Contents

Contents ........................................................................................................................................... 2

1.0 Introduction ................................................................................................................................ 3

2.0 Rationale for Cloud Computing ................................................................................................. 3

3.0 Infrastructure as a Service (IaaS) Use Cases ................................................................................. 5
  3.1 Virtual Environment Management ......................................................................................... 6
  3.2 Hybrid Cloud Management .................................................................................................... 7
  3.3 Storage Capacity ..................................................................................................................... 8
  3.4 Application On-Boarding ...................................................................................................... 11

4.0 Platform as a Service (PaaS) Use Cases ..................................................................................... 13
  4.1 Policy-driven Application Management ............................................................................... 15
  4.2 Development & Test – Problem Determination ................................................................... 20
  4.3 Development & Test – Application Promotion .................................................................... 21
  4.4 Database as a Service ........................................................................................................... 23
  4.5 Big Data Analytics ................................................................................................................. 25

5.0 Conclusion ................................................................................................................................ 27
1.0 Introduction

This white paper is a product of the Cloud Standards Customer Council\(^1\) (CSCC) and documents a collection of high priority use cases for both the Infrastructure as a Service (IaaS) and Platform as a Service (PaaS) cloud computing service models.\(^2\) In general, use cases highlight the benefits of cloud computing and are key to identifying specific technical requirements including the need for open standards that promote portability and interoperability.

The use cases included in this document are being leveraged by various CSCC working groups within the to help identify gaps within existing cloud based standards and to develop the supporting requirements which can be brought forward to the appropriate standards development organizations (SDO) for consideration. Current snapshots of the use cases are included in this document. The CSCC working groups will continue to evolve these use cases to the point where detailed standards requirements are documented. New use cases will be developed as appropriate. This white paper is a living document and will be updated on a regular basis to reflect changes to the use cases.

The paper opens with a short introduction on the rationale for cloud computing. This section highlights the benefits of cloud computing and defines the cloud terminology that is used throughout the paper. The remainder of the document is focused on specific use cases for IaaS and PaaS.

2.0 Rationale for Cloud Computing

Cloud computing offers a value proposition that is different than traditional enterprise IT environments. By offering a way to exploit virtualization and aggregate computing resources, cloud computing can offer economies of scale that would otherwise be unavailable. It can also offer opportunities to immediately exploit installed hardware and software, rather than expending time and resources to design, deploy and test a new implementation.

Because virtual instances can be provisioned and terminated at any time and the user organization pays only for the computing resource they are employing, costs can be lower. Likewise, fee structures need to be adequately clarified and understood to estimate future costs.

The benefits of cloud computing are compelling.

- Cloud computing permits the expense of infrastructure and its management to become an operational expense rather than a capital investment. This can be beneficial for the business both from a tax perspective, and also because it allows the organization to conserve capital for other purposes.

---

\(^1\) Refer to http://www.cloudstandardscustomercouncil.org/index.htm for more information on the CSCC.

\(^2\) Software as a Service use case will be included in the next revision of this document.
• Cloud computing offers a centralized, remote facility for computing, leading to economies of scale in both the use of hardware and software and a reduction in required resources for administrative management.

• The ability to make use of computing resources on an immediate basis, rather than a need to first invest time and skilled resources in designing, implementing and testing infrastructure (hardware and middleware). This leads to faster time to value which may mean enhanced revenue, greater business agility, larger market share, or other benefits.

Cloud computing does not exist in a vacuum. Most organizations will have a broad variety of applications already running in their data center. For most, cloud computing will extend their existing infrastructure. It can be used mainly for new projects. Or, an organization may use it for overflow, guaranteeing a certain level of performance for enterprise computing.

**NIST Defines Three Cloud Service Models**

**Cloud Software as a Service (SaaS).** The capability provided to the consumer is to use the provider’s applications running on a cloud infrastructure, typically through a pay-per-use business model. The applications are accessible from various client devices through a thin client interface such as a web browser (e.g., web-based email). The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, storage, or even individual application capabilities, with the possible exception of limited user-specific application configuration settings.

**Cloud Platform as a Service (PaaS).** The capability provided to the consumer is to deploy onto the cloud infrastructure consumer-created or acquired applications created using programming languages and tools supported by the provider. The consumer does not manage or control the underlying cloud infrastructure including network, servers, operating systems, or storage, but has control over the deployed applications and possibly application hosting environment configurations.

