Inner Range High Availability and Fault Tolerant Solution

Document created with reference to Software version V17.1

Inner Range Pty Ltd
ABN 26 007 103 933

1 Millennium Court, Knoxfield, Victoria 3180, Australia
PO Box 9292, Scoresby, Victoria 3179, Australia
Telephone: +61 3 9780 4300 Facsimile: +61 3 9753 3499
Email: enquiries@innerrange.com Web: www.innerrange.com

ASIAL MEMBER
AUSTRALIAN SECURITY INDUSTRY ASSOCIATION LIMITED
GOLD MEMBER

Certified System
Quality
ISO 9001

Minimum requirements

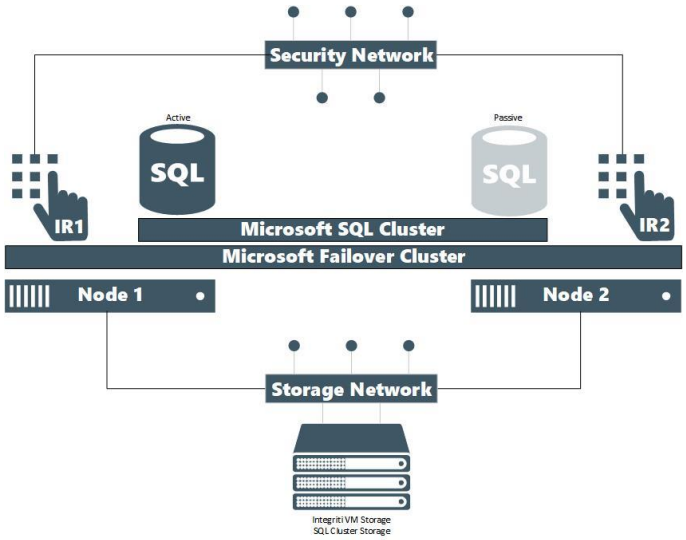
- Integriti Additional Server Node (High Availability) license (part no. 996965).
- Requires Integriti Professional version 17.1 or later.

Introduction

The Inner Range Integriti system has been designed to allow configuration in a manner that provides both high availability and fault tolerance. Multiple installations of the Integriti services are installed, with both Integriti controllers and clients being able to connect to any available server. In the event that one of the Integriti servers is no longer available, the system will automatically reconnect to one of the other servers still online.

Inner Range provides this solution using Microsoft Failover Clustering and a Microsoft SQL Cluster. An example diagram depicting this configuration can be seen below. It is important to note that the SQL cluster is built upon the Microsoft Failover Clustering configuration. It is recommended to have the Inner Range virtual machines, represented by IR1 and IR2 within the diagram, also running upon this failover cluster. However these measures are not required. This configuration has a requirement of a minimum of two nodes, as is displayed below in the diagram. The number of nodes can be extended to a maximum of 64 if the need should arise. The "Storage Network" shown is the storage solution underlying the Microsoft Failover Cluster on which the customer has deployed the SQL Cluster used by Integriti. It is not in addition to the storage used by the Microsoft SQL cluster.

For detailed information on network ports and connection methodology used by Integriti, please refer to the separate "Integriti Connectivity Guide" document.



What is Integriti Active Clustering?

The Integriti system consists of three server components:

1. Application Server
2. Controller Server
3. Integration Server

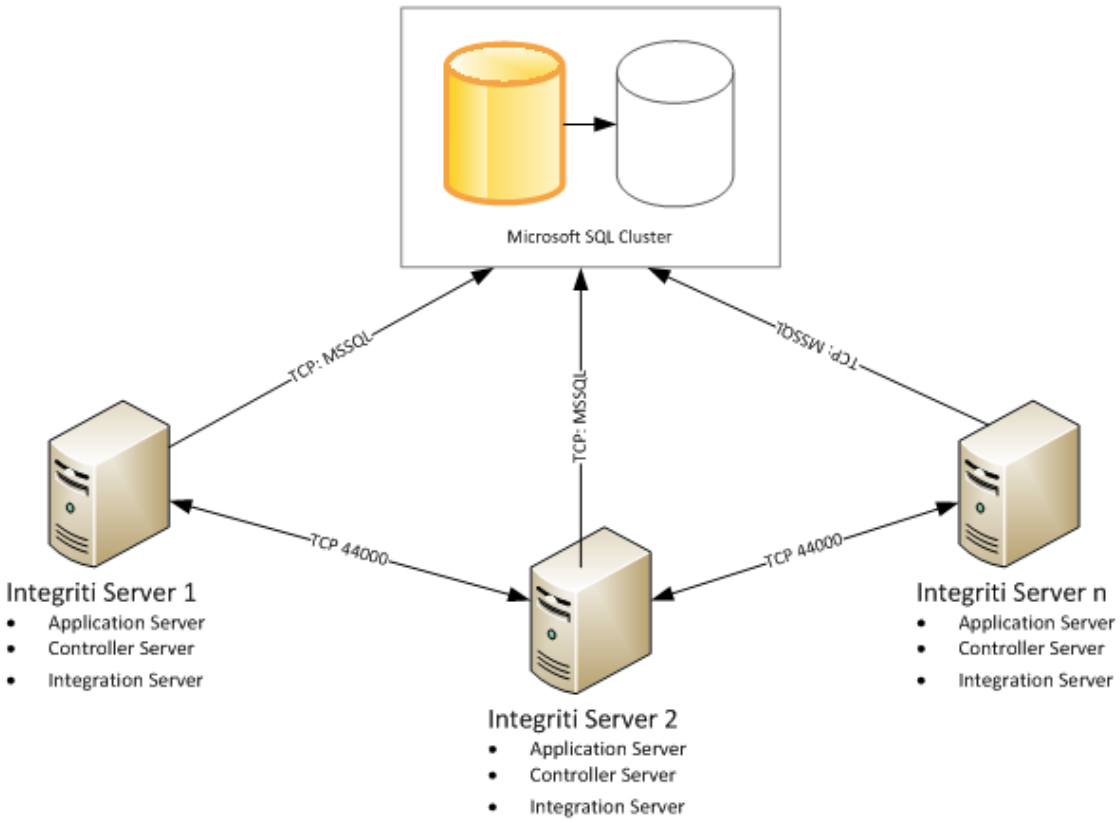
Each service performs a different role, but all maintain an independent connection to the SQL cluster, as well as a connection to all application servers in the cluster.

With Integriti Active Clustering, all services are always intended to be active, and are considered equal, rather than arranged in a primary / backup configuration.

When adding a new server to the cluster, an Additional Server Node (High Availability) license (part no. 996965) must be purchased via the Inner Range KeyPoint System. This will then generate a license key which allows activation of an additional application server on the same database cluster as the initial machine used to configure the system and create the schema (as per a usual installation).

This license key includes data which allows the machine to be cryptographically paired with the cluster. Pairing the machine to the cluster prevents unauthorised servers from attaching.

The diagram below depicts the connection ports and methodology of the high availability setup.



Adding a Server to Integriti Active Clustering

Under most circumstances, the only configuration required to add a new node to a clustered installation is a database connection string. The database connection string allows access from the server to the SQL cluster, along with the Additional Server Node License.

Each service connects to the SQL cluster and registers their connection details (IP Address, hostname, encryption keys, etc.) using the database as a conduit for disseminating the information necessary for the servers to establish a bi-directional TCP link between themselves. Should any server not be directly routable or behind a NAT, the address information can be configured manually in the database to allow servers to traverse NATs or connect to alternate IP Addresses.

Once each service has connected to the database, all controller servers and integration servers use the information published in the database to establish a TCP on port 44000 to allow real-time communications between each component of the cluster.

