**PIEP – Custom Symbol Manager**

**User Guide**

## VERSIONING AND CONTROL VALIDATION

VERSIONING HISTORY

|  |  |  |  |
| --- | --- | --- | --- |
| Version | AUTHOR | REVISION | Date |
| 1.0 | RCAS (CGI) | Creation | 29/09/2022 |

VALIDATION

|  |  |  |
| --- | --- | --- |
| Version | Name | Date |
| 1.0 |  |  |
|  |  |  |

## DOCUMENT CONTENT

1. VERSIONING AND CONTROL VALIDATION 2

2. DOCUMENT CONTENT 3

3. SCRIPT DESCRIPTION 4

3.1 Objective 4

3.2 Prerequisite 4

4. USER GUIDE 5

4.1 Prepare the input files 5

4.2 Run the PI Custom Symbol Manager 6

## SCRIPT DESCRIPTION

### Objective

**The purpose of the “PI Custom Symbol Manager” is to give a status of the custom symbols in remote PI Vision servers.**

The user provide the servers list and the referencial files for each symbols.

The application determine if the symbol is referenced, valid (all files required are installed), and the versioning.

### Prerequisite

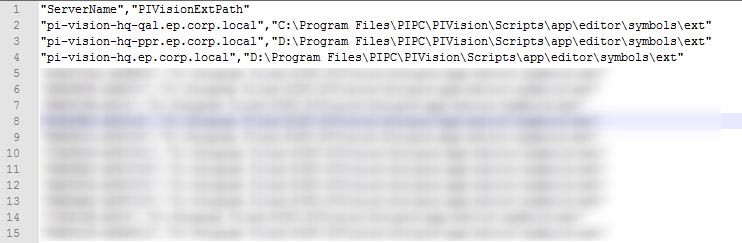
* Microsoft Powershell (5.0 or later).
* Package PI SMT 2018 or later.

## USER GUIDE

Copy the folder ExtractionPS in your local machine/server.

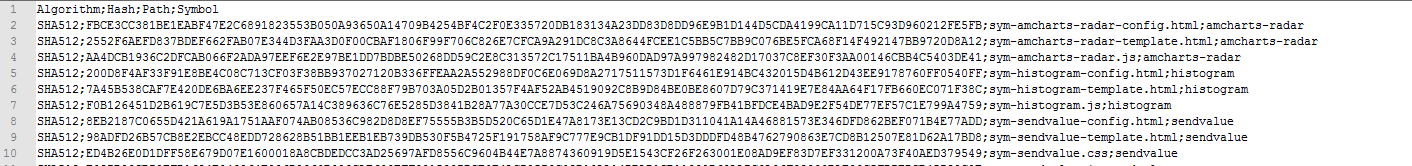
### Prepare the input files

Edit the file “Input\Server\_Path\_List.csv”.



Verify that every PI Vision server names and custom symbols root path are listed. Add them if necessary.

Edit the file “Input\Referential\_PICustomSymbolFiles.csv”.



Verify that every files are listed with their associated symbol.

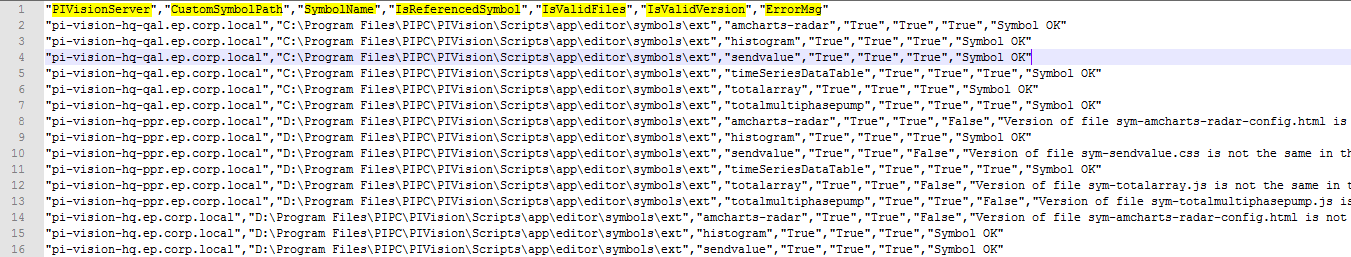
**NB** : you can run the script “PIVision-Export-Symbols-Hash.ps1” to generate a new file based on a PI Vision server without the Symbol column. You must specified manually the files for each symbols after that.

### 4.2 Run the PI Custom Symbol Manager

Run the script **PICustomSymbolsManager.ps1**

It generate a log file in .\logs folder and an analysis file in .\output folder.

Open ".\output\YYYY-MM-DD\_PICustomSymbolsAnalysis.csv" file. Example :



It countain a line per symbol per server with the following informations :

* "**PIVisionServer**" – The server name where the current symbol is installed
* "**CustomSymbolPath**" – The root path of the symbol js file
* "**SymbolName**" – The name of the current symbol
* "**IsReferencedSymbol**" – Indicate “true” if the symbol is present in the referential input file, “false” otherwise.
* "**IsValidFiles**" – Indicate “true” if all the files required to make the symbol works are present, “false” otherwise.
* "**IsValidVersion**" – Indicate “true” if the hash code of the files are the same than the referential one, “false” otherwise.
* "**ErrorMsg**" – Indicate the correction to make to this symbol is necessary, or indicate “Symbol OK” if everything is conform.