PatchFactory User Manual

© 2002-2008 by AgenSoft. All rights reserved.
Web: www.agensoft.com
Date: 21.04.2008

Table of Contents

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

1. Overview

1.1. What is PatchFactory?

PatchFactory™ version 3.3

Copyright © 2002-2008 by AgenSoft. All rights reserved.

Contacting Agensoft

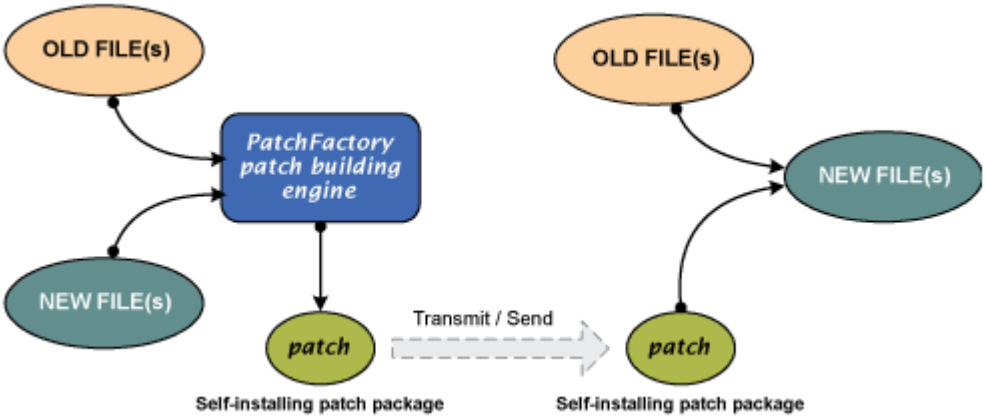
PatchFactory for Windows is a full-featured byte-level patching system for bandwidth-efficient software updating which is sold at a reasonable price comparing to systems of the similar class. PatchFactory provides a robust feature set that gives software developers the ability to tailor the update process to their specific needs.

Keeping customers on the most current version optimizes the performance of your product and significantly reduces your end-user support costs.
PatchFactory is a new and innovative software updating and delivery solution, ensuring that end-users always have the latest version of your software. When integrated into your application, PatchFactory performs almost any updating task, including synchronizing files, running an installer, updating your software, and even performing a custom action to keep your customers up-to-date and your technical support calls/inquiries to a minimum.

The patch-building engine is an innovative standard for producing version-to-version or cumulative software updates contained in one small, reliable package.
Generated patch packages are small size self-extracting executable update programs in a famous installer style with adjustable user-friendly interface (silent mode is also available). Patches can modify or replace the installed files to add new features, fix bugs, or add new security features.
Unlike setup files, which contain all files needed to run the software, the result update program only contains information concerning modifications (including changes in files/folders structure) made to old version software files relatively to new version files and has an easy-to-use interface with user-friendly dialogs.

Using PatchFactory you can create a complete yet compact update installation package for the application and make it available for download, or issue a media that contains the update package. You can also use PatchFactory to keep up-to-date any important data that must be updated very often and which freshness may be critical for your business.

The idea behind PatchFactory is simple: when modified file(s) must be transmitted, send only the changes (byte level differences) stored in one reliable self-installing update module rather than the entire software installation package. PatchFactory lossless byte-level compression is not content dependent, so it may be used whenever data is changed at one location and must be efficiently updated and/or archived at another.



NOTE: PatchFactory does not deal with any specific data structures, it operates with files as a binary data. Databases or files of other formats can be updated only as binary files (**warning:** database update can be implemented only if it is not changed on the end-user's machine).

Whatever you want to patch, update or upgrade - PatchFactory offers you the most comprehensive, professional, reliable, easy and safe to use, efficient and reasonably priced software update solution. Make your customers always feel up-to-date!

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

Key features of PatchFactory v3 :

- ***Patch building:***
  - Easily create self-applying patch file executables with customizable GUI interfaces
  - Powerful patch engine helping generate extremely small size patch packages.
  - Multiple version components support (with different target location on the end-user's machine).
  - Integrated compression technology.
  - Update of folders/files structure: addition of new/deletion of old folders or files, ignore missing files.
  - Effective patch building for executable EXE or DLL files.
  - Easy and visual navigation through images of old and new versions of your files.
  - Utilizing of the registry, INI files or a fixed path for smart file location.
  - Cumulative patch building to integrate several updates into one patch package.
  - Storage of difference building results in intermediate files helping significantly increase the speed of reiterated patch building.

- ***Patch applying module is based on the powerful InnoSetup engine which has the following features:***
  - Support for all Windows versions in use today -- Windows 95, 98, 2000, 2003, XP, Me, and NT 4.0 SP6.
  - Powerful integrated Pascal-based scripting engine providing lots of new possibilities to customize/enhance update Install or Uninstall functionality at run-time.
  - Silent install and silent uninstall.
  - Support for multilingual installs: *English (default), German, Spanish, Italian, French and Russian*.
  - Creation / changing of registry and .INI entries.
  - Run an external command or executable before, during, or after the patch process
  - Support for passworded and encrypted installs.
  - Optional license information and release notes presented to your end-user.
  - Apply Interface message management to customize messages presented to your end-user.

All features of patch module installation (except file update procedure itself) are implemented using standard InnoSetup resources.

Source codes of scripts used are available for free modification/enhancement.

- ***Additional features of Patch applying:***
  - Smart automatic version check to determine the installed version.
  - Ensure that no system is ever partially updated (automatic Rollback function during patch applying guarantees that end-user's system is successfully updated or not updated at all)
  - Capability of automatic backup of modified and deleted file(s).
  - Rollback / uninstall of the update after installation
  - Delayed patching support for locked files
  - Small size of base update modules (without patch data).
  - Log errors at apply time
  - Easy-to-use optimized interface that can be enhanced with the help of InnoSetup resources

System requirements :

- Minimum

- Microsoft Windows 98, NT4, 2000, ME, XP or above
- Microsoft Internet Explorer 5.0 or later
- Display resolution 800x600 hi-color (16 bits)
- Processor 486 and above, 32 Mb RAM

- Recommended

- Microsoft Windows 2000, XP or above
- Microsoft Internet Explorer 5.0 or later
- Display resolution 1024x768 hi-color (16 bits)
- Processor Pentium II and above, 128 Mb RAM

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

1.2. About Software Patching

Preface...

Onrush of the modern software development, and also wide Internet availability, have made a standard practice the frequent release of new versions of software programs. Prompt bugs / mistakes correction and feature adding are those major factors which oblige manufacturers to modify their software products and release updates on a frequent basis. And if the overall size of the distribution package of a software product is quite sizeable, then the necessity to download the full new version distribution package can become an expensive and tiresome procedure for end-users.

Most often, the size of brought changes from version to version is significantly smaller than the size of the full distribution package, so many developers could thought that it would be really a good idea to have an opportunity to implement software updating by delivering to end-users only those, presumably small changes by size which distinguish the new version from the version already installed on the end-user's machine.

Well, an attempt to reduce the update data size as much as possible is praiseworthy. However, how to make it really effective? In case of the text data, everything is clear. There are a lot of excellent algorithms for text files comparing. And moreover, the text data can be effectively compressed using well-known algorithms. But if you have to build an update package for executable files (exe, dll) then you'll see that those algorithms are completely useless to make something worthwhile.

Effective comparison of executable files or any other binary files requires absolutely other approaches. We'll use *"binary file"* to define files which structure is obviously unknown. Such files are considered as an arbitrary octet-byte stream.

Aspects of the software updating problem...

A problem of effective comparing of binary files is not the only one in the view of the software update task. The most important aspects which are really worthy of notice are automation of patch building and delivery of the update module to end-users.

Let's assume, that to some point of time the number of versions of your program has reached 21. It is obvious that a process of patch building for these 20 versions and the following publishing of those patches on the Internet will be quite complicated even if you have a files/catalogues comparing tool like PatchFactory v2.

Let's have a look what an end-user should perform to update a software product:

- define the installed version number;
- go to your web-site to make sure that a newer version exists;
- probably he would like to read the changes history for the newer version comparing to the version he already has;
- find in the published list the required version update;
- download and install the update package.

It is really impressing, isn't it?

And moreover active users of your program would like to receive new versions as soon as possible, having applied from their part to this process a minimum of efforts. And this wish is really reasonable!

PatchFactory v3 as a solution of software updating problem...

Projecting PatchFactory v3, we have set the developing of such product which provides automating of all stages of a software product, or even any other binary data update cycle (as fully as it can be possible) as our primary aim. Hereinafter, let's assume the certain set of operations including preparing of the update data, its publishing on the web-site and/or direct transfer to end-users, and also patch applying as an update cycle. Please refer to Concepts and definitions section of this manual.

Patch applying is a final stage of this update cycle which, actually, provides appearance of the required version of the software program or any other essential data on the end-users side.

There are no doubts that a task set is not so simple, as to find a completely universal decision is hardly possible. However, trying to offer a good enough decision covering the large class of subtasks of updating is quite feasible.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

1.3. What's new in this version?

PatchFactory v3 Version History

[**Legend**]
[+ **Added feature**]
[x **Improved/changed feature**]
[– **Bug fixed**]

==========

Version 3.3 (build 003) Date: 04-Oct-2005

==========

[x] InnoSetup compiler version updated up to 5.1.5

[*iss-script for patch applying module*]

[+] Spanish Language Pack (file Scripts\Languages\Spanish.isl)
[+] French Language Pack (file Scripts\Languages\French.isl)
[+] Italian Language Pack (file Scripts\Languages\Italian.isl)

[–] fixed the problem occurred when trying to replace a dll/exe in use under Win98
[–] miscellaneous bug fixes

==========

Version 3.2 (build 001) Date: 18-May-2005

==========

[+] New option 'Copy component images' added to the 'New Child Version' dialog. Setting this option results in that copying of the components images of the parent version (without copying of the image files).
[+] Additional patch option provides setting of the 'AppMutex' parameter value in default.iss

[–] Access violation when trying to edit 'OS Supported' or 'Language' version properties.
[–] Bug fixed concerned with loading of differences for version which is not child (thanks to Michael Boesch).
[–] Bug fixed which occurred when starting the program after closing the program in minimized state.

[x] If 'Comparing method' is set 'Ultimate' then the value of 'Memory usage' is ignored and is equal to 2GB.
[x] DevExpress components version updated (02 by internal number).
[x] Miscellaneous program interface improvements.

[*iss-script for patch applying module*]

[+] German Language Pack (file Scripts\Languages\German.isl)

==========

Version 3.1 (build 001) Date: 06-May-2005

==========

[+] New graphics of the program interface.
[+] "File Version" property added to Version File Editor and Version Differences Editor panels
[+] Warnings are displayed if compared versions contain files with equal file version numbers or old file version number is greater than the new one:
Warning #11: Version numbers of file '%s' in old %s and new %s versions are identical (%s).
Warning #12: File '%s': file version number (%s) in the old version %s is greater than file version number (%s) in the new version %s.
[+] Column "Modified" in file panels now indicates local last modified date of the file (was GMT)

[–] Miscellaneous program interface bug fixes:
dfbuild.exe: replacement of existing file was not performed

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

[*iss-script for patch applying module*]

[+] Automatic save / restore of file owner, group, access control and audit information (for NTFS file system only) while performing the backup / rollback procedures. Added patch option "Save/restore file security data" - using this option you can turn this automatic save / restore ON or OFF.
[+] New program main icon.
[x] Added/Replaced files: files with equal versions are replaced on the end-user's machine.
[x] Possible problems concerning changing of the access rights data while file updating (for NTFS file system only) are corrected.

==========

Version 3.0 RC1 (build 000) Date: 18-Mar-2005

==========

Major update. Beta.
PatchFactory v3 is greatly differs from versions 2.x.
You can read about features of PatchFactory v3 in **"What is PatchFactory"** section of this manual.
This is a beta version intended for beta-testing purposes only.

==========

Version 2.1.237 Date: 17-Jun-2004

==========

[+] Context help calling enabled from Patch Building Wizard dialogs
[+] Czech Language Pack (used to define the language of the output update program's dialogs)
[+] Italian Language Pack (/---/)
[+] French Language Pack (/---/)
[+] German Language Pack (/---/)
[+] Hungarian Language Pack (/---/)
[+] Swedish Language Pack (/---/)
[+] Spanish Language Pack (/---/)
[x] Help system is updated and converted to chm-format
[–] Patch applying SFP-modules: bug fixed concerned with reading patch applying parameters
[–] Patch applying SFP-modules: bug fixed occurred at patch applying for files with read-only attribute
[–] Patch applying SFP-modules: miscellaneous bug fixes

==========

Version 2.0.235 Date: 16-May-2003

==========

[+] Patch Building Wizard
[+] Added splash-window
[x] Patch applying module Default.sfp:
- Now it is a more powerful Update Installation Wizard with variety of professional interface and installation options;
- Author information included;
- License Agreement and Release Notes included;
- File "comment" is replaced by "notes.rtf";
- Size of the module Default.SFP is increased up to 35KB.

[–] Bug fixed that appeared at pressing "Close" in the "building patch" dialog for single and multiple files patches;
[–] Default.sfp Bug fixed that caused the error at patch applying if the full name of the patch file contained spaces.

==========

Version 1.1 Date: 14-Feb-2003

==========

First Public Release

==========

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

1.4. Patching Engine Efficiency

To evaluate the performance of byte-level patching engine used in PatchFactory to build a difference between two particular versions, we have prepared a comparison table (below) for well-known software products where you can see the original size of the new version files, the full original installation package (available from the author of the corresponding software product), the Update package size (from the author), the difference size (built with the help of PatchFactory), and the result compression ratio of the result difference size (prepared by means of PatchFactory) versus original installation package and uncompressed new version files of a software product.

Software Product	Versions	Uncompressed size		From the Author		Total DF size, bytes created with PatchFactory	DF compression factor vs.	
		old version files, bytes	new version files, bytes	Install package, bytes	Update package, bytes		Install package	Uncompressed new version
Nero Ahead Software, 151 files	**6.6.0.1 ⇒ 6.6.0.3**	70 798 812	**71 410 418**	29 974 779	NA	**3 248 356**	**89.2 %**	**95.5 %**
Adobe GoLive CS, 3 605 files	**7.0.0 ⇒ 7.0.2**	145 355 437	**145 478 267**	108 219 719	14 813 584	**2 527 744**	**98.0 %**	**98.3 %**
Adobe Acrobat CS, 819 files	**6.0.0 ⇒ 6.0.1**	143 978 136	**145 465 485**	138 860 743	15 919 168	**4 782 134**	**96.6 %**	**96.8 %**
Adobe Acrobat CS2, 1 942 files	**7.0.0 ⇒ 7.0.1**	458 286 849	**458 295 112**	211 754 411	4 905 816	**1 733 106**	**99.2 %**	**99.9 %**
Adobe Acrobat CS2, 2 017 files	**7.0.0 ⇒ 7.0.5**	458 286 849	**478 871 880**	#--#	26 598 760	**10 645 342**	**94.9 %**	**97.7 %**
K-Lite Mega Codec Pack, 147 files	**1.3.1 ⇒ 1.3.2**	27 233 341	**27 238 095**	25 195 329	NA	**189 289**	**99.2 %**	**99.4 %**
TheBat! Professional, 120 files	**3.50 ⇒ 3.51**	26 596 977	**26 703 858**	9 742 336	NA	**1 967 167**	**79.8 %**	**92.6 %**
TheBat! Professional, 120 files	**3.51 ⇒ 3.60**	26 703 858	**27 380 430**	9 967 104	NA	**2 201 632**	**77.9%**	**91.9 %**

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

1.5. Acknowledgments

PatchFactory includes the following sources, which are used with the permission of their authors for redistribution.

- -

bsdiff
Description: bsdiff and bspatch are tools for building and applying patches to binary files
Author: Colin Percival, Computing Lab, Oxford University
Web: http://www.daemonology.net/bsdiff/

- -

InnoSetup
Description: free installer for Windows programs
Author: Jordan Russell
Web: http://www.jrsoftware.org/

- -

bzip2 and libbzip2
Description: Freely available, patent free, high-quality data compressor
Author: Julian R. Seward
Web: http://sources.redhat.com/bzip2/

- -

libxml2
Description: XML C parser and toolkit developed for the Gnome project
Author: Daniel Veillard
Web: http://xmlsoft.org/

- -

synedit
Description: SynEdit is an advanced multi-line edit control, for Borland Delphi, Kylix and C++Builder
Author: Michael Hieke
Web: http://synedit.sourceforge.net/

- -

The Graphics32 Library
Description: Graphics32 is a set of functions, classes, components and controls designed for high-performance graphics programming.
Author: Alex Denissov
Web: http://graphics32.sf.net/

- -

And special thanks to **Fedor Mishin** (Fairdell Software) for his contribution to the development of the first public release of PatchFactory and for the idea of the first PatchFactory main icon image.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

2. Basics

2.1. Concepts and definitions

For the subsequent statement it is necessary to define some important terms which have specific meaning in the context of PatchFactory v3.

- **Product**

Let's assume a Product as a certain software program or simply any set of binary files which undergo some changes eventually. For example, we develop software product which is named PatchFactory. A set of attributes associated with this product represents essential information: global unique identifier (GUID), name of the product, manufacturer of this product and others.

- ***Product Parameters***

Each product has two main parameters: Product Name and Product GUID.
Product Name must form a valid file name, thus using of such symbols as < > : " / \ | * ? is not allowed.
Product GUID is generated automatically when new product is created and it is not changed later on.
Other parameters are optional and reserved for future use.

- **Version**

The version is the change of the product supposed to be published and fixed at some stage.

The obligatory attribute of the version is the identifier of the version which should be unique among all version identifiers for this product.

To define the number of the version the line identifier is used, for example: "1.2b" or " 3.7 (build 465) ". Restrictions on version number format and symbols are listed in the "Version Parameters" table.

To have an opportunity to present interrelation of versions among each other, let's introduce one more important version attribute - *Parent Version* or *previous version* which will designate the version the current version originated from. For each version there can be only one previous version (single inheritance). In other words, each version can have only one parent or not have it at all. At the same time, any version can be parental for any number of versions. Such approach allows us to present the whole set of versions of a software product as a *Versions Tree*. If versions have no previous one - they are so-called *Root Versions* or primary version. Any version can be a *Root Version*. Thus, the number of primary versions is not limited. From the above it follows that the Versions Tree does not contain a single root, so a product can be assumed as a root of the *Versions Tree*.

View available Version parameters...

- **Component**

All files included in a version are grouped into components.

Each version component may contain a set of folders and/or files. Components applying allows to easily allocate independent groups of folders/files which can have separate location in the tree of the file system on the end-user's machine, and probably can have an optional installation property, i.e. can be installed or not.

In general, the concept of components corresponds to the term accepted in the majority of modern install-generators.

The key parameters of a component are: *component identifier*, unique among all components of the given version; *name of this component* and *flag of optional installation* of this component.

To detect whether a component is installed on the end-user's machine or not, we use the ***key file method***. The idea of this method is that one file of each component is marked as the *Key*. Its presence in the system determines the fact that a component is installed or not.

Thus we have 3 main requirements for a component and a Key file:

- each component should have at least one *file*;
- each component should have one and only one *Key file*;
- *Key file* should exist during all "component's lifetime".

View available Component parameters...

- **Version Difference**

Let's name a set of the information which defines the difference between two versions, as a *Version Difference*. Besides difference in files, it can describe rules of update building and applying and other necessary data.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

- **Update module, Patch**

The program module which carries out the updating of the version of a software product, already installed on the end-user's side, up to some other version.
Update is performed on the basis of difference(s) between versions.

Hereinafter let's assume (when talking about a "patch" as a program object) that version which this patch belongs to, is new comparing to other versions of a software product and this new version contains at least one difference.

Using PatchFactory you can make patches of 2 types:

- **v2v-patch**
  Using this type of patches provides updating of only one version up to one another. Let's assume, that you have three versions of your software product: 1.0, 1.1 and 1.2. To update versions 1.0 and 1.1 up to version 1.2 using v2v-patches you should prepare two separate update modules.
- **Cumulative patch**
  This type of patches provides updating of a one of several specified versions up to one other version by means of only one patch module. For the example of the previous paragraph, a cumulative patch applying provides building of only one update module which can be used to update both versions 1.0 and 1.1 up to version 1.2.

View available Patch parameters...

- **Patch Publishing**

The patch prepared by software developer should be delivered to end-users. It is obvious, that taking into account its small size (comparing to the full new version distribution package), the most convenient way is to provide an access to these update modules via Internet. In an elemental case Patch Publishing can be implemented via placing of a patch-file on your company's web-server.

Disadvantages of such approach are obvious: both publishing and its delivering of a patch should be carried out "manually".

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

- **Update Server**

Let's name the Update Server as a server on which update modules are published.
Access methods to server and protocols used are insignificant at the current stage.

A schematical sketch of the overall update cycle with different types of patches is shown in figure 1.

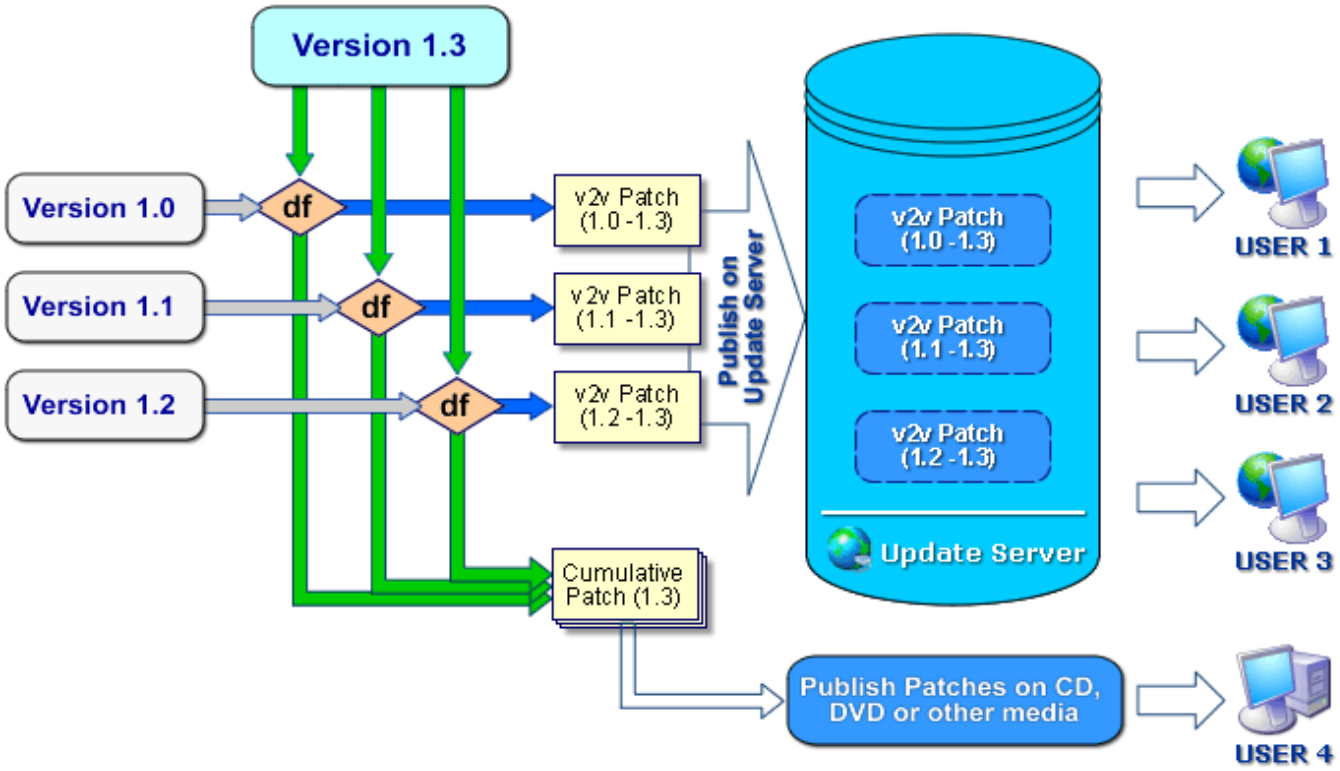


Figure 1. An example of an update cycle with different types of patches.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

2.1.1. Version Parameters

Main version parameters are listed in the table below.

Parameter	Description
Version Number	Can be changed after version is created. Version number must form a valid name of file, thus using of such symbols as < > : " / \ | * ? is not allowed.
Release Date / Time	Version release date/time
Version ID	Version Identifier. It is set automatically when a new version is created and cannot be changed later.
Parent Version	The Parent version number. This parameter can be changed later on.

See also: Version definition in terms of PatchFactory v3.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

2.1.2. Component Parameters

Main component parameters are listed in the table below.

Parameter	Description
Component ID	Version Identifier. It can be set automatically or manually when a new version is created and cannot be changed later on.
Required	Parameter which defines whether a component is required or not. Default value: YES.
Location	This parameter defines the record from Location Repository which is used to detect the location of this component on the end-user's machine.
Parent Version	The Parent version number. This parameter can be changed later on.
Dependencies	Optional parameter. List of components that affect the current one. It is supposed that a component can be installed in the system only if all other components it depends on are also installed. Currently "Dependencies" parameter is used only to check the correctness of setting the "Required" parameter for coherent components.

See also: Component definition in terms of PatchFactory v3.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

2.1.3. Patch Parameters

Main patch parameters are listed in the table below.

Parameter	Description
Patch Name	Patch name must form a valid name of file, thus using of such symbols as < > : " / \ | * ? is not allowed. It cannot be changed after patch creation.
Patch Type	Currently there are 2 types of patches: v-2-v and Cumulative 1. v-2-v. Building of this type of a patch leads to creation of independent patch modules, used to perform an update from one arbitrary version to another. Contents of old versions to be updated using this patch is defined by differences which exist in the new version. 2. Cumulative. This patch type is used to generate an update module capable to perform the update from several versions up to one another version. All necessary actions are taken to minimize the size of this patch module during the building process. Contents of old versions which can be updated using this patch module is defined by differences which exist in the new version.
Allow df	Flag, which defines whether difference between files with the same name is used during the update procedure. Value **YES (default)** allows such updating. If set to NO then full copies of new files are used if they differ from old version files.
Allow update on reboot	General flag. Provides replacing of files to be updated at system restart (often if such files are occupied by other processes). **YES by default.**
Allow uninstall on reboot	General flag. Provides replacing of files to be uninstalled/repaired at system restart (often if such files are occupied by other processes). **YES by default.**
Save/restore file security data	This option is useful only for NTFS file system and provides saving of file owner, group, access control and audit information. The patch module must have necessary privileges in order to use this facility. Processing of security data may decrease the speed of backup/restore operation, so, please, set this option to YES only if you understand its meaning and really need it, in most cases security processing is not required for home users. **YES by default.**
InnoSetup script	Name of the InnoSetup script file used to generate the update module. If only the name is specified (not the full path) then is it assumed that the script-file is stored in the "<PatchFactory Root Folder>\ Scripts" folder.
Wizard	
Uninstallable	Flag which defines whether backup information necessary to perform uninstall/rollback of patch is saved. If set to Yes, the backup copy of all replaced/modified files (for which backup is allowed) is saved and a record is added into Add/Remove programs list. If set to **No (default)** then rollback/uninstall of the patch after its successful applying is impossible. In this case backup copies of files are stored in the temporary system folder during patch applying only to provide rollback if an error occurred during patch applying to maintain an old version operating, and after patch is succesfully apllied they are deleted.
Show Backup page	Available values: **Yes (default)**, No. If set to No, then "Update Installation parameters" dialog page (asking an end-user to select whether to perform backup with further Rollback and to choose the backup folder) will not be displayed to an end-user during update installation.
Output	
Directory	Folder where the output update modules are saved to. If empty then it is

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

Parameter	Description
	assumed that result patch-files are saved in the "<Product Folder>/<New Version>/patches/<Patch Name>" folder.
File name Template	Template for update modules names. To prepare a template you can use the following specificators: {P} : Product Name. {U} : Patch Name. {O} : Old version name (empty string for Cumulative patch). {N} : New version name. {PID} : Product ID. {OID} : Old version ID (empty string for Cumulative patch). {NID} : New version ID. To include '{' character use '{{' instead.

See also: Patch definition in terms of PatchFactory v3.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

2.2. Binary Files Comparing

Preface...

Construction of a small-size patch module is closely related to the number of changes brought to the new version versus an old one, and also to the efficiency of the methods, capable to determine these changes and present them in a compact form. Well-known algorithms of text data comparing cannot help in comparing of the binary data when the structure of files is completely unknown in advance, and the nature of these changes can be poorly predicted. Attempt to apply LCS-search algorithms (Largest Common Subsequence) also have shown their incompetence. Comparison of binary files requires a special approach which is successfully implemented in PatchFactory3. In comparison with the previous versions of the program (1.x and 2.x) we have managed to greatly increase the quality and the speed of comparing process - presented in version 3 of PatchFactory.

