High Availability and Disaster Recovery Strategy

Status: Ready for Review by CSC

Recommendation

Type	Region 1 - AZ1	Region 1 - AZ2	S3	Region 2 - AZ1 (not yet available)
Gold	EC2 + EBS	EC2 + EBS (Cluster or Replication)	Backup	EC2 + EBS (Cluster or Replication)
Silver	EC2 + EBS	EC2 + EBS (Cluster or Replication)	Backup	Backup & Restore
Bronze	EC2 + EBS	Backup & Restore	Backup	Backup & Restore
Note: S3 is at the regional level and also has cross-region synchronization capability. Only 'green' is currently support.				

Change Healthcare will adopt a tiered approach to meeting SLA, RPO, and RTO requirements. Tier descriptions are as follows:

Gold Tier: Utilized for applications with an uptime SLA of 99.9% and with an RPO of less than 1 hour. The RTO will range from 1 hour to a maximum of 4 hours depending on the application needs. The RTO will only affect the level of “activeness” in relation to the passive environment. The default RTO is 2 hours, but the company will allow limited exceptions up to 4 hours.

Silver Tier: Utilized for applications with an uptime SLA of 99.9% and with an RPO of 4 hours. The expected default RTO is 4 hours with a maximum of 8 hours allowed on an exception basis.

Bronze Tier: Utilized for applications with an uptime SLA of 99.7% and with an RPO of 4 to 24 hours. No company applications should have an RPO greater than 24 hours. The expected RTO is 8 hours with a maximum of 24 hours granted for a limited number of exceptions. The default RPO model will be 4 hours with longer RPOs granted in limited cases.

Deployment Approach:

A laboratory DR test is required for any deployment approach before initial GA. The test is required for any major architectural change, such as a deployment architecture change, and must be repeated once a year.

The three-tiered approach also applies to database and application requirements and is described as follows:

Gold Tier: Utilizes multiple AZs and near real-time replication to a zone in another region. The company will ensure multiple AZ deployment and replication to DR regions for stores with these features as part of their definition, such as S3 and Cassandra. HA deployment across regions will be the standard for solutions such as SQL*Server, Oracle, and SoftNAS. Deployment across zones will also be the standard for application VMs. The goal is to bring up the stack in a few minutes at the DR site. Across AZs within the same region, the application VM capacity will scale up to 2X as the SLA approaches 100% (from 99.99%).[1]

Silver Tier: The features will be the same as those in the Gold tier within a region, but replication to another region will utilize database backups and logs. The replication will ensure an RPO within 4 hours.[2]

Bronze Tier: Utilizes a single AZ and has no real-time replication, but uses backup and snapshot capabilities to ensure data is backed up every few hours and pushed to a DR site. This will allow application restoration to the backup and snapshot point.[3]

The East region is our primary region. The company will deploy all the applications there for the coming year. Application split across US regions may be adopted over time, but for now all deployment is in the East region. At the Gold and Silver levels for the application tier, AZ selection for VMs should be across all AZs. For the Gold and Silver DB tier the company will standardize on 2 of the AZs across the board. For the Bronze tier, the company will standardize on 1 of the AZs selected for the Gold and Silver tiers and deploy all Bronze tier applications into that AZ for the coming year.[4]

Applications in the Gold and Silver tier must allow multiple VMs at the application tier. Applications that do not support this level of redundancy will require rearchitecture or will be automatically placed in the the Bronze tier.

DB Store Approach

All DB stores will be dedicated to applications in a production environment. Ideally, the DB stores will be dedicated in non-production environments as well.

1. Oracle – Replication will be based on Data Guard, synchronous for multiple AZs and asynchronous for cross region.
2. SQL*Server – Always On, as the understanding is mirroring is not a long-term strategy for Microsoft.
3. SoftNAS – Similar to SQL*Server.
4. Cassandra – Multiple AZ replication for Gold and Silver tiers, otherwise single AZ.

5. S3 – By definition a shared multiple AZ store within a region with replication to another region all of the time.
6. DynamoDB – Not considered part of standards and teams should move to Cassandra.
7. MySQL – [5]
8. AWS Relational Database Service (RDS) – Not discussed here. These are mechanisms to deploy databases.

Coordinating data synchronization loss between various stores for a given application is not discussed in this document, as that is expected to be in the application domain. However, proper S3 versioning use should allow accurate design of all the necessary features.[6]

Thought Exercise: Mapping applications to the various tiers

1. Legacy Change Healthcare – Silver (for the website and API endpoint)
2. eRX exchange – Silver[7]
3. Advocate Pro – Bronze
4. Assistant Web – Silver
5. CMS – Bronze
6. TC3 – Bronze
7. Payment Automation – Bronze
8. eCashiering/PPOL – Gold
9. Communication layer – Gold – The company must standardize this quickly with close to zero RPO. This will allow for batch situation recovery in many cases without having to go back to our clients.

