EUROPEAN

MIDDLEWARE INITIATIVE

APEL CLIENT USER GUIDE

Document Version:	**2.2**
EMI Component Version:	**4.1.0**
Date:	**01.07.2013**

DOCUMENT CHANGE LOG

Version	Date	Comment	Author
2.2	01/07/2013	Update to include file permission information for /etc/apel/parser.cfg and /etc/apel/client.cfg	Alison Packer
2.1	25/04/2013	Update to include SLURM parser and new configuration.	Will Rogers
2.0	13/02/2013	Rewritten for new implementation of APEL publisher in EMI3.	Will Rogers
1.0.2	18/05/2012	Reviewed, added funding notice	Will Rogers
1.0.1	12/09/2011	Updated	Cristina del Cano Novales
1.0.0	07/04/2011	First version	Cristina del Cano Novales

Contents

1. INTRODUCTION

APEL (Accounting Processor for Event Logs) is a tool that gathers CPU accounting information by parsing batch system and blah accounting log files and inserting the data into a local MySQL database. APEL then publishes the data into a centralized repository.

2. ARCHITECTURE

The APEL Client consists of two main components, the APEL parser and the APEL publisher (Figure 1). More detailed information about each component's architecture can be found in the following sections.
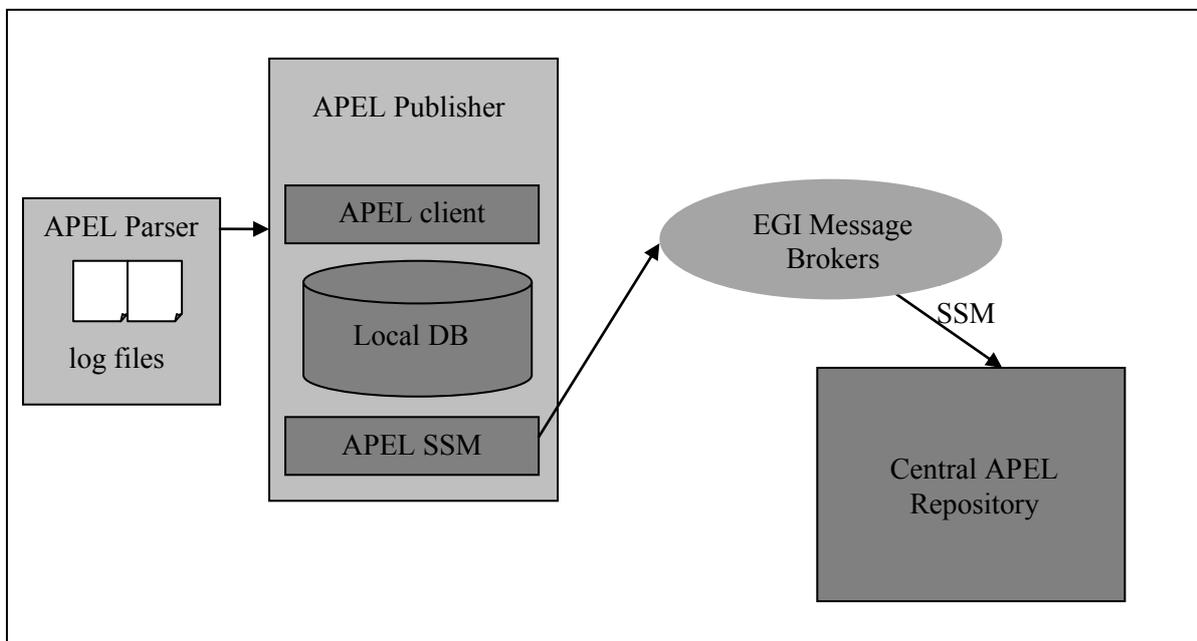
Figure 1: Overall Architecture

3. APEL PARSER

3.1. DESCRIPTION

To create CPU accounting information from a grid site, APEL requires information from three sources:

- Batch system log files
- CE accounting log files (also known as blah logs)
- CPU benchmarking information, available from LDAP.

The APEL parser is used to retrieve the first two of these.

Currently, the APEL parser can support four types of batch system: PBS/Torque, LSF, Grid Engine, and SLURM. The APEL parsers package is able to parse log files from any of these batch systems. Accounting records are created by the APEL parser from the batch system logs and blah logs and these accounting records are stored in a MySQL database, installed on an APEL Publisher node.

