Bristol OpenEnterprise Reference Guide

Event Replication
Contents

1 Event Replication........................................................................................................................................... 3
  1.1 Event Replication Overview .................................................................................................................... 3
  1.1.1 Real-time Event Replication .............................................................................................................. 3
  1.1.2 Back-fill Event Replication ................................................................................................................ 3
  1.2 User Interface........................................................................................................................................... 3
  1.3 Configuration.......................................................................................................................................... 4
  1.3.1 Event Replication Configuration Overview ..................................................................................... 4
  1.3.2 Database Configuration....................................................................................................................... 4
    1.3.2.1 Note ........................................................................................................................................ 5
  1.3.3 Latency .............................................................................................................................................. 6
  1.3.4 Command Line Configuration............................................................................................................ 6
  1.4 Example Configuration............................................................................................................................... 6
  1.4.1 Example System................................................................................................................................... 6
  1.4.2 Example Solution................................................................................................................................ 8
    1.4.2.1 Requirement 1 ............................................................................................................................ 8
      1.4.2.1.1 West Servers to collect signal values directly .................................................................... 8
    1.4.2.2 Requirement 2 ............................................................................................................................ 8
      1.4.2.2.1 Consistent Alarm and Event History during normal operation .................................. 8
    1.4.2.3 Requirement 3................................................................................................................................ 8
      1.4.2.3.1 Consistent Alarm and Event History after Server Failure ........................................... 8

2 Index .............................................................................................................................................................. 10
1 Event Replication

1.1 Event Replication Overview

The OpenEnterprise Event Replication component enables the EventLog from a Source OpenEnterprise Server database to be replicated to other OpenEnterprise Server databases.

In OpenEnterprise, the EventLog table temporarily stores the most recent one hundred events, sourced directly from the AlarmSummary table. The entries in this table are then logged historically, providing a history of alarms and events that have occurred within the OpenEnterprise system.

The Event Replication component enables OpenEnterprise Servers across a Wide Area Network, to synchronize the contents of their historical alarm and event logs.

There are two main modes of Event Replication:

1.1.1 Real-time Event Replication

This mode is employed during the normal operation of the Source and Target Servers. Events from the Source OpenEnterprise Server can be replicated to one or more Target OpenEnterprise Servers. This functionality will only be available if both Source and Target Servers are running, and communication between these is possible.

The Event Replication Component monitors the contents of the EventLog table within the Source OpenEnterprise Server, and when applicable events occur, those events will be copied to the EventLog table of the Target OpenEnterprise Server or Servers.

1.1.2 Back-fill Event Replication

Back-fill mode provides the Event Replication component with the ability to copy all applicable alarms/events that have occurred within the Source OpenEnterprise Server, during a loss of communication with the Target Server, back to the Target Server once communication has been resumed.

When a failed connection with the Target Server is restored, the Event Replication component queries the Event History table on the Source OpenEnterprise Server, and then feeds the returned data back through the Event Log table of the Target OpenEnterprise Server. This historical query is based on the time that communication with the Target Server was lost, plus a user configurable latency period. Note that only those alarms and events that are available from the log file and any on-line archives within the Source Server can be used in the back fill operation.

All Event Log updates on the Target OpenEnterprise Server will be performed through the id = -1 record, which will always exist as long as the standard DefaultLoad.sql has been included.

1.2 User Interface

The Event Replication component’s User Interface provides the user with a description of its actions, connection times, and the status of each action.
1.3 Configuration

1.3.1 Event Replication Configuration Overview

In order to run the Event Replication component, two areas of configuration are required:

1. Database Configuration
2. Command Line parameter

1.3.2 Database Configuration

The Event Replication component is configured through the EventReplication table in the OpenEnterprise database. Below is a table containing each attribute of this table, and a description of the way it should be configured.

