



XiP Advanced Usage

ver1.0

4-20-2012

| | |
|---|---|
| A. Constructing a workflow package with XiP | 3 |
|---|---|

A. Constructing a workflow package with XiP

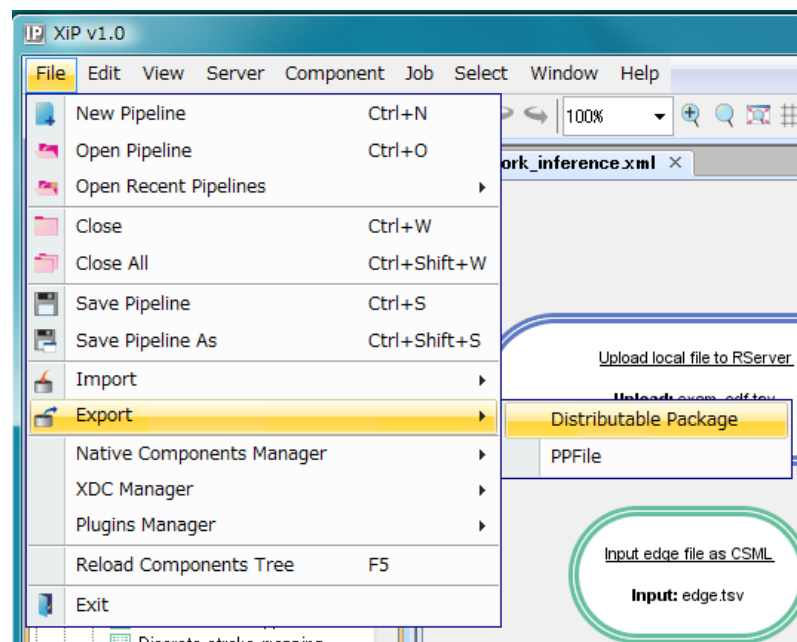
Here, we will describe how to build a stand-alone XiP package. We will assume that the workflow is the “network_inference.xml” constructed in the “Quick Guide 1.0”.

1. Construction of the package

1.1 At the XiP platform, open the file corresponding to the workflow that will be used to construct the package.

Go to the XiP menu bar and select the [Open] option ([File] -> [Open]) or go to the “File explorer” and drag and drop the workflow file to XiP canvas.

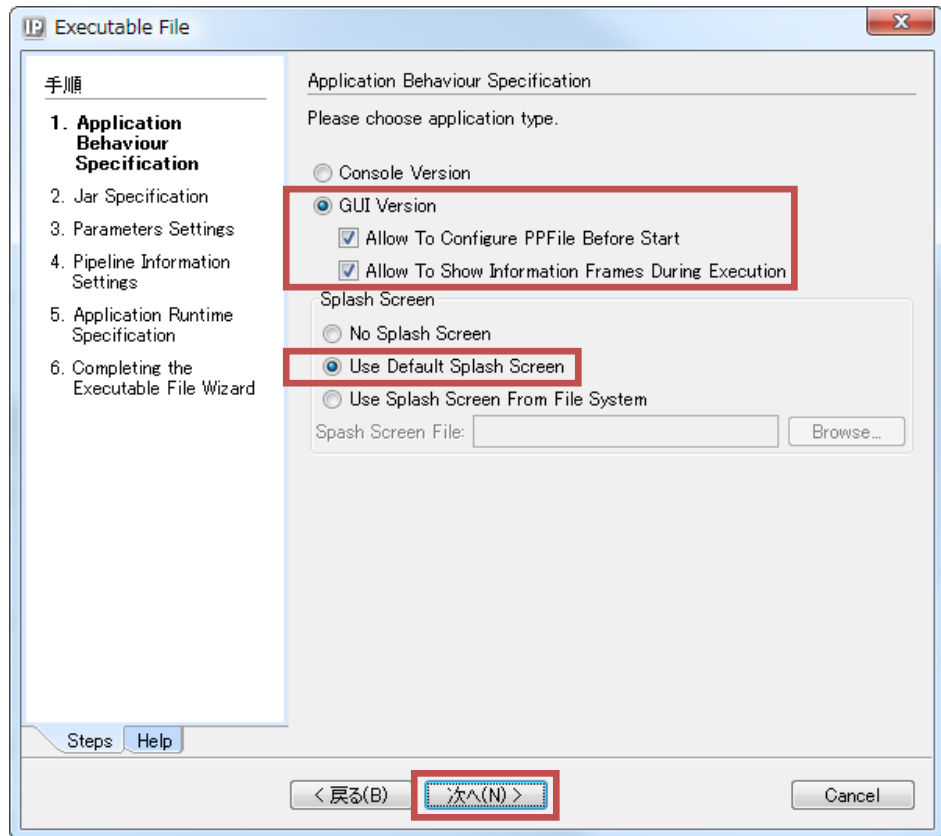
1.2 At the XiP menu bar, select the option [Distributable Package] ([File] -> [Export] -> [Distributable Package]).



