■ ■ ■ white paper

Rocket®

Rocket® U2 High Availability / Disaster Recovery Best Practices

Rocket U2 High Availability /Disaster Recovery Best Practices

Revised October 2014

Table of Contents

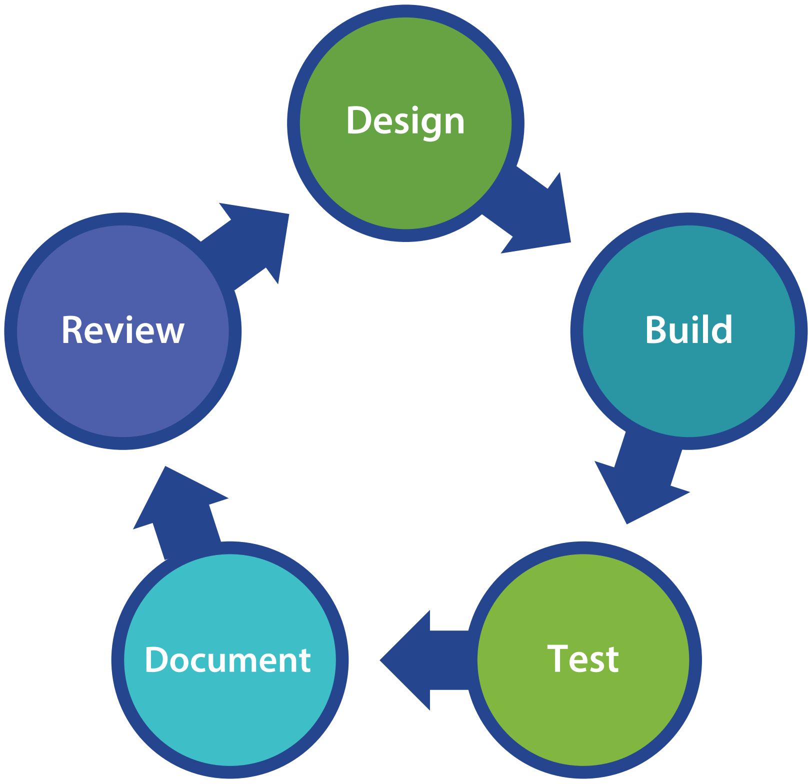
Rocket® U2 High Availability /Disaster Recovery Best Practices

Introduction

Designing and implementing a high availability and/or a disaster recovery architecture can be a challenging task given the disparate, yet related environments that are part of the plan and execution. Your plan should include the hardware, software, firmware, networks, power, and people that make up your architecture. A successful effort begins with clearly defined and thoroughly understood business requirements. Thorough analysis of the business requirements enables you to make intelligent design decisions and develop an architecture that addresses your business needs in the most cost-effective manner.

Once your business requirements are realized, you can begin to design the architecture that will achieve the required levels of availability, performance, scalability, and security. The high availability/disaster recovery (HADR) architecture you choose should have a clearly defined plan for deployment and ongoing management that minimizes complexity and business risk.

This document provides insight into the best practices Rocket MultiValue Professional Services has identified in their extensive experience implementing HADR solutions. *The Rocket Software's U2 High Availability/Disaster Recovery Best Practices provided herein are for use at your discretion. However, we highly recommend that these practices be followed during the implementation and management of your HADR solution for optimum results.*



Operational Considerations

Maximizing availability and recoverability while maintaining optimum performance requires a number of considerations.

In the following sections, we cover several key areas that will impact your results.

Determine Availability and Performance SLAs

First, determine your business need for availability and performance and set agreed upon service level agreements (SLAs). Once determined, document your high availability and performance SLAs and create an outage and solution matrix. Identify possible scheduled and unscheduled outages and document the business's cost of downtime. Establish Recovery Time Objectives (RTOs) and Recovery Point Objectives (RPOs or data loss tolerance) for the outages from business requirements and resource functionality data.

Be sure to include consideration for the following use cases for U2 Replication:

- ❖ Local and remote disaster failover
- ❖ High availability for scheduled outages due to maintenance and upgrades
- ❖ Improved performance by replicating and providing reporting on another server
- ❖ Development and QA environments from replicated production data

Design a High Availability / Disaster Recovery (HADR) Environment

Design and document a high availability environment to achieve the optimal high availability architecture. Consider all the components of your environment: power, hardware, network, storage, firmware, software, database, security, and peripherals.

Checklist:

- ❖ Review the latest certified patch sets available for your environment
- ❖ Document the existing and new environment including:
    - Upgrades and new additions
    - Development and test environments
- ❖ Document your current configurations
- ❖ Develop a plan to test, upgrade, add and implement HADR

Rocket Software has numerous documents, brochures, and webinars, as well as experienced professional services consultants, to help you with each step, from analysis to planning, implementation, review and improvement.

Understand and Design Security Best Practices

A key consideration in HADR is the security of the systems—both physical security and security from cyber-attacks. You should understand current security practices and how those will apply to a replicated environment. Many of the same security requirements will exist, but some may differ on a replicated unmonitored environment. Since replication can be used in a variety of ways—failover, development and QA, and reporting— be aware that these present different configurations and environments to secure.

