For the most recent version of this document, visit our documentation website.
# Table of Contents

1 Relativity service bus ........................................................................................................... 4
   1.1 Relativity service bus infrastructure .............................................................................. 4
   1.2 Service bus installation overview .................................................................................. 6

2 Monitoring service bus performance .................................................................................... 9
   2.1 Setting up Windows Performance Monitor ..................................................................... 9
   2.2 Tracking metrics .............................................................................................................. 10
      2.2.1 Service bus message container ............................................................................. 10
      2.2.2 Service bus messaging broker nodes .................................................................... 11
      2.2.3 Service bus messaging gateway nodes .................................................................. 11
   2.3 Reporting functions in Windows Performance Monitor .................................................. 11

3 Relativity service bus entities ................................................................................................ 13
   3.1 Conversion ..................................................................................................................... 13
      3.1.1 conversionresponses topic ................................................................................... 13
      3.1.2 conversions_rp<Resource Pool Artifact ID> topic .................................................. 15
      3.1.3 resourcepoolstatus topic ..................................................................................... 17
      3.1.4 SignalR Scaleout .................................................................................................... 18
   3.2 Data Grid ....................................................................................................................... 19
      3.2.1 datagridingestion topic ........................................................................................ 19
      3.2.2 datagridverification topic ................................................................................... 19
      3.2.3 Migration_queue_#### topic ............................................................................... 20
      3.2.4 seeding_queue topic ............................................................................................ 20
   3.3 Processing .................................................................................................................... 21
      3.3.1 imagingresponses topic ....................................................................................... 21

4 Troubleshooting the Relativity service bus .......................................................................... 23
   4.1 General troubleshooting tasks for service bus ................................................................. 23
      4.1.1 Ensure all services are running ................................................................................. 23
      4.1.2 Verify that the FarmDNS is set ................................................................................. 23
      4.1.3 Verify instance settings are correct ........................................................................... 23
      4.1.4 Check that the Relativity namespace exists ............................................................. 24
4.2 Service bus connection errors .................................................................................................................. 24
  4.2.1 Remote name can't be resolved ............................................................................................................. 24
  4.2.2 Error occurs on send .............................................................................................................................. 24
  4.2.3 Value is out of range ............................................................................................................................ 24
  4.2.4 Connect refused by target machine ...................................................................................................... 25
  4.2.5 Server returns bad request error .......................................................................................................... 25
  4.2.6 Trust relationship couldn't be established ........................................................................................... 25
  4.2.7 Connection errors in a multiple host environment .............................................................................. 25
  4.2.8 Service bus returns a 401 error ............................................................................................................ 26
4.3 Service bus PowerShell cmdlets ................................................................................................................ 27
4.4 Checking logs in the Event Viewer ............................................................................................................. 27
4.5 Using Service Bus Explorer ...................................................................................................................... 28
  4.5.1 Prerequisites for Service Bus Explorer ................................................................................................. 28
  4.5.2 Connecting to a Relativity namespace .................................................................................................. 29
  4.5.3 Viewing entities in a namespace .......................................................................................................... 29
  4.5.4 Troubleshooting with Service Bus Explorer ......................................................................................... 30
4.6 Uninstalling the service bus farm ............................................................................................................... 31
4.7 Creating a namespace for the service bus ................................................................................................ 31
1 Relativity service bus

The Relativity service bus is a message delivery service that communicates information about agent jobs to different application components. This infrastructure feature supports this communication by routing messages between application components. For example, Relativity uses the service bus for submitting conversion jobs to agents and returning converted documents.

**Note:** Before installing or upgrading, you must install and configure Service Bus for Windows Server. Next, install or upgrade your primary SQL Server and the Relativity service bus. You can find information about Service Bus for Windows Server in the Pre-Installation guide. For installation instructions, see the Relativity Installation.

The Relativity service bus supports the following features:

- Guaranteed delivery of messages to ensure reliable communication between application components.
- High throughput performance for successful message delivery over the service bus.
- Support for arbitrary messages.
- High scalability to ensure that service bus can support an increasing number of resources added to your Relativity environment.
- Fault tolerance and high availability to guarantee that the service bus continues operating even when a component fails.

1.1 Relativity service bus infrastructure

The Relativity service bus is built on Service Bus for Windows Server, so it leverages the capabilities offered by this industry-standard software. You must install it on a node in a Service Bus for Windows Server farm. The Relativity installer updates the Instance setting table on your primary SQL Server with information about the location of the Relativity service bus that you provide during installation. It updates the following instance settings:

ServiceBusFullyQualifiedDomainName

<table>
<thead>
<tr>
<th>Table section</th>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>localhost</td>
<td>Specifies the fully-qualified domain name for the machine hosting Service Bus for Windows Server. The Relativity installer automatically sets this value during an installation or upgrade based on the inputs in the RelativityResponse.txt file. For more information, see the Relativity Installation Guide.</td>
</tr>
</tbody>
</table>

**Note:** The value for this setting should match the FarmDNS output from the Get-SBFarm command available in the Service Bus PowerShell utility. For more information, see [Service bus PowerShell cmdlets on page 27](#).

ServiceBusHttpPort
Table section | Relativity.ServiceBus
---|---
Value | 9455
Description | Specifies the HTTP port for Service Bus for Windows Server. The Relativity installer automatically sets this value during an installation or upgrade based on the inputs in the RelativityResponse.txt file. For more information, see the Relativity Installation Guide.

**Note:** The value for this setting should match the HttpsPort output from the Get-SBFarm command available in the Service Bus PowerShell utility. For more information, see Service bus PowerShell cmdlets on page 27.

