Article
Mark Bolinsky · Jul 1, 2016 17m read

InterSystems Example Reference Architecture for Microsoft Azure Resource Manager (ARM)

++Update: August 2, 2018

This article provides a reference architecture as a sample for providing robust performing and highly available applications based on InterSystems Technologies that are applicable to Caché, Ensemble, HealthShare, TrakCare, and associated embedded technologies such as DeepSee, iKnow, Zen and Zen Mojo.

Azure has two different deployment models for creating and working with resources: Azure Classic and Azure Resource Manager. The information detailed in this article is based on the Azure Resource Manager model (ARM).

Summary

Microsoft Azure cloud platform provides a feature rich environment for Infrastructure-as-a-Service (IaaS) as a cloud offering fully capable of supporting all of InterSystems products. Care must be taken, as with any platform or deployment model, to ensure all aspects of an environment are considered such as performance, availability, operations, and management procedures. Specifics of each of those areas will be covered in this article.

Performance

Within Azure ARM there are several options available for compute virtual machines (VMs) and associated storage options, and the most directly related to InterSystems products are network attached IaaS disks stored as VHD files in Azure page blob storage. There are several other options such as *Blob (block)*, *File* and others, however those are more specific to an individual application ' s requirements rather than supporting the operations of Caché. There are two types of storage where the disks are stored: Premium and Standard. Premium storage is more suited for production workloads that require guaranteed predictable low-latency Input/Output Operations per Second (IOPs) and throughput. Standard storage is a more economical option for non-production or archive type workloads. Care must be taken when selecting a particular VM type because not all VM types can have access to premium storage.

Virtual IP Address and Automatic Failover

Most IaaS cloud providers lacked the ability to provide for a Virtual IP (VIP) address that is typically used in database failover designs. To address this, several of the most commonly used connectivity methods, specifically ECP clients and CSP Gateways, have been enhanced within Caché to no longer rely on VIP capabilities making them mirror-aware.

Connectivity methods such as xDBC, direct TCP/IP sockets, or other direct connect protocols, require the use of a VIP. To address those, InterSystems database mirroring technology makes it possible to provide automatic failover for those connectivity methods within Azure using APIs to interact with the Azure Internal Load Balancer (ILB) to achieve VIP-like functionality, thus providing a complete and robust high availability design within Azure. Details of this can be found in the Community article Database Mirroring without a Virtual IP address.

Backup Operations

Performing a backup using either traditional file-level or snapshot based backups can be a challenge in cloud deployments. This can now be achieved within Azure ARM platform using Azure Backup and Azure Automation

Run Books along with InterSystems External Freeze and Thaw API capabilities to allow for true 24x7 operational resiliency and assurance of clean regular backups. Alternatively, many of the third-party backup tools available on the market can be used by deploying backup agents within the VM itself and leveraging file-level backups in conjunction with Logical Volume Manager (LVM) snapshots.

Example Architecture

As part of this document, a sample Azure architecture is provided as a starting point for your application specific deployment, and can be used as a guideline for numerous deployment possibilities. This reference architecture demonstrates a highly robust Caché database deployment including database mirror members for high availability, application servers using InterSystems Enterprise Cache Protocol (ECP), web servers with InterSystems CSP Gateway, and both internal and external Azure load balancers.

Azure Architecture

Deploying any Caché based application on Microsoft Azure requires some specific considerations in certain areas. The section discusses these areas that need to be considered in addition to any regular technical requirements you may have for your application.

Two examples are being provided in this document one based on *InterSystems TrakCare* unified healthcare information system, and another option based on a complete InterSystems HealthShare health informatics platform deployment including: *Information Exchange, Patient Index, Health Insight, Personal Community, and Health Connect*.

Virtual Machines

Azure virtual machines (VMs) are available in two tiers: basic and standard. Both types offer a choice of sizes. The basic tier does not provide some capabilities available in the standard tier, such as load balancing and auto-scaling. For this reason, the standard tier is used for TrakCare deployments.

Standard tier VMs come in various sizes grouped in different series, i.e. A, D, DS, F, FS, G, and GS. The DS, GS, and new FS sizes support the use of Azure Premium Storage.

Production servers typically need to use Premium Storage for reliable, low-latency and high-performance. For this reason, the example TrakCare and HealthShare deployment architectures detailed in this document will be using either FS, DS or GS series VMs. Note that not all virtual machine sizes are available in all regions.

For more details of sizes for virtual machines see:

- Windows Virtual Machine Sizes
- Linux Virtual Machine Sizes

Storage

Azure Premium Storage is required for TrakCare and HealthShare servers. Premium Storage stores data on Solid State Drives (SSDs) and provides high throughput at low latencies, whereas Standard Storage stores data on Hard Disk Drives (HDDs) resulting in lower performance levels.

Azure Storage is a redundant and highly available system, however, it is important to notice that Availability Sets currently don ' t provide redundancy across storage fault domains and in rare circumstances this can lead to issues. Microsoft has mitigation workarounds and is working on making this process widely available and easier to end-customers. It is advisable to work directly with your local Microsoft team to determine if any mitigation is required.