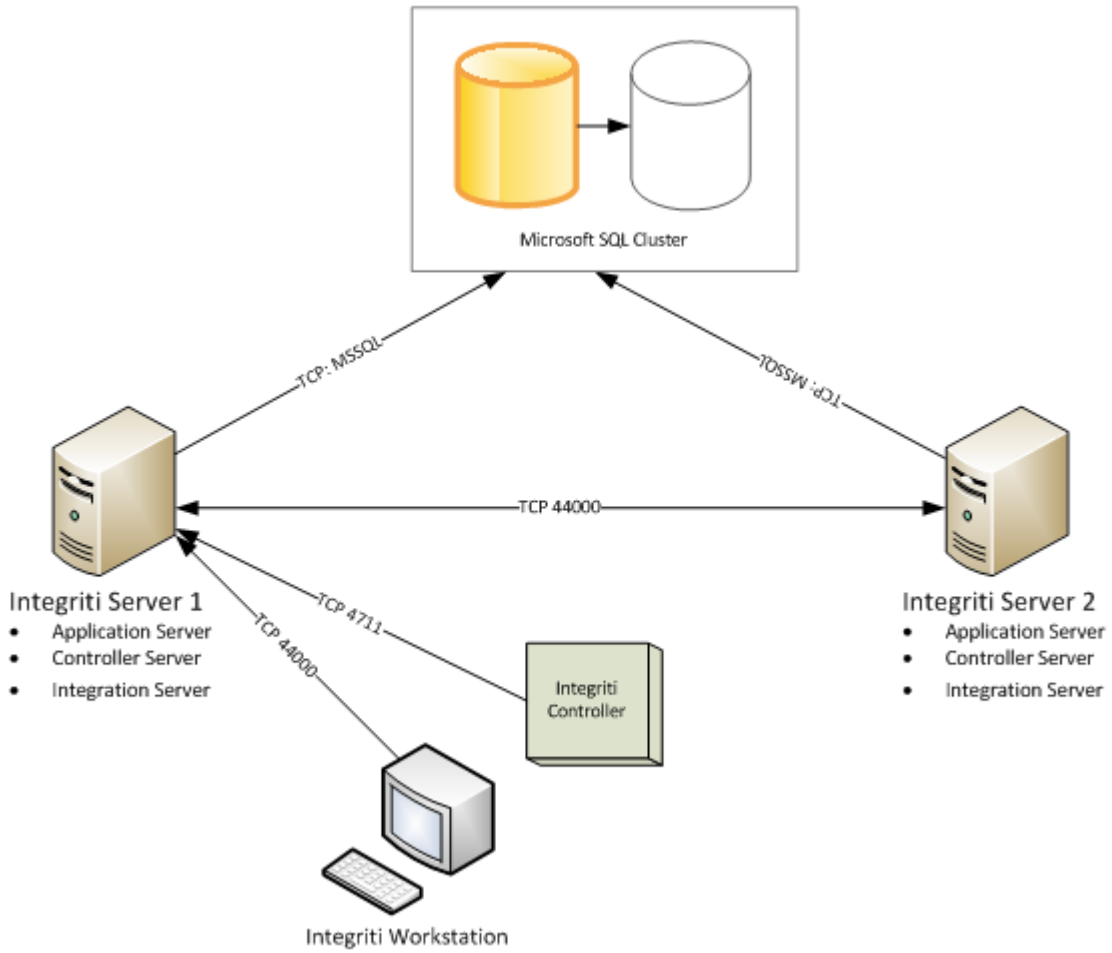
Integriti Active Clustering is designed such that all data pertinent to a connection is stored in the shared database, which allows all servers to be available at all times, both for the purposes of providing high-availability and load balancing. If any node was to fail, any clients or controllers connected to that node would automatically switch to any other available node in the cluster.

Note: Without the use of DNS, there is a limit of four pre-defined IP Addressable end-points per controller.

Should additional resources be required, or additional layers of application server redundancy are required, more servers can be added by purchasing Additional Server Node Licenses and installing the software on additional hardware, then entering the connection string of the SQL cluster.

What happens in the event of a node failure?

If one of the Integriti servers becomes unavailable, any controllers or client applications connected to it will automatically search for another available Integriti server node.



Each Integriti controller has a number (up to four) of prioritised static IP addresses for controller servers. In the event of one of these being unreachable, the controller will attempt to connect to these in sequence, connecting to the first that responds.

The controller remains connected to this alternate controller server until that becomes unreachable or a manual restart of communications is invoked, whereupon the controller will again attempt to connect to the first server on the list and any others in order.

