QualiSystems

7.0 CloudShell High Availability

Deployment Guide

Release date: March 2016

Document version v1.0

Quali systems

Contents

Overview

This document presents the recommended architecture for implementing CloudShell in a High Availability (HA) environment.

This document should be read in conjunction with the *CloudShell HA Installation and Configuration Guide*.

A failover cluster is a group of independent servers (nodes) that work together to increase the availability and scalability of clustered nodes. The clustered nodes are connected by physical cables and by software. If a disaster occurs and the active cluster node goes down, the clustering solution changes the active node automatically to the standby server and Quali server starts on the new active node.

Use cases

This document covers the following use cases:

- Planned downtime: scheduled by administrator. Used for maintenance purpose. The primary system is manually switched to the standby mode.
- Unplanned downtime: failure of one of the component in the system, automated failover.

Guiding principles

The HA architecture design follows a number of guiding principles that drive the choice of the deployment type for the High Availability solution:

Aspect	Description
Cost	Consolidate applications components, use virtual environment for all components and leverage from existing infrastructure (compute/storage/network).
Performance	Have separate tiers for web/app/database, use load balance (when possible) and use physical servers.
Simplicity	Use a consistent solution across the system for ease of support, for example, Windows clustering.
Layering	No SPOF means layering redundancy at all levels (physical host, networking, hypervisor, OS, application services). This document focuses on application level redundancy and host failure.
SLA	Time to switch from active to standby should be minimal once a failure has been detected.

Architecture

This architecture describes a QualiSystems certified automatic failover solution for all core components of the application, without a Single Point of Failure (SPOF). Alternative solutions are described in each respective section.

Component	Solution
Quali App Server	Quali Server service - Windows Server Clustering (active/passive)
	QuickSearch - Windows Server Clustering (active/active)
License server	Windows Server Clustering (active/standby)
Database Server	Windows Server Clustering (active/standby)
Web Server	Basic, performance setup: Windows ARR load balancing (active/active)
	Advanced setup: Windows ARR load balancing (active/active)
Execution Server	Natively supported by CloudShell (active/active)

The following diagram depicts the main components of the CloudShell application for an advanced set up, as deployed in an end-to-end HA solution within which Web server, license server, execution server and Application server may be co-hosted on each node of the application cluster. Each component may failover independently without impacting the integrity and user experience of the end user.

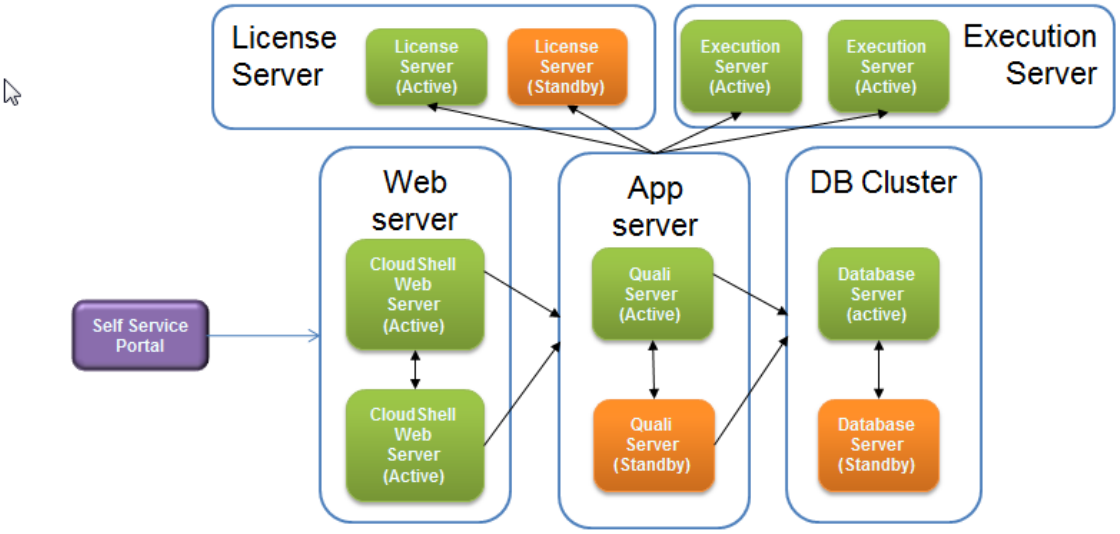


Figure 1: CloudShell HA architecture - logical view

Deployment types

The recommended machine configurations are described in this section.