When a disk is provisioned against a premium storage account, IOPS and throughput, (bandwidth) depends on the size of the disk. Currently, there are three types of premium storage disks: P10, P20, and P30. Each one has specific limits for IOPS and throughput as specified in the following table.

Premium Disks Type	P4	P6	P10	P15	P20	P30
Disk Size	32GB	64GB	128GB	256GB	512GB	1024GB
IOPS per disk	120	240	500	1100	2300	5000
Throughput per disk	25MB/s	50MB/s	100MB/s	125MB/s	150MB/s	200MB/s

Note
: Ensure there is sufficient bandwidth available on a given VM to drive the disk traffic.
For example, a STANDARD_DS13 VM has 256 MB per second dedicated bandwidth available for all premium storage disk traffic.
That means four P30 premium storage disks attached to this VM have a throughput limit of 256 MB per second
and not the 800 MB per second that four P30 disks could theoretically provide.

For more details and limits on premium storage disks, including provisioned capacity, performance, sizes, IO sizes, Cache hits, throughput targets, and throttling see:

- Premium Storage

High Availability

InterSystems recommends having two or more virtual machines in a defined Availability Set. This configuration is required because during either a planned or unplanned maintenance event, at least one virtual machine will be available to meet the 99.95% Azure SLA. This is important because during data center updates, VMs are brought down in parallel, upgraded, and brought back online in no particular order leaving the application unavailable during this maintenance window.

Therefore, a highly available architecture requires two of every server, i.e. load balanced web servers, database mirrors, multiple application servers and so on.

For more information on Azure high availability best practices see:

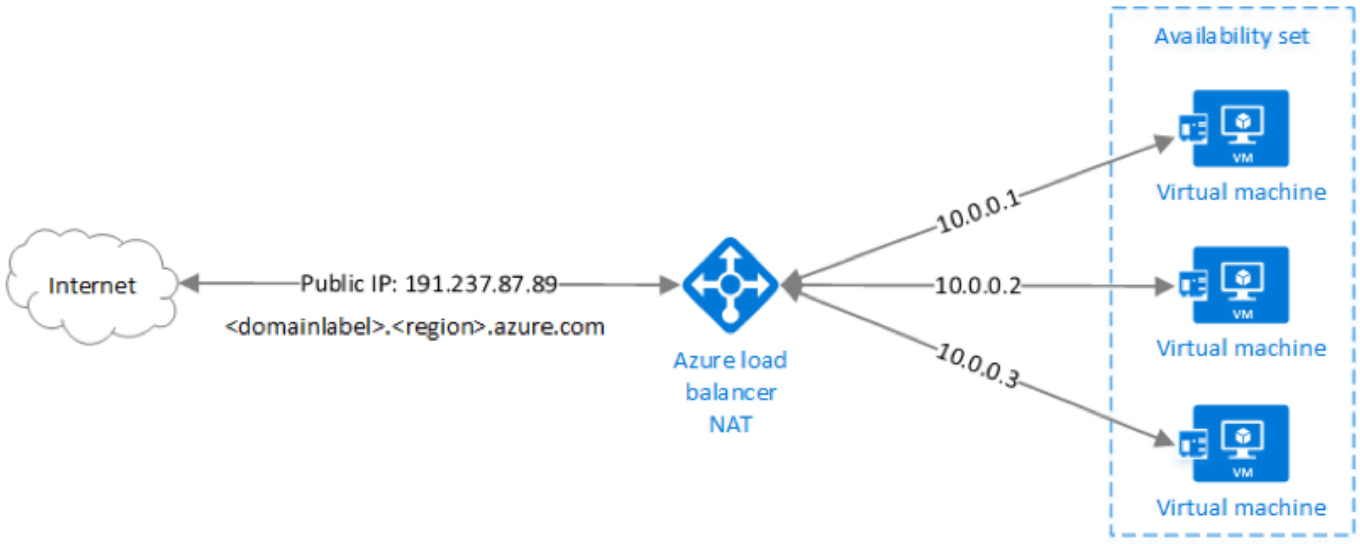
- Managing Availability

Web Server Load Balancing

External and internal load balanced web servers may be required for your Caché based application. External load balancers are used for access over the Internet or WAN (VPN or Express Route) and internal load balancers are potentially used for internal traffic. The Azure load balancer is a Layer-4 (TCP, UDP) type load balancer that distributes incoming traffic among healthy service instances in cloud services or virtual machines defined in a load balancer set.

The web server load balancers must be configured with client IP address session persistence (2 tuple) and the shortest probe timeout possible, which is currently 5 seconds. TrakCare requires session persistence for the period a user is logged in.

The following diagram provided by Microsoft demonstrates a simple example of the Azure Load Balancer within an ARM deployment model.



