

Rocket MultiValue Integration Server v1.3.2 Release Notes

Notices

Copyright

© Rocket Software, Inc. or its affiliates 1996-2023. All Rights Reserved.

© 1996-2023 Rocket Software, Inc. or its affiliates. All Rights Reserved.

Trademarks

Rocket is a registered trademark of Rocket Software, Inc. For a list of Rocket registered trademarks go to: www.rocketsoftware.com/about/legal. All other products or services mentioned in this document may be covered by the trademarks, service marks, or product names of their respective owners.

Examples

This information might contain examples of data and reports. The examples include the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

License agreement

This software and the associated documentation are proprietary and confidential to Rocket Software, Inc. or its affiliates, are furnished under license, and may be used and copied only in accordance with the terms of such license.

Note: This product may contain encryption technology. Many countries prohibit or restrict the use, import, or export of encryption technologies, and current use, import, and export regulations should be followed when exporting this product.

Corporate information

Rocket Software, Inc. develops enterprise infrastructure products in four key areas: storage, networks, and compliance; database servers and tools; business information and analytics; and application development, integration, and modernization.

Website: www.rocketsoftware.com

Contacting Technical Support

The Rocket Community is the primary method of obtaining support. If you have current

Rocket Global Headquarters

77 4th Avenue, Suite 100

Waltham, MA 02451-1468

USA

To contact Rocket Software by telephone for any reason, including obtaining pre-sales information and technical support, use one of the following telephone numbers.

Country and Toll-free telephone number

- United States: 1-855-577-4323
- Australia: 1-800-823-405
- Belgium: 0800-266-65
- Canada: 1-855-577-4323
- China: 400-120-9242
- France: 08-05-08-05-62
- Germany: 0800-180-0882
- Italy: 800-878-295
- Japan: 0800-170-5464
- Netherlands: 0-800-022-2961
- New Zealand: 0800-003210
- South Africa: 0-800-980-818
- United Kingdom: 0800-520-0439

Contacting Technical Support

The Rocket Community is the primary method of obtaining support. If you have current support and maintenance agreements with Rocket Software, you can access the Rocket Community and report a problem, download an update, or read answers to FAQs. To log in to the Rocket Community or to request a Rocket Community account, go to www.rocketsoftware.com/support. In addition to using the Rocket Community to obtain support, you can use one of the telephone numbers that are listed above or send an email to support@rocketsoftware.com.

Contents

Chapter 1: Release Notes.....	1
What's new.....	1
What's changed.....	1
System requirements.....	3
What's fixed.....	4

What's new

What's new describes new functionality that is included in MVIS v1.3.2.

New features in MVIS v1.3.2

Review a summary of the new features in release v1.3.2.

Admin API

Issue Number: MVIS-4667 | MVIS-4670:

MVIS provides a new Admin Development API that returns a single subroutine in the response, together with all Dynamic Arrays definitions used by that subroutine on the basis of the query string parameter `attachDynamicArrays` for a specific Account.

See [Admin API](#) in the "What's Changed" topic for complete details.

Issue Number: MVIS-4668:

MVIS provides a new Admin Development API that returns a single **Dynamic Array** definition for a specified Account.

See MVIS-4668 in the "What's Changed" topic for complete details.

Logging

Issue Number: MVIS-3348:

MVIS logs now feature a request/transaction/correlation **Trace-Id** when it is specified in the request header.

See [Logging](#) in the "What's Changed" topic for complete details.

REST

Issue Number: MVIS-4704:

You can now select the **Data Type** column value from a **Data Type** drop-down menu, and the **Data Type** value is stored in metadata of MVIS.

See [REST](#) in the "What's Changed" topic for complete details.

What's changed

Review changes to existing functionality from Version 1.3.1 to Version 1.3.2 of MVIS.

MVIS v1.3.2 contains changed functionality that you should keep in mind when you decide whether to upgrade to the new version. Some changes might require that you modify existing applications and influence how you develop new applications.

