Business Resiliency Framework for Enterprise Workloads in the Cloud

Valentina Salapura, Ruchi Mahindru and Richard Harper
IBM T.J. Watson Research Center, Yorktown Heights, NY, U.S.A

Keywords: Cloud Computing, High Availability, Enterprise Class Applications, Resiliency.

Abstract: Businesses with enterprise-level workloads - such as Systems Applications and Products (SAP) workloads - require business level resiliency including high availability, clustering, or physical server appliances. To enable businesses to use enterprise workloads in a cloud, the IBM Cloud Managed Services (CMS) cloud offers many SAP enterprise-level workloads for both virtualized and non-virtualized cloud environments. Based on our experience with enabling resiliency for enterprise-level workloads like SAP and Oracle, we realize that as the end-to-end process is quite cumbersome, complex and expensive. Therefore, it would be highly beneficial for the customers and the cloud providers to have a systematic business resiliency framework in place, which would very well fit the cloud model with appropriate level of abstraction, automation, while allowing the desired cost benefits. In this paper, we introduce an end-to-end business resiliency framework and resiliency life cycle. We further introduce an algorithm to determine the optimal resiliency pattern for enterprise applications using a diverse set of platforms in the IBM CMS cloud offering.

1 INTRODUCTION

Cloud computing is being rapidly adopted across the IT industry to reduce the total cost of ownership of increasingly more demanding workloads. It is becoming the new de facto environment for many system deployments in a quest for more agile on-demand computing with lower total cost of ownership. Medium and large enterprises, various agencies and institutions are quickly adopting cloud computing, with high expectations of resiliency that have heretofore been associated with the traditional dedicated datacenters.

Enterprises demand usage of Enterprise Resource Planning (ERP) (Hossain, 2001) workloads - such as Systems Applications and Products (SAP) workloads (Gargeya, 2005). The ERP workloads are used to manage business operations and customer relations that are commonly required for running business back-office operations. Such workloads are legacy applications which require an infrastructure with high availability, clustering, shared storage, or physical server appliances. Clustering enables redundancy, which in turns provides resiliency. Setting such resiliency features based on legacy processes is quite cumbersome, as it involves multiple teams performing different actions leading to expensive setup and steady state operations.

Business impact of loss of IT infrastructure can be huge. Enterprise-class clients, such as banks, financial institutions, hospitals, governments, utility companies, etc. can suffer business losses even from short outages and service interrupts. Cost of downtime could dissolve business, or cause irreparable brand damage, loss of customer data and reputation. To deliver the level of resiliency needed by various enterprise applications, a systematic way and a framework for delivering resilient systems is needed.

To satisfy a growing need of enterprise customers to run their enterprise-level workloads in cloud environment, IBM Cloud Managed Services (CMS) (IBM Corporation, 2017; Kochut, 2011) enables enterprise workloads. IBM CMS is a premier cloud offering with both shared and dedicated customer set up, with many resiliency features built it at the infrastructure and hypervisor level (Salapura, 2013). CMS provides a unique mix of virtualized and non-virtualized infrastructure, diverse types of platforms e.g. System x and Power systems and service level agreement (SLA) mechanisms. IBM CMS cloud offers a fully managed solution for many SAP applications in the cloud.

Salapura, V., Mahindru, R. and Harper, R.
Business Resiliency Framework for Enterprise Workloads in the Cloud.
DOI: 10.5220/0006376707140721
In *Proceedings of the 7th International Conference on Cloud Computing and Services Science (CLOSER 2017)*, pages 686-693
ISBN: 978-989-758-243-1