The problem of an entire AWS region going offline for a period of time, outside of a disaster or severe issue, remains to be solved. The company is not likely to invoke DR procedures if a region is offline due to causes such as network issues. This avoids the possibility of data loss.[8]

Focus the application infrastructure design on uptime requirements. Change Healthcare will design the right deployment architecture within a region and leverage another region for disaster or severe failure recovery.

In general, do not try to meet both high uptime SLA and DR requirements by leveraging multiple regions. However, uptime needs must influence the DR strategy. A limited number of applications can be designed active/active, and in these cases it is acceptable to leverage multiple regions to meet both HA and DR needs.

Basically, the default recommendation is an active/passive model across regions rather than an active/active model. The active/active model is difficult to design in the enterprise application space. The company will deploy active/active applications in a limited number of circumstances where the opportunity arises.

The most important design consideration for primary region redundancy must be uptime SLA requirements. RPO is a secondary consideration.

The level of passiveness of the secondary region must be primarily designed based on RPO. RTO is usually not a concern with the right automation.

Recall that a 99.7% uptime SLA equates to approximately 2 hours of allowed downtime a month. For a 99.9% uptime SLA, allowable downtime is approximately 30-45 minutes. EBS and EC2 instances have a 99.5% uptime SLA with 22 allowable downtime minutes a month. The company plans to deal with possible EBS disk failures with a RAID strategy and S3 has very high durability, therefore the durability problem is effectively solved in a different manner. S3 does pose an availability challenge, and this issue must be incorporated into application design. Application designers must account for the 99.9% of requests in a 5 minute window limitation and NOT place real-time transaction data in S3.

No company applications have a greater uptime requirement than 99.95%. Application recovery time when an underlying component is failing needs to be the focus of this discussion. However, there are no strict network connectivity SLAs that govern an AZ or region simply becoming unavailable at the network tier. This must be a design consideration.[9]

Typical failures include:

- DB node crashes due to load, a DB bug, or issues with OS/DB interference.
- Application VM crashes due to application issues such as load, a bug, or an OS issue.
- Internal networking issues, which can cause challenges with other types of failures.

Policies

- The company will create a three-tiered, multiple AWS AZ and region approach with clear SLA, RPO, and RTO objectives.

Must Haves

Rank	Description	Resolution
1	**Hardened Intranet**: Reliable and secure communication based on least privilege principle between VPCs in AWS and between on premise data centers.	

2	**Hardened Intra-VPC**: Reliable and secure communication based on least privilege principle between stacks within a VPC.	
3	**OS Hygiene**: An automated process to create and maintain hardened OS versions and usage enforcement per policies.	
4	**User/Account Hygiene**: Two-factor authentication to users with least privilege access, maintainable account structure.	
5	**Hardened Internet**: Reliable and secure communication to and from internet based on least privilege principle and meeting security operational requirements.	
6	**DR and HA**: Core infrastructure to enable DR and HA strategy.	• The CIE team will create reference implementations that use multiple availability zones for DR and HA. • The CIE team will automate setting up core infrastructure in the Shared Services account with integral, appropriate HA strategy. • Mission critical, Commvault, and DB backup S3 buckets will be replicated across regions for DR purposes. • Core infrastructure will be defined/deployed in code and allow for a DR strategy as detailed by InfoSec and guided by business approved RPO and RTO. • User access policy and role implementation will be automated and maintained in source control. This makes the environment reproducible and helps achieve HA and DR, if necessary.
7	**Backup and Restore**: Reliable backup and restore infrastructure to create and maintain backups per the policies.	
8	**Compliance**: Guardrails enforcing policies to host protected information in AWS.	

Appendix A - Key Terms and Definitions

RPO – recovery point objective – This defines how many minutes/hours of data loss are acceptable in the case of a disaster or failure of any sort.

RTO – recovery time objective – This defines the acceptable amount of time in which to bring a system back up in the case of a severe disaster or failure. Note that the related RTO does not typically apply to a simple failure such as a machine going down. Those types of failures are covered as part of the service-level agreement (SLA) uptime, response time, or transaction processing details, which tend to be more stringent than RTO allowances in cases of severe failure.

