



Adroit
Technologies



Adroit Edge Gateway

Microsoft Azure Microservice



Adroit
Edge Gateway

COPYRIGHT

Copyright © 2025 Advanced Worx 112 (Pty) Ltd. All rights reserved.

Information in this document is subject to change without notice. The software described in this document is furnished under a license agreement or nondisclosure agreement. The software may be used or copied only in accordance with the terms of those agreements. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or any means electronic or mechanical, including photocopying and recording for any purpose other than the purchaser's personal use without the written permission of Advanced Worx 112 (Pty) Ltd.

Advanced Worx 112 (Pty) Ltd t/a Adroit Technologies
20 Waterford Office Park
189 Witkoppen Road (c/o Waterford Drive)
Fourways
South Africa

www.adroitscada.com

Version	Date	Author	Comment	Applicable Edge Version	Approval
1.0	25.02.2025	D. Acuna	Initial Draft	10.0.5.5	
2.0	18.05.2025	D. Acuna	Final Draft	10.0.5.5	
3.0	19.08.2025	D. Wibberley	Final	10.0.5.5	

Contents

1. Introduction.....	3
2. Configuration and Usage	3
2.1. Azure Storage Connections	3
2.1.1. Adding a New Connection	3
2.2. Azure Storage Publishers	6
2.2.1. Adding Publishers.....	6
2.2.2. Publisher Settings	6
2.2.3. Substitutions and Payload	8
2.2.3.1. Substitutions	8
2.2.3.2. Payload.....	8
2.2.3.3. Preview and Test	8
2.3. Work Folder Location	10
3. Troubleshooting.....	11

1. Introduction

The Azure Storage Publisher is a Microservice that enables the Adroit Edge Gateway to publish operational data to the Azure Cloud. This data can then be used in Dashboards, Reports and Analytics using the AI capabilities of the Azure platform.

2. Configuration and Usage

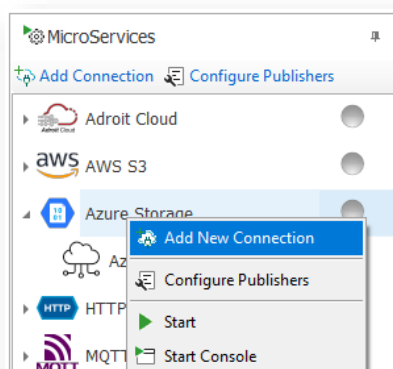
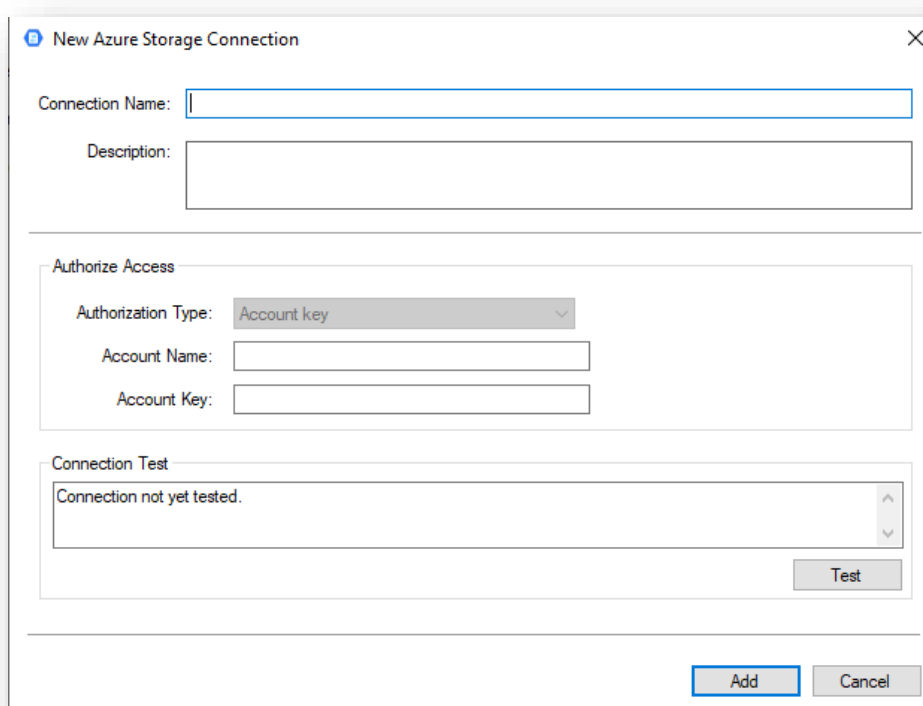
Configuration of all Adroit Edge Publishers is done within the **Config Editor**, under: **Configuration** → **Welcome** → **Edge Gateway Configuration**.