Deployment considerations

You may use a mix of virtual and physical hosts for each application component, based on performance expectations and hardware availability. It is recommended that the machines inside the cluster be of the same type.

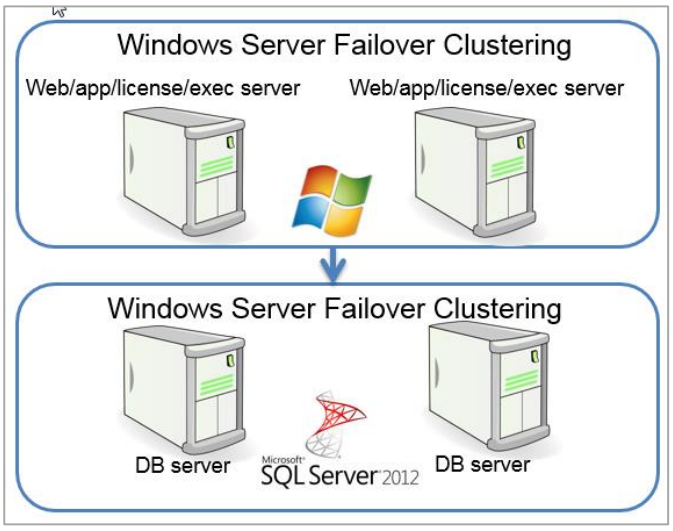
For example, you might have an Application server hosted on physical hosts and a database server hosted on Virtual Machines. For best performance, it is recommended to host the application server and web server on separate machines.

There are different OS platform requirements if application components are hosted on physical as compared to virtual hosts, as described in the following table:

Host Type	Description
Physical host	Windows 2012 Standard Edition on each node of the cluster
Virtual host	Windows 2012 Datacenter Edition (1 license for all nodes) or Windows 2012 Standard Edition for every pair of node

