



SyncBackSE v6

PDF User Guide

© 2011 2BrightSparks Pte Ltd.

Table of Contents

Foreword	1
Part I Welcome to SyncBackSE	4
1 Help using SyncBackSE	7
2 How to Buy SyncBackSE	12
3 Version Comparison	15
4 What's New in Version 6?	16
Part II Quick Start	20
1 Understanding Backup and Synchronization	22
Source and Destination	24
Backup	26
Synchronize	28
Mirror	30
Restore	31
Automating SyncBackSE	32
2 Licensing SyncBackSE	34
3 Profiles Backup for V5 Users	36
4 Importing V3 Profiles	37
5 Creating Your First Profile	40
6 First Run	48
Part III Using SyncBackSE	54
1 Basic Operation	54
The Main Window	55
Exporting and Importing Profiles	57
Creating a Profile	58
Running a Profile	67
Creating a Group Profile	70
Restoring a Backup	74
Creating a Schedule	80
Preferences Main Menu	89
Profiles Pop-Up Menu	93
Columns Main Menu	94
Global Settings	96
Dialogs	102
Comparison Programs	103
Log Settings	104
Shared Settings	106
Progress Bar	107
Accessibility	108
Exploring SyncBackSE	108
2 Easy Mode	110
Easy Mode Configuration	112

Searching the settings	114
Click For Options button	115
Simple	116
Sub-directories and Files	117
When	121
Fast Backup	122
Decisions - Files	131
Smart Synchronization	136
Notes	141
3 Expert Mode	142
Simple, Performance	147
Simple, History	147
When, Hot-key	148
When, Login/Logout	148
When, Changes	149
When, Insert	150
When, Periodically	151
When, Time Limit	151
When, Programs	152
Decisions, Folders	153
Copy/Delete	156
Copy/Delete, Folders	160
Copy/Delete, Advanced	161
Copy/Delete, Versioning	165
Compare Options	169
Compare Options, File Size	170
Compare Options, Date & Time	171
Compare Options, Attributes	172
Compression	173
Compression, Encryption	175
Compression, Advanced	176
Compression, NTFS	177
Compression, Compressed	177
Encryption	178
FTP	178
FTP, Advanced	180
FTP, Proxy	183
FTP, Firewall	184
Misc.	184
Misc., Media	185
Programs Before	186
Programs, After	188
Auto-close Settings	189
To Email	190
To Email, Advanced	193
Log	195
Log, Email Log	197
Log, Advanced	198
Network	200
Network, Advanced	201
CD/DVD Burner	201
CD/DVD Burner, Advanced	204
Cloud	205
Cloud, Advanced	207

Variables	209
Scripts	210
Backup Email	210
4 Runtime Help	212
The Differences Window	212
The File Collision Window	220
5 Technical Reference	224
Scripting	224
Base	227
Main Interface Scripts	229
Profile Configuration Scripts	233
Location Scripts	236
Runtime Scripts	249
SBLocation	265
SBProfile	268
SBRunning	272
SBSystem	291
SBVariables	303
SBHistory	308
Constants	313
Example Scripts	318
Command Line Parameters	320
Filter Settings	325
Open and Locked File Copying	329
Variables	330
Regular Expressions	350
Uninstalling SyncBackSE	358
SyncBack Management Service	358
Part IV Legal Information	364
1 Distribution	364
2 Privacy Statement	367
3 General Terms	369
4 Credits	377
Part V Glossary	380
Part VI System Requirements	400
Part VII Company Information	404
Index	407

SyncBackSE V6 PDF Help Manual

2BrightSparks Pte Ltd © 2003 - 2011. All Rights Reserved.

No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of 2BrightSparks Pte Ltd.

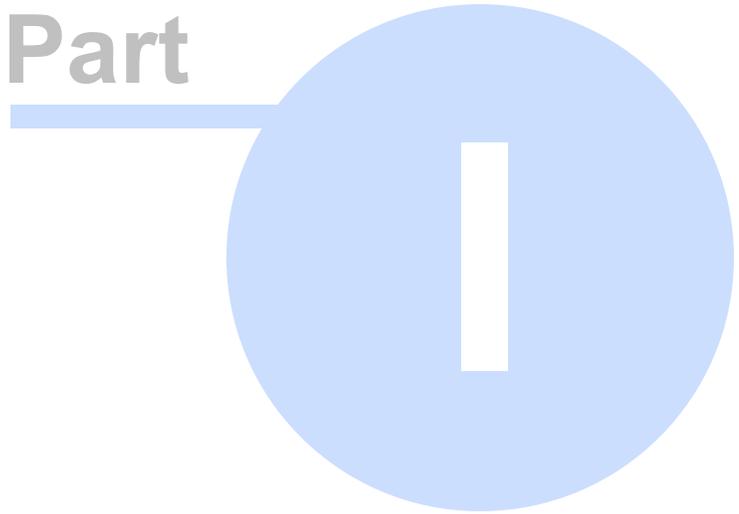
Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

2BrightSparks Pte Ltd. December 2011

Welcome to SyncBackSE

Part



1 Welcome to SyncBackSE



SyncBackSE V6

Welcome to the help and information guide of SyncBackSE V6.

Introduction

"You did have a backup... didn't you?..."

Three reasons why SyncBackSE is the right choice for you:

- 1: SyncBackSE helps you to protect yourself from data loss by allowing you to backup your important files. When disaster strikes all you have to do is click a single button to restore.
- 2: SyncBackSE helps you spend time on those things that matter. Trying to recover from the effects of lost data can be very costly and time consuming. SyncBackSE's proven backup and synchronization solution is a breeze to implement, reliable, and affordable.
- 3: SyncBackSE delivers great ease of use and outstanding functionality.

For most people, keeping a backup of their computer's data is not normally foremost in their minds, and is usually something they only get round to right after a disastrous data loss. We all assume that our files are going to sit there obediently unless we purposely delete them. Anyway, data can always be recovered or undeleted, right?

The harsh truth is that data is never secure unless a copy is kept in a *separate location*. If you accidentally delete a file, overwrite it, catch a destructive computer virus or suffer a catastrophic hard drive failure there's no easy way to get that data back without specialist software or expensive help.

The good news is that SyncBackSE makes backups easy. In its simplest form, this comprehensive program will enable you to keep copies of your data on another drive for safekeeping, be it on local or remote storage locations. Let's take a quick look at the many storage destinations you can use:



An internal drive (e.g. a second hard disk)



An externally-attached drive (e.g. USB, Firewire, or eSATA)



Removable media (e.g. a USB memory stick)



One or more writeable CDs, DVDs, or Blu-Rays

Of course, given SyncBackSE's comprehensive capabilities, it doesn't stop there. You can also keep safe copies of your data on the following destinations too:



A network drive (e.g. in a typical business environment)



A ZIP archive



An FTP or SFTP (PRO version) server



A user-defined location (using scripting) (PRO version)



An email server (PRO version)



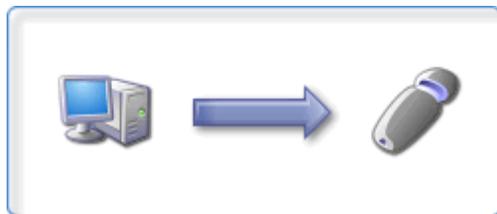
Amazon S3 or compatible server, e.g. Google Storage (PRO version)



Microsoft Azure server (PRO version)

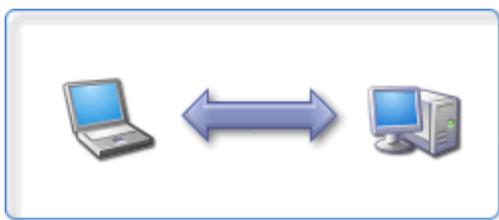
SyncBackPro can also make backups of your email (which is different from making backups of your files to an email server).

How does SyncBackSE achieve this?



At its simplest, backing up data is a one-way event, i.e. SyncBackSE will faithfully copy your live data on your PC to the storage destination. This is the simplest method of backing up and will suit most people for most situations. You may either perform the backup manually whenever you choose, or schedule a regular time slot for it to happen. In the latter's case, you never need to worry about forgetting to start the process.

Keep your data synchronized between two computers.



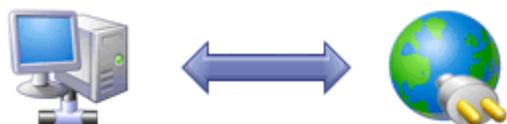
SyncBackSE can keep the same set of data up-to-date on both your PC and the storage destination. In short, any change made on one will be made on the other. For example, if you travel with a laptop and make changes to its data you may want to make sure that your desktop at home has the same set of changes. Conversely, someone at home may have made other changes to that data on the desktop too. SyncBackSE quickly achieves the consolidation of the two sets of data by only copying the changed data between the two PCs. This is known as Synchronization and is one of SyncBackSE's most versatile features.

Copy locked and/or open files



Another impressive trick is SyncBackSE's ability to copy and backup locked and/or open files. For example, if you are using Microsoft Outlook, your normally locked data file will be successfully copied even though it is in use. Other lesser-featured backup programs simply can't do this.

Need to keep backups offsite?



Best practice says that your backup should ideally be located in a different location to your original data. So how does SyncBackSE deal with offsite backups? Well, using a powerful FTP engine, SyncBackSE enables you to backup data to a remote server on the Internet. You could also use FTP to backup to a local NAS storage device or an FTP server somewhere else on a company network. SyncBackSE can also backup your files to an email server or to Amazon's S3 cloud storage service.

Centralized Management

Distributed installations of SyncBackPro can be managed and monitored centrally using the SyncBack Management Service (SBM Service). This allows for easy management of profiles and to check on the status of backups on those remote installations.

Other features

Among SyncBackSE's array of impressive features are the Fast Backup option, Smart Synchronization, file versioning, fast compression with AES encryption, configuration via scripting (🏠 Pro version), BZip2 and LZMA compression (🏠 Pro version), backup to CD or DVD with automatic disk spanning (🏠 Pro version), a database engine for the backup/sync of an unlimited number of files (🏠 Pro version), and a number of crucial advantages over its freeware namesake.

SyncBack Version Comparison

For a comparison of **SyncBackPro**, **SyncBackSE**, and **SyncBack freeware** go to Version Comparison.

For a list of what is only available in the 🏠 Pro version go to Pro Only Features.

Quick Links

To get up to speed quickly, we recommend taking a look at the following sections:

[New User Guide](#)

[Quick Start](#)

[Using SyncBackSE](#)

[Online Tutorials](#)

Thank you for choosing SyncBackSE. We are confident you will find it to be both powerful and a pleasure to use!

1.1 Help using SyncBackSE

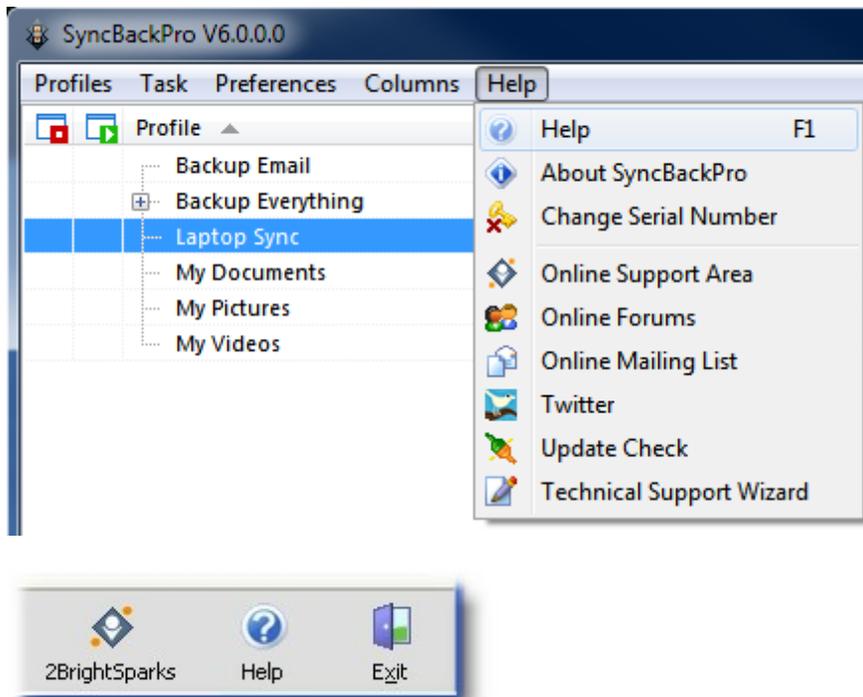
Help! Online and Printable Support Resources

As a user of SyncBackSE you can enjoy extensive support from 2BrightSparks Pte Ltd:

Your first port of call for support is to read this help file.

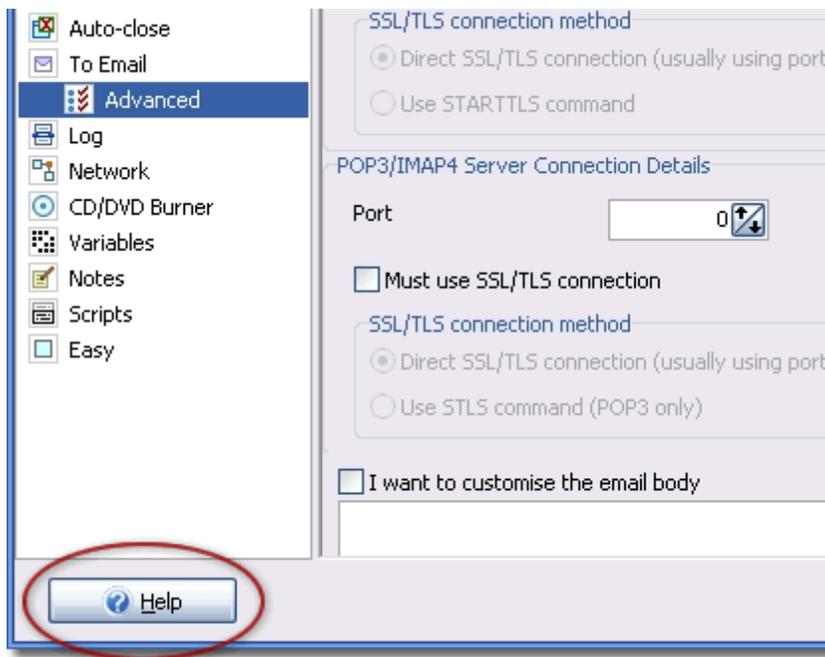
Top Tip! Simply click the F1 key (usually on the upper left of your keyboard) to open the help file at the particular page relating to the current SyncBackSE window.

Alternatively, click **Help** under the help menu, or the Help button on the main program window to open the help file:



If you have never backed up data before, we advise you read the Quick Start Guide which explains the fundamentals about different kinds of backup. Using SyncBackSE provides detailed explanations of all the functions in SyncBackSE including Basic Operation; Easy Mode; Expert Mode; Runtime Help; and a Technical Reference.

This help file also offers context sensitive guidance when using the program. You'll find a **Help** button in most of the program's windows. Clicking the help button will take you straight to the help page for that particular task or option. The following help button in the program for example will take you to the section of this help file called Email, Advanced:



Fully Searchable Settings

When editing the settings for a profile you can use the settings search feature to locate the settings you need.

Fully Searchable and Printable Help Manual

Do not directly print this help file from the Microsoft help viewer because the print quality of the HTML Help viewer is poor.

A fully searchable and printable SyncBackSE Help File is available as an Adobe PDF file (Portable Document Format).

This help file is available from our website in PDF format at:

<http://www.2brightsparks.com/syncback/SyncBackSE-PDF.zip>

Although Adobe Acrobat Reader is installed as standard on Windows computers, you may wish to download and install the very latest version which includes significant improvements over its predecessors:



[Get latest Adobe Acrobat Reader](#)

Online Support from 2BrightSparks

Our online support is among the best in the industry. You'll find [Tutorials](#) to get you up and running and our extensive [Support Area](#) which features our [Knowledge Base and FAQs](#) (Frequently Asked Questions).

Technical Support

You **must** use our [Support Ticketing](#) service for email support which is available at:

<http://www.2brightsparks.com/help/index.php?pf=usr>

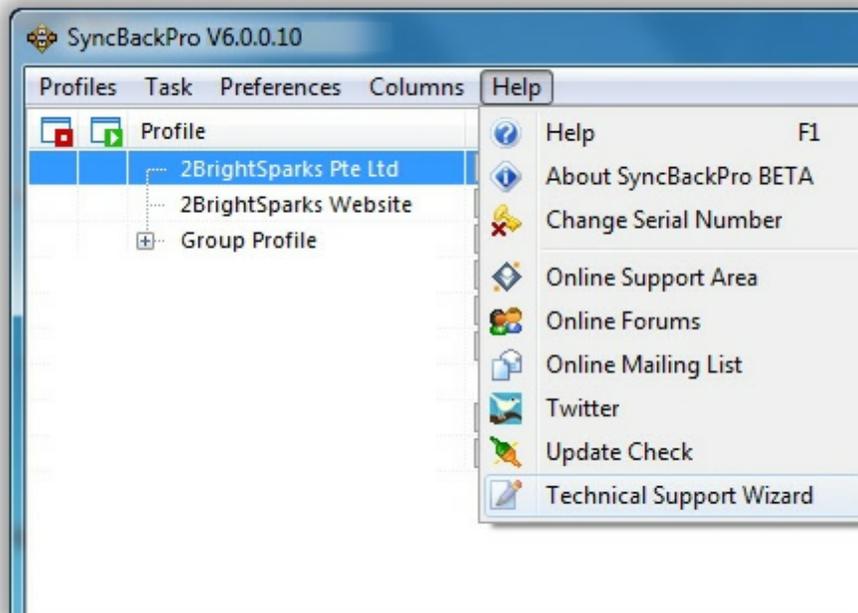
Support Ticketing ensures we keep the appropriate records of your support request and our response history to you. This helps us deliver better service to all our customers as we respond efficiently, audit the effectiveness of our responses and generally improve the quality of our support. In addition, analysis of our support performance and the frequency level of specific support requests also leads to improving our software. By using a support system that does not require email, the problems associated with spam filters are also avoided.

We trust you understand the importance we place on the high quality of our software and service, and therefore we must follow procedures that are of greatest benefit to the majority of our users. Direct email support requests will be ignored.

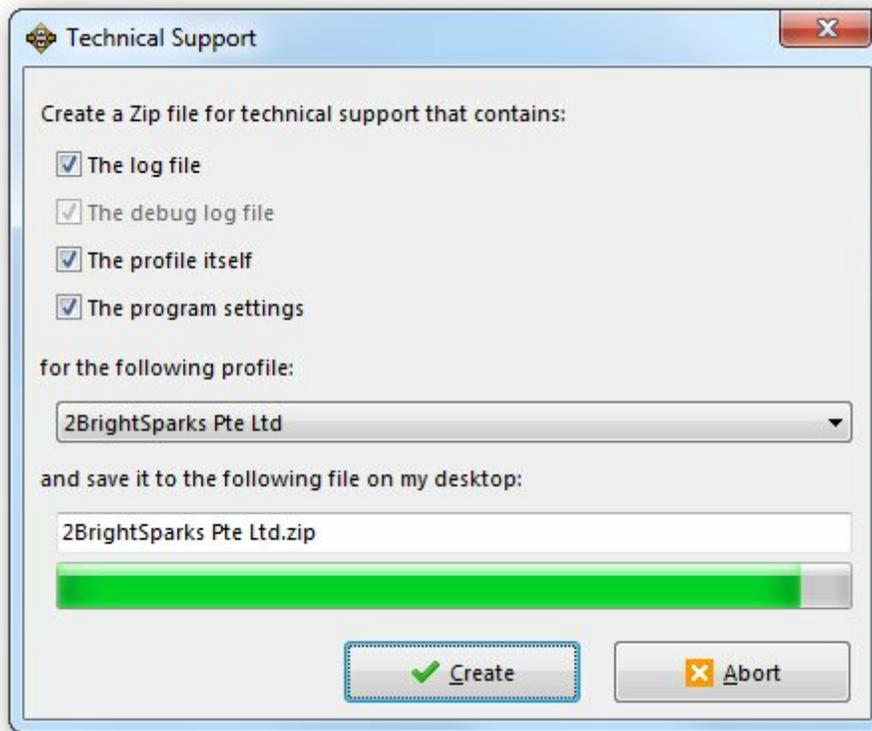
The Technical Support Wizard

If after reviewing the help file and online support resources you continue to experience difficulties using SyncBackSE, you may submit a support ticket. The **Technical Support Wizard** creates a special file that provides the essential information we require when you submit your ticket.

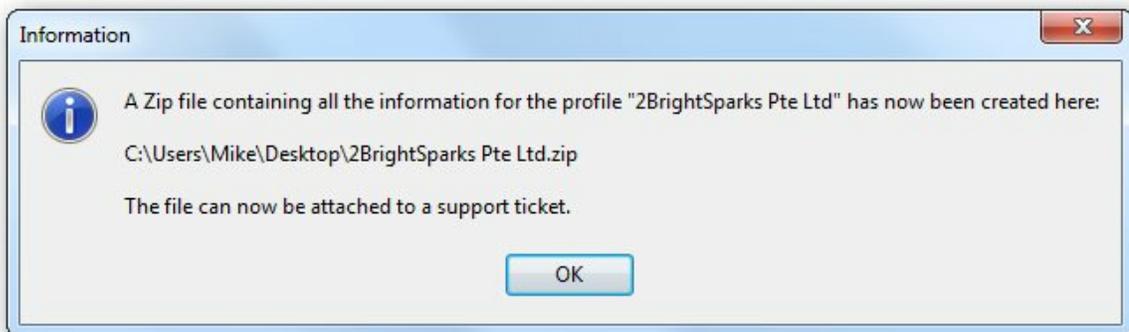
You can open the Technical Support Wizard by clicking on the **Help** menu and choosing the **Technical Support Wizard** item from the drop down list:



You may now choose items to include, select the profile, then click **Create**:



An informational window will open showing where the support zip file has been saved to:



The Technical Support Wizard will create a special zip file that contains the settings, logs, debug information, etc. that relates to the selected profile. The Support Zip File created by the wizard may then be attached to a Support Ticket.

Note that the Zip file produced is in a special format that is not readable by Windows Explorer or many other Zip programs. You should not attempt to modify the Zip file or its contents otherwise it may become corrupted.

Debug Logs

Debug logs are text files that are stored in the same folder as your profile settings. The

folder in which profiles are stored is defined using Global Settings.

1.2 How to Buy SyncBackSE

Purchase SyncBackSE Today and Enjoy Instant Delivery

We recognize that security is one of the major concerns for the shopper during an online transaction. Our payment processor uses state of the art security tools and techniques to ensure that you are protected against online fraud.

All payment transactions at 2BrightSparks are handled through the eSellerate payment system. At no time do we process or save customer payment card details on our website.

If you would like to learn more about eSellerate, visit their site at:

http://www.esellerate.net/company_overview.asp

Direct Ordering and Download

Pay via Credit Card, by Phone or FAX, or via PayPal...

Ordering SyncBackSE couldn't be simpler. Your serial number will be presented to you immediately following payment on eSellerate's secure server. Your order details and serial number will also be sent to you via email.

SyncBackSE, and other 2BrightSparks software, are only available from our website as downloadable products. Orders can be processed for VISA, Mastercard, DISCOVER, American Express, JCB, and via PayPal



Buy Now

Click the following link to order directly from our Web Store:

[Buy SyncBackSE](#)



Upgrade SyncBackSE to SyncBackPro

All SyncBackSE licensees can upgrade to SyncBackPro at reduced prices. [Visit Our Web Store](#) and choose your upgrade option today.

Phone and Fax Orders

Alternatively, you may use the following URLs for phone or Fax orders:

 <http://store.eSellerate.net/s.asp?s=STR2236179640&CMD=PHONE>
 <http://store.eSellerate.net/s.asp?s=STR2236179640&CMD=FAX>

Website

A fully functional evaluation version of SyncBackSE is also available from our website by clicking any of the 'Try' links:

Download the full evaluation of SyncBackSE today:

[Download SyncBackSE 30 Day Trial](#)

Sales and Support

Sales and Support are available by submitting a support ticket from our [Support Area](#). Find out more about our industry leading support resources.

Other Great Software from 2BrightSparks

InfoHesiveEP



A powerful program to organize your data files and share information. For Microsoft Windows XP/2003/2008/Vista/7. [Find out more...](#)

- InfoHesive Projects link to any data file on your computer.
- Write notes associated with any file or URL.
- Use the powerful on-board rich-text editor and save as an eBook or Help File.
- Share your InfoHesive WorkSpaces using the the Freeware InfoHesive Viewer.

Create InfoHesive WorkSpaces for your own needs: Support & Marketing; Personal Information Manager; Home Inventory; Boilerplate Documentation; Brain Storming; Photo & Music Album; Study or Research Aid; Recipe Book; Project Management; e-Authoring; help file dev. The list is endless...

Download the fully functional 30 day trial of InfoHesive

OnClick Utilities



A suite of easy to use software utilities for Microsoft Windows XP/2003/2008/Vista/7. [Find out more...](#)

- **FindOnClick** performs lightning fast file searches.

- **UndeleteOnClick** recovers deleted files.
- **DeleteOnClick** securely deletes data. A free version is available.
- **HashOnClick** helps guarantee files are identical. A free version is available.
- **EncryptOnClick** delivers military grade encryption. Freeware.
- **ScrambleOnClick** encrypts/decrypts files and text.
- **PatchOnClick** easily updates large files. A free version is available.

Select one or all of the OnClick Utilities programs for your needs:

Download OnClick Utilities

Company Postal Address

2BrightSparks Pte Ltd
PSA Building
PO Box 364
911143
Singapore

Read Company Information for more.

1.3 Version Comparison

Version Comparison

Below is a list of some of the major features and in which version of SyncBack they appear:

Feature	SyncBack Pro	SyncBackSE	SyncBack Freeware
Backup/sync via Amazon S3 and Microsoft Azure	Yes	No	No
Backup of emails	Yes	No	No
SyncBack Management Service (SBM Service)	Yes	No	No
SysLog integration	Yes	No	No
Backup/sync via an email server	Yes	No	No
Backup directly to CD/DVD	Yes	No	No
Scripting	Yes	No, requires 3rd party software	No, requires 3rd party software
Unlimited number of files	Yes	No, limited by RAM	No, limited by RAM
Drive failure detection using S.M.A.R.T.	Yes	No	No
BZip2 and LZMA compression	Yes	No	No
SFTP	Yes	No	No
ExtremeCopy	Yes	No	No
Detect and correct filename case differences	Yes	Yes	No
Unlimited filename lengths (except with Zip)	Yes	Yes	No
Copy open/locked files	Yes	Yes	No
File versioning	Yes	Yes	No
Incremental & differential backups	Yes	Yes	No
Fast Backup feature to reduce backup time	Yes	Yes	No
Smart Synchronization	Yes	Yes	No
Fully Unicode for multi-language support	Yes	Yes	No
Zip64 compression with 256-AES encryption	Yes	Yes	No
BWT compression	Yes	Yes	No
Create self extracting Zip files	Yes	Yes	No
Extended FTP support, with SSL/TLS support	Yes	Yes	No
Fine grained control over what to backup/sync	Yes	Yes	No
Regular expression filters	Yes	Yes	No
Bandwidth throttling	Yes	Yes	No
Emailing logs	Yes	Yes	No
Extensive variables	Yes	Yes	No
Automatic backup when files changed	Yes	Yes	No
Control over what actions to take	Yes	Yes	No
Move files	Yes	Yes	No
Detect renamed files	Yes	Yes	No
Enable & disable profiles	Yes	Yes	No

Run profiles on login or logout	Yes	Yes	No
Run profiles whenever media is connected	Yes	Yes	No
Run profiles whenever a program is started or stopped	Yes	Yes	No
Free technical support	Yes	Yes	No



Upgrade SyncBackSE to SyncBackPro

All SyncBackSE licensees can upgrade to SyncBackPro at reduced prices. [Visit Our Web Store](#) and choose your upgrade option today.

1.4 What's New in Version 6?

Great Features in SyncBackSE

The latest version of SyncBackSE offers significant advantages over previous versions, and is a free upgrade for all registered users of SyncBackSE.

New Features in V6

Listed below are just some of the new features included in this version. All these new features are covered in greater detail in this help file and in our [Online Support Area](#).

- **Start profile when a profile is started or stopped:** A profile can now be started automatically whenever a program is started and/or stopped.
- **Variables:** New variables have been added.
- **Case changes:** SyncBackSE can now detect changes in filename case and update accordingly.
- **Drive failure prediction:** Drive failure prediction has been improved so that drive failure can be predicted in more drives and external drives ( Pro version only).
- **Windows 7:** User interface changes to make use of the new user interface features in Windows 7.
- **Revised and extended Help File**

Features Only Available in SyncBackPro

The following features are only available in the  Pro version:

- **Backup and sync via the Amazon S3 and Microsoft Azure cloud storage services:** You can backup and synchronize your files via cloud storage services. 🏠 Pro version only.
- **Backup email messages:** You can backup your email messages from your email server. 🏠 Pro version only.
- **SyncBack Management Service (SBM Service):** Remote installations of SyncBackPro can now be managed and monitored from a central location. 🏠 Pro version only.
- **SysLog Integration:** SyncBackPro can be configured to communicate with your SysLog server so you can monitor remote SyncBackPro installations. 🏠 Pro version only.
- **Backup and sync via an email server:** You can backup and synchronize your files via an email server (POP3/IMAP4 and SMTP), e.g. Gmail, Yahoo!, etc. For example, you could synchronize your files with a friend or colleague via a dedicated shared email account. You could also sign up to a free email account and backup your files to it. There is no need to worry about spam - SyncBackPro can be configured to automatically delete all emails that are not used by SyncBackPro. There is also no need to worry about security because you can store your files using 256-bit AES encryption. 🏠 Pro version only.
- **Backup directly to a CD/DVD:** You can backup directly to CD or DVD (including **Blu-Ray**), with automatic disk spanning. There is no need to pre-format the CD/DVD, and compression does not need to be used for disk spanning. 🏠 Pro version only.
- **Scripting:** This powerful feature allows you to configure how SyncBackPro works and runs profiles. For example, you could create a script that lets SyncBackPro backup to a database (or anything else you can access). Scripting can also be used to change how profiles run. A number of new scripting functions have been added to version 6. Scripts can now also add their own profile configuration page for easy integration and configuration. 🏠 Pro version only.
- **Unlimited number of files:** SyncBackPro uses a database to store details of the files it is copying, instead of storing the information in RAM (memory). This means an unlimited number of files and folders can now be processed. SyncBackPro will use RAM for performance but will automatically and seamlessly switch to a database if memory is running low or the number of files reaches a threshold. Using a database will actually be significantly faster when the number of files is in the hundreds of thousands. 🏠 Pro version only.
- **Automatic drive failure detection:** Using S.M.A.R.T. technology, SyncBackPro can detect the impending failure of a hard drive. If a hard drive is going to fail (or has failed) then the log file will have details, and the profile status will indicate it. To use this feature your computers BIOS and hard drive must support S.M.A.R.T. (and it must also be enabled via the BIOS settings). S.M.A.R.T. failure prediction was improved in V6. 🏠 Pro

version only.

- **BZip2 and LZMA compression:** As well as supporting the industry standard Zip compression, and BWT compression, you can now also compress using BZip2 and LZMA. These newer compression methods compress most files more effectively than more traditional gzip or Zip, but are slower. If space is more of a concern than speed, then BZip2 or LZMA may be the solution. The BZip2 compression method is compatible with WinZip 11.0 and newer compression utilities. The LZMA compression method is compatible with WinZip 12.0 and newer compression utilities.  Pro version only.
- **SFTP:** As well as supporting traditional FTP and FTPS, SyncBackPro now also supports SFTP.  Pro version only.
- **ExtremeCopy:** Files over 100MBytes can be copied faster using the ExtremeCopy method which will reduce profile run time.  Pro version only.



Upgrade SyncBackSE to SyncBackPro

All SyncBackSE licensees can upgrade to SyncBackPro at reduced prices. **Visit Our Web Store** and choose your upgrade option today.

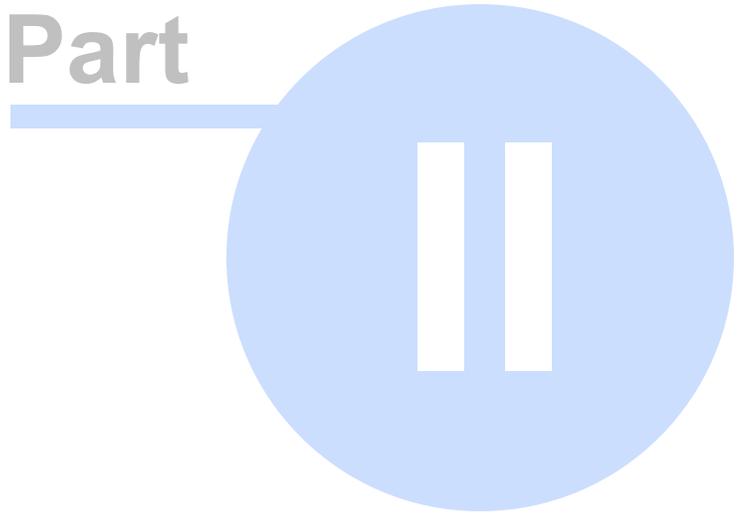
History of changes

For a complete history of changes which details all updates, fixes, and features, go to:

<http://www.2brightsparks.com/syncback/sbse-changes.html>

Quick Start

Part



2 Quick Start



Start Using SyncBackSE Straight Away

 If you have not used a backup program before, we strongly advise you spend time reading Understanding Backup and Synchronization which provides an essential introduction to the field.

 The 'take note' icon on the left is used throughout this help file and indicates advise that is of particular importance.

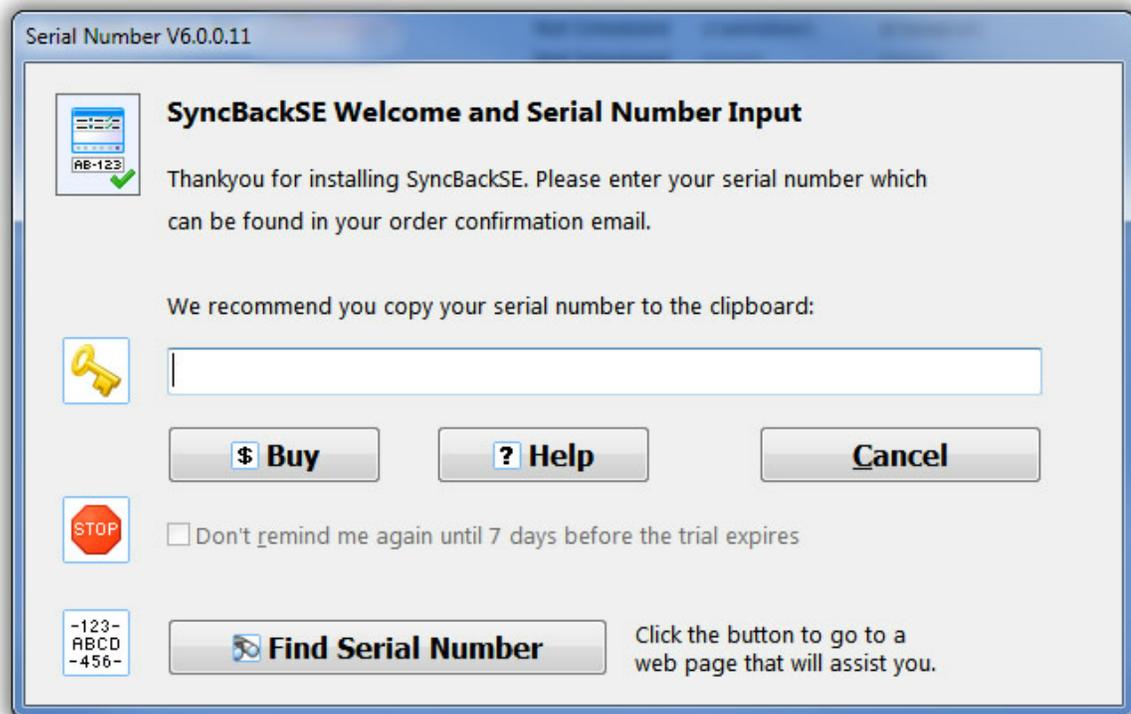
Three stage start

SyncBackSE can be up and running by following the following three stage process:

Step 1. When you first install SyncBackSE, the program will ask if you would like to:

- Evaluate the software
- Enter your Serial Number that you would have received by email after you have paid for SyncBackSE

Click the **Buy** button to visit our web store so you may pay for a license, or the **Evaluate** button to use the 30 day trial:



 When in evaluation mode, the user can opt not to be reminded again until 7 days before the trial expires by clicking the box.

Step 2. The program will then ask if you would like to import any existing SyncBack freeware profiles. Please remember to ensure you browse to the correct folder if you chose to install a previous version of SyncBack freeware in a customized location. If you wish to import version 3 profiles, read our special guide .

Step 3. If you have not created any profiles in the past click the "New" button located on the lower left of the program window:



Creating your first Profile is a simple process with the help of the profile wizard. Once a profile has been created you need only click a single button to run that profile.

Need Help?

Read the Help Using SyncBackSE section of this help file which details the many ways you will find guidance and support when using the program.

 Use the context sensitive help buttons that provide assistance by clicking the 'Help' button that you'll find on each program window. This will take you straight to the

appropriate section in the help file.

SyncBackSE Evaluation Version

You have 30 days to evaluate SyncBackSE after which time you must uninstall it from your computer. You may buy SyncBackSE at any stage by visiting our web store.

[Click to visit our Web Store](#)

2.1 Understanding Backup and Synchronization

What SyncBackSE Does

SyncBackSE is designed to help you prevent data loss. This section of the help file aims to provide you with essential knowledge about what the program does to achieve this. There are four core tasks SyncBackSE does:

- Backup
- Synchronize
- Mirror
- Restore

SyncBackSE allows the user to make many, many decisions about how these core tasks are achieved.

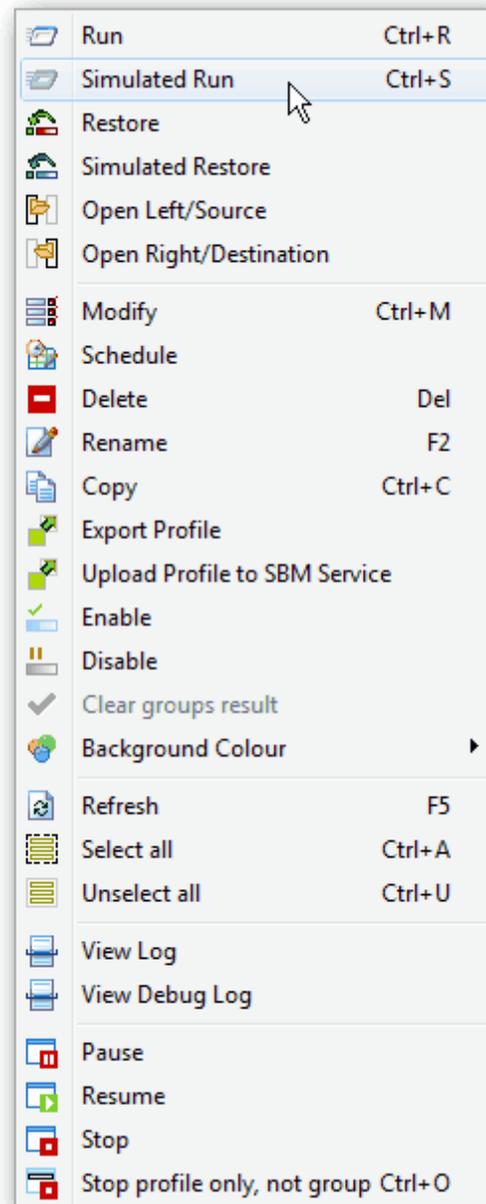
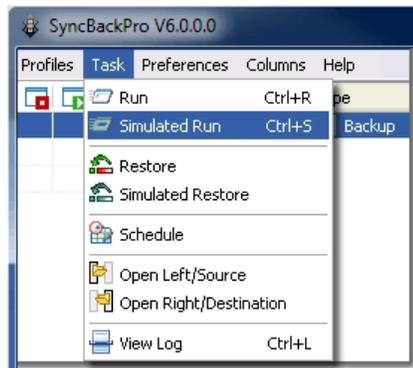


Important: SyncBackSE copies, moves, and deletes data. Always ensure you test your settings, ideally with test files, before using them with your actual files. We try to make it very clear during the installation process that SyncBackSE is designed to be able to delete and replace files, so it must be used with caution. We also ensure the default options are set to a safe mode.

Good data processing procedure dictates that any program should be thoroughly tested with non-critical data before relying on it. SyncBackSE also features a 'Simulated Run' feature so that users can check to ensure the program is processing data in the way they expect before making an actual run.

The Simulated Run or Simulated Restore commands are available from the 'Task' menu or by right clicking on a profile:

Task Menu: Simulated Run Right Click Menu: Simulated Run



What SyncBackSE can and cannot do

SyncBackSE can copy all of your files, but cannot make an exact copy of your system drive. SyncBackSE is designed to copy your important files, e. g. your pictures, documents, music, database, etc. You can always re-install the operating system and any programs, but you cannot recover your files unless you've made a backup. With SyncBackSE, making a backup of those files is quick and easy.

To make an exact copy of your system drive, you must use "disk imaging" software. With disk imaging, the entire disk (the parts that are used) is copied bit-by-bit. This means the

copy will take up a lot of disk space and take much longer to copy.

Your Windows operating environment is constantly changing. Programs are installed, updated, uninstalled and settings are changed. Many important security specific applications are also regularly and automatically updated. Anyone who uses their computer to connect to the Internet should have in place Anti-Virus, Firewall and Anti-Spyware programs that are all readily available from within the Windows operating system. These issues, combined with the longer, costlier (more disks) and less convenient (more time consuming) disk imaging process inevitably means that for the average user, disk imaging is carried out far less frequently than the kind of backup SyncBackSE delivers.

SyncBackSE copies all your important files in a fast, up to date and reliable manner.

2.1.1 Source and Destination

Understanding the 'Source' and 'Destination'

SyncBackSE copies, moves and deletes digital files from one location to another. This helps you in your aim to prevent data loss.

To make the process of backing up simple to understand, we use the terms **Source** and **Destination**. The **Source** is a particular location and the **Destination** is a different location. In the example below, the Source is a workstation and the Destination is an external drive:



You will create a Profile of both the Source and Destination, and define the Profile's actions (backup, synchronization, restore etc.).

You can also group profiles together so multiple actions occur, and then schedule these profiles to run automatically.

Backup

In the case of a simple backup operation, the Source is the place where files are copied **from** and the Destination is the place where files are copied **to**. For example, the Source could be the folder 'Your Computer Drive\My Documents\My Business Folder\' and the Destination could be a folder on an external USB drive 'My External Backup Drive\My Backup\My Business Folder\'.

SyncBackSE allows you to make choices about exactly what files are to be copied, moved,

ignored or deleted during the backup, synchronization or restore process. You may decide, for example, to ignore certain files or folders when backing up. Therefore, it is not necessary to backup every file from the Source to the Destination.

When SyncBackSE first runs a backup, the program will copy all the files you require from the Source to the Destination. The next time you run the same backup task, SyncBackSE does not copy the unchanged files, but rather scans both the Source and Destination, notes what files have been changed and then asks you to confirm the action the program is about to take in the Differences Window. This makes subsequent backups a lot faster than the initial backup.

Synchronization

When it comes to Synchronization, viewing the source as a location where files are copied from is not accurate. Think of the source simply as a location, rather than as a location that always has a fixed task associated with it. The source can be thought of as the **left** side, and the destination as the **right** side.

In a synchronization operation for example, a file may be copied to the source rather than from the source, and at the same time another file may be copied to the destination. This may occur as you may have chosen options in SyncBackSE that request certain actions occur given certain criteria. For example, you may require that a file that is older on either the source or destination, must be replaced by the newer file (given they have the same name and file type).

As you can tell by the example above, synchronization is a more complex process than a backup process.

Mirror

Mirroring is used when you want the destination to be an exact copy of the source. It assumes that the destination is not changed by other programs or users, so it is not the same as a synchronization. Files are copied from the source to the destination, and not vice-versa. It is not the same as a backup because any files that are only on the destination are deleted. This means there are no 'orphaned' files in the destination.

Good data procedures

When you first use SyncBackSE, we advise you use the default options and simply backup, as this will prevent any unexpected and potentially unwanted actions to occur. As you become familiar with the program, you will begin to gain a deeper understanding of the Backup and Synchronization processes.

We have tried very hard to put as many checks and warnings in place in order to prevent you from accidentally losing data. We make it very clear during the installation process that SyncBackSE is designed to be able to delete and replace files, so it must be used with caution. We also ensure the default options are set to safe settings that will reduce the possibility of unwanted data loss to a minimum.

Good data processing procedures dictate that any program is thoroughly tested with non-

critical data before relying on it. SyncBackSE also features a **Simulated Run** feature that you will be able to ensure the program is processing data in the way you expect before making an actual run. A Simulated Run does not move, copy, or delete any files, but will show the differences window for you to fully review what will occur in the case of an actual run.

2.1.2 Backup

An Explanation of the Backup Process

A backup copies files in one direction: from the source to the destination.

Note that a backup is not a synchronize process.

SyncBackSE can backup to the same drive; a different drive or medium (CDRW, CompactFlash, etc); an FTP server; a Network; or a Zip archive.

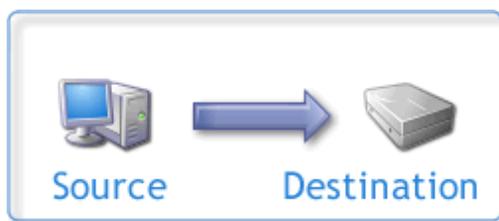
Backing Up

The examples on this page show different scenarios of when you might backup.



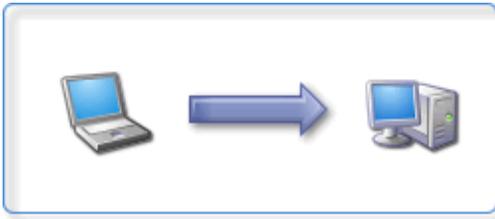
Note that the Destination needs to have enough free disk space to take all the backup data.

Here's an example of a local computer backing up to an external hard disk. The computer's drive is the Source, and the external hard disk is the Destination. Files will be copied from the source to the destination:



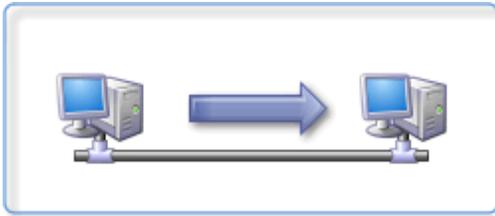
Usage: Backup files on your local C drive onto an external USB drive. This ensures you've got an accessible copy of your data, even if you experience problems with your main computer.

A second example shows a backup where the a laptop is the Source, and a workstation is the Destination:



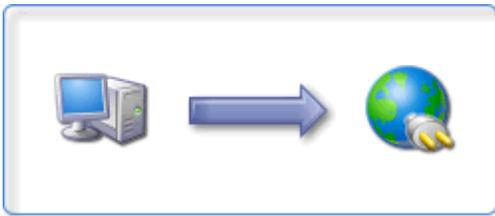
Usage: Perhaps you've been on a business trip or on holiday and you've continued to update your documents. When you get back home you'll want to copy any new or changed files back onto your computer. A simple backup will achieve this.

The third example shows how a backup can run from one network computer to another:



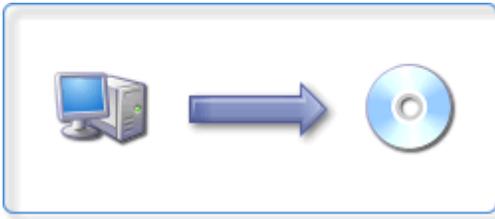
Usage: You've setup a home network, and you use one of the computers on that network to work on, and another to backup to.

The example below shows a backup taking place from a local workstation to another remote computer using FTP (File Transfer Protocol):



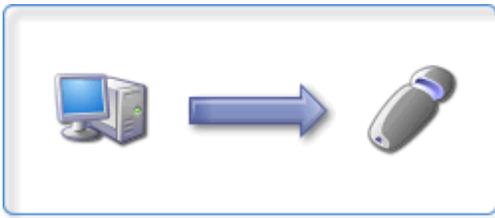
Usage: You're on the move and work with a laptop. As you've got Internet access and FTP access to a website to upload files, you backup a copy of your data so you know if disaster strikes and your laptop is lost, breaks down, or is damaged in transit, your data is safe and easily retrievable.

The next example shows a backup to CD-R/DVD-R etc:



Usage: You want to backup all your family photos and movies to CD-R or DVD for archival purposes.

The final example shows a backup between a local computer and a memory stick:



Usage: You're at work and need to copy the documents that are located in different folders on your work computer to a memory stick. Once you've defined your profiles and grouped them using SyncBackSE, you'll click one button, and everything will be copied quickly and simply in a single action.

2.1.3 Synchronize

An Explanation of the Synchronize Process

A Synchronisation copies files in both directions: from the source to the destination, and from the destination to the source.



Note that the Destination and Source need enough free disk space to exchange the data.

SyncBackSE V4 introduced Smart Synchronization which copies files in both directions, but it also keeps a history of where files were during the last synchronisation. This allows for much finer control over what actions to take based on what has changed, and also allows it to detect changes such as the file only being modified in the source or destination.

It is advisable to use Smart Synchronization instead of the old style synchronization. By default any synchronization profile created using SyncBackSE will be a Smart Synchronization profile. However, a profile imported from older versions of SyncBackSE (or SyncBack freeware) may not be.

Note that the synchronization process is not the same as a backup process.

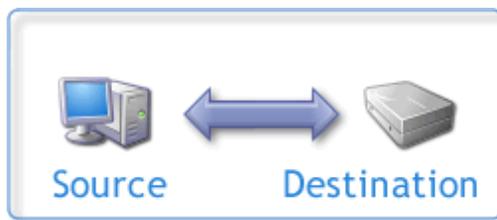
What is the difference between Synchronization and Smart Synchronization?

Smart Synchronization keeps track of what changed the last time the profile was run so that it knows if a file has been deleted, created, or changed since the last profile run. This helps you (and SyncBackSE) make an informed decision about what to do when something changes. It also gives you a comprehensive choice of options on what to do with a file when specific things happen, e.g. the file is deleted for the source but not the destination. Old style synchronization does not keep track of changes and has a limited set of options. Whenever possible you should use Smart Synchronization instead of the old style synchronization.

For example: you are synchronizing files between your laptop and desktop. You change a file on your laptop but delete that same file on your desktop. When you next run your Smart Synchronization profile SyncBackSE will be able to detect this and perform the configured action (in this case the default action is to prompt the user). However, if you had used an old style synchronization profile it would only have detected that the desktop file had deleted and would not have detected that the laptop file had also been modified (in this case the default action would have been to copy the file from the laptop back to the desktop).

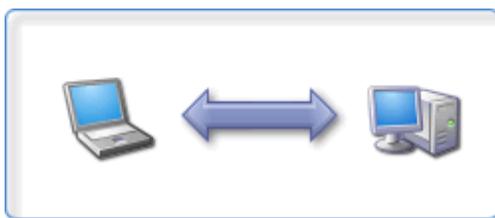
Synchronizing the Source and the Destination

Here's an example of a computer drive being synchronized with an external hard disk. Files will be synchronized between the source and the destination:



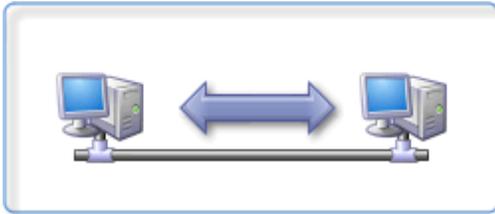
Usage: You use two separate drives, one for business, another for home use. Some files are on both drives, like your diary. Synchronization ensures that whatever drive you work on, the other drive is updated with your new diary items.

A second example shows a laptop (Source) and workstation (Destination) being synchronized. The workstation could be two different computers, e.g. your office computer and your home computer:



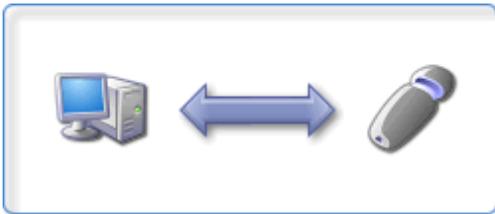
Usage: You work on the move and at home and want to ensure both your laptop and workstation have the same up to date files. You achieve this by Synchronizing your two computers.

The third example shows a synchronization running from one network computer to another:



Usage: You work on a networked computer and often change files that others will also view and change during the course of a day. Synchronization helps to ensure that whoever is working on the file does so with the most up to date version.

The final example shows a local computer and a memory stick being synchronized:



Usage: You've updated many documents over the course of the day. At one point you used a memory stick to load some documents as you worked on a computer other than your usual one, but you can't remember the exact name of those documents. When you return to your main computer you run your synchronization profile. SyncBackSE copies files in both directions, ensuring that both the memory stick and your main computer have the same up to date documents.

2.1.4 Mirror

An Explanation of the Mirror Process

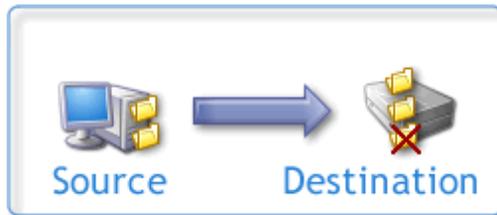
Mirroring ensures that one drive (or folder) contains the same files as another drive (or folder). It is not the same as a backup because it deletes files. It is also not the same as synchronization because it only copies files in one direction.



Note that the Destination needs to have enough free disk space to take all the data.

Mirroring

Here's an example of a local computer mirroring to an external hard disk. The computer's drive is the Source, and the external hard disk is the Destination. Files will be copied from the source to the destination and any files that are only on the Destination are deleted:



Usage: Mirror files from your local C drive to an external USB drive. This ensures you've got an identical copy of your data, even if you experience problems with your main computer. It is different from a backup because there are no 'orphaned' files in the destination.

2.1.5 Restore

An Explanation of the Restore Process

Being able to easily restore data is a core function of SyncBackSE. Here are some example scenarios of when you'll want to run a restore operation:

- You inadvertently delete a file and/or folder and need to recover it easily and quickly.
- As you're working on a document, the parent program unexpectedly crashes and your work is gone. If you've setup a backup to run in the background, you'll be able to restore to the last backup point.
- Your computer suffers a catastrophic failure, and you cannot access any files or folders. Having backed up onto an external drive or disk, you will be able to easily restore your valuable files onto a new computer.

A restore operation swaps the source and destination directories: i.e. the source directory becomes the destination directory and vice-versa:



Usage: You have inadvertently deleted the wrong files. You quickly restore your files from an external hard drive back onto your main computer, then continue to work with little impact

on your time or overall stress levels.

Usage: Your main computer suffers a serious security attack and your document files have been badly affected. After reinstalling Windows you are able to also restore your valuable document files to the state they were prior to the attack.



Note that a restore may not work in the way you expect it to: e.g. some of the files in the destination directory may be older than their equivalent entries in the source, and therefore may not replace the source entries (depending upon what your settings are).

If in doubt use the **Simulated Restore** feature in SyncBackSE. A simulated restore will show you what will happen to your files, without actually copying, moving, or deleting any files.

Running a restore operation is not reversible.

2.1.6 Automating SyncBackSE

Scheduling and Background Tasks

SyncBackSE provides a number of ways to run a profile (see below for a complete list) however, there are two methods to run a profile based on the date and time:

- Scheduling with the Windows Task Scheduler.
- Have a profile configured to run in the background.

A profile can be both scheduled and set to run in the background.

Scheduled Tasks



The Windows operating system comes with an integrated scheduler (the Windows Task Scheduler).

This scheduler lets the user configure Windows so that certain programs are run at certain times, e.g. every day at 9am. For example, most anti-virus software will prompt you to create a scheduled task to scan your computer for viruses every morning. There are a number of advantages to scheduling profiles:

- A scheduled profile can be configured to run even when you're not logged in or if someone else is logged in.
- A scheduled profile can switch the computer on (from hibernation or standby) to run the profile.
- The date & time of when the profile is run, and how often it is repeated (e.g. daily), is highly configurable.

- You do not need to have SyncBackSE running to have a profile run at the scheduled times.

This help file has a special section that shows you how to Create a Schedule.

Background Tasks



SyncBackSE has the ability to run profiles at periodic intervals, e.g. every 2 hours.

This is different from scheduling a profile because it is not based on a specific date & time but instead the frequency. There are a number of disadvantages to having a profile run in the background:

- You must be logged in for the profile to run.
- SyncBackSE must be running for the profile to run.
- The profile can only be configured to run every x seconds/minutes/hours and not at a specific date or time.

If profiles are configured to run in the background then it's best to configure SyncBackSE so that it starts automatically when you login to Windows.

This help file has a special section that shows you how to Create a Background Task.

Which method should be used?

In general it is better to schedule a profile instead of having it run in the background. However, if you want a profile to run frequently (e.g. hourly) then it is advisable to both schedule the profile and have it set to run in the background. This gives you the best of both worlds: your backup will be performed even when you are not logged in, and your backups will be performed frequently while you are working so that if you need to restore a file the backup copy will be more recent.

Situations under which a SyncBackSE Profile can be run

The list below shows all the situations and configurations in which a SyncBackSE profile can be run:

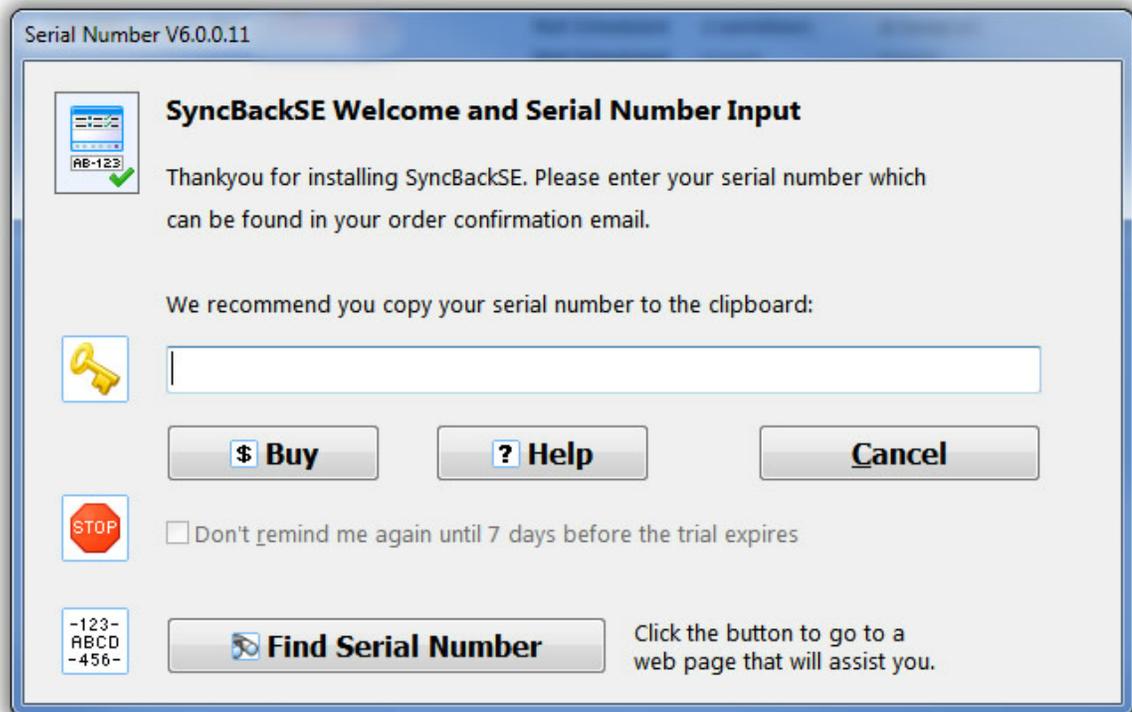
- Manually run, e.g. profile selected in main window and the **Run** or **Restore** buttons are pressed
- Scheduled (note that schedules can also be set to run every x minutes, hours, etc. and not just once or daily, weekly, monthly etc.)
- Run when a **hot-key** is pressed (see When -> Hot-key)
- Run on Windows startup or shutdown/logoff (see When -> Login/Logout)
- When files or folders in the source or destination are changed (see When -> Changes)
- Run when a device is attached/inserted, e.g. a CD, a USB flash key, etc. (see the When -> Insert)

- Set to run in the background (see When -> Periodically)
- Run when an external programs starts (see When -> Programs)
- Run externally from the command line, a batch file, or another program
- Run as part of a **Group Profile** which is run by one of the above methods

2.2 Licensing SyncBackSE

Program Launch

When SyncBackSE is first launched, the evaluation window will be shown:



Do not try to manually enter the serial number as errors can easily be made.

The easiest way for you to enter your serial number is to copy your entire order email to the clipboard. SyncBackSE will find the serial number and automatically paste it into the text field. You can copy the email before SyncBackSE is running, or even when the serial number window is being displayed.

Click "OK" once the serial number has been entered.



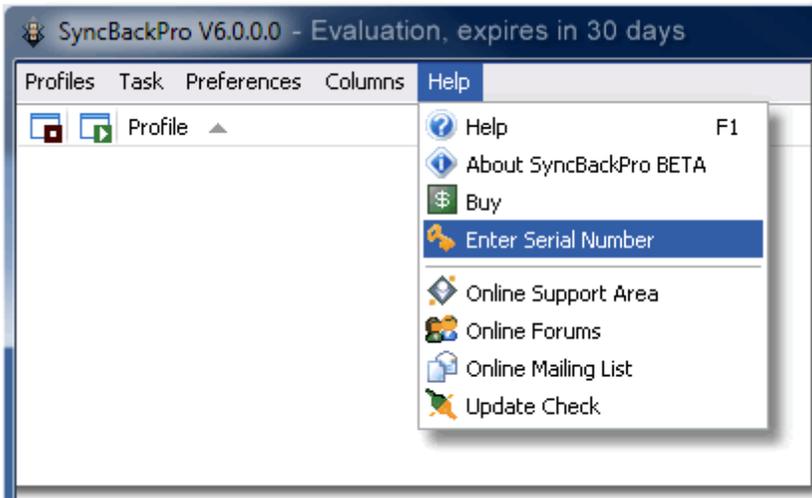
The "OK" button will only appear once the serial number has been entered correctly.

How to Copy and Paste

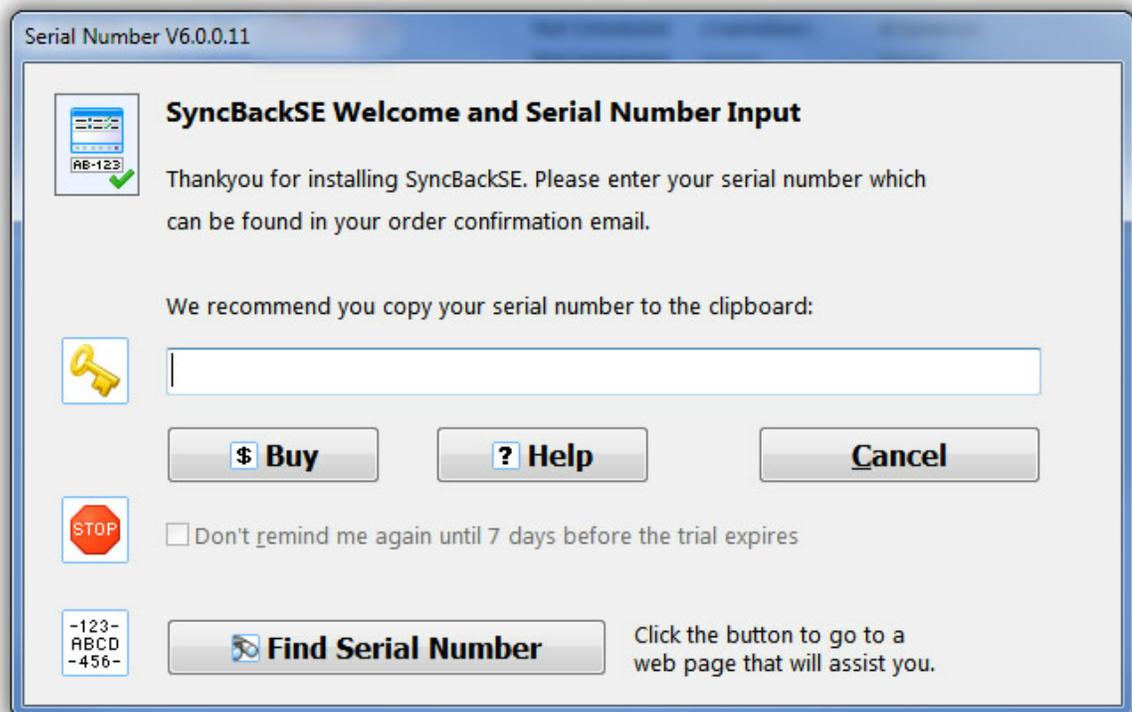
Copying and pasting is the best way to ensure that you have entered the correct serial number, as it is easy to mistake the number zero '0', for the letter 'O'.

For those who are unsure on how to quickly copy and paste your serial number, here's how:

1. Run SyncBackSE or select 'Enter Serial Number' available via the Help menu:



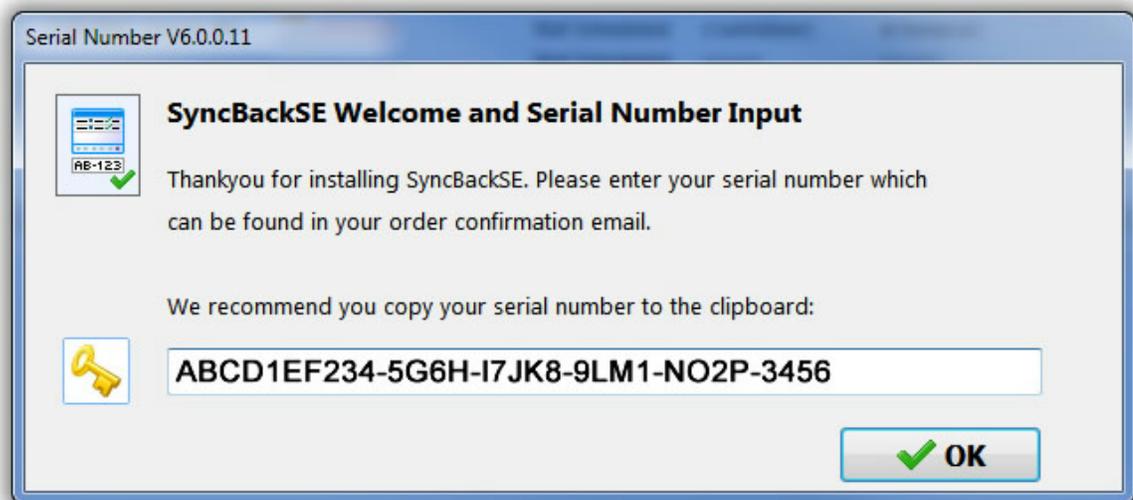
The Evaluation window will open:



2. Run your email program and click once in the body of your confirmation order email.

3. Hold down the 'Ctrl' and 'A' keys on the computer keyboard. This will select the whole body of your email.
4. Copy by holding down both the 'Ctrl' and 'C' keys then switch back to the SyncBackSE evaluation window.
5. Paste by holding down the 'Ctrl' and 'V' keys.

Your serial number will be automatically pasted into the evaluation window:



6. Click the 'OK' button

Congratulations! You have now licensed SyncBackSE.

Safe Installation

All versions of SyncBackSE can be safely installed over an existing installation. By doing this, you will ensure any profiles you have created continue to be active.

Importing Profiles

To import a Profile use the 'Import Profile' menu item under 'Profiles'. Read the special section Importing V3 Profiles.

2.3 Profiles Backup for V5 Users

Profiles Backup for SyncBackSEV5 Users

When SyncBackSE V6 is first run it will check to see if it has been installed over SyncBackSE V5. If so, it makes a backup of the existing profiles and settings of V5. The purpose of this backup is so that if you decide to uninstall V6 and install V5, then V5 can restore your old settings and profiles. It is important to note that you must install the latest version of SyncBackSE V5 to be able to restore your settings and profiles.

You can choose which folder to make a backup of the profiles and settings in, but it is recommended that you do not and instead use the default. The default location is a sub-directory of your users application data folder. After SyncBackSE has made the backup it will open a Windows Explorer window so you can see which folder it is and the files in it.

The backup files will be automatically deleted by SyncBackSE once a valid serial number is entered and 60 days or more have passed since V6 was first run. You can manually delete the backup files yourself if you are sure you are not going to revert back to V5.

2.4 Importing V3 Profiles

Upgrading from SyncBack/SE V3 to SyncBackSE V6

If you've been using SyncBack/SE V3 you would have already created profiles. To import your profiles you will be asked to browse to the appropriate location on your computer. If you installed SyncBack/SE V3 using the default settings, SyncBackSE V6 will locate your profiles automatically for you, and you need only click 'OK' in the pop-up explorer-type window.

You may create a profile at any time by using the 'New' button on the lower menu, or selecting 'New' from the 'Profiles' main menu located at the top of the program window.

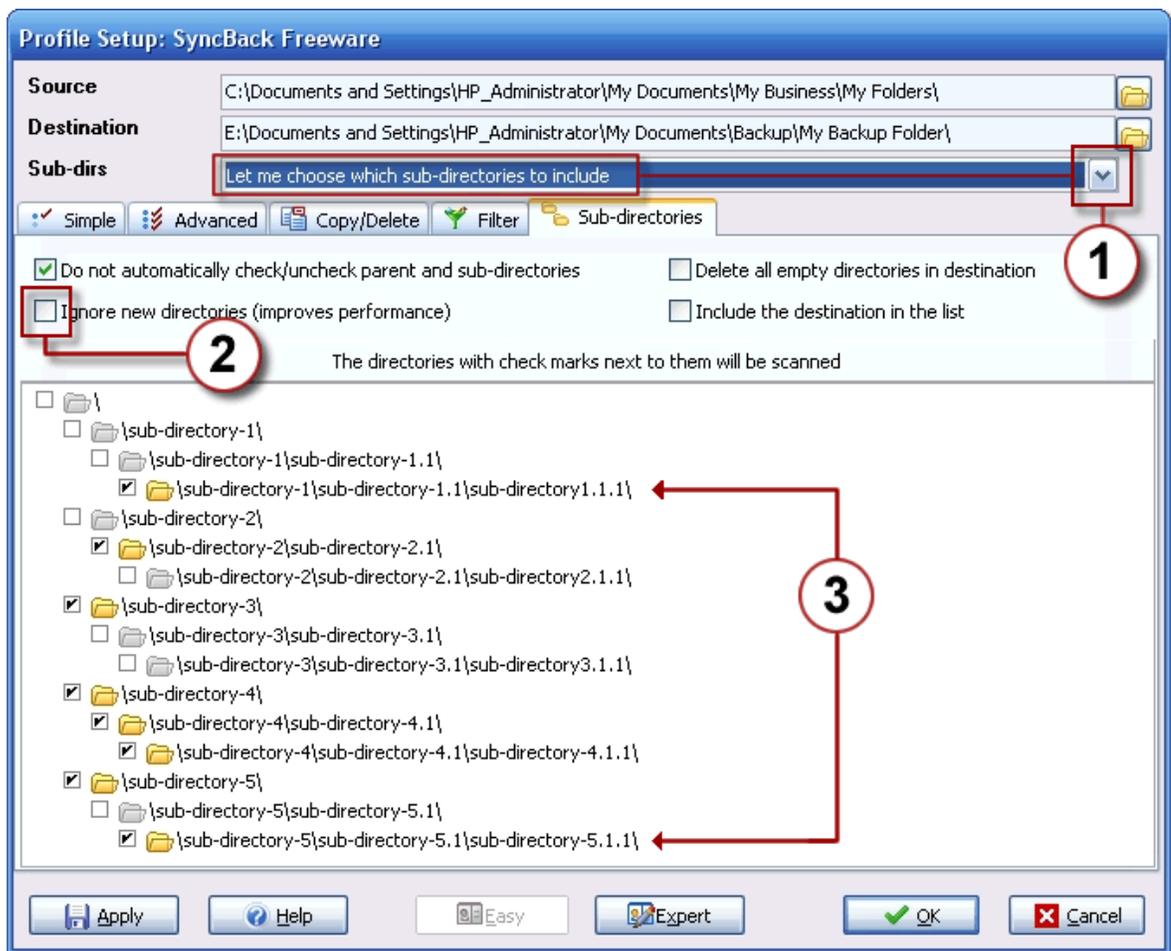
Learn more about Exporting/Importing Profiles in SyncBackSE.

'Difficult' to convert Profiles

Under certain situations the directory selections for a profile cannot be completely converted to the new format. Profiles that may not be converted correctly have the following characteristics in SyncBack V3 Freeware:

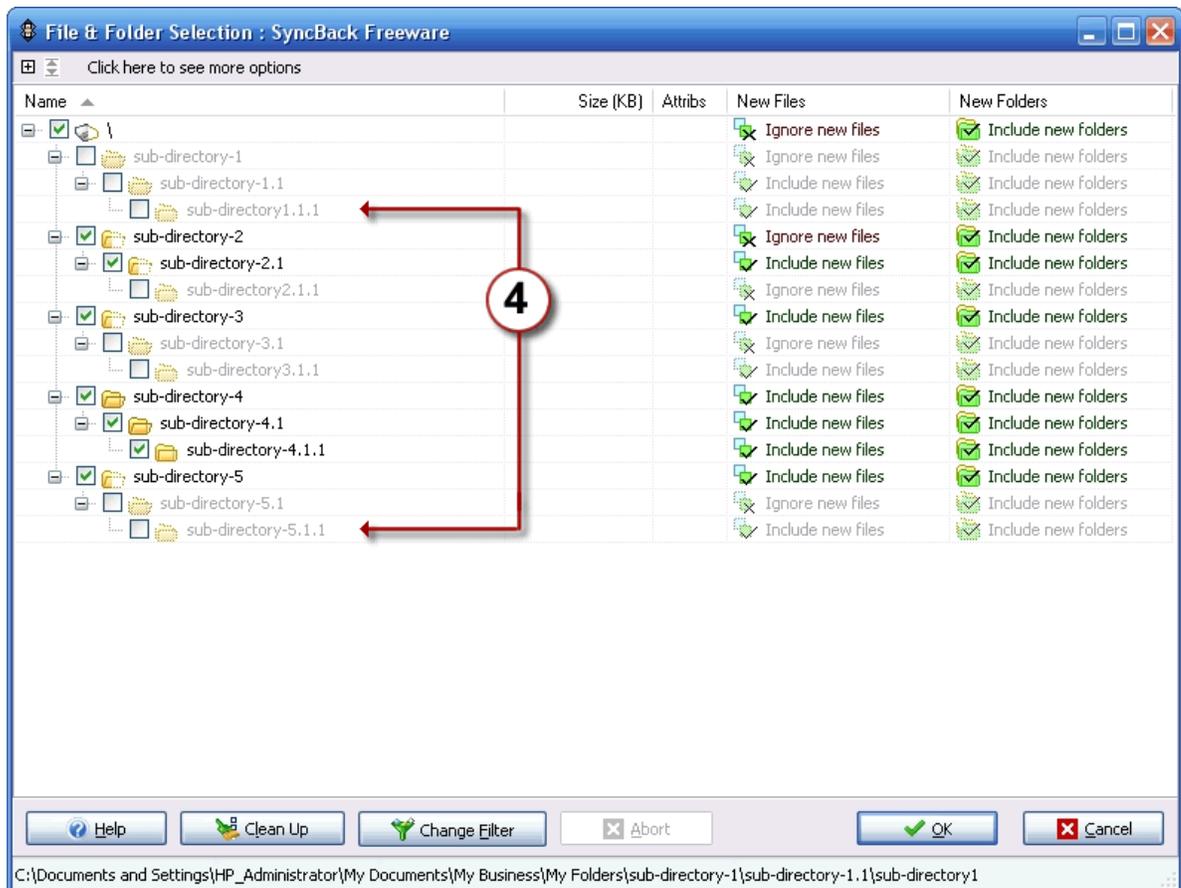
- The profile has the **Let me choose which sub-directories to include** option enabled (see point 1 below).
- The **Ignore new directories (improves performance)** option is not enabled (see point 2 below).
- A child directory (with no child directories of its own) has been selected but its parent directory has not been selected (see point 3 below).

SyncBack Freeware Screenshot



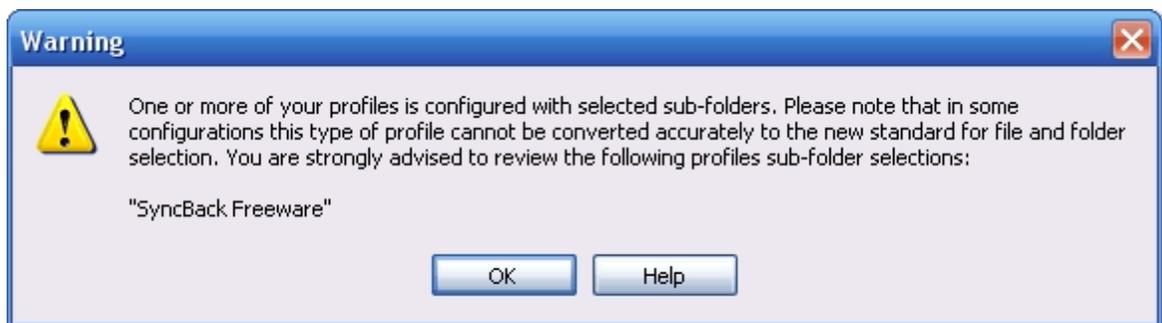
In the following situation when the settings are converted, two of the directory selections are lost (see point 4 below):

SyncBackSE V6 Screenshot



As you can see above, the directories **sub-directory 1.1.1** and **sub-directory 5.1.1** are no longer selected.

When SyncBackSE V6 is first run it will convert your profiles. If any of your profiles may have lost directory selections, then a warning message will appear listing the profiles that may be affected:



In the example shown above of a profile called **SyncBack Freeware** the sub-folders **sub-directory 1.1.1** and **sub-directory 5.1.1** are affected and so the warning dialog is given.

The new folder and file selections in V6 will now allow for complete control over what is to be copied and ignored. Unfortunately it is impossible to do a complete conversion of the old

folder selection settings due to the storage method used in previous versions. We apologize for the inconvenience.

2.5 Creating Your First Profile

Creating Your First 'Profile'

A profile stores information about the folders or files you would like to backup or synchronize using SyncBackSE. Profiles can be very specific as to what, when, and how a given task is performed, but we are going to be concentrating on creating a simple backup profile.



Be aware that different settings and choices will become available during the profile creation process depending on what you would like SyncBackSE to do. If you are uncertain in any way about the different options available, please read the Understanding Backup and Synchronization before you create a profile.

The Profile Setup Wizard walks you through the process of setting up your profile.

The default settings in the SyncBackSE Profile Setup Wizard will help ensure you will easily create a Backup profile.

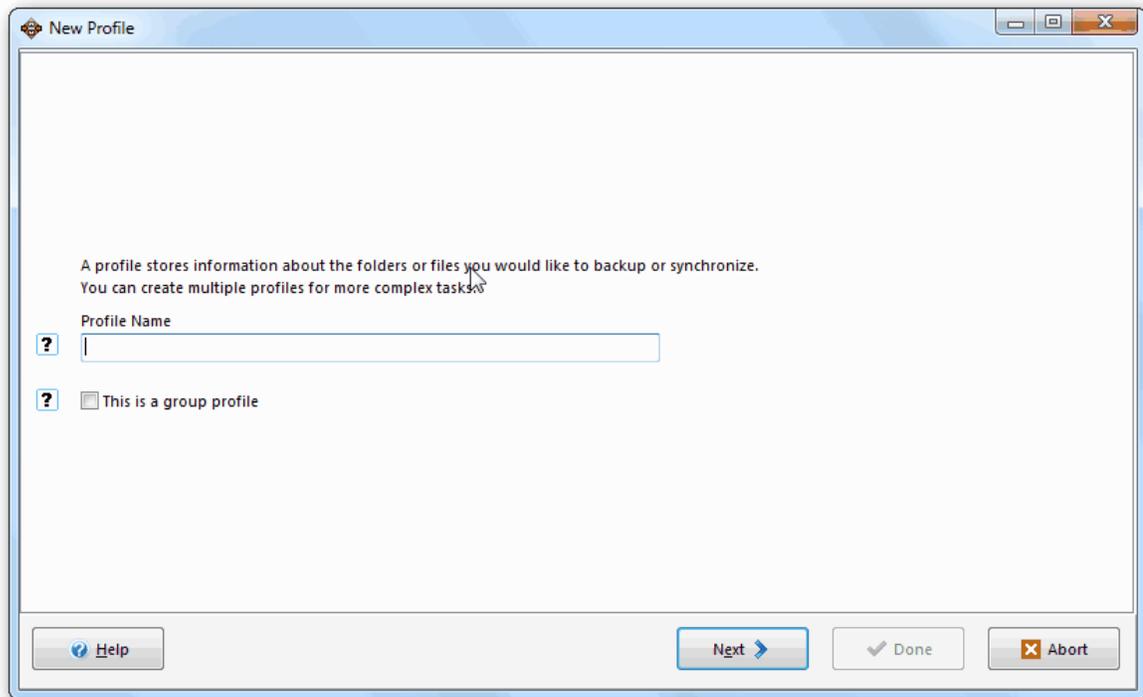
Click the **New** button located on the lower left of the program window:



Alternatively, choose **New** from the Profiles menu on the top left of the program window.

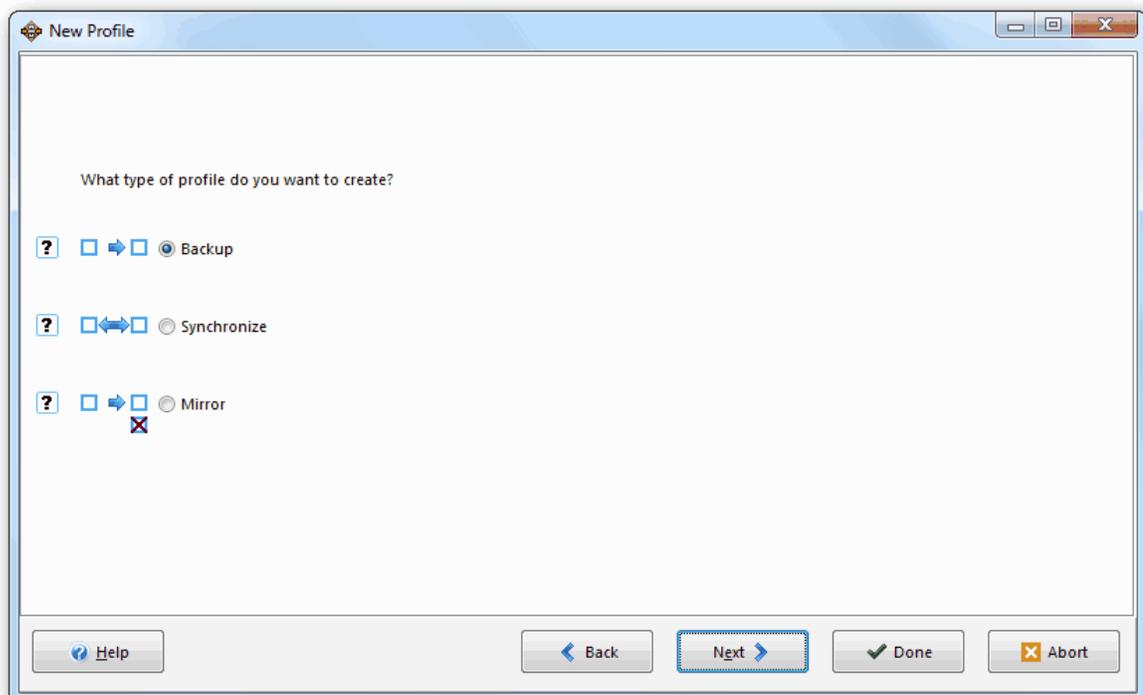
The Profile Wizard window will appear. The window is large to accommodate the varied settings and input fields that can appear during the profile setup process depending on the choices you make. If you would like to know more about a program window simply click the F1 key to view the help section relating to it.

Enter a name for your new Profile:



Click **Next** located at the lower right of the window.

For this example we are creating a backup profile which is the default option:

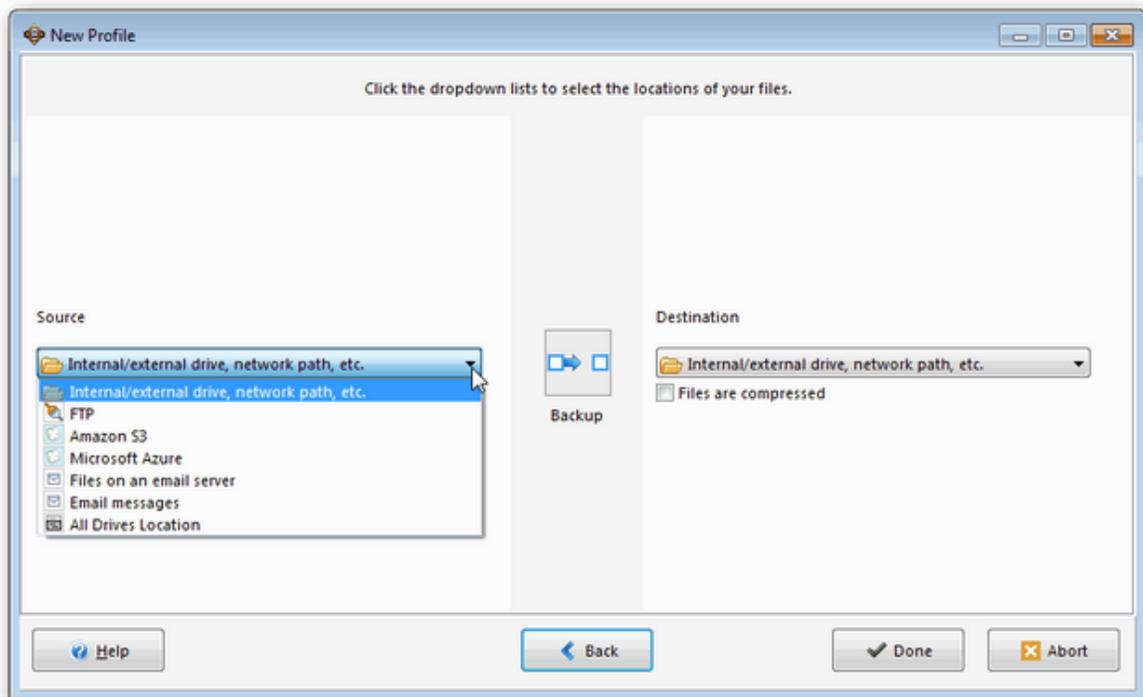


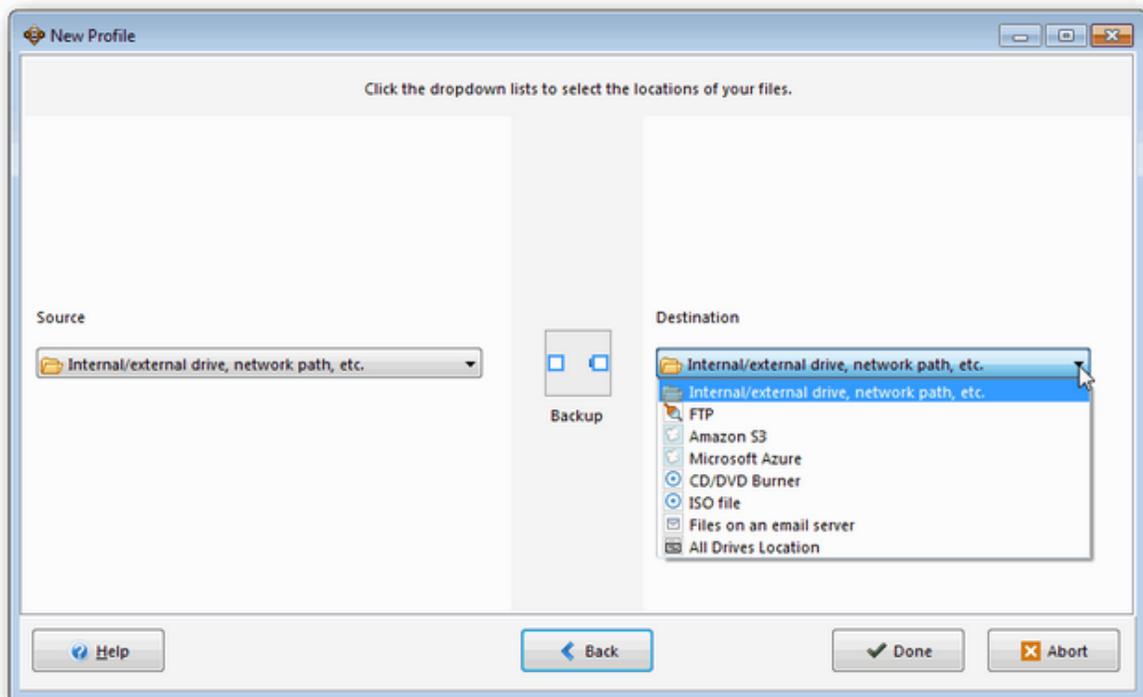
The wizard will ask whether you wish to choose the Source and Destination. The Source is

where you are copying your files from, and the Destination is where you are copying your files to.

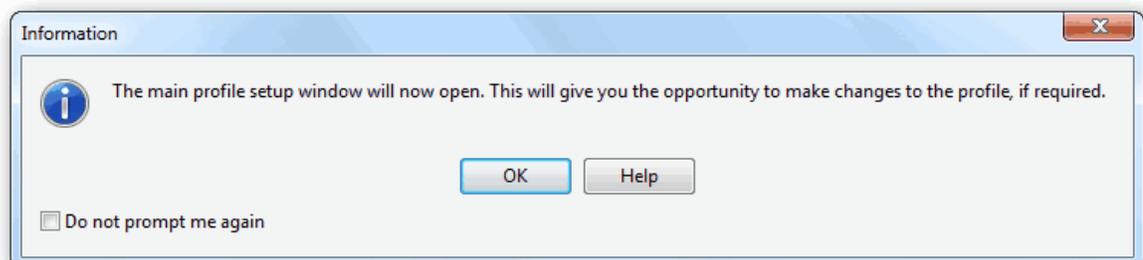
The **Internal/External drive, network path etc.** option is always the default on each side unless a different option is selected from the drop down lists. Available options can vary depending whether it will be the Source or Destination. It should be noted that selecting a non-default option for one side generally means you can only select the basic/default ('drive or network path') option for the other side. This is due to internal design aspects.

In this example where the default settings are used, you need only click the **Done** button. But if you choose non-default settings, an additional **Next** button may appear, whereby you can optionally specify additional settings (alternatively, you can complete them in the main Profile Setup window later). For example, if you choose FTP from the drop down list, and click **Next**, you will be prompted for your FTP details. For technical reasons, if you do click **Next** (where available), you will not be able to switch Back to the previous wizard screen and will thus - if you change your mind - need to **Abort** the wizard and restart the process.

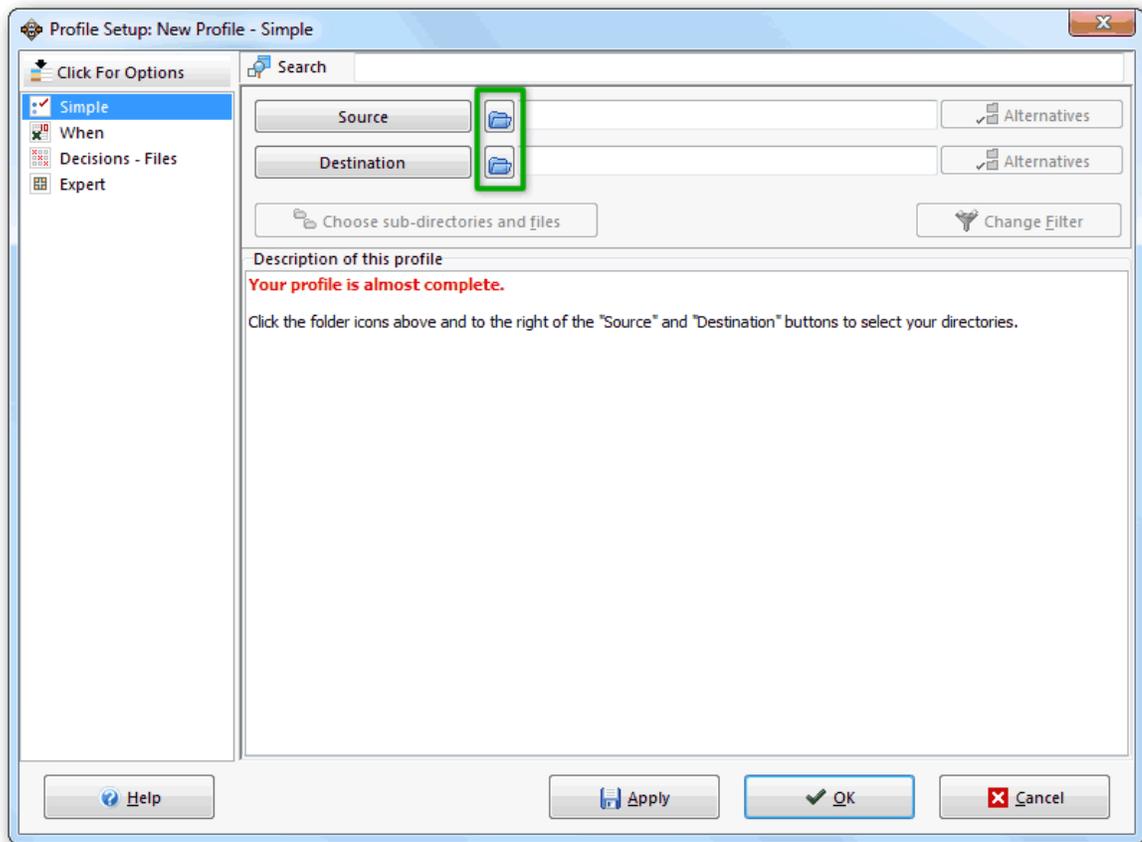




An information window now appears informing you that you will be able to view and make changes to your profile. Click **OK**:

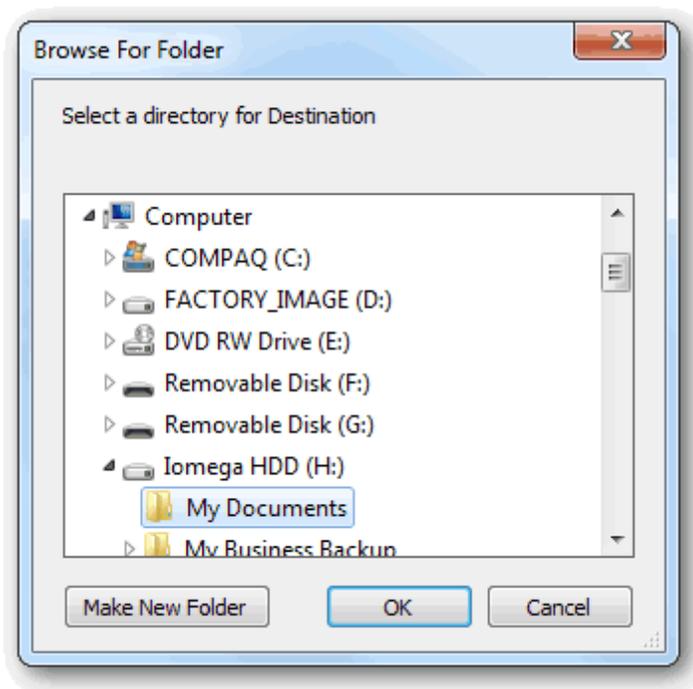


The Profile Setup window now opens. You will need to define your Source and Destination locations by clicking each of the flashing blue folder icons. Be aware these will change to yellow folder icons after a few seconds:

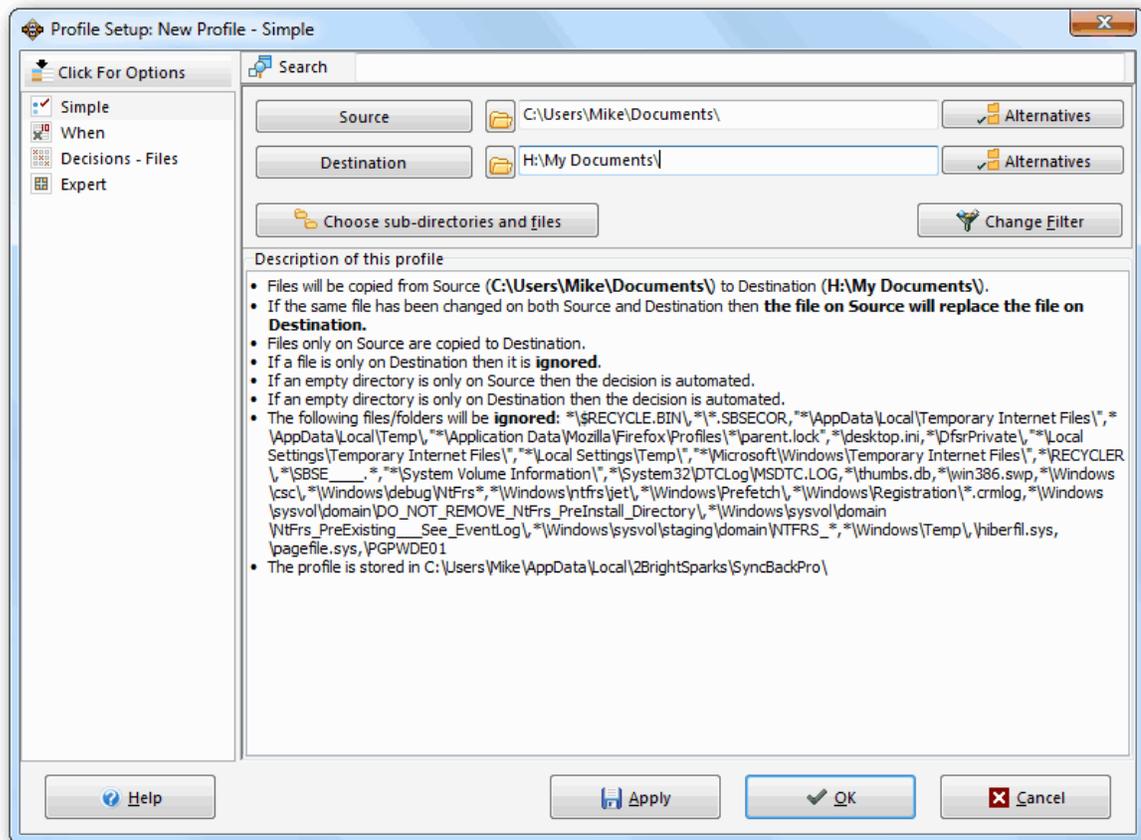


When you click the folder icon a **Browse For Folder** window will appear in which you will locate your source - Click **OK**.