ServiceBusTcpPort

Table section | Relativity.ServiceBus
---|---
Value | 9454
Description | Specifies the TCP port for Service Bus for Windows Server. The Relativity installer automatically sets this value during an installation or upgrade based on the inputs in the RelativityResponse.txt file. For more information, see the Relativity Installation Guide.

**Note:** The value for this setting should match the TcpPort output from the Get-SBFarm command available in the Service Bus PowerShell utility. For more information, see Service bus PowerShell cmdlets on page 27.

SharedAccessKey

Table section | Relativity.ServiceBus
---|---
Value | Default value varies by environment.
Description | Specifies the value of the shared access key / authorization rule used when authenticating with service bus.

The web and agent servers request information about the location of the Relativity service bus from the primary SQL Server. The Relativity service bus then facilitates communication between application components by sending and receiving messages. A typical Relativity installation requires only a single node in a farm. You can optionally install Service Bus for Windows Server on multiple hosts that you add to your service bus farm. The following diagram illustrates how the Relativity service bus integrates with your environment’s infrastructure in a typical installation.
1.2 Service bus installation overview

To install the Relativity service bus, you must first install and configure Service Bus for Windows Server. You next run the Relativity installer to add the Relativity service bus to your environment and complete other related tasks. The following diagram illustrates the installation process for a typical Relativity installation for service bus. You can optionally install the Service Bus for Windows Server on multiple hosts.
Use the following workflow to install the Relativity service bus in your environment:

1. For a typical installation, install Service Bus for Windows Server on the machine that you want included in the service bus farm. For a multiple host installation, install Service Bus for Windows Server on the servers or VMs for this purpose. The farm requires that you add an odd number of nodes, but you shouldn’t exceed a maximum of five nodes. For information about installing and configuring the service bus, see the Pre-Installation guide.

2. Set up a new service bus farm on a machine where you installed Service Bus for Windows Server. After completing this process, you can optionally add multiple hosts to the farm. For more information, see the Pre-Installation guide.
3. Run the Relativity installer to install or upgrade your primary SQL Server and any necessary distributed servers. For more information, see the following guides:
   - **New installation** - see the Relativity Installation guide.
   - **Upgrade** - see the Relativity Upgrade guide.

   **Note:** The Relativity Installation and Relativity Upgrade guides include information for troubleshooting installation issues for service bus. For general troubleshooting information, see [Troubleshooting the Relativity service bus on page 23](#).

4. Install or upgrade the Relativity service bus server. Ensure that the Relativity service bus server is a node in the Service Bus for Windows Server farm.

   **Note:** You need to install the Relativity service bus on only a single node or host in the farm. If you have a multiple host environment, you don’t need to install it on each node.

5. Install or upgrade the agent server.
6. Install or upgrade the web server.
7. Install or upgrade other servers used in your environment. For example, you might install the worker manager or Analytics server depending on your organization’s needs.
2 Monitoring service bus performance

You can use the Windows Performance Monitor to check the health of your service bus. This tool provides counters that monitor the behavior of various service bus components. Additionally, it provides data collection sets that you can use to aggregate the metrics collected by the configured counters and output reports on them. For more information about service bus health, see Relativity service bus entities on page 13.

- Relativity service bus entities on page 13

2.1 Setting up Windows Performance Monitor

Use the following steps to set up Windows Performance Monitor in your environment:

1. Log in to a server where you want to run the Performance Monitor.
2. Right-click on Start > Search, and locate the Performance Monitor.
3. Open the Performance Monitor, and expand the Monitoring Tools folder.
4. In the Monitoring Tools folder, click Performance Monitor to display the graphical interface.
5. Click + to display the Add Counters window.
6. To point the Performance Monitor to a computer in your environment, enter its Fully Qualified Domain Name (FQDN) or IP address in the Select counters from computer box.

![Add Counters](image)

7. Select the counters that you want to monitor from the list box. See Tracking metrics on the next page.
8. Select the service bus nodes that you want to monitor in the **Instances of select object** box.

![Add Counters](image)

### 2.2 Tracking metrics

The Windows Performance Monitor includes counters that monitor the performance of different components of the service bus. The following list includes key components of the service bus and the counters that are most helpful for monitoring them. For detailed information about these counters, see Service Bus for Windows Server Performance Counters ([https://msdn.microsoft.com/en-us/library/dn441380.aspx](https://msdn.microsoft.com/en-us/library/dn441380.aspx)).

#### 2.2.1 Service bus message container


The values for the following metrics should remain consistent. Additionally, normal values vary based on your environment’s hardware and network topology. When the values for these metrics are high, applications using the service bus may experience slower performance. For example, conversion requests may take longer to process. Monitor the following list of performance counters for the message container:
- Send Message Latency
- Receive Message Latency
- Complete message latency

### 2.2.2 Service bus messaging broker nodes

You can monitor performance counters scoped to the messaging broker nodes. For more information, see [Service Bus for Windows Server Messaging Broker Node](https://msdn.microsoft.com/en-us/library/dn441380.aspx).

If any of the following metrics have values outside of the normal range, review the current load on your system. Determine if your service bus has an unusual number of requests by checking the values for the Incoming Messages/sec and Outgoing Messages/sec metrics available in the Performance Monitor. Consider increasing the resources on a node with a high request volume, or increasing the number of nodes in your system. Monitor the following list of performance counters for messaging broker nodes:

- Cpu Throttle Level - the normal range for this counter is between 0 and 1.
- Memory Throttle Level - the normal range for this counter is between 0 and 2.

### 2.2.3 Service bus messaging gateway nodes

Monitor the following list of performance counters scoped to the messaging gateway nodes. For more information, see [Service Bus for Windows Server Messaging Gateway Node](https://msdn.microsoft.com/en-us/library/dn441380.aspx).