1.3 At the [Executable File] screen, set up the [Application Behaviour Specification] as described below, and then, press the [Next] button.

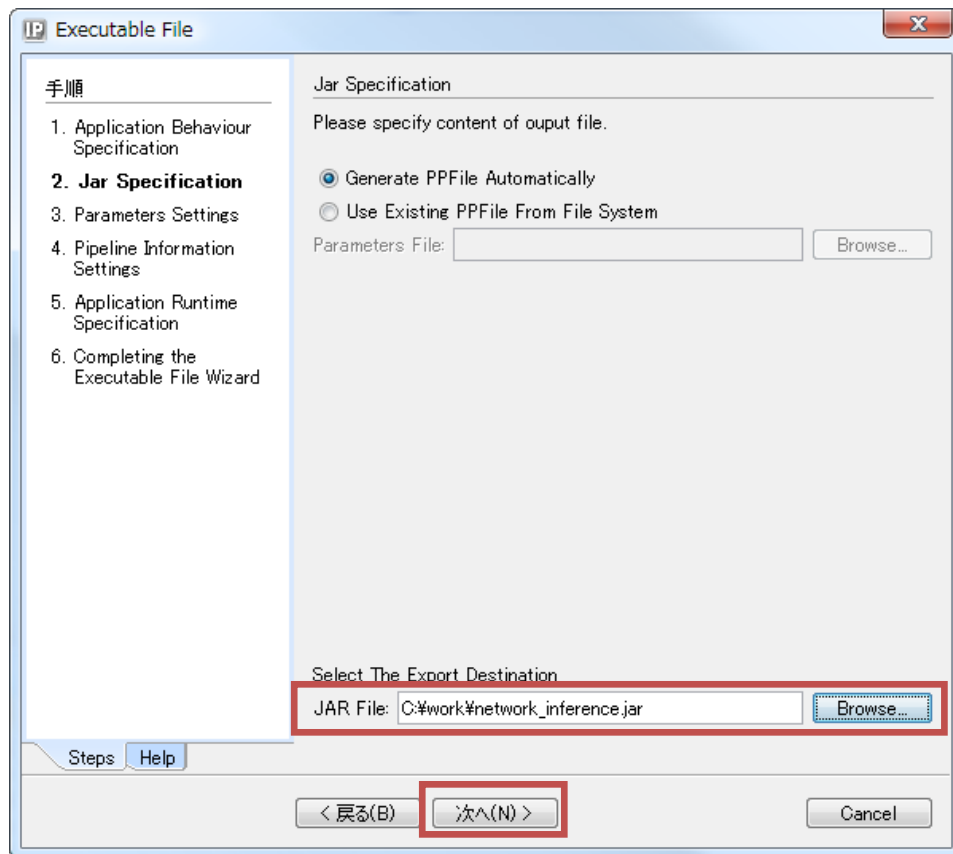
application type : select the “GUI Version”

Splash Screen : select the “Use Default Splash Screen”