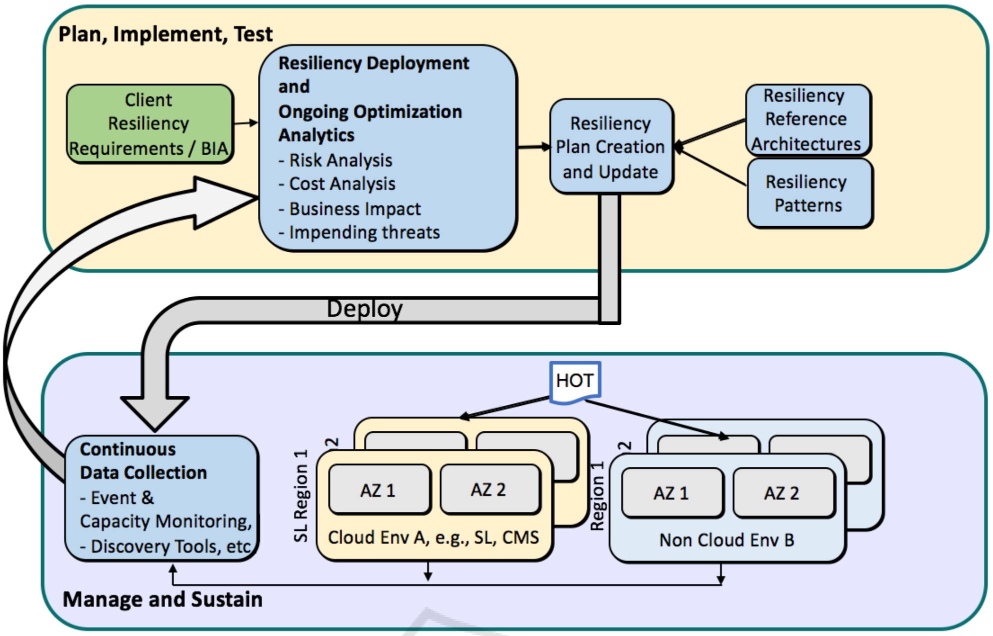

Plan, Implement, Test
Client Resiliency Requirements / BIA
Resiliency Deployment and Ongoing Optimization Analytics
- Risk Analysis
- Cost Analysis
- Business Impact
- Impending threats
Resiliency Plan Creation and Update
Resiliency Reference Architectures
Resiliency Patterns
Deploy
HOT
Continuous Data Collection
- Event & Capacity Monitoring,
- Discovery Tools, etc
SL Region 1
2
AZ 1
AZ 2
Cloud Env A, e.g., SL, CMS
Region 1
2
AZ 1
AZ 2
Non Cloud Env B
Manage and Sustain

Figure 1: Business resiliency framework for cloud.

In this paper, we introduce an end-to-end business resiliency framework we developed in the scope of IBM CMS cloud for a sample set of resiliency solutions. We show how various resiliency patterns can be implemented for enterprise applications for various supported platforms. An example of these applications is SAP High-performance Analytic Appliance (HANA) (Färber, 2012).

2 CLOUD BUSINESS RESILIENCY FRAMEWORK

Cloud computing is highly desirable for its main attributes like scalability, multi-tenancy, on-demand computing resources delivered over the network, and pay-per-use pricing. This offers flexibility in using as few or as many IT resources as needed at any point in time. Thus, the users do not need to predict resources that they might need in future, which makes cloud infrastructure attractive for businesses.

To ensure resiliency of workloads, a number of resiliency features are implemented. These features typically include VM restart upon failure or VM migration, and high availability clusters (HA clusters), where multiple OS images are used to enable continuous operation of enterprise applications. Implementing HA clusters requires several resiliency features such as VM anti-collocation, where VMs are placed on different physical hosts, or shared storage, so that multiple VMs might need to access the same DB data. It also avoids distributed solutions that are hard to manage between the cloud and non-cloud environments with a part of the workload running in the cloud, and the other part running in a non-cloud environment.

Given that deployment of such resiliency features can be complex, it warrants a need for a structured and ongoing approach to plan, maintain, test and continuously improve such business resiliency operations. To address this need, we introduce an end-to-end business resiliency framework and the lifecycle we developed in the scope of IBM CMS cloud.

Each enterprise customer has different workloads requirements and SLAs. Cloud is a multi-tenant environment with the goal to standardize the solutions and phases within them as much as possible to simplify the process associated with deployment and steady state operations to promote the asset reuse while maintaining the low cost. Such objective motivates the need for an end to end business resiliency framework, as described below.

