

Event Replicator for Adabas

Replication Monitoring

Version 4.1.1

October 2022

This document applies to Event Replicator for Adabas Version 4.1.1 and all subsequent releases.

Specifications contained herein are subject to change and these changes will be reported in subsequent release notes or new editions.

Copyright © 2022 Software AG, Darmstadt, Germany and/or Software AG USA, Inc., Reston, VA, USA, and/or its subsidiaries and/or its affiliates and/or their licensors.

The name Software AG and all Software AG product names are either trademarks or registered trademarks of Software AG and/or Software AG USA, Inc. and/or its subsidiaries and/or its affiliates and/or their licensors. Other company and product names mentioned herein may be trademarks of their respective owners.

Detailed information on trademarks and patents owned by Software AG and/or its subsidiaries is located at <http://softwareag.com/licenses>.

Use of this software is subject to adherence to Software AG's licensing conditions and terms. These terms are part of the product documentation, located at <http://softwareag.com/licenses/> and/or in the root installation directory of the licensed product(s).

This software may include portions of third-party products. For third-party copyright notices, license terms, additional rights or restrictions, please refer to "License Texts, Copyright Notices and Disclaimers of Third-Party Products". For certain specific third-party license restrictions, please refer to section E of the Legal Notices available under "License Terms and Conditions for Use of Software AG Products / Copyright and Trademark Notices of Software AG Products". These documents are part of the product documentation, located at <http://softwareag.com/licenses> and/or in the root installation directory of the licensed product(s).

Use, reproduction, transfer, publication or disclosure is prohibited except as specifically provided for in your License Agreement with Software AG.

Document ID: ARF-MONITORING-411-20220204

Table of Contents

Preface	v
1 About this Documentation	1
Document Conventions	2
Online Information and Support	2
Data Protection	3
2 Replication Monitoring Overview	5
3 Installing and Configuring Replication Monitoring	9
Replication Monitoring Requirements	10
Configuring Event Replicator for Adabas to send Statistics, Events and Heartbeats	10
Starting Elasticsearch and Kibana	12
Configuring Adabas Analytics Collector MF using the Event Replicator Target Adapter Administration	12
Starting Event Replicator Target Adapter	15
Copying IBM MQ Series Jar Files (Optional)	15
Restarting Adabas Analytics Collector MF	15
Verify the Replication Monitoring Installation	16
Import and Explore Adabas Analytics Collector MF Kibana Examples	16
4 Replication Monitoring Metrics	19
Event Replicator Server Statistics	20
Event Replicator Target Adapter Statistics	32
Heartbeat	33
5 Replication Monitoring Events	35

Preface

This documentation provides basic information for Replication Monitoring.

This document is organized as follows:

<i>Replication Monitoring Overview</i>	Provides a basic description of the concepts behind and the tools used with Replication Monitoring.
<i>Installing and Configuring Replication Monitoring</i>	Describes how to install Replication Monitoring and provides basic configuration instructions.
<i>Replication Monitoring Metrics</i>	Describes the various statistics (or metrics) sent from the Event Replicator Server and made available to the Replication Monitoring subsystem.
<i>Replication Monitoring Events</i>	Describes the various events that can be sent by the Event Replicator Server on the mainframe.

1 About this Documentation

- Document Conventions 2
- Online Information and Support 2
- Data Protection 3

Document Conventions

Convention	Description
Bold	Identifies elements on a screen.
Monospace font	Identifies service names and locations in the format <code>folder.subfolder.service</code> , APIs, Java classes, methods, properties.
<i>Italic</i>	Identifies: Variables for which you must supply values specific to your own situation or environment. New terms the first time they occur in the text. References to other documentation sources.
Monospace font	Identifies: Text you must type in. Messages displayed by the system. Program code.
{ }	Indicates a set of choices from which you must choose one. Type only the information inside the curly braces. Do not type the { } symbols.
	Separates two mutually exclusive choices in a syntax line. Type one of these choices. Do not type the symbol.
[]	Indicates one or more options. Type only the information inside the square brackets. Do not type the [] symbols.
...	Indicates that you can type multiple options of the same type. Type only the information. Do not type the ellipsis (...).

Online Information and Support

Product Documentation

You can find the product documentation on our documentation website at <https://documentation.softwareag.com>.

In addition, you can also access the cloud product documentation via <https://www.software-ag.cloud>. Navigate to the desired product and then, depending on your solution, go to “Developer Center”, “User Center” or “Documentation”.

Product Training

You can find helpful product training material on our Learning Portal at <https://knowledge.softwareag.com>.

Tech Community

You can collaborate with Software AG experts on our Tech Community website at <https://tech-community.softwareag.com>. From here you can, for example:

- Browse through our vast knowledge base.
- Ask questions and find answers in our discussion forums.
- Get the latest Software AG news and announcements.
- Explore our communities.
- Go to our public GitHub and Docker repositories at <https://github.com/softwareag> and <https://hub.docker.com/publishers/softwareag> and discover additional Software AG resources.

Product Support

Support for Software AG products is provided to licensed customers via our Empower Portal at <https://empower.softwareag.com>. Many services on this portal require that you have an account. If you do not yet have one, you can request it at <https://empower.softwareag.com/register>. Once you have an account, you can, for example:

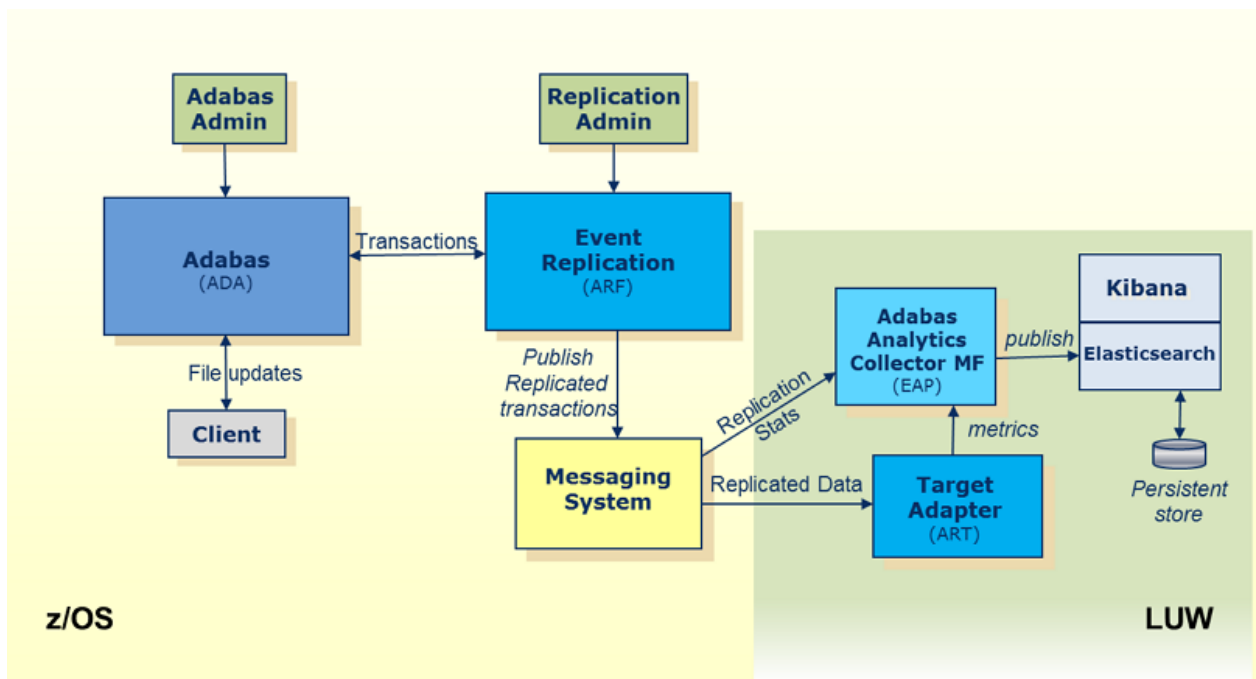
- Download products, updates and fixes.
- Search the Knowledge Center for technical information and tips.
- Subscribe to early warnings and critical alerts.
- Open and update support incidents.
- Add product feature requests.

Data Protection

Software AG products provide functionality with respect to processing of personal data according to the EU General Data Protection Regulation (GDPR). Where applicable, appropriate steps are documented in the respective administration documentation.

2 Replication Monitoring Overview

Replication Monitoring is a tool that will allow users to monitor the status and progress of data being replicated by the Event Replicator for Adabas. While statistical information (metrics) may be gathered by issuing DRPLSTAT operator commands, utilizing the Replication Monitoring tool provides an automated way to collect these metrics. The metrics may then be graphically viewed using the Kibana application. Kibana, along with Elasticsearch analytics engine, may be installed under a Windows, UNIX or Linux operating system.



Activating the Replication Monitoring subsystem is achieved by setting the value of the global variable parameter, `STATINTERVAL`, to a value other than zero. A new Destination, with a destination type (DTYPE) of `ETBROKER` or `MQSERIES` (or `NULL` for test purposes), should then be created,

and the `DSTATLOG` parameter set to "YES". The result of setting up this new Destination will be for the Replication metrics to be sent to the Adabas Analytics component of Replication Monitoring.

One additional parameter is the `DEVENTLOG`. Setting this to "YES" will result in events to be sent to the Adabas Analytics component. Examples of events include Destination(s) being Opened or Closed, or Subscription File(s) being Activated and Deactivated.

Another aspect of Replication Monitoring is the Heartbeat feature. There is one new parameter that needs to be set to activate this feature: `DHBINTERVAL`. The `DHBINTERVAL` determines how often the Event Replicator Server sends a heartbeat to a Destination. The Destination must be one used to send replication data to the Event Replicator Target Adapter.

Using Kibana, it is possible to visualize a number of metrics provided by the Event Replicator Server. Each visualization may be of a single or multiple metrics, or a customized dashboard may be created to have a single view of a number of visualizations. Then, with the flexibility of Elastic-Search, the dashboards may facilitate different time ranges of metrics; as much as one year, showing a view of the replication workload trends.

The Replication metrics have a number of categories, which are described in more detail in the section [Replication Monitoring Metrics](#). The categories are as follows:

- **Global General Statistics**

This provides an overview about the number of transactions received and sent by the Event Replicator Server.

- **Subscription**

Where a number of Subscriptions may have been defined to the Event Replicator Server, an overview of the activities of any/all subscriptions may be monitored. This includes any Replication Initial State requests (i.e. Initial State requests not done asynchronously using the ADARIS utility).

- **Subscription file**

For each Subscription, one or more Subscription files may be defined. Monitoring these provides details about the commands being issued for each file.

- **Destination**

Monitoring the Destinations provides an overview of the number of items being sent to specific targets and the amount of data. If Destination is manually or automatically closed, and logging is active, the amount of data being logged and de-logged may be tracked, i.e. Destination SLO-Ging activities are.

- **Destination file for an Adabas destination**

Where the Replication is from Adabas to Adabas. the modifications being made to the target destination files may be tracked.

- **Subtask**

Where the Destination types are either EntireX Broker or MQ Series, an overview of the replicated transactions may be monitored.

- **Input queue**

If Node to Node Replication is being used or other requests are being sent by an application to the Event Replicator Server, these may be monitored.

- **SLOG**

Under general statistics, the usage of the SLOG system file may be monitored. The number of items logged and de-logged from the file may be tracked.

- **Events**

When Replication is active, a number of events may occur during normal processing, e.g. a Destination may become closed or opened, a file may be inactive or active, a destination may become full, the Replication Pool may become short on storage. These events would be written to the console but, with Replication Monitoring, a specific visualization allows an overview of all events to be viewed.

- **Event Replicator Target Adapter (ART)**

The Event Replicator Target Adapter subsystem activities may, similarly, be tracked. There are a number of java beans that Event Replicator Target Adapter provides to monitor replicated transactions being processed. This will be done for each target (e.g. RDBMS, JMS, Terracotta). These metrics will also be viewable from Kibana, showing visualizations of the trends or current status of Event Replicator Target Adapter. See also *Reviewing Event Replicator Target Adapter Statistics* in the Event Replicator Target Adapter Administrator's Guide for further details.

- **Heartbeat**

When the Heartbeat feature is active, the heartbeats sent to the Event Replicator Target Adapter, will have various timestamps. These timestamps will reflect the latency at various points; points that would indicate if the delivery of the replicated data is being delayed. Examples would be, if the workload being sent to the Event Replicator Server and the data had to be written to the SLOG, or if it was queued up in the EntireX Broker or MQ Series.

Once the heartbeat reaches the Event Replicator Target Adapter, additional information is added before the heartbeat metrics are passed on to the Adabas Analytics component. In short, the heartbeat will provide information that reflects the rate at which replicated data is being processed and is likely to change based on the number of transactions being replicated.

See also *Reviewing Event Replicator Target Adapter Statistics* in the Event Replicator Target Adapter Administrator's Guide for further details.

