Rocket® MultiValue Integration Server

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

Organizations rely on the Rocket MultiValue (MV) Application Platform to build reliable, scalable, cost-effective applications—but they need to modernize those applications to stay relevant and competitive. When you can quickly and easily modify and enhance applications, you prove MV belongs in the mainstream while addressing end-user demands, coping with market competition and transformation, and attracting new market opportunities.

The Rocket MultiValue Integration Server (MVIS), the upgrade path for Web DE and U2REST users, helps extend the business value and technology options for MV-based applications. It makes building APIs that expose MV data and business logic to partner ecosystems and third-party services faster and easier. Continuous availability comes from connection pooling, failover support, and API and connectivity monitoring. An integration layer provides cloud-ready functionality and flexible deployment options so you can streamline costs, gain operational efficiencies, and lower TCO.

PRODUCT BENEFITS

1 Get new applications and features to market faster

2 Ensure optimized user experiences and 24x7x365 availability

3 Minimize Total Cost of Ownership (TCO) with flexible and efficient deployment options

> " The Rocket suite of tools has answered all my needs. Python is meeting the requirements of a maturing workforce, and we're doing quite a lot of modernization with this and MVIS as a combined technology stack. The MV platform continues to evolve to meet our business challenges and technical needs, making it the logical choice to modernize. "
>
> *Geoff Bishop,*
> *Engagement Manager, Eurotunnel Le Shuttle*

Ensure optimized user experiences and 24x7x365 availability

Your customers expect 24x7x365 application availability. MVIS provides the resilient connections to make that possible. Connection pool licenses can be shared across accounts (with min/max number of 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. MVIS can also be used in combination with orchestration technologies, such as Kubernetes, for deployment automation that enables patterns such as high availability, blue-green deployment, and elastic scalability. An orchestrator using monitoring information from MVIS can automate the spawning of new MVIS server instances to allow your applications to grow elastically as your load increases or recover from failure quickly without the need for human intervention.

Three features work together to ensure resiliency, eliminating a single point of failure.

- 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.

Avoid production downtime due to software updates 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.

Get new applications and features to market faster

MVIS lets you take advantage of new business opportunities—like monetizing data—and trends, like the API economy, faster than ever. MVIS makes it easy to update existing applications and develop new ones using modern frameworks and languages in a way that keeps MV relevant for new developers. For example, you can deploy your updated application with minimal training required on MV. Using MVIS, you'll not only build on your MV investment, but also transform the way you capitalize on your MultiValue IP.

MVIS lets you:

- Develop and use APIs to build integrations to other systems

- Quickly add features to your MV app

- Work with a new partner to take advantage of business opportunities

- Get new features and applications to market faster

- Deploy zero-install web apps

Quickly creating and publishing RESTful endpoints means you can deliver updates fast, exceed market expectations, and take advantage of revenue opportunities.

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 programmatically (via the admin API), or through a collaborative web interface.

MVIS supports Continuous Integration/Continuous Development (CI/CD) through an admin API that helps you get new features to market faster and with greater reliability. Support of oAuth allows your APIs to integrate with third-party authentication providers and authorization servers such as Auth0 and Okta, ensuring a good fit within your enterprise application security configuration.

For our Web DE and U2 REST customers, MVIS provides an easy migration. Customers who have been creating RESTful Web Services can simply export RESTful endpoints for server data resources, subroutines, and dynamic arrays using the U2RESTful services tool, then import endpoint definitions using the MVIS Administration Console. Migration for Web DE customers is completed using the Converter Tool. MVIS offers full support for RedBack Objects.

Minimize TCO, deploy automagically with flexible and efficient deployment options