An end to end business resiliency framework allows cloud provider and their customers to define a comprehensive resiliency plan in the cloud environment for both cloud native and cloud enabled workloads. The resiliency framework enables to systematically assess and evaluate customer workloads to identify resiliency requirements as determined by business impact analysis.

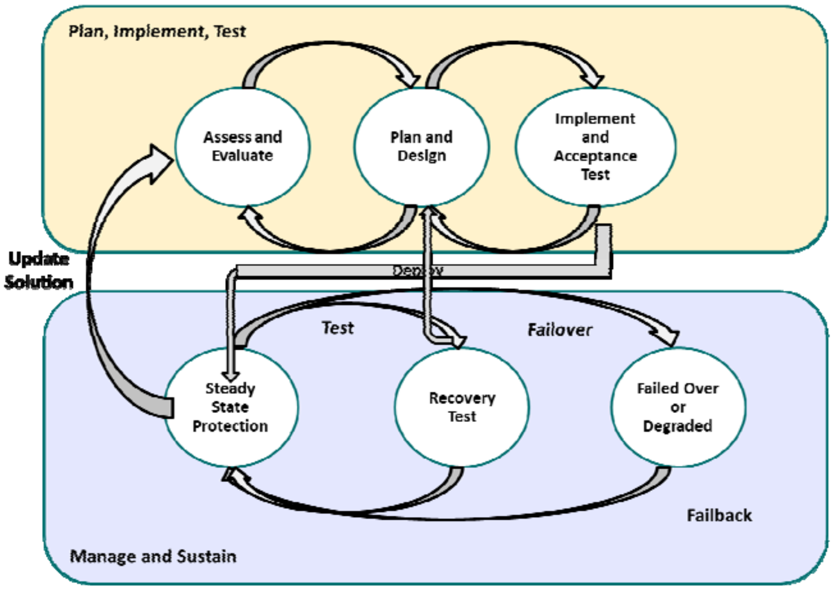


Figure 2: Business resiliency lifecycle.

Because of the business risk analysis and resiliency requirements, and referencing the resiliency reference architecture, an appropriate resiliency plan is created which uses the selected cloud resiliency patterns. For the cloud enabled workloads, the resiliency plan selects resiliency components, and gives configuration of the appropriate resilience elements, such as HA clusters or data replication. For new applications for which there are no resiliency patterns available, guidelines are provided to assist designing resilient applications from scratch via patterns, reference architectures, and wizards.

As illustrated in Fig. 1, the developed resiliency plan is deployed across the cloud and non-cloud environments available to the client. The ongoing operation of the customer's resiliency mechanisms is instrumented, and collected data is analysed to ensure that the required resiliency and SLA levels are being met. In addition, the framework provides recommendations on how to improve the resiliency posture and/or reduce the cost of resiliency while maintaining SLAs.

3 CLOUD BUSINESS RESILIENCY LIFECYCLE

The resiliency framework is used for both initial resiliency deployment, and for ongoing resiliency optimization. An important component in the framework is continuous monitoring of the deployed workload, the environment, risks, costs, and other parameters. Based on the variations in the workload, risk updates, impending events and disasters, cost variations (e.g., cost of datacenter, cost of replication network, datacenter saturation), or variation in client workload importance over time, the resiliency plans are revised and updated.

Any resiliency solution, whether for a high availability or disaster recovery, undergoes a life cycle, as shown in Fig. 2. Due to space limitations, only a summary of the key phases is presented here.

The two major phases of the life cycle are "Plan, Implement, and Test," and "Manage and Sustain." In the former phase the requirements are "Assessed and Evaluated", and the resiliency solution is "Planned and Designed," leveraging the business resiliency framework described earlier. At the end of this phase, the resiliency solution is "Implemented, Tested, and Deployed" into the production environment and enters service.

While in service (also called steady state), resiliency functionality is leveraged to "Protect" the workload from the anticipated failures. All resiliency solutions must periodically undergo "Recovery Test" to ensure that the resiliency mechanisms are functional. Such tests often reveal weaknesses in the resiliency solution which in turn requires a continuous revalidation of the "Plan, Implement, and Test" life cycle phase to update the weak elements of the solution.