2.1. Azure Storage Connections

2.1.1. Adding a New Connection

To create a new Azure Storage connection:

1. Right-click the **Azure Storage** Microservice in the treeview.
2. Select the option to create a new connection.

The 'New Azure Storage Connection' dialog box contains the following fields and sections:

- Connection Name:** A text input field.
- Description:** A text input field.
- Authorize Access:**
 - Authorization Type:** A dropdown menu with 'Account key' selected.
 - Account Name:** A text input field.
 - Account Key:** A text input field.
- Connection Test:**
 - A text area showing 'Connection not yet tested.'
 - A 'Test' button.
- Buttons:** 'Add' and 'Cancel' buttons at the bottom right.

In the **Add Azure Storage Connection** dialog, configure the following settings:

- **Connection Name:** The display name of the connection under the Azure Storage Microservice.
- **Description:** A description of the connection. Used as a brief tooltip description for the connection.
- **Authorize Access:**
 - **Authorization Type:** This specifies the method used to authorize access to the Azure storage containers. There are different types of authorization methods, such as Shared Key, Shared Access Signature (SAS), and Azure Active Directory (AAD) integration. Each method has its unique characteristics and security features.
 - **Account Name:** This specifies the name of the Azure storage account that the storage container belongs to. The account name is unique within Azure and is used to identify the storage account. It is essential for managing and accessing the storage resources.
 - **Account Key:** This specifies the primary or secondary access key associated with the Azure storage account. The account key is used to authenticate and authorize access to the storage account. It provides full control over all the resources in the storage account and should be kept secure. It is essential to rotate the keys periodically to maintain security.

Connection Testing

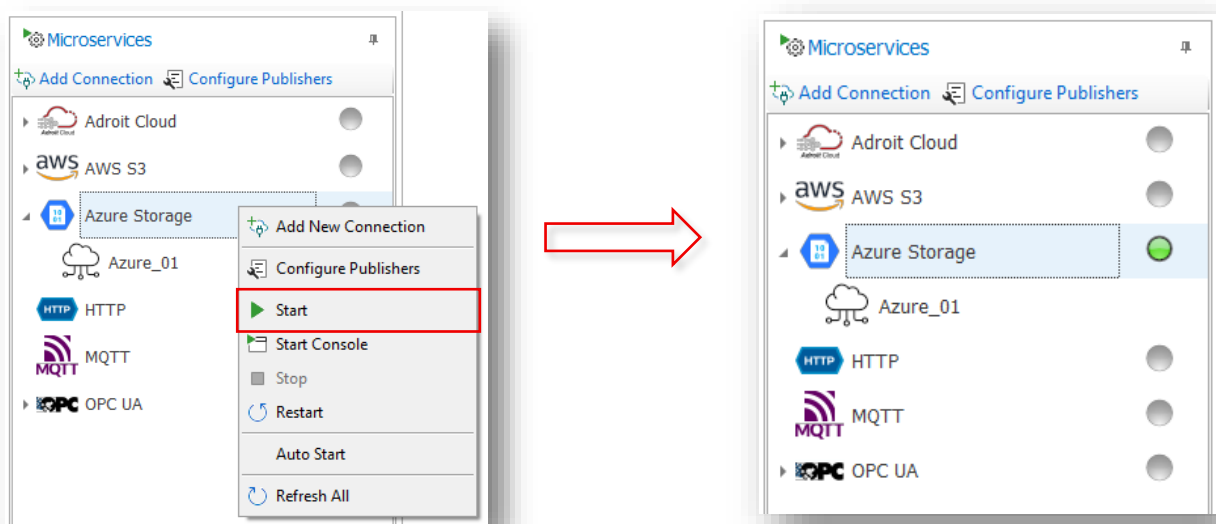
A connection test feature is available to verify the container credentials are correct.