For more information on Azure load balancer features such as distribution algorithm, port forwarding, service monitoring, Source NAT, and different types of available load balancers see:

- Load Balancer Overview

In addition to the Azure external load balancer, Azure provides the Azure Application Gateway. The Application Gateway is a L7 load balancer (HTTP/HTPS) with support for cookie-based session affinity and SSL termination (SSL offload). SSL offloading removes the encryption/decryption overhead from the Web servers, since the SSL connection is terminated at the load balancer. This approach simplifies management as the SSL certificate is deployed and managed in the getaway instead of all the nodes in the web farm.

For more information, see:

- Application Gateway overview
- Configure an Application Gateway for SSL offload by using Azure Resource Manager

Database Mirroring

When deploying Caché based applications on Azure, providing high availability for the Caché database server requires the use of synchronous database mirroring to provide high availability in a given primary Azure region and potentially asynchronous database mirroring to replicate data to a hot standby in a secondary Azure region for disaster recovery depending on your uptime service level agreements requirements.

A database mirror is a logical grouping of two database systems, known as failover members, which are physically independent systems connected only by a network. After arbitrating between the two systems, the mirror automatically designates one of them as the primary system; the other one automatically becomes the backup system. External client workstations or other computers connect to the mirror through the mirror Virtual IP (VIP), which is specified during mirroring configuration. The mirror VIP is automatically bound to an interface on the primary system of the mirror.

```
Note: In Azure, it is not possible to configure the mirror VIP, so an alternative sol
ution has been devised.
```

The current recommendation for deploying a database mirror in Azure is to configure three VMs (primary, backup, arbiter) in the same Azure Availability Set. This ensures that at any given time, Azure will guarantee external connectivity with at least two of these VMs with a 99.95% SLA, and that each will be in different update and fault domains. This provides adequate isolation and redundancy of the database data itself.

Additional details on can be found here:

- Azure Availability Sets
- Azure Server Level Agreements (SLAs)

A challenge within any IaaS cloud provider, including Azure, is the handling of automatic failover of the client connections to the application with the absence of Virtual IP capabilities. To retain automatic failover for client connections a couple directions have been taken.

Firstly, InterSystems has enhanced the CSP gateway to become mirror-aware so connectivity from a web server with the CSP Gateway to a database server no longer requires a VIP. The CSP gateway will auto-negotiate with both the of the mirror members and redirect to the appropriate member whichever is the primary mirror member. This goes along with the already mirror-aware capabilities of ECP clients if using them.

Secondly, connectivity outside of the CSP Gateways and ECP clients still requires a VIP-like capability. InterSystems recommends the use of the polling method with the *mirrorstatus.cxw* health check status page detailed in the community article Database Mirroring without a Virtual IP address.

The Azure Internal Load Balancer (ILB) will provide a single IP address as a VIP-like capability to direct all network traffic to the primary mirror member. The ILB will only distribute traffic to the primary mirror member. This method does not rely on polling, and allows for an immediate redirection upon any mirror member within a mirror configuration becoming the primary member. Polling may be used in conjunction with this method is some DR scenarios using Azure Traffic Manager.

Backup and Restore

There are multiple options available for backup operations. The following three options are viable for your Azure deployment with InterSystems products. The first two options incorporate a snapshot type procedure which involves suspending database writes to disk prior to create the snapshot and then resuming updates once the snapshot was successful. The following high-level steps are taken to create a clean backup using either of the snapshot methods:

- Pause writes to the database via database Freeze API call.
- Create snapshots of the OS + data disks.
- Resume Caché writes via database Thaw API call.
- Backup facility archives to backup location

Additional steps such as integrity checks can be added on a periodic interval to ensure clean and consistent backup.

The decision points on which option to use depends on the operational requirements and policies of your organization. InterSystems is available to discuss the various options in more detail.

Azure Backup

Backup operations can now be achieved within Azure ARM platform using Azure Backup and Azure Automation Runbooks along with InterSystems External Freeze and Thaw API capabilities to allow for true 24x7 operational resiliency and assurance of clean regular backups. Details for managing and automating Azure Backups can be found here.

Logical Volume Manager Snapshots

Alternatively, many of the third-party backup tools available on the market can be used by deploying individual backup agents within the VM itself and leveraging file-level backups in conjunction with Logical Volume Manager (LVM) snapshots.

One of the major benefits to this model is having the ability to have file-level restores of either Windows or Linux based VMs. A couple of points to note with this solution, is since Azure and most other IaaS cloud providers do not provide tape media, all backup repositories are disk-based for short term archiving and have the ability to leverage blob or bucket type low cost storage for long-term retention (LTR). It is highly recommended if using this method to use a backup product that supports de-duplication technologies to make the most efficient use of disk-based backup repositories.

Some examples of these backup products with cloud support include but is not limited to: Commvault, EMC Networker, HPE Data Protector, and Veritas Netbackup. InterSystems does not validate or endorses one product over the other.

