EGEE

R-GMA Gin

Document identifier:	**EGEE-JRA1-TEC-gin-manual**
Date:	**October 27, 2006**
Activity:	**JRA1: Middleware Engineering and Integration (UK Cluster)**
Document status:	**DRAFT**
Document link:	

Abstract: This document describes the R-GMA Gin tool

1 INTRODUCTION

Gin is a service to read information in LDIF format and publish it to R-GMA. It allows flexible mapping between LDIF attributes and R-GMA tables/columns.

1.1 RUNNING GIN

To start the Gin service, use:

```
/etc/init.d/rgma-gin (start|stop|status)
```

To run Gin directly from the command line, use:

```
$RGMA_HOME/bin/rgma-gin
```

1.2 CONFIGURATION

Gin has two configuration files:

```
$RGMA_HOME/etc/rgma-gin/gin.conf
$RGMA_HOME/etc/rgma-gin/security.conf
```

`gin.conf` defines the source of LDIF data and how it maps on to R-GMA tables and columns. `security.conf` defines the certificate/proxy used by Gin to authenticate itself to the R-GMA server.

These files can be edited manually to configure Gin – Their formats are described in section 2. However, Gin also comes with a setup tool `rgma-gin-config` which allows easy setup based on three predefined Gin configurations:

- `gip` The Generic Information Provider - publishes to Glue schema tables.

- `fmon` Publishes to Glue schema tables using the FMON program.

- `glite-ce` Information provider for a Glite Computing element - publishes to Glue schema tables.

To setup Gin using `rgma-gin-config`, use the following command:

```
$RGMA_HOME/bin/rgma-gin-config --gip-ce=yes|no --gip-se=yes|no \
                               --gip-service=yes|no --fmon=yes|no \
                               --glite-ce=yes|no
```

This will set up Gin to use the specified information providers. The setup script will prompt you to enter information about the certificates Gin will use to authenticate itself. When complete, a new configuration file will be written (overwriting any previous file).

To configure the security options on the command line, the `--cert, --key, --proxy, --insecure, --trusted-cert-dir` options are available.

1.2.1 EXAMPLES

To set up Gin to use the GlueCE tables with the generic information provider and no authentication:

```
$RGMA_HOME/bin/rgma-gin-config --gip-ce=yes --insecure
```

To set up Gin to use the GlueCE tables with the generic information provider and authentication using a certificate/key pair:

```
$RGMA_HOME/bin/rgma-gin-config --gip-ce=yes --cert=certificate.pem \
                        --key=privatekey.pem
```

To set up Gin to use the fmon information provider and authentication using a Grid proxy:

```
$RGMA_HOME/bin/rgma-gin-config --fmon=yes --proxy=/tmp/proxy.pem
```

To set up Gin to use the glite-ce information provider and authentication using a Grid proxy, with a specified directory containing trusted certificates:

```
$RGMA_HOME/bin/rgma-gin-config --glite-ce=yes --proxy=/tmp/proxy.pem \
                        --trusted-cert-dir=/etc/certificates/
```

A full description of `rgma-gin-config` and its options is available in section 2.3

1.3 LOGGING

The Gin log file is:

```
/var/log/glite/rgma-gin.log
```

Logging is controlled by the log4j properties file:

```
$RGMA_HOME/etc/rgma-gin/log4j.properties
```

2 REFERENCE GUIDE

2.1 GENERAL CONFIGURATION

The `gin.conf` file describes how Gin should obtain the LDIF data, how it should be mapped to R-GMA tables and columns, and how often the publication should take place.

The basic format of the file is:

```
infoprovider <name>

  # Table definitions

  # Information provider options

endinfoprovider

# More info provider definitions
```

Comments are allowed and are specified using the # character. Keywords are case insensitive.

2.1.1 TABLE DEFINITIONS

The most general table definition is as follows:

```
table <table>[=<objectclass>]
  column <column>[=<attribute>]+
  [multivalued <column>[=<attribute>]]
  table <link table>
    [column <column>[=<attribute>]]+
    [multivalued <column>[=<attribute>]]
  endtable
endtable
```

`<table>` gives the name of an R-GMA table we wish to publish to. This table will not be created by Gin - you must ensure that a suitable table already exists or create one using another tool.