**Cloud Infrastructure as a Service (IaaS).** The capability provided to the consumer is to provision processing, storage, networks, and other fundamental computing resources where the consumer is able to deploy and run arbitrary software, which can include operating systems and applications. The consumer does not manage or control the underlying cloud infrastructure but has control over operating systems, storage, deployed applications, and possibly limited control of select networking components (e.g., host firewalls).

**NIST Also Recognizes Four Deployment Models**

• A **private cloud** in which the cloud infrastructure is utilized by just one organization, though not necessarily operated by that one organization.

• A **community cloud** whereby several organizations with common concerns share a cloud.

• The **public cloud** provided by the private sector for all comers, such as Amazon’s EC2 service.
• A hybrid cloud in which two or more cloud types are discrete but networked together such that a burst of activity beyond the capabilities of one cloud is shifted for processing to another.

It should be noted that the deployment models do not reflect where the resources are actually hosted. For example a private cloud can be hosted internally (on-site) or externally (outsourced). The deployment models in conjunction with hosting need to be recognized and considered.

**What is the importance of Standards-Based Cloud Computing?**

Standards-based cloud computing ensures that clouds can readily interoperate based on open standard interfaces. This allows workloads to be readily moved from cloud to cloud and services created for one cloud computing environment to be employed in another cloud computing environment, eliminating the need to write redundant code.

Without open standards, organizations are forced to select proprietary environments that lead to vendor lock-in. This means that integrating applications or services across differing proprietary cloud platforms will be possible but will require extensive, expensive, and time-consuming work.

Some of the proposed standards are based on open source environments. This has the advantage of making all the code transparent, available for inspection, and more readily suited for an interoperable environment. However, whenever a new technology is attracting a great deal of attention, neither vendors nor customers are likely to wait for mature standards or rich open source environments. They will leverage the advantage of early adoption of emerging technology at the price of having to move to a standard (and perhaps an open source) environment at a later date.

**3.0 Infrastructure as a Service (IaaS) Use Cases**

While many enterprises today are using virtualization to consolidate their IT infrastructures, hardware consolidation is only one piece of virtualization’s benefit. Organizations that move beyond virtualization with IaaS capabilities such as integrated service management and self-service provisioning can realize significant benefits:

• Reduce IT operating expenses and capital expenses by improving resource utilization and administrator-to-server ratios

• Reduce time to market through increased efficiency and automation of common solutions across the enterprise helping to eliminate unique solutions at a departmental level

• Achieve simplified, integrated management including real-time monitoring and high-scale low-touch provisioning

• Scale operations to meet market dynamics and business strategy

**Approaches for adoption of IaaS:**

The adoption approach for IaaS varies depending upon the level of investment in capital IT assets, in-house operations management and systems support skills.
## Approach for IaaS adoption

### Large Organizations

1. Analyze IaaS offerings in terms of total cost of ownership (TCO)/return on investment (ROI) and risks such as vendor lock in/interoperability/existing IT infrastructure especially network.
2. Define a clear IaaS strategy before adopting specific IaaS offerings.
3. Start with an infrastructure virtualization project to establish a foundation that enables future cloud adoption.
4. Consider moving to a Private (On-site) deployment model which provides a good initial transition to IaaS with relatively low risk.
5. Consider Private (Outsourced) and Public deployment models which can potentially deliver added business value – closely consider security and reliability issues as well as integration with existing enterprise services.
6. For Public deployments, consider moving only non-critical applications in the early transition phases.

### SMBs

1. Analyze IaaS offerings in terms of TCO/ROI, risks (vendor lock in/interoperability/existing IT infrastructure especially network).
2. Define an IaaS strategy before adopting the IaaS offerings.
3. In many cases, the Private (On-site) deployment model will not be feasible given insufficient ROI associated with consolidating a relatively small number of existing IT assets.
4. Consider the Public deployment model which provides access to computing and storage capacity at the lowest cost.
5. For Public deployments, consider moving only non-critical applications in the early transition phases.
6. Consider Private (Outsourced) deployment to handle spill over of mission critical workloads during periods of high demand or as a backup resource for disaster recovery.
7. Application migration and admin costs must be taken into account for Public and Private (Outsourced) options.

The remainder of this section will take a close look at four common IaaS use cases from the perspective of both cloud consumers and cloud providers. A common theme of all these use cases is the need for standardized interfaces and formats to improve interoperability and portability.