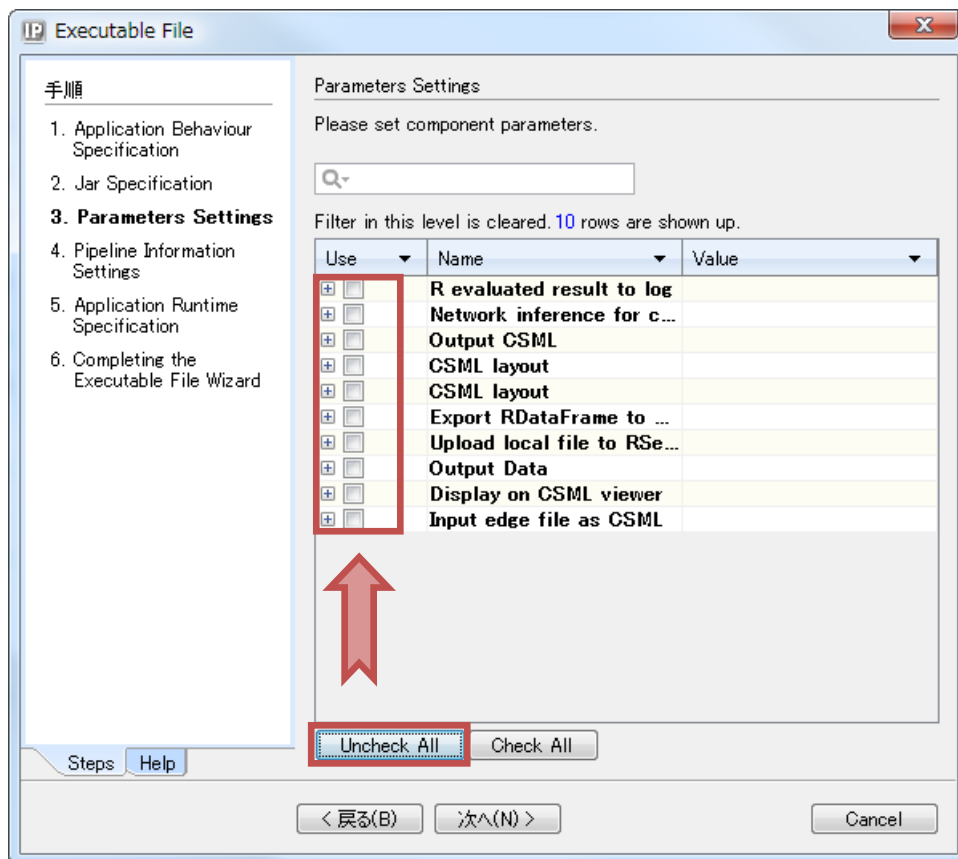
1.4 At the “Jar Specification” screen, set up the parameters as shown below and press the [Next] button.

JAR File : Press the “Browse” button and, at the “File Chooser”, select the directory where the file will be saved and the file’s name. Here, we will assume the following directory and file name: “C:¥work¥network_inference.jar”.