By default, a table will match an LDIF object class which has the same name. However, if `<objectclass>` is specified, this object class will be matched to the specified R-GMA table.

Where an LDIF attribute block refers to several LDIF object classes, the longest named object which matches an R-GMA table will be chosen to process the attributes.

A table column is specified using the `column` keyword. The name of the column in the R-GMA table is `<column>` . By default this column will be mapped onto the LDIF attribute with name `<objectclass><column>` , e.g. if you had the following table definition:

```
table Service=LdifService
  column Id
endtable
```

The `Id` column will be mapped onto the LDIF attribute `LdifServiceId` .

If you want an R-GMA column to map onto a different LDIF attribute, this can be specified using the `<attribute>` option.

It is also possible to specify more complex column mappings. For example to populate an R-GMA column with the concatentated values of two LDIF attributes, you can specify `<attribute1>+<attribute2>` . Any number of attributes may be joined in this way, for example:

```
table Service
  column Name=Type+Endpoint
endtable
```

will populate the Name column with the contatentated values of the LDIF attributes ServiceType and ServiceEndpoint.

You can also separate the concatenated values with a colon or slash using the following syntax.

```
table File
  column Endpoint=Host+:+Port+/+Path
endtable
```

You can also specify a - after an attribute name. This will remove any string from the start of the attribute up to =.

```
table Site
  Name=Info-
endtable
```

e.g. if `SiteInfo` LDIF attribute is `name=cern.ch`, then this will set the value to `cern.ch`.

The - modifier can be used to remove several sets of `name=value` pairs. The first - removes up to '=' inclusive, the second up to ',' inclusive, the third the next '=' the fourth the next ',' and so on.

```
 table Site
   Description=Info---
 endtable
```

Now if SiteInfo is `name=cern.ch, description=CERN`, it will be converted to `CERN`.

You can combine both of the above, e.g.

```
table Site
  FullName=Info-+:+Location
endtable
```

This will apply one '-' edit to the value of `SiteInfo` to remove a prefix, append a '/' to it and then append the value of `SiteLocation`.

If a particular column may have multiple values in an LDIF attribute block, you can specify it using the `multivalued` keyword. The syntax is identical to that for specifying a column, but for each LDIF attribute block a separate tuple will be published for each occurance of the multivalued attribute.

A multivalued column must form part of the primary key for the R-GMA table being published. In addition, you may only specify one multivalued column per table. Normally this is not the best way to map a multivalued attribute onto a relational table. Instead Gin allows you to define linked tables to publish multivalued attributes.

2.1.2 LINKED TABLES

In a relational system such as R-GMA, one-to-many relationships are generally modelled using linked tables. For example if you have a `Service` table and a single service may have multiple owners, a typical approach would be to create a `ServiceOwner` table which contains columns matching the primary key of the `Service` table and also an additional column in its primary key containing the `Owner` field.

This kind of mapping can be handled by Gin using linked tables. To define a linked table, you embed a `table` definition inside another table definition, e.g.

```
table Service
  column Name
  table ServiceOwner
    multivalued Owner
  endtable
endtable
```

Linked tables are published in the same way as ordinary tables and process the same attribute blocks as their parent tables. They also inherit the attribute mappings of the parent table with the columns prefixed by the table name.

For example in the definition above, the `Service` table is assumed to have a `Name` column which is populated by the LDIF attribute `ServiceName` . The `ServiceOwner` table will populate it's `Owner` column with the multiple values of the LDIF `ServiceOwner` attribute. It will also populate it's `ServiceName` column (if it exists) with the value of the LDIF attribute `ServiceName`. In this way, link tables can easily be defined to publish multivalued LDIF attributes.

In some cases the above syntax may not be sufficient to describe the mapping. For example, suppose the `ServiceOwner` table instead had columns `Service_Name` and `Value` . This could be mapped using the following table definition.

```
table Service
  column Name
  table ServiceOwner
    multivalued Value=ServiceOwner
    column Service_Name=ServiceName
  endtable
endtable
```

The automatic inherited mapping to the `ServiceName` column would be ignored in this case since it does not match a column in the link table.