3

Installing and Configuring Replication Monitoring

- Replication Monitoring Requirements 10
- Configuring Event Replicator for Adabas to send Statistics, Events and Heartbeats 10
- Starting Elasticsearch and Kibana 12
- Configuring Adabas Analytics Collector MF using the Event Replicator Target Adapter Administration 12
- Starting Event Replicator Target Adapter 15
- Copying IBM MQ Series Jar Files (Optional) 15
- Restarting Adabas Analytics Collector MF 15
- Verify the Replication Monitoring Installation 16
- Import and Explore Adabas Analytics Collector MF Kibana Examples 16

This chapter describes the steps necessary to install and configure Replication Monitoring.

Replication Monitoring Requirements

To use Replication Monitoring, the following products must be installed:

- Adabas Analytics Collector MF 1.1 or above.
- Event Replicator for Adabas 3.6.1 or above. Details on the installation of Event Replicator for Adabas can be found under *Installation* in the Event Replicator for Adabas Installation Guide.
- Event Replicator Target Adapter 3.3 or above. Details on the installation of Event Replicator Target Adapter can be found under *Installing Event Replicator Target Adapter* in the Event Replicator Target Adapter Installation Guide.



Note: This is a prerequisite if you want to use Heartbeat.

- Event Replicator Target Adapter Administration 3.3. or above. Details on the installation of Event Replicator Target Adapter Administration can be found under *Installing Event Replicator Target Adapter Administration* in the Event Replicator Target Adapter Administration Guide.
- Elasticsearch 6.2.4 or above. Details on the installation and setup of Elasticsearch can be found under <https://www.elastic.co/guide/en/elasticsearch/reference/current/index.html>.
- Kibana 6.2.4 or above. Details on the installation and setup of Kibana can be found under <https://www.elastic.co/guide/en/kibana/current/index.html>

Elasticsearch and Kibana are open source and available for download at <https://www.elastic.co/products>.

Configuring Event Replicator for Adabas to send Statistics, Events and Heartbeats

There are four Event Replicator for Adabas parameters that can be setup via the Adabas Event Replicator Subsystem, or in the Replication startup parameters (DDKARTE ADARPD statement) to receive Replication metrics, Replication Events, and Replication Heartbeats:

- STATINTERVAL
- DSTATLOG
- DEVENTLOG

- **DHBINTERVAL**

STATINTERVAL

```
ADARPD STATINTERVAL=interval
```

The value for *interval* specifies how often the Event Replicator Server should send out statistics. The value is specified in seconds and the minimum value is 60. A value of 0 means statistics are not sent.

For more information read about the STATINTERVAL parameter , in the *Event Replicator for Adabas Reference Guide*.

DSTATLOG

```
ADARPD DESTINATION NAME=destination_name  
...  
DSTATLOG=YES  
...
```

The DSTATLOG=YES parameter specified as part of a destination definition tells the Event Replicator Server to send the replication statistics to the Adabas Analytics Collector MF environment.

For more information read about the DSTATLOG parameter , in the *Event Replicator for Adabas Reference Guide*.

DEVENTLOG

```
ADARPD DESTINATION NAME=destination_name  
...  
DEVENTLOG=YES
```

The DEVENTLOG=YES parameter should be specified as part of the Adabas Analytics Collector MF destination set up. This option instructs the Event Replicator Server to send replication event information to the Adabas Analytics Collector MF environment.

For more information read about the DEVENTLOG parameter, in the *Event Replicator for Adabas Reference Guide*.

DHBINTERVAL

```
ADARPD DESTINATION NAME=destination_name
...
DHBINTERVAL=interval
DCLASS=SAGTARG
```

The value *interval* specifies how often Event Replicator Server should send heartbeats to this destination. The value is specified in seconds and the minimum value is 60. A value of 0 means heartbeat events are not sent. This parameter can be separately applied to each destination.

It can only be used with destinations that also specify `DCLASS=SAGTARG` i.e. where the Destination target is the Event Replicator Target Adapter.

It is possible to specify multiple destinations to receive the statistics, events, and heartbeats, or a single destination may be defined.

For more information read about `DHBINTERVAL` parameter, in the *Event Replicator for Adabas Reference Guide*.

Starting Elasticsearch and Kibana

For Elasticsearch, open a Command Prompt (Windows) or a telnet session (UNIX) and type:

```
cd <Elasticsearch_install_dir>/bin
elasticsearch
```

Where `<Elasticsearch_install_dir>` is the path to your Elasticsearch installation.

For Kibana, open a Command Prompt (Windows) or a telnet session (UNIX) and type:

```
cd <Kibana_install_dir>/bin
kibana
```

Where `<Kibana_install_dir>` is the path to your Elasticsearch installation.

Configuring Adabas Analytics Collector MF using the Event Replicator Target Adapter Administration

➤ To add an Adabas Analytics Collector MF environment:

- 1 Start Event Replicator Target Adapter Administration.
- 2 Go to **Window > Preferences**, select **Adabas Analytics Collector MF** and click **Add**.

- 3 Enter a name to identify this Adabas Analytics Collector MF environment.
- 4 In the **Root Directory** field:
 - If the installation of the Adabas Analytics Collector MF is a local installation, click the **Browse** button and navigate to `<SAG_install_root>/AdabasAnalyticsCollectorMF`.
 - If the installation of the Adabas Analytics Collector MF is on a remote host, check the **Remote** box and manually enter the fully qualified path to the product root.
- 5 Click **OK**.
- 6 This is an optional step. Click the **Validate** button to validate the settings you entered.
- 7 Click **OK** to add this Adabas Analytics Collector MF environment for management.
- 8 Click **OK** to exit Preferences.

Further details are also given in *Setting Configuration File and Preferences*, in the *Event Replicator Target Adapter Administrator's Guide*

➤ **To configure an Elasticsearch target to receive the Event Replicator for Adabas statistics and events:**

- 1 In the Event Replicator Target Adapter Administration, expand the node you just created for this Adabas Analytics Collector MF environment.
- 2 Expand the **Targets** node.
- 3 Right click on **Elasticsearch** and click **New**.
- 4 Enter the information to identify your Elasticsearch installation.
- 5 For the Index use "reptor_monitor" if you plan to use the provided Adabas Analytics Collector MF examples.
- 6 Save the target definition under a name of your choice using the Save icon or **File > Save**.

Further details are also given in *Configuring Target Definitions for Adabas Analytics Collector MF*, in the *Event Replicator Target Adapter Administrator's Guide*

➤ **To configure an Elasticsearch target to receive Event Replicator Target Adapter metrics and Event Replicator Server heartbeats:**

- 1 Please take care that the following ports are not in use already and ensure - depending on your configuration and/or firewall settings - that these ports are open for inbound communication.

