Configuring code pre-processing for your Universal Analyzer language

- Introduction
- Creating a pre-processor
  - launch.bat
- How it works

If the language you are targeting in the Universal Analyzer requires some pre-processing prior to analysis, CAST has provided a method for automating this. This section describes the process for configuring language pre-processing and how to integrate it into your language pack.
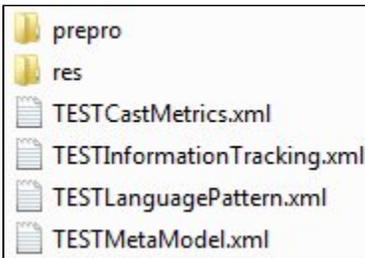
Introduction

Some languages supported as standard by CAST with their own dedicated CAST Analyzer (such as Cobol or C++) require that the source code is pre-processed prior to analysis. This pre-processing phase is built in to the CAST Analyzer and is executed seamlessly before the analysis takes place without the end-user having to do anything. In previous versions of CAST, although the Universal Analyzer was able to analyze any language not supported by CAST (via a dedicated analyzer) provided that a language package was correctly defined for the target language, the pre-processing phase (if the language required it to meet the configuration provided in the language pack) had to be carried out manually before the source code files were analyzed. This added an extra step in the analysis/configuration process.

This situation has changed - it is now possible to automate the pre-processing phase using a batch file (containing the pre-processing instructions), which is executed prior to the analysis of the source code files by the Universal Analyzer.

Creating a pre-processor

A pre-processor consists of a folder ("**prepro**") containing a batch file ("**launch.bat**") containing the pre-processing instructions. The folder must be placed in your language package at the same level as the xxxMetaModel.xml file and other XML files.

> ⓘ You can also include in the **prepro** folder **other batch files** or **tools** that may be required for the pre-processing. This will not affect the analysis of your source code - they will only be run if the launch.bat file contains specific instructions to do so.

launch.bat

The **launch.bat** file will contain the instructions for pre-processing your code to ensure that it can be analyzed correctly by the Universal Analyzer using your language package - what you put in this file entirely depends on what you need to do to your code before it is analyzed.

The only specific requirement is that the pre-processing does not interfere in any way with the structure or organization of the source code (i.e. number of files/folders, file/folder hierarchy, file/folder naming) - in other words, the **output folder** (i.e. the result of the pre-processing) **must be identical** to the **input folder** (i.e. the source code as packaged by the CAST Delivery Manager Tool) once the **launch.bat** file has been executed successfully.

How it works

After creating a version with the CAST Delivery Manager Tool containing your source code and having correctly packaged/installed your custom Universal Analyzer language package so that this source code is identified correctly, when you launch an analysis or a snapshot generation (including an analysis) with the CAST Management Studio, CAST will first check to see whether the language pack contains a folder called "**prepro**" containing a file called "**launch.bat**" - if these two criteria are met, then the analyzer will execute the launch.bat file using two parameters:

- the input folder (i.e. where your unprocessed source is stored - this will be in the **Source Code Deployment Folder** managed by the CAST Management Studio).
- the output folder (i.e. where your pre-processed source code will be stored for analysis - this will be in a temporary folder stored in the **LISA** location defined in the CAST Management Studio)

Once the launch.bat file has been successfully executed, the Universal Analyzer will analyze the source code contained in the output folder.

If the language pack does not contain a "**prepro**" folder containing a "**launch.bat**", then the analysis will continue as normal using the source code in the Source Code Deployment Folder.

> ⓘ Please note that if a source code file matches at least two different Universal Analyzer language packs, and both language packs contain a prepro folder containing a launch.bat file then the Universal Analyzer will ignore this file and it will not be analyzed and no metrics/Quality Rules will be run against the file.