Edit or Delete a Connection

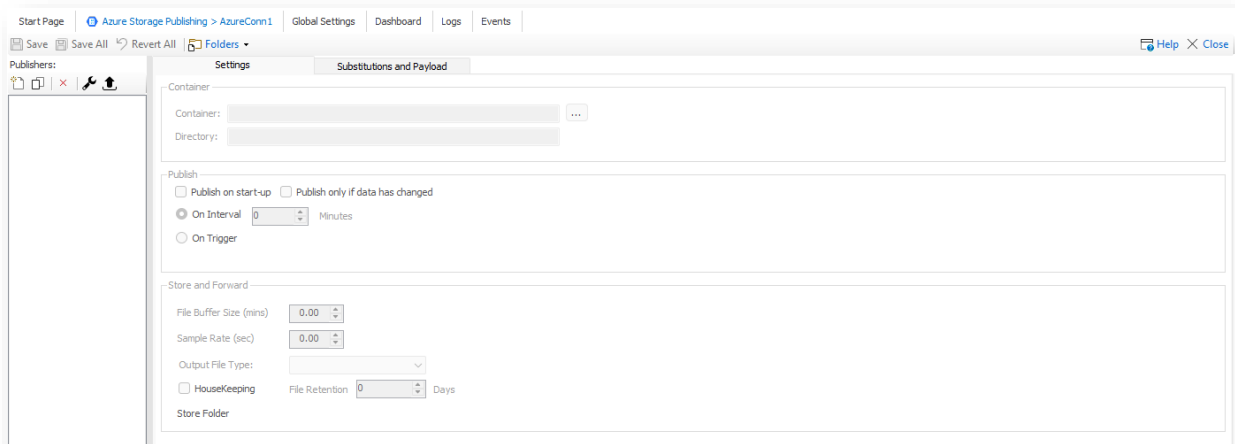
To edit or delete a connection, simply right-click on the connection and select the required option. NB: Deleting a connection will delete any associated Publishers configured for the connection.

Start the Connection

To start the Microservice, right-click on the required Microservice and select Start.



After creating a connection, double-clicking it will present the configuration options for **Azure Storage Publishers**. You can add multiple Publishers to a connection and configure each one with these two tabs:



These tabs are:

Tab	Description
Settings	General settings, including container configuration and publishing modes (e.g., interval or trigger-based).
Substitutions and Payload	Configure Dataelement substitutions and the payload message for the Azure Storage container. Validate and preview the final payload message with live data and test uploading the payload message to the Azure Storage container.

2.2. Azure Storage Publishers

After creating a connection, multiple Publishers can be added by selecting the connection in the treeview and clicking **Configure Publishers** from the toolbar or by double-clicking it.

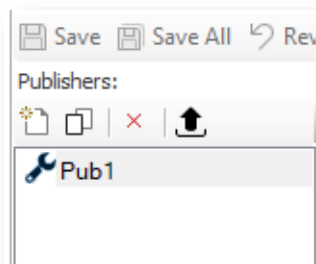
2.2.1. Adding Publishers

To add a Publisher:

1. Click the **Add New Publisher** from the toolbar.
2. Provide a name and description for the Publisher.

Each Publisher starts in **Maintenance Mode** (not actively publishing) by default. To switch to **Publish Mode**, use the context menu or toolbar buttons options. Additional options allow you to:

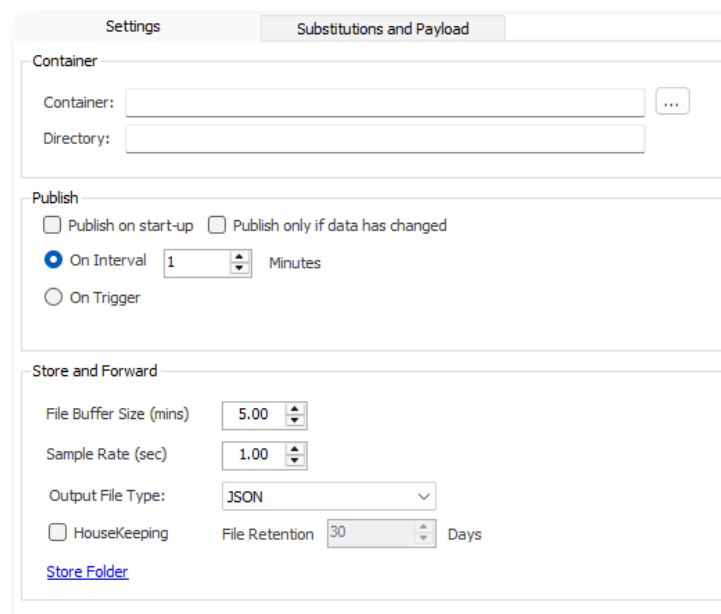
- Edit the Publisher's description.
- Delete the Publisher.
- Copy the Publisher, including all settings, substitutions, and payload configurations. The copy function also supports Find and Replace for substitution Dataelement names.



