Rocket
LEGACY POWERS LEGENDARY

DATASHEET

Rocket® MultiValue Integration Server

Modernization, improved performance and 24x7x365 uptime for better customer experiences

Organizations use the Rocket MultiValue Application Platform to build reliable, scalable, cost-effective applications—but they have to modernize those applications to stay relevant and competitive. When you can modify and enhance applications quickly and easily, you can address end-user demands, cope with market competition and transformation, and attract new market opportunities.

The Rocket® MultiValue Integration Server (MVIS) helps extend the business reach and technology options for applications developed with the MultiValue Application Platform, making it easier to build APIs that expose MultiValue (MV) data and business logic to partner ecosystems and third-party services. Continuous availability comes from horizontal scaling, failover support, and API and connectivity monitoring. An integration layer provides cloud-ready functionality and flexible deployment options so you can streamline costs and gain operational efficiencies.

> "MVIS provides opportunities for us to be more competitive in the market because our products will have the strength of a MultiValue database and the flexibility to integrate with other current and upcoming solutions."
>
> - Jamie Luna Morales, Managing Director, Alcomtec EIRL

Product benefit list

- Get new applications and features to market faster
- Ensure optimized user experiences and 24x7x365 availability
- Minimize costs with flexible and efficient deployment options

Get new applications and features to market faster

It's easy to develop and update applications using modern frameworks and languages with the Rocket MultiValue Integration Server (MVIS). Simply expose MV logic and data through RESTful services and connect through the language or framework of your choice. The Swagger definitions are automatically created when you develop your APIs, and the Swagger toolset lets your development team discover and interact with them, simplifying API development and consumption. Services can be created through a web interface that fosters collaboration. Continuous Integration/Continuous Development (CI/CD) is supported through an admin API to help you get new features to market faster and with greater reliability.

MVIS provides an easy migration path for customers using U2 RESTful Web Services or Web DE. Customers who have been creating RESTful Web Services can simply export REST server data resources, subroutines, and dynamic arrays using the U2RESTful services tool, then import the REST server using the Integration Server Administration Portal. Migration for Web DE customers is completed using the Converter Tool. RedBack Objects are also fully supported.

MVIS can help transform the way organizations capitalize on existing MultiValue IP. Creating and publishing RESTful endpoints quickly means updates are delivered fast, so you can exceed market expectations and take advantage of revenue opportunities.

Ensure optimized user experiences and 24x7x365 availability

Your customers expect 24x7x365 application availability. MVIS provides connection resiliency to make that possible. It communicates with orchestration technologies, such as Kubernetes, for horizontal scaling, where policies can be established to monitor and manage workload limits. Once a maximum limit is exceeded the orchestration service establishes a new instance of Rocket MVIS and the workload is distributed across multiple instances. Potential bottlenecks or server overload is eliminated, providing end users with consistent and uninterrupted service.

Three features work together to ensure resiliency.

- Continuous signals are sent between MVIS and the application server to prevent idle connections from unexpected termination.
- If a monitored connection fails, the bad connection is terminated and another is established.
- The orchestrator checks the health of MVIS at preconfigured intervals. If it's not available, the orchestrator replaces the instance.

Production downtime due to software updates is avoided with graceful failover. Two identical production instances of MVIS run in parallel (one live and one idle), and manual approval traffic is routed to the secondary instance after all in-flight requests to the first are complete. That means no data is lost and end users experience no downtime.