- Sql Operation Latency - the normal latency shouldn't exceed 100 milliseconds. Latency that exceeds the recommend range indicates that the SQL server has slowed down the processing of service bus requests. Check the health of your SQL server to determine whether your environment configuration needs updating.
- Sql Exceptions/sec - the normal value for this metric varies based on the size of your environment and the current load that it is handling. However, it shouldn't rapidly increase. An increased number of exceptions indicates that the SQL server is failing an increasing number of service bus requests. Verify that the SQL server connection is configured properly for the service bus. In addition, check the health of your SQL server to determine whether your environment configuration needs updating.
- Failed Operations/sec - the normal value for this metric varies based on the size of your environment and the current load that it is handling. However, it shouldn't rapidly increase. If this value increases, check the event viewer for information about service bus failovers.

### 2.3 Reporting functions in Windows Performance Monitor

The Windows Performance Monitor provides you with the ability to create entities called data collection sets. You can configure a set of configured counters into data collection set, and then log performance metrics on the set. You have the option to configure the data collection for specific timeframes. Additionally, you can configure the data collection set to generate notifications when counters exceed specified thresholds.

For more information about data collection sets, see the following pages on the Microsoft site:

When the data collection is completed, the data collector set performs various tasks, which are output to the event viewer. You can also schedule other Windows tasks to meet your current needs.

To send event viewer logs by email, you can use various Powershell scripts that point to an SMTP server. For information about setting up email notifications, see Send-MailMessage ([https://msdn.microsoft.com/en-us/powershell/reference/5.1/microsoft.powershell.utility/send-mailmessage](https://msdn.microsoft.com/en-us/powershell/reference/5.1/microsoft.powershell.utility/send-mailmessage)).

You may also want to consider gathering data with the Perfmon plugin from New Relic. For more information, see Perfmon ([https://newrelic.com/plugins/52projects/115](https://newrelic.com/plugins/52projects/115)).
3 Relativity service bus entities

The Relativity service bus relies on specialized entities to support message delivery between various application features. Each application feature uses its own set of entities to communicate over the service bus. This page provides information about each Relativity feature that uses the service bus, and its related entities. These entity types can be divided into the following categories:

- **Queue** - the messages on a queue are persisted until they are delivered to a client. They are delivered in the order that they are added to the queue. Each message is delivered once to a single client.

- **Topic** - a topic supports message distribution to multiple subscriptions. When a message is sent to a topic, the filters or rules for each associated subscription are evaluated. If the subscription doesn’t filter out the message, then a copy of it is delivered to the subscription. Messages aren’t directly persisted on or received from a topic.

- **Subscription** - the messages on a subscription are persisted until delivered to a client. The messages aren’t sent directly to a subscription. Instead, they are sent to the associated topic. By default, copies of all messages sent to the associated topic are persisted to the subscription. However, you can set up filters or rules so that the subscription only receives messages with certain property values. Messages are delivered in the order that they are added to the queue. After a message is placed on a subscription, the copy is delivered once to single a client.

You can use Service Bus Explorer to view the current status of entities on the Relativity service bus. For more information, see Using Service Bus Explorer on page 28.

3.1 Conversion

Conversion uses the following entities on the Relativity service bus.

3.1.1 conversionresponses topic

This section includes information about the conversionresponses topic and its subscriptions.

Click here for topic and subscription information

3.1.1.1 conversionresponses

<table>
<thead>
<tr>
<th>Entity name</th>
<th>conversionresponses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Topic</td>
</tr>
<tr>
<td>Entity description</td>
<td>Each CAPI in a Relativity environment has a corresponding subscription on the conversionresponses topic with any completed conversions.</td>
</tr>
<tr>
<td>Message description</td>
<td>The Service Bus message for conversion contains a dynamic number of variables which may include the following: the file path of the completed conversion; metrics; conversion type; ids; and other necessary components to conversion.</td>
</tr>
</tbody>
</table>
Processes - write to entity

The conversion agents write messages to the CAPI-<MD5> subscriptions on this topic.

Processes - read from entity

The CAPI reads messages from the CAPI-<MD5> subscriptions on this topic.

Troubleshooting tips

If there aren't any messages, either they were successfully picked up by the CAPI, or the conversion agents aren't completing requests. Try restarting the conversion agents. If no messages are picked up by the CAPI, you may need to recycle the CAPI application pool.

3.1.1.2 ConversionCompleteAgent_Priority<#>

Entity name
Add ConversionCompleteAgent_Priority<#>

Entity type
Subscription

Entity description
Each Conversion Priority in a Relativity environment has a corresponding subscription on the conversionresponses topic for use by the Document Viewer Service.

Message description
The Service Bus message for conversion contains a dynamic number of variables which may include the following: file paths of the native or image; conversion type; ids; and other necessary components to conversion.

Processes - write to entity

The Service Bus writes messages to this subscription.

Processes - read from entity

The SignalR backplane reads messages from this subscription.

Troubleshooting tips

If there aren't any messages, either they were successfully picked up by the conversion complete agents, or the SignalR backplane isn't completing requests. If no messages are picked up by the conversion complete agents, you may need to restart the conversion complete agents.

3.1.1.3 CAPI-<MD5 Hash of a CAPI URL/Filepath> subscription

Entity name
CAPI-<MD5 Hash of a CAPI URL/Filepath>

Entity type
Subscription

Entity description
Each CAPI in a Relativity environment has a corresponding subscription on the conversionresponses topic with any completed conversions. The name of each subscription is CAPI appended with the MD5 Hash of the corresponding CAPI URL/Filepath.