Note that you cannot specify an object mapping for a linked table – it takes data from attribute blocks that match the parent table. Also, you can only define one level of linking – link tables cannot contain other link tables.

If an attribute specified in a table mapping is not found, it is not published. Furthermore, If the R-GMA table contains columns which are not specified in the table mapping these columns will not be published (i.e. will be set to their default values or `NULL`). This enables Gin to support backwards-compatible changes to the R-GMA and LDIF schemas, such as addition of table columns.

2.1.3 INFORMATION PROVIDER OPTIONS

```
run <program to run to get information>
```

Specifies the script (or command) to generate LDIF data. More than one command can be specified using multiple run keywords (on seperate lines) – the output from each command will be processed and RGMA updated in turn.

```
staticinformation <table> \"<sql INSERT command>\"
```

`table` will be updated with specified command at the slow rate, i.e. as if it is a `slowtable`. **N.B. The table name appears twice in the line.**

Example:

```
staticinformation info \"INSERT INTO info (NAME, ADDRESS) VALUES ('ahost.com', '1.2.3.4')\"
```

```
delay <number of seconds>
```

Mandatory keyword specifies the number of seconds which must elapse between updates.

```
slowtable <tablename>
```

Specified a table that should be published less frequently than other tables. See `slowcounter` option for more details.

```
slowcounter <count>
```

Number of update iterations to skip in between updates to slow tables. If not present, slowtables are updated with the same frequencey as normal tables

```
foreignkeyattribute <name>
```

Specified the name of an LDIF attribute which marks a foreign key.

This is a special case in which the name of an LDIF attribute indicates that its value refers to a foreign key. The attribute value is then of the form `key=value` in which the key and value are the usual attribute name / attribute value pair. This is used in the Glue mapping with foreign keys marked using the `GlueForeignKey` attribute.

R-GMA does not need this flag since keys are defined in its own schema. However, if this option is set Gin will discard the special foreign key marker and interpret the value as a normal attribute key/value pair.

If this option is not present, no special foreign key interpretation is performed.

2.2 SECURITY CONFIGURATION

The `security.conf` file is a `name=value` properties file defining the authentication credentials used by Gin. Comments are allowed using the `#` character. The following properties are recognized:

Property	Description
sslCertFile	X509 certificate used to authenticate Gin to the R-GMA server
sslKey	X509 private key used to authenticate Gin to the R-GMA server
sslKeyPasswd	Password for private key specified with `sslKey`
gridProxyFile	File containing X509 proxy certificate used to authenticate Gin to the R-GMA server
sslCAFiles	Directory containing certificates of trusted Certificate Authorities
crlFiles	Directory containing Certificate Revocation Lists for trusted certificate authorities

Either a certificate/key pair or a proxy may be used for authentication. If both are specified, the proxy takes precedence.

2.3 SETUP SCRIPT OPTIONS

The `rgma-gin-config` setup script has the following command line options:

Property	Description	Default
`--help`	Display usage information	n/a
`--version`	Display Gin version	n/a
`--gip`	Whether to use the Generic Information Provider. NB this is deprecated, please use the gip-* options	no
`--gip-ce`	Whether to use the Generic Information Provider with the GlueCE tables	no
`--gip-se`	Whether to use the Generic Information Provider with the GlueSE tables	no
`--gip-service`	Whether to use the Generic Information Provider with the GlueService tables	no
`--fmon`	Whether to use the FMON Information Provider	no
`--glite-ce`	Whether to use the Glite CE Information Provider	no
`--insecure`	If specified, assume that Gin does not need to be authenticated to the R-GMA server	Authentication enabled
`--cert`	Path to file containing X509 certificate to use when authenticating Gin to the R-GMA server	`/etc/grid-security/hostcert.pem`
`--key`	Path to file containing private key for the certificate defined using `--cert`.	`/etc/grid-security/hostkey.pem`
`--proxy`	Path to file containing Grid proxy to use when authenticating Gin to the R-GMA server	None – certificate/key pair used by default
`--trusted-cert-dir`	Path to directory containing X509 certificates of trusted certificate authorities	`/etc/grid-security/certificates/`
`--silent`	If specified, produce no output and use default values for unspecified parameters	Interactive mode