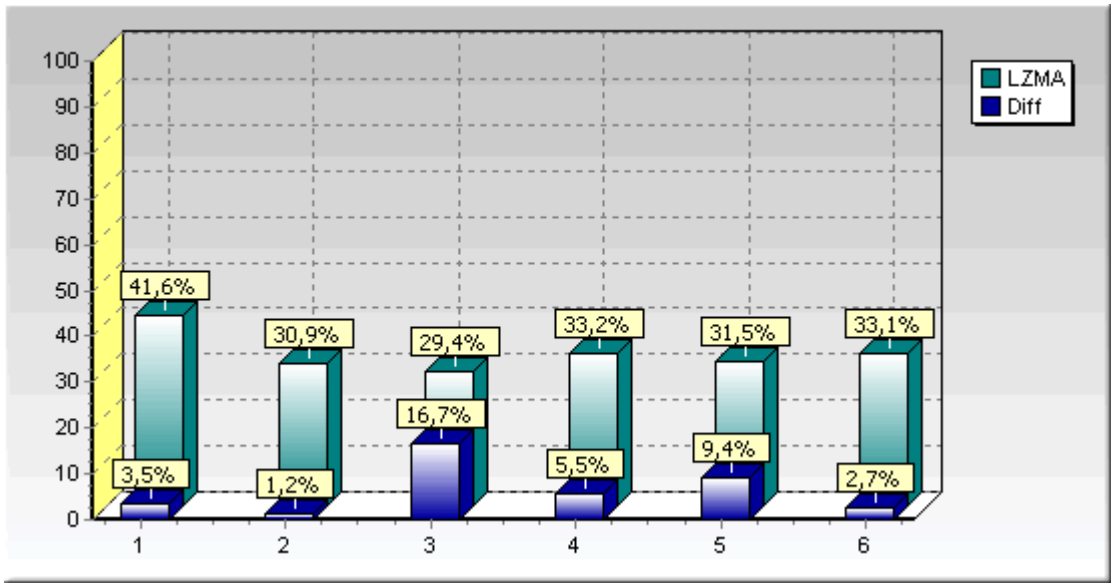
The basis of the algorithm is the ability to find concurrence in compared files at a level of octet-byte subsequences. Such byte-oriented nature of the used algorithm allows to make the efficiency of its work independent of a file format. The size of the resulting difference file, in the view of time spent on its construction is assumed as the main criterion of patch-building algorithm efficiency.

Key features of new algorithm:

- Significant improvement of algorithm quality parameters (generated difference files are smaller, work speed is greater);
- Special optimization providing considerable raise of comparison quality for executable modules (exe, dll);
- Comparing files with size up to 2^63 bytes;
- Flexible control under ratio "time/result" with the help of different comparing methods selection;
- Setting utilized resources constraints (memory size, process priority);
- Caching of comparing results. Storage of comparing results in intermediate files allows to significantly reduce the time of patch building.

The comparing algorithm is implemented as a separate console program dfbuild.exe In PatchFactory v3. Accepting as entrance parameters the file to be compared (old and new), dfbuild constructs a difference between them and saves it in a df-file.

The table below contains the examples (implemented for well-known software products main exe-files), illustrating the efficiency of df-files in comparison with LZMA-compression. Percentage values show the ratio of the LZMA-compressed file and df-file sizes to the size of a new file.



File	New file size, bytes	LZMA-compression		DF-file size	
		bytes	%	bytes	%

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

RAR.exe [3.40 beta3] – [3.42]	**297 472**	123 733	41.6	**10 291**	**3.5**
Winamp.exe [5.04] – [5.05]	**980 480**	303 275	30.9	**12 240**	**1.2**
TheBat.exe [3.4.0.933] – [3.5.0.1013]	**8 955 464**	2 632 520	29.4	**1 493 827**	**16.7**
HelpMan.exe [3.4] – [3.5] (Help&Manual)	**5 139 456**	1 704 393	32.2	**284 577**	**5.5**
SPECCTRA.exe [10.2] – [15.2] (SPECCTRA ShapeBased Automation Software	**13 307 978**	4 190 654	31.5	**1 245 494**	**9.4**
Fireworks.exe [7.0.0.288] – [7.0.2.295]	**14 696 448**	4 861 690	33.1	**400 728**	**2.7**

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

2.3. Byte-level Differencing

Byte-level differencing installs a software package using the byte-level differencing technology. The byte-level differencing function compares a software package to be installed (version package) with the base version (base package) that already exists on the target. On the source host a delta package is created for all the differences found between the version package and the base package, where both the base package and the version package reside on the source host. Because only the delta package, which is typically much smaller than either the version or base packages, is sent on the network, network traffic is considerably reduced. The new version of the software package is reconstructed on the target by applying the changes contained in the delta package to the base package.

- **How Byte-level Differencing Works**

Byte-level differencing compactly encodes a version of a file as a set of changes from a previous version. The delta file consists of any new files not in the base version and the differences between existing files in two versions. A differencing algorithm finds and outputs the changes made between two versions of the same file by locating common strings to be copied and unique strings to be added.

The process can be represented as follows:

```
Version File - Base File = Delta File
```

Reconstruction, the inverse operation, requires the base file and a delta file to rebuild a version:

```
Base File + Delta File = Version File
```

- **How Software Distribution Uses Byte-level Differencing**

Any distributed application that updates data frequently should take advantage of byte-level differencing to reduce the network traffic. Software Distribution can use byte-level differencing to distribute upgrades of software applications to a wide range of endpoints. To upgrade an already existing software package (base package) to a new version (version package), Software Distribution applies the differencing algorithm to each file contained in the version package that is delta compressible and that is found in the base package with the same fully qualified path.

To apply the delta installation, the base and the version packages must have the same nested structure. The above operation is performed on the source host and generates a delta package that is sent to the target.

On the target, Software Distribution uses files contained in the delta package to reconstruct each version file starting from the base file already installed on the target and applying to it the corresponding delta file contained in the delta package. If any of the installed files to be reconstructed are not found on the target, modified, or locked, the delta installation fails. Read-only files, if any, are overwritten.

The operations of loading and installing from a depository can also take advantage of the byte-level differencing technology because only the delta package is loaded on the depository for a subsequent byte-level differencing installation. The delta package can be also unloaded from a depository.

Byte-level differencing is the most efficient if the new version of a software package that you want to install contains few differences compared with the base version. If the difference is very significant, it can be more efficient to reinstall the new version.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

2.4. Location Repository

Each Component of the Version can have its own location in the file-system hierarchy on the end-user's system. To determine where exactly the particular Component should be placed PatchFactory uses a special table called the **Location Repository**. This table is the integral part of the product description (i.e. associated with each product), and should be filled by user.
Each item of the Location Repository is a named record. The parameters of each record define the way the full path to each component location is determined on the end-user's machine.

The "Location" property of a Component defines which record from the Location Repository is used to detect this Component location.

Currently the following types of records are available:

1. Registry

The full path to component's files location is taken from one of the Registry keys.

Parameter	Possible values	Description
Root	HKEY_CLASSES_ROOT, HKEY_CURRENT_USER, **HKEY_LOCAL_MACHINE,** HKEY_USERS, HKEY_CURRENT_CONFIG	Root registry key.
Subkey	<string>	Subkey path.
Value	<string>	Registry value.
Additional	<string>	Additional part of the Path.
Default value	<string>	Default value. Used if Path value can not be taken from the Registry (specified Registry value does not exist, access is denied, etc.).

2. INI-file

The full path to component's files location is taken from ini-file.

Parameter	Possible values	Description
File	<string>	Name of INI-file.
Section	<string>	Name of section inside INI-file.
Key	<string>	Name of a key within section.
Additional	<string>	Additional part of the Path.
Default value	<string>	Default value. Used if the Path value cannot be taken from INI-file (INI-file cannot be found, specified section/value does not exist, etc.).

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3. Special folder

The full path to component's files location is defined by one of the special folders.

Parameter	Possible values	Description
Folder	[Program Files]	The path of the system's Program Files directory, typically "C:\Program Files".
	[Common Files]	The path of the system's Common Files directory, typically "C:\Program Files\Common Files".
	[Windows]	The system's Windows directory.
	[System]	The system's Windows System directory (System32 on Windows NT platforms).
	[System Drive]	The drive letter where Windows is installed, typically "C:". On Windows NT platforms, this directory constant is equivalent to the SystemDrive environment variable.
	[Fonts]	Fonts directory. Normally named "FONTS" under the Windows directory.
Additional	<string>	Additional part of the Path.

4. Environment variable

The full path to component's files location is defined by an environment variable.

Parameter	Possible values	Description
Variable name	<string>	Name of environment variable.
Additional	<string>	Additional part of the Path.
Default value	<string>	Default value. Used if specified environment variable doesn't exist.

5. Fixed path

The full path to component's files location

Parameter	Possible values	Description
Path	<string>	Full path to component's files location.

6. Location ref

The full path to component's files location is defined by another item of the Location Repository table.

Parameter	Possible values	Description
Based on location	<reference>	Reference to another item of the Location Repository table.
Additional	<string>	Additional part of the Path.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

2.5. Installed Version Detection

Version detection method...

There are different methods to detect which version of a software product is installed on the end-user's computer. For example, you can store version number in the system Registry or obtain it from the resources of one of executable files. It is impossible to list all the variety of available methods, therefore we have implemented just one, which in our opinion is the most universal method, and will satisfy the overwhelming majority of PatchFactory's users. We have named this method as the **version key method**. But if its usage for any reason is impossible, there is always an opportunity to use a custom detection method.

The idea of version keys method is based on the assumption that it is possible to observe changes both in structure of files, and in their content from version to version of a software product. The number of such files which are constantly being changed from version to version is usually insignificant.

To perform the accurate identification of the version PatchFactory uses the key defined by the set of version-key files names and the MD5-values of their content. MD5 is used to replace a content of a file, thus, to provide a small size of a version key with a high probability of its uniqueness.
Such files which participate in the version key composition let's name ***version key files*** or just ***version keys***.

A set of files which form the *version key*, must be defined by user.

There are two main requirements for version key files:

- none of those files can be changed / deleted on the end-user's side;
- version key files must be certainly present on the end-user's machine to correctly detect the installed version, therefore version keys must be specified only inside the required components, or they will be ignored;

And besides try to minimize the number of version key files.

NOTE: As the most simple and at the same time the most reliable method to provide the presence of unique file in each version, we suggest placing a file named "vernum", for example, which contains the number of the current version installed inside one of the required components. So if you define the file "vernum" as a version key file you may be sure that problems with installed version detection will not arise anymore.

Hereinafter we suppose that only one version of a software product to be updated is installed. In some case it is also possible to detect the installed version using the *version key method* if several versions are installed on the same computer, though the correctness in this case is dependent on how the version keys are defined and what folder they are stored in.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3. User interface

3.1. Overview

This topic briefly describes the main elements of the PatchFactory interface.

PatchFactory v3 main window has the following menus: **File**, **View**, **Group**, **Product**, **Version**, **Component(s)**, **Difference(s)**, **Patch(es)** and **Help**.
File, ***View***, and ***Help*** menus are always shown while ***Group***, ***Product***, ***Version***, ***Component(s)***, ***Difference(s)***, ***Patch(es)*** - only when the appropriate node is selected within the Product Tree.

Also **Version Files Editor** and **Version Difference Editor** have their own menus.

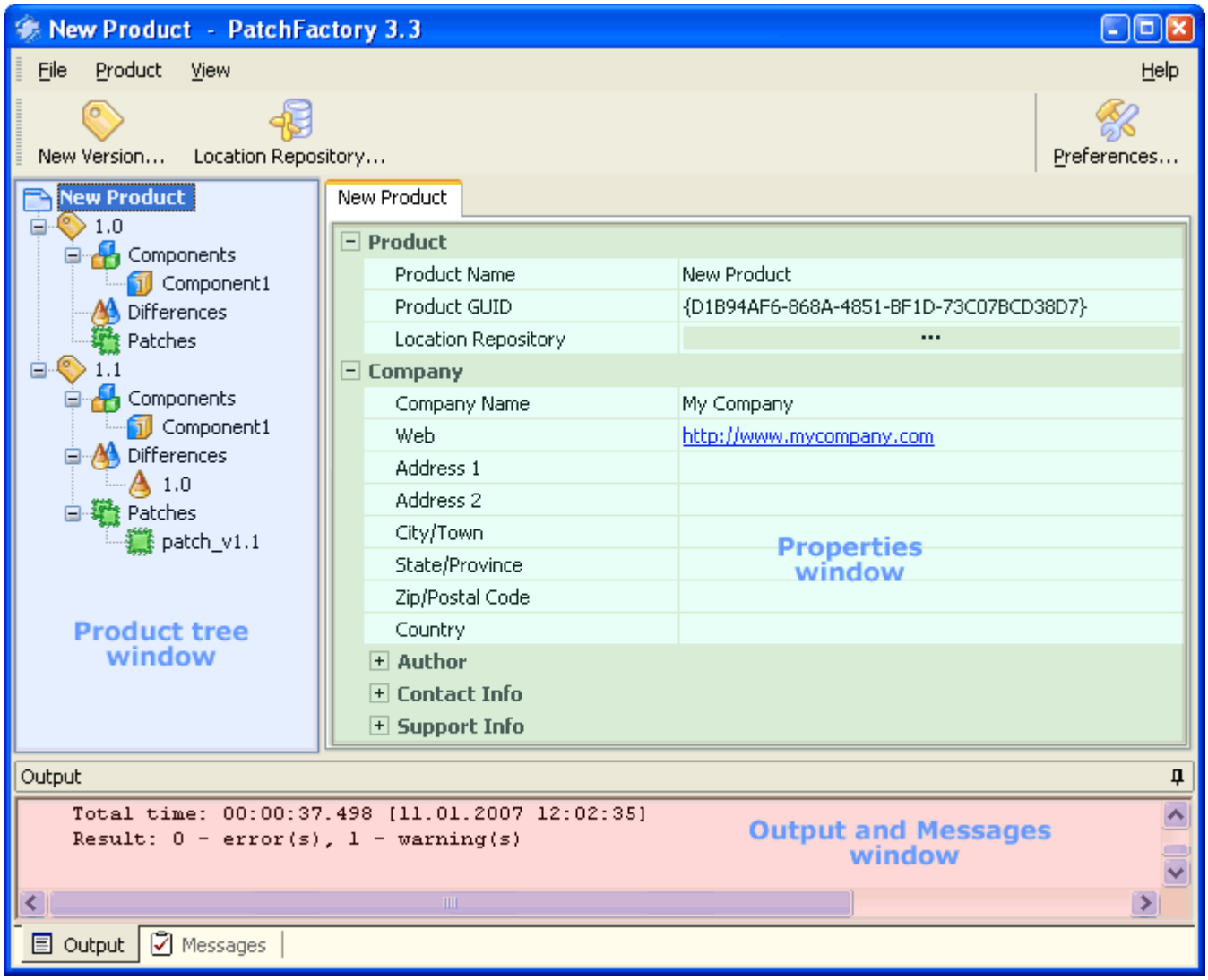
Select the appropriate topic to read about commands of a particular menu.

The next interface item is the control panel. It is placed below menus. You may use control panel buttons to:

- open/save/add Groups (pfg-files), Products (pfp-files), or Versions (pfv-files);
- compile/build differences and patches;
- add files to version components;
- view/edit differences;
- view/edit global program Preferences;

Main program window consists of four subwindows.
They are **Product Tree** window, **Properties** window, **Output** log window, and **Messages** window.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



- **Product Tree** window is used to navigate through Groups of software products, software products, versions of software products, their differences and patches.

Available actions with tree elements can be made if you right-click with your mouse on the appropriate element.

- **Properties** window is used to view / edit properties of the corresponding Product Tree element, such as *Product*, *Version*, *Difference*, or *Patch*.

Select the appropriate node within the Product Tree to view/modify properties of tree elements .

You can also switch between properties of available products within Group using tabstops at the head of the Properties window.

- **Output** log window is used to view output log of performed operations.

- **Messages** window is used to view output messages such as warnings or errors.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3.2. Keyboard layout

Key	Action
Project	
Ctrl+N	Create new software product.
Ctrl+W	Close current software product.
Ctrl+O	Open existing software product.
View	
Ctrl+Alt+O	Output
Ctrl+Alt+M	Messages
Alt+P	Preferences
Product Tree	
Shift+Ctrl+Up	Move selected node Up.
Shift+Ctrl+Down	Move selected node Down.
Shift+Ctrl+Right	Move selected node Inside.
Shift+Ctrl+Left	Move selected node Outside.
Version Files Editor	
Ctrl+S	Save selected version image.
Ctrl+Q	Calculate occupied space.
F5	Refresh active window.
Shift+F5	Refresh all windows.
Ctrl+A	Select All files and folders within active Version Image window.
Ctrl+D	Deselect All files and folders within active Version Image window.
Version Difference Editor	
Ctrl+I	Enable synchronous selection of files and folders in old and new version image windows.
Help	
F1	View context Help contents.
Alt+F1	About PatchFactory

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3.3. Menu

3.3.1. File menu

Command	Shortcut	Function
New >		
Product	Ctrl+N	Create new software product.
Group	Shift+Ctrl+N	Create new group of software products.
Open...	Ctrl+O	Open existing software product / product group.
Save Group '...' As ..		Save active Group to another file / location.
Close Product "..."		Close active Product.
Close All		Close all Groups and Products.
Recent Products >		Display recent software products list.
Recent Groups >		Display recent product groups list.
Exit	Alt+F4	Exit PatchFactory.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

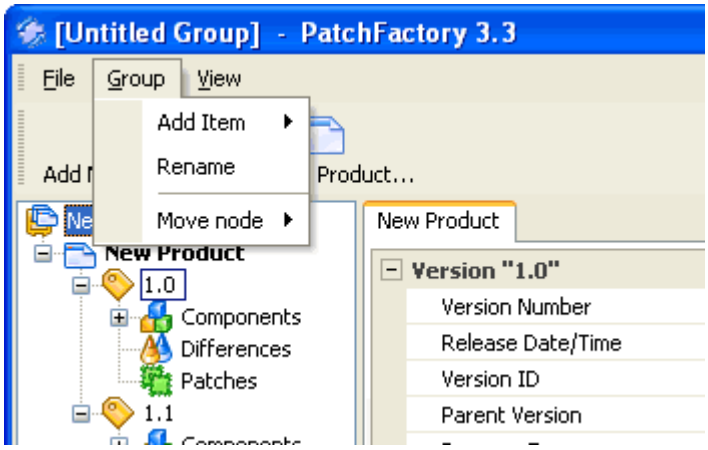
3.3.2. View menu

Command	Shortcut	Function
Output	Ctrl+Alt+O	Show / Hide the Output window.
Messages	Ctrl+Alt+M	Show / Hide the Messages window.
Preferences	Alt+P	Invoke General Preferences Dialog.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3.3.3. Group menu

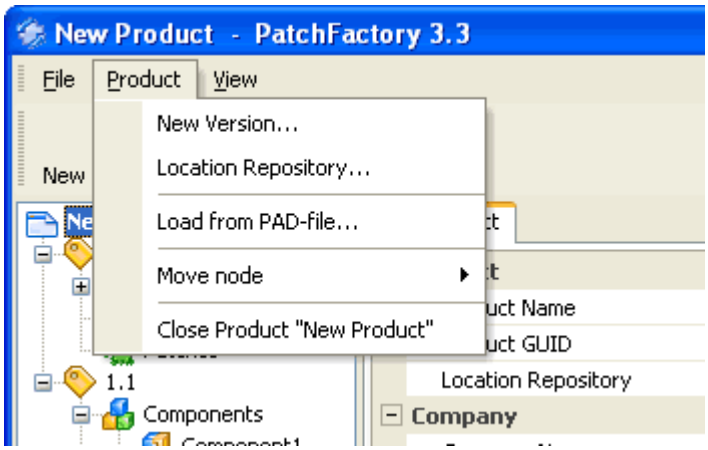
Command	Shortcut	Function
Add item >		
Add New Product...		Add New Product to the selected Group.
Add Existing Product...		Add Existing Product to the selected Group.
Add New Group		Add New Group of Software Products.
Rename		Rename selected Group.
Move node >		
Move Up	Shift+Ctrl+Up	Move selected Group node Up.
Move Down	Shift+Ctrl+Down	Move selected Group node Down.
Move In	Shift+Ctrl+Right	Move selected Group inside.
Move Out	Shift+Ctrl+Left	Move selected Group outside.



© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3.3.4. Product menu

Command	Shortcut	Function
New Version...		Add new Version to the selected Product.
Location Repository...		Invoke Location Repository dialog.
Load from PAD-file...		Import Product information from PAD-file*.
Move node >		
Move Up	Shift+Ctrl+Up	Move selected Product node Up.
Move Down	Shift+Ctrl+Down	Move selected Product node Down.
Move In	Shift+Ctrl+Right	Move selected Group inside.
Move Out	Shift+Ctrl+Left	Move selected Group outside.
Close Product "..."		Close selected Product.
Delete		Delete selected Product.



© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3.3.5. *Version menu*

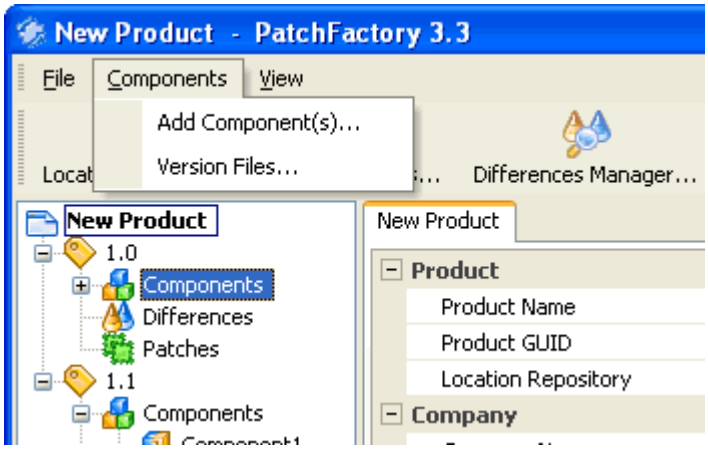
Command	Shortcut	Function
New Child Version...		Add new Child Version.
Add Component(s)...		Add component(s) to the selected Version.
Version files...		Invoke "Version Files" dialog.
Differences manager...		Invoke "Differences Manager" dialog.
Move node >		
Move Up	Shift+Ctrl+Up	Move selected Version node Up.
Move Down	Shift+Ctrl+Down	Move selected Version node Down.
Lock / Unlock		Lock / Unlock selected Version.
Delete		Delete selected Version.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3.3.6. *Component(s) menu*

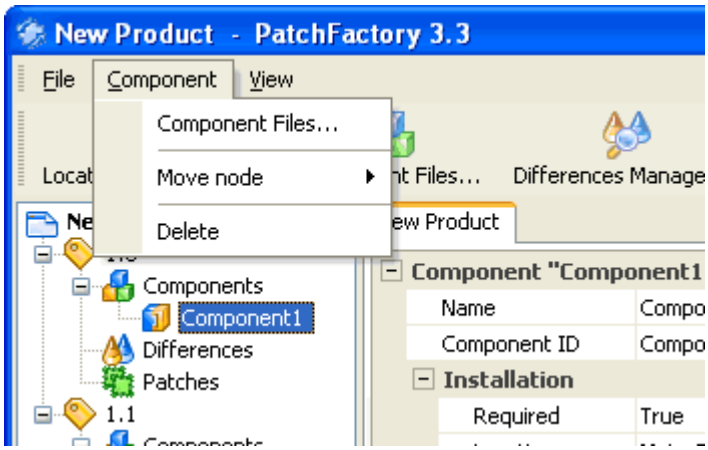
- **Components menu:**

Command	Shortcut	Function
Add Component(s)...		Add component(s) to the selected Version.
Version files...		Invoke "Version Files" dialog.



- **Component menu:**

Command	Shortcut	Function
Component files...		Invoke "Version Files" dialog with selected component as active.
Move node >		
Move Up	Shift+Ctrl+Up	Move selected Component node Up.
Move Down	Shift+Ctrl+Down	Move selected Component node Down.
Delete		Delete selected Component.

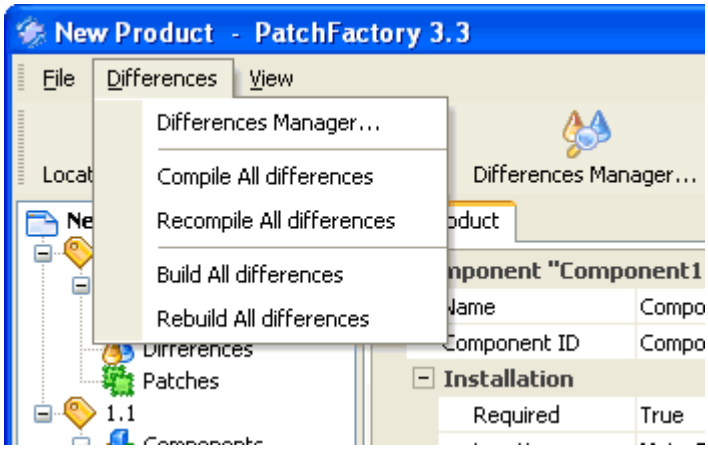


© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3.3.7. *Difference(s) menu*

- **Differences menu:**

Command	Shortcut	Function
Differences Manager...		Invoke "Differences Manager" dialog.
Compile All differences		Compile All Differences of the current Version.
Recompile All differences		Recompile All Differences of the current Version.
Build All differences		Build All Differences of the current Version.
Rebuild All differences		Rebuild All Differences of the current Version.

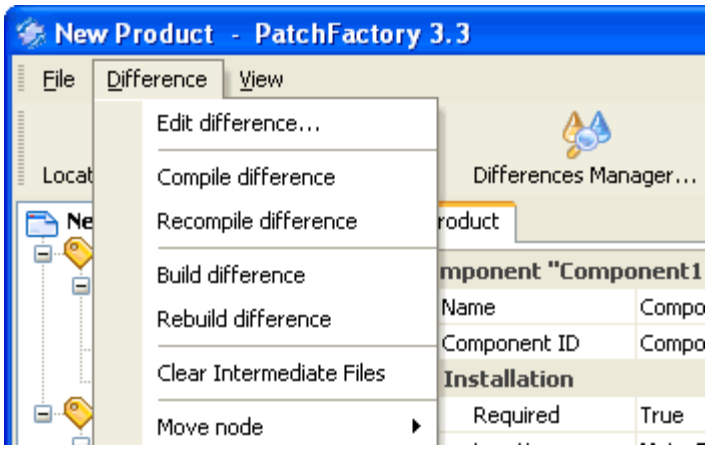


- **Difference menu:**

Command	Shortcut	Function
Edit difference...		Invoke "Version Difference" dialog.
Compile difference...		Compile selected Difference.
Recompile difference...		Recompile selected Difference.
Build difference...		Build selected Difference.
Rebuild difference...		Rebuild selected Difference.
Clear Intermediate Files		Clear all intermediate files concerned with selected Difference.
Move node >		
Move Up	Shift+Ctrl+Up	Move selected Difference node Up.
Move Down	Shift+Ctrl+Down	Move selected Difference node Down.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

Command	Shortcut	Function
Delete		Delete selected Difference.

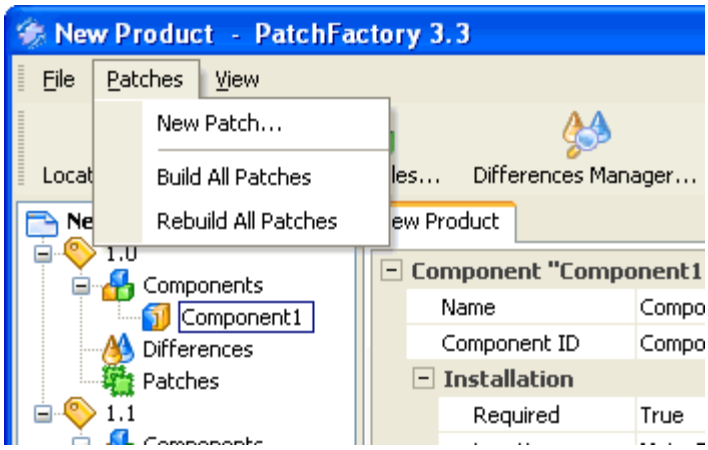


© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3.3.8. *Patch(es) menu*

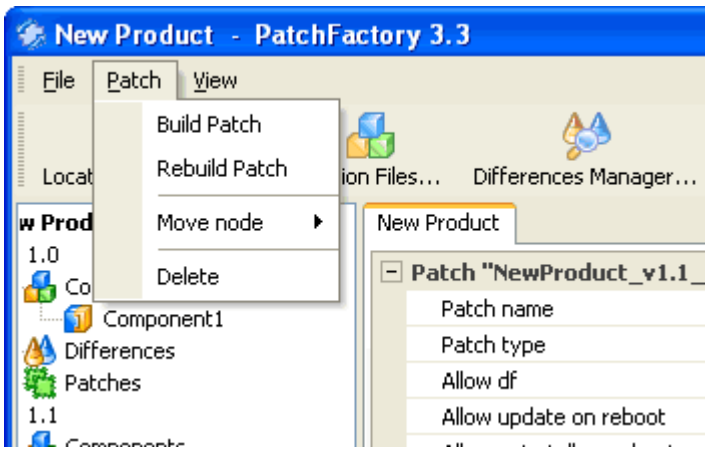
- **Patches menu:**

Command	Shortcut	Function
New Patch...		Invoke "New Patch" dialog.
Build All Patches		Build All Patch modules of the current Version.
Rebuild All Patches		Rebuild All Patch modules of the current Version.