Message description
The Service Bus message for conversion contains a dynamic number of variables which may include the following: file paths of the completed con-
3.1.1.4 CAPI-<Web Server/Machine Name>-Mass subscription

<table>
<thead>
<tr>
<th>Entity name</th>
<th>CAPI-&lt;Web Server/Machine Name&gt;-Mass</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Subscription</td>
</tr>
<tr>
<td>Entity description</td>
<td>Each CAPI in a Relativity environment has a corresponding subscription on the conversionresponses topic with any completed mass conversions. The name of each subscription is CAPI appended with the web server/machine name and mass. This subscription only picks up priority three mass conversions.</td>
</tr>
<tr>
<td>Message description</td>
<td>The Service Bus message for conversion contains a dynamic number of variables which may include the following: file paths of the completed conversion; metrics; conversion type; ids; and other necessary components to conversion.</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>The conversion agents write messages to these subscriptions.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>The CAPI reads messages from these subscriptions.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>If there aren't any messages, either they were successfully picked up by the CAPI, or the conversion agents aren't completing requests. Try restarting the conversion agents. If no messages are picked up by the CAPI, you may need to recycle the CAPI application pool.</td>
</tr>
</tbody>
</table>

3.1.2 conversions_rp<Resource Pool Artifact ID> topic

This section includes information about the conversions_rp<Resource Pool Artifact ID> topic and its subscriptions.
### 3.1.2.1 conversions_rp<Resource Pool Artifact ID> topic

<table>
<thead>
<tr>
<th>Entity name</th>
<th>conversions_rp&lt;Resource Pool Artifact ID&gt;, example &quot;conversions_rp1015400&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Topic</td>
</tr>
<tr>
<td>Entity description</td>
<td>Each Conversion Priority in a Relativity environment has a corresponding subscription on the <code>conversions_rp</code> topic with any completed conversions.</td>
</tr>
<tr>
<td>Message description</td>
<td>The Service Bus message for conversion contains a dynamic number of variables which may include the following: file paths of the native or image; conversion type; ids; and other necessary components to conversion.</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>The CAPI writes messages to the <code>ConversionAgent_Priority</code> subscriptions.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>The Conversion Agents read messages from the <code>ConversionAgent_Priority</code> subscriptions.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>If there aren't any messages, either they were successfully picked up by the conversion agents, or the CAPI isn't completing requests. Try recycling the CAPI. If no messages are picked up by the conversion agents, you may need to restart the conversion agents.</td>
</tr>
</tbody>
</table>

### 3.1.2.2 ConversionAgent_Priority1 subscription

<table>
<thead>
<tr>
<th>Entity name</th>
<th>ConversionAgent_Priority1</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Subscription</td>
</tr>
<tr>
<td>Entity description</td>
<td>This subscription corresponds to requests of doc-to-doc conversions.</td>
</tr>
<tr>
<td>Message description</td>
<td>The Service Bus message for conversion contains a dynamic number of variables which may include the following: file path of the native or image; conversion type; ids; and other necessary components to conversion.</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>The CAPI writes messages to this subscription.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>The conversion agents read messages from this subscription.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>If there aren't any messages, either they were successfully picked up by the conversion agents, or the CAPI isn't completing requests. Try recycling the CAPI. If no messages are picked up by the conversion agents, you may need to restart the conversion agents.</td>
</tr>
</tbody>
</table>
3.1.2.3 ConversionAgent_Priority2 subscription

<table>
<thead>
<tr>
<th>Entity name</th>
<th>ConversionAgent_Priority2</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Subscription</td>
</tr>
<tr>
<td>Entity description</td>
<td>This subscription corresponds to requests of pre-convert Conversions</td>
</tr>
<tr>
<td>Message description</td>
<td>The Service Bus message for conversion contains a dynamic number of variables which may include the following: file path of the native or image; conversion type; ids; and other necessary components to conversion.</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>The conversion agents read messages from this subscription.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>The CAPI writes messages to this subscription.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>If there aren't any messages, either they were successfully picked up by the conversion agents, or the CAPI isn't completing requests. Try recycling the CAPI. If no messages picked up by the conversion agents, you may need to restart the conversion agents.</td>
</tr>
</tbody>
</table>

3.1.2.4 ConversionAgent_Priority3 subscription

<table>
<thead>
<tr>
<th>Entity name</th>
<th>ConversionAgent_Priority3</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Subscription</td>
</tr>
<tr>
<td>Entity description</td>
<td>This subscription corresponds to requests of mass convert conversions</td>
</tr>
<tr>
<td>Message description</td>
<td>The Service Bus message for conversion contains a dynamic number of variables which may include the following: file path of the native or image; conversion type; ids; and other necessary components to conversion.</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>The CAPI writes messages to this subscription.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>The Conversion Agents read messages from this subscription.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>If there aren't any messages, either they were successfully picked up by the conversion agents, or the CAPI isn't completing requests. Try recycling the CAPI. If no messages are picked up by the conversion agents, you may need to restart the conversion agents.</td>
</tr>
</tbody>
</table>

3.1.3 resourcepoolstatus topic

This section includes information about the resourcepoolstatus topic and its subscription.
3.1.3.1 resourcepoolstatus topic

<table>
<thead>
<tr>
<th>Entity name</th>
<th>resourcepoolstatus</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Topic</td>
</tr>
<tr>
<td>Entity description</td>
<td>Whenever agent servers are added or removed from a resource pool group, a message is sent to each subscription. Each conversion agent has a subscription on this topic in order to find out what agent server to be a part of.</td>
</tr>
<tr>
<td>Message description</td>
<td>The Service Bus message contains a list of agent servers for each conversion agent.</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>Relativity Web writes messages to this entity.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>Conversion agents read messages from this entity.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>N/A</td>
</tr>
</tbody>
</table>

