

Git 1.2

On this page:

- [Extension description](#)
 - [In what situation should you install this extension?](#)
 - [Technical information](#)
- [What's new?](#)
 - [Git 1.2](#)
- [Supported GIT hosting sites](#)
- [CAST AIP compatibility](#)
- [Download and installation instructions](#)
- [Extension interface](#)

Target audience:

CAST Administrators



Summary: Information and Release Notes about the **Git File Extractor 1.2.x** for the CAST Delivery Manager Tool. This extension is a **file extractor**.

Extension description

This extension provides the means to extract source code from a **Git** repository. In other words, file based source code that does not reside in a simple local or network folder. This extension is similar in function to the extractors for **SVN** and **TFS** that are provided "out of the box" in the CAST Delivery Manager Tool.

In what situation should you install this extension?

This extension should be used when you want to extract file based source code that is stored in a **Git** repository.

Technical information

- The extension **does not contain a source code "discoverer"** (to determine the type of source code project) and therefore relies on other discoverers that are already installed in the CAST Delivery Manager Tool to do so.
- This extension **supports both local and remote Git repositories**:
 - The URL format for a **local folder** should be for example **C:/[path]/.git**.
 - For **remote repositories**, **http/https** and **SSH** protocols are supported, however the remote repositories are cloned to the hard drive on which the CAST Delivery Manager Tool is being run from. The %TEMP% folder is used to store the cloned repository. When the packaging action has been completed in the CAST Delivery Manager Tool, the cloned repository will be removed from the local disk.



Note that with regard to **SSH protocol**, you must ensure that you have an **SSH key** setup already in your GitHub account for the machine on which you will run the extension via the CAST Delivery Manager Tool. If the key is not set up and you attempt to use the SSH protocol, then an error message will be displayed.

What's new?

Git 1.2

- Bug Fixing:
 - EXTGIT-21: Cleanup of cloned repository from %temp% folder
 - EXTGIT-28: Invalid GIT URL generates an error that doesn't indicate what went wrong
 - EXTGIT-29: Checking out the revision with full hash id
 - EXTGIT-32: Extraction from private repository over HTTPS doesn't work
- New features:
 - EXTGIT-23: Make folder parameter not mandatory
 - Explanation: In previous releases of this extension, if no sub-folder was entered in the Repository URL field in the CAST Delivery Manager Tool the package action would fail. This issue has been resolved and now, if no sub-folder is entered in the Repository URL field, the entire repository will be extracted instead.
 - EXTGIT-33: Make it possible to select the branch to be cloned
 - Explanation: In previous releases of this extension, it was not possible to select a branch for extraction - instead only the default main branch was extracted. This feature adds a "Branch" field to the Git extractor GUI in the CAST Delivery Manager Tool enabling you to select the branch you require.

- EXTGIT-31: Cloning with SSH
 - Explanation: In previous releases of the extension, support for the SSH protocol did not function correctly. This has now been fixed.
- EXTGIT-35: Support self-signed certificates
 - Explanation: In previous releases of the extension, if the HTTPS protocol was used in the extraction and the remote certificate was deemed "invalid" (self-signed, unknown root CA, expired) then the extraction would fail. This feature enables you to deactivate the check for invalid certificates if the remote repository has one in place and the extraction will succeed.

Supported GIT hosting sites

- GitLab
- Github
- Bitbucket
- RocketGit

CAST AIP compatibility

This extension is compatible with:

CAST AIP release	Supported
8.2.x	✓
8.1.x	✓
8.0.x	✓
7.3.x	✓

Download and installation instructions

Please see:

- <http://doc.castsoftware.com/display/EXTEND/Download+an+extension>
- <http://doc.castsoftware.com/display/EXTEND/Install+an+extension>