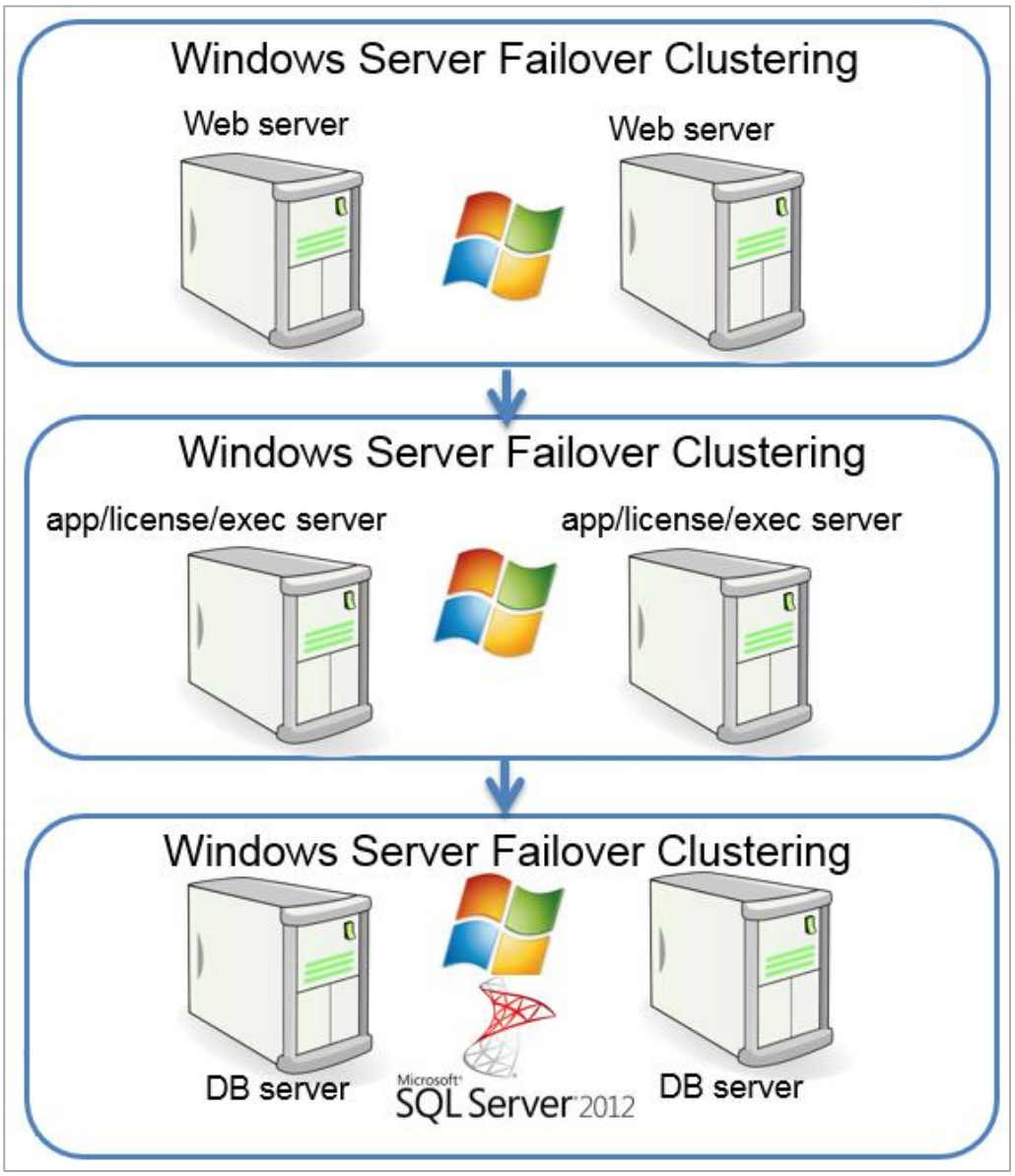
Basic 4 server configuration

Uses two WSFC clustered front end servers, combining the web portal and the Quali application server, with two servers for the database server clustering.