Port number	Used for
4981	Adabas Analytics Collector MF shutdown
4000	Event Replicator Target Adapter JMX metrics
9200	Elasticsearch
9300	Elasticsearch
5601	Kibana

- 2 Expand the **Targets** node.
- 3 Right click on **Elasticsearch** and click **New**.
- 4 Enter the information to identify your Elasticsearch installation.
- 5 For the Index use "targetadapter" if you plan to use the provided Adabas Analytics Collector MF examples.
- 6 Save the target definition under a name of your choice using the Save icon or **File > Save**.

Further details are also given in *Configuring Target Definitions for Adabas Analytics Collector MF*, in the *Event Replicator Target Adapter Administrator's Guide*

➤ **To configure a source to receive statistics and events from mainframe Event Replicator for Adabas:**

- 1 Expand the **Sources** node.
- 2 Right click on either **EntireX Communicator** or **Websphere MQ** and click **New**.
- 3 Enter the information to identify the message queue where you will receive statistics and/or events.
- 4 For the target pick the target you created above to receive Event Replicator for Adabas statistics and events.
- 5 Save the source definition under a name of your choice using the Save icon or **File > Save**.

Further details are also given in *Configuring Source Definitions for Adabas Analytics Collector MF*, in the *Event Replicator Target Adapter Administrator's Guide*

➤ **To configure a source to receive Event Replicator Target Adapter metrics and Event Replicator Server heartbeats:**

- 1 Right click on **Target Adapter** under the **Sources** node and click **New**.
- 2 Enter the information to identify where your Event Replicator Target Adapter publishes its metrics to JMX beans. This is the port identified as the "JMX Port" during installation of the Event Replicator Target Adapter, by default port 4000.
- 3 If you are unsure of what port to use, look for the value set for the variable `ART_JMX_PORT` in the file `installArtPath.bat` (`.sh` for UNIX) under the Event Replicator Target Adapter installation location.

- 4 For the Target pick the target you created above to receive Event Replicator Target Adapter metrics and Event Replicator Server heartbeats.
- 5 Save the source definition under a name of your choice using the Save icon or **File > Save**.

Further details are also given in *Configuring Source Definitions for Adabas Analytics Collector MF*, in the *Event Replicator Target Adapter Administrator's Guide*

Starting Event Replicator Target Adapter

Start the Event Replicator Target Adapter from the Event Replicator Target Adapter Administration as described in *Starting Event Replicator Target Adapter from the Administration Tool*, in the *Event Replicator Target Adapter User Guide*.

Copying IBM MQ Series Jar Files (Optional)

If you are planning to use IBM MQ Series as a messaging source, you will need to obtain the following jar files and copy them into your Adabas Analytics Collector MF installation:

- com.ibm.mq.jar
- com.ibm.mqjms.jar
- dhbcore.jar

Copy these jar files into this location: `<SAG_install_root>/AdabasAnalyticsCollectorMF/jars/lib`.

Restarting Adabas Analytics Collector MF

➤ To restart the Adabas Analytics Collector MF

Restart the now configured Adabas Analytics Collector MF service (Windows) or daemon (UNIX).

- 1 On Windows, use the Services applet to restart the service called **Software AG Adabas Analytics Collector MF v1.0.0.x Service**.
- 2 On Linux, execute the commands `eapStopDaemon.sh` and `eapStartDaemon.sh` found in the `<SAG_install_root>/AdabasAnalyticsCollectorMF/sh` directory.
- 3 Examine the Adabas Analytics Collector MF log file at `<SAG_install_root>/AdabasAnalyticsCollectorMF/log/EAPMonitorMain.log`.
 - Look for any lines that contain ERROR to identify any potential problems.

Verify the Replication Monitoring Installation



Important: Follow this step in order for the provided Adabas Analytics Collector MF examples to work.

> To verify the installation

- 1 Ensure your replication environment is set up to perform replication from Adabas to an Event Replicator Target Adapter target.
- 2 Perform at least one update or insert or delete on your source Adabas nucleus. This will:
 - cause replicated data to flow from Event Replicator for Adabas to Event Replicator Target Adapter;
 - cause Event Replicator for Adabas statistics, events, and heartbeats and Event Replicator Target Adapter metrics to flow to Adabas Analytics Collector MF and on to Elasticsearch;
 - populate the Elasticsearch data points including the necessary schema;
 - allow Kibana to interpret the data points and schema and allow the provided Adabas Analytics Collector MF examples to work.

Import and Explore Adabas Analytics Collector MF Kibana Examples

> To import Adabas Analytics Collector MF Kibana Examples

- 1 Open Kibana:
 - Open a browser and point to your Kibana installation (default is *http://localhost:5601*).
- 2 Create index patterns:
 - In the Kibana menu (left pane) click **Management**.
 - In the Management pane (right pane) click **Index Patterns**.
 - Under **Define index pattern** locate the **Index pattern** text box and enter "targetadapter-*".
 - Click button **Next step**.
 - Under **Configure settings** locate the **Time Filter field name** drop down box and select field **timestamp**.
 - Click button **Create index pattern**.

- Near the upper left corner of the window click button + **Create Index Pattern** to create another index pattern.
 - Under **Define index pattern** locate the **Index pattern** text box and enter "reptor_monitor-*".
 - Click button **Next step**.
 - Under **Configure settings** locate the **Time Filter field name** drop down box and select field **timestamp**.
 - Click button **Create index pattern**.
- 3 Import the Kibana Examples:
- Near the upper left corner of the window click tab **Saved Objects**.
 - Near the upper right corner click button **Import**.
 - Navigate to the Adabas Analytics Collector MF Kibana examples directory.
 - The default location should be `<install_root>/third-party/Kibana-Examples`.
 - Select file `EAP-everything.json` and click button **Open**.
 - If prompted with "Automatically overwrite all saved objects?" click button **Yes, overwrite all objects**.
 - If prompted with "Index Pattern conflicts":
 - o use the drop down boxes under **New index pattern** to map:
 - ID "targetadapter-*" to the new index pattern "targetadapter-*" and
 - ID "reptor_monitor-*" to "reptor_monitor-*".
 - Click button **Confirm all changes**.
 - You should then see 13 saved Dashboards and 44 Visualizations.

➤ To explore Adabas Analytics Collector MF Kibana Examples

- 1 Open Kibana (if not already open):
 - Open a browser and point to your Kibana installation (default is `http://localhost:5601`).
- 2 Open any dashboard:
 - Click **Dashboard** in the left hand pane.
 - Click on any dashboard name. Suggested starting points are:
 - **ReptorMonitor: Global** for Event Replicator for Adabas statistics and/or events.
 - **Event Replicator Target Adapter Dashboard** for Event Replicator Target Adapter metrics.