3.1.3.2 resourcePoolStatusMonitor_<Conversion Agent Artifact ID> subscription

<table>
<thead>
<tr>
<th>Entity name</th>
<th>resourcePoolStatusMonitor_&lt;Conversion Agent Artifact ID&gt;, example &quot;resourcePoolStatusMonitor_1035454&quot;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Subscription</td>
</tr>
<tr>
<td>Entity description</td>
<td>Whenever agent servers are added or removed from a resource pool group, a message is sent to this subscription. Each conversion agent reads this subscription to find out which agent server to be a part of. The name is comprised of resourcePoolStatusMonitor appended with the conversion agent artifact id.</td>
</tr>
<tr>
<td>Message description</td>
<td>The Service Bus message contains a list of Agent Servers for each Conversion Agent.</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>Relativity Web writes to this entity.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>Conversion Agents reads from this entity.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>N/A</td>
</tr>
</tbody>
</table>

3.1.4 SignalR Scaleout

This section includes information about the conversions_rp<Resource Pool Artifact ID> topic and its subscriptions.
3.1.4.1 Signalr_topic_dvs_<#>

<table>
<thead>
<tr>
<th>Entity name</th>
<th>Signalr_topic_dvs_&lt;#&gt;</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Topic</td>
</tr>
<tr>
<td>Entity description</td>
<td>This topic corresponds to the SignalR backplane for the Document Viewer Service. Every hub that is connected to this Service Bus using the SignalR backplane with the same topic prefix will make a subscription to these topics.</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Message description</th>
<th>N/A</th>
</tr>
</thead>
<tbody>
<tr>
<td>Processes - write to entity</td>
<td>SignalR hubs and Conversion Complete Agents</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>SignalR hubs</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>N/A</td>
</tr>
</tbody>
</table>

3.2 Data Grid

Data Grid uses the following entities on the Relativity service bus.

3.2.1 datagridingestion topic

This section includes information about the <Your Entity Name> topic and its subscription.

3.2.1.1 datagridingestion topic

<table>
<thead>
<tr>
<th>Entity name</th>
<th>datagridingestion</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Topic</td>
</tr>
<tr>
<td>Entity description</td>
<td>The job queue for Lucene index building.</td>
</tr>
<tr>
<td>Message description</td>
<td>Index Document Job</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>Import writes to this entity.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>The Data Grid Worker Agent reads from this entity.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>Verify that Data Grid Worker Agents exist.</td>
</tr>
</tbody>
</table>

3.2.2 datagridverification topic

This section includes information about the <Your Entity Name> topic and its subscription.
3.2.2.1 datagridverification topic

<table>
<thead>
<tr>
<th>Entity name</th>
<th>datagridverification</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Topic</td>
</tr>
<tr>
<td>Entity description</td>
<td>The job queue for Lucene index building.</td>
</tr>
<tr>
<td>Message description</td>
<td>Verify Document Job</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>The Data Grid Worker Agent writes to this entity.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>The Data Grid Worker Agent reads from this entity.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>Verify that Data Grid Worker Agents exist.</td>
</tr>
</tbody>
</table>

3.2.3 Migration_queue_####
This section includes information about the <Your Entity Name> topic and its subscription.

3.2.3.1 Migration_queue_####

<table>
<thead>
<tr>
<th>Entity name</th>
<th>Migration_queue_####</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Queue</td>
</tr>
<tr>
<td>Entity description</td>
<td>The job queue for Text Migration per workspace.</td>
</tr>
<tr>
<td>Message description</td>
<td>Text Migration Document Job</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>The Data Grid Migration Manager writes to this entity.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>The Data Grid Migration Worker reads from this entity.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>Verify that Data Grid Migration Workers exist.</td>
</tr>
</tbody>
</table>

3.2.4 seeding_queue
This section includes information about the <Your Entity Name> topic and its subscription.

3.2.4.1 Seeding_queue

<table>
<thead>
<tr>
<th>Entity name</th>
<th>Seeding_queue</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Queue</td>
</tr>
<tr>
<td>Entity description</td>
<td>The job queue for Text Migration per workspace.</td>
</tr>
</tbody>
</table>
3.3 Processing

Processing uses the following entities on the Relativity service bus.

3.3.1 imagingresponses topic

This section includes information about the imagingresponses topic and its subscription.

3.3.1.1 imagingresponses topic

<table>
<thead>
<tr>
<th>Entity name</th>
<th>imagingresponses</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity type</td>
<td>Topic</td>
</tr>
<tr>
<td>Entity description</td>
<td>Each CAPI in a Relativity environment has a corresponding subscription on the imagingresponses topic with any completed images.</td>
</tr>
<tr>
<td>Message description</td>
<td>The Service Bus message for imaging contains a dynamic number of variables which may include the following: the file path of the completed images; page count; identified type; imaging set id; document id; workspace id; error messages; and other necessary components to images.</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>The Invariant worker servers write messages to the CAPI-&lt;MachineName&gt; subscriptions on this topic.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>The CAPI reads messages from the CAPI-&lt;MachineName&gt; subscriptions on this topic.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>If there aren't any messages, either they were successfully picked up by the CAPI, or the Invariant worker servers aren't completing requests. On the Invariant side, check that jobs are being picked up and completed, and if so, check the tracking log to look for errors communicating with service bus. If no messages are picked up by the CAPI, you may need to recycle the CAPI application pool.</td>
</tr>
</tbody>
</table>