- **Patch menu:**

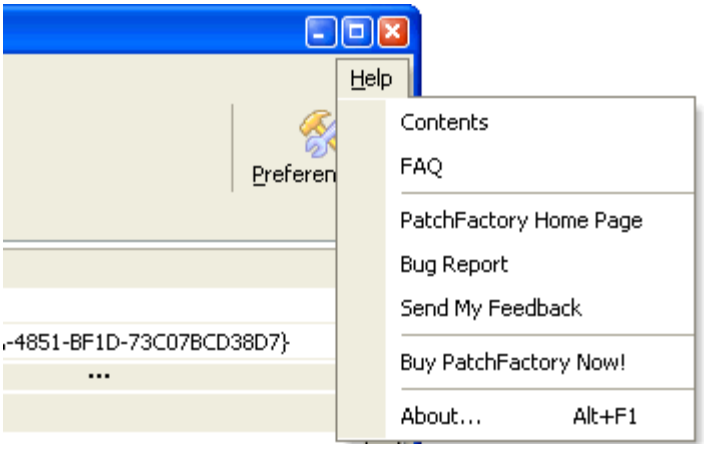
Command	Shortcut	Function
Build Patch		Build selected Patch module.
Rebuild Patch		Rebuild selected Patch module.
Move node >		
Move Up	Shift+Ctrl+Up	Move selected Patch node Up.
Move Down	Shift+Ctrl+Down	Move selected Patch node Down.
Delete		Delete selected Patch.



© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

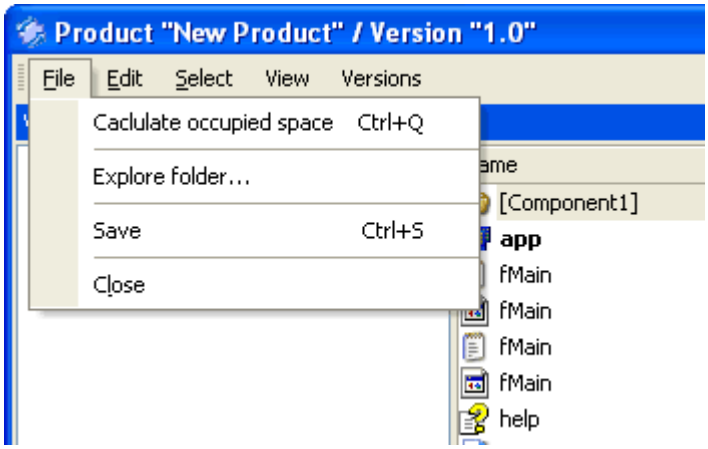
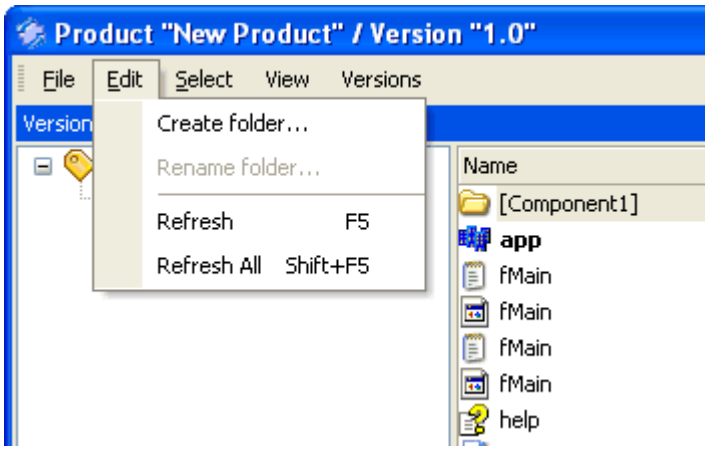
3.3.9. Help menu

Command	Shortcut	Function
Contents	F1	View context Help contents.
Getting Started		View Getting Started information.
FAQs		Frequently Asked Questions
Flash Tutorial		View local copy of PatchFactory interactive Flash-based Tutorial.
Ordering Information		How to register
PatchFactory Homepage		Visit PatchFactory Home page http://www.agensoft.com
PatchFactory Support Forums		Visit PatchFactory Support Forums http://www.agensoft.com/forums
Support Information		Update and Support
Contact Support Team		Send your feedback, bug-report or other inquiry using the online email form (guarantees that your email will reach our Support Team)
Buy PatchFactory		Buy PatchFactory Online http://www.agensoft.com/order.html (displayed only in unregistered version).
About	Alt+F1	Display the dialog box containing version & copyright information.

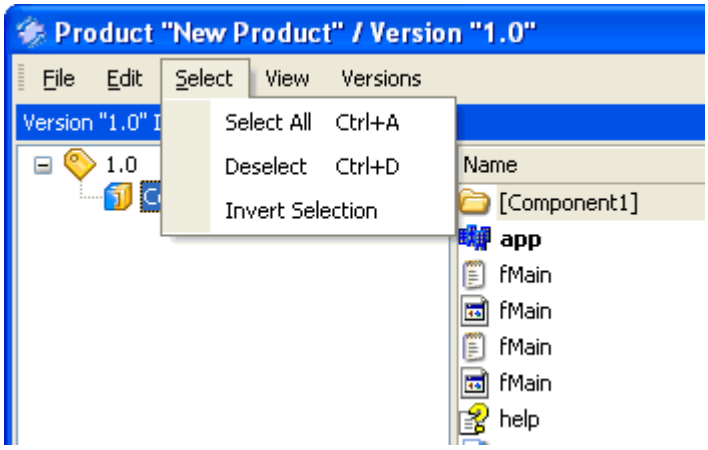
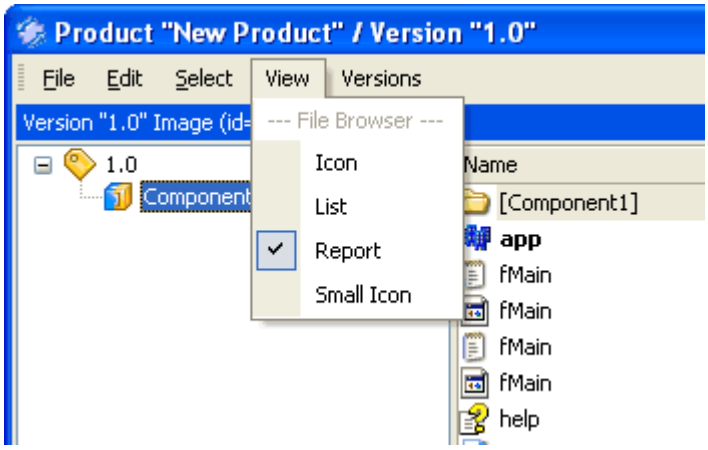


© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3.3.10. Version Files Editor menus

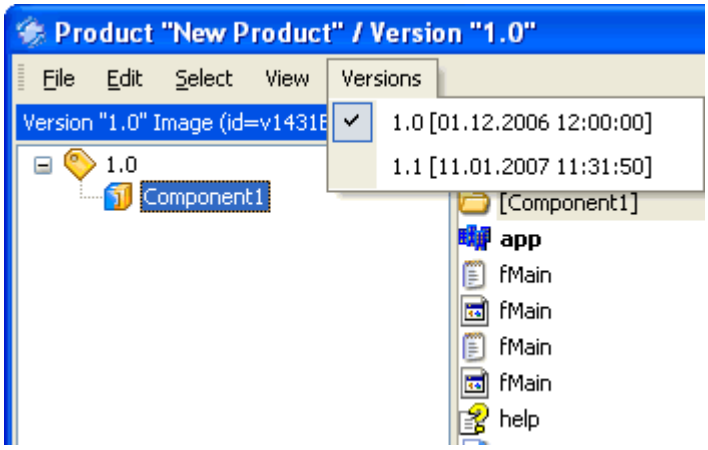
Command	Shortcut	Function
File >		
Calculate Occupied space	Ctrl+Q	Calculate Occupied space of the selected folder or Component.
Explore folder		Open selected folder or Component in a new Explorer window.
Properties		Display selected file/folder properties.
Save	Ctrl+S	Save selected Version image.
Close		Close Version File Editor window.
		
Edit >		
Create folder		Create folder in the selected Component.
Rename folder		Rename folder in the selected Component.
Set as Key		Set selected file as **Key**.
Refresh	F5	Refresh selected window.
Refresh All	Shift+F5	Refresh All windows.
		
Select >		
Select All	Ctrl+A	Select All files and folders within active Version Image window.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

Command	Shortcut	Function
Deselect	Ctrl+D	Deselect All files and folders within active Version Image window.
Invert Selection		Invert file selection.
		
View >		This menu is used for File Browser window only.
Icon		Display files / folder in File Browser window as Icons.
List		Display files / folder in File Browser window as a List.
Report		Display files / folder in File Browser window as a Detailed List.
Small Icon		Display files / folder in File Browser window as Small Icons.
		
Versions >		Here you can select which Version Files you want to view/add.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

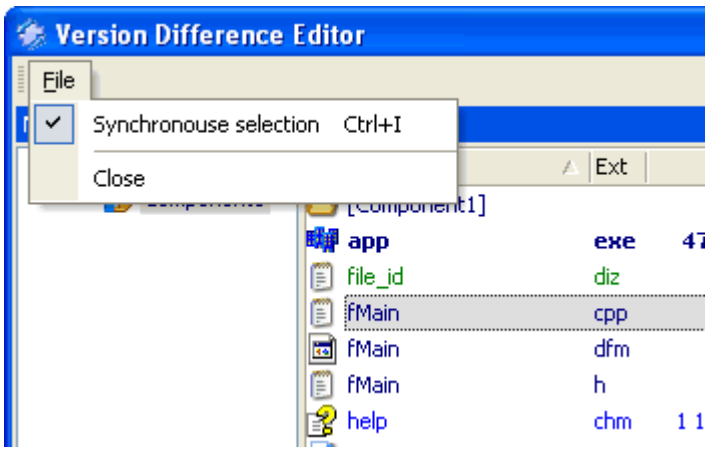
Command	Shortcut	Function



© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3.3.11. Version Difference Editor menus

Command	Shortcut	Function
File		
Synchronous selection	Ctrl+I	Enable synchronous selection of files and folders in old and new version image windows.
Close		Close Version Difference Editor window.

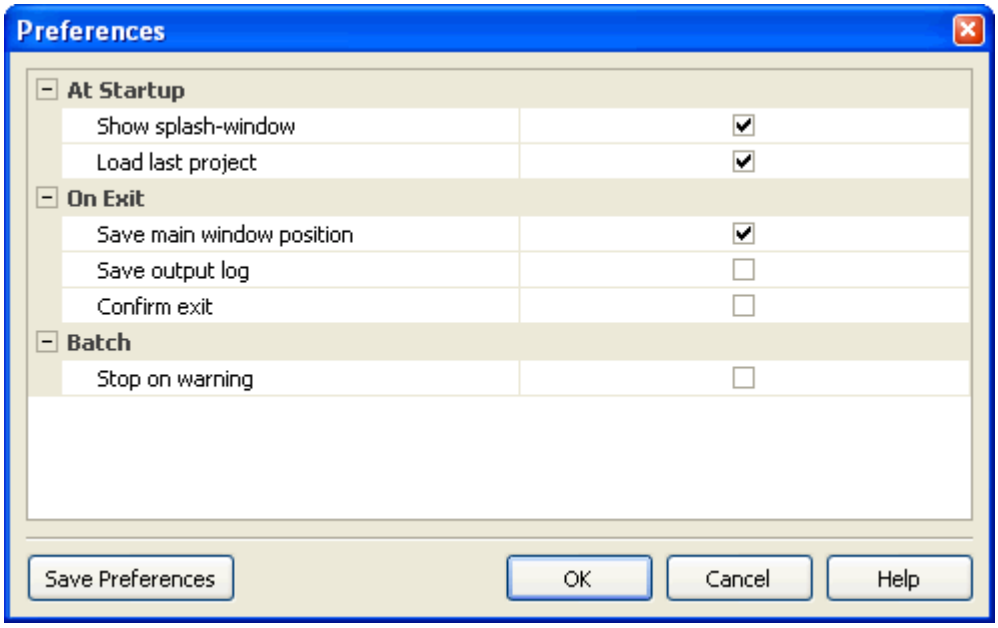


© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

3.4. Preferences

By pressing Alt+P or selecting menu <View/Preferences...> you can invoke the "Preferences" dialog.

Here you can configure a set of different program options:



Dialog options:

- **At Startup:**
  - **Show splash-window**
    Select this option if you want splash-window to be displayed at PatchFactory startup.
  - **Load last project**
    Select this option if you want to automatically load the project you last worked on.
- **On Exit:**
  - **Save main window position**
    Select this option if you want to save main window position on program exit.
  - **Save output log**
    Select this option if you want to save output log into file (which will automatically loaded at next startup) on program exit.
  - **Confirm exit**
    Select this option if you want to be asked for confirmation before program exit.
- **Batch:**
  - **Stop on warning**
    Select this option if you want to stop job processing on any warning.
    If this option is not selected - then warnings are collected and displayed in the *Output* and *Messages* subwindows.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4. Operation

4.1. Getting Started

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aliquam velit risus, placerat et, rutrum nec, condimentum at, leo. Aliquam in augue a magna semper pellentesque. Suspendisse augue. Nullam est nibh, molestie eget, tempor ut, consectetuer ac, pede. Vestibulum sodales hendrerit augue. Suspendisse id mi. Aenean leo diam, sollicitudin adipiscing, posuere quis, venenatis sed, metus. Integer et nunc. Sed viverra dolor quis justo. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis elementum. Nullam a arcu. Vivamus sagittis imperdiet odio. Nam nonummy. Phasellus ullamcorper velit vehicula lorem. Aliquam eu ligula. Maecenas rhoncus. In elementum eros at elit. Quisque leo dolor, rutrum sit amet, fringilla in, tincidunt et, nisi.

Donec ut eros faucibus lorem lobortis sodales. Nam vitae lectus id lectus tincidunt ornare. Aliquam sodales suscipit velit. Nullam leo erat, iaculis vehicula, dignissim vel, rhoncus id, velit. Nulla facilisi. Fusce tortor lorem, mollis sed, scelerisque eget, faucibus sed, dui. Quisque eu nisi. Etiam sed erat id lorem placerat feugiat. Pellentesque vitae orci at odio porta pretium. Cras quis tellus eu pede auctor iaculis. Donec suscipit venenatis mi.

Aliquam erat volutpat. Sed congue feugiat tellus. Praesent ac nunc non nisi eleifend cursus. Sed nisi massa, mattis eu, elementum ac, luctus a, lacus. Nunc luctus malesuada ipsum. Morbi aliquam, massa eget gravida fermentum, eros nisi volutpat neque, nec placerat nisi nunc non mi. Quisque tincidunt quam nec nibh sagittis eleifend. Duis malesuada dignissim ante. Aliquam erat volutpat. Proin risus lectus, pharetra vel, mollis sit amet, suscipit ac, sapien. Fusce egestas. Curabitur ut tortor id massa egestas ullamcorper. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec fermentum. Curabitur ut ligula ac ante scelerisque consectetuer. Nullam at turpis quis nisl eleifend aliquam. Sed odio sapien, semper eget, rutrum a, tempor in, nibh.

4.1.1. Creating New Group

To create a new Group of Products select from menu {**File** > New > Group}.

The next step after you are finished with New Product creation, is adding a New Product > to this Group.

4.1.2. Creating New Product

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Aliquam velit risus, placerat et, rutrum nec, condimentum at, leo. Aliquam in augue a magna semper pellentesque. Suspendisse augue. Nullam est nibh, molestie eget, tempor ut, consectetuer ac, pede. Vestibulum sodales hendrerit augue. Suspendisse id mi. Aenean leo diam, sollicitudin adipiscing, posuere quis, venenatis sed, metus. Integer et nunc. Sed viverra dolor quis justo. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis elementum. Nullam a arcu. Vivamus sagittis imperdiet odio. Nam nonummy. Phasellus ullamcorper velit vehicula lorem. Aliquam eu ligula. Maecenas rhoncus. In elementum eros at elit. Quisque leo dolor, rutrum sit amet, fringilla in, tincidunt et, nisi.

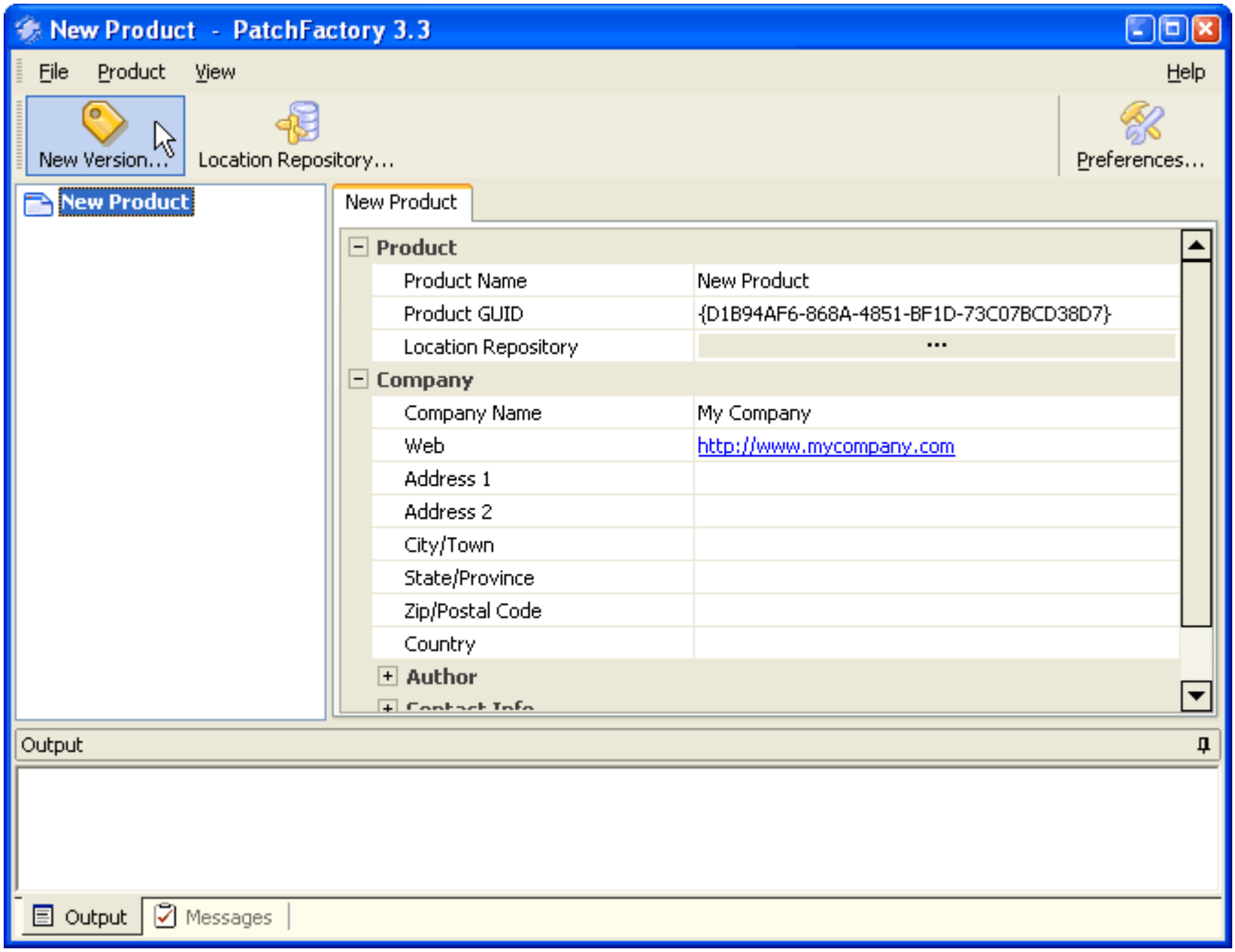
Donec ut eros faucibus lorem lobortis sodales. Nam vitae lectus id lectus tincidunt ornare. Aliquam sodales suscipit velit. Nullam leo erat, iaculis vehicula, dignissim vel, rhoncus id, velit. Nulla facilisi. Fusce tortor lorem, mollis sed, scelerisque eget, faucibus sed, dui. Quisque eu nisi. Etiam sed erat id lorem placerat feugiat. Pellentesque vitae orci at odio porta pretium. Cras quis tellus eu pede auctor iaculis. Donec suscipit venenatis mi.

Aliquam erat volutpat. Sed congue feugiat tellus. Praesent ac nunc non nisi eleifend cursus. Sed nisi massa, mattis eu, elementum ac, luctus a, lacus. Nunc luctus malesuada ipsum. Morbi aliquam, massa eget gravida fermentum, eros nisi volutpat neque, nec placerat nisi nunc non mi. Quisque tincidunt quam nec nibh sagittis eleifend. Duis malesuada dignissim ante. Aliquam erat volutpat. Proin risus lectus, pharetra vel, mollis sit amet, suscipit ac, sapien. Fusce egestas. Curabitur ut tortor id massa egestas ullamcorper. Cum sociis natoque penatibus et magnis dis parturient montes, nascetur ridiculus mus. Donec fermentum. Curabitur ut ligula ac ante scelerisque consectetuer. Nullam at turpis quis nisl eleifend aliquam. Sed odio sapien, semper eget, rutrum a, tempor in, nibh.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

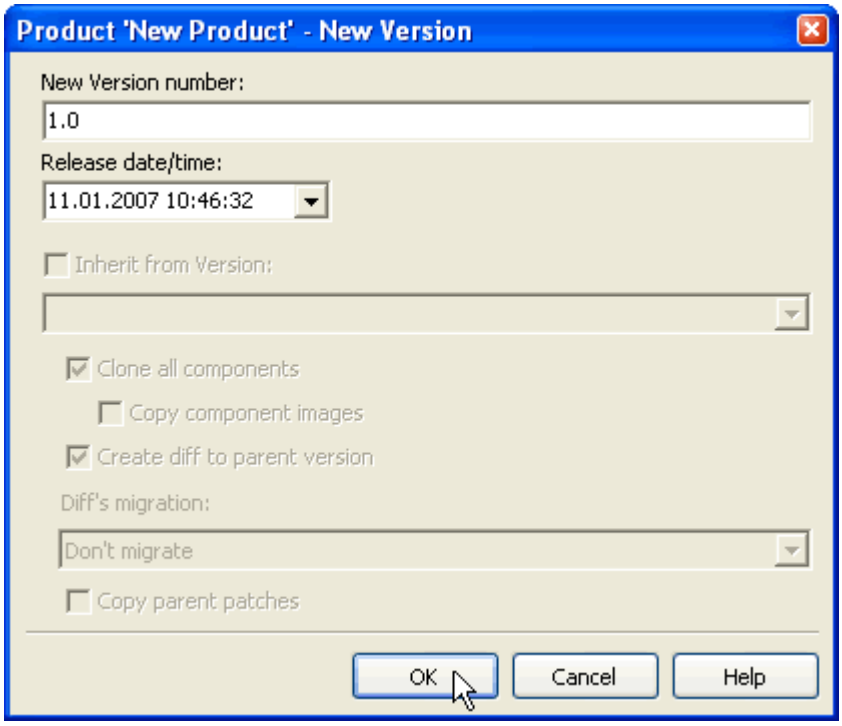
4.1.3. Adding New Version

To create a new Version select the desired Product node within the Product tree and select from menu {**Product** > New Version} or click the **New Version** button in the main tollbar.



This will initiate a "New Version Dialog" which you can below.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



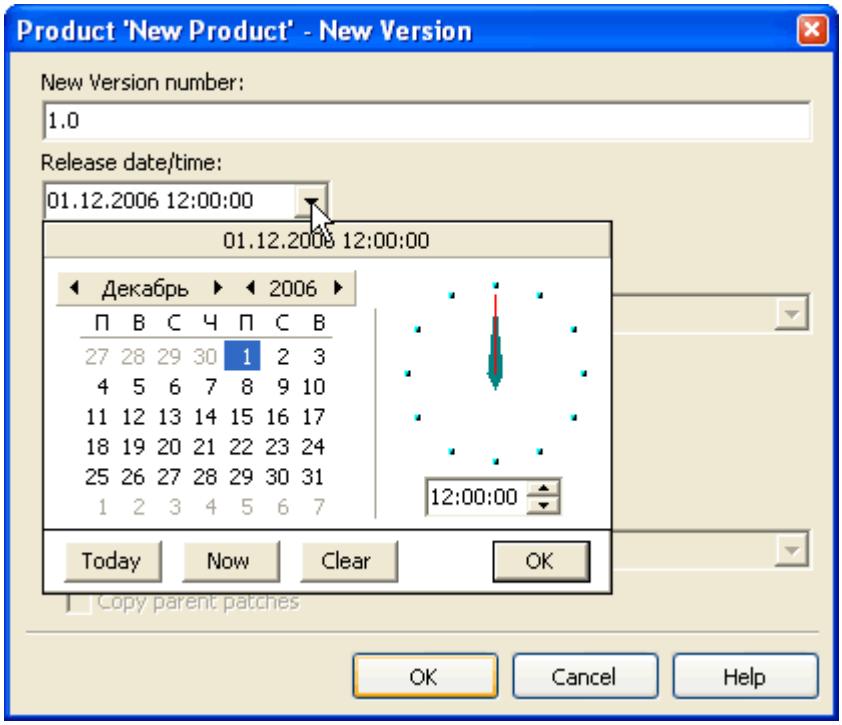
Dialog options:

- **New version number:**

Enter the New Version Number. It can be changed later after version creation.
Version number must form a valid name of file, thus using of such symbols as < > : " / \ | * ? is not allowed.

- **Release date/time:**

Enter the Version Release date and time here.
Click drop-down arrow select date and time using a calendar.



- **Inherit from version:**

Select version which is parent relative to this one. Parent version properties are inherited by the new version.

- ***Clone all components:***

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

Select this option to create empty components identical to the components of the parent version in the new version.

- ***Create diff to parent version:***

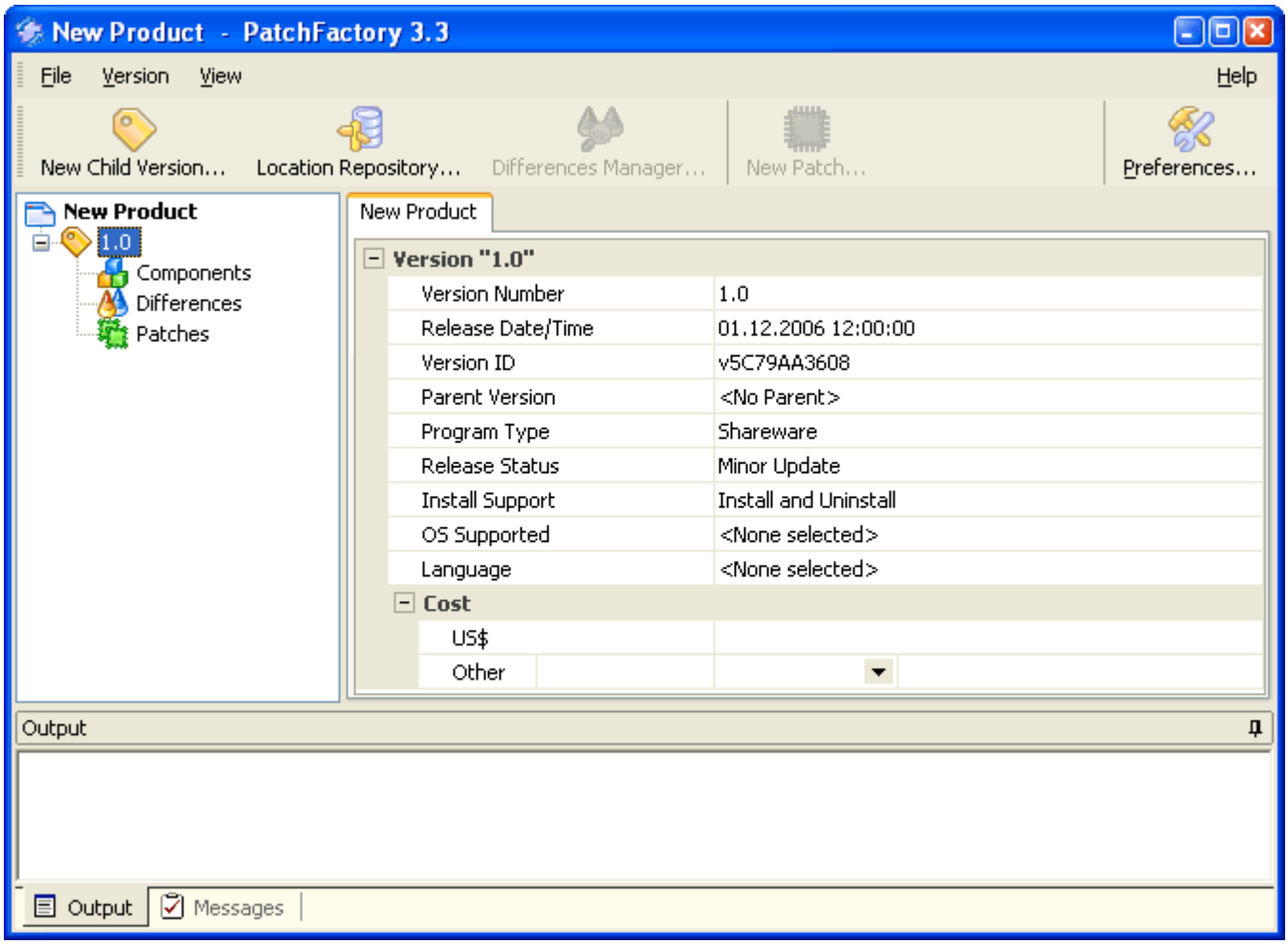
Select this option automatically create difference to selected parent version.