In this example the destination has been defined as the My Documents folder in an external drive:



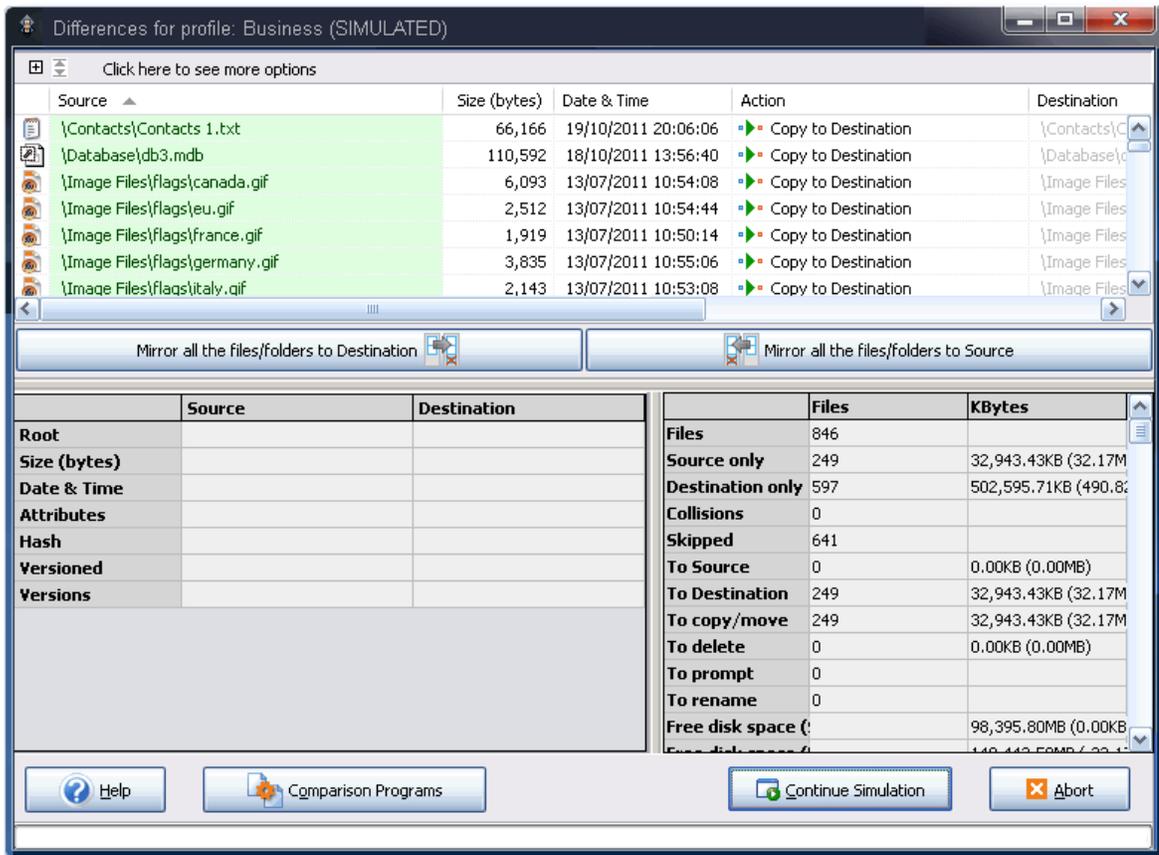
The Profile Setup Window will now display all the selections you have made. You will notice that by default SyncBackSE has automatically built options that will make your backup proceed more reliably and quickly than if you simply copy the contents of the My Documents folder into another drive:



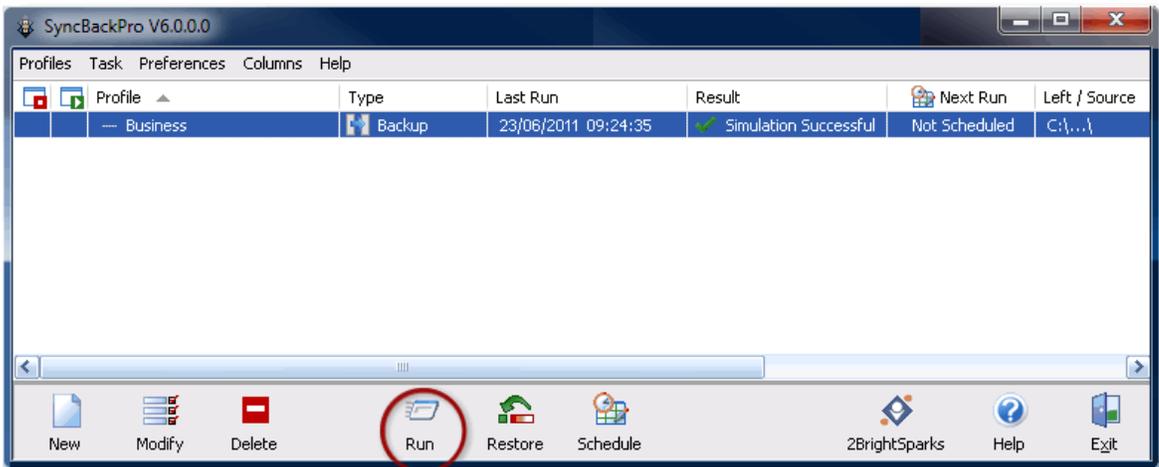
Click **OK**. A Window will open which asks whether you would like to perform a simulated run. This allows you to check the profile functions correctly without actually copying any files. Click **Yes**.

In the following example a Business profile has been defined, the simulation accepted, and the Differences window opens.

The Differences window shows all the files that would be copied in an actual run. Click **Continue Simulation**:



The main program window will now open. You may now run this backup profile at any time by selecting a profile and then clicking the **Run** button:



The final section of this Quick Start guide shows you how to Run Your Profile.

2.6 First Run

Running a Profile for the First Time

SyncBackSE processes files very quickly. Therefore, if you are only backing up or synchronizing a few files, you will may not see the progress bar at the bottom of the window.

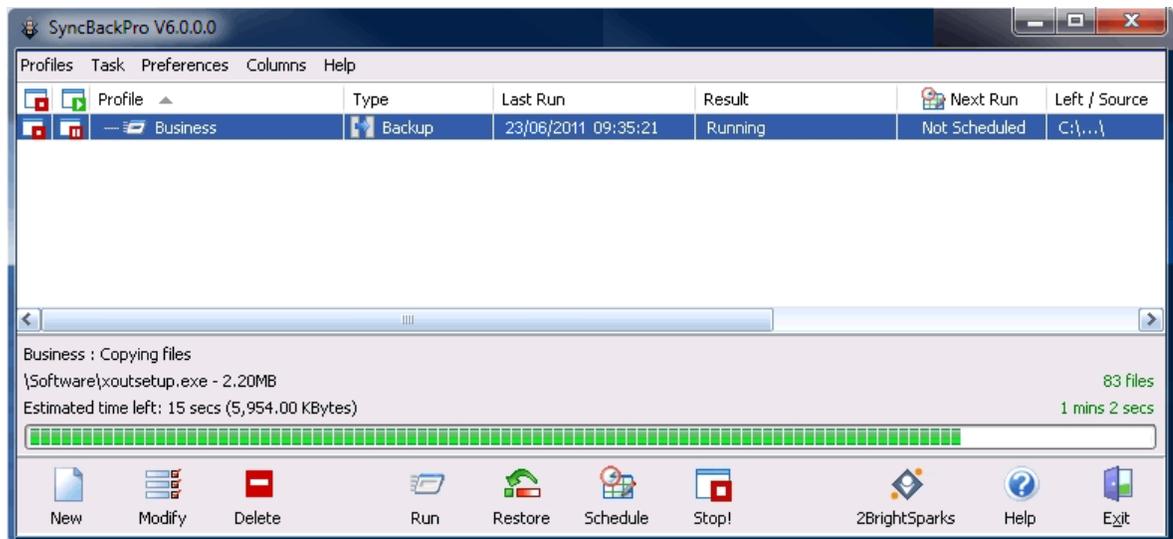


When a profile is running (or paused) it will have two icons shown to the left of it:

Click the  icon to stop the profile.

Click the  icon to pause the profile. If the profile is already paused then it will instead show the  icon. Click it to continue the profile.

A progress bar will appear at the bottom of the window as the files are being processed:



As the profile is being processed, an icon will also appear in the System Tray located on the bottom right of your screen:

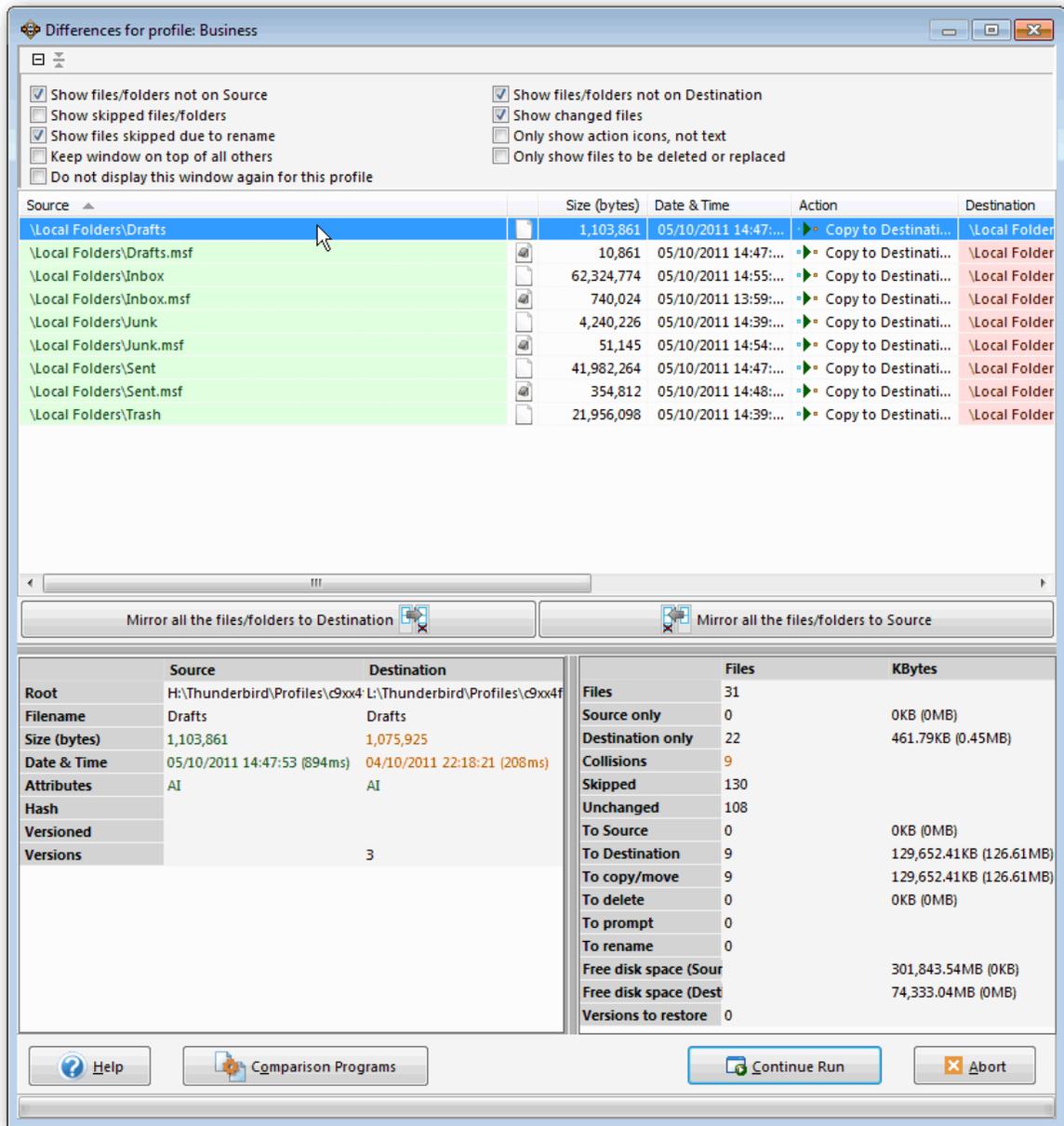


SyncBackSE icon when a profile is not running



Animated SyncBackSE icon when a profile is running

Depending on your settings, the Differences Window may appear after the initial scan takes place. The Differences Window shows what will happen to the files (whether they will be copied, deleted, or moved). Once you have reviewed the differences, click "Continue":



 Note how in the Differences Window screenshot below, the toggle icon has been clicked to show more options:



In this example the Differences Window shows 2 collisions. A "collision" is when a file in the source and destination differ but have the same name. In other words, the file is in both the source and destination but is modified in some way, perhaps by date, size etc.

	Files	KBytes
Files	31	
Source only	0	0KB (0MB)
Destination only	22	461.79KB (0.45MB)
Collisions	9	
Skipped	130	
Unchanged	108	
To Source	0	0KB (0MB)
To Destination	9	129,652.41KB (126.61MB)
To copy/move	9	129,652.41KB (126.61MB)
To delete	0	0KB (0MB)
To prompt	0	
To rename	0	
Free disk space (Source)		301,843.54MB (0KB)
Free disk space (Destination)		74,333.04MB (0MB)
Versions to restore	0	

Buttons:

A notification of collisions occur in the **Differences** window which appears by default when making a backup (note however there are circumstances when the **Differences** window does not appear, for example when the user has chosen not to show the window).

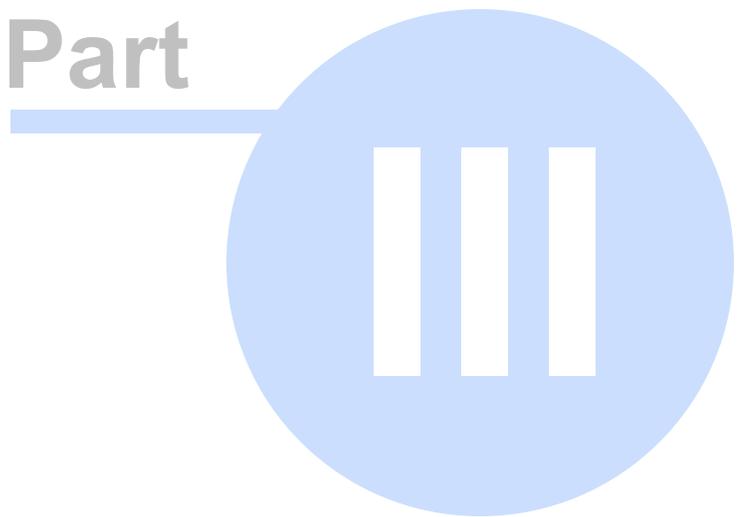
Collisions are shown in red in the **Differences** window to highlight there are going to be changes made when you continue the profile task. If the user views the **Differences** window carefully, the user has the option to make choices about whether they want to accept the changes SyncBackSE will make. The user has the option to bypass these choices by selecting a specific item in the **Differences** window with a right click. A different action may then be chosen.

After the profile has been processed the main window will look slightly different as the temporary **stop** and **pause** icons are no longer viewable.

You have now successfully created a simple backup profile. To create a Group Profile you'll need to create two profiles or more. You'll then have the opportunity of running these profiles as one. To find out more about this feature go to [Creating a Group Profile](#).

Using SyncBackSE

Part



3 Using SyncBackSE



A Guide to Using SyncBackSE

SyncBackSE is a powerful, easy to use commercial program that helps you backup and synchronize your files to: the same drive; a different drive or medium (CDRW, CompactFlash, etc); an FTP server; a Network; or a Zip archive. SyncBackSE can copy locked and open files as easily as the usual closed files. This allows you to backup, synchronize and restore any data you wish to.

Using SyncBackSE guides you through the essential functions and operation of SyncBackSE.

3.1 Basic Operation

Essential Knowledge

If you have never used SyncBackSE before, you are strongly advised to read this section of the help file:

Basic Operation

The Main Window

Exporting and Importing

Creating a Profile

Running a Profile

Creating a Group Profile

Restoring a Backup

Creating a Schedule

Preferences Main Menu

Columns Main Menu
Global Settings
Dialogs
Comparison Programs
Log Settings
Shared Settings
Exploring SyncBackSE

3.1.1 The Main Window

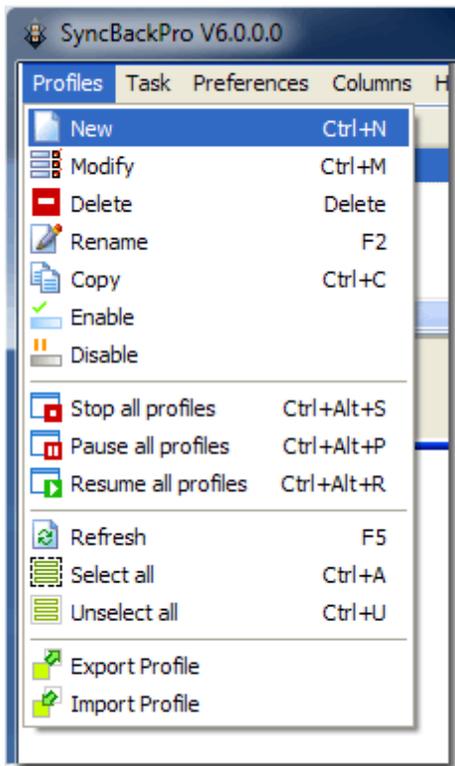
SyncBackSE: The Program Interface

SyncBackSE is a simple program to use with many options and user definable parameters. The program is perfect for the novice and expert alike.

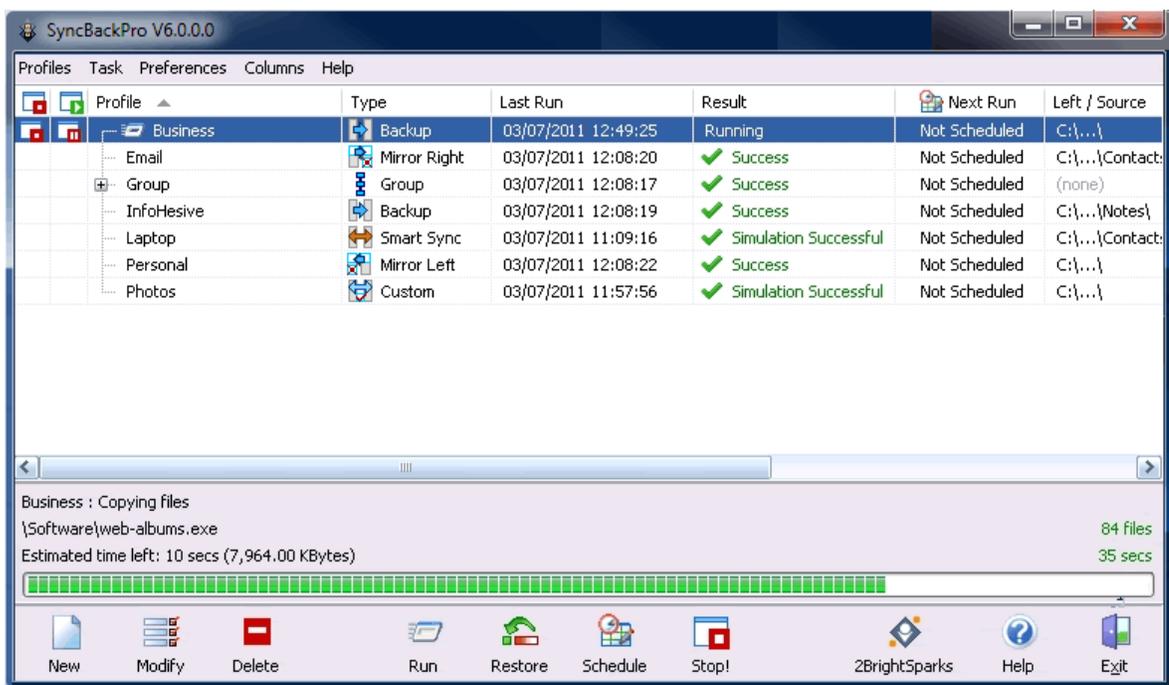
All the essential tasks in SyncBackSE can be carried out from the main window by clicking an icon on the lower toolbar:



SyncBackSE also has a main menu that provides easy access to all its functions, and a right click pop-up menu that is available when a profile is highlighted:

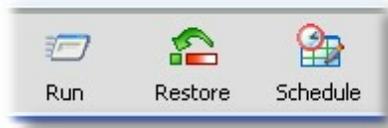


The main window below shows SyncBackSE running a backup 'Profile' called 'Business':



Once a Profile has been defined, the user need only click the 'Run' or 'Restore' buttons to carry out the task. Alternatively the Profile can be scheduled to run at certain times and

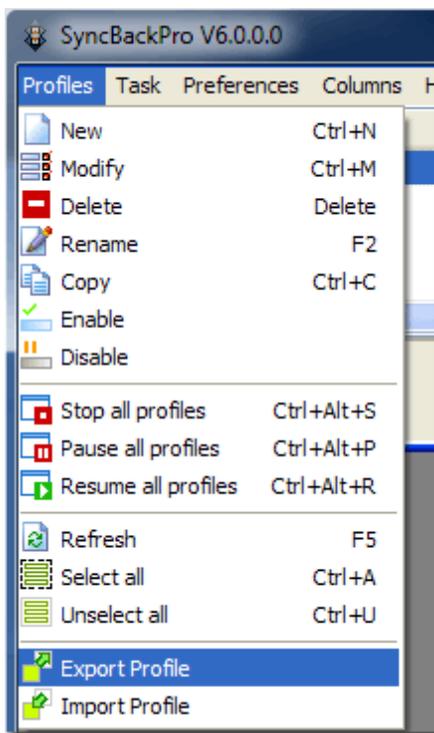
dates:



3.1.2 Exporting and Importing Profiles

Exporting and Importing Profiles

You can copy a profile from one installation of SyncBackSE to another by exporting and importing profiles. This also lets you make a backup of your profiles settings.



- **To export a profile:** select the profile (or profiles) in the main window and select **Profiles -> Export Profile** from the main menu. Then choose where to save the profile. To export all your profiles press Ctrl-A then select **Profiles -> Export Profile** from the main menu. Note that unless your groups are expanded the profiles in them will not be selected.
- **To import a profile:** select **Profiles -> Import Profile** from the main menu then choose the file(s) that contains the profile. You can import profiles from any current or previous

version of SyncBackSE or SyncBack freeware. When importing profiles exported from versions of SyncBack freeware or SyncBackSE V3 then some directory selections may not be imported. It is recommend that **group profiles** are imported last to ensure the profiles in the group exist at the time the group is imported, otherwise the group may fail to import.

Important

- It is not recommended to import profiles exported from newer versions of SyncBackSE.
- Profiles exported from old versions of SyncBackSE or SyncBack (V3 and before) will import incorrectly if the Windows XP/2003 non-Unicode locale is different. For example, if you import a V3 profile exported from XP where Chinese was the non-Unicode locale, then you must set your non-Unicode locale to Chinese.

Importing SyncBackSE Version 3 Profiles

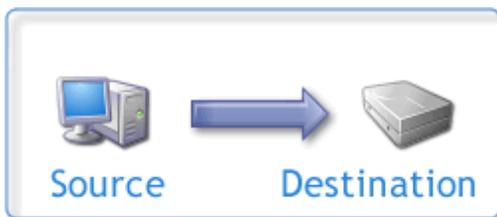
If you are upgrading from SyncBack freeware V3, or SyncBackSE V3, and wish to import your profiles, read our special guide to Importing V3 Profiles.

3.1.3 Creating a Profile

SyncBackSE Profile Types

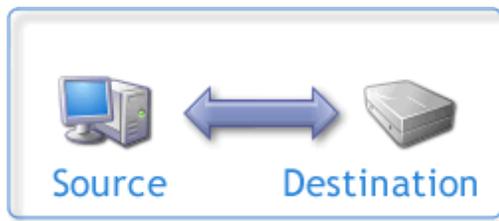
Profile Types

Choose a Backup profile when you want to backup (copy) your new and modified files to another place, e.g. an external hard drive or an FTP server.



Backup profile

Choose a Synchronization profile when you have two directories in which files are changing and you want them both to contain the same files. For example, you may have a directory on your computer containing documents you are working on. A colleague may also be working on documents on his computer. If you want to copy his new and modified files to your local directory, and copy your new and modified files to his directory, then you need to use a Synchronization profile.



Synchronization profile

Choose a Mirror profile when you want one directory to be identical to another directory. It is not the same as a backup because it deletes files. It is also not the same as synchronization because it only copies files in one direction.



Mirror profile

What is the difference between Synchronization and Smart Synchronization?

Smart Synchronization keeps track of what changed the last time the profile was run so that it knows if a file has been deleted, created, or changed since the last profile run. This helps you (and SyncBackSE) make an informed decision about what to do when something changes. It also gives you a comprehensive choice of options on what to do with a file when specific things happen, e.g. the file is deleted for the source but not the destination. Regular synchronization does not keep track of changes and has a limited set of options. Whenever possible you should use Smart Synchronization instead of regular synchronization. If you import a sync profile from SyncBack/SE V3 then it will not be using Smart Synchronization (as that feature was not available in older versions).

Choose a Group profile when you want to create a profile that contains other Backup or Synchronization profiles. This allows you to run many profiles at once.

So what is the difference between a backup process and a synchronization process?



A backup process copies files in one direction: from the source to the destination. A Synchronization process copies files in both directions: from the source to the destination and from the destination to the source.

Creating Profiles

A profile stores information about the folders or files you would like to backup or synchronize using SyncBackSE. Profiles can be very specific as to what, when, and how a given task is performed, but we are going to be concentrating on creating a simple backup profile.



Be aware that different settings and choices will become available during the profile creation process depending on what you would like SyncBackSE to do. If you are uncertain in any way about the different options available, please read the Understanding Backup and Synchronization before you create a profile.

The Profile Setup Wizard walks you through the process of setting up your profile.

The default settings in the SyncBackSE Profile Setup Wizard will help ensure you will easily create a Backup profile.

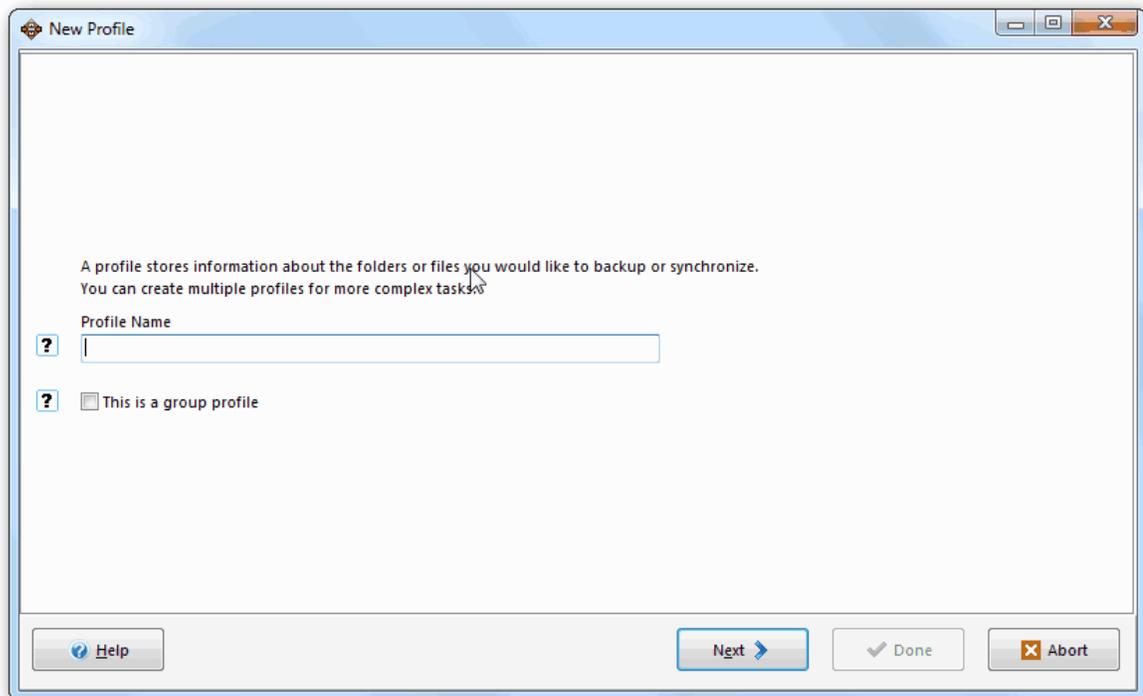
Click the **New** button located on the lower left of the program window:



Alternatively, choose **New** from the Profiles menu on the top left of the program window.

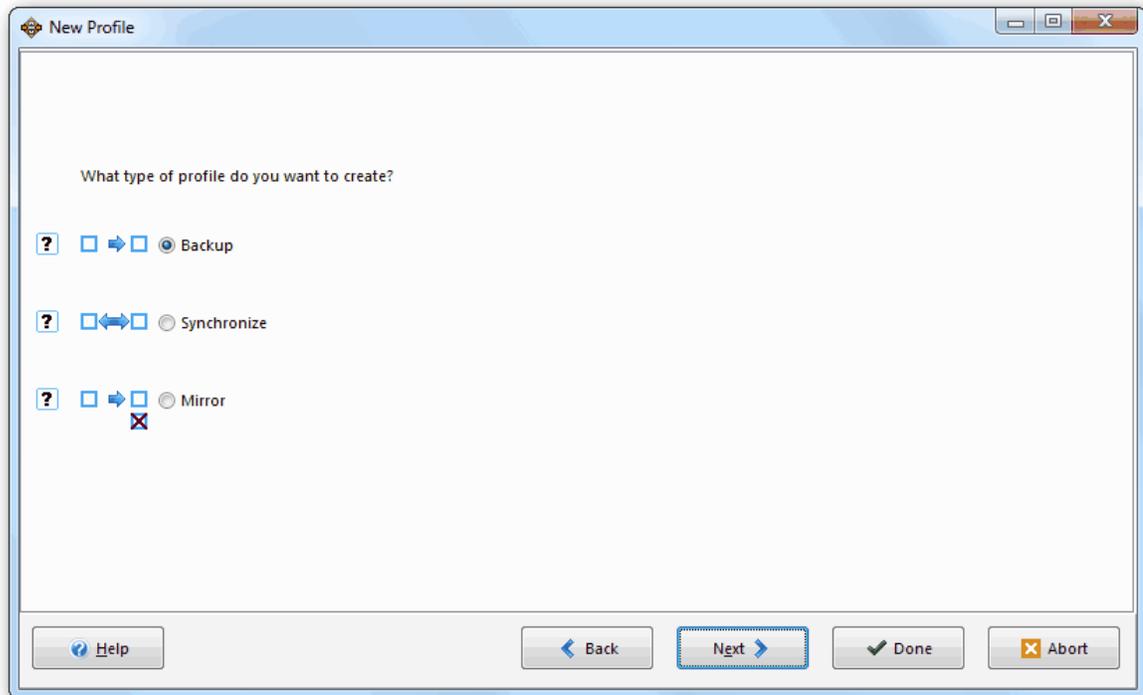
The Profile Wizard window will appear. The window is large to accommodate the varied settings and input fields that can appear during the profile setup process depending on the choices you make. If you would like to know more about a program window simply click the F1 key to view the help section relating to it.

Enter a name for your new Profile:



Click **Next** located at the lower right of the window.

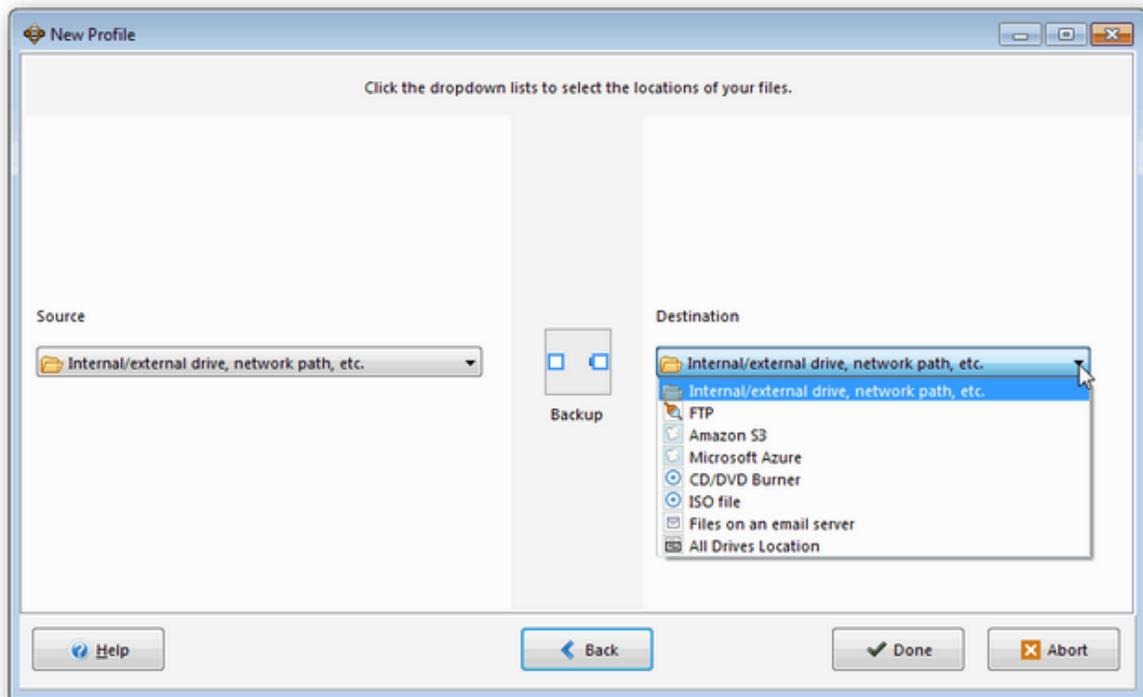
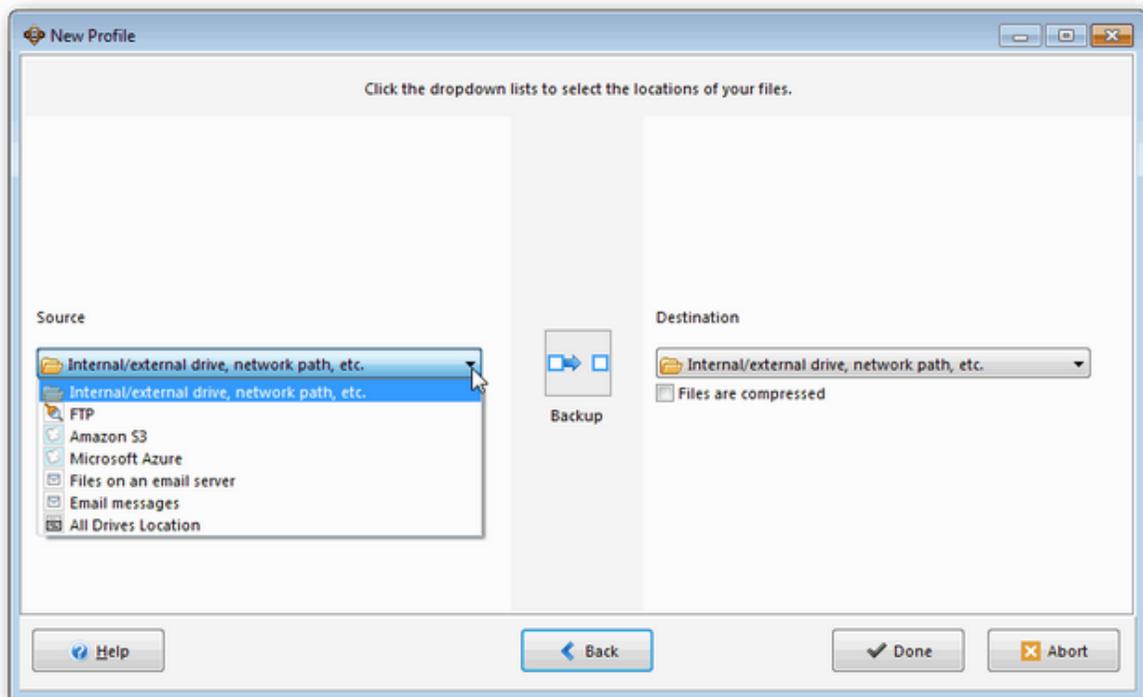
For this example we are creating a backup profile which is the default option:



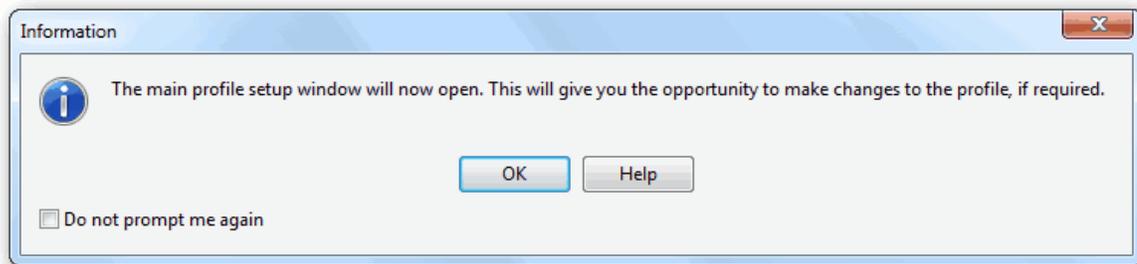
The wizard will ask whether you wish to choose the Source and Destination. The Source is where you are copying your files from, and the Destination is where you are copying your files to.

The **Internal/External drive, network path etc.** option is always the default on each side unless a different option is selected from the drop down lists. Available options can vary depending whether it will be the Source or Destination. It should be noted that selecting a non-default option for one side generally means you can only select the basic/default ('drive or network path') option for the other side. This is due to internal design aspects.

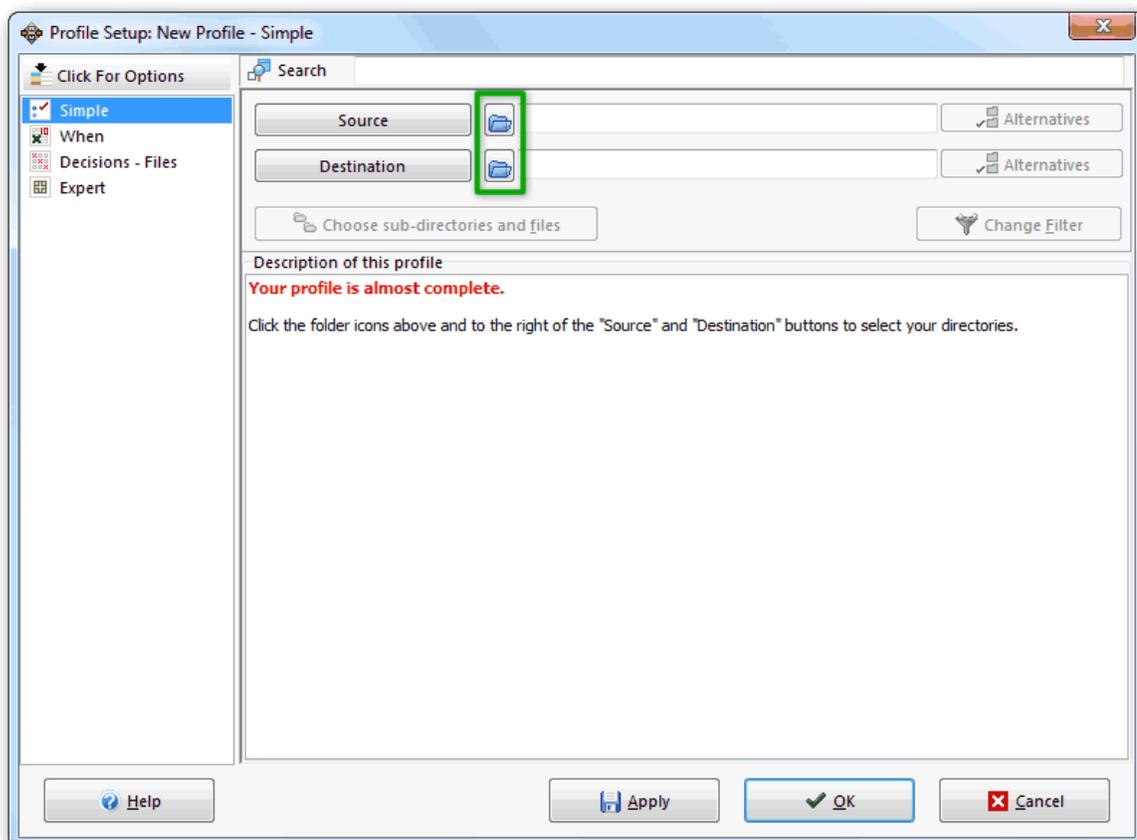
In this example where the default settings are used, you need only click the **Done** button. But if you choose non-default settings, an additional **Next** button may appear, whereby you can optionally specify additional settings (alternatively, you can complete them in the main Profile Setup window later). For example, if you choose FTP from the drop down list, and click **Next**, you will be prompted for your FTP details. For technical reasons, if you do click **Next** (where available), you will not be able to switch Back to the previous wizard screen and will thus - if you change your mind - need to **Abort** the wizard and restart the process.



An information window now appears informing you that you will be able to view and make changes to your profile. Click **OK**:

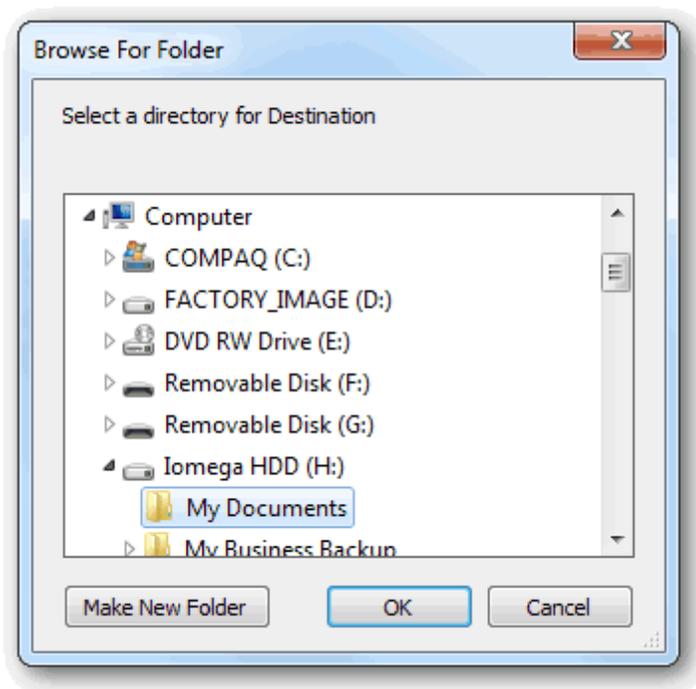


The Profile Setup window now opens. You will need to define your Source and Destination locations by clicking each of the flashing blue folder icons. Be aware these will change to yellow folder icons after a few seconds:

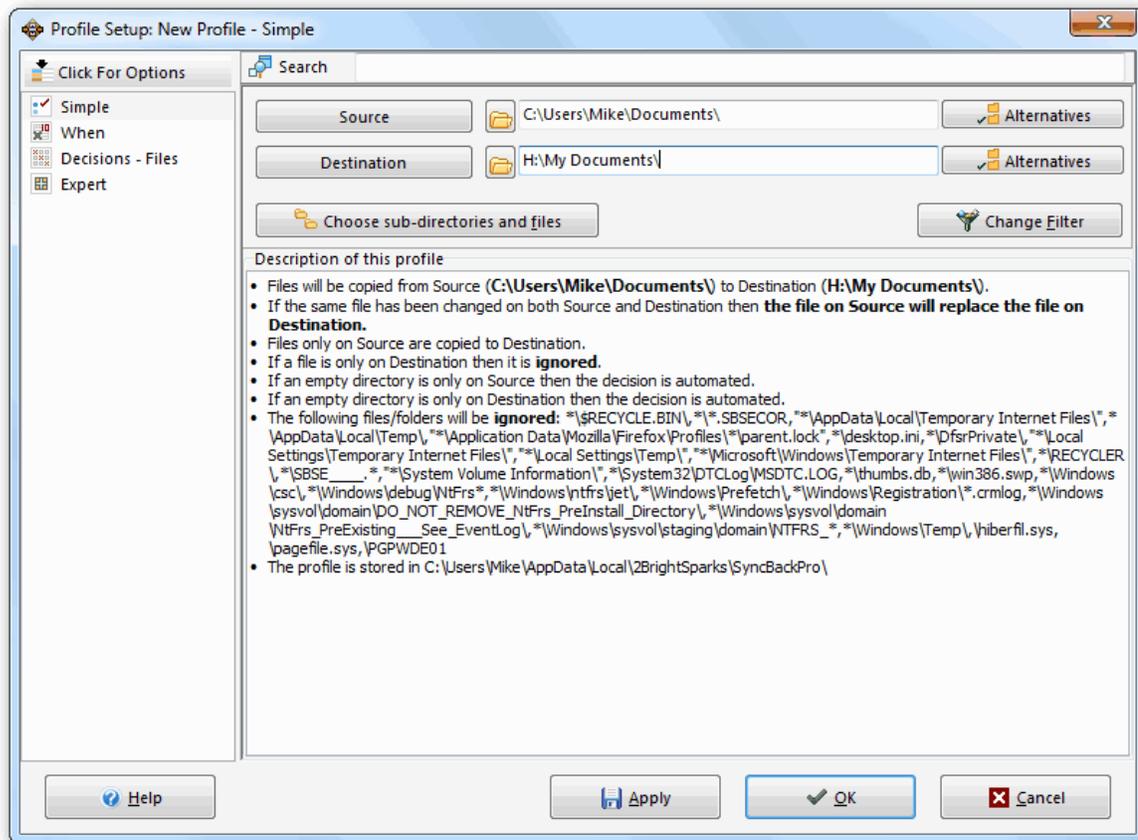


When you click the folder icon a **Browse For Folder** window will appear in which you will locate your source - Click **OK**.

In this example the destination has been defined as the My Documents folder in an external drive:



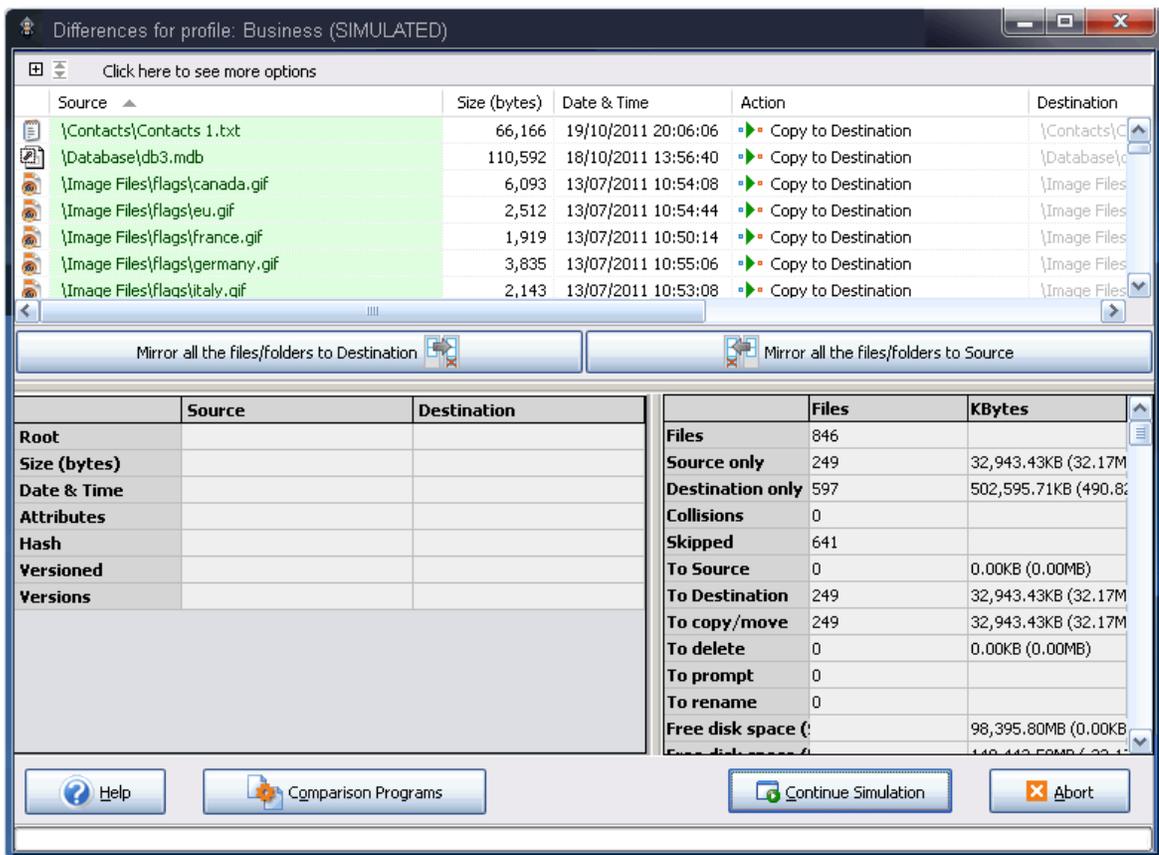
The Profile Setup Window will now display all the selections you have made. You will notice that by default SyncBackSE has automatically built options that will make your backup proceed more reliably and quickly than if you simply copy the contents of the My Documents folder into another drive:



Click **OK**. A Window will open which asks whether you would like to perform a simulated run. This allows you to check the profile functions correctly without actually copying any files. Click **Yes**.

In the following example a Business profile has been defined, the simulation accepted, and the Differences window opens.

The Differences window shows all the files that would be copied in an actual run. Click **Continue Simulation**:



The main program window will now open. You may now run this backup profile at any time by selecting a profile and then clicking the **Run** button:

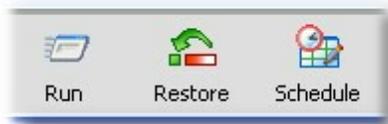


The next section of this help file shows you how to 'Run' the Profile.

3.1.4 Running a Profile

SyncBackSE in Action

To run the Profile select the profile in the main program window, then click the 'Run' button:

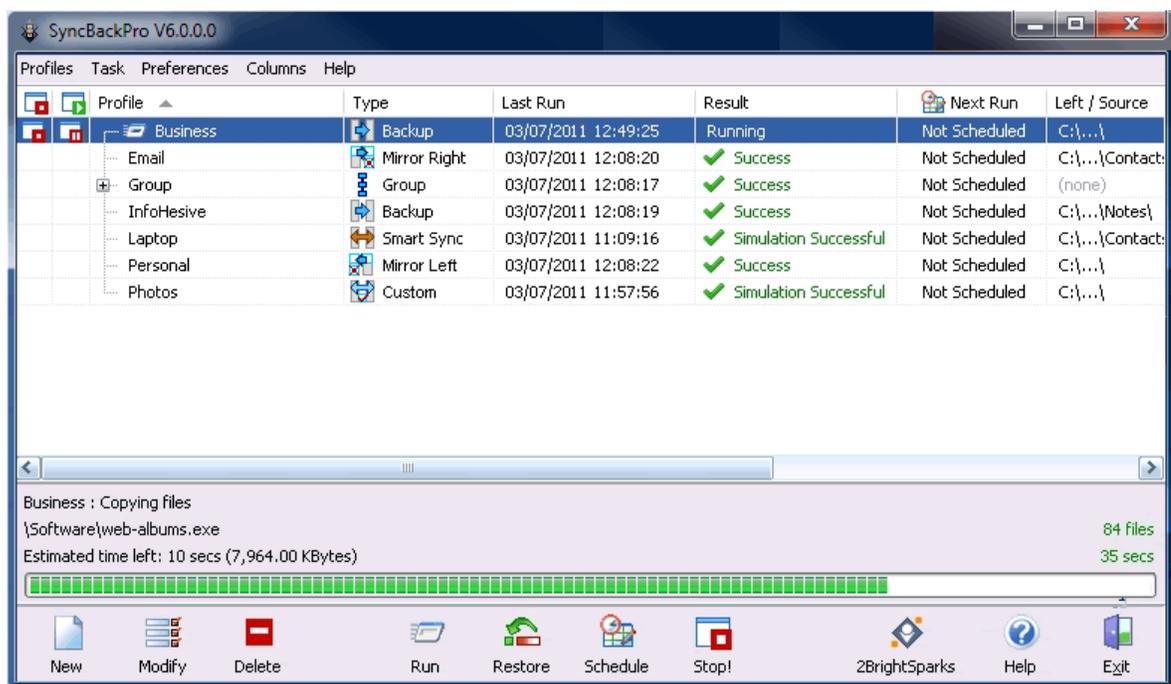


 When a profile is running (or paused) it will have two icons shown to the left of it:

Click the  icon to stop the profile.

Click the  icon to pause the profile. If the profile is already paused then it will instead show the  icon. Click it to continue the profile.

A progress bar will appear as the files are being processed:



As the profile is being processed an icon will also appear on the Windows System Tray located on the bottom right of your screen.

SyncBackSE icon when a profile is not running:



Animated SyncBackSE icon when a profile is running:



Depending on your settings, the Differences Window may appear after the initial scan takes place. The Differences Window shows what will happen to the files (whether they will be copied, deleted, or moved). Once you have reviewed the differences, click "Continue":

Differences for profile: Business

Show files/folders not on Source
 Show skipped files/folders
 Show files skipped due to rename
 Keep window on top of all others
 Do not display this window again for this profile

Show files/folders not on Destination
 Show changed files
 Only show action icons, not text
 Only show files to be deleted or replaced

Source	Size (bytes)	Date & Time	Action	Destination
\Local Folders\Drafts	1,103,861	05/10/2011 14:47:...	▶ Copy to Destinati...	\Local Folder
\Local Folders\Drafts.msf	10,861	05/10/2011 14:47:...	▶ Copy to Destinati...	\Local Folder
\Local Folders\Inbox	62,324,774	05/10/2011 14:55:...	▶ Copy to Destinati...	\Local Folder
\Local Folders\Inbox.msf	740,024	05/10/2011 13:59:...	▶ Copy to Destinati...	\Local Folder
\Local Folders\Junk	4,240,226	05/10/2011 14:39:...	▶ Copy to Destinati...	\Local Folder
\Local Folders\Junk.msf	51,145	05/10/2011 14:54:...	▶ Copy to Destinati...	\Local Folder
\Local Folders\Sent	41,982,264	05/10/2011 14:47:...	▶ Copy to Destinati...	\Local Folder
\Local Folders\Sent.msf	354,812	05/10/2011 14:48:...	▶ Copy to Destinati...	\Local Folder
\Local Folders\Trash	21,956,098	05/10/2011 14:39:...	▶ Copy to Destinati...	\Local Folder

Mirror all the files/folders to Destination
 Mirror all the files/folders to Source

	Source	Destination
Root	H:\Thunderbird\Profiles\{9xx4 L:\Thunderbird\Profiles\{9xx4f	
Filename	Drafts	Drafts
Size (bytes)	1,103,861	1,075,925
Date & Time	05/10/2011 14:47:53 (894ms)	04/10/2011 22:18:21 (208ms)
Attributes	AI	AI
Hash		
Versioned		
Versions		3

	Files	KBytes
Files	31	
Source only	0	0KB (0MB)
Destination only	22	461.79KB (0.45MB)
Collisions	9	
Skipped	130	
Unchanged	108	
To Source	0	0KB (0MB)
To Destination	9	129,652.41KB (126.61MB)
To copy/move	9	129,652.41KB (126.61MB)
To delete	0	0KB (0MB)
To prompt	0	
To rename	0	
Free disk space (Sour		301,843.54MB (0KB)
Free disk space (Dest		74,333.04MB (0MB)
Versions to restore	0	

 Note how in the Differences Window, the toggle icon has been clicked to show more options:



 In this example the Differences Window shows 6 collisions. A "collision" is when a file in the source and destination differ but have the same name. In other words, the file is in both the source and destination but is modified in some way, perhaps by date, size etc.

	Files	KBytes
Files	31	
Source only	0	0KB (0MB)
Destination only	22	461.79KB (0.45MB)
Collisions	9	
Skipped	130	
Unchanged	108	
To Source	0	0KB (0MB)
To Destination	9	129,652.41KB (126.61MB)
To copy/move	9	129,652.41KB (126.61MB)
To delete	0	0KB (0MB)
To prompt	0	
To rename	0	
Free disk space (Sour		301,843.54MB (0KB)
Free disk space (Dest		74,333.04MB (0MB)
Versions to restore	0	

A notification of collisions occur in the "Differences" window which appears by default when making a backup (note however there are circumstances when the "Differences" window does not appear, for example when the user has chosen not to show the window).

Collisions are shown in red in the "Differences" window to highlight there are going to be changes made when you continue the profile task. If the user views the Differences window carefully, the user has the option to make choices about whether they want to accept the changes SyncBackSE will make. The user has the option to bypass these choices by selecting a specific item in the Differences window with a right click. A

different action may then be chosen.

After the profile has been processed the main window will look slightly different as the temporary "stop" and "pause" icons are no longer viewable.

You have now successfully created a simple backup profile. To create a Group Profile you'll need to create two profiles or more. You'll then have the opportunity of running these profiles as one. To find out more about this feature go to Creating a Group Profile.

3.1.5 Creating a Group Profile

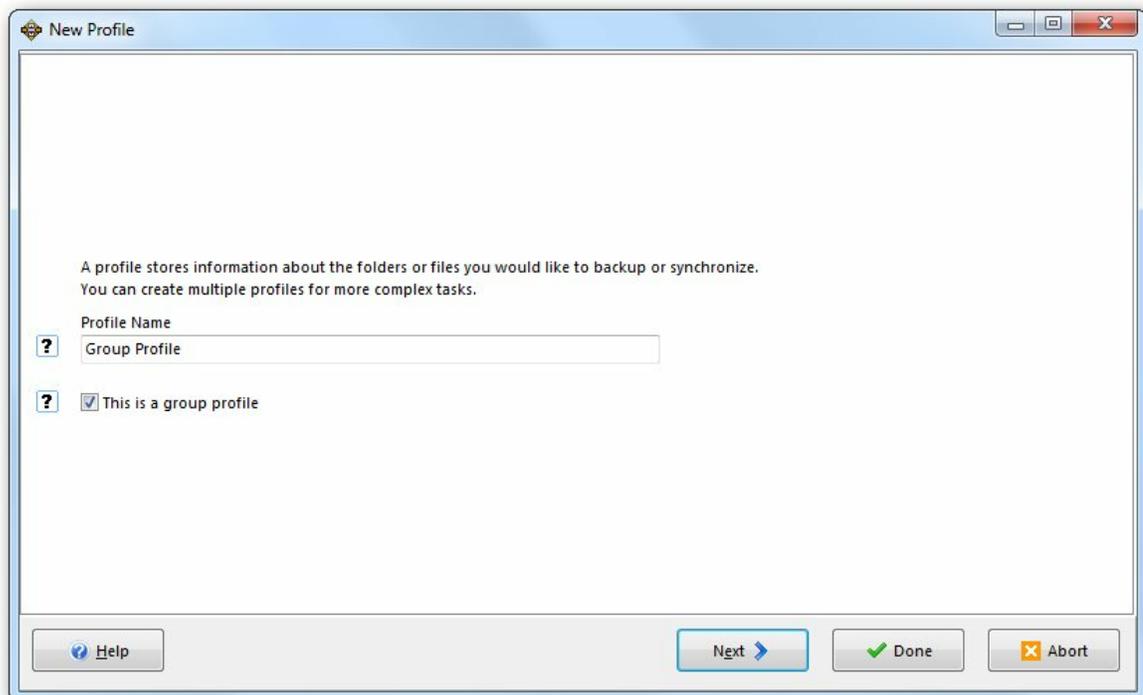
Group Profiles

Once you have created two or more profiles you may group them together. Group profiles are collections of profiles. This allows you to run multiple profiles in one go instead of having to run each profile one after another. This page of SyncBackSE Help runs through the simple procedure you'll need to follow.

Creating a Group Profile

Click the **New** button from the Toolbar (or **New** from the main menu or right-click menu).

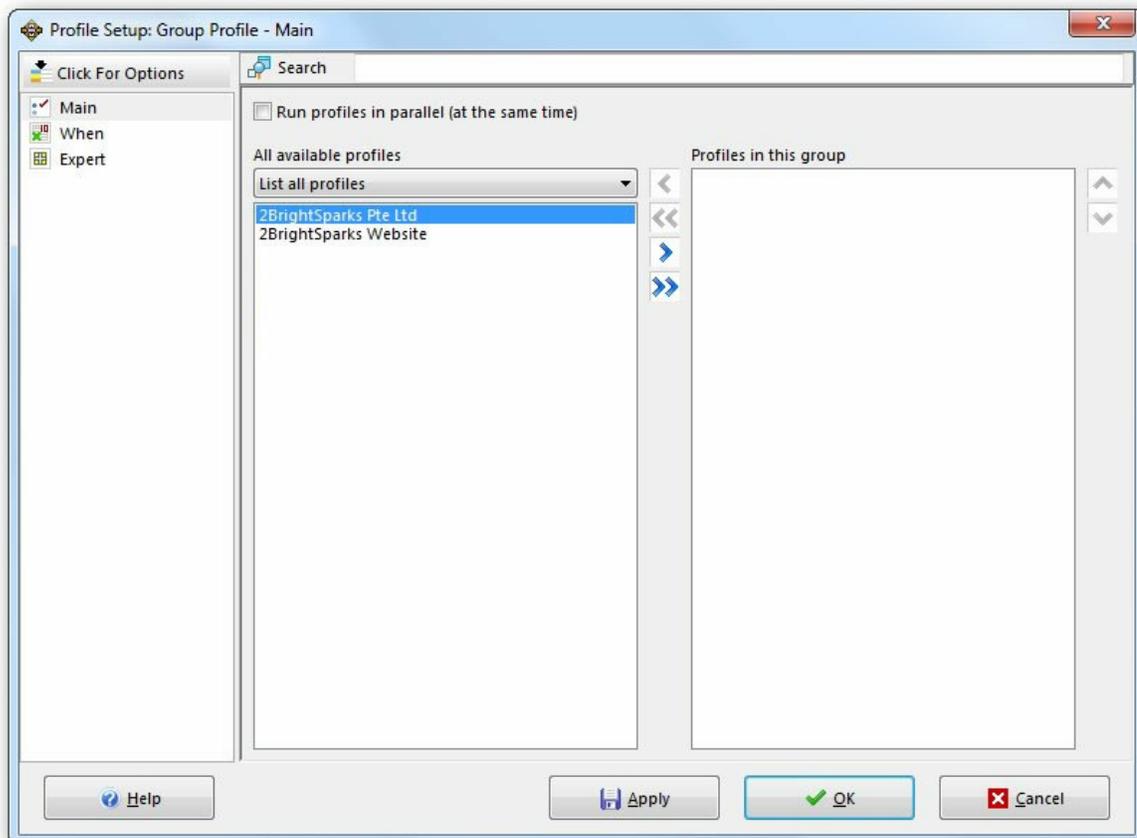
Enter a name for your profile and tick the **This is a group profile** checkbox:



Click the **Next** button. An information window informs you the Profile Setup window will open. You may click the **Do not prompt me again** if you do not wish to be notified of this in the future:



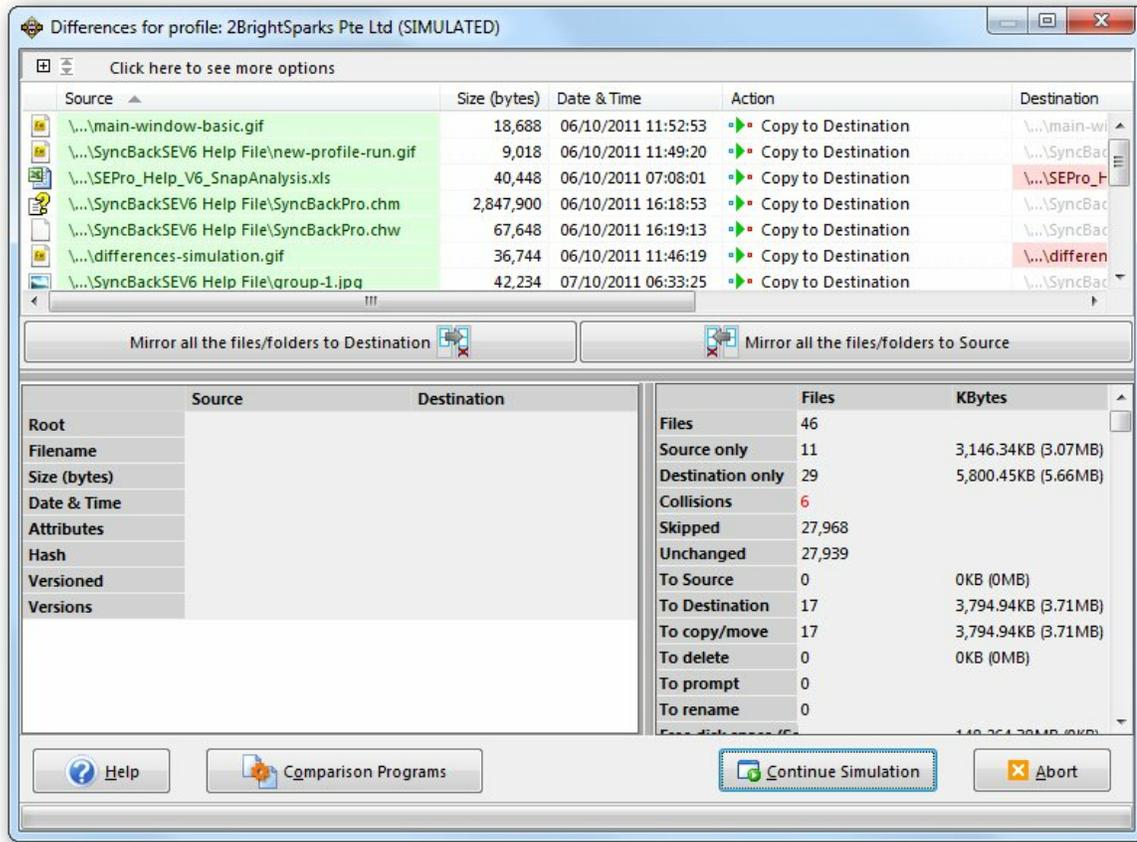
The Group Profile Setup window will open. The example below shows two profiles that can be grouped together. If you have many categories of profiles, there is an drop-down option to filter the categories that are listed. You can add and remove profiles from the group using the <, <<, >, and >> buttons. You can order the profiles in the group by selecting one or more profiles and using the up and down buttons on the right which are currently grayed out. Profiles in the group are executed in the order displayed (first to last):



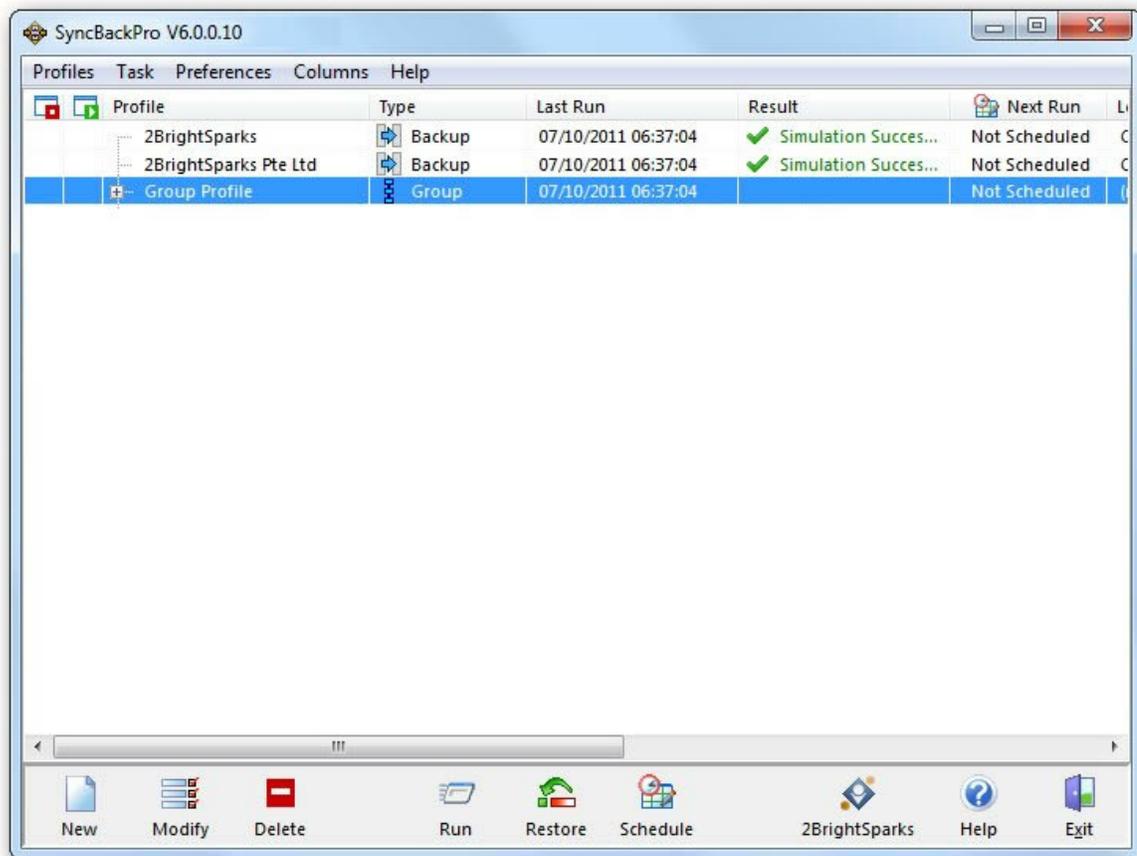
After you click **OK** an informational window will ask whether you wish to perform a

simulated run which provides a report on what will be copied or deleted without actually copying or deleting any files.

In the example below the Simulated Run option has been chosen (recommended), and the Differences Window shows the files that will be copied if an actual run is performed. Click **Continue Simulation**:



The main program window now shows when the Group Profile was last run, and also that the simulation was successful for both profiles that were run:



After you click "OK" the main program window will show the group. Note how the small box to the left of the "Group" profile can be clicked to show what profiles are in the group.

To alter the profiles in a group, select a Group Profile in the main window (you may also select 'Profiles' > 'Modify' from the main menu or right click and select 'Modify'). The 'Group Profile' window will then appear so you may redefine or change your group choices.

Modifying Group Profiles

Select your group profile and click the Modify button. The Profile Setup Window will open. You can change what profiles are associated with a group and the settings for the group. For help on the other settings see the Easy Mode and Export Mode Configuration sections.

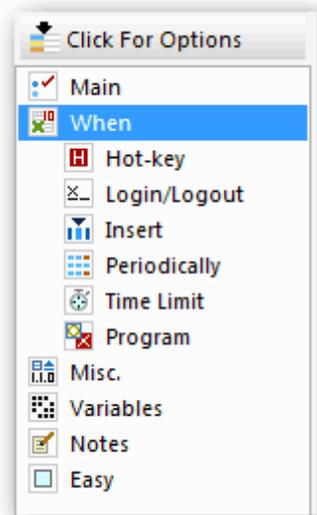
Under '**All available profiles**' you can see the profiles that can be put into the group. There is also a drop-down list so that you can filter the list to include all profiles, non-group profiles, or group profiles. You can add and remove profiles using the horizontal arrow buttons between the two lists, and can order the profiles using the up and down arrow buttons on the right of the window.

Run profiles in parallel: By default, the profiles in a group are run in the order you provide, from first to last. The next profile is started once the previous one finishes. You can choose to have all the profiles in a group run at the same time by enabling this option. It is not

recommended you select this option because it will probably increase the time taken to run the profiles and overload your computer, drive, network, memory, etc. Note that if your group contains other groups then the group will be run serially and not in parallel.

Expert Options

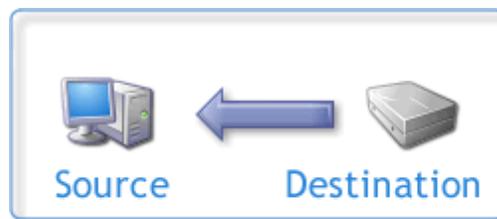
Additional settings are available when choosing the "Expert" options from the Options drop down menu or by going to the Expert settings page:



3.1.6 Restoring a Backup

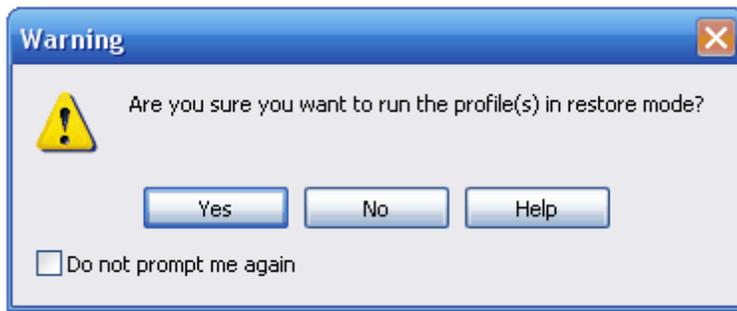
Backup Restoration

Restoring a backup in SyncBackSE is a simple matter of selecting a profile and clicking the 'Restore' button located in the lower menu (you may also select 'Restore' from the main menu under 'Task', or 'Restore' from the right-click menu). Note that you cannot run a restore with a Synchronize profile.



When a profile is run as a Restore the following dialog appears (unless you kept the **SHIFT**

key pressed while clicking the Restore button, or you have previously switched off this dialog):



Click **Yes** to continue. The **Restore Wizard** then analyzes your profile and will ask a set of questions related to how you want the files restored. Note that some of the question may not appear if your profile is configured in such a way that the question is not required or relevant. Once you have gone through all the steps (or click the **Restore Now** button to skip all the steps) the restore process will begin.

After these steps, and the source/left and destination/right are scanned, the Differences window will appear. This gives you another opportunity to abort and also examine and optionally change which files will be restored and how they are restored.



If your backups are incremental or differential, i.e. you are using variables in your destination, then you'll need to think about what order to restore in.

For a differential backup you would restore the full backup first, followed by the newest differential backup. However, if your last backup run was a full backup then you only need to restore from the full backup.

For an incremental backup you would restore the full backup first, followed by oldest incremental backup to the newest incremental backup. However, if your last backup run was a full backup then you only need to restore from the full backup. For example, if your backups are stored at D:\My Backup\%DAYOFWEEK%, and you do a full backup on Monday, and today is Wednesday (and today's backup has already run) then you'd restore Monday first (D:\My Backup\1\), then Tuesday, and finally Wednesday. This makes sure you have the newest files restored as they will overwrite older files already restored. It also makes sure you restore all your files because the incremental backup directories will only contain some of your files.

Last Run

If the last run of the profile was not a success then the following question will be asked:



Click **Yes** to continue to the next step, or click **No** (or **Abort**) to abort the restore.

Moving

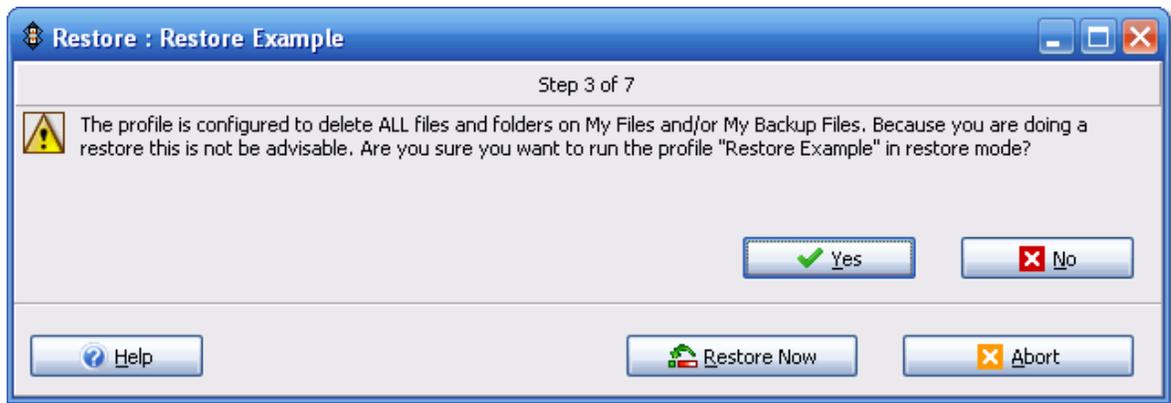
If the profile, or one of the profiles in the group, is configured to move files then the following question will be asked:



Obviously when restoring you would not want your backup files moved as you would lose your backup files. It would be better to copy them so after the restore you still have the backup files. It is recommended that you abort the restore and reconfigure your profile to not move files. You can then restore and afterwards change the profile configuration back to its original settings.

Delete All Files

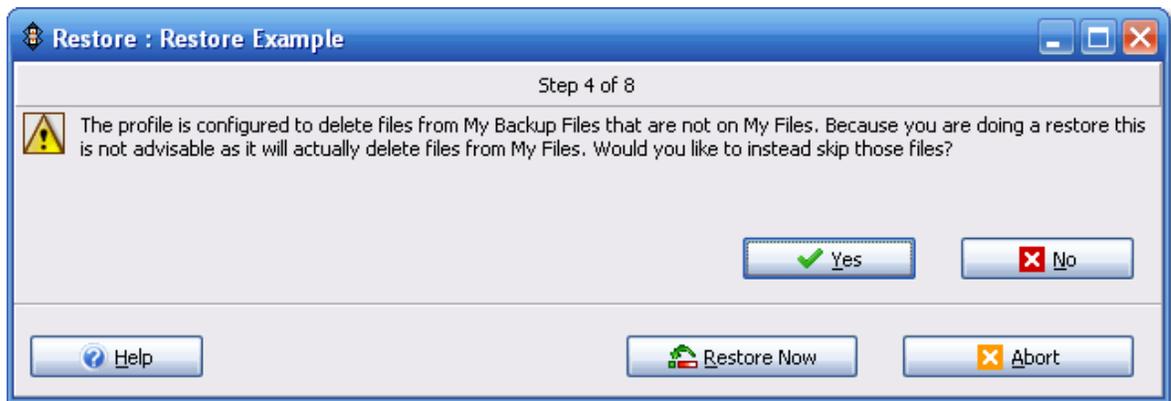
If the profile, or one of the profiles in the group, is configured to delete all the files in the source/left and/or destination/right then the following question will be asked:



Obviously when restoring you would not want your backup or original files deleted. It is recommended that you abort the restore and reconfigure your profile to not delete all the files. You can then restore and afterwards change the profile configuration back to its original settings.

Delete Files

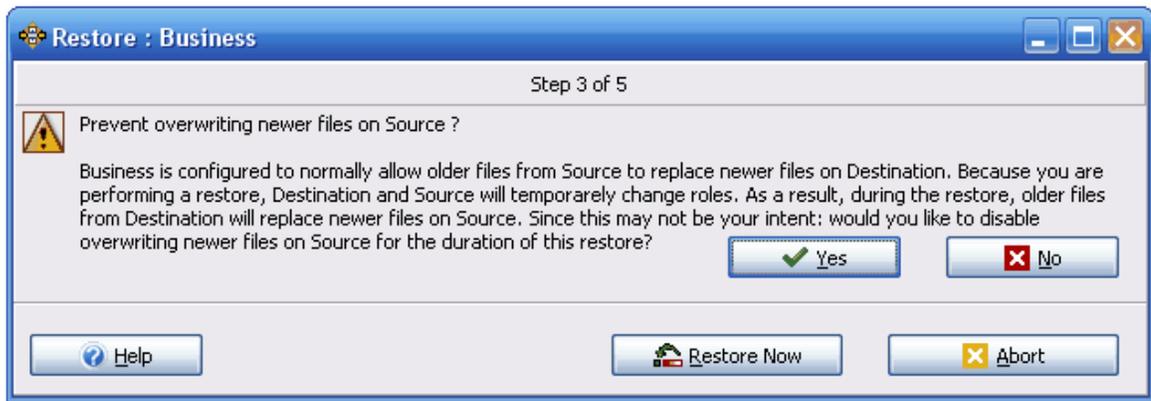
If the profile, or one of the profiles in the group, is configured to delete files that are only in the destination/right then the following question will be asked:



When restoring you would probably not want files only in the source/left to be deleted (as this is a restore the source/left becomes the destination). It's recommended you click **Yes** to skip the files instead of deleting them.

Newer Files

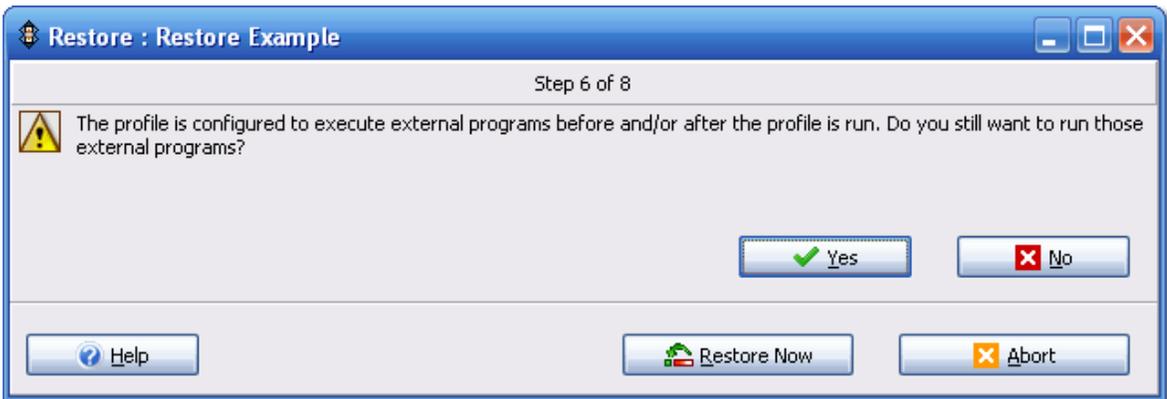
If the profile, or one of the profiles in the group, is not configured to keep newer files then the following question will be asked:



When restoring you probably do not want to replace any newer original files with older backup files. It's recommended you click **Yes** to skip the older files.

Before/After Programs

If the profile, or one of the profiles in the group, is configured to run a program before and/or after the profile then the following question will be asked:



When restoring you may not wish to run those programs. Click **Yes** to still run the programs, or **No** to not have them run.

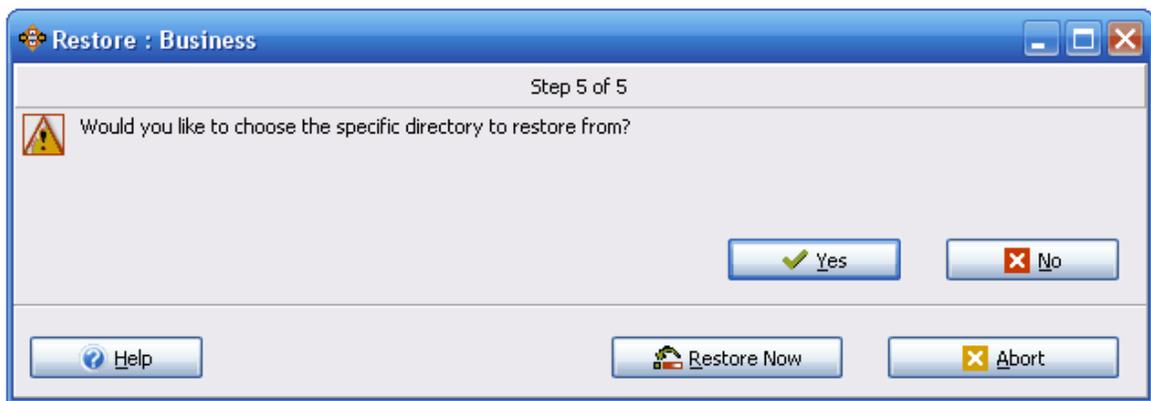
Restore To

You are given the opportunity to restore to a different folder. This is especially important if your source/left folder (where the files will be restored to) contains variables:



Restore From

You are given the opportunity to restore from a different folder. This is especially important if your destination/right folder (where the files will be restored from) contains variables:



Reverse Group

In some cases you may want the group to run in reverse. For example, if you have a group that does a backup to a Zip file then copies the Zip file to an FTP server you probably need to run this in reverse, i.e. first retrieve the Zip file then unzip.

Replace Prompt

If you are restoring from a spanned/split Zip file and the profile is configured to prompt if a file cannot be replaced because it is busy, then you are asked if you'd instead like to replace those files on reboot. The reason is because no prompt can be made in this situation.

3.1.7 Creating a Schedule

Scheduling SyncBackSE

SyncBackSE interfaces with the Windows Task Scheduler to allow you to run profiles automatically at certain times, e.g. run a backup profile every day at 5am. On Windows XP you can access the task scheduler via the Start menu (All Programs > Accessories > System Tools > Scheduled Tasks). On Windows Vista and newer you can access the task scheduler via the Start menu (All Programs > Accessories > System Tools > Task Scheduler).

For an overview of scheduling and background tasks go to Automating SyncBackSE.

Creating a Schedule

To create a schedule for a profile, select the profile on the main screen and click the Schedule button (you can also click the Schedule button when modifying a profile). If there is no schedule for the profile, a dialog box will appear (if you have a schedule already then a different window will appear - see Modifying a Schedule below).



Click Yes.

On Windows, by default, you are not allowed to use blank/empty passwords when scheduling tasks. If this is the case on your installation of Windows then a prompt will appear asking you if you would like the restriction removed (**no prompt is given if the restriction has already been removed**):



Click Yes to remove the restriction. If your Windows login password is blank/empty, and this restriction is not removed, then your scheduled task will fail to run.

Users on Windows Vista, Windows 7, and newer

The following windows will appear if you are using Windows Vista, Windows 7, or newer. For Windows XP see the section Users on Windows XP below.

You can choose to run a profile daily, weekly, or monthly (for other repeating type you'll need to edit the schedule directly using the Task Scheduler). You can also choose how often the profile should be run (repeating).

Screenshot of the Windows Task Scheduler "Schedule" dialog box. The dialog is titled "Schedule" and has a close button. It asks "When do you want the profile to run?" with radio buttons for "Daily" (selected), "Weekly", and "Monthly". Below this are tabs for "Daily", "Weekly", and "Monthly". The "Daily" tab is active, showing a "Start" date of 6/18/2009 and a time of 9:00:00 AM. The "Recur every" field is set to 1 days. The "Repeating" section has checkboxes for "Run this profile every" (checked), "for a duration of" (checked), and "Indefinitely" (checked). The "Security" section has checkboxes for "Run only when user is logged on" (unchecked), "Run whether user is logged on or not" (checked), and "Do not store password. The profile will only have access to local resources." (checked). At the bottom are "OK" and "Cancel" buttons.

- **Run only when user is logged on:** If this checkbox is ticked then the profile will only run at the scheduled times if you are logged into Windows. The benefit of this method is that you do not need to enter your Windows login password. The drawback is that you must be logged on for the profile to run.
- **Run whether user is logged on or not:** If this checkbox is ticked then the profile will run even if you are not logged in. You will need to enter your login password, unless the "Do not store password" option is ticked.

- **Do not store password:** If you are performing a backup/sync with an internal drive, or an external drive, then you can tick this checkbox. In this case you do not need to enter your Windows login password. However, if you are performing a backup/sync with a network drive, or are accessing the network, you should not use this option otherwise the schedule may fail to run. Also, if this option is enabled you will not have access to NTFS encrypted files.

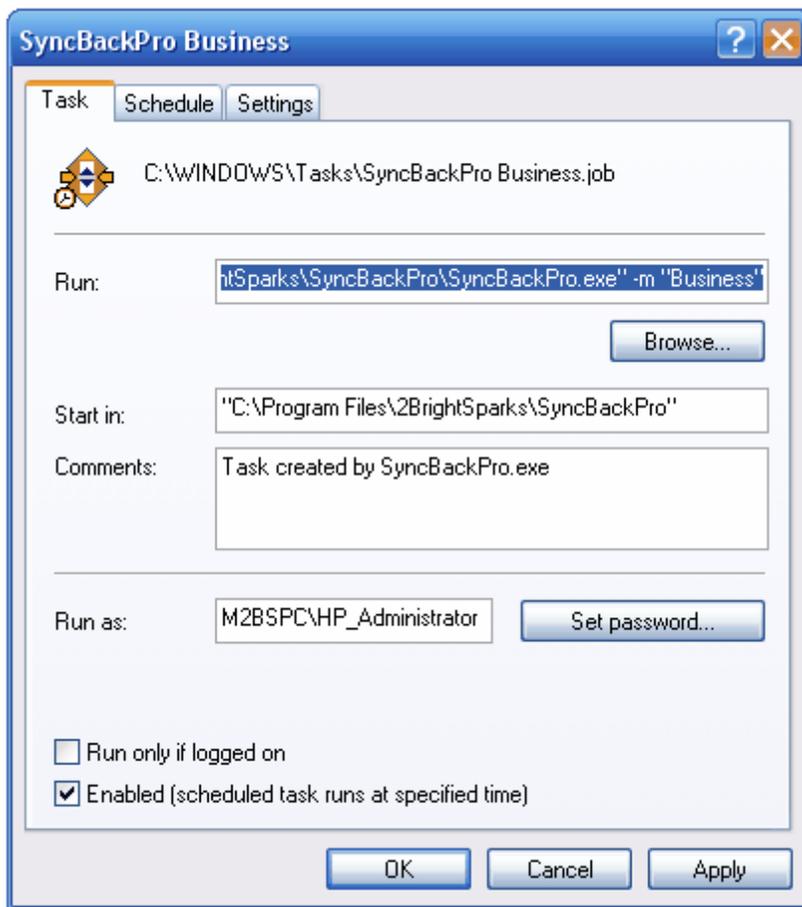
Note that if you have no password then SyncBackSE may not prompt you for your Windows login password (as it will test to see if your password is blank).

Users on Windows XP

A dialog will then appear to remind you that you must set your login password for the schedule even if you do not have a password. This dialog box is a reminder to tell you to enter your password on the next screen (otherwise SyncBackSE will not run at its scheduled time). Click OK.



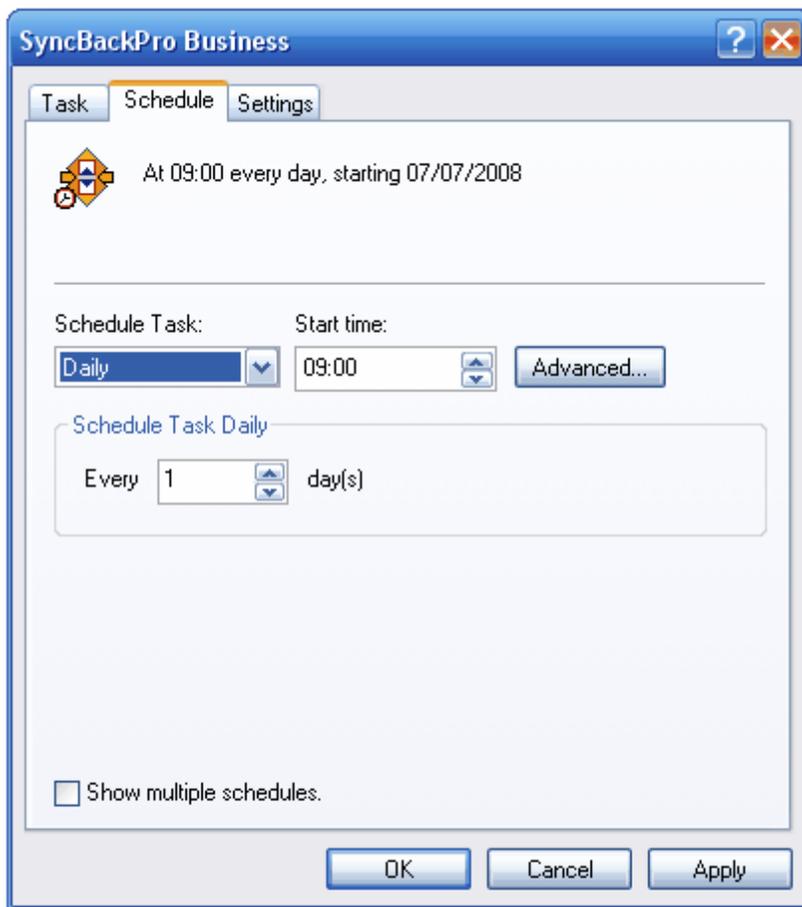
Click "OK".



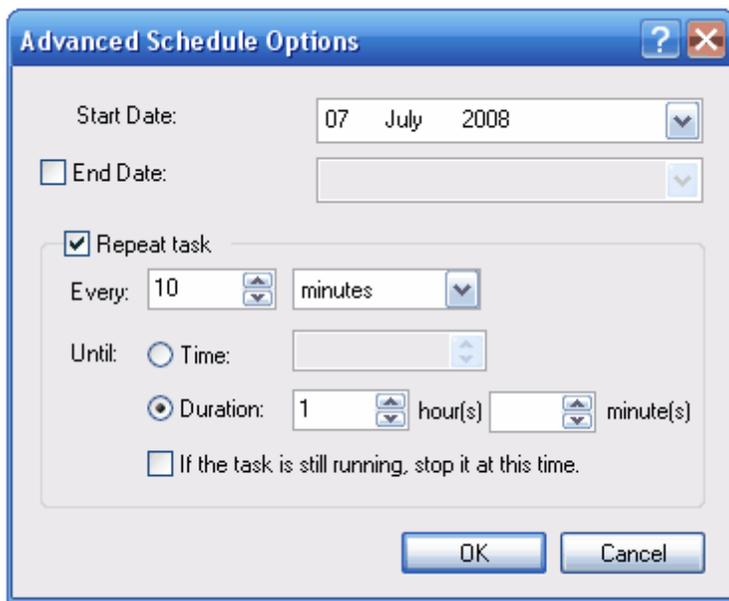
 Note that by default a profile is run minimized (hidden) and will not prompt you.

If there is a 'Set password' button then click it. You **must** now enter your Windows login password twice, and then click OK. If you have no Windows login password then do not enter anything and click OK. You are then returned to the previous dialog box.

Click the Schedule tab.



By default, SyncBackSE will create a schedule to run the profile every day at 9am. You can change the schedule on this screen. There is also an "Advanced" button which provide additional scheduling settings:



See the Windows help file for more information.

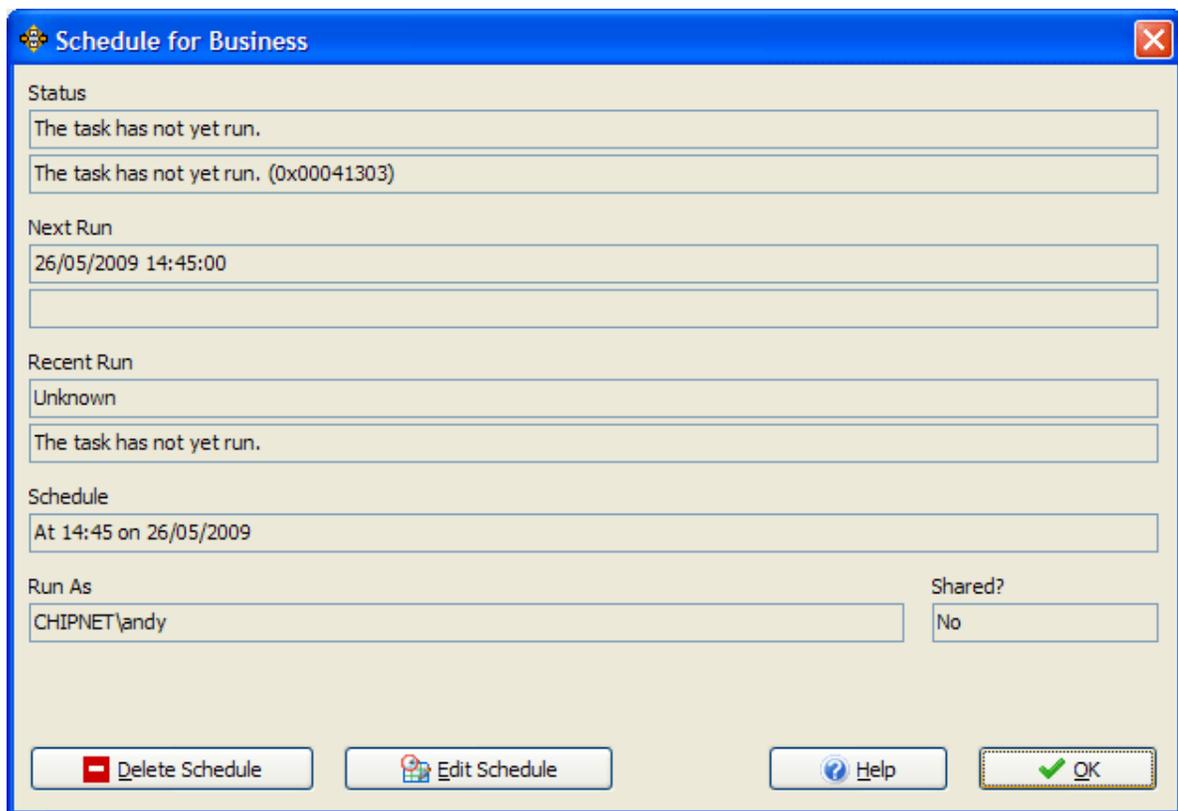
Click the Settings tab. See the Windows help file for more information on this screen:



See the Important Scheduling Information section below for more information.

Modify a schedule

If you already have a schedule for a profile, when you click the Schedule button the following window will appear.



This window displays information on the schedule for the profile, e.g. what its schedule is and the current status. You can delete or edit the schedule by clicking Edit Schedule.

Important Scheduling Information

There are some important points to remember about the Windows scheduler:

- If you are using power saving features in Windows, e.g. your computer goes to stand-by or hibernate after a certain period of inactivity, then you must enable the 'Wake the computer to run this task' option in the scheduler. This will wake your computer to run the task at the appointed time.
- If you change your Windows login password then you must remember to change the password for your scheduled tasks. If you are using Windows Vista, Windows 2008, Windows 7, or newer, then you only need to update the password for one schedule. Windows will automatically update the password for all other scheduled tasks. If you are using XP, or 2003, then you need to update the password for every schedule.
- SyncBackSE can accept a number of Command Line Parameters.
- To have SyncBackSE start with Windows you need to select **Preferences > Options** from the main menu, then tick the option **Start With Windows**.

Common Problems

There are a number of possible reasons why the scheduled task may not run:

- Check to see what error message is returned from the Windows Task Scheduler. On Windows XP you can do this by **Start > Control Panel > Scheduled Tasks**. You can now see a list of all scheduled tasks on your PC. Scroll to the right and look at the Status column. This will contain the error message. On Windows Vista or newer you can do this by **Start > All Programs > Accessories > System Tools > Task Scheduler**. Expand the tree on the left so it is **Task Scheduler (Local) > Task Scheduler Library > 2BrightSparks > SyncBack > [your username]**.
- You must use your Windows login password.
- You are using the wrong username and/or password. You must use your Windows login password.
- The scheduled task may not be set-up correctly to wake the computer if it is hibernating or in standby mode.
- The scheduler may be stopped or not installed. See your Windows documentation for details on how to start or install it. For Windows XP you can start the task scheduler by: **Start > Control Panel > Scheduled Tasks** then selecting **Advanced > Start Using Task Scheduler** from the main menu.
- The scheduler may be paused. For Windows XP you can resume the task scheduler by: **Start > Control Panel > Scheduled Tasks** then selecting **Advanced > Continue Task Scheduler** from the main menu.
- Profiles are user specific, they are not visible to every user on the computer. This means when you create a profile under a Windows username, and logout and login as a different Windows user, then you will not see the profile created as the other user. When scheduling a profile make sure your scheduled task is being run as the user who created the profile (this is the default when new schedules are created).
- If you are a member of a domain check that the correct username is being used for the scheduled task. By default your local (machine) username may be used, but it may be that you must use your domain username (domain\username).
- You should also make sure the user account has the necessary Windows user rights. To do this, run the Local Security Policy control panel applet (in the Administrative Tools section of the control panel). If you are using the home version of Windows then you may not have access to the Local Security Policy control panel applet (Microsoft have removed the feature from home versions of Windows).