Caché Online Backup

For small deployments the built-in Caché Online Backup facility is also a viable option as well. This InterSystems database online backup utility backs up data in database files by capturing all blocks in the databases then writes the output to a sequential file. This proprietary backup mechanism is designed to cause no downtime to users of the production system.

In Azure, after the online backup has finished, the backup output file and all other files in use by the system must be copied to an Azure File share. This process needs to be scripted and executed within the virtual machine.

The Azure File shares should use an Azure RA-GRS storage account for maximum availability. Note Azure File shares have a maximum share size of 5TB, a maximum file size of 1TB, and maximum 60 MB/s throughput per share (shared by all clients).

Online backup is the entry-level approach for smaller sites wishing to implement a low cost solution for backup. However, as databases increase in size, external backups with snapshot technology are recommended as a best practice with advantages including the backup of external files, faster restore times, and an enterprise-wide view of data and management tools.

Disaster Recovery

When deploying a Caché based application on Azure, Disaster Recovery (DR) resources including network, servers and storage are recommended to be in different Azure region. The amount of capacity required in the designated DR Azure region depends on your organizational needs. In most cases 100% of the production capacity is required when operating in a DR mode, however lesser capacity can be provisioned until more is needed as an elastic model.

Asynchronous database mirroring is used to continuously replicate to the DR Azure region ' s virtual machines. Mirroring uses database transaction journals to replicate updates over a TCP/IP network in a way that has minimal performance impact on the primary system. Compression and encryption is highly recommended to be configured with these DR Asynchronous mirror members.

All external clients on the general Internet who wish to access the application will be routed through an Azure Traffic Manager as a DNS service. Microsoft Azure Traffic Manager (ATM) is used as a switch to direct traffic to the current active data center. Azure Traffic Manager supports a number of algorithms to determine how end users are routed to the various service endpoints. Details of various algorithms can be found here.

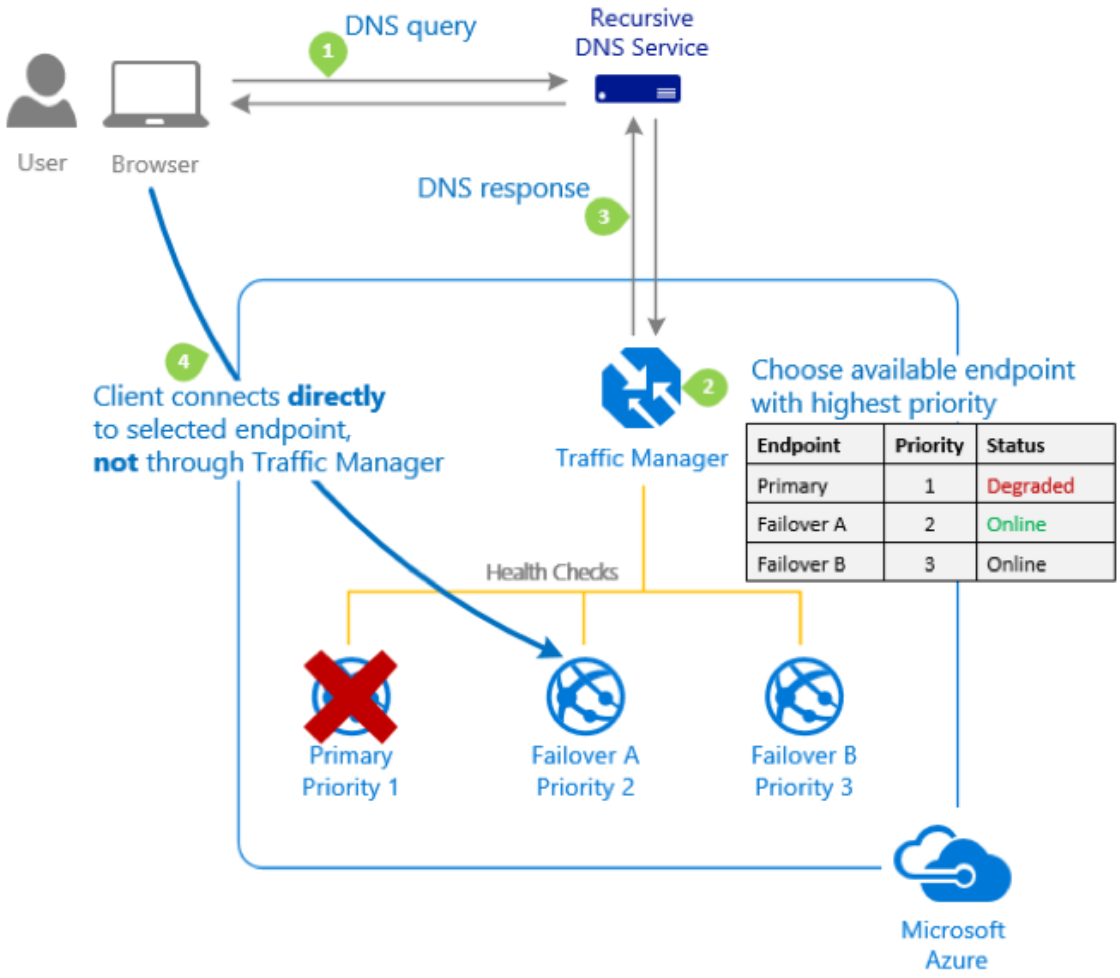
For the purpose of this document, the ' priority ' traffic-routing method will be used in conjunction with Traffic Manager endpoint monitoring and failover. Details of endpoint monitoring and failover can be found here.