- Virtual environment monitoring
- Hybrid cloud management
- Storage capacity
- Application On-boarding

### 3.1 Virtual Environment Monitoring

A company that provides computerized reservation systems and other services to the travel industry has recently invested in private cloud technologies to host their applications. The company has a multitude of applications leveraging the shared virtualized environment (servers, storage and network) that is used for development and
test, as well as production. They need to effectively manage the health of this environment, ensure they are alerted to capacity bottlenecks and plan for future growth.

**Desired Cloud Implementation**

The company needs to assess the health of their entire environment in a single view to determine whether potential problems are related to their physical server, a virtual machine (VM), storage, or their network environment. This will help them to quickly find the issue that caused the problem so that they can apply appropriate changes.

The company also needs to understand capacity requirements that include future growth of existing systems. There is a need to optimize the existing environment by leveraging automation tools to place workloads appropriately to reduce bottlenecks and align with corporate policies (dev/test on different hosts), and to improve overall capacity (rightsizing). The company needs to report on capacity consumed by and allocated to individual users, notifying users of policies as required (i.e., deleting VMs if they are not used).

**Business Goal**

The primary business goals of the company are to improve the availability of the services it offers and to reduce IT management costs. The combination of an integrated management system to quickly resolve issues along with capacity planning capabilities to optimize usage of resources helps to achieve these goals.

**Necessary Conditions**

The following conditions must be met for the application on-boarding process to be implemented successfully.

- **Security.** Proper access and identity management is needed for the company to provide services to their customers.
- **Interoperability.** The system needs to work with open standard interfaces that provide provisioning through the full life cycle of an on-premise image. In addition, the system requires open standard interfaces to manage the use of internal cloud resources including monitoring usage and performance.
- **Portability.** Virtual machine and application images need to work with any cloud technology provider’s implementation.

**3.2 Hybrid Cloud Management**

Cloud consumers need to dynamically expand and contract use of infrastructure-as-a-service compute resources to an off premise cloud provider based on business demand. The selection of the hybrid cloud model is driven by the enterprise’s need to maintain certain mission critical, oftentimes security-sensitive, applications on-premise while reducing their time to deliver less mission critical applications off premise.

**Desired Cloud Implementation**

While looking to save time and cost by moving to elastic off premise resources, customers require that their security and data standards be maintained. After security standards have been established, the customer will be looking to actually launch the
resources into the cloud. The off premise cloud's provisioning capability must encompass the full life cycle of an off-premise image - create, maintain, and teardown the images at will. Finally, after establishing the use of off-premise resources, the client will be looking to manage the use of cloud resources in a centralized way, including monitoring usage and performance of the off-premise resources using a direct connection to their existing on-premise monitoring system.

**Business Goal**

Customers are interested in reducing their overall capital expense and operational cost by integrating on-premise and off-premise compute resources. They would like to reduce costs and improve overall delivery times while maintaining a high standard of security and governance. In addition, they want to efficiently handle expected and unexpected spikes in workload capacity providing consistent levels of performance to their end users.

**Necessary Conditions**

The following conditions must be met for the provisioning process to be implemented successfully.

- **Security.** The customer will need to establish a secure virtual private network between themselves and the off premise cloud provider. There must be no compromise of their internal firewall and security screens. In addition, on-premise user access levels must be respected during the move of data off premise.

- **Interoperability.** The system needs to work with open standard interfaces that provide provisioning through the full life cycle of an off-premise image. In addition, the system requires open standard interfaces to manage the use of external cloud resources including monitoring usage and performance. Automation tools for workload optimization need to work with any cloud service provider. Lastly, open standard formats and interfaces are required to request pricing and QoS attributes offered by different cloud service providers.

- **Portability.** Virtual machine and application images need to work with any cloud service provider.

**3.3 Storage Capacity**

All IT environments maintain and store information. Because data growth is almost always inescapable, the size and cost of the infrastructure required to store and manage this information is growing. Driven by a desire to become more efficient, IT managers are exploring the promises of cloud for pieces of their storage capacity. The characteristics that distinguish a storage capacity cloud from a traditional IT infrastructure for data storage are:

- The services are standardized rather than customized. Often there is a catalog of available storage capacity and associated quality-of-service (QoS) characteristics that application owners choose from as opposed to individually designing and maintaining custom approaches for each application. The catalog might include

---

3 One specific example is the use of cloud for storage backup infrastructure.
such QoS characteristics as availability, performance, operational recovery, and disaster recovery.