3.3.1.2 CAPI-<Web Server/Machine Name>-Mass subscription

<p>| Entity name | CAPI-&lt;Web Server/Machine Name&gt;-Mass |</p>
<table>
<thead>
<tr>
<th>Entity type</th>
<th>Subscription</th>
</tr>
</thead>
<tbody>
<tr>
<td>Entity description</td>
<td>Each CAPI in a Relativity environment has a corresponding subscription on the imagingresponses topic with any completed images. The name of each subscription is CAPI appended with the web server/machine name and mass. This subscription only picks up imaging responses that were from an imaging set or a mass imaging request.</td>
</tr>
<tr>
<td>Message description</td>
<td>The Service Bus message for imaging contains a dynamic number of variables which may include the following: the file path of the completed images; page count; identified type; imaging set id; document id; workspace id; error messages; and other necessary components to images.</td>
</tr>
<tr>
<td>Processes - write to entity</td>
<td>The Invariant worker servers write messages to these subscriptions.</td>
</tr>
<tr>
<td>Processes - read from entity</td>
<td>The CAPI reads messages from these subscriptions.</td>
</tr>
<tr>
<td>Troubleshooting tips</td>
<td>If there aren't any messages, either they were successfully picked up by the CAPI, or the Invariant worker servers aren't completing requests. On the Invariant side, check that jobs are being picked up and completed, and if so, check the tracking log to look for errors communicating with service bus. If no messages are picked up by the CAPI, you may need to recycle the CAPI application pool.</td>
</tr>
</tbody>
</table>
4 Troubleshooting the Relativity service bus

You can troubleshoot the Relativity service bus by viewing the log files and by running diagnostic commands that the Service Bus PowerShell utility provides. Additionally, you can use the Service Bus Explorer to view information about the topics and subscriptions used for distributing messages.

Note: The Relativity Installation and Relativity Upgrade guides include information for troubleshooting installation issues for service bus.

4.1 General troubleshooting tasks for service bus

Use these general troubleshooting tasks to resolve issues with the Relativity service bus. For more information about troubleshooting with PowerShell commands, see Service bus PowerShell cmdlets on page 27.

4.1.1 Ensure all services are running

Complete these steps as necessary:

- To ensure that all services are running, execute the Get-SBFarmStatus command.
- If you find that the services aren't running, execute the Stop-SBFarm, and then the Start-SBFarm command.
- If the services still don't start, see Troubleshooting the service bus farm in the Pre-Installation guide.

4.1.2 Verify that the FarmDNS is set

To verify that the FarmDNS is set, execute the Get-SBFarm command.

If you find that the FarmDNS isn't set, execute the following commands:

```
Stop-SBFarm
Set-SBFarm -FarmDns 'YOUR_DNS'
Update-SBHost
Start-SBFarm
```

4.1.3 Verify instance settings are correct

Verify that the instance settings for the service bus match their respective values from the farm. Execute the Get-SBFarm command. Compare the values returned by this command to those in the following instance settings:

- ServiceBusFullyQualifiedDomainName
- ServiceBusHttpPort
- ServiceBusTcpPort

Note: For more information, see the Instance Setting guide.
4.1.4 Check that the Relativity namespace exists

To check that the Relativity namespace exists, execute the Get-SBNamespace command.

If you find that the namespace doesn't exist, complete the following steps:

1. On your machine, click **Start > Control Panel > Programs > Programs and Features**.
2. Check for Relativity service bus in the program list. If you find the Relativity service bus, uninstall it.
3. Run the Relativity installer with the INSTALLSERVICEBUS=1. For more information, see the Relativity Installation or Relativity Upgrade guide.
4. Verify that the namespace called Relativity exists.

4.2 Service bus connection errors

Review the following list of errors and resolutions to troubleshoot your service bus connection. For more information about troubleshooting with PowerShell commands, see [Service bus PowerShell cmdlets on page 27](#).

4.2.1 Remote name can't be resolved

| Error Message | Microsoft.ServiceBus.Messaging.MessagingCommunicationException: The remote name could not be resolved: 'incorrect.domain.name'. |
| Resolution | Verify that the ServiceBusFullyQualifiedDomainName instance setting contains the correct value. Ensure that it matches the DNS for the service bus farm and that it is a fully qualified domain name. Use the Get-SBFarm cmdlet to check this configuration. |

4.2.2 Error occurs on send

| Error Message | Microsoft.ServiceBus.Messaging.MessagingException: The underlying connection was closed: An unexpected error occurred on a send. ---> System.Net.WebException: The underlying connection was closed: An unexpected error occurred on a send. ---> System.IO.IOException: Unable to read data from the transport connection: An existing connection was forcibly closed by the remote host. |
| Resolution | Verify that the ServiceBusHttpPort and ServiceBusTcpPort instance settings contain the correct values. They must match your service bus configuration. Use the Get-SBFarm cmdlet to check this configuration. Contact Client Services. |

4.2.3 Value is out of range

| Error Message | System.ArgumentOutOfRangeException: Specified argument was out of the range of valid values. Parameter name: value at System.UriBuilder.set_Port(Int32 value). |
Resolution: Verify that the ServiceBusHttpPort and ServiceBusTcpPort instance settings contain the correct values. They must match your service bus configuration. Use the Get-SBFarm cmdlet to check this configuration.

4.2.4 Connect refused by target machine


Resolution: Verify that the service bus farm is running by using the Get-SBFarmStatus cmdlet. You can use the Start-SBFarm cmdlet to start the services, if necessary.

4.2.5 Server returns bad request error


Resolution: Verify that the service bus is configured with the namespace for Relativity used when configuring the Service Bus for Windows Server. Use the Get-SBNamespace cmdlet to retrieve this information. For configuration information, see the Pre-Installation guide.

4.2.6 Trust relationship couldn't be established


Resolution: Verify that you have the required certificates for the Service Bus for Windows Server on your web and agent servers. For more information, see the Pre-Installation guide.