Traffic Manager works by making regular requests to each endpoint and then verifying the response. If an endpoint fails to provide a valid response, Traffic Manager shows its status as Degraded. It is no longer included in DNS responses, which instead will return an alternative, available endpoint. In this way, user traffic is directed away from failing endpoints and toward endpoints that are available.

Using the above methods, only the specific region and specific mirror member will only ever allow traffic to it. This

is controlled by the endpoint definition which is a mirrorstatus page presented from the InterSystems CSP Gateway. Only the primary mirror member will ever report “ success ” as a HTTP 200 from the monitor probing.

The following diagram provided by Microsoft demonstrates at a high-level the priority traffic-routine algorithm.



The Azure Traffic Manager will yield a single endpoint such as: "https://my-app.trafficmanager.net" that all clients can connect to. In addition, an A record could be configured to provide a vanity URL such as "https://www.my-app-domain.com". The Azure Traffic Manager shall be configured with one profile that contains the addresses of both regions ’ end point.

At any given time, only one of the regions will report online based on the endpoint monitoring. This ensures that traffic only flows to one region at a given time. There are no added steps needed for failover between the regions since the endpoint monitoring will detect the application in the primary Azure region is down and the application is now live in the secondary Azure region. This is because the DR Async mirror member being promoted to primary and then allows the CSP Gateway to report HTTP 200 to the Traffic Manager endpoint monitoring.

There are many alternatives to the above described solution, and can be customized based on your organization operational requirements and service level agreements.

Network Connectivity

Depending on your application ’ s connectivity requirements, there are multiple connectivity models using either Internet, IPSEC VPN, or a dedicated link using Azure Express Route are available. The method to choose will depend on the application and user needs. The bandwidth usage for each of the three methods vary, and best to check with your Azure representative or Azure Portal for confirmation of available connectivity options for a given

region.

If you are using Express Route, there are several options including multiple circuits and multi-region access that can be enabled for disaster recovery scenarios. It is important to work with the Express Route provider to understand the high availability and disaster recovery scenarios they support.

Security

Care needs to be taken when deciding to deploy an application in a public cloud provider. Your organization ' s standard security policies, or new ones developed specifically for cloud, should be followed to maintain security compliance of your organization. Cloud deployments have the added risk of data now outside client data centers and physical security control. The use of InterSystems database and journal encryption for data at rest (databases and journals) and data in flight (network communications) with AES and SSL/TLS encryption respectively are highly recommended.

As with all encryption key management, proper procedures need to be documented and followed per your organization ' s policies to ensure data safety and prevent unwanted data access or security breech.

When access is allowed over the Internet, third party firewall devices may be required for extra functionality such as intrusion detection, denial of service protection etc.