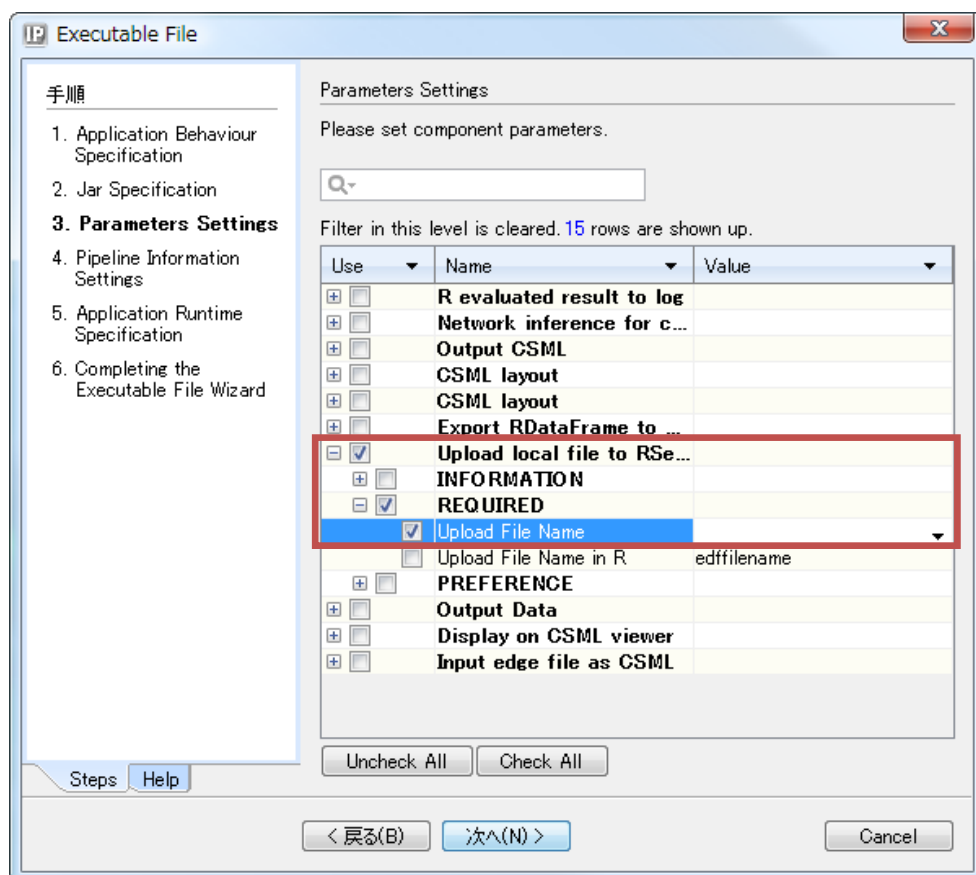


1.5 At the “Parameters Settings” screen

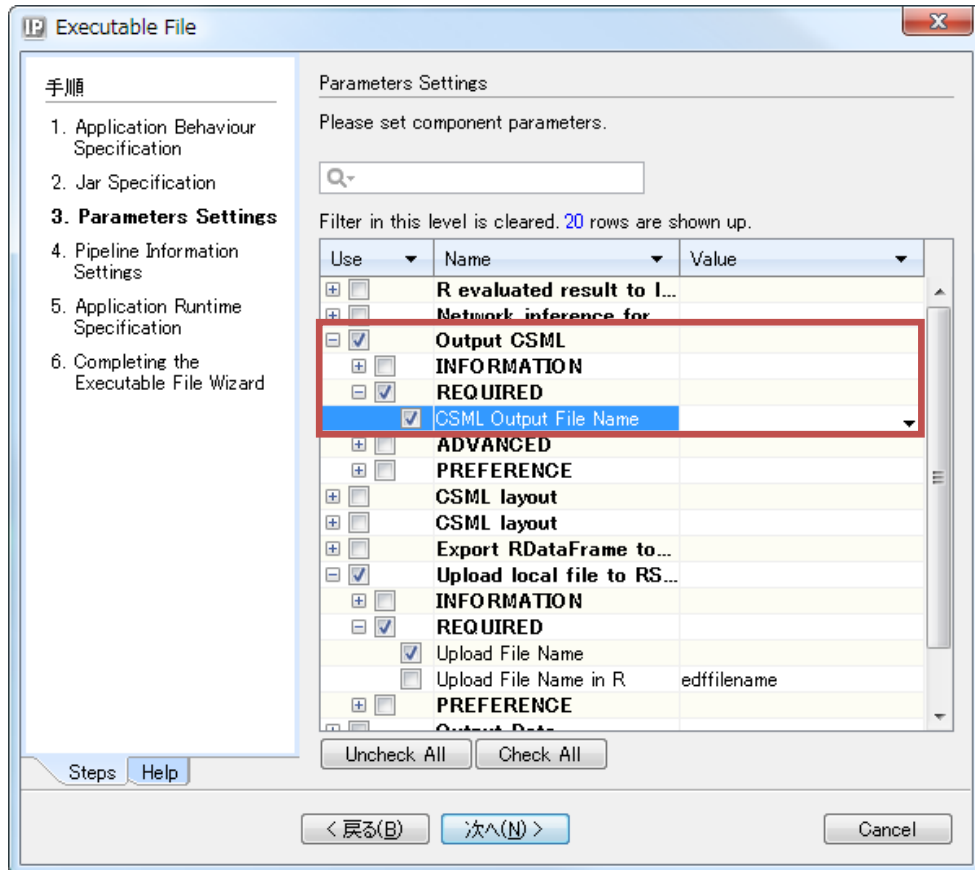
1.5.1 Since this time, the parameters that will be opened to users are only a few ones, firstly, press the “Uncheck All” button. Note that the order of the displayed parameters may differ from that in the following screenshot.