MVIS gives you fast, efficient development, and set-it-and-forget-it deployment. 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 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, on-premises, or in a hybrid-cloud configuration. Get more value from your applications by connecting to the cloud-provider analytics found in Azure Application Insights or AWS CloudWatch to diagnose issues and understand what users do with your application.
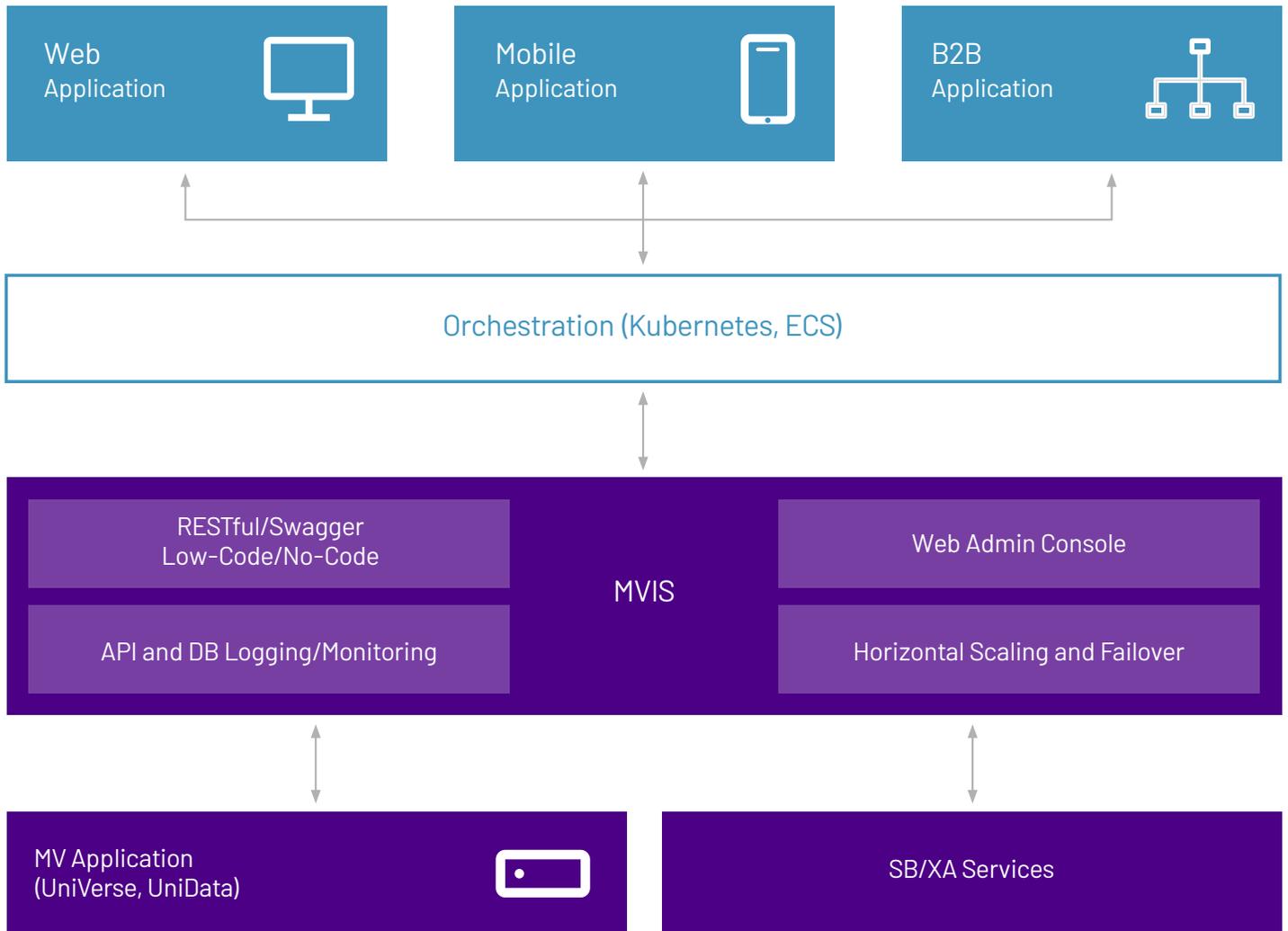
Vanity URLs, defined pre-deployment, give you the flexibility to define and configure endpoint URLs for segmenting and managing access to your application and data to meet your business requirements. For example, you can choose a unique URL for internal and a different unique URL for external APIs for your MV assets.