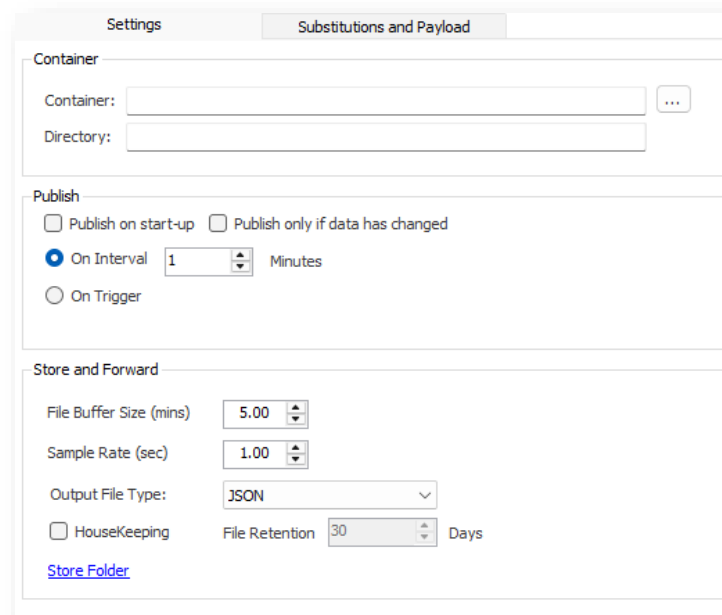
2.2.2. Publisher Settings

The **Settings** tab includes:

- **Container:** This provides the name of the container we are uploading too. A Container browser can be launched to browse in any of the available containers.
- **Directory:** This provides the name of the directory of the container we are uploading to.
- **Publish Mode Settings:**
 - **On Interval:** Publishes messages at a regular interval (defined in minutes).
 - **Interval** (in minutes): Specifies the publishing interval.



- **On Trigger:** Publishes messages triggered by a Dataelement value change, provided that the trigger Dataelement has changed and holds a non-zero value.
 - **Deadband:** Defines the waiting period (in minutes) before accepting the next trigger after the last published time.

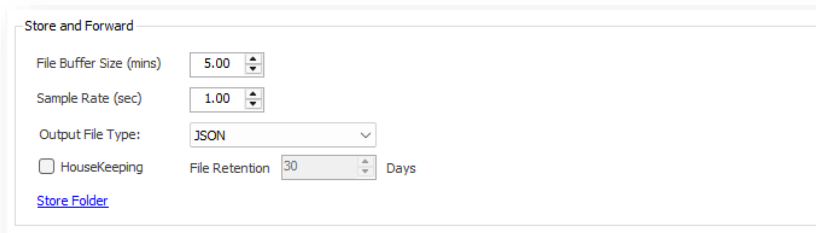


For both publishing modes above you can also:

- **Publish on start-up:** This publishes on startup of the Microservice and when the Publisher is set to publish mode.
- **Publish only if data has changed:** Publishes only if any substitution Dataelement values have changed.
- **Store and Forward Settings:**

Payload messages will be appended to a file based on the configured Sample Rate. The file will be uploaded when the configured File Buffer Size is reached.

 - **File Buffer Size (mins):** Specify the file buffer size in minutes. This determines how long the buffer will accumulate messages before being uploaded.
 - **Sample Rate (sec):** Specify the sample rate in seconds. This determines how frequently the payload messages are sampled and appended to the file.
 - **Output File Type:** Indicate the output file type(s) (e.g., JSON, CSV, Parquet).
 - **Housekeeping:** After uploading the file, the file is moved to a backup folder. The Housekeeping deletes any files older than the File Retention days.
 - **Store Folder:** This opens a Windows Explorer to the Publishers Store and Forward folder where the processing of the upload folder has been done.



2.2.3. Substitutions and Payload

In the Substitutions and Payload tab we can configure the substitution Dataelements for the Azure Storage Publisher as well as the payload messages which will be uploaded.

We are also provided with an option to validate, preview and test the substituted payload message before it's been uploaded.

2.2.3.1. Substitutions

The Substitutions grid allows users to define multiple substitutions with corresponding Dataelements and formatting.

To add Dataelements to the Substitutions grid:

1. Navigate to the Dataelements tree view under the relevant Adroit Datasource.
2. Select the desired Dataelements (you can multiselect) and drag them into the Substitutions grid.
3. Alternatively, you can directly input the full names of the Dataelements into the Substitutions grid.

Additional Substitutions functionalities include:

- **CSV Import/Export:** Handle substitution lists and their configurations using CSV files. Import or export them seamlessly within the Substitution Grid.
- **Find and Replace:** Locate and modify text within substitution Dataelement names directly in the Substitution Grid by simply typing the name into the grid directly.