Changed functionality

Review the details of changed feature functionality in release 1.3.2.

Admin API

Issue Number: MVIS-4667 | MVIS-4670:

MVIS provides a new Admin Development API that returns a single subroutine in the response, together with all Dynamic Arrays definitions used by that subroutine on the basis of the query string parameter `attachDynamicArrays` for a specified Account.

Users should provide a subroutine and an **Account Name**, and the query string parameter `attachDynamicArrays=true`.

The `attachDynamicArrays=true` parameter will return the details of each dynamic array used.

The `attachDynamicArrays=false` parameter will return a list of dynamic array names used.

The API Path contains the **Account Name** and **Subroutine Name**. The response contains:

- The subroutine definition.
- Each **Dynamic Array** definition that is associated with a subroutine parameter.

Previously, an existing API was able to get all the subroutines for an account, an approach that could involve a high number of subroutines.

API GET Reference: `GET /api/manager/rest/{accountName}/subroutine/{subroutineName}`

Issue Number: MVIS-4668:

MVIS provides a new Admin Development API that returns a single **Dynamic Array** definition for a specified Account.

The API Path contains the **Account Name** and **Dynamic Array Name**. The response contains:

- The single **Dynamic Array** definition that is associated with the Account.

Previously, an existing API was able to get only all the Dynamic Arrays for an account.

API GET Reference: `GET /api/manager/rest/{accountName}/dynarray/{dynArrayName}`

Logging

Issue Number: MVIS-3348:

MVIS logs now feature a request/transaction/correlation Trace-Id when it is specified in the request header.

To include the Trace-Id, set a value for the request header key `X-MVIS-Trace-Id`.

The Trace-Id will display as the first element of the log line. It is enclosed in square brackets:

For example:

```
[X-MVIS-Trace-Id:SELF: TraceIdFromRequestHeader]
```

REST

Issue Number: MVIS-4704:

The **Data Type** column value is now populated by default from the SQL data type that is defined in attribute 8 of the UniVerse dictionary. This is now the case if the connection string that is defined for the MVIS account is set to *uvcs*. The **Data Type** column is not displayed for a connection string of any other value.

You can now select the **Data Type** column value from a **Data Type** drop-down menu, and the **Data Type** value is stored in the metadata of MVIS.

Previously, the **Data Type** column that appears for **Data Resources** did not show any values, nor was the data type stored in the metadata of MVIS.

System requirements

Before you install MVIS 1.3.2, review the current platform and hardware requirements for MVIS.

See *System Requirements* in the MVIS Installation and User Guide, in the respective product category on the Rocket Documentation Library at <https://docs.rocketsoftware.com>.

MVIS hardware requirements

The hardware requirements for MVIS 1.3.2 are:

- CPU: 2vCPU
- A 2 GB minimum amount of available hard disk space.
- At least 4 GB of RAM available memory.

Windows Server 2012 requirements

The Windows Server 2012 platform requirements for MVIS are:

- UniVerse 11.3.1 database or higher, or UniData 8.2.1 database or higher running on a local or remote machine that is licensed for connection pooling.
- Java (JDK or JRE) version 1.8, OpenJDK 1.8.
- Python 3.7 with the latest version of pip (Python's package manager). Do not use Python version 3.4 that currently comes with UniVerse and UniData.

Note: Python 3.8 is not supported for Windows platforms. For a workaround to this issue, see "Tech Note 000017690 - MVIS Install Fails When Using Python 3.8 or Later On Windows" available from the [Rocket Community](#).