In addition, while in service the workload may suffer failures. The resiliency features will engage and the workload will enter the "Failed Over or Degraded" state. The exact configuration of this state of course depends on the resiliency solution in effect. If the failure did not result in any physical destruction of the originating environment, then a “Non-Reconstructive Failback” to that environment is performed when that environment has been repaired. However, if the originating environment has been irremediably damaged, then a “Reconstructive Failback” process is performed. This equates strongly to re-entering the “Implement and Acceptance Test” state.

Next sections demonstrate an end-to-end scenario with the application of resiliency framework, along with various phases and states of the lifecycle.

4 WORKLOADS AND RESILIENCY PATTERNS CHARACTERIZATION

Enterprise applications can be deployed in several different ways, depending on the features needed, performance requirements, or if high availability support is needed. Each of these different configurations provide a different level of resiliency inherent to that configuration. For example, SAP HANA can be deployed in a single node, or multiple nodes configuration. As a single node deployment, it can be scaled up to include resources and provide high availability for data. As a multi node scale our configuration, it can be configured to support high availability clusters, or not. Each of these configurations achieves different levels of resiliency, satisfying different SLA requirements. Also, each of these configurations has a different cost base.

Since for the cloud environment we want to provide economy of scale, we want to provide the required level of resiliency while minimizing cost. Resiliency is increased in highly automated environments, thus eliminating human errors, and reducing cost.

To select the optimal configuration which provides required level of resiliency, we introduce an algorithm to be used with our resiliency framework. While the framework has several phases, design and plan, test, steady state, etc., resiliency evaluation and optimization can be performed in each phase. In this paper, we focus on the optimization during the “Plan and design” phase. In the future, we plan to work on optimization for the other framework phases, such as for “Test and validate”, and for “Steady state”.

To determine the optimal resilient architecture for a workload, we use application attributes to qualify each application. The attributes describe applications’ properties in terms of memory consistency, state-full and scaling. The attributes we use are a result of our observation of the workloads deployed, and attributes that must be considered for resiliency deployment. By no means it represents the exhaustive list of workloads’ attributes. The attributes are listed below:

Relaxed Consistency vs. Sequential consistency: Sequential consistency model requires a write by any processor to be seen by all processors in real time, maintaining the overall order of writes between the processors, but which can impact performance. Relaxed consistency requires programmers to implement the memory consistency explicitly by applying synchronization.

Stateless vs. Stateful: A stateless applications do not record data generated in one session for use in the next session. A stateful application must record changes in state caused by events during a session.

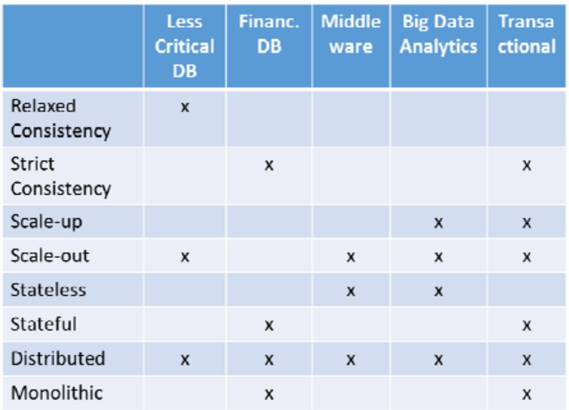
Distributed vs. Monolithic: A monolithic application is a single-tiered application in which the user interface and data access code are within a single program. A multitier application is a client–server architecture in which web interface, application processing, and data management functions are physically separated.

Scale-up vs. Scale-out: Scale up (after referred to as Vertical Scaling) approach adds more resources (processors and memory) to a server, providing a more robust server. Scale out (or Horizontal Scaling) approach adds more servers without increasing individual servers.

To capture characteristics of different workloads, we distinguish a set of different workload groups. We characterize different workload groups for each of the given attributes. For example, we differentiate between less critical database workloads, financial databases, and transactional workloads, to name a few. Other workload groups can be characterized following our nomenclature. For example, a less critical database workload can use a relaxed consistency, and is implemented as a distributed system that can grow by adding more servers. A financial database is stricter, and it must preserve the exact order of transactions thus demanding sequential consistency.