Focus areas:

- Security around physical and login access
- Administration security
- Sensitive data access and storage

Implementation in a Separate Test and QA Environment

In order to measure performance impacts and to compare different configurations, we recommend setting up your HADR solution in a separate environment, such as one you use for testing and QA. Besides ensuring in advance that your production implementation is performant, this setup will give you high availability for your test and QA environment itself.

Optimally, the disaster recovery testing environment will be a close replica of the production environment - including a replica of the standby environment. Baselines for the environment should be established for performance, as well as storage and other delta analytics. You should validate the application in a test environment and ensure the HADR solution meets or exceeds your functionality, performance, and availability requirements. We also recommend automating the procedure, and also documenting and testing a fallback procedure. This requires comparing metrics captured before and after a patch application on the test system and against metrics captured on the production system.

Real application testing can be used to capture workload on the production system and replay it on the test system. A process to replicate workload to execute functional tests, performance tests, and availability tests must be available and used to understand the effects of replication. Any changes should be validated in the test environment first, including evaluating the effect of changes and the fallback procedures before introducing the changes in the production environment.

A disaster recovery simulation should also be performed and documented to establish actual SLAs and procedures. This simulation will test procedures, failover devices and software as well as provide training, and validate that a recovery can be successful. You will also be able to validate your SLAs through this disaster recovery simulation.

Recovery and Performance

Often the biggest issue when implementing a solution is maintaining performance while meeting your HADR SLAs. is often the biggest issue when implementing a solution. A great deal of your Rocket U2 replication performance and its possible effects on your production performance depends on the modes of replication—Real-time, Immediate, Deferred, the publisher, subscriber, and listener configurations—as well as the published replications groups processes. Many different combinations and configurations depend on fluctuations and other factors in your day-to-day processes. Other considerations include storage updates, timing, and processes, as well as network and other processing considerations.

Recovery Process Design and Documentation

For both scheduled and unscheduled outages, a process should be documented that describes what happens and the person responsible for that action Automate as much of this process as possible using scripts and document any manual steps where automation is not feasible. We recommend documenting both the automated processes and the manual steps required to recover from each type of outage. Include external validation that the recovery was successful before allowing users to resume processing. For each outage type, determine a reasonable recovery time objective.

Outage Type	Recovery Steps	Recovery Time Objective
Server Down		
Network Outage		
Storage Failure		
Human Error		
Security Breach		

Configure Monitoring Infrastructure

To maintain your high availability environment, you should configure the monitoring infrastructure that can detect and react to performance and high availability thresholds. Again, automate wherever possible, although a mix of both automatic and manual alerts is acceptable.

The monitoring infrastructure should do the following:

- Monitor system, network, application, database, and storage statistics
- Monitor performance and service statistics
- Create performance and high availability thresholds as early warning indicators of system or application problems

Establish Escalation Management Procedures

We recommend that you establish escalation management and support procedures so the process is not hindered due to deviations from the documented procedures. Most recovery challenges occur when the primary database or system is not meeting availability or performance SLAs and failover procedures are not automatic or procedures are not successful. Downtime can be prolonged if proper escalation policies are not followed and decisions are not made quickly. Availability is the top priority and a contingency plan should be created to gather sufficient data for future root cause analysis (RCA).

Review and Improvement Cycles

It is important to design and implement regular cycles to review your HADR solutions and look for ways to improve. You will want to regularly test your HADR solution and continuously look for ways to strengthen it, improve performance, or increase automation.

Semiannual or Annual Testing of the Plan

As a comparative analysis, organizations may choose perform semiannual or annual testing depending on the exposure (cost of being down) of the organization related to the operational systems involved. While it is rare that testing of the plan occurs more often than twice a year, it does sometimes happen. Annual testing should be considered mandatory. Without an annual test, existing staff may forget how to accomplish certain tasks and newly hired staff may not have learned the proper procedures to follow in case of an outage.

Use Objective Observers

Objective observers can provide valuable insights to those performing the recovery tasks. Request that observers watch what is occurring and allow them to ask questions of your staff. We have been amazed with the usefulness of objective observers for the insights they provide. Observers do not need to be experienced with your processes and it is helpful to consider that even simple questions can generate insights into recovery procedures that should be added. Ask your observers to validate the recovery processes, both automatic and manual.

white paper

Rocket

www.rocketsoftware.com

info@rocketsoftware.com

Improvement Cycles

A test to improve the recovery processes should be instituted whenever a significant change is made to the environment, business processing cycles, or quantity of work being performed by the system. For example, if your company acquires another organization and the volume of data increases significantly, a recovery test improvement cycle should be performed to ensure there is enough capacity to recover within your SLA requirements.

Allow your staff to self-generate a request for an improvement cycle. If your computing staff is concerned about the ability to recover, then schedule a test to regain their confidence by validation.

Quantify Results

Always quantify the results to ensure you are meeting your SLAs. By having timings quantified, you remove personal assessments that may come into play. Also, corrections are much easier to implement when personalities are removed from the circumstances observed.