2.2.3.2. Payload

The Payload configuration provides a markup editor for creating the JSON payload. Substitutions from the grid can be inserted into the JSON by selecting a substitution and using the Payload toolbar buttons to insert a substitution of specific Dataelement's data i.e. its value, quality and timestamp:

- **Value:** Inserts a `${substitution name}`, which will substitute in the Dataelements value.
- **Quality:** Inserts a `${substitution name}ST`, which will substitute in the Dataelements quality i.e. bad, good etc.
- **Timestamp:** Inserts a `${substitution name}TS`, which will substitute in the Dataelements timestamp

These Substitutions can be also added manually into the JSON in the markup editor.

Additional Payload features include:

- **Templates:** Create JSON payloads using predefined imported templates. A default template is available to generate a JSON structure containing key-value pairs for each substitution, such as *substitution name: \${substitution name}*
- **Validation:** Verify the JSON payload to ensure accuracy and receive feedback on any errors.

2.2.3.3. Preview and Test

The **Preview** tab allows you to:

- **View** – View the final JSON payload with live substituted values.
- **Test** Will execute a publish event of the payload message to ensure functionality. If the test fails, an error response will be displayed.

Settings

Substitutions and Payload

Substitutions

Provide Substitutions that will be replaced with its corresponding Data Elements values in the payload

Import CSV

Export CSV

Find & Replace

Refresh/Validate

Substitution	Data Element	Format
1	Adroit.A1.value	...
2	Adroit.A2.value	...
3	Adroit.A3.value	...
4	Adroit.A4.value	...

Payload

Create the structure of the payload content to be sent with Data Elements substitutions

Language: JSON

Template: Default

Generate

Check

Insert Substitution: ☐ Value ☐ Timestamp ☒ Quality

```

1 {
2   "d": [
3     {
4       "tag": "1",
5       "value": "${1}"
6     },
7     {
8       "tag": "2",
9       "value": "${2}"
10    },
11    {
12      "tag": "3",
13      "value": "${3}"
14    },
15    {
16      "tag": "4",
17      "value": "${4}"
18    }
19  ],
20  "ts": "${1}TS",
21  "container": ""
22 }

```

Chars 277 (0.27 KB)

Ln 1

Col 1

Pos 0

JSON

Payload

Preview

Settings

Substitutions and Payload

Substitutions

Provide Substitutions that will be replaced with its corresponding Data Elements values in the payload

Import CSV

Export CSV

Find & Replace

Refresh/Validate

Substitution	Data Element	Format
1	Adroit.A1.value	...
2	Adroit.A2.value	...
3	Adroit.A3.value	...
4	Adroit.A4.value	...

Payload

Shows a preview of the structure of the payload content to be sent with live substituted data

Last: 2025/05/14 14:18:23

Refresh

```

1 {
2   "d": [
3     {
4       "tag": "1",
5       "value": 55.00
6     },
7     {
8       "tag": "2",
9       "value": 676.00
10    },
11    {
12      "tag": "3",
13      "value": 600.00
14    },
15    {
16      "tag": "4",
17      "value": 700.00
18    }
19  ],
20  "ts": "2025/05/14 14:18:23",
21  "container": ""
22 }

```

Payload

Preview

2.3. Work Folder Location

All configuration files related to the Azure Storage are stored in:

C:\ProgramData\Adroit Technologies\SmartDataServices\AzureStorage

Users typically do not need direct access to these folders, as configuration can be managed via the Config Editor. However, key files include:

- *config.xml*: Stores all settings, Dataelement substitutions, and payload configurations.
- *AzureStorage.log*: Available only if "Log to file" is enabled. This log file must be opened manually.

Subfolders include:

- CSV: Contains CSV files for importing and exporting Dataelement substitutions.
- Templates: Contains predefined JSON templates for generating specific payload messages.

