



High-Performance Reservoir Risk Assessment (Jacta Cluster) Configuration Guide

GOCAD Suite 2.5

 **Paradigm™**

Powered by GOCAD

Copyright © 2008, Paradigm Geophysical Corp. All rights reserved.

The information in this document is subject to change without notice and should not be construed as a commitment by Paradigm Geotechnology B.V. or any of its affiliates (collectively, "Paradigm"). Paradigm assumes no responsibility for any errors that may appear in this document.

The Copyright Act of the United States, Title 17 of the United States Code, Section 501 prohibits the reproduction or transmission of Paradigm's copyrighted material in any form or by any means, electronic or mechanical, including photocopying and recording, or by any information storage and retrieval system without permission in writing from Paradigm. Violators of this statute will be subject to civil and possible criminal liability. The infringing activity will be enjoined and the infringing articles will be impounded. Violators will be personally liable for Paradigm's actual damages and any additional profits of the infringer, or statutory damages in the amount of up to \$150,000 per infringement. Paradigm will also seek all costs and attorney fees. In addition, any person who infringes this copyright willfully and for the purpose of commercial advantage or private financial gain, or by the reproduction or distribution of one or more copies of a copyrighted work with a total retail value of over \$1,000 shall be punished under the criminal laws of the United States of America, including fines and possible imprisonment.

The following are trademarks or registered trademarks of Paradigm Geotechnology B.V. or any of its affiliates (collectively, "Paradigm") in the United States or in other countries: Paradigm™, Paradigm logo, Alea®, Artist™, Cement-IT™, EarthStudy 2D™, e-Geoscience™, Epos™, Explorer™, EZdip™, Focus™, GeoDepth™, GeoLAN™, Geolog®, GeoScene™, GeoSec™, GOCAD®, iMap™, IntegralPlus™, Jacta®, LogLAN™, MapLAN™, NexModel™, OpenGeo Tools™, PlaNet™, Probe™, Pump-IT™, Reservoir Navigator™, ResSim™, Rock & Fluid Canvas™, SeisEarth™, SeisFacies™, SeisX™, SKUA™, Slider™, SolidGeo™, Stratimagic™, Surveyor™, Sysdrill™, Sysdrill Director™, Sysdrill DirectorGeo™, Sysdrill IDEAS™, Sysdrill SaG™, Vanguard™, Velocity Navigator™, VG Tools™, VolLAN™, VoxelGeo™, VXplot™, and/or other Paradigm products referenced herein.

All rights not expressly granted are reserved.

The following are trademarks or registered trademarks of their respective companies or organizations:

Linux is a registered trademark of Linus Torvalds. Windows is a registered trademark of Microsoft Corporation in the United States and/or other countries.

All other trademarks are owned by their respective owners.

Before You Begin

About This Guide

Instructions in this guide assume GOCAD Suite is already installed on the computer that will submit batch jobs to the computer cluster. Jacta Cluster is installed automatically when you install GOCAD Suite. If GOCAD Suite is not already installed, install it now. For instructions, see the *Installation Guide*.

This guide is intended for experienced UNIX system administrators. Instructions in this guide assume a familiarity with the UNIX operating systems, TCP/IP networking, and intermediate shell programming.

Contents

Before You Begin	iii
About This Guide	iii
Configuring Jacta Cluster.....	1
Installing a Batch Management System.....	2
About Batch Management Systems	2
Supported Batch Management Systems	2
Installing the Batch Server.....	3
Installing the Clients	3
Configuring the Batch Manager	4
Configuring Jacta Cluster.....	7
Submitting Jobs from Jacta	8
Jacta Cluster Cookbook	9

Configuring Jacta Cluster

In this chapter

- "Installing a Batch Management System," page 1-2
- "Configuring the Batch Manager," page 1-3
- "Configuring Jacta Cluster," page 1-7
- "Submitting Jobs from Jacta," page 1-8

Overview

With High-Performance Reservoir Risk Assessment (Jacta Cluster), you can use a cluster of computers to carry out Reservoir Risk Assessment (Jacta) calculations. One workstation controls this cluster by using a batch management system.

To use Jacta Cluster, you must install and configure a batch management system to work with the Jacta system. Then you configure Jacta and submit the jobs.