- ***Diff's migration:***
  - Don't migrate
    Do not copy or move any differences from parent version.
  - Copy all diffs from parent version
    All differences created in parent version will be copied to the new version.
  - Move all diffs from parent version
    All differences created in parent version will be moved to the new version.
- ***Copy parent patches:***

Select version which is parent relative to this one.

Refer to the "*Concepts and definitions*" section for more information concerning Version definition in terms of PatchFactory and Version parameters list.

After New Version is created you can define additional (optional) Version parameters.



Dialog options:

- **Version Number:**

Contains Version Number you have entered at the previous step.

- **Release date:**

Contains Release date of this version. *Optional parameter.*

- **Version ID:**

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

Generated automatically and cannot be changed after version is created.

- **Parent version:**
Indicates Parent version number relating to the selected version. *Optional parameter.*

- **Program type:**
Indicates program type. Can be defined as ***Shareware***, ***Freeware***, ***Adware***, ***Demo***, ***Commercial***, or ***Data Only***. *Optional parameter.*

- **Release Status:**
Indicates program release status type. Can be defined as ***Major Update***, ***Minor Update***, ***New Release***, ***Beta***, ***Alpha*** , or ***Media Only***. *Optional parameter.*

- **Install Support:**
Indicates program install support properties. Can be defined as ***Install and Uninstall***, ***Install Only***, ***No Install Support***, or ***Uninstall Only***. *Optional parameter.*

- **OS Supported:**
Shows program's supported OS'es list. Choose OS'es your software supports. *Optional parameter.*

- **Language:**
Shows program's supported Languages list. Choose Languages your software supports. *Optional parameter.*

- **Cost:**
  - US$: program cost in US$
  - Other: select the appropriate currency from the drop-down list to specify program cost in other currency then US$.

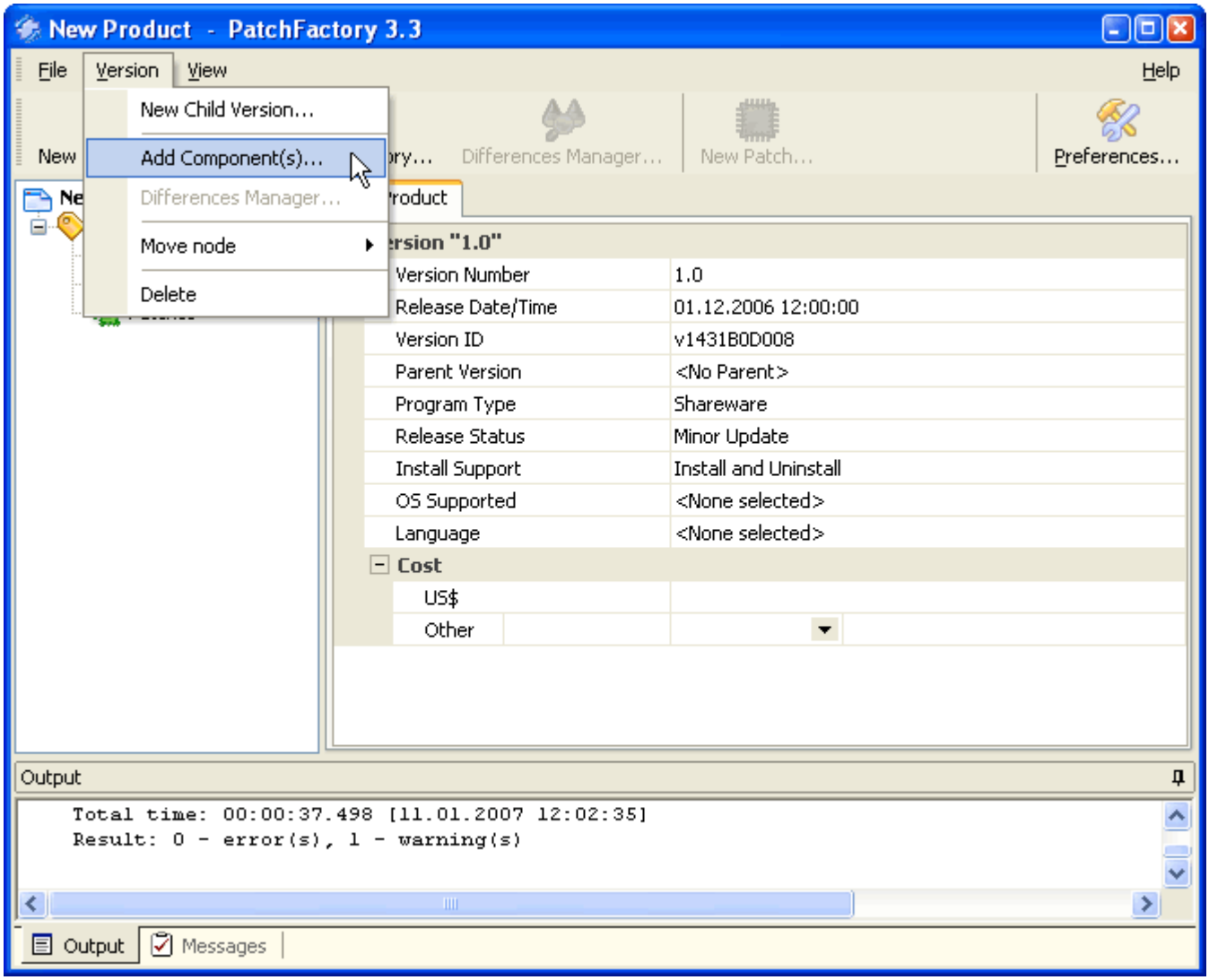
The next step after you're finished with New Version creation, is Adding Component(s) to this Version >.

4.1.4. Adding Component(s)

To create a new Component select the desired Version node within the Product tree and choose from menu { **Version** > Add Component(s)}.
This will initiate "Add Component(s)" which you can see below.

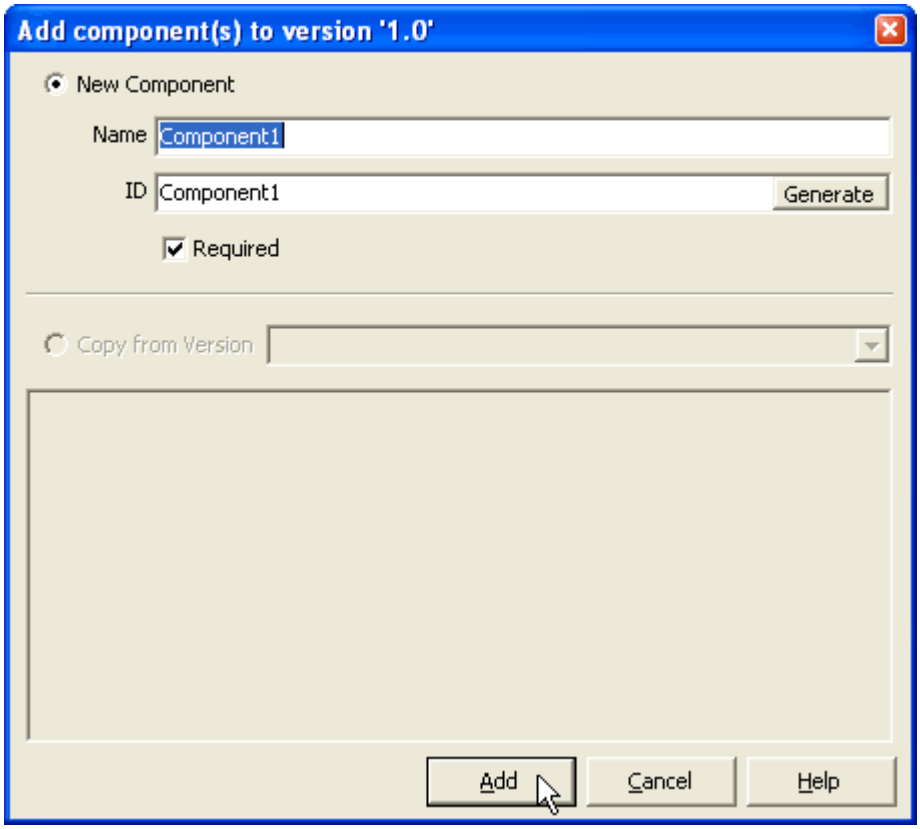
There are two available choices. You can either create a new Component or copy a Component (and namely all its properties) from selected version.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



This will initate a "Add New Component" dialog:

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



New Component:

- **Name:**

Enter New Component name here. It can be changed later after component creation.

- **ID:**

Generated automatically and cannot be changed after component is created.

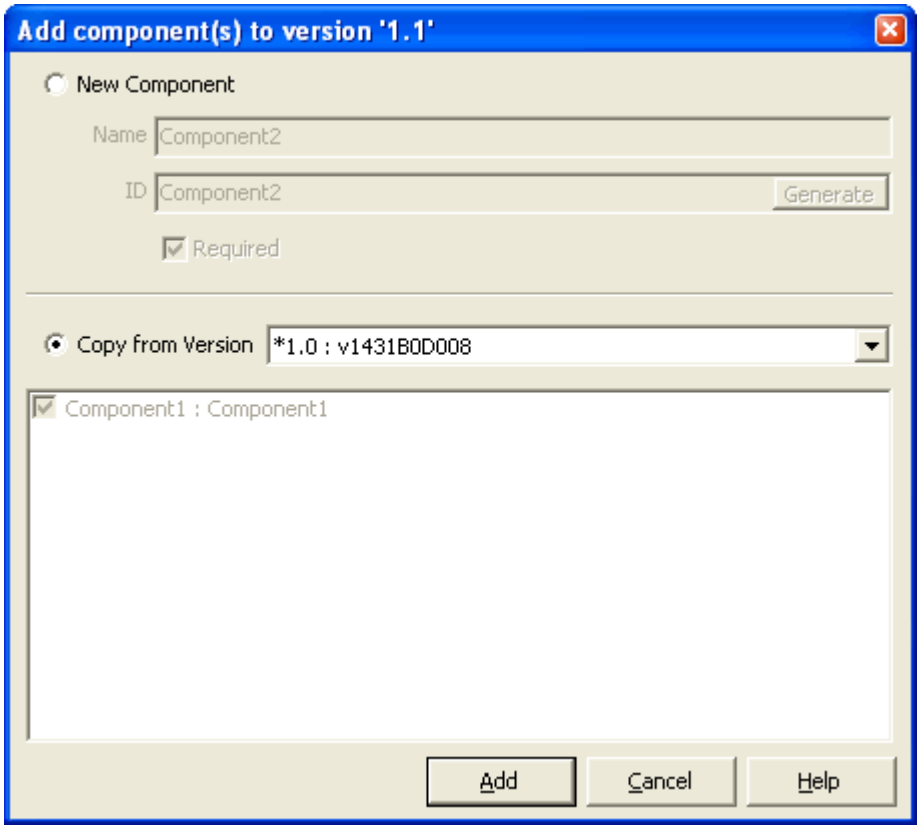
- **Required:**

Select this option if a component to be added is Required.

Refer to the "*Concepts and definitions*" section for more information concerning Component definition in terms of PatchFactory and Component parameters list.

Copy from version:

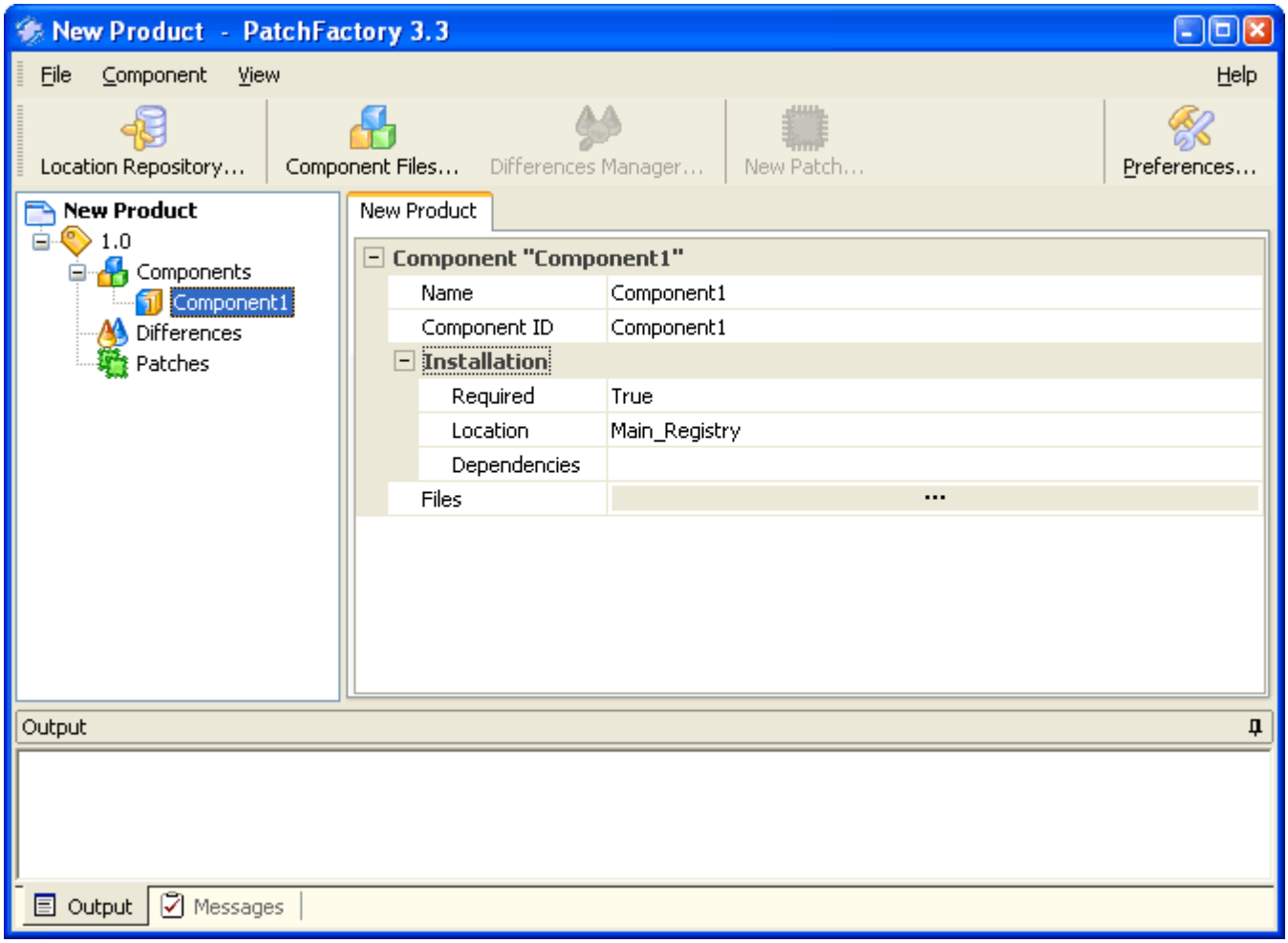
© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



Here you can clone some components from one of the previous versions upon your selection.
Select a version from a drop-down box. In the field below check components which you want to transfer to the created Component.
According to the choice made empty components will be created with identical properties (as the selected components from selected version).

After New Component is created you can define additional (optional) Component parameters.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



Dialog options:

- **Name:**
Component name. It can be changed after component creation.

- **ID:**
Generated automatically and cannot be changed after component is created.

- ***Installation:***

  - **Required:**
  Values: True or False. Parameter which defines whether a component is required or not.

  - **Location:**
  This parameter defines the record from Location Repository which is used to detect the location of this component on the end-user's machine.
  Select record to be used from the drop-down list.

  - **Dependencies:**
  Optional parameter. Select components that affect the current component.

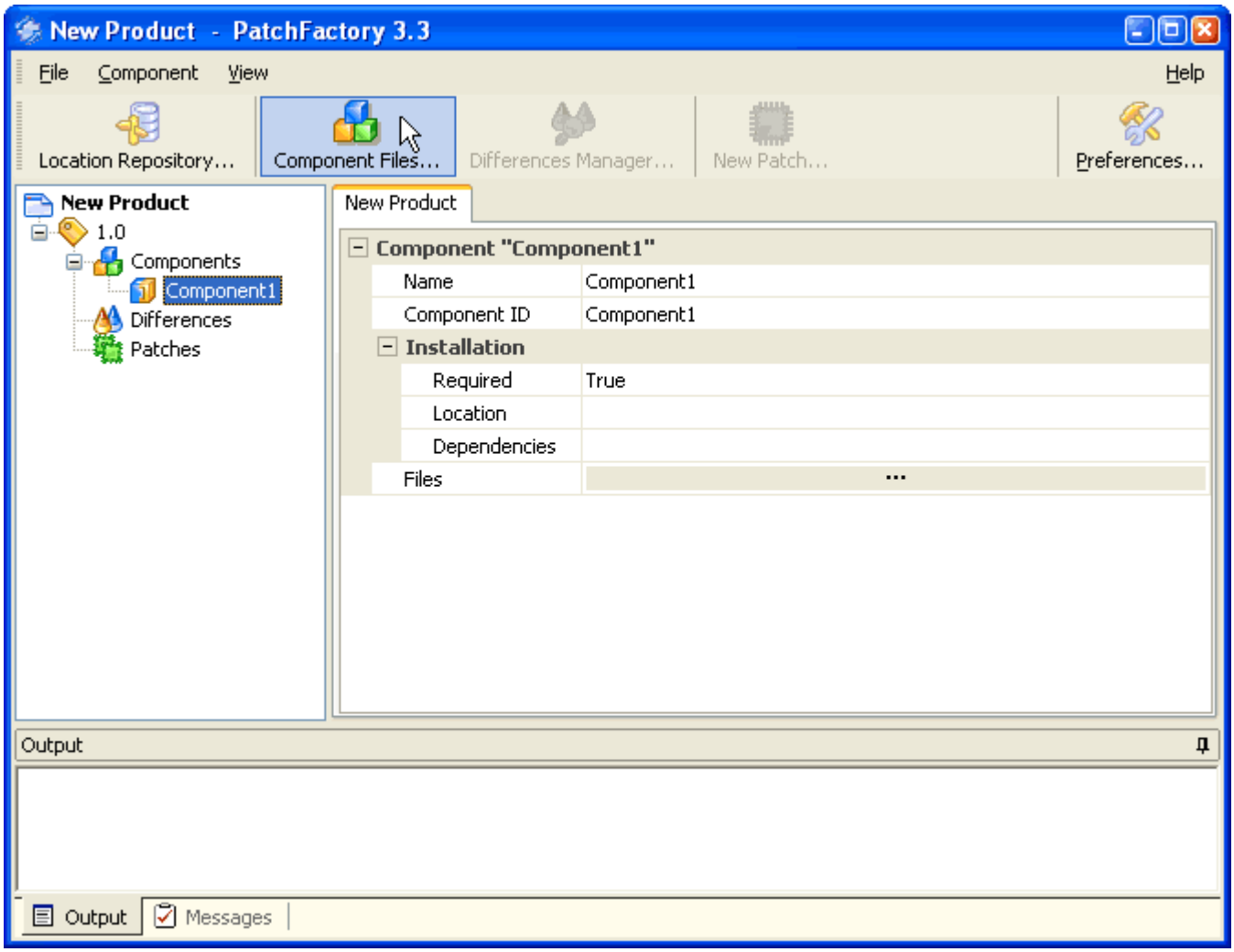
  - **Files:**
  Click "**...**" to add files to the current component.

The next step after you're finished with New Version creation, is Adding Files to the Component >.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

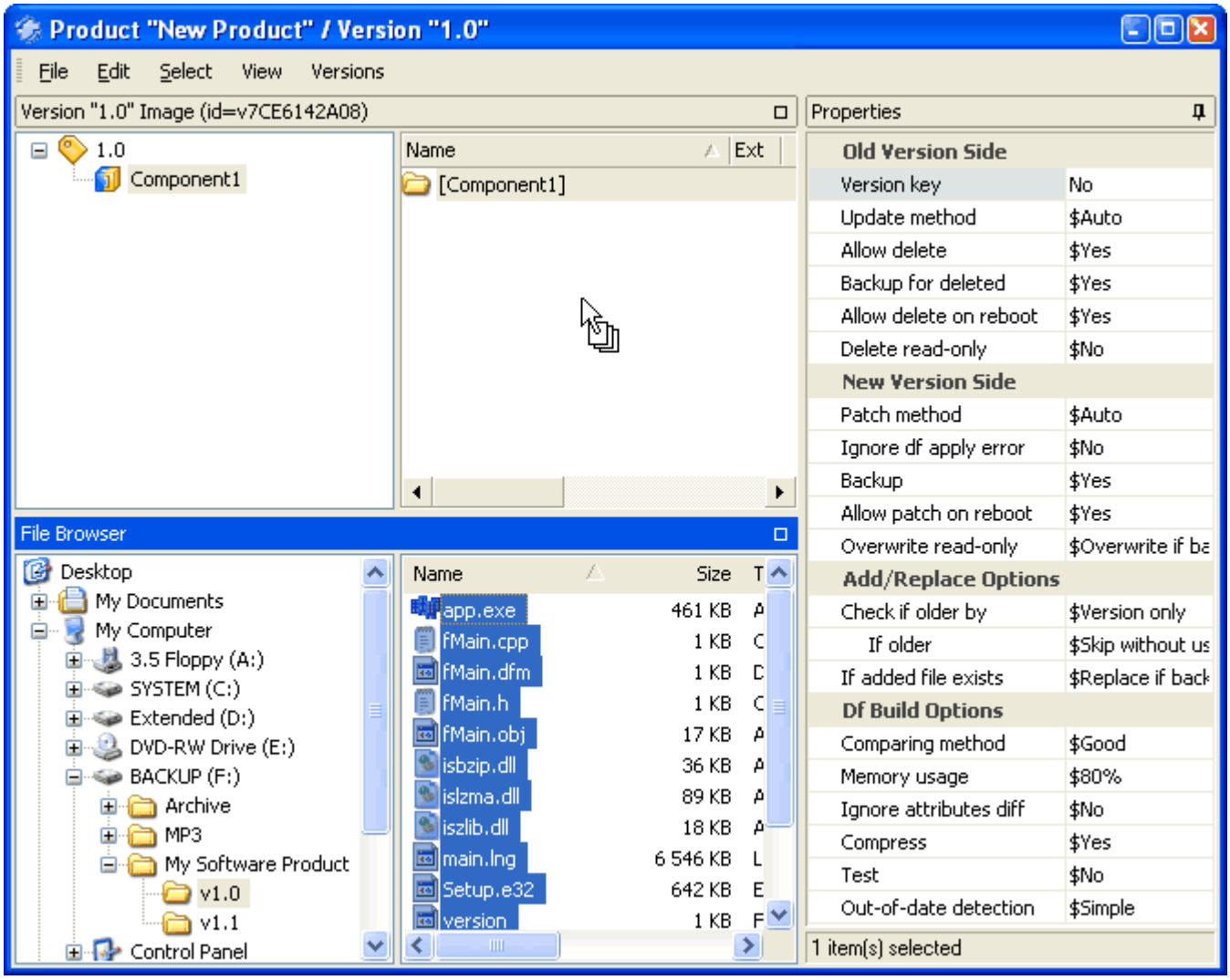
4.1.5. Adding Component Files

To add files to the Component select the desired Component node within the Product tree choose from menu { **Component** > Component Files} or click the **Component Files** button in the main tollbar.



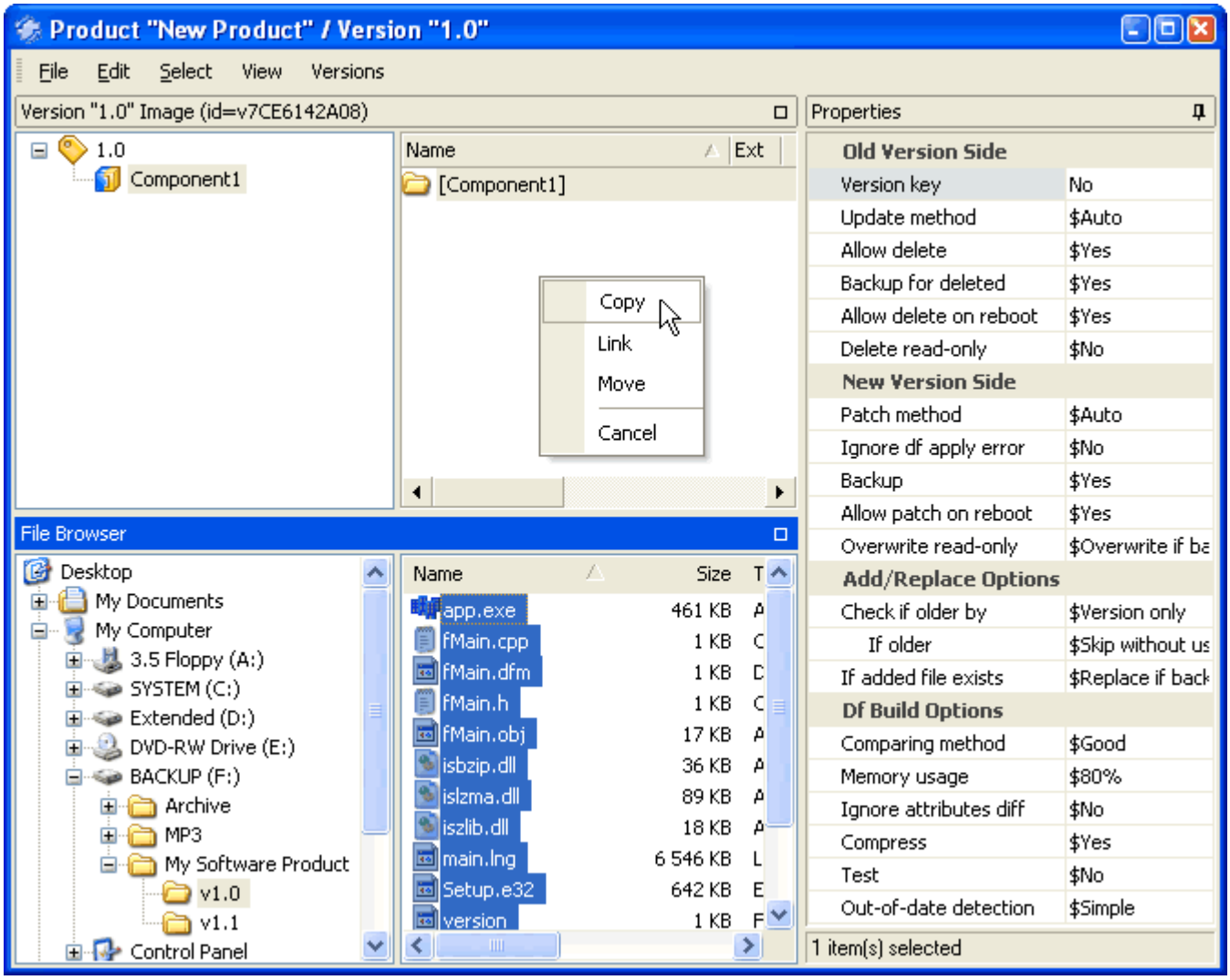
This will invoke "Version Files Editor" (see screenshot below).