Make sure that the user account has the following user rights:

- Act as part of the operating system
- Log on as a batch job
- Log on as a service

Make sure the user account is not listed in the following user rights:

- Deny logon as a batch job
- Deny logon as a service

Alternatives

Note that you are not forced to use the Windows Task Scheduler. You can use 3rd party schedulers instead, but the drawback is that SyncBackSE will not know the task is scheduled (it will still run as that is the job of the Windows scheduler), and the schedule options (to create and delete a schedule) will still use the Windows Task Scheduler. There are a number of free alternatives to the Windows Task Scheduler, e.g. Kirby Alarm.

3.1.8 Preferences Main Menu

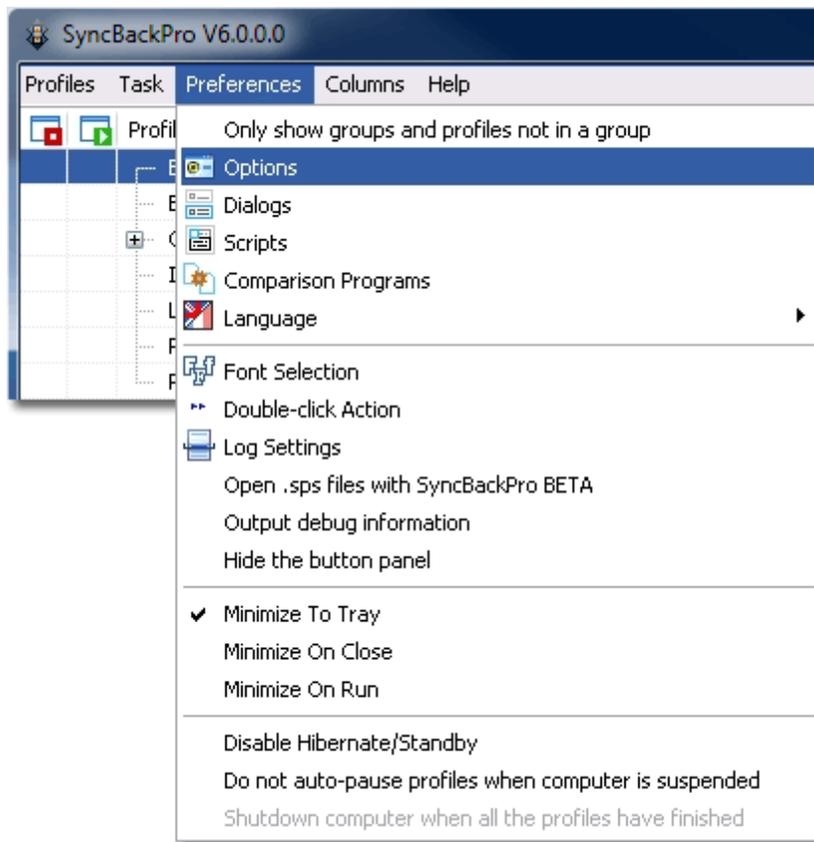
Program Options

Only show groups and profiles not in a group

If unticked then all profiles and groups are listed in the main window. If the option is ticked then the top level list of profiles will not include profiles that are part of a group or groups that are in other groups. The top level list will only show profiles that are not in any group, and groups that are not in any other group. Ticking this option helps unclutter the list.

Options

This opens a window where you can change settings that apply to the entire program. See this page of the help file for more details. The Options window is available from the main menu: 'Preferences' > 'Options':



Dialogs

This opens a window where you can change which dialog boxes appear. See this page of the help file for more details.

Scripts

 Scripting is a Pro only feature. This opens a window where you can extend the abilities of SyncBackSE by using scripts. See this page of the help file for more details.

Comparison Programs

This opens a window where you can change which programs are used in the Differences window to compare the contents of files. See this page of the help file for more details.

Language

The language used can be changed using this menu. Note that only the translated languages installed are listed. By default the language selected during installation is used.

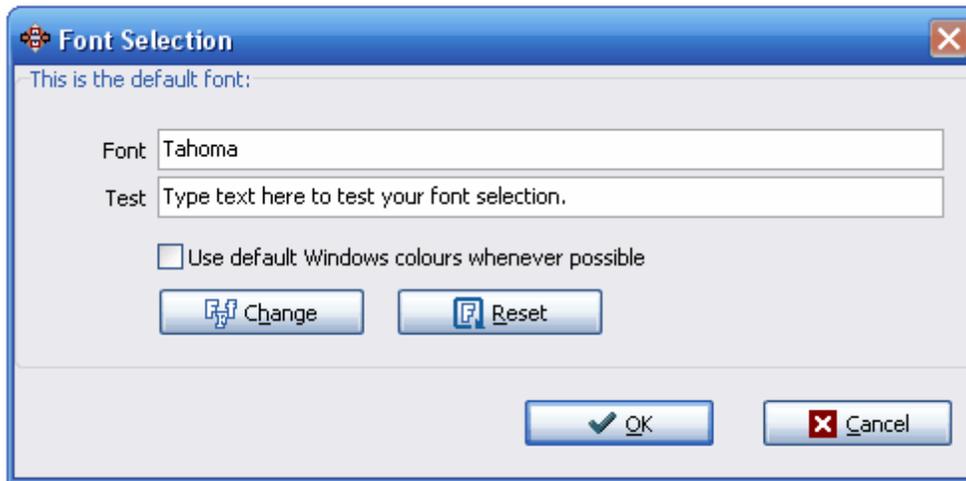
Management Service Settings

 Scripting is a Pro only feature. See the SyncBack Management Service help page for

more details.

Font Selection

You can change the default font used by SyncBackSE. This font is used on all screens. Note that the style, e. g. bold, will be used when displaying filenames, but will not be used for text labels, buttons, etc. The font size is fixed at 8 (9 if you are using Windows Vista). Changing the font (and its Script) is useful when using other languages with SyncBackSE.



- **Use default Windows colours whenever possible:** If you are visually impaired and use a special Windows colour scheme, e.g. High Contrast, then you should enable this option. It will force SyncBackSE to use the default colours whenever possible. Note that you must exit and restart SyncBackSE if you change this option.

Double-click Action

You can select what action SyncBackSE should take when you double-click on a profile.

Log Settings

This opens a window where you can change how profile log files are created. See this page of the help file for more details.

Open .sps files with SyncBackSE

Exported profiles have the filename extension of SPS. If another program has set itself as the default program for opening SPS files, then this menu item will appear allowing you to change it back to SyncBackSE.

Output Debug Information

If this menu item is ticked (enabled) then when profiles are run, or profiles modified, then a debug file is created. This debug file will contain detailed information that allows 2BrightSparks to pin-point any errors or problems with the program (it is used by the

Technical Support Wizard). By default this item should not be ticked. Only tick this menu item when instructed to do so by 2BrightSparks technical support. When enabled it can significantly reduce performance.

Hide The Button Panel

If this menu item is ticked (enabled) then the button panel is hidden. To change where the button panel is displayed then right-click on it and choose from the pop-up menu.

Minimize To Tray

If this item is ticked then SyncBackSE will minimize to the system tray (also called the Notification area in Windows XP) instead of to the task bar.

Minimize On Close

If this item is ticked then SyncBackSE will minimize instead of closing (exiting) when the Close button (the X button in the applications title bar) is clicked. In this case you must click the Exit button to close SyncBackSE.

Minimize On Run

If this item is ticked SyncBackSE will minimize automatically when a profile is run.

Disable Hibernate/Standby

If this item is ticked then SyncBackSE will stop your computer from hibernating or going into standby power saving modes if any profiles are running. If your computer is using batteries this option is disabled and cannot be used. Note that if your computer is running on batteries, and your batteries are low, SyncBackSE will automatically pause all running profiles and display a warning message. SyncBackSE can put your computer into hibernate or standby mode after profiles are run. For more information, see the page Command Line Parameters.

Do not auto-pause profiles when computer is suspended

If this item is ticked then SyncBackSE will not pause all the profiles when the computer is suspended. Otherwise if the computer is suspended, e.g. you close the lid on a notebook computer, then SyncBackSE will pause all the profiles immediately. After the computer is woken it will automatically resume the profiles that were suspended.

Shutdown computer when all the profiles have finished

If this item is ticked then SyncBackSE will shutdown the computer as soon as all the profiles have completed. This can be useful if you need to leave the office and a long running profile is currently running.

3.1.9 Profiles Pop-Up Menu

Profiles Pop-Up Menu

If you right-click on a profile then a pop-up menu appears with a number of menu items. Many of these functions can be used with multiple-profiles. There are a number of ways to select multiple profiles, for example:

- Press **Ctrl-A** to select all profiles (to unselect all press **Ctrl-U**).
- While holding down the **Ctrl** key click on the profiles you want selected.
- You can select a range of profiles by clicking on a profile, then while holding down the **Shift** key, click another profile.

Some of the menu items may not be available due to limitations. For example, some functions cannot be used with groups (e.g. Open Left/Source) or they can only be used with a single profile and not multiple profiles (e.g. Modify). Also, many of the menu items will also show the shortcut key. For example, you can run profiles by selecting them and using the shortcut key **Ctrl-R**

- **Run (Ctrl-R):** Runs the selected profiles.
- **Simulated (Ctrl-S):** Runs the selected profiles in **simulation** mode.
- **Restore:** Runs the selected profiles in restore mode.
- **Simulated Restore:** Runs the selected profiles in **simulated restore** mode.
- **Open Left/Source:** Using Windows Explorer, it will open to the source/left folder on the selected profiles.
- **Open Right/Destination:** Using Windows Explorer, it will open to the destination/right folder on the selected profiles. Note that some destinations cannot be opened, e.g. if you are performing a backup to the cloud then it will not do anything. For FTP, it will use the FTP URL `ftp://[user ID:password@]<host name>[:port]/[path/]`
- **Modify (Ctrl-M):** Modifies the selected profile.
- **Schedule:** Creates or modifies the profiles schedule.
- **Delete (Del):** Deletes the selected profiles.
- **Rename (F2):** Renames the selected profiles.
- **Copy (Ctrl-C):** Makes a copy of the selected profile.
- **Export Profile:** Exports the selected profiles. Exporting profiles lets you import them into other installations of SyncBackSE and is also a way of making a backup of a profile. In the Global Settings window, you can configure SyncBackSE to automatically make backups of all your profiles when SyncBackSE exits.

- **Upload Profile to SBM Service:** The selected profile is uploaded to the SBM Service. See the SyncBack Management Service help page for more details.
- **Enable:** Enables a previously disabled profile.
- **Disable:** Disables the selected profiles. A disabled profile cannot be run by any method (including scheduled, manually, etc.). You cannot disable group profiles.
- **Clear groups result:** The select groups will have their last **Result** cleared.
- **Background Colour:** This allows you to change the background colour of a profile. To reset/clear a profiles background colour select **Clear**. Background colours are purely cosmetic and have no affect on the profile itself.
- **Refresh (F5):** The display is refreshed so the information display is current. This can be useful if you have profiles running in another instance of SyncBackSE, e.g. via a schedule, and the display isn't refreshing automatically.
- **Select all (Ctrl-A):** All the **visible** profiles are selected. If you have an unexpanded group then those profiles in the group will not be selected.
- **Unselect all (Ctrl-U):** The current selections are de-selected.
- **View Log:** The log for the selected profiles is displayed. Note that you may have a history of logs for a profile so you can choose which log to display using the sub-menu. To display the latest log you can press **Ctrl-L**.
- **View Debug Log:** The debug log for the selected profiles is displayed. Debug logs are for internal use.
- **Pause:** The selected profiles are paused. To resume them select **Resume**.
- **Resume:** If paused, the selected profiles are resumed.
- **Stop:** The selected profiles are stopped. If the profile is being run as part of a group then the entire group is stopped.
- **Stop profile only, not group (Ctrl-O):** The selected profiles are stopped, but not the group it is running as a part of.

3.1.10 Columns Main Menu

Program Options

The main window has a number of columns that provide information on the profiles. You can choose which columns to display via the **Columns** main menu.

- **Stop:** When a profile is running you can click on the icon shown in the Stop column to

stop the profile. If you stop a profile that is being run as part of a group then all the other profiles in the group will also be stopped.

- **Play/Pause:** When a profile is running you can click on the icon shown in the Stop column to pause/continue the profile. Note that when a group profile is run (and it is not set to run the profiles in parallel) it will run the profiles in the group one after another. Those in the group waiting to run are paused until the previous profile in the group has finished. SyncBackSE will then automatically continue the profile when it is its turn to run.
- **Type:** The type of profile, e.g. Backup, Fast Backup, Smart Sync, etc.
- **Last Run:** When the profile was last run. If it has not been run yet then it will show **Never**.
- **Result:** The result of the last profile run. Note that if any file fails to be copied, moved, or deleted, for any reason at all, then the profile run is considered a failure. Success is when there were no errors.
- **Next Run:** When the profile will next be run. You can run the profile manually at any time (unless it is disabled or already running). The next run date & time takes into account schedules and periodic runs. If the profile is in a group then it will also take that into account. For example, a profile may be set to run at 2pm but could be in a group that is set to run at 1pm. In the main window it will show the appropriate time depending on whether the profile is shown in the group or not.
- **Source / Left:** The source/left folder.
- **Destination / Right:** The destination/right folder.
- **Background:** If the profile is set to run in the background, e.g. every 30 minutes or whenever there are changes in the source or destination, then this column will contain the details.
- **Progress:** When a profile is running this column shows the same information shown in the progress bar.
- **Last Successful Run:** The date & time the profile was last run without error. If it has not yet run without any errors then it will show **Never**.
- **Last Scheduled Run Error:** If the Windows Task Scheduler had a problem running SyncBackSE, e.g. password is incorrect, then this column will show the reason why SyncBackSE could not be started to run the profile at the scheduled time. If the profile is not scheduled, or there was no error the last time it was run at the appointed schedule, then this column will be empty.
-  With the Pro version you can add your own columns by using scripting.

3.1.11 Global Settings

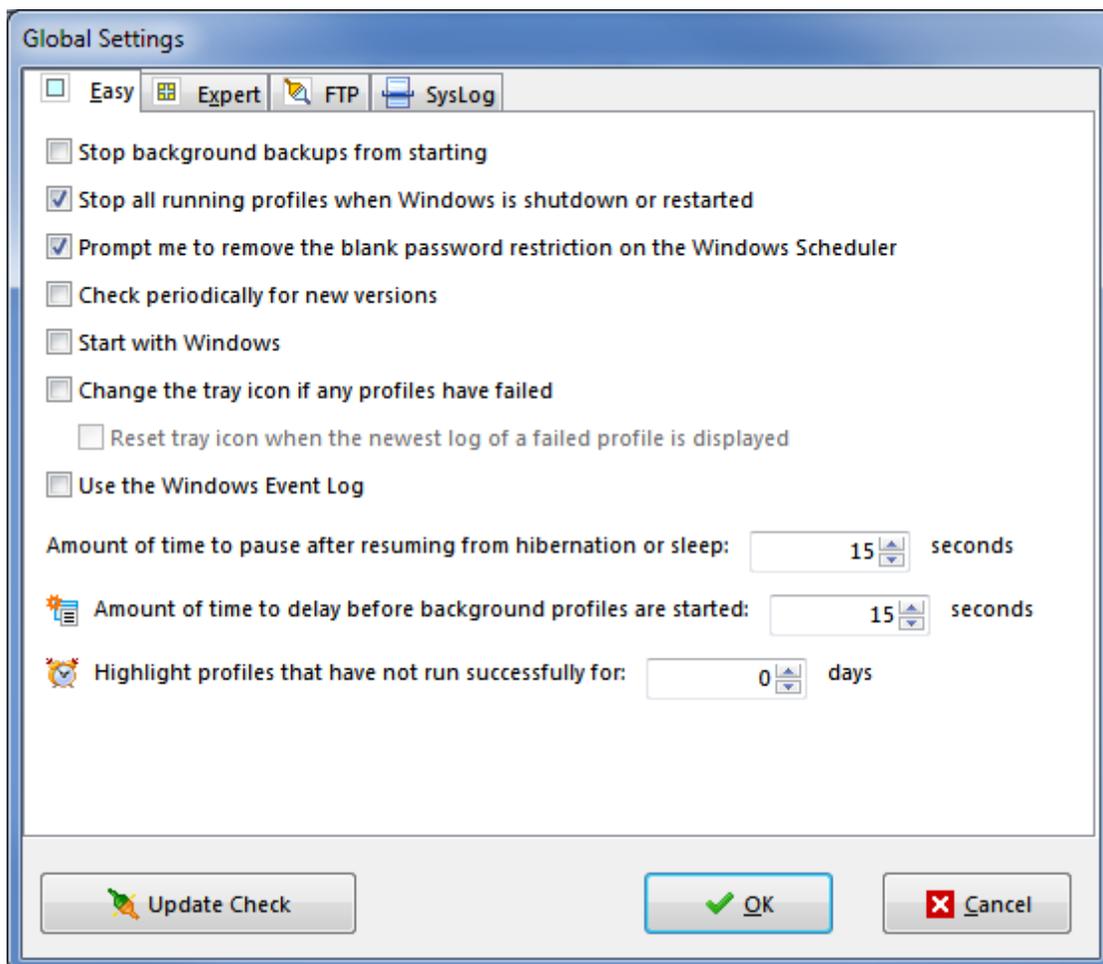
Program Options: Global Settings

Global Settings - Easy

- **Stop background backups from starting:** If ticked, any profiles set to run in the background (periodically, or via a file/folder change) will not be started. This does not affect scheduled tasks.
- **Stop all running profiles when Windows is shutdown or restarted:** If enabled then when you shutdown or restart Windows then all running profiles will be stopped.
- **Prompt me to remove the blank password restriction on the Windows Scheduler:** By default you cannot use a blank/empty password when scheduling tasks. If this item is ticked then whenever you schedule a profile, and you are using Windows with this restriction, you will be prompted to ask you if you'd like SyncBackSE to remove that restriction. Note that you will not be prompted if the restriction has already been removed. If you do not have the access rights in Windows to remove the blank password restriction then the checkbox is disabled.
- **Check periodically for new versions:** If ticked then SyncBackSE will check, every 30 days, if a new version of SyncBackSE is available. See also the **Update Check** button (to check for a new version immediately) and also the **Help -> Update Check** main menu item. Internet access is required to check for new versions.
- **Start with Windows:** If ticked, SyncBackSE will start after Windows is rebooted. It will start minimized. If you have profiles set to run periodically (not via a schedule), or to run whenever there are changes, it is recommended you use this option.
- **Change the tray icon if any profiles have failed:** If ticked, and any profiles fail when run, then the icon shown in the system tray will change to indicate this. To reset the icon, e.g. you have reviewed the failed profile(s) and are aware of the problem, then right-click on the SyncBackSE tray icon and select **Reset tray icon**. The tray icon will show an orange cross if background backups are being stopped from starting (see option above) and a red cross if background backups can start.
 - **Reset tray icon when the newest log of a failed profile is displayed:** If ticked then the tray icon is automatically reset when the newest log of a profile (that has failed) is displayed. Viewing older logs, or logs of profiles that haven't failed on their last run, will not reset the tray icon.
- **Use the Windows Event Log:** If ticked then errors are recorded in the Windows Event Log. These are not profile errors but errors related to using the program when the user cannot be prompted. They are typically used by technical support staff to help debug problems with the software.
- **Amount of time to pause after resuming from hibernation or sleep:** If the computer is put into sleep or hibernate mode, then resumed, this is the number of seconds that SyncBackSE will pause (sleep) before it does anything. This gives Windows time to re-

establish network connections etc.

- **Amount of time to delay before background profiles are started...:** When SyncBackSE is started it will wait this amount of time before the first background profile is started. This gives you time to disable background tasks or to exit SyncBackSE before they start.
- **Highlight profiles that have not run successfully for...:** If the value is above zero then any profiles that have not run successfully for that number of days will have an alarm clock graphic placed next to their name in the main window. This lets you clearly see which profiles are not running as expected. Note that disabled profiles are ignored and it does not apply to group profiles.



Global Settings - Expert

- **Put new, imported, and copied profile settings files into...:** This setting defines where to put the settings files for new profiles (existing profiles settings files are not moved). It is recommended that **Automatic** is used as this will store new profiles in the same folder as SyncBackSE when it is run from removable media, or the **C:\Users**

[username]\AppData\Local\2BrightSparks\SyncBack folder when not (if using Windows XP it will be **C:\Documents and Settings\[username]\Local Settings\Application Data\2BrightSparks\SyncBack**). If SyncBackSE is being run from an external drive then you will not be able to choose where to store profiles (they will always be stored in the same folder as the program). If you do not have write access to the folder SyncBackSE is being run from then you cannot choose that option. The roaming application directory is for people using corporate networks where files are typically stored on a server.

- **Backup all profiles when the program exits:** By default a backup is made of all the profiles when SyncBackSE exits. This setting defines where to put those files (they are exported profile files). If you have a large number of profiles then you may wish to disable this option and backup your profiles manually (you can use the **-export** command line parameter to export all profiles).



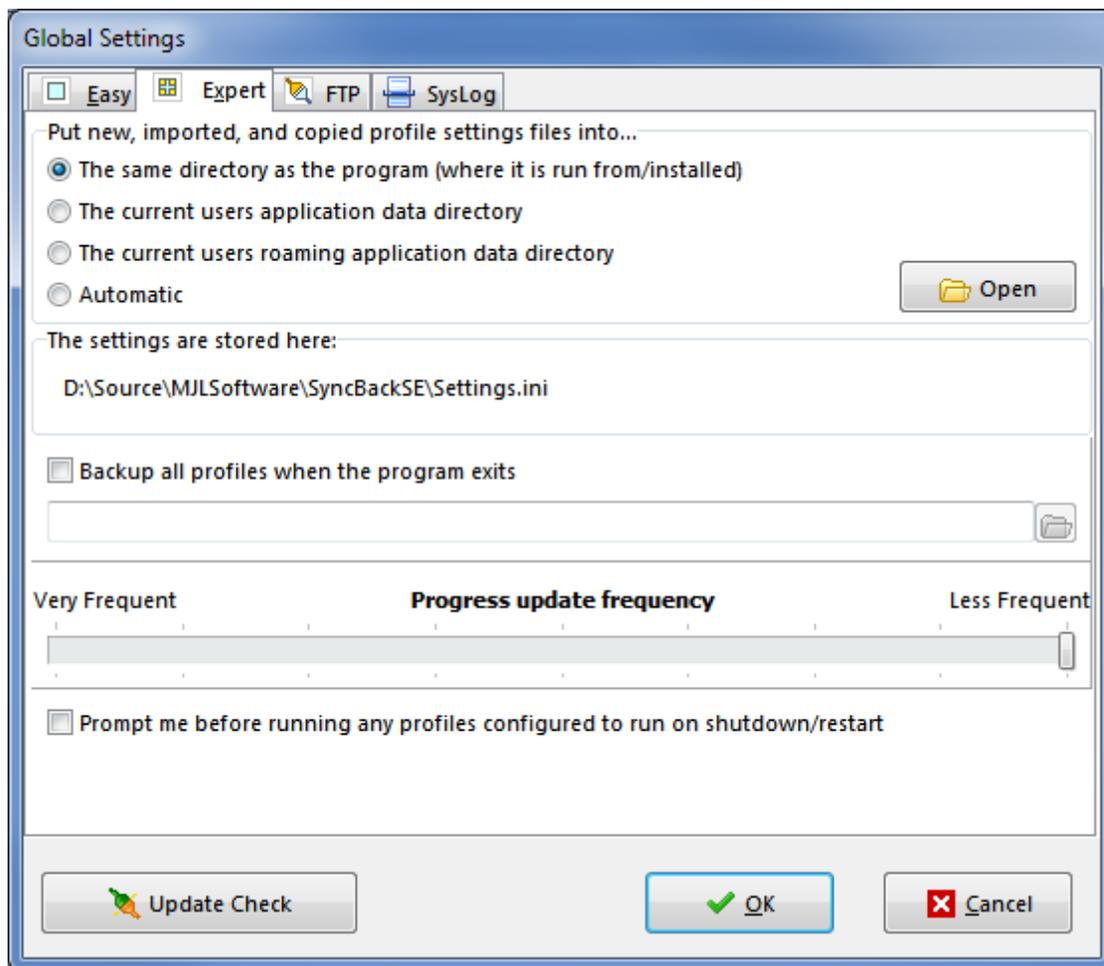
So as not to interfere with the Windows shutdown/restart/logoff process, profiles will not automatically be backed up if SyncBackSE is exiting due to a Windows shutdown/restart/logoff.

- **Progress update frequency:** If you move the slider to the far left then the progress bar will be updated very frequently. If you move it to the far right then they will be updated less frequently. Note that the more frequently the progress bar is updated then the slower the profiles will run. You can reduce the profile run time by reducing the update frequency.
- **Prompt me before running any profiles configured to run on shutdown/restart:** If this option is ticked, and you have profiles that are set to run on Windows shutdown or restart, then you will be prompted before they are run. For example, you may want to shutdown quickly and therefore do not want the profiles to be run.
- **... after profiles have run when configured to run on shutdown/restart:** If you have a profile that is set to run on Windows shutdown or restart, then this setting tells SyncBackSE what to do after that profile has run. Unfortunately Windows does not tell SyncBackSE whether the user wants to shutdown or restart, so this setting is used to define what to do.



This option is not required on Windows Vista, Windows 7, or Windows 2008 R2, as Microsoft have changed the shutdown/logoff process and corrected this flaw. If you are using Vista or newer this option will not be displayed.

- **Update Check:** If this button is clicked a check is made to see if a new version of SyncBackSE is available for download. You can also check via the **Help** main menu. Internet access is required to check for new versions.

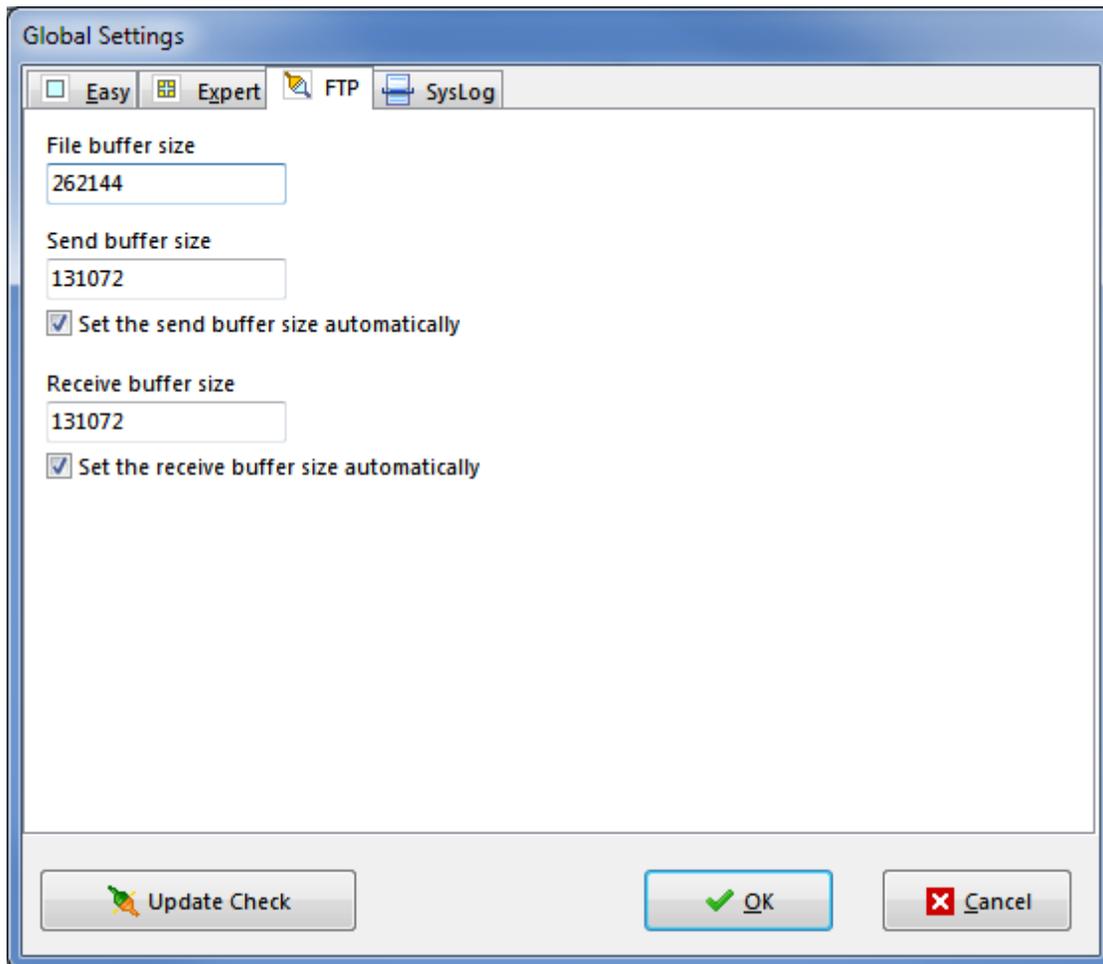


Global Settings - FTP

This page contains settings related to how SyncBackSE communicates with FTP, FTPS, and SFTP servers. It is strongly recommended that you do not change any of these values and that the defaults are used.

- **File buffer size:** This is the size, in bytes, of the buffer used to send and receive files. The default size is 262144 bytes, and the minimum size is 8192 bytes.
- **Send buffer size:** This value directly affects the TCP window size. If the value is set to 0 the default Windows value is used and `setsockopt()` is not called. The default SyncBackSE value is 131072 bytes.
- **Set the send buffer size automatically:** When enabled the send buffer is automatically increased after the socket is connected until it reaches the optimal size. This option is enabled by default.
- **Receive buffer size:** This value directly affects the TCP window size. If the value is set to 0 the default Windows value is used and `setsockopt()` is not called. The default SyncBackSE value is 131072 bytes.

- **Set the receive buffer size automatically:** When enabled the receive buffer is automatically increased after the socket is connected until it reaches the optimal size. On Windows Vista, Windows 2008, Windows 7, or higher, the auto tuning feature provided by Windows is used. In this case the receive buffer size value is ignored. This option is enabled by default.

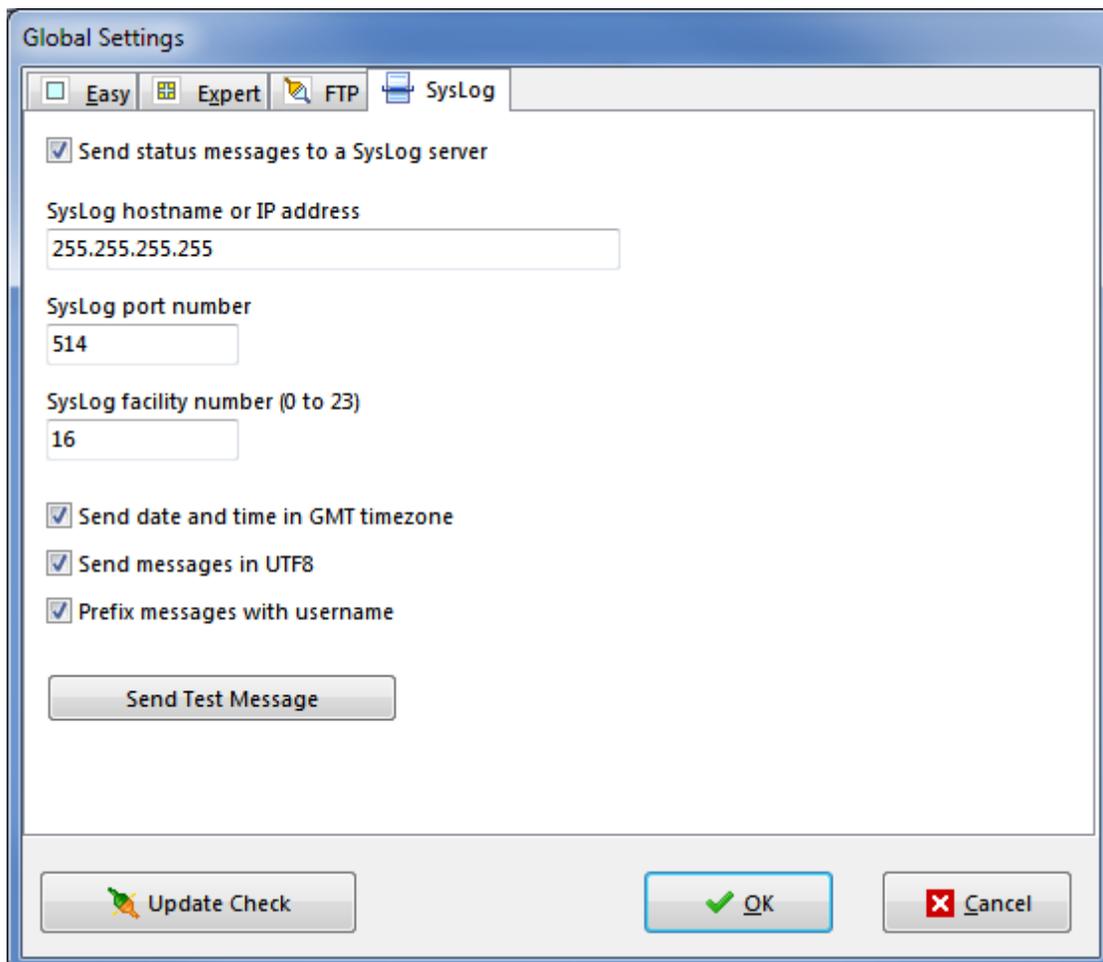


Global Settings - SysLog

 These options are only available in the Pro version. This page contains settings related to how SyncBackSE communicates with a SysLog server (as specified in RFC 3164). When certain tasks are performed, e.g. profiles run, a message will be sent to the SysLog server.

- **Send status messages to a SysLog server:** If enabled, messages will be sent to the SysLog server.
- **SysLog hostname or IP address:** The hostname or I.P. address of the SysLog server. To broadcast use the I.P. address 255.255.255.255

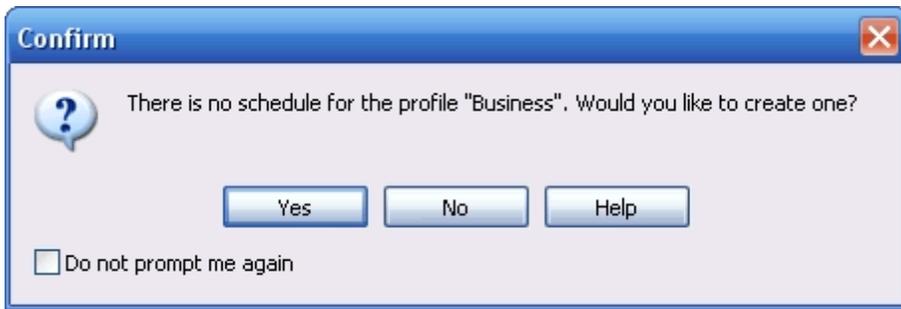
- **SysLog port number:** The port number of the SysLog server. The default is 514
- **SysLog facility number (0 to 23):** The facility number that SyncBackSE should use in messages sent to the SysLog server. The default is 16. Note that usually the numbers 0 to 15 are used for system messages.
- **Send date and time in GMT timezone:** The dates & times of the messages can either be in the local timezone (of the computer running SyncBackSE) or in the GMT/UTC timezone.
- **Send messages in UTF8:** Messages can either be in ASCII or UTF8.
- **Prefix messages with username:** To help with tracking, the messages sent to the SysLog server can optionally be prefix with the Windows username.
- **Send Test Message:** Press to send a test message to the SysLog server.



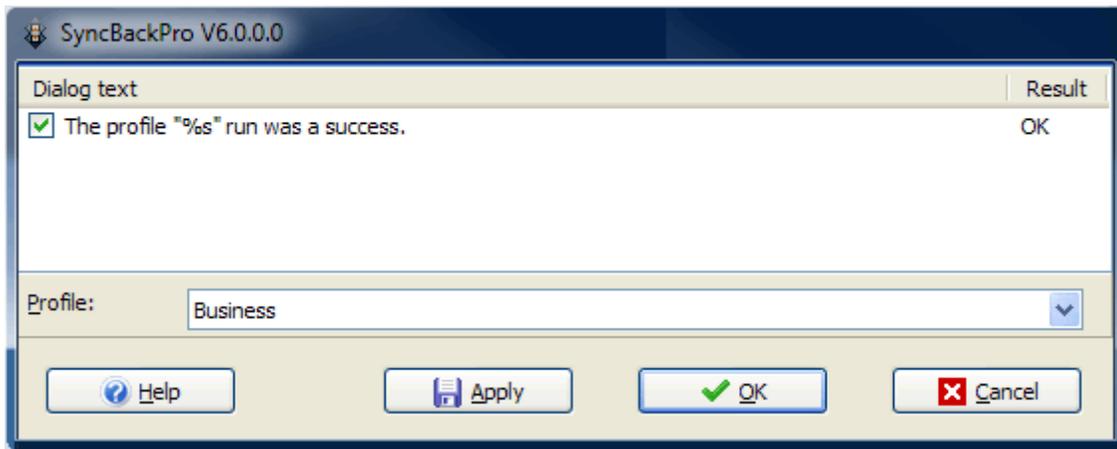
3.1.12 Dialogs

Program Options: Dialog Settings

Many dialogs displayed by SyncBackSE have the “Do not prompt me again” checkbox:



When ticked the dialog will not be displayed again and instead your answer will be used in future. Sometimes, however, you may want the dialog to be displayed again.



Profile: Some dialog boxes are profile specific, meaning whether they are displayed or not depends on the profile being run. This list-box lists all the profiles and ***Program***, which is for dialog boxes that are not for profiles but for the program itself. When the selection is changed the items in **Dialog text** is changed.

Dialog text shows which dialog boxes have been configured not to be displayed again. It shows the text of the dialog box along with the result that is used instead of showing the dialog box, e.g. OK. If you want the dialog box to be displayed simply untick the item. If you want all dialog boxes for a profile to be displayed press **Ctrl-A** (to select all items), right-click, and choose **Untick selected** from the pop-up menu.

3.1.13 Comparison Programs

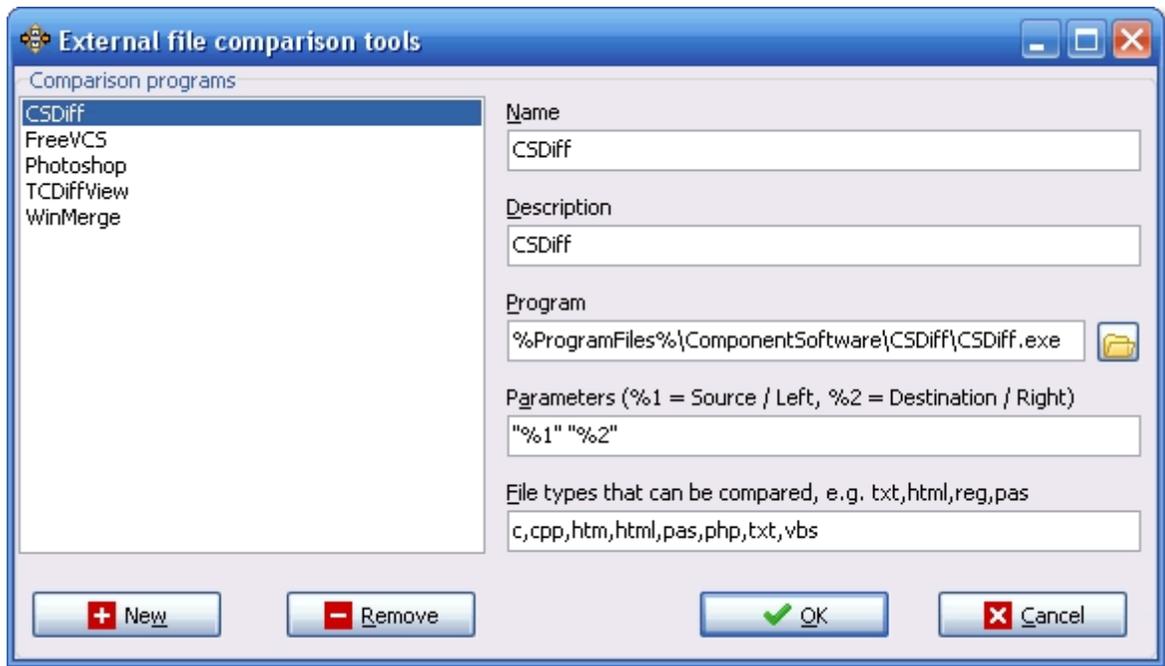
Program Options: Comparison Programs

External File Comparison Tools

When the Differences window is displayed, which lists which files are going to be copied, deleted, moved, etc., then you have the chance to compare two files. For example, you may have changed the same Word document in both the source/left and destination/right. In this case you may want to visually compare the actual Word documents to decide what action you want SyncBackSE to take.

The **External file comparison tools** window lets you tell SyncBackSE which external programs to use to visually compare files of various types.

- **Name:** This is the unique name to give to the entry. You can use whatever name you wish, e.g. **Compare Word Documents**
- **Description:** An optional description of the program, e.g. Compares word documents
- **Program:** This is the filename of the program that will be used to compare the two files. You must enter a complete path and filename. Note that Windows environment variables can be used, e.g. **%ProgramFiles%\WinMerge\WinMerge.exe**. If the program does not exist then the text will be highlighted in red.
- **Parameters:** These are the command line parameters that are used with the comparison program. Check the documentation of the comparison program to see what parameters can be used and how they are used. There are two special strings that you must use in the parameters: **%1** is used to represent the filename of the source/left file, and **%2** is for the destination/right filename. SyncBackSE will automatically replace these two parameters with the correct filenames. Note that you will probably need to wrap the **%1** and **%2** strings with double-quotes, e.g. **-left "%1" -right "%2"**, as Windows treats a space character as a break between parameters.
- **File types that can be compared:** A comma-delimited list of file types that the comparison program can compare. Do not use the initial period of the filename extension, e.g. use **txt,c,h** and not **.txt,.c,.h**



3.1.14 Log Settings

Program Options: Logging Settings

By default the log file, produced when a profile is run, is created in HTML format. By selecting this menu item you can change the log file format, or choose not to create a log file at all.

- **Store all my log files in the following folder:** This is the folder SyncBackSE will store the log files created after every profile run. You can use environment variables, e.g. **%APPDATA%**
- **Use the following filename for my log files:** This is the filename of the log files SyncBackSE created. Windows environment variables and SyncBackSE variables can be used here, including a special variable called **%PAGE%** which is replace with the page number of the log file.



Note that this setting is just the filename, not the path. The **Store all my log files in the following folder** setting is for the path.

- **Keep a history of...:** SyncBackSE can be configured to keep a certain number of log files, e.g. the log files of the last 3 profile runs. This is a program wide setting, meaning all profiles will keep this number of log files. However, it is possible to override this value at a profile level.



Text Log Files Format

Text format log files are mainly for use by other programs. HTML log files are for people to use. The text log file is a comma-delimited list of lines with all line elements (columns) wrapped in double-quotes:

"Date & time", "Is A Control Message?", "Filename/Control Message", "Status", "File Status", "Error Type"

Date & time Date & time line was written to the log file

Is A Control Message? 1 if this is a control message. If so the [Filename] is actually an informational message. For a value of zero, it really is a filename.

Filename/Control Message Either a filename or a control message

Control Message

Status e.g. Not in destination, source copied

File Status Integer value for file status:

- 0 = Not relevant to current line, i.e. ignore the file status
- 1 = File was skipped & was in both
- 2 = File was skipped & was in source/left only
- 3 = File was skipped & was in destination/right only
- 4 = File was deleted
- 5 = File was copied
- 6 = File attributes and/or date & time changed
- 7 = File warning

- 8 = Error
- 9 = Non-critical error
- 10 = File was copied, but a reboot is required
- 11 = File was ignored during scan of source, e.g. filtered out
- 12 = File was ignored during scan of destination, e.g. not selected
- 13 = File was ignored during comparison, e.g. read-only
- 14 = File was unchanged (fast backup only)
- 15 = An exception report
- 16 = A version was restored
- 17 = The file was in neither the source nor destination
- 18 = File was renamed
- 19 = File was renamed, but a reboot is required

Error Type The end of the line will have a letter if the line refers to an error or warning:

- W** for warning messages
- E** for error messages
- N** for non-critical error messages
- X** for exceptions (SyncBackSE caused an exception error)
- O** for no error

3.1.15 Shared Settings

Shared Settings

Each profile has its own settings, e.g. what the source/left directory is. However, in some cases you may want to share settings between several profiles. For example, you'll probably have the profiles log files emailed to the same email account. Instead of setting the email connection details for each profile it would be simpler to just set it once and then have those profiles use that one set of settings. That way, if the email login password changes (for example) then you only need to change it in one place instead of in each profile.

You can share the settings on the following profile settings pages:

- Auto-close
- Backup to email
- Backup of email
- CD/DVD burning
- Cloud
- Emailing the log
- FTP
- Network
- Variables (both for group and non-group profiles)
- Zip filter

Creating New Shared Settings

To create new shared settings simply click the **New** button (in the Shared Settings box) and enter the unique name you wish to use for the shared settings, e.g. My FTP Server. Note that if you've made changes to the settings on the current settings page then they will be lost unless you apply (save) them first.

Copying Shared Settings

To copy existing shared settings simply create some new shared settings. The existing settings will be copied to the new shared settings.

Deleting Shared Settings

To delete existing shared settings simply click the **Delete** button (in the Shared Settings box) and choose the settings to delete. Note that you cannot delete shared settings that are being used, or shared settings that are set to be the default values.

Renaming Shared Settings

To rename existing shared settings simply click the **Rename** button (in the Shared Settings box) and enter the new name to use. Shared settings names must be unique, and also cannot use the name **None**. Renaming shared settings does not affect which profiles are using the shared settings.

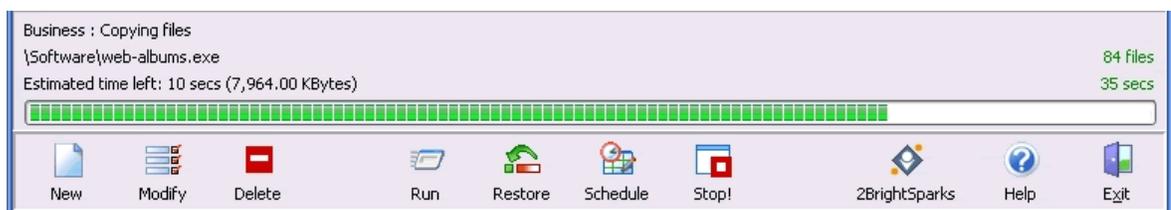
Security

Shared settings are not password protected, but profiles can be. You should be aware that if you have a password protected profile that is using shared settings, and another profile that is not password protected and using the same shared settings, then a user could change those shared settings via the unprotected profile.

3.1.16 Progress Bar

Progress Bar

The progress bar appears when a task is running and provides visual feedback of the progress of that task to the user:



A progress bar also appears behind a profiles name in the main window, and if you are

running Windows 7 or newer, in the taskbar itself. Note that the progress bar behind a profiles name isn't visible if the profile is highlighted.

3.1.17 Accessibility

Accessibility

SyncBackSE V6 has been updated to help the visually impaired and work with screen reader software:

- To change the font select Preferences -> Font Selection from the main menu. Note that this cannot be used to change the font size, only the font to be used.
- To stop SyncBackSE from using colors to highlight items, e.g. stop using red text for warnings, then select Preferences -> Font Selection from the main menu and enable the option to use the default Windows colors.
- Rich edit controls do not work with many screen readers and may cause SyncBackSE to crash. For this reason if a screen reader is being used then rich edit controls are hidden. This means the Notes setup page is not available and the profile description is not displayed.
- If a screen reader is being used then some buttons in the SyncBackSE interface will have their captions changed to allow for more feedback on what the button is for. The caption text may not fit horizontally into the button so it may look corrupted, but obviously when a screen reader is being used that is not important.

3.1.18 Exploring SyncBackSE

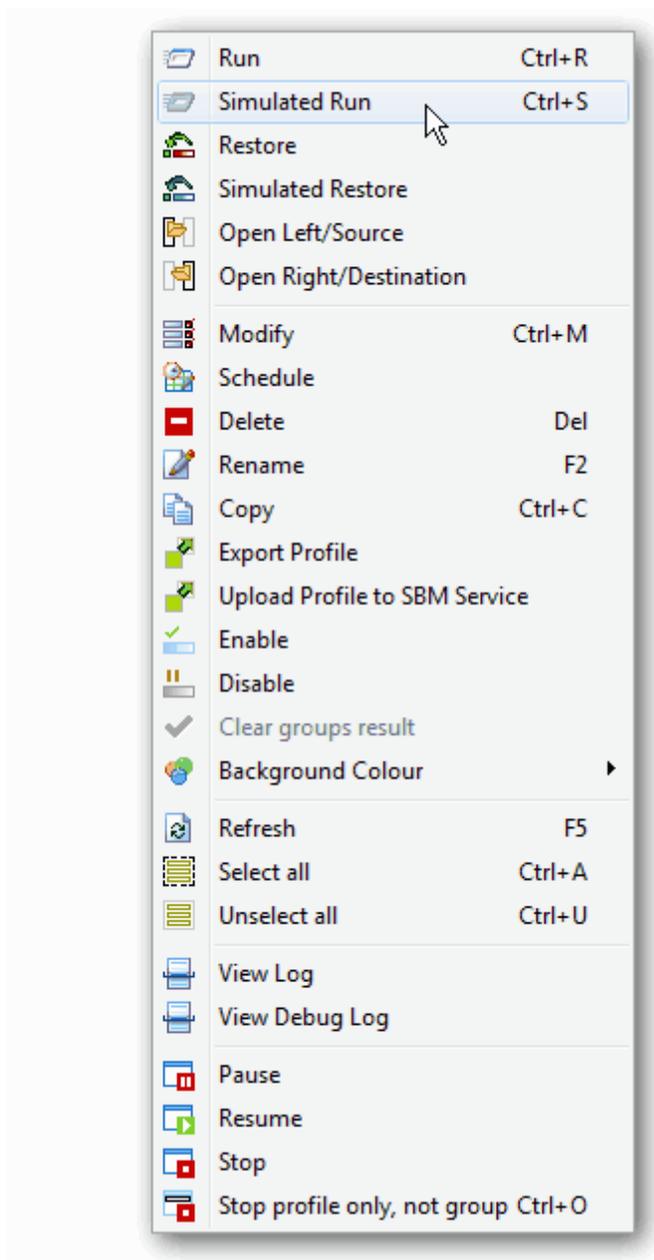
Discovering SyncBackSE

Now you've gained the essential knowledge of using SyncBackSE, we'll turn to taking advantage of some of the options available in the program. SyncBackSE has two Modes that allow you to change the settings for any given profile: Easy Mode and Expert Mode.

SyncBackSE is a very flexible program. As you gain confidence in using the program you'll discover the power and ease that backing up and synchronizing can bring to your daily work at the computer. The default settings in SyncBackSE are defined to help you manage your data backup tasks in a simple straightforward manner.



Remember at all times that SyncBackSE copies, moves, and deletes data. Please ensure that you test your settings before running them. You can achieve this easily by using the Simulated Run or Simulated Restore commands available from the 'Task' menu:



The next section of this help file presents the settings and options available to you in Easy Mode.

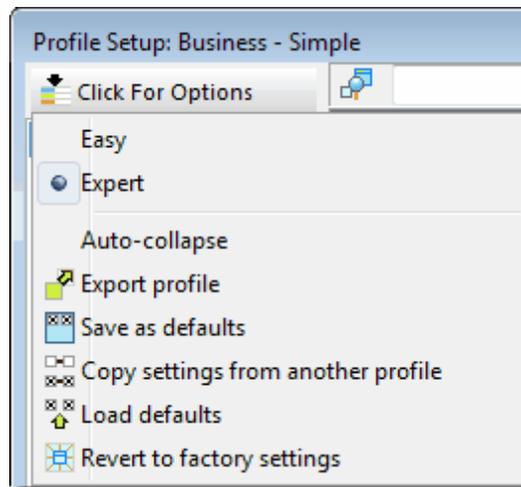
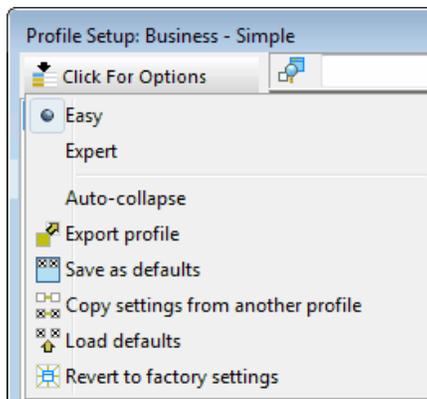
3.2 Easy Mode



Easy Operation

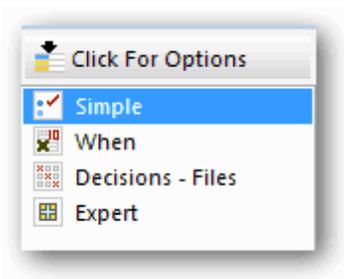
SyncBackSE provides two modes to view and change your Profiles: Easy and Expert. Easy mode presents far fewer options to modify your Profile making the choices you make less complex than in Expert mode. The Easy and Expert Mode buttons can be found at the top left of the Profile Setup Windows (select a Profile, then click the 'Modify' button at the base of the main SyncBackSE window). You can also click the Easy or Expert items in the list.

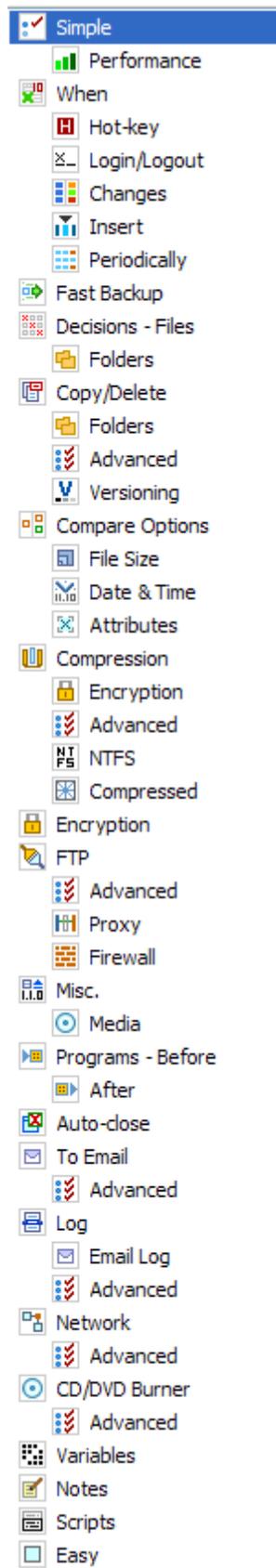
The following screenshots show the option menus in Vista:



Easy Options

Expert Options





Profile Settings in Easy Mode

The following Settings pages detail the options available for each settings tab in Easy Mode:

Easy Mode Configuration

Searching the settings

Click For Options button

Simple Settings

Sub-directories and Files

When

Fast Backup

Decisions - Files

Smart Synchronization

Notes

3.2.1 Easy Mode Configuration

Easy Mode: Configuring SyncBackSE

After a profile has been created in SyncBackSE you may modify the settings for that profile at any time. If you haven't already defined a profile view Creating a Profile which will guide you through this simple process.

Easy Mode Overview

SyncBackSE has two Modes that will allow you to change the settings for any given profile: Easy Mode and Expert Mode.

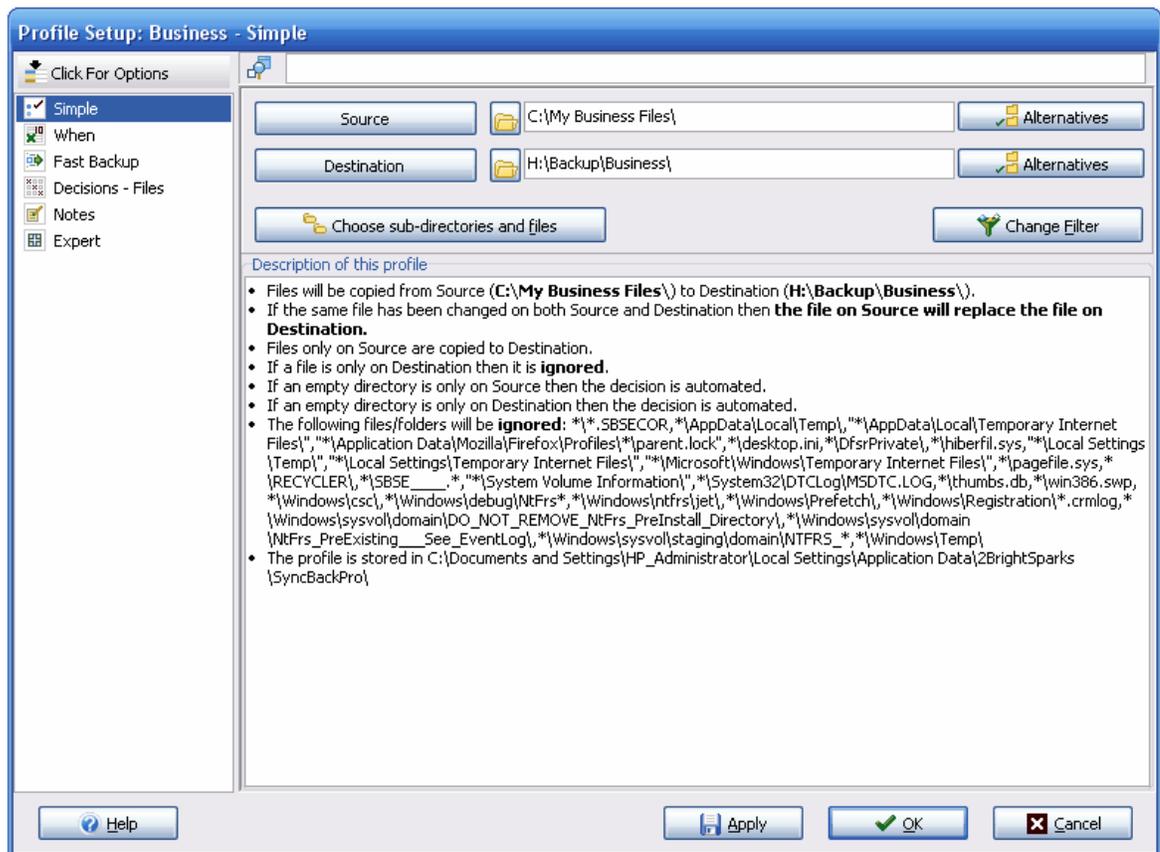


Note that these modes affect a single profile and not all profiles.

To find out how Group Profiles may be modified go to Creating a Group Profile.

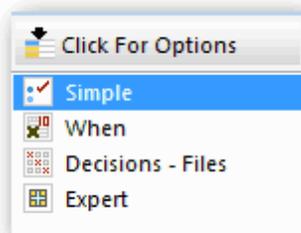
This help page details the options available in the Easy Mode. To modify a profile either use the 'Modify' button on the toolbar, or select 'Modify' from the 'Profiles' main menu located at the top of the program window.

The Easy Mode Profile Setup window is shown below with the default 'Simple' settings tabs:



 Note how the text in the white informational area helps you decide what option best suits your requirements by summarizing your profile in a text statement.

Spend time getting to know what options are available under the four additional tabs in the Profile Setup window. The 'Simple', 'When', 'Fast Backup', 'Advanced', and 'Decisions - Files' options on the left hand tree contain a range of options that provide a great deal of flexibility in the way you can perform tasks:



Profile Settings in Easy Mode

The following Settings pages detail the options available for each settings tab:

Simple Settings

When

Sub-directories and Files

Fast Backup

Decisions - Files

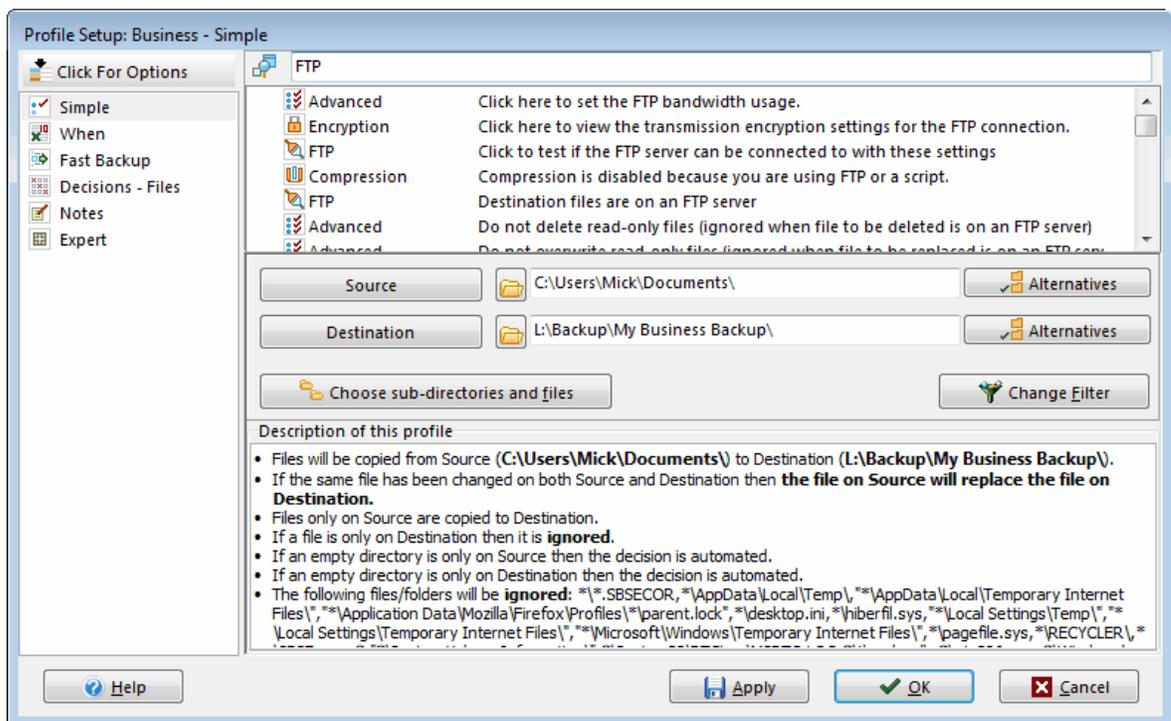
Smart Synchronization

Copy/Delete Settings

3.2.2 Searching the settings

Easy Mode/Expert Mode: Searching the settings

At the top of the Profile Setup window is a text entry area which can be used to search for settings. For example, in the following the user has typed **FTP**. The search results are updated immediately after each key press. To go to the setting you want simply click on it in the search results as shown in the Vista screenshot below:



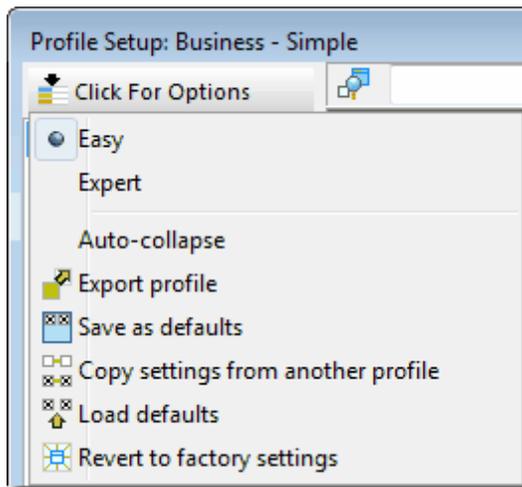
You can expand/hide the search results by clicking on the magnifying glass icon next to the

search box.

3.2.3 Click For Options button

Easy Mode/Expert Mode: Click For Options

At the top-left of the Profile Setup window is a button labeled **Click For Options**. When pressed a menu appears below the button:



- **Easy:** When selected the window goes into easy mode. In easy mode a number of features are hidden.
- **Expert:** When select the window goes into expert mode. In expert mode all features are made available.
- **Auto-collapse:** By default, when a settings page is selected it will automatically collapse the previously selected page (if it was expanded). For example, if you are in Expert mode, and on the Simple page, then go to the Compression page, then it will automatically collapse the Simple page so that the Performance page is no longer in the list.
- **Export profile:** If clicked then the profile being edited is exported to a file.
- **Save as default:** If clicked then the settings for the current page as saved as defaults. This means whenever a new profile is created then those settings will be used in the new profile. Note that this is a page specific setting, meaning it does not save the whole profile as the default, just the settings on the current page. It cannot be used on some settings pages.
- **Copy settings from another profile:** If clicked then the settings for the **current page** can be replaced with the settings from another profile. Note that this is a page specific setting, meaning it does not copy all the settings from another profile, just the settings for

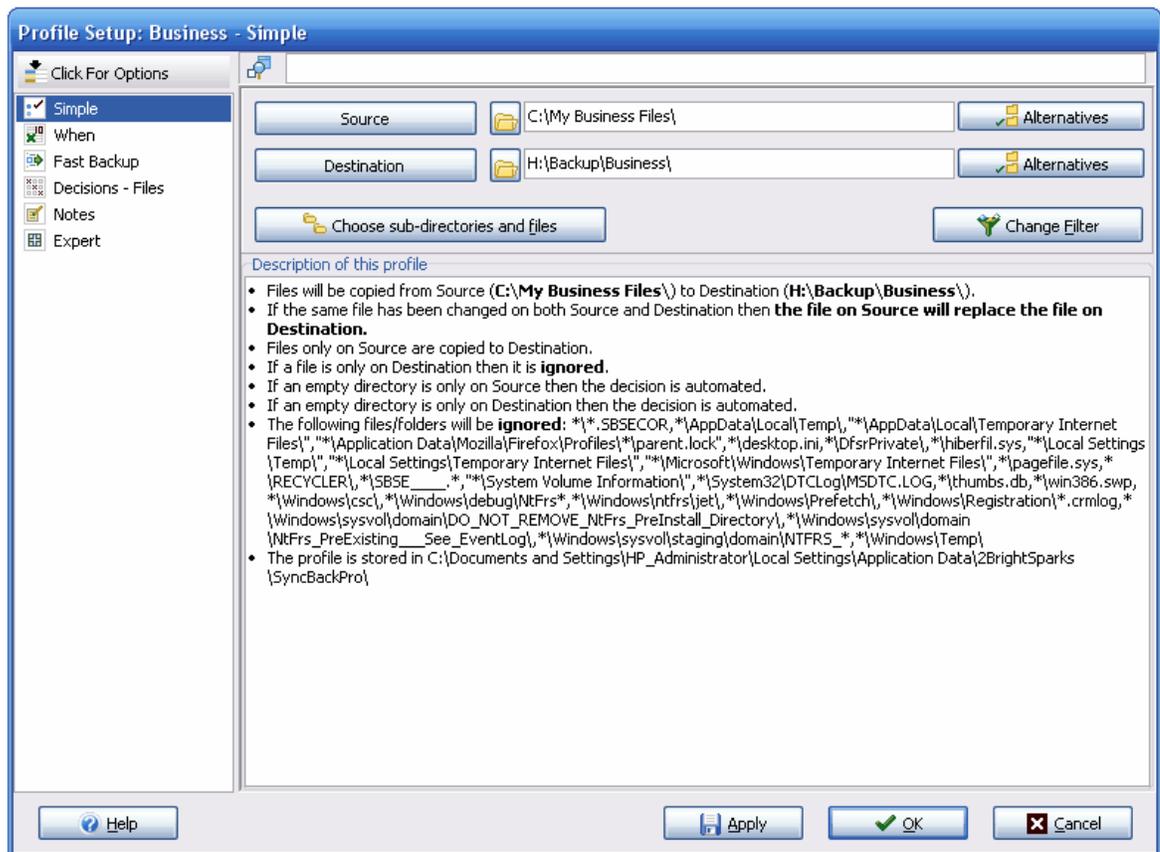
the current page.

- **Load defaults:** If clicked then the current pages settings are replaced by the default settings for this page. This is a page specific setting. It cannot be used on some settings pages.
- **Revert to factory settings:** If clicked then the current pages settings are replaced by the factory default settings. Note that these are not the same as the default settings (which can be changed by using the **Save as default** menu item). This is a page specific setting.

3.2.4 Simple

Easy Mode/Expert Mode: Simple Settings

This page gives you an overview of what the profile is configured to do. It also lets you change where files are copied to and from, which files and folders are copied, and what name you want to give to the places where files are copied to and from.

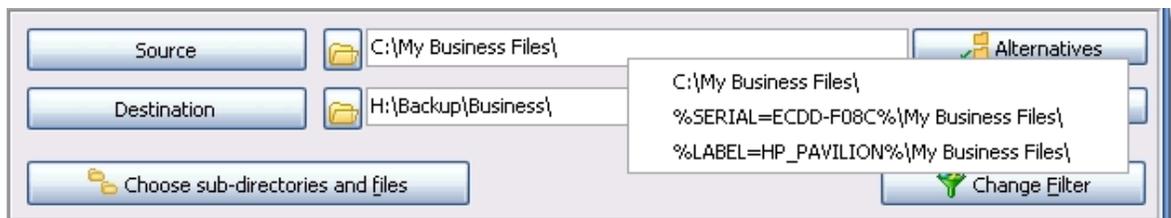


Next to the source/left and destination/right paths there are two buttons (by default labeled

Source or Left, and Destination or Right). You can click these buttons to change the name of that location. For example, you may be doing a backup from your desktop to an external hard drive. Instead of using the labels **Source** and **Destination** you may instead want them to be called **Desktop** and **External Drive**.

By default all your files and folders in the chosen folders are copied (with the exception of things like the swap file). However, if you'd like to not copy some specific folders or files, or choose which types of files not to copy (e.g. temporary files), then you can click the **Choose sub-directories and files** button to make those selections. Read more about choosing sub-directories in the Sub-directories and files section of this help file.

The **Alternatives** button can be used to change the source/left or destination/right paths to something that is more flexible and/or portable. For example, if the destination is a removable drive then you can have it changed so that the serial number of the drive is used instead of the drive letter. This means if you connect the drive, and it gets a different drive letter assigned to it by Windows, then the profile will still work as expected. Alternatives may also be given if part of the path can be substituted with Windows variables. A tick will appear next to the path that SyncBackSE suggests you use:



To filter out files and folders based on their name, e.g. you may not want to copy any .exe files, then click the **Change Filter** button. Read more about the filters in the Filter Settings section of this help file.

3.2.5 Sub-directories and Files

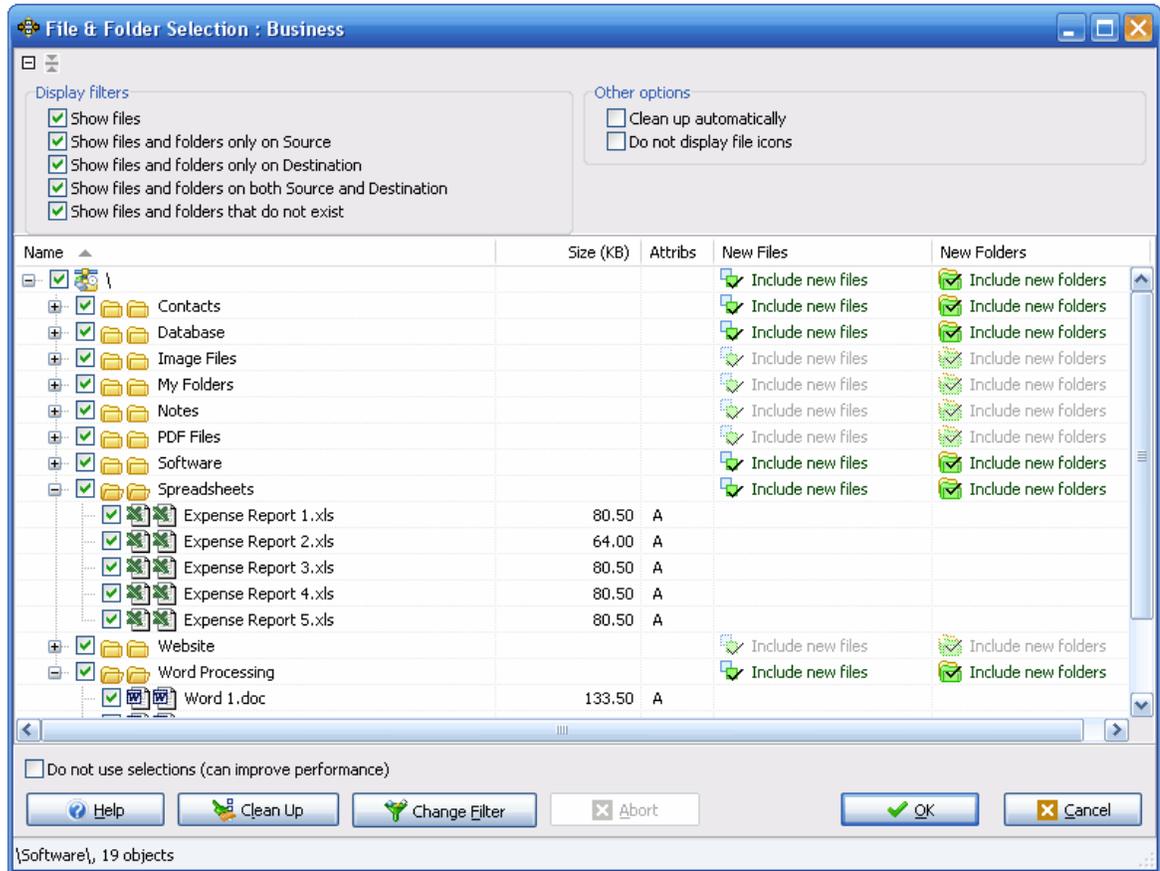
Choosing Sub-directories and Files Settings

To select exactly which folders and files you want to include in your profile, click the **Choose sub-directories and files** button on the Simple page. By default all files and sub-folders (with the exception of some folders and files you wouldn't want to copy, e.g. your swap file) are included. SyncBackSE always takes the approach that is better to include files than to exclude them.



If the profile is a traditional backup profile, i.e. it copies files to the destination and does not delete destination only files, then only the files and folders in the source are shown. If the profile is a traditional restore profile, i.e. it copies files from the destination, and does not delete source only files, then only the files and folders in the destination are shown. For all other profile types the tree will show files and folders in the source/left and destination/right. If you have a backup or restore profile

and want to force the tree to display files and folders in both the source and destination then hold down the **Ctrl** or **Shift** keys when you click the **Choose sub-directories and files** button.



The tree displayed is similar to the one used in Windows Explorer. You can collapse and expand the folders to see the files and folders within. A folder or file with a tick next to it is included in the profile.

Icons

When all the files and child folders in a folder are included in the profile, the color of the folders will be solid yellow. If any files or child folders are not included, then the folder will be half yellow. Note that in some cases it is not known if all files and child folders are included until the folder has been fully expanded.

If the profile is a Smart Synchronization profile then two icons are shown next to the files and folders: the one on the left refers to the source/left, and the one on the right refers to the destination/right. If the file or folder is ticked then the icons will be 'ghosted' if the file or folder does not exist. This lets you easily see if a file or folders exists in the source/left and/or destination/right.

New Files and Folders

On the right of the tree are two columns: **New Files** and **New Folders**. These options let you decide on whether the profile should include any new files or folders that are created within that folder. By default all new files & folders are included. If a folder is not ticked then no files or folders will be included in the profile. These options are not available until you expand the folder. This is because SyncBackSE cannot know what files and folders are new until a folder is expanded revealing what the current contents of the folder is.

As an example: you may have a folder in which you only want two specific files included and no other files or folders. You would tick those two files and untick all the other files and sub-folders. Then you would change the folder settings to ignore new files and folders. This way even if a new file is created in that folder it will be ignored, and so will any new sub-folders.

Do not use selections (can improve performance)

If this checkbox is ticked then file & folder selections are ignored. This can reduce a profiles run time, sometimes dramatically. To further improve performance you may also want to disable the filters.

Click here to see more options

If you click the bar (with the + sign at the left) at the top of the window then a number of options appear:

- **Show files:** This allows you to show or hide files in the tree.
- **Show files and folders...:** These options let you show or hide files and folders depending on whether they exist in the source/left or destination/right. If your profile is synchronizing files then the file icons also inform you if a file exists in the source/left and/or destination/right.
- **Show files and folders that do not exist:** You may have previously selected a file and/or folder to include in your backup. However, that file or folder may no longer exist, e.g. it has been deleted. In this case it will still appear in the tree but will be highlighted in red. If this option is unticked then those non-existent files and folders will be hidden. You can permanently remove them by clicking the **Clean up** button or ticking the **Clean up automatically** option.
- **Clean up automatically:** Files or folders that do not exist will be automatically removed from the tree. You can manually remove them by clicking the **Clean up** button.
- **Do not display file icons:** To improve performance on slower computers you can have file icons not displayed.

Pop-up menu

If you right-click on the tree then a pop-up menu appears with a number of options. These options apply to the selections you have made in the tree:

- **Tick selected:** All the items selected in the tree will be ticked.
- **Untick selected:** All the items selected in the tree will be unticked.
- **Collapse all:** All the folders in the tree will be collapsed.
- **Collapse selected:** All the selected folders in the tree will be collapsed.
- **Exclude folders with this name:** Folders with the same name as the ones you've selected in the tree will be added to the filters (folders not to copy) list.
- **Exclude files with this name:** Files with the same filename as the ones you've selected in the tree will be added to the filters (files not to copy) list.
- **Exclude files with this extension:** Files with the same filename extension as the ones you've selected in the tree will be added to the filters (files not to copy) list.
- **Include folders with this name:** Folders with the same name as the ones you've selected in the tree will be added to the filters (folders to copy) list.
- **Include files with this name:** Files with the same filename as the ones you've selected in the tree will be added to the filters (files to copy) list.
- **Include files with this extension:** Files with the same filename extension as the ones you've selected in the tree will be added to the filters (files to copy) list.
- **Ignore new files:** Any new files created in the selected folders will be ignored.
- **Include new files:** Any new files created in the selected folders will be included in the profile
- **Ignore new folders:** Any new sub-folders created in the selected folders will be ignored.
- **Include new folders:** Any new sub-folders created in the selected folders will be included in the profile.
- **Add file:** Add one or more files to the selected folder (or the parent folder of the selected file). To add multiple files separate them with a forward slash (/), e.g. file 1.txt/file 2.txt/file 3.txt. This menu item is hidden if the automatic clean-up checkbox is ticked. It is also hidden if a file or folder has not been selected (or multiple selections have been made). Why would you want to add a non-existent file? The file may not currently exist, but you know that it will in future. Note that as the file does not currently exist, if you click the **Clean Up** button then it will be removed from the tree.
- **Add folder:** Add one or more sub-folders to the selected folder (or to the parent folder of the selected file). To add multiple folders separate them with a forward slash (/), e.g. folder 1/folder 2/folder 3. This menu item is hidden if the automatic clean-up checkbox is ticked. It is also hidden if a file or folder has not been selected (or multiple selections have been made). Why would you want to add a non-existent folder? The folder may not

currently exist, but you know that it will in future. Note that as the folder does not currently exist, if you click the **Clean Up** button then it will be removed from the tree.

You can select multiple items in the tree by clicking the mouse and using the Shift and Ctrl keys.

Buttons

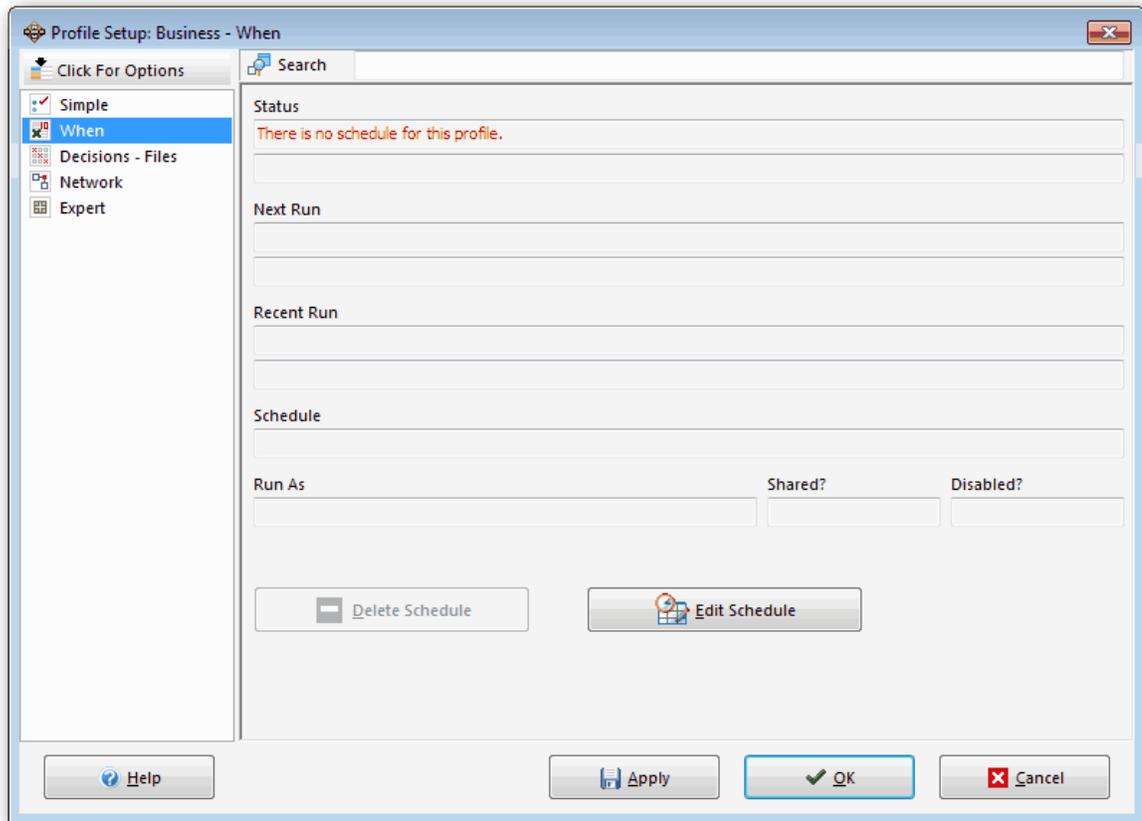
At the bottom of the window are some options available via buttons:

- **Clean Up:** Files or folders that do not exist will be removed from the tree if this button is clicked. Files and folders that do not exist are highlighted in red.
- **Change Filter:** Clicking this button displays the filter window. This lets you choose which types of files to include or exclude, and also choose which folders to include or exclude based on their name. See the Filter Settings section for more information.

3.2.6 When

Easy Mode/Expert Mode: When Settings

This page shows you if a profile has been configured to run on a schedule, e.g. 9am every morning. To create or change the schedule click the **Edit Schedule** button. For details on scheduling, and solutions to scheduling problems, see the Creating a Schedule section of this help file. Click the **Delete Schedule** button to delete an existing schedule.

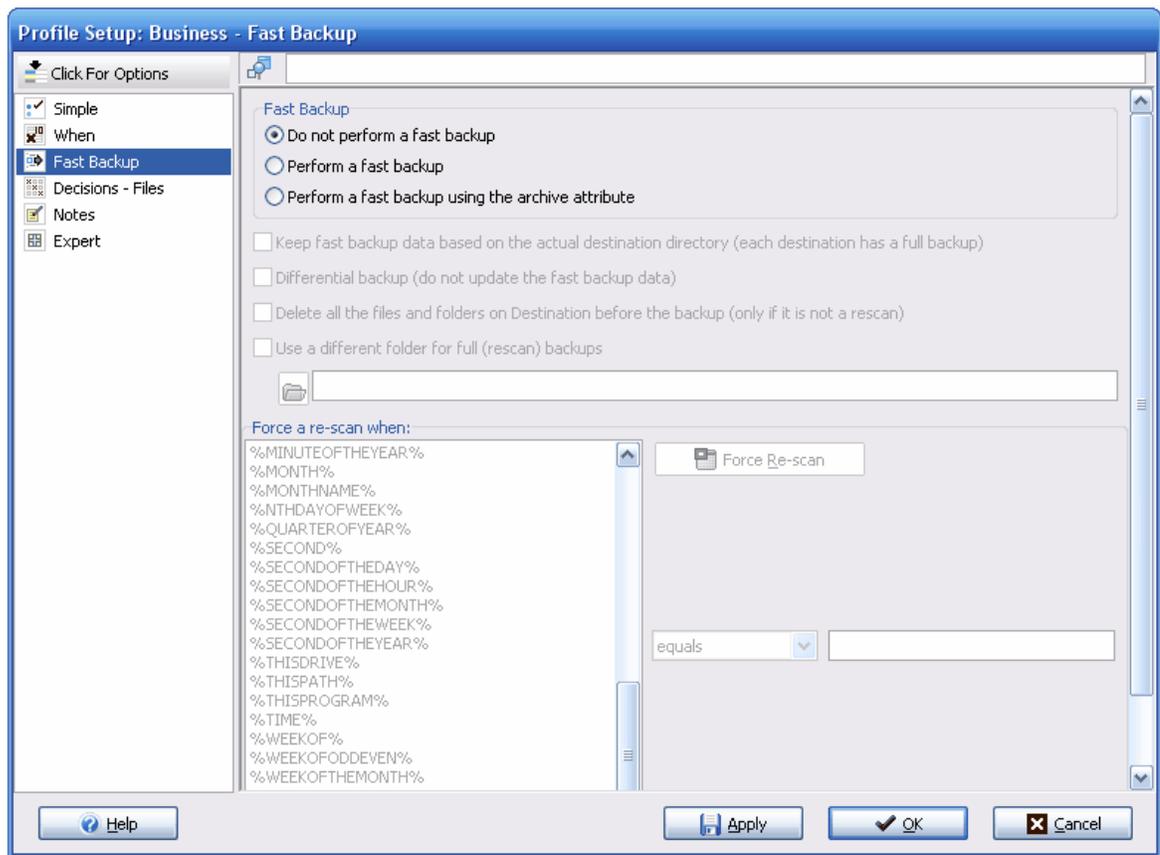


3.2.7 Fast Backup

Easy Mode/Expert Mode: Using Fast Backup

SyncBackSE can make backing up even faster if you choose the 'Fast Backup' option. **Read this help page carefully** before you continue as there are some important considerations to make when choosing this option. You'll also find some Frequently Asked Questions about Fast Backup below along with some examples.

'Fast Backup' option is displayed when you create or modify a profile and select the option from the tree:



 Some of the descriptions below refer to "Fast Backup data". This is data that is kept by SyncBackSE to keep track of what has changed between each profile run. It is used internally by SyncBackSE.

- **Do not perform a fast backup:** Enable this option to perform a standard backup, i.e. the source and destination will be scanned and compared to decide which files must be copied.
- **Perform a fast backup:** Enable this option to greatly improve the performance of a backup profile by not scanning the destination. Note that this option is only available when a profile is configured in a certain way (i.e. the source is not being changed by the settings). For more information, including side effects of fast backups, please see the section below.
- **Perform a fast backup using the archive attribute:** Enable this option to greatly improve the performance of a backup profile by not scanning the destination. This option is different from the one above because it uses the traditional backup method of using the archive attribute of a file to decide if a file should be copied or not. Note that this option is only available when a profile is configured in a certain way. For more information, including side effects of fast backups, please see the section below.
- **Keep fast backup data based on the actual destination directory (each destination has a full backup):** If this option is unticked then a fast backup works much

the same as an incremental backup. This means that only new or modified files are copied from the source to the destination regardless of where the destination is. If you are not using variables, e.g. %DAYOFWEEK%, in the destination then you can leave this option unticked as it will make no difference. This option is not available when using a Fast Backup with the archive attribute.

If you are using variables in the destination then you should consider ticking this option. If this option is ticked then the fast backup works in a different way. It keeps track of which files and folders are in each destination. This means each destination directory will have a complete backup and not just contain the new/changed files.

- **Differential backup (do not update the fast backup data):** If this option is ticked then a fast backup works the same as a differential backup. This means that only new or modified files since the last full backup are copied from the source to the destination. If you are not using variables, e.g. %DAYOFWEEK%, in the destination then you can leave this option unticked as it will make no difference. For example, if your destination is X:\%DAYOFWEEK%, and you force a rescan on Mondays, then your Monday backup will be a full backup. Your Tuesday backup will contain new and changed files since Monday, Wednesday's backup will contain new and changed files since Monday, etc. If the Fast Backup is using the archive attribute then the archive bit is not cleared on the original file when it is copied unless it is a full/rescan backup.
- **Delete all the files and folders in the destination before the backup (only if it is not a rescan):** If this option is ticked then all the files and folders (just the Zip file if compressing to one single Zip file) in the destination are deleted before the backup is made. The files and folders are not deleted if a re-scan has been forced or is required.

This option is best used when "Keep fast backup data based on the actual destination directory" is unticked. For example, if you keep 7 days worth of backups (using the %DAYOFWEEK% variable in the destination), and force a rescan each Monday, then by enabling this option you'll ensure that the Monday backup contains a complete backup and that the backups for all the other days just contain new/changed files since the previous day.

- **Use a different folder for full (rescan) backups:** If this option is ticked then you can define which folder should be used for full backups, i.e. where to backup to if there is a re-scan. This can be very useful when you always want full backups to go into one folder, and incremental/differential backups to go into the usual destination folder. If you tick this option then SyncBackSE will automatically set the full folder to your destination folder. However, it is unlikely this is the folder that you wish to use so you should modify it as appropriate.

For example, if your destination is X:\%DAYOFWEEK%, and you have it set to rescan on Monday, then you probably want your full backup folder set to X:\1. This means the full backups will always go into the Monday (1) folder even if you force a rescan on a Friday, for example.

- **Force re-scan:** Click this button to force SyncBackSE to scan the destination next time the profile is run. Please read the notes below about the consequences of forcing a rescan when using FTP. If the button is disabled it is either because the profile is not a

Fast Backup profile or SyncBackSE will already be performing a rescan on the next profile run, e.g. the button has already been pressed.

- **Force a re-scan when:** This lets you define when SyncBackSE should perform a complete re-scan of the destination. For example, to force a complete re-scan every Monday you would select **%DAYOFWEEK%** from the list, select equals from the drop-down, and type in **1** (1=Monday, 7=Sunday) in the edit box. See the section below for details on why this may be required and what it does. Please read the notes below about the consequences of forcing a rescan when using FTP. Note that you cannot enter a list of values, for example you cannot use **%DAYOFWEEK%** and enter 1,3 to rescan on Mondays and Wednesdays. Only one value may be entered. With the  Pro version it is possible to use scripting to decide when a rescan should occur, meaning far more complex evaluations can be made (see the **IncVar.vbs** example script).

Important Information About Fast Backups

Explaining Fast Backups

When you backup files to the destination it is assumed that no other application, or person, will be changing the files in the destination. For example, if you backup your files to another drive you are not going to be editing or changing those backup files (except using SyncBackSE to replace them as appropriate). Because of this SyncBackSE should be able to remember what files, and directories, are on the destination without needing to scan it to find out.

How 'Fast Backup' works

First, you need to enable a Fast Backup option on the Fast Backup tab. The Fast Backup option is not available if the profile is configured such that it cannot use the Fast Backup option, e.g. it's a Smart Synchronization profile.

How Fast Backup works depends on whether it's using the archive attribute or not:

- **Not using the archive attribute:** When the profile is next run, SyncBackSE will remember which files and directories it copied to (or deleted from) the destination directory. This means the first run of a profile, after Fast Backup is enabled, will take the same amount of time as without Fast Backup enabled. However, for the second and subsequent runs of the profile it will not need to scan the destination directory because it remembers what it did the last time the profile was run.
- **Using the archive attribute:** Each file has what is called an **archive attribute** (just like files have read-only, hidden, etc. attributes). Whenever a file is changed the archive bit is automatically set (by Windows), and when SyncBackSE copies a file it clears the archive bit. So when SyncBackSE needs to know which files are to be copied to the destination it just needs to see if the archive attribute is set. There is no need to scan the destination. The main advantage to using the archive attribute is that there is no need to keep information on the state of the files (so less disk space is used). It may also be very slightly faster (as it doesn't need to read and save the information on the files).

Using Fast Backup means the scan time is substantially lower (at least twice as fast, often

far more) especially if the destination is on a slow device, e.g. networked drive, FTP server, etc.

Rescan with archival backups

If a rescan is done, either by clicking the Force Rescan button or by other means, then when the profile is next run it will scan both the source and destination, compare the files, then copy new and changed files. This has a side effect with archival backup in that it will not copy a file (even if it has its archive attribute set) if the source and destination files are the same. If the profile is run again (so it is not a rescan) then it will scan the source, see the file has the archive attribute set, and copy it to the destination regardless.

Can I use Fast Backups on all profile types?

No. The Fast Backup option is only possible when no changes are being made to the destination by other programs, profiles, or users. This means it cannot be used with backups from FTP or Zip files, synchronization profiles, or profiles that use prompting. It is for backup profiles only.

If you are using archival backups you must keep in mind SyncBackSE is relying on the archive attribute being set once a file is created or changed and is not reset by anything else other than that profile in SyncBackSE. Once the archive attribute is set then SyncBackSE knows the file needs to be backed up. However, some other programs, e.g. other backup software, may also use and reset file archive attributes. You must also be careful not to have more than one profile that copies the same files and resets the archive attribute.

Sometimes SyncBackSE scans the destination directory even though I've enabled Fast Backups. Why?

There are a number of reasons why SyncBackSE may scan the destination directory:

- The **Force Re-scan** button has been pressed for that profile.
- The settings on the **Fast Backup** tab specify a re-scan should be performed under certain conditions.
- The **-full** command line parameter was used.
- The fast backup data has been deleted.
- The filters or file & folder selections have been modified.

What options does using Fast Backup disable?

When using Fast Backups you cannot enable the following options in your profile:

- Reset the archive file attribute on files once they have been copied.
- Files cannot be moved, and files cannot be copied to the source or deleted from the source (this option can be used with archive attribute fast backups)

- The destination cannot be watched for file changes.

What side effects are there with using Fast Backups?

If the profile is set to delete destination only files, SyncBackSE may not know a new file has been created in the destination (see this section for more details).

Because a Fast Backup will not scan the destination (except on the first run after it is enabled for that profile) that means only the changes will be applied to the destination without regard to what is actually on the destination. For example, you could change the destination directory, run the profile, and then only the new/modified files would be copied to the destination (and not all the files as would normally be the case).

This has important consequences when your destination directory is dynamic, i.e. it uses environment variables that can change in value. For example:

- Create a normal backup profile and set the source directory to **C:\My Documents** and the destination directory to **D:\Backup\%DAYOFWEEK%**
- Enable Fast Backup for the profile.
- On the first run of the profile (let's assume it's Monday and the destination directory is empty) all the files will be copied to **D:\Backup\1**
- When the profile is run on Tuesday then only the new or modified files will be copied to **D:\Backup\2**
- On Wednesday new and changed files will be copied to **D:\Backup\3** and so on until Monday.
- When it is run again on the following Monday then only the new and changed files will be copied to **D:\Backup\1**
- Enable the option "Keep fast backup data based on the actual destination directory". This will create full backups for each day and not just incremental backups for Tuesday to Sunday.

Or

- Enable the option "Delete all the files and folders in the destination before the backup" and force a rescan on Mondays. There are three ways to do this (using this example):
 1. The best and easiest option is to configure the "Force a re-scan when:" settings to force a re-scan every Monday (select %DAYOFWEEK% from the list, select equals from the drop-down, and type in 1 (1=Monday, 7=Sunday) in the edit box).
 2. You can do this manually by clicking the Force Re-scan button on the Mondays.
 3. Use the -full command line option (for Monday only when scheduling).

This will mean that the Monday backup is a complete backup, and the backups on Tuesday to Sunday contain just the new/changed files since the previous days backup.

What about FTP and Fast Backups?

One of the benefits of using Fast Backup and FTP is that it can really improve the backup time. Apart from not having to scan the FTP server to find changes, SyncBackSE also does not need to set the date & time of the file on the FTP server to match that of its equivalent file on your PC. This can further reduce the backup time.

The Archival Fast Backup option is not available when doing multi-zip backup to an FTP server. When multi-zip files are stored on an FTP server SyncBackSE must name the Zip files in a special way (to store information like their uncompressed size, for example). Because of this the destination must be scanned to know what those filenames are. The non-archival Fast Backup method can be used (in most cases) because it knows what the destination filename is without scanning (because it is in the fast backup database). However, if the destination folder is dynamic, e.g. it is using a variable, then it can cause problems as it will not always be able to know what the destination filename is.

If you don't care if the date & time of the files on your FTP server match those on your PC then you can untick the **"If the FTP server cannot set a files date & time then change the local files date & time to match that on the server"** option on the FTP tab.

There are important consequences to doing this: the date & time the file should be set to (on the FTP server) is kept in the Fast Backup data. Therefore, if you force a re-scan (so erasing that data) then the last date & time information is permanently lost. What does this mean?

- When you do a restore all the files will be retrieved from the FTP server along with their last modification date & time, which is not the original value. As the date & times won't match, all the files will be restored unless you've configured your profile to ignore file date & times.
- When you next run a backup all the date & times will be mismatched so forcing a complete backup (unless you've configured your profile to ignore file date & times).

What about backup to a single Zip file on an FTP server?

Making a backup to a single Zip file on a remote FTP server creates some interesting challenges:

1. To update a remote Zip file it would need to be downloaded, updated, and then uploaded. Depending on the size of the Zip file, this could be extremely slow.
2. To know what files and folders are in a remote Zip file it would need to be entirely downloaded.

Because of this SyncBackSE will always replace the existing Zip file and therefore assume there are no files in the remote Zip file. The solution is to use a Fast Backup profile, i.e. an incremental or differential backup. For example, to keep 7 days worth of backups, and have

a full-backup on Mondays and incremental backups on all the other days of the week:

- Create a backup profile and configure your source as appropriate, and set your destination as appropriate (i.e. a single Zip file on an FTP server)
- Set the **Destination** to something (the %DAYOFWEEK% is required) like **\My Backups\%DAYOFWEEK%.zip**
- Go to the Fast Backup tab and enable the options: **Perform a fast backup** and set the full-backup folder to **\My Backups\1.zip**
- In the **"Force a re-scan when:"** box select the %DAYOFWEEK% item, select **Equals** from the drop-down list to the right of the box, and enter **1** into the box to the right of the drop-down list (we use 1 because Monday is day 1, Tuesday is day 2, etc).
- You should schedule the profile to run just once every day

Note about "Delete all the files and folders in the destination before the backup"

This option should be used with care because it will delete all the files and folders in the destination before running the profile. However, if you are compressing to a single Zip file it will just delete that Zip file and no other folders or files.

An important detail to remember about this setting is that the destination files are not deleted if the profile run is doing a rescan. For example, if you've configured your profile to rescan on Mondays (%DAYOFWEEK% equals 1) then when the profile is run on a Monday it will not delete the destination files. However, it will delete them if there is no rescan. This has consequences if you run your profile more than once and also if you do not ever do a rescan. For example, if you configured your profile to rescan on Mondays then when the profile is run on a Tuesday it will delete the destination files and copy over the new or changed files since the last profile run. However, if you run it again immediately (and assuming it is still Tuesday) it will delete those files and then copy over any new or changed files since the last run, which may be no files at all.