1.5.2 Check the option [Upload File Name] ([Upload local file to RServer] -> [REQUIRED] -> [Upload File Name]) and delete the value.

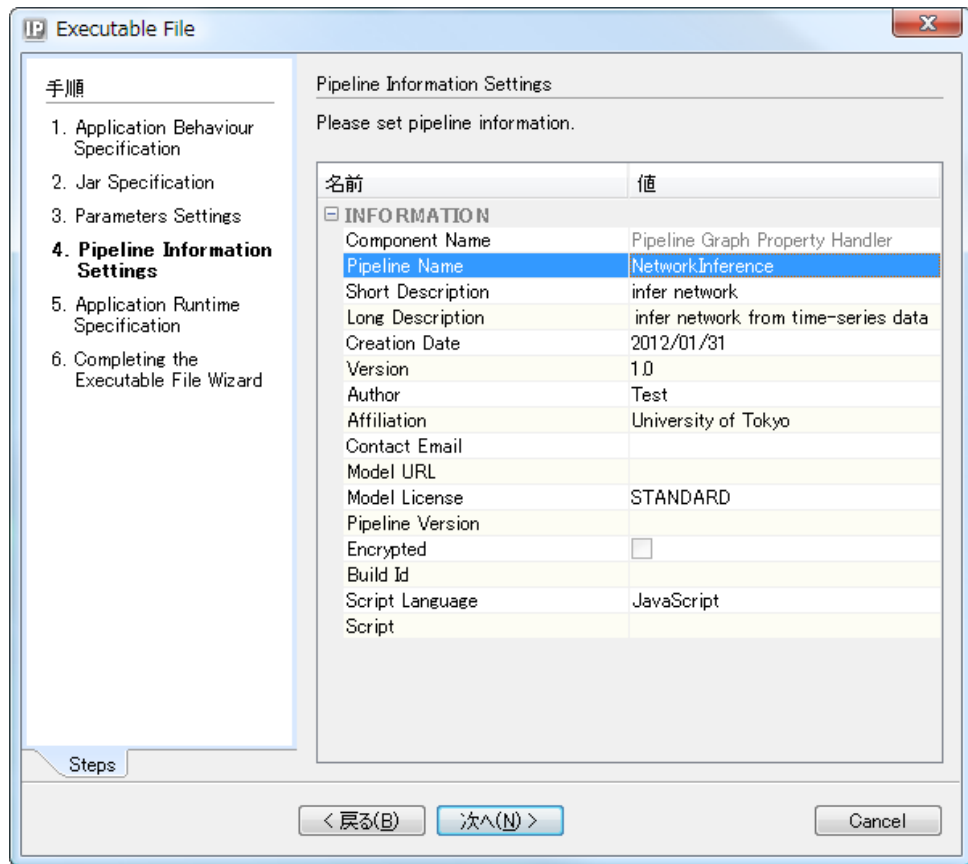


1.5.3 Mark the [CSML Output File Name] ([Output CSML] -> [REQUIRED] -> [CSML Output File Name]) option and delete the value.