- **Latency** for flow timing of the whole system.
- **Replicator Heartbeat Dashboard** for heartbeats.

You should see many visualizations that change as replication flows from the source Adabas nucleus through Event Replicator for Adabas through messaging down to Event Replicator Target Adapter and on to the target RDBMS(s) or other Event Replicator Target Adapter target types.

➤ **To import your own dashboards and visualizations from Kibana 5.1.2**

- If you have an existing installation of Elasticsearch/Kibana 5.1.2 and you have created your own dashboards and visualizations that you want to preserve, it is suggested that you do an "Export Everything" from your Kibana 5.1.2 installation. Then back up your Elasticsearch data and delete your *.kibana* index. Then follow normal Elastic upgrade instructions and finally import all of your dashboards and visualizations into your new Kibana 6.2.4 installation.

4 Replication Monitoring Metrics

- Event Replicator Server Statistics 20
- Event Replicator Target Adapter Statistics 32
- Heartbeat 33

This chapter is organized as follows:

Event Replicator Server Statistics

This section describes the various statistics (or metrics) sent from the Event Replicator Server and made available to the Replication Monitoring subsystem. Snapshots of these statistics may also be seen via the DRPLSTAT operator command, or through the Adabas Online System (AOS). The format of these statistics may differ slightly between the different facilities.

While each metric reflects specific information about a certain aspect of the replication process, combinations of some metrics could make the statistical information and trends more meaningful; for example:

If monitoring the SLOG status, a change in items logged and de-logged, to the SLOG file, may reflect the load on the Replication system. For instance, if the number of de-logged items increases at a faster rate than new items are logged, then the Event Replicator Server through-put could be deemed to be optimistic. However, if the logged items are increasing faster than items are being de-logged, it may imply that the Event Replicator Server is receiving a higher load than normal or a bottleneck may be occurring. This may be also verified by looking at the Input Transactions.

While a single snapshot may seem concerning, it may merely mean that a large transaction or batch job, doing heavy updates, may be in progress. Further monitoring of the metrics would verify this trend more accurately.

This section covers the following statistical categories originating from the Event Replicator Server:

- [Global Statistics](#)
- [Subscription Statistics](#)
- [Subscription File Statistics](#)
- [Deleted Subscription Statistics](#)
- [Destination Statistics](#)
- [Adabas Destination Statistics](#)
- [Deleted Destination Statistics](#)
- [Subtask Statistics](#)
- [Input Queue Statistics](#)

- Deleted Input Queue Statistics

Global Statistics

Sample output

```

Input transactions           :           25
Pending input transactions  :            0
Output items                :           470
Pending output items       :            0
Output messages            :           498
  Bytes sent                :       1,601,617
Input messages             :           29,364
  Bytes received            :       26,374,377
Input commits              :           29,357
Input backouts             :            0
Items de-logged            :            41
Items logged               :            41
Items on SLOG              :            0

Database-related input transactions:
  De-logged                 :            0
  Logged                   :            0
  On SLOG                  :            0
  FCBs on SLOG             :            0 ←

```

Labels and values shown for Global Statistics

The global statistics section shows statistics collected for the entire Event Replicator Server session.

Label	Value Description	Elasticsearch Field
	ReptorStatisticGlobal	type
	URBS reptor status ("RSTA")	reptorStatus
	Target ID of originating Event Replicator Server	targetID
	Nucleus ID of originating Event Replicator Server	nucleusID
Input transactions	Shows the number of fully processed input replicated transactions.	inputTransactions
Pending input transactions	Shows the number of in-flight input transactions.	pendinInputTransactions
Output items	Shows the total number of output transactions and events (i.e. URBS) for all destinations.	
Pending output items	Shows the total number of in-flight output transactions and events for all destinations.	
Output messages	Shows the total number of output messages sent for all destinations.	totalMessagesAllDests

Label	Value Description	Elasticsearch Field
Bytes sent	Shows the total number of bytes in the output messages.	totalBytesAllDests
Input messages	Shows the total number of input messages received for all input queues.	totalMessagesAllQueues
Bytes received	Shows the total number of bytes in the input messages.	totalBytesAllQueues
Input commits	Shows the total number of commits issued for all input queues.	totalCommitsAllQueues
Input backouts	Shows the total number of backouts issued for all input queues.	totalBackoutsAllQueues
Items de-logged	Shows the total number of transactions and events read from the SLOG system file for all destinations including database-related input transactions.	totalItemsDeloggedSLOG
Items logged	Shows the total number of transactions and events written to the SLOG system file for all destinations including database-related input transactions.	totalItemsLoggedSLOG
Items on SLOG	Shows the total number of transactions and events currently residing on the SLOG system file for all destinations including database-related input transactions.	totalItemsOnSLOG
	Java Date instance containing URBS timeTransProcessedStart equivalent	timestamp
	Java Date instance containing URBS timeTransCommitEnd equivalent	timestampEnd
	interval length in milliseconds timestampEnd minus timestamp	intervalLength

Labels and values shown for Database-related input transactions

The Database-related input transactions section has the following statistics:

Label	Value Description	Elasticsearch Field
De-logged	Shows the number of database-related input transactions read from the SLOG system file.	dbTransDeloggedSLOG
Logged	Shows the number of database-related input transactions written to the SLOG system file.	dbTransLoggedSLOG
On SLOG	Shows the number of database-related input transactions currently residing on the SLOG system file.	dbTransOnSLOG
FCBs on SLOG	Shows the number of Adabas File Control Blocks (FCBs) currently residing on the SLOG system file.	dbFCBFDTTTransOnSLOG

Subscription Statistics

Sample output

```

SUB1
  Input items           :                24
  Output items         :                26
  C5 data              :                 0
  Data lost            :                 1
  Initial-state completed :             1
  Initial-state data   :                23
  Security functions   :                 1
  User transactions    :                 0
  Utility functions    :                 0
  Item time at source  : 2017-10-17 18:45:41
  Item processed       : 2017-10-17 18:45:41

```

Labels and values shown for subscription statistics

The subscription statistics section is repeated for each subscription defined to the Event Replicator Server.