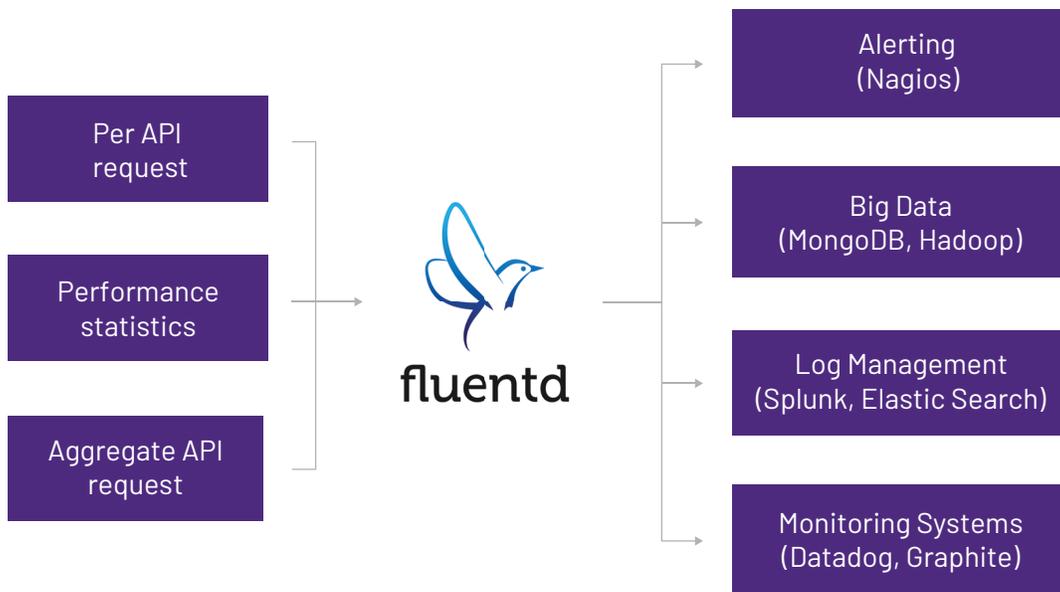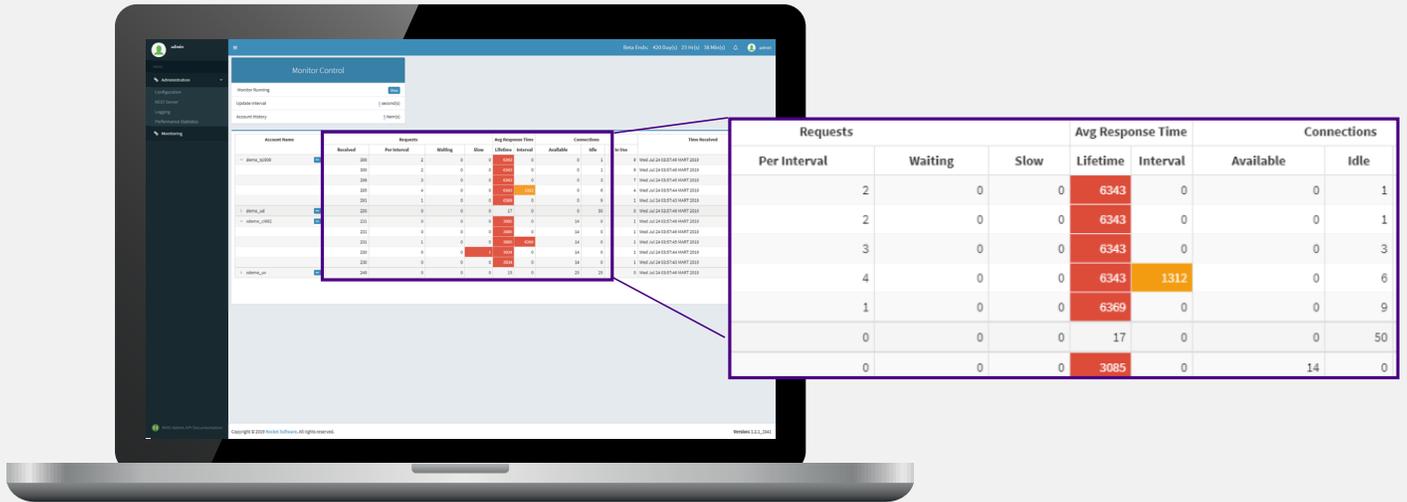**Figure 1:** MVIS supports flexible deployment
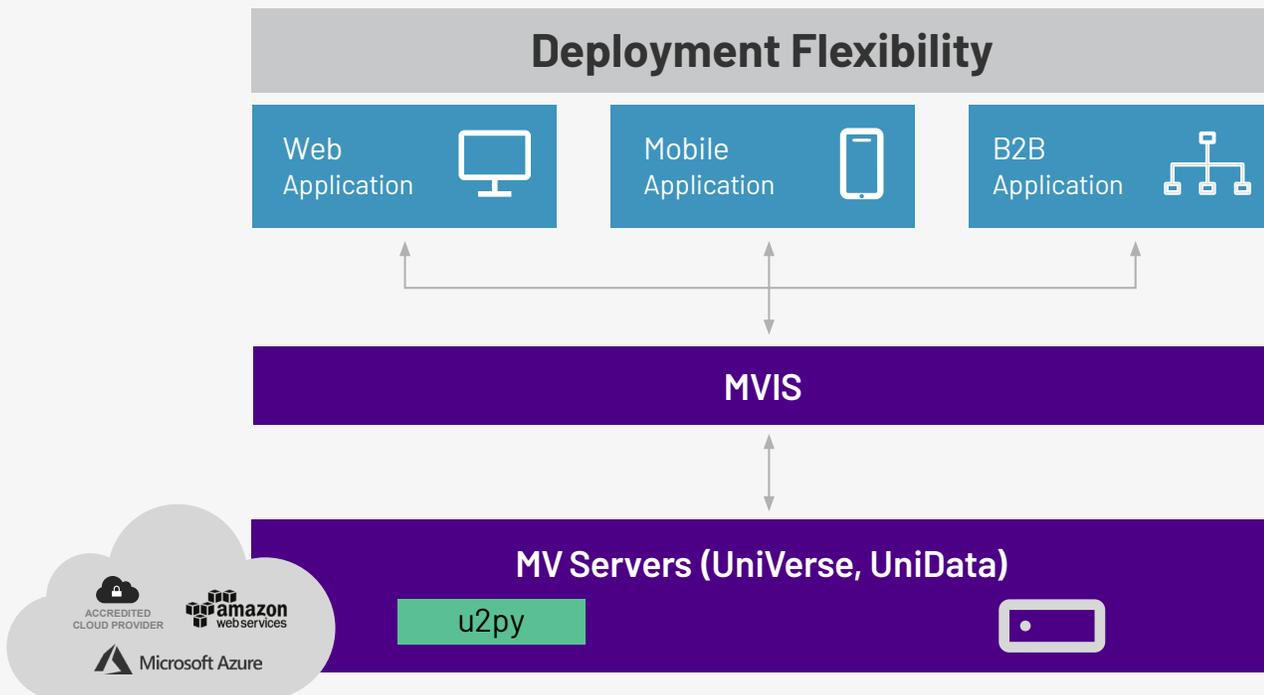
The MVIS architecture supports flexible deployment options. You can run it stand alone on the data server, or on a separate server in front of multiple data servers to support partners who have multiple customers with their own data servers, or direct customers that need DevOps functionality (such as 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 or use a container manager such as Docker.

Figure 2: The MVIS architecture ensures availability, for a better user experience, by using Connection Pools and supporting HADR and elastic scaling

Web Application	Mobile Application	B2B Application

Orchestration (Kubernetes, ECS)

MVIS

RESTful/Swagger Low-Code/No-Code

API and DB Logging/Monitoring

Web Admin Console

Horizontal Scaling and Failover

MV Application (UniVerse, UniData)

SB/XA Services

Figure 3: API and connection logging and 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

Per API request

Performance statistics

Aggregate API request

fluentd

Alerting (Nagios)

Big Data (MongoDB, Hadoop)

Log Management (Splunk, Elastic Search)

Monitoring Systems (Datadog, Graphite)

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.

Tech Specs

Platform	• Windows Server 2012, 2016, 2022 • Linux RHEL7, RHEL8 • AIX 7.1, 7.2
Hardware Requirements	• 2 vCPUs • 2 GB RAM • 2 GB disk
Software Requirements	• Rocket UniVerse 11.3.3 or later, or Rocket UniData 8.2.1 or later • 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)

Rocket