4.2.7 Connection errors in a multiple host environment

When connection errors occur in a multiple host environment, the ServiceBusHosts table on the EDDS database may not include the fully qualified domain name (FQDN) for each of the service bus hosts. For
more information, see Service bus PowerShell cmdlets in the Relativity service Bus guide.

Use the following procedure to verify the FQDN in the ServiceBusHosts table:

1. Obtain FQDN for each of the service bus nodes. If you don't know the FQDNs, run the `Get-SBFarm` command and copy the FQDNs for the hosts from the output.

2. Log in to Microsoft SQL Server Management Studio on your primary SQL Server.

3. Run the following SQL command to obtain the list of hosts added to the ServiceBusHosts table:

   ```
   Select * From EDDS.eddsdbo.ServiceBusHosts
   ```

4. Verify that the entries returned from this command match the FQDNs of your service bus nodes obtained in step 1 or contain only a single value that matches the FarmDns. Complete the following tasks if the entries don't match:

   - **Missing an FQDN** - insert a row with the FQDN into the table. See the following sample command:

     ```
     INSERT INTO EDDS.eddsdbo.ServiceBusHosts Values('<FQDN of the host>')
     ```

   - **Incorrect host name** - execute an UPDATE statement to add the correct FQDN for the host.

   - **Extraneous host name** - execute a DELETE statement to remove the names of hosts not currently used in your environment.

### 4.2.8 Service bus returns a 401 error

<table>
<thead>
<tr>
<th>Error Message</th>
<th>Resolution</th>
</tr>
</thead>
<tbody>
<tr>
<td>The service bus returns the HTTP Error 401.</td>
<td>To resolve this issue, use the following steps to verify that your SharedAccessKeyName and SharedAccessKey instance settings are correct.</td>
</tr>
</tbody>
</table>

1. Open the Service Bus PowerShell tool.

2. Execute the following cmdlet:

   ```
   Get-SBAuthorizationRule -NamespaceName Relativity
   ```

3. Verify that you have an authorization rule named Relativity. If it doesn’t exist, complete the instructions in [Creating a namespace for the service bus](#) on page 31.

4. Verify that the authorization rule has send, manage, and listen rights. See the output from step 2.

5. Verify the SharedAccessKeyName instance setting is set to Relativity.

6. Verify that the value for the SharedAccessKey instance setting matches the output from step 2.

For more information, see the Pre-Installation guide.
4.3 Service bus PowerShell cmdlets


**Note:** We don't recommend using these commands to change any Authorization rules at this time.

Log in to the machine where you installed the Relativity service bus. Run the PowerShell commands as necessary. The following list includes commands that are useful for debugging:

- Get-SBFarm - retrieves the configuration for the Service Bus for Windows Server farm.
- Get-SBFarmStatus - outputs a list of services on each service bus server in a farm. It also retrieves their statuses, such as Running, Stopped, Start Pending, Stop Pending, and others. Use this cmdlet to quickly get a list of services that are running on different servers.
- Stop-SBFarm - stops all services on all servers in a farm. You need to stop the services to make any configuration changes to a farm or the namespace.
- Start-SBFarm - starts all services on all servers in the farm. You can use this cmdlet to restart an entire service bus farm.
- Stop-SBHost - stops all services on the server where you are currently running this cmdlet.
- Start-SBHost - starts all services on the server where you are currently running this cmdlet. You can use this cmdlet to restart specific service bus servers.

4.4 Checking logs in the Event Viewer

Similar to other applications running on Windows server, the Relativity service bus logs all errors, warnings, and information events to the event viewer. You can view these events in the application logs of the Windows Event Viewer application. By default, the verbose events aren't logged, but you can enable them.

Use the following steps to enable verbose logs:

1. Open the Event Viewer on your server.
2. On the View menu, click Show Analytic and Debug Logs.
3. Under Applications and Service Logs, open the Microsoft-ServiceBus > Analytic Trace.
4. Right-click on the Analytic Trace and click Enable Log.
5. Repeat these steps to enable the **Debug Trace**.

### 4.5 Using Service Bus Explorer

You can use Service Bus Explorer to troubleshoot the Relativity service bus. With this Microsoft tool, you can also view entities contained in the Relativity service bus, and their current statuses. The following information includes guidelines for using the Service Bus Explorer in a Relativity environment. For additional information, see Service Bus Explorer ([https://code.msdn.microsoft.com/Service-Bus-Explorer-f2abca5a](https://code.msdn.microsoft.com/Service-Bus-Explorer-f2abca5a)) on Microsoft Developer Network site.

**Note:** We recommend using the Service Bus Explorer only when other troubleshooting methods have failed. Use the Service Bus Explorer to view the status of Relativity service bus entities. Don’t modify any settings through this tool, because such actions may negatively affect the functionality and performance of your Relativity instance. In addition, don’t delete messages unless we explicitly recommend this action to resolve an issue with the service bus. Manipulating data in the service bus can adversely affect Relativity functionality.

#### 4.5.1 Prerequisites for Service Bus Explorer

You must complete the following prerequisites before using Service Bus Explorer:

- Obtain the executable for the Service Bus Explorer:
Extract the contents of the Service Bus Explorer.zip file.

In the C# folder, locate the 2.1.zip file. Create a new folder and copy this file to it.

**Note:** Relativity service bus requires Service Bus Explorer 2.1.3.0.

Extract the contents of the 2.1.zip file.

Double-click on the ServiceBusExplorer.exe to launch the application.

- Run Service Bus Explorer on any machine that can connect to the Relativity service bus, and the required certificates.
- Run Service Bus Explorer as a user with administrative and manage user privileges on the Service Bus for Windows Server.