Architecture Diagram Examples

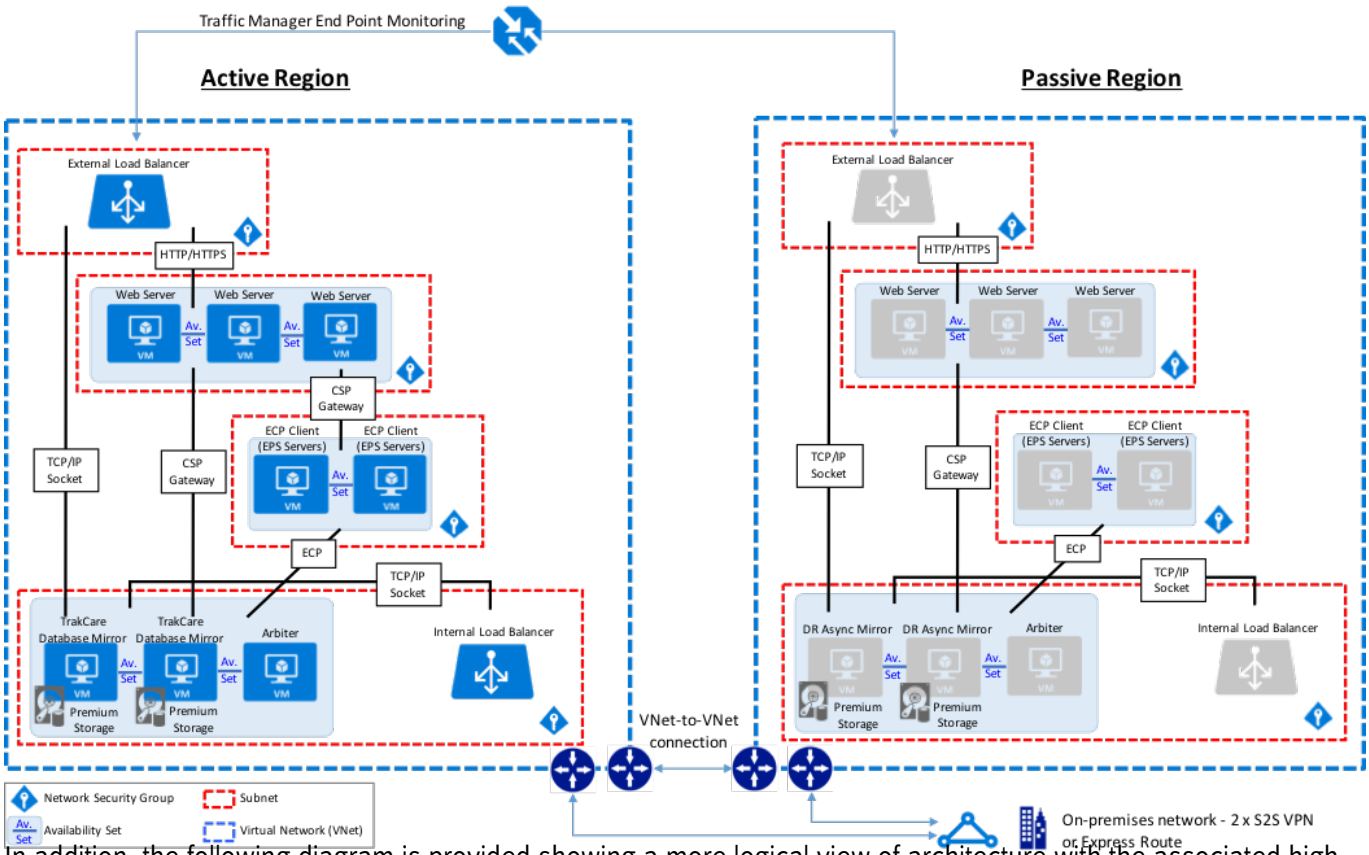
The diagrams below illustrates a typical Caché installation providing high availability in the form of database mirroring (both synchronous failover and DR Asynchronous), application servers using ECP, and multiple load balanced web servers.

TrakCare Example

The following diagram illustrates a typical TrakCare deployment with multiple load balanced webservers, two EPS print servers as ECP clients, and database mirror configuration. The Virtual IP address is