Uptime SLA – The expected uptime for the user-facing, real-time transactional application. This metric is typically measured monthly and an uptime of 99.9% translates to approximately 44 minutes of downtime a month. Most Change Healthcare applications tend to operate in US time

zones and allow for scheduled downtime at night. Typical running hours are from approximately 5 a.m. EST to 9 p.m. PST. Conservatively speaking, these circumstances allow for approximately 30-44 minutes of downtime per month for a 99.9% uptime application. For a 99.95% uptime application, approximately 15-20 minutes of downtime are allowed. For a 99.7% application, approximately 1.5-2 hours per month are allowed.

Batch transaction processing SLA – Batch transaction processing like Claims or ERA usually require processing completion within 24-72 hours, depending on the transaction time. This is not discussed separately, as with the appropriate RTO current applications can meet these requirements because the batch SLAs are not an hour. However, it is important to consider this requirement when designing the RTO for an application.

Transaction processing SLA – This determines the amount of time, typically in seconds, that real-time transactions should take to process.

Response time SLA – The expected time, typically in seconds, for a fully-formed response to return for end users or APIs.

Transaction processing and response time SLAs are not in the scope of this discussion. These concern application design and scalability. These issues have some overlap with HA/DR, but are also separate and distinct.

Customer contracts – Customer contracts are modeled around the SLA and do tend to cover RPO and RTO. However, the SLA applies first and Change Healthcare is penalized for not meeting an SLA. In the case of a disaster, or severe failure categorized as a disaster, the company can refer to the RPO and RTO to determine the amount of time available to recover as well as the consequences of data loss, but the penalty is still defined in the SLA. The RTO and RPO are regarded more as a means to deal with disasters and set expectations in the case of a disaster, not define the terms around point failures.

Appendix B - Background Information

Amazon provides a 10% monthly expense service credit if availability for EC2 instances or EBS volumes is below 99.95%, and a 30% credit if availability is less than 99%. This does not include the upfront amount paid for Reserved Instances (RI), therefore the credit is fairly small. As a result, the SLA is fairly weak. The 99.95% uptime equates to approximately 22 minutes a month and 4 hours, 23 minutes a year of allowable downtime.

For S3, the SLA is based on the number of requests. Availability is modeled on the 99.9% figure, but with a 5 minute window for error responses. This means within a 5 minute window if there are 1,000 requests, one may get an error or service unavailable response with no SLA penalty. The S3 SLA is actually weaker than the EC2 or EBS SLAs as concerns overall availability, but is more stringent in some ways when it comes to system wide downtime as the measurement window is 5 minutes versus monthly for EC2 and EBS.

Note that this doesn't map to durability or data loss. AWS replicates each EBS volume within the Availability Zone (AZ) and claims an annualized failure rate (AFR) of 0.1-0.2%, per volume, per year. To address the EBS volume failure rate Change Healthcare will need redundancy in the form of RAID 1 or some other RAID*xy*. For the S3 service durability is defined as 99.999999999% which practically equates to no data loss.

For EBS volumes, when considering data loss Change Healthcare must consider storage tier durability as well as AZ availability. If RTOs are such that the system must be up faster than the AZ can recover, then the only option is to restore a backup from an earlier point in time unless the data is replicated in another AZ.

For high availability (HA), the challenge centers around the possibility that a running database (DB) EC2 instance may go offline, taking the DB with it. A similar concern for the application tier is the possibility that if the OS virtual machine (VM) goes down, Change Healthcare must consider how to recover the application tier. Currently, not all Change Healthcare applications can run the application tier in a multi-VM model.

With NoSQL stores like Cassandra, Change Healthcare has a choice regarding how to deploy the system across AZs and the number of copies to make.

Similarly, for a software-based network-attached storage (NAS) solution like SoftNAS, the concerns are similar to those of DBs. It is worth noting that a NAS solution will rely on EBS plus S3 for storage and a compute tier (VMs) for access.

Note that S3 is also often used for internal data transfer between the components of an application. S3 is also used as a data store, of sorts, for certain applications. In addition, Change Healthcare will use Glacier for cold storage for some data. The Glacier characteristics must be better understood when it comes to how teams use it for their applications. For example, is Glacier only for cold storage, or will someone ever require it to ensure an application can respond to an API? Current Glacier use cases all follow the cold storage idea and Change Healthcare must firm up this stance.

Change Healthcare applications also use Redshift for analytics/warehouse purposes. If Redshift is needed to meet application API/end-user needs, teams must model Redshift deployment according to their HA/DR form. An example of modeling Redshift deployment is the current use of data shares to WebMD from Redshift as part of the DR in the application code.

A single region with deployment across AZs will not meet Change Healthcare contractual DR requirements for some solutions. This issue is discussed in more detail in the following sections. Change Healthcare must provide guarantees that if a certain region is fully offline, a complete strategy for data and service recovery exists.