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



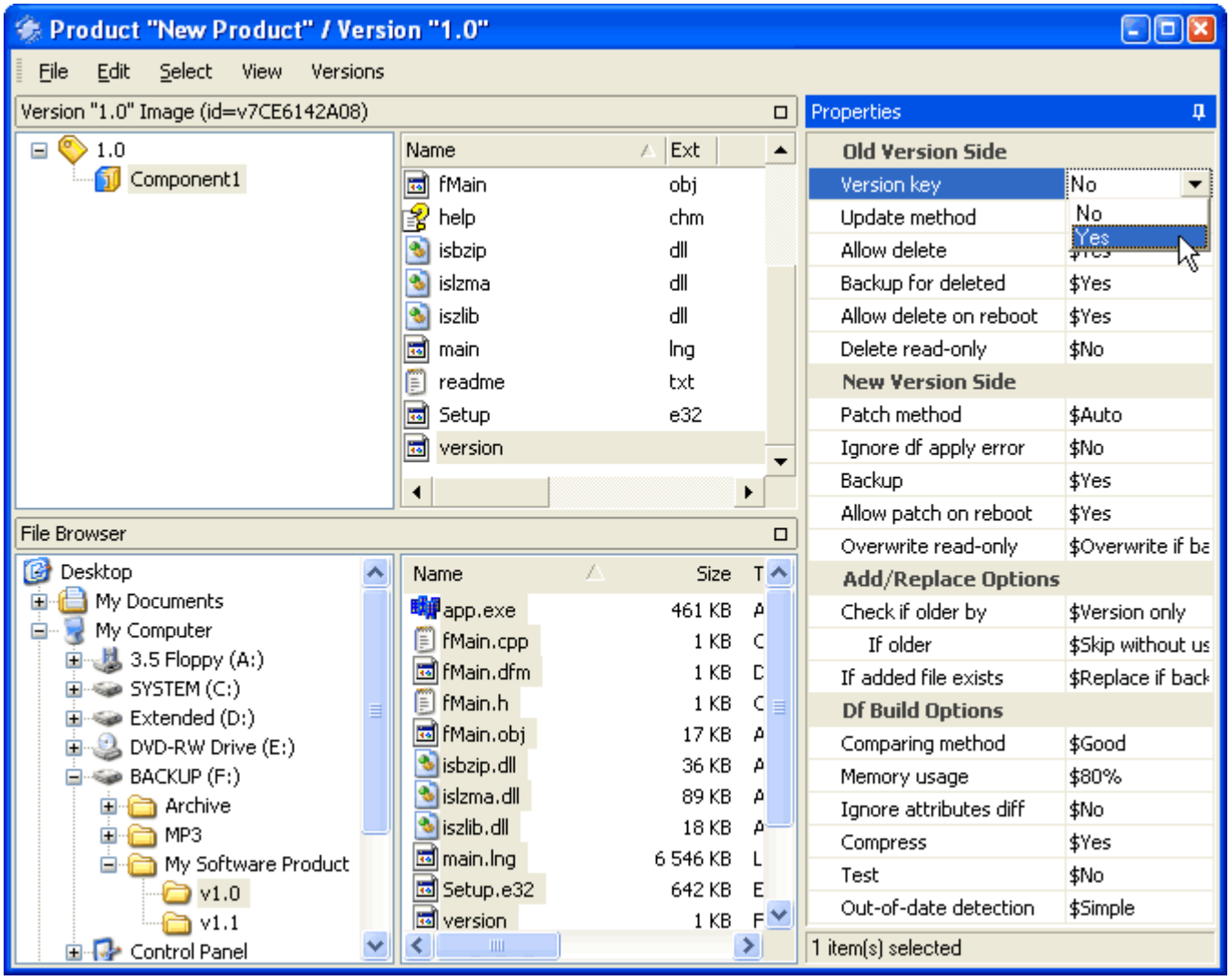
To add files to the selected component navigate to the desired folder in the right-bottom "File Browser" subwindow and drag-and-drop files to the "Version ... Image (id=v....)" window.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



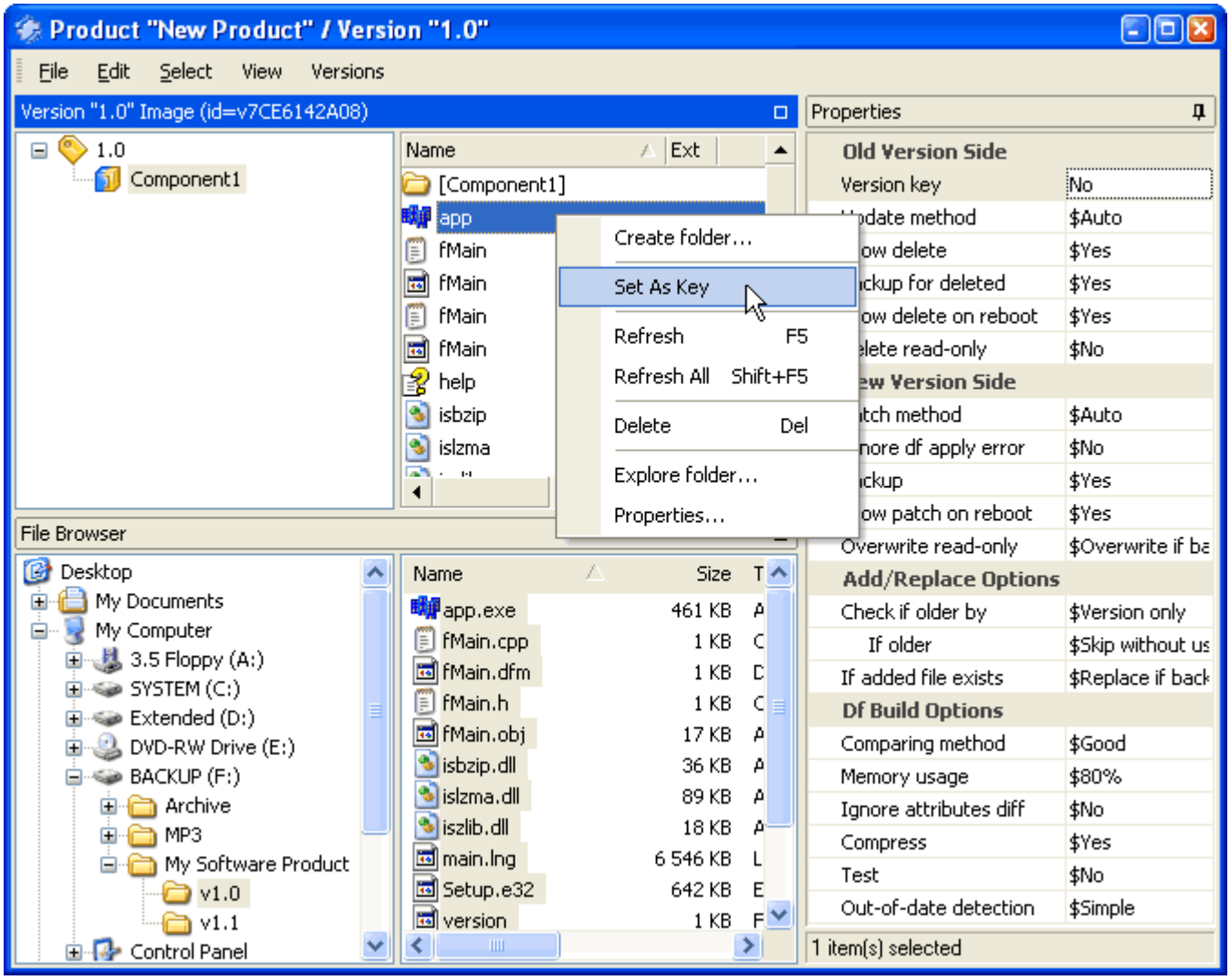
Select **Version Key** file(s) (used to determine the version number on the end-user's machine). You can do it by selecting the appropriate file within selected Component and changing its "Version Key" property to Yes (in the "Properties" subwindow to the right). Files, marked as *Version Keys* are underlined in the "Version Image" window. Refer to the **Installed Version Detection** section of this manual for more information on **Version Keys** definition and possible usage hints.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



Then set Component's **Key File(s)**. Right-click on the desired file and select "Set as Key" in the drop-down or select a file you want to make *Key* and then choose from menu {**Edit** > Set as Key}. *Key Files* are used to determine whether a whole Component is installed on the end-user's machine or not. Thus they must exist during all Version Component's life.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



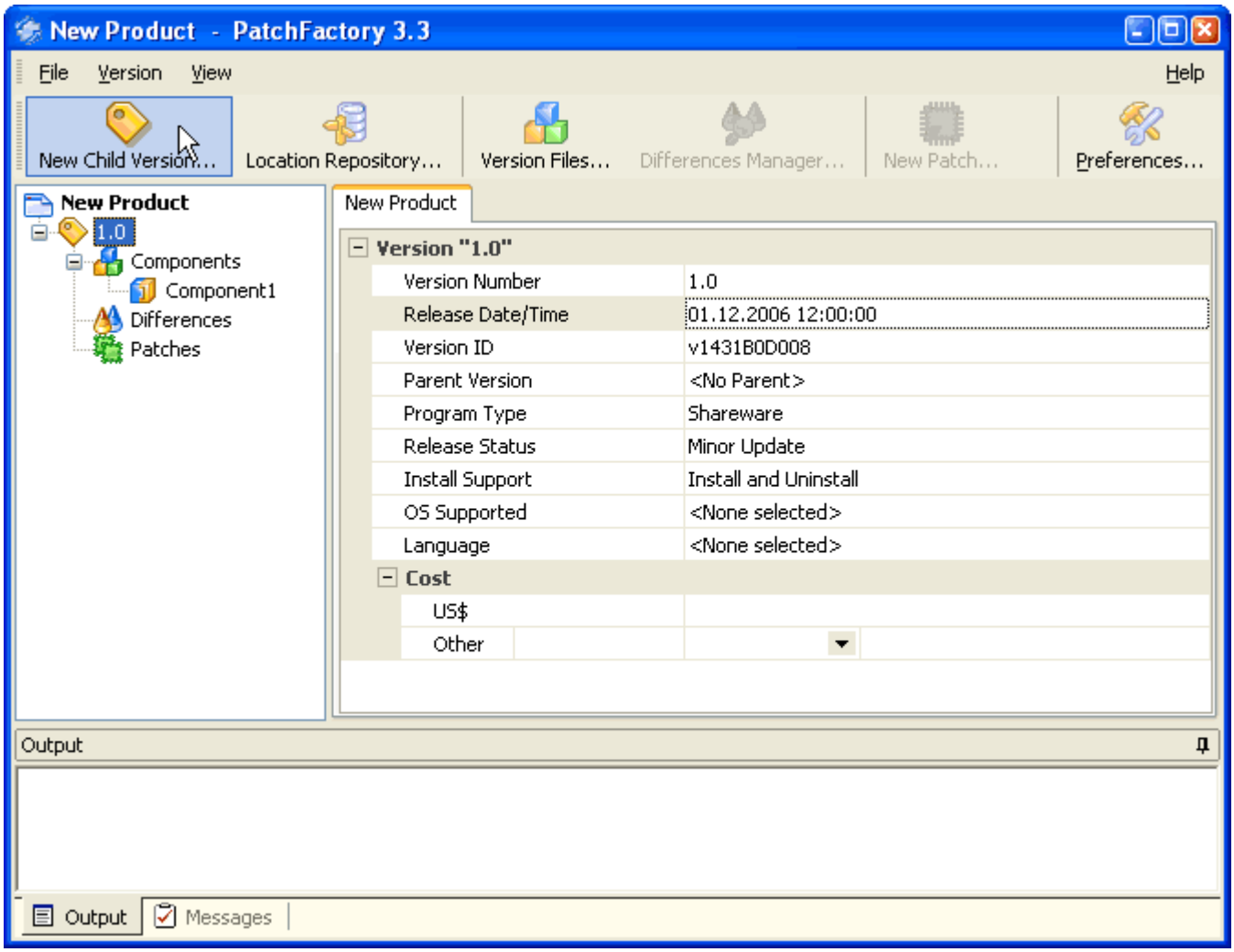
- To get the description of all menu items, please, refer to the Version Files Editor Menu section of this manual.
- For more information on using Version Files Editor and for detailed description on the , please, refer to Version Files Editor section of this manual.

The next step after you're finished with New Version creation, is Adding Child Version >.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

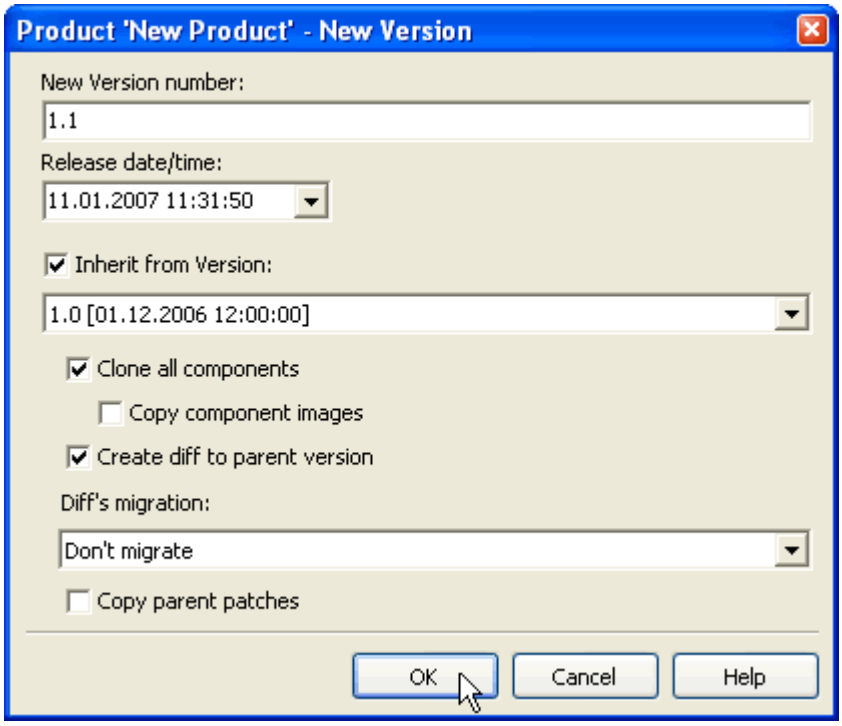
4.1.6. Adding Child Version

To create a new Child Version select the desired Version node within the Product tree and select from menu { **Version** > New Child Version} or click the **New Child Version** button in the main tollbar..



This will initiate a "New Version Dialog" which you can below.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

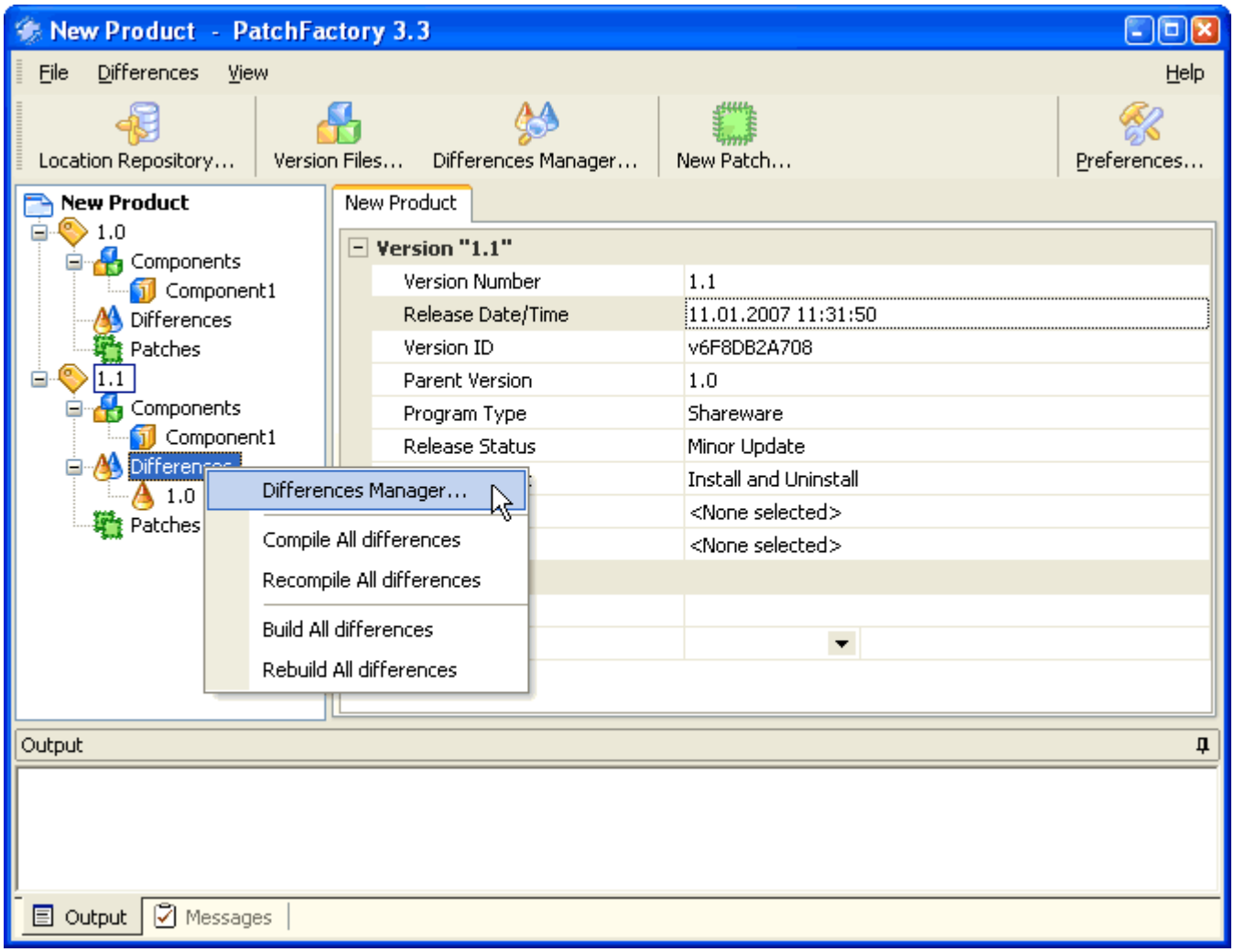


The next step after you're finished with a Child Version creation, is Adding Differences for the current Version >.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4.1.7. Creating Differences

To add Difference(s) to the Version select the desired *Version* node within Product tree and select *Differences* node within Product tree and choose from menu {**Differences** > Differences Manager} or right-click on the appropriate **Differences** node within Product tree.



This will invoke "Difference Manager" (the screenshot you can see below).

Refer to the "*Concepts and definitions*" section for more information concerning Difference definition in terms of PatchFactory.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

Here you can see all available Versions for this Product that you can make Differences.
Select appropriate versions from the list that you want to add a Difference.
You can sort available versions list by *Version Number*, *Release date*, *Ancestor* status or version *GUID*.

Tabs description:

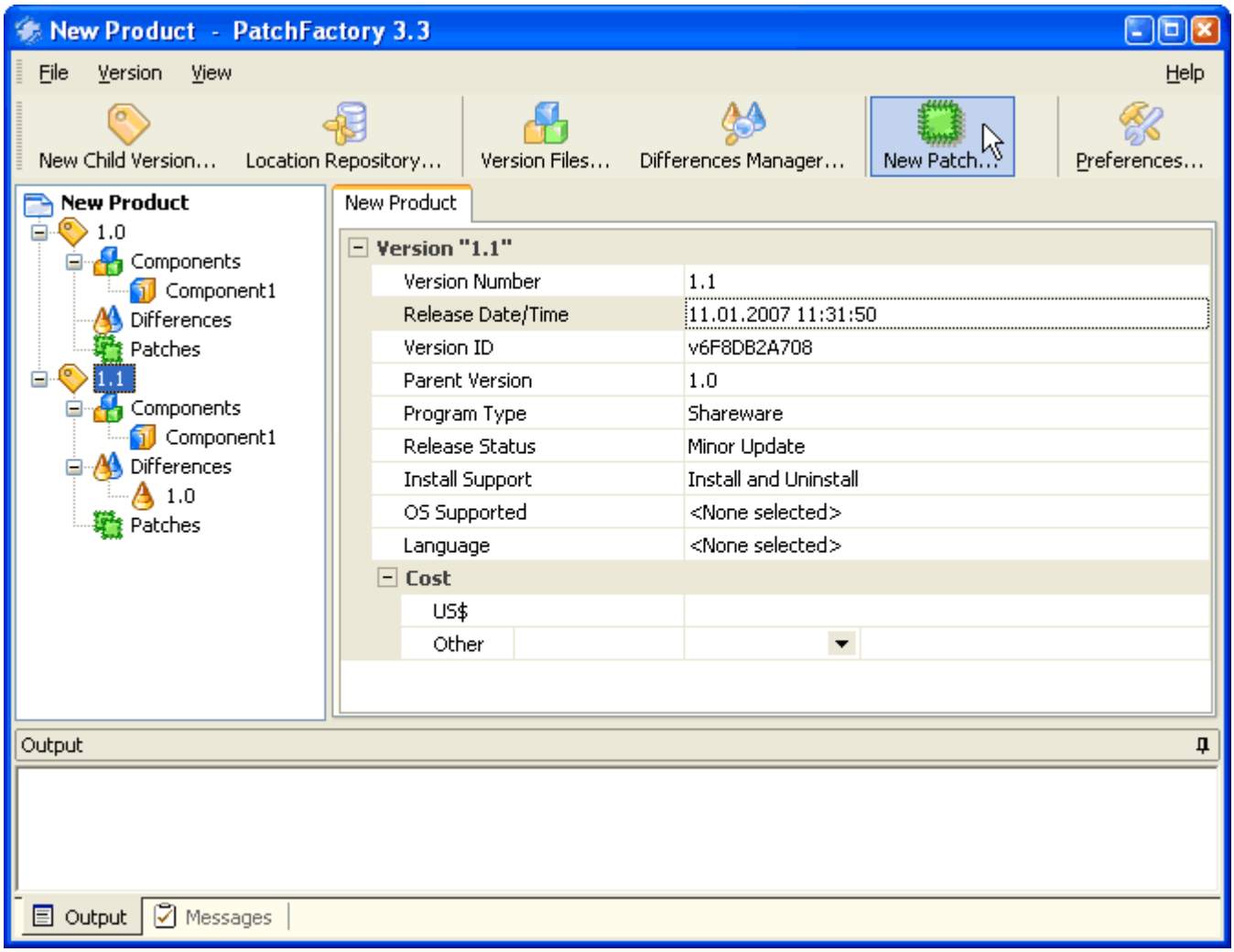
- **Version**
  Indicates version number string.
- **Release date/time**
  Indicates version release date/time.
- **Ancestor**
  "+" indicates that this version is an ancestor to the active Version.
- **GUID**
  Automatically generated Version's unique identifier.

The next step after you're finished with New Version creation, is Creating New Patch Module for the current Version >.

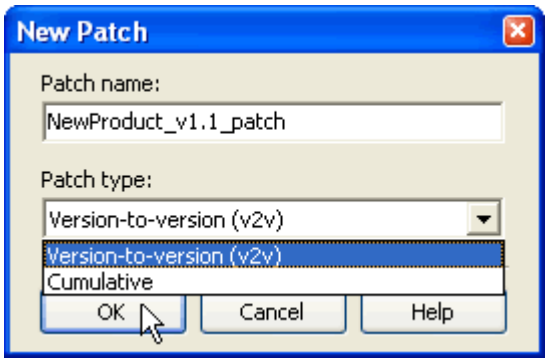
© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4.1.8. Creating New Patch

To add *Patch module* select *Patches* node inside the desired *Version* node within Product tree and select *Patches* node within Product tree and choose from menu {**Patches** > New Patch}.



This will invoke "New Patch" dialog (the screenshot you can see below).



Dialog options:

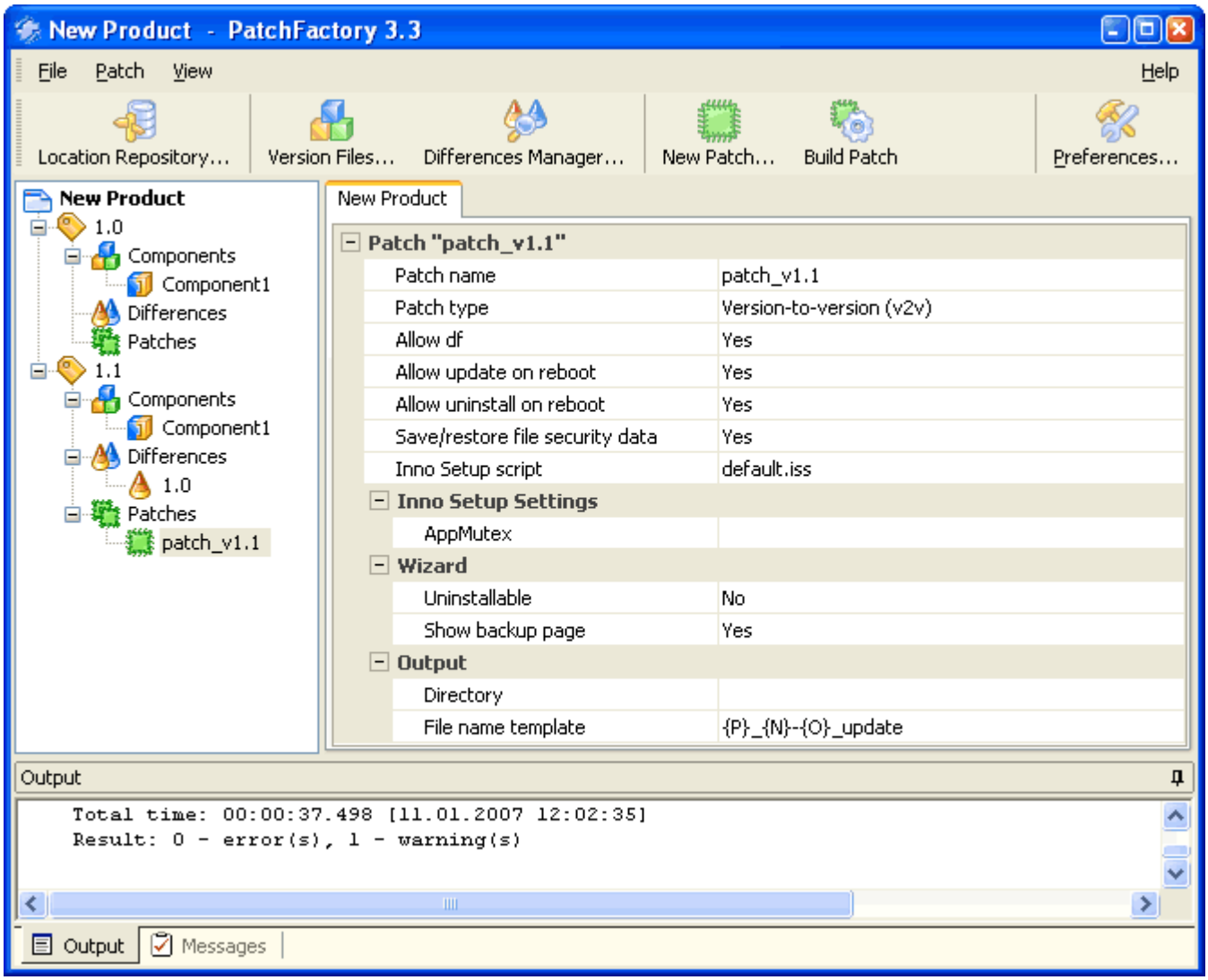
- **Patch Name:**
Enter the Patch Name you want to build. Patch name cannot be changed after patch creation.
Patch Name must form a valid name of file thus using of such symbols as < > : " / \ | * ? is not allowed.

- **Patch Type:**
Enter Patch Type: *Version-to-version* or *Cumulative*.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

Refer to the "Concepts and definitions" section for more information concerning Patch types in terms of PatchFactory.
- Use *Version-to-version* patch type if you want to make a patch module that updates **one old version to the current one**.
- Use *Cumulative* patch type if you want to make a patch module that updates **several old versions to the current one**.

After pressing OK button a New Patch is created. Now you're ready to view/edit essential patch options.



Dialog options:

- **Patch Name:**
Patch Name which you have specified at the previous step.

- **Patch Type:**
Version-to-version or *Cumulative*.

- **Allow df:**
Set to **YES (default)** if want to use difference between files of the same name during update procedure.
If set to **NO** then full copies of new files are used if they differ from old version files.

- **Allow update on reboot:**
Provides replacing of files to be updated at system restart (most often if they are occupied by other processes).
YES by default.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

- **Allow uninstall on reboot:**

Provides replacing of files to be uninstalled/repaired at system restart (most often if they are occupied by other processes).

