Cresset Engine Broker^{TM} V3.0.1 Release Notes

About Cresset Engine Broker

The Cresset Engine Broker (CEBroker) is a utility to connect your Cresset applications to FieldEngine resources on other computers. It enables Flare™ and Spark™ to use the full power of distributed computing resources with little or no configuration required.

Cresset's tools have always had the capability to distribute their calculations remotely using our FieldEngine technology. However, this could sometimes be difficult to configure. This tool sits in between the desktop application and existing FieldEngines and can also spawn new FieldEngine processes in response to demand. The result is a dynamically configured computing resource that is simple to set up and use and puts the power of a supercomputer on your desktop.

- Use thousands of cores to process calculations from one Flare or Spark project
- Submit calculations to queueing systems such as Slurm, Grid Engine or Platform LSF
- Submit calculations on the cloud (for example, AWS resources)
- Connect the spare processing capacity of computers in your lab to your desktop
- Seamlessly join Windows and Linux machines into a single processing pool
- Run large virtual screening experiments in a few hours.

Supported Platforms

Small clusters specifications with ~50 computation nodes

Hardware	Specifications
RAM	1GB
Processor	Intel i3 processor or better
Disk space	200MB

Medium clusters specifications with ~500 computation nodes

Hardware	Specifications
RAM	10GB
Processor	Intel i5 processor or better
Disk space	200MB

Supported Operating Systems

Red Hat Enterprise Linux 7® (64 bit), Red Hat Enterprise Linux 8® (64 bit).

Installation

1. Install the packages required for running the CEBroker:

   RHEL 7 and CentOS & 7

   ```
   yum install mesa-libGLU libXScrnSaver
   ```

RHEL 8 and CentOS Stream 8

```
dnf install mesa-libGLU libXScrnSaver libnsl
```

Debian and Ubuntu

```
apt-get install libglu1-mesa libxss1
```

2. Browse to the Linux folder of the CEBroker distribution media.
3. Install the CEBroker re-locatable rpm package using one of the command lines below (where `<vers>` is the version number, `<arch>` is the architecture and `<path>` is the install location (`/opt` by default):

   ```
   yum install cresset-cebroker-<vers>-1.<arch>.rpm
   ```

   or

   ```
   rpm -i [--prefix <path>] cresset-cebroker-<vers>-1.<arch>.rpm
   ```

4. Alternatively, unpack the `CEBroker_<vers>.<arch>.tar.xz` file to a suitable location (e.g. `$HOME` or `/usr/local` or `/opt`). Instructions to configure the CEBroker to run as a service on RHEL or Debian can be found at `documentation/INSTALL.Linux.txt.`

In the default RPM install, CEBroker will be installed as a service called ‘cresset-cebroker3’, run as the ‘cebroker’ user. The service will be set to start automatically (see the ‘Configuration’ section below).

Licensing

To run CEBroker you will require a valid license file, which will be supplied by Cresset, usually as an attachment to an email. If you cannot locate your license file, then please contact support@cresset-group.com. The license file should be placed in one of the directories detailed at https://www.cresset-group.com/support/support-resources/license-file-locations/.

Configuration

The behavior of the CEBroker is controlled by the config.ini file. If the CEBroker is being run as a service, the default path to the file is /etc/cebroker3/config.ini. If the CEBroker is being started from the command line then the path to the config.ini file is passed to it as an argument, for example:

```
<install directory>/cresset/CEBroker3/bin/CEBroker3 config.ini
```

An example configuration is provided at `documentation/examples/example_config.ini.`

The key settings in the config.ini which should be updated are:

Port

The port to listen for clients (Flare and Spark) and FieldEngine connections on. If the firewall is enabled, then this port should be opened to allow the network connections.

ScriptFile

The broker daemon calls a script to start FieldEngines. There are example scripts in the ‘`documentation\examples\`’ directory of the installation tree (`/opt/cresset/CEBroker3` by default). The example configuration scripts cover starting FieldEngine processes on the local machine and for submitting them to a Grid Engine queue.

Additional configuration of the xxx_engine.sh scripts may be required to make the Cresset Engine Broker handle both Spark 10.7 and Flare V6 calculations.

MaxCpuSingleCore, MaxCpuMultiCore, MaxGpuSingleCore

These settings control the maximum number of FieldEngines which may be started.

SSL (Encryption)

The CEBroker can be configures to encrypt all network connections. To do this you require an SSL certificate and the private key which was generated with the certificate. The CertificateFile and PrivateKey settings should be set to the absolute path to then files. Then the Enable setting should be set to true.

The CEBroker will automatically reload the certificate and private key if they are updated while the CEBroker is running. This allows expiring certificates to be updated without restarting the CEBroker.

Self-signed certificates may be used but this is not recommended. To use a self-signed certificate the environment valuable CRESSET_ALLOW_SELF_SIGN_CERT=1 must be set on all machines running Flare, Spark and FieldEngines.

To check that encryption is enabled press the “Test Connection” button in the CEBroker section of Flare or Sparks preferences.