Replication of data across EBS volumes across AZs, in the same or different regions, is not something AWS supports out of the box. This kind of replication is left to the application tier to replicate the data as appropriate. AWS does support replicating S3 data across regions in a mostly real-time fashion.[10]

Appendix C - Requirements/Approach Considerations

AZs are similar to independent data centers, although the network fabric within a region acts as if the AZs are one data center. Certain services like S3 are region specific and not AZ specific. In general, region downtime is an uncommon phenomenon for AWS, and historically has occurred with a frequency of approximately once every 1-2 years. The downtime is usually due to networking issues, either specific to AWS or due to a network backbone challenge faced by a large carrier. Therefore, one class of issues Change Healthcare must consider is when and what kind of infrastructure can go down and how the company plans to recover. Problems of this class generally belong in the HA category.

However, these AZs are within tens of miles of each other. The company must also have a natural disaster recovery strategy. AWS has historically recovered from a system-wide infrastructure outage that takes down an entire region, such as a network fabric issue or S3 outage, in less than 24 hours. The company must decide if this type of issue is an HA or DR consideration. There exists a sliding scale between HA and DR with no clear dividing line unless defined by the company with specific circumstances and purposes in mind. In an ideal world, the HA application approach also meets the application DR needs, though most applications in the enterprise space struggle to achieve this equivalence. The company cannot assume parity between HA and DR solutions when determining a comprehensive HA/DR approach for the Change Healthcare set of applications.

Other cost considerations:

DB licenses and storage end up being the most significant costs in deploying enterprise applications. The more copies the company creates, the higher the costs incurred.

Oracle charges for a DB license for any DB instance. SQL*Server provides only one free passive instance and DB storage is still an issue for SQL*Server.

In our current on-premise DC, in general there exists an N+1 model for the node where the databases run with a shared storage feature. If a system goes down, staff can bring up the database on the other node. In some instances this incurs a period of downtime while other configurations recover automatically. The company also replicates the data to a DR site for most applications either in real-time or using backups. An N+1 model is not feasible for AWS because EBS volumes cannot be shared across EC2 instances.

Customer SLAs tend not to distinguish between a natural disaster or an infrastructure failure. Therefore, contractual RTO and RPO overlap with both HA and DR issues. However, customers tend to be more forgiving when it comes to a natural disaster. Even though SLA penalties are not necessarily technically differentiated, meeting an SLA outside of a natural disaster is something that should receive a higher priority when considering a cost-benefit analysis. The belief is that if the AWS East region, for example, experiences a regional outage due to an AWS issue, customers would be approximately as forgiving of downtime as for a natural disaster. The company will receive some sympathy, but without a prompt failover, if necessary, to another region the company will most likely face serious questions and suffer reputation hits. Any other type of failure that results in application downtime will likely elicit little sympathy from customers.

Currently, the interdependencies between applications are not clearly delineated. Many applications run in a hybrid model between on-premise resources and AWS, and many will continue to do so. Therefore, when modeling HA or DR, it is critical to draw clear application boundaries to determine a complete and accurate failover model. Otherwise, it will be nearly impossible to react with a failover plan during a real regional outage, even with data and systems theoretically replicated across the regions.

AZs experience downtime more often than regions, so the HA/DR plan must assume that AZ downtime is not a rare occurrence.[11]

Appendix D - Information to Consider, Action Items

1 - Historically, for a given account, there has been throttling by AWS with regards to how quickly VMs can be spun up. Validate with AWS the best performance they will allow. This will also be impacted by our account strategy.

2 - Sree – We believe we can replicate backups and logs at the right rate to stay within 4 hrs with S3 snapshot replication, but we need to confirm. Also, we need to model out recovery time on the other side. Based on our initial modeling, we believe this is workable unless the database is huge or the change rate of the application is extremely high. This is another validation item. Note that this is of concern as we are trying to avoid having a DB license in the other region, and restoring from a backup from S3 takes time for large DBs.

3 - This requires us to automate bringing up a DB node if the node goes down by mounting the EBS volumes for that node quickly on another node in a different AZ if needed for an in-region failure.

4 - We may have to randomize the AZ picked for Bronze tier to reduce the impact in the case of an AZ failure.

5 - Need info from Sree – I forget.

6 - May need to define architectural patterns.

7 - Do we need Gold? Main difference is RPO.

8 - Need to discuss if we can manage this from a customer expectation perspective. The SLA hit I am not worried about – it's more a reputation hit we need to get our head around.

9 - Need to get to the bottom of AZ failure use cases.

10 - What is the expected lag time for this replication?

11 - Get some real data on the kind of AZ failures we have encountered versus region failures. Note that we are not discussing data durability, but rather the availability of the needed infrastructure in an AZ for the application residing in it to function.