only used for connectivity not associated with ECP or the CSP Gateway. The ECP clients and CSP Gateway are mirror-aware and do not require a VIP.

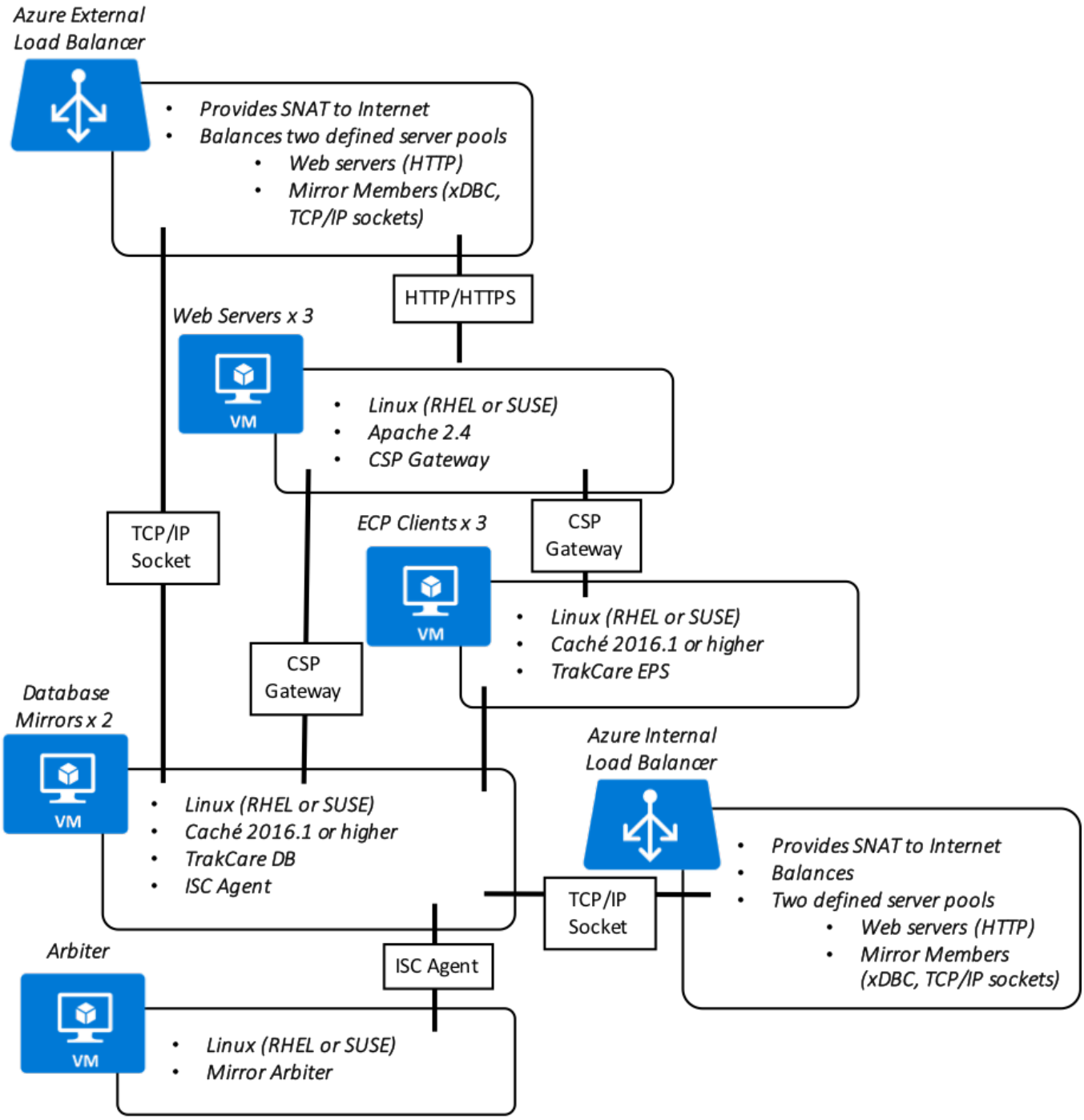
The sample reference architecture diagram below includes high availability in the active or primary region, and disaster recovery to another Azure region if the primary Azure region is unavailable. Also within this example, the database mirrors contain the TrakCare DB, TrakCare Analytics, and Integration namespace all within that single mirror set.