1.1 Installing a Batch Management System

You must install a batch management system to run Jacta Cluster. For more information, see the following:

- "About Batch Management Systems," page 1-2
- "Supported Batch Management Systems," page 1-2
- "Installing the Batch Server," page 1-2
- "Installing the Clients," page 1-3

About Batch Management Systems

Batch management systems consist of a head node, which divides submitted instructions into subsets and distributes them among compute nodes.

The compute nodes finish their calculations and return that information to the head node. That output can then be viewed from the workstation which submitted the job. A workstation is a computer with the program sending the batch job (in this case, Jacta). The workstation may be the same physical machine as a head or compute node.

The instructions in this guide use Torque as an example of how to install a batch management system. Torque refers to the head node as a server computer and the compute nodes as client computers, so that terminology will be used.

PBS-compatible versions

The following versions of the PBS can be used:

- PBS Gridworks (<http://www.pbsgridworks.com/>)
- Torque Resource Manager (<http://www.clusterresources.com/pages/products/torque-resource-manager.php>), commonly referred to as "Torque"
- OpenPBS (<http://www.openpbs.org/>)

Installing the Batch Server

The first task is to download the Torque source files and compile them on the server computer.

Note This installation requires root privileges.

To install the batch management system on the server

Download the Torque source files (<http://www.clusterresources.com/downloads/torque>) from the Web site and then uncompress them.

```
tar -xzvf torque-x.x.x.tar.gz
```

1 Run the configuration utility.

```
cd torque-x.x.x  
./configure
```

2 Compile the sources into a program.

```
make
```

3 Build the program on the server computer.

```
make install
```

The default installation folder is:

/usr/local/bin or /usr/local/sbin

The configuration file is in this folder:

/var/spool/torque

Installing the Clients

The next task is to generate software for the client computers and then install the client software on each computer that will receive commands from the batch management server. The Torque software on each client computer downloads and parses instructions from the batch management server.

Note This installation requires root privileges.

To install the program on each client

- 1 On the server computer, continuing from the previous procedure, create the client package.

make packages

The make script generates sh scripts named *torque-package-server-operating system-architecture.sh* and *torque-package-client-operating system-architecture.sh*, such as *torque-package-server-linux-i686.sh*

- 2 On each client, copy the client package to a temporary folder.
- 3 On each client, run the client package. For example:

```
./torque-package-client-linux-i686
```

Repeat this procedure for each client machine.

Configuring the Batch Manager

After you install the batch management system, the next task is to configure the various parts of the system to communicate with each other. You must also configure a Jacta submission template for the type of job you will generally submit.

For instructions, see:

- "To configure the server computer," page 1-3
- "To configure the client computers," page 1-4
- "To configure SSH," page 1-4
- "To configure the submission template," page 1-5
- "To verify the configuration," page 1-7

To configure the server computer

- 1 Specify your client configuration by creating or editing the node file `/var/spool/torque/server_priv/node` to add an entry for each trusted node.

```
node1 np=2
node2 np=2
node3
node4
```

The node names correspond to the name of each computer node computer. Each client is assumed to have a single processor unless you use the argument `np` to specify a number of processors. In this example, `node1` and `node2` have two processors, whereas `node3` and `node4` have one.

- 2 Run the configuration script, where *Username* designates the user who will administer this server.

```
./torque.setup Username
```

- 3 Add additional users as needed with the **qmgr** command.

qmgr

Qmgr: **set server** *Group* += *User*

- *User* is the UNIX account you want to grant rights to.
- *Group* is either **manager** (can configure) or **operator** (can submit jobs).

- 4 Do the following:

- a Start the server that receives batch jobs:

pbs_server

- b Start the scheduler that prioritize tasks and manages queues.

pbs_sched

To configure the client computers

On each client, do the following:

- 1 Edit the server name. The following file must contain the name of the server.

```
/var/spool/torque/server_name
```

Note If you used the generated packages to set up your clients, this information should already be present.

- 2 Edit the server location in the following file.

```
/var/spool/torque/mom_priv/config
```

The file must contain the following line:

```
pbsserver ip-address
```

Where *ip-address* is a dot-delimited IP address in the form 10.10.10.1.

- 3 Run the command daemon.

pbs_mon

This daemon waits for commands.

To configure SSH

Torque uses the secure copy protocol (SCP) to copy files between server and clients. For scp to work correctly, you must configure the underlying transfer system, secure shell (SSH), to work automatically.