- You should upgrade to the latest version of pip before beginning the installation.
- If you are installing Python for the first time, select Add Python 3.X to PATH from the Setup window. If you already have Python installed, ensure that the `path-to-Python\Scripts` directory is included in your *PATH* environment variable so the pip executable can be located. To update pip to the latest version, enter the following command:

```
python -m pip install -U pip
```

Linux/AIX: Red Hat Enterprise Linux 7 and AIX 7.2 requirements

The Linux/AIX platform requirements for MVIS are:

- UniVerse 11.3.1 or higher, or UniData 8.2.1 or higher running on a local or remote machine that is licensed for connection pooling.
- Java (JDK or JRE) version 1.8, OpenJDK 1.8.
- Python 3.7 with the latest version of pip (Python's package manager). Do not use Python version 3.4 that currently comes with UniVerse and UniData. Each Linux distribution has its own recommended mechanism to install Python 3.X. Follow the recommendations for your particular Linux distribution.
- You should upgrade to the latest version of pip before beginning the installation. To update pip to the latest version, type the following command:

```
pip install -U pip
```

What's fixed

What's fixed describes resolutions to problems that were found in a previous release.

Admin API

Issue Number: MVIS-4621:

OAuth provides delegated authorization for REST APIs that enables applications to obtain limited access to a user's data without defining new users.

MVIS provides OAuth provider name support in V1.3.2. When you install MVIS V1.3.2 (upgrade from V1.3.1) and restart the server, the `oauthPublicKeyPath` `authPublicKeyPassword` keys are removed from the MVIS `cm.ini` configuration file.

Previously, the `oauthPublicKeyPath` `authPublicKeyPassword` keys should not have been present in the `cm.ini` configuration file in the MVIS release V1.3.1 installation build.

This issue is fixed.

Issue Number: MVIS-4665:

MVIS Admin API documentation now features improved API description messages for one data resource and one dynamic array Admin API.

The POST `/api/manager/rest/{accountName}/dataresource` API now features this API description:
`Creates data resources for the specified account.`

The POST `/api/manager/rest/{accountName}/dynarray` API now features this API description:
`Creates dynamic arrays for the specified account.`

Previously, the descriptions for the above cited Admin APIs began with the inaccurate verb "lists" instead of the required term "creates".

This issue is fixed.

Installation/Configuration

Issue Number: MVIS-2862:

MVIS deploys the U2XfrWrite subroutine utility to the Multivalue Server. This subroutine compiles and catalogs other basic subroutines.

When you compile and catalog a program, MVIS now follows these steps:

1. MVIS checks if U2XfrWrite is already cataloged.
2. If U2XfrWrite is not cataloged, then MVIS deploys U2XfrWrite to the same basic program file as the subroutine you are compiling and cataloging.

Previously, MVIS always deployed U2XfrWrite to the BP file. This resulted in issues for instances in which the BP file was a hashed file rather than a directory.

This issue is fixed.

Integrations

Issue Number: MVIS-5002

When MVIS stops, topic subscriptions are removed, a new autoIdlOnDelete property automatically deletes a subscription if it is not used after 30 days in the event the subscription was not deleted when MVIS stops. Unused subscriptions are thus auto-purged by Microsoft Azure.

MVIS now uses a guaranteed random Universally Unique Identifier (UUID) unique name instead of a Media Access Control (MAC) address as the subscription name to ensure that name conflicts are avoided.

Previously, MVIS was creating subscriptions that were never purged. MVIS subscribed to a topic upon the start of the service, yet when MVIS stopped the topic was not unsubscribed.

This issue is fixed.

Logging

Issue Number: MVIS-2512:

When you enable or restart an account in MVIS, an error occurs if it is not currently licensed for connection pooling.

The error message associated with this scenario is updated so that it now reads with greater clarity.

For UniData, the error message now reads: `The user limit has been reached on the server.`

For UniVerse, the error message now reads: `No licensed connection pools available.`

Previously, if you attempted to connect to a database server that did not have connection pool licenses the following error was logged: `The UniVerse/UniData license has expired.`

This issue is fixed.

Issue Number: MVIS-4756:

In MVIS release v1.3.1 an option was added to include the account name in log entries.