Label	Value Description	Elasticsearch Field
	ReptorStatisticSubscription	type
	URBS reptor status ("RSTA")	reptorStatus
	Target ID of originating Event Replicator Server	targetID
	Nucleus ID of originating Event Replicator Server	nucleusID
SUB1 (sample)	Subscription name	subscriptionName
Input items	Shows the number of input transactions processed for this subscription.	
Output items	Shows the total number of output transactions and events (i.e. URBS) for this subscription.	subscriptionOutputItems
C5 data	Shows the number of C5 events created for this subscription.	subscriptionC5DataTransactions
Data lost	Shows the number of data lost events created for this subscription.	subscriptionDataLostTransactions
Initial-state completed	Shows the number of initial-state completed events created for this subscription.	subscriptionInitialStateCompletions
Initial-state data	Shows the number of initial-state transactions processed for this subscription.	subscriptionInitialStateDataTransactions
Security functions	Shows the number of security function events created for this subscription.	subscriptionSecurityFunctions

Label	Value Description	Elasticsearch Field
User transactions	Shows the number of replicated transactions processed for this subscription.	subscriptionUserDataTransactions
Utility functions	Shows the number of Adabas utility-related events created for this subscription.	subscriptionUtilityFunctions
Item time at source	Is the commit time of the last processed transaction or event for this subscription.	subscriptionLastItemCommitTime
Item processed	Is the end of subscription processing time for the last processed transaction or event for this subscription.	subscriptionLastItemProcessedTime
	subscriptionLastItemProcessedTime minus subscriptionLastItemCommitTime	subscriptionLastItemDurationTime
	subscriptionOutputItems divided by intervalLength	subscriptionItemsPerSecond
	Java Date instance containing URBS timeTransProcessedStart equivalent	timestamp
	Java Date instance containing URBS timeTransCommitEnd equivalent	timestampEnd
	interval length in milliseconds timestampEnd minus timestamp	intervalLength

Subscription File Statistics

Sample output

```

DBID 28857 File      1
  Deletes              :           0
  Filtered out        :           1
  Initial-state       :           0
  Inserts              :       1,106
  Updates              :           2
    
```

Labels and values shown for subscription file statistics

The subscription file statistics section is repeated for each database ID (SFDBID) and file (SFILE) combination defined to an individual subscription.

Label	Value Description	Elasticsearch Field
	ReptorStatisticSubscriptionFile	type
	URBS reptor status ("RSTA")	reptorStatus
	Target ID of originating Event Replicator Server	targetID
	Nucleus ID of originating Event Replicator Server	nucleusID
	Subscription name	subscriptionName
DBID 28857 (sample)	Is the database ID in the above sample.	subscriptionFileDBID
File 1 (sample)	Is the file ID in the above sample.	subscriptionFileFNR
Deletes	Shows the number of deletes processed for the file.	subscriptionFileDeletes
Filtered out	Shows the number of records which were excluded from replication processing by transaction filtering for the file.	
Initial-state	Shows the number of initial-state transactions processed for the file.	subscriptionFileInitialStateRecords
Inserts	Shows the number of inserts processed for the file.	subscriptionFileInserts
Updates	Shows the number of updates processed for the file.	subscriptionFileUpdates
	Java Date instance containing URBS timeTransProcessedStart equivalent	timestamp
	Java Date instance containing URBS timeTransCommitEnd equivalent	timestampEnd
	interval length in milliseconds timestampEnd minus timestamp	intervalLength

Deleted Subscription Statistics

Sample output

```
Deleted subscriptions
Output items           :           1
```

Labels and values shown for deleted subscription statistics

Label	Value Description
Output items	Shows the total number of output transactions and events (i.e. URBS) for all deleted subscriptions.

Destination Statistics

Sample output

```

DEST1
Output items           :           137
Pending items         :             0
Item time at source   : 2017-10-17 18:59:31
Messages              :           165
Bytes                 :       1,401,542
Commits               :           128
Commit time at destination : 2017-10-17 18:59:31
Pending messages     :             1
Pending bytes        :           110
De-logged             :             11
Logged                :             11
On SLOG               :             0
    
```

Labels and values shown for destination statistics

The destination statistics section is repeated for each destination defined to the Event Replicator Server.

Label	Value Description	Elasticsearch Field
	ReptorStatisticDestinationBrokerMQNull	type
	URBS reptor status ("RSTA")	reptorStatus
	Target ID of originating Event Replicator Server	targetID
	Nucleus ID of originating Event Replicator Server	nucleusID
DEST1 (sample)	Is the destination name in this example.	destinationName
Output items	Shows the number of output transactions and events (i.e. URBS) for the destination.	destOutputItems
Pending items	Shows the number of in-flight output transactions and events for the destination.	destPendingItems
Item time at source	Is the commit time of last processed transaction or event for the destination.	destItemTimeAtSource
Messages	Shows the number of output messages sent for the destination.	destMessages
Bytes	Shows the number of bytes in the output messages.	destBytes
Commits	Shows the number of messaging system commits for the destination.	destCommits
Commit time at destination	Is the time of the last commit issued to the messaging system for the destination.	destCommitTimeAtDestination

Label	Value Description	Elasticsearch Field
Pending messages	Shows the number of messages that have been written but not yet committed to the messaging system for the destination.	destPendingMessages
Pending bytes	Shows the number of bytes for the pending messages.	destPendingBytes
De-logged	Shows the number of transactions and events read from the SLOG system file for the destination.	destItemsDeloggedSLOG
Logged	Shows the number of transactions and events written to the SLOG system file for the destination.	destItemsLoggedSLOG
On SLOG	Shows the number of transactions and events currently residing on the SLOG system file for the destination.	destItemsOnSLOG
	destLastMessagingCommitTime minus destLastTransCommitTime	destLastItemDurationTime
	destReplicatedTrans divided by intervalLength	destItemsPerSecond
	Java Date instance containing URBS timeTransProcessedStart equivalent	timestamp
	Java Date instance containing URBS timeTransCommitEnd equivalent	timestampEnd
	interval length in milliseconds timestampEnd minus timestamp	intervalLength

Adabas Destination Statistics

Sample output

```

ADAFI15
Output items           :           54,710
  Pending items       :              0
Item time at source   : 2017-10-17 19:12:36
BT                    :              0
ET                    :           21,500
ET time at destination : 2017-10-17 19:14:07
Input: DBID 28857 File 15 Target: DBID 28858 File 15
A1                    :           6,600
E1                    :           7,450
HI                    :              0
N1                    :              1
N2                    :           7,450
Security              :              1
S4                    :              1
UTI                   :              0

```

Labels and values shown for Adabas destination statistics

The Adabas destination statistics section is repeated for each Adabas destination defined to the Event Replicator Server.