The SSH system uses cryptographic keys to authenticate users. You must generate a key pair for a user account that can log on to both server and client computers. After you generate these keys, you can then copy them from the server to the clients so the two can communicate without your intervention.

- 1 On the server, make sure you are logged on as the user account that will run batch jobs and then do the following:

- a If `~/.ssh/id_rsa` and `~/.ssh/id_rsa.pub` do not exist, generate a key.

```
ssh-keygen -t rsa
```

- b Copy the user key to `~/.ssh/authorized_keys`.

```
cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys
```

- c Test the connection to each client.

```
ssh Username
```

Where *Username* includes node information, such as user@node1. Reply **yes** if the server asks if you want to accept this key.

- 2 On each client, make sure you are logged on as the user account that will run batch jobs and then do the following:

- a Copy the SSH keys to each client.

```
scp -r .ssh Username
```

Where *Username* includes node information, such as user@node1.

- b Test the connection to the server.

```
ssh Username
```

Where *Username* includes node information, such as user@server1. Reply **yes** if the server asks if you want to accept this key, but do not enter any password.

Note Do not enter a password. If you are prompted to do so, your SSH is not configured correctly. Start the procedure over.

To configure the submission template

The submission template defines options that are consistent across batch jobs. Some can be changed later when a job is submitted to the Jacta Cluster. For more information, see "Submitting Jobs from Jacta" on page 1-8.

You can modify the standard Jacta submission template (see Figure 1) to fit your needs by using the keywords defined in Table 1.

Figure 1 Standard Jacta submission template

```
#!/bin/sh
# *****
# *      Jacta for Clusters - job submission template
# *****
#PBS -o %OUTPUT_PATH%/%SIM_RUN%.%JOB_NAME%.out
#PBS -e %OUTPUT_PATH%/%SIM_RUN%.%JOB_NAME%.err
#PBS -q %SUBMISSION_QUEUE%
#PBS -m n
#PBS -N %JOB_NAME%
#PBS -l nodes=1

GOCADLMD_LICENSE_FILE=%LICENSE_SERVER%
GOCAD_LICENSE=jcrtl
GOCADPATH=%GOCADPATH%
GOCADHOME=%GOCADHOME%
LD_LIBRARY_PATH=%LD_LIBRARY_PATH%

export LD_LIBRARY_PATH GOCADLMD_LICENSE_FILE GOCAD_LICENSE GOCADPATH GOCADHOME

%JACTABATCH% "%PROJECT_PATH%" --test-mode -out-runs "%OUTPUT_FILE%" -iteration-start
%FIRST_ITERATION% -iteration-end %LAST_ITERATION% -simulation-run %SIM_RUN% -ssid %SSID% -
crc %CRC%
```

Table 1 Submission batch file keywords

Keyword	Type ¹	Changeable in submission interface?	Description
%LICENSE_SERVER%	Global	Yes	Host name of license server to be used for Jacta.
%SIM_RUN%	Global	Yes	Name of the output of this version of the simulation.
%PROJECT_PATH%	Global	No	Absolute path to the current project. Default is the location of the current Jacta project.
%OUTPUT_PATH%	Global	Yes	Absolute path to where simulation results will be stored.
%SUBMISSION_QUEUE%	Global	No	Name of queue where the job will be submitted.
%JACTABATCH%	Global	No	Absolute path to the JactaBatch executable. Default is defined by Jacta Cluster.
%TEMP%	Global	No	Absolute path to the temporary folder. Default is defined in the Preferences dialog box in GOCAD Suite. For more information, see "Configuring Jacta Cluster" on page 1-7.
%FIRST_ITERATION%	Local	No	Number of the first realization computed by the node.
%LAST_ITERATION%	Local	No	Number of the last realization computed by the node, plus one.
%JOB_NAME%	Local	No	Job name. Default is automatically defined by Jacta Cluster based on the run name and job number.
%SSID%	Local	No	Session ID. Default is defined by Jacta Cluster.
%CRC%	Local	No	Cyclic redundancy check. Default is defined by Jacta Cluster.

1. Global variables are consistent across all nodes. Local variables are defined only on one node.

To verify the configuration

After you install the Torque batch management system, you may want to verify your configuration. For more information, see "To test your Torque configuration" on page 1-10.

The next task is to configure Jacta to work with your batch management system. For more information, see "Configuring Jacta Cluster" on page 1-7.