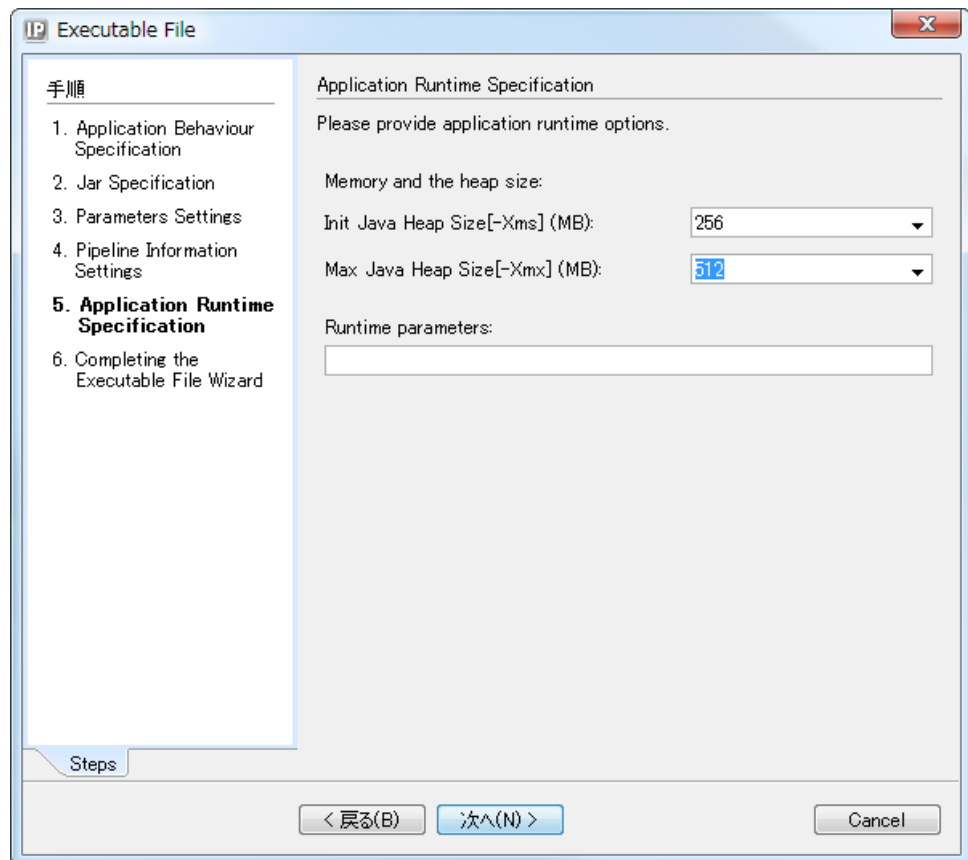


1.5.4 Click on the [Next] button.

1.6 At the [Pipeline Information Settings] screen, set up parameters such as the [Pipeline Name](the title of this package) and [Author] and press the [Next] button.



1.7 Set up the [Application Runtime Specification], [Java Heap Size] and [Runtime parameters] and press the “Next” button.



1.8 Check out the “Summary” settings and press the [Finish] button. Finishing the “Progress animation”, the construction of the jar package file is concluded.