- The infrastructure is shared rather than dedicated. Consolidated storage capacity is shared by individuals, departments or companies who previously operated their own island of capacity and attempted to provide for needed QoS.

- The environment is paid per use or subscription rather than from a centralized budget. Consumers of storage capacity (departments or application owners) pay for their capacity at varying rates based on the associated QoS as opposed to having capacity covered in a CIO-level overhead budget – shifting mindset away from “well, if it’s covered in some other budget, then I want all my capacity with Tier-1 QoS”.

Desired Cloud Implementation

IT managers maintain storage capacity often operated by a storage administration team. The infrastructure is comprised of sufficient (sometimes significant) disk array resources with associated capabilities for creating tiers, snapshots, mirrors, and other QoS characteristics such as the type of connection to servers that need to operate on the data (iSCSI, NAS, Fibre Channel, SAS, etc).

Driven by a desire to become more efficient, there are two scenarios being explored by IT managers.

- A private storage capacity cloud involves enhancing an existing disk capacity infrastructure with cloud-like techniques. Starting with an existing, centralized disk array infrastructure, IT managers define a storage capacity service catalog that describes standardized tiers of storage capacity with associated QoS, assign per-use or subscription prices for capacity consumed under each tier or service level, and provide a self-service mechanism for clients to request and pay for new services. Depending on the use case, the request for storage capacity in a given service level can be initiated either by an individual via a portal or, more likely through a programmatic call initiated as part of a larger virtual server provisioning action.

- A public storage capacity cloud is really a matter of ownership. From the end user perspective, a private storage cloud and a public storage cloud are quite similar – self-service access to a catalog of per-use priced storage capacity tiers or services. The difference is in ownership of the back-end infrastructure. In a public storage cloud implementation, the infrastructure is owned by an external third-party service provider. As a result, it is critical that the SLA with the cloud provider ensure that an appropriate QoS is delivered and that the correct level of security, privacy and location of the data is maintained.

Hybrid IT infrastructures are needed when an IT manager chooses to maintain a server/application infrastructure (virtual or physical) on premise and leverage a public storage capacity cloud for storing data. Due to the network latency involved in connecting on-premise servers and applications to off premise public storage clouds, workloads which are not performance critical, for example, backup and archive workloads, are a good fit for the hybrid model. For a hybrid backup cloud, an IT manager
begins with a private backup infrastructure (backup software with associated servers, storage capacity and networking) and then chooses to move the storage capacity piece of the infrastructure to a public storage capacity cloud. Again, it is critical that the SLA with the cloud provider ensures that an appropriate QoS is delivered and that the correct level of security, privacy and location of the data is maintained.

There are two modes of use.

- An individual user requiring storage capacity accesses a service portal, selects a service level from the catalog, and gains access to the needed capacity.

- Provisioning automation working on a larger job, requests storage capacity from the service catalog via a programmatic interface (ex1: new virtual server and associated storage, ex2: provisioning new clients to a hybrid backup cloud that needs additional storage capacity for the backups).

The most common deciding factors in an IT manager’s choice to implement storage capacity as a private or public cloud will be the scale of the enterprise’s environment and the security sensitivity of the data being backed up. Smaller enterprises can not internally generate the needed economies of scale required to realize all of the cloud benefits. Similarly, as data volumes increase, the network latency associated with public cloud implementations becomes a concern leading some IT managers back to a private cloud implementation. Confidential data, such as personnel and customer information, is typically not appropriate for a public cloud backup.

**Business Goal**

Data growth is inescapable – and moving faster than the density or cost per GB improvements being made by infrastructure suppliers. Left alone, the percentage of IT budget spent on all facets of data storage will continue to increase. The business goal with “cloudification” of the storage environment is to leverage cloud concepts to improve efficiency and cope with data growth.

Private storage administration teams benefit from “cloudification” through improved data storage utilization and efficiency, an accelerated reduction in the cost per GB to store data, and labor productivity gains.

Public service providers are able to create cost efficient offerings that compete well for business from organizations that do not have the competency or scale to create similar private implementations.

**Necessary Conditions**

The following conditions must be met for the storage capacity processes to be implemented successfully.