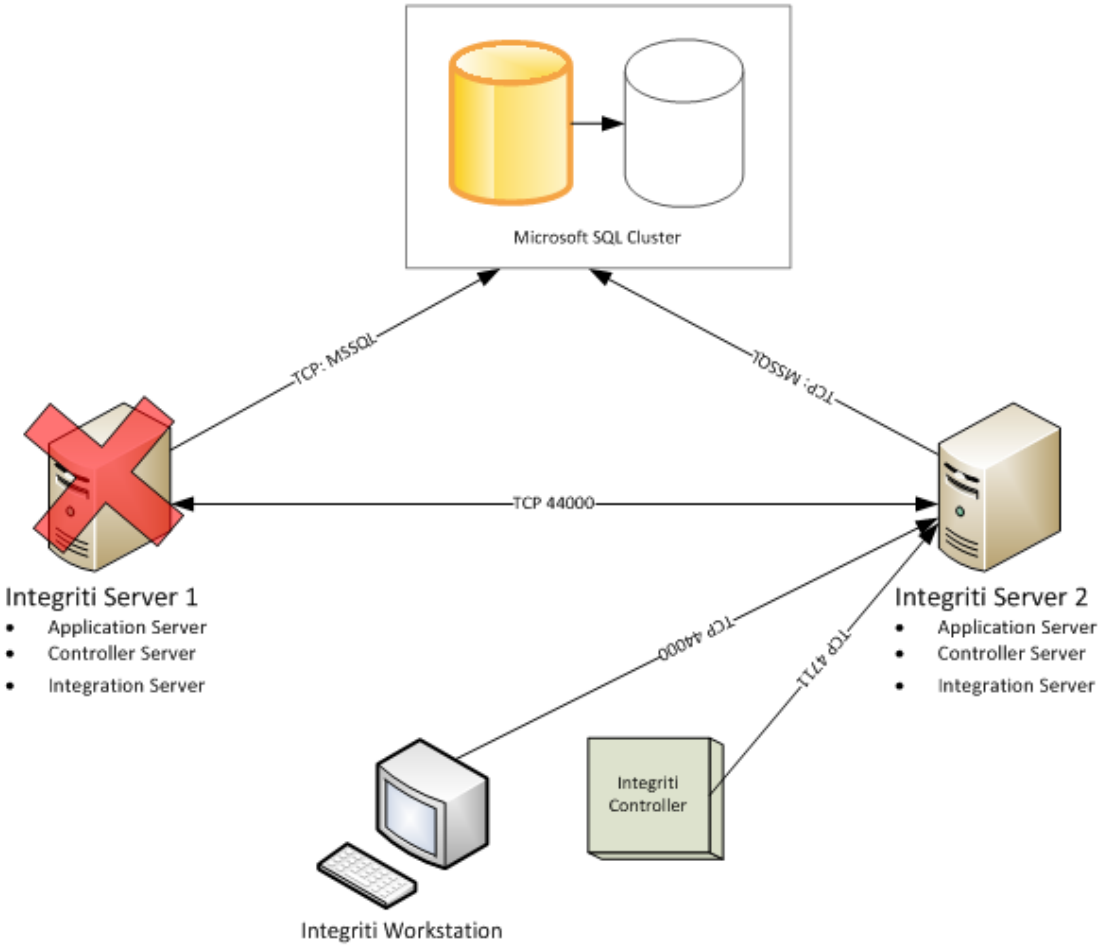
Alternatively, as the Integriti controller can be connected to a single URL, instead of a sequence of IP addresses, the customer's IT department can set up their DNS server(s) to provide this feature of the HA solution using DNS to target a backup server on failover.

Each Integriti client has a set of URLs addresses for Integriti Application Servers. On start-up, or connection failure, the client attempts to connect to the servers in this set, and connects to the first to respond.

Once connection to another Integriti node is established, regular operation is automatically resumed, as any data stored in the shared database that is related to the connection is stored in the shared SQL cluster.

Each Integriti Controller Server must be reachable from each Integriti controller via IP or DNS. Each Application Server must be reachable from each client via IP or DNS.

If DNS is used to provide the route for controllers to reach a backup controller server, it is the responsibility of the customer to ensure that this mechanism propagates changes in the required response time.



Once a controller or workstation has moved to another server, it will remain using that server until either its connection is lost (due to network or node failure) or until administrative action is taken to shift it to another working node.

What is Microsoft Failover Clustering?

Microsoft Failover Clustering is a cooperative system component that enables applications, services, and even scripts to increase their availability without human intervention. Anywhere from one to sixty-four physical hosts are combined into a single unit, called a “failover cluster”.

These hosts share one or more common networks and at least one shared storage location. One or more of the previously listed services running on these physical hosts are presented to Microsoft Failover Clustering as roles. A “role” is any service running on the computer. The cluster can move these roles from one host to another very quickly in response to commands, or automatically in response to environmental events.

Many roles, including Inner Range Integriti servers running upon a Hyper-V virtual machine, can be intentionally moved without any service interruption. All roles can be quickly and automatically (without human intervention) moved to a surviving node in the event of a node failure. For these reasons, clusters are called “high availability technology” because they can decrease – and potentially eliminate – the amount of time that a service is unreachable outside of its own scheduled maintenance windows.