Label	Value Description	Elasticsearch Field
	ReptorStatisticDestinationAdabasFile	type
	URBS reptor status ("RSTA")	reptorStatus
	Target ID of originating Event Replicator Server	targetID
	Nucleus ID of originating Event Replicator Server	nucleusID
ADAFI15 (sample)	Is the Adabas destination name in the above sample.	destinationName
Output items	Shows the number of output transactions and events (i.e. URBS) for the destination.	destOutputItems
Pending items	Shows the number of in-flight output transactions and events for the destination.	destPendingItems
Item time at source	Is the commit time of last processed transaction or event for the destination.	destItemTimeAtSource
BT	Shows the number of BT (backout transaction) commands successfully issued for the destination.	destBTCommands
ET	Shows the number of ET (end transaction) commands successfully issued for the destination.	destETCommands
ET time at destination	Is the time of the last successful ET command that was issued for the destination.	destETTimeAtDestination
	Items de-logged from the SLOG system file.	destItemsDeloggedSLOG
	Items logged to the SLOG system file.	destItemsLoggedSLOG
	Items on the SLOG system file.	destItemsOnSLOG
	destLastETTime minus destLastTransCommitTime	destLastTransactionDurationTime
	destReplicatedTrans divided by intervalLength	destTransactionsPerSecond
	Java Date instance containing URBS timeTransProcessedStart equivalent	timestamp
	Java Date instance containing URBS timeTransCommitEnd equivalent	timestampEnd
	interval length in milliseconds timestampEnd minus timestamp	intervalLength
Input: DBID <i>nnnnn File xx</i> Target: DBID <i>nnnnn File xx</i>	Is the input database/file and the target database/file combination.	destInputDBID destInputFNR destTargetDBID destTargetFNR

Label	Value Description	Elasticsearch Field
		destNumberErrors
A1	Shows the number of A1 (update) commands issued for the destination.	destUpdateCommands
E1	Shows the number of E1 (delete) commands issued for the destination.	destDeleteCommands
HI	Shows the number of HI (hold record) commands issued for the destination.	destHoldCommands
N1	Shows the number of N1 (add) commands issued for the destination.	destInsertCommands
N2	Shows the number of N2 (add) commands issued for the destination.	destInsertISNCommands
Security	Shows the number of Adabas security operations replicated to the target database/file.	destSecurityFunctions
S4	Shows the number of S4 (find) commands issued for the destination.	destFindCommands
UTI	Shows the number of Adabas utility operations replicated to the target database/file.	destUtilityFunctions
	Replication method ('ISN' / 'KEY')	destReplicationMethod

Deleted Destination Statistics

Sample output

```
Deleted destinations
Output items      :          1
Messages         :          1
Bytes            :         110
Commits          :          1
De-logged        :          1
Logged           :          1 ↵
```

Labels and values shown for deleted destination statistics

Label	Value Description
Output items	Shows the number of output transactions and events (i.e. URBS) for all deleted destinations.
Messages	Shows the number of output messages sent for all deleted destinations.
Bytes	Shows the number of bytes in the output messages.
Commits	Shows the number of messaging system commits for all deleted destinations.
De-logged	Shows the number of transactions and events read from the SLOG system file for all deleted destinations.

Label	Value Description
Logged	Shows the number of transactions and events written to the SLOG system file for all deleted destinations.

Subtask Statistics

Sample output

```

Output subtask items:
Main           :           0
  1           :          268
  2           :          101
  3           :          101

Output subtask message totals:
Main           :           0
  1           :           1
  2           :           1
  3           :           1
    
```

The output subtask items detail line is repeated for each output subtask defined to the Event Replicator Server. Each detail line shows the number of output transactions and events (i.e. URBS) processed for the subtask.

The output subtask message totals detail line is repeated for each output subtask defined to the Event Replicator Server. Each detail line shows the number of messages processed for the subtask.

Labels and values shown for subtask statistics

Label	Value Description	Elasticsearch Field
	ReptorStatisticOutputTask	type
	URBS reptor status ("RSTA")	reptorStatus
	Target ID of originating Event Replicator Server	targetID
	Nucleus ID of originating Event Replicator Server	nucleusID
Main (sample)	taskNumber (0 = main task)	taskNumber
items	Shows the number of output transactions and events (i.e. URBS) processed for the subtask.	items
message totals	Shows the number of messages processed for the subtask.	messagesSent
	Java Date instance containing URBS timeTransProcessedStart equivalent	timestamp
	Java Date instance containing URBS timeTransCommitEnd equivalent	timestampEnd
	interval length in milliseconds timestampEnd minus timestamp	intervalLength

Input Queue Statistics

Sample output

```

INPUTQ1
  Input items           :           54,721
  Pending items        :              0
  Item time at source   : 2017-10-17 19:10:45
  Messages             :           29,364
  Bytes                :        26,374,377
  Pending messages     :              1
  Bytes                :             110
  Commits              :           29,357
  Commit time at input queue : 2017-10-17 19:15:45
  Backouts             :              0 ↵

```

Labels and values shown for input queue statistics

The input queue statistics section is repeated for each input queue defined to the Event Replicator Server.

Label	Value Description	Elasticsearch Field
	ReptorStatisticInputQueue	type
	URBS reptor status ("RSTA")	reptorStatus
	Target ID of originating Event Replicator Server	targetID
	Nucleus ID of originating Event Replicator Server	nucleusID
INPUTQ1 (sample)	Is the input queue name in this example.	queueName
Input items	Shows the number of input transactions and events (i.e. URBS) for the input queue.	
Pending items	Shows the number of in-flight input transactions and events (i.e. URBS) for the input queue.	
Item time at source	Is the time of the last input transaction or event processed for the input queue.	
Messages	Shows the number of messages read and committed for the input queue.	queueMessagesReceived
Bytes	Shows the number of bytes in the above messages.	queueBytesReceived
Pending messages	Shows the number of messages read from the input queue but not yet committed.	queuePendingMessages
Bytes	Shows the number of bytes in the above messages.	queuePendingBytes
Commits	Shows the number of commits issued for the input queue.	queueMessagingCommits
Commit time at input queue	Is the time the last commit was issued for the input queue.	
Backouts	Shows the number of back outs issued for the input queue.	queueMessagingBackouts

Label	Value Description	Elasticsearch Field
	Java Date instance containing URBS timeTransProcessedStart equivalent	timestamp
	Java Date instance containing URBS timeTransCommitEnd equivalent	timestampEnd
	interval length in milliseconds timestampEnd minus timestamp	intervalLength

Deleted Input Queue Statistics

Sample output

```
Deleted input queues
Input items      :          1
Messages        :          1
Bytes           :         110
Commits         :          1
Backouts        :          0
```

Labels and values shown for deleted input queue statistics

Label	Value Description
Input items	Shows the number of input transactions and events (i.e. URBS) for all deleted input queues.
Messages	Shows the number of messages read and committed for all deleted input queues.
Bytes	Shows the number of bytes in the above messages.
Commits	Shows the number of commits issued for all deleted input queues.
Backouts	Shows the number of back outs issued for all deleted input queues.