- **Security.** Data must be secured (encrypted) before being sent to a public backup cloud (and sometimes to a private backup cloud as well). In addition, encryption keys are typically kept by the user and are not available even to the public backup cloud provider. There must be a guarantee that multiple clients can not see any data from or about other clients. Private backup clouds are sometimes implemented to avoid such security exposures.
- **Interoperability.** Consistent, open standard interfaces for accessing and managing private and public storage capacity clouds are required. In addition, open standard formats and interfaces are required to request pricing and QoS attributes offered by different cloud service providers.

- **Portability.** Open standards must be leveraged to ensure that persisted data can be migrated across different cloud storage providers.

### 3.4 Application On-Boarding

Application on-boarding enables ISVs and other application providers to create and manage product definitions with an on-boarding process that certifies a package and makes it available through a product catalog. In addition, the application on-boarding process includes the loading of the image and metadata into a repository.

**Desired Cloud Implementation**

An ISV or other application provider contracts with a cloud service provider to provide base services (compute, storage and/or network) for its application to be executed. As part of the registration process the application provider must be authenticated by the cloud service provider in order to provide its application package. This level of security is important to avoid the introduction of malware into the cloud service provider’s cloud management system as well as preventing an unauthorized partner from accessing the cloud service provider’s infrastructure.

Next, the application must go through a certification process to ensure it executes properly in the cloud service provider’s environment. In most cases, the cloud service provider will have a separate test bed. Once certified, the application package is placed in the cloud service provider’s package management system.

The application is now ready to be provisioned as a part of a service by the cloud management system. In addition, the application can be included as a product definition within the cloud provider’s product catalog.

Figure 1 illustrates the flow for on-boarding a package from a partner to a cloud service provider’s package management system.

1. The application provider is authenticated by the cloud service provider.
2. The application images are registered with the cloud service provider.
3. The application images are maintained and managed by the cloud service provider leveraging an image library.
4. The application metadata is maintained and managed by the cloud service provider leveraging a metadata library.

---

4 A **package** is the combination of application content and/or metadata that comprise the content to be managed in the cloud. A **product** defines what is offered to a customer, and references one or more cloud services that represent packages with their configurations.

5 The ISV or application provider is responsible for certification.
Figure 1: On-boarding a partner package

Figure 2 illustrates the flow for creating a product definition in the product catalog.

1. The application provider is authenticated by the cloud service provider.
2. The application images are registered with the cloud service provider.
3. The application image and metadata information is obtained.
4. The application image and metadata information is included in the product definition within the cloud service provider’s product catalog.
**Business Goal**

Cloud service providers have incentive to expand their portfolio to make their cloud offerings attractive to their service consumers. This requires developing a partner ecosystem. Providing a standard interface to on-board third party applications speeds time to market which, in turn, helps generate revenue.

**Necessary Conditions**

The following conditions must be met for the application on-boarding process to be implemented successfully.

- *Security*. Proper access and identity management is needed for the partner to provide their applications to the cloud service provider.

- *Interoperability*. One of the goals of the use case is to develop an open standard package format and interface to on-board (i.e., deploy) third party applications into a cloud service provider infrastructure. In addition, open standard management interfaces are required once the application is deployed.

- *Portability*. The application package that’s developed during the on-boarding process needs to work with any cloud service provider.

**4.0 Platform as a Service (PaaS) Use Cases**

PaaS leverages an integrated development and runtime platform optimized for creating, deploying and managing cloud applications. A PaaS environment dynamically adjusts
workload and infrastructure characteristics to meet existing business priorities and service level agreements (SLAs).

PaaS helps eliminate the need for developers to work at the image-level, enabling developers to completely focus on application development. It will also help reduce software design steps and enable faster time-to-market using predefined workload patterns.

The incentives for an enterprise to transition to a PaaS environment differ based on the size and IT maturity of the enterprise. For large enterprises, a key motivation for considering PaaS is the ability to quickly and inexpensively develop and deploy new applications. Large enterprises have additional incentives for considering a move to PaaS:

- Integrated development and runtime platform for specific workloads
- Consistent, pattern-based deployments for most common workloads
- Integrated workload management for SLA enforcement, dynamic resource management, high availability and business priorities
- Awareness and optimization of workloads based on business priorities and SLAs
- Consolidated workloads under a simplified management system

**Approaches for adoption of PaaS:**

Organizations with mature IT systems already have significant investments in their development and runtime platforms along with significant investments in human resources associated with solution development and testing. As a result, they will initially look to re-factor these assets as they transition to cloud computing.