Any log files which have changed since the last operation of the parser will be parsed to check for new records.

3.2. DEPLOYMENT

The APEL Parser can be deployed in a variety of ways depending upon the setup at your site.

APEL Accounting records will only be created when the accounting log files and the batch system log files are parsed. Sites must therefore ensure the APEL parser has access to both types of log file, either by installing an instance of the APEL parser on each machine containing logs, or by making the logs available to the machine where the APEL parser is installed.

An overview of some of the typical deployments is as follows:

3.2.1 SEPARATE LRMS/CE

An instance of the APEL parser must be installed in both the CE and the LRMS. The parser must be configured to read the Event Log Files (from the batch system) in the CE installation. The LRMS APEL parser must be configured to read the Accounting Log Files.
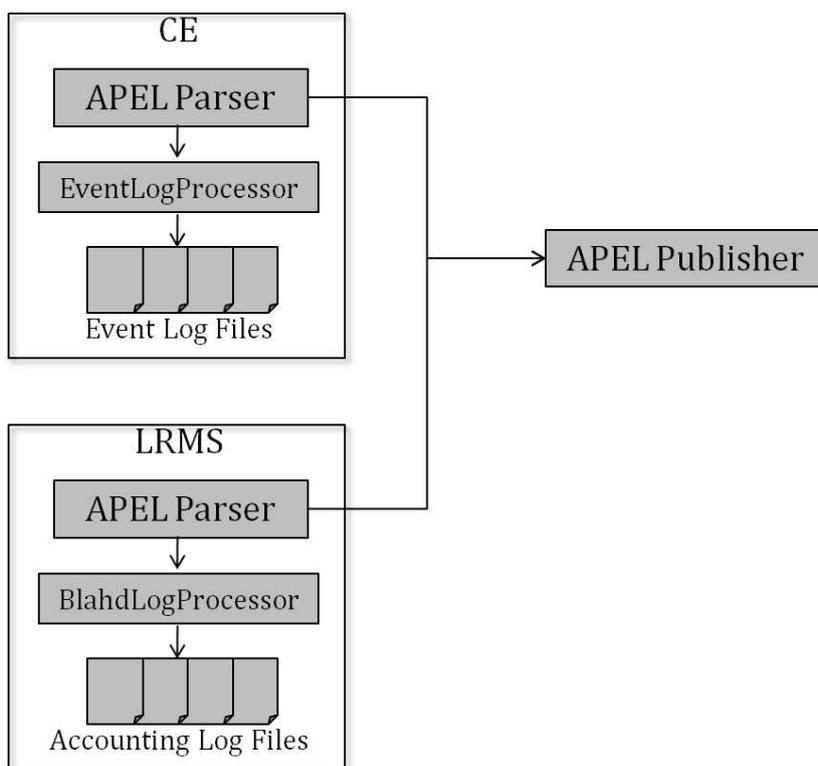
Figure 2: Separate LRMS/CE

3.2.2 COMBINED LRMS/CE

Only one instance of the APEL parser is needed in the CE/LRMS. The parser must be configured to parse both the batch system log files and the accounting log files.
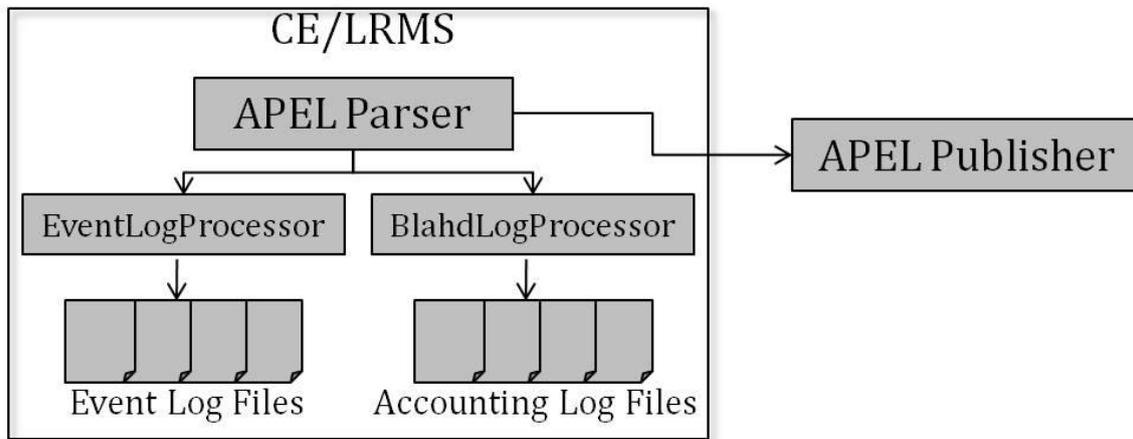
Figure 3: Combined LRMS/CE