Event Replicator Target Adapter Statistics

This section describes the various statistics (or metrics) sent from the Event Replicator Target Adapter and made available to the Replication Monitoring subsystem. The Event Replicator Target Adapter has always collected statistics and made them available for display using *Event Replicator Target Adapter Administration*. These statistics are described in the Event Replicator Target Adapter documentation at *Reviewing Event Replicator Target Adapter Statistics* in the Event Replicator Target Adapter Administrator's Guide.

The statistics collected and displayed by the Event Replicator Target Adapter and its Administration tool are now also sent to Elasticsearch and made available to Kibana dashboards and visualizations as part of Replication Monitoring. The "Category Type" in the Event Replicator Target Adapter

statistics documentation maps directly to the Elasticsearch fields "type" and "id". Some examples of this mapping are shown in the table below:

Category Type	Elasticsearch "type" value	Elasticsearch "id" value
/engine/...	engine	<i>engine id</i> e.g. "Engine-0"
/source/ <i>sourcename</i>	source	<i>sourcename</i> e.g. "HOST3-S1"
/jdbc/ <i>targetname</i>	jdbc	<i>targetname</i> e.g. "host05 SQL Server"
/subscription/ <i>subname</i>	subscription	<i>subname</i> e.g. "EAPSUB3"
/table/ <i>tablename</i>	table	<i>tablename</i> e.g. "Employees"
/heartbeat/ <i>destname</i>	heartbeat	<i>destname</i> e.g. "DEST2"
/heartbeat_etb/ <i>destname</i>	heartbeat_etb	<i>destname</i> e.g. "DEST2"
/heartbeat_processing/ <i>destname</i>	heartbeat_processing	<i>destname</i> e.g. "DEST2"
/heartbeat_slog/ <i>destname</i>	heartbeat_slog	<i>destname</i> e.g. "DEST2"
/latency/ <i>source.target.subname</i>	latency	<i>source.target.subname</i> e.g. "HOST3-S1.host05 SQL Server.EAPSUB3"

Please refer to *Reviewing Event Replicator Target Adapter Statistics* in the Event Replicator Target Adapter Administrator's Guide for further details.

Heartbeat

The statistics reported by the Event Replicator Target Adapter include a Heartbeat functionality that merits some additional description. A heartbeat is a special short message generated and sent down the pipe through the messaging system and into the Target Adapter. The purpose of this message is to ensure that all pieces of the replication system are working properly at a regular interval. The Target Adapter recognizes these heartbeat messages and reports statistics about the flow timing as part of Replication Monitoring.

The full set of statistics that includes all properties of the heartbeat is written at the conclusion of all heartbeat processing as an Elasticsearch data point with field "type" set to "heartbeat". Additional data points are written containing certain sub-types that comprise the full heartbeat. These are written with field "type" set to "heartbeat_etb", "heartbeat_processing", and "heartbeat_slog". All data points contain a "timestamp" field with different values appropriate for the sub-type. The final full heartbeat data point is written with "timestamp" set to the final time when all heartbeat statistics have been fully recognized and processed. Some important points in the heartbeat timeline include:

Timestamp Name	Timestamp Description
creationTime	Time the URBS heartbeat message was created on the mainframe by Event Replicator for Adabas.
readFromSlog	Time the URBS heartbeat message was read from the SLOG file on the mainframe by Event Replicator for Adabas. This timestamp will only have a value if the mainframe processing load was sufficiently high to cause the message to be written to the SLOG file for later processing.
infoTime	Time when Unit-of-Work status was obtained from EntireX Broker by Event Replicator Target Adapter.
artTime	Time when the heartbeat message is fully received and recognized by Event Replicator Target Adapter.

The table below shows all heartbeat "type" values with the "timestamp" value used and the list of Elasticsearch fields included in the data points for that sub-type.

Elasticsearch "type" value	Elasticsearch "timestamp" value	Elasticsearch fields
heartbeat_etb	infoTime	maxBytes maxMessages newestItem numberBytes numberMessages numberUows oldestItem
heartbeat_processing	infoTime	processingTime
heartbeat_slog	readFromSlog	slogTime
heartbeat	artTime	Complete field list described at <i>Reviewing Event Replicator Target Adapter Statistics</i> in the Event Replicator Target Adapter Administrator's Guide in table entry Heartbeat.

Please consult *Reviewing Event Replicator Target Adapter Statistics* in the Event Replicator Target Adapter Administrator's Guide for more details.

5 Replication Monitoring Events

This section describes the various events that can be sent by the Event Replicator Server on the mainframe. Events are represented in Elasticsearch as records with type "ReptorEvent". Each event is further described by the value of field "reptorStatus".

Each possible reptorStatus value is listed along with a description in the table below.

reptorStatus	Description
ASEC	Adabas security replication
AUTI	Adabas utility service replication
C5DA	Replication user data from C5 command
CLOS	Destination closed
CMPL	Initial-state processing completed
DBCO	Database connected
DBDI	Database disconnected
DBNP	Database not present
DCMP	Decompression error: AI, BI, FAI, FBI, or KEY
DEAC	Replication deactivated
DERR	Destination error
DFUL	Destination full
ERRO	Initial-state erroneous request rejected
INIT	Initial-state processing commenced
IQCL	Input Queue closed
IQOP	Input Queue opened
LODE	ADALOD LOAD ended
LODS	ADALOD LOAD started
LOST	Possibly lost replication data

reptorStatus	Description
OPEN	Destination opened
REAC	Replication reactivated
REFG	Refresh Globals
REFR	Resource refreshed
RESE	ADARES REGENERATE/BACKOUT ended
RESS	ADARES REGENERATE/BACKOUT started
REST	Reptor restart after abnormal end
RPLE	Replay process (ADARPL) ended
RPLS	Replay process (ADARPL) started
SAVE	ADASAV RESTORE ended
SAVS	ADASAV RESTORE started
SLDD	SLOG for DB-related input disabled
SLDF	SLOG for DB-related input turned off
SLDO	SLOG for DB-related input turned on
SLDR	SLOG for DB-related input resumed
SLDS	SLOG for DB-related input suspended
SLOF	SLOG turned off for destination
SLON	SLOG turned on for destination
STRT	Normal Reptor session start
SUBS	Subscription status information
TERM	Normal Reptor session termination
TRSP	Response to prior-transaction request
UPDE	ADALOD UPDATE ended
UPDS	ADALOD UPDATE started