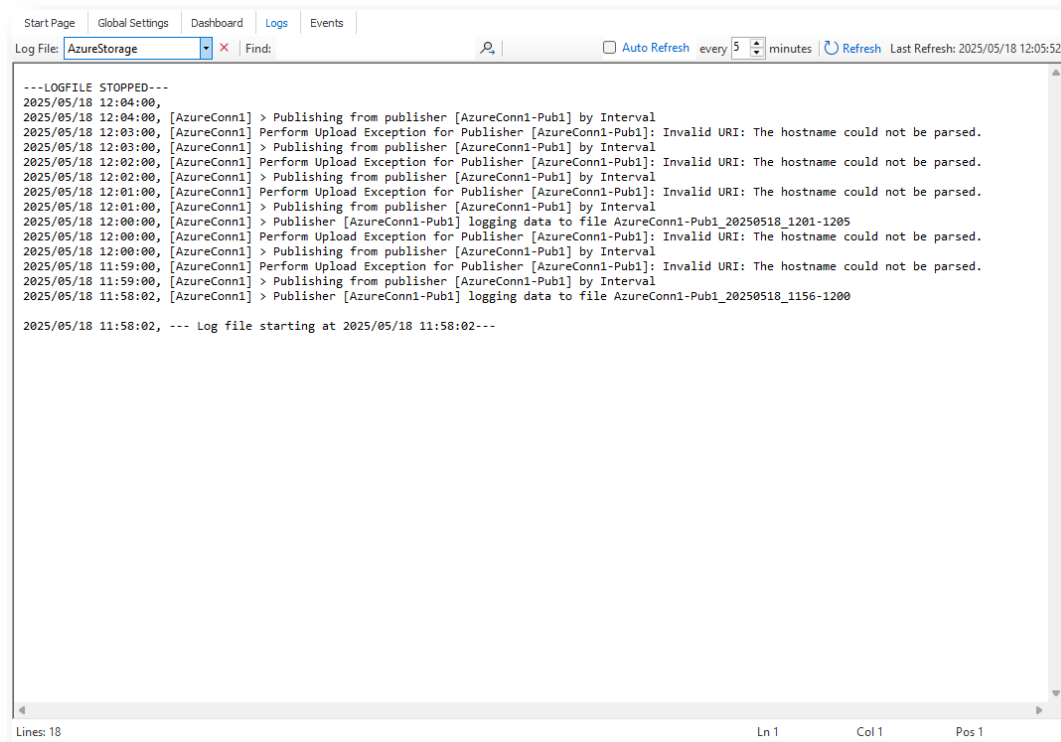
3. Troubleshooting

The Azure Storage Microservice provides various tools to help resolve issues:

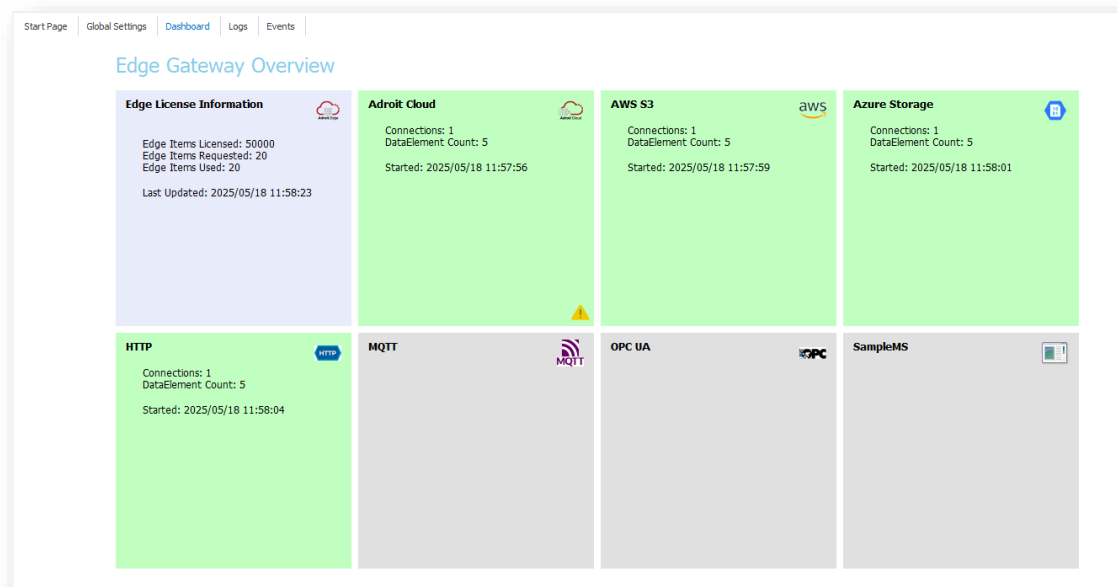
- **Connection Testing:** The Azure Storage connection settings include a test functionality to verify the connection to the Azure Storage container.
- **Substitution Validation:** In the Azure Storage Publishers configuration, the substitution grid highlights invalid Dataelements of a substitution in red for easy identification.
- **Payload Validation:** The payload configuration features a validation tool to ensure the JSON payload is properly formatted, providing detailed feedback on any errors.
- **Preview:** The Preview tab allows users to preview the final JSON payload with live substituted values.
- **Diagnostics:** Access additional tools on the main Adroit Edge Gateway tabs for advanced troubleshooting:
 - **Event Log Viewer:** View all Microservice event log messages sourced from the SmartDataServices event log.