<table>
<thead>
<tr>
<th>Attribute Name</th>
<th>Data Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>DataService</td>
<td>Char primary key</td>
<td>The data service of the Target OpenEnterprise Server to which events will be copied. This conforms to the standard OpenEnterprise data service format of [Host:]Service[,[Host:]Service] for a redundant Server database - e.g. OEServer1:rtadb1,OEServer2:rtadb1, and [Host:]Service for a standalone database - e.g. OES standaloneServer:rtadb1</td>
</tr>
<tr>
<td>Priority</td>
<td>Char</td>
<td>A CSV list that contains the list of those priorities of alarms/events that will be copied to the Target OpenEnterprise Server. A NULL value for this attribute will indicate that all alarms/events should be copied.</td>
</tr>
</tbody>
</table>
### Event Replication

<table>
<thead>
<tr>
<th>Attribute</th>
<th>Type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Disable</td>
<td>Boolean</td>
<td>If set to TRUE then real-time Event replication will be disabled for this table entry. The default value is FALSE.</td>
</tr>
<tr>
<td>CopyInitialEvents</td>
<td>Boolean</td>
<td>Allows the user to configure whether on connection to the Source Server the current standing events of the system are replicated to the Target OpenEnterprise Server. The default value is FALSE.</td>
</tr>
<tr>
<td>Backfill</td>
<td>Boolean</td>
<td>If this attribute value is set to TRUE, then whenever reconnection is obtained to a Target Server, the event Replication component will attempt to backfill that server with any Events that have occurred while the Target Server was not available. The default value is FALSE.</td>
</tr>
<tr>
<td>Latency</td>
<td>Integer</td>
<td>A value, in seconds that is added to the start and end of the query range, in order to allow for the delay between an OpenEnterprise Server becoming available/unavailable and the Event Replication Component being informed of that change of state. Default value is 30 seconds.</td>
</tr>
<tr>
<td>QueryPeriod</td>
<td>Integer</td>
<td>A value, in seconds, that indicates the period of time covered by each query on the Source OpenEnterprise Server's EventHistory, when performing a backfill operation. Default value is 300 seconds. If this value is 0 or NULL then this indicates that a single query will be used to obtain all event history records.</td>
</tr>
<tr>
<td>MinimumDisconnectionPeriod</td>
<td>Integer</td>
<td>The minimum time, in seconds, between a disconnection and reconnection to the Target OpenEnterprise Database, before a back-fill will be performed. Default value is 0.</td>
</tr>
<tr>
<td>TransactionPackageSize</td>
<td>Integer</td>
<td>While performing a back-fill operation this will define the number of updates of the Event Log table to be packaged in a single transaction to the Target OpenEnterprise Server. Default value is 1.</td>
</tr>
<tr>
<td>AccessArea</td>
<td>Char</td>
<td>The Access Area to which the Event Replication object belongs. The default value is ‘ALL’.</td>
</tr>
</tbody>
</table>

#### 1.3.2.1.1 Note

It is possible to replicate alarms and events to multiple target OpenEnterprise Servers by adding multiple entries into the EventReplication table.
1.3.3 Latency

The Latency period allows the user to define a period of time that is subtracted from the disconnection time, and added to the reconnection time, thus increasing the time range of the EventHistory queries on the Source OpenEnterprise Server. This is mainly to handle the delay between the Target OpenEnterprise Server actually failing, and the Event Replication component being informed of that failure. Adding the Latency ensures that no events are missed during the back-fill operation.

1.3.4 Command Line Configuration

The OpenEnterprise Event Replication Component will support the following command line option, which will identify the Source Server from which Events are to be sourced.

\[-s \text{<dataservice>}\]

Where \text{<dataservice>} conforms to the standard \text{[Host:]Service[,][Host:]Service} format, e.g. \text{rtrdb1} or \text{OEServer1:rtrdb1,OEServer2:rtrdb1}.

