

EGEE

R-GMA Flexible Archiver

Document identifier:	EGEE-JRA1-TEC-flexy-manual
Date:	May 11, 2006
Activity:	JRA1: Middleware Engineering and Integration (UK Cluster)
Document status:	Released
Document link:	

Abstract: This document describes the R-GMA flexible archiver

1 INTRODUCTION

The flexible archiver allows users to setup an RGMA secondary producer for a set of user configurable tables. Reasons for archiving tables in this way include:

- To allow the use of latest queries on data that is published by continuous primary producers
- To allow queries that involve joining tables published by different primary producers

The flexible archiver allows each table to be archived for a different length of time so that quickly and slowly changing information can be archived appropriately.

2 CONFIGURATION

2.1 CONFIGURATION FILE

The flexible archiver reads its configuration from a file using the format:

```
<property>=<value>
```

Lines beginning with # are comments and are ignored.

2.1.1 CONFIGURATION PARAMETERS

The following parameters are recognized in the configuration file:

Type	<p>The type of consumer queries the producer supports. Possible values are:</p> <p>Latest History</p>
DBName	<p>This optional property is used to define the database name to be used. It is defined as a JDBC URL in the form <code>jdbc:mysql://localhost/pudding</code>. If this property is undefined, the RGMA server will create a database dynamically for you. However, when the secondary producer is terminated, the database will be deleted (losing any archived material). To keep data persistent you'll want to create your own database and then use this property to define its name. See 2.2 for more help.</p>
DBUsername	Database username
DBPassword	Database password
HistoryRetentionPeriod	Defines a time value after which tuples will be deleted from the archiver, defined in minutes.
Tables	Table names to archive, defined as a space separated list of tables.
x_HistoryRetentionPeriod	Retention period for table x, over-rides the value defined by HistoryRetentionPeriod

2.1.2 EXAMPLES

An example configuration file can be found under `$RGMA_HOME/etc/rgma-flexible-archiver/`.

2.2 DATABASE SETUP

The R-GMA flexible archiver requires a MySQL database with sufficient user privileges to store archived data. Your RGMA server already runs a MySQL server so all you need is access to set one up. To simplify this process, a script is available to do this for you

```
$RGMA_HOME/bin/rgma-flexible-archiver-db-setup
```

This script dumps a series of SQL statements to stdout. To install the database, you'll need mysql root access to perform the following:

```
mysql -u root -p < my-sql-file
```

Where my-sql-file contains output from running the rgma-flexible-archiver-db-setup script (of course, this assumes you're running the script on the same host as the MySQL server). Alternatively, you can just pipe the output into mysql, for example:

```
rgma-flexible-archiver-db-setup -n test -u me -p password | mysql -u root -p
```

3 RUNNING THE FLEXIBLE ARCHIVER

To create your flexible archiver:

```
rgma-flexible-archiver --start --config my.config --output my-id-file
```

Where `--config` defines the config file to create your secondary producer (see configuration file). The `--output` option defines a file where the secondary producer id will be stored.

Once started, the secondary producer will be created on your RGMA server. You'll need to ping the secondary producer on a regular basis to ensure the resource persists on the server see 3.1.

To ping the producer:

```
rgma-flexible-archiver --ping --input my-id-file
```

The `--input` defines the archiver id file. The id within the file is used to identify the secondary producer on the RGMA server.

To terminate your secondary producer:

```
rgma-flexible-archiver --stop --input my-id-file
```

Again the `--input` defines the secondary producer id file. The id within the file is used to identify the secondary producer on the RGMA server.

3.1 KEEPING IT ALIVE

Once your flexible archiver is running, you'll want to keep it that way. To do so, you'll need to routinely ping your archiver on a regular basis. The recommended approach is to run a cron script at regular intervals of 60 minutes. If you schedule cron to perform a ping using longer intervals, you will lose your flexible archiver altogether.

To guard against R-GMA server outages, you should add extra scripting in your cron job to check the return status of the ping. The following example will create a new flexible archiver if the ping attempt has failed.

```
#!/bin/sh

$RGMA_HOME/bin/rgma-flexible-archiver --ping --input my-id-file
if [ $? != 0 ]
then
    rm -f my-id-file
    $RGMA_HOME/bin/rgma-flexible-archiver --start --config \
my.config --output my-id-file
    exit $?
fi
```