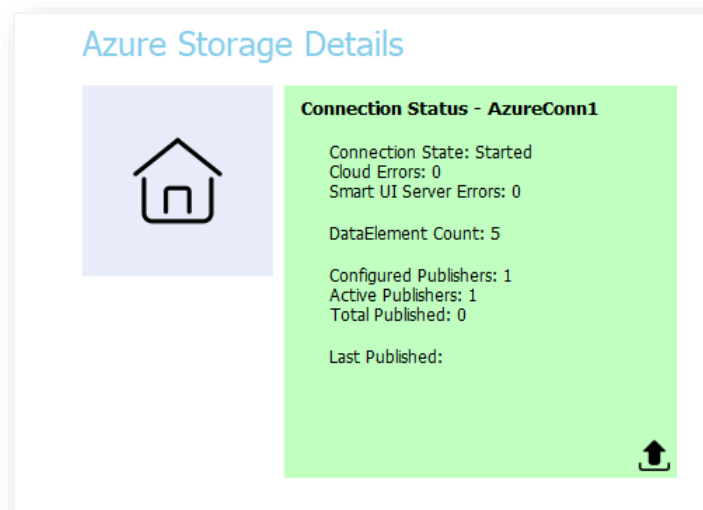
Start Page	Global Settings	Dashboard	Logs	Events
Last Updated: 2025/05/18 12:07:16 Event Viewer ✕ Clear ↶ Export ↷ Refresh				
Enter text to search... Find Clear				
Drag a column header here to group by that column				
Time Generated	Entry Type	Source	Message	Data
2025/05/18 12:07:04	Error	HTTPPost	[HttpConn1] < Publisher [HttpConn1-...	{
2025/05/18 12:07:00	Information	AdroitCloud	[CloudConn1] > Publishing from publis...	
2025/05/18 12:07:00	Information	AzureStorage	[AzureConn1] > Publishing from publis...	
2025/05/18 12:07:00	Information	HTTPPost	[HttpConn1] > Publishing from publish...	
2025/05/18 12:07:00	Information	AWSS3	[AWSConn1] > Publishing from publish...	
2025/05/18 12:07:00	Error	AzureStorage	[AzureConn1] Perform Upload Excepti...	
2025/05/18 12:07:00	Information	AWSS3	[AWSConn1] > Publisher AWSConn1-...	
2025/05/18 12:07:00	Error	AWSS3	[AWSConn1] < Publisher AWSConn1-...	
2025/05/18 12:07:00	Information	AWSS3	[AWSConn1] > Publisher AWSConn1-...	
2025/05/18 12:07:00	Error	AWSS3	[AWSConn1] < Publisher AWSConn1-...	
2025/05/18 12:07:00	Information	AWSS3	[AWSConn1] > Publisher AWSConn1-...	
2025/05/18 12:07:00	Error	AWSS3	[AWSConn1] < Publisher AWSConn1-...	
2025/05/18 12:07:00	Information	AWSS3	[AWSConn1] > Publisher AWSConn1-...	
2025/05/18 12:07:00	Error	AdroitCloud	[CloudConn1] < Publisher [CloudConn...	{
2025/05/18 12:06:04	Error	HTTPPost	[HttpConn1] < Publisher [HttpConn1-...	{
Total: 1246 records				
Time Generated: 2025/05/18 12:07:04		Data:		
Entry Type: Error		<pre>{ "d": [{ "tag": "1", "value": 0.0 }, { "tag": "2", "value": 0.0 }, { "tag": "3", "value": 0.0 }] }</pre>		
Source: HTTPPost				
Message:				
[HttpConn1] < Publisher [HttpConn1-Pub1] post Failed to send! Something went wrong with post				

- **Log File Viewer:** Examine the complete contents of a selected Microservice log file.