3.2.3 MULTIPLE CES
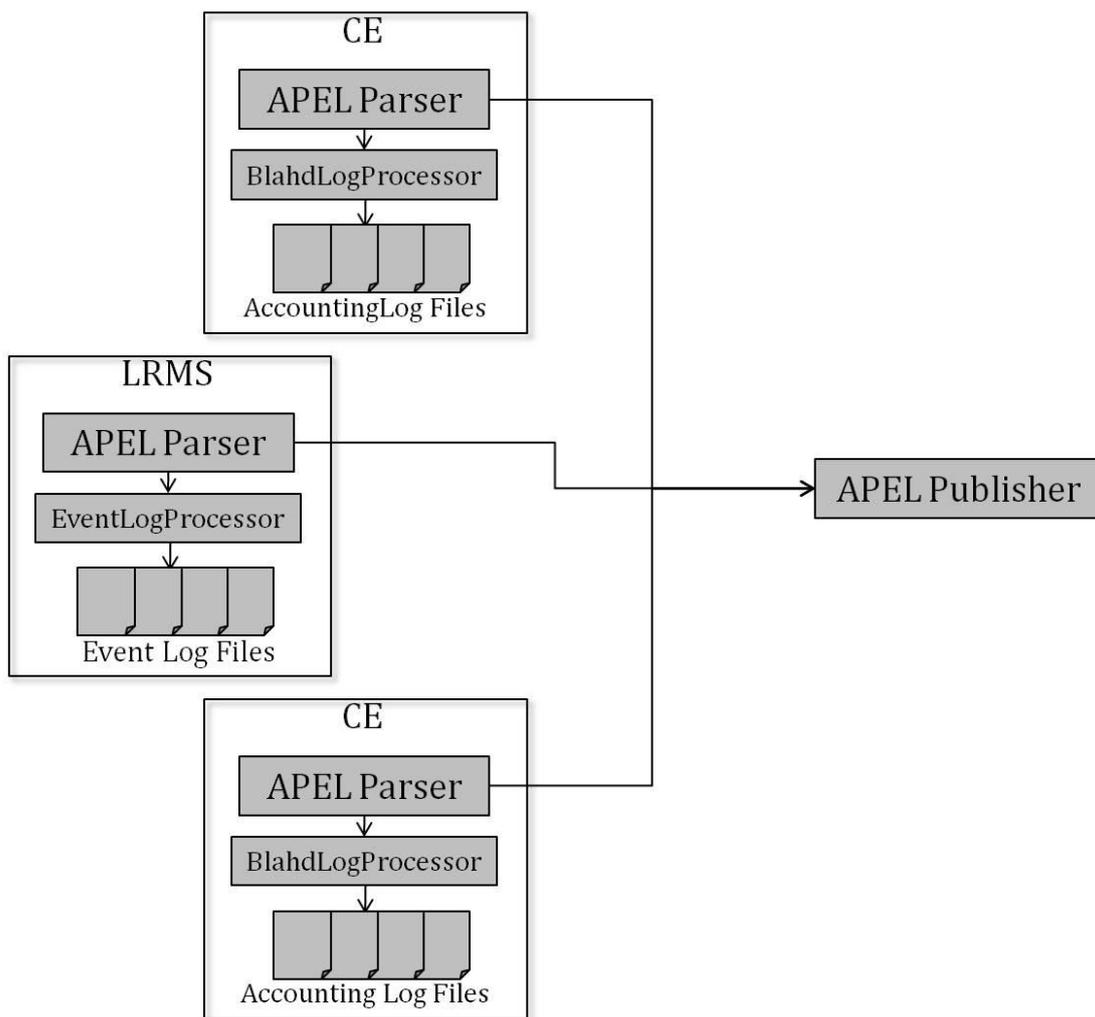
Figure 4: Multiple CEs

3.3. INSTALLATION

To run the APEL parser the following packages should be installed (they are available for SL5 or SL6):

```
apel-lib
apel-parsers
```

3.4. CONFIGURATION

(For more detailed installation information please see the APEL Parsers System Administrator Guide.)