Versioning

As an alternative to doing incremental or differential backups, you may want to consider using Versioning instead. With versioning you can keep a defined number of versions of a file. This means you can keep old versions of files that have been changed or deleted. Note that you can use versioning with Fast Backup, but it can become complex and will slow down a Fast Backup (as SyncBackSE must scan the destination to know what versions are available).

Example Fast Backup Configurations

The following section provides example backup configurations.

To keep 7 days worth of backups, and have a full-backup on Mondays and incremental

backups on all the other days of the week:

- Create a backup profile and configure your source as appropriate
 - Set the **Destination** to something (the %DAYOFWEEK% is required) like **D:\My Backups\%DAYOFWEEK%**
 - Go to the Fast Backup tab and enable the options: **Perform a fast backup** and **Delete all the files and folders in the destination before the backup**, and set the full-backup folder to **D:\My Backups\1**
 - In the "**Force a re-scan when:**" box select the %DAYOFWEEK% item, select **Equals** from the drop-down list to the right of the box, and enter **1** into the box to the right of the drop-down list (we use 1 because Monday is day 1, Tuesday is day 2, etc).
 - You should schedule the profile to run just once every day
-

To keep 7 days worth of backups, and have a full-backup on Mondays and differential backups on all the other days of the week:

- Create a backup profile and configure your source as appropriate
 - Set the **Destination** to something (the %DAYOFWEEK% is required) like **D:\My Backups\%DAYOFWEEK%**
 - Go to the Fast Backup page and enable the options: **Perform a fast backup**, **Differential backup (do not update the fast backup data)**, **Delete all the files and folders in the destination before the backup**, and set the full-backup folder to **D:\My Backups\1**
 - In the "**Force a re-scan when:**" box select the %DAYOFWEEK% item, select **Equals** from the drop-down list to the right of the box, and enter 1 into the box to the right of the drop-down list (we use 1 because Monday is day 1, Tuesday is day 2, etc).
 - You should schedule the profile to run just once every day
-

To keep 7 days worth of backups and have full-backups for each day:

- Create a backup profile and configure your source as appropriate
 - Set the **Destination** to something (the %DAYOFWEEK% is required) like **D:\My Backups\%DAYOFWEEK%**
 - Go to the Fast Backup tab and enable the options: **Perform a fast backup** and **Keep fast backup data based on the actual destination directory**
-

- Go to the **Decisions - Files** page and configure the profile to delete files from the destination that do not exist on the source. This ensures that your backups don't get cluttered with redundant files.
 - You should schedule the profile to run just once every day
-

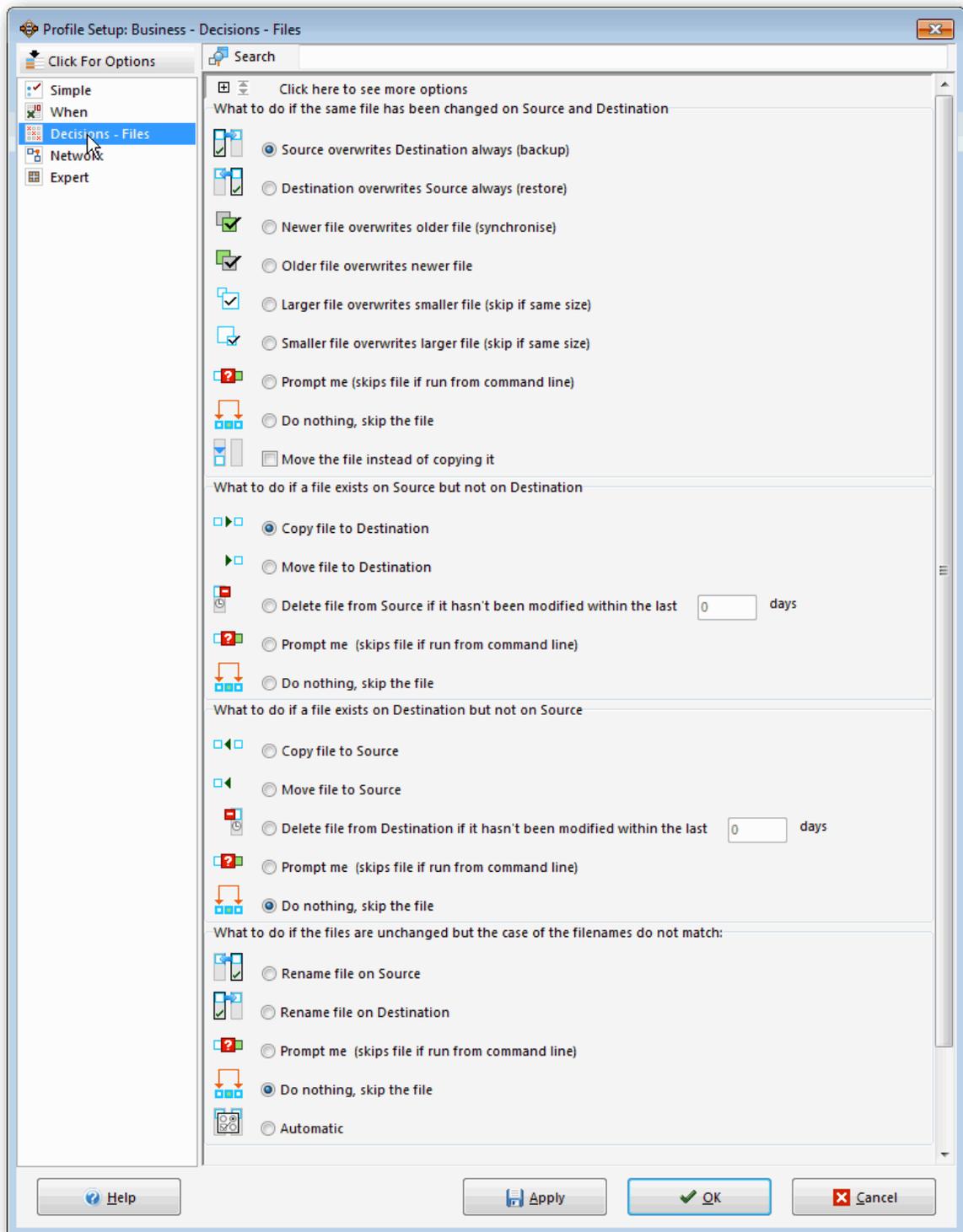
To keep full-backups on a set of disks:

- Create a backup profile and configure your Source as appropriate
- **Set the Destination, e.g. X:\%SERIAL%, where X:** is a drive that accepts removable media, e.g. a CD/DVD drive, Zip drive, floppy disk drive, etc.
- **Go to the Fast Backup tab and enable the options:** Perform a fast backup **and** Keep fast backup data based on the actual destination directory
- Go to the **Decisions - Files** page and configure the profile to delete files from the destination that do not exist on the source. This ensures that your backups don't get cluttered with redundant files.
- Each time you run the profile use a different disk. You could keep 10 days worth of backups by rotating a set of 10 disks. Each disk will have a complete backup.

3.2.8 Decisions - Files

Easy Mode/Expert Mode: Decisions - Files Settings

Define how SyncBackSE will perform its task under different circumstances.



For example, what SyncBackSE will do if the same file is in the destination but not the source.

The **Decisions - Files** page lets you tell SyncBackSE which files to copy, delete, move, or

rename. If this is a new profile then these settings have already been correctly chosen for you and there is no need to change them. Which options are displayed depends on whether you have a Smart Synchronization profile or not.

There are four different situations in which SyncBackSE must decide what action to take:

1. When there is a file (with the same name and in the same directory) that is both in the source and the destination but their contents are not the same. For example, you may have changed the file in the source.
2. When there is a file that is in the source, but not in the destination. For example, you may have deleted the file in the destination.
3. When there is a file that is in the destination, but not in the source.
4. When the files have not been changed but the case is different. For example, the source file may be called **abc.txt** and the destination file called **ABC.TXT**. See the Renaming Case Change section for important information.

Pre-configured Settings

At the top of the window is a bar that when clicked shows several buttons:



Clicking on these buttons will change the settings on this page to perform that task. For example, if you want to reset the settings so it's a backup then click the "Backup Source to Destination" button. If a button is highlighted that was the original settings of the profile. This helps you go back to your original settings if you need to.

Non-Smart Synchronization Options

For Smart Synchronization profiles please see this section of the help file.

First Advanced Settings Group

The first group of settings on this page let you decide what SyncBackSE should do for situation 1, i.e. when a file is in the source and destination, but they are not the same file:

- **Source overwrites destination always (backup):** A file from the source directory will always replace a file in the destination directory. Choose this option when doing backups.

- **Destination overwrites source always (restore):** A file from the destination directory will always replace a file in the source directory.
- **Newer file overwrites older file (synchronize):** The newer file will replace the older file, i.e. the file last modified replaces the older file. Choose this option when synchronizing directories. If the files have the same date & time then they are skipped. However, if it is a Smart Synchronization profile, and the dates and times are the same, and it knows which file was changed, then the changed file will replace the unchanged file.
- **Older file overwrites newer file:** The older file will replace the newer file. This is the exact opposite of the previous option. If the files have the same date & time then they are skipped.
- **Larger file overwrites smaller file (skip if same size):** The larger file will replace the smaller file (and no copy is made if they are the same size).
- **Smaller file overwrites larger file (skip if same size):** The smaller file will replace the larger file (and no copy is made if they are the same size). This is the exact opposite of the previous option.
- **Prompt me (skips file if run from command line):** If both files have been changed then you are prompted and will be able to decide what to do. Note that if SyncBackSE is run from the command line, or from the Windows Task Scheduler, then no prompt will appear and the file will not be copied.
- **Do nothing, skip the file:** In this case no file is copied and nothing is done.
- **Move the file instead of copying it:** The file will be moved from the source to the destination. Note that this option is not always available, e.g. if your profile is a Fast Backup profile.

Second Advanced Settings Group

The second group of settings on this page let you decide what SyncBackSE should do for situation 2, i.e. when a file is in the source but not the destination:

- **Copy file to destination:** The file is copied from the source to the destination.
- **Move file to destination:** The file is moved from the source to the destination. Note that this option is not always available, e.g. if your profile is a Fast Backup profile.
- **Delete file from source if it hasn't been modified within the last x days:** The file is deleted from the source. If the days value is greater than zero then the file is deleted from the source only if it has not been modified in that number of days.
- **Prompt me (skips file if run from command line):** You will be prompted on what action to take. Note that if SyncBackSE is run from the command line, or from the Windows Task Scheduler, then no prompt will appear and the file will be ignore.

- **Do nothing, skip the file:** Nothing will be done and the file will be ignored.

Third Advanced Settings Group

The third group of settings on this page let you decide what SyncBackSE should do for situation 3, i.e. when a file is in the destination but not the source:

- **Copy file to source:** The file is copied from the destination to the source.
- **Move file to source:** The file is moved from the destination to the source. Note that this option is not always available, e.g. if your profile is a Fast Backup profile.
- **Delete file from destination if it hasn't been modified within the last x days:** The file is deleted from the destination. If the days value is greater than zero then the file is deleted from the destination only if it has not been modified in that number of days.
- **Prompt me (skips file if run from command line):** You will be prompted on what action to take. Note that if SyncBackSE is run from the command line, or from the Windows Task Scheduler, then no prompt will appear and the file will be ignore.
- **Do nothing, skip the file:** Nothing will be done and the file will be ignored.

Fourth Advanced Settings Group

The fourth group of settings on this page let you decide what SyncBackSE should do for situation 4, i.e. when a file is unchanged but the case of the filename is different. See the Renaming Case Change section for important information.

- **Rename file on source:** The file is renamed on the source so that it becomes the same as the destination filename.
- **Rename file on destination:** The file is renamed on the destination so that it becomes the same as the source filename.
- **Prompt me (skips file if run from command line):** You will be prompted on what action to take. Note that if SyncBackSE is run from the command line, or from the Windows Task Scheduler, then no prompt will appear and the file will be ignored.
- **Do nothing, skip the file:** Nothing will be done and the differences in filename case will be ignored. This is the default action.
- **Automatic:** The appropriate file is renamed. Which file is renamed? It depends on what type of backup location you are using and the type of profile you have. If you are using FTP, compressing to a single Zip file, burning to CD/DVD, an email server, or the cloud, then it will rename the file stored on a disk or network drive. If you are copying to and from a disk or network drive then it will rename the destination file if it's a backup or mirror to the destination. If it's a backup or mirror to the source then it will rename the source file.

Fast Backup

If you are using Fast Backup then you must keep in mind that SyncBackSE does not scan the destination unless it is a rescan. For example, if you are using a Fast Backup profile that uses the archive attribute, and it is not a rescan, and you've configured the profile to delete destination only files, then it does not know what files are in the destination. This means if you create a new file in the destination, or delete a file from the source, then it will do nothing. If you are using a fast backup that does not use the archive attribute, and it is not a rescan, and you've configured the profile to delete destination only files, then it works slightly differently than the archive attribute method. This is because SyncBackSE keeps track of what files were previously in the source (which should therefore be the same as the destination). If you delete a file from the source, and there is an equivalent destination file, then it will delete the destination file. However, if you create a new file in the destination then it will not be deleted.

3.2.9 Smart Synchronization

Using 'Smart Synchronization'

Smart Synchronization Options

With Smart Sync you are given a large number of options of what to do in each situation. Although the list and options appear daunting, taking the default values is nearly always the best option.

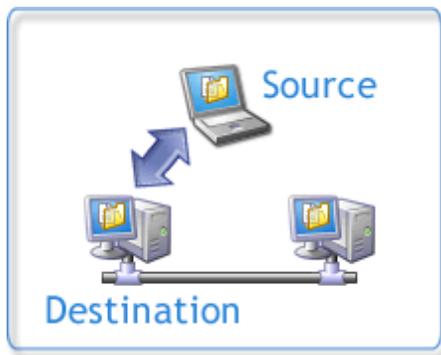


Smart Synchronization cannot be used when compressing all files into one single file or when burning to a CD/DVD.

In the description of the options below an example follows about how this option may be triggered when you do a Smart Sync.

Example Scenario

You have a local copy of the files on your laptop computer. In your profile setup this is the Source directory. There is also a copy of those files on the company network. In your profile setup this is the Destination directory.



People in the company are making changes to the files on the network, and when you are out of the office you are changing your files on your laptop. Once you return to the office you are connecting your laptop to the company's network and synchronizing your files using SyncBackSE.

Note that when you first run a Smart Sync profile it has no history of data to look at and so, for example, cannot know if a file was only changed in the source but not the destination.

What to do if...:

-  **...the same file has been changed in both the source & destination:** You have changed a file on your laptop and someone has also changed the same file on the network. In this situation, it's best to be prompted on what to do. You may need to manually merge the file contents. Tick the **Move the file instead of copying it** checkbox to move the file.
-  **...the file has only been changed in the source (unchanged in destination):** You have changed a file on your laptop and nobody changed that file on the network. In this situation the default is to copy your changed file to the network (Source overwrites destination always). Tick the **Move the file instead of copying it** checkbox to move the file.
-  **...the file has only been changed in the destination (unchanged in source):** A file was changed on the network and you didn't change your local copy of the file. In this situation the default is to replace your local copy with the changed one (Destination overwrites source always). Tick the **Move the file instead of copying it** checkbox to move the file.
-  **...a file is deleted from the source (but was changed or created in the destination):** You have deleted a file on your laptop but someone has changed that file on the network or created a file with the same name on the network. In this situation the default is to be prompted as you may either want to delete the network file, or copy the

network file to your laptop.

-  **...a file is deleted from the source (but is unchanged in the destination):** You have deleted a file on your laptop and the file on the network was not changed. In this situation the default is to also delete the file from the network (Delete file from destination).
-  **...a file is deleted from the destination (but was changed or created in the source):** Someone has deleted a file from the network but you've changed your local copy of that file (or created a new file with the same name). In this situation the default is to be prompted as you may either want to delete your local copy of the file, or copy the local file to the network.
-  **...a file is deleted from the destination (but is unchanged in the source):** Someone deleted a file on the network and the file your laptop was not changed. In this situation the default is to also delete the file from your laptop (Delete file from source).
-  **...a new file has been created in both the source & destination and are different:** The same file has been created on the network and your laptop. This happens when you first run a Smart Sync profile. The default is to be prompted, and usually you would choose to copy the newer file over the older file (Newer file overwrites older file). Tick the **Move the file instead of copying it** checkbox to move the file.
-  **...a new file has been created in the source only, or is only in the source:** You have created a new file on your laptop and it doesn't exist on the network. This happens when you first run a Smart Sync profile. The default is to copy the file to the network (Copy file to destination).
-  **...a new file has been created in the destination only, or is only in the destination:** Someone has created a new file on the network and it doesn't exist on your laptop. This happens when you first run a Smart Sync profile. The default is to copy the file to your laptop (Copy file to source).
-  **...the case of a file on source has been changed (unchanged on destination):** The file on the source and destination is identical except for the case of its filename. The case of the source filename has been changed so that it is no longer the same as the destination. For example, the source file was previously called **abc.txt** but has been renamed to **ABC.TXT**. See the Renaming Case Change section below for important information.

-  **...the case of a file on destination has been changed (unchanged on source):** The file on the source and destination is identical except for the case of its filename. The case of the destination filename has been changed so that it is no longer the same as the source. See the Renaming Case Change section below for important information.
-  **...the case of a file on both source and destination has been changed:** The file on the source and destination is identical except for the case of its filename. The case of both the source and destination filename has been changed and they are not the same. See the Renaming Case Change section below for important information.
- **Detect file renames on Source (note this may reduce performance):** If this option is ticked then SyncBackSE will try to detect files that have been renamed/moved in the source. If a file has been renamed on the source then it will rename the destination file to match it. Note that this option requires that file contents be compared, which means this option could be very slow when there are many files or very large files. It will only compare files when it needs to.
- **Detect file renames on Destination (note this may reduce performance):** If this option is ticked then SyncBackSE will try to detect files that have been renamed/moved in the destination.
- **When prompting, and the run is unattended, ignore the changes:** For some of the decisions you may have chosen to be prompted. However, if the profile is being run unattended then you cannot be prompted. In this case the file is ignored and a warning is recorded in the log file. With a SmartSync profile you may want SyncBackSE to ignore the changes. See the Ignoring Changes section for important details on what this means.
- **Clear History:** If clicked, then the Smart Sync history (information) will be cleared. If it is cleared then when the profile is next run SyncBackSE will not have any history to base its decisions on, so it cannot know if a file was only changed in the source but not the destination. Clearing the history is equivalent to not having yet run the profile. This button is disabled if there is no history to clear.

Ignoring Changes

Smart Synchronization works because SyncBackSE keeps a database of the state of files and folders, to which it can then refer to detect what has changed since the last time the profile was run. For example, if SyncBackSE knows a file existed on the source, and then when you next run the profile the file no longer exists, it knows the file was deleted from the source. Without that database it would have no idea that the file had existed before. After a profile has finished it updates this database so it can be used on the next run.

However, if a file is being skipped because you cannot be prompted on the decision to make, then you may not want SyncBackSE to update that database for those files that were skipped because you could not be prompted. Let's use an example of why you may want to

do this:

1. You have the same file, file.txt, in the left folder (**C:\Left\file.txt**) and the right folder (**C:\Right\file.txt**).
2. You modify the left file and also modify the right file.
3. The profile is run unattended. Both files have been modified and you've chosen to be prompted in both files have been modified. However, the run is unattended so the files are skipped because you cannot be prompted.
4. **After the profile has finished SyncBackSE updates the SmartSync database with the new details of the files.**
5. Later you modify the left file.
6. The profile is run attended, i.e. you can be prompted. SyncBackSE will now copy the left file over the right file. Why? Because since the last run (in step 3) the left file has been modified and the right one has not. You've selected that if one file is modified and the other isn't then the modified file replaces the unmodified file.

As you can see the problem here is that you have lost the changes made to the right file. This was because in step 4 SyncBackSE replaced the old file details with the new ones. If you choose to ignore changes when you cannot be prompted then it would work differently:

1. You have the same file, file.txt, in the left folder (**C:\Left\file.txt**) and the right folder (**C:\Right\file.txt**).
2. You modify the left file and also modify the right file.
3. The profile is run unattended. Both files have been modified and you've chosen to be prompted in both files have been modified. However, the run is unattended so the files are skipped because you cannot be prompted.
4. **After the profile has finished SyncBackSE does not update the SmartSync database for those files that were skipped because you could not be prompted.**
5. Later you modify the left file.
6. The profile is run attended, i.e. you can be prompted. SyncBackSE will now prompt you about the file because it still has the old details for a previous attended run, and because of this it sees that both the left file and right file have been modified.

The example above highlights that the update (or not) to the SmartSync database (in step 4) is the reason for the differences.

Renaming Case Changes

What does a **change in case** actually mean? Case means upper case or lower case. A file called **abc**, for example, has the same name but different case from a file called **ABC**. In most situations it doesn't matter if the case is different. For example, although Windows keeps the case of a filename it doesn't care if a file is called **abc** or **ABC**. To Windows, and all programs running in Windows, they are treated as the same name. On Windows you cannot have a folder with a file called **abc** and a file called **ABC**. Windows is not case sensitive.

However, in some situations the case does matter and is important. For example, if files are being stored on a UNIX or Linux system, e.g. via an FTP server, then a file called **abc** is different from a file called **ABC**. A directory could contain both a file called **abc** and a file

called **ABC**. Cloud systems are usually case sensitive (Amazon S3 and Microsoft Azure are) so would act the same way as an FTP server on a UNIX/Linux system.

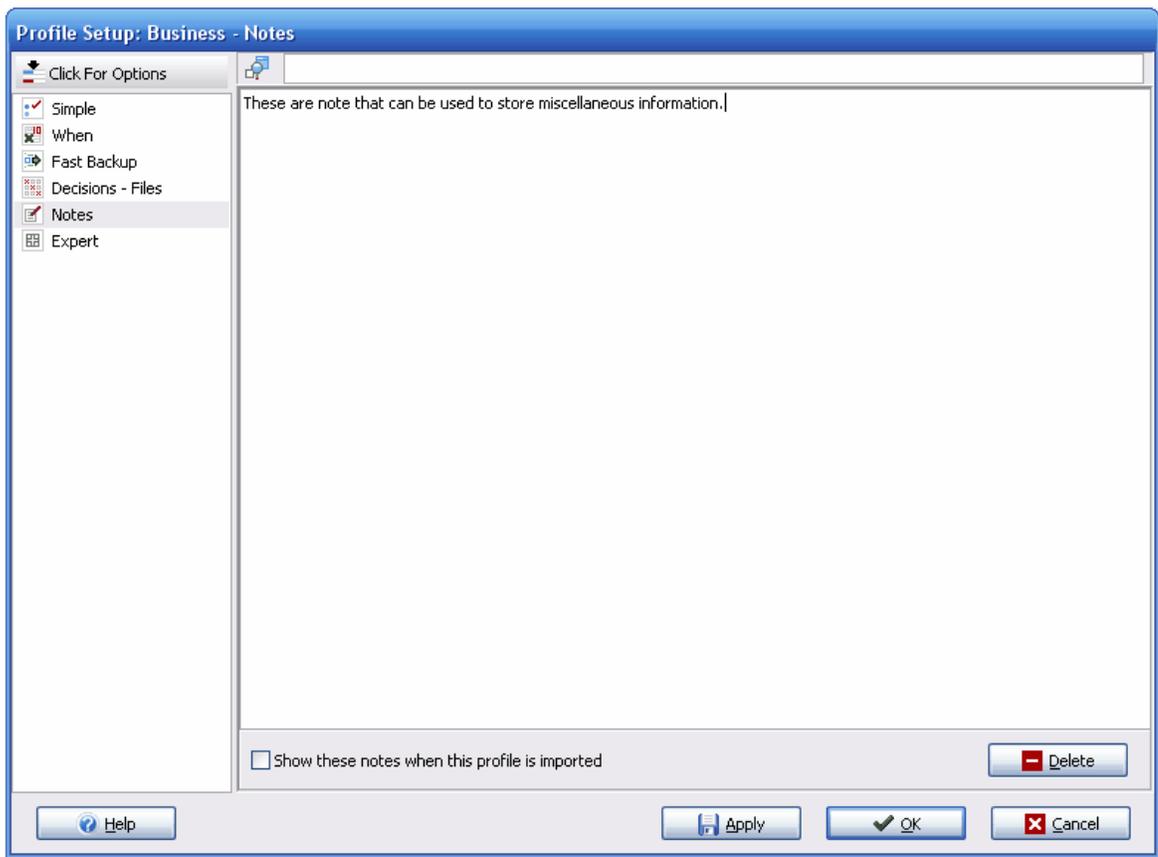
When SyncBackSE gets a list of files it will check to see if there are files with the same name but different case. If so, an error is recorded in the log file because SyncBackSE can only use one of those files so the others must be skipped and ignored. For example, if there are files called **abc**, **ABC**, and **Abc** all in the same folder then two of them will be ignored and only one of them used. Which is used? That cannot be determined or defined. SyncBackSE will use the first one it finds, but the order of the list of files it receives is often system dependent and could be random.

3.2.10 Notes

Easy Mode/Expert Mode: Notes

In some situations you may want to record some free-format notes about a profile, e.g. what the profile does, what needs to be done before the profile is used, etc. This settings page lets you enter those notes and optionally have them displayed when the profile is imported. Note that the notes are rich-text, which means you could copy & paste the notes from Microsoft Word, for example, and it will retain the font styles, sizes, colors, etc. Images cannot be used. The Notes feature is not intended to be a full blown text editor and is intended for entering a few simple notes.

- **Show these notes when this profile is imported or copied:** If this checkbox is ticked, and the profile is exported and imported into another installation of SyncBackSE, then this note is displayed. The note will also be displayed if the profile is copied. This is useful when distributing profiles and you want to tell the user what the profile does or if they need to change something before using it, for example. Note that if a profile is imported in unattended mode then the note is not displayed.



3.3 Expert Mode

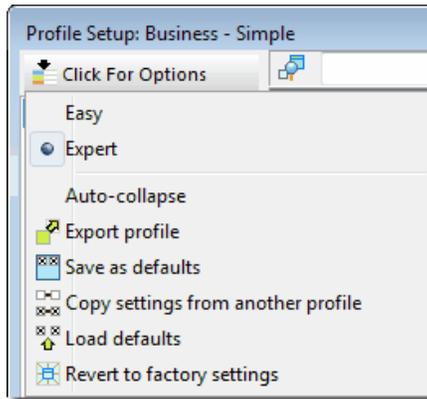


Expert Mode Overview

To modify a profile either use the 'Modify' button on the toolbar, or select 'Modify' from the 'Profiles' main menu located at the top of the program window. Spend time getting to know what options are available under the many pages in the Profile Setup window.



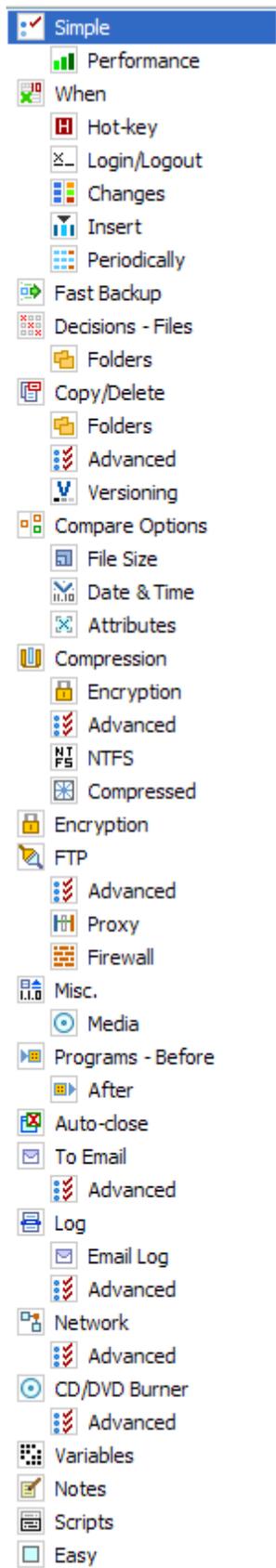
SyncBackSE provides two modes to view and change your Profiles: Easy and Expert. Easy mode presents far fewer options to modify your Profile making the choices you make less complex than in Expert mode. To access the Expert mode, when modifying a profile, you'll need to select the 'Expert' mode item located in the 'Click For Options' menu.



Click For Options Menu

The Click For Options menu allows you to:

- Select Easy or Expert mode
- Save profile settings as a default profile
- Copy profile settings from an existing profile
- Load defaults
- Revert to factory settings

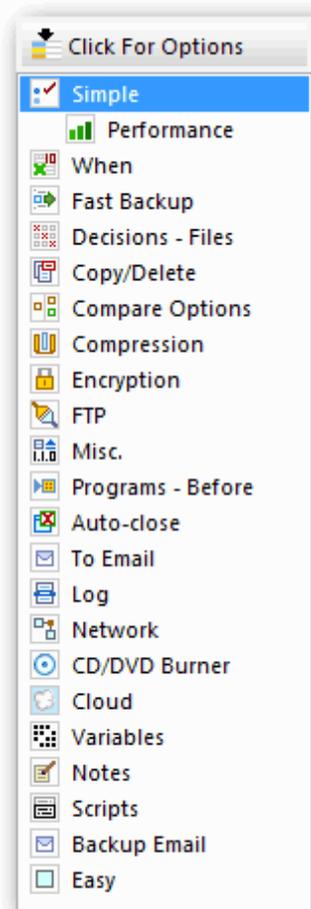


Expert Options

The left hand Options Tree contains a range of settings and choices that provide a great deal of flexibility in the way you can perform and control tasks, many of which are not available in the default Easy Mode.

Spend time getting to know what options are available.

The screenshot on the left shows all the options in Expert mode that are available. In practice only the selected branch of the Option Tree will show its associated branch:



Profile Settings in Expert Mode

The following Settings pages detail the options available for each settings tab (some of the headings are also available in Easy Mode):

Simple (also available in Easy Mode)

Sub-directories and Files (also available in Easy Mode)

Simple, Performance

When (also available in Easy Mode)

When, Hot-key

When, Login/Logout

When, Changes

When, Insert

When, Periodically

Fast Backup Settings (also available in Easy Mode)

Decisions - Files (also available in Easy Mode)

Smart Synchronization (also available in Easy Mode)

Decisions, Folders

Copy/Delete

Copy/Delete, Folders

Copy/Delete, Advanced

Compare Options

Compare Options, File Size

Compare Options, Date & Time

Compare Options, Attributes

Compression

Compression, Encryption

Compression, Advanced
Compression, NTFS
Compression, Compressed
Encryption
FTP
FTP, Advanced
FTP, Proxy
FTP, Firewall
Misc.
Misc., Media
Programs - Before
Programs, After
Auto-close
Email
Email, Advanced
Log
Log, Email
Log, Advanced
Network
Network, Advanced
CD/DVD Burner
CD/DVD Burner, Advanced
Variables
Notes (also available in Easy Mode)
Scripts
Backup Email

3.3.1 Simple, Performance

Expert Mode: Simple, Performance Settings

Simple | Performance

This page shows you which settings may be affecting the performance of your profile. It is important to keep in mind that speed is not everything, and that what is more important is that the files are copied correctly. For example, verifying that your backup files are correct will slow down the profile but will guarantee your backup files are not corrupted at backup time. The **safe copy** option is enabled by default for most profiles and it is strongly recommended that you do not switch off this option just to slightly reduce the backup time.

You can jump directly to the appropriate settings page by simply clicking on the items listed in the "**The following settings are slowing down the profile:**", etc., sections.

3.3.2 Simple, History

Expert Mode: Simple, History

Simple | History

This page shows you the history of a profile, e.g. when it was run, where it was run, the result, etc. This is similar to the log files except it's provided in table form. The history is also recorded separately from the log files and uses less disk space, so you may want to keep a longer history. A profiles history is also sent to the SyncBack Management Service (🏠 PRO Pro version only) to enable remote monitoring of profiles. Note that no history is kept of simulated runs.

If you right-click on the history then a pop-up menu appears. Using this pop-up menu you can choose which columns to show or hide. The history is always ordered from most recent (on the left) to oldest (on the right).

Scripts (🏠 PRO Pro version only) have access to most of this information via the SBHistory object.

- **Maximum run history to keep:** The number of profile runs to keep a history of. The default is 50 and the maximum is 500.

3.3.3 When, Hot-key

Expert Mode: When, Hot-Key Settings

When | Hot-key

- **Run the profile when this hot-key is pressed:** To run the profile when a certain key combination is pressed, click this edit box and press the hot-keys you want to assign. For example, if you want to run the profile whenever you press Ctrl-Shift-P, then press those keys. Now whenever you press Ctrl-Shift-P, no matter what application you are using, and even if SyncBackSE is minimized, the profile will be run. Please note that SyncBackSE must be running for hot-keys to function (you may want to configure SyncBackSE to start automatically when you login to Windows). To remove a hot-key, click the hot-key edit box and press the Backspace key. If you try to use a hot-key that is already being used (either by another profile or as a hot-key in another application) then the hot-key will be set to **None**.
- **Run unattended, i.e. do not prompt me:** If a profile is run via the hot-key, by default it is run attended, i.e. dialog box and prompts related to the profile will be displayed when required. If this option is ticked then the profile will be run silently without any prompting. Note that if this is a group profile then all the profiles in the group will be run attended or unattended, i.e. for a group this setting overrides the profiles setting.

3.3.4 When, Login/Logout

Expert Mode: When, Login/Logout Settings

When | Login/Logout

- **Run this profile on Windows shutdown/logoff:** If this option is enabled, the profile will run when you shutdown, restart, or logoff from Windows. Please note that when using Windows XP, or 2003, then SyncBackSE cannot differentiate between a shutdown and a restart (there is no problem with logoff). So after the profile has run, the computer will shutdown even if you selected to restart, which is the default. This is due to a limitation in the way Windows informs applications of a Windows shutdown. You can change this via the Global Options settings. This is not an issue with Windows Vista, Windows 7, 2008 R2, or newer versions of Windows. If you wish to be prompted before profiles are run on shutdown, restart, or logoff from Windows, then there is a Global Options setting to enable this. Also, by default, SyncBackSE will warn you there are profiles set to run on shutdown/logoff when you exit SyncBackSE. This is because SyncBackSE must be running at the time of logoff or shutdown for the profiles to be run.
-  Starting with Windows Vista, Microsoft introduced many restrictions on how programs can react to and handle the shutdown or restart of a computer. Due to these restrictions Run Before and Run After programs will silently fail and not even start if the profile is set to run on shutdown/logoff and the computer is shutdown or restarted (the programs will still be run as per normal if its a logoff).

- **On shutdown/logoff run unattended, i.e. do not prompt me:** If this option is ticked then the profile will be run silently with any prompting. Note that if this is a group profile then all the profiles in the group will be run attended or unattended, i.e. for a group this setting overrides the profiles setting. By default it is run attended for pre-Vista versions of Windows and unattended for Vista and newer. Due to changes introduced in Windows Vista, a profile set to run on shutdown/logoff must be run unattended.
- **Run this profile when I login to Windows:** On Windows XP, or 2003, a shortcut to SyncBackSE is created in the Windows Startup folder so when you login the profile will be run. For Windows Vista and later a special entry is created in the Windows Task Scheduler (this avoids receiving UAC elevation prompts). Note that a profile set to run on login will be run unattended. Due to the nature of this setting it cannot be set to be on by default. However, unlike in earlier versions of SyncBackSE this setting can now be exported/imported and copied to another profile.

3.3.5 When, Changes

Expert Mode: When, Changes Settings

When | Changes

A profile can be configured to watch for changes to files, and when any changes are made, the profile will be run. There are some important points to keep in mind:

1. SyncBackSE must be running for it to be able to detect changes. You may want to configure SyncBackSE to start with Windows (see **Preferences -> Global Settings**).
2. If you make changes to those files while SyncBackSE is not running it will not detect them and so when you start SyncBackSE it will not run the profile. Because of this you may still want to schedule the profile to run periodically.
3. Changes cannot be detected on FTP, email servers, or cloud servers. It will probably also fail to detect changes on NAS devices that are not running Windows (many NAS devices use a version of Linux) and on networked drives that are not running Windows.

Instead of detecting changes in files, you may want to have the profile run when a program closes. For example, if you are editing files then you may instead want SyncBackSE to backup the files as soon as you close the editor software. Having SyncBackSE backup files every time they are changed may not be practical in some cases.

- **Run this profile when any files or directories are changed on source:** If this option is enabled then the source/left folder will be watched, and if any changes are made to any files (including files in sub-folders), or if any files are deleted, then the profile will be run.
- **Run this profile when any files or directories are changed on destination:** As per

the above setting but the destination/right is watched. If you using a backup profile then there is no point watching the destination for changes. Also note this option is not available with Fast Backup profiles.

- **Run interactively, i.e. prompt me if required:** If a profile is run when changes to the files are made, by default it is run unattended, i.e. dialog box and prompts related to the profile will not be displayed. If this option is ticked then you will be prompted as and when required.
- **Wait a number of seconds for no changes before running the profile:** A common problem with running a profile when files are changed is that there is no way for SyncBackSE to know when a program has finished writing to or updating a file. For example, a video editing program may take several seconds to save the changes you've made to a video. In that case SyncBackSE will run once it sees the video file being updated, but the video editing program may not have finished saving the file before SyncBackSE starts copying that file. To avoid these kinds of problems you can configure SyncBackSE to not start the profile until there have been no changes for a certain number of seconds. For example, a program may need to update many files before it exits. Updating all those files may take several seconds. Let's say you set this profile setting to 5 seconds. If the program updates one file then SyncBackSE will see the change but won't yet run the profile. The program may then update another file a couple of seconds later. SyncBackSE will detect the change but still won't run the profile because changes were made less than 5 seconds after the previous change. Once the program has saved all its files, and 5 seconds have passed without any file updates, SyncBackSE will then run the profile.



Important: this delay starts from the last alert by Windows of a change (and is reset/zeroed by any fresh alert). But if the delay is set to 5 seconds, for example, and the last alert by Windows of a change was 5 seconds ago, the profile will then start. If that last change is still ongoing (video program is still writing after 5 seconds) then an error may occur as the file is still being written to. It is recommended that you set the delay time to exceed the duration of the slowest/longest change-event anticipated.

3.3.6 When, Insert

Expert Mode: When, Insert Settings

When | Insert

A profile can be configured to run whenever an external device is attached, e.g. a USB key is inserted. If Windows is assigning a random drive letter to the device when inserted please use the %LABEL% or %SERIAL% Variables. If you use these options you may want to configure SyncBackSE to start automatically when you login to Windows.

- **Run this profile when...:** This set of options enables this profile to be automatically run whenever an external device is connected or a CD/DVD, USB memory key, etc. is attached to your computer. You can configure it to be broad, e.g. any device into any

drive, or very specific, e.g. a drive with a specific label, serial number, and with a certain drive letter. You can use Windows environment variables in the label and serial settings, but you cannot use user defined profile variables. Note that there important differences between volume serial numbers and hardware serial numbers. See the HWSERIAL description in the Variables section for more details.

- **Run unattended, i.e. do not prompt me:** If a profile is run via a device insert, by default it is run attended, i.e. dialog box and prompts related to the profile will be displayed when required. If this option is ticked then the profile will be run silently with any prompting. Note that if this is a group profile then all the profiles in the group will be run attended or unattended, i.e. for a group this setting overrides the profiles setting.

3.3.7 When, Periodically

Expert Mode: When, Period Settings

When | Periodically

Running profiles in the background is similar to running profiles via the Windows Task Scheduler, except SyncBackSE must be running for background profiles to run. Also background profiles usually run much more frequently than scheduled tasks, e. g. every 30 minutes. If you use these options you may want to configure SyncBackSE to start automatically when you login to Windows.

For example, you could create and configure a profile to run in the background every 30 minutes that makes a backup of the documents you are working on. This helps ensure that you lose the least amount of work possible if, for example, there was a power cut. For more details see the Automating SyncBackSE section.

- **Run this profile every...:** This is the interval at which the profile will be run, e. g. every 30 minutes.
- **Run interactively, i.e. prompt me if required:** If a profile is in the background by default it is run unattended, i.e. dialog box and prompts related to the profile will not be displayed. If this option is ticked then you will be prompted as and when required. Note that if this is a group profile then all the profiles in the group will be run attended or unattended, i.e. for a group this setting overrides the profiles setting.
- **Warn me when exiting SyncBackSE...:** If this option is enabled, and you exit SyncBackSE while a background profile is waiting to run, then you will be prompted as a reminder.

3.3.8 When, Time Limit

Expert Mode: When, Time Limit

When | Time Limit

In some cases, you may want to limit the amount of time a profile can run for. There are some important points to keep in mind when using this feature:

- The time limit setting is ignored for simulated runs and when the profile is run in restore mode.
- It cannot be guaranteed that the profile will be terminated within the time limit. For example, if the user is being prompted (e.g. the Differences window is being displayed) then the profile is only terminated after the window is closed. Also, during some situations the profile cannot be terminated, e.g. if there are network issues.
- The time limit includes any paused time. For example, if the profile is run as part of a group (run sequentially, which is the default) then the profile is started paused and only continues once the previous profile in the order has finished.
- If a profile is stopped because it has reached its time limit, and is run as part of a group (run sequentially), then all the other profiles in the group will also be stopped (which is the same as if the profile had been stopped manually).
- When a profile is part of a group then the earliest time limit is used. For example, if a profile has a time limit of 20 minutes, but the group it is in (and being run as a part of) has a time limit of 10 minutes, then the profile has a time limit of 10 minutes and not 20 minutes. This makes sense because the profile is being run as part of a group, and the group has a time limit, so the group itself will be terminated before the profile will be.
- **Stop the profile if it runs for more than...:** If enabled, this is the maximum amount of time that the profile can run for.

3.3.9 When, Programs

Expert Mode: When, Programs

When | Programs

A profile can be configured to run whenever another program starts or exits (closes). For example, you may want SyncBackSE to make a backup of your documents as soon as your word processor is closed. It can be configured to run interactively or silently. Also, you can have it monitor several programs.

This profile settings page can use and create shared settings.

- **Run this profile when any of the following programs start:** The list-box lists all the programs that will trigger the profile if they start. To add to the list click the **Add** button. You can use variables, but you cannot use variables from a parent group. To remove

one or more entries from the list first select them and then click the **Remove** button.

- **Run interactively, i.e. prompt me if required:** By default a program is run unattended when one of the programs specified starts, i.e. dialog box and prompts related to the profile will not be displayed. If this option is ticked then you will be prompted as and when required. Note that if this is a group profile then all the profiles in the group will be run attended or unattended, i.e. for a group this setting overrides the profiles setting.
- **Run this profile when any of the following programs stop:** The list-box lists all the programs that will trigger the profile if they stop/close/exit. To add to the list click the **Add** button. You can use variables, but you cannot use variables from a parent group. To remove one or more entries from the list first select them and then click the **Remove** button.
- **Run interactively, i.e. prompt me if required:** By default a program is run unattended when one of the programs specified stops, i.e. dialog box and prompts related to the profile will not be displayed. If this option is ticked then you will be prompted as and when required. Note that if this is a group profile then all the profiles in the group will be run attended or unattended, i.e. for a group this setting overrides the profiles setting.

3.3.10 Decisions, Folders

Expert Mode: Folders Settings

Decisions - Files | Decisions, Folders

These settings are related to how **empty** folders are created & deleted and how case changes are handled. Note that if a file is created then obviously its folder must also be created (to contain the file). By default it is handled automatically (based on your file decisions). If you are using a Smart Synchronization profile then you are given different options on what to do when a directories case is changed.

What to do if a directory exists on source/left but not on destination/right

- **Create directory on destination/right:** Folders only on the source/left will be created in the destination/right.
- **Delete directory from source/left (if empty):** If the folder is empty it will be deleted from the source/left. Note that the folder will not be deleted if it has files in it, including any hidden files or file versions. You can configure which files SyncBackSE can automatically delete to make a folder empty on the Copy/Delete Folders page.
- **Delete directory from source/left only if a file is moved/deleted from it:** In some cases you may only want a folder to be deleted if a file was deleted/moved from it. As above, the folder must be empty for it to be deleted.
- **Prompt me (skips folder if run unattended):** You will be prompted on what action to

take.

- **Do nothing:** Nothing will be done.
- **Automatic:** SyncBackSE will decide the best action to take based on your profile type. This is the default.

What to do if a directory exists on destination/right but not on source/left

- **Create directory on source/left:** Folders only on the destination/right will be created in the source/left.
- **Delete directory from destination/right (if empty):** If the folder is empty it will be deleted from the destination/right. Note that the folder will not be deleted if it has files in it, including any hidden files or file versions. You can configure which files SyncBackSE can automatically delete to make a folder empty on the Copy/Delete Folders page.
- **Delete directory from destination/right only if a file is moved/deleted from it:** In some cases you may only want a folder to be deleted if a file was deleted/moved from it. As above, the folder must be empty for it to be deleted.
- **Prompt me (skips folder if run unattended):** You will be prompted on what action to take.
- **Do nothing:** Nothing will be done.
- **Automatic:** SyncBackSE will decide the best action to take based on your profile type. This is the default.

Automatic

The decision SyncBackSE makes for the Automatic option depends upon the type of profile:

- **Backup (copying files from source/left to destination/right):** Folders only in the source/left will be created in the destination/right. Folders only in the destination/right are deleted (if empty).
- **Restore (copying files from destination/right to source/left):** Folders only in the destination/right will be created in the source/left. Folders only in the source/left are deleted (if empty).
- **SmartSync:** Folders only in the source/left will be created in the destination/right, and vice-versa. However, if a folder was deleted from the source/left then it will be deleted from the destination/right, and vice-versa.
- **Mirror Right (copying files from source/left to destination/right, deleting files only in destination/right):** Folders only in the source/left will be created in the destination/right. Folders only in the destination/right are deleted (if empty).
- **Mirror Left (copying files from destination/right to source/left, deleting files only**

in source/left): Folders only in the destination/right will be created in the source/left. Folders only in the source/left are deleted (if empty).

- **Old-style Synchronize (copying files to and from source/left and destination/right):** Folders only in the source/left will be created in the destination/right, unless files only in the source/left are being deleted. If files only in the source/left are being deleted then folders only in the source/left are deleted (if empty). Folders only in the destination/right will be created in the source/left, unless files only in the destination/right are being deleted. If files only in the destination/right are being deleted then folders only in the destination/right are deleted (if empty).

What to do if the case of the directory names do not match

See the Renaming Case Change section for important information. A different set of options are available if this is a SmartSync profile. See the section below for SmartSync options when a directories case changes.

- **Rename directory on source:** The directory is renamed on the source so that it becomes the same as the destination directory name.
- **Rename directory on destination:** The directory is renamed on the destination so that it becomes the same as the source directory name.
- **Prompt me (skips folder if run unattended):** You will be prompted on what action to take. Note that if SyncBackSE is run from the command line, or from the Windows Task Scheduler, then no prompt will appear and the changes in directory case will be ignored.
- **Do nothing:** Nothing will be done and the differences in directory case will be ignored. This is the default action.
- **Automatic:** The appropriate directory is renamed. Which directory is renamed? It depends on what type of backup location you are using and the type of profile you have. If you are using FTP, compressing to a single Zip file, burning to CD/DVD, an email server, or the cloud, then it will rename the directory stored on a disk or network drive. If you are copying to and from a disk or network drive then it will rename the destination directory if it's a backup or mirror to the destination. If it's a backup or mirror to the source then it will rename the source directory.

What to do if...

See the Renaming Case Change section for important information. These options are only available for SmartSync profiles. For other profile types see the section above (What to do if the case of the directory names do not match).

-  **...the case of a folder on source/left has been changed (unchanged on destination/right):** The case of the source folder name has been changed so that it is no longer the same as the destination. For example, the source folder was previously called **abc** but has been renamed to **ABC**. See the Renaming Case Change section

below for important information.

-  **...the case of a folder on destination/right has been changed (unchanged on source/left):** The case of the destination folder name has been changed so that it is no longer the same as the source. See the Renaming Case Change section below for important information.
-  **...the case of a folder on both source/left and destination/right has been changed:** The case of both the source and destination folder name has been changed and they are not the same. See the Renaming Case Change section below for important information.
- **When prompting, and the run is unattended, ignore the changes:** For some of the decisions you may have chosen to be prompted. However, if the profile is being run unattended then you cannot be prompted. In this case the folder change is ignored and a warning is recorded in the log file. With a SmartSync profile you may want SyncBackSE to ignore the changes. See the Ignoring Changes section for important details on what this means.



If you get the error "**Failed to rename to [new folder name]: The process cannot access the file because it is being used by another process**" it is because another process is currently using the folder that needs to be renamed. For example, a program may be using a file that is in the folder that is needs to be renamed. You must close the process/program that is using that folder.

3.3.11 Copy/Delete

Expert Mode: Copy and Delete Settings

Define how SyncBackSE will handle the copying, deleting, and moving of files.

SyncBackSE can optionally use the exact same routines for copying, deleting, and moving files that Windows Explorer uses. This ensures that files are copied as expected, e.g. the file attributes are also copied which provide benefits such as putting deleted files in your Recycle Bin.

File copying method: Three different methods of copying (and deleting and moving) files are provided to allow for maximum compatibility and flexibility:

- **Standard Windows file copying:** This is the default method for copying files. It is the most efficient and quickest way to copy files. Not using the shell (see next item) may fix network problems, especially when using Novell networks.
- **Windows Explorer method of file copying:** The Windows Shell (Explorer) is asked to

copy, delete, and move files on behalf of SyncBackSE. When using this method you have the benefit of moving deleted files to the recycle bin. Using the shell may cause problems when SyncBackSE is used on non-standard versions of Windows, e.g. BartPE.

- **Backup read/write file copying:** This method (which can only be used by Windows users who have the necessary backup access rights – if not then the **Standard** method is silently used) can be used to copy files that the user has no access rights to. So if you are receiving **Access denied** error messages you may want to try this method. Note that the backup copy method cannot be used with Zip files, FTP, backup to email, or backup to CD/DVD/ISO (the Standard method will be used silently).

Summary of file copying methods			
	Standard	Explorer	Backup
Copies extended attributes (1)	Yes	Yes	Yes
Copies OLE structured storage (1)	Yes	Yes	Yes
Copies NTFS file system alternate data streams (1)	Yes	Yes	Yes
Copies file attributes	Yes	Yes	Yes
Copies security attributes (2)	Yes (3)	Yes	Yes (3)
Copies encrypted files (5)	Yes	Yes	Yes (4)
Can copy files user has no access to	No	No	Yes (4)
Deleted files can be moved to Recycle Bin (8)	No	Yes (6)	No
File deletions or overwrites can be confirmed first (8)	No	Yes (6)	No
Directory creation can be confirmed first	No	Yes (6)	No
Copy Performance	Fast	Fast	Slow
Copy progress feedback (7)	Yes	No	Yes

(1) The destination must be an NTFS formatted volume.

(2) The destination must be an NTFS formatted volume, and if not a local volume then the security attributes may not be copied.

(3) The destination must be an NTFS formatted volume, and security attributes are only copied if the profile is configured to copy file security permissions. By default they are not copied.

(4) The Standard method is automatically used to copy NTFS encrypted files.

- (5) The copy may not be encrypted, e.g. if the destination is not NTFS.
- (6) Not enabled by default.
- (7) This means the copy progress is shown, meaning you can abort a file copy. Without progress feedback the profile can only be aborted after a file has been copied, not during the copy.
- (8) If you would like to keep files that are deleted or replaced then consider using versioning.

Shell File Copying & Deleting: The following options are only displayed if the 'File copying method' is set to **Windows Explorer method of file copying**:

- **Move deleted files to recycle bin:** If a file is deleted from a local drive (not a removable drive or network drive) it can be moved to the recycle bin instead of being deleted. This is identical to what would happen if you deleted the file using Windows Explorer.
- **Confirm file deletions, replacing files, etc.:** When a file is to be deleted or replaced by another file then you can be prompted on whether you want SyncBackSE to delete or replace the file. An important point to remember is that if you are using the recycle bin (see the previous item) then you will not be prompted when a file is deleted (instead it is silently moved to the recycle bin). Also, if you are running the profile in unattended mode, e.g. from the Windows Scheduler, then you will not be prompted and the file will be deleted or replaced. If you would like to keep files that are deleted or replaced then consider using versioning.
- **Confirm directory creation:** If a new directory must be created then you can be prompted on whether you want the directory to be created or not. If you are running the profile in unattended mode, e.g. from the Windows Scheduler, then you will not be prompted and the directory will be created. This option is only available if "Do not display a progress dialog box" is unchecked and "Display error messages and prompts" is enabled.
- **Display error messages:** If an error occurs when a file is copied or deleted then you can choose to be prompted with an error message. Note that these error messages are also recorded in the profiles log file. If you are running the profile in unattended mode, e.g. from the Windows Scheduler, then you will not be prompted with any error messages.
- **Do not display a progress dialog box:** When copying large files, using a slow network connection, or slow storage devices, it can sometimes take a long time to copy a file. If this option is enabled then the standard Windows file copy progress dialog box will be displayed (it is only displayed if a file copy will take more than a few seconds). The benefit of this dialog box is that you can cancel the file copy, for example it may be taking too long, and you can see how long it will take to copy the file. If you are running the profile in unattended mode, e.g. from the Windows Scheduler, then the progress dialog box will not be displayed. Please note that on

some systems disabling this option may drastically reduce performance.

- **Verify that files are copied correctly:** After a file is copied, SyncBackSE can check to guarantee that the newly created file is identical to the original file.



Enabling this option can significantly increase the time taken for a profile to run.

Note that this option will not work if you are using an FTP server that does not support the XCRC extension (the log will contain the warning message "**The FTP server does not support hashing**").

When moving files, if the copy cannot be verified, then the source file is not deleted, i.e. it becomes a copy not a move. The verification option is best used with the "**Make safe copies**" option also enabled as it ensures that your backup doesn't contain corrupted files. If both these options are enabled then when a file is copied it is first copied to a temporary file, i.e. the destination file is not yet overwritten. If the temporary file does not match the original file then the temporary file is deleted and an error is recorded. Without the safe copies option you will know the copy is corrupt but by then it is too late as the file has already been replaced with a corrupted file.

- **Warn me (abort profile if unattended) if 100% or more of my files are going to be deleted due to my settings :** In some situations you may have wrongly configured your profile or made a mistake. For example, you may have set the source folder incorrectly to an empty folder and configured your profile to delete all files in the destination that are only in the destination. In this case when you run the profile all your destination files would be deleted. This setting is to avoid situations like this. You may also want to make sure that no files are going to be deleted. If SyncBackSE sees that it is going to delete the specified percentage of your files then it will automatically abort the profile and do nothing (if the run is unattended). If the profile is being run attended then a warning message will appear so that you can choose to abort or not. This setting is enabled by default and set at 100%, i.e. it will only abort/warn if **all** your files are going to be deleted. If you want to abort or be warned if any files are to be deleted then set the value to 0%. Otherwise, it will only abort/warn if the specified percentage (or more) of your files are going to be deleted.
- **Automatically create the source/left and destination/right folders if they do not exist:** By default the source/left and destination/right folders will be created automatically if they don't exist. However, in some situations you may not want this, e.g. if using a SmartSync profile, and instead want the profile to fail if either folders don't exist.
- **Improve performance by using ExtremeCopy to copy large files:**  **Pro only.** If enabled, any files that are 100MBytes or larger in size are copied using the special **ExtremeCopy** method. In most cases this will increase the file copy speed and so reduce the profile run time. Note that this is of no benefit when using FTP, compression, email, the cloud, or burning to CD/DVD and is not available when using the **Backup Copy** file copying method. It is only used when copy files from one drive to another (which includes external drives, internal drives, and over the network to drives on other computers or devices). Also, when a file is being copied using ExtremeCopy then

bandwidth throttling settings are ignored. Only one file at any one time can be copied using ExtremeCopy, so if two or more profiles are running at the same time then only one of them will be used ExtremeCopy at any one time.

How much faster is ExtremeCopy? (🏠 Pro only)

A test profile that copied 4 files (one 5GByte file, two 400MByte files, and a 500MByte file) was used to test the benefit of using ExtremeCopy. To reduce interference in the times the system was rebooted after each profile run, all other programs were terminated, and no other computers or programs were using the network or drives at the time. The results were:

1. Copying the files from one partition to another partition on the same internal SATA drive: the non-ExtremeCopy profile took 287 seconds, whereas the ExtremeCopy profile took 244 seconds which is 43 seconds faster.
2. Copying the files from an internal SATA drive to an external drive connected via Firewire: the non-ExtremeCopy profile took 241 seconds, whereas the ExtremeCopy profile took 219 seconds which is 22 seconds faster.
3. Copying the files from an internal SATA drive to a network path (100Mbps wired LAN connection): the non-ExtremeCopy profile took 1,206 seconds, whereas the ExtremeCopy profile took 1,174 seconds which is 32 seconds faster.

Keep in mind that many factors will affect the performance of files being copied, e.g. CPU load, RAM load, network load, drive load, Windows file caching, etc.

3.3.12 Copy/Delete, Folders

Expert Mode: Copy/Delete, Folders Settings

Define how SyncBackSE will manage folders (directories).

- **Copy sub-directory and file security permissions (only valid for NTFS):** If ticked then NTFS folder security is copied when new directories are created. Note that if the security is changed on a folder then the new security is not copied over. NTFS file security will also be copied with files.
- **Copy sub-directory attributes and creation date (only when new directories are created):** If ticked then folder attributes and the creation date are copied when new directories are created. Note that if the attributes or creation date are changed on a folder then the new settings are not copied over.
- **Delete all empty directories in destination/right:** If ticked then all empty directories in the destination/right will be deleted at the end of the profile run. Note that a folder will not be deleted if it has files in it, including any hidden files or file versions. You can configure which files SyncBackSE can automatically delete to make a folder empty with the “If a folder cannot be deleted because it’s not empty...” setting below.
- **Do not delete the empty directories if the profile run fails:** By default empty folders

are not deleted in the profile run was not a success.

- **Delete all the empty directories on source/left:** If ticked then all empty directories in the source/left will be deleted at the end of the profile run. Note that a folder will not be deleted if it has files in it, including any hidden files or file versions. You can configure which files SyncBackSE can automatically delete to make a folder empty with the “If a folder cannot be deleted because it's not empty...” setting below.
- **Do not delete the empty directories if the profile run fails:** By default empty folders are not deleted in the profile run was not a success.
- **Ignore NTFS junction points (reparse points):** If ticked then junction/reparse points are ignored. This is the default and recommended.
- **When the desktop.ini file (used by Explorer) is copied configure the folder it is copied to use it:** The desktop.ini file is a special file created and managed by Windows Explorer. It helps define what a folder looks like when viewed in Windows Explorer. It is only used if the folders attributes are set correctly. You can tell SyncBackSE to automatically set the correct folder attributes if a desktop.ini file is copied to it. If you have desktop.ini filtered out, then desktop.ini is shown in yellow in the filters window. Note that the desktop.ini file does not entirely define the appearance of a folder in Windows Explorer.
- **If a folder cannot be deleted because it's not empty...:** Only empty folders can be deleted. Windows Explorer often creates special hidden files in folders that contain settings related to that folder but contain no user created information. This setting defines which files can be deleted to make a folder empty. By default the **thumbs.db** and **desktop.ini** files will be deleted, along with temporary SyncBackSE files. Note these files are only deleted if they are the only files in the folder. Wildcards can be used, e.g. *.tmp

3.3.13 Copy/Delete, Advanced

Expert Mode: Copy/Delete, Advanced Settings

- **Automatically update shortcuts when they are copied or moved:** SyncBackSE can be configured to automatically change shortcuts that are copied so that the copy points to the correct file. For example, if your source is C:\ and your destination is D:\ and you copy a shortcut that points to C:\abc.txt then you may want it to be changed (in the destination) to instead point to D:\abc.txt. In this example you would set the root folder for shortcuts on the source/left to C:\ and the root folder for shortcuts on the destination/right to D:\. In general the shortcut root folder is the same the source and destination folders, except when you are copying to UNC paths or network drives. In that case the shortcut root folder should be based on the drive on that remote computer.
- **Make safe copies (copy files using a temporary filename and then rename the file on success - reduces performance):** When this option is ticked then SyncBackSE will

copy files in a two stage process: first it will copy the file and use a temporary filename, then it will then replace the destination file with the temporary copy if the copy was a success. This avoids problems where the destination file may be deleted because the original file could not be copied, e.g. because it is locked and cannot be replaced. With FTP it is actually a three stage process as it must delete the file being replaced before renaming the temporary copy. With multi-zip, safe copying is always used when creating the compressed file. This option is enabled by default and is recommended. If you have enabled versioning, on either the source or destination, then you cannot change this setting as safe copies must be used when versioning is used.



We strongly recommend using the safe copy feature to avoid corrupting your backup files due to unpredictable failures or errors. However, there may be cases where performance is the most important factor. If you are copying thousands of files, especially small files, or you are copying over a network (this includes FTP and the cloud) then switching off safe copies can significantly reduce the backup time. Note that if you are using versioning then you cannot switch off safe copies (as versioning requires its use).

- **Prompt to retry if a file is locked or cannot be copied:** If this option is enabled then SyncBackSE will prompt you if it cannot copy or delete a file, e.g. because it is locked, no access rights, etc. The prompt gives you the opportunity to close the program that has the file locked so the file can be copied or deleted by SyncBackSE. If you are running the profile in unattended mode, e.g. from the Windows Scheduler, then no prompt is made and the file will be skipped (an error message will be recorded in the log file). There is another option in SyncBackSE to close certain programs before a profile is run (see the Auto-close page).



Note that if you've also enabled the option to replace/delete after a reboot then you will not be prompted.

- **If a file cannot be replaced/deleted because it is locked then replace/delete it after a reboot:** If a file cannot be replaced/deleted because it is locked then SyncBackSE can configure Windows so that on the next reboot the file will be replaced/deleted. This option must be used with care because it cannot be guaranteed that the file will actually be replaced/deleted and there is no way to know if it did or did not.



Note that if you've also enabled the option to prompt to retry if a locked file cannot be copied then you will not be prompted.

These features cannot be used to replace in-use files on FTP servers. Also, if NTFS compression is being used the replaced file will not be NTFS compressed.

- **Prompt to retry if source/left or destination/right drives do not exist:** If a drive does not exist, e.g. because it has not been connected or the network connection is not available, then a prompt will be displayed. Note that if the profile is run unattended then no prompt is displayed and the profile will fail.
- **If a file cannot be copied because of security (Access Denied) then try Backup Read/Write copy method:** The Backup Read/Write file copying method lets a user (who is a member of the Backup Operators user group) backup files that they have no access

to. When this option is enabled if a file cannot be copied because you have no access rights to it then the backup method is used to try and bypass the file security.

- **Do not replace newer files with older files:** If ticked then new files are not overwritten (replaced) by older files. This option is only available under certain conditions.
- **Force the file modification date & time to be correct (may be required when using SAMBA or network drives):** Sometimes the file system that files are copied to cannot correctly record the files last modification date & time. For example, when using SAMBA shares (or some NAS devices) there can be occasions when the file will have the current date & time as it's last modification date & time. To resolve this SyncBackSE can forcibly change the new files last modification date & time to be correct. Note that this option is not available when using compression. There is another option (see the Compare Options, Date & Time page) to ask SyncBackSE to ignore small date & time differences. This can also be useful in avoiding problems where the file system cannot accurately record dates & times.
- **Copy file creation date & times:** By default copies of files are given the current date & time as their creation date & time. If this option is enabled then the creation date & time is copied. Note that FTP servers cannot store file creation date & times.
- **Reset the archive file attribute on files once they have been copied:** When enabled the archive attribute on a file, both on the source/left and destination/right, will be cleared once the file has been copied. Enabling this option will slightly decrease performance. This option is not available when doing Fast Backups. Note that when doing a backup to CD/DVD, the archive attribute is cleared before the file is burned to disk. This means the archive attribute could be cleared even if the file failed to be saved to CD/DVD.
- **Remove the read-only attribute from copies of files (useful when copying from a CD-ROM):** In some situations, when a file is copied from a CD/DVD, then the file may automatically be marked as read-only (not by SyncBackSE, but by the file system driver in Windows). If this option is enabled then any read-only flag put on the copy of the file will be removed automatically.
- **Do not overwrite read-only files (ignored when file to be replaced is on an FTP server):** If the file to be replaced is read-only, and this option is enabled, then the file will not be overwritten.
- **Do not delete read-only files (ignored when file to be deleted is on an FTP server):** If the file to be deleted is read-only, and this option is enabled, then the file will not be deleted.
- **Do not use the Volume Shadow Copy service (VSS) to copy open files:** SyncBackSE can copy open/locked files by using the Volume Shadow Copy service that is part of Windows. However, if you cannot or don't wish to use this service then you can enable this option.
- **Copy all files from the shadow volume (snapshot):** SyncBackSE can optionally copy all files from the shadow volume (by using the Volume Shadow Copy service). This is

useful when you want all the files to be copied at exactly the same time so you have a consistent file state across the backup set. A snapshot is taken of the drive at a single point in time so that essentially the files are frozen at that point. This does not stop users and other programs from modifying and deleting those files because they are changing files on the actual volume and not the shadow volume. It is recommended that this setting be used when doing a backup to CD/DVD or in situations where you have a set of files that are related to each other, e.g. backup of a set of database files.

- **Delete ALL the files and folders in source/left before scanning for changes:** This option should be used with extreme care! It will delete all files and folders in the source/left folder before it scans for changes. **Be extremely careful with this option as you could very easily delete all your files**, e.g. if your source/left is C:\ then you will delete every single file on your C: drive. You have been warned.
- **Delete ALL the files and folders in destination/right before scanning for changes:** This option should be used with extreme care! It will delete all files and folders in the destination/right folder before it scans for changes (if the destination/right is a single Zip file then just the Zip file will be deleted). **Be extremely careful with this option as you could very easily delete all your files**, e.g. if your destination/right is C:\ then you will delete every single file on your C: drive. You have been warned.
- **Silently fail if source/left cannot be reached through the network, drive does not exist, or there is no disk:** When enabled, and the source/left drive cannot be accessed, then the profile will silently fail when run. No log file will be produced, and no failure result will be recorded. This option is useful when you have a profile scheduled to run periodically but are not always connected to the source/left.
- **Silently fail if destination/right cannot be reached through the network, drive does not exist, or there is no disk:** When enabled, and the destination/right drive cannot be accessed, then the profile will silently fail when run. No log file will be produced, and no failure result will be recorded. This option is useful when you have a profile scheduled to run periodically but are not always connected to the destination/right, e.g. you backup to a network drive via a wireless network connection. If the destination is an email server, FTP server, or a cloud service, then SyncBackSE will attempt to connect to the server (at the relevant address and port) to see if it can be accessed. In previous versions a **ping** was used instead, but often pings are blocked by routers and firewalls.
- **Copy symbolic links as-is instead of copying the file the link points to:** If enabled, and a symbolic link file is being copied, then instead of copying the file the link points to it copies the link. This means the link on the destination will point to the same file the source link points to. Because of this you may want to make sure the symbolic links are relative and not absolute. For example, let's say you have a symbolic link on the source that points to C:\abc\def.txt. When copied to the destination the destination link will still point to that same file. However, if the link was relative and instead pointed to ..\def.txt then the destination copy of that link would defer to the def.txt file on the destination. This option is only available on Windows Vista and newer, and is only used when copying from NTFS to NTFS file systems. It also cannot be used with compression or when Extreme Copy is being used.

- **Limit bandwidth usage to...:** If a value above zero is entered then when copying files the bandwidth usage is limited to what is specified (kilobytes per second). Note that there are some cases where the bandwidth usage cannot be limited, e.g. when using compression. Note that this bandwidth is used even when copying files to internal or external drives, as well as to other computers or NAS devices. There is a separate setting to limit the FTP bandwidth usage on the FTP, Advanced page.

3.3.14 Copy/Delete, Versioning

Expert Mode: Copy/Delete, Versioning

For a description of versioning see the section What Is Versioning below. Versioning cannot be used (on the destination) when compressing to a single Zip file, or when doing a backup to an email server.

- **Enable versioning on source/left/destination/right:** If enabled, versions of files will be kept in that location. You can, for example, choose to only keep versions in the destination/right.
- **Where versions files are stored...:** This setting defines where the versions files are kept. They are even kept in a sub-folder of each folder, or kept within a folder in the source/destination. Once you choose a setting it is not recommended that you change it otherwise you will no longer have access to your versions.
- **Keep a maximum of x versions:** The number of versions of a file to keep can be specified here. Note that obviously the more versions of files you keep the more disk space will be used. When the profile is next run it will automatically delete any excessive versions (starting with the oldest version). If the value is set to zero then versions are never deleted based on the number of versions.
- **Keep versions for a maximum of x days:** When a version of a file is made the date & time when the version was made is recorded. Note that this is not the last modification date & time of the file, nor is it the creation date & time of the file, it is the date & time when the version was made. Using this option you can specify how long a version should be kept. Once the version is older than the specified number of days it is automatically deleted when the profile is next run. If the value is set to zero then versions are never deleted based on their age.
- **Change Filter:** When this button is clicked you can choose what types of files are versioned or not. The filter applies to versions on the source/left and destination/right. For example, entering `*.exe` and `*\temp*` in the "Files NOT to version" field will not version any EXE files, or any files in sub-folders called `\temp\`.

Versioning is available in Expert Mode and is associated with the Copy/Delete settings.

What is versioning?

A version of a file is automatically created when:

- A file is to be replaced (a copy of the file to be replaced is made before it is replaced)
- A file is to be deleted (a copy of the file to be deleted is made before it is deleted)
- A file is to be moved (a copy of the file to be moved is made before it is moved, which is essentially the same as being deleted)

Assuming versioning is enabled, here are some examples of how it works:

- You choose to delete a file. SyncBackSE makes a version of the file and then deletes the file. At a later time you may decide that you actually wanted to keep that file. If so you can run the profile and retrieve the version and so restore the file.
- You make some modifications to a document then make a backup of it. SyncBackSE will make a version of the backup file that is about to be replaced then make the backup. At a later time you may decide that you did not want those changes. If so you can run the profile and retrieve the version and so restore the file to before the changes were made.

Where are the version files kept?

Where the versions files are kept depends on the choice you made:

- **In a hidden sub-folder of the folder that contains the original file:** The versions of files are kept in a special hidden sub-folder called **\$SBV\$**. Each folder will have this special sub-folder if there any versions of files in that folder. SyncBackSE will automatically mark the folder as hidden when it creates it. You should not rename the folder or the files inside it otherwise the versions files cannot be used. You are free to delete the folder and the version files in it (you will obviously lose those versions) but it will not affect SyncBackSE as no database of those versions files is kept (it is always built at profile run time).
- **In a hidden sub-folder of the base folder:** The versions of files are kept in a special hidden folder called **\$SBV\$** which is in the base folder. For example, if your destination directory is **X:\My Backup\Documents** then the versions folder will be **X:\My Backup\Documents\SBV\$**. You should not rename the folder or the files inside it otherwise the versions files cannot be used. You are free to delete the folder and the version files in it (you will obviously lose those versions) but it will not affect SyncBackSE as no database of those versions files is kept (it is always built at profile run time).

When deciding where to keep the versions files, please consider the following:

- **In a hidden sub-folder of the folder that contains the original file:** If you have more than one profile that is using the same folder, and is using versioning, then the advantage of choosing this option is that all the profiles will have access to the same versions. Another advantage is that you can change the base folder and not lose the versions (as long as they are still sub-folders). The disadvantage of choosing this option is that it becomes impossible for SyncBackSE to know if a directory is truly empty or not. For example, if there are versions of files in the folder, but no actual files (e.g. they've all been

deleted), then SyncBackSE does not know if the folder should be left empty or whether it shouldn't exist. This has implications for SmartSync profiles as it may cause empty folders to be created on the opposite side when you don't want them.

- **In a hidden sub-folder of the base folder:** The advantage of this option is that it removed the "empty folder" issue. This is because the versions are not stored in a sub-folder of the actual folder, so the actual folder can be deleted without affecting the versions. A disadvantage of this option is that the versions folder is pinned to the source/destination folder, so if you change the source/destination folder then you lose the versions. Also, if more than one profile is using the same folders then they will not share the versions (unless the source/destination path is the same).

Changing where to store the versions will result in losing those versions.

How to restore versions

Versions can be restored from the Differences window (or from the File Collision window). When a profile is run as a **Restore**, and versioning is used, the Differences window will automatically show skipped files. This allows you to restore old versions of files that no longer exist, and restore versions of files where there is no change.

If you wish to restore versions without using Restore (e.g. you are using a SmartSync profile and so cannot run it in Restore mode) then select the profile in the main window, hold down the **Ctrl** key, keep it pressed, then click the **Run** button. This will ensure the **Differences** window is shown. You then need to enable it to show skipped files (click the bar at the top of the Differences window to see the options).

See the Differences help page for details on retrieving versions of files.

Frequently Asked Questions

Q: Can versioning be used with Fast Backups?

A: Yes, but it can greatly reduce the performance gains you get with Fast Backup. If your profile is configured to display the Differences window then it is recommended you switch off this option as displaying the Differences window forces SyncBackSE to scan every folder to see which versions are available for each file.

Q: Can versioning be used with SmartSync profiles?

A: Yes.

Q: Can versioning be used with single zip files?

A: No.

Q: Can versioning be used with email servers?

A: No.

Q: What if I switch from not using compression to compressing each file into its own zip file (or vice-versa)?

A: You will be warned that you can no longer use the existing versions. This is because if compression is used then the old versions are also stored compressed (and encrypted, if

configured so). If no compression is used then the versions are not stored compressed.

Q: What if I switch off versioning? What happens to the versions files?

A: If a profile does not use any versioning (on the source/left or destination/right) then the old versions files will be treated like any other kind of file. If you were using versioning on both the source/left and destination/right, and then switched off versioning on one side, then the old versions on that side are ignored. For example, if you had versioning on the source/left and destination/right, and then switched off versioning in the source/left, then the old versions files in the source/left will be ignored, i.e. SyncBackSE will pretend those files do not exist.

Q: What if I change where versions are stored? What happens to the versions files?

A: You will lose access to the existing versions. You must manually delete the versions files, or configure your profile to ignore the **\$SBV\$** folders.

Q: Are the versions stored compressed or encrypted?

A: Only if the files themselves are.

Q: Why aren't the versions stored compressed?

A: Because it would slow down the profile considerably.

Q: What if I decrease the number of versions to keep, or how long they are kept?

A: When the profile is next run any excess versions will be automatically deleted.

Q: Can I choose to store the versions in a directory I choose?

A: No.

Q: What if I change the source/left and/or destination/right path? Are the versions files automatically moved?

A: No (this is the same as using variables).

Q: What if I used variables in the source/left and/or destination/right path?

A: If versions are kept in a sub-folder of where the original file actually is, then using variables has no effect, except obviously the versions files will be scattered across various folders based on the variable values.

Q: If a folder is filtered out (or unselected) does SyncBackSE still manage the versions in that folder?

A: No, because the profile is specifically configured not to use that folder.

Q: If a file is filtered out (or unselected) does SyncBackSE still manage the versions in that folder?

A: Yes. When looking at file versions it ignores any filtering or selection rules. This makes sense because the original file may no longer exist anyway, for example.

Q: Does versioning affect performance?

A: Normally it has a very small effect on performance. SyncBackSE has been developed to make versions as quickly as possible. However, there are cases where a version cannot be made quickly (i. e. the file to be replaced or deleted has to be copied, instead of being

moved, in which case it can affect performance if the file is large).



Note that if you are using Fast Backups then enabling versioning on the destination can greatly reduce the performance gains you get with Fast Backup.

Q: My log file has the error “The profile "*profile name*” was automatically disabled: reason why disabled.” What do I do?

A: What happened is that a version of the file was made (i.e. it was moved into the versions sub-folder, called **\$SBV\$**), but the copy failed. SyncBackSE then attempted to move the file back from the versions sub-folder to where it was originally. However, something went wrong and the file cannot be moved back. You must manually move the file back from the **\$SBV\$** folder to its parent folder and rename it. Once done you should re-enable the profile via the main window (right-click on the profile and select **Enable** from the pop-up menu). This situation is extremely unlikely to happen.

3.3.15 Compare Options

Expert Mode: Compare Options Settings

Fine tune the way SyncBackSE handles file change detection.

When a profile is run, SyncBackSE only copies files that have been changed and new files. It uses four different methods to check if a file is different in the source and destination:

Last file modification date & time: All files record the date & time of when they were last changed.

File size: All files record the number of bytes they contain.

Hash value: A unique value can be computed based on the contents of a file. These values can be used to check if a files contents is the same as another's.

File attributes: Files have special attributes, e.g. read-only, hidden, etc., and SyncBackSE can optionally check for changes in these attributes

You have the option to not use some of these methods, and also change how they are used.

- **Skip the Differences screen when this profile is run (it is never shown when unattended):** Whenever you run a profile, it will compare the source and destination, and display the results in the Differences window. However, you can skip this window by enabling this option. Note that the Differences window is never displayed if you run SyncBackSE with command line parameters (e.g. from the Windows Task Scheduler), or if a profile is run in the background, or if there are no differences. In most cases there is no need to display this window, except if you want to check to see what the differences are. The Differences window is always displayed when doing a Simulated Run, Simulated Restore, or Restore.

- **Do not show the Differences screen if it is empty (due to filter settings):** If this option is enabled, and the filter settings on the Differences window are such that no files are displayed, then the Differences window will be automatically closed and the profile run will continue. The Differences window is always displayed when doing a Simulated Run, Simulated Restore, or Restore.
- **Use slower but more reliable method of file change detection:** By default, SyncBackSE will not compute the hash value of a file. The reason is that it can dramatically increase the time taken for a profile to run. However, if you want to be absolutely certain that SyncBackSE detects if a file has changed, so that it is copied, then you can enable this option. The only reason to enable this option is if you do not trust the last file modification date & time of the files, and the file size may not change. For example, by default, **TrueCrypt** drive container files never change size or last modification date & time (note that you can configure TrueCrypt to change the contains last modification date & time via the Settings->Preferences main menu). Note that this option will not work if you are using an FTP server that does not support the XCRC extension (the log will contain the warning message "**The FTP server does not support hashing**").
- **Always use slower but more reliable method of file change detection:** SyncBackSE will not calculate a files hash value (to detect file differences) if it has already discovered a file is different anyway, e.g. the files are not the same size. However, if you have a (non-archival) Fast Backup profile, there may be situations when you always want a files hash value to be calculated even if there is no destination file, for example, and even when the files are obviously different. You may want to use this option so that a hash value can be used with incremental and differential backups.
- **Display a message if the profile run is a success (never shown when unattended):** If the profile runs without error then a dialog box is displaying stating that. Normally no message is shown after a profile is run except when simulated or an error occurs.

3.3.16 Compare Options, File Size

Expert Mode: Compare Options, File Size Settings

Note that SyncBackSE has no limitation in regards to the size of files. However, some file systems, e.g. FAT32, do have limits on the size of files.

- **Ignore file size changes (not recommended - may reduce performance):** You can tell SyncBackSE to completely ignore any differences in file sizes. Ignoring the file size has no impact on performance except in some circumstances when using FTP servers. Note that this option is not available if your profile is a Smart Sync profile.
- **Ignore files less than... and ignore files more than...:** To ignore files of a certain size, or within a size range, set these options as appropriate. A value of zero is ignored,

e.g. you cannot ignore all files over zero bytes in size.

3.3.17 Compare Options, Date & Time

Expert Mode: Compare Options, Date & Time Settings

- **Ignore file modification date & time changes (not recommended):** You can tell SyncBackSE to completely ignore the last modification date & time of a file. Ignoring the date & time has no impact on performance. Note that this option is not available if your profile is a Smart Sync profile.
- **Ignore file date & time changes that are because of Daylight Savings Time (DST) changes:** By default if a file's last modification date & time is exactly one hour different then it is ignored. This avoids problems with changes in the time due to Daylight Savings Time.
- **Ignore date & time changes of x seconds or less (differences are rounded down to nearest second):** In some situations, the file system itself may not accurately record the correct date & time when a file was last modified. This can occur when using SAMBA shares (e.g. on NAS devices) and FAT formatted file systems. With this option, you can tell SyncBackSE to ignore small differences in the date & times, e.g. ignore differences of 2 seconds or less. Note that the difference is rounded down to the nearest second, so a difference of 2.99 seconds is treated as 2 seconds. To avoid inaccuracies in file systems, and differences between file systems, SyncBackSE will ignore differences of 2 seconds by default. You can of course change this if you require finer accuracy.
- **Ignore files that have/have not been modified...:** This setting allows you to ignore files that were modified within a certain date range, e.g. within the last x days, or since January 1st 2004. Note that it uses the files last modification date & time, and not the file creation date & time.



This feature may not work as you expect it when using the **within the last** option. The following explains how the date & time comparison works:

Seconds: Fractional seconds do not count. For example, if a file was modified 30.5 seconds ago, and you want to ignore files that have been modified within the last 30 seconds, then it will ignore the file, i.e. 30.5 seconds is treated as 30 seconds.

Minutes: Fractional minutes do not count. For example, if a file was modified 2 minutes and 31 seconds ago, and you want to ignore files that have been modified within the last 2 minutes, then it will ignore the file, i.e. the seconds are not used in the comparison.

Hours: Fractional hours do not count. For example, if a file was modified 2 hours, 31 minutes, and 32 seconds ago, and you want to ignore files that have been modified within the last 2 hours, then it will ignore the file, i.e. the minutes and seconds are not

used in the comparison.

Days: Fractional days do not count. For example, if a file was modified 2 days, 13 hours, 31 minutes, and 32 seconds ago, and you want to ignore files that have been modified within the last 2 days, then it will ignore the file, i.e. the hours, minutes, and seconds are not used in the comparison.

Weeks: Fractional weeks do not count. For example, if a file was modified 2 weeks, 4 days, 13 hours, 31 minutes, and 32 seconds ago, and you want to ignore files that have been modified within the last 2 weeks, then it will ignore the file, i.e. the days, hours, minutes, and seconds are not used in the comparison.

Months: Because months are not all the same length, SyncBackSE assumes there are 30.4375 days per month. Also, fractional months do not count. For example, if a file was modified 2 months, 3 weeks, 4 days, 13 hours, 31 minutes, and 32 seconds ago, and you want to ignore files that have been modified within the last 2 months, then it will ignore the file, i.e. the weeks, days, hours, minutes, and seconds are not used in the comparison.

Years: Because years are not all the same length (e.g. leap years), SyncBackSE assumes of 365.25 days per year. Also, fractional years do not count. For example, if a file was modified 2 years, 7 months, 3 weeks, 4 days, 13 hours, 31 minutes, and 32 seconds ago, and you want to ignore files that have been modified within the last 2 years, then it will ignore the file, i.e. the months, weeks, days, hours, minutes, and seconds are not used in the comparison.

3.3.18 Compare Options, Attributes

Expert Mode: Compare Options, Attributes Settings

- **Do not copy read-only files:** If enabled then read-only files are not copied. This option is useful when a source code control system is used, for example.
- **Do not copy hidden files:** If enabled then hidden files are not copied.
- **Do not copy system files:** If enabled then system files are not copied.
- **Do not copy offline files:** If enabled then offline files are not copied. The offline attribute is only available on NTFS file systems.
- **Do not copy NTFS encrypted files:** If enabled then encrypted files are not copied. The encryption attribute is only available on NTFS file systems. Note that this does not mean it will not copy files encrypted using 3rd party utilities. Files stored on NTFS can optionally be encrypted by Windows itself. It is this type of encryption that this option refers to.
- **Only copy files that do have the archive attribute set:** If enabled then only files which

have the archive attribute set (on) will be copied.

- **Attributes to watch:** SyncBackSE can be configured to watch for file attribute changes. Note that some attributes are not supported on some file systems. Only the NTFS file system supports all the attributes. FAT32/FAT16 file systems support only the Archive, Hidden, Read Only, and System attributes.



Note that SAMBA and non-Windows file systems may not support some of the attributes, or may require further configuration to support them.

When a files attributes are changed, SyncBackSE will copy the file. Which file is copied depends upon the **"What to do if the same file has been changed"** setting on the Decision - Files page. For a **Smart Sync** profile which file is copied depends upon the **"... the same file has been changed"** setting. If hashing is used, then the file is not copied if the files contents are identical, and instead only the files attributes are copied.

3.3.19 Compression

Expert Mode: Compression Settings

SyncBackSE has the ability to compress files using the industry standard ZIP format as well as the newer BZip2 format. Compression reduces the size of the file and has the potential to save a lot of disk space, especially when files such as text and office documents are being copied. Two methods of compression are supported: all the files can be placed into one single compressed file, or each file can be placed into it's own individual compressed file. The first option uses the least amount of disk space, but has the disadvantage that "all the eggs are placed in one basket", so to speak. Also, versioning cannot be used if all files are placed into a single compressed file.

To increase compression performance SyncBackSE can be configured not to try and compress already compressed files, e.g. MP3's, JPG images, etc. Instead of compressing files of these types it will instead **store** them (without compression) in the Zip file. Note that they are still stored in a Zip file but are not compressed within the Zip file.



SyncBackSE is Unicode enabled, so it can store files and folders with names in any language, e.g. Chinese, in a Zip file. However, some compression programs are not Unicode enabled. Because of this if you open a Zip file produced by SyncBackSE using a non-Unicode enabled compression program then it will show the filenames incorrectly (probably with question marks). The problem is with the compression program, not the Zip file produced by SyncBackSE. The solution is to use a compression utility that is Unicode enabled, such as **7Zip**, for example.

Also note that if you open a split Zip file created by SyncBackSE in a compression program, e.g. WinZip, then it may give an error saying the file is corrupted. The problem is that it is expecting the split filenames to be named differently. See this section for details.

Compression Settings

- **Compress the files on destination/right into a Zip file:** Enable this option to compress files copied to the destination/right into a Zip file. If you are backing up files to an FTP server, please read the Fast Backup section for tips on getting the best results.
- **Put all the files into a single compressed file (by default each file will be placed in its own compressed file):** If this option is enabled, then the files will be put into a single compressed Zip file. If this option is not enabled (the default), then each file will be placed into its own Zip file.



When each file goes into its own Zip file, and those Zip files are being stored on an FTP server, we have the problem of knowing what is in a Zip file on a remote FTP server. To know this SyncBackSE would need to download the Zip file and open it to see what file is inside it, what its uncompressed size is, and what its last modification date & time is. To avoid this SyncBackSE changes the filename of Zip files stored on an FTP server by embedding this information in the filename itself.

However, if the filename does not contain this information, e.g. it was created on the FTP server by some other utility, then SyncBackSE will not know the files uncompressed size or its last modification date & time. This means (depending on your profile configuration) it will always assume the file has been changed since the last profile run.

- **Try and compress files that are being used:** By default any file that is being used will not be copied. This is done to avoid corruption, i.e. a file could be changed by another program while a compressed version of it is being created. Enable this option to ignore this safe-guard.
- **Type of compression:** There are five types of compression: Deflated (which is the default), Deflated64, Burrows Wheeler, BZip2, and LZMA.

Deflated provides the normal type of compression used by the older Zip format (traditional PKZIP 2.04g compression method).

Deflated64 (also know as Enhanced Deflate) provides a greater level of compression, but note that it will increase the compression time and is not compatible with older Zip compression programs. Deflate64™ is a trademark of PKWARE Inc.

Burrows Wheeler (popularized by the UNIX and Linux BZip2 program) offers significantly better compression than Deflate but takes longer to compress and decompress data. Tests have shown BWT (Burrows Wheeler Transform) to often achieve between 20% to 30% better compression than Deflate on many popular file types such as databases, pictures, text and executable files. BWT is considered to be one of the most efficient compression algorithms for compressing XML data. In comparison to BWT, Deflated64 is slightly faster but does not compress as well.

BZip2 is similar to the **Burrows Wheeler** compression method except that it is compatible with some compression programs, e.g. WinZip 11. Note that in some cases, e.g. with highly random data, the compression speed can be very slow as compared to the other compression methods. However, it generally provides the best compression

level.  BZip2 support is a Pro only feature.

LZMA (Lempel-Ziv-Markov chain-Algorithm) uses an improved and optimized version of the Lempel-Ziv (LZ77) compression algorithm, backed by a Markov chain range encoder. It uses a variable dictionary size. It is compatible with some compression programs, e.g. WinZip 12. LZMA typically provides much better compression than the Deflate and Deflate64 algorithms at the expense of speed and memory usage when compressing. It also typically provides compression ratios a little better than BZip2/BWT while being a little faster.  LZMA support is a Pro only feature.



Note that the **Burrows Wheeler** compression method is not supported by any other compression program. Only SyncBackSE can be used to restore Burrows Wheeler compressed files.

- **Level of compression:** There are ten levels of compression ranging from level 0 (no compression, files are stored in a compressed file but are not actually compressed) to level 9 (highest compression). The more a file is compressed, the slower it takes to compress the file (it will also use more memory). This option allows you to make a trade-off between speed and file size. The Type of compression setting also influences the compression speed and file size.

3.3.20 Compression, Encryption

Expert Mode: Compression, Encryption Settings

- **Encryption method:** If you wish to password protect the contents of the files in the Zip file then choose the encryption method to use. AES encryption is more secure but not as portable. For example, "Old style" encrypted Zip files can be decrypted and extracted using practically any 3rd party Zip program. However, AES encrypted Zip files can only be decrypted and extracted using newer 3rd party Zip programs, e.g. WinZip 9.
- **Password:** If you want to password protect the files in the compressed archive then enter a password here. It's important to note that if you change the password then the existing files in the destination will still use the old password. Each file in a compressed archive has its own password. The maximum password length is 79 characters.
- **Prompt for the password when run (profile will fail if run unattended):** If this option is enabled then every time the profile is run SyncBackSE will prompt you for the compression password. If the profile is being run unattended, then no prompt will be displayed and the profile run will fail.



Note that you are entirely responsible for remembering the password used. It is not possible under any circumstances for 2BrightSparks to recover lost passwords.

3.3.21 Compression, Advanced

Expert Mode: Compression, Advanced Settings

Advanced Compression Settings

- **Create self-extracting Zip files:** If ticked then a self-extracting Zip file will be created. A self-extracting Zip file is an executable program, that when run, will extract its contents. The benefit is that a 3rd party Zip program is not required to uncompress the files. Please note that as the executable is 32-bit, it is limited to roughly 2GB in size (it's not exactly 2GB as it depends on the executable extractor included in the file).
- **Enable disk spanning (note that this is not possible if unattended):** If you are performing a backup to a removable media, e.g. a floppy-disk, etc. then it may be that the backup cannot fit onto one disk. If this option is enabled then the Zip file will be split across each disk. An important point to note about spanned/split Zip files are that they cannot be modified once created (this is a limitation of the compression format, and not SyncBackSE). The existing Zip file will be automatically deleted and rebuilt. For this reason you may wish to use a Fast Backup profile that does full and incremental backups.
- **Limit the disk usage to...:** If **Enable disk spanning** is enabled then this is the maximum amount of disk space to use on each disk.
- **Create a multi-part Zip file with each part having the maximum size of...:** If **Enable disk spanning** is not enabled then setting a value means you want a split zip file with each part being no more than the size specified. An alternative to disk spanning is to split a Zip file into several files. These separate parts can then be copied manually to a disk, FTP server, etc. An important point to note about spanned/split Zip files are that they cannot be modified once created (this is a limitation of the compression format, and not SyncBackSE). The existing Zip file will be automatically deleted and rebuilt. For this reason you may wish to use a Fast Backup profile that does full and incremental backups. Note that the naming standard used by SyncBackSE is not compatible with some Zip programs. See the section below for more details.
- **Filename extension:** This option is only available when compressing each file to its own file. Enter the filename extension to use for the destination compressed files. By default '.zip' is used.
- **Store the filenames in UTF8 format:** WinZip 12 and newer compression utilities support storing filenames in a special format (UTF8). This allows for non-English filenames, e.g. Chinese, to be correctly recognized.
- **Temporary directory:** By default, temporary files produced during compression are stored in your standard Windows temporary directory. You can however change this using this option. For example, you may be using a small RAM disk as your temporary directory and so when using compression you would like the temporary files stored on your RAM disk. It is recommended you leave this setting empty so that the default temporary directory is used.

Split Zip Files



There is no standard way to name split Zip files. The naming standard used by SyncBackSE for split Zip files is different from that used by **WinZip**, **WinRAR**, and possibly other compression utilities. To open split Zip files you may need to rename the Zip files. For example, the following files may be created by SyncBackSE:

```
Test.zip
Test.z02
Test.z03
```

If you attempt to open the Zip file using WinZip or WinRAR it will incorrectly report that the Zip file is corrupt. If you wish to open the Zip file using WinZip or WinRAR you must rename the files as follows:

```
Rename test.zip to test.z01
Rename test.z03 to test.zip (i.e. change the extension of the last file to .zip)
```

At time of writing, split Zip files produced by SyncBackSE cannot be opened using the 7-Zip archiver, even after renaming the Zip files.

3.3.22 Compression, NTFS

Expert Mode: NTFS Compression Settings

- **Use NTFS compression on files copied to the source/left (only valid on NTFS volumes):** If enabled, and the source/left is on a volume formatted with the NTFS file system, then files copied from the destination/right to the source/left will be compressed using NTFS compression.
- **Use NTFS compression on files copied to the destination/right (only valid on NTFS volumes):** If enabled, and the destination/right is on a volume formatted with the NTFS file system, then files copied to the destination/right from the source/left will be compressed using NTFS compression.

3.3.23 Compression, Compressed

Expert Mode: Already Compressed Files Settings

To increase compression performance SyncBackSE can be configured not to waste time trying to compress already compressed files, e.g. MP3's, JPG images, etc. Instead of compressing files of these types it will instead **store** them (without compression) in the Zip file. Note that they are still stored in a Zip file but are not compressed within the Zip file.

This profile settings page can use and create shared settings.

3.3.24 Encryption

Expert Mode: Encryption Settings

For file encryption please review the Compression Encryption page in this help file. For FTP transmission encryption please review the FTP, Advanced page.

3.3.25 FTP

Expert Mode: File Transfer Protocol Settings

SyncBackSE uses a modern FTP engine that takes advantage of newer FTP standards and extensions, e.g. SFTP (Pro version only), FTPS, XCRC, MODE Z transmission compression, UTF8, and SSL/TLS transmission encryption. This increases performance and compatibility with FTP servers.

This tab is only displayed in Expert mode. The FTP options are not available when the destination/right is using compression, email, scripting, or a CD/DVD.

This profile settings page can use and create shared settings.

FTP Servers and File Dates and Times

When SyncBackSE transfers a file to an FTP server it will try and set the last modification date & time of the file so that it matches the date & time of the local file. However, many older or basic FTP servers simply do not provide the functions to perform this. There are standards for communicating with FTP servers, but not all FTP servers follow those standards, or they implement them incorrectly. There are also many FTP features which are optional, and one of those optional features is setting the last modification date & time of files on the FTP server. This is not a limitation of SyncBackSE, but of the FTP server.

SyncBackSE has an advanced FTP engine that will try several methods to set the last modification date & time of files on an FTP server:

1. If the server supports the **MFMT** extension then that will be used (this does not apply to SFTP servers)
2. If the server supports the **SITE UTIME** extension then that will be used (this does not apply to SFTP servers)
3. If the server supports the **MDTM** extension then that will be used. Note that there are two forms of the MDTM extension: one to get a files date & time, and one to set a files date & time. Many FTP servers support retrieval of a files date & time, but fewer

also support setting of a files date & time using MDTM. This does not apply to SFTP servers.

4. If the server supports none of the above then there are two options: have SyncBackSE change the date & time of the local files to match the date & time of the file on the FTP server (which will be the date & time the file was transferred to the FTP server), or change the profile to a Fast Backup profile. With a Fast Backup profile it will not set the date & time of the files on the FTP server, but it will avoid any problems with the dates & times being different.

SFTP servers only support one way to set a files date & time. However, some SFTP servers do not implement the feature, in which case see point 4 above.

FTP Servers and Filenames

Windows has restrictions on which characters can be used in a filename. It does not allow filenames to contain the following characters: * ? : " < > |

However, some systems, e.g. UNIX, have no such restrictions on which characters can be used in a filename. Because of this, when FTP is used, by default SyncBackSE will automatically convert the filenames so that they are valid for the system they are on. For example, if a file is copied from FTP which has the filename **This * is <> an example?** then when copied to Windows it will have its filename changed to **This %2A is %3C%3E an example%3F**. When the same file is copied back to the FTP server it's name will be changed back to **This * is <> an example?**

You can switch off automatic filename translation on the **FTP -> Advanced** settings page.

The table below shows which characters are converted to and from which codes:

<u>Original Characters</u>	<u>Code Used</u>
:	%3A
*	%2A
?	%3F
"	%22
<	%3C
>	%3E
	%7C
(trailing space)	%20

FTP Server Connection Details

- **Destination/right files are on an FTP server:** If ticked, then the destination/right is an FTP server, i.e. you are backing up to or synchronizing with an FTP server. If the destination is an **SFTP** server, or an **SFTP** server (Pro version only), then define this on the FTP -> Advanced page. If you are backing up files to a compressed Zip file on an FTP server then please read the Fast Backup section for tips on getting the best results.
- **Hostname:** This is the hostname of the FTP server that has the destination directory on it, e.g. ftp.myserver.com. Variables can be used. Note that you must just use the

hostname or IP address. Do not enter an URL, e.g. ftp://ftp.myserver.com/folder/

- **Username:** Your server login username. Variables can be used.
- **Password:** Your server login password. If you are using an SFTP server with key authentication ( Pro version only) then you can leave this blank. If you prefer to be prompted for the password (see the next option) then this edit box is disabled.
- **Prompt for the password when run (profile will fail if run unattended):** If this option is enabled then every time the profile is run SyncBackSE will prompt you for the password. If the profile is being run unattended, then no prompt will be displayed and the profile run will fail.
- **SFTP Key:** If you are using an SFTP server with key authentication then this is the filename of the key file.
- **SFTP Key Password:** If the key file requires a password to be used then enter the key file password here.
- **If the FTP server cannot set a files date & time then change the local files date & time to match that on the server:** SyncBackSE will force the date & time of the file on the FTP server to match that of the file on your PC. **If it cannot set the time of the file on the server, and this option is ticked, then it will change your local files date and time.** However, if this option is not ticked (the default) then the date and time of the file on the FTP server is not changed and neither is the local files date & time. If the FTP server cannot set a files date & time, and you do not want to change your local files dates & times, then the solution is to use a Fast Backup profile.

Once all the appropriate FTP settings have been set, you can test them by clicking the **Test FTP settings** button. SyncBackSE will then attempt to connect and login to the FTP server with its progress being shown in the window below the button.

3.3.26 FTP, Advanced

Expert Mode: Advanced FTP Settings

- **This is an SFTP server (port 22):**  SFTP support is a Pro only feature. If the FTP server is an SFTP server then tick this checkbox.



In many cases the root folder, or login folder, for an SFTP login is different from FTP. If you switch from SFTP to FTP, or vice-versa, it is strongly recommended that you check to make sure the directory is still valid. When using SFTP it is often the case that you have access to all the directories on the server, but with FTP it is typically the case that you only have access to a sub-directory that is pretending to be the root directory (for security reasons).