TrakCare Azure Reference Architecture Diagram - PHYSICAL ARCHITECTURE



In addition, the following diagram is provided showing a more logical view of architecture with the associated high-level software products installed and functional purpose.

TrakCare Azure Reference Architecture Diagram - LOGICAL ARCHITECTURE

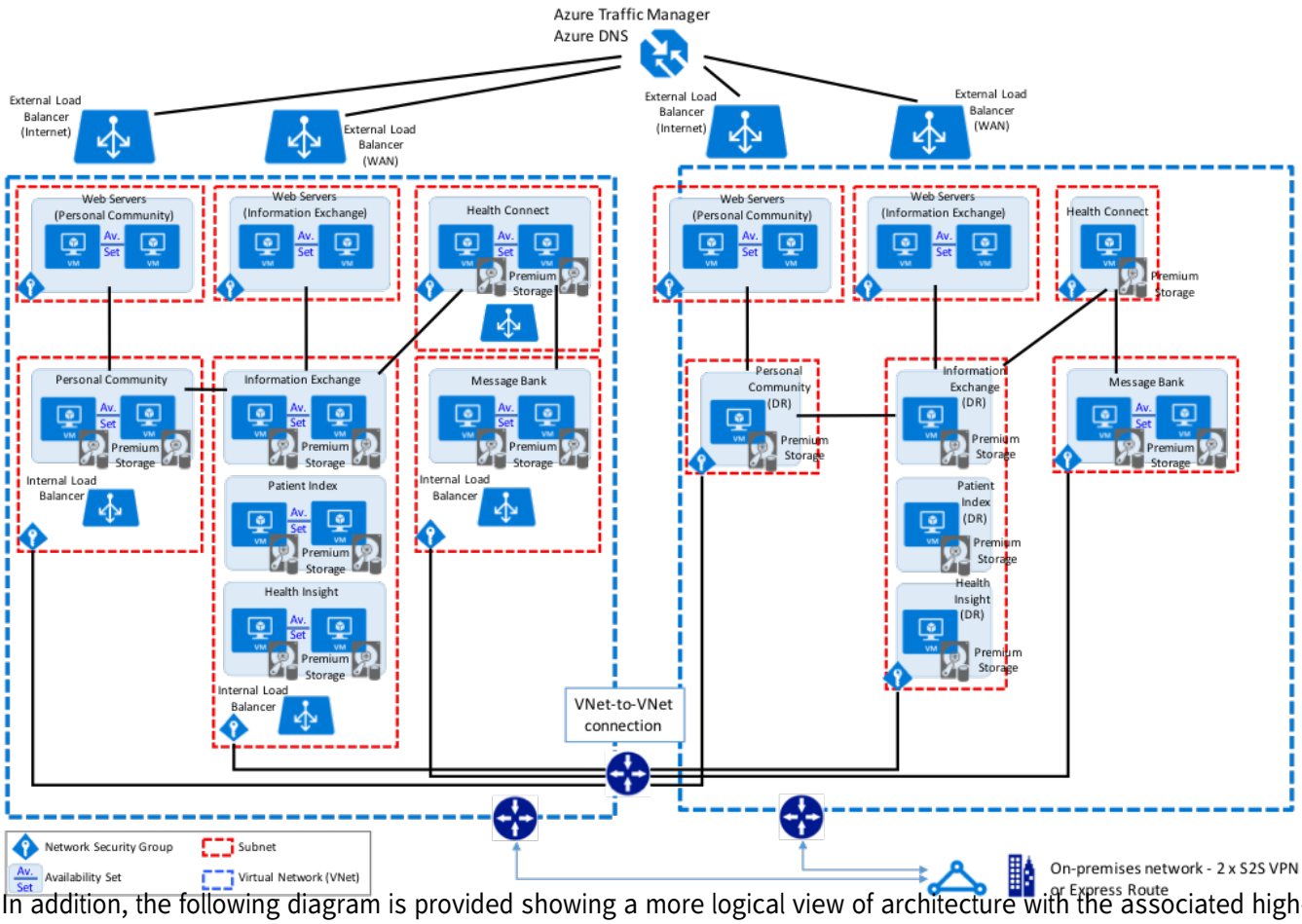


HealthShare Example

The following diagram illustrates a typical HealthShare deployment with multiple load balanced webservers, with multiple HealthShare products including Information Exchange, Patient Index, Personal Community, Health Insight, and Health Connect. Each of those respective products include a database mirror pair for high availability within an Azure availability set. The Virtual IP address is only used for connectivity not associated with ECP or the CSP Gateway. The CSP Gateways used for web service communications between the HealthShare products are mirror-aware and do not require a VIP.

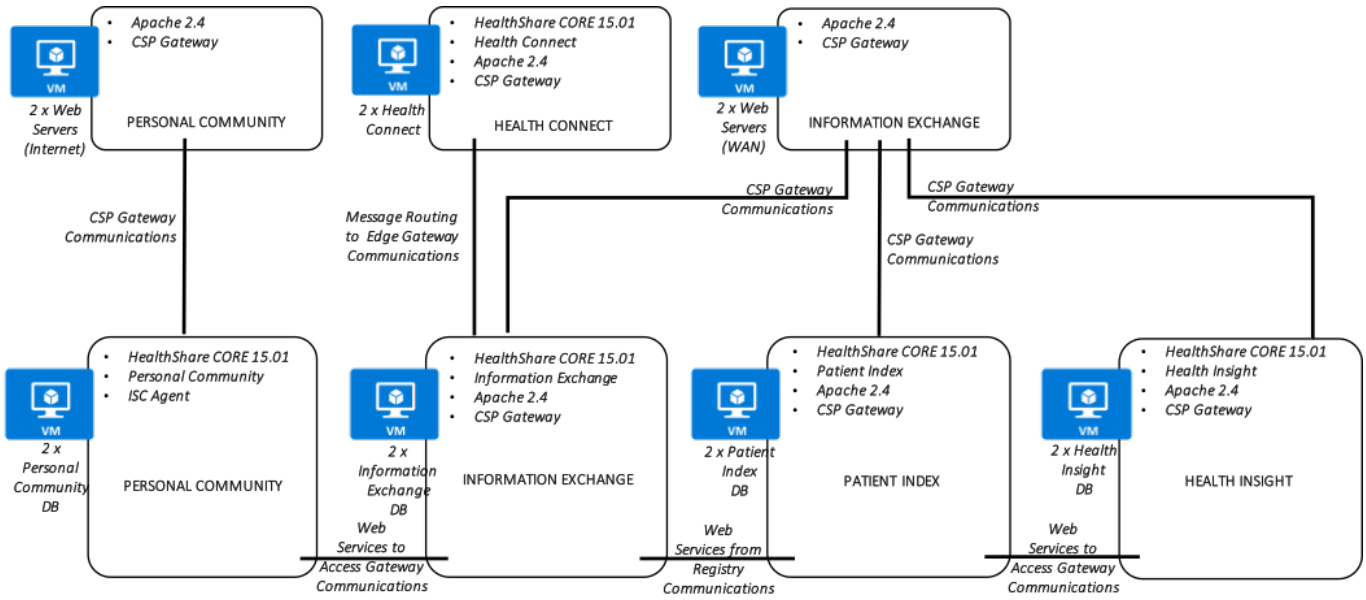
The sample reference architecture diagram below includes high availability in the active or primary region, and disaster recovery to another Azure region if the primary Azure region is unavailable.

HealthShare Azure Reference Architecture Diagram – PHYSICAL ARCHITECTURE



In addition, the following diagram is provided showing a more logical view of architecture with the associated high-level software products installed, connectivity requirements and methods, and the respective functional purpose.

HealthShare Azure Reference Architecture Diagram – LOGICAL ARCHITECTURE



#Azure #Best Practices #Cloud #High Availability #InterSystems Business Solutions and Architectures #Mirroring #Caché #HealthShare #InterSystems IRIS #InterSystems IRIS for Health

Source URL:https://community.intersystems.com/post/intersystems-example-reference-architecture-microsoft-azure-resource-manager-arm