Cloud providers offer different level of service level agreement (SLA) to describe level of availability. SLAs are contractual obligations and in many cases, include penalties for noncompliance.

Table 1: Application characterization based on their attributes.

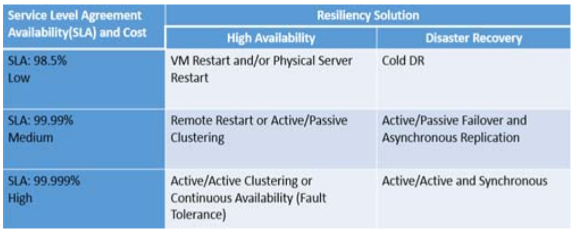


	Less Critical DB	Financ. DB	Middleware	Big Data Analytics	Transactional
Relaxed Consistency	x				
Strict Consistency		x			x
Scale-up				x	x
Scale-out	x		x	x	x
Stateless			x	x	
Stateful		x			x
Distributed	x	x	x	x	x
Monolithic		x			x

Typically offered SLA levels are: 99.999%, 99.99%, 99.9%, 98.5%, which describe different allowed down time. This translates in tolerated maximum downtime from 26.3 seconds per month for the highest SLA level, to 14.4 hours of downtime per month for servers with the lowest SLA level (Schmidt, 2006).

Different resiliency patterns achieve different level of availability. We distinguish between high availability (HA) solutions and disaster recovery (DR) solutions for each SLA level. For example, to achieve SLA of 98.5% the use virtual or physical server restart mechanisms is sufficient. To achieve the highest SLA level, more sophisticated methods must be used such as high availability clustering with servers configured in active-active configuration. We list some existing resiliency patterns for HA and DR for achieving different level of availability in Table 2.

Table 2: Service level agreement levels, cost and resiliency solution.



Service Level Agreement Availability(SLA) and Cost	Resiliency Solution	
	High Availability	Disaster Recovery
SLA: 98.5% Low	VM Restart and/or Physical Server Restart	Cold DR
SLA: 99.99% Medium	Remote Restart or Active/Passive Clustering	Active/Passive Failover and Asynchronous Replication
SLA: 99.999% High	Active/Active Clustering or Continuous Availability (Fault Tolerance)	Active/Active and Synchronous

Each of the resiliency pattern is associated with a cost base to implement it. Thus, for a higher SLA level, more resources must be used, which results in a higher cost solution. For example, using a cluster of servers to implement high availability cluster offers a higher availability solution, but it also costs more than restarting a single server, as in a lower level availability solution.

5 RESILIENCY PATTERN OPTIMIZATION

For our resiliency pattern optimization algorithm, we quantify different resiliency patterns we can use as a solution architecture to ensure high availability to workloads.

Each resiliency solution has a range of availability numbers, cost and recovery time associated to it. The *cost* of any solution has multiple contributing components such as *cost*{*overhead, operational cost, deployment cost, maintenance cost, resource cost*}. Recovery time is defined as a range of minutes it takes to recover, which could be a *range of minutes to recover*.

For example, active replication pattern ensures advance high availability but comes at high operational cost, whereas virtual machine restart provides moderate availability at a low operational cost. However, operation disruption may not be acceptable for the mission critical workloads. The attributes associated with resiliency patterns are captured by system matter experts.

When submitting a request for a business resiliency solution a user may specify the attributes application attributes based on the system's guidance or select all the standard attributes for a given application listed in the best practices catalog by the service provider. We list only a subset of possible attributes and their mapping to applications. The mapping is continuously evolving for new applications and identified attributes. These attributes may be reprioritized over time, and revised as learnt through the system to eliminate correlated attributes.

For a given SLA, our algorithm selects the optimal resiliency pattern that matches the given application attributes and the availability while minimizing the total cost. The combination of the attributes of a workload and the desired SLA level drives the cost of the appropriate resiliency solution.

The algorithm performs the following steps:

- For a given workload, enumerate the attributes of the workload
- Select the required SLA
- For the given SLA and attributes, select possible resiliency patterns.
- From possible resiliency patterns, select lowest cost pattern for which the desired SLA is met.

