

# PowerBuilder - Prepare and deliver the source code

- [Preparation - source code discovery](#)
- [Using CAST AIP Console](#)
  - [Source code delivery](#)
  - [PowerBuilder version](#)
- [Using legacy CAST Delivery Manager Tool](#)
  - [How do I add a source code package to my delivery](#)
  - [What you should package?](#)
  - [How do I fine-tune my Version ?](#)
  - [How do I deliver the Version for analysis?](#)
  - [Delivery acceptance](#)



**Summary:** This section describes how to prepare and deliver the source code of your PowerBuilder application.

## Preparation - source code discovery

Discovery is a process that is actioned during the delivery process CAST will attempt to automatically identify "projects" within your application using a set of predefined rules. Discoverers are currently **embedded in CAST AIP Core**:

- [PowerBuilder Discoverer](#)

You should read the relevant documentation for each discoverer (provided in the link above) to understand how the source code will be handled.

## Using CAST AIP Console



- Supported in AIP Console v. **1.17**.
- See [Application onboarding](#) for more information.

## Source code delivery

AIP Console expects either a **ZIP/archive file** or **source code located in a folder** configured in AIP Console. You should include in the ZIP/source code folder all PowerBuilder source code:

- .PBT
- .PBL
- etc.

CAST highly recommends placing the files in a folder dedicated to PowerBuilder. If you are using a ZIP/archive file, zip the folders in the "temp" folder - but do not zip the "temp" folder itself, nor create any intermediary folders:

```
D:\temp
  |-----PowerBuilder
  |-----OtherTechno1
  |-----OtherTechno2
```

## PowerBuilder version

In order for a PowerBuilder analysis to complete successfully

- the **PowerBuilder IDE** must be installed on the machine on which the analysis is running - i.e. the Node
- **or** the path to the **pborxx.dll** file and the **PowerBuilder version** must be populated in the `%PROGRAMDATA%\CAST\AipConsole\AipNode\aip-node-app.properties` file if the **PowerBuilder IDE** is not installed on the AIP Node.

In essence, AIP Console needs to know the **version** of PowerBuilder you are delivering/analyzing and where the **pborxx.dll** file is located:

- AIP Console will **first** attempt to discover the location of the pborcXXX.dll file/PowerBuilder version using the **Windows registry** (on the AIP Node) during the source code delivery process. If AIP Console finds a registry key on the AIP Node, it will populate the relevant fields in the AIP Console GUI (see [PowerBuilder - Analysis configuration](#)).
- If no registry key is found (i.e. the PowerBuilder IDE is not installed on the AIP Node), then AIP Console will look in the **following file on the target AIP Node**. For example - these locations can be filled in BEFORE delivering the source code (you must restart the AIP Node so that the changes are taken into account):

```
2.x
%PROGRAMDATA%\CAST\AIP-Node\application.yml

1.x
%PROGRAMDATA%\CAST\AipConsole\AipNode\aip-node-app.properties
```

The following sections relevant and should be uncommented/filled in as appropriate:

```
2.x
# =====
# PowerBuilder configuration
# -----
# Provide the full paths of the pborcXXX.dll that are available on this host separated by ';'
#powerbuilder:
# pborcdll:
#   location: C:\\Users\\ABD\\Documents\\PB9\\Shared_PB9\\pborc90.dll;C:
\\Users\\ABD\\Documents\\PB10\\Shared_PB10\\pborc100.dll
# Maximum number of parallel job executions
# If maximum number of parallel job executions is reached, the job will wait in queue

1.x
# =====
# PowerBuilder configuration
# -----
# Default version of Power Builder to analyze between (Version8, Version9, VersionX, VersionX5, VersionXI,
VersionXI5, VersionXII, VersionXII5, VersionXII6, Version2017, Version2018)
powerbuilder.compile.version=Version8
# Path of the pborcXXX.dll file associated to the default PowerBuilder version
powerbuilder.pborcdll.location=
```

If neither method reveals the location of the pborcXXX.dll file/PowerBuilder version, then **the analysis will fail**. Above all, the options defined in the **AIP Console GUI have priority** over the **Windows registry** and the **aip-node-app.properties** file (in that order).


## Using legacy CAST Delivery Manager Tool

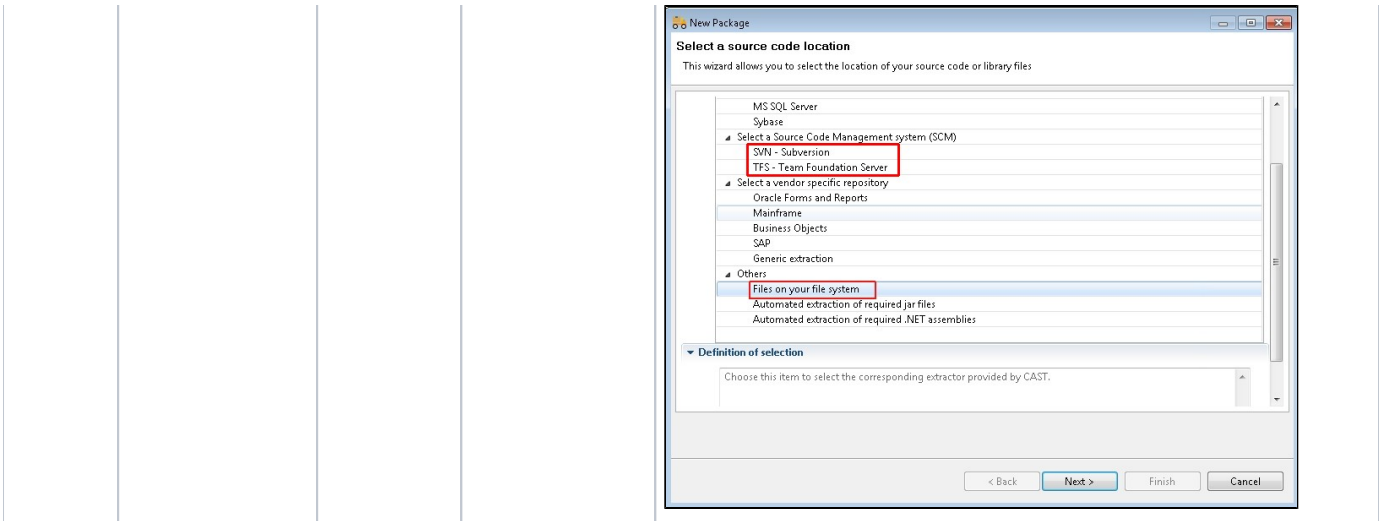
### How do I add a source code package to my delivery

See [How do I add a source code package to my delivery](#).

### What you should package?

When creating packages to discover and extract your PowerBuilder application you should create them as listed below:

Package	Package name /type	Mandatory?	Location/path	Notes
1	Source code		Source code root folder	Use the " <b>Files on your file system</b> " / <b>SVN</b> / <b>TFS</b> options in the CAST Delivery Manager Tool:  <i>Click to enlarge</i>



## How do I fine-tune my Version ?

See [How do I fine-tune my Version](#) for more information.

## How do I deliver the Version for analysis?

See [How do I deliver the Version for analysis](#) for more information.

## Delivery acceptance

See [Validate and Accept the Delivery](#) for more information.