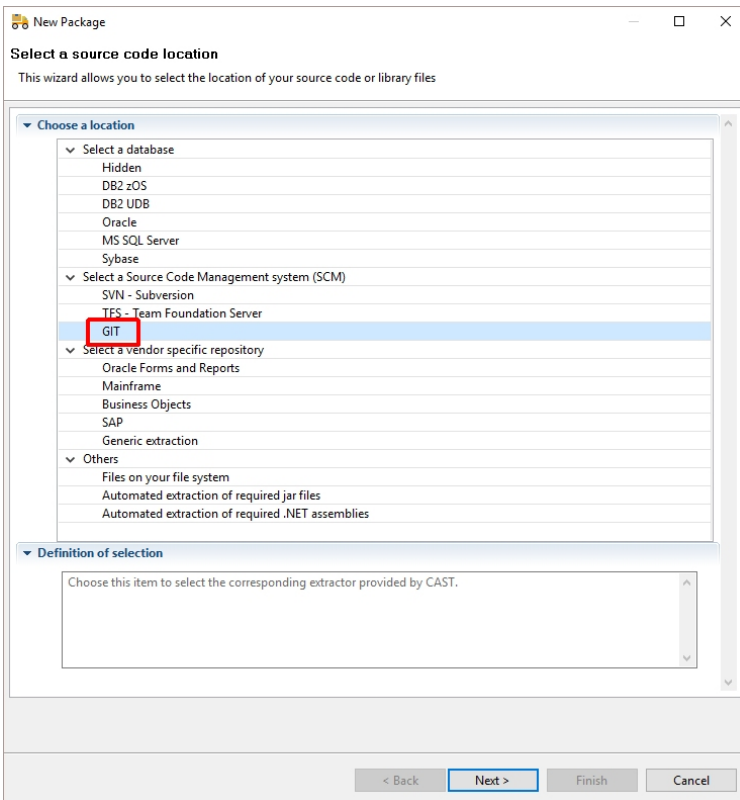


- This extension contains a **File extractor** and you should take note of the specific instructions in the installation guide that explains how to package your source code with the CAST Delivery Manager Tool when you have an **existing Version**.
- The latest [release status](#) of this extension can be seen when downloading it from the CAST Extend server.

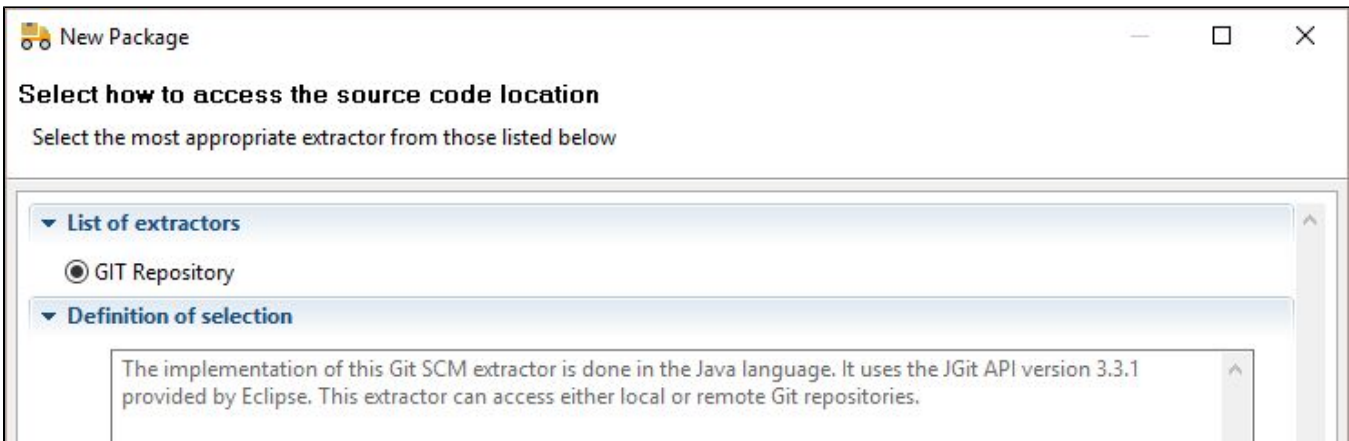
Extension interface

The following screen shots show the differences in the product when the extension is installed:

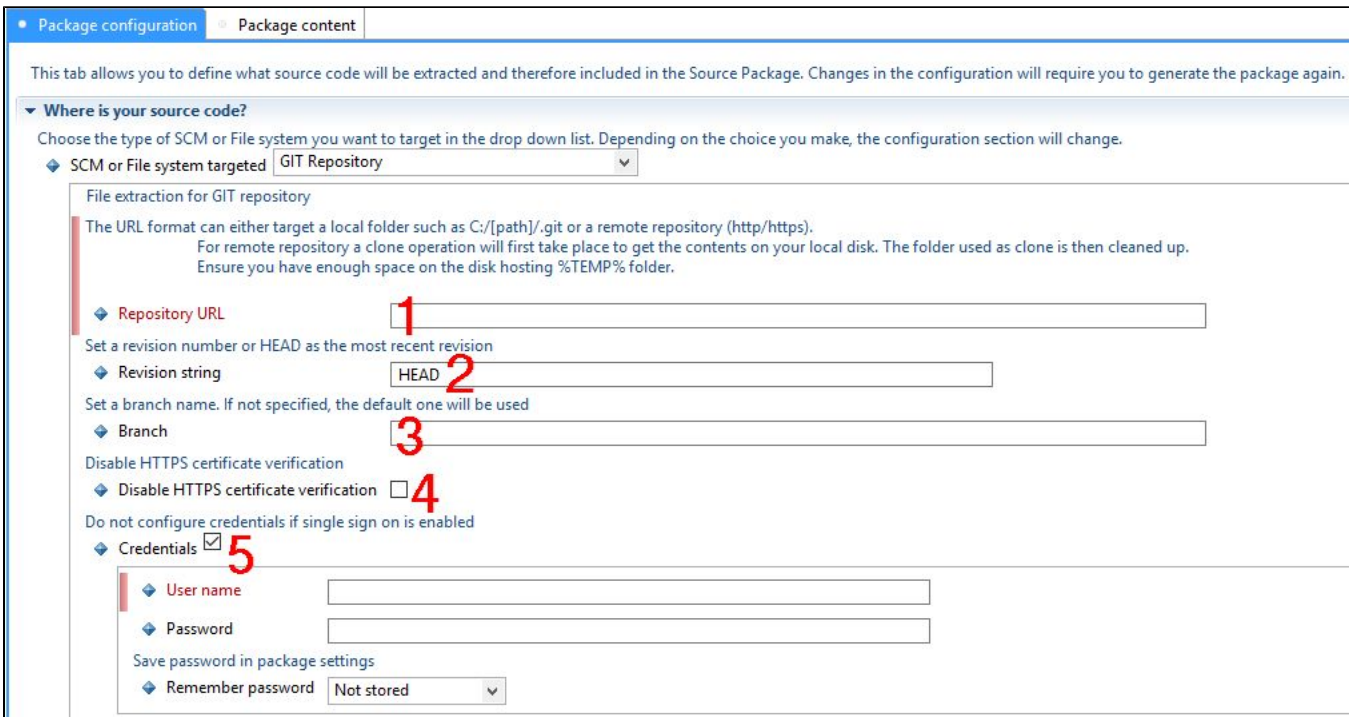
- In the CAST Delivery Manager Tool a new entry called "GIT" will be added to the **Add Package wizard**:



- When selected the above item provides the following screen with the **GIT Repository** option already selected:



- The **Package Configuration** tab then offers the following interface to access your GIT repository:



Option	Explanation								
1	Enter the direct URL for your Git repository in the form: <table border="1" style="margin-left: 20px;"> <thead> <tr> <th>Repository</th> <th>URL format</th> </tr> </thead> <tbody> <tr> <td>Local</td> <td>C:/[path]/.git</td> </tr> <tr> <td>http/https</td> <td>https://github.com/<account>/TEST.git</td> </tr> <tr> <td>SSH</td> <td>git@github.com:<account>/TEST.git</td> </tr> </tbody> </table>	Repository	URL format	Local	C:/[path]/.git	http/https	https://github.com/<account>/TEST.git	SSH	git@github.com:<account>/TEST.git
Repository	URL format								
Local	C:/[path]/.git								
http/https	https://github.com/<account>/TEST.git								
SSH	git@github.com:<account>/TEST.git								
2	Enter the commit hash of the revision you want to extract. If omitted, the HEAD revision of the selected branch is chosen. Note that: <ul style="list-style-type: none"> • if the Branch parameter is used (see option 3 below), this parameter is ignored • selecting the revision based on a revision tag is currently not supported 								
3	If the code you require is located in a specific Git branch , enter the branch name here to ensure that the code from that branch is extracted. If nothing is entered, the default main branch will be used.								

4	When this option is unticked (default position), if the certificate for the GIT repository is invalid (self-signed, unknown root CA, expired), then extraction fails. If this option is ticked, extraction succeeds in cases described previously - in other words, invalid HTTPS certificates are ignored. Note that ticking this option has no effect on an extraction which already succeeds (HTTP, valid HTTPS).
5	<p>Tick the Credentials option if the Git repository requires authenticated access.</p> <p>Remember Password</p> <p>This option enables you to force the CAST Delivery Manager Tool to save the access credentials you have entered above. Choosing an option or not has no impact on the extraction (i.e. the CAST Delivery Manager Tool can still access the required resources). However, if you are creating subsequent Versions using the same repositories, you can choose to store the password in which case you will not need to re-enter it.</p> <p>There are two save options:</p> <ul style="list-style-type: none">• Local > The credentials are saved in the user's local workspace on the current machine. Choose this option if you do not want the password to be available to other Delivery Managers.• Server > The credentials are saved locally (as above) and are also synchronized back to the CAST AIC Portal (i.e. the Source Code Delivery Folder). Choose this option if you want the password to be available to other Delivery Managers.