- Add to library of resiliency pattern solutions for that application and given SLA.

This algorithm effectively maps the user provided input workload attributes to the attributes captured for each of the resiliency solutions. Every new determined resiliency pattern is added to the library of statically defined pattern-workload mapping, which contains pre-matched set of solutions for combination of attributes selected.

Each resiliency solution has an embedded availability model *{number of nodes, heartbeat, type of box, type of storage}* that can be adjusted at any stage of the process.

6 CASE STUDY: BUSINESS RESILIENCY FRAMEWORK FOR HANA

SAP HANA appliance can be deployed on a single node server (without high availability), scale-up with high availability, scale-out without high availability, or a scale-out multi node cluster to provide high availability. Due to complex deployment and high cost associated with deploying SAP HANA solution, it is generally recommended to first scale-up the solution as much as possible (i.e., to add more resources to the server) before considering the option to scale-out (to distribute the application on multiple servers). Scale-out is primarily available for analytics workloads like BW on HANA or DataMart scenarios. Scale-up is generally available for the transactional workloads like SAP Business Suite on HANA including ERP, CRM, SRM, SCM, etc.

Figure 3 shows a scenario where customer initially requires to host a small-to-medium sized critical BW analytics application to provide real-time feeds to its sample users. In the "Plan, Implement, Test" phase, first the requirements are assessed and evaluated. Based on the assessment and evaluation in the "Assess and Evaluate" state, a scale up solution with high availability scenario is planned and designed. The solution is planned, designed, implemented, tested, and deployed with an active-passive configuration.

During the "Manage and Sustain" phase, the solution is maintained in steady state, where it is monitored for performance and capacity constraints. The high-availability set-up is tested on some periodic basis to pro-actively validate and fine-tune the setup, in case of an actual failure. In case of an actual failure of the primary node, the workload is failed-over to the standby node. The deployed solution is scaled-up to its maximum capability, based on the event and capacity monitoring and recovery test functionality.

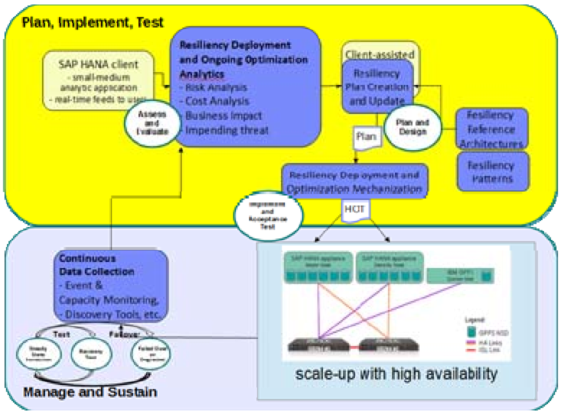

Plan, Implement, Test
SAP HANA client
- small-medium analytic application
- real-time feeds to users
Assess and Evaluate
Resiliency Deployment and Ongoing Optimization Analytics
- Risk Analysis
- Cost Analysis
Business Impact
Impending threat
Client-assisted
Resiliency Plan Creation and Update
Plan
Plan and Design
Resiliency Reference Architectures
Resiliency Patterns
Resiliency Deployment and Optimization Mechanization
HOT
Implement and Acceptance Test
Continuous Data Collection
- Event & Capacity Monitoring,
- Discovery Tools, etc.
Test
Failover
Legend
scale-up with high availability
Manage and Sustain

Figure 3: Use of elements of the resiliency framework across the resiliency life cycle.