- **Port:** The port number of the FTP server. Most FTP servers use port 21, except when

using Implicit SSL/TLS encryption in which case most FTP servers use port 990. Most SFTP servers users port 22.

- **Reconnect Attempts:** Whenever the connection to the FTP server is lost, for whatever reason, SyncBackSE will reconnect and resume from where it left off. This is done in the background and does not require any user intervention. This setting specifies the number of attempts SyncBackSE should make to reconnect to the FTP server before it gives up. Note that this number refers to sequential attempts, not the total number of attempts that may be made over the entire profile run (i.e. once a reconnection is made the attempt counter is reset back to zero). This value is also used for the initial connection attempt, so if it is set to 1, and no connection can be made, then the profile will fail instead of trying again.
- **Seconds Between Attempts:** The number of seconds SyncBackSE should pause before making another attempt to reconnect to the FTP server.
- **Read timeout (seconds):** In some situations, e. g. when the connection is dropped, an FTP server may not respond to requests from SyncBackSE. This setting tells SyncBackSE how long to wait (in seconds) for an answer from the server whenever one is required. By default SyncBackSE will wait for 60 seconds before disconnecting and reconnecting to try again.

Encryption and compression options

Note than none of these options are available when using an SFTP server as they are not applicable.

- **Encryption the communication channel:** If enabled the communication channel will be encrypted. This means that all commands sent to and received from the server will be encrypted, e.g. the password is encrypted. Note that this does not encrypt any file communication. For that you need to enable the option **Encrypt the data channel**. If the FTP server does not support encryption then this option and all the encryption settings are ignored.
- **Client certificate to use:** If FTP encryption is being used then you can use this setting to specify which certificate SyncBackSE should use (or if one should be used at all). The certificate list is taken from the Personal certificates installed on the computer.
- **Encrypt the data channel:** If enabled the data channel will also be encrypted. Note this will significantly slow down the profile. For the data channel to be encrypted you must have the communication channel encrypted.
- **Use implicit connection (port 990):** If enabled an implicit SSL connection will be made (most FTP servers use port 990 for implicit connections). If disabled, an explicit SSL connection is made (most FTP servers use the standard FTP port 21 for explicit connections). This option is not available when using an SFTP server.
- **Reduce bandwidth by using compression (MODE Z):** To increase performance on slower networks enable this option. It will transmit data to and from the server in compressed form to reduce the send and receive time. Note that this option requires that

the FTP server supports the MODE Z FTP extension. If not, this option will be ignored. Also note that enabling this option when the FTP server is on a LAN will actually decrease performance.

Misc.

- **Limit bandwidth usage to...:** This option lets you restrict the amount of bandwidth used for the FTP connection. For example, you may also be using the network for other things the same time the profile is run and do not want the profile to use all available bandwidth on the network.
- **Quote Character:** If your FTP server supports wrapping quotes around filenames, e.g. if they have spaces, then enter the quotation character here. If your FTP server does not require or support quoting (most do not) then leave this empty. In most cases uses any value here will cause problems. This option is not available when using SFTP.
- **Server timezone:** If the FTP server is in a different timezone then enter the number of minutes difference from GMT/UTC, e.g. +120 (meaning 120 minutes ahead of GMT/UTC), -60 (meaning 60 minutes behind GMT/UTC). Generally you do not need to enter a value, but in case the FTP server is incorrectly changing the date & times of files then you can correct it here. This option is not relevant and is ignored if your FTP server supports the **MFMT** or **SITE UTIME** extensions. See also the **MDTM syntax** and **Self-correct** settings. This option is not available when using SFTP.
- **Use Unicode (UTF8):** This setting tells SyncBackSE to either use or not use the UTF8 extension on the FTP server. Some FTP servers do not correctly support UTF8 so you may wish to tell SyncBackSE not to use it. Also, some FTP servers do support it but don't tell FTP clients that they can and do support it.
- **MDTM syntax:** In most situations it's best to leave this setting as **Default**. However, some FTP servers may require that a different command format be used. The **MDTM** command is used to set the last modification date & time of a file on the FTP server. This option is not relevant and is ignored if your FTP server supports the **MFMT** or **SITE UTIME** extensions. See also the **Server timezone** and **Self-correct** settings. This option is not available when using SFTP.
- **Use LIST command without options:** Some old and very basic FTP servers cannot provide detailed listings of all the files and sub-folders in a folder. When asked for a list of files they return no information (so SyncBackSE thinks there are no files on the FTP server). To resolve this issue enable this option. This option is not available when using SFTP.
- **Self-correct when setting a files date and time:** Unfortunately many FTP servers do not set the correct date & time of a file, e.g. they incorrectly assume the date & time given to them is a local date & time. To avoid these kinds of problems you can ask SyncBackSE to check to see if the server is setting the dates & times correctly, and if not, to compensate. Note, however, that this is not always possible (it will not work on some old or basic FTP servers).

- **Server requires Allocate command:** Some (old) FTP servers require that the FTP client reserve disk space before transferring files to the server. In general, the majority of FTP servers do not require or support this. This option is not available when using SFTP.
- **Force binary transfers:** To increase performance SyncBackSE will not tell the FTP server it wants to transfer files in binary mode before every file is transferred. If this option is not enabled (default) then it will tell the FTP server to use binary mode immediately after the connection is made, and it will not tell it again. If this option is enabled then SyncBackSE will force the FTP server into binary mode before every file transfer. This will increase profile run times but may be required for some FTP servers. This option is not available when using SFTP.
- **Use the HOST command as this is a virtual host:** Some FTP servers, usually when using the Microsoft IIS FTP server, are virtual hosts, meaning the server hosts multiple web sites. In this case, when SyncBackSE connects to the server it needs to tell the FTP server which host it needs to use (by using the `HOST` command). By default, this option is not enabled, and if it is enabled, and the server is not a virtual host, then it may stop SyncBackSE from connecting. This option is not available when using SFTP.
- **Automatically translate invalid filenames:** By default SyncBackSE will translate filenames so they are compatible with Windows file systems. See this section for details.

3.3.27 FTP, Proxy

Expert Mode: FTP, Proxy Server Details

- **I use a proxy server:** If you must use a proxy server to connect to external FTP or SFTP servers then you must enable this option.
- **Hostname:** This is the hostname of the proxy server, e.g. proxyserver.com. Variables can be used.
- **Username:** Your proxy login username. If you do not need to login to your proxy server then leave this blank. Variables can be used.
- **Password:** Your proxy login password. If you do not need to login to your proxy server then leave this blank.
- **Port:** The port number of the proxy server. This value varies depending upon the type of proxy server software used. Although "1" is the default, it will almost certainly not be this number.
- **Proxy Type:** This setting defines what type of proxy server you are using. It is important the correct setting is used otherwise SyncBackSE will not be able to login to your proxy server. Check with your Network Administrator on which proxy setting to use.

3.3.28 FTP, Firewall

Expert Mode: FTP, Firewall/Router Settings

Note that none of these settings are available when using SFTP as they are not applicable.

- **Passive:** If the FTP server is behind a firewall then you may need to enable this option. If SyncBackSE can login to the FTP server, but cannot transfer files or retrieve a folder listing, then try enabling this option to see if it fixes the problem.
- **Try to keep the connection alive during file transfers:** It is not recommended that this option is used. Use of this option with many FTP servers can cause odd errors (because the commands sent to the FTP server, and the replies given by it, can become "out of sync"). If this option is enabled then when files are sent to, or received from the FTP server, a NOOP command is sent to the FTP server every 30 seconds. In some rare cases this can stop the FTP server from assuming the connection has been broken during long file transfers. However, in the majority of cases the use of this option instead causes more problems.
- **Use Clear Command Channel (CCC):** Choose this option if you are behind a firewall or router that uses NAT (Network Address Translation) and you are having connection problems. Some routers can dynamically change the FTP communication to translate I. P. addresses but this can only be done if that part of the FTP communication is not encrypted.
- **External I. P. address:** If you are behind a router, and are using NAT (Network Address Translation), then you may need to enter your external I.P. address here. This is usually the I.P. address of your router and is not a local I.P. address. This option is not required (or enabled) when using passive connections. Variables can be used.
- **Use ports ranging from...:** If you are behind a firewall then this is the range of TCP ports that you must open on your firewall to allow the FTP server to contact SyncBackSE. If the ports are not open then files cannot be transferred. This option is not required (or enabled) when using passive connections (because with a passive connection you must configure the FTP server to specify which ports it tells the client, SyncBackSE, to use).

3.3.29 Misc.

Expert Mode: Additional Options in SyncBackSE Settings

Miscellaneous Settings in Expert Mode

This page is for miscellaneous settings that don't fit into other categories.

- **Priority:** These are the priorities used when running this profile. You can define the

priority used when run manually, and also when run automatically. A profile with a higher priority will run faster than a profile with a lower priority, if both the profiles are run at the same time. There are seven levels of priority from **Idle** (the slowest) to **TimeCritical** (the fastest). It is not recommended that you use **TimeCritical** as it may cause your entire computer to slow down or lock up. If you want a profile to use the least amount of CPU time, then select **Lowest**. If **Idle** is used the profile may never be run. Note that the -priority command line parameter will override this setting.

- **Flush all open files before running profile:** If enabled then all changes to files that are still in the cache are written to disk. This will add a few seconds to the time taken to run a profile.
- **Stop Windows from sleeping while this profile is running:** In some situations you may not want Windows to sleep while a profile is running, e.g. when burning to CD/DVD. If so, enable this option. Note that this option does not stop a user from putting Windows to sleep. It only stops it sleeping due to the power saving settings in Windows.
- **Pause for...:** In some cases you need to give Windows time to reinitialize network connections and devices (or spin-up CD's) once it comes out of hibernate or standby. This option lets you have the profile pause for a number of seconds before the profile starts running. The pause will be ignored if the profile is set to run on shutdown/logoff and is run at that time (otherwise it would delay Windows shutdown). You can achieve something similar by using the command line parameter -countdown
- **Password protect this profile from modification or deletion:** To protect the profile from modification or deletion enter a password here. If a password is entered then whenever any attempt is made to modify or delete the profile then the user will be prompted for the password. Note that if a password protected profile is using shared settings then you must take note that those shared settings could be changed via an unprotected profile. **Important: It is your responsibility to remember the password.**
- **Speech:** You can configure SyncBackSE to speak (or to play a .WAV sound file) when certain events occur. To play a .WAV file use the filename of a .WAV file. If the file exists then it will attempt to play it. Note that it must be a .WAV file and not any other type of file, e.g. MP3. To play other sound file formats you must convert them to .WAV files using 3rd party programs. You can use variables. For example, if you used %DATE% then it would say the current date.

3.3.30 Misc., Media

Expert Mode: Miscellaneous, Media Settings

- **Eject source/left media after the profile has run:** If enabled, the source/left media will be ejected once the profile has completed. For example, if your source/left directory is on a CD then the CD will be ejected. Note that this option may not work with USB devices.
- **Eject destination/right media after the profile has run:** If enabled, the destination/

right media will be ejected once the profile has completed. Note that this option may not work with USB devices.

- **Load source/left media before the profile is run:** If enabled, the source/left media will be loaded/inserted before the profile is run. For example, if your source/left directory is on a CD then the CD will be loaded.
- **Load destination/right media before the profile is run:** If enabled, the destination/right media will be loaded/inserted before the profile is run.

You can click the **Test Eject** and **Test Load** buttons to test if the media can be ejected or loaded. It will work with most removable media, e. g. CD's, DVD's, Compact Flash cards, etc.

3.3.31 Programs Before

Expert Mode: Programs Before Settings

Define what programs will run before and/or after the profile runs.

You have the option of running an external program before the profile starts and/or after the profile has finished. This useful option allows you to prepare the files being copied before the profile has run, for example. There are many possible uses for these settings in SyncBackSE, so read through the following and consider how they may be applied to your own computer setup.

- **Run before profile:** To have a program run before the profile is run, type the name into 'Run before profile' edit-box or click the folder icon next to it. For example, you could type **notepad.exe** so whenever the profile starts Notepad will be run. You can use variables to define the program name, path, etc. See the section below for more details.



Note that if the program name, or the folder it is in, contains spaces then you must wrap the entire program name with double-quotes otherwise the return value will always be 1. For example, **C:\Program Files\Company Name\A Program.exe** must be specified as **"C:\Program Files\Company Name\A Program.exe"**. Any parameters passed to the program should also be wrapped in their own pair of double-quotes, especially if they contain spaces, e.g. **"C:\Program Files\Company Name\A Program.exe" "param 1" "another param"**



Starting with Windows Vista, Microsoft introduced many restrictions on how programs can react to and handle the shutdown or restart of a computer. Due to these restrictions **Run Before** and **Run After** programs will silently fail and not even start if the profile is set to run on shutdown/logoff and the computer is shutdown or restarted (the programs will still be run as per normal if its a logoff).

- **Wait until the program has finished before running profile:** If this option is enabled, then when the program is run, SyncBackSE will pause the profile until the program has

finished. The program must exit/close before the profile will continue. If you do not enable this option then the program will be run and the profile will carry on running without waiting for it to finish.

- **Wait for a maximum of...:** If this option is enabled, then you can choose how long SyncBackSE should wait for the program to finish before it continues. If the program does not finish within the specified time then SyncBackSE will continue with the profile. Note that it is advisable to set a maximum waiting time otherwise SyncBackSE may get "stuck" waiting for a program that is not going to exit.
- **Abort the profile if the program fails to finish in time:** If this option is ticked then if the program does not return in the time stated then the profile will stop.
- **Abort the profile if the program fails to execute:** If enabled, and the **before** program fails to start (because the program doesn't exist or cannot be run), then the profile will not run. By default the profile will continue to run with a failure.
- **Abort the profile if the programs return value is not ...:** Most programs, batch files, and scripts, have a numeric return/exit value. This usually indicates if it ran without error, and if there was an error, what kind of error occurred. If you tick this checkbox then you can specify which return values the program must return for the profile to be run. A comma-delimited list of values can be entered, and a hyphen used to specify a range of values. For example, if the program returns 0, 1, or a value between 10 and 50 (inclusive) if it was run successfully, and any other value on failure, then in the edit box you would type 0, 1, 10-50.
- **...but do not create a log or record it as a failure:** If this option is ticked then if the program does not return the expected value then the profile will stop. No log will be created, and the last run result will remain unchanged. An example is when you need to connect to something that may not always be available, and if it isn't available then it can be safely ignored, e.g. connecting to a network from a laptop.
- **Run the program when simulating:** By default the program is not run when doing a simulated run. This is advisable as the program may change or delete files, which is not something you generally want to do during a simulated run.

Variables and switches

As with the **Source/Left** and **Destination/Right** directories, you can use Windows environment variables. For example, if you typed "**Notepad %HOMEPATH%\test.txt**" then this would run notepad and open a file in your home directory called **test.txt**. You can see a list of what typical Windows environment variables are available on the following web page:

<http://windowsitpro.com/article/articleid/23873/what-environment-variables-are-available-in-windows.html>

By default all external programs are run in a normal window. You can have the program instead run minimized so that they do not appear as a window on the screen. To do this prefix the program with one of the following:

/min This will run the program minimized and will not make it the currently active window. For example: **/min "C:\Program Files\Company Name\A Program.exe"**

/max This will run the program maximized. For example: **/max "C:\Program Files\Company Name\A Program.exe"**

/hide This will run the program minimized and will also hide the window. For example: **/hide "C:\Program Files\Company Name\A Program.exe"**

Unlike the **Source/Left** and **Destination/Right** directories, you can also use special SyncBackSE variables. These are used in the same way as environment variables, except they have a leading underscore character, e.g. **%_Source%**. The value returned is for the profile being run. Below is a list of the most common variables that you can use:

_Compression = Returns 1 if the destination/right is compressed

_Destination = Destination/right directory/filename

_DestIsFTP = Returns Y if the destination/right is an FTP server

_Priority = Run priority of the profile

_SingleFile = Returns 1 if compressing to a single file (ignore if **_Compression** returns 0)

_Source = Source/left directory

For a full list see the profile settings INI file.

3.3.32 Programs, After

Expert Mode: Programs, After Settings

- **Run after profile:** To have a program run after the profile has finished, type the name into 'Run after profile' edit-box or click the folder icon next to it. You can use the same variables and switches as per the Run Before program.



Starting with Windows Vista, Microsoft introduced many restrictions on how programs can react to and handle the shutdown or restart of a computer. Due to these restrictions **Run Before** and **Run After** programs will silently fail and not even start if the profile is set to run on shutdown/logoff and the computer is shutdown or restarted (the programs will still be run as per normal if its a logoff).



Note that the special variable **%PROFILEFAILED%** can be used to pass the result of the profile run. If the value is 1 (one) then the profile failed (or was aborted), if the value is 0 (zero) then the profile run was a success.

- **Wait until the program has finished before running next profile or exiting:** If this option is enabled, then when the program is run, SyncBackSE will pause and not finish the profile until the program has finished. The program must exit/close before the profile will complete running. If you do not enable this option then the program will be run and the profile finish as per normal without waiting.
- **Wait for a maximum of...:** If this option is enabled, then you can choose how long SyncBackSE should wait for the program to finish before the profile run ends. If the program does not finish within the specified time then SyncBackSE will end the profile run. Note that it is advisable to set a maximum waiting time otherwise SyncBackSE may get "stuck" waiting for a program that is not going to exit.
- **Run the program even if the profile fails:** Select this option to run the **after** program even if the profile fails. By default the program is not run if the profile fails, e.g. a file could not be copied. If the source or destination is a UNC path (\\server\share\folder\), and it cannot be connected to, then the **Run After** program is not run.
- **Run the program when simulating:** By default the program is not run when doing a simulated run. This is advisable as the program may change or delete files, which is not something you generally want to do during a simulated run.
- **Run the program after the log file has been closed (any failure will not be recorded in the log):** By default the **after** program is run before the log file is closed and created. This is so the result of the program run can be recorded. It also means the log file has not yet been created. In some situations you may want the after program to use the log file, and if so, you need to enable this option. The variable %LOGFILENAME % can be used to get the filename of the first page of the log file.
- **Run the program only if any file changes were made:** If you only want the program to run if any files were copied, deleted, or moved, then tick this option.

3.3.33 Auto-close Settings

Expert Mode: Auto-Close Settings

Define what programs will close before running a profile by entering the contents of their title bar.

This profile settings page can use and create shared settings.

Although SyncBackSE can copy locked files (under the correct circumstances) it sometimes cannot copy files that are being used by other programs. For example, you cannot copy a Word document while it is being edited in Microsoft Word and the file is on a network drive. One option available is to automatically close those programs so SyncBackSE can copy the files being used. These settings allow you to choose which programs to automatically close before the profile is run.

To add a program to the list, click the **Add** button. You are then prompted to enter the words

that appear in the application title bar of the program you want closed, or select one from the drop-down list. For example, if you want to close Microsoft Word then type **Microsoft Word**. An important point to remember is that it is case sensitive. For example, if you use **microsoft word** then it would fail. You only need to enter a fragment, i.e. you do not need to type in the exact title but just a portion of it. To be sure the setting is correct try running the program you wish to close then click the **Test** button.

To remove entries from the list, click on them and click the **Remove** button.

Before a profile is run, SyncBackSE will try and close all the programs with those words in their title bar. If you are using **Microsoft Word** at the time then **Word** will prompt you to save the document before it closes. However, some programs may not prompt you and refuse to close, or you may want them closed even if they prompt. In this case you must tick the **Forcibly close programs that will not close gracefully** option. If this option is enabled then SyncBackSE will forcibly close those programs if they don't close gracefully.



Note that this will very probably result in you losing data, so this option should not be used without careful consideration.

Due to security changes introduced with **Windows Vista**, auto-close will not work with scheduled profiles (unless they are set to run only if the user is logged on). This is because processes that are run via the scheduler are run in a different session and have no access to the desktop processes.

3.3.34 To Email

Expert Mode: Email

In SyncBackSE V4 the Email settings page referred to emailing of the log file. From V5 those settings have now been moved to the Log -> Log Email settings page. This settings page is now for backup to an email server.  These options are only available in the Pro version.

This profile settings page can use and create shared settings.

Using these settings it is possible to backup (or synchronize) your files with an email server. This is not to be confused with synchronizing emails or making a backup of your emails. SyncBackSE allows you to store your files on an email server. For example: you may be in the office working on some important documents and want to leave the office and work on them at home. You could copy them to a USB key (for example) and take that home, but if you don't have any external storage medium you could instead send them to your email server. At home you would then retrieve them from your email server, work on them, back them up to the email server, go to work in the morning, and restore them from your email server. It's similar to using an FTP server, except you are using your email server. The advantage is that there are numerous free email servers, e.g. Gmail, Yahoo!, etc., on which you could store your files.

A tutorial on backing up files to a **Gmail** account is available for viewing online:

<http://www.2brightsparks.com/howto/sbpro/backupgmail/index.htm>

SMTP Server Connection Details

- **Destination/right files are on an Email server:** Tick this checkbox to backup/sync your source/left files with an email server.
- **Hostname:** The hostname (or IP address) of your SMTP server, e.g. mail.server.com. Check with your email provider or systems administrator on what that is.
- **To:** The email address of where to send the files on the SMTP server. Note that some email servers require a full email address, e.g. [name@email.address.com](#). In today's world of junk email, many email servers are configured to reject any email that appears to be spam. If your email server is rejecting your email please make sure you have filled out the settings correctly, e.g. valid **To** and **From** addresses are used. You can enter multiple email addresses by separating them with **semi-colons** or **commas**, e.g. you@email.com; you@yahoo.com
- **From:** The email address of the sender. This is usually a full email address, e.g. [name@email.address.com](#). It is usually the same as the **To** email address, however, for debugging purposes (e.g. to check if your email is being rejected) you may want to use a different email address.
- **Subject:** The subject of the emails that store your files. Note that you can use a range of variables to help you see which files are in which email message, although you don't strictly need to know this (as you should use SyncBackSE to restore/sync the files).
- **Must login to SMTP server:** If you must login to your SMTP email server then tick this checkbox and enter your login username and password. Note that some servers require a login whereas others may fail if you do attempt to login. Check with your email provider or systems administrator.
- **Prompt for the password when run (profile will fail if run unattended):** If this option is enabled then every time the profile is run SyncBackSE will prompt you for the password. If the profile is being run unattended, then no prompt will be displayed and the profile run will fail.

POP3/IMAP4 Server Connection Details

- **POP3 or IMAP4:** Choose the type of server used to retrieve emails. Typically it is a POP3 email server. Check with your email provider or systems administrator.
- **Hostname:** The hostname (or IP address) of your POP3 or IMAP4 server. Check with your email provider or systems administrator on what that is. Note that in some cases, e.g. Gmail and other web based email services, you may need enable access to your emails via a POP3 (or IMAP4) server.

- **IMAP Folder:** If you are using an IMAP4 email server then you can optionally enter the name of the folder to retrieve your files from. Note that you can use variables here. Also, you may need to first create the folder on your IMAP4 server. It is important to remember that SyncBackSE cannot choose where to store your files as that is controlled by your SMTP server (where your files are stored depends on the settings on your SMTP server). The IMAP4 server is used to read your files from.
- **Must login to POP3/IMAP4 server:** If you must login to your POP3/IMAP4 email server then tick this checkbox and enter your login username and password. Note that some servers require a login whereas others may fail if you do login. Check with your email provider or systems administrator.
- **Prompt for the password when run (profile will fail if run unattended):** If this option is enabled then every time the profile is run SyncBackSE will prompt you for the password. If the profile is being run unattended, then no prompt will be displayed and the profile run will fail.

Important notes about using backup to email

- **Backup to email should not be your primary backup location!**
- Use a dedicated email account to store your files. This allows SyncBackSE to delete any non-backup related emails, e.g. spam.
- There can be a potentially long delay between sending an email and it being stored in your email inbox, so do not run the backup too frequently.
- Email servers may have flood control to stop many emails from one address/server.
- Most email servers have a limit on the size of attachments. This means you may not be able to backup large files. To get around this it is recommended you use single Zip compression with splitting (not spanning).
- Use Deflate64 and maximum compression level. This reduces the amount of space used on your email server and also reduces the time it takes to send and receive files.
- Use 256-bit AES encryption (with compression) if your files contain private/sensitive information.
- If you use splitting with compression then note that a split Zip file cannot be updated. This means it will be a full backup every time.
- If you have very large files you should use single Zip compression with splitting (set the split size as appropriate).
- If you have many small files it is recommended you use multi-zip compression.
- When using single zip, if the Make safe copies option is not enabled, then it will delete the existing Zip file(s) before uploading the new one(s).
- It is recommended to mask the attachment filename to avoid the email from being blocked.

Gmail

If you are using a Gmail account, the following explains how to configure it so it can be used with SyncBackSE:

- Login to your Gmail account
- Click **Settings** link in top-right

- Go to **Forwarding and POP** tab in Settings
- Enable **Enable POP for all mail (even mail that's already been downloaded)**
- Change **When messages are accessed with POP** to **keep Gmail's copy in the Inbox**
- Click **Save Changes**

The problem with Gmail is that it sometimes forgets these settings and so you may have the problem of SyncBackSE saying there are no emails. This is because the Gmail POP server is saying there aren't any emails because the **Enable POP for all mail** setting is sometimes "forgotten" by Gmail. Also, sometimes Gmail doesn't appear to delete emails that SyncBackSE asks it to delete.

3.3.35 To Email, Advanced

Expert Mode: Email, Advanced



These options are only available in the Pro version.

- **This email account is only used by this program. Delete any emails not created by this program:** If you are using a dedicated email account then it is highly recommend that you enable this option. When enabled SyncBackSE will automatically delete any emails that have not been created by SyncBackSE, e.g. spam. This ensures your email account is only storing your files. **Warning: if you are using the email account to receive emails not from SyncBackSE then do not enable this option.**
- **Mask the attachment filename to attempt to avoid it being blocked, e.g. by Gmail:** Some email providers, e.g. Google Gmail, will block attachments of a certain kind, e.g. executable programs, pictures, etc. SyncBackSE can bypass this by using a different filename as the attachment filename, e.g. attachment.txt. This does not affect SyncBackSE as it does not depend on the filename of the attachment. However, if you are restoring files manually, for example, then you may wish to keep the original filename. Please note that you could put the original filename in the email body (as well as the email subject) for reference. To help avoid problems, new profiles will have this option enabled by default.
- **If an email cannot be deleted because it is not a re-scan then ignore it:** This option is only available for Fast Backup profiles and by default is not ticked. If the profile is configured to delete files or folders that are only on the email server, and the profile run is not a re-scan, then SyncBackSE cannot delete the email because it does not know which email to delete. It does not know which email to delete because it is a Fast Backup and so has not asked the POP3/IMAP4 email server for a list of emails. If this setting is not ticked, and an email cannot be deleted, then the log file will contain the error message **The email cannot be deleted because it is not a re-scan.**

SMTP Server Connection Details

- **Port:** The port number of your SMTP email server. It is recommended you leave it as

zero (then SyncBackSE will use the default port number based on your settings).

- **Local computer name:** Leave this blank unless you are familiar with the HELO SMTP command. SyncBackSE tells the SMTP server that this is the name of the computer (the hostname). Some SMTP servers are configured to reject attempts to use them from computers that identify themselves incorrectly or via an I.P. address. SyncBackSE attempts to detect and correct this by not identifying itself. To tell SyncBackSE to not identify itself set this value to * (a single asterisk). To tell SyncBackSE to send the local computers name (instead of I.P. address) set this value to + (a single plus sign).
- **Reply To:** The email address any replies should be sent to. You can leave this blank.
- **CC:** Carbon-copies of the email can be sent to other email addresses. You can enter multiple email addresses by separating them with **semi-colons** or **commas**, e.g. you@email.com; you@yahoo.com
- **BCC:** Blind-carbon-copies of the email can be sent to other email addresses. You can enter multiple email addresses by separating them with **semi-colons** or **commas**.
- **Receipt:** If you require a delivery receipt to be sent then enter the email address of the person to receive that receipt. Note that a receipt is only sent if the email client or email server supports this feature.
- **Must use SSL/TLS connection:** If your SMTP server requires an SSL/TLS encrypted connection, or it supports one and you want your email to be transmitted to the server in encrypted form, then enable this option. Note that your server must support SSL/TLS encryption.
- **Direct SSL/TLS connection (usually using port 465):** If your SMTP server supports a direct encrypted connection then select this option.
- **Use STARTTLS command:** This is different from the above setting in that it connects to the SMTP server then requests that the connection be encrypted by sending the STARTTLS command to the email server. Choose this option if your SMTP server does not support a direct encrypted connection.

POP3/IMAP4 Server Connection Details

- **Port:** The port number of your POP3/IMAP4 email server. It is recommended you leave it as zero (then SyncBackSE will use the default port number based on your settings).
- **Must use SSL/TLS connection:** If your POP3/IMAP4 email server requires an SSL/TLS encrypted connection, or it supports one and you want your email to be transmitted from the server in encrypted form, then enable this option. Note that your server must support SSL/TLS encryption.
- **Direct SSL/TLS connection (usually using port 995 for POP3, 993 for IMAP4):** If your POP3/IMAP4 server supports a direct encrypted connection then select this option.

- **Use STLS command (POP3 only):** This is different from the above setting in that it connects to the POP3 server then requests that the connection be encrypted by sending the STLS command to the email server. Choose this option if your SMTP server does not support a direct encrypted connection. This option is not available if an IMAP4 server is being used.

SMTP Encoding

- **Header encoding:** It is recommended that you do not change this setting. This setting defines the encoding format used for the email header.
- **Transfer encoding:** It is recommended that you do not change this setting. This setting defines the encoding format used for the email body.
- **I want to customize the email body:** If ticked then you are able to create a custom email body instead of using the default. The email body can use Windows environment variables, e.g. %HOMEPATH%, as well as all the SyncBackSE Variables and some special variables that are only available when using an email server to store your files. Note that the email body is not stored as part of the shared settings. You can use either a plain text body and/or a HTML body. Keep in mind that the readers email client will need to be able to display HTML email. Because of this you may want to have both a test body and HTML body so that email clients that do not understand HTML emails will display the text body instead.

3.3.36 Log

Expert Mode: Log Settings

In SyncBackSE V4 the Email settings page referred to emailing of the log file. From V5 those settings have now been moved to this page, and the To Email page is for backup to email.

- **Display log after running a profile:** Once a profile has finished, if this option is enabled then the log file will be displayed. There are some cases where the log file will never be displayed: if the media cannot be loaded, a pre-run pause was aborted by the user, the Run Before program fails, or if the source or destination cannot be connected to via the network.



Note: when the profile is run via a schedule, and **Windows Vista, Windows 7**, or newer is being used, then the log may not be displayed. This is due to security changes introduced in Windows Vista (basically when SyncBackSE is run from a schedule it cannot interact with the user interface).

- **Only when errors occur:** If this option is enabled, then the log file will only be displayed if any errors occurred during the profile run, e.g. a file could not be copied.
- **Log the reason why files/folders on source/left are ignored/skipped:** In some

cases you may want to know why a file or folder was not copied from the source/left, for example. If this option is enabled then the log file will state the reason. Note that this is the reason why SyncBackSE did not wish to copy, move, or delete the file/folder. The file/folder may still have been copied if you told SyncBackSE to do so in the Differences window.

- **Log the reason why files/folders on destination/right are ignored/skipped:** In some cases you may want to know why a file or folder was not copied from the destination/right, for example. If this option is enabled then the log file will state the reason. Note that this is the reason why SyncBackSE did not wish to copy, move, or delete the file/folder. The file/folder may still have been copied if you told SyncBackSE to do so in the Differences window.
- **Do not log skipped files or folders:** If ticked then skipped files are not recorded in the log. A skipped file is one which has been skipped due to the settings on the Decisions - Files settings page or set to skipped via the Differences window. Note that skipped files will also be recorded in the ignored section of the log file if the **Log the reason why files/folders on...** setting has been enabled.
- **Do not log successfully copied, moved, or deleted files:** If ticked then if a file is copied, moved, or deleted without any problems then it is not recorded in the log file. The main reason you may want to use this option is to reduce the size of the log file, e.g. thousands of files are likely to be copied and the log file is going to be emailed. Even if this option is selected it will still log renames or if a file is copied and a reboot is required to complete the copy.
- **Log the drive serial numbers (internal and external drives only):** If ticked then the drives hardware serial number is recorded in the log file. The hardware serial number is not the same as the serial number given to a partition when you format it. Drive serial numbers cannot be changed and are stored by the drive hardware itself.
- **Log the S.M.A.R.T. information to check for possible drive failure (internal drives only):**  S.M.A.R.T. support is a Pro only feature. S.M.A.R.T. is an acronym for Self-Monitoring, Analysis, and Reporting Technology. If ticked a check is made to see if the drive may fail in the near future (or if it has already failed). This feature only works if your computers BIOS supports the SMART standard (most do), and it has also been enabled in your BIOS, and the drive supports SMART (most do). Note that there are countless types and versions of BIOS's available so please refer to your BIOS manual for information on how to do this. Although using this may help detect impending drive failure, it is not a perfect technology. It should not be relied upon to always detect an impending drive failure.
- **If a file cannot be copied because it was deleted before it could be copied then make it a warning and not a failure:** In some cases you may get run failures because a file (usually temporary files) cannot be copied because they have been deleted (or moved) by something else before they could be copied. If you prefer to have these errors instead recorded as warnings (so they don't cause a profile run failure) then enable this option. Care should be taken when using this option as you are telling SyncBackSE that it should not treat an error as an error. Note that this option is ignored when running a

profile as a restore, and it is also ignored when copying from email or single zip files, and is also ignored when copying to or from a script location (🏠 Pro only).

- **If the user cannot be prompted to choose an action then:** On the Decisions - Files and Decisions - Folders settings pages you may have chosen to be prompted in some situations. However, if the profile is being run unattended, e.g. from a schedule, then you cannot be prompted. In this case the file is skipped and a warning is recorded in the log file. However, using this setting you can change what (if anything) is recorded in the log file. The file or folder will always be skipped, but if you are using a SmartSync profile then you may want the file or folder to be skipped and the changes to be ignored.
- **Override the program wide setting for the number of log files to keep:** Using the Log Settings window you can specify how many log files all profiles should keep. However, using this setting it is possible to override that program wide setting and set the number of log files to keep for this specific profile.
- **Delete log files:** Click the button to delete all the profiles log files. The button is disabled if the profile has no log files to delete. To delete the logs files of all profiles use the Log Settings window (via the Preferences main menu).
- **S.M.A.R.T. Test:** 🏠 S.M.A.R.T. support is a Pro only feature. Click the button to have all connected drives tested to see if a drive is predicted to fail. Note that some drives cannot be predicted to fail. An error message is only displayed if a drive is predicted to fail or if there is no way to predict failure of the drive.

3.3.37 Log, Email Log

Expert Mode: Log, Email

After a profile has been run, SyncBackSE can email the log file. This is especially useful when SyncBackSE is located on a remote machine. The log file can optionally be sent as an email attachment.

In today's world of junk email, many email servers are configured to reject any email that appears to be spam. If your email server is rejecting your log email please make sure you have filled out the settings correctly, e.g. valid **To** and **From** addresses are used.

This profile settings page can use and create shared settings.

- **Email the log file after the profile has run:** To email the log file, enable this option. If the profile is disabled then no log will be sent.

SMTP Server Connection Details

- **Hostname:** This is the hostname of your SMTP email server, e. g. **smtp.mailserver.com**. Note that it is the SMTP server (which is used for sending email), not the POP3 or IMAP4 server (which is used for receiving email).

- **To:** The email address to send the log file to. You can enter multiple email addresses by separating them with **semi-colons** or **commas**, e.g. you@email.com; you@yahoo.com
- **From:** The email address from which the log file has been sent. Typically you would put your own email address here. **Note that some SMTP servers may reject the email if it is not from a valid email address or an email address on that server.**
- **Subject:** The subject to use for the email, e.g. **SyncBackSE Log File**. Note that you can use environment variables in the subject. You can also use the special variable **%PROFILENAME%** to put the profiles name in the subject, e.g. **Log file for %PROFILENAME%**
- **Must login to SMTP server:** If you must login to your SMTP server then enable this option. Due to spam, most email servers now require you to login. SyncBackSE can login to SMTP servers that require a username and password in clear-text, NTLM, CRAM-MD5, or MSN.
 - **Username:** The username to use to login to your SMTP server.
 - **Password:** The password to use to login to your SMTP server.
 - **Prompt for the password when run (profile will fail if run unattended):** If this option is enabled then every time the profile is run SyncBackSE will prompt you for the password. If the profile is being run unattended, then no prompt will be displayed and the profile run will fail.

3.3.38 Log, Advanced

Expert Mode: Email, Advanced Settings

- **Only send the email if an error occurs:** If you only want the log file to be emailed if the profile failed then tick this check box, otherwise the log file will be emailed after every run. The email subject will be prefixed with **[Failed]** if the profile run failed.
 - **...or there are differences:** If you only want the log file to be emailed if the profile run failed, or differences were found between the source and destination, tick this check box.
- **Do not send the email if it is a simulated run:** In most cases you probably don't want to have the log emailed if it is a simulated run. If so, enable this option.
- **Do not attach the log file, just send the email body:** If you only want the email body, and not the log files attached, then tick this option.
- **Only attach the log file if an error occurs:** Tick this option if you only want the log file attached to the email if the profile run fails.

- **Compress the attached log file and give the attachment a filename of ...:** This option is enabled by default. When enabled the log file will be compressed into a single zip file and attached to the email. Some email programs, or corporate environments, will not allow attachments of certain types. If this is the case try changing the filename to **log.txt**, for example. You can use variables in the filename, e.g. %PROFILENAME%.zip
- **Encryption method:** If you only want the log file to be encrypted then choose the type of encryption you wish to use. Note that you can only encrypt the log if it is also compressed. The receiver may need compression software to decrypt the log file.
- **Password:** The password to encrypt (and decrypt) the log file.

SMTP Server Connection Details

- **Port:** The TCP/IP port number which your email server uses. Most SMTP email servers use port 25.
- **Local computer name:** Leave this blank unless you are familiar with the HELO SMTP command. SyncBackSE tells the SMTP server that this is the name of the computer (the hostname). Some SMTP servers are configured to reject attempts to use them from computers that identify themselves incorrectly or via an I.P. address. SyncBackSE attempts to detect and correct this by not identifying itself. To tell SyncBackSE to not identify itself set this value to * (a single asterisk). To tell SyncBackSE to send the local computers name (instead of I.P. address) set this value to + (a single plus sign).
- **Reply To:** The email address any replies should be sent to. You can leave this blank.
- **CC:** Carbon-copies of the email can be sent to other email addresses. You can enter multiple email addresses by separating them with **semi-colons** or **commas**, e.g. you@email.com; you@yahoo.com
- **BCC:** Blind-carbon-copies of the email can be sent to other email addresses. You can enter multiple email addresses by separating them with **semi-colons** or **commas**.
- **Receipt:** If you require a delivery receipt to be sent then enter the email address of the person to receive that receipt. Note that a receipt is only sent if the email client or email server supports this feature.
- **Must use SSL/TLS connection with SMTP server:** If your SMTP server requires an SSL/TLS encrypted connection, or it supports one and you want your email to be transmitted to the server in encrypted form, then enable this option. Note that your server must support SSL/TLS encryption.
- **Direct SSL/TLS connection (usually using port 465):** If your SMTP server supports a direct encrypted connection then select this option. Usually port 465 (instead of port 25) is used in this case. To change the port number you must go to the Email - Advanced page.

- **Use STARTTLS command:** This is different from the above setting in that it connects to the SMTP server then requests that the connection be encrypted by sending the STARTTLS command to the email server. Choose this option if your SMTP server does not support a direct encrypted connection.

SMTP Encoding

- **Header encoding:** It is recommended that you do not change this setting. This setting defines the encoding format used for the email header.
- **Transfer encoding:** It is recommended that you do not change this setting. This setting defines the encoding format used for the email body.
- **I want to customize the email body:** If ticked then you are able to create a custom email body instead of using the default. The email body can use Windows environment variables, e.g. %HOMEPATH%, as well as all the SyncBackSE Variables and some variables that are used especially in the subject and email body. Note that the email body is not stored as part of the shared settings. You can use either a plain text body and/or a HTML body. Keep in mind that the readers email client will need to be able to display HTML email. Because of this you may want to have both a text body and HTML body so that email clients that do not understand HTML emails will display the text body instead.

3.3.39 Network

Expert Mode: Network Settings

If your source/left and/or destination/right directory are on a network, and you are using a UNC path, e.g. \\machine\share\directory, then this page allows you to set the username and password required to connect to the network shares. You can only edit the values if you are using a UNC path or if the path contains variables (because it is impossible to know if it will be a UNC path or not until the profile is actually run).

This profile settings page can use and create shared settings.

- **Username:** Your network username for the source/left or destination/right (as appropriate).
- **Password:** Your network password for the source/left or destination/right (as appropriate).
- **Test Connection:** Click this button to test if SyncBackSE can connect to the **source/left** and/or **destination/right** using the supplied username and password. This button is disabled if the path is not a UNC path (if variables are being used in the path then keep in mind that variables can change value and so the currently expanded variables may not make the path a UNC path).

If you use the Network settings in SyncBackSE take account of the following Windows issues:

- To use a network for the source/left or destination/right you must specify a valid UNC path, e.g. \\Machine Name\ShareName\Folder\
- Your current username and password are used before the supplied username and password (unless the option **Use this username and password before trying my current username and password** is ticked).
- For **Windows XP Home** you may need to supply the **Guest** password and not enter any username.
- The username may need to be in the form **Domain\Username** or **MachineName\Username** for it to work correctly.
- If the destination computer is configured to use simple file sharing (the default on Windows XP) then it may connect even if an invalid username and password is used. Note that it may also connect even if an invalid username and password is used because Windows caches connection information and so may use valid cached credentials instead.
- Windows networking has many quirks and problems. Please experiment with various settings before seeking help.

3.3.40 Network, Advanced

Expert Mode: Network, Advanced Settings

You can only edit the values if you are using a UNC path or if the path contains variables (because it is impossible to know if it will be a UNC path or not until the profile is actually run).

- **Use this username and password before trying my current username and password:** If enabled then SyncBackSE will try the supplied username and password before using your current username and password. By default SyncBackSE will use your current username and password before trying the ones supplied.
- **Do not disconnect from the network after the profile has run:** By default SyncBackSE will disconnect the network connection once the profile run has finished. If you change this setting so that SyncBackSE does not disconnect you should be aware that it may cause problems with the computer you are connecting to and/or from. Windows, and other operating systems, have limits on the number of network connections that they can have open at any one time. The limit could depend on a number of factors, e.g. technical or licensing. If the connection is not dropped then the limit may be reached and problems may occur, e.g. computers may not be able to connect to the remote server.

3.3.41 CD/DVD Burner

Expert Mode: CD/DVD Burner

 These options are only available in the Pro version. SyncBackSE can backup to the following types of CD/DVD (your CD/DVD drive also needs to support these types):

- BD-R / BD-RE (Blu-Ray)
- DVD+R Dual Layer
- DVD-R Dual Layer
- DVD+R / DVD+RW
- DVD-R / DVD-RW
- DVD-RAM
- CDR / CDRW
- HD-DVD

It can also backup to an ISO image file which can be mounted as a virtual drive (using 3rd party software, e.g. [DAEMON tools](#)) or burned to disk using 3rd party [burning software](#). SyncBackSE also performs automatic disk spanning, i.e. if your files do not fit onto the CD/DVD then it will split the files and continue onto another CD/DVD. Disk spanning does not require the use of compression, but using compression means fewer disks are used.

SyncBackSE can only backup to disks pre-formatted in the **UDF** format. It also does not support the automatic disk spanning feature. **SyncBackPro** does not require UDF, and supports automatic disk spanning. If you import a profile from an older version of SyncBackSE (or SyncBack freeware) that copies to a UDF formatted CD or DVD, and you want to make use of the new disk burning features in SyncBackPro, then you must modify the profile and enable the option **Destination/right is a CD/DVD burner, burn the files to the disk in this drive**. **Note that the contents of the CD/DVD is entirely replaced, i.e. it does not update the existing files on the CD/DVD but instead overwrites everything. Anything already on the disk will be deleted.**

This profile settings page can use and create shared settings.



Backing up to CD/DVD is not recommended for use as your primary backup. It is cheaper, faster, and more reliable to backup to an external drive or USB key. However, if you must backup to CD/DVD, e.g. for offsite storage, it is highly recommended that you check to see if you have the latest firmware for your CD/DVD drive and that you use quality media that is compatible with your drive.



If you are using single Zip compression then note that the Zip file is entirely created on the hard drive first, and then copied to the CD/DVD or ISO file. It is then deleted from the drive once entirely copied to CD/DVD. This means you must have the necessary free hard disk space. Also, if there are **any compression errors** then nothing will be burned to CD/DVD and no ISO file will be created. This is to avoid wasting disks with an incomplete backup.



When burning to CD/DVD you may want to stop Windows from going into sleep mode due to the power savings settings. To do this enable an option on the Misc. settings page.

- **Destination/right is a CD/DVD burner, burn the files to the disk in this drive:** If enabled then the files are copied to a CD/DVD. **Note that the contents of the CD/DVD is**

entirely replaced, i.e. it does not update the existing files on the CD/DVD but instead overwrites everything. Anything already on the disk will be deleted.

- **Create an ISO file on destination/right, do not burn the media:** If enabled then the files are copied to an ISO file. Note that SyncBackSE cannot restore or read from an ISO file. You need to use 3rd party software to access the ISO file. The ISO file can be mounted as a virtual drive (using 3rd party software, e.g. DAEMON tools) or burned to disk using 3rd party burning software.
- **I understand the CD/DVD will be entirely erased. Do not prompt me for confirmation before it is erased:** The contents of a CD/DVD is always entirely replaced, i.e. it does not update the existing files on the CD/DVD but instead overwrites everything. Anything already on the disk will be deleted. Because of this you are prompted before the disk is erased so you are given an opportunity to abort. If you run the profile unattended, e.g. on a schedule, and the profile is set to prompt you before erasing a disk then SyncBackSE will take the safer option (i.e. it will not assume you want the disk erased) and abort the profile. To stop it aborting, and to stop it prompting before erasing a disk, tick this checkbox. When you are prompted to erase the disk you can also stop it prompting again by ticking the "Do not prompt me again" checkbox in the dialog window. For safety reasons, this setting cannot be ticked by default and is not part of the shared settings.
- **Limit the disk usage (or ISO file size) to:** You can optionally limit how much disk space on the CD/DVD that SyncBackSE uses. If the files will not fit onto the CD/DVD then you will be prompted to insert a new disk so the backup can continue. This means you must be physically present to change the disks, i.e. it cannot be an unattended profile run. If the profile run is unattended then it will not prompt you and the backup will fail. If you are copying to an ISO file then this is the maximum size of the ISO file. If the files will not fit into the ISO file then a new ISO file is created to put the remainder of the files into.
- **Do not split the files to fit them onto the disk/ISO:** To make maximum use of the disk space, SyncBackSE will automatically split a file if it cannot fit onto the free space available on the disk. This ensures that all of a disks space is used and none is wasted. However, you may not want SyncBackSE to do this, e.g. the CD/DVD is not going to be restored using SyncBackSE. In that case you can enable this option, but note that no single file can be larger than one CD/DVD, i.e. it will not split such a file over multiple disks.
- **Format:** This is the file system type to use on the CD/DVD or ISO file. It is recommended you leave this setting as **default**. Each format type has different restrictions on the filename lengths and what characters they can contain. SyncBackSE records the original filename of a file so when you restore it will restore using the correct filename, even though the filename on the CD/DVD may be different. If you use compression then the Zip file will contain the original filename. Also note that if you are creating an ISO file then some formats may not be compatible with the 3rd party software used to read or burn the ISO file. Check the documentation of that software for its format requirements.
- **If a file or folders name is changed due to the formats requirements then...:** As

explained in the **Format** setting, the filename stored on the CD/DVD may not be the same as the original filename. This is a limitation of the file system used, not of SyncBackSE. SyncBackSE gets around this limitation by recording the original name of a file so when it is restored the correct filename is used. However, you may wish to know if a filename has been changed, and can do so with this setting.

3.3.42 CD/DVD Burner, Advanced

Expert Mode: CD/DVD Burner, Advanced

 These options are only available in the Pro version.

- **Make the disk/ISO bootable:** Enable this option to make the CD/DVD bootable. This means you can insert the disk into a drive and boot the computer using it (assuming the computer can boot from a CD/DVD disk). To boot from a CD/DVD you must supply a boot image. **SyncBackSE does not provide boot images as they are typically copyrighted software and not free.**
- **ISO9660 Level:** This defines the filename limitations and affects the compatibility of ISO files. If you are copying to an ISO file then note that most 3rd party software can only use level 1. [This Wikipedia entry](#) has detailed information on the levels.
- **Use ISO file delimiters (';1') (less compatible):** This defines if delimiters are used in the filenames. It is not recommend that this setting be enabled.
- **Use long Joliet names:** This defines the filename limitations and affects the compatibility of ISO files. If you are copying to an ISO file then note that some 3rd party software may not be able to read ISO images if this option is enabled.
- **Cache the files (useful when there are many small files or copying from the network):** If enabled then part of your hard drive is used to cache files. If you are copying many small files, copying from a network, or copying from a slow device, then it is recommended you enable this option.
- **Do not automatically add the profile to the backup:** By default the profile is automatically exported and included on the CD/DVD/ISO. It is called **SYNCBACK.sps** and placed in the root folder on the CD/DVD/ISO. If you need to restore from the CD/DVD/ISO, and do not have the profile, then you can re-create the profile by importing this SPS file. If you do not want the profile included on the CD/DVD/ISO then enable this option. **If you are using compression with encryption then note that the password is stored in the profile.**

3.3.43 Cloud

Expert Mode: Cloud

 These options are only available in the Pro version. This profile settings page can use and create shared settings.

SyncBackSE can backup and synchronize files with the Amazon S3 or Microsoft Azure cloud storage services. This means your files will be securely stored on their servers, and having an offsite backup of your files is highly recommended. Before you can use this feature you must create an account with Amazon or Microsoft to use their cloud service. Once you've created an account you will receive your account details which you will need to use their service.

There are some cloud services which are compatible with Amazon S3. For example, Google Storage (not to be confused with Google Docs) is an Amazon S3 compatible service. To use such a service with SyncBackSE you simply need to set the **Cloud Service** and **Service URL** as appropriate.

- **Destination/right files are on a cloud storage server:** If ticked, then the destination/right is a compatible cloud storage service, i.e. you are backing up to or synchronizing with a cloud service.

Server Connection Details

- **Cloud Service:** Select the appropriate type of cloud storage service that is to be used.
- **Service URL:** If you are using Amazon S3 or Microsoft Azure then it is recommended that you leave this setting as **[default]**. You only need to change this setting if you are using a service which is compatible with Amazon S3 or Microsoft Azure. For example, Google Storage is an S3 compatible service. In these cases you must use the URL supplied by the compatible service (otherwise it would connect to the Amazon or Microsoft servers). For example, at time of writing the URL for Google Storage is **comdatastorage.googleapis.com** (note that you **do not** prefix the URL with http or https).
- **Username:** On Amazon S3 this is called the access key you want to use to connect to the S3 service. Microsoft Azure calls this the account.
- **Password:** On Amazon S3 this is called the secret key. Microsoft Azure calls this the access key. You can optionally have SyncBackSE prompt you for the password instead of entering it here.
- **Prompt for the password when run (profile will fail if run unattended):** If this option is enabled then every time the profile is run SyncBackSE will prompt you for the password. If the profile is being run unattended, then no prompt will be displayed and the profile run will fail.
- **Use encrypted (https) connection:** If this option is enabled then all communication with the cloud servers is encrypted. This does not mean the files are encrypted, it means

that all the communication is encrypted. Note that encrypting the connection may reduce performance. If you want to store your files encrypted you must use the encryption settings.

- **Bucket / Container:** On the Amazon S3 service, all files must be stored within a **bucket**. Microsoft Azure has the same concept but calls it a **container**. You can have multiple buckets/containers (like you can have multiple drives on a computer) but a profile can only backup/sync with one bucket/container (other profiles can of course use other buckets/containers). An Amazon S3 bucket name must be globally unique (meaning nobody else can use the same bucket name). A Microsoft Azure container name does not need to be globally unique. Buckets/containers need to adhere to some naming restrictions (these are restrictions of the service, and not of SyncBackSE, but you when using Amazon S3 you can optionally relax the rules):
 - The maximum length of a bucket/container name is 63 bytes and the minimum length is 3 bytes
 - Bucket/container names must start with a lowercase letter or a number
 - Bucket names can only contain lowercase letters, numbers, periods (.), and dashes (-). Azure containers have the same restriction except they cannot contain periods (.)
 - Bucket names cannot contain two adjacent periods, e.g. a bucket cannot be called **bad..name**. Azure containers cannot contain any periods.
 - Bucket names cannot contain dashes next to periods, e.g. a bucket cannot be called **a.-bad-.name**. Azure containers cannot contain any periods.
 - Containers cannot contain consecutive dashes, e.g. a container cannot be called **bad--name**
 - Bucket/container names should not end with a dash, e.g. a bucket cannot be called **bad-name-**
 - The name must not be formatted as an IP address (e.g., 192.168.5.4). Azure containers cannot contain any periods.

Amazon S3 Bucket Names and Locations

Files within a bucket can be accessed via a web browser, so you may want to keep this in mind when deciding on a bucket name. For example, if you created a bucket called **companyname.com** then you could access the files in that bucket using the URL **http://companyname.com.s3.amazonaws.com/filename**. By default files created in a bucket cannot be accessed via a browser because the default access policy is **private**. You can change this on the advanced settings page.

Another important factor is that buckets are location dependent, which means the files in a bucket are physically located in a specific place. When you create a bucket you can choose its physical location. Obviously performance is going to be affected by where you are accessing the files in the bucket from and where the bucket is located.

Microsoft Azure \$root container

The name **\$root** is a special container name in Microsoft Azure. A root container serves as a default container for your storage account. A storage account may have one root container. The root container must be explicitly created and must be named \$root. A blob

(file) stored in the root container may be addressed without referencing the root container name, so that a blob can be addressed at the top level of the storage account hierarchy. For example, you can now reference a blob that resides in the root container in the following manner:

`http://myaccount.blob.core.windows.net/mywebpage.html`

Which cloud storage service should I use?

When deciding on which cloud storage service to use you should base it on what is most important to you:

1. **Price:** The prices of the services differ, but only very slightly. Amazon S3 has the RRS option to reduce costs, but even so Microsoft Azure has very similar pricing. Note that pricing changes and may be based on where (regionally) you decide to store your files
2. **Speed:** The upload and download performance largely depends on where your files are physically located. The closer they are to you, the faster it is that they can be accessed. However, SyncBackSE can upload larger files to Microsoft Azure much quicker than Amazon S3 (unless you choose to split your files on S3). If possible, try the services using a typical set of files (at the same time of day you would use the service) to see any differences in performance.
3. **Size:** Microsoft Azure can store files up to 200GB in size. For Amazon S3 it is 5GB, but you can have bypass this limitation by splitting your files. Also, you could reduce the size of your files by using compression.
4. **Security:** The security and integrity of your files may be paramount. It is impossible to say which service is more secure. You can reduce security risks by telling SyncBackSE to encrypt your files.

Safe Copy performance

The safe copy option will have an impact on performance. If possible, you should switch off this option. Note that you cannot switch off safe copies if you are using versioning. You should also keep in mind that safe copies is designed to avoid file corruption problems to due network failures.

3.3.44 Cloud, Advanced

Expert Mode: Cloud, Advanced

 These options are only available in the Pro version.

- **Allow HTTP compression:** When this option is enabled HTTP compression is used for communication with the cloud server. Note that this does not mean that the files are stored compressed, only that the communication with the server is compression during transmission.

- **Keep HTTP connection alive:** Defines whether the connection to the cloud server is automatically closed after each request. Not all servers support keeping the connection alive.
- **Use Amazon S3 Reduced Rate Storage (RRS):** Amazon introduced the Reduced Rate Storage option to reduce prices. Basically enabling this option means there's a higher risk that they may lose your files. The benefit is that it is cheaper for you to store your files on their servers. You may decide that the risk is acceptable, especially if the files are not critical or that you have other backup copies of the files. For more details on the risk and price reductions from using this option please refer to [Amazon](#). This option is not available with Microsoft Azure (they do not provide such a service). Also, S3 compatible services may not support RRS either, e.g. Google Storage does not. In that case SyncBackPro will silently ignore this setting (however, to improve performance you should switch off RRS).
- **Retrieve a list of all the files and folders then filter (faster in most situations):** There are two methods for SyncBackPro to request a list of all the files and folders from the cloud storage server. It can either request the lists folder by folder (which is how it does it with local drives, FTP servers, etc.) or it can ask the server for a list of all the files and folders in one go (the default). One method may be faster than the other depending on how many of the files and folders on the cloud service are going to be filtered out by your profiles. If lots of the files are going to be ignored (due to filter and file & folder selections) then the profile may run faster if this option is switched off.
- **Ignore Amazon S3 bucket naming rules (not recommended):** Amazon S3 has some strict rules on what bucket names are allowed. You can bypass many of the rules but at the expense of accessing the bucket via a URL or other utilities. If you enable this option then SyncBackPro will ignore the restrictions and let the cloud server decide if the name is allowed or not. Note that the cloud service may modify the name you give it to enforce region/server specific rules. For example, if you create a bucket in the Singapore region then it cannot contain upper-case letters, so if you try and create a bucket called **XYZ** it will instead create a bucket called **xyz**.
- **Number of upload/download threads to use (too many will degrade performance):** When SyncBackPro uploads or downloads files, or retrieves the meta-data on the files, it will do so in parallel by using a number of different threads. This can significantly improve performance, but if too many threads are used then it can also significantly reduce performance (by overloading the network, CPU, and increasing memory usage). By default 5 threads are used.
- **Send and receive timeout (seconds):** Sometimes network problems and disconnections can cause communication with the cloud storage server to freeze, e.g. SyncBackPro may be waiting for a response from the server which may never arrive. You can set a limit for how long SyncBackPro should wait for a response before it tries again. By default the timeout value is 60 seconds.
- **Access Policy applied to uploaded files:** Cloud storage services yet you apply a security access policy for the files you store on their server. By default SyncBackPro will use the highest security setting, i.e. only you and nobody else can access the files.

However, you may be storing files that you want others to be able to access (e.g. via a web page) and so would want a different access policy. Note that there is also an access policy on the bucket/container. Please refer to the cloud service documentation on the meanings of the access policies.

- **To increase upload performance split files into parts, each part with a maximum size of (MBytes):** This option is only available when using Amazon S3 or a compatible cloud service. There are two reasons why you may want to use this option: 1) the maximum file size on Amazon S3 is 5GB so by splitting a larger file into parts you can get around this limitation, 2) when uploading a file only one thread is used so the upload is not as fast as it could be if it was uploaded in parts in parallel. The drawback of using this option is that your files are split into parts, which means if you want to access the files using 3rd party utilities then they'll not know how to reassemble the parts into a whole file. When SyncBackPro accesses the files it will reassemble them transparently, so there is no need for you to do anything or be concerned about it. So, if you are only accessing the cloud files using SyncBackPro then it is recommended that you enable this option so that there is no maximum file size and uploads are quicker. This option is not available when using Microsoft Azure because a file can be uploaded in parallel without needing to split it. Also, the maximum file size on Azure is 200GB, which is considerably larger than 5GB.

Cloud Limitations

- **Maximum Filename Length:** Both for Amazon S3 and Microsoft Azure the maximum length of a filename is around 1024 characters.
- **Maximum File Size:** For Microsoft Azure the maximum file size is 200GB. For Amazon S3 it is 5GB, but you can get around the maximum file size by selecting to split the files into parts.
- **Filenames:** Microsoft Azure does not allow filenames to end with periods (.)

3.3.45 Variables

Expert Mode: Variables

This profile settings page can use and create shared settings.

Variables are strings that are replaced by something else when the profile is run. For example, you may want to backup to a folder that has the current date in its name. To do this you would use the variable `%DATE%`, e.g. set the destination to `X:\Backup\%DATE%\`. This section has more details on what variables are available.

As well as Windows environment variables, SyncBackSE variables, and getting values from the registry, you can also define your own variables. Those variables can reference other variables. For example, you could create a variable called **THEDATETIME** and set it to **The current date is %DATE% and the time is %TIME%**

The variables defined for a profile are only available to that profile. If you define a variable in a group profile then it is available to all profiles in that group. However, group level variables are only available to the non-group profiles in the group, i.e. they do not cascade to the profiles in sub-groups. If a profile has a variable with the same name as one in its parent group then the profile variables value will replace the group variables value, i.e. profile variables take precedence. This setup window will also highlight this by showing the variables in red. As well as this, the log file will also tell you which profile variables are replacing group variables (if at all).

In SyncBackPro , this setup window also displays any variables that are defined by scripts that are run as part of the profile.

To add a variable simply click the **Add** button. You then enter a unique name for the variable, e.g. **MyVariable**, and then enter a value (the value can use variables itself). Do not use percentage signs around the variables name, e.g. use **VarName** instead of **%VarName %**.

3.3.46 Scripts

Expert Mode: Scripts

 Scripting is only available in the Pro version. Using this window you can specify which Runtime scripts will be used with the profile, and also if the destination/right files are managed by a location script. You must first install the script.

- **Destination/right files are managed by the following script:** If the destination/right should be managed by a location script then tick this checkbox and then select the location script.
- **Scripts that should be used when this profile is run (and in the order specified):** Tick the runtime scripts you wish to use when the profile is run. You can define the sort order by clicking on a script then using the up and down buttons to move it.



The order in which the scripts are set to run is important. This is because, in some cases, only one script can perform an action. For example, if you have a runtime script that renames a file then obviously a file can only be renamed once. This means the first script to rename a file is the one that will rename it. Any following scripts cannot rename the file.

3.3.47 Backup Email

Expert Mode: Backup Email

 These options are only available in the Pro version. This profile settings page can use and create shared settings.

Using these settings it is possible to backup your email messages (this is not to be confused with making a backup of your files to an email server or emailing the log). SyncBackSE can make a backup of emails stored on any POP3 or IMAP4 email server.

How do you restore your emails? You cannot restore the emails to your (SMTP) email server using SyncBackSE. Instead you must import the EML files into your email client.

How do you have the emails deleted from the email server after a copy has been made? You must configure the Decisions - Files settings so that the emails are **moved** instead of just being copied.

POP3/IMAP4 Server Connection Details

- **POP3 or IMAP4:** Choose the type of server used to retrieve emails. Typically it is a POP3 email server. Check with your email provider or systems administrator.
- **Hostname:** The hostname (or IP address) of your POP3 or IMAP4 server. Check with your email provider or systems administrator on what that is. Note that in some cases, e.g. Gmail and other web based email services, you may need enable access to your emails via a POP3 (or IMAP4) server.
- **IMAP Folder:** If you are using an IMAP4 email server then you can optionally enter the name of the folder to retrieve your files from. Note that you can use variables here. Also, you may need to first create the folder on your IMAP4 server. It is important to remember that SyncBackSE cannot choose where to store your files as that is controlled by your server.
- **Port:** The port number of your POP3/IMAP4 email server. It is recommended you leave it as zero (then SyncBackSE will use the default port number based on your settings).
- **Must login to POP3/IMAP4 server:** If you must login to your POP3/IMAP4 email server then tick this checkbox and enter your login username and password. Note that some servers require a login whereas others may fail if you do login. Check with your email provider or systems administrator.
- **Prompt for the password when run (profile will fail if run unattended):** If this option is enabled then every time the profile is run SyncBackSE will prompt you for the password. If the profile is being run unattended, then no prompt will be displayed and the profile run will fail.
- **Must use SSL/TLS connection:** If your POP3/IMAP4 email server requires an SSL/TLS encrypted connection, or it supports one and you want your email to be transmitted from the server in encrypted form, then enable this option. Note that your server must support SSL/TLS encryption.
- **Direct SSL/TLS connection (usually using port 995 for POP3, 993 for IMAP4):** If your POP3/IMAP4 server supports a direct encrypted connection then select this option.
- **Use STLS command (POP3 only):** This is different from the above setting in that it

connects to the POP3 server then requests that the connection be encrypted by sending the STLS command to the email server. Choose this option if your SMTP server does not support a direct encrypted connection. This option is not available if an IMAP4 server is being used.

Filenames

- **Filename of EML files:** This is the filename to use for the email files. Each email is downloaded and stored in its own self-contained EML file, i.e. the EML file contains the email body and all attachments. EML is a standard file format used by many of the popular email clients, e.g. Mozilla Thunderbird. You can use special variables for the filename. The default filename is `%EMAIL_SUBJECT% [%EMAIL_ID%].eml`
- **Also export the email bodies and attachments:** If you also want the email stored in plain text and the email attachments saved as-is then you should enable this option.
- **Sub-folder to export the emails and attachments to...:** This is the sub-folder to store the emails in. You can use special variables in the sub-folder name. The default sub-folder is `\\%EMAIL_DATE%\\%EMAIL_ID%\\%EMAIL_SUBJECT%`

3.4 Runtime Help



Run SyncBackSE

Runtime Help refers to operations which occur as SyncBackSE is performing a task like a backup. This section of the help file covers the Differences and File Collision Windows which allow you a great deal of flexibility in fine tuning your profile task.

3.4.1 The Differences Window

Differences

When you run a profile, if there are any differences detected, e.g. files have changed, this differences window will appear. You can configure your profile never to show this window (see the Compare Options page when creating/modifying the profile).



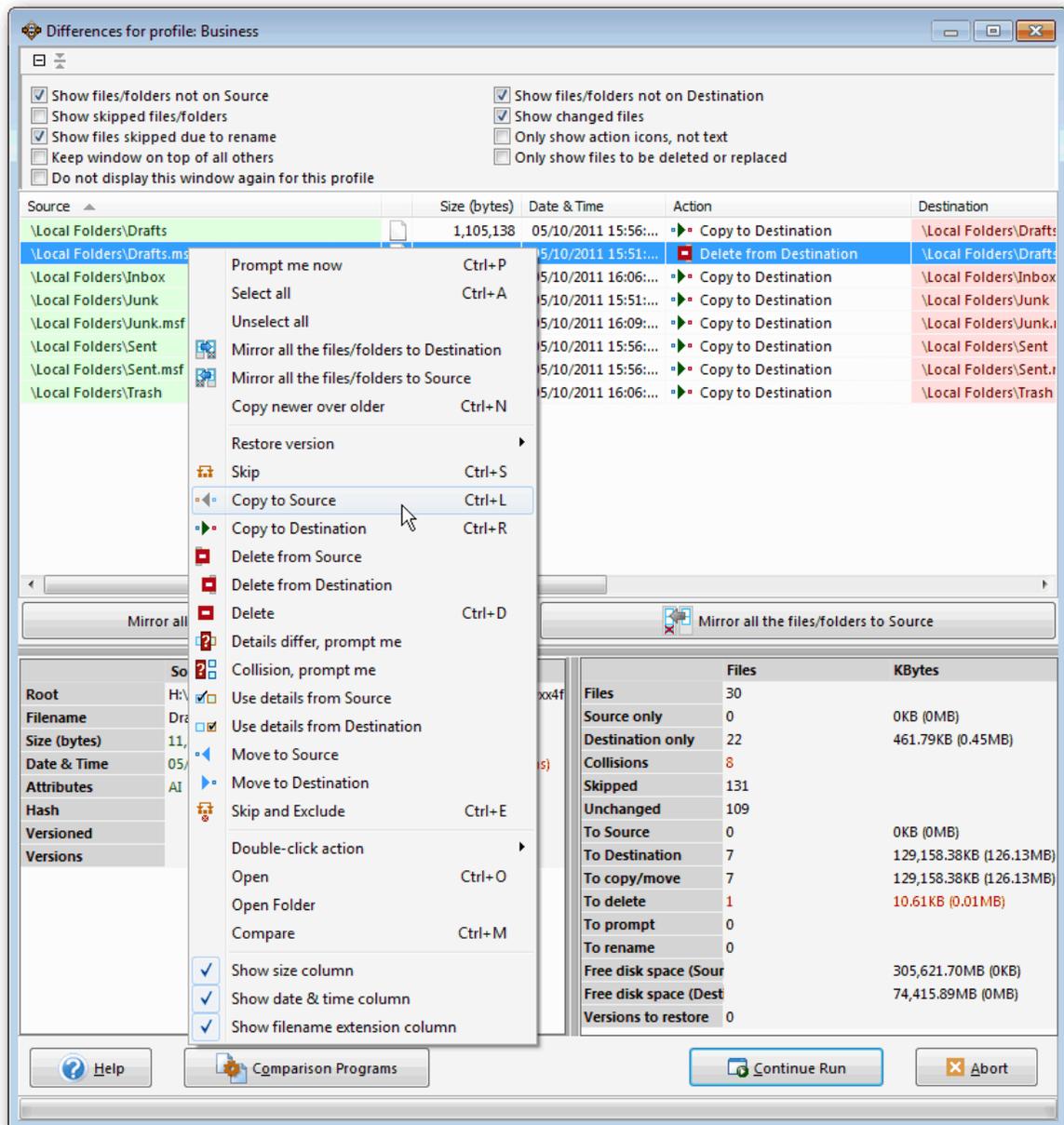
The **Differences** window is not shown for unattended runs, e.g. scheduled tasks, if you run SyncBackSE with command line parameters, if the Skip the Differences

screen option is enabled for the profile, or if there are no differences. There is also an option to automatically close the Differences window if it is empty (due to the filter settings).

The **Differences** window will always be shown when doing a restore or simulated run.

If your profile contains tens of thousands of files (or more) then it is not recommended that the Differences window be displayed. This is because displaying so many files uses a lot of CPU time and RAM, and takes a long time to sort.

When a profile is run, SyncBackSE will compare the source/left and destination/right files to find the differences. There may be files in the source/left that are not in the destination/right, for example. After it has made the comparison it will display the **Differences** window that lists all the differences between the source/left and destination/right. This gives you a chance to see what will happen and change what actions will be taken. For example, a file may be marked for deletion but you may decide not to delete that particular file.



At the **bottom left** of the window are details on the currently selected files. It shows the differences between the file on the left and the file on the right. Newer dates & times, and larger file sizes, are shown in green, older dates & times, and smaller file sizes, are red. If the profile is a **SmartSync** profile then a checkbox labeled **Show previous details of the files** is shown in the bottom left. To show the previous details of a file you must first select the appropriate file and then tick the checkbox. It will now show the details of the file the last time the profile was run. Obviously new files will not have previous details.

The **bottom right** panel gives details on the number of files, and the amount of data, that will be copied, deleted, and skipped.



Note that the free disk space shows the estimated free disk space on the source/left and destination/right based on the actions made. For example, if you choose to delete files from the right then its free disk space will increase.

Mirror all the files to the right: When this button is clicked all the files on the left will have their action changed so that they are copied to the right. **Any files on the right which are not on the left will have their action changed so they are deleted.** This button lets you quickly tell SyncBackSE that the right should have exactly the same files as the left. Note that you can mirror a selection of files via the pop-up menu.

If you keep the **SHIFT** key pressed while clicking the button, then the files will be **moved** instead of copied.

Mirror all the files to the left: This does the opposite of the above button, i.e. the left should have the same files as the right. Note that you can mirror a selection of files via the pop-up menu. You cannot mirror files to the left if you have a traditional fast backup profile.

Between the Left and Right columns is the **Action** column. This shows what will happen to that file. Note that not all the options are available for each file as it depends on what the situation is with that file. For example, you cannot copy the left file if it doesn't exist:

Skip: Ignore the file and do nothing.

Skip and Exclude: Ignore the file and do nothing. The file will be ignored in all future profile runs as well. See the help in the File Collision window for more details.

Copy to right: Copy the left file to the right. This will overwrite the right file, if it exists.

Copy to left: Copy the right file to the left. This will overwrite the left file, if it exists.

▶ □ **Move to right:** Move the left file to the right. This will overwrite the right file, if it exists, and delete the file from the left.

□ ◀ **Move to left:** Move the right file to the left. This will overwrite the left file, if it exists, and delete the file from the right.

□ = □ **Unchanged:** This option is only available when running a **Fast Backup** profile. By selecting this option you are telling SyncBackSE that the file actually has not changed and nothing should be done. This is useful, for example, when you have a copy of a file on an FTP server but the date & times do not match. However, you know the file is identical.

Delete: Delete both the source and destination files.

Delete from left: Delete the file on the left.

Delete from right: Delete the file on the right.

Collision, prompt me: You will be prompted what choice to make after you close this window.

Missing, prompt me: You will be prompted what choice to make after you close this window.

Details differ, prompt me: If the files are identical, but their attributes are different, then you can choose to be prompted on which files attributes to use.

Use details from left: The files are identical, but their attributes are different, so copy the attributes from the left file to the right file.

Use details from right: The files are identical, but their attributes are different, so copy the attributes from the right file to the left file.

There are three ways to change the **Action** for a file:

- Click on the Action item and select the new action from the drop-down list
- Right-click on the row and select the new action from the pop-up menu
- Use hot-keys

You can change the action for multiple files by clicking on the rows and pressing the **Shift** and **Ctrl** keys. You can press **Ctrl-A** to select all the files. Right-click on the selection and choose the action from the pop-up menu. Only actions that are available for all the selected files are shown.

You can use the following hot-keys to change the action of the selected items:

- **Ctrl-S:** Skip the files
- **Ctrl-E:** Skip and exclude the files
- **Ctrl-L:** Copy files to the source/left
- **Ctrl-R:** Copy files to the destination/right
- **Ctrl-D:** Delete the files from both the source/left and destination/right
- **Ctrl-U:** Mark the files as unchanged (Fast Backup profiles only)
- **Ctrl-N:** The newer file will replace the older file (any version changes are lost)

To be prompted immediately on what to do with a file you can select the file (by clicking on its row) and pressing **Ctrl-P** (or by double-clicking the row, or selecting '**Prompt me now**' from the pop-up menu). The File Collision Window will appear from which you can make a choice. Note that nothing is done with the file until after the **Differences** window is closed.

There is a roll-out bar at the top of the Window. Click on it to display other options:



- **Show files/folders not on left:** If not selected then files and folders that are only on the right are hidden.
- **Show files/folders not on right:** If not selected then files and folders that are only on the left are hidden.
- **Show changed files:** If not selected then files that are on both the left and right, and are different, are hidden.
- **Show skipped files/folders:** If not selected, then skipped files and folders are not shown.
- **Only show action icons, not text:** If selected then the **Action** column will just show icons and not text. This reduces the width of the Action column giving you more space.
- **Keep window on top of all others:** If selected then the Differences window will be placed above all other windows on your desktop.
- **Only show files to be deleted or replaced:** If selected then only files that are going to be deleted or replace will be shown. No folders will be displayed. Note that if a file is being moved, and is not replacing a file, then that is not regarded as a file that is to be deleted. To avoid performance problems the display is not automatically updated if you change the action of a file, so to refresh the display you must disable and re-enable this option. Also, unlike the other settings in the roll-out bar, this setting is not saved.
- **Do not display this window again for this profile:** If ticked then this window will not be displayed again when this profile is next run. This is identical to the "**Skip the Differences screen when this profile is run (it is never shown when unattended)**" option in the **Compare Options** page in the profiles configuration. Note that this checkbox is not shown when the run is a simulation or a restore (as the **Differences** window is always shown when a profile is run in simulation or restore mode).

After you have reviewed the differences, made whatever changes are required (if any), and are ready to continue with the profile run, you can click either the '**Continue**' button or the '**Abort**' button. Aborting will stop the profile run immediately and no files will be copied, deleted, or moved.

Restoring Versions

Versions of files are restored via the **Differences** window. One way to ensure the

Differences window is displayed is to run the profile by using **Ctrl-R**. If it's a backup profile you could also run it as a Restore.

When there are versions of a file available then a V graphic () is displayed next to the filename. If a file doesn't exist, but it does have versions, then the filename will be post-fixed (suffixed) with the text **[Versions]**.

To restore a version of a file simply click on the V graphic and choose the version to restore from the drop-down list. The list gives the date & time when the version was made (it is not the last modification date & time, or creation date & time, of the file). Once a version to restore is chosen the filename shown will be post-fixed with the date & time of the version to be restored.

Versions cannot be restored if the **Action** is such that restoring the version would be pointless. For example, if you want to copy a source file to the destination (to replace the existing destination file), then you cannot restore a version of the destination file. It would make no sense because SyncBackSE would need to restore the destination version, then copy the source file to the destination, which would then replace the version you wanted to restore. In this case the text **[unavailable]** will appear in the pop-up menu. If you just want to restore the version change the Action to Skip and then choose the version to restore.

Keep in mind that restoring a version means any existing file will be versioned before it is replaced. For example, you have a source file that has one version. If you restore the version then a version of the existing source file will be made before the version is restored (basically the files are switched). This makes sense because you may later realize you made a mistake then it's very simple to correct it (you just restore the latest version, which was your original file).

It's possible to restore versions for multiple files. First, select the files you want to restore versions for (e.g. press Ctrl-A to choose all the files, or hold down the Ctrl key and click on a file to add it to the selection, or use Shift key and click to select a block of files). Next, right-click on the filename of one of the files you have selected. A pop-up menu will appear. Choose Restore Version, then Source or Destination (depending on where you want the versions restored), then you can choose either of the following options:

- **Restore latest version (except if backup file available)** - If the file has one or more versions, and doesn't exist in the source/destination, then it will restore the newest version of the file. If the file does exist then it won't do anything with that file.
- **Restore latest version (even if backup file available)** - If the file has one or more versions then it will restore the newest version of the file.

Comparing Files

SyncBackSE can tell you files are different, and using third party programs, you can ask SyncBackSE to show you the actual differences between the files. Click the Comparison Programs button to tell SyncBackSE which programs to use to compare which types of files.

To compare files simply click the file (or files to compare multiple files) and press **Ctrl-M**, or select **Compare** from the pop-up menu. You can also configure the double-click action to

compare files. If a suitable comparison program is not available for the file type then the files will be opened.



Note that the files must be retrieved to the local drive for comparison. If you have large files on FTP servers, or slow networks, then there may be a delay while retrieving the files.

Opening Files

To open (view) a file simply click the file (or files to compare multiple files) and press **Ctrl-O**, or select **Open** from the pop-up menu. You can also configure the double-click action to open files.

You can also drag & drop files from the Differences window onto your desktop, for example. Click on the file, keep the left-mouse button pressed, drag it to your desktop, and release the mouse button.



Note that the files must be retrieved to the local drive to be opened. If you have large files on FTP servers, or slow networks, then there may be a delay while retrieving the files.

Collisions

A "collision" is when a file in the source and destination differ, yet have the same name. That is, the file is both in the source and destination but is modified in some way, perhaps by date, size etc.

A notification of collisions occur in the "Differences" window which appears by default when making a backup (note however there are circumstances when the "Differences" window does not appear, for example when the user has chosen not to show the window):

	Files	KBytes
Files	146	
Source only	130	443.50KB (0.43MB)
Destination only	10	48,638.18KB (47.50M)
Collisions	6	
Skipped	507	
To Source	0	0.00KB (0.00MB)
To Destination	136	902.51KB (0.88MB)
To copy/move	136	902.51KB (0.88MB)
To delete	0	0.00KB (0.00MB)
To prompt	0	

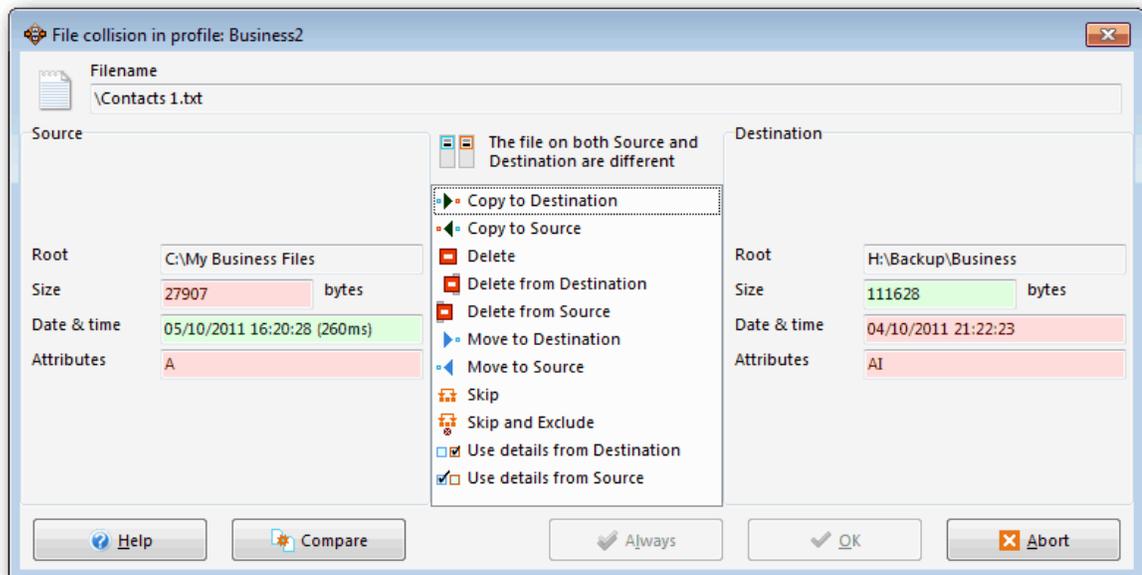
At the bottom of the window are two buttons: **Continue Run** (with a green checkmark icon) and **Abort** (with a yellow 'X' icon).

Collisions are shown in red in the "Differences" window to highlight there are going to be changes made when you continue the profile task. If the user views the Differences window carefully, the user has the option to make choices about whether they want to accept the changes SyncBackSE will make, or bypass them with a right click and choose a different action. For more read about the Collision Window.

3.4.2 The File Collision Window

File Collision

When you run a profile, and you have enabled prompting, then the **File Collision** window will appear when a decision is required from you on what to do with the file. Note that for unattended profile runs, e.g. when scheduled, this window does not appear and instead the file will be skipped.



On the **Decisions Files** and **Folders** pages, in Profile Setup, you have the option of asking SyncBackSE to prompt you under certain circumstances, e.g. if a file is in both the source/left and destination/right, but the files are not the same. For example, if you created a backup profile, then run the profile, and edit a file in the source, you would then have a file in the destination that is not the same as the one in the source. On the next run of the profile, if you have configured your profile to be prompted, then a window will appear asking you what action to take for this file.

The window has the filename at the top. If the filename is too long to fit in the edit box, you can resize the window. Information is also shown on where the file is in the source/left and destination/right, the size of the file, its last modification date & time, its attributes, and its hash value (if you have configured the profile to use hashing for file comparisons). When there are differences the values are highlighted in green or red. For example, if the left/source file is newer than the right/destination file then its date & time is shown in green whereas the right/destination files date & time is shown in red.

There are two sets of information: for the file on the left, and for the file on the right. Between

the left and right files details is a box listing what actions you can take. There are a number of options available to you, some of which may not be shown depending on how you've configured your profile and what the differences are:

Skip: No action will be taken with this file. The file will not be copied, deleted, or moved, and it will be ignored. If you click the **Always** button then you won't be prompted again, during this profile run and in this kind of situation, and instead the files will be skipped.

Skip and Exclude: No action will be taken with this file. The file will not be copied, deleted, or moved, and it will be ignored. The difference between this action and the plain **Skip** option is that the file will now always be skipped, including in any future profile runs. It is equivalent to de-selecting the file in the Sub-directories and files window in the profile configuration. If it is a simulated run then the file is simply skipped as per normal and not excluded from future profile runs. Note that the **Always** button cannot be used with Skip and Exclude to avoid accidentally excluding numerous files and folders. Also, this action is not available if the profile is configured not to use file and folder selections or it is a Fast Backup profile that does not use the archive attribute.

Copy to left: The file on the right will replace the file on the left. If you click the **Always** button then you won't be prompted, during this profile run and in this kind of situation, and instead the left file will be replaced by the right file.

Copy to right: The file on the left will replace the file on the right. If you click the **Always** button then you won't be prompted again, during this profile run and in this kind of situation, and instead the right file will be replaced by the left file.

- ▶ **Move to right:** Move the left file to the right. This will overwrite the right file, if it exists, and delete the file from the left. If you click the **Always** button then you won't be prompted again, during this profile run and in this kind of situation, and instead the file on the left will be moved to the right.
- ◀ **Move to left:** Move the right file to the left. This will overwrite the left file, if it exists, and delete the file from the right. If you click the **Always** button then you won't be prompted again, during this profile run and in this kind of situation, and instead the file on the right will be moved to the left.
- ◻ **Unchanged:** This option is only available when running a **Fast Backup** profile. By selecting this option you are telling SyncBackSE that the file actually hasn't changed and nothing should be done. This is useful, for example, when you have a copy of a file on an FTP server but the date & times do not match. However, you know the file is identical.

Delete: The file will be deleted from the left and right. If you click the **Always** button then you won't be prompted again, during this profile run and in this kind of situation, and instead the files will be deleted.

Delete from left: The file on the left will be deleted. If you click the **Always** button then you won't be prompted again, during this profile run and in this kind of situation, and

instead the file on the left will be deleted.

Delete from right: The file on the right will be deleted. If you click the **Always** button then you won't be prompted again, during this profile run and in this kind of situation, and instead the file on the right will be deleted.

Missing, prompt me: The file is either not on the left or right, and you want to be prompted later on what to do when the profile run continues. Note that this option is only available when prompted from the **Differences** screen. After that screen is shown (or if you've configured the profile not to show that screen) then this option is not available.

Collision, prompt me: The file is both on the left and right, and you want to be prompted later on what to do when the profile run continues. Note that this option is only available when prompted from the **Differences** screen. After that screen is shown (or if you've configured the profile not to show that screen) then this option is not available.

Details differ, prompt me: The files are identical but the attributes and/or last modification date & time are different, and you want to be prompted later on what to do when the profile run continues. Note that this option is only available when prompted from the **Differences** screen. After that screen is shown (or if you've configured the profile not to show that screen) then this option is not available.

Use details from left: The files are identical but the attributes and/or last modification date & time are different, but you want to copy the attributes and date & time from the file on the left to the file on the right. If you click the **Always** button then you won't be prompted again, during this profile run and in this kind of situation, and instead the details from the file on the left will be used.

Use details from right: The files are identical but the attributes and/or last modification date & time are different, but you want to copy the attributes and date & time from the file on the right to the file on the left. If you click the **Always** button then you won't be prompted again, during this profile run and in this kind of situation, and instead the details from the file on the right will be used.

If the profile is a **SmartSync** profile then a checkbox labeled **Show previous details of the files** is shown in the bottom left of the window. To show the previous details of a file you must first select the appropriate file and then tick the checkbox. It will now show the details of the file the last time the profile was run. Obviously new files will not have previous details.

After making your decision you can click the **OK** or **Always** button. If you click the **Abort** button then no action is taken and the profile run is immediately stopped.

Always

What other files are affected when you click the **Always** button is dependent on the situation. There's an **Always** action each for:

- Collision (file on both sides and they are not the same)
- The file details differ (same file contents but different file attributes)
- File only on the left/source
- File only on the right/destination

For example: if you had a file on both sides, and they are different (i. e. it's a collision), and you selected an action and clicked **Always**, then that action will be automatically used for files that are on both sides and are different (instead of you being prompted). If, for example, there is a file only on the left/source then the action does not apply to that. Note that the **Always** decision applies only for the current profile run, i. e. it doesn't apply the next time the profile is run, or the next profile in a group.

The **Always** button cannot be used with the **Skip and Exclude** action.

Compare

SyncBackSE can tell you files are different, and using 3rd party programs, you can ask SyncBackSE to show you the actual differences between the files. To compare the files simply click the **Compare** button. Only when there is a file both on the left and right can the files be compared. If a suitable comparison program is not available for the file type then the files will be opened.

Note that the files must be retrieved to the local drive for comparison. If you have large files on FTP servers, or slow networks, then there may be a delay while retrieving the files.

Shortcut Keys

A number of shortcut keys are available to help users who are familiar with the program make choices quickly:

L - Copy the file to the left

R - Copy the file to the right

D - Delete the file Note **both** files will be chosen for deletion, but if that option is not available, then the file on the right will be chosen for deletion, and if that option is not available, then the file on the left will be chosen for deletion.

P - Prompt later for what action to be taken

S - Skip the file

E - Skip and exclude the file

U - The file is unchanged (available only with Fast Backup profiles)



Note that if you use the **Ctrl** key with these shortcut keys then the action is immediate. For example, if you press **Ctrl-R** then the action to copy the file to the right is chosen, and the prompt window is closed.

3.5 Technical Reference



Technical Articles

The Technical Reference of this help file provides detailed information about:

Scripting

Command Line Parameters

Filter Settings

Open and Locked File Copying

Variables

Regular Expressions

Uninstalling SyncBackSE

SyncBack Management Service

3.5.1 Scripting

Scripting

 Scripting is a Pro only feature. This section of the help file provides information about scripting support. Scripting is a way in which the functionality of SyncBackPro can be changed or extended by writing small scripts. A script is a set of instructions and is similar to the macro support in Microsoft Office, and Java Script in web pages. It is also similar to plug-ins in other software, e.g. WinAmp. SyncBackPro can use scripts written in any scripting language (supported by the Windows Scripting Host) that is installed on the computer, but typically they are written in **VBScript** or **JScript**. We cannot provide a tutorial on how to write scripts, but the following web pages can get you started on learning VBScript:

VBScript Tutorial: <http://www.w3schools.com/vbscript/default.asp>

VBScript Introduction: <http://www.devguru.com/technologies/VBScript/home.asp>

VBScript Reference: <http://www.pctools.com/guides/scripting/id/2/?act=reference>

VBScript Language Reference: <http://msdn.microsoft.com/en-us/library/d1wf56tt.aspx>

Microsoft Scripting Center: <http://www.microsoft.com/technet/scriptcenter/default.mspx>

Scripting: <http://msdn.microsoft.com/en-us/library/ms950396.aspx>

SyncBackPro also comes with many example scripts.

Scripting Overview

When SyncBackPro performs certain actions it will check to see if there are any scripts installed that can be called when the action is performed. For example, whenever the profiles are listed in the main window it checks to see if there are any main interface scripts installed. If so, it checks if those scripts can be called, and if so, calls the appropriate function in the script. In technical terms, these are events. You write functions in the scripts that perform actions when certain events occur. To make it easier for the scripts to communicate with SyncBackPro it also provides helper objects, e.g. SBSsystem. Depending on the script type, one or more of those helper objects will be available to the script.

Installing Scripts

To start using a script you must first install it via **Preferences -> Scripts** in the main menu. Note that installing a script does not automatically make the script active. If it's a Main Interface or Profile Configuration script then you must install it then go to the relevant tab and tick the checkbox for that script. To use a Runtime script in a profile, after installing it you must select it in the Scripts page of the profiles configuration. If it's a Location script then you must create (or modify) a profile so it does a backup or sync with that script.

You can also install scripts via the command line interface by simply passing the filename of the script (the same way as importing profiles).

Exporting Scripts

To export a script, go to the Scripts window (via **Preferences -> Scripts** in the main menu), select the script (or scripts) you wish to export, then right-click on the selection and choose **Export** from the pop-up menu.

Script Order

The order in which the scripts are set to run is important. This is because, in some cases, only one script can perform an action. For example, if you have a runtime script that renames a file then obviously a file can only be renamed once. This means the first script to rename a file is the one that will rename it. Any following scripts cannot rename the file.

Script Types

There are four different types of scripts:

1. **Main Interface** scripts: these are scripts that can be used with the main user interface in SyncBackPro. For example, you could write a script that adds columns to the main window.
2. **Profile Configuration** scripts: these are scripts that be used when configuring a profile.
3. **Location** scripts: these are scripts that are used with profiles to store and retrieve files. For example, you could write a location script that copies files to and from a database.
4. **Runtime** scripts: these are scripts that are used by profiles when they are run. For example, you could write a profile to decide on which files to copy, or add columns to the Differences window.

A single script file can be more than one type of script. For example, you could write a script that is both a main interface and runtime script. SyncBackPro knows what type a script is because the script tells SyncBackPro via the Description function. It also knows what scripting language is being used based on comments in the header (first 10 lines) of a script, and failing that, the filename extension of the script file.

For example, the following VBScript is a profile configuration, runtime, and location script:

```
'
' Use SBLang to define what language this script is in:
'
' SBLang=VBScript
'
Function Description(ScriptType)
    Description = "The description of this script!"
    ScriptType = 11
End Function
```

Scripts Objects

SyncBackPro makes a number of objects available to scripts to help them interface with SyncBackPro. Which objects are available depends upon the type of script:

- **SBLocation**: This object is only accessible from Location scripts. It provides information and control over the source/left and destination/right locations.
- **SBProfile**: This object is only accessible from Profile Configuration scripts. It allows you to create a page in the profile setup window.
- **SBRunning**: This object is only accessible from Runtime scripts. Using this object you can access the runtime information of a profile.
- **SBSystem**: This object is accessible from any type of script. It provides general functions, e.g. hashing.
- **SBVariables**: This object is accessible from Main Interface, Runtime and Profile Configuration scripts. It allows you access to the profile and program variables.
- **SBHistory**: This object is accessible from Main Interface, Runtime and Profile Configuration scripts. Using it you can get access to the runtime history of a profile.

Scripts Online

If you have scripts you wish to share, or want to download more scripts, visit the following web page:

<http://www.2brightsparks.com/syncback/scripts/index.html>

3.5.1.1 Base

These are functions that should be defined in all scripts regardless of their type. All scripts must implement the Description function, but the others are optional. The functions are called by SyncBackPro at the appropriate time.

Function Description(ByRef ScriptType)

ScriptType: Set this to the type of script

Return value: A short description of the script (this is shown to the user)

This function is mandatory. All scripts must declare this function.

Called to get a description of the script and what type of script it is. The description is not stored and so can be dynamic, however this should not be relied upon in future. The description is stripped of newline and carriage return characters.

The script type is an integer that defines when the script should be used by SyncBack. It is a bitmask (i.e. the values can be or'ed together) of the following types:

0 = Not a valid script

1 = A script that can be used when configuring profiles (a 'configuration' script)

2 = A script that can be used by profiles at run-time (a 'run-time' script)

4 = A script that can be used by in the main user interface (a 'main' script)

8 = A script that can be used by profiles to store and retrieve files (a 'location' script)

A single script file can be used in more than one place. For example, if a script was both a location and a run-time script then its ScriptType value would be 10 (which is 2 + 8). The script type is stored by SyncBack when the script is installed.

```
Function Description(ByRef ScriptType)
    Description = "A short description of what this script does"
    ScriptType = 10
End Function
```

Function FilesToExport(Interactive, Counter)

Interactive: Passed as True if the script can prompt the user

Counter: This is initially passed as 0 (zero) and incremented on each call

Return value: The complete filename of a file to be exported along with the script

Called when a script is being exported as part of a profile. This function is called repeatedly until an empty string is returned. For the first call the Counter value is zero, and is incremented on each call.

The script should return the filenames (one per call) of files that SyncBack should include into the exported profile (the .SPS file). The filenames must be complete filenames including the drive and path. Note that when the script is imported, the files (and the script) will all be put into their own unique folder and then the Install function will be called (the script can then move those files, if required, and do any other installation tasks, e.g. register COM objects).

Do not include the filename of the script itself as that is included automatically. This function does not need to be defined if the script has no accompanying files.

If Interactive is True then the script can prompt for user input, otherwise it must not ask for user input (e.g. dialogs boxes should not be displayed).

```
Function FilesToExport(Interactive, Counter)
  If Counter = 0 Then
    FilesToExport = "C:\abc\def\ghi.txt"
  ElseIf Counter = 1 Then
    FilesToExport = "D:\another\folder\file.exe"
  Else
    FilesToExport = ""
  End If
End Function
```

Function Install(Interactive)

Interactive: Passed as True if the script can prompt the user

Return value: An error message on failure

Called when the script is installed into SyncBack. On failure it should return an error message, in which case the script will not be installed.

This function is called when a script is installed, either by the user via the Script window, or via the import of a profile that uses scripts. It can be used, for example, to move files used by the script to their correct places or to register COM objects. Note that if a script is already installed then Install is not called.

You should not prompt the user, or expect user input, if Interactive is passed as FALSE.

```
Function Install(Interactive)
  If Interactive Then
    SBSsystem.Say "Installed"
  End If
End Function
```

Sub Uninstall()

This function is called when the script is uninstalled from SyncBack. It should assume no user is present, i.e. it should not prompt the user for input.

```
Sub Uninstall()  
    SBSsystem.Say "Uninstalled"  
End Sub
```

3.5.1.2 Main Interface Scripts

These are functions that are defined in your **Main Interface** script and are called by SyncBackPro. A main interface script can enhance or change the main user interface. For example, you can add columns to show extra information about profiles, or perform some action when a key is pressed. The functions are called by SyncBackPro at the appropriate time. Main Interface scripts have access to the SBSsystem, SBVariables, and SBHistory objects.

Function MainColumnHint(Col, IsGroup, ProfileName)

Col: The column number

IsGroup: True if the profile is a group

ProfileName: The name of the profile

Return value: The hint string to display

This subroutine is called from the main window when a custom column hint is required. The first custom column is column zero (0). The script is only called for its custom columns and not for columns created by other scripts.

Function MainColumnsCount()

Return value: The number of custom columns, or zero if none are required

This function is called from the main window to ask the script how many custom columns it wants created in the main window. Note that this value is cached so the function is only called once (when the main window appears, i.e. when the program is run).

Function MainColumnSort(Col, IsGroup1, IsGroup2, ProfileName1, ProfileName2)

Col: The column the display is being sorted on

IsGroup1: True if ProfileName1 is a group

IsGroup2: True if ProfileName2 is a group

ProfileName1: The name of a profile

ProfileName2: The name of a profile

Return value: An integer (<0 if profile 1 should go before profile 2, 0=same profile, > 0 if profile 1 should go after profile 2)

This function is called from the main window when a custom column is being sorted. The first custom column is column zero. The script is only called for its custom columns and not for columns created by other scripts.

Important: As little processing or disk reading as possible should be done in this subroutine. Whenever possible use in-memory caching. See the History.vbs script as an example.

Function MainColumnText(Col, IsGroup, ProfileName)

Col: The column number

IsGroup: True if ProfileName refers to a group profile

ProfileName: The name of the profile

Return value: The text to display in the column

This function is called from the main window when text for a custom column is required. The first custom column is column zero (0). The script is only called for its custom columns and not for columns created by other scripts.

Important: As little processing or disk reading as possible should be done in this subroutine. Whenever possible use in-memory caching. See the History.vbs script as an example.

Function MainColumnTitle(Col, ByRef Width)

Col: The column number

Width: Set it to the width the column should be (in pixels)

Return value: The columns title

This function is called from the main window when the title for a custom column is required. The first custom column is column zero (0). The script is only called for its custom columns and not for columns created by other scripts.

Function NewVersionCheck(ByRef ErrMsg)

ErrMsg: If a check cannot be made then set an error message, else set to an empty string

Return value: If a new version is available then return the URL to open the web browser with, else return empty string

This function is called when SyncBack checks to see if a new version of SyncBack is available. The function should not prompt the user as update checks are usually made in the background. Also take into account that it may be unattended.