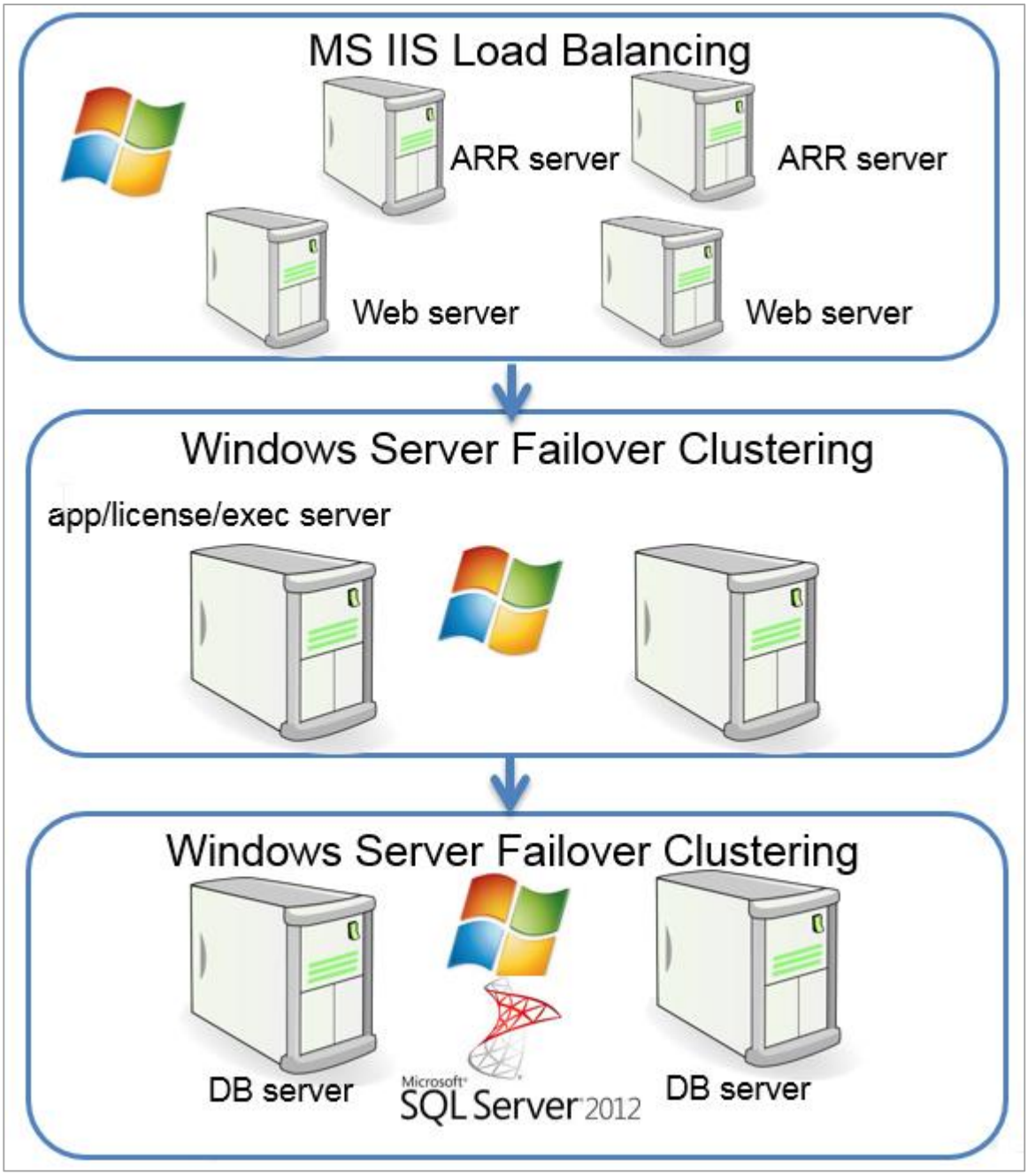
Basic 6 server configuration

Uses two WSFC clustered nodes to host the web portal, two WSFC clustered nodes to host the Quali server and two nodes to host the SQL server database cluster.



Basic 8 server configuration

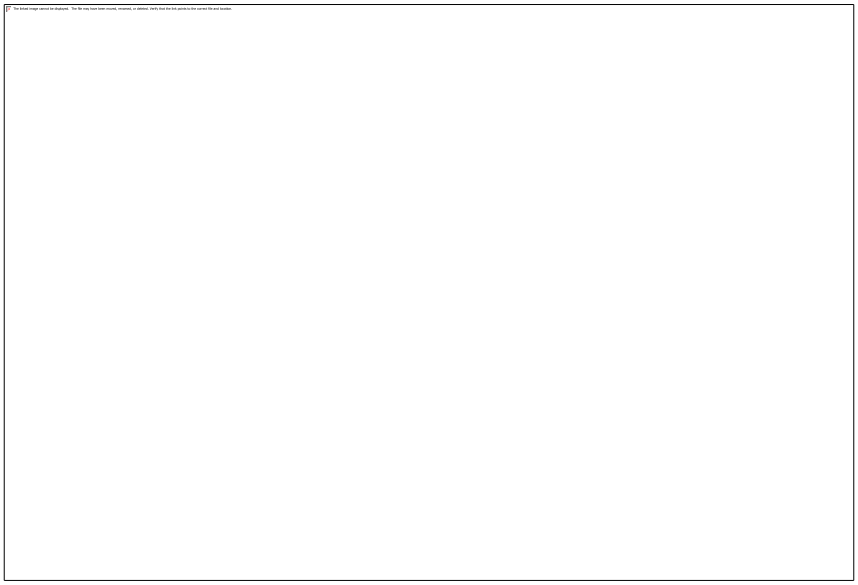
Uses two servers for the ARR load balancing solution, two servers to host the web portal in active/active mode, two WSFC clustered nodes to host the Quali server and two nodes to host the SQL server database cluster.