FileStorePath

When large files are sent from Flare and Spark to the CEBroker the files are cached on the CEBroker and sent to the FieldEngines when needed. If the CEBroker and FieldEngine have access to the same file system, then instead of sending the files to the FieldEngines via the network the FieldEngines can load the files from the file system. This reduces network traffic and increases performance. The FileStorePath can be set to an absolute path to a directory to enable this.

Please contact Cresset support (support@cresset-group.com) for assistance in configuring CEBroker to your computing facilities.

Cresset desktop and command line applications can be configured to contact the broker using the machine name and port in the ‘Processing’ preference settings (for GUIs) or using the ‘-g’ switch (command lines). Alternatively, the environment variable ‘`CRESSET_BROKER`’ can be used to specify the machine and port using the format `CRESSET_BROKER=hostname:port`.

Uninstalling the CEBroker

1. To uninstall the Cresset Engine Broker if installed from RPM, do

```
rpm -e cresset-cebroker2
```

2. To uninstall if installed from a `tar.xz` file, just delete the CEBroker directory and remove any `init` scripts or system unit files that you installed to run it as a service.

Changes in version 3.0.1

Fixed bug in the SLURM_engine.sh example script.

Changes in version 3.0

Version 3.0 of the Cresset Engine Broker is only compatible with Flare 6 and Spark 10.7.

- Re-designed architecture
- Simplified configuration
- Added encryption support using SSL certificates
- Local file caching, enabling a major speedup for docking calculations in Flare and Spark
- Reduced CPU usage, with the Broker now making use of all available CPU cores
- Reduced open file handle usage
- Only a single open port is required instead of requiring a range of ports
- Improved scaling, now supporting calculations on thousands of computing resources (CPUs/GPUs)

Changes in version 2.6

- Implemented stricter queuing system in support of FEP calculations
- Improved performance when working with low-bandwidth internet connections
- Cresset Engine Broker now prints out errors from start_engine_scripts to stderr to facilitate troubleshooting
- Fixed issue which caused the Cresset Engine Broker to mishandle GPU tasks during an FEP run

Changes in version 2.5

- Updated the following example scripts to support queuing systems which use the CUDA_VISIBLE_DEVICES environment variable to control which GPU a FieldEngine runs a calculation on.
    - start_LSF_engine.sh
    - start_PBS_engine.sh
    - start_SGE_engine.sh
    - start_SLURM_engine.sh
- Fixed rare crash when a connection between a FieldEngine and the CEBroker is unexpectedly lost.

Changes in version 2.4.2

- Improved performance and behavior of the CEBroker when used to run ligand-based alignment, ligand minimization and docking experiments at a 10K scale from Flare
- Improved performance when working on low bandwidth internet connections
- Fixed issue which caused the CEBroker fail to detect that a FieldEngine had shut down
- Fixed bug which on rare occasions caused a critical error on Flare.

Changes in version 2.4

- New queueing system, handling multiple users connecting to the broker and competing for available resources
- Multiple start_XXX_engine.sh scripts are now available to support different queueing systems
- Improved handling of hostname resolution and subnet choice
- Fixed bug which caused manually connected FieldEngines to become non-contactable after a period of time

Changes in version 2.3

- New ‘keepalive’ messages to correctly track connected Field Engine instances

- New feature to abort any running jobs when losing the connection to the network, leaving the Field Engines connected and available
- Updated SysV and system scripts to honor the -g parameter to use GPUs
- Fixed bug which caused the CEBroker to silently fail when trying to connect to Field Engines whose name could not be resolved.

Changes in version 2.2

- Improved system integration for RHEL 7
- Fixed some possible deadlocks with network sockets
- Improved and more informative logging
- Updated to use Qt 5.9: fixes some network issues
- The username is now passed to the broker and can be passed on to the start-engines script. This allows computational resources started by the broker to be assigned to the user requesting them, facilitating cost management
- Fixed bugs that occurred if an obsolete FieldEngine tries to connect to the broker
- Fixed problems that occurred when using CEBroker without a start-engines script
- Fixed DNS issues with using FieldEngines on the Amazon cloud.

Changes in version 2.1

- Fixed several threading bugs.

Changes in version 2.0

- Version 2 represented a major change in architecture. In version 1, the CEBroker referred FieldEngine locations to the application, and subsequent communication happened directly between the application and the FieldEngine. This caused issues in cases where the FieldEngines were running in a disconnected network and could not be contacted directly by the application. In version 2 the CEBroker acts as a relay: the application communicates solely with the CEBroker, and the CEBroker communicates with the FieldEngines.
- Version 2 is not backwards compatible with Version 1. Early application versions (e.g. Spark and Forge with version number earlier than 10.4) cannot use version 2.

Version 1.0

- Initial release.

Known Issues and Bugs

There are no known bugs or issues.

Found a bug or problem that is not listed? Please contact us and help improve our software!

Support

Cresset
New Cambridge House
Bassingbourn Road
Litlington
Cambridgeshire, SG8 0SS, UK

+44 (0)1223 858890
support@cresset-group.com
www.cresset-group.com
Twitter: @cressetgroup
LinkedIn: Cresset group