- **Uninstallable:**

If set to **YES**, the backup copy of all replaced/modified files (for which backup is allowed) is saved and a record is added into Add/Remove programs list.
If set to **NO** then rollback/uninstall of the patch after its successful applying is impossible. Backup copies of files are stored in the temporary folder during patch applying only to provide rollback if an error occurred during patch applying to maintain an old version operating.

- **InnoSetup script:**

Name of the InnoSetup script file used to generate the update module. **"default.iss" by default.**

- **Directory:**

Folder where the output update modules are saved to.

- **File name Template:**

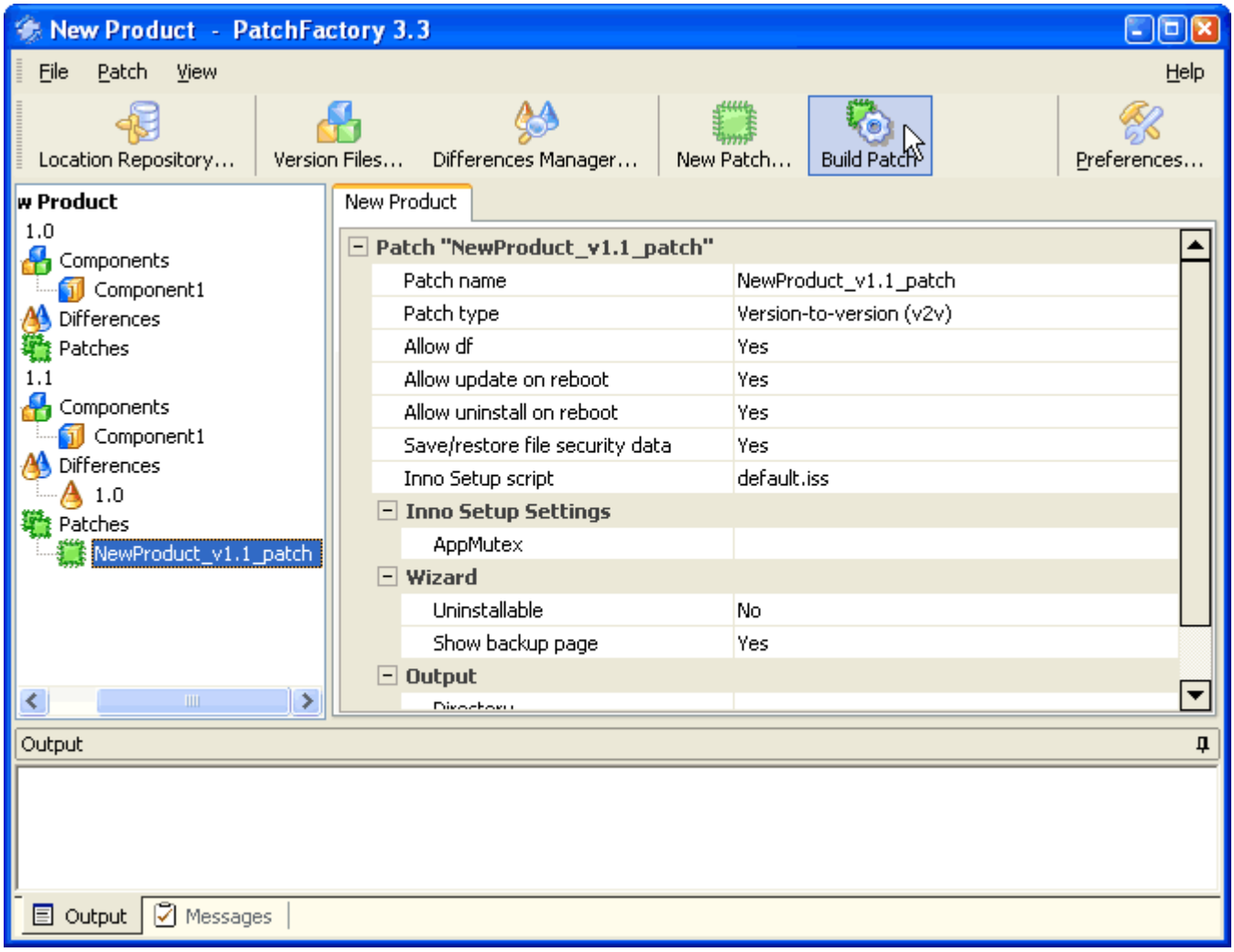
Template of update modules names.

Refer to the "Concepts and definitions" section for more detailed description concerning Patch parameters in terms of PatchFactory.

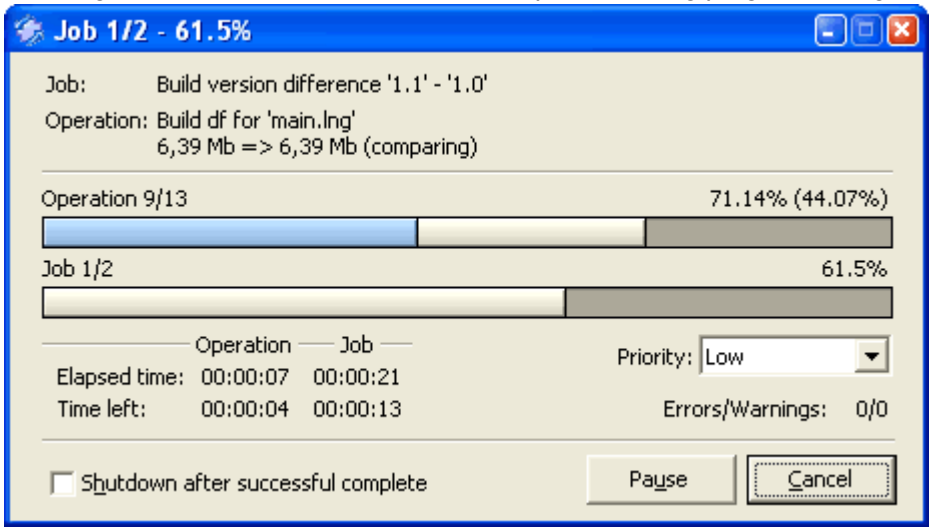
And at last we are ready to build the Update Module.

To build the Update module select "*Patch module"* node inside the desired *Version's* "*Patches"* node within Product tree and click on the **Build Patch** button or select *Patches* node within Product tree and choose from menu {**Patch** > Build Patch}.
You can also right-click with your mouse on the *Patch module* node you want to build within Product tree and select *"Build Patch"* from drop-down list.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



Pressing the **Build Patch** button initiates the patch building progress dialog to appear:



- **Operation** indicates ***execution progress of the current operation***. (blue-coloured progress bar indicates the ***similarity ratio*** of the current old version file versus new version file during their comparing).
- **Job** – indicates ***execution progress*** of the current job..
- **Priority** – indicates current system priority of the task.

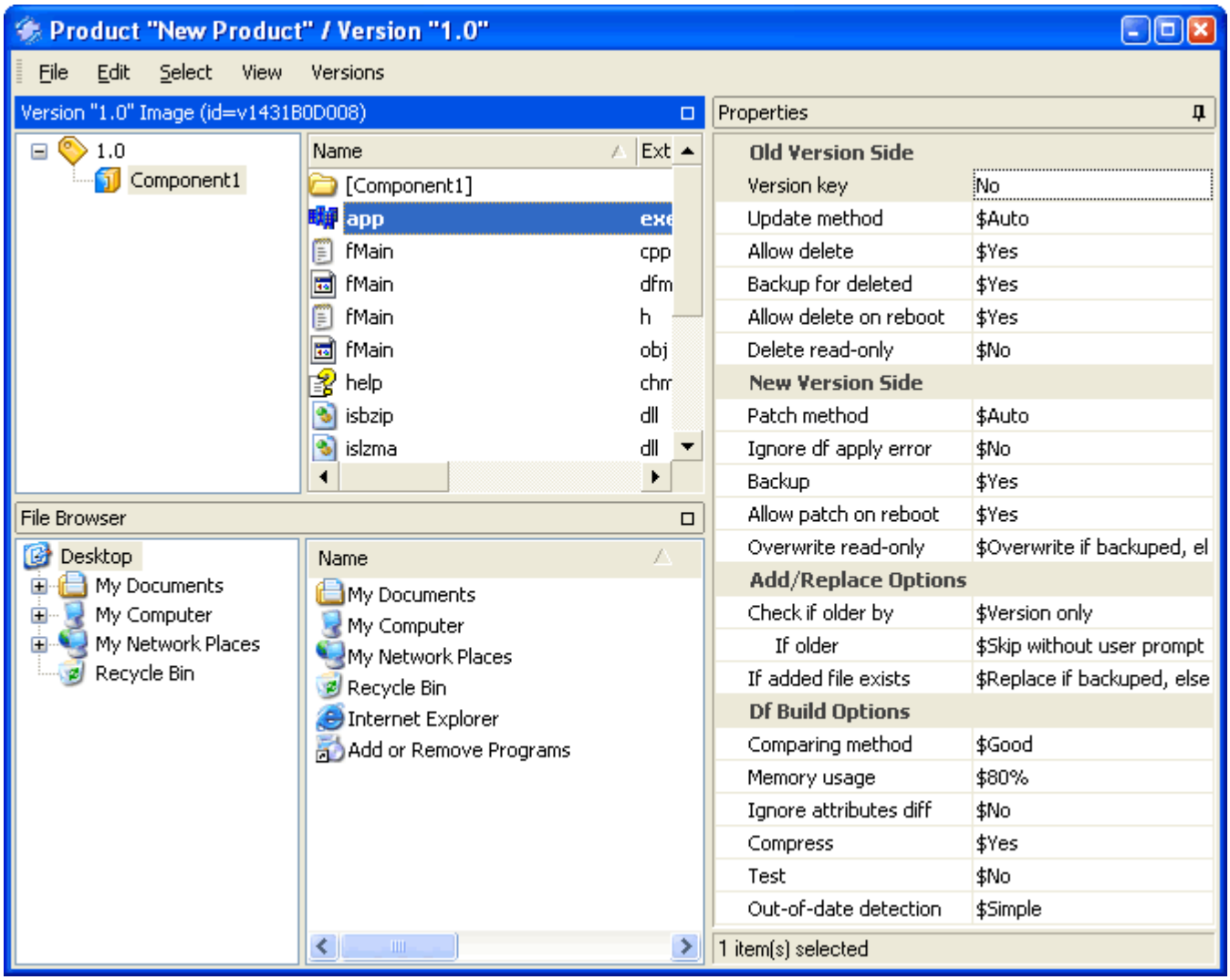
To pause patch building process press "Pause" button.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

To cancel patch building process press "Cancel" button.

4.2. Version Files Editor

Version Files Editor can be used to Add/Remove files from version's Components, to set different file attributes necessary for difference / patch building or update installation.



Version Files Editor main window consists of 3 subwindows:

- *Version Image Browser window* - used to navigate through Version / component image files;
- *File Browser window* - used to navigate through files on your system and to add them to component images;
- *Properties window* - used to set properties necessary for difference / patch building or update installation.

Version Image Browser

This window can be used to view version components, folders/files, to set/observe their properties or to add folders/files to components. To set file property - select it in the appropriate Version Image and then change necessary property value in the Properties window.

File Browser

Similar to Windows Explorer. Use this file browser window to navigate to the appropriate file(s) or directory that you want to add to the active Version Image (which you have selected in the *Version Image Browser*).

To add files or directories to the active *Version Image* select appropriate files / folders (as you do it in Windows Explorer) and drag&drop them to the *Version Image* window. If you use right mouse button then selected files/folders are copied into the Version Image and if left mouse button is used - then you can select from either you want to make

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

a hard copy, move the selected files/folders or to copy only references/links to them.

Tabstops header on the top of each window allows sorting of the file list by *Name*, *Date* and *Size* by clicking on the appropriate header. Clicking a second time on the same header reverses the order. Arrow to the left of the header text shows the sort direction.

Properties

All properties are divided into 2 main categories: ***Old Version Side*** and ***New Version Side***.
Properties of the *first category* are used during compilation of the difference between versions when selected version is considered as an *old* one, while properties of the *second category* are used during compilation of the difference between versions when selected version is considered as a *new* one.

Df build Options and ***Add/Replace Options*** subgroups are covered by the 2nd category.

Each property can have its own value or it can inherit (*by default*) the value from a Parent object (***valid for all properties except Version Key***).

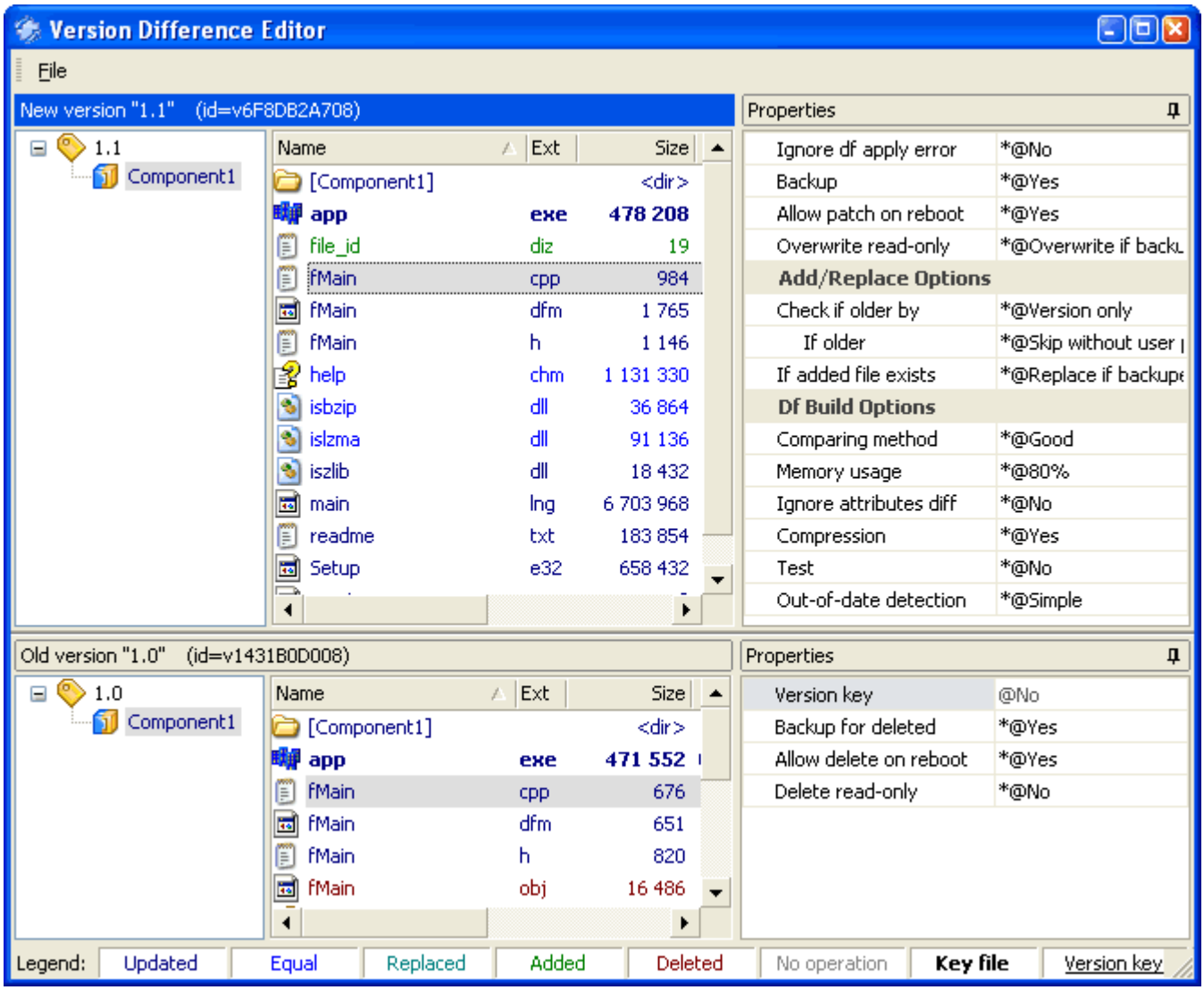
If property Value is prefixed with *' **$** '* symbol - it means that this property inherits its value from a Parent object.
Select **<Inherit>** option in the list of available values for selected property to make this property inherited from a parent object.

You can set properties values simultaneously for several elements which were selected in the Version Image.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4.3. Version Difference Editor

Version Difference Editor can be used to View files/folders both in New & Old Version Images, to change various file attributes necessary for difference / patch building or update installation or to change update commands and their parameters for particular difference.



Version Difference Editor main window consists of 2 main subwindows:

- *New Version Image window* - used to navigate through New Version Image components, folders and files and to set their properties (upper window);
- *Old Version Image window* - used to navigate through Old Version Image components, folders and files and to set their properties (lower window);

Depending on the operation and status, files and folders are highlighted with a certain color.
Legend Bar is placed at the bottom of Version Difference Editor window. To change the default color - double click on the appropriate item in the Legend Bar.

Each of those windows consist of two subwindows:

- *Version Image file browser window* - used to navigate through New Version Image objects;
- *Properties window* - used to set properties necessary for difference / patch building or update installation.

Old/New Version Image Browser

This window can be used to select necessary Version Image or a Component inside it, to set/observe properties of selected objects.
To change object property - select it in the appropriate *Version Image* and then change necessary property value in

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

the *Properties window*.

By default ***Synchronous selection*** mode when pairs of files in old and new version images are highlighted (selected) synchronously is turned on. You can switch it off by selecting the appropriate item in ***Files*** menu.

Tabstops header on the top of each window allows sorting of the file list by *Name*, *Date* and *Size* by clicking on the appropriate header. Clicking a second time on the same header reverses the order. Arrow to the left of the header text shows the sort direction.

Properties

Main properties that can be set are listed in **Old_Version_Side_Properties**, **New_Version_Side_Properties**, ***Add/Replace Options*** and ***Df build Options*** sections of this manual.

Each property of difference elements has such implicit attribute as "***Source of the obtained value"***.
This attribute defines the method of determining the particular value of the Update command.
The value can be defined by:

- by the corresponding version's property values (**<Defined by version>**, value is prefixed with ' @ ' symbol in the list),
- by version's property value only if the corresponding parent object's property value is defined by version's property values (**<Defined by parent>**, value is prefixed with ' * ' symbol in the list)
- by parent's property value. (**<Inherit>**, value is prefixed with ' $ ' symbol in the list)
- individual value, which you can set by clicking with right mouse button on the appropriate file. Available values and their descriptions are listed in the **Update Command generation** section of this manual.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4.4. Old Version Side Properties group

Old Version side Properties are used during compilation of the difference between versions when selected version is considered as an *old* one.

Parameter	Available options	Function
Old Version Side		
Version Key	Yes, **No**	Set selected file as a ***version key***
Update Method	**Auto,** Don't allow update by diff, Don't update	Set the **update method** .
Allow delete	**Yes**, No	Allow old file/folder deletion on the end-user's machine.
Backup for deleted	**Yes**, No	Backup file/folder if it should be deleted on the end-user's machine at patch applying.
Allow delete on reboot	**Yes**, No	Allow file/folder deletion after next system reboot (if selected file/folder was locked during update installation).
Delete read-only	Yes, **No**	Allow selected file/folder deletion if it is marked as read-only on the end-user's machine.

** Default properties values are marked with Bold font.*

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4.5. New Version Side Properties group

New Version side Properties are used during compilation of the difference between versions when selected version is considered as a *new* one.

Parameter	Available options	Function
New Version Side		
Patch method	**Auto,** Added or Replaced, Always Added, Always Replaced, Don't patch	Set the **patch method** .
Ignore df apply error	Yes, **No**	Ignore df apply error during update installation. If set to Yes and the file is updated with the "*Updated*" command then the old file will remain unchanged, and no message will be displayed to an end-user if update was not performed correctly due to some changes in the old file.
Backup	**Yes**, No	Make backup copy.
Allow patch on reboot	**Yes**, No	Allow file/folder patch after system reboot (if selected file/folder was locked during update installation).
Overwrite read-only	**Overwrite if backuped, else user prompt,** Overwrite if backuped, else skip, Overwrite without user prompt, Skip without user prompt, Defined by user	Overwrite selected file if is marked as read-only on the end-user's machine.

** Default properties values are marked with Bold font.*

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4.6. Add/Replace Options

Add/Replace Properties define the replacement procedure of the file that already exists on the end-user's machine (this group do not affect "Updated" files , i.e. when the update procedure is performed using a difference between two files).
If a file id defined as added or replaced in the update module, then it can be necessary to prevent replacement of newer files. To determine if existing file is newer the update program tries to compare last modified date and time, or version information (for PE-files). Thus if according to comparing results the file to be replaced is considered as *older* then it is replaced without any confirmations (similarly for "Added" files). Otherwise property "If older" is used to define the action to be made in such situation.

Parameter	Available options	Function
Add/Replace Options		
Check if older by	**Version only,** Version or Date/time, Date/time only, Don't compare	Defines method to determine if the file is newer/older or not determine at all during update installation on the end-user's machine. If set to "Don't compare" (i.e. do not perform this check) then replacement is always made. "Added" files replacement is controlled by "If added file exist" property.
If older	Skip without user prompt, Defined by user	Defines action to be made if the file to be replaced is newer (and replacing file is older).
If added file exists	**Replace if backuped, else user prompt,** Replace if backuped, else skip, Replace without user prompt, Skip without user prompt, Defined by user	Defines action to be made if the file (that is set as added) already exists on the end-user's machine and the check which file is older or newer is not implemented. Valid only for *Added* files and only if "Check if older by" = Don't Compare, or identification which file is older or newer failed (for instance, if file resources do not contain version information).

** Default properties values are marked with Bold font.*

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4.7. Df Build Options

Parameter	Available options	Function
Df build Options		
Comparing method	Fastest, Normal, **Good**, Best, Paranoid, Ultimate	Set the comparing method used to build file difference. **WARNING!** *'Paranoid'* and *'Ultimate'* methods can be very slow for big-size files!
Memory usage	5% - 100% (**80%** by default)	Set limit of the memory (available in the system) usage.
Ignore attributes diff	**Yes**, No	Ignore attributes difference during difference building.
Compress	**Yes**, No	Compress the result difference file(s). Do not set to No without necessity, especially if you want to build a cumulative patch module performing the update to selected version.
Test	**Yes**, No	This option forces df-file verification test via its simulation after df-file building successfully completed.
Out-of-date detection	Don't use, **Simple**, MD5, Test	Check whether df-file rebuild is necessary. Defines df-file out-of-date detection method. "Don't use": always rebuild df-file. "Simple": simple detection method which includes comparing of file sizes, last modified date/time, attributes and file names. "MD5": in addition to "Simple" method provides md5 checksum verification of both old and new files. "Test": in addition to "MD5" method provides df-file verification test via its simulation.

** Default properties values are marked with Bold font.*
*** More detailed description of "Df build options" you can find in dfbuild.txt or by executing "dfbuild.exe -h".*

NOTE: Setting *Memory usage* limit can be used in conjunction with selecting of comparing algorithm to tune up the ratio of quality and speed. Increasing the memory usage limit will result first of all in the advanced comparing quality. **Be careful with setting of this parameter!** Setting the limit above total physical memory available in your system can result in significant slowing down of the comparing process.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4.8. Update Command generation

- **Update command generation**

The result Update command is defined by the combination of the following properties:

- *Update Method* and *Allow Delete* properties values of the old version element (if exists),
- *Patch Method* property value of the new version element (if exists), and the fact of presence/absence of this element.

- ***Update method:***

Parameter "*Update method*" is used to specify how the selected element is updated if this element is placed inside an *Old Version image* of a difference.

The result value of the Update command is defined in concordance with the value of "*Patch method*" of the same-name element in the *New Version image* (if exists) and also by "*Allow delete*" parameter value.

- ***Patch method:***

Parameter "*Patch method*" is used to specify how the selected element is updated if this element is placed inside a *New Version image* of a difference.

The result value of the Update command is defined in concordance with the value of "*Update method*" of the same-name element (if exists) in the *Old Version image*.

The Update command value can take on the following values:

- "Updated" : new file will be created on the basis of an old file contents and binary difference between old and new files which is stored in the update module. Therefore an old file must remain unchangeable on the end-user's machine to perform the update procedure successfully.
- "Replaced/Added" : the update module contains the full copy of the new file if it differs from an old file. Command "Added" (unlike Replaced) usage assumes that there is no old file with the same name on the end-user's side, and the action in case of its presence is defined by the value of "Check if older by" and "If added file exists" properties.
- "Deleted" : deletion of an old file or empty folder.
- "No Operation" : no action with file or folder is made on the end-user's side. This command is set for identical files (in compliance with the property "Ignore attributes diff"), or can be manually set by user.

Particular value of the generated *Update command* for selected element is defined by the following table below.

			<Element does not exist>	**Update method**		
				Auto	**Don't allow update by diff**	**Don't update**
<Element does not exist>	**Allow delete**	YES	-	Deleted	Deleted	No operation
		NO	-	No operation	No operation	No operation
Patch method	**Auto**		Added	Updated	Replaced	No operation
	Added or Replaced		Added	Replaced	Replaced	No operation
	Always Added		Added	Added	Added	No operation
	Always Replaced		Replaced	Replaced	Replaced	No operation
	Don't patch		No operation	No operation	No operation	No operation

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4.9. Command line parameters

The Update Installation program accepts optional command line parameters. These can be useful to system administrators, and to other programs calling the Setup program.

/SILENT, /VERYSILENT
Instructs Setup to be silent or very silent. When Setup is silent the wizard and the background window are not displayed but the installation progress window is. When a setup is very silent this installation progress window is not displayed.
If a restart is necessary and the '/NORESTART' command isn't used (see below) and Setup is silent, it will display a Reboot now? message box. If it's very silent it will reboot without asking.

/LOG
Causes Setup to create a log file in the user's TEMP directory detailing file installation and [Run] actions taken during the installation process. This can be a helpful debugging aid. For example, if you suspect a file isn't being replaced when you believe it should be (or vice versa), the log file will tell you if the file was really skipped, and why.

The log file is created with a unique name based on the current date. (It will not overwrite or append to existing files.)
The information contained in the log file is technical in nature and therefore not intended to be understandable by end users. Nor is it designed to be machine-parseable; the format of the file is subject to change without notice.

/LOG="filename"
Same as /LOG, except it allows you to specify a fixed path/filename to use for the log file. If a file with the specified name already exists it will be overwritten. If the file cannot be created, Setup will abort with an error message.

/NOCANCEL
Prevents the user from cancelling during the installation process, by disabling the Cancel button and ignoring clicks on the close button. Useful along with '/SILENT' or '/VERYSILENT'.

/NORESTART
Instructs Setup not to reboot even if it's necessary.

/RESTARTEXITCODE=exit code
Specifies the custom exit code that Setup is to return when a restart is needed. Useful along with '/NORESTART'.
Also see Setup Exit Codes.

/LANG=language
Specifies the language to use. language specifies the internal name of the language as specified in a [Languages] section entry.
When a valid /LANG parameter is used, the Select Language dialog will be suppressed.

/PASSWORD=password
Specifies the password to use. If the [Setup] section directive Password was not set, this command line parameter is ignored.
When an invalid password is specified, this command line parameter is also ignored.

- **Exit codes :**

0 Setup was successfully run to completion.

1 Setup failed to initialize.

2 The user clicked Cancel in the wizard before the actual installation started, or chose "No" on the opening "This will install..." message box.

3 A fatal error occurred while preparing to move to the next installation phase (for example, from displaying the pre-installation wizard pages to the actual installation process). This should never happen except under the most unusual of circumstances, such as running out of memory or Windows resources.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4 A fatal error occurred during the actual installation process.
Note: Errors that cause an Abort-Retry-Ignore box to be displayed are not fatal errors. If the user chooses Abort at such a message box, exit code 5 will be returned.

5 The user clicked Cancel during the actual installation process, or chose Abort at an Abort-Retry-Ignore box.

6 The Setup process was forcefully terminated by the debugger (Run | Terminate was used in the IDE).

Before returning an exit code of 1, 3, or 4, an error message explaining the problem will normally be displayed.
Future versions of InnoSetup may return additional exit codes, so applications checking the exit code should be programmed to handle unexpected exit codes gracefully. Any non-zero exit code indicates that Setup was not run to completion.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

4.10.Patch applying

Patch module prepared with the help of PatchFactory is an executable module (Windows GUI application), that incorporates update data and apply parameters.
You can customize the appearance of update program by yourself at patch building step.

NOTE: It is recommended to apply the patch created on systems under Windows NT4/W2K/XP/2003 (on NTFS volumes) with Administrator rights.

- **Interface of the update program:**

All features of patch module installation (except file update procedure) are implemented using standard InnoSetup resources.
Source codes of scripts used are available for free modification / enhancement.