CloudShell application server clustering

QuickSearch is one of the main components in the CloudShell application server. This component is responsible for the high speed cache that enables the application server to perform high speed calculations for search and resource availability. This cache should be in sync at all times with the Quali DB.

In the High Availability environment, the deployment is with several application servers where at any time only one is actively working (Active-Passive approach). In cases where the application server service has stopped, the QuickSearch cache must be synced at all times with the active application server. This sync ensures that all cluster nodes are using the same data at all times and that the failover time is minimal.



Failure trigger

The application services that are monitored are listed in the following table:

Component	Monitor
Quali App server	Quali Server service
License server	License server service
Web Server – Basic and Performance setup	IIS services, CloudShell IIS Application Pool and CloudShell IIS site
Web server - Advanced	ARR performs a health test of a diagnostic page inside the CloudShell portal
Execution server	Execution server not responding

Before actual failover, one attempt is made to restart the failed service on the local server (Windows) for the Quali App server and License server components.

Additionally, the solution also detects if the entire node fails (is non-responsive).

Performance

Component	Retry Time (since fail is discovered to operational status on the same node) - seconds	Full Failover (since fail is discovered to operational status on another node) - seconds
Quali App server	30	40
License server	10	15
Web Server	15	20

Best practices

This section provides best practice examples.

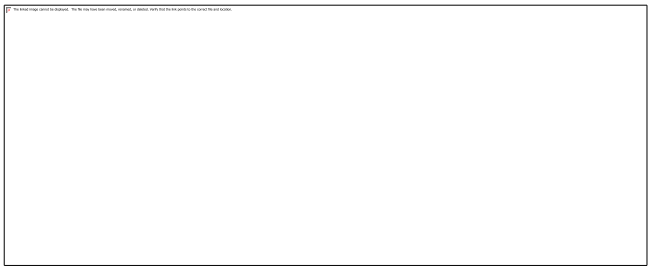
Execution server with HA setup

Tests and Commands, for example, resource commands and environment commands, run in the Execution Server and not in the App server.

These commands usually use TestShell API or Quali API to communicate with and use methods in Quali server (for example GetReservationDetails). To use the APIs, the Quali Server address or IP must be provided in advance, connect to the server and then run the code.

In an HA scenario, you cannot predict in advance which App server is the one that is online during the command execution phase in the Execution Server, so therefore the virtual front address of the App server must be provided.

By using the **connectionInfo** matrix in CloudShell Resource Manager drivers, there is no need to change all drivers to use the virtual front specific IP. An example of the matrix is depicted below:

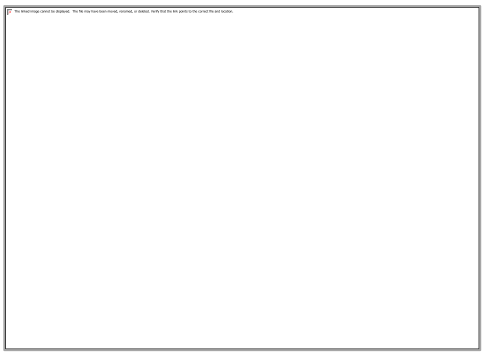


The best practice is to use the values in the matrix to operate and connect to the App server API's.