In many cases, SMBs do not possess the resources to invest significantly in development and runtime platforms and they lack the in-house human resources to develop and test home-grown applications. Many SMBs are dependent on ISVs to deliver their application functionality. As a result, they are dependent on an external cloud provider to support a PaaS environment that is consistent with their ISVs’ applications.

The approaches that could be adopted by both these types of organizations are depicted below:
### Approach for PaaS adoption

<table>
<thead>
<tr>
<th>Large Organizations</th>
<th>SMBs</th>
</tr>
</thead>
<tbody>
<tr>
<td>1. Analyze PaaS offerings in terms of total cost of ownership (TCO)/return on investment (ROI) and risks such as vendor lock in/interoperability/existing IT infrastructure especially network.</td>
<td>1. Analyze PaaS offerings in terms of TCO/ROI, risks (vendor lock in/interoperability/existing IT infrastructure especially network)</td>
</tr>
<tr>
<td>2. Define a clear PaaS strategy before adopting specific PaaS offerings</td>
<td>2. Define an PaaS strategy before adopting the PaaS offerings</td>
</tr>
<tr>
<td>3. Consider starting with either the Private (On-site) or Private (Outsourced) deployment model which provide a good initial transition to PaaS for both mission critical and non-mission critical workloads with relatively low risk</td>
<td>3. Determine if there’s sufficient in-house development resource to justify the expense of a PaaS environment – if not, SaaS may be the best alternative</td>
</tr>
<tr>
<td>4. For Public deployments, consider moving only non-critical applications in the early transition phases</td>
<td>4. If sufficient in-house development resources exist, both the Private (Outsourced) and Public deployment models are viable options - selection will be dependent upon the mission criticality of the services being developed and deployed.</td>
</tr>
</tbody>
</table>

The remainder of this section will explore five common PaaS use cases from the perspective of both cloud consumers and cloud providers:

- Policy-driven application management
- Development & Test – Problem Determination
- Development & Test – Application Promotion
- Database as a Service
- Big data analytics

### 4.1 Policy-driven Application Management

This use-case enables the policy-driven management of applications, delivered by the PaaS layer, also referred to as the *cloud application platform*. This system leverages an application-centric approach that packages the business logic (.war, .ear, etc) with metadata that represents the functional and non-functional requirements. In the current environment, the functional and non-functional requirements were design documents that informed systems administrators how they should install, configure, and integrate middleware components to meet the requirements of the line of business. This new system enables those requirements to be represented as meta-data, and the cloud application platform is able to instantiate the middleware components in the precise configuration needed to meet the requirements. Moreover, the cloud application platform is able to perform full life-cycle management of the deployed middleware components, including patch management, runtime optimizations such as auto-scaling, and so on.
Definitions

- **Application Deployer.** An entity that produces an application and packages it for deployment to a Cloud Application Platform (aka Platform as a Service or PaaS).
- **Application Model.** The combination of application content (.ear, .war, etc files) and metadata that describes the functional (DB connections, queue connections, etc) and non-functional requirements (availability, scalability, security, etc) that the underlying PaaS must implement.
- **Cloud Application Platform Administrator.** The entity that manages the collection of application runtimes, application runtime topologies, global policies for non-functional and functional requirements, and shared system services.
- **Runtime Plugins.** The combination of application runtime & configuration scripts that an application can run on.
- **Cloud Application Service Catalog.** A catalog within the cloud application platform that contains the list of plugins, global policies, and shared-system services.
- **Cloud Application Platform.** The system that manages the application model, including application life-cycle, instantiation and enforcement of non-functional and functional requirements. The system instantiates the plugins necessary to meet the requirements described within the application model. The cloud application platform can instantiate the Application Model across one or more Cloud Resource Groups.
- **Cloud Resource Group.** A collection of hardware resources including compute, storage, and network.
- **Infrastructure as a Service API’s.** The API’s that represent one or more cloud resource groups that the cloud application platform interacts with to instantiate the application model. These APIs need to be defined and freely available as an open standard.

**Desired Cloud Implementation**

In the current environment, application deployers must describe via design documents the functional and non-functional requirements needed for a given application. IT operations consumes these design documents, and must install, configure, integrate, and maintain the software stacks necessary to meet the described process.

In the desired implementation, an application deployer describes the functional and non-functional requirements via meta-data (JSON, XML, etc). This description is known as the application model. The application deployer governs the life-cycle of the application model, where the application model is a contract between the application deployer and the underlying system. The underlying system consumes this contract and based on the services within the system’s catalog, interacts with the underlying IaaS layer to instantiate the middleware configuration that implements requirements within the contract.