The APEL parser configuration file is located at /etc/apel/parser.cfg. You will include database username and password information in this file so ensure it is not world readable, e.g. by running:

```
chmod 600 /etc/apel/parser.cfg
```

It has two main tasks:

- Parses blah log files

- Parses batch system log files

Either of these may be disabled.

Each section of the configuration file is described below.

3.4.1 [db]

This section contains the APEL client database connection information.

```
# Information about the apel client database
[db]
hostname = localhost
port = 3306
name = apelclient
username = root
password =
```

3.4.2 [site_info]

```
[site_info]
# Specify the site name from GOCDB
site_name =
# Batch system hostname.  This does not need to be a definitive hostname,
# but it should uniquely identify the batch system.
# Example: pbs.gridpp.rl.ac.uk
lrms_server =
```

This contains two options:

- `site_name`: this must be the site name which is specified in GOCDB
- `lrms_server`: this must uniquely define the batch system being used. It does not have to be a real hostname, but it should not be the same for two different batch systems.

3.4.3 [blah]

This section contains information about the blah log files.

```
[blah]
enabled = true
dir = /var/log/cream/accounting
# typical logfile name is blahp.log-yyyymmdd
filename_prefix = blahp.log
subdirs = false
```

- `enabled:` allows you to enable or disable the blah parser
- `dir:` specifies the directory to search for blah logs
- `filename_prefix:` specifies the prefix that each file must have in order to be parsed
- `subdirs:` whether to recursively search subdirectories of the specified directory for log files

3.4.4 [batch]

This section contains information about the batch system log files.

```
[batch]
enabled = true
```

```
# Reparse all batch system log files and replace them in the database.
# Use with caution!

reparse = false

# Batch system specific options.
# Valid types are LSF, PBS, SGE, SLURM
type =
# Whether to try to parse multi-core details
parallel = false
# Directory to search for accounting logfiles
dir =
# Prefix shared by all log files to parse
filename_prefix =
# Whether to search subdirectories for logfiles
subdirs = false

# LSF only: scale CPU and wall durations according to
# 'HostFactor' value in logfiles
#scale_host_factor = false
```

- `enabled`: allows you to enable or disable the blah parser
- reparse: allows you to reparse all the batch system files and replace them in the database.
  - This will cause a lot of new data to be generated. Use this option only if you know what you're doing!
- `type`: must be PBS, LSF, SGE or SLURM
- `parallel`: whether to include information about multiple nodes and cores used by a job
- `dir`: specifies the directory to search for batch system logs
- `filename_prefix`: specifies the prefix that each log file must have in order to be parsed
- `subdirs`: specifies whether to search subdirectories of the specified directory recursively for log files
- `scale_host_factor` (LSF only): LSF can be configured to publish a host scaling factor in the accounting log file. This option will scale the CPU and wall durations according to these values.

3.4.5 [logging]

```
[logging]
logfile = /var/log/apelparser.log
level = INFO
console = true
```

This controls the logging options:

- `logfile`: specifies the location of the log file to write
- `level`: can be DEBUG, INFO, WARNING, ERROR, CRITICAL. It is recommended to leave this as INFO by default.
- `console`: whether to log to the console as well as to the log file.

3.5. LOGGING

The default log file for the APEL parser is `/var/log/apelparser.log`.