### 4.5.2 Connecting to a Relativity namespace

After installing the Service Bus Explorer, you must configure a connection string for your Relativity service bus namespace.

Use the following steps to connect to a namespace:

1. Open Service Bus Explorer.
2. Click **File > Connect**.
3. In the Service Bus Namespaces drop-down, select **Enter connection string**.
4. Enter the following string in the **Connection String** textbox. Replace `<FarmDns>`, `<HttpsPort>`, and `<TcpPort>` with these values from your service bus farm configuration.

   ```
   End-point=
   =sb://<FarmDns>/Relativity;St-
   sEndpoint=https://<FarmDns>:<HttpsPort>/Relativity;RuntimePort=<TcpPort>;ManagementPort=<HttpsPort>
   ```

   **Note:** You can also use the Get-SbClientConfiguration command instead of manually entering the connection string. This command outputs two connection strings. The second connection string is used for Relativity. Copy this connection string in to the textbox.

5. Click **OK**. You now view topics and subscriptions in the Service Bus Namespace browser.

### 4.5.3 Viewing entities in a namespace

The Service Bus Namespace browser displays a list of topics and subscriptions that your namespace contains. You can view additional information and settings for an entity by selecting it in the browser. You need to manually refresh the data when you select an entity. Select an entity and press **F5**, or right-click on an entity in the browser and click **Refresh**.

The following screen shot illustrates how the Service Bus Explorer displays conversion topics and subscriptions on the Relativity service bus. The integers that appear after a subscription indicate the number of messages and dead letters respectively.
For additional information about using this tool, see Service Bus Explorer (https://code.msdn.microsoft.com/Service-Bus-Explorer-f2abca5a) on Microsoft Developer Network site.

### 4.5.4 Troubleshooting with Service Bus Explorer

To troubleshoot with the Service Bus Explorer, you can use the Peek Messages operation to view messages in a subscription or queue without modifying them.

**Note:** Don't use the Receive operation because it removes messages from the entity, and may cause application failures.

Use the Service Bus Explorer to perform the following troubleshooting tasks. You need to refresh the data when selecting a different entity. See [Viewing entities in a namespace on the previous page](#).

- Verify that your Relativity namespace exists.
- Confirm that the expected topics and subscriptions exist in your Relativity namespace.
- Determine whether topics are receiving messages and whether messages are backed up in topics. You can also use it to determine whether the dead-letter queues are holding undelivered or unprocessed messages.
4.6 Uninstalling the service bus farm

To remove the Service Bus for Windows Server from a farm node or from the farm entirely, see Uninstalling [https://msdn.microsoft.com/en-us/library/dn441431.aspx](https://msdn.microsoft.com/en-us/library/dn441431.aspx) on the Microsoft website.

4.7 Creating a namespace for the service bus

You can use these steps to create a namespace for the service bus without installing or upgrading Relativity. In most cases, you would use these steps when you are troubleshooting your service bus as described in Service bus returns a 401 error on page 26.

Use the following steps to create a namespace:

1. Open the Service Bus PowerShell tool.
2. Execute the following cmdlet to create the Relativity namespace, if it doesn't exist:
   ```powershell
   New-SBNamespace -Name Relativity -ManageUsers yourDomain\yourUserName
   ```
3. Execute the following cmdlet to create the Relativity authorization rule:
   ```powershell
   New-SBAuthorizationRule -NamespaceName Relativity -Name Relativity -Rights Manage,Send,Listen
   ```
4. Complete the following steps to update the instance settings for the Relativity service bus:
   a. Execute the `Get-sbfarm` cmdlet. In the list of hosts, note the values for the `HttpsPort`, `TcpPort`, and `Name`.
   b. Execute the following cmdlet: `Get-SBAuthorizationRule -NamespaceName Relativity -Name Relativity -Rights Manage,Send,Listen`. Note the value of the `PrimaryKey`.
   c. Update the following instance settings with the values obtained from the previous steps. All of the settings are located in the Relativity.ServiceBus section of the instance setting table.

<table>
<thead>
<tr>
<th>Name</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>ServiceBusHttpPort</td>
<td><code>HttpsPort</code> value from step 4a</td>
</tr>
<tr>
<td>ServiceBusTcpPort</td>
<td><code>TcpPort</code> value from step 4a</td>
</tr>
<tr>
<td>SharedAccessKey</td>
<td><code>PrimaryKey</code> value from step 4b</td>
</tr>
<tr>
<td>SharedAccessKeyName</td>
<td><code>Relativity</code></td>
</tr>
</tbody>
</table>

5. If you have a multi-node service bus setup, update the ServiceBus hosts table.
6. Verify that the EDDS.eddsdbo.ServiceBusHosts table has an entry for each host in your farm. In each row, the value of the hostname should match the output from the `get-sbfarm` cmdlet.
7. Verify that table doesn't contain any extraneous entries.
Proprietary Rights

This documentation ("Documentation") and the software to which it relates ("Software") belongs to Relativity ODA LLC and/or Relativity’s third party software vendors. Relativity grants written license agreements which contain restrictions. All parties accessing the Documentation or Software must: respect proprietary rights of Relativity and third parties; comply with your organization’s license agreement, including but not limited to license restrictions on use, copying, modifications, reverse engineering, and derivative products; and refrain from any misuse or misappropriation of this Documentation or Software in whole or in part. The Software and Documentation is protected by the Copyright Act of 1976, as amended, and the Software code is protected by the Illinois Trade Secrets Act. Violations can involve substantial civil liabilities, exemplary damages, and criminal penalties, including fines and possible imprisonment.

©2019. Relativity ODA LLC. All rights reserved. Relativity® is a registered trademark of Relativity ODA LLC.