- **Title bar:**

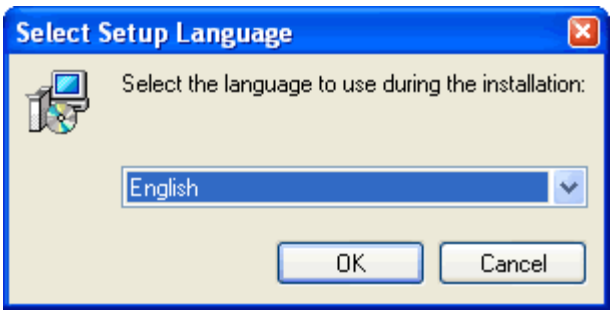
The title shows the application name including version: < Setup - "Product name" update >.

- **Update Installation Wizard Dialogs:**

These wizard dialogs guide you to apply the patch package for updating your application/software files or folders.
To apply the patch you should execute patch file and follow the instructions in *Update Installation Wizard*.

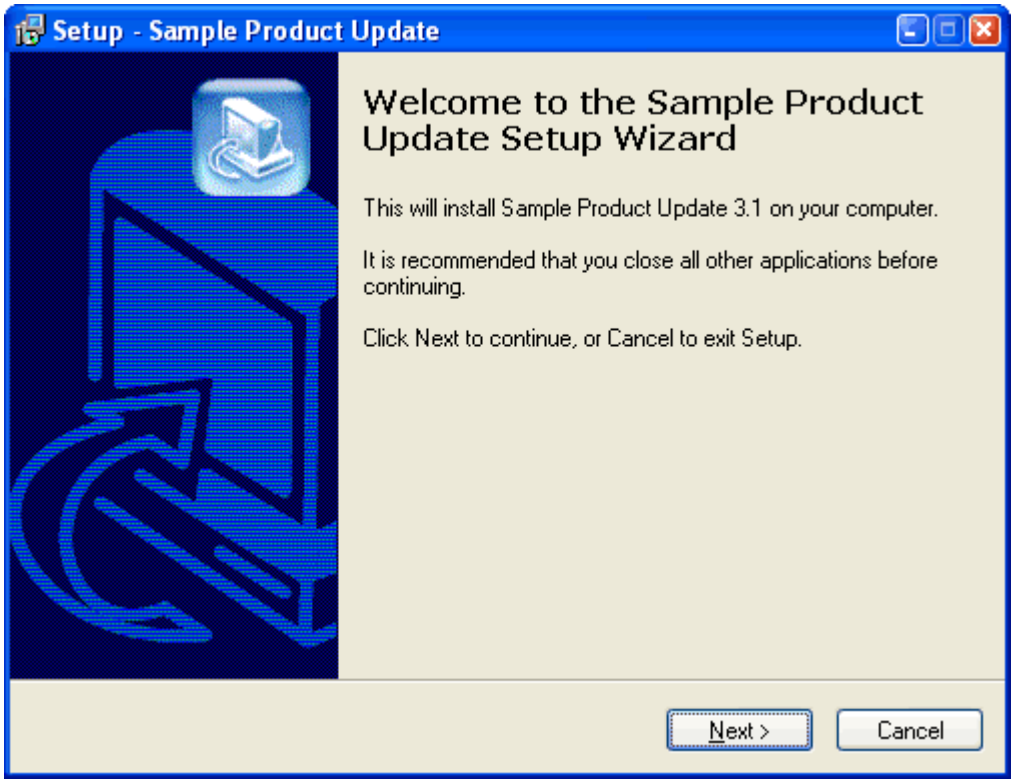
The maximum number of main *Update Installation Wizard* dialogs is 7 which include:

- **Select Setup Language dialog** : Here you can select the appropriate language for all wizard dialogs.

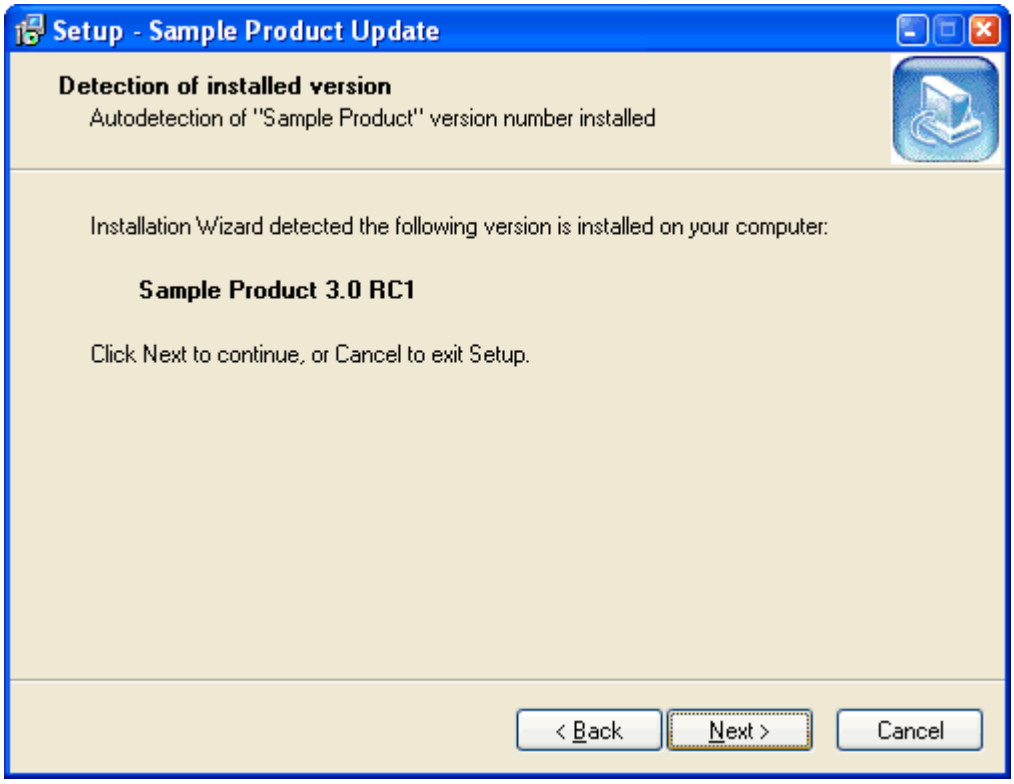


- **Welcome dialog** : This is a welcome dialog that shows main installation instructions to your end-user.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

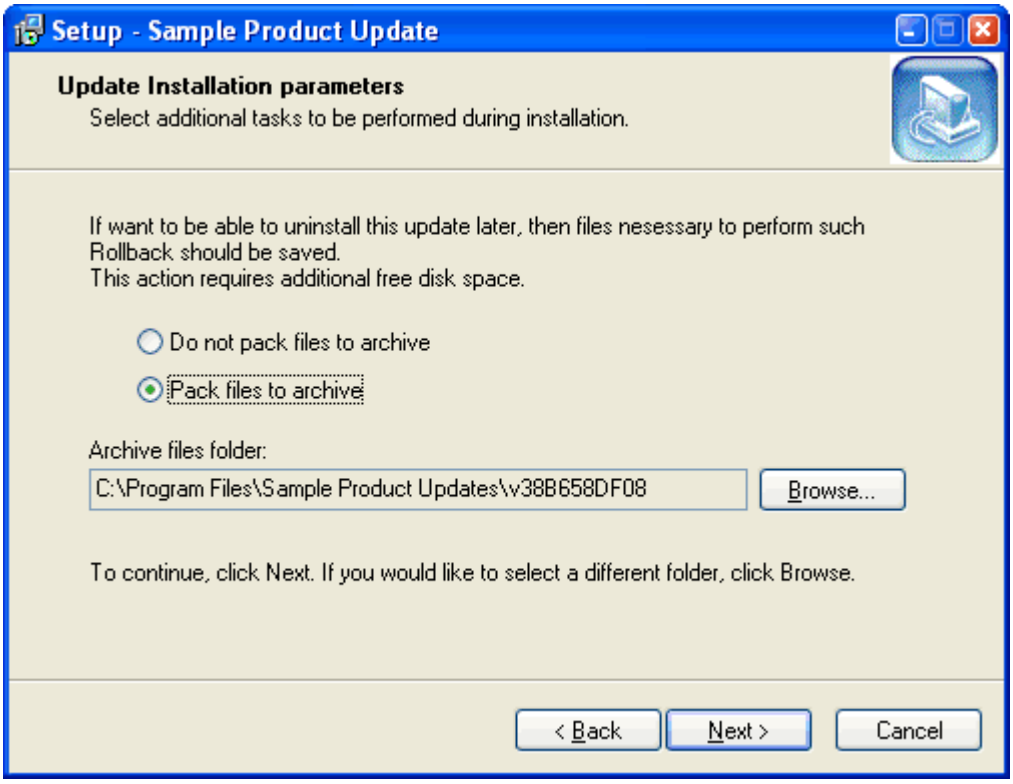


- **Detection of installed version** : Here you can observe which version number is determined. If Installation Wizard is not able to determine the version that he can update the application to then you'll see a warning - in this case installation will not continue.

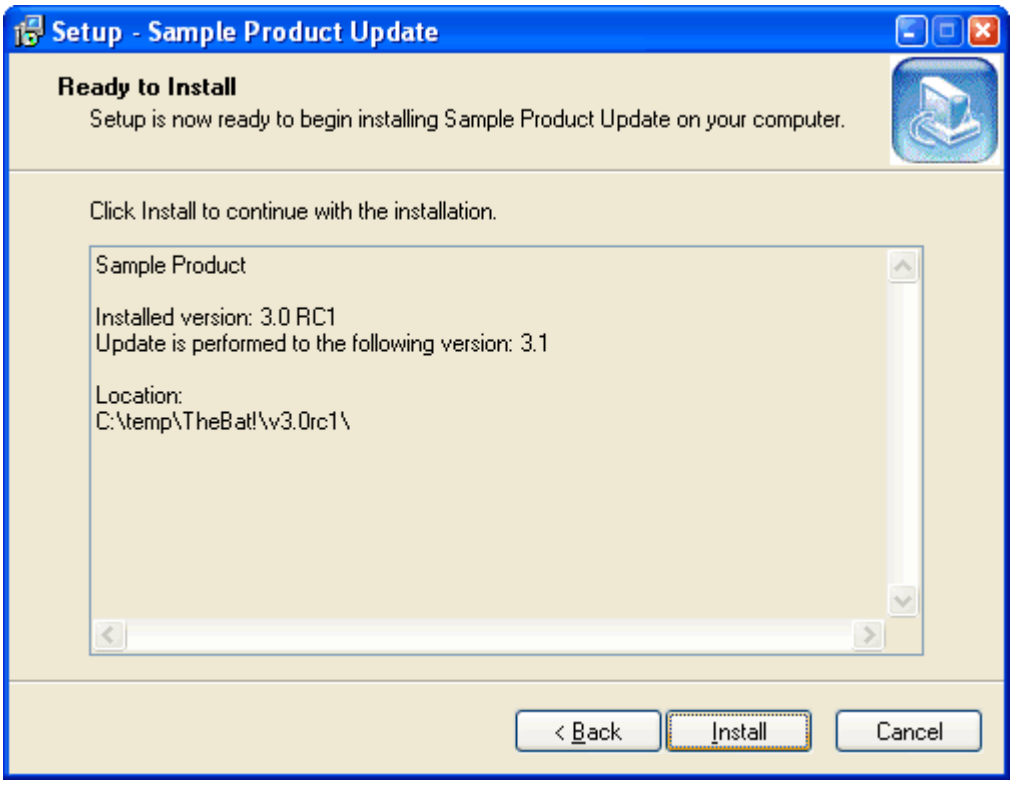


- **Update Installation parameters** : This dialog (if enabled) provides you with an option to be able to perform Rollback of the Update module installed. You can specify a folder where all files necessary to perform a Rollback will be stored.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

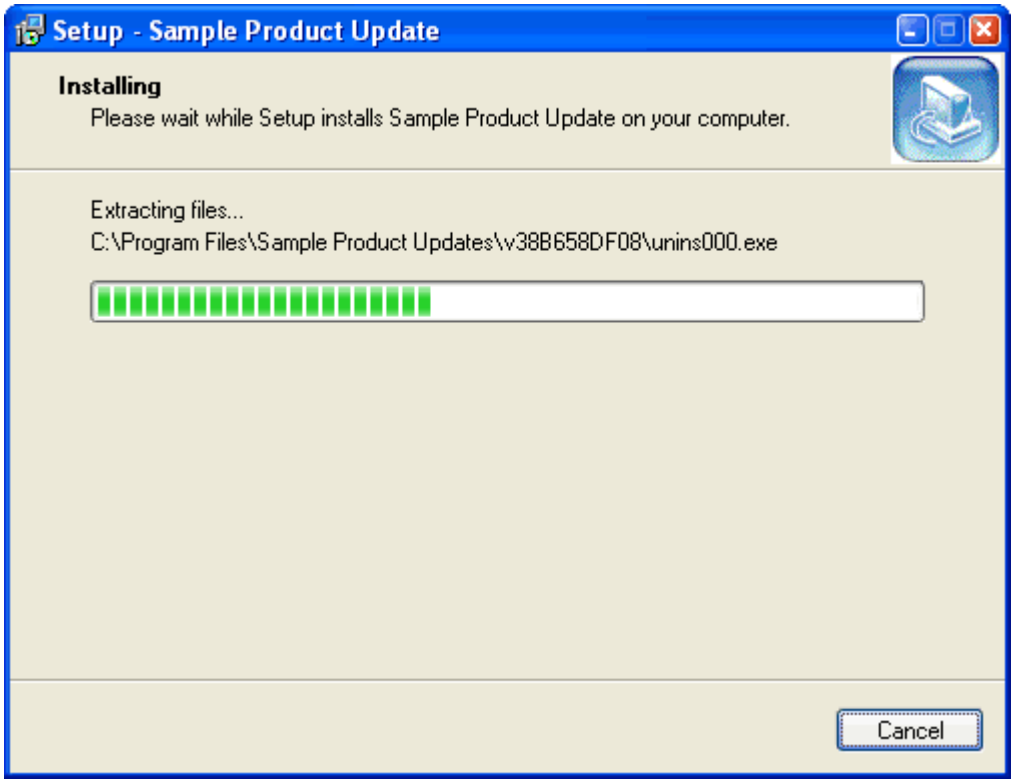


- **Ready to Install** : Here you can see the summary of all update installation tasks. Click <Install> to perform the Update Installation.

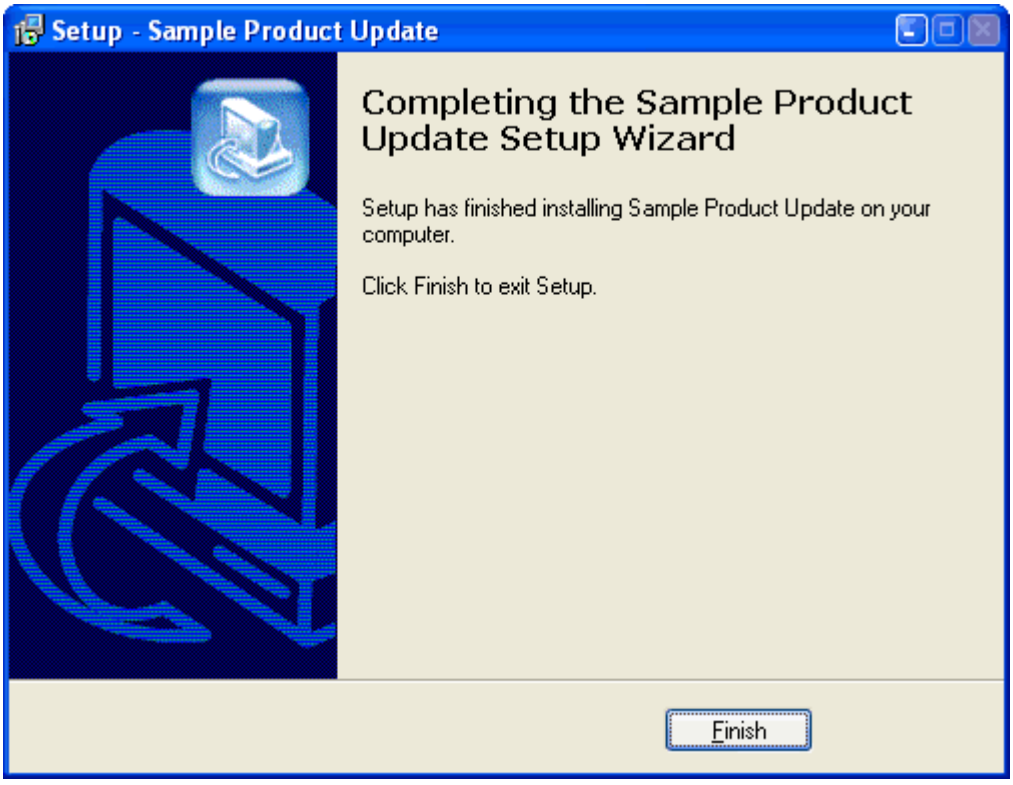


- **Installation progress** : This page is shown during the actual installation process and displays all files that are being updated.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com



- **Finish** : This dialog indicates the status of patch installation: whether it was completed successfully or not.



© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

5. Frequenlty asked questions

To view the latest FAQs you can visit our public website at http://www.agensoft.com/faq.html.

1. General Questions:

2. Technical Questions:

1. General Questions:

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

1.1 What is the main idea of building update modules?

The main idea of building patches is that patch file represents only information concerning changes made to an old version software product files relatively to a new version software product files. And if these changes are not significant relatively to total size of new version files than such delivering of update module (patch file/patch module) can become more effective method of update delivering.

Software has its bugs. These bugs are often discovered after the official release of the product. You are getting bug reports from your users. It's terrible that you spent your money to create fancy box, to record CD-ROM's and to distribute your software and several days after they appear. So, you should use "patch" which just contains description of changes you have made to your product since the official (or just previous) release. What's more, the difference between previous and current version to keep the patches as small as possible.

1.2 What's the advantage of buying a Commercial or a Corporate License? What is your licensing policy?

- the discounted **Personal License** is offered as a service to the industry, primarily for single person companies with little revenue (such as shareware authors). The software is licensed to the name of the individual purchasing the license.
- the standard **Commercial License** is intended for small companies. The software is licensed to the name of the company purchasing the license. Maximum number of employees using the software is limited to 5. It also entitles an organization to receive high-priority support *(with guaranteed answer within 2 business days)* via email. If you would like to obtain more extended services or more employees to use the software - consider buying the Corporate license.
- the premium **Corporate License** is intended for large companies and corporates with many employees. The software is licensed to the name of the company purchasing the license. And besides this type of license entitles an organization to receive one copy of the distribution software and to duplicate the software for any number of people or workstations within the corporation. It also entitles an organization to receive high-priority support *(with guaranteed answer within 1 business day)* via email and free major version upgrades during lifetime of the product.

All licenses include the royalty-free distribution of the PatchFactory Client for an unlimited number of applications, free minor version updates with special 50% discount for major upgrades and priority technical support via email.

1.3 Can I get the software on disk or CD media?

You can include CD with installation file if you order PatchFactory software at ShareIt!. It is shipped via Air Mail, within 2 business weeks. CD-ROM is free if buying a Commercial or a Corporate license, and 9.95$ value for Personal license. At the subscription time in the future, if you lose your copy of the software due to a hard drive failure you can download the latest trial version from our site. If for any reason your registration code becomes invalid or lost - just send an e-mail to Sales Support with details of your previous registration. You will receive a new code via e-mail at no charge as soon as we validate your old registration data.

1.4 Will I get 'SPAM' if I give you my e-mail address?

No. We never, ever give out customer information or e-mail addresses in any manner.

1.5 Can I order via fax machine or phone?

Our sales agency now offers both a Toll-Free Voice Registration Service and a Fax Registration Service for buying PatchFactory Software. To order software using the Toll-Free Voice Registration Service, please check links within online secure order forms after selecting an appropriate license at our Order page. You must have PatchFactory Product ID ready.

1.6 Can I purchase with a check?

Our sales agency gladly accepts payment by check or money order. However, please allow adequate time for the funds to be processed by our bank. For information about where to send your check or money order, please check links within online secure order forms after selecting an appropriate license at our Order page. You must have PatchFactory Product ID ready.

1.7 What are your product IDs at RegSoft and ShareIt?

Please, visit our online FAQs page at our public website www.agensoft.com/faq.html

1.8 How long does it take to get my registration code after I purchase a product online?

After our sales agency receive your online credit card order, it may take up to several hours to authorize your transaction. As soon as your charge is authorized, you will receive an authorization e-mail with your Tracking ID# as well as instructions on how to obtain the full registered version the product you ordered. It is important that the

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

customer check his or her e-mail to obtain the charge authorization and instructions. In the unlikely event that your credit card is declined, you will receive an e-mail stating the reason. If you do not receive any e-mail within 48 hours - there may be a problem with your order. In that case, kindly contact us by e-mail at sales@agensoft.com with your name, Tracking ID# and the approximate date and time of your order to obtain the status of your order.

1.9 Do I have to license each patch file created...?

To use PatchFactory software and to create as many patches as you wish you have to buy only one license (if you are planning PatchFactory to be used by more than one user on one computer of the local network (if applicable) than you should order the corresponding number of licenses).

And furthermore you may distribute created patch packages to as many users as you want.

1.10 What is the policy on updates...how much do they cost?

All minor updates are free as of this writing (subject to change). Minor updates are those where the software version number to the right of the decimal change (minor updates usually slightly differs from each other), but the digit to the left of the decimal stays the same. For example, updates from 3.0 to 3.1 are free, however, 2.x to 4.0 will be on a cost basis.

Additionally, if a new major version does get issued, it is offered to our current customers at a discounted rate (50% discount) - it means that to register PatchFactory major version update you will have to pay only 50% of its total price for each license to be renewed (for instance, $50 USD for one license renewal if the price is $100) to renew your registration.

To renew your registration of PatchFactory software just send an email message with your previous registration information to sales@agensoft.com using your email client or using our online email-form to get the instructions of how to renew your registration. As soon as we are notified that your order has been processed, a new registration key will be sent to you via email to unlock your copy of PatchFactory software.

1.11 How does PatchFactory can help prevent software piracy?

PatchFactory helps to prevent the illegal distribution of your software, by requiring that the software be properly installed at the time of the update. The patching process demands that the original files be installed before they can be modified. Files that have been tampered with will not be changed, and the update will fail.

Pirated versions of your software, that have files modified in them, will not be updated.

1.12 Which development environments can I use with PatchFactory?

PatchFactory is a stand-alone application that does not require the use of a separate development environment.

Consequently, you may use PatchFactory to provide updates for any type of Windows-based application, regardless of which programming language was used to develop it.

1.13 Does the PatchFactory *client-side* patch program require any third-party libraries to run?

The PatchFactory Client-side patch program is entirely self-contained: it does not use any 3rd party libraries, and only requires libraries which are part of the base operating system.

With PatchFactory, you have the assurance that customers with different hardware and software configurations will always be able to update your software.

1.14 What are the advantages for distributing patches?

Distributing the changes as the "patch" has several significant advantages over distributing new version of the product. First, you don't have to make new CD-ROM's (or multiple floppy disks) and new boxes.

The patches are usually small and easy to distribute on single floppy or over the Internet. Due to the differential nature of the patches, you can also distribute your patches freely (from your web page for example), because it is impossible to install patch without previous (bought and registered) version of product.

1.15 What are the limitations of the evaluation version of PatchFactory v3?

Evaluation version of PatchFactory has the following limitations:

- *30-day trial period*, during which you can evaluate PatchFactory for free. When this period expires you must either purchase our software or stop using it and remove it from your computer.
- *number of files*, contained in either old or new version images for which you can build a difference *is limited to 100* (one hundred).

1.16 What is the difference between patching and incremental updating?

Incremental update contains all the files which have been changed between two versions.

Update modules made with PatchFactory consists only of the changes from within each individual file with the

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

help of byte-level differencing technology used by our patching engine, resulting in a significantly smaller update size.

1.17 ERROR#1: Limitation of evaluation version: the number of files/folders in the difference can not exceed 100.

- Evaluation version of PatchFactory which is available for download from our public web-site contains a functional limitation, and namely the number of files, contained in either old or new version images for which you can build a difference is limited to 100 (one hundred).
- To request for a fully-functional demo, please use our online email-form to contact us (*guarantees that your email will reach our Support Team*). Do not forget to fill in all necessary fields. **Only requests from our online email form are taken into consideration!** We request your apologies for any inconvenience.

2. Technical Questions:

2.1 I am trying to compare two files which changed a bit. I used ASPack (or other compression utility) to reduce the file size. When I create the patch - its size is larger, which is not really what I wanted.

The basis of the comparing algorithm is the ability to find concurrence in compared files at a level of octet-byte subsequences.

Use of packing utilities to lower the file size most often leads to that the result archive files lose the similarity at the byte-level. Such behavior is peculiar for most of lossless compression algorithms.

In this case the most appropriate solution to achieve effective comparing results is not to use aspack (or other compression utility) at all.

Here are some recommendations to reduce the size of output patch file in this case:

- Try to reduce files size by moving unchanged parts of the program (viz. their invariability from version to version or their insignificant changes) to dll-modules.
- Do not apply Exe-compression utilities to deflate executable files (if there is no extreme necessity).

But nevertheless PatchFactory considers all these features of EXE and DLL files and provides optimal patch building in these cases.

2.2 How can I provide backup copy of old files on the end-user's machine to prevent their loss at patch applying?

PatchFactory v3 provides such feature as Rollback / Uninstall of the update module. To provide Uninstall capability you should set "***Uninstallable***" property to **Yes** for your Patch module. If set to **Yes**, the backup copy of all replaced/modified files (for which backup is allowed) is saved and a record is added into Add/Remove programs list.

If you don't want a dialog window offering to select whether to perform backup copy or not to be shown to your end-user - set property "***Show Backup page***" to **No**.

2.3 I need to update a database with things like adding columns or so. Does your software could help me to patch the database using sql or it can replace old files...?

PatchFactory does not deal with any specific data structures, it operates with files and directories.

Databases or files of other formats can be updated only as binary files (**warning:** database update can be implemented only if it is not changed on the end-user's machine).

2.4 How to create an update module that can run *silently* (i.e. no user intervention, pop-up windows, etc..) ...?

By default the Update Installation is done in a wizard mode (i.e. user's interaction during installation is required) but nevertheless the capability to run installation in a silent mode is provided for. To do a silent install you can use the following optional command line parameters of the update module.

/SILENT, **/VERYSILENT**

- *Instructs Installation to be silent or very silent. When Installation is SILENT the wizard is not displayed but the installation progress window is. When Installation is VERYSILENT this installation progress window is also not displayed.*

These parameters can be useful to system administrators, and to other programs calling the Update Installation program.

2.5 How can I localize PatchFactory client dialogs into my language?

To localize patch module dialogs to your language please, carefully read the Localization Instructions.

You can get a considerable discount! Please, do not send your language file prior to contacting us (required) via email or via an appropriate Support Forum, as a new version with exactly this language support can be already

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

under development.

2.6 I'm trying to build a patch for a Gigabyte-sized file. Can your product handle files of this size? Any ideas on time per Gb and processor utilization?

The speed and quality of the comparing depends on the similarity of files to be compared. Please, have a look at the parameters description of the console program dfbuild.exe (see dfbuild.txt or dfbuild.exe /?) and tuning recommendations prior to perform the comparing of such big files.

Additional recommendations:

- Try to perform comparing directly using console program dfbuild.exe in order to choose optimal comparing parameters.
- Perform comparing with options "-1 -mem=10" set. If the obtained results do not satisfy your requirements - try executing without additional parameters (i.e. with parameters set by default).
- Add these files to version images as links.

dfbuild program has a set of parameters which provides control under speed & quality of the comparing algorithm performance.

Here are brief tuning recommendations for setting of these parameters which can be helpful in some applications:

- If you are processing large files (~500Mb and more) first try to use 'dfbuild' with parameters providing the maximum speed:

```
dfbuild -1 -mem=1 <old_file> <new_file>
```

- If the result is not acceptable (df-file size is too big) try to increase the amount of system memory utilized using '-mem' parameter. And only after exhausting its potentialities try to change the 'comparing method' parameter.
- However the size of the result patch-file and the speed of its building depends not only on the parameters set of the comparing algorithm but also on the nature and the similarity ratio of particular files to be compared.

2.7 Error applying df-file with EDF_OLD_MD5_ERROR code. What is wrong?

In this case there was a mismatch of MD5 checksum for an old (updated) file, which is stored on the end-user's machine and MD5 checksum calculated during patch building (i.e. the expected one), i.e. these files were different by content.

If this occurrence is expected and normal (i.e. if this file on the end-user's machine can be changed), then you should set the option "Update method" to the value "Don't allow update by diff" in the UPDATED (OLD) version.

You can also set option "Patch method" within the NEW version unequal to "Auto" value (the particular value should be selected according to your specific requirements).

Otherwise if this file is not expected to be changed on the end-user's machine then you should look for the reason of its modification by yourself.