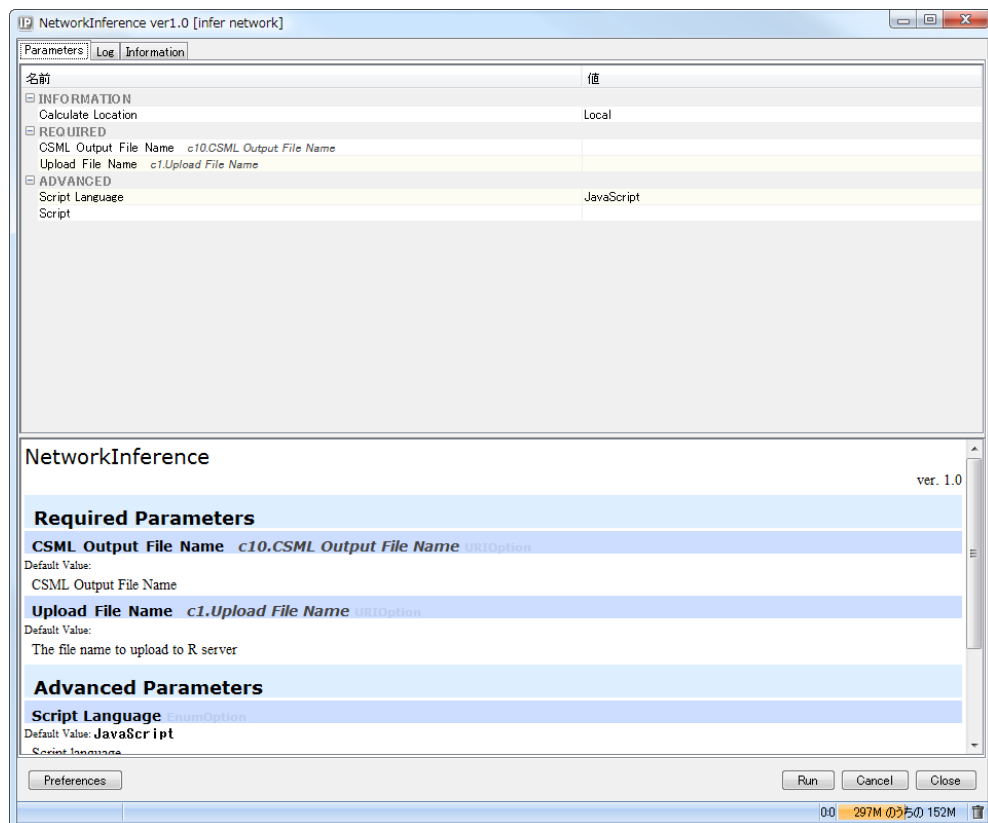
2. Starting the package application

2.1 By using the [File explorer], go to the directory where the package application is and double click the jar file with the mouse.

In case the jar file does not start or the jar file is not associated to a java command, go to the directory of the package by using the console screen and type the following command:

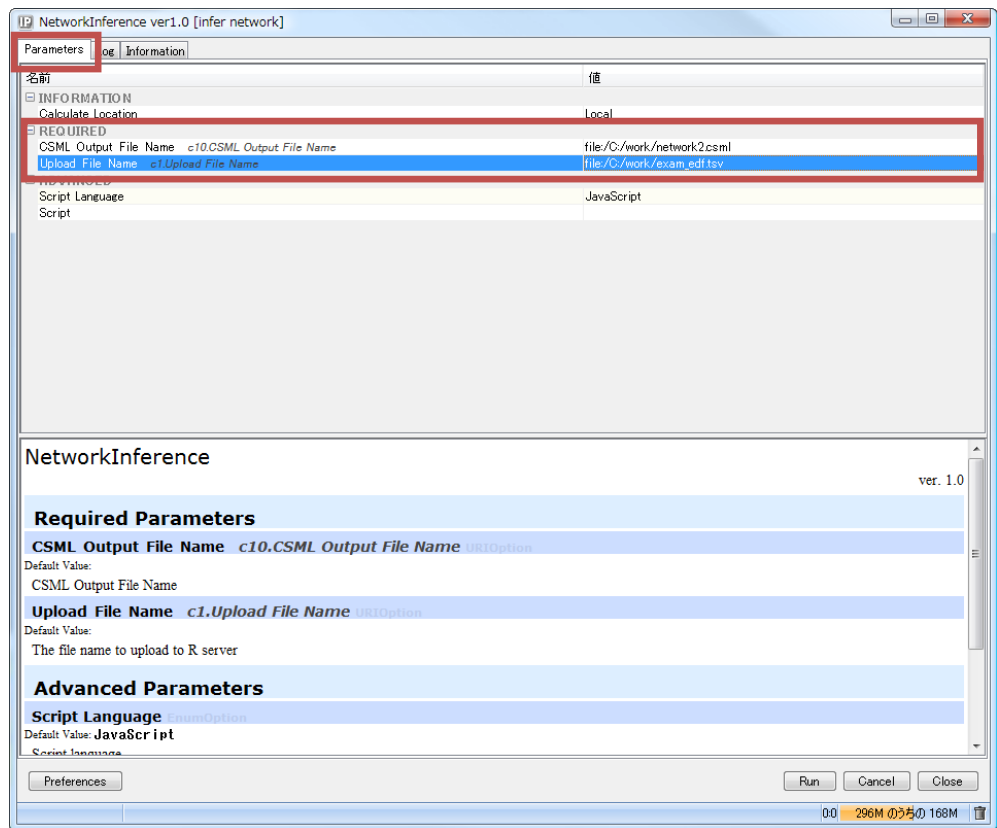
```
> java -jar <name of the package>
```

Then, the package starts as shown below.



2.2 Press the [Preferences] button at the left-bottom part of the screen. Then, at the R Preferences screen, set the [Host Name], [Port], [User Name] and [Password] to connect to the R server and press the [OK] button at the end.

2.3 At the [Parameters] tab screen, set up both the “CSML Output File Name” and the “Upload File Name” parameters.



2.4 Press the [Run] button to execute the workflow.