- If any script returns an error message then the update check is aborted.
- If any script returns a URL then the update check is aborted and the user is told a new version is available.
- If a script returns the URL string "*" then the update check is aborted and the user is told there is no new version.
- If all the scripts return an empty string URL, and no error messages, then SyncBack will check the 2BrightSparks web site to see if there is a new version available.

Function PollingRefresh()

Return value: True if the script wants RefreshDisplayEx or RefreshDisplay called even in there is no reload or refresh of profile data

This function is called from the main window to ask the script if it wants to have RefreshDisplayEx or RefreshDisplay called even when no profiles have been reloaded or refreshed.

It is called just before MainColumnsCount and is only called once on program startup. If the function is not defined then it is assumed the script does not want to be called.

Sub MainColumnClicked(Col, IsGroup, ProfileName)

Col: The column number

IsFile: True if it is a group profile

ProfileName: The name of the profile

This subroutine is called from the main window when a custom column is clicked. The first custom column is column zero (0). The script is only called for its custom columns and not for columns created by other scripts.

Sub MainEnded()

This subroutine is called when SyncBack stops. The script should not prompt the user or expect any user interaction. It should also not try to stop the program from exiting.

Sub MainFocusChanged(IsGroup, ProfileName)

IsGroup: True if ProfileName refers to a group

ProfileName: The name of the profile

This subroutine is called from the main window when the focused node changes.

Sub MainKeyPress(Key, Shift, IsGroup, ProfileName)

Key: The key that as pressed

Shift: The shift state

IsGroup: True if ProfileName refers to a group profile

ProfileName: The name of the profile

This subroutine is called from the main window when a key is pressed. It is called for each selected row. It is not called if the Delete key is pressed (as that is handled by SyncBack itself).

The Key value refers to the virtual key codes.

The Shift state can be a selection values (see RunDiffKeyPress for details)

Sub MainStarted(Unattended)

Unattended: If True then do not prompt the user or expect any user interaction

This subroutine is called when SyncBack starts.

Sub RefreshDisplay()

This subroutine is called when SyncBack refreshes or updates the main display. It is deprecated and instead RefreshDisplayEx is recommended.

Important: As little processing or disk reading as possible should be done in this subroutine. Whenever possible use in-memory caching. See the History.vbs script as an example.

Sub RefreshDisplayEx(Reloading, Refreshing)

Reloading: Passed as TRUE if the list was reloaded

Refreshing: A Scripting.Dictionary object containing the names of the profiles that were refreshed

This subroutine is called when SyncBack refreshes or updates the main display.

Reloading is TRUE if the list of profiles has been reloaded. This usually happens when a profile has been deleted, created, or renamed. If a profile has been renamed then the old profile name is passed in Refreshing. On program start, and when the users refreshes the list by pressing V5, for example, Reloading is passed as TRUE and Refreshing contains just one empty string (meaning all profiles are being refreshed).

Refreshing is a Scripting.Dictionary object that lists the names of all the profiles that were refreshed (in the key, the item is always empty). This usually happens when a profile has

been modified. If Refreshing contains just one key, and the key is an empty string, then all the profiles have been refreshed.

If Reloading is FALSE and Refreshed is empty then this is just a polling call. For example, if nothing is happening (which includes profiles running), then it may be a polling call. Note that when a profile starts and stops then RefreshDisplayEx (or RefreshDisplay) is called with the name in Refreshing.

How frequently a polling call is made depends on the refresh rate set by the user in the Global Settings. In most cases you won't need to do anything and so your script will not be called. However, if you do want the script to be called in these cases then see PollingRefresh.

Important: As little processing or disk reading as possible should be done in this subroutine. Whenever possible use in-memory caching. See the History.vbs script as an example.

3.5.1.3 Profile Configuration Scripts

These are functions that are defined in your **Profile Configuration** script and are called by SyncBackPro. The functions are called by SyncBackPro at the appropriate time. Profile Configuration scripts have access to the SBProfile, SBSsystem, SBVariables, and SBHistory objects.

Function ConfigCanClose()

Return value: TRUE if the setup page should not be allowed to close

This function is called by SyncBack to ask the script if the setup page can be closed. The script may not want the setup page to close if, for example, some settings are incorrect (however, in this case it is advised that the script simply not save invalid settings).

Function ConfigCanRevert()

Return value: TRUE if the script supports reverting to factory defaults

This function is called by SyncBack to ask the script if it can revert its settings to the factory defaults. Note that it should not revert to the factory settings in this call (for that see ConfigFactoryDefaults).

Function ConfigLoadSettings()

Return value: Return an error message on failure

This function is called to tell the script to load its settings and update the display to show those settings. For example:

```
' Create setup window display
Sub ConfigSetupDisplay()
    SBProfile.AddEdit "The name of the process you want to have finished running",
128, FALSE, FALSE, 1
    SBProfile.AddEdit "The number of seconds to wait before re-checking if it is still
running", 10, TRUE, FALSE, 2
End Sub

' Load settings. Return error message on failure.
Function ConfigLoadSettings()
    ConfigLoadSettings = ""

    SBProfile.SetEdit SBVariables.GetProperty("WFProcessName", "Notepad.exe", FALSE),
1
    SBProfile.SetEdit SBVariables.GetProperty("WFRetrySecs", 5, FALSE), 2
End Function
```

Function ConfigNodeCaption()

Return value: The caption to use in the profile setup window

This function is called to get the caption to use in the profile setup window.

Function ConfigSaveSettings(Silent)

Silent: Is TRUE if the script should not display any prompts or interact with the user

Return value: Return an error message on failure

This function is called when the script should save its settings. It should also check to make sure the settings are valid. For example:

```
' Save settings. Return error message on failure.
Function ConfigSaveSettings(Silent)
    ConfigSaveSettings = ""

    if SBProfile.GetEdit(1) = "" then
        ConfigSaveSettings = "The process name cannot be empty!"
        Exit Function
    end if

    if Int(SBProfile.GetEdit(2)) < 1 then
        ConfigSaveSettings = "The retry seconds cannot be less than 1!"
        Exit Function
    end if

    SBVariables.SetProperty "WFProcessName", SBProfile.GetEdit(1)
    SBVariables.SetProperty "WFRetrySecs", SBProfile.GetEdit(2)
End Function
```

Function ConfigWantSetupNode(IsGroup)

WillShow: Is TRUE if the profile is a group profile

Return value: True if a node in the setup window for the profile is required

Should a node in the profile setup window be created for this script to use? Note that the result should be consistent, e.g. it should not be based on what the current time is. This is because this function is called several times once the profile setup window is displayed for a profile, and each time it is called the result should be the same.

Sub ConfigFactoryDefaults()

This subroutine is called when the user has reverted to factory defaults. In this case the script should delete all the profile settings it manages. By doing this the default values will be used when the settings are read.

For example:

```
' Reset to factory defaults
Sub ConfigFactoryDefaults()
    SBVariables.DeleteProperty "WFProcessName"
    SBVariables.DeleteProperty "WFRetrySecs"
End Sub
```

Sub ConfigSetupDisplay()

This subroutine is called when the script should tell SyncBack what items the setup page should have on it. For example:

```
' Create setup window display
Sub ConfigSetupDisplay()
    SBProfile.AddEdit "The name of the process you want to have finished running",
128, FALSE, FALSE, 1
    SBProfile.AddEdit "The number of seconds to wait before re-checking if it is still
running", 10, TRUE, FALSE, 2
End Sub
```

Sub ConfigUpdateConditionals()

This subroutine is called when an item on the profile settings page has been changed. The script can then verify the new values, enable or disable other items based on the new values, etc.

Sub InitialiseVars(Passed)

Checking: Passed as True if the variables are being checked to see if they exist

This subroutine is called when the variables are initialised when a profile is being created or modified. Checking is passed as TRUE.

Note that if your script is also a run-time script then this sub-routine is shared. A configuration script does not have access to the SBRunning object.

In the example below the variable MyScriptVar is set to a dummy value if it's a call to check if the variable is valid.

```
'
' Called very early when a profile is run (Checking is False)
' Also called in profile config when asking what variables the script sets (Checking
is True)
'
Sub InitialiseVars(Checking)
  If Checking Then
    ' Profile is being saved
    SBVariables.SetVar "MyScriptVar", "?"
  Else
    ' Profile is being run
    ' Do nothing
  End If
End Sub
```

3.5.1.4 Location Scripts

These are functions that are defined in your **Location** script and are called by SyncBackPro. A location script is one that controls how files and folders are stored. For example, you could create a location script to backup to a 7zip archive file, or to a database, for example. The functions are called by SyncBackPro at the appropriate time. For a profile to use a location script you must configure the profile to use it. Location scripts have access to the SBLocation and SBSsystem objects.

Function LocAbilities()

Return value: The abilities of the script

This function is called so SyncBack knows what abilities (functions) the location script supports. Note that the value is cached, so the first value returned is used through-out the entire profile run.

The script can optionally support a number of features (the values can be OR'ed together, e.g. CAN_USEATTRIBUTES + CAN_NTFSATTRIBUTES:

For example:

```
Function LocAbilities()  
    ' Cannot use CAN_EXACTDATETIME because some drives may be FAT32  
    ' Cannot use CAN_NTFSATTRIBUTES because some drives may be FAT32  
    LocAbilities = 32 + 128 + 256 + 4096 + 131072  
End Function
```

Function LocConnect(MainThread)

MainThread: Passed as True if it is being called from a user interface

Return value: An error message if the script cannot be connect, otherwise an empty string

This function is called to tell the script to connect to the storage location.

There are two different ways in which this function is called: from the profile thread, or from the main thread (which is the user interface, e.g. the Differences window or the File Prompt window). When a profile is run then an instance of the script is created, and that instance is used while the profile is running. However, when there is user interaction (e.g. from the Differences window) then a new script instance is created. This means the state is different between the scripts, i.e. they have different global variables.

Return an error message if the script cannot connect to the storage location, e.g. the network is down.

Function LocConnected()

Return value: An error message if the script is not connected, otherwise an empty string

This function is called to tell ask the script if it is connected to its storage location. If it is connected (or doesn't need to connect to anything) then return an empty string, otherwise return an error message.

See also LocReconnect and LocConnect

Function LocCRC32(Filename, ByRef CRC32)

Filename: The complete path of the file to get the hash value of

CRC32: Set this to the CRC32 hash value of the file, in string format

Return value: An error message if the CRC32 cannot be retrieved, otherwise an empty string

This function is called when the CRC32 hash value of a file is required, e.g. for verification. The entire filename is passed, including the base path. If the CRC32 hash value cannot be retrieved then an error message should be returned.

Note that the CRC32 hash value should be returned in string format, e.g. E75A6A52

For example:

```
Function LocCRC32(FullPath, CRC32)
  CRC32 = ""
  LocCRC32 = "CRC32 not supported"
End Function
```

Function LocDeleteFile(Filename, ByRef DoesNotExist)

Filename: The complete path of the file to delete

DoesNotExist: Set to True if the file does not exist, otherwise set to False

Return value: An error message if the file exists and cannot be deleted, otherwise an empty string

This function is called when the script must delete a file. The entire filename is passed, including the base path. If the file cannot be deleted then an error message should be returned. If the file does not exist then set DoesNotExist to True, but do not return an error message.

For example:

```
Function LocDeleteFile(FullPath, DoesNotExist)
  LocDeleteFile = ""

  DoesNotExist = Not FSO.FileExists(FullPath)

  If Not DoesNotExist Then
    FSO.DeleteFile FullPath, True
  End If
End Function
```

Function LocDirExists(FullPath)

FullPath: The complete directory path including the base path

Return value: An empty string if the directory exists, otherwise an error message

This function is called when the script must check if a directory (folder) exists. The entire directory path will be passed, including the base path. If the directory exists then an empty string should be returned, otherwise return an error message.

For example:

```
Function LocDirExists(FullPath)
  If FullPath = "" Then
    LocDirExists = ""
  ElseIf FSO.FolderExists(FullPath) Then
    LocDirExists = ""
  End If
End Function
```

```
Else
  LocDirExists = "Folder does not exist"
End If
End Function
```

See also [LocFileExists](#)

Function [LocDisconnect\(\)](#)

Return value: An error message if the disconnect failed

This function is called when the location should disconnect from its storage, e.g. when the profile has finished or the user has aborted. If the script does not need to disconnect, or it disconnects without any problem, then it should return an empty string.

For example:

```
Function LocDisconnect()
  LocDisconnect = ""

  Set Drives = nothing
  Set FSO = nothing

  if Not (DebugLog is nothing) then
    DebugOut "---Disconnected: " & CStr(Now)
    DebugLog.WriteBlankLines(1)
    DebugLog.Close
    Set DebugLog = nothing
  End If
End Function
```

See also [LocConnect](#)

Function [LocFileExists\(FullFilename\)](#)

FullFilename: The complete filename including the base path

Return value: An empty string if the file exists, otherwise an error message

This function is called when the script must check if a file exists. The entire filename will be passed, including the base path. If the file exists then an empty string should be returned, otherwise return an error message.

For example:

```
Function LocFileExists(FullPath)
  If FullPath = "" then
    LocFileExists = "File does not exist"
  ElseIf FSO.FileExists(FullPath) Then
    LocFileExists = ""
  Else
    LocFileExists = "File does not exist"
  End If
End Function
```

```
End If
End Function
```

See also DirExists

Function LocFreeSpace()

Return value: The free space in bytes, else -1

This function is called when the location should return how much free space (in bytes) the storage location has. If it's not relevant or practical then return -1. Take note of the 32-bit integer limit in VBScript, so return the value as a string, e.g.

```
LocFreeSpace = CStr(CCur(3221225472))
```

Function LocGet(fromFName, toFName)

fromFName: The complete filename of the file to retrieve from the scripts storage location

toFName: Where script should store the file on the local filesystem

Return value: If the file cannot be retrieved and stored then return an error message

This function is called when SyncBack needs the location to retrieve one if its files and store it on the filesystem.

For example:

```
Function LocGet(fromFName, toFName)
  If Not FSO.FileExists(fromFName) Then
    LocGet = "File does not exist"
  Else
    ' Read-only?
    If FSO.FileExists(toFName) Then
      Set FileObj = FSO.GetFile(toFName)
      Attrs = FileObj.Attributes
      If Attrs And 1 Then
        FileObj.Attributes = Attrs - 1
      End If
      Set FileObj = nothing
    End If

    FSO.CopyFile fromFName, toFName, True
    LocGet = ""
  End If
End Function
```

Function LocGetAttributes(Filename, ByRef Attributes)

Filename: The complete path of the file or directory

Attributes: The filesystem attributes of the file or directory

Return value: An error message if the attributes cannot be retrieved, otherwise an empty string

This function is called when the filesystem attributes for a file or directory need to be retrieved. The entire path of the file or directory to get the attributes of is passed. Note that if it's a directory then it will have a trailing backslash. If the attributes cannot be retrieved then an error message should be returned.

For example:

```
Function LocGetAttributes(Filename, Attrs)
  If Filename = "" Then
    ' Its a special folder
    LocGetAttributes = ""
    Attrs = 1 + 2 + 4 + 16
    Exit Function
  End If

  If SBSSystem.IsFolder(Filename) Then
    '
    ' A folder
    '
    If not FSO.FolderExists(Filename) Then
      LocGetAttributes = "Folder does not exist"
      Exit Function
    End If

    Set FolderObj = FSO.GetFolder(Filename)
    Attrs = FolderObj.Attributes
    LocGetAttributes = ""
  Else
    '
    ' A file
    '
    If not FSO.FileExists(Filename) Then
      LocGetAttributes = "File does not exist"
      Exit Function
    End If

    Set FileObj = FSO.GetFile(Filename)
    Attrs = FileObj.Attributes
    LocGetAttributes = ""
  End If
End Function
```

Function LocMakeDir(FullPath)

FullPath: The complete path of the directory to create

Return value: An error message if the directory cannot be created, otherwise an empty string

This function is called when the location should create a directory. The entire path of the directory to create is passed. If the directory cannot be created then an error message should be returned.

Note that an empty string should be returned if the directory already exists. Do not return an error message.

For example:

```
Function LocMakeDir(FullPath)
    LocMakeDir = ""

    If FullPath = "" Then
        Exit Function
    ElseIf Not FSO.FolderExists(FullPath) Then
        FSO.CreateFolder(FullPath)
    End If
End Function
```

Function LocPut(fromFName, toFName, fromAttrs, fromDateTime, fromFileSize, DoSafeCopy, ByRef SafeFName)

fromFName: The complete filename of the file to retrieve from the local filesystem

toFName: Where script should store the file in its storage location

fromAttrs: The filesystem attributes of the file (ignore if < 0)

fromDateTime: The last modification date & time of the file (local timezone) (ignore if <=1.0)

fromFileSize: The size of the file (in bytes). Note this is a string to avoid the 32-bit integer limit in VBScript. (ignore if < 0)

DoSafeCopy: Passed as True if the file should be copied to a temporary file first and not to toFName

SafeFName: Set this to the filename used to store the file if DoSafeCopy was passed as True

Return value: If the file cannot be stored then return an error message

This function is called when SyncBack needs the location to store a file in its storage location. The file to store can be copied from the local filesystem.

If DoSafeCopy is True then the file should be copied to a temporary file first and not to the filename specified in toFName. The full path of the safe filename used should be returned in SafeFName.

For example:

```
Function LocPut(fromFName, toFName, fromAttrs, fromDateTime, fromFileSize,
DoSafeCopy, SafeFName)
    If Not FSO.FileExists(fromFName) Then
        LocPut = "File does not exist"
    Else
        ' Safe copy?
        If DoSafeCopy Then
            ' Note that we must return a SafeFName that we will understand when
            ' it is passed back to us (we will be asked to move the file)
            FulltoFName = toFName & ".$$$"
        End If
    End If
End Function
```

```
    SafeFName = toFName & ".$$$"
Else
    FulltoFName = toFName
    SafeFName = ""
End If

' Destination file read-only?
If FSO.FileExists(FulltoFName) Then
    Set FileObj = FSO.GetFile(FulltoFName)
    Attrs = FileObj.Attributes
    If Attrs And 1 Then
        FileObj.Attributes = Attrs - 1
    End If
    Set FileObj = nothing
End If

' SBSSystem.UpdateFileStatus("Copying " & fromFName & "...")
FSO.CopyFile fromFName, FulltoFName, True

If fromAttrs >= 0 Then
    Set FileObj = FSO.GetFile(FulltoFName)
    FileObj.Attributes = fromAttrs
    Set FileObj = nothing
End If

If fromDateTime > 1.0 Then
    SBSSystem.SetLastModDateTime FulltoFName, fromDateTime
End If

LocPut = ""
End If
End Function
```

Function LocReconnect()

Return value: An error message if the script cannot reconnect, otherwise an empty string

This function is called to tell the script to reconnect to the storage location. This is called when LocConnected indicates it is not connected, for example. Return an error message if the reconnect failed, e.g. it cannot reconnect because the network is down.

Function LocRemoveDir(FullPath)

FullPath: The complete path of the empty directory to delete

Return value: An error message if the directory cannot be deleted, otherwise an empty string

This function is called when the location should delete an empty directory. The entire path of the directory to delete is passed. If the directory cannot be deleted then an error message should be returned.

Note that an empty string should be returned if the directory does not exist. Do not return an

error message.

IMPORTANT: Do not delete a directory unless it is empty, i.e. it contains no files and no sub-directories.

For example:

```
Function LocRemoveDir(FullPath)
    LocRemoveDir = ""

    If FullPath = "" Then
        Exit Function
    ElseIf FSO.FolderExists(FullPath) Then
        Set Folder = FSO.GetFolder(Path)

        If Folder.Files.Count > 0 Then
            LocRemoveDir = "Directory contains files"
            Exit Function
        End If

        If Folder.SubFolders.Count > 0 Then
            LocRemoveDir = "Directory contains directories"
            Exit Function
        End If

        Set Folder = nothing
        FSO.DeleteFolder FullPath, True
    End If
End Function
```

Function LocRenameDir(OldFullPath, NewFullPath)

OldFullPath: The existing path

NewFullPath: The new path (rename it to this)

Return value: An error message if the directory cannot be renamed, otherwise an empty string

This function is called when the script must rename a directory. Entire paths are passed, including the base path. If the directory cannot be renamed then return an error message.

For example:

```
Function LocRenameDir(FromPath, ToPath)
    If FromPath = "" Then
        LocRenameDir = "Root directory cannot be moved"
        Exit Function
    ElseIf ToPath = "" Then
        LocRenameDir = "Cannot move to root directory"
        Exit Function
    End If

    If Not FSO.FolderExists(FromPath) Then
        LocRenameDir = "Source directory does not exist"
```

```
Exit Function
End If

If FSO.FolderExists(ToPath) Then
    LocRenameDir = "To destination directory already exists"
    Exit Function
End If

FSO.MoveFolder FromPath, ToPath
LocRenameDir = ""
End Function
```

Function LocRenameFile(OldFullFilename, NewFullFilename)

OldFullFilename: The existing filename

NewFullFilename: The new filename (rename it to this)

Return value: An error message if the file cannot be renamed, otherwise an empty string

This function is called when the script must rename a file. Entire filenames are passed, including the base path. If the file cannot be renamed then return an error message.

For example:

```
Function LocRenameFile(FromName, ToName)
    If Not FSO.FileExists(FromName) Then
        LocRenameFile = "Source file does not exist"
        Exit Function
    End If

    If FSO.FileExists(ToName) Then
        LocRenameFile = "To destination file already exists"
        Exit Function
    End If

    FSO.MoveFile FromName, ToName
    LocRenameFile = ""
End Function
```

Function LocScanList(FullPath)

FullPath: The complete directory path including the base path

Return value: An error message if the directory cannot be scanned, otherwise an empty string

This function is called when the script must tell SyncBack what files and sub-directories are in a directory. The entire path of the directory to scan is passed. For each file it must call SBLocation.AddFile, and for each folder it must call SBLocation.AddDir. If the directory cannot be scanned, e.g. access denied or it doesn't exist, then an error message should be returned.

For example:

```
Function LocScanList(FullPath)
  LocScanList = ""

  if FullPath = "\" or FullPath = "" then
  '
  ' Return the list of drives
  '
  For Each DiskDrive in Drives
    If DiskDrive.IsReady then
      If SBLocation.AddDir(DiskDrive.DriveLetter) = False then
        Exit Function
      End If
    End If
  Next
else
  '
  ' Return the list of folders and files
  '
  DriveLetter = Mid(FullPath, 2, 1)
  if DriveLetter = "" then
    Exit Function
  End If
  DrivePath = Mid(FullPath, 3, Len(FullPath) - 2)
  If DrivePath = "" then
    DrivePath = "\"
  End If
  Path = DriveLetter & ":" & DrivePath

  ' Return a list of sub-folders
  Set Folder = FSO.GetFolder(Path)
  Set FolderCol = Folder.SubFolders
  For Each SubFol in FolderCol
    If SBLocation.AddDir(SubFol.Name) = False then
      Exit Function
    End If
  Next

  ' Return a list of files
  Set FilesCol = Folder.Files
  For Each FileItem in FilesCol
    CFileSize = CStr(CCur(FileItem.Size))
    If SBLocation.AddFile(FileItem.Name, "", CFileSize, FileItem.Attributes,
FileItem.DateLastModified) = False then
      Exit Function
    End If
  Next
end if
End Function
```

Function LocScanPrepare()

Return value: An error message if the storage location cannot be prepared, otherwise an empty string

This function is called to tell the script to prepare the storage location for scanning. It is

called after LocConnect.

For example:

```
Function LocConnect(MainThread)
    LocConnect = ""

    IsMainThread = MainThread

    Set FSO = CreateObject("Scripting.FileSystemObject")
    Set Drives = FSO.Drives

    Set Folder = FSO.GetFolder("C:\")
    If Not IsMainThread Then
        Set DebugLog = Folder.CreateTextFile("debug.txt", True, True)
    Else
        Set DebugLog = Folder.CreateTextFile("debug_main.txt", True, True)
    End If

    DebugOut "---Connected: " & CStr(Now)
End Function
```

Function LocSetAttributes(Filename, Attributes)

Filename: The complete path of the file or directory

Attributes: The filesystem attributes to set the file or directory to

Return value: An error message if the attributes cannot be set, otherwise an empty string

This function is called when the filesystem attributes for a file or directory need to be set. The entire path of the file or directory to change the attributes of is passed. Note that if it's a directory then it will have a trailing backslash. If the attributes cannot be changed then an error message should be returned.

For example:

```
Function LocSetAttributes(Filename, Attrs)
    If Filename = "" Then
        ' Its a special folder
        LocSetAttributes = "Cannot set roots attributes"
        Exit Function
    End If

    If SBSSystem.IsFolder(Filename) Then
        '
        ' A folder
        '
        If not FSO.FolderExists(Filename) Then
            LocGetAttributes = "Folder does not exist"
            Exit Function
        End If

        Set FolderObj = FSO.GetFolder(Filename)
        FolderObj.Attributes = Attrs
        LocSetAttributes = ""
    End If
End Function
```

```
Else
'
' A file
'
If not FSO.FileExists(Filename) Then
    LocGetAttributes = "File does not exist"
    Exit Function
End If

Set FileObj = FSO.GetFile(Filename)
FileObj.Attributes = Attrs
LocSetAttributes = ""
End If
End Function
```

Function LocSetCreateDateTime(Filename, ModDateTime)

Filename: The complete path of the file or folder

CreateDateTime: The creation date & time (local timezone)

Return value: An error message if the date & time cannot be changed, otherwise an empty string

This function is called when the creation date & time of a file or folder must be changed. The entire path of the file to change the date & time of is passed. If the creation date & time cannot be changed then an error message should be returned.

For example:

```
Function LocSetCreateDateTime(FullPath, CreateDateTime)
    LocSetModDateTime = SBSsystem.SetCreateDateTime(FullPath, CreateDateTime)
End Function
```

Function LocSetModDateTime(Filename, ModDateTime)

Filename: The complete path of the file

ModDateTime: The last modification date & time (local timezone)

Return value: An error message if the date & time cannot be changed, otherwise an empty string

This function is called when the last modification date & time of a file must be changed. The entire path of the file to change the date & time of is passed. If the last modification date & time cannot be changed then an error message should be returned.

For example:

```
Function LocSetModDateTime(FullPath, ModDateTime)
    LocSetModDateTime = SBSsystem.SetLastModDateTime(FullPath, ModDateTime)
End Function
```

3.5.1.5 Runtime Scripts

These are functions that are defined in your **Runtime** script and are called by SyncBackPro. A runtime script can be used to change what happens when a profile is run. For example, you could stop a profile from being run if certain conditions aren't met, or perform actions when some files are copied, deleted, or renamed. The functions are called by SyncBackPro at the appropriate time. Runtime scripts are used by profiles and have access to the SBRunning, SBSsystem, SBVariables, and SBHistory objects.

Function RunBeforeCopyFile(ToLeft, Filename, ByRef ToDirCreated, ByRef FromFileLocked, ByRef ToFileLocked, ByRef DoneCopy)

ToLeft: True if the left/source file is to be copied from the right/destination

Filename: The file to be copied (not including the base path)

ToDirCreated: Set to True if a directory was created

FromFileLocked: Set to True if the file to be copied is locked

ToFileLocked: Set to True if the file to be replaced is locked

DoneCopy: Set to True if the file was copied

Return value: An error message if the copy failed

This function is called before a file is to be copied, and before any version is made. On failure the function should return an error message (do not return an error message if the script is not copying files). If a directory was created in order to copy the file then ToDirCreated should be returned as True. If the from file is locked then FromFileLocked should be returned as True (and an error message should be returned). If the to file is locked then ToFileLocked should be returned as True (and an error message should be returned). If the file was copied then DoneCopy should be returned as True.

Note that once a script has copied a file then no other scripts will be called to copy the file. This means the order of run-time scripts used in a profile is important.

This function is not called if the run is a simulation.

See also RunAfterCopyFile

Function RunBeforeDeleteFile(Left, Filename, ByRef FileLocked, ByRef DoneDelete)

Left: True if the left/source file is to be deleted

Filename: The file to be deleted (not including the base path)

FileLocked: Set to True if the file to be deleted is locked

DoneDelete: Set to True if the file was deleted

Return value: An error message if the delete failed

This function is called before a file is to be deleted (and before a version has been made). On failure the function should return an error message (do not return an error message if the script is not deleting files). If the file is locked then FileLocked should be returned as True (and an error message should be returned). If the file was deleted then DoneDelete should be returned as True.

Note that once a script has deleted a file then no other scripts will be called to delete the file. This means the order of run-time scripts used in a profile is important. Obviously a file can only be deleted once.

This function is not called if the run is a simulation.

See also RunAfterDeleteFile

Function RunBeforeEmailLog(ProfileResult, Failed, Diffs)

ProfileResult: The result of the profile run

Failed: True if the profile run had failures

Diffs: True if the profile found differences between the left/source and destination/right and the profile is set to email if there are differences

Return value: Return False to not send the email, else return True

This function is called before the log is to be emailed. This gives the script an opportunity to not send the email.

Note this subroutine is not called if the profile is not configured to send the log via email.

IMPORTANT: This function is called in the context of the main user interface thread.

Function RunBeforeRenameFile(Left, FromFilename, ToFilename, ByRef ToDirCreated, ByRef FileLocked, ByRef DoneRename)

Left: True if the left/source file is to be renamed

FromFilename: The old name of the file (not including the base path)

ToFilename: The new name of the file (not including the base path)

ToDirCreated: Set to True if the directory for ToFilename was created

FileLocked: Set to True if one of the files is locked so cannot be renamed

DoneRename: Set to True if the file was renamed

Return value: An error message if the rename failed

This function is called before a file is to be renamed. On failure the function should return an error message (do not return an error message if the script is not renaming files). If a directory was created in order to rename the file then ToDirCreated should be returned as

TRUE. If one of the files is locked then FileLocked should be returned as TRUE (and an error message should be returned). If the file was renamed then DoneRename should be returned as True.

Note that once a script has renamed a file then no other scripts will be called to rename the file. This means the order of run-time scripts used in a profile is important. Obviously a file can only be renamed once.

This function is not called if the run is a simulation.

See also RunAfterRenameFile

Function RunBeforeSetAttrs(Left, Filename, useAttrs, ByRef DoneSet)

Left: True if the left/source files attributes are to be changed

Filename: The name of the file (not including the base path)

useDateTime: The attributes to use

DoneSet: Set to True if the files attributes were changed

Return value: An error message on failure

This function is called when a files filesystem attributes are going to be changed. On error return an error message (do not return an error message if the script is not changing the files attributes). If the files attributes were changed then DoneSet should be returned as True. It is called only when the action is CACTION_USE_SRC_DETAILS or CACTION_USE_DEST_DETAILS, and is not called when a file is copied and its attributes are to be changed, for example.

Note that once a script has set a files attributes then no other scripts will be called to set the files attributes. This means the order of run-time scripts used in a profile is important.

This function is not called if the run is a simulation.

See also RunAfterSetAttrs

Function RunBeforeSetModDateTime(Left, Filename, useDateTime, ByRef DoneSet)

Left: True if the left/source files date & time is to be changed

Filename: The name of the file (not including the base path)

useDateTime: The date & time to change it to (local timezone)

DoneSet: Set to True if the files date & time was changed

Return value: An error message on failure

This function is called when a files last modification date & time is going to be set. On error return an error message (do not return an error message if the script is not setting the files date & time). If the files date & time was changed then DoneSet should be returned as True.

It is called only when the action is CACTION_USE_SRC_DETAILS or CACTION_USE_DEST_DETAILS, and is not called when a file is copied and its date & time is changed, for example.

Note that once a script has set a files date & time then no other scripts will be called to set the files date & time. This means the order of run-time scripts used in a profile is important.

This function is not called if the run is a simulation.

See also RunAfterSetModDateTime

Function RunDeleteAll(Left, WillDeleteAll)

Left: True if the check is for the Left/Source

WillDeleteAll: True if the profile is currently set to delete all

Return value: Return False to not delete all, else True to delete all

This function is called before any attempt is made to delete all the files and folders in either the source/left or destination/right.

Function RunDiffColumnsCount()

Return value: The number of custom columns, or zero if none are required

This function is called from the Differences window to ask the script how many custom columns it wants created in the Differences window. Note that this value is cached so the function is only called once (when the Differences window appears).

```
Function RunDiffColumnsCount()  
    RunDiffColumnsCount=2  
End Function
```

Function RunDiffColumnSort(Col, IsFile1, IsFile2, Filename1, Filename2)

Col: The column the display is being sorted on

IsFile1: True if Filename1 is a file, otherwise it's a folder

IsFile2: True if Filename2 is a file, otherwise it's a folder

Filename1: The name of a file/folder

Filename2: The name of a file/folder

Return value: An integer (<0 if file 1 should go before file 2, 0=same filename, > 0 if file 1 should go after file 2)

This function is called from the Differences window when a custom column is being sorted. The first custom column is column zero. The script is only called for its custom columns and not for columns created by other scripts.

Note that sorting can be extremely slow if there are hundreds of thousands of items to sort.

See also `RunDiffColumnsCount`

Function `RunDisabledCheck(ByRef NoLog)`

NoLog: Set this to TRUE if a log should not be created

Return value: A reason why the script should not continue running

This function is called very early on in the profile run and gives the script a chance to stop the profile from running. If the script does not want the profile to run it should return the reason, otherwise it should return an empty string if the profile should continue.

To stop a log file from being produced then set `NoLog` to True. For example, you may expect certain conditions, e.g. the network being unavailable, and so do not want a log file to be produced.

```
Function RunDisabledCheck(NoLog)
    NoLog = FALSE

    If SBRunning.Restore Then
        RunDisabledCheck = "You cannot restore your files"
    Else
        RunDisabledCheck = ""
    End If
End Function
```

Function `RunEmailLogAttachFilename(Cnt)`

Cnt: On the first call this is zero, and then incremented on each call

Return value: The filename of the file to attach. The file must exist and be readable.

This function is called when the log file is about to be emailed. It gives the script an opportunity to add custom file attachments to the email.

Note that `RunEmailLogAttachToAdd` is called first, and then `RunEmailLogAttachFilename()` is called the appropriate number of times.

IMPORTANT: This function is called in the context of the main user interface thread.

See `RunEmailLogAttachToAdd` for an example.

Function `RunEmailLogAttachToAdd`

Return value: The number of files to attach

This function is called when the log file is about to be emailed. It gives the script an opportunity to add custom file attachments to the email.

Note that RunEmailLogAttachToAdd is called first, and then RunEmailLogAttachFilename is called the appropriate number of times.

IMPORTANT: This function is called in the context of the main user interface thread.

```
Function RunEmailLogAttachToAdd
    RunEmailLogAttachToAdd = 2
End Function

Function RunEmailLogAttachFilename(Cnt)
    If Cnt = 0 Then
        RunEmailLogAttachFilename = "c:\path\filename1.txt"
    Else
        RunEmailLogAttachFilename = "c:\path\filename2.txt"
    End If
End Function
```

Function RunLogLocationInfoCaption(Left, Cnt)

Left: True is this call is for information on the left/source

Cnt: On the first call this is zero, and then incremented on each call

Return value: The caption string to use

This function is called when the log file is being created. It gives the script an opportunity to add custom information to the log file about the left/source and/or right/destination locations.

Note that RunLogLocationInfoToAdd is called first, and then RunLogLocationInfoCaption() is called the appropriate number of times.

See RunLogLocationInfoToAdd for an example.

Function RunLogLocationInfoInfo(Left, Cnt)

Left: True is this call is for information on the left/source

Cnt: On the first call this is zero, and then incremented on each call

Return value: The information string to use

This function is called when the log file is being created. It gives the script an opportunity to add custom information to the log file about the left/source and/or right/destination locations.

Note that RunLogLocationInfoToAdd is called first, and then RunLogLocationInfoInfo is called the appropriate number of times.

See RunLogLocationInfoToAdd for an example.

Function RunLogLocationInfoToAdd(Left)

Left: True if this call is for information on the left/source

Return value: The number of caption/info calls to make

This function is called when the log file is being created. It gives the script an opportunity to add custom information to the log file about the left/source and/or right/destination locations.

Note that RunLogLocationInfoToAdd is called first, and then RunLogLocationInfoInfo is called the appropriate number of times.

```
Function RunLogLocationInfoToAdd(Left)
  If Left = True Then
    RunLogLocationInfoToAdd = 2
  Else
    RunLogLocationInfoToAdd = 0
  End If
End Function

Function RunLogLocationInfoCaption(Left, Cnt)
  If Left = True Then
    If Cnt = 0 Then
      RunLogLocationInfoCaption = "Caption 1"
    Else
      RunLogLocationInfoCaption = "Caption 2"
    End If
  Else
    RunLogLocationInfoCaption = ""
  End If
End Function

Function RunLogLocationInfoInfo(Left, Cnt)
  If Left = True Then
    If Cnt = 0 Then
      RunLogLocationInfoInfo = "Info 1"
    Else
      RunLogLocationInfoInfo = "Info 2"
    End If
  Else
    RunLogLocationInfoInfo = ""
  End If
End Function
```

Function RunPreCopyCheck()

Return value: A reason why the script should not continue running

This function is called after the files and folders have been compared, and after the Differences window is displayed (if it is to be displayed). It is called before any files are copied, moved, deleted, or renamed, and gives the script a chance to abort the profile, e.g. the script may decide there is not enough free space to copy everything and so abort. If the script does not want the profile to run it should return the reason, otherwise it should return

an empty string if the profile should continue.

```
Function RunPreCopyCheck()  
  If NotEnoughDiskSpace Then  
    RunPreCopyCheck = "There is not enough disk space"  
  Else  
    RunPreCopyCheck = ""  
  End If  
End Function
```

Function RunRunAfterBefore(Filename)

Filename: The full filename and command line parameters of program to be called in Run After

Return value: The filename and command line parameters of program to be called in Run After

This function is called before the Run After program is called. It is passed the full and expanded filename (and any command line arguments) that are going to be used, which could be an empty string. This function can decide not to run the program (by returning an empty string) or it could change the string to have a different program be called. If it wants the same program to be called it should return what was passed in Filename.

See also RunRunAfterAfter and RunRunBeforeBefore

Function RunRunBeforeBefore(Filename)

Filename: The full filename and command line parameters of program to be called in Run Before

Return value: The filename and command line parameters of program to be called in Run Before

This function is called before the Run Before program is called. It is passed the full and expanded filename (and any command line arguments) that are going to be used, which could be an empty string. This function can decide not to run the program (by returning an empty string) or it could change the string to have a different program be called. If it wants the same program to be called it should return what was passed in Filename.

See also RunRunBeforeAfter and RunRunAfterBefore

Function RunShowDiffWindow(WillShow)

WillShow: Is the Differences window going to be displayed?

Return value: True if the Differences window should be displayed

Should the Differences window be displayed? Note that this function is not called if the profile run is unattended. In such cases the Differences window is never displayed.

Sub InitialiseVars(Checking)

Checking: Passed as True if the variables are being checked to see if they exist

This subroutine is called when the variables are initialised. Checking is passed as FALSE. Note that you need to be careful as many things haven't been initialised or created when this is called, e.g. no locations. It is called very early during the profile run (it is called just after RunDisabledCheck is called).

If you are setting variables then it is recommended you also make your script a Configuration script. This is so the users can see which variables the script sets, and also they can use the variables, e.g. in the source or destination paths.

In the example below the variable MyScriptVar is set to a dummy value if it's a call to check if the variable is valid.

```
'  
' Called very early when a profile is run (Checking is False)  
' Also called in profile config when asking what variables the script sets (Checking  
is True)  
'  
Sub InitialiseVars(Checking)  
  If Checking Then  
    ' Profile is being saved  
    SBVariables.SetVar "MyScriptVar", "?"  
  Else  
    ' Profile is being run  
    ' Do nothing  
  End If  
End Sub
```

Sub RunAfterConfig()

This function is called very early on in the profile run and gives the script a chance to do any early initialisation work. It is called before any attempt is made to connect to the left/source or destination/right. It is called after RunBeforeConfig and at this point the left/source folder is known, for example.

Sub RunAfterCopyFile(ToLeft, Filename, Failed)

ToLeft: True if the file was copied from the right/destination

Filename: The name of the file (not including the base path)

Failed: True if the copy failed

This function is called after a file has been copied. If Failed is True then either the copy failed or the user has aborted. Note that Failed being False does not necessarily mean the

file was actually copied, i.e. it does not guarantee a file was created because the location may not actually copy the file, e.g. it may actually simply send it to a printer. What a script location does when it copies a file is beyond SyncBack's control.

This function is not called if the run is a simulation.

See also RunBeforeCopyFile

Sub RunAfterDeleteFile(Left, Filename, Failed)

Left: True if the left/source file was deleted

Filename: The name of the file (not including the base path)

Failed: True if the delete failed

This function is called after a file has been deleted. If Failed is True then either the delete failed or the user has aborted. Note that Failed being False does not necessarily mean the file was actually deleted, e.g. it may not have existed.

This function is not called if the run is a simulation.

See also RunBeforeDeleteFile

Sub RunAfterEmailLog(ErrMsg)

ErrMsg: An error message (if the log could not be sent via email)

This subroutine is called after the log is emailed. If there was a problem sending the email then ErrMsg contains an error message.

Note this subroutine is not called if the profile is not configured to send the log via email.

IMPORTANT: This function is called in the context of the main user interface thread.

Sub RunAfterFileCompare(Filename, Diff, WhyIgnored, ByRef Action)

Filename: The name of the file (not including the base path)

Diff: The differences between the files

WhyIgnored: The reason why the file is being ignored (if it is)

Action: The action that has been decided upon based on the profiles settings

This function is called after a file is compared. The script can change the Action to something else, or leave it as-is.

See also RunBeforeFileCompare

Sub RunAfterFolderCompare(Filename, ByRef Action)

Filename: The name of the folder (not including the base path)

Action: The action that has been decided upon based on the profiles settings

This function is called after a folder (directory) is compared. The script can change the Action to something else, or leave it as-is.

See also `RunBeforeFolderCompare`

Sub `RunAfterRenameFile(Left, FromFilename, ToFilename, Failed)`

Left: True if the left/source file was renamed

FromFilename: The old name of the file (not including the base path)

ToFilename: The new name of the file (not including the base path)

Failed: True if the rename failed

This function is called after a file has been renamed. If Failed is True then either the rename failed or the user has aborted. Note that Failed being False does not necessarily mean the file was actually renamed, i.e. it does not guarantee a file was renamed because the location may not actually rename the file. What a script location does when it renames a file is beyond SyncBack's control.

This function is not called if the run is a simulation.

See also `RunBeforeRenameFile`

Sub `RunAfterSetAttrs(Left, Filename, useAttrs, Failed)`

Left: True if the left/source files attributes were changed

Filename: The name of the file (not including the base path)

useAttrs: The attributes that were used

Failed: True if the change failed

This function is called after a files filesystem attributes have been changed. It is called only when the action was `CACTION_USE_SRC_DETAILS` or `CACTION_USE_DEST_DETAILS`, and is not called when a file is copied and its attributes are changed, for example. If Failed is True then either the change failed or the user has aborted. Note that Failed being False does not necessarily mean the files attributes were actually changed.

This function is not called if the run is a simulation.

See also `RunBeforeSetAttrs`

Sub `RunAfterSetModDateTime(Left, Filename, useDateTime, Failed)`

Left: True if the left/source files date & time was changed

Filename: The name of the file (not including the base path)

useDateTime: The date & time it was changed to (local timezone)

Failed: True if the change failed

This function is called after a file's last modification date & time has been changed. It is called only when the action was `CACTION_USE_SRC_DETAILS` or `CACTION_USE_DEST_DETAILS`, and is not called when a file is copied and its date & time is changed, for example. If **Failed** is True then either the change failed or the user has aborted. Note that **Failed** being False does not necessarily mean the date & time of the file was actually changed.

This function is not called if the run is a simulation.

See also `RunBeforeSetModDateTime`

Sub `RunBeforeConfig()`

This function is called very early on in the profile run and gives the script a chance to do any early initialisation work. It is called before any attempt is made to connect to the left/source or destination/right. It is called after `RunDisabledCheck` and before the profile is prepared. Note that very little is initialised at this point, e.g. there is no left/source folder.

See also `RunAfterConfig`

Sub `RunBeforeFileCompare(Filename, ByRef Skip)`

Filename: The filename of the file (not including the base path)

Set: Skip to True to ignore/skip the file

This subroutine is called before a file is compared for differences. Note that the hash values will be probably empty except for: Fast Backup profiles which may have the destination hash value, and when using compression as the hash value is retrieved from the Zip file.

Sub `RunBeforeFolderCompare(Filename, ByRef Skip)`

Filename: The name of the folder (not including the base path)

Skip: Set to True to skip this folder

This function is called before a folder (directory) is compared. It gives the script an opportunity to skip a folder (it's action will be set to `CACTION_SKIP`)

See also `RunAfterFolderCompare`

Sub `RunBeforeScanning()`

This function is called just before scanning of the files and folders is about to start.

Sub RunDiffColumnClicked(Col, IsFile, Filename)

Col: The column number

IsFile: True if Filename refers to a file, else it's a folder

Filename: The name of the file/folder (not including the base path)

This subroutine is called from the Differences window when a custom column is clicked. The first custom column is column zero (0). The script is only called for its custom columns and not for columns created by other scripts.

See also RunDiffKeyPress

```
Sub RunDiffColumnClicked(Col, IsFile, Filename)
    SBSsystem.Say "Custom Column " & Col
End Sub
```

Sub RunDiffColumnHint(Col, IsFile, Filename)

Col: The column number

IsFile: True if Filename refers to a file, else it's a folder

Filename: The name of the file/folder (not including the base path)

Return value: The hint string to display

This subroutine is called from the Differences window when a custom column hint is required. The first custom column is column zero (0). The script is only called for its custom columns and not for columns created by other scripts.

```
Function RunDiffColumnHint(Col, IsFile, Filename)
    RunDiffColumnHint = Filename
End Function
```

Sub RunDiffColumnText(Col, IsFile, Filename)

Col: The column number

IsFile: True if Filename refers to a file, else it's a folder

Filename: The name of the file/folder (not including the base path)

This subroutine is called from the Differences window when text for a custom column is required. The first custom column is column zero (0). The script is only called for its custom columns and not for columns created by other scripts.

```
Function RunDiffColumnText(Col, IsFile, Filename)
    If Col = 0 Then
        RunDiffColumnText = "Custom Column 1: " & Filename
    Else
        RunDiffColumnText = "Custom Column 2: " & Filename
    End If
End Function
```

See also RunDiffColumnsCount

Sub RunDiffColumnTitle(Col, ByRef Width)

Col: The column number

Width: Set it to the width the column should be (in pixels)

This subroutine is called from the Differences window when the title for a custom column is required. The first custom column is column zero (0). The script is only called for its custom columns and not for columns created by other scripts.

```
Function RunDiffColumnTitle(Col, Width)
  If Col = 0 Then
    RunDiffColumnTitle = "Custom Column 1"
    Width = 100
  Else
    RunDiffColumnTitle = "Custom Column 2"
    Width = 150
  End If
End Function
```

Sub RunDiffFocusChanged(IsFile, Filename)

IsFile: True if Filename refers to a file, else it's a folder

Filename: The name of the file/folder (not including the base path)

This subroutine is called from the Differences window when the focused node changes.

```
Sub RunDiffFocusChanged(IsFile, Filename)
  SBSsystem.Say "Focus changed"
End Sub
```

Sub RunDiffKeyPress(Key, Shift, IsFile, Filename)

Key: The key that as pressed

Shift: The shift state

IsFile: True if Filename refers to a file, else it's a folder

Filename: The name of the file/folder (not including the base path)

This subroutine is called from the Differences window when a key is pressed. It is called for each selected row. It is not called if the Delete key is pressed (as that is handled by SyncBack itself).

The Key value refers to the virtual key codes.

The Shift state can be a selection of the following values:

0 = no shift state

1 = Shift key is pressed (ssShift)

2 = Alt key is pressed (ssAlt)
4 = Ctrl key is pressed (ssCtrl)
8 = Left mouse button is pressed (ssLeft)
16 = Right mouse button is pressed (ssRight)
32 = Middle mouse button is pressed (ssMiddle)
64 = Mouse double-clicked (ssDouble)

See also RunDiffColumnClicked

```
' This example says the filename if S is pressed
Sub RunDiffKeyPress(Key, ShiftState, IsFile, Filename)
  If Key = 83 Then
    SBSsystem.Say Filename
  End If
End Sub
```

Sub RunDoFullBackup(LeftDir, RightDir, ByRef FullBackup)

LeftDir: The base directory of the left/source

RightDir: The base directory of the right/destination

FullBackup: True if this is a full backup run. Change as required.

This function is called when a decision must be made on whether the profile run should be a full backup or an incremental/differential backup.

Note that once a script has decided on whether it is a full backup or not then no other scripts will be called to decide. This means the order of run-time scripts used in a profile is important.

See also FullBackup

Sub RunFileCompareDiff(Filename, Diff, WhyIgnored, ByRef Skip)

Filename: The filename of the file (not including the base path)

Diff: The differences between the files

WhyIgnored: The reason why the file is being ignored (if it is)

Skip: Set to True to ignore/skip the file

This subroutine is called after a file has been compared for differences and the left/source and right/destination files are different.

Sub RunFileCompareSame(Filename, ByRef Diff)

Filename: The filename of the file (not including the base path)

Diff: Set to the difference between the files, or CDIFF_IDENTICAL

This subroutine is called after a file has been compared for differences and the left/source and right/destination files are found to be identical.

Sub RunProfileResult(ProfileResult, ErrMsg)

ProfileResult: The result of the profile run

ErrMsg: If a fatal error occurred, this is the error message

This function is called when the result of the profile run is known and has been saved. It is usually done before the profile finishes. For example:

```
Sub RunProfileResult(ProfileResult, ErrMsg)
    SBRunning.DebugOut CStr(ProfileResult), ErrMsg, 1
End Sub
```

Sub RunRunAfterAfter(Filename, ReturnValue, ReturnErrMsg, TimedOut)

Filename: The full filename and command line parameters of program called in Run After

ReturnValue: The numeric return value of the program

ReturnErrMsg: If the program could not be run then this is the error message

TimedOut: If the program timed out then this is TRUE

This function is called after the Run After program has been called. It is passed the full and expanded filename (and any command line arguments) that were used. If there was a problem running the program then ReturnErrMsg will contain an error message. RetVal will contain the numeric return value from the program, but note that this value should be ignored if the program failed to run, if TimedOut is TRUE, or if the profile was not configured to wait for the program to finish. TimedOut is TRUE if the program took too long to run.

See also RunRunBeforeAfter and RunRunAfterBefore

Sub RunRunBeforeAfter(Filename, ReturnValue, ReturnErrMsg, TimedOut)

Filename: The full filename and command line parameters of program called in Run Before

ReturnValue: The numeric return value of the program

ReturnErrMsg: If the program could not be run then this is the error message

TimedOut: If the program timed out then this is TRUE

This function is called after the Run Before program has been called. It is passed the full and expanded filename (and any command line arguments) that were used. If there was a problem running the program then ReturnErrMsg will contain an error message. RetVal will contain the numeric return value from the program, but note that this value should be ignored if the program failed to run, if TimedOut is TRUE, or if the profile was not configured to wait for the program to finish. TimedOut is TRUE if the program took too long to run.

See also RunRunBeforeBefore and RunRunAfterAfter

3.5.1.6 SBLocation

These are functions that can be accessed from scripts via the **SBLocation** object. For example:

```
SBLocation.BaseDir
```

The **SBLocation** object is only accessible from Location scripts.

Function AddDir(Name)

Name: The name of the directory, without the path

Return value: Return FALSE if the script wants to abort the scan

This function should be called by the Script inside the LocScanList whenever a new directory (folder) is found, i.e. the script must call this function for every directory in the directory being scanned.

It is recommended that you use the newer AddDirEx function instead of this function.

Name: The name should not include the path. It is just the name of the directory. It must be a unique name for that folder, i.e. two sub-directories in the same directory cannot have the same name. Note that you do not need to pass the special folders . or ..

Function AddDirEx(Name, Attrs)

Name: The name of the directory, without the path

Attrs: The directories attributes

Return value: Return FALSE if the script wants to abort the scan

This function should be called by the Script inside the LocScanList whenever a new directory (folder) is found, i.e. the script must call this function for every directory in the directory being scanned.

Name: The name should not include the path. It is just the name of the directory. It must be a unique name for that folder, i.e. two sub-directories in the same directory cannot have the same name. Note that you do not need to pass the special folders . or ..

Function AddFile(Name, CRC32, FileSize, Attrs, ModDateTime)

Name: The name of the file, without the path

CRC32: The CRC32 hash value of the file, or empty string if unknown

FileSize: The size of the file, in bytes, or -1 if unknown

Attrs: The filesystem attributes of the file, or -1 if unknown

ModDateTime: The last modification date & time of the file, or 1.0 if unknown

Return value: Return FALSE if the script wants to abort the scan

This function should be called by the Script inside the LocScanList function whenever a new file is found, i.e. the script must call this function for every file in the folder being scanned.

Name: The name should not include the path. It is just the name of the file. It must be a unique name for that folder, i.e. two files in the same folder cannot have the same name.

CRC32: Do not waste time calculating the CRC32 hash value. Only provide it if it is already known, e.g. it's a Zip file so you can get the hash value quickly, otherwise pass an empty string.

FileSize: The size of the file, in bytes. If the size is unknown then pass -1. If IsIgnoringSize is returning True, and retrieving the file size would incur more processing time, then you can pass -1. Note that this parameter is a string to avoid the 32-bit limit on integers used by VBScript. If you are using VBScript, then convert the number to a currency and then to a string, e.g. CFileSize = CStr(CCur(FileItem.Size))

Attrs: The filesystem attributes of the file. If the attributes are unknown then pass -1. Note that these must be standard Windows filesystem attributes.

ModDateTime: The last modification date & time of the file (in the local timezone). If the last modification date & time is unknown then pass 1.0. Also, if IsIgnoringDateTime is returning True, and retrieving the date & time would incur more processing time, then you can pass 1.0

Property Abort

Returns TRUE if the profile is being aborted. Set it to TRUE to abort the profile. Note that the abort may not be immediate, and an abort cannot be aborted (i.e. once True you cannot change it to False).

Property BaseDir

Returns the base directory of the location.

This is a read-only property.

Property IsIgnoringDateTime

Returns True if this location can ignore file last modification date & times. Note that this should only be used in the LocScanList script function otherwise the return value is always False. This can be used to optimize the location, e.g. if it takes time to get a files last

modification date & time, and `IsIgnoringDateTime` is returning `True`, then the script should not waste time trying to retrieve it (in the `LocScanList` script function).

This is a read-only property.

Property `IsIgnoringSize`

Returns `True` if this location can ignore file sizes. Note that this should only be used in the `LocScanList` script function otherwise the return value is always `False`. This can be used to optimize the location, e.g. if it takes time to get a file size, and `IsIgnoringSize` is returning `True`, then the script should not waste time trying to retrieve it (in the `LocScanList` script function).

This is a read-only property.

Property `IsLeft`

Returns `True` if this location is the left/source location, otherwise it is the right/destination location.

This is a read-only property.

Property `VersionsType`

Returns an integer value that states what is being done with versions folders:

0 = They are being treated as any other directory (usually because neither location is using versioning)

1 = Versions directories are being skipped and assumed not to exist (usually because this location doesn't use versioning, but the other one does)

2 = File versions are being read from the versions directories, i.e. the location uses versioning

3 = File versions are being read from a sub-folder of the base folder

Note that this information is for reference only as SyncBack will manage all aspects of versioning for the script.

This is a read-only property.

Property `VersionSubDir`

Returns the name of the versions sub-folder, e.g. `SBV`. Note that this information is for reference only as SyncBack will manage all aspects of versioning for the script.

This is a read-only property.

3.5.1.7 SBProfile

These are functions that can be accessed from scripts via the **SBProfile** object. For example:

```
SBProfile.Name
```

The **SBProfile** object is only accessible from Profile Configuration scripts.

Function GetCheckbox(Tag)

Tag: The tag ID of the checkbox control

Return value: TRUE if the checkbox is ticked

This function returns the state of a checkbox control (created with AddCheckbox). To change the state of a checkbox you can use SetCheckbox.

Function GetComboBoxIndex(Tag)

Tag: The tag ID of the combobox control

Return value: The index of the selected item (zero is the first item, -1 if no item is selected)

This function returns the index of the currently selected item in a combobox control (created with AddComboBox). Note that if the control can be edited then the index can be -1, e.g. the user has entered their own text instead of selecting an item. In that case you can get the text entered by using GetComboBoxText.

To change the state of a checkbox you can use SetComboBoxItem or SetComboBoxText.

Function GetComboBoxText(Tag)

Tag: The tag ID of the combobox control

Return value: The text in the combobox control

This function returns the text entered into a combobox control (created with AddComboBox).

To change the state of a checkbox you can use SetComboBoxItem or SetComboBoxText.

Function GetEdit(Tag)

Tag: The tag ID of the edit control

Return value: The text in the edit control

This function returns the text in an edit control (created with AddEdit). To change the text in an edit control you can use SetEdit.

Function GetRadioGroup(Tag)

Tag: The tag ID of the radiogroup control

Return value: The index of the selected item (zero is the first item, -1 if no item is selected)

This function returns the index of the currently selected item in a radiogroup control (created with AddRadioGroup).

To change the state of a checkbox you can use SetRadioGroup.

Function SetCheckbox(Value, Tag)

Value: Pass TRUE to tick the checkbox

Tag: The tag ID of the edit control

This sub routine sets the state of a checkbox control (created with AddCheckbox). To get the state of a checkbox control you can use GetCheckbox.

Function SetComboBoxItem(ItemIndex, Tag)

ItemIndex: The item index to use (0 is the first item, -1 means nothing selected)

Tag: The tag ID of the combobox control

This sub routine sets the selected item index of a combobox control (created with AddComboBox).

To get the current index of a checkbox control you can use GetComboBoxIndex.

Function SetComboBoxText(ItemText, Tag)

ItemText: The text to set the combobox control to use

Tag: The tag ID of the combobox control

This sub routine sets the text of a combobox control (created with AddComboBox).

To get the text of a checkbox control you can use GetComboBoxText.

Function SetEdit(Value, Tag)

Value: The text to put into the edit control

Tag: The tag ID of the edit control

This sub routine sets the text in an edit control (created with AddEdit). To get the text in an edit control you can use GetEdit.

Function SetLabel(Caption, Tag)

Value: The text set the label control to

Tag: The tag ID of the label control

This sub routine sets the text of a label control (created with AddLabel).

Function SetRadioGroup(ItemIndex, Tag)

ItemIndex: The item index to use (0 is the first item, -1 means nothing selected)

Tag: The tag ID of the radiogroup control

This sub routine sets the selected item index of a radiogroup control (created with AddRadioGroup).

To get the current index of a radiogroup control you can use GetRadioGroup.

Property Group

Returns the TRUE if this is a group profile.

This property is read-only.

Property Name

Returns the name of the profile.

This property is read-only.

Property UsesScript

Returns the TRUE if this profile is using this script. For example, you may have a script that is both a configuration and run-time script. Although the script may be enabled as a

configuration script, the profile that is currently being edited may not be using the profile as a run-time script.

This property is read-only.

Sub AddCheckbox(Caption, Tag)

Caption: Pass the caption text to use for the control

Tag: Pass the unique tag ID for this control

Adds a checkbox control to the profile setup window.

This function should only be called from ConfigSetupDisplay.

Sub AddComboBox(ListOnly, Tag)

ListOnly: If the combobox text cannot be edited then pass TRUE

Tag: Pass the unique tag ID for this control

Adds a combobox control to the profile setup window. To add items to the combobox use AddComboBoxItem.

This function should only be called from ConfigSetupDisplay.

Sub AddComboBoxItem(Value, Tag)

Value: The text to add to the combobox

Tag: The tag ID of the combobox to add the item to

Adds an item to a combobox control.

This function should only be called from ConfigSetupDisplay.

Sub AddEdit(Caption, MaxLen, NumbersOnly, Password, Tag)

Caption: Pass the caption text to use for the edit control

MaxLen: The maximum length of text that can be entered into the edit control

NumbersOnly: Pass as TRUE if the edit control is for numbers only

Password: Pass as TRUE if the edit control is for passwords

Tag: Pass the unique tag ID for this control

Adds an edit control to the profile setup window.

This function should only be called from ConfigSetupDisplay.

Sub AddLabel(Caption, Tag)

Caption: Pass the caption text to use for the label

Tag: Pass the unique tag ID for this label

Adds a label to the profile setup window.

This function should only be called from ConfigSetupDisplay.

Sub AddRadioGroup(Caption, Tag)

Caption: The caption to use for the control

Tag: Pass the unique tag ID for this control

Adds a radio group control to the profile setup window. To add items to the combobox use AddRadioGroupItem.

This function should only be called from ConfigSetupDisplay.

Sub AddRadioGroupItem(Caption, Tag)

Caption: The caption text to use for the new radio item

Tag: The tag ID of the radio group to add the item to

Adds an item to a radio group control.

This function should only be called from ConfigSetupDisplay.

Sub EnableControl(Enable, Tag)

Value: Pass TRUE to enable the control, and FALSE to disable it

Tag: The tag ID of the control to enabled or disable

Enables or disables a control.

3.5.1.8 SBRunning

These are functions that can be accessed from scripts via the **SBRunning** object. For example:

```
SBLocation.Warning "Filename", "Warning message"
```

The **SBRunning** object is only accessible from Runtime scripts.

Function FolderHasFiles(Name)

Name: The name of the folder (not including the base folder)

Return value: True if the file contains files

This function returns True if a folder that has been scanned contains any files (that have not been filtered out). Note that the folder name should not include the base folder.

Function GetCurrentFileVer(Filename, Left)

Filename: The filename (not including the base folder)

Left: Pass True to get the details of the left/source file

Return value: The current file version, or -2 if it isn't set to use a different version

This function returns the version of the file that is going to be restored, or -2 if it isn't set to use restore a version.

See the function GetFileVerCount to get the number of versions a file has.

Note that the filenames do not include the base folder.

Function GetFileAction(Filename)

Name: The name of the file (not including the base folder)

Return value: The action to be performed

This function returns the action that will be performed for a particular file. Note that the filename should not include the base folder. If the file does not exist then 0 (CACTION_ERROR) is returned.

For folders see GetFolderAction

Function GetFileAttrs(Filename, Left)

Filename: The filename (not including the base folder)

Left: If True then return the attributes of the left/source file

Return value: The files attributes

This function returns a files filesystem attributes. Note that the filename should not include the base folder. If the file does not exist then -2 is returned. If the file has no file attributes,

then -1 is returned.

To set a files attributes use SetFileAttrs

See also GetFileDetails

Function GetFileDateTime(Filename, Left)

Filename: The filename (not including the base folder)

Left: If True then return the date & time of the left/source file

Return value: The files last modification date & time

This function returns the last modification date & time of a file (in the local timezone). Note that the filename should not include the base folder. If the file does not exist then 0.9 is returned. If the file has no date & time then 1.0 is returned.

To set a files modification date and time use SetFileDateTime

See also GetFileDetails

Function GetFileDetails(Filename, Left, ByRef StoredName, ByRef Size, ByRef Attrs, ByRef Hash, ByRef ModDateTime, ByRef Exists)

Filename: The filename (not including the base folder)

Left: Pass True to get the details of the left/source file

StoredName: The stored name of the file (if there is one)

Size: The size of the file in bytes (note this is a string)

Attr: The filesystem attributes of the file

Hash: The CRC32 hash value of the file (as a string)

ModDateTime: The last modification date & time of the file (local timezone)

Exists: Returned as True if the file exists on either the left/source or right/destination

Return value: False if the file does not exist on either the left/source or right/destination

This function returns all the details of a file in a single call. It is more efficient to use this function if more than a single piece of information on a file is required. Note that the filename should not include the base folder.

If the file does not exist at all on either the left/source or right/destination then False is returned and Exists is set to False. However, if you request the details of the file on the left/source and the file only exists on the right/destination, for example, then False is returned but Exists is set to True. If Exists is True then the file does exist on the left/source or right/destination (or both).

For example:

```
DoesNotExist = SBRunning.GetFileDetails(Filename, True, StoredName, Size, _
```

Attrs, Hash, ModDateTime, Exists)

Function GetFileDiff(Filename)

Filename: The filename (not including the base folder)

Return value: The difference between the files

This function returns the differences between a file in the left/source and/or right/destination. Note that the filename should not include the base folder. If the file does not exist then 0 (CDIFF_IDENTICAL) is returned.

For folders see GetFolderDiff

Function GetFileHash(Filename, Left)

Filename: The filename (not including the base folder)

Left: If True then return the hash value of the left/source file

Return value: The files CRC32 hash value (as a string)

This function returns the calculated CRC32 hash value (in string format) of a file. This function does not calculate the hash value, it merely returns the hash value that has already been calculated. Note that the filename should not include the base folder. If the file does not exist, or has no CRC32 hash value, then an empty string is returned.

To set a files hash value use SetFileHash

See also GetFileDetails, MD5, and CRC32

Function GetFileSize(Filename, Left)

Filename: The filename (not including the base folder)

Left: If True then return the size of the left/source file

Return value: The files size in bytes

This function returns the size of a file (in bytes). Note that the filename should not include the base folder. If the file does not exist then -2 is returned. If the file size is unknown then -1 is returned. Note that this function returns a string to avoid the 32-bit integer limit in VBScript.

To set a files size use SetFileSize

See also GetFileDetails

Function GetFileVerCount(Filename, Left)

Filename: The filename (not including the base folder)

Left: Pass True to get the details of the left/source file

Return value: The number of versions a file has, or zero if it has no versions

This function returns the number of versions a file has, or zero if it has no versions.

See the function GetCurrentFileVer to get the version number which will be restored.

Note that the filenames do not include the base folder.

Function GetFolderAction(Name)

Name: The name of the folder (not including the base folder)

Return value: The action to be performed

This function returns the action that will be performed for a particular folder. Note that the folder name should not include the base folder. If the folder does not exist then 0 (CACTION_ERROR) is returned.

For files see GetFileAction

Function GetFolderDiff(Name)

Name: The name of the folder (not including the base folder)

Return value: The difference between the folders

This function returns the differences between a folder in the left/source and/or right/destination. Note that the folder name should not include the base folder. If the folder does not exist then 0 (CDIFF_IDENTICAL) is returned.

For files see GetFileDiff

Function GetMovedName(Filename)

Filename: The filename (not including the base folder)

Return value: The file it has been moved from or to

This function returns the name a file has been moved from or to. For example, if SyncBack detects that file \folder\file.txt has been moved to \somewhere\else.txt then calling GetMovedName("\folder\file.txt") will return '\somewhere\else.txt' and calling GetMovedName("\somewhere\else.txt") will return '\folder\file.txt'

Note that the filenames do not include the base folder.

If the file has not been moved, or does not exist, then an empty string is returned.

Function `GetRuntimeValue(ValueToGet)`

ValueToGet: The runtime value to return

Return value: The value as a signed 32-bit integer

Returns a runtime value for the current profile. For more details see the description of the `GetRuntimeValueStr` function.

Note that this function returns 32-bit values, so if the value is 64-bit you should instead use the `GetRuntimeValueStr` function to return it as a string.

Function `GetRuntimeValueStr(ValueToGet)`

ValueToGet: The runtime value to return

Return value: The value as a string

Returns a runtime value for the current profile. This is useful to retrieve constantly updating information about the current profile, e.g. how many bytes it has copied so far. Sometimes this information is also available using variables, but often the variables aren't updated until certain stages have completed.

The value is returned as a string to get around 64-bit limitations in VBScript. Some of the values are 32-bit integers so they can safely be retrieved using the `GetRuntimeValue` function instead.

`ValueToGet` can be one of the following:

The following counters are updated during the scanning stage. A file or folder with the same name on both the source/left and destination/right is counted as one file or folder and not two. These are signed 32-bit integers.

0: The number of files scanned.

The following counters are updated during the comparison stage (what the differences are). A file or folder with the same name on both the source/left and destination/right is counted as one file or folder and not two. These are signed 32-bit integers.

- 1: The number of files that have been changed
- 2: The number of files whose contents have changed (hash values are different)
- 3: The number of files that are only in the destination/right
- 4: The number of files that are only in the source/left

- 5: The number of files whose modification date & time have changed
- 6: The number of files whose size has changed
- 7: The number of files whose attributes have changed
- 8: The number of files whose name is same but the case has changed
- 9: The number of directories that have been changed
- 10: The number of directories that are only in destination/right
- 11: The number of directories that are only in source/left
- 12: The number of files that are identical files or only have versions

The following are comparison counters that indicate what is left to be done. They are set during the comparison stage and decremented as the profile runs. These are signed 32-bit integers.

- 13: The number of files to skip
- 14: The number of files to be prompted on
- 15: The number of files to delete from source/left
- 16: The number of files to delete from destination/right
- 17: The number of files to copy from source/left
- 18: The number of files to copy from destination/right
- 19: The number of files to move from source/left
- 20: The number of files to move from destination/right
- 21: The number of files in the source/left to have date & time, attributes, and/or case changed
- 22: The number of files in the destination/right to have date & time, attributes, and/or case changed
- 23: The number of files to restore old versions of in source/left
- 24: The number of files to restore old versions of in destination/right
- 25: The number of files to be renamed on the source/left
- 26: The number of files to be renamed on the destination/right

The following are comparison counters that indicate what is left to be done. They are set during the comparison stage and decremented as the profile runs. These are signed 64-bit integers.

- 27: The total number of bytes to be copied to the source/left. This includes files to be moved to the source/left.
- 28: The total number of bytes to be copied to the destination/right. This includes files to be moved to the destination/right.
- 29: The total number of bytes to delete from the source/left. This includes files to be moved to the destination/right.
- 30: The total number of bytes to delete from the destination/right. This includes files to be moved to the source/left.

The following are results counters that indicate what has been done so far. They are incremented while the profile is running. These are signed 32-bit integers.

- 31: The number of files skipped so far
- 32: The number of files and folders prompted on so far
- 33: The number of files and folders renamed on the source/left so far
- 34: The number of files and folders renamed on the destination/right so far

- 35: The number of files deleted from the source/left so far
- 36: The number of files deleted from the destination/right so far
- 37: The number of files copied from the source/left so far
- 38: The number of files copied from the destination/right so far
- 39: The number of files moved from the source/left so far
- 40: The number of files moved from the destination/right so far
- 41: The number of files on the source/left that so far have had their last modification date and time updated
- 42: The number of files on the destination/right that so far have had their last modification date and time updated
- 43: The number of files on the source/left that so far have had their attributes updated
- 44: The number of files on the destination/right that so far have had their attributes updated
- 45: The number of files on the source/left that so far have old versions restored
- 46: The number of files on the destination/right that so far have old versions restored

The following are bytes counters that indicate what has been done so far. They are incremented while the profile is running. These are signed 64-bit integers.

- 47: The total number of bytes copied to the source/left so far. This includes files being moved to the source/left.
- 48: The total number of bytes copied to the destination/right so far. This includes files being moved to the destination/right.
- 49: The total number of bytes deleted from the source/left so far. This includes files being moved from the source/left.
- 50: The total number of bytes deleted from the destination/right so far. This includes files being moved from the destination/right.
- 51: The total number of bytes replaced (overwritten) on the source/left so far.
- 52: The total number of bytes replaced (overwritten) on the destination/right so far.

The following are error counters that indicate the number of errors so far. They are incremented while the profile is running. These are signed 32-bit integers.

- 53: The total number of compression errors so far
- 54: The total number of files that have failed to be copied, deleted, or moved so far
- 55: The total number of files that cannot have their hash value calculated so far
- 56: The total number of non-critical errors so far

The following are various values that are 64-bit signed integers.

- 57: The total number of bytes free initially on the source/left. This is set at the beginning of the profile and not updated. If the bytes free cannot be retrieved then -1 is returned.
- 58: The total number of bytes free initially on the destination/right. This is set at the beginning of the profile and not updated. If the bytes free cannot be retrieved then -1 is returned.
- 59: The number of files that will be updated. This is not changed once set.
- 60: The number of kilo-bytes that will be updated. This is not changed once set.

Function GetStoredFileName(Filename, Left)

Filename: The filename (not including the base folder)

Left: If True then return the stored name of the left/source file

Return value: The files stored filename

This function returns the stored (actual) filename of a file. Note that the filename should not include the base folder. If the file does not exist then an empty string is returned. If it doesn't have a stored name then it's normal filename is returned.

See the function `GetStoredFolderName` for information on what a stored name is.

See also `GetFileDetails`

Function `GetStoredFolderName(Filename, Left)`

Filename: The folder name (not including the base folder)

Left: If True then return the stored name of the left/source folder

Return value: The folders stored filename

This function returns the stored (actual) name of a folder. Note that the filename should not include the base folder. If the folder does not exist then an empty string is returned. If it doesn't have a stored name then it's normal name is returned.

A folder may have a different name on one side, e.g. the right/destination name may be different from the left/source name. For example, if a folder is on a DVD then its name may be different from the name it had on the hard-drive (when it was copied to the DVD). This is because the filesystem on the DVD has stricter rules on what a valid filename is and the maximum length of that name.

Generally stored names are only used when restoring from a CD/DVD

For example:

```
GetStoredFolderName("\Folder\A long folder name\", False)
```

may return "FOLDER\ALONGFOL"

To get the stored name of a file use `GetStoredFileName`

Function `GetVersionDetails(Filename, Left, VersionNumber, ByRef VersionFilename, ByRef Size, ByRef Attrs, ByRef ModDateTime, ByRef WhenVersioned)`

Filename: The filename (not including the base folder)

Left: Pass True to get the details of the left/source file version

VersionFilename: The filename for the version of the file

Size: The size of the file in bytes (note this is a string)

Attr: The filesystem attributes of the file

ModDateTime: The last modification date & time of the file (local timezone)

WhenVersioned: When the version was created (local timezone)

Return value: False if the file version does not exist

This function returns all the details of for a specific version of a file. Note that the filename should not include the base folder. Version numbers are zero based, with zero being the oldest version. Passing -1 as the version number will get you the newest version.

If the file or version does not exist then False is returned.

For example:

```
DoesNotExist = SBRunning.GetVersionDetails(Filename, True, 0, VersionName, Size, _  
    Attrs, ModDateTime, WhenVersioned)
```

See also `GetFileVerCount` and `GetCurrentFileVer`

Function `LeftFileExists(Filename)`

Filename: The filename to check for

Return value: True if the file exists in the left/source

Returns True if the file exists in the left/source. Note that the filename should not include the base folder.

This property is read-only.

See also `LeftFolderExists` and `RightFileExists`

Function `LeftFolderExists(Name)`

Name: The name of the folder to check for

Return value: True if the folder exists in the left/source

Returns True if the folder exists in the left/source. Note that the name should not include the base folder.

This property is read-only.

See also `RightFolderExists` and `LeftFileExists`

Function `RightFileExists(Filename)`

Filename: The filename to check for

Return value: True if the file exists in the right/destination

Returns True if the file exists in the right/destination. Note that the filename should not include the base folder.

This property is read-only.

See also RightFolderExists and LeftFileExists

Function RightFolderExists(Name)

Name: The name of the folder to check for

Return value: True if the folder exists in the right/destination

Returns True if the folder exists in the right/destination. Note that the name should not include the base folder.

This property is read-only.

See also LeftFolderExists and RightFileExists

Function SetFileAttrs(Filename, Left, NewAttrs)

Filename: The filename (not including the base folder)

Left: Pass True to set the attributes of the left/source file

NewAttrs: The file attributes to set

Return value: The files attributes, or -1 on failure

This function sets the filesystem attributes of a file. Note that the filename should not include the base folder.

This function does not actually change the attributes of the file. It instead tells SyncBack what the attributes of the file are.

See also GetFileAttrs

Function SetFileDateTime(Filename, Left, NewDateTime)

Filename: The filename (not including the base folder)

Left: Pass True to set the date & time of the left/source file

NewDateTime: The date & time to use (local timezone)

Return value: The date & time of the file, or 1.0 on failure

This function sets the last modification date & time of a file. Note that the filename should not include the base folder.

This function does not actually change the last modification date & time of the file. It instead tells SyncBack what the last modification date & time of the file is.

See also GetFileDateTime

Function SetFileHash(Filename, Left, NewHash)

Filename: The filename (not including the base folder)

Left: Pass True to set the hash value of the left/source file

NewHash: The CRC32 hash value of the file

Return value: The hash value of the file, or empty string on failure

This function sets the CRC32 hash value (string format) of a file. Note that the filename should not include the base folder.

This function does not actually change the hash value of the file. It instead tells SyncBack what the hash value of the file is. Note that the hash value is not used unless the profile is configured to use hashing for file comparisons.

See also GetFileHash

Function SetFileSize(Filename, Left, NewSize)

Filename: The filename (not including the base folder)

Left: Pass True to set the size of the left/source file

NewSize: The size of the file

Return value: The size of the file, or -1 on failure

This function sets the size of a file. Note that the filename should not include the base folder. To avoid the 32-bit limit in VBScript the NewSize is a string, and so is the return value.

This function does not actually change the size of the file. It instead tells SyncBack what the size of the file is.

See also GetFileSize

Function SetFileVersion(Filename, Left, NewVersion)

Filename: The filename (not including the base folder)

Left: Pass True to set the details of the left/source file

NewVersion: The version number to use

Return value: The current file version, -1 if an error occurred, or -2 if it isn't set to use a version

This function sets the current version of a file. Version numbers are zero based, with zero being the oldest version. Passing -1 as the version number will get you the newest version. Passing -2 as the version number will set it so no version is restored.

The only place you can change the version of a file is in the call to RunPreCopyCheck. If SetFileVersion is called before or after that point then the results are undefined.

You cannot change the current version of a file if the file is to be deleted or replaced, for example. In this case the function will return -1

Note that the filenames do not include the base folder.

See the functions GetFileVerCount and GetCurrentFileVer

Function Sleep(Seconds)

Seconds: The number of seconds to sleep

Return value: True if the user has aborted the profile

Sleep for the specified number of seconds, but also checks to see if the profile has aborted while sleeping. If the user has aborted the profile run then the function returns True immediately, i.e. the sleep is aborted.

Property Abort

Returns TRUE if the profile is being aborted. Set it to TRUE to abort the profile. Note that the abort may not be immediate, and an abort cannot be aborted (i.e. once True you cannot change it to False).

Property DifferentialBackup

Returns True if the profile is being run as a differential backup. If it is then that also implies it is a Fast Backup profile.

This property is read-only.

See also FastBackupType and DynamicFastBackup

Property DynamicFastBackup

Returns True if the profile is being run as a dynamic fast backup backup.

This property is read-only.

See also `FastBackupType` and `DifferentialBackup`

Property `FastBackupType`

Returns the type of Fast Backup the profile is being run as:

- 0 = It is not a Fast Backup
- 1 = It is an original Fast Backup
- 2 = It is an archival Fast Backup

This property is read-only.

See also `DynamicFastBackup` and `DifferentialBackup`

Property `FileCount`

Returns the number of files in the profile.

This property is read-only.

For the number of folders see `FolderCount`

Property `FolderCount`

Returns the number of folders (directories) in the profile.

This property is read-only.

For the number of folders see `FileCount`

Property `FullBackup`

Returns `TRUE` if this is a full-backup profile run. The value can be set to `TRUE` or `FALSE`, but note that any change to this value will only be relevant for the next run of the profile. Because of this any change made will not be reflected in reading the value, i.e. if `FullBackup` returns `FALSE` and you set it to `TRUE` then it will still return `FALSE`. You can check the current value by using `GetProperty`, e.g. `SBVariables.GetProperty("S_FULLBACKUP", "N", True)`

The setting will be automatically reset to `FALSE` after the profile has run (if it was a re-scan). Because of this it is recommend that you set the value in a call to `RunProfileResult` as this is called after any change the program makes to the value.

See also `FastBackupType`

Property GroupName

Returns the name of the group profile, or empty string if the profile is not being run as part of a group.

This property is read-only.

See also VisualGroupName

Property LeftAbilities

Returns the abilities of the source/left location.

This property is read-only.

See also RightAbilities

Property LeftFolder

Returns the left/source base folder, e.g. C:\My Files\To Backup\

This property is read-only.

See also RightFolder and LeftName

Property LeftName

Returns the name of the left/source, e.g. My Files To Backup

This property is read-only.

See also RightName

Property Name

Returns the name of the profile.

This property is read-only.

See also GroupName and ProfileType

Property ProfileType

Returns the type of the profile.

This property is read-only.

See also ProfileTypeDesc

Property ProfileTypeDesc

Returns a description of the profile based on its type (see ProfileType).

This property is read-only.

Property Restore

Returns True if the profile is being run in restore mode.

This property is read-only.

See also Simulated

Property RightAbilities

Returns the abilities of the destination/right location.

This property is read-only.

See also LeftAbilities

Property RightFolder

Returns the right/destination base folder, e.g. C:\My Backup Files\

This property is read-only.

See also LeftFolder and RightName

Property RightName

Returns the name of the right/destination, e.g. My Backup Files

This property is read-only.

See also LeftName

Property Simulated

Returns True if the profile is being run in simulation mode.

This property is read-only.

See also Restore

Property Unattended

Returns True if the profile is being run unattended, i.e. the script should not prompt the user or expect any user interaction.

This property is read-only.

Property VisualGroupName

If the profile is being run as part of a group, or it is being run from the user interface from within a group, then this is the name of the group. Note that the property `GroupName` is returned as an empty string if the profile is not being run as part of a group, but `VisualGroupName` may still return a group name if the user has clicked on the profile in a group and then run it.

This property is read-only.

Sub CriticalError(Filename, TheError, IncErrorCount)

Filename: The filename of the file the error relates to

TheError: The error message

IncErrorCount: Pass True if the critical error count should be incremented

This subroutine records a critical error in the profiles log file. The error must be related to a particular file. Note that the filename should not include the base folder. For example, if the base folder is `C:\My Files\`, and the file is `C:\My Files\Folder\Filename.txt`, then the filename to pass should be `\Folder\Filename.txt`

A critical error means the profile has failed. The profile will continue running but once completed its last run status will be 'Failure'.

For example:

```
CriticalError "\folder\file.txt", "Your wife says you cannot copy this file", TRUE
```

See also `NotCriticalError` and `Warning`

Sub DebugOut(Str1, Str2, Level)

Str1: Typically this is a filename or refers to the object the error is about

Str2: Typically this is the error message

Level: The severity of the message

This subroutine records a message in the debug log. Note that nothing is recorded if debug output is not enabled. Level refers to the severity of the message (1=error, 5=warning, 10=information). The user can set the minimum level a message should be, e.g. they could filter out anything above level 5.

For example:

```
DebugOut "\folder\file.txt", "Klingons off the starboard bow", 1
```

To record a message in the Windows event log use EventOut

Sub EventOut(Msg, Level)

Msg: The message to record in the Windows event log

Level: The severity of the message

This subroutine records an event in the Windows Applications Event Log. Level refers to the severity of the message (1=error, 5=warning, 10=information).

For example:

```
EventOut "The toilet seat has not been left down", 1
```

To record a message in the debug log for the profile use DebugOut

Sub Exception(ExceptionReport)

ExceptionReport: The exception report

This subroutine records an exception report in the profiles log file. An exception report is typically a number of lines of text that detail where an unexpected error occurred in the program/script. This is usually done when something serious unexpectedly happens in the program, e.g. an attempt is made to read from a nil pointer.

To record a message in the debug log for the profile use DebugOut

To record a message in the Windows event log use EventOut

Sub NotCriticalError(Filename, TheError)

Filename: The filename of the file the error relates to

TheError: The error message

This subroutine records a non-critical error in the profiles log file. The error must be related to a particular file. Note that the filename should not include the base folder. For example, if the base folder is C:\My Files\, and the file is C:\My Files\Folder\Filename.txt, then the

filename to pass should be \Folder\Filename.txt

A non-critical error does not mean the profile has failed.

For example:

```
NotCriticalError "\folder\file.txt", "The date and time of the file could not be set"
```

See also `CriticalError` and `Warning`

Sub `RebootRequired()`

This subroutine tells SyncBack that the computer should be rebooted after the profile has completed. For example, if a file can only be replaced on reboot then this subroutine should be called. Note that a reboot is not guaranteed to occur as the user may decide not to reboot.

Sub `SysLogMessage(Msg, Severity)`

Msg: The message to send to the SysLog server

Severity: The severity of the message (ranging from 0 to 7)

This subroutine sends a message to the SysLog server. Note that nothing is sent unless a SysLog server has been configured to be used. The severity is an integer value ranging from 0 (Emergency) to 7 (Debug).

For example:

```
SysLogMessage "Klingons off the starboard bow", 3
```

To record a message in the Windows event log use `EventOut`

Sub `Warning(Filename, TheWarning)`

Filename: The filename of the file the warning relates to

TheError: The warning message

This subroutine records a warning message in the profiles log file. The warning must be related to a particular file. Note that the filename should not include the base folder. For example, if the base folder is C:\My Files\, and the file is C:\My Files\Folder\Filename.txt, then the filename to pass should be \Folder\Filename.txt

A warning does not mean the profile has failed.

For example:

```
Warning "\folder\file.txt", "The file has your credit card number in it"
```

See also NotCriticalError and CriticalError

3.5.1.9 SBSsystem

These are functions that can be accessed from scripts via the **SBSsystem** object. For example:

```
SBSsystem.Say "Hello"
```

The **SBSsystem** object is accessible from any type of script.

Function BuildDateTime(Day, Mon, Year, Hour, Min, Sec, MSec)

Day: Day of the month

Mon: Month of the year

Year: Year

Hour: Hour of the day (24-hour clock)

Min: Minute of the hour

Sec: Second of the minute

MSec: Milli-seconds of the second

Return value: A date & time value, or 1.0 if the parameters passed are invalid

This function builds a date & time that can be used with SyncBack, e.g. in the AddFile function. Note that the date variable type in VBScript is already compatible with SyncBack.

For example, for 24th June 1971 15:31:02.123 you would call:

```
BuildDateTime(24, 6, 1971, 15, 31, 02, 123)
```

Function CheckNetworkDrive(UNCOrPath)

UNCOrPath: UNC path or drive path to check

Return value: An error message on failure

This function checks to see if a network drive can be reached. A UNC path, e.g. \\server\share\folder\, or networked drive path, e.g. Z:\, can be passed. If it can be reached then an empty string is returned, otherwise an error message is returned.

To check if a computer can be reached (via its hostname or IP address) use the function Ping

Function CompareFileNames(Filename1, Filename2, CaseSensitive)

Filename1: A Unicode filename to compare with Filename2

Filename2: A Unicode filename to compare with Filename1

CaseSensitive: TRUE if the comparison should be case sensitive

Return value: Returns zero if the strings are ordinally identical

This function compare two filenames for ordinal (not linguistic) equality. Digits in the strings are considered as numerical content rather than text. For performance reasons, if you are just testing for equality (or not) then use SameFileNames instead.

* Returns zero if the strings are identical.

* Returns > 0 if the string pointed to by Filename1 has a greater value than that pointed to by Filename2.

* Returns < 0 if the string pointed to by Filename1 has a lesser value than that pointed to by Filename2.

See also SameFileNames

Function CompressFile(Filename)

Filename: The full filename of the file to compress

Return value: Returns error message on failure

This function compresses a file or folder using NTFS compression. If the file or folder is already compressed then no error is returned.

See also DecompressFile

Function COMRegister(Filename)

Filename: Filename of the COM/OCX/ActiveX component to register

Return value: An error message on failure

This function registers an COM/OCX/ActiveX component (DLL or EXE). Note that typically the user must have Administrator privileges to register components. On success an empty string is returned, otherwise an error message is returned.

Function CRC32(Filename)

Filename: The filename of the file to calculate the hash value of

Return value: CRC32 hash value of the file, or empty string on failure

This function returns the CRC32 hash value of a file in string format. Note that it may take a long time to calculate the hash values of large files, or files accessed via a slow connection.

See also MD5

Function DecodeString(Str)

Str: The string to decode

Return value: The decoded string

This function decodes a string that has been previously encoded with the EncodeString function.

Function DecompressFile(Filename)

Filename: The full filename of the file to decompress

Return value: Returns error message on failure

This function decompresses an NTFS compressed file. If the file does not exist, or is not compressed, then no error message is returned. Note that it cannot decompress folders.

See also CompressFile

Function DecryptFile(Filename)

Filename: The full filename of the file to decrypt

Return value: Returns error message on failure

This function decrypts an NTFS encrypted file. If the file does not exist, or is not encrypted, then no error message is returned. Note that it cannot decrypt folders.

See also EncryptFile

Function DecryptString(Str)

Str: The string to decrypt

Return value: The decrypted string

This function decrypts a string that has been previously encrypted with the EncryptString function.

Function EncodeString(Str)

Str: The string to encode

Return value: The encoded string

This function encodes a string so that it can safely be stored in INI files, the registry, etc. It is useful for when the string may contain characters that may be invalid for the storage medium.

See also DecodeString

Function EncryptFile(Filename)

Filename: The full filename of the file to encrypt

Return value: Returns error message on failure

This function encrypts a file using NTFS encryption. If the file does not exist, or is already encrypted, then no error message is returned. If the file is compressed, EncryptFile will decompress the file before encrypting it. Note that it cannot encrypt folders.

See also DecryptFile

Function EncryptString(Str)

Str: The string to encrypt

Return value: The encrypted string

This function encrypts a string so that it is no longer plain text. Note that the encrypted string is formatted as a list of numbers (seperated by spaces). This means it can safely be stored in INI files, the registry, etc. so there is no need to use the EncodeString function.

See also DecryptString

Function Exec(CmdLine, WaitSecs, ByRef RetVal, ByRef ErrMsg)

CmdLine: The full program name and parameters

WaitSecs: The number of seconds to wait

RetVal: The integer return value of the program executed

ErrMsg: An error message if the program could not be executed

Return value: Returns True if the program was executed

This function executes a program and optionally waits for it to complete.

CmdLine can contain command line parameters. It must be a fully qualified filename, e.g. C:\abc\def param1 param2

If WaitSecs < 0 then it will wait forever the program to finish

If WaitSecs = 0 then it will not wait for the program to finish

If WaitSecs > 0 then it will wait that number of seconds for it to finish

You need to wrap double quotes around the program name (no need if it does not contain spaces) otherwise the return value will always be 1. Also, command line parameters should be wrapped in double quotes if they contain spaces. For example:

```
"D:\Documents and Settings\Mick\Desktop\deldest.bat" param1 "param 2"
```

and NOT:

```
D:\Documents and Settings\Mick\Desktop\deldest.bat param1 param 2
```

CmdLine can be prefixed with special parameters:

/min to minimize the window (and not make it active)

/max to maximize the window

/hide to hide the window

e.g. /min "D:\Documents and Settings\Mick\Desktop\deldest.bat" "param 1"

If you choose to wait for the program to terminate then its return value is returned in RetVal. Otherwise the value in RetVal is unknown and should be ignored.

On failure ErrMsg will be set to an error message.

If the program was executed then True is returned. False is returned if the program could not be executed, e.g. doesn't exist. Note that True does not mean the program did what you required. To verify that check RetVal.

For example:

```
Executed = SBSsystem.Exec("C:\Windows\System32\notepadx.exe", 0, RetVal, ErrMsg)
```

To view a web page use the function OpenBrowser, and to open a file use the function OpenFile

Function GetProfileName(Idx)

Idx: The profile name to retrieve (first profile is zero)

Return value: The name of the profile, or empty string if there is no such profile

This function retrieves the name of a profile. To get the number of profiles use the

ProfileCount property. Note that the first profile is zero (0), and the last is ProfileCount - 1

Note that you should call ProfileCount to refresh the list.

Function GMTToLocal(GMTTime)

GMTTime: A date & time in the GMT/UTC timezone

Return value: A local date & time

This function convert a GMT/UTC date & time to a local date & time. The function LocalToGMT will convert a local date & time to a GMT/UTC date & time.

Function IsFolder(Filename)

Filename: The filename to check

Return value: True if Filename has a trailing backslash

This function checks if a string has a trailing backslash, and if so, returns True. Note that the function does not check if the file or folder actually exists.

For example:

```
SBSystem.IsFolder("c:\abc\def")
```

will return False, but

```
SBSystem.IsFolder("c:\abc\def\")
```

will return True.

Function LanguageCode(DomainName, ToTranslate)

DomainName: The translation domain to use (use default for SyncBack translations)

ToTranslate: The string to translate

Return value: The translated string, or ToTranslate if it cannot be translated

This function translates a string from English to the current language, assuming a translation is available. See also AddTranslationDomain to add translations.

Function LocalToGMT(GMTTime)

LocalTime: A date & time in the local timezone

Return value: A GMT/UTC date & time

This function convert a local date & time to a GMT/UTC date & time. The function GMTToLocal will convert a GMT/UTC date & time to a local date & time.

Function MD5(Filename)

Filename: The filename of the file to calculate the hash value of

Return value: MD5 hash value of the file, or empty string on failure

This function returns the MD5 hash value of a file in string format. Note that it may take a long time to calculate the hash values of large files, or files accessed via a slow connection.

See also CRC32

Function OpenBrowser(URL)

URL: The URL of the web page to open in a browser

Return value: True if the browser was opened to display the web page

This function opens the default web browser and directs it to go to the URL given. Note that it returns True only if the browser is opened, but it cannot know if the URL itself is valid or accessible.

For example:

```
SBSystem.OpenBrowser("http://www.2BrightSparks.com/")
```

To open a file use the function OpenFile

To check if a server can be reached use the function Ping

Function OpenFile(Filename)

Filename: The complete filename of the file to open

Return value: An error message of failure, else an empty string

This function opens a file, e.g. a text document, with the default program for that file type, e.g. Notepad. Note that it returns an empty string only if the appropriate program was opened, but it cannot know if the file itself was opened by the program.

For example:

```
SBSystem.OpenFile "c:\folder\file.txt"
```

On failure an error message is returned.

To open a web page use the function `OpenBrowser`

Function `Ping(HostnameOrIP)`

HostnameOrIP: An I.P. address or hostname

Return value: An error message on failure

This function 'pings' a server to see if it is accessible. Note that some servers do not respond to ping requests (e.g. at time of writing microsoft.com does not). If the server responds to the ping requests then an empty string is returned, otherwise an error message is returned.

For example:

```
SBSystem.Ping("google.com")
```

To check if a network drive is accessible use the function `CheckNetworkDrive`

Function `SameFileNames(Filename1, Filename2, CaseSensitive)`

Filename1: A Unicode filename to compare with `Filename2`

Filename2: A Unicode filename to compare with `Filename1`

CaseSensitive: TRUE if the comparison should be case sensitive

Return value: Returns TRUE if the strings are ordinally identical

This function compare two filenames for ordinal (not linguistic) equality. Digits in the strings are considered as numerical content rather than text.

See also `CompareFileNames`

Function `SBCmdLineParam(ParamIdx)`

ParamIdx: The command line parameter to retrieve (first param is zero)

Return value: The command line parameter, or empty string if there is no such parameter

This function retrieves the command line parameters passed to SyncBack. To get the number of parameters use the `SBCmdLineParamsCount` property. Note that the first parameter is zero (0), and the last is `SBCmdLineParamsCount - 1`

Function `SBVersion(Filename, ByRef Major, ByRef Minor, ByRef Release, ByRef Build)`

Filename: The filename of an executable, or an empty string for SyncBack

Major: The major version number of the executable

Minor: The minor version number of the executable

Release: The release version number of the executable

Build: The build version number of the executable

Return value: If an empty filename was passed then the name SyncBack is using

This function gets the version information from an executable. If Filename is passed as an empty string then it will return the version information of SyncBack itself. Also, passing an empty string will return the application name of SyncBack. In some situations SyncBack may be branded under a different name. If a filename is passed then the filename itself (stripped of the path) is returned.

For example:

```
AppName = SBSSystem.SBVersion(Filename, Major, Minor, Release, Build)
```

Function SetCreateDateTime(Filename, LocalDateTime)

Filename: Complete filename of the file to change the creation date & time of

LocalDateTime: The local date & time to change the creation date & time to

Return value: An error message on failure

This function sets the creation date & time of a file to the one supplied. The filename must be a complete filename, and the date & time should be in the local timezone. It returns an error message on failure.

For example:

```
SBSSystem.SetCreateDateTime("c:\folder\file.txt", Now)
```

Function SetFileAttributes(Filename, Attribs)

Filename: Complete filename of the file to change the attributes of

Attribs: The attributes to use

Return value: An error message on failure

This function sets the Windows file attributes of file. The filename must be a complete filename. It returns an error message on failure.

For example:

```
SBSSystem.SetFileAttributes("c:\folder\file.txt", 1)
```

Function SetLastModDateTime(Filename, LocalDateTime)

Filename: Complete filename of the file to change the modification date & time of

LocalDateTime: The local date & time to change the modification date & time to

Return value: An error message on failure

This function sets the last modification date & time of a file to the one supplied. The filename must be a complete filename, and the date & time should be in the local timezone. It returns an error message on failure.

For example:

```
SBSystem.SetLastModDateTime("c:\folder\file.txt", Now)
```

Function UpdateFileStatus(Status)

Status: The file status message to display in the SyncBack main window

Return value: True if the profile is terminating

This function updates the current file status shown in the SyncBack main window. Note that SyncBack itself will display the appropriate status messages when files are being copied, deleted, etc. The current file status message is the message displayed below the current status messages.

For example:

```
SBSystem.UpdateFileStatus("Taking the stereo from your car...")
```

See also UpdateStatus

Function UpdateStatus(Status)

Status: The status message to display in the SyncBack main window

Return value: True if the profile is terminating

This function updates the current status shown in the SyncBack main window. Note that SyncBack itself will display the appropriate status messages when tasks are being performed, e.g. Scanning for changes. The status message is the message displayed above the current file status message.

For example:

```
SBSystem.UpdateStatus("Taking your dog for a walk...")
```

See also UpdateFileStatus

Property LanguageCode

This property returns the language code of the user interface, e.g. "en" for English. See also AddTranslationDomain and TranslateString

This is a read-only property.

Property NoDesktop

This property returns True if there is a desktop. It would return False, for example, if no user is currently interactively logged in, or if a different user is currently interactively logged in.

This is a read-only property.

Property ProfileCount

This property returns the number of profiles. To retrieve the names of the profiles use GetProfileName

This is a read-only property.

Property SBCmdLineParamsCount

This property returns the number of command line parameters passed to SyncBack. The parameters themselves can be retrieved using the SBCmdLineParam function.

This is a read-only property.

Property SBFilename

This property returns the complete filename of the SyncBack program.

This is a read-only property.

Property ScriptFilename

This property returns the complete filename of the script itself. This should not be stored as it may change, e.g. if imported onto another computer then the path may be different. To get just the path the script is in use ScriptPath

This is a read-only property.

Property ScriptPath

This property returns the path (directory) the script itself is in. This should not be stored as it may change, e.g. if imported onto another computer then the path may be different. To get the complete filename of the script use the function ScriptFilename

This is a read-only property.

Property UniqueID

This property returns a universally unique 32 character long string.

This is a read-only property.

Sub AddTranslationDomain(DomainName)

DomainName: The name of the translation domain

This function adds a custom translation domain so that a script can provide its own translations for strings. Strings are translated using TranslateString.