2.8 How can I customize the Update Installation wizard images?

There are two methods available which you can use to customize the Update Installation wizard images:

1. Replace the original wizard images in "<PatchFactory Installation Folder>\Scripts\" folder, where
   Setup.bmp - vertical installer image
   SetupSmall - right-top small installer image.
2. Modify the "default.iss" patch applying script-file, located in "<PatchFactory Installation Folder>\Scripts\" folder.

- find the lines
  WizardImageFile=${SCRIPT_PATH}Setup.bmp and
  WizardSmallImageFile=${SCRIPT_PATH}SetupSmall.bmp
  and replace "${SCRIPT_PATH}Setup.bmp" and "${SCRIPT_PATH}SetupSmall.bmp" with the full (or relative to the Scripts folder) path to your installer images.

2.9 How can I Register a DLL/OCX library? I need to somehow register dll's after they are installed.

To register/unregister a DLL/OCX library, you can add several lines to the end of the 'default.iss' script-file according to the example listed below.

'default.iss' script-file is located in the "<PatchFactory3 Installation Folder>\Scripts" folder.

```
#----------------------------------#
function GetMyAppPath(new_comp_id: String): String;
 begin
   Result := GetPathForComp(NewVer,new_comp_id);
 end;
[Run]
```


© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

```
;; Register app.dll, stored in the Component with "MyComponent1" ID
  Filename: "{sys}\regsvr32.exe"; Parameters: """{code:GetMyAppPath|MyComponent1}\app.dll
;; Unregister app.dll, stored in the Component with "MyComponent1" ID
  Filename: "{sys}\regsvr32.exe"; Parameters: "/u ""{code:GetMyAppPath|MyComponent1}\app.
#----------------------------------#
```

Replace "MyComponent1" with the ID of the component specified in the "Component ID" property of your version component (not its Name but ID) where app.dll is located.

For more information, please read Inno Setup manual - section "How to use: Setup Script Sections -> [Run] section".

Inno Setup manual location : **"<PatchFactory3 Installation Folder>\Inno\ISetup.hlp"**

2.10 How can I write to the system Registry?

To add a record to the system Registry, you can add several lines to the end of the 'default.iss' script-file according to the example listed below.

'default.iss' script-file is located in the "<PatchFactory3 Installation Folder>\Scripts" folder.

```
#----------------------------------#
[Registry]
     Root: HKCU; Subkey: "Software\My Company\My Program"; ValueType: string;
ValueName: "InstallPath"; ValueData: "{app}"
#----------------------------------#
```

For more information, please read Inno Setup manual - section "How to use: Setup Script Sections -> [Registry] section".

Inno Setup manual location : **"<PatchFactory3 Installation Folder>\Inno\ISetup.hlp"**

2.11 How can I run an executable or open another file after update is finished?

To run an executable, you can add several lines to the end of the 'default.iss' script-file according to the example listed below.

'default.iss' script-file is located in the "<PatchFactory3 Installation Folder>\Scripts" folder.

```
#----------------------------------#
function GetMyAppPath(new_comp_id: String): String;
   begin
   Result := GetPathForComp(NewVer,new_comp_id);
 end;
[Run]
       Filename:   """{code:GetMyAppPath|MyComponent1}\app.exe""";   Flags:   nowait
skipifdoesntexist
#----------------------------------#
```

Replace "MyComponent1" with the ID of the component specified in the "Component ID" property of your version component (not its Name but ID) where app.exe is located.

To open a file which is not a directly executable file (an .exe or .com file) add the "shellexec" flag. (Flags: nowait skipifdoesntexist shellexec) This flag is required if Filename is not a directly executable file (an .exe or .com file). When this flag is set, Filename can be a folder or any registered file type -- including .hlp, .doc, and so on. The file will be opened with the application associated with the file type on the user's system, the same way it would be if the user double-clicked the file in Explorer.

By default, when the shellexec flag is used it will not wait until the spawned process terminates. If you need that, you must add the flag waituntilterminated. Note that it cannot and will not wait if a new process isn't spawned -- for example, if Filename specifies a folder.

For more information, please read Inno Setup manual - section "How to use: Setup Script Sections -> [Run] section".

Inno Setup manual location : **"<PatchFactory3 Installation Folder>\Inno\ISetup.hlp"**

2.12 How can I specify the minimum user privilege required to run the update?

To check if the user is logged on as administrator and not run the patch if they are not, please add the following line inside the [Setup] section

of the 'default.iss' script-file.

'default.iss' script-file is located in the *"<PatchFactory3 Installation Folder>\Scripts"* folder.

Please find the following line

```
;PrivilegesRequired
```


© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

inside 'default.iss' file and replace it with the desired value according to the example listed below.

`PrivilegesRequired=admin`

Valid values: none, poweruser, admin

Default value: none

Description:

This directive specifies the minimum user privileges required to run the update. When set to poweruser or admin, Setup will give an error

message at startup (e.g. "You must be logged in as an administrator when installing this program") if the user doesn't have at least Power User

or administrative privileges, respectively. This only applies to Windows NT platforms.

For more information, please read Inno Setup manual - section "How to use: Setup Script Sections -> [Setup] section".

Inno Setup manual location : **"<PatchFactory3 Installation Folder>\Inno\ISetup.hlp"**

If any question is not covered here, ask us: support@agensoft.com regarding technical issues (such as bug reports, feature suggestions, etc.)

and at sales@agensoft.com regarding sales issues (ordering problems, partnership suggestions, etc.)

You can also use our online email-form (preferably) to contact us.

We'll get in touch with you as soon as possible (usually within two business days).

* Do not forget to provide us with necessary technical information (Windows version, Detailed description of your problem,

and your registration information, if you are a registered user).

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

6. Registration and Support

6.1. License Agreement

END-USER LICENSE AGREEMENT

IMPORTANT: This END-USER LICENSE AGREEMENT ("EULA") is a legal agreement between you, either as an individual or a single entity ("Customer"), and AgenSoft Inc. ("AGENSOFT") for all AGENSOFT products which may include executable programs, redistributable modules, controls, dynamic link libraries, source code, demos, intermediate files, media, printed materials, and "online" or electronic documentation ("SOFTWARE PRODUCT(S)" or "SOFTWARE").

AGENSOFT grants to you as an individual, a personal, nonexclusive license to install and use the SOFTWARE PRODUCT(S) for the sole purposes of designing, developing, testing, and deploying application programs which you create. By installing, copying, or otherwise using the SOFTWARE PRODUCT(S), you agree to be bound by the terms of this EULA. If you do not agree to any part of the terms of this EULA, DO NOT INSTALL, USE, EVALUATE, OR REPLICATE IN ANY MANNER, ANY PART, FILE OR PORTION OF THE SOFTWARE PRODUCT(S).

All SOFTWARE PRODUCT(S) are licensed, not sold. If you are an individual, you must acquire an individual license for the SOFTWARE PRODUCT(S) from AGENSOFT or its authorized resellers. If you are an entity, you must acquire and assign an individual license for each Customer within your organization using and/or developing with the SOFTWARE PRODUCT(S) or acquire the appropriate license type (if applicable) from AGENSOFT or its authorized resellers.

If the SOFTWARE PRODUCT(S) you have obtained is marked as a "TRIAL" or "EVALUATION," you may install one copy of the SOFTWARE PRODUCT(S) for testing purposes for a period of 30 calendar days from the date of installation ("Evaluation Period"). You may not attempt to bypass or disable any of these limitations. These restrictions may only be removed by purchasing a license to use the SOFTWARE PRODUCT(S) from AGENSOFT. Upon expiration of the Evaluation Period, the SOFTWARE PRODUCT(S) must be uninstalled and all copies destroyed.

RIGOROUS ENFORCEMENT OF INTELLECTUAL PROPERTY RIGHTS. If the licensed right of use for this SOFTWARE PRODUCT(S) is purchased by you with any intent to reverse engineer, decompile, create derivative works, and the exploitation or unauthorized transfer of, any AGENSOFT intellectual property and trade secrets, to include any exposed methods or source code where provided, no licensed right of use shall exist, and any PRODUCT(s) created as a result shall be judged illegal by definition of all applicable law. Any sale or resale of intellectual property or created derivatives so obtained will be prosecuted to the fullest extent of all local, federal and international law.

1. LICENSE GRANT

AGENSOFT grants you a license to use the version of this SOFTWARE PRODUCT(S) according to the license type you have purchased. You may not modify the SOFTWARE PRODUCT(S) or disable any licensing or control features of the SOFTWARE PRODUCT(S) except as an intended part of the SOFTWARE PRODUCT(S)'s programming features. This license is not transferable to any other company, entity, or individual. You may not publish any registration information (serial numbers, registration keys, etc.) or pass it to any other company, entity, or individual.

Permission to distribute the SOFTWARE is not transferable, assignable, saleable, or franchisable. Each entity wishing to distribute the package must independently satisfy the terms of the distribution license.

2. COPYRIGHT

Except for the licenses granted by this agreement, all right, title, and interest in and to the SOFTWARE PRODUCT(S) (including, but not limited to, all copyrights in any executable programs, modules, controls, libraries, electronic documentation, text and example programs), any printed materials and copies of the SOFTWARE PRODUCT(S) are owned by AGENSOFT. The SOFTWARE PRODUCT(S) is protected by copyright laws and international treaty provisions. Therefore you must treat the SOFTWARE PRODUCT(S) like any other copyrighted material except that you may (a) make one copy of the Software solely for backup or archival purposes, or (b) transfer the Software to a single hard disk, provided you keep the original solely for backup or archival purposes. You may not copy any printed materials that may accompany the SOFTWARE PRODUCT(S).

3. REDISTRIBUTION

a) In addition to the rights granted in Section 1, AGENSOFT grants you the right to use and modify the source code version of those portions of the Software designated as "sample code" for the sole purposes of designing,

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

developing, and testing your software product(s), and to reproduce and distribute the sample code, along with any modifications thereof, only in object code form, provided that you comply with Section 3.c.

b) In addition to the rights granted in Section 1, AGENSOFT grants you a non-exclusive, royalty-free right to reproduce and distribute the object code version of any portion of the SOFTWARE PRODUCT(S), along with any modifications thereof, in accordance with the above stated conditions.

c) If you redistribute the sample code or redistributable components, you agree to: (i) distribute the redistributables in object code only, in conjunction with and as a part of a software application product developed by you which adds significant and primary functionality to the Software; (ii) not use AGENSOFT's name, logo, or trademarks to market your software application product; (iii) include a valid copyright notice on your software product ; (iv) indemnify, hold harmless, and defend AGENSOFT from and against any claims or lawsuits, including attorney's fees, that arise or result from the use or distribution of your software application product; (v) not permit further distribution of the redistributables by your end user.

AGENSOFT PRODUCT(s) may include certain files ("Redistributable(s)") intended for distribution by you to the users of software applications which you create. Redistributables include, for example, those files identified in printed or on-line documentation as redistributable files, or those files preselected for deployment by an install utility provided with the SOFTWARE PRODUCT(S) (if any). In all circumstances, the Redistributables for the SOFTWARE PRODUCT(S) are only those files specifically designated as such by AGENSOFT.

Subject to all of the terms and conditions in this EULA, you may reproduce and distribute copies of the Redistributables, provided that such copies are made from the original copy of the Redistributables included with the SOFTWARE PRODUCT(S) or modified versions of the Redistributables which are provided to you by AGENSOFT or those which you create. Copies of Redistributables may only be distributed with and for the sole purpose of executing application programs permitted under this EULA that you have created using the SOFTWARE PRODUCT(S).

AT NO TIME MAY CUSTOMER CREATE ANY TOOL, REDISTRIBUTABLE, OR SOFTWARE PRODUCT(S) THAT DIRECTLY OR INDIRECTLY COMPETES WITH AGENSOFT SOFTWARE PRODUCT(S) WHICH UTILIZES ALL OR ANY PORTION OF THE SOFTWARE PRODUCT(S) contained within this installation.

Distribution by the CUSTOMER of any design-time tools (EXE's OCX's or DLL's), executables, and source code distributed to Customer by AGENSOFT as part of this SOFTWARE PRODUCT(S) and not explicitly identified as a redistributable file is strictly prohibited. The Customer shall not develop software applications that provide an application programming interface to the SOFTWARE PRODUCT(S) or the SOFTWARE PRODUCT(S) as modified.

The Customer may NOT distribute the SOFTWARE PRODUCT(S), in any format, to other users for development or application compilation purposes. Specifically, if Customer creates a control using the SOFTWARE PRODUCT(S) as a constituent control, Customer may NOT distribute the control created with the SOFTWARE PRODUCT(S) (in any format) to users to be used at design time and or for ANY development purposes.

Customer MAY NOT REDISTRIBUTE any SOFTWARE PRODUCT(s) files if using an evaluation, trial, Not for Resale, or demo version of the SOFTWARE PRODUCT(s).

4. TRANSFER

You may NOT permanently or temporarily transfer ANY of your rights under this agreement to any individual or entity without prior written approval from AGENSOFT. Regardless of any modifications which you make and regardless of how you might compile, link, and/or package your programs, under no circumstances may the libraries, Redistributables, and/or other files of the SOFTWARE PRODUCT(S) (including any portions thereof) be used for developing programs by anyone other than you. Only you as the licensed Customer have the right to use the libraries, redistributables, or other files of the SOFTWARE PRODUCT(S) (or any portions thereof) for developing programs created with the SOFTWARE PRODUCT(S). In particular, you may not share copies of the Redistributables with other co-developers. You may not reproduce or distribute any AGENSOFT documentation without AGENSOFT's explicit permission.

If you are an entity (Company), you must acquire and assign a license to each Customer within your organization using and or developing with the SOFTWARE PRODUCT(S). With written notification to AGENSOFT, Company may transfer the license obtained for a Customer to another Customer employed or otherwise engaged by Company if the initial Customer is no longer employed or engaged by Company or is reassigned to another function within Company and no longer develops software applications using the SOFTWARE PRODUCT(S). In addition, with written notification to AGENSOFT, Company may transfer its license of the SOFTWARE PRODUCT(S) to a successor Company.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

5. TRADE SECRETS AND CONFIDENTIALITY

a) The Software contains information or material which is proprietary to AGENSOFT ("Confidential Information"), which is not generally known other than by AGENSOFT, and which you may obtain knowledge of through, or as a result of the relationship established hereunder with AGENSOFT. Without limiting the generality of the foregoing, Confidential Information includes, but is not limited to, the following types of information, and other information of a similar nature (whether or not reduced to writing or still in development): designs, concepts, ideas, inventions, specifications, techniques, discoveries, models, data, source code, object code, documentation, diagrams, flow charts, research, development, methodology, processes, procedures, know-how, new product or new technology information, strategies and development plans (including prospective trade names or trademarks).

b) Such Confidential Information has been developed and obtained by AGENSOFT by the investment of significant time, effort and expense, and provides AGENSOFT with a significant competitive advantage in its business.

c) You agree that you shall not make use of the Confidential Information for your own benefit or for the benefit of any person or entity other than AGENSOFT, except for the expressed purposes described in the paragraph hereof entitled "REDISTRIBUTION", in accordance with the provisions of this Agreement, and not for any other purpose.

d) You agree to hold in confidence, and not to disclose or reveal to any person or entity, the Software, other related documentation, your product registration key file or serial number (if any) or any other Confidential Information concerning the Software other than to such persons as AGENSOFT shall have specifically agreed in writing to utilize the Software for the furtherance of the expressed purposes described in the paragraph hereof entitled "REDISTRIBUTION", in accordance with the provisions of this Agreement, and not for any other purpose.

e) You acknowledge the purpose of this section entitled "TRADE SECRETS AND CONFIDENTIALITY" is to protect AGENSOFT’s ability to limit the use of the data and the Software generally to licensees, and to prevent use of Confidential Information concerning the Software by other developers or vendors of software.

6. OTHER RESTRICTIONS

You may not rent, lease or transfer the Software. You may not reverse engineer, decompile or disassemble the Software, except to the extent applicable law expressly prohibits the foregoing restriction. Without prejudice to any other rights, AGENSOFT may terminate this License Agreement if you fail to comply with the terms and conditions of the agreement. In such event, you must destroy all copies of the SOFTWARE PRODUCT(S).

7. LIMITED WARRANTY

If within 30 days of your purchase of this SOFTWARE PRODUCT(S), you become dissatisfied with the Software for any reason, you may return the software to AGENSOFT (or your dealer, if you did not purchase it directly from AGENSOFT) for a refund of your purchase price. To return the Software, you must contact AGENSOFT and obtain a return material authorization (RMA) number. AGENSOFT will not accept returns of opened or installed software without an RMA number. Returns are subject to the deduction from your purchase price of a 20% restocking fee and all shipping costs. You agree that this limited warranty does not apply to a Software source code license as specified in section 5.e.

8. TECHNICAL SUPPORT

Technical support is usually provided via email only. At any time, if the unmodified SOFTWARE PRODUCT(S) fails to satisfy the performance described in the documentation, AGENSOFT will try to issue a workaround to help the Customer solve the technical problems. However, it is understood that AGENSOFT may not always be able to solve such problems. A service fee for such workarounds may be charged by AGENSOFT. In such case, AGENSOFT will notify the Customer with a quote in prior to issuing the workarounds.

9. NO OTHER WARRANTIES

AGENSOFT DISCLAIMS ALL OTHER WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO IMPLIED WARRANTY OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE, WITH RESPECT TO THE SOFTWARE PRODUCT(S), THE ACCOMPANYING WRITTEN MATERIALS, AND ANY ACCOMPANYING HARDWARE.

10. LIMITATION OF LIABILITY

IN NO EVENT SHALL AGENSOFT OR ITS SUPPLIERS BE LIABLE FOR ANY DAMAGES WHATSOEVER (INCLUDING, WITH LIMITATION, INCIDENTAL, CONSEQUENTIAL, SPECIAL, OR EXEMPLARY DAMAGES OR LOST PROFITS, BUSINESS INTERRUPTION, OR OTHER PECUNIARY LOSS) ARISING OUT OF THE USE OR INABILITY OF THIS SOFTWARE PRODUCT(S), EVEN IF AGENSOFT HAS BEEN ADVISED OF SUCH DAMAGES.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

APART FROM THE FOREGOING LIMITED WARRANTY, THE SOFTWARE PRODUCT(S) ARE PROVIDED "AS-IS", WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED. THE ENTIRE RISK AS TO THE PERFORMANCE OF SOFTWARE PRODUCT(S) IS WITH THE PURCHASER. AGENSOFT DOES NOT WARRANT THAT THE OPERATION OF THE PROGRAMS WILL BE UNINTERRUPTED OR ERROR-FREE. AGENSOFT ASSUMES NO RESPONSIBILITY OR LIABILITY OF ANY KIND FOR ERRORS IN THE SOFTWARE PRODUCT(S), OF/FOR THE CONSEQUENCES OF ANY SUCH ERRORS.

11. GOVERNMENT-RESTRICTED RIGHTS

Use, duplication, or disclosure by the U.S. Government of the computer software and documentation in this package shall be subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.277-7013 (Oct 1988) and FAR 52.227-19 (Jun 1987). The Contractor is AGENSOFT.

12. GOVERNING LAW

This agreement shall be governed by the laws of Moscow, Russian Federation, excluding the application of its conflicts of law rules and shall inure to the benefit of AGENSOFT and any successors, administrators, heirs and assigns. Any action or proceeding brought by either party against the other arising out of or related to this agreement shall be brought only in a STATE or FEDERAL COURT of competent jurisdiction located in Russian Federation, Moscow. The parties hereby consent to in personal jurisdiction of said courts. The United Nations Convention on Contracts for the International Sale of Goods is specifically disclaimed.

In addition to any other relief granted the prevailing party shall be entitled to recover its attorney’s fees and costs. THE PARTIES EXPRESSLY WAIVE THE RIGHT TO A TRIAL BY JURY. The parties acknowledge that any breach of this agreement may result in irreparable harm to AGENSOFT Inc., thereby entitling AGENSOFT to injunctive relief for any such breach in addition to any other rights or remedies that AGENSOFT may have. This Agreement is the entire agreement between you and AGENSOFT and supersedes any other communications or advertising with respect to the SOFTWARE PRODUCT(S). If any provision of this License is held invalid, the remainder of this License shall continue in full force and effect.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

6.2. How to register

AgenSoft's PatchFactory is "try-before-you-buy" software. You can evaluate it 30 days for free. When this period expires you must either purchase our software or stop using it and remove it from your computer.

➢ **Please visit our public web-site for the latest Ordering information: http://www.agensoft.com**

As soon as we are notified that your order has been processed, a registration key will be sent to you via email within 24 hours or one business day (usually within 12 hours and depends on the country you reside in) to unlock your copy of PatchFactory. Ordering gives you the right to use PatchFactory after the 30-day trial period, receive technical support and use features available only for registered users.

We offer a 30-day money back guarantee on all our software. If you are not completely satisfied with your purchase, we will refund its total price excepting shipping fees within 30 days of receipt.

➢ **To buy PatchFactory you can use one of our secure payment processing services:** *RegNow* or *ShareIt!* (services of Digital River), which are industry leaders in E-Commerce Payment Processing Services. All data exchanged during the payment process is SSL-secured and protected with a high level of encryption.

Our registration services currently accept the following bank cards: **Visa**, **Mastercard/Eurocard**, **American Express** , **Discover/Novus**, **JCB**, **Diners Club**. You can also order by **Cheque**, **Cash**, **Fax**, **Postal mail**, **Fax&Phone**, **Bank/Wire Transfer**, **Purchase Order** or **Paypal** - check links on the online secure order form. Don't forget to supply exact personal information and a valid e-mail address to which we can send your registration and update information.

The simplest and cheapest way to purchase PatchFactory software is to order it online.
Customer information is considered confidential and will not be shared or distributed to any third party.

To get more info on PatchFactory discounts and special offers visit PatchFactory order page at http://www.agensoft.com/order.html

To allow for all types of developers to use our software, we offer three types of licenses:	
• **Discounted Personal License**	offered as a service to the industry, primarily for single person companies with little revenue (such as shareware authors). The software is licensed to the name of the individual purchasing the license.
• **Standard Commercial License**	intended for small companies. The software is licensed to the name of the company purchasing the license. Maximum number of employees using the software is limited to 5. It also entitles an organization to receive high-priority support (with guaranteed answer within 2 business days) via email. If you would like to obtain more extended services or more employees to use the software - consider buying the Corporate license
• **Premium Corporate License**	intended for large companies and corporates with many employees. The software is licensed to the name of the company purchasing the license. And besides this type of license entitles an organization to receive one copy of the distribution software and to duplicate the software for any number of people or workstations within the corporation. It also entitles an organization to receive high-priority support (with guaranteed answer within 1 business day) via email and free major version upgrades during lifetime of the product.

NOTE: All licenses include the royalty-free distribution of created patch package both for you and your customers for an unlimited number of applications, free minor version updates with special 50% discount for major upgrades during lifetime of the product and high-priority technical support via email and public support forums.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

NOTE: The price for Personal, Commercial and Corporate licenses as well as other service terms are subject to change without any notice. To get the latest information about our pricing and discounting policy and other services, you can *visit our public web-site at* www.agensoft.com.

If you do not receive your registration code within a reasonable amount of time (usually two business days for credit card payments or two weeks for other payments), please notify us by email at sales@agensoft.com. We request your apologize for any inconvenience caused by those delays if any.

If you have any problems after you placed an order, visit our Ordering Problem and FAQ pages or send an e-mail to our Sales Support Team at sales@agensoft.com. Please include your name, e-mail address, and detailed description of your problem. We make every effort to reply to all e-mail inquiries within two to four hours with a maximum of 48 hours or two business days.

Competitive Upgrade - up to 50% OFF for migration to PatchFactory. If you have previously purchased an updating solution and have experienced any options, performance, or reliability issues, AgenSoft offers you the chance to upgrade from your current vendor's software to PatchFactory. Please forward a copy of the order confirming your purchase of that updating software at sales@agensoft.com and we'll gladly e-mail you the special instructions to obtain your upgrade to PatchFactory.

Purchasing a PatchFactory license entitles you to the following:
Royalty-free distribution of the PatchFactory Client for an unlimited number of applications.
Free minor updates with special 50% discount for major upgrades during lifetime of the product.
Removal of the 30-day trial and functional (if any) limitations.
The registration reminder (nag-screen) will never popup again.
Friendly and quick-response Technical support (currently only via email) at support@agensoft.com
Easy and efficient software updating tool to get the full control under your software versions

Information about our registration services:

- **RegNow** (Issaquah, Washington, USA). **Billing Currency:** USD, EUR, GBP, AUD and others. **Payment options:** Credit/Debit Cards, Cheque, Cash, Postal mail, Fax&Phone Ordering, Bank/Wire Transfer, PayPal. **Language:** Multilingual.
- **ShareIt!** (Koln, Germany). **Billing Currency:** EUR, USD, GBP, AUD and others. **Payment options:** Credit/Debit Cards, Cheque, Cash, Postal mail, Bank/Wire Transfer, Fax&Phone Ordering, Purchase Order. **Language:** MultiLingual.
- **RegSoft** (Atlanta GA, USA). **Billing Currency:** USD,EUR, GBP, AUD and others. **Payment options:** Credit/Debit Cards, Cheque, Cash, Fax, Postal mail, Fax&Phone Ordering, Bank/Wire Transfer, PayPal. **Language:** English.

"Free minor version updates" means that by purchasing version 3.0, you have also purchased 3.1, 3.2 and all other 3.x versions, but not version 4.0 and later. However, you will get free major version update if it has passed less than 3 months since the date of your purchase.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

6.3. Update and Support

- **Update**

  As a registered user of PatchFactory v3 or less you will be able to receive all minor updates of PatchFactory free of charge.
  It means that by purchasing version 3.1, you have also purchased 3.2, 3.3 and all other 3.x versions, but not version 4.0 and later. However, at the present time there we are not planning to release a new major version update and it is entirely possible that only minor versions will ever be published.

  Just download the latest release of PatchFactory from our company's URL and install it without uninstalling the previous version. This will preserve your registration. You can also download the latest update module.

  Update information: www.agensoft.com/updates
  Download information: www.agensoft.com/download.html

  - **Registration renewal:**

  Registration renewal: to renew your registration of PatchFactory you can use the appropriate discounted upgrade link at http://www.agensoft.com/order.html
  As soon as we check your registration info and your payment is processed, a new registration key will be sent to you via email to unlock your copy of PatchFactory software.
  Major version upgrade is offered to our current customers (Personal and Commercial licenses) at a discounted rate (50% discount). Corporate customers receive any version updates & upgrades free of charge during lifetime of the product.

  If the registration code is no longer present on your system just re-enter it like the first time and restart the program.

  If for any reason your registration code becomes invalid or lost send an e-mail to sales@agensoft.com with details of your previous registration. You will receive a new code via e-mail at no charge as soon as we validate your old registration data.

  You can subscribe to our newsletter at our site www.agensoft.com to be informed of the latest information about product updates, site news and corporate events.

- **Support**

  If you have a question regarding the operation of PatchFactory, please take a moment to review the most common Frequently Asked Questions (FAQs). You may find there the answer you are looking for! If there is no answer do not hesitate to contact us. We'll help you and respond to your message as soon as possible (usually within 24 to 48 hours).

  Support information: http://www.agensoft.com/support

  You can use our online email-form (preferably) to contact us. Just fill all necessary fields and click on "Send Email" button. Do not forget to select the correct Question Category for your message and write the detailed description of your problem - it can significantly shorten the respond time.

  All comments and suggestions concerning improvements to PatchFactory are appreciated!
  If you can help us with implementing of other languages support for our web-site, PAD-file descriptions of PatchFactory, PatchFactory program interface, or update installation dialogs, please email us, you can get a free registration!
  But, please, don't send any translations prior to contacting us (***required***), because this language support can be already under development.

  When asking for technical support please inform us about the following:
  - PatchFactory version installed (from the "About" dialog including build number)
  - Full OS version installed (including Service Pack installed and Localization)
  - Detailed description of your problem (with a screenshot if possible)
  - Your registration information (if you are a registered user)

NOTE: before requesting technical support, please ensure that you are using the latest version of PatchFactory available from http://www.agensoft.com/download.html

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com

7. Contact information

Contacting AgenSoft...

Please refer to the E-mail addresses below for further information.

info@agensoft.com
General information and feedback. Any suggestions and comments will be welcomed!

support@agensoft.com
Customer technical support service or bug report.

sales@agensoft.com
Purchase or registration code affairs. We will be happy to help you.
Software distribution, promotion in software compilations, or business cooperation. We are open to various levels of cooperation.

webmaster@agensoft.com
Exchanging links or placing advertisement.

You can also use our online email-form (preferably) to contact us.

Visit our public web-site www.agensoft.com for further information.

© 2002-2008 by AgenSoft. All rights reserved.
www.agensoft.com