Paradigm™

Reservoir Simulation Interface

SKUA® 2009.2 and GOCAD® 2009.2

Rock & Fluid Canvas™ 2009 | Epos™ 4.0 Rollup 1

Configuration Guide

Published October 28, 2009

Configuring Reservoir Simulation Interface

If you have a Reservoir Simulation Interface license and you want to launch flow simulations from Paradigm™ SKUA® 2009.2 or Paradigm™ GOCAD® 2009.2, your Information Technology department must configure simulator launch parameters for Reservoir Simulation Interface based on:

- The simulator used (ECLIPSE or 3DSL)
- The local operating system (Windows, Linux, or both)
- The mechanism through which the simulator executable is accessed (locally via the command line, remotely via a remote shell (RSH) on another host, or through a Web portal by specifying a Web address)

A default configuration template (default_run_config.xml), which resides in the following location, must be edited and renamed or copied to run_config.xml in the same location:

InstallationPath/PDGM/GOCAD-SKUA-2009.2/ReservoirProduction/lib/app-defaults/

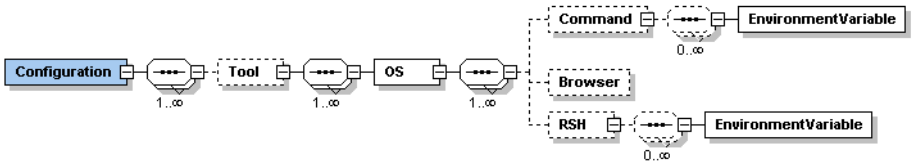
Modifying this template may require the assistance of your site system engineer or a Paradigm consultant. The configuration file can contain information relating to multiple simulators, operating systems, or launch mechanisms.

Your Information Technology department can incorporate custom modifications to the launch panel by defining a user interface (UI) component that allows you to enter or select certain launch parameters directly from the Reservoir Simulation Interface UI, rather than storing them in the configuration file. Default parameters can be set for this UI component.

Configuration File Format

Figure 1 shows the hierarchy of elements that make up the configuration file.

Figure 1 Hierarchy of elements in the configuration file



Below the mandatory **Configuration** description, the following elements must be specified (bold text corresponds to the labels in Figure 1; italicized text corresponds to words in the example configuration file in Figure 2):

- **Tool** is where you specify the *name* of the simulator, that is, "3DSL" or "Eclipse."
- **OS** is where you specify the *type* of operating system for which the launch mechanism description that follows will be valid, that is, "Windows," "Linux," or "All."

One of the following launch mechanisms must then be specified:

- **Command** to launch the executable through a command line
- **Browser** to go to a specific Web address for job submittal
- **RSH** to remotely log on to another computer and launch the simulation

Each will have the following components:

- **description.** A unique name given to a particular launch configuration that will appear on the RSI launch panel

- **value.** An instruction executed at launch (see Table 1)
- **uicomponent** (optional)**.** The location of the .ui file containing the custom parameter entry UI component (see "Customizing the User Interface" on page 1-4).

Environmental variables, for example specifying the location of the license file, that need to be set before the simulation is launched, must be specified as **EnvironmentVariable** which includes the following components:

- *name*. The name of environment variable
- *value*. The corresponding value

Table 1 Instruction executed for each launch mechanism

Launch mechanism	Launches
Command	The *value* attribute which should be the path to the simulator executable and the parameter file
Browser	The local default browser with the Web address specified in the *value* attribute
RSH	The RSH command with the parameters described in the value attribute

Configuration File Example

Figure 2 shows an example configuration file, run_config.xml.

Figure 2 Example configuration file

```
<?xml version="1.0" encoding="UTF-8" ?>
- <Configuration xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="Configuration.xsd">
  <!--   Sample configuration for job submission   -->
  - <Tool name="3DSL">
    - <OS type="Win32">
      - <Command description="3DSL 2.20" value="C:\Users\Jack\EarthDecision\3dsl-pc-060705.exe
          "$PARAMETER_FILE"">
          <EnvironmentVariable name="LSFORCEHOST" value="ODEON" />
        </Command>
      - <Command description="Show Env. Variables" uicomponent="default_custom_form.ui" value="cmd /C
          echo $PARAMETER_FILE () $PARAMETER_FILE_WO_EXT () $UI.input_1 () %USERNAME%">
          <EnvironmentVariable name="LSFORCEHOST" value="ODEON" />
        </Command>
      - <Command description="3DSL 2.40 (for testing only)" value="C:\Users\Jack\EarthDecision\3dsl-pc-
          060705.exe "$PARAMETER_FILE"">
          <EnvironmentVariable name="LSFORCEHOST" value="ODEON" />
        </Command>
      </OS>
    - <OS type="linux-i86">
      - <Command description="Launch local" value="/apps/3dsl/bin/3dsl $PARAMETER_FILE">
          <EnvironmentVariable name="LSFORCEHOST" value="odeon" />
        </Command>
      </OS>
    - <OS type="All">
        <Browser description="Launch browser" value="http://portal.fr/3dsl?deck=$PARAMETER_FILE" />
        <RSH description="Launch remote" value="host_name 3dsl $PARAMETER_FILE" />
      </OS>
    </Tool>
  - <Tool name="Eclipse">
    - <OS type="Win32">
      - <Command description="Launch local" value="W:\Applications\Eclipse $PARAMETER_FILE_WO_EXT">
          <EnvironmentVariable name="LM_LICENSE_FILE" value="7321@odeon" />
        </Command>
      </OS>
    - <OS type="linux-i86">
      - <Command description="Launch local" value="/apps/eclipse/bin/eclipse
          $PARAMETER_FILE_WO_EXT">
          <EnvironmentVariable name="LM_LICENSE_FILE" value="=7321@odeon" />
        </Command>
      </OS>
    - <OS type="All">
        <Browser description="Launch browser" value="http://portal/eclipse?deck=$PARAMETER_FILE" />
        <RSH description="Launch remote" value="host_name eclipse $PARAMETER_FILE" />
      </OS>
    </Tool>
</Configuration>
```