Configuring Jacta Cluster

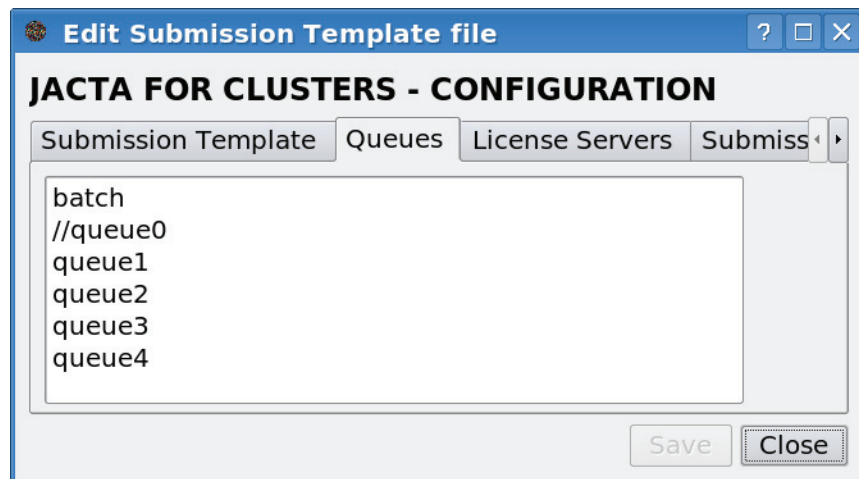
Once the batch management system is working, the next task is to tell Jacta Cluster what commands it must use to communicate with your batch processing system. The information you enter into Jacta will reflect what you set up with your batch management system in "Configuring the Batch Manager" on page 1-3.

To configure job submission options

- 1 On the batch management server, start GOCAD Suite and load your project. For more information, see the *User Guide, Part I: Getting Started*.
- 2 In the **GOCAD** window window, click the **Edit** menu, and then click **Preferences**.

Note You must have write permissions to the default working folder in order to save changes to your preferences.

- 3 In the **Default working folder for finding and saving objects and projects** box, enter the folder where you want to store the temporary files from cluster processing.
Note One temporary file will be created per job, and it can be safely removed after the job is processed.
- 4 In Jacta, ensure all simulation parameters in the **Parameter Browser** window are correct. For more information, see the *User Guide, Part IX: Reservoir Modeling*.
- 5 Open the **Uncertainty Viewer** window. For more information, see the *User Guide, Part IX: Reservoir Modeling*.
- 6 On the **Cluster Tools** menu, click **Edit Sub. Template File** to open the **Edit Submission Template File** dialog box.
- 7 Click the **Queues** tab, type the names of your queues as defined in your batch management system, and then click **Save**.



- 8 Do the following to define the license server for each client computer:
 - a Click the **License Servers** tab.
 - b Add the license server the clients should use. Specify one license server per line in the form *port@license server name*. For example:
27000@examplelicenseserver
 - c Click **Save**.
- 9 Click the **Submission Command** tab, enter the appropriate command for the type of batch management system you have installed, and then click **Save**.

Batch system	Submission command
PBS	qsub

If your batch system installation folder is not in your system path, you can enter an absolute path. For example:

/usr/sbin/qsub

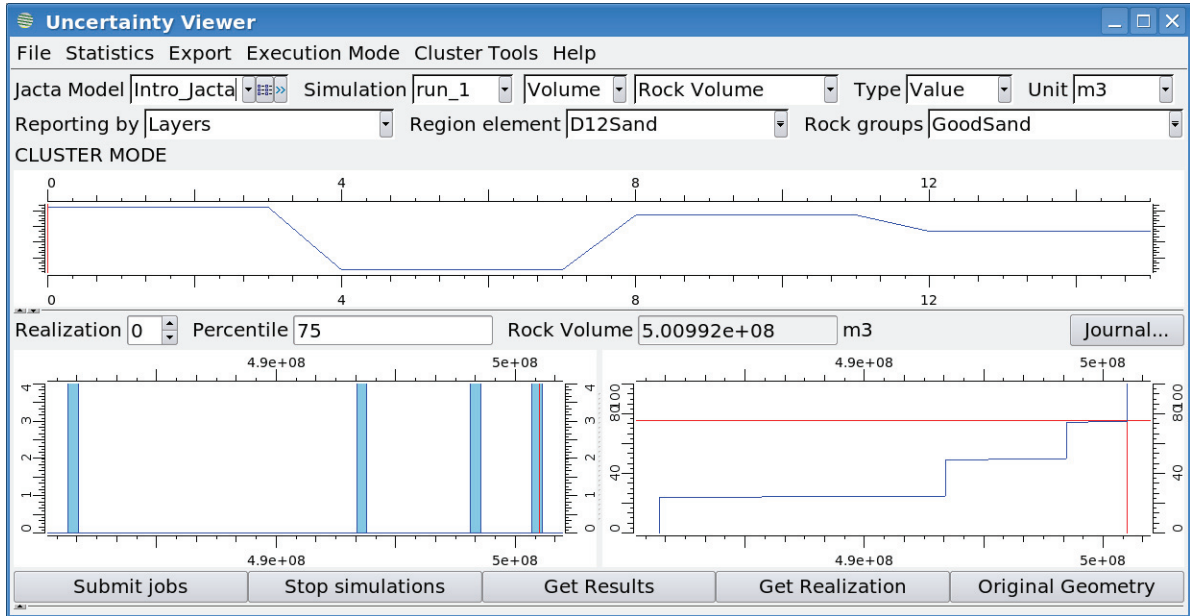
- 10 When you are finished, click **Close**.

The next task is to submit jobs to your Jacta cluster. For more information, see ["Submitting Jobs from Jacta" on page 1-8](#)

Submitting Jobs from Jacta

Each time you submit a job to Jacta Cluster, you must ensure that the submission options match your configuration. The following first-time configuration shows what options must be configured for Jacta Cluster jobs.

- To submit jobs** 1 In the **Uncertainty Viewer** window, click the **Execution Mode** menu, and then click **Cluster** to switch to cluster mode.



- 2 Click the **Submit Jobs** button to open the dialog box, and then do the following:

Submit Jobs to Cluster

JACTA FOR CLUSTERS - JOB SUBMISSION

Current Project

Jacta Model	Simulation Run	Number of realizations
Intro_Jacta	run_1	16

License Server

License Server: @riesling1

Queue

Queue Name: batch

Paths

Project Location: /data/projects/jacta_project.prj

Saving Location: /data/projects/jacta_project.prj/simulation_results

Jobs

Number of jobs to submit: 4

Submit Jobs

- a In the **License Server** box, select a license server. Available license servers are defined in your Jacta Cluster configuration. For more information, see ["To configure job submission options" on page 1-7.](#)
- b In the **Queue Name** box, select a queue. Available queues are defined in your batch management system and in your Jacta Cluster configuration. For more information, see ["To configure job submission options" on page 1-7.](#)
- c In the **Saving Location** box, type the path to the folder where you want to store the results of the simulation.
- d In the **Number of jobs to submit** box, type a number that is at least one, and less than or equal to the total number of simulations you want to run. If you have a license for four clients and want to run at least four simulations, setting four jobs would make sense.

- 3 Click **Submit Jobs** to run your simulations.

When your simulations are finished, you can click the **Get Results** button in the **Uncertainty Viewer** window to load the simulation results as a project.

Jacta Cluster Cookbook

The following "recipes" are both standard and troubleshooting procedures for common tasks and errors. For more information, see:

- ["To test your Torque configuration," page 1-10](#)
- ["To track errors using logs," page 1-10](#)
- ["To find equivalent commands in PBS and LSF," page 1-10](#)

To test your Torque configuration

On the server, do the following:

- 1 Enter test commands.
 - a Shut down the server.
qterm -t quick
 - b Start the server.
pbs_server
 - c Verify queue configuration.
qstat -q
 - d View additional configuration data.
qmgr -c 'p s'
 - e Submit a test job.
echo "sleep 30" | qsub
 - f Check jobs display.
qstat
- 2 Start the scheduler.
pbs_sched
- 3 Run your test job.
qrun job-id

Where *job-id* is the number of the jobs revealed by the `qstat` command in [step f](#).

To track errors using logs

Table 1–1 lists the location where Torque stores logs.

Table 1–1 Location of Torque logs

Type of node	Log location
Server	/var/spool/torque/server_log/date
Client	/var/spool/torque/mom_log/date

Use *date* to specify the day whose logs you want to see.