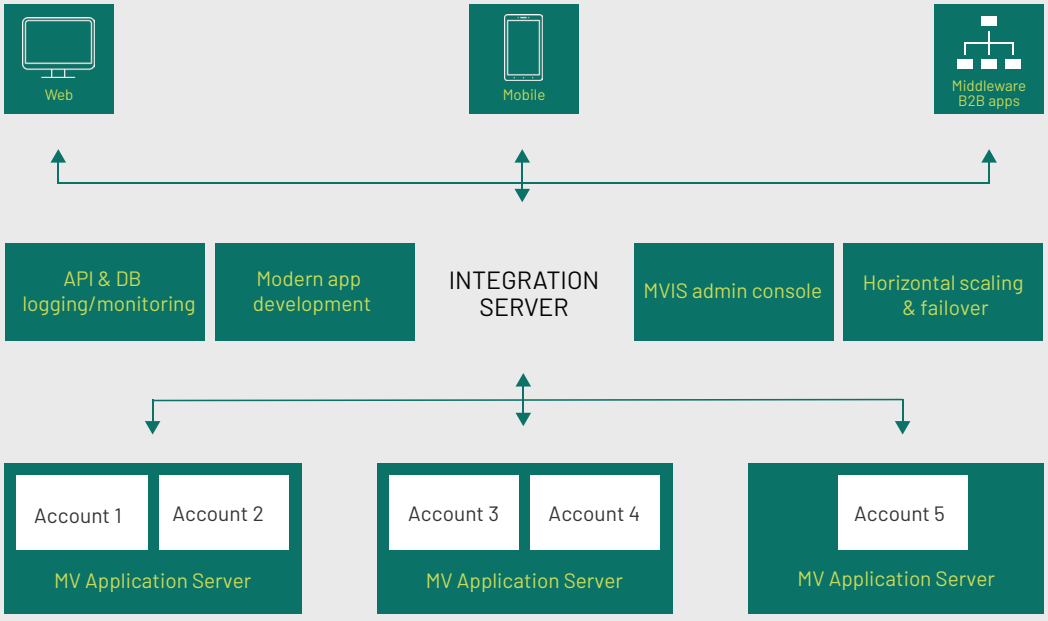
Minimize costs with flexible and efficient deployment options

The MVIS architecture supports flexible deployment options. You can run it natively on the data server, or on a separate server in front of multiple data servers to support partners who have multiple customers with their own data servers, or direct customers that need DevOps functionality (such as graceful failover or connection resiliency) for 24x7x365 uptime. You can also run MVIS as a container for a more lightweight solution—simply point your Orchestrator to the container.

Applications that require high availability and reliability can use a cluster of MVIS(s) to eliminate single points of failure. The shared configuration file (which ensures there are replica images) can be located in an Amazon Web Service (AWS) S3 bucket, in Azure Blob Storage, or in a local Redis storage.

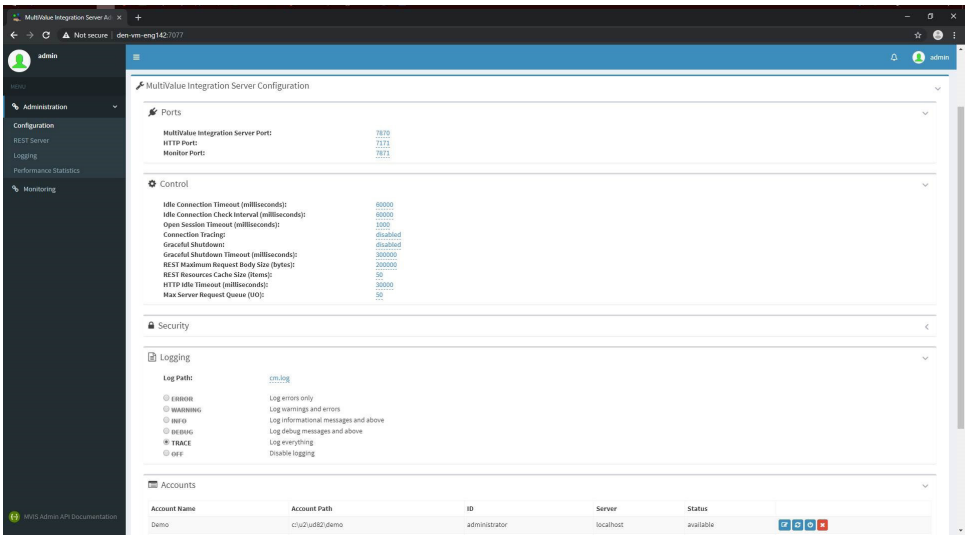
MVIS can also be deployed and managed in the cloud. Increase the value of your application by connecting to cloud provider services such as Azure Application Insights or AWS CloudWatch to access the analytics needed to diagnose issues and understand what users do with your application.

Figure 1: Enhanced pooling for a better user experience



Connection pool licenses can be shared across accounts (with min/max connections set per account) on each server so "available" licenses can be used by applications with heavy demands. Ensuring proper distribution of licenses minimizes end-user wait time, maximizes license usage, and safeguards customer SLAs. Individual pools within a set of pools can be restarted without shutting down the server, eliminating application downtime.