A domain is the name of a .MO translation file (without the .MO extension). The .MO file must be placed into the appropriate \locale\[language code]LC_MESSAGES\ sub-folders of the SyncBack installation folder. There should be a .MO file for each language that strings can be translated into. MO files are created using the freeware POEdit program.

See the TranslationExample.vbs for an example.

Sub Say(ToSay)

ToSay: What the computer should say, or the filename of a .WAV file

This subroutine uses the speech engine in Windows to have the computer say what you request. It can also be used to play .WAV files (by passing the filename).

For example:

```
SBSystem.Say "2 bright sparks rock my world"
```

Sub Sleep(Seconds)

Seconds: The number of seconds to sleep

This function sleeps for the specified number of seconds. Note that you should not sleep for more than a few seconds in case the user wants to abort. The script (and profile or anything else) cannot abort while sleeping.

For example:

```
SBSystem.Sleep 2
```

See also `SBRunning.Sleep`

3.5.1.10 SBVariables

These are functions that can be accessed from scripts via the **SBVariables** object. For example:

```
SBSystem.SetProperty "MyVar", "Value"
```

The **SBVariables** object is accessible from Main Interface, Runtime and Profile Configuration scripts.

Function Count()

Return value: The number of variables defined

This function returns the number of variables defined.

Function GetGlobalProperty(PropName, PropDefault, Internal)

PropName: The name of the property

PropDefault: The value to return if the property does not exist

Internal: Pass True to retrieve properties that SyncBack uses, otherwise it's a property created by a script

Return value: The value of the property (as a string) or an empty string on error

This function retrieves a global property (setting) value. This is different from `GetProperty` and `GetProfileProperty` because those are used with profiles (the property is profile specific).

Note that `PropName` must be a single property name.

To check if a property exists or not pass a default value that cannot be valid, e.g.

```
If SBVariables.GetGlobalProperty("PropName", "!NOTEXIST!", False) = "!NOTEXIST!"  
Then  
    ' Does not exist  
Else  
    ' Exists  
End If
```

See also SetGlobalProperty

Some internal properties are encrypted, so you must decrypt the result using the SBSSystem.DecryptString function.

Function GetProfileProperty(ProfileName, PropName, PropDefault, Internal)

ProfileName: The name of the profile to read the property from

PropName: The name of the property

PropDefault: The value to return if the property does not exist

Internal: Pass True to retrieve properties that SyncBack uses, otherwise it's a property created by a script

Return value: The value of the property (as a string) or an empty string on error

This function retrieves a profile property (setting) value from a specific profile. It works the same way as GetProfileProperty except you can specify the profile.

See also SetProfileProperty

Some internal properties are encrypted, so you must decrypt the result using the SBSSystem.DecryptString function.

Function GetProperty(PropName, PropDefault, Internal)

PropName: The name of the property

PropDefault: The value to return if the property does not exist

Internal: Pass True to retrieve properties that SyncBack uses, otherwise it's a property created by a script

Return value: The value of the property (as a string) or an empty string on error

This function retrieves a profile property (setting) value. The difference between properties and variables is that properties are stored as part of the profiles settings, but variables are not. That means their value is kept between profile runs. Note that PropName must be a single property name.

To check if a property exists or not pass a default value that cannot be valid, e.g.

```
If SBVariables.GetProperty("PropName", "!NOTEXIST!", False) = "!NOTEXIST!" Then
    ' Does not exist
Else
    ' Exists
End If
```

See also SetProperty

Note that this function will do nothing if called from a Main Interface script. You must use the GetProfileProperty function.

Some internal properties are encrypted, so you must decrypt the result using the `SBSystem.DecryptString` function.

Function `GetVar(VarName)`

VarName: A string containing variables

Return value: `VarName` with the variables expanded

This function retrieves a variable value. The difference between properties and variables is that properties are stored as part of the profiles settings, but variables are not. Also, variables like environment variables are set by the operating system or other programs.

```
VarValue1 = SBVariables.GetVar("%USERPROFILE%")
VarValue1 = SBVariables.GetVar("Username is %USERNAME% and profile is %USERPROFILE%")
```

See also `SetProperty`, `GetVarName`, and `SetVar`

Function `GetVarName(Idx, ByRef Value)`

Idx: The number of the variable to get the name of (0=first variable)

Value: Value is set to the value of the variable

Return value: The name of the variable, or empty string on failure

This function retrieves the name of a variable. The first variable is variable zero (0). `SBVariables.Count` returns the number of variables defined.

For example:

```
VarName = SBVariables.GetVarName(0, VarValue)
```

Function `SetGlobalProperty(PropName, NewPropValue)`

PropName: The name of the property

NewPropValue: The new value of the property

Return value: The new value of the property (as a string) or an empty string on error

This function sets a global property (setting) value. This is different from `SetProperty` and `SetProfileProperty` because those are used with profiles (the property is profile specific).

Note that `PropName` must be a single property name. See `GetGlobalProperty` for retrieving global property values.

If you want to store the value encrypted, see the `SBSystem.EncryptString` function.

See DeleteGlobalProperty to delete global properties.

Note that you cannot change internal SyncBack properties.

Function SetProfileProperty(ProfileName, PropName, NewPropValue)

PropName: The name of the profile to delete the property from

PropName: The name of the property

NewPropValue: The new value of the property

Return value: The new value of the property (as a string) or an empty string on error

This function sets a profile property (setting) value for a specific profile. It works the same way as SetProperty except a profile can be specified. See GetProfileProperty for retrieving property values from a specific profile and DeleteProfileProperty to delete properties from a specific profile.

If you want to store the value encrypted, see the SBSSystem.EncryptString function.

Note that you cannot change internal SyncBack properties.

Function SetProperty(PropName, NewPropValue)

PropName: The name of the property

NewPropValue: The new value of the property

Return value: The new value of the property (as a string) or an empty string on error

This function sets a profile property (setting) value. Note that PropName must be a single property name. See GetProperty for retrieving property values.

See DeleteProperty to delete properties, and SetProfileProperty to set the property for a specific profile.

Note that you cannot change internal SyncBack properties.

If you want to store the value encrypted, see the SBSSystem.EncryptString function.

Note that this function will do nothing if called from a Main Interface script. You must use the SetProfileProperty function.

Function SetVar(VarName, NewVarValue)

VarName: The name of the variable to set

NewVarValue: The new value of the variable

Return value: The new value of the variable (string) or an empty string on failure

This function sets the value of a profile variable. Note that variables cannot be deleted.

```
SBVariables.SetVar("MyVariable", "The value")
```

See also GetVar

Sub DeleteGlobalProperty(PropName)

PropName: The name of the property to delete

This subroutine deletes a global property (setting) value. This is different from DeleteProperty and DeleteProfileProperty because those are used with profiles (the property is profile specific).

Note that PropName must be a single property name, and you cannot delete internal SyncBack properties.

Sub DeleteProfileProperty(ProfileName, PropName)

ProfileName: The name of the profile to delete the property from

PropName: The name of the property to delete

This subroutine deletes a profile property (setting) value from a specific profile. Note that PropName must be a single property name.

Note that you cannot delete internal SyncBack properties.

See also DeleteProperty

Sub DeleteProperty(PropName)

PropName: The name of the property to delete

This subroutine deletes a profile property (setting) value. Note that PropName must be a single property name.

Note that you cannot delete internal SyncBack properties.

Note that this function will do nothing if called from a Main Interface script. You must use the DeleteProfileProperty function.

3.5.1.11 SBHistory

These are functions that can be accessed from scripts via the **SBHistory** object. They allow you to refer to a profiles history, e.g. when it was run, who ran it, what the result was, etc. The SBHistory.ProfileName property will already have been set for you unless you are using a Main Interface script, in which case you must set it yourself as appropriate. Once the profile name has been set you can get the number of history records available using SBHistory.RecordCount. The number of records available depends on how many times the profile has been run and the maximum history for that profile. Next you need to specify which history record you want to get the values of. You can do this by setting SBHistory.RecordIndex. By default it is set to zero, which is the index of the oldest history record. When you've set the profile name and record index you can then get the history information, e.g. SBHistory.RunResult.

The **SBHistory** object is accessible from Main Interface, Runtime and Profile Configuration scripts.

Property AbortReason

If the profile was aborted then this read-only property returns the reason why it was aborted. Ignore this value if the profile was not aborted. See RunResult for the profiles run result.

Returns -1 on error or if the profile name has not been set.

Property BackupType

This read-only property returns the backup type. If the profile is a SmartSync profile, or was run as a restore, then ignore this value. If the profile is not a Fast Backup profile then it will also return that it was an incremental backup.

Returns -1 on error or if the profile name has not been set.

Property CloudContainer

This read-only property returns the name of the cloud bucket/container that was used. If the cloud was not used then an empty string is returned. For the type of cloud service used see CloudType.

Returns an empty string on error or if the profile name has not been set.

Property CloudType

This read-only property returns the type of cloud service used. If the cloud was not used then it will return 0.

Returns -1 on error or if the profile name has not been set.

Property ComputerName

This read-only property returns the name of the computer used to run the profile. For the name of the user see UserName.

Returns empty string on error or if the profile name has not been set.

Property DestDir

This read-only property returns the destination/right directory used for the profile. If the directory was overridden, e.g. via the command line, then you can check this using IsDestOverride.

Returns empty string on error or if the profile name has not been set.

Property DestSerial

This read-only property returns the destination drives volume serial number. There will not be a serial number if the profile used FTP, email, cloud, etc.

Returns empty string on error or if the profile name has not been set.

Property EmailHostname

This read-only property returns the hostname of the SMTP email server used to backup files to. If the profile is making a backup of emails then it is the POP3/IMAP4 email server. If the profile does not use an email server then an empty string is returned.

Returns empty string on error or if the profile name has not been set.

Property ErrMsg

If the profile failed because of a critical error then this read-only property contains the error message. See also RunResult.

Property FTPHostname

This read-only property returns the hostname of the FTP server used to copy files to and from. If the profile did not use an FTP server then an empty string is returned.

Returns empty string on error or if the profile name has not been set.

Property **GroupName**

If the profile was run as part of a group then this read-only property returns the name of that group. See `GroupStartTime` to get the date & time when the group was started.

Returns empty string on error or if the profile name has not been set.

Property **GroupStartTime**

This read-only property is the date & time when the profiles parent group was started. To get the name of the group see `GroupName`.

Returns 1.0 on error or if the profile name has not been set. It also returns 1.0 if the profile was not run as part of a group.

See also `ProfileStartTime`

IMPORTANT: The date & time is in the GMT/UTC timezone. Use `SBSystem.GMTToLocal` to convert it to your local timezone.

Property **Is64Bit**

This read-only property returns `TRUE` if the version of Windows that ran the profile was 64-bit. To get the version of Windows use `WindowsVersion`.

Returns `FALSE` on error or if the profile name has not been set.

Property **IsDestOverride**

If the destination/right folder was overridden, e.g. via the command line, then this read-only property will return `TRUE`.

Returns `FALSE` on error or if the profile name has not been set.

Property **IsRestore**

If the profile was run as a Restore then this property returns `TRUE`.

Returns `FALSE` on error or if the profile name has not been set.

Property **IsSourceOverride**

If the source/left folder was overridden, e.g. via the command line, then this read-only property will return `TRUE`.

Returns FALSE on error or if the profile name has not been set.

Property ProfileName

This property is used to set and return the name of the profile the history is for. If it has not been set then an empty string is returned. The profile name must be set before you can use any other function or properties in the SBHistory object.

Note that the profile name will have been set automatically unless it is a main interface script.

If you set the profile name to what it already is then nothing will happen.

The history data is cached, so to refresh it you must call Refresh

Property ProfileStartTime

This read-only property is the date & time when the profile started running. It is different from ThreadStartTime because the ProfileStartTime may never be set.

Returns 1.0 on error or if the profile name has not been set. It also returns 1.0 if the profile never started.

See also GroupStartTime

IMPORTANT: The date & time is in the GMT/UTC timezone. Use SBSSystem.GMTToLocal to convert it to your local timezone.

Property ProfileType

This read-only property returns the type of profile.

Returns -1 on error or if the profile name has not been set.

Property RecordCount

This read-only property returns the number of history records there are for the profile. If the profile name has not yet been set (see ProfileName) then -1 is returned. If there is no profile history then 0 is returned.

Records are numbered from 0 (which is the oldest history record) upto RecordCount - 1 (which is the newest history record). Use the RecordIndex property to change the current history record.

Property RecordedInSBM

This read-only property returns TRUE if the history has been recorded in the SyncBack Management Service.

Returns FALSE on error or if the profile name has not been set.

Property RecordIndex

This property is used to set and return the index number of the current history record for the profile.

When a profile name is set (see ProfileName) then current index is set to 0, i.e. the oldest history record for the profile. If the profile name has not been set, or there is no profile history, then -1 is returned.

A record index can range from 0 to RecordCount - 1, i.e. the record list is zero based. If you attempt to set an invalid record index then it is ignored.

Property RunResult

This read-only property is the result of the profile run. See also ErrMsg for any critical error message. If the profile was aborted then see AbortReason for the reason it was aborted.

Returns -1 on error or if the profile name has not been set.

Property SourceDir

This read-only property returns the source/left directory used for the profile. If the directory was overridden, e.g. via the command line, then you can check this using lsSourceOverride .

Returns empty string on error or if the profile name has not been set.

Property SourceSerial

This read-only property returns the source drives volume serial number. There will not be a serial number if the profile used FTP, email, cloud, etc.

Returns empty string on error or if the profile name has not been set.

Property ThreadStartTime

This read-only property is the date & time when the profile was prepared so it would be ready to be run when needed. It is different from ProfileStartTime because the ProfileStartTime may never be set. For example, if a profile is run as part of a group then the profile may never start because a previous profile in the group may be aborted. In that

case the ProfileStartTime is not set, but the ThreadStartTime always is. So the ProfileStartTime is when the profile actually started running (if at all).

IMPORTANT: The date & time is in the GMT/UTC timezone. Use SBSSystem.GMTToLocal to convert it to your local timezone.

Returns 1.0 on error or if the profile name has not been set.

See also GroupStartTime.

Property UserName

This read-only property returns the users Windows login username. For the name of the computer see ComputerName.

Returns empty string on error or if the profile name has not been set.

Property WindowsVersion

This read-only property returns the version of Windows used to run the profile. To see if the version of Windows was 64-bit see Is64Bit.

Returns -1 on error or if the profile name has not been set.

Sub Refresh()

This subroutine refreshes the profiles history. Note that the RecordCount may changed and the RecordIndex is reset to zero.

3.5.1.12 Constants

These are constant values. In the Scripts sub-directory there is a file called **SBCconstants.vbs** which has all these values defined.

Abilities

The following are location abilities. Note that some of these cannot be returned by LocAbilities:

- 1 = Does the location use Windows folders? Cannot be used by LocAbilities().
- 2 = Can the directory attributes and date & times be copied to a new directory? Cannot be

used by LocAbilities().

4 = Can file/folder NTFS security be copied? Cannot be used by LocAbilities().

8 = Can the BackupRead/BackupWrite API's be used? Cannot be used by LocAbilities().

16 = Can use MD5 hashing? Normal only. Cannot be used by LocAbilities().

32 = Do files/folders have Windows filesystem attributes? (CAN_USEATTRIBUTES)

64 = Are file dates & times stored exactly (including milli-seconds)?

(CAN_EXACTDATETIME)

128 = Can file/folder attributes be changed? (CAN_CHANGEATTRIBUTES)

256 = Can file/folder last modification date & times be changed?

(CAN_CHANGEDATETIME)

512 = Are all the files stored in a single Zip file? Cannot be used by LocAbilities().

1024 = Can the base path be empty? (CAN_HAVEEMPTYPATH)

2048 = Is the location on a Windows filesystem? e.g. a drive or UNC path. Cannot be used by LocAbilities().

4096 = Does the location support versioning? (CAN_VERSION)

8192 = Are files stored on a Windows filesystem uncompressed etc? Cannot be used by LocAbilities().

16384 = Files & folders have NTFS extended attributes? (CAN_NTFSATTRIBUTES)

32768 = Accessed via a drive letter? This is different from CAN_STOREDONWINDOWS because the CD/DVD burner uses a drive letter but you cannot read from it. Cannot be used by LocAbilities().

65536 = Write-once, meaning cannot read from it only write to it, e.g. burn location and spanned/split zip (CAN_WRITEONCE)

131072 = Can files be moved/renamed? (CAN_MOVE_FILES)

262144 = Can use CRC32 hashing? (CAN_USECRC32)

524288 = Can folders be moved/renamed? (CAN_MOVE_FOLDERS)

1048576 = It is better to use the case of files on this location

(CAN_PREFERRED_FILECASE)

2097152 = It is better to use the case of folders on this location

(CAN_PREFERRED_FOLDERCASE)

4194304 = Files cannot be copied to this location, but can be deleted from it, e.g. backup of email (CAN_CANNOT_COPYTO)

AbortReason

The following are the reasons why a profile was aborted:

0 = User chose to stop this profile

1 = Profile has run out of time (ELR_TimeLimit)

2 = SyncBack is closing

3 = User wants to stop all the profiles

4 = Profiles are stopping because of Windows shutdown

5 = Stopping because it's part of a group that is being stopped

6 = Aborted by a script

7 = Too many files would be deleted (ELR_TooManyDeletes)

Actions

The following are actions that can be performed on files and folders (note that many of these cannot be used with folders):

- 0 = Skip the file - there was an error (CACTION_ERROR)
- 1 = Do nothing, skip/ignore the file (CACTION_SKIP)
- 2 = Copy from right/destination to the left/source (CACTION_COPY_TOSOURCE)
- 3 = Copy from left/source to the right/destination (CACTION_COPY_TODEST)
- 4 = Delete from left/source (CACTION_DELSOURCE)
- 5 = Delete from right/destination (CACTION_DELDEST)
- 6 = Delete left/source & right/destination (CACTION_DELBOTH)
- 7 = A file is in left/source or right/destination, but not both, prompt the user (CACTION_MISSING_PROMPT)
- 8 = Contents same, but attributes and/or date & time changed, prompt the user (CACTION_DETAILS_PROMPT)
- 9 = The file has changed in both, prompt the user (CACTION_BOTH_PROMPT)
- 10 = Use the file attributes, security, filename case, and/or date & time from the left/source (CACTION_USE_SRC_DETAILS)
- 11 = Use the file attributes, security, filename case, and/or date & time from the right/destination (CACTION_USE_DEST_DETAILS)
- 12 = Move from right/destination to the left/source (CACTION_MOVE_TOSOURCE)
- 13 = Move from left/source to the right/destination (CACTION_MOVE_TODEST)
- 14 = The file is actually unchanged (for Fast Backup only) (CACTION_UNCHANGED)
- 15 = Rename the left/source file (for SmartSync only when the file has been renamed) (CACTION_RENAME_SOURCE)
- 16 = Rename the right/destination file (for SmartSync only when the file has been renamed) (CACTION_RENAME_DEST)

BackupType

The following are profile backup types:

- 0 = Unknown
- 1 = Full
- 2 = Incremental,
- 3 = Differential

CloudType

The following are cloud service types:

- 0 = Unknown
- 1 = Amazon S3 or compatible
- 2 = Microsoft Azure or compatible

Differences

The following are differences. Note that the values can be or'ed together if there is more

than one difference:

- 0 = Skipped because of settings or files are identical (CDIFF_IDENTICAL)
- 1 = Different sizes (CDIFF_SIZE)
- 2 = Different hash values (CDIFF_HASH)
- 4 = Different modification date & time (CDIFF_DATETIME)
- 8 = In right/destination only (CDIFF_DESTONLY)
- 16 = In left/source only (CDIFF_SRONLY)
- 32 = Different attributes (CDIFF_ATTRIB)
- 64 = Only the versions exist (will show in Differences window) (CDIFF_VERSION)
- 128 = File case is different (CDIFF_CASE)

IgnoredReason

The following are reasons why a file or folder is ignored:

- 0 = Unknown (EIRUnknown)
- 1 = Not select in folder & file selectin tree (EIRNotSelected)
- 2 = File/folder does not exist (EIRDoesNotExist)
- 3 = Is a junction point and settings say they are to be ignored (EIRJunctionPoint)
- 4 = A new folder and parent folder settings say new folders are to be ignored (EIRNewFolder)
- 5 = A new file and parent folder settings say new files are to be ignored (EIRNewFile)
- 6 = Filtered out due to filter settings (EIRFiltered)
- 7 = File is not a Zip file (when multi-zip and a non-zip file is found) (EIRNotZipFile)
- 8 = Skipped due to error (EIRError)
- 9 = Wasn't modified within the required time (EIRNotModifiedWithin)
- 10 = File size is too small or too large (EIRSizeOutOfBounds)
- 11 = The source & destination file are the same (EIRIdentical)
- 12 = Many reasons, e.g. in destination but not source (EIRAdvancedSetting)
- 13 = Old file (only in source or dest) not deleted as it's not old enough (EIRNotOldEnoughToDelete)
- 14 = Source cannot be modified, e.g. Fast Backup (EIRCannotModifySource)
- 15 = The destination file is newer so cannot be replace (EIROtherFileNewer)
- 16 = The destination file is read-only so cannot be replace (EIRCannotReplaceReadOnly)
- 17 = The destination file is read-only so cannot be deleted (EIRCannotDelReadOnly)
- 18 = Source is read-only so cannot be copied (EIRCannotMoveCopyReadOnly)
- 19 = Source file does not have archive attribute set so cannot be copied (EIRCannotMoveCopyNotAttrib)
- 20 = Source file has hidden attribute set so cannot be copied (EIRCannotMoveCopyHidden)
- 21 = Source file has system attribute set so cannot be copied (EIRCannotMoveCopySystem)
- 22 = Not copied due to DST time difference (EIRDST)
- 23 = Files same size (EIRSameSize)
- 24 = Ignored by script (EIRScript)
- 25 = Source file has offline attribute set so cannot be copied (EIRCannotMoveCopyOffline)
- 26 = Source file has NTFS encrypted attribute set so cannot be copied (EIRCannotMoveCopyEncrypted)
- 27 = Destination cannot be modified (EIRCannotModifyDestination)

28 = Files same date and time (EIRSameTime)

ProfileTypes

The following are profile types:

- 0 = Error (EPTError)
- 1 = Unknown profile type (EPTUnknown)
- 2 = Custom (EPTCustom)
- 3 = Backup from source to destination (EPTBackupFromSrc)
- 4 = Backup from destination to source (EPTBackupFromDest)
- 5 = Sync (old style) (EPTOldSync)
- 6 = SmartSync (EPTSmartSync)
- 7 = Group (EPTGroup)
- 8 = Mirror from source to destination (EPTMirrorRight)
- 9 = Mirror from destination to source (EPTMirrorLeft)

Results

The following are profile run results:

- 0 = No result (ELR_None)
- 1 = Unknown profile (ELR_UnknownProfile)
- 2 = Profile is already running (ELR_AlreadyRunning)
- 3 = Profile has been imported and not run yet (ELR_Imported)
- 4 = Profile is running (ELR_Running)
- 5 = Internal error (ELR_InternalError)
- 6 = Simulated run was aborted (ELR_SimAborted)
- 7 = Simulated run failed (ELR_SimFailed)
- 8 = Simulated run was a success (ELR_SimSuccess)
- 9 = Restore was aborted (ELR_RestAborted)
- 10 = Restore failed (ELR_RestFailed)
- 11 = Restore was a success (ELR_RestSuccess)
- 12 = Run was aborted (ELR_Aborted)
- 13 = Run failed (ELR_Failed)
- 14 = Run was a success (ELR_Success)
- 15 = Failed due to network connection (ELR_NetFailed)
- 16 = Failed because left or right could not be scanned (ELR_ScanFailed)
- 17 = Failed because the left and right could not be compared (ELR_CompFailed)
- 18 = Run failed because Run Before program stopped profile (ELR_RunBeforeFailed)
- 19 = Run failed because profile is disabled (ELR_Disabled)
- 20 = Run failed because drive errors were found (ELR_SMARTFailed)
- 21 = Run failed because the log could not be emailed (ELR_EmailFailed)
- 22 = Run failed because the Volume Shadow Copy service failed (ELR_SnapshotFailed)
- 23 = Profile was aborted because it reached its run-time limit (ELR_TimeLimit)
- 24 = Profile was aborted because too many files would be deleted (ELR_TooManyDeletes)

Windows

The following are the versions of Windows:

- 0 = Unknown
- 1 = Windows 2000
- 2 = Windows XP
- 3 = Windows 2003
- 4 = Windows Vista
- 5 = Windows 2008
- 6 = Windows 7
- 7 = Windows 2008 R2
- 8 = Windows 8
- 9 = Windows 8 Server

3.5.1.13 Example Scripts

SyncBackPro comes with following example scripts:

- **AllDrives.vbs:** This is a location script which gives access to all the drives on a computer. This means you could use it as your source or destination, set the base directory to \, and then choose files and folders from multiple drives.
- **CompressCopy.vbs:** This is a runtime script which will NTFS compress the copies of files if the original file was also NTFS compressed. This should only be used in profiles that copy to and from NTFS partitions.
- **CreateRestorePoint.vbs:** This is a runtime script which creates a system restore point in Windows whenever the profile is run.
- **DecryptCopy.vbs:** This is a runtime script which will remove the NTFS encryption from the copy of files. For example, you may have a profile that does a backup from one NTFS partition to another, but you may not want the backup copies to be encrypted. This should only be used in profiles that copy to NTFS partitions. See also the **EncryptCopy.vbs** script.
- **EncryptCopy.vbs:** This is a runtime script which will NTFS encrypt the copies of files. For example, you may have a profile that does a backup from one NTFS partition to another, and want the backup copies to be NTFS encrypted. This should only be used in profiles that copy to NTFS partitions. See also the **DecryptCopy.vbs** script.
- **ExtraInfo.vbs:** This is both a runtime and main interface script. It adds a column to the main interface that shows how many files were copied, deleted, etc. in the last profile run. Note that after installing it you must both enable it (as a main interface script), and also configure your profiles to use the script.
- **History.vbs:** This is a main interface script that adds a column to the main interface showing how much history information there is for a profile. The purpose of this example is to demonstrate how

to reduce overhead.

- **IncVar.vbs**: This is a runtime script that shows how to create and use your own variables in a profile, and also how to set when a rescan for a Fast Backup profile should be performed. **Edit the script before using it, or use the IncVarEx.vbs script instead.**
- **IncVarEx.vbs**: This is the same as the **IncVar.vbs** script, except you can configure it from the profile setup window, so there is no need to edit the script file itself.
- **MOTD.vbs**: This is a main interface script that shows a "Message of the day". **Edit the script before using it.**
- **OncePerDay.vbs**: This is a runtime script that shows how to restrict a profile so that it can only be run at most once per day. You may want to use **OncePerDayEx.vbs** instead as it lets you see if the profile has already run today.
- **OncePerDayEx.vbs**: This is the same as the **OncePerDay.vbs** script, except you can also see if the profile has already run from the profile setup window.
- **OnlyYesterday.vbs**: This is a runtime script that ignores any files if the source file has not been modified yesterday. The script is not used if it is a Restore.
- **SBConstants.vbs**: This is simply a definition file defining all the constants.
- **SendResultViaSMS.vbs**: This is a runtime script the sends an SMS if a profile fails. **Edit the script before using it or use the SendResultViaSMSEx.vbs script instead.**
- **SendResultViaSMSEx.vbs**: This is the same as the **SendResultViaSMS.vbs** script, except you can configure it from the profile setup window, so there is no need to edit the script file itself.
- **SendResultViaTwitter.vbs**: This is a runtime script the sends a Twitter message after a profile run. **Edit the script before using it, or use the SendResultViaTwitterEx.vbs script instead. Note: Twitter have since changed their authentication method so this script no longer works.**
- **SendResultViaTwitterEx.vbs**: This is the same as the **SendResultViaTwitter.vbs** script, except you can configure it from the profile setup window, so there is no need to edit the script file itself. **Note: Twitter have since changed their authentication method so this script no longer works.**
- **StripZeros.vbs**: This is a runtime script that shows how to create and use your own variables in a profile. In this example two new variables are created: %NoZeroDay% and %NoZeroMonth%. They are the same as %DAY% and %MONTH% except the leading zero is not included.
- **WaitForFinish.vbs**: This is a runtime script that stops a profile from starting until a program has finished (or isn't running). **Edit the script before using it or use the WaitForFinishEx.vbs script instead.**
- **WaitForFinishEx.vbs**: This is the same as the **WaitForFinish.vbs** script, except you can configure it from the profile setup window, so there is no need to edit the script file itself.
- **TranslateExample.vbs**: This script shows how you can internationalize the messages displayed by your script.

3.5.2 Command Line Parameters

Command Line

A Definition of the Command Line

A Command Line is a space provided directly on the screen where users type specific commands. A CLI (command line interface) is a user interface to a computer's operating system or an application in which the user responds to a visual prompt by typing in a command on a specified line, receives a response back from the system, and then enters another command. The MS-DOS Prompt application in the Windows operating system is an example of the provision of a command line interface. Today, most users prefer the graphical user interface (GUI) offered by Windows or Macs.

SyncBackSE Command Line Parameters

SyncBackSE accepts a number of command line parameters. Some of these are used when SyncBackSE is called from the Windows Task Scheduler. Note that when SyncBackSE is run with command line parameters, by default it will run in **Unattended** mode and be minimized, which means that it will not prompt the user and will not be visible on screen.

- **r**: The profiles following this parameter are run in restore mode. By default profiles are run as backups/Synchronizations.
- **i**: The profiles following this parameter are run in interactive mode. By default profiles run from the command line are run unattended, i.e. no prompts are displayed.
- **m**: The profiles following this parameter are run in minimized mode, i.e. nothing is displayed on the screen. This is the default.
- **n**: The profiles following this parameter are run in normal mode, i.e. dialogs and windows are displayed on the screen. By default profiles run from the command line are run in minimized mode.
- **p**: The profiles following this parameter are run in parallel, i.e. they are all run at the same time. Normally the profiles are run in serial (one after the other).
- **s**: The profiles following this parameter are run in simulated mode, i.e. no files are actually copied or deleted.
- **hibernate**: Place the computer into hibernate mode (if the computer supports it).
- **standby**: Place the computer into standby mode (if the computer supports it).
- **shutdown**: Logoff, shutdown, and switch off the computer (if the computer supports it)
- **monoff**: Switches off all attached display monitors. This is unlikely to work if the profile is

run from a different session, e.g. run via the task scheduler.

- **reboot:** Reboot the computer (if the computer supports it).
- **rebootfreq:** Reboot the computer (if the computer supports it) if a profile was run that required a reboot for a file to be replaced/deleted.
- **full:** Perform a full backup, i.e. rescan the destination for changes. This is only for profiles using the "Fast Backup" setting.
- **countdown [seconds]:** When used this parameter will cause a small window appear with a countdown timer in it. For example, if you passed **-countdown 10** then a window will appear counting down from 10 seconds. Once it reaches zero then the profiles following it in the command line will be run. You can abort the countdown (and exit the program) and so abort running of the profiles by clicking the **Cancel** button. If the user aborts the countdown then the exit code -100 is returned. If you click **OK** then the countdown is cleared and the program continues (and so the profiles are run). A countdown is also useful if you want to shutdown the computer after the profiles have run, but also want to abort the shutdown just in case you are using the computer at the time, e.g. **-countdown 10 -shutdown**. If you use this command line parameter in a scheduled task then make sure the **-m** parameter is not also used. If the task is scheduled in Vista (or newer versions of Windows) then you must enable the checkbox "**Run only when user is logged on**" for the scheduled task otherwise the countdown window will not appear (this is due to the new security in Vista and newer versions of Windows).
- **countdownmsg [message] [seconds]:** This is the same as the **countdown** parameter, except you can define what message you want to appear in the countdown window. If the user aborts the countdown then the exit code -100 is returned. Remember to wrap the message in double-quotes, e.g. **-countdownmsg "Shutdown in 10 seconds" 10 -shutdown**
- **export [profile name] [filename]:** This parameter is used to export a profile, or export all profiles. For example: **-export "My Profile" "C:\Profiles\My Profile.sps"** will export the profile called **My Profile** to the file **C:\Profiles\My Profile.sps**. To export all profiles you must use an asterisk for the profile name and supply a directory instead of a filename, e.g. **-export * "C:\My Profiles"** will export all profiles to the directory C:\My Profiles. The filenames will be the same as the profile names (with the .SPS filename extension). If a profile fails to be exported then the exit code -102 is returned.
- **importprofile [profile filename]:** This parameter allows you to import many profiles by using wildcards. For example, **-importprofile "C:\My Folder*.sps"**, will import all the profiles from the "C:\My Folder" folder. If no directory is given in the filename then the current directory is used. If it fails the exit code -101 is returned.
- **importscript [script filename]:** ( Pro version only) This parameter allows you to import many scripts by using wildcards. For example, **-importscript "C:\My Folder*.sbs"**, will import all the scripts from the "C:\My Folder" folder. If no directory is given in the filename then the current directory is used. **For security reasons scripts are always**

imported interactively and with the users consent. See the note below for more details. If it fails the exit code -108 is returned.

- **source**: Set the source folder for all the following profiles to use. Variables can be used, but please see the important information at the end of the Variables section on how Windows expands environment variables.
- **dest**: Set the destination folder for all the following profiles to use. Variables can be used, but please see the important information at the end of the Variables section on how Windows expands environment variables.
- **nosplash**: The splash screen will not be displayed.
- **delete [profile name]**: The named profile is deleted. If a profile fails to be deleted then the exit code -103 is returned.
- **donotexit**: By default SyncBackSE will automatically exit after performing the tasks given on the command line (that is unless the user interface is used while performing those tasks). To stop it exiting use this parameter.
- **password [password]**: When deleting or importing profiles, an existing profile may exist with that name and be password protected. Use this parameter to provide the password. The password parameter must come before the import filename or **-delete** parameter, e.g. `-password "the password" -delete "profile name"`
- **priority [priority]**: The profiles following this on the command line will be run at this priority. 1 (Idle) is the lowest priority, and 7 (Time Critical) is the highest. The default is 4 (Normal). Note that this is the priority of the thread that runs the profile. It is not the priority of the entire SyncBackSE process (which is the priority that you'll see in the Windows Task Manager). To set the priority of the entire SyncBackSE process using the **procpriority** command line parameter. The SyncBackSE process can only have one priority but each thread can have its own priority.
- **procpriority [priority]**: This sets the priority of the entire SyncBackSE process (which is the priority that you'll see in the Windows Task Manager). 1 (Idle) is the lowest priority, and 5 (High Priority) is the highest. The default is 3 (Normal). Note that the range of values, and default value, are different from that for the **priority** command line parameter. The SyncBackSE process can only have one priority but each thread can have its own priority.
- **affinity [CPU mask]**: This function is only of use on computers that are multi-core or multi-CPU. It ties the SyncBackSE process to one or more specific processors. This can be very useful if you want to limit the CPU resources SyncBackSE can use. The parameter is a bitmap mask (in decimal) with each bit representing a CPU. For example, to force SyncBackSE to only use processor 1 then pass 1, to force SyncBackSE to only use processor 2 then pass 2, to force SyncBackSE to use both processor 1 and 2 then pass 3, to force SyncBackSE to only use processor 3 then pass 4, etc.
- **posreset**: This will reset the positions of all windows.

- **noprofbackup:** The profiles will not be backed up on exit.
- **profbackup:** The profiles will be backed up on exit.
- **clearss:** The SmartSync data will be cleared for all profiles following it on the command line.
- **log [profile name]:** The latest log file for the profile will be displayed. If it is a group profile then the logs for all the profiles in the group will be displayed.
- **sbmssync:** ( Pro version only) Using the defined connection parameters, it will login to the SBM Service, upload profile run history (if required), and download/delete the managed profiles (if required). If it fails the exit code -109 is returned.
- **winpassword [password]:** If you are importing profiles with a schedule then Windows may need your password in order to create the schedule (this is a security requirement of Windows). To avoid being prompted you can pass your Windows login password on the command line. If you do not give a password, and Windows needs your password, and you have not specified the **-i** parameter, then the schedule may silently fail to import.

All other parameters are assumed to be profile names. If a profile name has a space in it use double quotes around the profiles name, e.g. "All Profiles". The profile names are not case sensitive. So for example:

```
SyncBackSE profile1 -i profile2 profile3 -n -r "profile 4" -hibernate
```

This command will run profile1 as a backup/sync in unattended and minimized modes, profile2 and profile3 as a backup/sync in interactive and minimized modes, profile 4 as a restore in interactive and normal mode, and once all the profiles have finished the computer will hibernate.

If SyncBackSE is run with no command line parameters then it will first check to see if any other instances of SyncBackSE are running which also started with no command line parameters. If so, it will not start. This helps ensure only one instance of SyncBackSE is running.

However, whenever SyncBackSE is run with command line parameters, it will run regardless of whether any other instances of SyncBackSE are running or not. Also, if run with command line parameters, SyncBackSE will exit once it has finished its tasks, but if you use any part of its user interface while it is running then it will not exit.

Importing Profiles

You can automatically import profiles by passing the filename of the exported profile on the command line. For example, if you exported a profile and saved it as 'MyProfile.SPS' then if you pass this on the command line to SyncBackSE it will automatically import the profile. The filename extension must be **.SPS** otherwise it is assumed to be the name of a profile to run. If you want to import multiple profiles using wildcards then you can use the **-importprofile** command line parameter. You can also import profiles by dragging the file onto the main window of SyncBackSE.

If a profile has a schedule then when you import it Windows may require you to provide your password (this is a security requirement of Windows). To avoid being prompted you can pass your Windows login password on the command line via the **-winpassword** command line parameter. If you do not give a password, and Windows needs your password, and you have not specified the **-i** parameter (so SyncBackSE cannot prompt you), then the schedule may silently fail to import.

Importing Scripts

You can also automatically import scripts by passing the filename of the exported script on the command line. The filename extension must be **.SBS** otherwise it is assumed to be the name of a profile to run. **For security reasons scripts are always imported interactively and with the users consent.** There is no way to bypass this. If you want to import multiple profiles using wildcards then you can use the **-importscript** command line parameter. You can also import scripts by dragging the file onto the main window of SyncBackSE.

Exit Codes

When running SyncBackSE from the command line, a batch file, etc. then it will return an exit code that gives an indication of whether the task was completed successfully or not. As a number of tasks can be done via the command line at the same time, e.g. import a profile, run it, then delete it, the exit code relates to the last task done on the command line. Also note that if a group profile is run then the exit code is undefined. In the list below, the second column, e.g. 0xFFFFF97, is what the **Windows Task Scheduler** will show in the **Last Result** column.

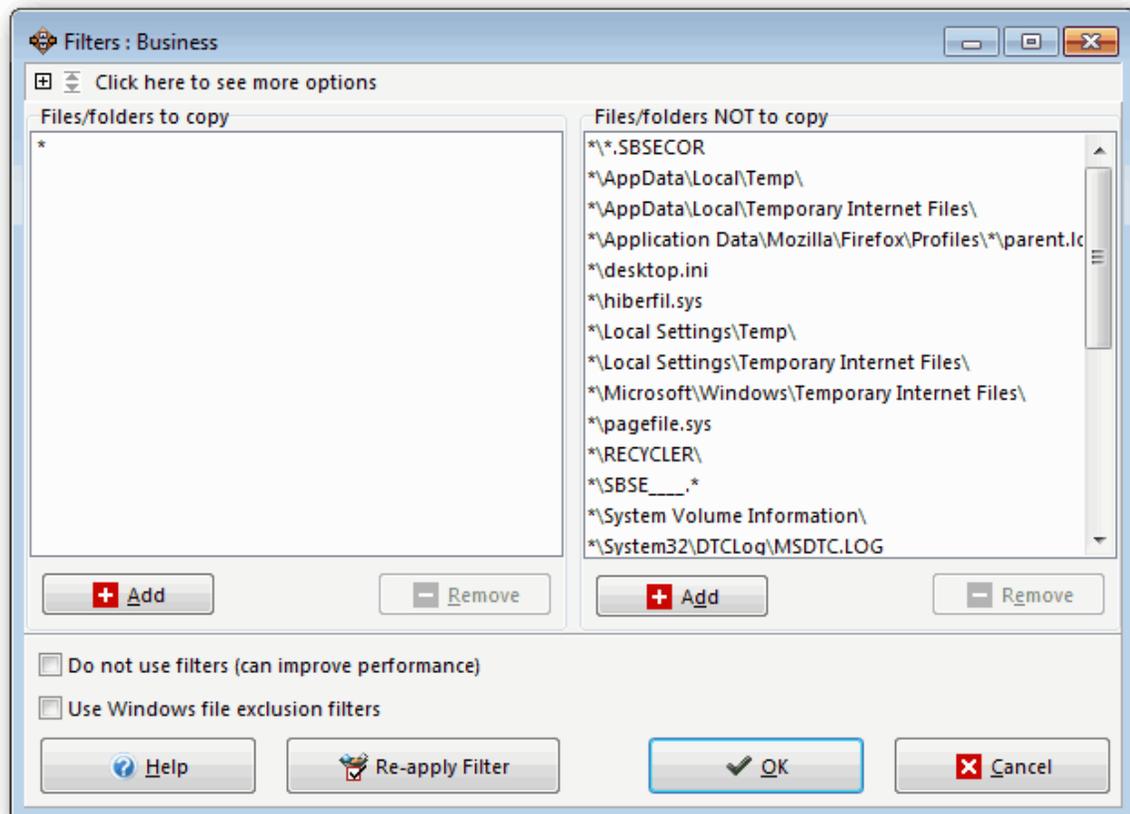
0	Success, no error.
100	SyncBackSE did not close because of user interaction or the donotexit parameter was used
-100	0xFFFF The countdown parameter was used and the user aborted it FFF9C
-101	0xFFFF An attempt was made to import a profile from the command line and it failed FFF9B
-102	0xFFFF The export parameter was used and a profile export failed FFF9A
-103	0xFFFF The delete parameter was used and the profile failed to be deleted FFF99
-104	0xFFFF The user aborted the profile run FFF98
-105	0xFFFF The profile name given does not exist FFF97
-106	0xFFFF The profile was not run because it is disabled FFF96
-107	0xFFFF The profile run failed (note that the result of a group profile run is unknown) FFF95
-108	0xFFFF An attempt was made to import a script from the command line and it failed FFF94
-109	0xFFFF An attempt was made to synchronize with the SBM Service from the command FFF93 line and it failed

-110 0xFF The SyncBackSE serial number is invalid or the evaluation period has expired
FFF92

3.5.3 Filter Settings

Filter Settings

Define what file types and directories are copied and which are not. Note that if you filter out the **desktop.ini** file, and have the option to use the desktop.ini file set, then it will be highlighted in yellow in the filter settings window.

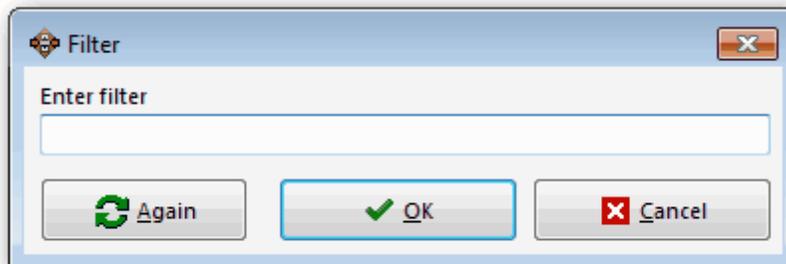


- **Roll-out:** If you click the bar (with the + sign at the left) at the top of the window then a number of options appear.
 - **Style of filter to use:** SyncBackSE supports three different types of filters: **Old V3 style filters**, Regular expressions, and **DOS** expressions. See the help below for details on how to use these. Note that you should not use semi-colons (;) if you are using DOS expressions. This is due to a limitation in the Windows operating system call that is used to compare filenames using filters.

- **Make these the defaults:** If selected then the filters you have set for this profile will become the default filters for all newly created profiles.
- **Add (Files/folders to copy):** If clicked a dialog box appears letting you enter a filter expression for files and folders to include. You can enter a filename, a folder name, a complete path, etc. Anything matching this expression will be copied. You can enter multiple filters at the same time by using the forward slash (/) to separate them, e.g. **item1/item2/item3**. You can also press the **Again** button to immediately enter another filter item.
- **Remove (Files/folders to copy):** If clicked then the selected items will be removed from the list.
- **Do not use filters (can improve performance):** If this checkbox is ticked then no filtering is performed. This can reduce the run time of a profile, sometimes substantially. To further improve performance you may also want to disable file & folder selections.
- **Use Windows file exclusion filters:** If this option is enabled then the standard Windows backup file filters are also used. These are files that Windows itself recommends are not included in any backups. For example, any files in the temporary folder. Note that this option is not available if the profile is using Fast Backup (without the archive attribute).
- **Add (Files/folders NOT to copy):** If clicked a dialog box appears letting you enter a filter expression for files and folders to exclude. You can enter a filename, a folder name, a complete path, etc. Anything matching this expression will be not be copied. You can enter multiple filters at the same time by using the forward slash (/) to separate them, e.g. **item1/item2/item3**. You can also press the **Again** button to immediately enter another filter item.



To edit an existing filter you can double-click on the filter entry or press F2.



- **Remove (Files/folders NOT to copy):** If clicked then the selected items will be removed from the list. You can also double-click on an item in the list to change it.
- **Re-apply Filter:** If clicked, then the filter will be re-applied to the selections you have made in the tree. For example, if you have selected to exclude all files with the extension **.tmp** then if you have selected a file in the tree which does have that extension then you

will be prompted if you'd like to unselect that file. Use the "**Re-apply Filter**" button instead of the **OK** button when you haven't made any changes to the filters but want to re-apply them.

When you make a change to the filters, and click OK to save them, then the new filters will be re-applied to the selections you made in the tree.



Note: to modify an existing filter double-click it or select it and press F2. You can also click on a filter item (in **Files/folders to copy** or **Files/folders NOT to copy**) and press **Ctrl-C** to copy all the filter items to the Windows clipboard. Pressing **Ctrl-A** will select all filter items.



Note: to copy all the filters from another profile, right-click on the **Files/folders to copy** or **Files/folders NOT to copy** box and select "Copy these settings from another profile" from the pop-up menu.



Note: you can use SyncBackSE and Windows Environment variables in the filters, e. g. **%HOMEPATH%**

What are the filters compared with?

A filter is compared with the folders and filenames (which include the path). The root source and destination directories are not used. For example, if your source directory is **C:\my files** and it includes a folder called **subfolder** and a file called **file.txt** then the filters would be compared against the following:

```
\
\subfolder\
\file.txt
```

An important point to remember is that folders include a trailing backslash, but files do not.

Example filters for SyncBackSE

The filters in SyncBackSE allow for files and folders to be filtered based on their name. There are three filter types to choose from. This section gives examples using **DOS Expressions**, which is the default filter type and the simplest to use.

First, there are some important rules to remember about filters:

1. The selections in the tree override the filters. For example, you can filter out all .txt files but still select some .txt files in the tree.
2. Exclude filters override include filters. For example, the include filter may be set to * (which means include everything), and your exclude filter could have *.*.temp in it, which means any file with the extension .temp will be excluded.
3. Filters apply to the entire filename, including the path, but the source and destination base are not part of the path. For example, if your source directory is C:\My Documents\ then that will not be in the filename used with the filter. This makes sense if you remember that the source and destination root directories are different, but their sub-directories are going to have the same names.

4. Folder names end with a backslash, where as files do not, e.g. `\My Documents\` and `\My Documents\filename.txt`
5. All file and folder names start with a backslash.

Remember to include folders and files in the filter

If you set your **Files/folders to copy** filter to just `**.txt` then it will only include text files in the root folder and unselect all child folders. Why? Because you forgot to also include folders. You need to also add `*\` to **Files/folders to copy** to include all the folders (or change that as appropriate to include only folders with certain names, for example).

Another example is if you set your **Files/folders to copy** filter to just `*\` then it will only include folders and no files. Why? Because all folders end with a backslash (`\`) but no files do. If you want all folders and files the filter should be `*` (or `**`)

Filters are applied top-down

When SyncBackSE scans a folder it starts from the top (the root) and works its way down the child folders. For each file and folder it first looks to see if it has been specifically selected, or not, in the file & folder selection tree. Selections override filters. If no selection decision has been made then it uses the filters. It first checks to see if the file or folder matches any inclusion filter. If not, it is filtered out, i.e. skipped (ignored). If it's a folder that is being filtered out then all files and child folders of that folder are ignored. If it matches the filter then it checks to see if it matches an exclusion filter. If it does then it is filtered out.

For example, say you have the following folder structure:

```
\
\Parent\
\Parent\Child\
\Parent\Child\GrandChild\
```

The root folder (`\`) is always included and cannot be filtered out. After scanning the files in the root folder, it would then scan Parent, then Child, then GrandChild. Each folder must match a filter (or have been selected in the tree) otherwise everything in it is ignored. For example, if you had the following filter to copy all text files in the Child folder then it would fail:

```
\Child*.txt
```

Why would it fail? Two reasons. First, because the Parent folder does not match that filter so the Child folder is never looked at. Second, the Child folder doesn't match that filter (it is `\Parent\Child\`). Folders end with a backslash. Therefore the filters must be changed to the following so the folders are included:

```
\Parent\
\Parent\Child\
\Parent\Child*.txt
```

Example DOS Expressions

Notice that many of the examples below also include filters to include folders.

<code>*\</code>	All text files (.txt) in all folders. The <code>*\</code> filter ensures all folders are looked at.
<code>**.txt</code>	

*\	All text files in all folders called temp and any sub-folders of those temp folders.
----	--

\temp\ \temp*.txt	All text files in the root folder called temp . For example, if your source directory is C:\My Documents\ then this filter is for all text files in C\My Documents\temp\
-----------------------	---

*\test\	All folders called test . Note that no files will be copied unless another filter is added to include files, e.g. **.txt
---------	--

*\parent\ \parent\child	All folders called child whose parent directory is called parent . Notice the filter *\ is required otherwise it will never look inside folders called parent. Note that no files will be copied unless another filter is added to include files, e.g. **.txt
----------------------------	--

\temp*\	All root folders whose name starts with temp or is called temp . Note that no files will be copied unless another filter is added to include files, e.g. **.txt
---------	--

Examples of wrong DOS Expressions

The examples below are examples of **wrong** filters. An explanation is given of why it is wrong.

*.txt	This will match any file, or folder, whose name ends with .txt . If you are trying to just include all text files then you should also remember to add *\ to the filters otherwise no child folders will be selected (see the notes below). If you just want text files in the root then the filter is valid if used on its own.
-------	---

temp*.txt	This filter will fail to match anything because all folder and filenames start with a backslash (\) character.
------------	--

3.5.4 Open and Locked File Copying

Locked and Open Files

Open / locked files can only be copied by SyncBackSE when the following criteria are met:

- SyncBackSE is being used on Windows XP or newer, i.e. it is not supported on Windows 2000. Both 32-bit and 64-bit versions of Windows are supported.

- The open / locked file is on a local volume that is formatted with NTFS, or it is on a local volume that is formatted with FAT32 and you also have a local volume on an internal drive that is formatted with NTFS. You can copy locked files to any other drive (external or internal), network drive, Zip file, FTP, etc. The restriction is on where the locked file is, not where it is copied to.
- You are running SyncBackSE with a user who is a member of the Backup Operators group (or an Administrator)

"Local" means a volume on a drive that is physically connected to the computer, i. e. open / locked files cannot be copied from drives over the network.

"Internal" means a drive that is connected via IDE, SCSI, or SATA. Drives connected via USB, Firewire, etc. are considered external. Basically, if the drive is inside your computers case then it is considered internal. Note that the first page of the log file will contain a warning if open / locked files cannot be copied because either the above criteria is not met, or there is another problem.

Possible reasons why an open / locked file cannot be copied:

- Only one profile can copy open / locked files at any one time. If two profiles are running at the same time then only one of them will be able to copy the open / locked files. If you are running profiles in a group then unselect the option to run them in parallel.
- Desktop search programs like Copernicus Desktop Search (CDS), Google Desktop Search (GDS), and MSN Search Toolbar, can interfere with the copying of open / locked files. You may need to close those programs to guarantee that open / locked files can be copied.
- The Volume Shadow Copy Service (VSS) is not installed or working correctly. VSS is a part of Windows and not SyncBackSE. It is used to copy open / locked files. If there is a problem with VSS then the log file will contain the error messages.
- You are not using the latest version of SyncBackSE. You may download the latest version from [our website](#).

3.5.5 Variables

Defining Variables

SyncBackSE includes a whole range of variables that can be used in various profile settings, e.g. the Source and Destination. Variables are values which are not known until the profile is run. At runtime the variables are replaced by their value. Note that you can also define your own profile variables, both at the group level and at the profile level. You can also get values from the registry.

For variables see the following sections on this page:

- Days
- Weeks
- Months
- Years
- Dates
- Times
- Drives, Files, and Folders
- Misc.
- Backup to email
- Backup from email
- Emailing the log
- Registry
- Order of evaluation
- Important notes

Examples of variable usage

Although variables appear to be complex, they are in fact very simple. Just remember that a variables is replaced with its value when the profile is run. A couple of examples will make it clear:

- You are the administrator for a number of employees computers and want to create a backup profile that makes a backup of all the users documents. This profile will be imported on each users computer so you don't need to manually create it on each computer. Each user has their own 'My Documents' folder on a computer, so if you set the source folder to one users folder then it wouldn't work for other users (because they have different usernames, so the path would be different). To avoid this you can simply set the source to **%CSIDL_PERSONAL%**. Now when the profile is run it will replace the **%CSIDL_PERSONAL%** string with the users My Documents path.
- You want to backup to a Zip file and use the current date in the filename of the Zip file. To do this simply set your destination (for example) to **X:\Backup%DATE%.Zip**

Variables are user specific

Remember that the value of a variable may be user-specific. For example, the variable **%CSIDL_PERSONAL%** (see the example above) has a different value for each user (because every users has their own My Documents folder). So if you have a profile set to run as a specific user, e.g. via **Run As** or via the scheduler, then keep in mind that the value returned depends on the user who is running the profile.

Days

The following variables are related to days of the week, month, etc:

%DAY% Current day of the month, e. g. 10

%DAYOFMONTH% Alias for %DAY%

%DAYODDEVEN% Odd or even day of the year (O = odd day, E = even day)

%DAY_P% Yesterdays day of the month (could be previous month)

%DAY_N% Tomorrows day of the month (could be next month)

%DAYOFWEEK% Current day of the week, (1 = Monday, 7 = Sunday)

%DAYOFWEEK_P% Yesterdays day of the week

%DAYOFWEEK_N% Tomorrows day of the week

%DAYOFYEAR% Current day of the year (January 1st = 1)

%DAYOFYEAR_P% Yesterdays day of the year

%DAYOFYEAR_N% Tomorrows day of the year

%DAYOFQUARTER% Current day of the current quarter of the year (January 1st, April 1st, July 1st, October 1st = 1)

%DAYOFQUARTER_P% Yesterdays day of the quarter of the year

%DAYOFQUARTER_N% Tomorrows day of the quarter of the year

%NTHDAYOFWEEK% Note that this value may differ from the value that the **WeekOfTheMonth** variable returns, because NthDayOfWeek counts every occurrence of the given weekday, while WeekOfTheMonth only counts a week if it includes 4 or more days in the month. Thus, for example, if today is a Saturday and is the first day of a month, NthDayOfWeek returns 1, while WeekOfTheMonth returns 5 (or maybe 4), indicating the last week of the previous month.

%DAYSINMONTH% Number of days in current month.

%DAYSINYEAR% Number of days in current year.

These new variables allow you, for example, to keep 7 days worth of backups, e. g. you could set your destination to D:\Backup\%DAYOFWEEK%\ so that you'll always have backups of the last seven days worth of files.

%DAYOFWEEKNAME% The first three letters of the day of the week, e.g. Mon. Note that English is always used.

%DAYOFWEEKNAME_P% The first three letters of yesterday

%DAYOFWEEKNAME_N% The first three letters of tomorrows

%LASTRUNDAY% The day of the month that the profile was last run (empty string if it has not yet been run)

%LASTSUCCESSRUNDAY% The day of the month that the profile was last run without error (empty string if it has not yet been run without error)

Weeks

The following variables are related to weeks of the month, year, etc:

%WEEKOF% Week of the year (1 to 53). WeekOf uses the ISO 8601 standard to define the week of the year. That is, a week is defined as running from Monday through Sunday, and the first week of the year is defined as the one that includes the first Thursday of the year (the first week that includes four or more days in the year). This means that if the first calendar day of the year is a Friday, Saturday, or Sunday, then for the first three, two, or one days of the calendar year, WeekOf returns the last week of the previous year. Similarly, if the last calendar day of the year is a Monday, Tuesday, or Wednesday, then for the last one, two, or three days of the calendar year, WeekOf returns 1 (the first week of the next calendar year).

%WEEKOFODDEVEN% Odd or even week of the year (O = odd week, E = even week)

%WEEKOFTHEMONTH% Week of the month (1 to 6). WeekOfTheMonth uses the ISO 8601 standard definition of a week. That is, a week is considered to start on a Monday and end on a Sunday. The first week of a month is defined as the first week with four or more days in that month. Thus, if the first day of the month is a Friday, Saturday, or Sunday, the first one, two, or three days of the month are defined as belonging to the last week of the previous month. Similarly, if the last day of the month is a Monday, Tuesday, or Wednesday, then the last one, two, or three days of the month are defined as belonging to the first week of the next month.

%WEEKOFQUARTER% Current week of the current quarter of the year (January 1st, April 1st, July 1st, October 1st = 1)

%WEEKOFQUARTER_P% Yesterdays week of the quarter of the year

%WEEKOFQUARTER_P% Tomorrows week of the quarter of the year

%WEEKSINYEAR% The number of weeks in the year (52 or 53). WeeksInYear defines the first week of the year according to the ISO 8601 standard. That is, the first week of the year is the one that includes the first Thursday of the year (the first week that has 4 or more days in the year). This means that WeeksInYear always returns either 52 or 53.

Months

The following variables are related to months of the year:

%MONTH% Current month, e. g. 12

%MONTH_P% Previous month

%MONTH_N% Next month

%MONTH_Y% Yesterdays month

%MONTH_T% Tomorrows month

%MONTHNAME% The first three letters of the current month, e.g. Jan. Note that English is always used.

%MONTHNAME_P% The first three letters of last month

%MONTHNAME_N% The first three letters of next month

%MONTHNAME_Y% The first three letters of yesterdays month

%MONTHNAME_T% The first three letters of tomorrows month

%LASTRUNMONTH% The month that the profile was last run (empty string if it has not yet been run)

%LASTSUCCESSRUNMONTH% The month that the profile was last run without error (empty string if it has not yet been run without error)

%MONTHOFQUARTER% The number of the month of the quarter of the year. For example, January=1, February=2, March=3, April=1, May=2, etc.

%MONTHOFQUARTER_P% The number of the previous month of the quarter of the year. For example, if it is April then this returns 3 (for March), and if it is March then it would return 2 (for February).

%MONTHOFQUARTER_N% The number of the next month of the quarter of the year. For example, if it is March then this returns 1 (for April), and if it is April then it would return 2 (for May).

%MONTHOFQUARTER_Y% The number of yesterdays month of the quarter of the year.

%MONTHOFQUARTER_T% The number of tomorrows month of the quarter of the year.

Years

The following variables are related to years:

%YEAR% Current year in 4 digits, e. g. 2010

%YEAR2% Last two digits of current year, e. g. 09

%YEAR_P% Previous year

%YEAR_N% Next year

%YEAR_Y% Yesterdays year

%YEAR_T% Tomorrows year

%QUARTEROFYEAR% Returns the current quarter for the current year, i.e. 1 for January to March, 2 for April to June, 3 for July to September, and 4 for October to December.

%QUARTEROFYEAR_P% Returns the previous quarter for the current year. For example, if the current quarter is 2 then it returns 1, and if it is 1 then it returns 4.

%QUARTEROFYEAR_N% Returns the next quarter for the current year. For example, if the current quarter is 1 then it returns 2, and if it is 4 then it returns 1.

%QUARTEROFYEAR_Y% Returns yesterdays quarter for the current year.

%QUARTEROFYEAR_T% Returns tomorrows quarter for the current year.

%LASTRUNYEAR% The year that the profile was last run (empty string if it has not yet been run)

%LASTSUCCESSRUNYEAR% The year that the profile was last run without error (empty string if it has not yet been run without error)

Dates

The following variables are related to dates:

%DATE% Current date (it will be in the short date format configured in your installation of Windows)

%DATE_P% Yesterdays date

%DATE_N% Tomorrows date

Times

The following variables are related to times:

%TIME% Current time (it will be in the short time format configured in your installation of Windows)

%HOUR% Current hour (24 hour clock format), e. g. 19

%MINUTE% Current minute

%SECOND% Current second

%MILLISECOND% Current millisecond (0 to 999)

%HOUROFTHEYEAR% The number of complete hours between the current date & time and 12:00 AM on Jan 1 of the year.

%HOUROFTHEMONTH% The number of complete hours between the current date & time and 12:00 AM on the first day of the month.

%HOUROFTHEWEEK% The number of complete hours between the current date & time and 12:00 AM on Monday of the week.

%MINUTEOFTHEYEAR% The number of minutes between the current date & time and 12:00:00:00 AM on Jan 1 of the year.

%MINUTEOFTHEMONTH% The number of minutes between the current date & time and 12:00 AM on the first day of the month.

%MINUTEOFTHEWEEK% The number of minutes between the current date & time and 12:00 AM on Monday of the week (the week starts on Monday).

%MINUTEOFTHEDAY% The number of minutes between the current date & time and 12:00 AM on the same day.

%SECONDOFTHEYEAR% The number of seconds between the current date & time and 12:00:00:00 AM on Jan 1 of the year.

%SECONDOFTHEMONTH% The number of seconds between the current date & time and 12:00:00 AM on the first day of the month.

%SECONDOFTHEWEEK% The number of seconds between the current date & time and 12:00:00 AM on Monday of the week (the week starts on Monday).

%SECONDOFTHEDAY% The number of seconds between the current date & time and 12:00:00 AM on the same day.

%SECONDOFTHEHOUR% The number of seconds between the current date & time and the start of the same hour on the same day.

%MILLISECONDOFTHEYEAR% The number of milliseconds between the current date & time and 12:00:00:00 AM on Jan 1 of the year.

%MILLISECONDOFTHEMONTH% The number of milliseconds between the beginning (Midnight on the first day) of the month and the current date & time.

%MILLISECONDOFTHEWEEK% The number of milliseconds between the current date & time and 12:00:00:00 AM on Monday of the current week (the week starts on Monday).

%MILLISECONDOFTHEDAY% The number of milliseconds between the current date & time and the beginning (midnight) on the same day.

%MILLISECONDOFTHEHOUR% The number of milliseconds between the current time and the start of the same hour on the same day.

%MILLISECONDOFTHEMINUTE% The number of milliseconds between the current time and the start of the same minute on the same day.

Drives, Files, and Folders

The following variables are related to drives, files, and folders:

%THISDRIVE% The drive that SyncBackSE is running on, e. g. C:

%THISPATH% The path that SyncBackSE is running from, e.g. C:\Program Files\2BrightSparks\SyncBackSE

%THISPROGRAM% The path and filename of the SyncBackSE program itself.

%SYNCBACKFOLDER% The **default** local (not roaming) folder that SyncBackSE stores its profiles in, e.g. C:\Documents and Settings\[username]\Local Settings\Application Data\2BrightSparks\SyncBack\ (for Vista and newer it's %LOCALAPPDATA%\SyncBack\). Note that this is not necessarily where the user has decided to store his profiles (you may have configured it to store them in %THISPATH%, for example)

%SYNCBACKBACKUPFOLDER% The **default** (not roaming) folder that SyncBackSE stores its profile backup files in, e.g. C:\Documents and Settings\[username]\Local Settings\Application Data\2BrightSparks\SyncBackSE\Profiles Backup\ (for Vista and newer it's C:\Users\[username]\AppData\Local\2BrightSparks\SyncBackSE\Profiles Backup\). Note that this is not necessarily where the user has decided to store his profile backups.

There are several special variables than can be used to identify drives based on their label or serial number. These are very useful when using external USB drives, for example, which may have a different drive letter each time they are plugged-in:

%LABEL=?% The entry is replaced by the drive letter with that label. For example, if your C drive is labeled My Disk then %LABEL=My Disk%\Documents would be translated into C:\Documents. Note that you can only use one label per string but can use it multiple times. Also, the drives A and B are ignored (use LABELALL to include drives A and B, see below).

%LABELALL=?% This is the same as the LABEL variable except it also includes the drives A and B, which are typically floppy disk drives. When the floppy drive is accessed to get the label of the disk in it may become irritating, which is why you should use LABEL unless you need the label of a disk in drive A or B.

%SERIAL=?% The entry is replaced by the drive letter with that serial number. For example, if your D drive has a serial number of BC46-F69E then %SERIAL=BC46-F69E %Program Files will be translated at runtime into D:\Program Files. Note that you can only use one serial per string but can use it multiple times. Also, the drives A and B are ignored (use SERIALALL to include drives A and B, see below).

%SERIALALL=?% This is the same as the SERIAL variable except it also includes the drives A and B, which are typically floppy disk drives. When the floppy drive is accessed

to get the serial number of the disk in it may become irritating, which is why you should use SERIAL unless you need the label of a disk in drive A or B.

%HWSERIAL=?% The entry is replaced by the drive letter with that hardware serial number. There are important differences between the hardware serial number and the volume (also called a partition) serial number used by the variables **SERIAL** and **DISKSERIAL**. The volume serial number is changed every time a volume is formatted. However, the hardware serial number is stored in the drive hardware itself and never changes. Another important difference is that the hardware serial number is the same for all volumes on a physical drive. For example, if you have a hard drive you could have two volumes on it, e.g. C: and D:. The hardware serial numbers for the volumes C: and D: will be identical because they are stored on the same physical drive. Because of this you should not use a hardware serial number if there is more than one volume on the drive (because which volume is returned is undefined in that case). If the drive is one of two or more connected to a RAID controller then the RAID controller will likely (but not guaranteed) return the serial number of the first available drive in the RAID array. Hardware serial numbers can only be used with **Windows Vista or newer**. Note that you can only use one serial per string but can use it multiple times.

%DISKLABEL% The label of the disk (volume) in the drive.

%DISKSERIAL% The unique serial number of the disk (volume) in the drive.

%DISKHWSERIAL% The hardware serial number of the drive.

%CSIDL_DESKTOP% The virtual folder representing the Windows desktop, the root of the namespace.

%CSIDL_PROGRAMS% The file system directory that contains the user's program groups (which are themselves file system directories). A typical path is C:\Documents and Settings\[username]\Start Menu\Programs.

%CSIDL_PERSONAL% The virtual folder representing the My Documents desktop item. A typical path is C:\Documents and Settings\[username]\My Documents\

%CSIDL_FAVORITES% The file system directory that serves as a common repository for the user's favorite items. A typical path is C:\Documents and Settings\[username]\Favorites.

%CSIDL_STARTUP% The file system directory that corresponds to the user's Startup program group. The system starts these programs whenever any user logs onto Windows. A typical path is C:\Documents and Settings\[username]\Start Menu\Programs\Startup.

%CSIDL_RECENT% The file system directory that contains shortcuts to the user's most recently used documents. A typical path is C:\Documents and Settings\[username]\My Recent Documents.

%CSIDL_SENDTO% The file system directory that contains Send To menu items. A typical path is C:\Documents and Settings\[username]\SendTo.

%CSIDL_STARTMENU% The file system directory containing Start menu items. A typical path is C:\Documents and Settings\[username]\Start Menu.

%CSIDL_DESKTOPDIRECTORY% The file system directory used to physically store file objects on the desktop (not to be confused with the desktop folder itself). A typical path is C:\Documents and Settings\[username]\Desktop.

%CSIDL_NETHOOD% A file system directory containing the link objects that may exist in the My Network Places virtual folder. A typical path is C:\Documents and Settings\[username]\NetHood.

%CSIDL_FONTS% A virtual folder containing fonts. A typical path is C:\Windows\Fonts.

%CSIDL_TEMPLATES% The file system directory that serves as a common repository for document templates. A typical path is C:\Documents and Settings\[username]\Templates.

%CSIDL_COMMON_STARTMENU% The file system directory that contains the programs and folders that appear on the Start menu for all users. A typical path is C:\Documents and Settings\All Users\Start Menu.

%CSIDL_COMMON_PROGRAMS% The file system directory that contains the directories for the common program groups that appear on the Start menu for all users. A typical path is C:\Documents and Settings\All Users\Start Menu\Programs.

%CSIDL_COMMON_STARTUP% The file system directory that contains the programs that appear in the Startup folder for all users. A typical path is C:\Documents and Settings\All Users\Start Menu\Programs\Startup.

%CSIDL_COMMON_DESKTOPDIRECTORY% The file system directory that contains files and folders that appear on the desktop for all users. A typical path is C:\Documents and Settings\All Users\Desktop. Valid only for Windows NT systems.

%CSIDL_APPDATA% The file system directory containing application data for all users. A typical path is C:\Documents and Settings\All Users\Application Data.

%CSIDL_PRINTHOOD% The file system directory that contains the link objects that can exist in the Printers virtual folder. A typical path is C:\Documents and Settings\[username]\PrintHood.

%CSIDL_LOCAL_APPDATA% The file system directory that serves as a data repository for local (non-roaming) applications. A typical path is C:\Documents and Settings\[username]\Local Settings\Application Data.

%CSIDL_ALTSTARTUP% The file system directory that corresponds to the user's non-localized Startup program group. This is usually an empty value.

%CSIDL_COMMON_ALTSTARTUP% The file system directory that corresponds to the non-localized Startup program group for all users. This is usually an empty value.

%CSIDL_COMMON_FAVORITES% The file system directory that serves as a common

repository for favorite items common to all users.

%CSIDL_INTERNET_CACHE% The file system directory that serves as a common repository for temporary Internet files. A typical path is C:\Documents and Settings\[username]\Local Settings\Temporary Internet Files.

%CSIDL_COOKIES% The file system directory that serves as a common repository for Internet cookies. A typical path is C:\Documents and Settings\[username]\Cookies.

%CSIDL_HISTORY% The file system directory that serves as a common repository for Internet history items.

%CSIDL_PROFILE% The user's profile folder. A typical path is C:\Documents and Settings\[username]

%CSIDL_COMMON_MUSIC% The file system directory that serves as a repository for music files common to all users. A typical path is C:\Documents and Settings\All Users\Documents\My Music

%CSIDL_MYMUSIC% The users music files folder.

%CSIDL_COMMON_PICTURES% The file system directory that serves as a repository for image files common to all users. A typical path is C:\Documents and Settings\All Users\Documents\My Pictures

%CSIDL_MYPICTURES% The users image files folder.

%CSIDL_COMMON_VIDEO% The file system directory that serves as a repository for video files common to all users. A typical path is C:\Documents and Settings\All Users\Documents\My Videos

%CSIDL_MYVIDEO% The users video files folder.

%CSIDL_CDBURN_AREA% The file system directory acting as a staging area for files waiting to be written to CD. A typical path is C:\Documents and Settings\username\Local Settings\Application Data\Microsoft\CD Burning

%CSIDL_WINDOWS% The Windows directory. This corresponds to the %windir% or %SYSTEMROOT% environment variables. A typical path is C:\Windows

%CSIDL_SYSTEM% The Windows System folder. A typical path is **C:\Windows\System32** for both 32-bit and 64-bit versions of Windows.

%CSIDL_SYSTEMX86% The 32-bit Windows System folder (even if you are using 64-bit Windows). For 64-bit versions of Windows this may be **C:\Windows\SYSWOW64**

%CSIDL_PROGRAM_FILES% The 32-bit program files folder. A typical path is **C:\Program Files** for 32-bit versions of Windows, and **C:\Program Files (x86)** for 64-bit versions of Windows.

%CSIDL_PROGRAM_FILESX86% The 32-bit program files folder (even if you are using 64-bit Windows).

%CSIDL_PROGRAM_FILES_COMMON% A folder for 32-bit components that are shared across applications. A typical path is **C:\Program Files\Common** for 32-bit versions of Windows, and **C:\Program Files (x86)\Common** for 64-bit versions of Windows.

%CSIDL_PROGRAM_FILES_COMMONX86% The folder for 32-bit components that are shared across applications (even if you are using 64-bit Window).

%CSIDL_COMMON_APPDATA% The file system directory containing application data for all users. A typical path is C:\Documents and Settings\All Users\Application Data

%CSIDL_2BS_APPDATA% This is the same as **%SYNCBACKFOLDER%** and **%CSIDL_2BS_LOCAL_APPDATA%**

%CSIDL_2BS_LOCAL_APPDATA% This is the same as **%SYNCBACKFOLDER%** and **%CSIDL_2BS_APPDATA%**

%CSIDL_2BS_ROAM_APPDATA% The **default** roaming (not local) folder that SyncBackSE would store its profiles. This is not necessarily where the user has decided to store the profiles.

%CSIDL_2BS_APPDATA_PROFILESBACKUP% This is the same as **%SYNCBACKBACKUPFOLDER%** and **%CSIDL_2BS_LOCAL_APPDATA_PROFILESBACKUP%**

%CSIDL_2BS_LOCAL_APPDATA_PROFILESBACKUP% This is the same as **%SYNCBACKBACKUPFOLDER%** and **%CSIDL_2BS_APPDATA_PROFILESBACKUP%**

%CSIDL_2BS_ROAM_APPDATA_PROFILESBACKUP% The **default** roaming (not local) folder that SyncBackSE would store backups of profiles. This is not necessarily where the user has decided to store the profile backups.

Misc.

%SBVERSION% The complete version number of the SyncBackSE program itself, e.g. 5.6.0.13

%PROFILENAME% The profile name. This can be used in the source/left and/or destination/right path.

%GROUPNAME% The group name. This can be used in the source/left and/or destination/right path. If the profile is not being run as part of a group then the value returned is an empty string. See also **%VISUALGROUPNAME%**

Backup to email

Some special variables can be used in the subject and email body when performing a

backup to an email server (Pro version only).

%FILENAME% The filename of the file being backed up

%UPLOADDATETIME% The date & time the file was sent to the email server

%UPLOADDATETIME_GMT% As per **%UPLOADDATETIME%** except the GMT/UTC date & time is used

%MODIFIEDDATETIME% The last modification date & time of the file sent to the email server

%MODIFIEDDATETIME_GMT% As per **%MODIFIEDDATETIME%** except the GMT/UTC date & time is used

%CREATEDDATETIME% The creation date & time of the file sent to the email server

%CREATEDDATETIME_GMT% As per **%CREATEDDATETIME%** except the GMT/UTC date & time is used

%FILEATTRIBS% The attributes of the file sent to the email server

%FILESIZE% The size of the file (in bytes) sent to the email server

%FILEHASH% The hash value of the file sent to the email server

%ISCATALOG% Y if this is a catalog email, else N. A catalog email is an email that contains details on the files that are on the email server.

Backup (from) email

Some special variables can be used in the EML filename and sub-folder when performing a backup of email (Pro version only).

%EMAIL_ID% Unique email message ID. The format is decided by the email server.

%EMAIL_SUBJECT% Email subject.

%EMAIL_SIZE% Size of email in bytes. This is not the size of the EML file.

%EMAIL_DATE% Date email sent. The format used is the short date format set in Windows.

%EMAIL_TIME% Time email sent. The format used is the long time format set in Windows.

%EMAIL_DATEYEAR% Date email sent (year).

%EMAIL_DATEMONTH% Date email sent (month). This is always two digits, e.g. 03 for April.

%EMAIL_DATEDAY% Date email sent (day). This is always two digits, e.g. 05 for the 5th day of the month.

%EMAIL_DATEHOUR% Date email sent (hour) in 24-hour format. This is always two digits, e.g. 09 for 9am.

%EMAIL_DATEMIN% Date email sent (minute). This is always two digits, e.g. 05 for 5 minutes past the hour.

%EMAIL_DATESEC% Date email sent (second). This is always two digits, e.g. 03 for 3 seconds past the minute.

%EMAIL_FROMNAME% From friendly name. If there is no sender's friendly name then the email address is returned (as per **%EMAIL_FROMADDRESS%**).

%EMAIL_FROMADDRESS% From email address.

%EMAIL_REPLYTO% Email address to reply to.

%EMAIL_FIRSTTNAME% To friendly name (of first recipient). Note that an email can be sent to more than one person, so this refers to the first person in the To list. If there is no friendly name then the email address is returned (as per **%EMAIL_FIRSTTOADDRESS%**).

%EMAIL_FIRSTTOADDRESS% To email address (of first recipient). Note that an email can be sent to more than one email address, so this refers to the first email address in the To list.

%EMAIL_IMAPFOLDER% The name of the IMAP4 folder the email is being retrieved from. Note that this will be an empty string if IMAP4 is not being used. The value is modified to ensure it is a valid Windows file/folder name. This means it can be used in the EML filename and sub-folder settings.

Emailing the log and late setting variables

There are a number of special variables that cannot be used in the source or destination settings, for example, because their value is not set until the profile is run (or at some later stage during the profile run). Some variables values are not set until near the end of a profile run. Because of this they can only be used correct in certain settings, e.g. the email body.

%_SOURCE% The source/left path.

%_DESTINATION% The destination/right path.

%LOGFILENAME% Filename of first page of latest log file. This is not set until the log file is closed, so it can only be used in Run After when the profile has been configured to run the 'after' program after the log file has been closed.

%SNAPSOURCE% If a shadow volume is being used to copy locked files from the source/left then this is the path of that shadow volume.

%SNAPDEST% If a shadow volume is being used to copy locked files from the destination/right then this is the path of that shadow volume.

%VISUALGROUPNAME% If the profile is part of a group, and it is run on its own from the main user interface (not necessarily as part of the group), then this is the name of the group. It is different from **%GROUPNAME%** because that value is only set if it is run as part of a group.

%ISFULLBACKUP% If this is a full-backup, i.e. a rescan is being done of the destination, then this is Y, else N

%ISUNATTENDED% If this is an unattended profile run then Y is returned, else N

%ISRESTORE% If this is a restore then Y is returned, else N

%ISSIMULATION% If this is a simulated run then Y is returned, else N

%PROFILEFAILED% If the profile run was a success then 0 is returned, else 1 is returned on error/abort (this is the same variable that can be used in **Run After**).

%ATTACHMENTSTOTAL% The total number of attachments for the email.

%CRITICALERROR% If there was a critical error then this is the error message, otherwise it is an empty string.

%DELETEDTOTAL% The total number of files that were deleted.

%SKIPPEDTOTAL% The total number of files that were skipped, e.g. only in the destination and the profile was configured to ignore files that are only in the destination.

%COPIEDTOTAL% The total number of files that were copied.

%MOVEDTOTAL% The total number of files that were moved.

%DATECHANGEDTOTAL% The total number of files whose last modification date & time were copied.

%ATTRIBCHANGEDTOTAL% The total number of files whose attributes were copied.

%RUNBEFOREERROR% If the Run Before program failed (e.g. because it doesn't exist, or couldn't be started) then this is the error message.

%RUNAFTERERROR% If the Run After program failed (e.g. because it doesn't exist, or couldn't be started) then this is the error message.

%COPYERRORSTOTAL% The total number of file copy/delete errors.

%COMPRESSERRORSTOTAL% The total number of errors related to compression.

%NONCRITICALERRORSTOTAL% The total number of non-critical errors.

%STARTTIME% The date & time the profile was ready to run. If the profile was started as a part of a group then the date & time returned by this variable is not necessarily the date & time the profile actually started (for that see **%STARTTIME2%**). When a group (that is set to run profiles serially and not in parallel) is run SyncBackSE prepares all the profiles in the group so they can be started immediately once the proceeding profile has finished. So the **%STARTTIME%** value is the date & time when the profile was prepared but not necessarily started. To get the date & time when the profile was actually started use the variable **%STARTTIME2%**. If the profile is not part of a group then the time returned by **%STARTTIME%** and **%STARTTIME2%** will be almost identical. To get the date & time when the group was started (if the profile was run as part of a group) use the variable **%GROUPSTARTTIME%**. If you ran a group with just one profile in it then the times returned by **%STARTTIME%**, **%STARTTIME2%**, and **%GROUPSTARTTIME%** would be almost identical.

%STARTTIME2% The date & time the profile actually started to run. See **%STARTTIME%** for an explanation of the differences between that variable and this.

%GROUPSTARTTIME% The date & time when the (root) group started. If the profile is not part of a group then an empty string is returned. You can have groups within groups, so this is the date & time when the root group started. To get the date & time when the profile started see **%STARTTIME%** and **%STARTTIME2%**. See **%STARTTIME%** for an explanation of the differences between the three start time variables.

%ENDTIME% The date & time the profile completed.

%STARTSCANTIME% The date & time the scan for changes started.

%ENDSCANTIME% The date & time the scan for changed ended.

%BYTESCOPIED% The total number of bytes that were copied. See also **%KBYTESCOPIED%** and **%MBYTESCOPIED%**.

%KBYTESCOPIED% The total number of kilobytes that were copied. This is a whole (integer) number that is rounded up or down as appropriate.

%MBYTESCOPIED% The total number of megabytes that were copied. This is a whole (integer) number that is rounded up or down as appropriate.

%BYTESDELETED% The total number of bytes that were deleted. See also **%KBYTESDELETED%** and **%MBYTESDELETED%**.

%KBYTESDELETED% The total number of kilobytes that were deleted. This is a whole (integer) number that is rounded up or down as appropriate.

%MBYTESDELETED% The total number of megabytes that were deleted. This is a whole (integer) number that is rounded up or down as appropriate.

%BYTESREPLACED% The total number of bytes that were replaced/overwritten. See also **%KBYTESREPLACED%** and **%MBYTESREPLACED%**.

%KBYTESREPLACED% The total number of kilobytes that were replaced/overwritten. This is a whole (integer) number that is rounded up or down as appropriate.

%MBYTESREPLACED% The total number of megabytes that were replaced/overwritten. This is a whole (integer) number that is rounded up or down as appropriate.

%COMPARECHANGEDTOTAL% The total number of files that have changed.

%COMPAREHASHCHANGEDTOTAL% The total number of files which have different hash values.

%COMPAREDESTONLYTOTAL% The total number of files in the destination only.

%COMPARESOURCEONLYTOTAL% The total number of files in the source only.

%COMPAREDATETIMETOTAL% The total number of files whose last modification date & time is different.

%COMPARESIZETOTAL% The total number of files whose size is different.

%COMPAREATTRIBTOTAL% The total number of files whose attributes are different.

%COMPARECASETOTAL% The total number of files whose filename case are different, e.g. the source file is called **ABC** and the destination file is called **abc**.

%COMPAREHASHERRORTOTAL% The total number of files whose hash value could not be calculated to compare them.

%TOSKIPCNT% The number of files that are going to be skipped.

%TOPROMPTCNT% The number of files where the user will be prompted on the action to take.

%TODELETESRCNT% The number of files that are to be deleted from the source/left.

%TODELETEDESTCNT% The number of files that are to be deleted from the destination/right.

%TOCOPYTODESTCNT% The number of files to be copied to the destination/right.

%TOCOPYTOSRCNT% The number of files to be copied to the source/left.

%TOMOVETODESTCNT% The number of files to be moved to the destination/right.

%TOMOVETOSRCNT% The number of files to be moved to the source/left.

%TOCHANGESRCATTRIBSCNT% The number of files in the source/left that will have their attributes/date & time changed.

%TOCHANGEDESTATTRIBSCNT% The number of files in the destination/right that will have their attributes/date & time changed.

%TORENAMESRCCNT% The number of files in the source/left that will be renamed.

%TORENAMEDESTCNT% The number of files in the destination/right that will be renamed.

%BYTESCOPYTOSRC% The total number of bytes to be copied (includes moved files) to the source/left.

%KBYTESCOPYTOSRC% The total number of kilobytes to be copied (includes moved files) to the source/left. This is a whole (integer) number that is rounded up or down as appropriate.

%MBYTESCOPYTOSRC% The total number of megabytes to be copied (includes moved files) to the source/left. This is a whole (integer) number that is rounded up or down as appropriate.

%BYTESCOPYTODEST% The total number of bytes to be copied (includes moved files) to the destination/right.

%KBYTESCOPYTODEST% The total number of kilobytes to be copied (includes moved files) to the destination/right. This is a whole (integer) number that is rounded up or down as appropriate.

%MBYTESCOPYTODEST% The total number of megabytes to be copied (includes moved files) to the destination/right. This is a whole (integer) number that is rounded up or down as appropriate.

%BYTESDELETFROMSRC% The total number of bytes to be deleted (includes moved files) from the source/left.

%KBYTESDELETFROMSRC% The total number of kilobytes to be deleted (includes moved files) from the source/left. This is a whole (integer) number that is rounded up or down as appropriate.

%MBYTESDELETFROMSRC% The total number of megabytes to be deleted (includes moved files) from the source/left. This is a whole (integer) number that is rounded up or down as appropriate.

%BYTESDELETFROMDEST% The total number of bytes to be deleted (includes moved files) from the destination/right.

%KBYTESDELETFROMDEST% The total number of bytes to be deleted (includes moved files) from the destination/right. This is a whole (integer) number that is rounded up or down as appropriate.

%MBYTESDELETEFROMDEST% The total number of bytes to be deleted (includes moved files) from the destination/right. This is a whole (integer) number that is rounded up or down as appropriate.

%PAGE% Used only in log filename.

%COMPAREDIRSCHANGEDTOTAL% The number of directories that have been changed.

%COMPAREDIRSDESTONLYTOTAL% The number of directories that are only in the destination.

%COMPAREDIRSSOURCEONLYTOTAL% The number of directories that are only in the source.

%FTPCONNECTCNT% The number of (re)connections made to the FTP server

%VERSIONSRESTOREDOTAL% The total number of versions restored

%COMPAREUNCHANGEDTOTAL% The total number of unchanged files

%TORESTOREVERSRCNT% The total number of versions files to be restored on the source/left

%TORESTOREVERDESTCNT% The total number of versions files to be restored on the destination/right

%FREEBYTESSOURCEBEFORE% The number of free bytes on the source/left before the profile started copying, moving, and deleting files

%FREEKBYTESSOURCEBEFORE% The number of free kilobytes on the source/left before the profile started copying, moving, and deleting files. This is a whole (integer) number that is rounded up or down as appropriate.

%FREEMBYTESSOURCEBEFORE% The number of free megabytes on the source/left before the profile started copying, moving, and deleting files. This is a whole (integer) number that is rounded up or down as appropriate.

%FREEBYTESDESTBEFORE% The number of free bytes on the destination/right before the profile started copying, moving, and deleting files

%FREEKBYTESDESTBEFORE% The number of free kilobytes on the destination/right before the profile started copying, moving, and deleting files. This is a whole (integer) number that is rounded up or down as appropriate.

%FREEMBYTESDESTBEFORE% The number of free megabytes on the destination/right before the profile started copying, moving, and deleting files. This is a whole (integer) number that is rounded up or down as appropriate.

Registry

As well as variables, you can also get values from the registry. For example, the following will retrieve the current version of Firefox that is installed:

```
%@HKEY_LOCAL_MACHINE\SOFTWARE\Mozilla\Mozilla Firefox\CurrentVersion  
%
```

To get values from the registry you must use **%@** followed by one of the following (these define which part of the registry to read):

```
HKEY_CLASSES_ROOT  
HKEY_CURRENT_USER  
HKEY_LOCAL_MACHINE  
HKEY_USERS  
HKEY_PERFORMANCE_DATA  
HKEY_CURRENT_CONFIG  
HKEY_DYN_DATA
```

Then specify the path in the registry, e.g. `\SOFTWARE\Mozilla\Mozilla Firefox\CurrentVersion`, and finally finish with a single percentage sign (%). If there is no such value in the registry then the variable is not expanded.

If you are using a 64-bit version of Windows then note that because SyncBackSE is a 32-bit program it will read from the 32-bit registry by default. To read from the 64-bit registry you must use **%64@**, e.g.

```
%64@HKEY_LOCAL_MACHINE\SOFTWARE\Mozilla\Mozilla  
Firefox\CurrentVersion%
```

If you are not using a 64-bit version of Windows, and try to read from the 64-bit registry, then it will instead try to get the value from the 32-bit registry (because there is no 64-bit registry on 32-bit versions of Windows). Because of this it is recommended that you always use **%64@** instead of **%@** because it will work correctly on both 64-bit and 32-bit versions of Windows.

Order of evaluation

Variables are evaluated in the following order:

1. Registry variables
2. Windows environment variables
3. User defined (profile and group) variables and run-time variables, e.g. `%PROFILENAME%`
4. Group variables
5. SyncBack variables

If a user defined profile variable has the same name as a user defined group variable then the profile variable replaces the group variable.

Important note about Variable usage

An important point to remember is that Windows has its own environment variables, e.g. %USERNAME%. When these variables are used in a batch file, or on the command line, then Windows automatically expands them. Unknown variables are simply deleted. For example, if you had the following batch file:

```
@echo off
"c:\program files\2brightsparks\SyncBackSE\SyncBackSE.exe" -source "x:\
%DAY%"
```

Then when run it would actually be expanded to do the following:

```
@echo off
"c:\program files\2brightsparks\SyncBackSE\SyncBackSE.exe" -source "x:\
\"
```

Note that the %DAY% has been removed because it's an unknown Windows variable (it's a SyncBackSE variable). To stop Windows from changing SyncBackSE variables you must use two percentage signs, e.g.

```
@echo off
"c:\program files\2brightsparks\SyncBackSE\SyncBackSE.exe" -source "x:\
%%DAY%%\"
```

3.5.6 Regular Expressions

Regular Expression Filters in SyncBackSE

This section of the help file provides information and guidance about regular expression filters. Regular expressions are a system for matching patterns in text data. They provide a powerful set of tools for finding particular words or combinations of characters in strings.

Note that by default SyncBackSE will be case insensitive with the filters and it is not recommended that you use case sensitivity (via the modifiers). You also need to keep in mind that regular expressions can match any part of a filename, unlike DOS expressions which must match the entire filename. SyncBackSE works with line separators as recommended at www.unicode.org, however there are no line separators within a filename. To add flexibility, SyncBackSE adds the backslash character (\) as a line separator. This means that filenames are essentially broken down into their parts with each part being treated as a separate line. See the Line Separator section below on how this is useful.

Important: 2BrightSparks cannot provide technical support for helping you create regular expressions.

Simple matches

Any single character matches itself, unless it is a meta-character with a special meaning described below.

A series of characters matches that series of characters in the target string, so the pattern **blah** would match **blah** in the target string.

You can cause characters that normally function as meta-characters or escape sequences to be interpreted literally by 'escaping' them by preceding them with a backslash (\), for instance: meta-character **^** match beginning of string, but **^** match character **^**, **** match **** and so on.

Examples:

foobar matches string **foobar**
^FooBarPtr matches **^FooBarPtr**

Escape sequences

Characters may be specified using escape sequences syntax much like that used in C and Perl: **\n** matches a newline, **\t** a tab, etc. More generally, **\xnn**, where **nn** is a string of hexadecimal digits, matches the character whose ASCII value is **nn**. If you need a Unicode character code, you can use **\x{nnnn}** where **nnnn** is one or more hexadecimal digits.

\xnn	character with hex code nn
\x{nnnn}	character with hex code nnnn (one byte for plain text and two bytes for Unicode)
\t	tab (HT/TAB), same as \x09
\n	newline (NL), same as \x0a
\r	carriage return (CR), same as \x0d
\f	form feed (FF), same as \x0c
\a	alarm (bell) (BEL), same as \x07
\e	escape (ESC), same as \x1b

Examples:

foo\x20bar matches **foo bar** (note space in the middle)
\tfoobar matches **foobar** predefined by tab

Character classes

You can specify a character class, by enclosing a list of characters in square brackets (**[]**), which will match any one character from the list.

If the first character after the opening square bracket **[** is **^**, the class matches any character not in the list.

Examples:

foob[aeiou]r finds strings **foobar**, **foober**, etc. but not **foobbr**, **foobcr**, etc.
foob[^aeiou]r finds strings **foobbr**, **foobcr**, etc. but not **foobar**, **foober**, etc.

Within a list, the dash/minus character - is used to specify a range, so that **a-z** represents all characters between **a** and **z**, inclusive.

If you want - itself to be a member of a class, put it at the start or end of the list, or escape it with a backslash. If you want a closing square bracket **]** then you may place it at the start of list or escape it with a backslash.

Examples:

[-az]	matches a , z and -
[az-]	matches a , z and -
[a\ -z]	matches a , z and -
[a-z]	matches all twenty six small characters from a to z
[\n-\x0D]	matches any of #10 , #11 , #12 , #13
[\d-t]	matches any digit, - or t
[]-a]	matches any character from] to a

Meta-characters

Meta-characters are special characters which are the essence of Regular Expressions. There are different types of meta-characters, described below.

Meta-characters - line separators

^	start of line
\$	end of line
\A	start of text
\Z	end of text
.	any character in line

Examples:

^foobar	matches string foobar only if it's at the beginning of line
foobar\$	matches string foobar only if it's at the end of line
^foobar\$	matches string foobar only if it's the only string in line
foob.r	matches strings like foobar , foobbr , foob1r and so on

The **^** meta-character by default is only guaranteed to match at the beginning of the input string/text, the **\$** meta-character only at the end. Embedded line separators will not be matched by **^** or **\$**.

You may, however, wish to treat a string as a multi-line buffer, such that the **^** will match after any line separator within the string, and **\$** will match before any line separator. You can do this by switching on the modifier **m**.

The **\A** and **\Z** are just like **^** and **\$**, except that they won't match multiple times when the

modifier **m** is used, while **^** and **\$** will match at every internal line separator.

The **.** meta-character by default matches any character, but if you switch off the modifier **s**, then **.** won't match embedded line separators.

^ is at the beginning of a input string, and, if modifier **m** is on, also immediately following any occurrence of ****, **\x0D\x0A**, **\x0A**, **\x0D**, **\x2028**, **\x2029**, **\x0B**, **\x0C**, or **\x85**. Note that there is no empty line within the sequence **\x0D\x0A**.

\$ is at the end of a input string, and, if modifier **m** is on, also immediately preceding any occurrence of ****, **\x0D\x0A**, **\x0A**, **\x0D**, **\x2028**, **\x2029**, **\x0B**, **\x0C**, or **\x85**. Note that there is no empty line within the sequence **\x0D\x0A**.

. matches any character, but if you switch off modifier **s** then **.** doesn't match ****, **\x0D\x0A**, **\x0A**, **\x0D**, **\x2028**, **\x2029**, **\x0B**, **\x0C**, or **\x85**.

Note that **^.*\$** (an empty line pattern) does not match the empty string within the sequence **\x0D\x0A**, but matches the empty string within the sequence **\x0A\x0D**.

Meta-characters - predefined classes

\w	an alphanumeric character (including underscore _)
\W	a non-alphanumeric
\d	a numeric character
\D	a non-numeric
\s	any space (same as [\t\n\r\f])
\S	a non space

You may use **\w**, **\d** and **\s** within custom character classes.

Examples:

foob\d r matches strings like **foob1r**, **foob6r** and so on but not **foobar**, **foobbr** and so on
foob[\ws]r matches strings like **foobar**, **foob r**, **foobbr** and so on but not **foob1r**, **foob=r** and so on

Meta-characters - word boundaries

\b	Match a word boundary
\B	Match a non-(word boundary)

A word boundary (**\b**) is a spot between two characters that has a **\w** on one side of it and a **\W** on the other side of it (in either order), counting the imaginary characters off the beginning and end of the string as matching a **\W**.

Meta-characters - iterators

Any item of a regular expression may be followed by another type of meta-characters - iterators. Using these meta-characters you can specify number of occurrences of previous

characters, meta-characters or sub-expressions.

*	zero or more ("greedy"), similar to {0,}
+	one or more ("greedy"), similar to {1,}
?	zero or one ("greedy"), similar to {0,1}
{n}	exactly n times ("greedy")
{n,}	at least n times ("greedy")
{n,m}	at least n but not more than m times ("greedy")
*?	zero or more ("non-greedy"), similar to {0,}?
+?	one or more ("non-greedy"), similar to {1,}?
??	zero or one ("non-greedy"), similar to {0,1}?
{n}? 	exactly n times ("non-greedy")
{n,}? 	at least n times ("non-greedy")
{n,m}? 	at least n but not more than m times ("non-greedy")

So, digits in curly brackets of the form **{n,m}** specify the minimum number of times to match the item **n** and the maximum **m**. The form **{n}** is equivalent to **{n,n}** and matches exactly n times. The form **{n,}** matches n or more times. There is no limit to the size of **n** or **m**, but large numbers will chew up more memory and slow down execution.

If a curly bracket occurs in any other context, it is treated as a regular character.

Examples:

foob.*r	matches strings like foobar , foobalkjdfllkj9r and foobr
foob.+r	matches strings like foobar , foobalkjdfllkj9r but not foobr
foob.?r	matches strings like foobar , foobbr and foobr but not foobalkj9r
fooba{2}r	matches the string foobaar
fooba{2,}r	matches strings like foobaar , foobaaar , foobaaaar etc.
fooba{2,3}r	matches strings like foobaar , or foobaaar but not foobaaaar

A little explanation about *greediness*. "Greedy" takes as many as possible, "non-greedy" takes as few as possible. For example, **b+** and **b*** applied to string **abbbbc** return **bbbb**, **b+?** returns **b**, **b*?** returns empty string, **b{2,3}?** returns **bb**, **b{2,3}** returns **bbb**.

You can switch all iterators into "non-greedy" mode (see the modifier **g**).

Meta-characters - alternatives

You can specify a series of alternatives for a pattern using **|** to separate them, so that **fee|fie|foe** will match any of **fee**, **fie**, or **foe** in the target string (as would **f(e|i|o)e**). The first alternative includes everything from the last pattern delimiter (**(**, **[**, or the beginning of the pattern) up to the first **|**, and the last alternative contains everything from the last **|** to the next pattern delimiter. For this reason, it's common practice to include alternatives in parentheses, to minimize confusion about where they start and end.

Alternatives are tried from left to right, so the first alternative found for which the entire expression matches, is the one that is chosen. This means that alternatives are not necessarily greedy. For example: when matching **fool|foot** against **barefoot**, only the **foo**

part will match, as that is the first alternative tried, and it successfully matches the target string. (This might not seem important, but it is important when you are capturing matched text using parentheses.)

Also remember that `|` is interpreted as a literal within square brackets, so if you write `[fee]fie[foe]` you're really only matching `[feio]`.

Examples:

`foo(bar|foo)` matches strings **foobar** or **foofoo**

Meta-characters - sub-expressions

The bracketing construct (...) may also be used for defining sub-expressions. Sub-expressions are numbered based on the left to right order of their opening parenthesis. First sub-expression has number '1'.

Examples:

`(foobar){8,10}` matches strings which contain 8, 9 or 10 instances of the **foobar**
`foob([0-9][a+])r` matches **foob0r**, **foob1r**, **foobar**, **foobaar**, **foobaar** etc.

Meta-characters - back-references

Meta-characters `\1` through `\9` are interpreted as back-references. `\<n>` matches previously matched sub-expression `#<n>`.