When setting up HA, App server virtual front connectivity details are specified, for example:

This IP or address (DNS name) is also used in the 'connectivityInfo' matrix when running commands. For example, in the image below, a simple driver is running in a HA setup and is printing the matrix content:

This IP address is the App server virtual front IP, as you can see in the Windows cluster manager:



Appendix

This appendix contains reference information and a glossary of terms.

References

Topic	URL
Application Request Routing Version 2 Overview	http://www.iis.net/learn/extensions/planning-for-arr/application-request-routing-version-2-overview
CloudShell distributed execution server	https://support.qualisystems.com/entries/87064507-Distributed-Provisioning-DisPro-CloudShell-6-2-feature-
Create a New Network Load Balancing Cluster	https://technet.microsoft.com/en-us/library/cc771008.aspx
Create a new Network Load Balancing Port Rule	https://technet.microsoft.com/en-us/library/cc733056.aspx
Define and Configure an Application Request Routing Server Farm – step-by-step guide	http://www.iis.net/learn/extensions/configuring-application-request-routing-(arr)/define-and-configure-an-application-request-routing-server-farm
Health check	http://blogs.iis.net/richma/application-request-routing-health-check-features
Setting up TestShell Portal on IIS including HTTPS	https://support.qualisystems.com/entries/61196243-Setting-up-Testshell-Portal-on-IIS-including-HTTPS-
Virtual environment deployment on VMware vCenter 5.5 or above, with HA clustering configured across two different ESXi hosts	https://pubs.vmware.com/vsphere-55/index.jsp#com.vmware.vsphere.avail.doc/GUID-E90B8A4A-BAE1-4094-8D92-8C5570FE5D8C.html
Windows load balancing manager	https://technet.microsoft.com/en-us/library/cc776931%28v=ws.10%29.aspx
Windows Server Manager Step-by-Step Guide	https://technet.microsoft.com/en-us/library/cc753762(v=ws.10).aspx

Documentation

Additional technical documentation is available in the QualiSystems' Download Center.

Operational documentation for all Single Sign-On applications is available by clicking the Help option in any CloudShell application.

For our discussion forums, you can access the QualiSystems Customer Portal.

Glossary

Terms used in this guide are described in the following table.

Term	Description
Active-Active	All nodes in the cluster are active. A load balancing algorithm/policy determines the preferred node for a given session. This is a more scalable architecture. However, it is more complex to manage.
Active-Passive	A fully redundant instance of each node is present. The passive node is brought online when its associated primary node fails.
Active-Standby	One node in the cluster is active. The other node is inactive until failover is triggered (warm standby).
AlwaysOn Availability Groups	A high-availability and disaster-recovery solution that provides an enterprise-level alternative to MSSQL database mirroring.
ARR	Active Request Routing. This is an IIS server native load balancing solution.
ARR Server Farm	A logical group of application servers where HTTP requests are routed based on HTTP inspection rules and load balance algorithm.
Availability databases	A failover environment for a discrete set of user databases (an availability group) that fail over together.
Availability replica	An instantiation of an availability group that is hosted by a specific instance of SQL Server and that maintains a local copy of each availability database that belongs to the availability group. Two types of availability replicas exist: a single primary replica and one to four secondary replicas. The server instances that host the availability replicas for a given availability group must reside on different nodes of a single Windows Server Failover Clustering (WSFC) cluster.
NLB	Network Load Balancing. Use the NLB Manager to create and manage NLB clusters from a single computer.
SAN	Storage area network, dedicated network used to enhance storage devices. It is a high-speed network, providing a direct connection between servers and storage, including shared storage, clusters, and disaster-recovery devices.
SPOF	Single Point of Failure.
Warm Upgrade	An administrator is able to upgrade one node of the cluster to a new version of CloudShell (Quali Server) while the other node is active, then fall back to the new node and upgrade the standby without any downtime for the end user.

Term	Description
Windows load balancing manager	Windows load balancing manager enables you to create and manage Network Load Balancing (NLB) clusters from a single computer. By centralizing NLB administration tasks, many common configuration errors are eliminated.
WSFC	Windows Server Failover Clustering