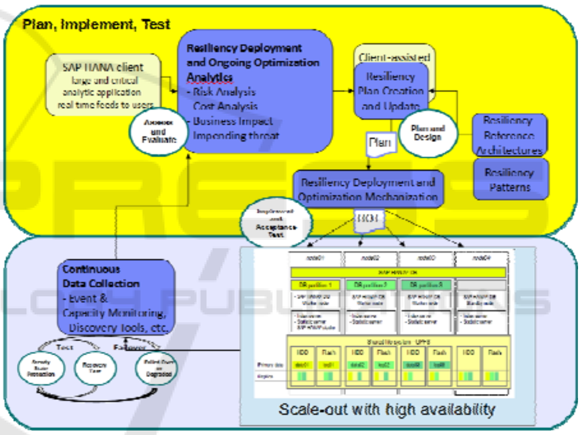

Plan, Implement, Test
SAP HANA client
large and critical analytic application
real time feeds to users
Assess and Evaluate
Resiliency Deployment and Ongoing Optimization Analytics
- Risk Analysis
Cost Analysis
- Business Impact
Impending threat
Client-assisted
Resiliency Plan Creation and Update
Plan
Plan and Design
Resiliency Reference Architectures
Resiliency Patterns
Resiliency Deployment and Optimization Mechanization
HOT
Implement and Acceptance Test
Continuous Data Collection
- Event & Capacity Monitoring,
Discovery Tools, etc.
Test
Failover
node01
node02
node03
node04
HDD
Flash
Scale-out with high availability

Figure 4: Use of elements of the resiliency framework across the resiliency life cycle.

Overtime, based on the performance data collected during the "Manage and Sustain" phase along with further assessment and evaluation of customer's growing needs to host a large analytics application with larger number of real users, a scale-out solution is selected. It offers high availability to provide increased benefit with the corresponding increased investment. The new solution is planned, designed, implemented and tested with the right sized business resiliency solution to cater the customer requirements, as shown in Figure 4.

7 RELATED WORK

Enterprise-class customers (e.g., banks, insurances and airlines) need management services such as monitoring, patching, backup, change control, high availability and disaster recovery to support systems running complex applications with stringent IT process control and quality-of-service (QoS) requirements. Such features are typically offered by IT service providers in strategic outsourcing (SO) engagements, a business model for which the provider takes over several aspects of management of a customer's datacenter resources, software assets, and processes. Servers with such support are characterized as being managed.

This should be contrasted with unmanaged servers provisioned using basic Amazon Web Services (AWS) (Miller, 2010; AWS Corporation, 2017) and IBM's SoftLayer (SoftLayer, 2017) offerings, where the cloud provider offers automated server provisioning. To make a server managed, these cloud service providers have networked with other service partners that customers can engage to fill all the gaps up and down the stack. This enables the user to add services to the provisioned server, but the cloud provider assumes no responsibility for their upkeep or the additional services added. Therefore, it puts burden on the customer to obtain a fully managed solution for their enterprise workload rather than the cloud service providing an end-to-end fully managed solution for the customers.

AWS provides the IT resources so that the customers can launch entire SAP enterprise software stacks on the AWS Cloud. AWS Cloud is SAP verified and certified. AWS provides highly reliable services and multiple fault-tolerant Availability Zones for disaster recovery implementations.

The IBM Cloud Managed Services (CMS) product (IBM Corporation, 2017) from IBM is an enterprise cloud which provides managed services for critical workloads and enterprise-level SLA mechanisms. CMS supports several software services on CMS, such CMS4SAP CMS4ORCALE and AMM4SAP.

HANA is fully certified to run on VMware platform (King, 2014). vSphere 5.5 has a limitation in that the largest VM can be created with 1 TB of disk storage only. Depending on the usage of the data, both warm and cold data can reside together on the disk. This enables extension of the total size of the SAP HANA database above 1 TB. Currently, several cloud providers that are enabling themselves to support more options for SAP and SAP HANA workloads.

In (Dekel, 2003), the authors have described a system that focuses on performance aware high availability which is achieved through cloning and replication of application's state. Our work focuses on a resiliency framework to determine and deploy the optimal resiliency support for a given workload based on its characteristics.

8 LESSONS LEARNT AND CONCLUSIONS

During enablement of enterprise workloads in the IBM's CMS cloud, several points became apparent. First insight is that each enterprise customer has a varied set of resiliency requirements for the workload that they are running depending on the nature of their business. Therefore, the cloud service providers must handle such heterogeneous requirements with least amount of customization possible that must be delivered in a tight scheduled while maintaining the low cost.