1.4 Example Configuration

1.4.1 Example System

Under normal operating conditions our Example system has the following architecture. The system is using Bristol RTUs, which allows the RDI3000 to connect to OpenBSI for data collection.
The East and West Servers are separated by several hundred miles, and communicate over a 128Kb fixed link. However they are mirrors of each other in terms of their actual database contents. Under normal operating conditions all RTU’s report alarms to the East Servers, (as well as RBE), whilst the West Servers use template collection to obtain values etc from the RTU’s for all signals.

The East Workstations source all signal and alarm information directly from the East Servers. The West Workstations source all signal information from the West Servers, but alarms are sourced from the East Servers.

If the East Servers fail then, following manual intervention, all alarms are redirected to the West Servers.

The functional requirements of the Example system are as follows.

1. The West Servers must be kept up to date with all signal alarm information, (i.e. the values of the inalarm, acknowledged etc. attributes), in order to satisfy the requirements of the mimics on the West Workstations.

2. An identical history of any alarms reported on the system must be available from both the East and West Servers.

3. Following a failure of the East Servers, the Alarm Summary contents of the West Servers must be consistent with those of the East Servers prior to the failure. The same requirement also exists when the East Servers subsequently become available again.

Example Solution
1.4.2 Example Solution

These are the solutions to the requirements of the example system:-

1.4.2.1 Requirement 1

1.4.2.1.1 West Servers to collect signal values directly

This requirement is fulfilled by enabling the RDI3000 on the West Servers to collect signal values through template polling. See the NW3000 Poll List Builder Help file for more information.

1.4.2.2 Requirement 2

1.4.2.2.1 Consistent Alarm and Event History during normal operation

This is fulfilled by configuring the Event Replication Component on the East Servers to provide real-time replication of the East Server Event Log to the West Server Event Log. The following Event Replication table entry attribute values on the East Servers would be required:

- Dataservice = “WestServer1:rtrdb1,WestServer2:rtrdb1”
- Disable = FALSE
- Priority = “0,1,2,3”
- BackFill = TRUE

Because the Disable attribute for Event Replication on the East Servers is set to False (i.e. it is enabled), during normal operation the East Server Event Log will be replicated on the West Server database.

Because the Backfill attribute is set to True, if the West Servers fail, then when communication is restored, the East Servers will backfill the West Servers Event Log with any missed events.

1.4.2.3 Requirement 3

1.4.2.3.1 Consistent Alarm and Event History after Server Failure

The East Servers are already configured for Event Log back-fill replication should there be a failure of the West Servers. What if the East Servers fail though? In this case, the West Servers would be manually configured to receive the remote alarm data from the RTUs, which would enable the West Event Log to be updated directly by the RTU data. Then, when the East Servers are back online, their Event Log should be updated by the West Servers, remote alarm data should be re-routed to the East Servers and from that point on real-time event replication should continue from the East to the West Servers.

Therefore the Event Replication table entry attribute values for the West Servers would be.

- Dataservice = “EastServer1:rtrdb1,EastServer2:rtrdb1”
- Disable = TRUE
- Priority = “0,1,2,3”
- BackFill = TRUE
Note, that it is vitally important that the Disable attribute is set to TRUE on the West Servers to prevent them from also attempting real-time replication once the East Servers have resumed operation. This would cause the equivalent of a spin, whereby both OpenEnterprise Server pairs would repeatedly be updating each other with the same alarms and events.

The values of the attributes used to tune the backfill operation, (e.g. QueryPeriod, Latency, MinimumDisconnectionTime and TransactionPackageTime) for both Server pairs would need to be decided once the performance and durability of the link between the two Server pairs had been determined, but the defaults should suffice for initial setup.

EventLog Replication Overview
2 Index

C
Command Line Configuration ..................... 5

D
Database Configuration .............................. 2

E
Event Replication Configuration Overview.. 2
EventLog Replication Overview ................. 1
Example Solution ................................. 7
Example System ..................................... 5

L
Latency.................................................... 4

U
User Interface.......................................... 2