Figure 3, Figure 4, and Figure 5 show the resulting **General Settings** (launch) panel with **Browser**, **Local host**, and **RSH** selected, respectively.

Figure 3 General Settings panel with Browser selected

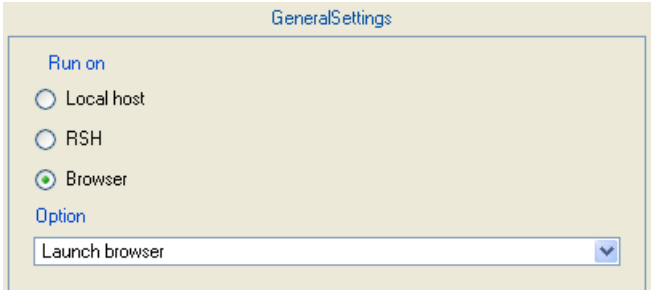


Figure 4 General Settings panel with Local host selected

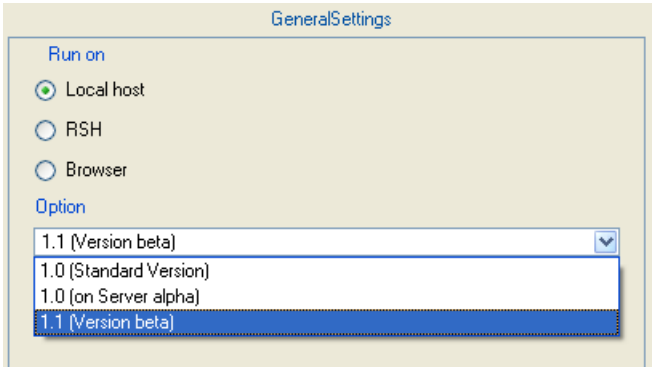


Figure 5 General Settings panel with Local host selected and custom UI component specified

Customizing the User Interface

The customization of the UI ("uicomponent" attribute) is limited to the following controls:

- **QLineEdit** (single line edit control)**.** The corresponding string variable will be replaced by the content of edit control.
- **QCheckBox** (check box)**.** If selected, the corresponding string variable is replaced by the label of the check box; if cleared, the string variable is replaced by empty text.
- **QButtonGroup** (exclusive and filled with QRadioButton)**.** The corresponding string variable is replaced by the label of the selected radio button.

- **QSpinBox** (edit control and spin buttons)**.** The corresponding string variable is replaced by the value of the edit control.
- **QListBox** (single selection mode)**.** If a selection exists, the corresponding string variable is replaced by the selection text; if a selection does not exist, the macro is replaced by empty text.
- **QComboBox** (can be editable)**.** The string variable is replaced by the text of the combo selection.

Notes

- You should store the .ui file or files in the folder where run_config.xml resides: */InstallationPath/*PDGM/GOCAD-SKUA-2009.2/ReservoirProduction/lib/app-defaults/. A default file (default_custom_form.ui) is available as a template.
- Every "value" attribute (in Command, Browser, Rsh, and EnvironmentVariable) can contain references to UI fields. Those references are in the form of a string variable, "$UI.field_name". This is illustrated in the preceding example, where the first widget is called "input_1," and its value is used through the string variable, "$UI.input_1".

Predefined String Variables

Table 2 lists the predefined string variables of the *value* attribute.

Table 2 Predefined string variables

Macro	Description
$PARAMETER_FILE	Full path name of parameter file (for example, "C:\Users\My_Models\study.dat"). This file will be generated by RSI compliant with the simulator selected.
$PARAMETER_FILE_WO_EXT	Full path name of parameter file without extension (for example, "C:\Users\My_Models\study"). This file will be generated WITH the extension by RSI compliant with the simulator selected.