An application deployer can submit an application model for execution to a PaaS running in any of the cloud deployment models. Cloud service providers can then on-board the deployed applications and offer them as revenue-generating services.
The figure below illustrates the steps for defining and instantiating an application model in a cloud.

1. Application deployer defines an application model and submits it to the cloud application platform for instantiation.
2. The cloud application platform determines the correct physical architecture based on the application model, assembles the necessary runtime plugins to instantiate the application model.
3. The cloud application platform interacts with the IaaS to instantiate the necessary components.

The figure below illustrates the steps for exposing runtime plugins to application deployers.

1. The cloud application platform administrator installs the new runtime plugins to the cloud application platform.
2. The cloud application platform exposes the new plugins via additional application model elements which are made visible to application deployers.
The figure below illustrates the steps for instantiating runtime plugins across different cloud resource groups.

1. Application deployer defines an application model with different elements of the model deployed to different cloud resource groups.
2. The cloud application platform determines the correct architecture based on the runtime plugins and application model.
3. The cloud application platform instantiates the runtime plugins across the different cloud resource groups in accordance to the application model.
Business Goal
The primary business goal is to reduce the cost of management applications and expose middleware services without exposing the complexity of the middleware. The tasks of installing, configuring, integrating, and maintaining middleware are simplified (and eliminated) by relying on the underlying cloud application platform to provision middleware platforms as a service.

Necessary Conditions
The following conditions must be met for the application on-boarding process to be implemented successfully.

- **Security.** The application deployer will need to establish a secure virtual private network between themselves and the off-premise cloud provider.

- **Interoperability.** One of the goals of the use case is to develop a standard application model and to enable the creation and multi-provider deployment of applications. An application model, defined and available as an open standard, is the central, governed artifact in the cloud system. A standard application model enables applications to be deployed to multiple PaaS technologies, which in turn are running on multiple IaaS layers, with minimal disruption.

- **Portability.** The application model must be portable across IaaS, and only prereqs a PaaS that can consume the application model. The PaaS layer is responsible for
managing and executing the application. The IaaS layer is responsible for providing the necessary compute resources. The application model provides an abstraction of the application and its needs that is decoupled from the vendor-specific nuances of both the PaaS and IaaS.

- **Other.** If no standardized IaaS API’s exist, or at least a manageable set of API’s, the application model’s portability, and the cloud application platform’s ability to manage the application are severely limited.

### 4.2 Development & Test – Problem Determination

Developers often spend countless hours trying to reproduce complex scenarios where a problem was detected from a tester in the test organization. In many circumstances the problem cannot be reproduced within a development environment causing more risk to the release. The intention of this use case is to use cloud to speed the delivery of applications to be tested by a test organization and, more importantly, provide the ability to save the state of a cloud environment to be used for problem determination.

**Desired Cloud Implementation**

In the current system, it is often a difficult and manually intensive task for developers to reproduce problems that were detected either within production and/or during a normal testing cycle conducted by the test organization. Developers are typically provided a description of the problem and the environment in which they found the problem. If lucky, the tester will accurately document the necessary steps to reproduce the problem. Even with accurate description of the steps, a developer could spend countless hours trying to setup the same environment and circumstances to reproduce the problem that was detected.

The desired outcome is to automate the deployment of application required infrastructure and platforms to the cloud to improve the efficiency and productivity of the development and test organizations. But more importantly, it is desired that when a problem is detected during the testing of an application deployed to infrastructure in the cloud, a service is available from the cloud provider to capture a snapshot of the running state of the environment exhibiting the problem.

A tester would select an action from the cloud provider to capture a snapshot of a set of related VMs working together in a topology and a URL would be returned providing a link to the stored location of the snapshot. The URL may then be associated a defect record within a change management system.

The cloud provider must provide an API that allows the developer to resume the saved snapshot topology by passing the URL defining the stored state. The API will retrieve the saved topology state and resume it including resuming each VM that is part of the stored topology. With the state restored the developer can more effectively solve the problem versus spending time provisioning an environment and recreating the conditions that caused the problem.
Business Goal
Using cloud services to capture the running state of a test environment reduces critical
time and effort spent recreating problems that are often due to specific conditions that are
sometimes difficult to reproduce.

Necessary Conditions
The following conditions must be met for the problem determination process to be
implemented successfully.