Second insight is that there is a variety of cluster set up configurations that may be possible and the required set up may vary from workload to workload. Additionally, the cluster set up may evolve overtime based on the changing requirements of the workload. Additionally, the cloud provider must support the application level replication technology depending on the applications being deployed. As the requirements are highly variable and may evolve overtime as the workload evolves, it is crucial to systematize and standardize the end to end process of the resiliency solution planning, implementation, testing and delivery.

Another insight is that multiple levels of resiliency at infrastructure, middleware and application levels are required for increased system reliability. Implementing multiple levels of resiliency delivers a more robust system, while enabling operation of these different levels of resiliency seamlessly.

Enterprise-class customers, such as banks, financial institutions, hospitals, governments, utility companies, etc. can suffer high business losses even from short outages and service interrupts in the IT infrastructure. Cost of downtime could dissolve business, or cause irreparable brand damage, loss of customer data and reputation. A structured and continuously improving mechanism is required to deliver the level of resiliency needed by the various enterprise applications.

We introduced an end-to-end business resiliency framework and resiliency life cycle. We further discussed various resiliency patterns implemented for enterprise applications using a diverse set of platforms in the IBM CMS cloud offering. To determine the optimal resiliency pattern for various applications, we introduce an optimization algorithm which takes into consideration application attributes and the desired SLA level, to determine the optimal resiliency pattern. We showcased an end to end application of the resiliency framework and resiliency life cycle for a SAP HANA scenario.

REFERENCES

L. Hossain, J. D. Patrick, and M. A. Rashid, Enterprise Resource Planning: Global Opportunities and Challenges. Hershey Park, PA: Idea Group Publishing. 2001.

Gargeya, VB 2005, ‘Success and failure factors of adopting SAP in ERP system implementation’, Business Process Management Journal, Vol.11, No.5, pp501–516.

IBM Corporation, IBM Cloud Managed Services. [Online]. Available: http://www.ibm.com/marketplace/cloud/managed-cloud/us/en-us. Last Accessed: 2017-03-15.

A. Kochut, Y. Deng, M. R. Head, J. Munson, A. Sailer, H. Shaikh, C. Tang, A. Amies, M. Beaton, D. Geiss, D. Herman, H. Macho, S. Pappe, S. Peddle, R. Rendahl, A. E. T. Reyes, H. Sluiman, B. Snitzer, T. Volin, and H. Wagner, "Evolution of the IBM cloud: Enabling an enterprise cloud services ecosystem," IBM Journal of Research and Development, vol. 55, pp. 397-409, Nov. 2011.

V. Salapura, R. Harper, and M. Viswanathan. "Resilient cloud computing." IBM Journal of Research and Development, vol. 57 no. 5, 2013.

F. Färber, N. May, W. Lehner, P. Große, I. Müller, H. Rauhe, and J. Dees, "The SAP HANA Database--An Architecture Overview," in IEEE Data Eng. Bull. vol. 35, no. 1, pp. 28-33, 2012.

F. Färber, S. K. Cha, J. Primsch, C. Bornhövd, S. Sigg, and W. Lehner, "SAP HANA database: data management for modern business applications," in ACM Sigmod Record, vol. 40, no. 4, pp 45-51, 2012.

K. Schmidt, High Availability and Disaster Recovery: Concepts, Design, Implementation. Springer Science and Business Media, 2006.

F. P. Miller, A. F. Vandome, and J. McBrewster, Amazon Web Services. Alpha Press, 2010.

AWS Corporation, Amazon Elastic File System. [Online]. Available: https://aws.amazon.com/efs/. Last Accessed: 2017-03-15.

SoftLayer. [Online]. Available: http://www.softlayer.com/. Last Acessed: 2017-03-15.

C. King, “Demystifying Production SAP HANA on VMware vSphere Implementations” VMWare White Paper: (2014). [Online]. Available: http://info.vmware.com/content/31421_Whitepaper_Reg?asset=whitepaper&cid=70180000000Nlj1&src=wsite. Last Accessed : 2017-03-15.

E. Dekel, O. Frenkel, G. Goft, Y. Moatti, "Easy: engineering high availability QoS in wServices", Reliable Distributed Systems 2003. Proceedings. 22nd International Symposium on, pp. 157-166, 2003, ISSN 1060-9857.