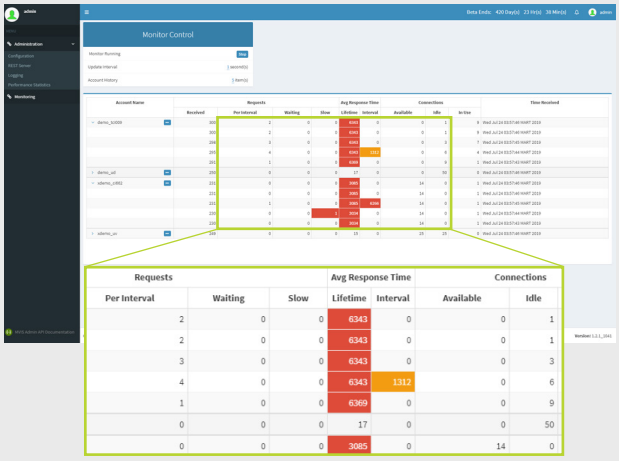
Figure 2: MVIS administration console



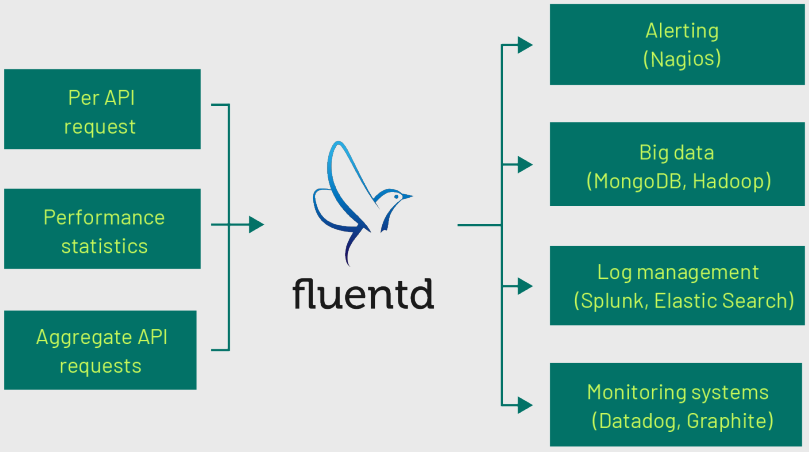
Administration and monitoring functions are available through a web-based user console or a REST API. Easily create REST services, modify connections, enable security, configure cloud-based services and more.

Figure 3: API and connection logging and monitoring

Aggregate real-time analysis using the admin console



Requests			Avg Response Time		Connections	
Per Interval	Waiting	Slow	Lifetime	Interval	Available	Idle
2	0	0	6343	0	0	1
2	0	0	6343	0	0	1
3	0	0	6343	0	0	3
4	0	0	6343	1312	0	6
1	0	0	6369	0	0	9
0	0	0	17	0	0	50
0	0	0	3085	0	14	0



API requests, performance statistics, and license data are logged for review. The administration console provides aggregate analysis. For detailed data discovery MVIS communicates with Fluentd, which formats the logs for consumption by popular log management tools. Investigate issues that impact performance or end-user experience and determine if license redistribution or more licenses are needed.

Technical specifications and system requirements

Platform	Requirements
Windows Sever 2012, 2016	• Rocket® UniVerse 11.3.1, or Rocket UniData® 8.2.1 or later running on a local or remote machine. • Java (JDK or JRE) version 1.8, OpenJDK 1.8. • Python 3.5 or later with the latest version of pip (Python's package manager).
Linux RHEL7, RHEL8	• Rocket UniVerse 11.3.1, or Rocket UniData 8.2.1 or later running on a local or remote machine. • Java (JDK or JRE) version 1.8, OpenJDK 1.8. • Python 3.5 or later with the latest version of pip (Python's package manager).
AIX 7.1, 7.2	• Rocket UniVerse 11.3.1, or Rocket UniData 8.2.1 or later running on a local or remote machine. • Java (JDK or JRE) version 1.8, IBM JDK 1.8

RS MV DS IntegrationServer Feb 2020 v3

Rocket
LEGACY POWERS LEGENDARY