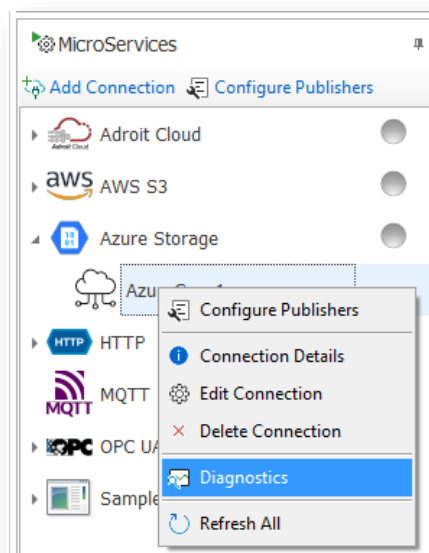
- **Dashboard:** Gain real-time dashboard-like feedback from Microservices, featuring specific Microservices statuses, counters, and indicators.

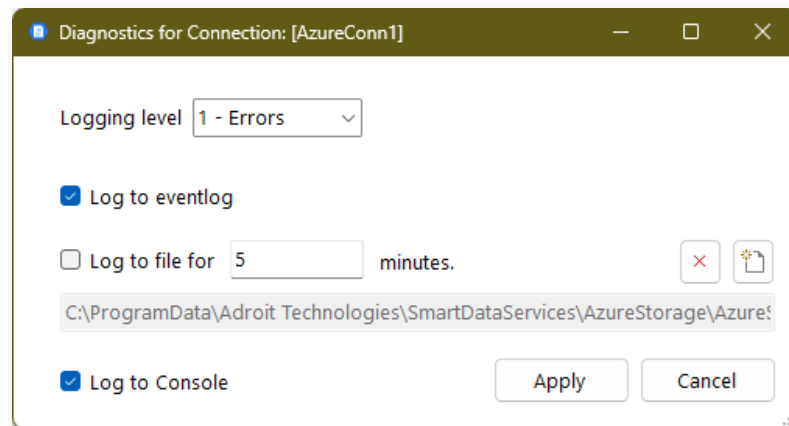




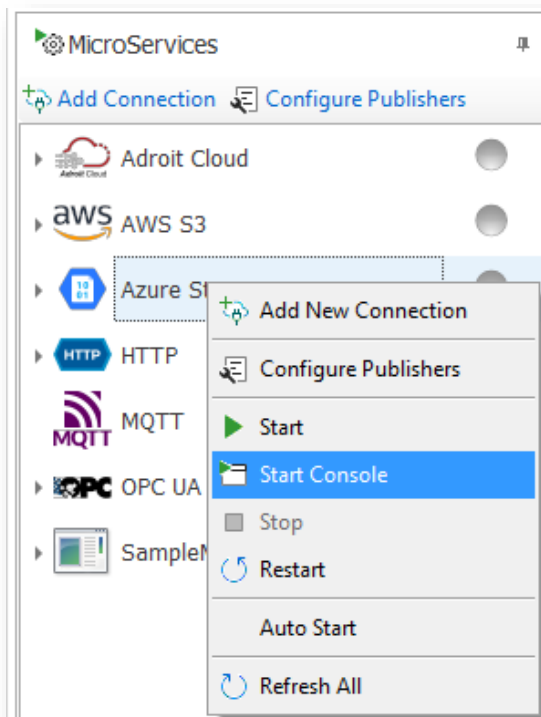
Diagnostics information logging, including its log level (i.e. Errors, Warnings, Information and Debug levels etc.) and output destination (i.e. Event log, log file, console etc.) is configured through the Diagnostics dialog for each Azure Storage connection.

To configure the Diagnostics for an Azure Storage connection, right-click the Azure Storage connection and click Diagnostics.





- **Microservice Console View:** Operate each Microservice as a console application to observe its real-time console messages. To run a Microservice as a console right- click the Microservice and click Start Console.



```
Y:\debug\SDS\SDSAzureStorageExec.exe
11:39:27: [AzureStorage] >>>>>>>> Loading MicroService <<<<<<<<<
11:39:27: [AzureStorage]
11:39:27: [AzureStorage] Connecting to Smart UI Server...
11:39:28: [AzureStorage] Smart UI Server Connection successful :)
11:39:28: [AzureStorage] Reading MicroService configuration ...
11:39:28: [AzureStorage] Reading Connection [AzureConn1] Publishers ...
11:39:28: [AzureStorage] + Added Publisher: AzureConn1-Pub1
11:39:28: [AzureStorage] Subscribing to Publisher [AzureConn1-Pub1] data elements
11:39:28: [AzureStorage] Reading Publisher [AzureConn1-Pub1] data elements initial values
11:39:28: [AzureStorage] Starting Publisher [AzureConn1-Pub1]: PollInterval Mode
11:39:28: [AzureStorage]
11:39:28: [AzureStorage] >>>>>>>> MicroService Loading Complete <<<<<<<<<
11:39:28: [AzureStorage]
```