3.6. RUNNING THE PARSER

To run the parser using the default configuration, run:

```
# /usr/bin/apelparser
```

4. APEL PUBLISHER

4.1. DESCRIPTION

The APEL Publisher:

- Retrieves benchmarking information from the BDII;
- Joins the EventRecords and BlahdRecords inserted in the database by the APEL parser to create JobRecords;
- Optionally creates Summary records from the JobRecords;
- Writes messages containing accounting data (either JobRecords OR Summary records) to the filesystem;
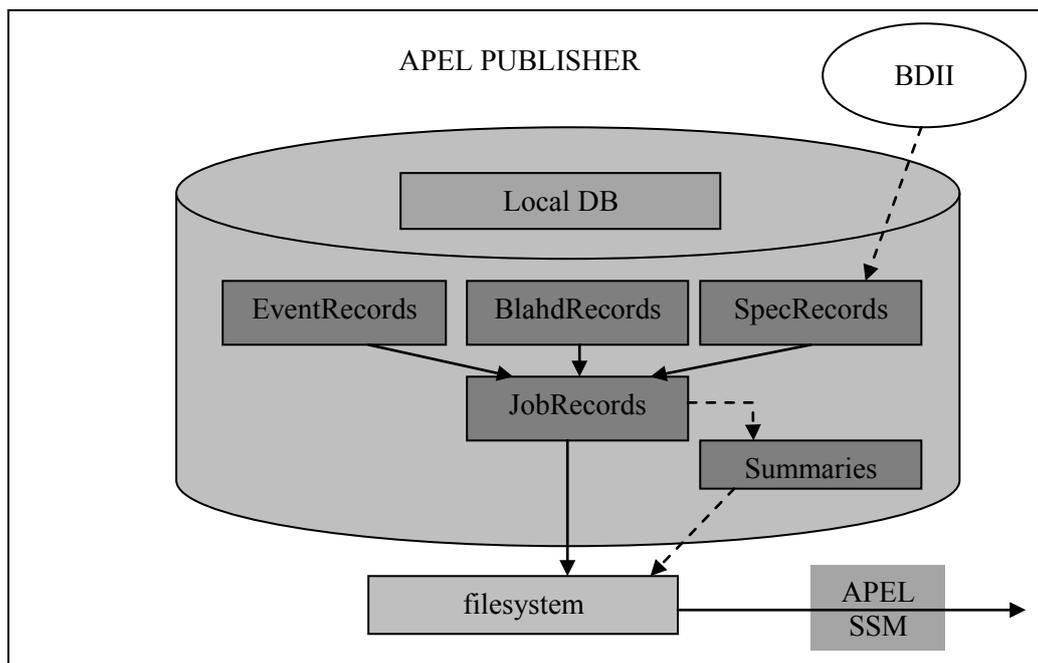- Runs SSM to send the messages to the central APEL repository.

Figure 5: APEL Publisher

4.2. DEPLOYMENT

The APEL Publisher is deployed by installing the following packages (they are available for SL5 or SL6):

```
apel-ssm

apel-lib
```

4.3. CONFIGURATION

(For more detailed installation information please see the APEL Publisher System Administrator Guide.)

The APEL publisher has two configuration files, one for apel-client and one for apel-ssm. A MySQL database must also be installed and the client schema loaded.

The APEL client database schema is located in `/usr/share/apel/client.sql`

The APEL client configuration file is located in `/etc/apel/client.cfg`

The SSM configuration file for sending messages is located at `/etc/apel/sender.cfg`.

4.3.1 APEL Client Database

Install MySQL and load the APEL client database schema.

4.3.2 APEL Client

Each section of the APEL Client configuration file is described below. You will include database username and password information in this file so ensure it is not world readable, e.g. by running:

```
chmod 600 /etc/apel/client.cfg
```

4.3.2.1 [db]

This section contains the APEL client database connection information.

```
# Information about the apel client database
[db]
hostname = localhost
port = 3306
name = apelclient
username = root
password =
```

4.3.2.2 [spec_updater]

This section contains information about the ldap server containing benchmarking information.

```
[spec_updater]
enabled = true
# The GOCDB site name
site_name =
ldap_host = lcg-bdii.cern.ch
ldap_port = 2170

## The following information is necessary for
## local jobs only.  Provide an average value for
## machines which your batch system uses.
## Supported specs are HEPSPEC and Si2k
## Provide also a hostname for your batch system.
## This does not need to be a definitive hostname,
## but it should uniquely identify the batch system.
## Example: pbs.gridpp.rl.ac.uk
```

```
#lrms_server =
#spec_type = HEPSPEC
#spec_value = 1.0
```