If an endpoint attempted to call a subroutine which was not cataloged, the log line was formed incorrectly. The error message and account names were swapped in the log information, with one log line being broken into multiple lines.

This issue is fixed.

Issue Number: MVIS-4956:

In release MVIS v1.3.1 it is possible to include the MVIS account name as part of MVIS log entries.

In the event that no UniRPC connection was active, the MVIS ERROR level log shows ?COMOBACKUP? instead of the MVIS account name.

Here are examples of MVIS ERROR level log lines at v1.3.1 versus v1.3.2:

MVIS 1.3.1 log:

```
com.rs.u2.u2cm.a - UniPool::adjustMinPoolSizeConnections: Attempting a connection for
account ?COMOBACKUP?
com.rs.u2.u2cm.a - In connection manager log como backup [ROCKET U2][UniObjects for
Java][UniSession Exception][ErrorCode: 81002]No RPC Connection active.
```

MVIS 1.3.2 log:

```
com.rs.u2.u2cm.a - [MVIS_Account_Name] UniPool::adjustMinPoolSizeConnections:
Attempting a connection for account MVIS_Account_Name
com.rs.u2.u2cm.a - [MVIS_Account_Name] In connection manager log como backup [ROCKET
U2][UniObjects for Java][UniSession Exception][ErrorCode: 81002]No RPC Connection
active
```

TRACE level logs will still show the account name ?COMOBACKUP?, since this is an indication that the specific connection is performing the task of backing up U2 server COMO logs to the BackupComoLogs folder on the MVIS server.

This issue is fixed.

Open API (Swagger)

Issue Number: MVIS-4550:

Previously, the Swagger payload for generated Subroutine Vanity URLs was including output parameters in the endpoint request body.

This issue is fixed.

Issue Number: MVIS-4559:

Support for a fields query string parameter is added in MVIS v1.3.2 to ensure compatibility with U2RESTful Web Services support for the fields query string parameter with a value of a comma separated list of fields to be returned.

If this query string parameter was added to a Data Resources endpoint in previous versions of MVIS, an error was returned.

This issue is fixed.

Issue Number: MVIS-4766:

Swagger subroutine models, when expanded for the output parameter, now show correct values for the Dynamic Array and the string parameter output.

Previously, MVIS did not produce or define an output array reference to support the Swagger **Definition** record for **Parameter Type** = output and for **Data Type** = Dynamic Array.

This issue is fixed.

REST

Issue Number: MVIS-3300:

For MVIS connections to UniData accounts on Linux servers, MVIS no longer erroneously reports the following error when compiling and cataloging a subroutine for instances in which the user that is used to connect to the account only has Group privileges to the cataloged subroutine:

```
Error 25001: Failed to compile and catalog subroutine on MV server.  
Compiling and cataloging the subroutine 'NEW@SUBNAME' failed:  
Check the subroutine compiling and cataloging options and source code in REST service  
definition.
```

Previously, MVIS erroneously reported this error.

This issue is fixed.

Issue Number: MVIS-4805:

For instances where a subroutine returns an incorrect JSON Object or Array, both the Subroutine name and the output Parameter name are now included in an optimized informational error message.

Previously, the word "object" was spelled incorrectly in error message. In addition, the error detail itself did not provide the most useful and specific information unless users increased the log level to DEBUG.

This issue is fixed.

Issue Number: MVIS-4820:

From MVIS release v1.3.2 and higher, the following log lines should be recorded in the `mvis.log` file for a vanity URL request:

```
com.rs.u2.u2rest.b.c - Started vanity url execution  
com.rs.u2.u2cm.o - Executing the vanity url endpoints. Name : Vanity/URL/Path  
com.rs.u2.u2cm.o - [AccountName] The tracer data sent to dataserver: 'REST=1'
```

Previously, INFO level logs were not shown for a successful Vanity URL request.

This issue is fixed.