Examples:

`(.)\1+` matches **aaaa** and **cc**
`(+)\1+` also match **abab** and **123123**
`(["']?)(\d+)\1` matches **"13"** (in double quotes), or **'4'** (in single quotes) or **77** (without quotes) etc

Modifiers

Modifiers are for changing behaviour of the regular expression engine. Any of these modifiers may be embedded within the regular expression itself using the `(?...)` construct. If the construction is in-lined into a sub-expression then it affects only that sub-expression.

i By default this is on. Do case-insensitive pattern matching (using installed in you system locale settings). **SyncBackSE uses case insensitive searches by default and it is not recommended that you use case sensitivity.**

m By default this is off. Treat string as multiple lines. That is, change `^` and `$` from matching at only the very start or end of the string to the start or end of any line anywhere within the string. This is important because in SyncBackSE a backslash is treated as a line separator. See the Line Separator section.

s By default this is on. Treat string as single line. That is, change `.` to match any

character whatsoever, even a line separators, which it normally would not match. This is important because in SyncBackSE a backslash is treated as a line separator. See the Line Separator section.

g Non standard modifier. Switching it **off** will switch all following operators into non-greedy mode (by default this modifier is **on**). So, if modifier **g** is off then **+** works as **+**?, ***** as ***?** and so on. By default this is on.

Examples:

(?i)Saint-Petersburg matches **Saint-petersburg** and **Saint-Petersburg**
(?i)Saint-(?-i)Petersburg matches **Saint-Petersburg** but not **Saint-petersburg**
(?i)(Saint-)?Petersburg matches **Saint-petersburg** and **saint-petersburg**
((?i)Saint-)?Petersburg matches **saint-Petersburg**, but not **saint-petersburg**

(?#text)

A comment, the text is ignored. Note that comment is closed at the first close bracket **)**, so there is no way to put a literal close bracket **)** in the comment.

Line Separator

SyncBackSE treats the backslash character as a line separator. All filenames start with a backslash, and all folders end with a backslash. The backslash character also delineates the parts of a file. By treating the backslash character as a line separator you can change how the **.** meta-character works and so have more flexibility. For example, let's say you only want text files in any folder that start with the name **temp**. A first attempt would be:

```
\\$
.*\\temp.*\\.*.txt
```

The first one (**\\\$**) makes sure all folders are scanned, which is the same as the DOS expression *****. Although the second expression looks correct, it won't work correctly. It would match **\\temp\\folder\\test.txt**. A second try could be:

```
\\temp.*?\\[^\\]*\\.txt$
```

But this would also match **\\temp\\folder\\test.txt**. Why? The part **\\temp.*?** will correctly match **\\temp** but **[^\\]*\\.txt\$** will match **test.txt** no matter what is after **\\temp**. This is where the point about the backslash being a line separator is important. Because it's a line separator you can change the way the meta-character **.** works. Normally it will match any character at all. But using the **s** modifier you can stop it matching line separators, so the following will work:

```
(?-s)\\temp.*\\.*.txt$
```

To explain why this would work, let's use the example filename of **\\temp\\folder\\text.txt** and see why it would not match:

(?-s) is a modifier that tells SyncBackSE to treat the filename as separate lines. Because backslash is a line separator it means you can think of the filename as being broken up into its parts with each part effectively on its own line:

```
temp
folder
text.txt
```

\\temp.* matches `\\temp\\`, so we are now onto the next line/part (folder)

.*\\.txt\$ means the end of the current line must match any number of characters (but not backslash) and end with `.txt`. You could also have **.*\\.txt\\Z**

If the expression was **(?-s)\\temp.*\\..*\\.txt** (so it doesn't have \$ at the end) then it would wrongly match `\\temp\\folder.txt\\test.txt` because `folder.txt` matches **.*\\.txt**

What if you only wanted root temp* folders? The expression would be:

```
(?-s)\\A\\temp.*\\..*\\.txt$
```

The meta-character **\\A** ensures that `\\temp\\` must be at the beginning.

SyncBackSE Examples

Notice that many of the examples below also include filters to include folders.

<code>\\\$ \\.txt\$</code>	All text files (<code>.txt</code>) in all folders. The <code>\\\$</code> filter ensures all folders are looked at.
--------------------------------	--

<code>\\\$ (?-s)\\temp\\..* txt\$</code>	All text files in all folders called temp . It would not include <code>.txt</code> files in any sub-folders of folders called temp . To include all <code>.txt</code> files in all sub-folders of folders called <code>temp</code> you would omit the <code>(?-s)</code>
--	--

<code>\\temp\\\$ (?-s)\\temp\\..* txt\$</code>	All text files in the root folder called temp . For example, if your source directory is <code>C:\\My Documents\\</code> then this filter is for all text files in <code>C:\\My Documents\\temp\\</code>
--	---

<code>.*\\test\\\$</code>	All folders called test . Note that no files will be copied unless another filter is added to include files.
---------------------------	---

<code>.*\\parent\\\$.*\\parent\\ \\child\\\$</code>	All folders called child whose parent directory is called parent . Notice the filter <code>.*\\</code> is required otherwise it will never look inside folders called <code>parent</code> . Note that no files will be copied unless another filter is added to include files.
--	--

<code>(?-s)\\A\\temp. *?\\\$</code>	All root folders whose name starts with temp or is called temp . Note that no files will be copied unless another filter is added to include files. If you omit
---	---

(?-s) then it would wrongly also match anything inside \temp\

3.5.7 Uninstalling SyncBackSE

How to Uninstall SyncBackSE

If you used the default installation procedure for SyncBackSE you may uninstall the program by going to **Windows Start > All Programs > 2BrightSparks > SyncBackSE > Uninstall SyncBackSE**.

The exception to this is if you installed the program in a custom location, in which case you will have to find out where SyncBackSE is located. You can easily find this using the Windows search facility.

Please be aware that uninstalling SyncBackSE will delete your profiles.

All versions of SyncBackSE can safely be installed over an existing installation. By doing this, you will ensure any profiles you have created continue to be active.

Lastly, make sure that SyncBackSE is not running while you attempt to uninstall. Remember that SyncBackSE can be seen in the task bar on the lower right when active (you may need to click the little left pointing arrow to see the SyncBackSE icon). You may have also created a scheduled task which you will need to disable if uninstalling.

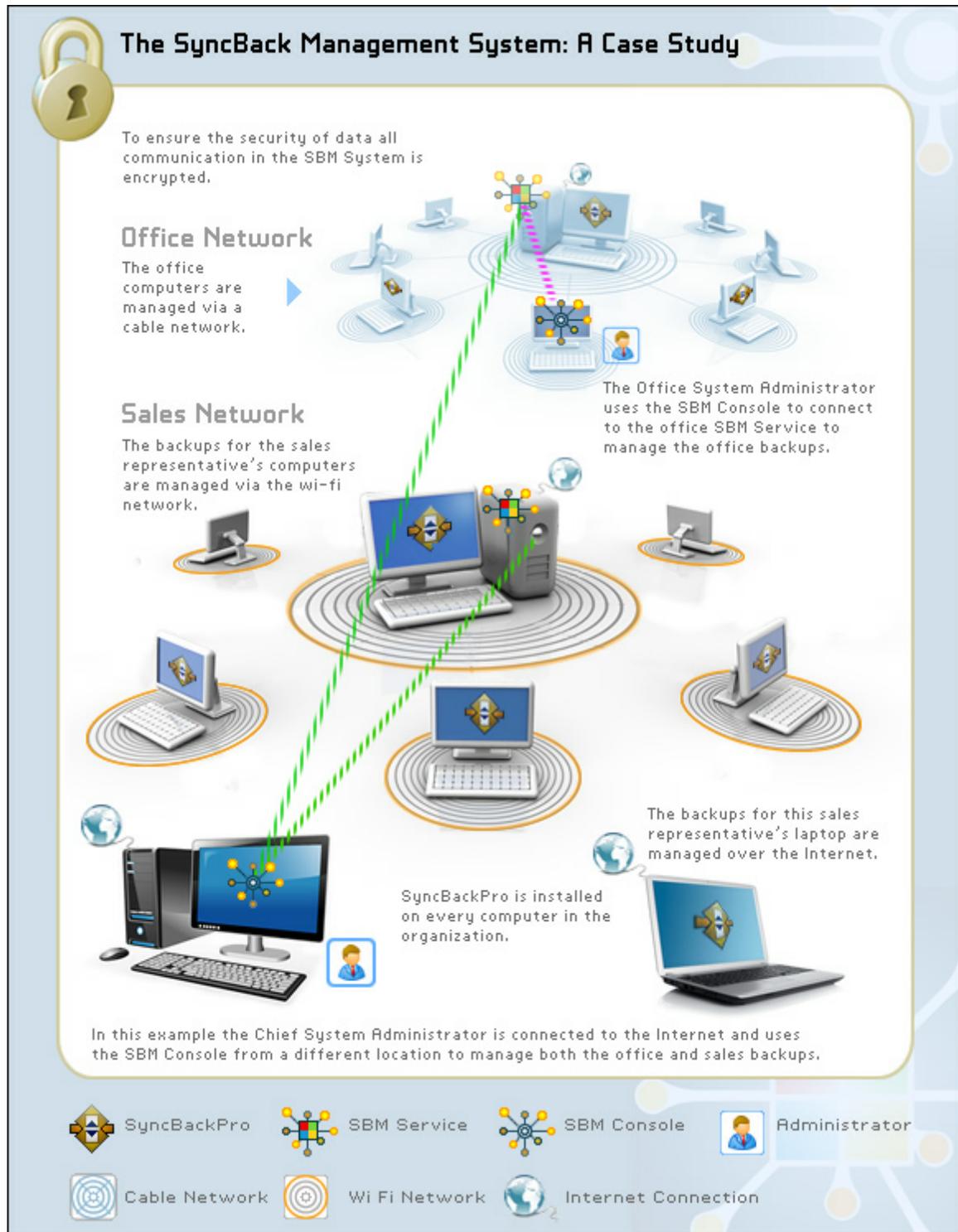
3.5.8 SyncBack Management Service

The SyncBack Management System

Remote installations of SyncBackPro ( Pro version only, not available with SyncBackSE) can now be managed and monitored from a central location. A server, the SyncBack Management Service (**SBM Service**), must be installed on a Windows server. You can then configure your installations of SyncBackPro to communicate with your SBM Service so that they can report on when profiles are run and also update their profiles. The communication between SyncBackPro and the SBM Service is done over the network and is encrypted. This means an installation of SyncBackPro can communicate with your SBM Service over an intranet or the Internet.

Configuration and management of the SBM Service is done via the SyncBack Management Console (**SBM Console**). The console can be used from any computer as it communicates with the SBM Service in the same way as SyncBackPro, i.e. over a network connection. The console allows you to view the profile run history of remote SyncBackPro installations, for example. For detailed information on how to use the console please refer to

the console help file. The SBM Console is free software which can be downloaded from the 2BrightSparks web site.



Management Service Settings

To set the connection details for SyncBackSE to connect to the SBM Service select **Preferences -> Management Service Settings** from the main menu.

- **Hostname:** This is the hostname, or URL, of the SBM Service that you want to connect to. If you are connecting via HTTP then the URL is in the format http://hostname:port/BIN/, e.g. http://myserver.com:8099/BIN/. If you are using a TCP connection then enter just the hostname or IP address. By default the server uses a HTTP connection.
- **Port:** A port number is only required when connecting via TCP/IP, i.e. you are not using a HTTP connection. When using a HTTP connection the port number is embedded in the URL.
- **Username:** The login username for the SBM Service. You must have a username to login to the SBM Service.
- **Password:** The login password for the SBM Service.

To test the settings click the **Test Settings** button. SyncBackSE will then attempt to connect and login to the SBM Service.

Once these settings have been set (and validated) you may not have the access rights to change them. If not then the settings will be read-only and the **Modify** button will be visible. To change them you must click the **Modify** button and then you will be asked for the username and password of a user, e.g. an administrator, that has the access rights to change these settings.

Upload Profile to SBM Service

To upload profiles to the SBM Service first select the profile to upload then select **Preferences -> Upload Profile to SBM Service**. Optionally right-click on the profile and select **Upload Profile to SBM Service** from the pop-up menu. You can only upload profiles if you are an administrator. You must enter a description for the profile and then specify if the schedule for the profile should also be exported. After the profile has been uploaded a message will be displayed giving the unique GUID for the profile. This is for information purposes only so that when you edit the profiles details using the SBM Console you can check to make sure that the profile is the same one you uploaded. Every profile has a universally unique GUID.

Important: Once a profile has been uploaded you must use the **SBM Console** to assign the profile to one or more groups. If you are updating an existing profile then the profile will still be in the same groups as it was before the update.

Managed Profiles

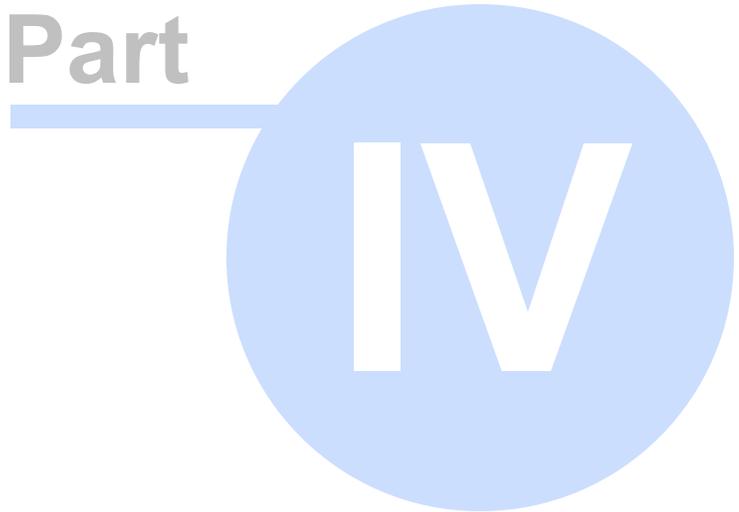
Every hour a check is made to see if there are any new or updated managed profiles. It also checks to see if any managed profiles should be deleted. If so they are downloaded from the SBM Service and installed, or deleted as necessary. This is only done when online. You can do an immediate check by pressing **Ctrl-F5**.

Offline

If SyncBackSE cannot connect to the SBM Service, e.g. there is no network connection, then it will proceed in offline mode. The window caption for SyncBackSE will show if it is in offline mode. When in offline mode it will use the cached security settings (retrieved during the last online login) so that the user will still be restricted in what they can or cannot do. Any profile history created while in offline mode will be automatically and silently uploaded once SyncBackSE can connect to the SBM Service and it has been an hour or more since it last uploaded cached history. You can do an immediate upload by pressing **Ctrl-F5**.

Legal Information

Part



4 Legal Information



Legal Documentation

Please read the following legal documentation which presents 2BrightSparks Pte Ltd's policies on licensing and distribution.

[Distribution](#)

[Privacy Statement](#)

[General Terms](#)

[Credits](#)

4.1 Distribution

Software Distribution Agreement for SyncBackSE Shareware Version

This Software Distribution Agreement (hereinafter referred to as "AGREEMENT") is a legal AGREEMENT between you, and 2BrightSparks Pte Ltd (hereinafter referred to as "AUTHOR") for distributing the computer software program entitled SyncBackSE (hereinafter referred to as "SOFTWARE").

This AGREEMENT describes the terms and conditions by which AUTHOR will license other parties to distribute the SOFTWARE which is intended solely for distribution as SHAREWARE. No use, distribution or reproduction of the SOFTWARE or copies of the SOFTWARE is authorized except in compliance with the terms and conditions herein. Distribution of the SOFTWARE in accordance with the provisions of this Software Licence Agreement is encouraged.

You should carefully read the following terms and conditions before distributing this SOFTWARE. Unless you have a different licence AGREEMENT signed by AUTHOR, your use of this SOFTWARE indicated your acceptance of this licence AGREEMENT.

By copying or distributing this SOFTWARE, you agree to be bound by the terms and conditions of this AGREEMENT as well as those of the "Software Licence Agreement".

GENERAL DEFINITIONS

As stated, the SOFTWARE is marketed as SHAREWARE.

Definition of Shareware

Shareware distribution gives users a chance to try software before buying it. If you try a Shareware program and continue using it, you are required to register it (or purchase the Licensed version).

Copyright laws apply to both Shareware and retail software, and the copyright holder retains all rights, with a few specific exceptions as stated below. The author specifically grants the right to copy and distribute the software, either to all and sundry or to a specific group.

Shareware is a distribution method, not a type of software.

GENERAL TERMS AND CONDITIONS

- AUTHOR shall be credited as the owner of the SOFTWARE in all distribution of the SOFTWARE. AUTHOR is the exclusive world-wide licensor of the SOFTWARE, and the copyrights and other proprietary rights therein. The SOFTWARE is intended solely for distribution as SHAREWARE (i.e., try-before-you-buy software); it is not public domain or free software or freeware.
- The SOFTWARE shall be identified by name and shall be identified as SHAREWARE in all distribution.
- You may copy and/or distribute the SOFTWARE only in its original, unaltered form, with all files included unmodified, and without making any additions, modifications or deletions except as provided in this paragraph. You may not modify the SOFTWARE or any of its files, and the SOFTWARE must be distributed as a complete package. You may not change, delete, merge or rename any files or elements of the SOFTWARE in any manner, and you may not add any files or new elements (except for installation routines which do not interfere with the proper operation or installation of the SOFTWARE).
- Since the SOFTWARE is intended for distribution only as SHAREWARE, you shall not charge any fee or other compensation for the SOFTWARE, although you may charge a distribution fee for costs associated with distributing the SOFTWARE. You are permitted, and encouraged, to make and distribute copies of the SOFTWARE to your friends, family members and co-workers for your and their private non-commercial use, in compliance with the terms and conditions hereof.
- You recognize that your right to distribute the SOFTWARE is nonexclusive and that AUTHOR can terminate the license granted to you at any time for any reason upon notice. AUTHOR reserves the right to withhold or withdraw permission to distribute the SOFTWARE from anyone at any time for any reason. The other provisions hereof shall survive any expiration or termination of this AGREEMENT.
- You shall take reasonable steps to ensure that the SOFTWARE and any other software, documentation and other materials distributed with the SOFTWARE are free from viruses.

- You may not use, copy, modify, distribute or transfer the SOFTWARE or any element thereof in whole or in part, except as expressly provided for herein.
- You may not rent or lease the SOFTWARE to anyone.
- AUTHOR reserves the right to update the contents of the SOFTWARE and its associated files, documentation and/or other elements, at its discretion from time to time, without the consent of, or any obligation to, any licensed users or distributors.
- You will hold AUTHOR, family members, distributors, licensees, sub-licensees and lawyers harmless from and against any and all claims, actions, damages, losses, liabilities, costs and expenses arising directly or indirectly from your acts and omissions in copying and distributing the SOFTWARE.
- If any provision of this AGREEMENT is held to be void, invalid or unenforceable, it will not affect the validity of the balance of this AGREEMENT, which shall remain valid and enforceable according to its terms and conditions.
- This agreement shall be governed by the laws of the United Kingdom.

SPECIAL TERMS AND CONDITIONS

- Distribution by BBS, on-line Services, FTP, FSP, News, WWW, Satellite, Other File Transfer Protocols: The SOFTWARE and associated files may be copied, used and posted without charge and permission as long as the "General Terms and Conditions" set forth above are complied with.
- Distribution on Floppy Disk / CD-ROM / DVD / Other Disk Types in a non-retail environment: The SOFTWARE and associated files may be copied, and used and posted without charge and permission as long as the "General Terms and Conditions" set forth above are complied with.
- Distribution on Floppy Disk / CD-ROM / DVD / Other Disk Types by Anonymous access FTP/WWW Shareware Archives: The SOFTWARE and associated files may be copied, used and posted without charge and permission as long as the "General Terms and Conditions" set forth above are complied with.
- Distribution on Magazine Companion Disk / CD-ROM / Other Disk Types: The program and associated files may be copied, used and posted without charge and permission as long as the "General Terms and Conditions" set forth above are complied with. We would greatly appreciate if you would inform 2BrightSparks about any review(s) you write about the SOFTWARE. Thanks.
- Distribution in a Retail Environment, Book Companion Disk / CD-ROM / Other Disk Types (Book): You may not distribute the program without obtaining explicit permission from AUTHOR.
- Other (e.g. Retail not covered above) CD-ROM Shareware Distribution: You may not distribute the program without obtaining explicit permission from AUTHOR.

- Internet Providers Disk / CD-ROM / Demo Disks / Connection Kits / etc.: You may distribute this SOFTWARE on your disk/CD only as bundled shareware with other programs without charge and permission as long as the "General Terms and Conditions" set forth above are complied with. If you intend to provide the program for your own diagnostic purposes then you may not distribute the program without obtaining explicit permission from the AUTHOR.
- Software/Hardware Manufacturers & Suppliers: You may not distribute the program pre-installed or otherwise on the machines you manufacture/distribute/etc. or bundled with your own products without obtaining explicit permission from AUTHOR.
- Other Type of Distribution: Please contact AUTHOR for details.

BY DISTRIBUTING THE SOFTWARE YOU ACKNOWLEDGE THAT YOU HAVE READ AND UNDERSTOOD THIS AGREEMENT AND YOU AGREE TO BE BOUND BY THIS AGREEMENT'S TERMS AND CONDITIONS. YOU ALSO AGREE THAT THIS AGREEMENT IS THE COMPLETE AND EXCLUSIVE STATEMENT OF THE RIGHTS AND LIABILITIES OF THE PARTIES AND SUPERSEDES ALL PROPOSALS OR PRIOR AGREEMENTS, ORAL OR WRITTEN AND ANY OTHER COMMUNICATION BETWEEN THE PARTIES RELATING TO THE SUBJECT MATTER OF THIS AGREEMENT.

4.2 Privacy Statement

Privacy Statement

We have created this privacy statement in order to demonstrate our firm commitment to privacy. The following discloses our information gathering and dissemination practices for 2BrightSparks.

We use your IP address to help diagnose problems with our server, and to administer our website. Your IP address is used to help identify you and to gather broad demographic information.

Cookies

We use cookies to identify you so we can retrieve your information so you don't have to re-enter it each time you visit our site. We also use cookies to help ensure our surveys are only completed once.

Subscriptions

Our site's Support Ticketing, Mailing List, and other forms requires users to give us contact information, like their name and email address. We use customer contact information from forms to send the user information about 2BrightSparks. The customer's contact information is also used to contact the visitor when necessary if they have subscribed to the Mailing List. Users may opt-out of receiving future mailings by choosing to non-subscribe.

Unique identifiers are collected to verify the user's identity and for use in our record system.

Collecting Unique Identifiers

2BrightSparks and its associated Web Store uses an order form for customers to request information, products, and services. We collect visitor's contact information and unique identifiers. Contact information from the order form is used to send orders and information about our company to our customers. The customer's contact information is also used to get in touch with the visitor when necessary. Users may opt-out of receiving future mailings. Unique identifiers are collected from website visitors to verify the user's identity and for use as account numbers in our record system.

External Linking

2BrightSparks contains links to other sites. We are not responsible for the privacy practices or the content of such websites.

Security

This site has security measures in place to protect the loss, misuse and alteration of the information under our control. All data is protected using the most advanced methods available. We do not store financial information like credit card numbers or personal information like social security numbers on this site.

Un-Subscribe

2BrightSparks gives users the option to [contact us](#) for removing their information from our database to not receive future communications or to no longer receive mail list messages.

Correct/Update

You can send an email to us by visiting our [Support Area](#) to change and/or modify information previously provided.

Additional Information

Why we use cookies

A cookie is a small data file that certain Web sites write to your hard drive when you visit them. A cookie file can contain information such as a user ID that the site uses to track the pages you have visited. The only personal information a cookie can contain is information you supply yourself. A cookie can't read data off your hard disk or read cookie files created by other sites.

We use cookies to assist in searches. Cookies further enhance our service by enabling us to fast-track much of our User's browsing habits with the aim of improving our services.

Do you share data with outside companies and/or organisations?

No. we do not share any 'off-line data' with any commercial entity (e-mail addresses

collected for our mailing list, and any data collected for purposes other than for on-line publication on our website).

We only disclose data when required to do so by International law.

4.3 General Terms

2BrightSparks Pte Ltd: General Terms and Conditions

Last Modified: 17 June 2008

Products, Services, and Conditions of Use

You should carefully read the following Terms and Conditions before using our software products. Unless you have a different license agreement signed by 2BrightSparks, your use of our software indicates your acceptance of this license agreement and warranty.

Use of all software available from 2BrightSparks (hereinafter 'SOFTWARE') is contingent on your agreement our software licensing terms found on this web page.

As part of our service, we agree to provide you with software, information, and other services that we may decide to offer, subject to the terms of this agreement. Upon notice published through the service, we may modify this agreement at any time. You agree and continue to agree to use our products and services in a manner consistent with all applicable laws and regulations and in accordance with the terms and conditions set out in the policies and guidelines outlined below. Please note that you will be referred to as 'customer' in this agreement.

Limitation Of Liability And Warranty

The customer agrees that use of the services and products provided by 2BrightSparks is entirely at the customer's own risk. Services and products are provided 'as is,' without warranty of any kind, either express or implied, including without limitation any warranty for information, services, uninterrupted access, or products provided through or in connection with the service, including without limitation the software licensed to the customer and the results obtained through the service. Specifically, we disclaim any and all warranties, including without limitation: 1) any warranties concerning the availability, accuracy or content of information, products or services; and 2) any warranties of title or warranties of merchantability or fitness for a particular purpose.

This disclaimer of liability applies to any damages or injury caused by any failure of performance, error, omission, interruption, deletion, defect, delay in operation or transmission, computer virus, communication line failure, theft or destruction or unauthorized access to, alteration of, or use of record, whether for breach of contract, tortious behavior, negligence, or under any other cause of action. Customer specifically acknowledges the service is not liable for the defamatory, offensive or illegal conduct of

other customers or third-parties and that the risk of injury from the foregoing rests entirely with customer.

Neither the software, products or services delivered by 2BrightSparks, nor any of its agents, affiliates or content providers shall be liable for any direct, indirect, incidental, special or consequential damages arising out of use of it or inability to gain access to or use the software or out of any breach of any warranty. Customer hereby acknowledges that the provisions of this section shall apply to all content on the service.

Governing Law

This agreement shall be governed by the laws of Singapore.

Trademarks

All trademarks appearing on the service are trademarks of their respective owners.

Modification

2BrightSparks reserves the right, at its discretion, to revise these Terms and Conditions at any time and without prior notice, and such revision shall be effective immediately upon the posting of the revised Terms and Conditions at the company website:

<http://www.2brightsparks.com/terms.html>

Software License Agreement

You should carefully read the following Terms and Conditions before using our software products. Unless you have a different license agreement signed by 2BrightSparks, your use of our software indicates your acceptance of this license agreement and warranty.

Use of all software available from 2BrightSparks (hereinafter 'SOFTWARE') is contingent on your agreement our software licensing terms found on this web page.

SyncBackSE Licensing Terms and Conditions

You are advised to revisit the following web page periodically as we reserve the right to change these terms and conditions at any time and without notification:

CONDITIONS OF USE

You agree and continue to agree to use our software in a manner consistent with all applicable laws and regulations and in accordance with the terms and conditions set out in the policies and guidelines outlined below.

Please note that you will be referred to as 'customer' in this agreement.

SOFTWARE LICENSE AGREEMENT

You should carefully read the following Terms and Conditions before using SyncBackSE.

Unless you have a different license agreement signed by 2BrightSparks Pte Ltd, your use of our software indicates your acceptance of this license agreement.

Use of all software available from 2BrightSparks Pte Ltd, including SyncBackSE, (hereinafter 'SOFTWARE') is contingent on your agreement to the following terms:

LIMITATION OF LIABILITY

The use of all software including SyncBackSE available from 2BrightSparks Pte Ltd ('SOFTWARE') is contingent on your agreement to the following Limitation of Liability:

SOFTWARE is provided as is, and without warranty of any kind. To the maximum extent permitted by applicable law, 2BrightSparks Pte Ltd its suppliers, its distributors, and its affiliates, or others who may offer SOFTWARE, will not be liable for any damages whatsoever, whether direct or indirect, special, incidental, consequential, or punitive of any kind (including but not limited to damages for: loss of profits, loss of confidential or other information, business interruption, personal injury, loss of privacy, failure to meet any duty - including of good faith or of reasonable care - negligence, and any other pecuniary or other loss whatsoever) arising out of, or in any way related to the use of, or inability to use our SOFTWARE or support services, or the provision of or failure to provide support services, or otherwise under, or in connection with SOFTWARE documentation, or any provision of these terms and conditions, even if 2BrightSparks Pte Ltd or any supplier, distributor, or its affiliates has been advised of the possibility of such damages.

The Limitations on, and Exclusions of liability for damages in this agreement apply regardless of whether liability is based on breach of contract, tort (including negligence), delict, strict liability, breach of warranties or conditions, or any other legal theory. 2BrightSparks Pte Ltd furthermore disclaims all warranties, including without limitation any implied warranties of merchantability, fitness for a particular purpose, and on infringement.

The entire risk arising out of the use or performance of the SOFTWARE and documentation remains with the recipient. To the maximum extent permitted by applicable law, in no event shall 2BrightSparks Pte Ltd be liable for any consequential, incidental, direct, indirect, special, punitive, or other damages whatsoever (including, without limitation, damages for loss of business profits, business interruption, loss of business information, or other pecuniary loss) arising out of this agreement or the use of or inability to use the product, even if 2BrightSparks Pte Ltd has been advised of the possibility of such damages.

The SOFTWARE and the accompanying files are sold "as is" and without warranties as to performance or merchantability or any other warranties whether expressed or implied. Because of the various hardware and software environments into which SOFTWARE may be put, no warranty of fitness for a particular purpose is offered.

Good data processing procedure dictates that any program be thoroughly tested with non-critical data before relying on it. The user must assume the entire risk of using the SOFTWARE. Any liability of the seller will be limited exclusively to product replacement or refund of purchase price.

OWNERSHIP

You may not reverse engineer, decompile or disassemble the SOFTWARE. 2BrightSparks Pte Ltd shall retain title and all ownership rights to the SOFTWARE.

COPYRIGHT

This SOFTWARE is protected by copyright laws and international copyright treaties, as well as other intellectual property laws and treaties.

MAINTENANCE

2BrightSparks Pte Ltd is not obligated to provide support, maintenance, or updates for the SOFTWARE (either by email, phone, or otherwise). However, any maintenance or updates provided by 2BrightSparks Pte Ltd shall be covered by this Agreement.

SOFTWARE USAGE AGREEMENT

One registered copy of SOFTWARE may either be used by a single person who uses the software personally on up to 5 computers for non-commercial use (home). A separate license must be purchased for each installation of the program for commercial use. Installing the program on a server counts as one installation.

In a commercial setting you may access the registered version of the SOFTWARE through a network, provided that you have obtained individual licenses for the SOFTWARE covering all workstations that will access the software through the network. For instance, if 4 different workstations access the SOFTWARE on the network, each workstation must have its own SOFTWARE license, regardless of whether they use the SOFTWARE at different times or concurrently.

RETURNS POLICY

Before purchasing software from 2BrightSparks Pte Ltd, you are strongly encouraged to 'test drive' it using the evaluation version. In the event you encounter a problem with the product, contact our support department for assistance. You may do this by submitting a Support Ticket in our Support Area at:

<http://www.2brightsparks.com/help/>

After your purchase, refunds will only be given at the discretion of the Company Management.

EVALUATION AND REGISTRATION

Subject to the terms of this agreement, you are hereby licensed to use the SOFTWARE for evaluation purposes without charge. The evaluation version of the SOFTWARE has limitations and reminders that the SOFTWARE is in use. For a full-featured, unrestricted version, a registration fee is required. Upon payment you will be sent an email that will provide the Serial Number to unlock the SOFTWARE.

SyncBackSE DISTRIBUTION AGREEMENT

This Software Distribution Agreement (hereinafter referred to as "AGREEMENT") is a legal AGREEMENT between you, and 2BrightSparks Pte Ltd (hereinafter referred to as "AUTHOR") for distributing the computer software program entitled SyncBackSE (hereinafter referred to as "SOFTWARE").

This AGREEMENT describes the terms and conditions by which AUTHOR will license other parties to distribute the SOFTWARE which is intended solely for distribution as SHAREWARE. No use, distribution or reproduction of the SOFTWARE or copies of the SOFTWARE is authorised except in compliance with the terms and conditions herein. Distribution of the SOFTWARE in accordance with the provisions of this Software Licence Agreement is encouraged.

You should carefully read the following terms and conditions before distributing this SOFTWARE. Unless you have a different licence AGREEMENT signed by AUTHOR, your use of this SOFTWARE indicated your acceptance of this licence AGREEMENT.

By copying or distributing this SOFTWARE, you agree to be bound by the terms and conditions of this AGREEMENT as well as those of the "Software Licence Agreement".

GENERAL DEFINITIONS

As stated, the SOFTWARE is marketed as SHAREWARE.

Definition of Shareware

Shareware distribution gives users a chance to try software before buying it. If you try a Shareware program and continue using it, you are required to register it (or purchase the Licensed version).

Copyright laws apply to both Shareware and retail software, and the copyright holder retains all rights, with a few specific exceptions as stated below. The author specifically grants the right to copy and distribute the software, either to all and sundry or to a specific group.

Shareware is a distribution method, not a type of software.

GENERAL TERMS AND CONDITIONS

- AUTHOR shall be credited as the owner of the SOFTWARE in all distribution of the SOFTWARE. AUTHOR is the exclusive world-wide licensor of the SOFTWARE, and the copyrights and other proprietary rights therein. The SOFTWARE is intended solely for distribution as SHAREWARE (i.e., try-before-you-buy software); it is not public domain or free software or freeware.
- The SOFTWARE shall be identified by name and shall be identified as SHAREWARE in all distribution.
- You may copy and/or distribute the SOFTWARE only in its original, unaltered form, with all files included unmodified, and without making any additions, modifications or deletions except as provided in this paragraph. You may not modify the SOFTWARE or any of its

files, and the SOFTWARE must be distributed as a complete package. You may not change, delete, merge or rename any files or elements of the SOFTWARE in any manner, and you may not add any files or new elements (except for installation routines which do not interfere with the proper operation or installation of the SOFTWARE).

- Since the SOFTWARE is intended for distribution only as SHAREWARE, you shall not charge any fee or other compensation for the SOFTWARE, although you may charge a distribution fee for costs associated with distributing the SOFTWARE. You are permitted, and encouraged, to make and distribute copies of the SOFTWARE to your friends, family members and co-workers for your and their private non-commercial use, in compliance with the terms and conditions hereof.
- You recognise that your right to distribute the SOFTWARE is nonexclusive and that AUTHOR can terminate the license granted to you at any time for any reason upon notice. AUTHOR reserves the right to withhold or withdraw permission to distribute the SOFTWARE from anyone at any time for any reason. The other provisions hereof shall survive any expiration or termination of this AGREEMENT.
- You shall take reasonable steps to ensure that the SOFTWARE and any other software, documentation and other materials distributed with the SOFTWARE are free from viruses.
- You may not use, copy, modify, distribute or transfer the SOFTWARE or any element thereof in whole or in part, except as expressly provided for herein.
- You may not rent or lease the SOFTWARE to anyone.
- AUTHOR reserves the right to update the contents of the SOFTWARE and its associated files, documentation and/or other elements, at its discretion from time to time, without the consent of, or any obligation to, any licensed users or distributors.
- You will hold AUTHOR, family members, distributors, licensees, sub-licensees and lawyers harmless from and against any and all claims, actions, damages, losses, liabilities, costs and expenses arising directly or indirectly from your acts and omissions in copying and distributing the SOFTWARE.
- If any provision of this AGREEMENT is held to be void, invalid or unenforceable, it will not affect the validity of the balance of this AGREEMENT, which shall remain valid and enforceable according to its terms and conditions.
- This agreement shall be governed by the laws of the United Kingdom.

SPECIAL TERMS AND CONDITIONS

- **Distribution by BBS, on-line Services, FTP, FSP, News, WWW, Satellite, Other File Transfer Protocols:** The SOFTWARE and associated files may be copied, used and posted without charge and permission as long as the "General Terms and Conditions" set forth above are complied with.
- **Distribution on Floppy Disk / CD-ROM / DVD / Other Disk Types in a non-retail environment:** The SOFTWARE and associated files may be copied, and used and posted

without charge and permission as long as the "General Terms and Conditions" set forth above are complied with.

• **Distribution on Floppy Disk / CD-ROM / DVD / Other Disk Types by Anonymous access FTP/WWW Shareware Archives:** The SOFTWARE and associated files may be copied, used and posted without charge and permission as long as the "General Terms and Conditions" set forth above are complied with.

• **Distribution on Magazine Companion Disk / CD-ROM / Other Disk Types:** The program and associated files may be copied, used and posted without charge and permission as long as the "General Terms and Conditions" set forth above are complied with. We would greatly appreciate if you would contact us about any review(s) you write about the SOFTWARE at:

<http://www.2brightsparks.com/contact.html>

• **Distribution in a Retail Environment, Book Companion Disk / CD-ROM / Other Disk Types (Book):** You may not distribute the program without obtaining explicit permission from AUTHOR.

• **Other (e.g. Retail not covered above) CD-ROM Shareware Distribution:** You may not distribute the program without obtaining explicit permission from AUTHOR.

• **Internet Providers Disk / CD-ROM / Demo Disks / Connection Kits / etc.:** You may distribute this SOFTWARE on your disk/CD only as bundled shareware with other programs without charge and permission as long as the "General Terms and Conditions" set forth above are complied with. If you intend to provide the program for your own diagnostic purposes then you may not distribute the program without obtaining explicit permission from the AUTHOR.

• **Software/Hardware Manufacturers & Suppliers:** You may not distribute the program pre-installed or otherwise on the machines you manufacture/distribute/etc. or bundled with your own products without obtaining explicit permission from AUTHOR.

• Other Type of Distribution: Please contact AUTHOR for details.

BY DISTRIBUTING THE SOFTWARE YOU ACKNOWLEDGE THAT YOU HAVE READ AND UNDERSTOOD THIS AGREEMENT AND YOU AGREE TO BE BOUND BY THIS AGREEMENT'S TERMS AND CONDITIONS. YOU ALSO AGREE THAT THIS AGREEMENT IS THE COMPLETE AND EXCLUSIVE STATEMENT OF THE RIGHTS AND LIABILITIES OF THE PARTIES AND SUPERSEDES ALL PROPOSALS OR PRIOR AGREEMENTS, ORAL OR WRITTEN AND ANY OTHER COMMUNICATION BETWEEN THE PARTIES RELATING TO THE SUBJECT MATTER OF THIS AGREEMENT.

Freeware

Licensing Terms and Conditions

2BrightSparks grants you a limited non-exclusive license to use FREEWARE downloadable from 2BrightSparks for personal, educational, charity, and commercial use, and donations are entirely optional.

If you are using the SOFTWARE free of charge under the terms of this Agreement, you are not entitled to support although we will respond to support requests if they relate to any SOFTWARE that is not performing it's task correctly (bugs etc).

Our Freeware is licensed to you in accordance with the terms and conditions of this Agreement. You represent and warrant that you will not violate any of the requirements of this Agreement and further represent and warrant that:

- You will not, and will not permit others to:

(i) reverse engineer, decompile, disassemble, derive the source code of, modify, or create derivative works from our Freeware, or

(ii) copy, distribute, publicly display, or publicly perform content contained in this Freeware other than as expressly authorized by this Agreement.

- You will not use our Freeware to engage in or allow others to engage in any illegal activity.

- You will not engage in using our Freeware that will interfere with or damage the operation of the services of any third parties by overburdening/disabling network resources through automated queries, excessive usage or similar conduct.

- You will not sell our Freeware or charge others for use of it (either for profit or merely to recover your media and distribution costs) whether as a stand-alone product, or as part of a compilation or anthology, without explicit prior written permission.

- You will not use our Freeware to engage in any activity that will violate the rights of third parties, including, without limitation, through the use, public display, public performance, reproduction, distribution, or modification of communications or materials that infringe copyrights, trademarks, publicity rights, privacy rights, other proprietary rights, or rights against defamation of third parties.

- You may not claim any sponsorship by, endorsement by, or affiliation with our company.

Limitation of Liability

Please read the LIMITATION OF LIABILITY above which applies to Freeware and Commercial software provided by 2BrightSparks Pte Ltd.

4.4 Credits

Program Credits

SyncBackSE Program Developers

2BrightSparks Pte Ltd © 2003 - 2011

Website: www.2BrightSparks.com

Additional Credits

SyncBackSE makes use of the following components and services:

SyncBackSE is written in Delphi XE2 by Embarcadero®: www.embarcadero.com

Resources: AbleStable, Serving the Creative Community: www.ablestable.com

VirtualTreeview	www.delphi-gems.com
Zip Compression Library	www.xceedsoft.com
SmartFTP	www.smartftp.com
EASendMail SMTP	www.emailarchitect.com
Component and EAGetMail	
POP3 & IMAP4 Component	
MD5 hashing by David Barton	crypto@cityinthesky.co.uk
Project JEDI	www.delphi-jedi.org
TRegExpr Library	RegExpStudio.com
madExcept	madshi.net
DISQLite3	www.yunqa.de
NMS DVD Burning SDK	www.numediasoft.com
ExtremeCopy	www.easersoft.com
IP*Works, Amazon Integrator,	www.nsoftware.com
Azure Integrator	
RemObjects SDK	www.remobjects.com
GnuGetText	dxgettext.po.dk
ICS	www.overbyte.be
Inno Setup	jrsoftware.org
FinalBuilder	www.finalbuilder.com

International Language Translations



Armenian translation by Hrant Ohanyan



Catalan translation by Jordi Rodellar Fustero



Czech translation by Vlamo



Danish translation by Mads Andersen



Dutch translation by Gerard Entius and Hans Fraiponts



Finnish translation by Jarkko Mäkineva



French translation by Philippe Septier



German translation by Stefan Aicher



Hungarian translation by Jozsef Tamas Herczeg



Italian translation by Roberto Lodigiani and Massimo Pedrazzoli



Japanese translation by Kawai & Tsu_Kun



Norwegian translation by Kjetil Birkeland Moe



Polish translation by Lukasz Puzon Brodowski



Portuguese (Brazilian) translation by Edson Wunderlich



Simplified & Traditional Chinese translation by Larry Ho



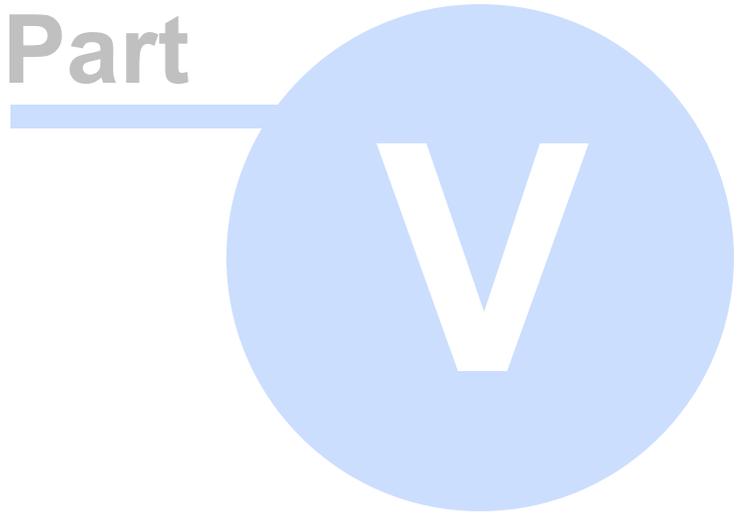
Spanish (Argentinean) translation by Gustavo Santiago



Swedish translation by Linus Rörstad

Glossary

Part



5 Glossary



Glossary

AES

Advanced Encryption Standard

This is an encryption method used to secure communications or files. It is very hard to crack; to the extent that the U.S. Government use it for top secret documents.

Alt Tag

The alt tag is a label describing an image. It appears when the mouse is rolled over an image on a webpage (text within a small yellow window in Windows, white in Macs). It is particularly helpful for people who view pages in text-only mode and/or who have special software that converts the text to an audio equivalent.

AFTP

Anonymous file transfer protocol

Abbreviation: anonymous FTP; AFTP. A system for using the standard FTP without requiring a user ID or password (see also **FTP** and **SFTP**).

Anti-Virus

Anti-virus software

Anti-virus software is used to scan files for possible rogue instructions (viruses) that may have been attached to them. These instructions, if run by an application, might perform unwanted actions.

Applet

Small Java programs that 'auto install' when browsing (given the browser has Java enabled). Applets are designed to run on any system. Unlike an application, applets cannot be executed directly from the operating system. Sun Microsystems certifies applets or applications written in 100% pure Java will run in all systems equipped with a Java Virtual Machine, not just in Microsoft Windows environments.

ASCII

American Standard Code for Information Interchange

ASCII is the standard method for encoding characters as 8-bit sequences of binary numbers, allowing a maximum of 256 characters. Text files are customarily called 'ASCII files'. ASCII also includes control characters such as carriage return and tab (see also Unicode).

ASP

Active Server Pages

ASP stands for Active Server Pages. A blend of traditional HTML and database server language. When a server accesses an active content page, the requested page is passed through the database server where the code is processed, and a new HTML page is generated. This page is then returned to the regular web server and sent on to the user.

Attachment

A file which is 'attached' and sent as part of an e-mail message, e.g. much in the same way as one would attach a photo to a written letter with a paper clip.

Attribute

Read-only attribute

Describes a file that may only be read and not changed, e.g. a files on a CD-ROM are by default read-only. This attribute can be assigned to most files or folders. A read-only file or folder cannot be modified or deleted, but a read-only folder may have its contents modified or deleted. If the device (e.g. CDRom) is not read-only in nature (e.g. a hard disk drive) then the read-only attribute may be removed to make the file or folder writeable if desired.

Archive attribute

The file has been marked by the operating system as an archive file. Applications or the operating system use this attribute to mark files ready for backup or removal. Every time a file is changed, or created, the file is marked with the archive attribute.

Hidden attribute

The file is hidden. It is not included in an ordinary directory listing. The hidden attribute is typically given to important operating system files, the reason being that they don't normally need to be seen or changed by the user.

System attribute

The file is part of the operating system or is used exclusively by it. As with the hidden attribute, this is typically given to important operating system files.

Temporary attribute

The file is being used for temporary storage.

Offline attribute

The data of the file is not immediately available. This attribute indicates that the file data has been physically moved to offline storage, e.g. tape, CD or DVD storage.

Not indexed attribute (not content indexed)

The file will not be indexed by the Windows content indexing service.

Encrypted attribute

A Microsoft file-based encryption technology that enables users to encrypt files and folders on NTFS-formatted volumes. EFS helps protect the confidentiality of data by ensuring that only authorized users can decrypt the encrypted files or folders.

Compressed attribute

This attribute denotes that the file has been compressed by the operating system to make it generally much smaller than the original size. Supported by NTFS volumes (Windows NT and newer). The benefits are the saving of disk space and under certain conditions, increased speed of access.

Creation date attribute

This attribute specifies the original creation date and time of file or folder. When the file or folder is created / newly made, creation date and time are recorded in file system.

Last modified date attribute

Indicates at the file system level when the last time the file or folder was modified. When the file's contents are modified or saved, the last modified date and time stamp is set to correspond with the time of those changes.

Last accessed date attribute

Indicates at the file system level when a file was last accessed. If the file is only opened for viewing, copying,

moving, etc., the last accessed date is changed to when the operation took place.

AVI

Audio Video Interleave

AVI is a common format on the Internet for movies and videos (typically Windows based). Contrary to popular belief, an AVI file is simply a container for any number of different video formats (see also **MPG**).

Background Backup (SyncBack/SE/Pro)

The process of running a backup profile in the background, i.e. without user-intervention. Running profiles in the background is similar to running profiles via the Windows Task Scheduler, except SyncBack must be running for them to run. Also background profiles typically run much more frequently than scheduled tasks, e.g. every 30 minutes.

Backup

A Backup is where files are copied one-way from the Source to the Destination. The backup process prohibits files being copied the other way round (see also **Mirror** and **Synchronize**).

Bandwidth

This is the maximum amount of data which can be carried at a given time by your Internet connection. Generally, the larger the bandwidth, the quicker the data will be received or sent. Example: ADSL Broadband has a higher bandwidth than an analog modem.

BMP

Bitmap

A type of lossless graphic file format used to save a digital image containing a color value for each pixel in a picture. It is a simpler format than a JPEG or GIF (see also **GIF**, **JPG** and **TIFF**).

Boolean Search

A Boolean search is one formed by joining simple terms in a logical way with "AND", "OR" and "NOT". Boolean terms can also be expressed in symbols like "-" instead of "minus" or "+" instead of "plus".

Bug

An unintentional programming error that causes a program or computer system to perform erratically, produce incorrect results, or crash.

BWT

Burrows-Wheeler Transform

A compression algorithm used in BZip2 (see **BZip2**, **Compression** and **Zip**).

BZip2

BZip2 is a compression technique. It compresses most files more effectively than more traditional GZip or Zip, but it is slower (see also **BWT**, **Compression** and **Zip**).

C and C++

These are both widely used computer programming languages.

Cache

A small area of fast memory provided to increase the effective speed of a large amount of slower memory. Your browser uses a 'cache' to store web pages and parts of web pages you have visited during a session. Instead of retrieving the page again from the Internet, your browser will get it faster from the cache.

CAD

Computer aided design

CAD is the process of using a computer to assist in the design process, usually by automating the production of drawings. CAD techniques are widely used in engineering and architecture.

CGI**Common Gateway Interface**

CGI is the standard for running programs on a server from a Web page. Gateway programs, or scripts, are executable programs which can be run by themselves.

CHMOD

Originally from the UNIX world, CHMOD is used by FTP (File Transfer Protocol) programs for PC and Mac that allow the directory and file permissions to be changed over the Internet.

Character Set

The basic set of letters and symbols that a computer uses, or are included in a particular font.

Compression

Compression is the process of making a file smaller by using a complex algorithm of bit reduction (see also **BWT**, **BZip2** and **Zip**).

Cookie

A cookie is a small data file that certain web sites write to your hard drive when you visit them with your browser. A cookie file can contain information such as a user ID that the site uses to track the pages you have visited. The only personal information a cookie can contain is information you supply yourself. A cookie can't read data off your hard disk or read cookie files created by other sites.

CSS**Cascading Style Sheets**

Cascading Style Sheets (CSS) refers to a style 'language' used by web designers to define presentational aspects of a web document (typeface, background, text, link colors, margin controls, and the placement of objects on a web page).

Database

A collection of data that is organized so that its contents can easily be accessed, managed, and updated.

Delphi

Delphi is a programming language and software development environment. It is produced by Embarcadero (formerly CodeGear, formerly Inprise, and originally Borland). The Delphi language, formerly known as Object Pascal (Pascal with object-oriented extensions) originally targeted only Microsoft Windows, but now also builds native applications for Linux and the Microsoft .NET framework.

Destination

The Destination is where files and directories are copied to in a backup profile (they are copied from the Source). In a synchronize profile the destination can generically be thought of as the "right side".

Differences Window (SyncBack products)

During a profile run, the Differences Window shows what will happen to files (whether they will be copied, deleted, or moved). For example, it will show you how many "collisions" have occurred. A collision is when a file in the source and destination differ but have the same name. In other words, the file is in both the source and destination but is modified in some way, perhaps by date, size etc. The differences window will help you decide what course of action to take based upon these results.

Differential Backup

Differential backups include backing up all files that have changed since the last full backup. Hence to restore all your data, all you would need are the last full and differential backups. The difference between differential and incremental backups is that incremental backups include only the files that have changed since the last full or incremental backup (see also **Incremental Backup**). In SyncBackSE, differential backups are possible using Fast Backup profiles.

Directory (another name for Folder)

In computing, a directory, catalog, or folder, is an entity in a file system which contains a group of files and other directories. A typical file system contains thousands of files, and directories help organize them by keeping related files together. A directory contained inside another directory is called a subdirectory (or subfolder) of that directory. Together, the directories form a hierarchy, or tree structure.

Disk Spanning (SyncBackPro)

If SyncBackPro is performing a backup to a removable medium, e.g. a floppy-disk, or CD/DVD then it may be that the backup cannot fit onto one disk. If this option is enabled then the Zip file will be split across a number of disks.

DLL

Dynamic Link Library

DLL files are a method for storing a program's components in separate files to the main program. DLL files typically have the extension .DLL and cannot be launched directly.

DNS

Domain Name System

When you type in a website address (e.g. www.ablestable.com) into your browser, a domain name server (typically at your ISP) translates this into a numerical address, known as an IP Address, so your request can be routed to the correct site.

Domain

Internet Domain

An Internet Domain A 'logical' region of the Internet. People sometimes refer to them loosely as 'sites'. Generally, a domain corresponds to one or more IP addresses or an area on a host. A domain is organized in levels. The top level identifies geographic or purpose commonality. The second level identifies a unique place within the top level domain and is, in fact, equivalent to a unique address on the Internet (an IP address).

Windows Domain

In Windows NT and newer, a domain binds together a set of network resources (applications, printers, and so forth) for a group of users. The user only has to log in once to the domain to gain access to the resources, as opposed to having to authenticate to each one.

Download

Downloading is typically the transfer of data from the Internet down to a computer. A download in its widest definition is the calling up of web pages from a browser. The more usual and focused use for the term download is when a user requests a file from a location on the Internet and makes a copy of that file onto their own computer (see also **Upload**).

Encryption

Encryption is a process whereby information is scrambled so no unauthorized person can access it (see also **EFS** and **TrueCrypt**).

EFS

Encrypting File System

The Encrypting File System (EFS) is a file system driver that provides filesystem-level encryption in Microsoft Windows (2000 and later) operating systems, except Windows XP Home Edition, Windows Vista Basic, and

Windows Vista Home Premium. The technology enables files to be transparently encrypted on NTFS file systems to protect confidential data from attackers with physical access to the computer.

Ethernet

Ethernet is one of the most popular standards for connecting PC's to form a local area network. It can be carried over a wide number of media, including copper, fiber-optics and wireless.

Extension

On computers using MSDOS or Windows, the last part of a filename, after a dot (period) is known as the extension. These systems use the extension for indicating the type of information that the file contains. For example, the main program file for SyncBackPro is called SncBackPro.exe, the 'exe' indicating that it is an executable program. A file called ReadMe.txt is a text file.

FAQ

Frequently Asked Questions

This is simply a compilation of frequently asked questions, the intention being that it should be the first port of call when a user is looking for answers to a problem.

Fast Backup

This is an option within SyncBackSE/Pro to greatly improve the performance of a backup profile by not scanning the destination first. This allows for other backup methods such as Incremental and Differential. It is recommended that the FAQ is read carefully before using this feature.

FAT

File Allocation Table

File Allocation Table (FAT, FAT12, FAT16 and FAT32) are file systems that were developed for MS-DOS and used in consumer versions of Microsoft Windows up to and including Windows 7. The FAT file system is considered relatively uncomplicated, and because of that, it is a popular format for floppy disks; moreover, it is supported by virtually all existing operating systems for personal computers, and because of that it is often used to share data between several operating systems booting on the same computer (a multi-boot environment). It is also used on removable memory cards and other similar devices. It is important to note that the different variants of FAT have their various limitations. For example, the maximum file size you can store on a FAT system is 32MB, 2GB on FAT16 and 4GB on FAT32 (see also FAT32).

FAT32

This is the last in the line of Microsoft's FAT file systems. In order to overcome the volume size limit of FAT while still allowing memory-constrained DOS real-mode code to handle the format, Microsoft decided to implement a newer generation of FAT, known as FAT32, with 32-bit cluster numbers, of which 28 bits are currently used.

File Versioning

See Versioning

Firewall

Firewalls are special devices, computers or computer programs that are installed on a network to prevent intruders from stealing files, snooping or disabling the host computer(s). In the home environment the firewall is usually a piece of software installed on the individual's computer. In the corporate environment it is an 'appliance' type device that intercepts, blocks or allows, all data coming in and out of the whole organization.

Folder (another name for Directory)

A folder is a file container on a disk. Like a folder in a filing cabinet, you can store related files in the same folder to help organize your information.

FTP

File Transfer Protocol

The most widely-used method of downloading and uploading (getting and putting) files between two computers on the Internet. FTP is a simple network protocol based on Internet Protocol and also a term used when referring to the process of copying files when using FTP technology (see also **AFTP** and **SFTP**).

FTPS

File Transfer Protocol Secure

FTPS (also known as FTP Secure and FTP-SSL) is an extension to the commonly used File Transfer Protocol (**FTP**) that adds support for the Transport Layer Security (TLS) and the Secure Sockets Layer (SSL) cryptographic protocols. FTPS should not be confused with the SSH File Transfer Protocol (**SFTP**), an incompatible secure file transfer subsystem for the Secure Shell (SSH) protocol. It is also different from Secure FTP, the practice of tunneling FTP through an SSH connection. (see also **SFTP** and **FTP**).

GIF

Graphics Interchange Format

A type of image file format. It is the most common way to compress and store images for transfer over the Internet. It supports animations and allows a separate palette of 256 colors for each frame. Color palette limitations makes the GIF format unsuitable for reproducing color photographs, but it is well-suited for simpler images such as graphics or logos with solid areas of color (see also **BMP** and **JPG** and **TIFF**).

Group Profile

In the SyncBack series of products, a Group Profile is a collection (set) of Profiles. They allow you to run a number of profiles in parallel or a specific order.

Hash (as in CRC32 or MD5 value)

A hash function is a process that converts an input from a (typically) large domain into an output in a (typically) smaller range (the hash value, often a subset of the integers). Hash functions vary in the domain of their inputs and the range of their outputs and in how patterns and similarities of input data affect output data. It is typically used in the verification of file provenance – e.g. as a checksum to detect accidental data corruption during a download.

CRC

A cyclic redundancy check (CRC) is a type of hash function used to produce a checksum, which is a small number of bits, from a large block of data, such as a packet of network traffic or a block of a computer file, in order to detect errors in transmission or storage. A CRC is computed and appended before transmission or storage, and verified afterwards to confirm that no changes occurred.

MD5

In cryptography, MD5 (Message-Digest algorithm 5) is a widely-used cryptographic hash function with a 128-bit hash value. As an Internet standard, MD5 has been employed in a wide variety of security applications, and is also commonly used to check the integrity of files.

Host

1. A company that rents out space on the Internet to allow you to place web pages on it (see also Web Host)
2. The generic name for a computing device connected to a network, be it a LAN or the Internet.

Hot-Key

A Hot-Key is the name given to a keyboard shortcut (also known as an accelerator key, shortcut key, or hot-key). It comprises a set of keyboard keys that when pressed simultaneously, perform a predefined task. Such a task could be done with the mouse (or other analog input such as a trackball), but would require much longer. Hence, they are a shortcut in that they save the user time.

HTML

Hypertext Mark-up Language

HTML is the common language that lies behind most web sites and their pages found on the on the World Wide Web. It defines a set of standards that enable the author to format a page in a variety of different styles and appearances.

HTTP

Hypertext Transfer Protocol

This is the name given to the Internet protocol standard for defining the way computers transmit data on the Internet during a browser session. An exchange between your browser and web site will consist of an HTTP 'conversation' that finally results in you receiving and displaying web pages. Many Internet addresses begin http, for example: <http://www.2brightsparks.com>

HTTPS

Hypertext Transfer Protocol Secure

This protocol elevates HTTP to a secure level and is required for pages that require a SSL (secure sockets layer) connection.

Hypertext

A hypertext link is a special word or phrase in a web page that 'points' to another page. When clicked upon, you are taken to the page the link refers to, thus enabling navigation. Visually, links are typically underlined or contained within graphic elements.

Icon

A small image on the computer's display which represents some action or object.

IMAP4

Internet Message Access Protocol

This is an Internet protocol that defines and controls how emails are received by an email program from an email server that supports IMAP4. It is primarily used by users who don't have permanent connections to their email server (delivery on demand). The ISP's IMAP4 server receives and holds messages that have been sent to the user until their email program connects up and retrieves them (see also POP3 and SMTP).

IP Address

Internet Protocol Address

The 32-bit address defined by the Internet Protocol. Every resource on the Internet has a unique numerical IP address, represented in dotted decimal notation. IP addresses are the closest thing the Internet has to phone numbers. By calling that number you get connected to the computer that 'owns' that IP address (see also **IP**).

Incremental Backup

An Incremental Backup just backs up the data which has its Archive bit set, or has been changed since the last full or incremental backup (see also **Differential Backup**). In SyncBackSE, incremental backups are possible using Fast Backup profiles.

InterNIC

Internet Information Centre

InterNIC is the combined name for the regulatory body that provide registration, information, and database services to the Internet.

Intranet

A private network inside a company or organization that uses the same kinds of software and resources that you would find on the public Internet, but which is only for internal use.

IP

Internet Protocol

An industry standard, connectionless, best-effort packet switching protocol used as the network layer in the TCP/IP Protocol Suite. It has the task of delivering distinguished protocol datagrams (packets) from the source host to the destination host solely based on their addresses.

ISO (file format)

International Organization for Standardization

An ISO file is an archive file (also known as a disk image) of an optical disc such as a CD or DVD. The term ISO has been somewhat hijacked simply because the format is defined by the International Organization for Standardization (ISO). ISO image files typically have a file extension of .ISO.

ISP

Internet Service Provider

Primarily a company that gives you access to the Internet, but will normally offer other services such as email, website hosting and online databases.

Jar

Java Archive

A file format used to bundle all components required by a Java applet. JAR files simplify the downloading of applets since all components (.class files, images, sounds, etc.) can be packaged into a single file.

Java

A programming language developed by Sun Microsystems. Java is an object-oriented language similar to C++, but simplified to eliminate language features that cause common programming errors. Many websites rely upon the applications it produces, and whilst being a predominantly Internet-based language, it can also be found in devices such as mobile phones.

Joliet

Joliet is a file system extension designed by Microsoft for storing long filenames on optical media. It extends the previous standard (ISO9660 Level 1) which was only able to store filenames with 8.3 filenames (an 8 character filename followed by a 3 character extension).

JVM

Java Virtual Machine

A self-contained operating environment that behaves like a separate computer and is designed to minimize the effects of badly-behaved programs. For example, Java applets run in a Java virtual machine (JVM) that has no access to the host operating system.

Java Bean

JavaBeans are reusable software components for Java that can be manipulated visually in a programming tool.

JavaScript

Not to be confused or associated in any way with Java, JavaScript is a scripting language which is used to embed small programs such as pop-up windows into the HTML code of a webpage.

JPG / JPEG

Joint Photographic Experts Group

A standard type of image file commonly found on the Web. It uses a variable compression technique to reduce the size of the file and is especially suitable for photographic images (see also **BMP**, **GIF** and **TIFF**).

Junction Point (also called a Reparse Point)

In computing, a NTFS junction point (JP) is a type of NTFS reparse point in the NTFS file system. It requires a

NTFS 5.0 file system, which can be created (or converted from a FAT partition) under Windows 2000 or newer. It can be used in a similar way to symbolic links - allowing you to create a link to a folder that is, for most intents and purposes, the same as the folder itself. This has many benefits over a windows shortcut (.lnk) file, such as allowing you to access files within the shortcut via explorer, the console, etc.

KB

Kilobyte

A kilobyte is a unit of memory capacity equal to 1024 bytes. It isn't 1000 bytes as might be expected because computers tend to favor sizes that are a power of two (1024 is two to the power of 10).

LAN

Local Area Network

A LAN is a Local Area Network allowing several connected computers and/or peripherals to work together and share resources. The various devices are typically connected using a high speed link (10mb/s or greater) over cabled or wireless Ethernet.

Locale

A subset of a user's environment that defines conventions for a specified culture, such as time formatting, numeric formatting, monetary formatting, and character classification, conversion, and collation.

Locked files

A file becomes locked when an application opens it for writing. SyncBackSE/Pro is able to backup these files by utilizing Windows' VSS (see also **VSS**).

Log-in

A user or program logs in by providing a user name and password to gain access to a restricted area of a network, web site or computer.

MB

Megabyte

A megabyte is a unit of memory capacity equal to 1,048,576 bytes. It isn't 1,000,000 bytes as might be expected because computers tend to favor sizes that are a power of two (1024 is two to the power of 10).

Merchant

The organization accepting credit card or other e-payments for the goods or services they provide.

MDTM

MoDification TiMe [of a file]

When communicating with an FTP server, this command returns the last-modified time of the given file. A few types of FTP server allow *setting* the last-modified time of a file using this command (see also **MFMT**).

MFMT

MODIFY FACT: MODIFICATION TIME

MFMT is a command used in FTP to modify the last modification time of a folder or file on the destination file system.

MIME

Multipurpose Internet Mail Extensions

An extension to the Simple Mail Transfer Protocol (SMTP) that allows different forms of data including video, audio, or binary data to attach to e-mail, without requiring translation into plain ASCII text.

Mirroring

Mirroring is the process whereby the source is copied in its entirety to the destination. During this process,

extraneous files are also deleted from the destination until it is identical to the source.

MLSD

MLSD is a machine readable format for directory listings. MLST and MLSD are FTP commands intended to provide detailed, standardized directory listings across different server platforms.

MODE Z Compression

Mode Z compression compresses files-on-the-fly as they are being transferred from the local computer to the remote computer and remote to local, saving bandwidth and improving transfer time.

MPG, MPEG

Motion Picture Experts Group

MPG is a common video format for movies and videos, especially those on DVD. Any computer with DVD playing facilities will be able to play MPG files (see also **AVI**).

NAS

Network Attached Storage

A NAS device is a server that runs an operating system specifically designed for the storage and serving of files. Network-attached storage is accessible directly on the network through protocols such as TCP/IP. Think of it as a dedicated Windows computer that serves little or no other role than to store and serve files (see also **Server**).

NAT

Network Address Translation

NAT is a technique that hides a private IP address behind a single IP address in another, often public address space. It is commonly used in home ADSL broadband routers for adding an extra layer of protection from the Internet.

Network

Two or more computers working together so they can exchange information with each other (see also **LAN**).

Network Drive

See **Share**

NTFS

New Technology File System

The NTFS File System is the standard file system of Windows NT and its descendants. Windows version 95, 98, 98SE and ME, cannot natively read NTFS file systems, although utilities do exist for this purpose. NTFS replaced Microsoft's previous FAT file system, used in MS-DOS and early versions of Windows. NTFS has several improvements over FAT such as improved support for meta-data and the use of advanced data structures to improve performance, reliability and disk space utilization plus additional extensions such as security access control lists and file system journaling.

OSP

Online Service Provider

An OSP offers specific proprietary content in addition to the usual World Wide Web and Internet access, for example, AOL.

Payment Service Provider

A third party service provider directly linked with credit-card authorization network for the acceptance of credit cards and e-payments for orders placed online.

PGP

Pretty Good Privacy

Software that encrypts important information so it can be sent over the Internet securely. PGP offers military grade encryption and is available free to home users.

PDF**Portable Document Format**

Adobe® Portable Document Format (PDF) is the cross-platform standard for electronic document distribution worldwide. Adobe PDF is a universal file format that preserves all the fonts, formatting, graphics, and color of any source document, regardless of the application and platform used to create it.

POP3**Post Office Protocol 3**

This is an Internet protocol that defines and controls how emails are received by an email program from a POP3 server. It is primarily used by users who don't have permanent connections to their email server (delivery on demand). The ISP's POP3 server receives and holds messages that have been sent to the user until their email program connects up and retrieves them (see also **IMAP4** and **SMTP**).

Port

In TCP/IP communications, devices communicate with each other over certain port numbers. Each side of a TCP connection has an associated 16-bit port number assigned by the sending or receiving application. For example, web pages are sent over port 80 (HTTP), FTP communications are sent over port 21.

Profile (as in a SyncBack/SE/Pro profile)

A Profile defines and stores information about the folders or files you would like to backup or synchronize using SyncBack/SE/Pro. Once you've created a Profile you'll be able to click a single button on the toolbar to carry out a specified task in the future. They can be edited to fine-tune any type of backup process.

Protected Files

Protected files are critical system files that are installed as part of Windows (for example, files with a .dll, .exe, .ocx, and .sys extension and some True Type fonts). Windows uses a system to verify if protected system files are the correct Microsoft versions. If a program tries to replace these files, Windows will restore the original ones.

Protocol

Protocol is the term used to describe the standard for communication between computers. If two computers have a protocol in common, then they should be able to communicate even if they are completely different. There are a range of standard protocols to cover the different types of communication application. For example: DNS (naming); FTP (file transfer); HTTP (World-Wide Web documents); NNTP (news); POP, SMTP (e-mail).

Proxy Server

A proxy server is a computer network service which allows clients to make indirect network connections to other network services. A client connects to the proxy server, then requests a connection, file, or other resource available on a different server. The proxy provides the resource, possibly by connecting to the specified server, or by serving it from a cache. In some cases, the proxy may alter the client's request or the server's response for various purposes. One of the benefits of such a scheme is in the speeding up of web content to the client.

Public Domain

If a work is in the 'public domain' it is generally freely distributable although at times companies may have rights over distribution.

Redirect URL

A Web site address that when called redirects the user to a different URL. For example when a user enters

www.bbc.com into their web browser they will be automatically redirected to www.bbc.co.uk

Registry

The Registry is a storage area in Windows that keeps a record of nearly every Windows setting you are able to change, plus all the user-invisible settings that it needs to keep track of the operating system itself. For example, your desktop settings, such as the screen saver, screen color and background image are all stored here so that when you log in to Windows these will be applied as per the settings you previously defined. It should not be manually edited unless you are confident you know what you are doing.

Regular Expression

A regular expression (abbreviated as regexp, regex, or regxp, with plural forms regexps, regexes, or regexen) is a string that describes or matches a set of strings, according to certain syntax rules. Regular expressions are used by many text editors and utilities to search and manipulate bodies of text based on certain patterns.

Relational Database

A relational database is one whose structure is made up of numerous separate but linked tables. The key advantage of a relational database is that duplication of entries is significantly reduced or even eliminated, allowing for the efficient management of larger databases.

Removable Media

This is a type of storage device that may be physically inserted and removed from the computer, assuming the operating system allows it. Examples of this are: USB memory sticks, external hard drives, CDs, DVD's or tapes. These can be especially useful if simple offsite backups are required.

Reparse Point

Reparse points provide a way to extend the NTFS file system by adding extra information to the directory entry, so a file system filter can interpret how the operating system will treat the data. This allows the creation of junction points and NTFS symbolic links. They also can act as hard links, but aren't limited to point to files on the same volume: they can point to directories on any local volume.

Restore

A Restore copies files from a previous backup on the Destination back to the Source, such as in the event of a file being accidentally deleted. It is the opposite of a Backup therefore no files are copied from the Source to the Destination.

SAMBA

Samba is a piece of free software that allows file and print sharing between computers running Windows and computers running UNIX. It is widely used on NAS servers to provide compatibility with Windows clients (see also NAS).

Schedule (as in scheduling using Windows Task Scheduler)

SyncBack/SE/Pro interfaces with the Windows Task Scheduler to allow you to run profiles automatically at certain times, e.g. run a backup profile every day at 5am. On Windows XP you can access the task scheduler via the Start menu (All Programs > Accessories > System Tools > Scheduled Tasks).

Scripting

Scripting is a method of automating or tailoring a process by means of a sequence of commands contained within a script file. SyncBackPro may be controlled in this way, negating the need for manual user input. This is particularly useful where repetitive tasks are called for, or functionality needs to be extended or changed.

Secure Credit Card Transactions

When goods are bought on the Internet, there are two main systems which are used to transfer credit card

details securely: SSL (Secure Sockets Layer) and SET (Secure Electronic Transactions). These processes encrypt the transaction details in order for them to be sent securely over the internet and then decrypt them at the sales point.

Share

A network or file share is a resource on a computer network, typically allowing multiple computer users on the same network to have a centralized space on which to store files (documents, spreadsheets, etc) and share amongst each other.

Server

A server is a powerful, high-storage capacity computer or device on a network that contains and publishes resources, such as file shares and printers for users to utilize. Its only role is to serve and is not used as a traditional computer, thus keeping its responsiveness and availability high at all times.

Servlet

A servlet is an applet that runs on a server. The term usually refers to a Java applet that runs within a Web server environment. This is analogous to a Java applet that runs within a Web browser environment.

SFTP

Secure File Transfer Protocol

A secure version of FTP: the most widely-used method of downloading and uploading (getting and putting) files between two computers on the Internet. It makes use of the SSH protocol to secure the data (see also **FTP** and **FTPS**).

Shareware

Shareware is software distributed on the basis of an honor system. Most shareware is delivered free of charge, but the author usually requests that you pay a fee if you like the program and use it regularly or after a specified trial period.

S.M.A.R.T.

Self-Monitoring, Analysis, and Reporting Technology

This technology is built in to many hard disk drives and if monitored by software, provides an early warning of impending failure, based upon various status indicators.

Smart Synchronization

SyncBackSE/Pro uses this method to copy files in both directions, whilst keeping a history of where files were during the last synchronization. This allows for much finer control over what actions to take based on what has changed, and also allows it to detect changes such as the file only being modified in the source or destination (see also **Synchronize**).

SMTP

Simple Mail Transfer Protocol

The Internet protocol responsible for specifying how two mail systems interact and the format of control messages they exchange to transfer mail. Most ISP's will have one or more SMTP servers that receive and forward emails from other SMTP servers around the world. From a home user point of view they send email using SMTP and receive it using POP3.

Sparse file

A file containing large sections of data composed only of zeros, which is marked as such in the NTFS. The file system saves disk space by only allocating as many ranges on disk as are required to completely reconstruct the non-zero data. When an attempt is made to read in the non-allocated portions of the file (also known as holes), the file system automatically returns zeros to the caller.

SQL

Structured Query Language

SQL Server is Microsoft's relational database management system (RDBMS). It uses a client/server model whereby the data access logic is executed on the server. This is as opposed to a file-based database like Microsoft Access where queries are executed on the user's PC.

SSL

Secure Sockets Layer

SSL is a process encrypts the channel between a Web browser and Web server to ensure the privacy and reliability of data. It is used extensively in online banking and purchases.

Synchronize

The Synchronize operation is when files are copied to and from the source and destination. The aim of this process is to maintain identical copies of the data on both machines, regardless of which side the data changed. One of the caveats of using this method is the possibility of collisions (conflicts). For example if the same file is changed on both sides SyncBack will prompt the user for a decision, or it can be configured for an automated action for convenience. After synchronization, the source and destination should contain the same files and directories, i.e. are a mirror of each other.

TCP-IP

Transport Control Program/Internet protocol

TCP/IP is the well-defined and almost exclusive system of protocols used for communication over the Internet. Whereas the IP packet portion of the protocol is connectionless and only makes a best-effort attempt to communicate, TCP makes sure the packets have arrived and that the message is complete.

TIFF

Tagged image file format

A type of image file format. TIFF files may contain multiple images, and support a variety of color depths and may use lossless compression, making the file sizes a good deal bigger than the same with JPG formatting. There are several variations and cross-compatibility is sometimes a problem (see also **BMP** and **JPG**).

TLS

TLS is a cryptographic protocol that supersedes SSL. Like SSL it provide high security and data integrity for communications between computers (see also **SSL**).

Tray Icon

Tray icons appear in the Windows System Tray. This is the small bar at the bottom right-hand area of your screen. It is normally used to show the status of "utility" type programs running in the background. Usually, if you double-click or right-click them, they'll open up or bring up a menu. SyncBack can be minimized or 'closed' to appear in this area.

TrueCrypt

TrueCrypt is a proprietary software application used for real-time on-the-fly encryption (see also **EFS** and **Encryption**).

UDF

Universal Disk Format

UDF is a standardized common file system for all optical media, e.g. CDs and DVDs. The format is designed to make a common file system for read-only and re-writable optical media.

UAC

User Account Control

User Account Control is a security component in Windows Vista and newer. UAC elevates the user's account

privileges thus enabling them to perform common tasks as non-administrators (called standard users in Windows Vista) as administrators without having to switch users, log off, or use Run As.

UNC

Universal Naming Convention or **Uniform Naming Convention**

A UNC specifies a common syntax to describe the location of a network resource, such as a shared file, directory, or printer. The Windows UNC syntax is as follows:

`\\ComputerName\SharedFolder\Resource`

Additionally, one can connect a drive letter to the UNC for ease of access in Windows, but the raw UNC name can be used to access resources in the same location just as well.

Unicode

Unicode is a character encoding standard developed by the Unicode Consortium. The aim of the standard is to provide universal way of encoding characters of any language, regardless of the computer system, or platform, being used. The system achieves this by using two bytes (16 bits) for every character rather than the one byte (8 bits) as used by ASCII.

UNIX

An old but venerable operating system very widely used on medium sized, multi-user servers, and in top-of-the-range desktop computers. Its characteristics are reliability and efficient use of computer hardware but some variations have a somewhat arcane user command-line interface. Most modern versions now have a graphical user interface that allows mouse input, but it is still the domain of the technically proficient. Modern day incarnations now include Linux and all its desktop variants.

Upload

Uploading is the process of sending bulk information from your computer to another computer or server on the Internet. For example if you use a photo sharing website, you are uploading your files to them (see also **Download**).

URL

Universal Resource Locator

A URL is the technical term for 'Web site address'. This is usually the address of a website or document on the Web (e.g. <http://www.2brightsparks.com>)

Usenet

Usenet is short for *User's Network*. It is a collection of thousands of online bulletin boards residing on the Internet. Each bulletin board is arranged in a hierarchical fashion and contain discussion groups (or newsgroups) dedicated to a myriad of topics. Messages are posted and responded to by readers either as public or private emails.

User Interface (or UI)

That part of a computer program that controls interaction with the user. For example, in Windows, you are presented with a Start button, task bar, desktop, system tray and moveable mouse pointer. This whole experience is termed the User Interface. Another example is the 2BrightSparks series of products, which strive to give you the most attractive and useable User Interface possible.

Variables

In computer languages, a variable is simply a name assigned to a mathematical value, text, or multitude of other objects that exist within that environment. Windows has a set of predefined variables that the operating system uses to reference all sorts of values (type in *set* at a command prompt for an example). SyncBackSE/Pro use a range of variables that can be used in the Source and Destination settings for a profile when scripting a backup task.

Versioning

When this SyncBackSE/Pro feature is switched on, a backup version of a file is automatically created before it is moved, replaced or deleted. If one of these operations subsequently turned out to be a mistake then you can restore one of the previously saved versions.

Virus

A virus is an unwanted, malicious piece of software that is designed to reproduce itself and adversely affect data on your computer or its performance. New viruses are written and distributed around the Internet every day by unprincipled persons. If your computer is connected to the Internet you should always use Anti-Virus software and a firewall to combat these. Also ensure that it is up-to-date with your publisher's latest anti-virus definitions. You should also download and install the latest Microsoft critical updates as they are published. Use the Security Center in Control Panel to configure your system appropriately.

VSS

Volume Shadow Copy Service

Volume Shadow Copy Service is a background service in Windows XP and newer that provides a method of creating snapshots of files and directories at predefined points in time. The service runs at the block-level, not the file level which means it is able to backup open or locked files. These snapshots can then be used to restore data files and folders from a previous point in time. It is used by the Windows Backup utility and of course SyncBackSE/Pro.

WAV

A WAV file is an audio format file and denoted by the extension .wav. It typically contains an uncompressed PCM (Pulse Coded Modulation) audio bit stream unlike MP3, which has various stages of compression to achieve smaller file sizes relative to the raw version.

Web Host

This is company that rents out space on the Internet to allow you to place your website on it. They provide the hardware, servers, backbone connections, backup system etc. where your data is housed. They also make sure your site is available to site visitors at all times.

Webmaster

A webmaster is someone who manages a web site. They make decisions about its content, style and administration. Large sites may employ additional content directors or editors. On smaller sites, webmasters may make these decisions and accept news releases directly.

Web Site

A Web Site is a collection of web pages, graphical images, videos and other digital resources that are hosted on one or more web servers. It is usually (but not always - see Intranet) accessible via the Internet. The content is constructed into a web page (normally an HTML document) that the web server sends to the user's web browser for display. The pages of a website can usually be accessed from a common root URL called the homepage. The URL's of these pages are organized into a structured hierarchy, although the hyperlinks between them control the orderly navigation of the whole site.

WAN

Wide Area Network

A WAN is network which covers a large geographical area. A well-designed WAN has some form of redundancy built in to allow for link faults between, or with, the various nodes. In the event of such a fault, the traffic will be re-routed through another node, thus maintaining resilience. The best-known wide area network is the Internet.

WinRAR

WinRAR is a shareware file archiver and data compression utility. It is one of the few applications that is able to create native RAR archives, because of its proprietary encoding algorithm (see also **BWT**, **BZip2**, **Compression** and **ZIP**).

XCRC

The XCRC is a CRC (Cyclic Redundancy Check) algorithm to calculate the hash value of the file being copied. This value can be used to verify the integrity of the file transfers. If the CRC values of the local and remote file match, both files are considered equal and therefore the transfer is successful.

X-Window

UNIX has traditionally been a command-line, text-based operating system and can be daunting for novices. For people seeking an easier way to use UNIX, X-Windows is a standard for providing it with a windowed user interface, much in the same way of Microsoft Windows. The ability to divide and order the screen into windows is an important feature in the provision of a graphical user interface, as is the use of a pointing device. X-Windows provides both in an effort to ease the learning UNIX curve.

Zip Compression, Zip File

Zip Compression is a method of compressing one or more files and folders into a single file with the aim of reducing the overall file size. This file can then be emailed, transmitted by FTP or copied onto removable media for the recipient to unzip (uncompress) it. Text files, such as Word and Notepad are highly compressible, as are other Microsoft Office documents. Photos and binary files are less so, but will nearly always compress a certain amount. SyncBack is

able to backup data to and restore from a ZIP file. There are also a wide range of 3rd party tools that will unzip these files, ranging from WinZip to the command line PKUnzip (see also **BWT**, **Compression** and **BZip2**).

Legal Information

© 2003 – 2010 2BrightSparks Pte Ltd. All Rights Reserved.

No parts of this work may be reproduced in any form or by any means - graphic, electronic, or mechanical, including photocopying, recording, taping, or information storage and retrieval systems - without the written permission of 2BrightSparks Pte Ltd. Some portions of this glossary used sections from the open source encyclopedia <http://www.wikipedia.org/>

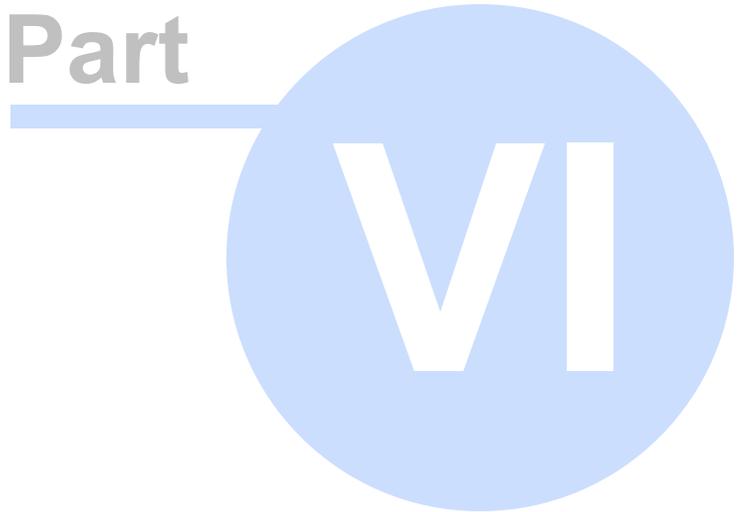
Products that are referred to in this document may be either trademarks and/or registered trademarks of the respective owners. The publisher and the author make no claim to these trademarks.

While every precaution has been taken in the preparation of this document, the publisher and the author assume no responsibility for errors or omissions, or for damages resulting from the use of information contained in this document or from the use of programs and source code that may accompany it. In no event shall the publisher and the author be liable for any loss of profit or any other commercial damage caused or alleged to have been caused directly or indirectly by this document.

May 2009

System Requirements

Part



6 System Requirements



System Requirements

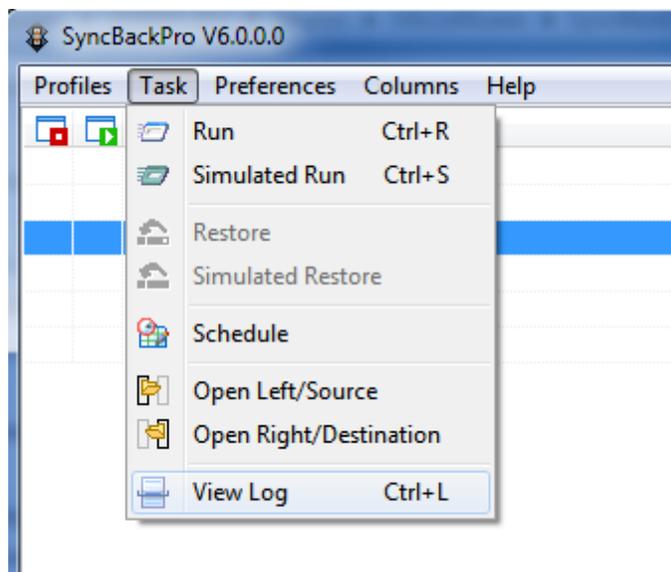
SyncBackSE requires Windows XP SP3, Windows 2003, Windows Vista, Windows 2008, Windows 7, or newer. Both 32-bit and 64-bit versions of Windows are supported.

Windows 95, 98, 98SE, ME, NT 4, and 2000 are not supported.

 To copy locked/open files, you **must** be running as a user who is a member of the Backup Operators group (administrator users are). Both 32-bit and 64-bit versions of Windows are supported. In addition, the file being copied (which is locked) must be on a local volume (i.e. a drive physically connected to the computer). If it is on a FAT32 volume then you must also have a local NTFS volume on an internal drive.

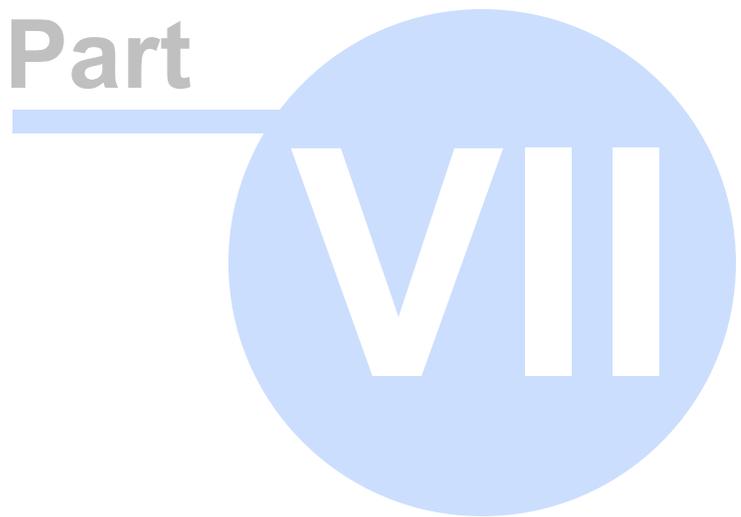
How to find out what system a drive is using

 SyncBackSE produces log reports which will show what File System you are currently using. To view a log report select a profile, then **View Log** from the Task menu - you may also use the shortcut keys "Ctrl" and "L" after selecting a profile.



Company Information

Part



7 Company Information



2BrightSparks Pte Ltd

Overview

2BrightSparks Pte Ltd was incorporated in 2004 with Michael J. Leaver, Mike de Sousa, and Richard N. Gascoigne at its helm. The team is now seven strong and growing as it continues to deliver high quality utility software solutions used by IT professionals, corporations, and non-profit organizations across the globe.



For more about the 2BrightSparks team go to:

<http://www.2brightsparks.com/about.html>

Company Website

www.2brightsparks.com

Sales and Support

Sales and Support are available by submitting a support ticket from our [Support Area](#).

Company Postal Address

2BrightSparks Pte Ltd
PSA Building
PO Box 364
Singapore 911143

Payment Processor

All payment transactions at 2BrightSparks are handled through the eSellerate payment system. At no time do we process or save customer payment card details at www.2BrightSparks.com.

If you would like to learn about eSellerate visit their site at:

http://www.esellerate.net/company_overview.asp

Index

- \$ -

\$root 205

- % -

%20 178
%22 178
%2A 178
%3A 178
%3C 178
%3E 178
%3F 178
%7C 178

- . -

.WAV 184

- [-

[unavailable] 212

- 0 -

0xFFFFF9? (Windows Task Scheduler return codes)
320

- 2 -

2BrightSparks Privacy Statement 367
2BrightSparks: Company Information 404
2BrightSparks: General Terms and Conditions 369

- 3 -

32-bit 400
3rd party components 377

- 6 -

64-bit 400

- 7 -

7-Zip 176

- A -

Access Denied 156, 161
Advanced Compression Settings 176
Advanced Email Log Settings 198
Advanced Email settings 193
Advanced FTP Settings 180
Advanced Network Settings 201
AES Encryption 175
Alarm Clock 96
AllDrives.vbs 318
Already compressed files 177
Alternatives 116
Amazon S3 205
An overview of Backing up 24
An overview of Mirroring 24
An overview of Restoring 24
An overview of Smart Synchronization 28
An overview of Synchronization 24
Archive attribute 122
Auto-close Settings 189
Auto-collapse 115
Automating SyncBackSE 32
Azure 205

- B -

Backing up for the first time 40
Backup Email 210
Backup scenarios 26
Backup to email 190
Bandwidth 161
Basic Operation 54
Blocked email attachments 193
Blu-ray 201
Bucket names 205
Burn 201
BWT 173

BZip2 173

- C -

Cannot open file: it does not appear to be a valid archive. 176
 CD 201
 CD/DVD Burner 201
 CD/DVD Burner Advanced Settings 204
 Centralized Management 358
 Changes 149
 Changing preferences 89
 Changing Simple Settings 116
 Click For Options 115
 Close programs 189
 Cloud 205
 Cloud Advanced Settings 207
 Cloud storage: choosing a service 205
 Collisions 48, 212
 Colours 89
 Columns 94
 Columns Main Menu 94
 Command Line Parameters 320
 Common Settings 106
 Company Information 404
 Compare Files 103
 Compare Options Attributes Settings 172
 Compare Options Date and Time Settings 171
 Compare Options Settings 169, 170
 Comparing Programs in SyncBackSE 103
 Comparison Programs 103
 Components used 377
 Compressed Settings 177
 Compression Encryption Settings 175
 Compression filter 177
 Compression Settings 173
 Configuring Easy Mode 112
 Constants 313
 Container names 205
 Copy and Delete Advanced Settings 161
 Copy and Delete Settings 156, 160
 Copy Delete Versioning 165
 Copy open and locked files requirements 400
 Copy settings 115
 Corrupted Zip 176
 CreateRestorePoint.vbs 318
 Creating a Group Profile 70
 Creating a Profile 58

Creating a Schedule 80
 Creating your first Profile 40
 Credits 377

- D -

Daylight Savings Time 171
 Debug Mode 89
 Decisions - Files 131
 Decisions, Folders Settings 153
 DecryptCopy.vbs 318
 Defaults 115
 desktop.ini 160
 Detect changes 149
 DFP_GET_VERSION failed: Incorrect function 195
 Dialogs in SyncBackSE 102
 Differences Window 48
 Differences Windows 169
 Differential 122
 Direct ordering and download 12
 Directories 160
 Distributing SyncBackSE 364
 Distribution 364
 Do not prompt me again 102
 Download Scripts 224
 Drive Failure 195
 DST 171
 DVD 201

- E -

Easy 115
 Easy Mode 110
 Easy Mode Configuration 112
 Eject 185
 EMail 190
 Email variables 330
 Empty folder on FTP server 180
 Encoding 193, 198
 EncryptCopy.vbs 318
 Encryption 175, 178
 Entering Serial Number 34
 Evaluating SyncBackSE 20
 Evaluation Mode 34
 Evaluation screen 20
 Event Log 96
 Example scripts 318

Exit Codes 320
Expert 115
Expert Mode Configuration 142
Explaining Backup for the new user 26
Explaining Synchronizing for the new user 28
Exploring SyncBackSE 108
Export Scripts 224
Exporting Profiles 57
Expression Filters 350
Extending SyncBackPro 224
ExtraInfo.vbs 318

- F -

Factory settings 115
Failed to rename to... 153
Fast Backup 122
Fast Backup versioning 165
File Collision Window 220
File Collisions 212
File Size 170
File Versioning 165
File Versions 165
Filter Settings 325
Filter Versions 165
Firewall 184
First Run 48
First Steps 34
First time user 40
Folders 160
Folders Settings 153
Format of text log file 104
Freeware Comparison 15
Frequently Asked Questions 7
FTP and Zip backup 122
FTP Firewall Settings 184
FTP Passive 184
FTP Port 180
FTP Proxy 183
FTP Settings 178
FTPS 180
Full backup 122

- G -

General Terms and Conditions 369
Global Options 96

Global Settings 96, 106
Glossary 380
Gmail 190
Good data procedures 24
Google Storage 205
Group Profiles 70
Group Profiles: Background Settings 70
Guide 7
Guide for new users 22
Guide to Smart Synchronization 136

- H -

Help 7
Help and information 7
Help for particular controls 54
Help on SyncBackSE 7
Helpful Internet links 7
Hibernate 96
History 147
HOST command 180
Hot key 148
How do I
 Get support for SyncBackSE 7
 Print the online documentation 7
 Start 20
How to buy SyncBackSE 12

- I -

IMAP 190
Import Profiles 320
Import Scripts 224, 320
Importing Profiles 57, 58
Importing SyncBackSE Version 3 Profiles 37
Incremental 122
IncVar.vbs 318
Insert 150
Installing Scripts 224
Introducing Easy Mode 110
Introduction 4
Invalid Windows filenames 178
ISO 201

- J -

Join the SyncBackSE community 7

- K -

KnowledgeBase and FAQ 7

- L -

Label destination 116
 Label source 116
 Legal Information 364
 Limit Time 151
 Linux server cannot detect changes 149
 LIST 180
 Location Scripts 236
 Log Email Settings 197
 Log Off Windows 148
 Log Settings 104, 195
 Login and Logout 148

- M -

Main Interface Scripts 229
 Main Program Window 55
 Making your first backup 40
 Management Service Settings 358
 MDTM 178
 MFMT 178
 Microsoft Azure 205
 Mirror 30
 Mirroring 30
 Miscellaneous Media Settings 185
 Miscellaneous Settings 184
 MODE Z 180

- N -

NAS drive cannot detect changes 149
 Network Settings 200
 New features and changes 16
 New User Guide 22
 New Version 96
 Notes 141
 Novell 156
 NTFS Compression Settings 177

- O -

Offline (SBM Service) 358
 OncePerDay.vbs 318
 Online support resources 7
 OnlyYesterday.vbs 318
 Open and Locked File Copying 329
 Options 89, 96
 Options: Fast Backup 122
 Orange cross on tray icon 96

- P -

Passive 184
 Password protect a profile 184
 Payment processor 12
 PDF Manual 7
 Performance 147
 Periodically 151
 Phone and Fax purchases 12
 Plug-ins 224
 POP3 190
 Preferences Main Menu 89
 Previous versions of Windows 400
 Printable Manual 7
 Priority 184
 Privacy Statement 367
 Problems with Serial Numbers 34
 Profile Configuration Scripts 233
 Profile History 147
 Profile notes 141
 Profile Setup 121
 Profile Types 58
 Profile: First Run 48
 Profiles 58
 Program Arguments 320
 Program Credits 377
 Program Options 89, 96
 Program Settings 96
 Programs After Settings 188
 Programs Before Settings 186
 Programs exiting 152
 Programs starting 152
 Progress Bar 107
 Protect a profile 184

- Q -

Question marks 173
Quick Start 20

- R -

Red cross on tray icon 96
Registry 330
Remote Management 358
Rescan 122
Restart Windows 148
Restore 31, 74
Restore Versions 165
Restoring a Backup 74
Restoring data 31
Resume from sleep or hibernation 96
Return Values 320
Right-Click Menu 55
Run After 188
Run Before 186
Run Group Profile 70
Run History 147
Running A Profile 67
Running SyncBackSE 32
Runtime Help 212
Runtime Scripts 249

- S -

S3 205
Safe copies and performance 161
Save as defaults 115
Say 184
SBHistory object 308
SBLocation object 265
SBM Console 358
SBM Service 358
SBProfile object 268
SBRunning object 272
SBSystem object 291
SBVariables object 303
Schedule 80
Scheduling and Background Tasks 32
Screen readers 108
Script base 227

Script constants 313
Script examples 318
Script Types 224
Scripting 210
Scripting Reference 224
Scripts 210
Search 7
Search Settings 114
Searchable Help File Index 7
Searching Settings 114
SendResultViaSMS.vbs 318
Settings in SyncBackSE 96
SFTP 180
SFTP Keys 178
Shared Settings 106
Shutdown Windows 148
Simple History 147
Simple Performance 147
Simple Settings 116
Simulated Restore 74
SITE UTIME 178
Skip Differences Windows 169
Sleep 96
SMART 195
Smart Synchronization 136
SMS 318
SMTP 190
Snapshot 161
Source and Destination 24
Spanned Zip 176
Speak 184
Split Zip 176
Start using SyncBackSE 20
StripZeros.vbs 318
Sub-directories and Files 117
Supporting Windows Versions 400
SyncBackSE V3: importing profiles 37
Synchronize 28
Synchronize scenarios 28
SysLog settings 96
System Requirements 400
System Restore 318

- T -

Technical Reference 224
Tell me about 16
 SyncBackSE 4

Text log file format 104
 The CD/DVD in drive X: is about to be completely
 erased before files are copied to it. 201
 The Differences Window 212
 The File Collision Window 220
 The FTP server does not support hashing 156
 The Main Window 55
 The Profile Setup Screen 121
 thumbs.db 160
 Time Limit 151
 Time scheduling SyncBackSE 80
 Translators names 377
 Tray Icon 96
 Troubleshooter 7
 TrueCrypt 169
 Tutorial on backup to email 190
 Tutorials 7
 Twitter 318

- U -

U3 variables 330
 UDF 201
 Uninstalling SyncBackSE 358
 UNIX filenames 178
 Update Check 96
 Update Version 16
 Upgrade to SyncBackPro 12
 Upload Profile to SBM Service 358
 User-defined notes 141
 User-defined variables 209
 Using FTP 178
 Using SyncBackSE 54
 Using SyncBackSE for the first time 48
 UTF8 176, 180

- V -

Variables (built-in) 330
 Variables (user-defined) 209
 Verify 156
 Version Comparison 15
 versioning 165
 Versions 165
 Virtual host 180
 Vista 400
 Visually Compare Files 103

Visually impaired 108
 Visually Impaired 89
 Volume Shadow Copy service 161
 VSS 161

- W -

WaitForFinish.vbs 318
 Welcome 4
 What about backup to a single Zip file on an FTP
 server? 122
 What about FTP and Fast Backups? 122
 What is a Profile? 40
 What is Smart Synchronization? 28
 What side effects are there with using Fast Backups?
 122
 What's new 16
 What's new? 16
 When - The Profile Setup Screen 121
 When, Changes 149
 When, Hot key 148
 When, Insert 150
 When, Login and Logout 148
 When, Periodically 151
 When, Programs 152
 When, Time Limit 151
 Windows 400
 Windows 2003 400
 Windows 2008 400
 Windows 2008 R2 400
 Windows 7 400
 Windows Event Log 96
 Windows Vista 400
 Windows XP 400

- X -

XML Compression 173

- Z -

Zip 173
 Zip filter 177
 Zip64 173

SyncBackSE V6

PDF User Guide

© 2011 2BrightSparks Pte Ltd.