- `enabled:` allows you to enable or disable the blah parser
- `dir:` specifies the directory to search for blah logs
- `filename_prefix:` specifies the prefix that each file must have in order to be parsed
- `subdirs:` whether to search subdirectories of the specified directory recursively for log files

4.3.2.3 [joiner]

This section contains information about how to create the accounting data.

```
[joiner]
enabled = true
local_jobs = false
```

- `enabled:` allows you to enable or disable the blah parser
- `type:` must be PBS, LSF or SGE
- `parallel:` whether to include information about multiple nodes and cores used by a job
- `dir:` specifies the directory to search for batch system logs
- `filename_prefix:` specifies the prefix that each log file must have in order to be parsed
- `subdirs:` whether to search subdirectories of the specified directory recursively for log files

4.3.2.4 [unloader]

This section contains configuration of which data to send to the APEL server.

```
[unloader]
enabled = true
dir_location = /var/spool/apel/

# You may send only summaries of your data to the APEL server,
# rather than individual job records.
# This reduces the network load.
send_summaries = false

# You may send 'withheld' instead of the user's DN in the
# GlobalUserName field.  This is only valid for individual
# job records.
withhold_dns = false

# Optional: send ONLY these VOs to the APEL server.
# This overrides exclude_vos.
#include_vos = atlas,cms,lhcb

# Optional: do not send these VOs to the APEL server.
# This does not take effect if include_vos is set.
#exclude_vos = atlas,cms

# Which records to send:
# latest - just send the new records to the server
# gap    - send records from between the specified dates (inclusive)
#          this is only for individual job records
# all    - send all records to the server.  Don't do this for individual
#          job records without talking to the apel team!
interval = latest
## only used if interval = gap
```

```
#gap_start = 2012-01-01
#gap_end = 2012-01-31

# Send CAR-format records - only for job records
send_ur = false
```

- `enabled:` allows you to enable or disable the unloader
- `dir_location:` leave this value unchanged
- `send_summaries:` send processed data instead of each individual job record. This is recommended for large sites.
- `withhold_dns:` do not send user DNs as part of the data
- `include_vos` and `exclude_vos`: specifies if only certain VOs should be sent to the server
- `interval:` controls which dates records are sent from
- `send_ur:` allows sending of records in CAR format. This is only useful for compatibility with other accounting servers. If sending to the APEL server, leave this as false.

4.3.2.5 [ssm]

This section allows you to enable sending the data from the filesystem to the central APEL server.

```
[ssm]
enabled = true
```

For SSM configuration update the file `/etc/apel/sender.cfg` – see section 6.4

4.3.2.6 [logging]

This section controls the logging options:

```
enabled = true
[logging]
```

```
logfile = /var/log/apel/client.log
level = INFO
console = true
```

- logfile: specifies the location of the log file to write
- level: can be DEBUG, INFO, WARNING, ERROR, CRITICAL. It is recommended to leave this as INFO by default.
- console: whether to log to the console as well as to the log file.

4.3.3 Logging

The log file for the APEL client can be found in `/var/log/apel/client.log`

4.3.4 APEL SSM Configuration

The SSM configuration file for sending messages is located at `/etc/apel/sender.cfg`.

Each section of the SSM configuration file is described below:

4.3.4.1 [broker]

The default configuration will find the production EGI message brokers.

4.3.4.2 [certificates]

This specifies

- the location of the host certificate
- the location of the host key
- the location of the trusted CA certificates directory

If you wish messages to be explicitly encrypted in transit, you may specify the APEL server's certificate using the server option.

4.3.4.3 [messaging]

The correct queue must be specified. The path is the location on the file system in which the sending SSM will look for messages.

4.3.4.4 [logging]

This section controls the logging options:

```
enabled = true
[logging]

logfile = /var/log/apel/sender.log
level = INFO
console = true
```

- logfile: specifies the location of the log file to write
- level: can be DEBUG, INFO, WARNING, ERROR, CRITICAL. It is recommended to leave this as INFO by default.
- console: whether to log to the console as well as to the log file.

4.3.5 Logging

The log file for the sending SSM can be found in `/var/log/apel/sender.log`

4.4. RUNNING THE PUBLISHER

The script `/usr/bin/apelclient` will run the publisher using the configurations (client and SSM) specified above. To run the APEL publisher once per day, set up a cron job.