Microsoft Failover Clustering is always in an active/passive configuration. This means that any single role can only operate on a single cluster node at any given time. A cluster can operate multiple roles simultaneously, however. Depending on the role type, it may be possible for roles to operate independently on separate hosts. As this specifically applies to Hyper-V, an individual virtual machine is considered only a single role, not the entire Hyper-V service. Therefore, a single virtual machine can only run on one Hyper-V host at any given time, but each host can run multiple virtual machines simultaneously.

If a host crashes, all of its virtual machines will also crash, but they will be automatically restarted on another cluster host. Because a virtual machine cannot run on two hosts simultaneously, Hyper-V virtual machines are not considered fault tolerant. This fault tolerance is provided within the Integriti product itself with multiple options for servers being provided to both Integriti controllers and Integriti software clients. In the event of a virtual machine no longer being available, the Integriti controllers and clients will detect the failure and simply reconnect to one of the remaining online virtual machines.

The Inner Range Integriti server software will typically be installed as a Hyper-V virtual machine upon the Microsoft Failover Cluster. Each one of these virtual machines can service the Integriti controllers and clients while connecting back to a common Microsoft SQL Database. This database is configured as an SQL cluster to protect it against hardware and software failures.

What is an SQL Cluster?

Microsoft Windows Failover Clustering can also be used to increase the uptime of SQL Server instances. One Failover Cluster Member is identified as the active node, on which a SQL Server instance is running the production workload, and the other is a passive node, on which SQL Server is installed but not running. If the SQL Server instance on the active node fails, the passive node becomes the active node and begins to run the SQL Server production workload with some minimal failover downtime.

This definition is straightforward, but it has a lot of unclear implications, which is where many clustering misunderstandings arise.

What SQL Clustering Can Do

SQL clustering is designed to improve the availability of the physical server hardware, operating system, and SQL Server instances but not the shared storage. Should any of these aspects fail, the SQL Server instance fails over. The other node in a cluster automatically takes over the failed SQL Server instance to reduce downtime to a minimum.

Additionally, the use of a Windows Failover Cluster can help reduce downtime when maintenance is performed on cluster nodes. For example, if hardware needs to be updated on a physical server or a new service pack installed on the operating system, this can be achieved one node at a time.

What Clustering Cannot Do

Clustering is just one part of many important and required pieces in a puzzle to ensure high availability. Other aspects of high availability, such as ensuring redundancy in all hardware components, are just as important. Without hardware redundancy, the most sophisticated cluster solution in the world can fail.

Some Database Administrators (DBA's) believe that clustering can reduce downtime to zero. This is not the case; clustering can mitigate downtime, but it cannot eliminate it. For example, the failover itself causes an outage lasting from seconds to a few minutes while the SQL Server services are stopped on one node then started on the other node and database recovery is performed.

Clustering is also not designed to intrinsically protect data, as the shared storage is a single point of failover in clustering. Data must be protected using other options, such as backups, log shipping, or disk mirroring. In actuality, the same database drives are shared, albeit without being seen at the same time, by all servers in the cluster, so corruption in one would carry over to the others.

Clustering is usually deployed within the confines of a data centre, but can be used over geographic distances (geoclusters). Geoclusters are more sophisticated and typically require the involvement of the storage vendor to accomplish a storage solution spanning a geographic distance and synchronisation across multiple disk arrays.

Frequently Asked Questions

Q: How long does it take to fail over?

A: There are several factors to consider in calculating failover time. One must adjust for the time for the SQL Server instance's service to go down on one node, be initiated on another node, and start up. This time for instances to start and stop includes normal database recovery times. If failover times are needed within an SLA, the time for failover testing should be conducted in a planned downtime. However, one should also estimate how long failover might take if occurring during peak load.

Q: Can I use geo-clustering for disaster recovery?

A: Yes, but it requires some sophisticated setup. Most SQL Server clusters are installed in the same subnet in a single datacentre and are suitable for high availability.

Q: Why is shared storage so important?

A: Shared storage is important because not everyone has robust amounts of storage space available. It's important to make sure the shared storage solution has redundancy in all the right places, because in a failover cluster, shared storage is a single point of failure. This also means that if data is corrupted, it will be corrupted no matter which node accesses it.