- **Security.** A secure connection is used for all of the interactions between the
development and test organizations and the cloud-hosted VMs.

- **Functionality.** It will be required that each cloud service provider support an API
to snapshot a running topology that would consist of one or more running VMs.\(^6\)
The snapshot will contain the running state of each VM. The service provider
must return an HTTP URL that addresses the snapshot stored within the cloud. A
cloud service provider must provide an API to resume a snapshot given the URL
provided.\(^7\)

- **Portability.** At this time, there is not a requirement to share snapshots of a running
environment (i.e., instances of VMs) across different cloud service providers.

4.3 Development & Test – Application Promotion
Almost all organizations follow a development process that requires deployment to
multiple test and stage environments prior to moving applications into production. This is
necessary to properly test the application to detect errors well before application changes
are released into production. When working with multiple environments there are
challenges detecting differences between the environments. There are also points within
the development process when the application changes and automation tasks have been
tested and it is desired to simply move (i.e., promote\(^8\)) the application and configurations
from one environment to another with only minimal changes (environmental specific
only). Having a process to automate the promotion process reduces risks by reducing
errors caused by manual or automated tasks.

Desired Cloud Implementation
In the current system, organizations often manually move applications from one
environment to another environment by trying to redefine the necessary configurations.
More sophisticated organizations will utilize consistent automation tasks/scripts.
Whenever manual and even automated tasks are run there is an opportunity for errors to

\(^6\) In the future, it may make sense to standardize the snapshot API to provide a level of consistency across
different cloud service providers. This is particularly true for hybrid cloud environments.

\(^7\) The format of the snapshot is out of scope for this use case because the assumption is that a snapshot
cannot be transferred between cloud service providers.

\(^8\) Promote. To move applications and deployment configurations from a source hosting environment to a
target hosting environment using transformation rules to automate a select set of environmental specific
properties.
be injected into the process. When moving from a staging environment into a production environment there is no room for even a slight opportunity of an error.

It is desired that development teams can use the cloud to provision the necessary infrastructure and platforms, and use automated tasks for provisioning middleware configurations and install applications. This is necessary for each test environment. However at certain points in the development life cycle (typically after the application has passed all necessary quality tests) it is desired to simply move or promote the application from one environment to another with only minimal environmental specific changes (e.g., host names). To aid development teams in this task it is expected that a cloud service provider allows teams to define and manage deployment environments (sometimes called topologies). The cloud service provider is expected to provide a utility that allows users to compare two environments and generate a report with the differences. This capability will aid in problem determination as well as validation that a move of an application was successful.

To aid development teams in moving applications and their deployment configuration from one environment to another environment, a cloud service provider will provide a service to “promote” applications and configurations from a source hosting environment to a target hosting environment. The cloud service provider can restrict promotion to compatible environments. It is expected that the cloud service provider would provide a mechanism to replace environmental specific configuration values during the promotion process. This is necessary because deployment configurations often contain values that are specific to the target hosting environment. To ensure the promote service worked correctly teams can use the environment comparison support to ensure only environmental specific differences exist between the environments.

**Business Goal**

Using cloud services to promote applications and deployment configurations from one environment to a target environment dramatically reduces risk by removing almost all manual and/or scripted changes. Such a service ensures the greatest degree of consistency between environments.

**Necessary Conditions**

The following conditions must be met for the application promotion process to be implemented successfully.

- **Security.** A secure connection is used for all of the interactions between the development and test organizations and the cloud-hosted VMs.

- **Functionality.** It will be required that each cloud service provider will support an API to compare two running environments and generate a report with the differences. A cloud service provider MAY support comparison of two off-line environments (i.e., not provisioned).


9 At this time, there does not appear to be a strong interoperability requirement for defining a formal standard for this API.
environments must be under the management of the cloud. There is not an expectation that the promotion API will support environments outside of the cloud’s management.\(^\text{10}\)

- **Other.** Promoting applications from one environment to another may require specific knowledge of the installed runtime platform such that platform-specific configuration information can be promoted correctly.

### 4.4 Database as a Service

Large enterprise customers have many departmental systems such as payroll, accounting, call centers and personnel management that could be consolidated into private cloud. These customers have thousands of servers across their infrastructure. The goal of consolidating onto the private cloud is to reduce cost of ownership by eliminating redundant systems and administration of these typically separate systems.

**Definitions**

- **DBaaS.** Any cloud is analogous to a vending machine with the main difference being what it provisions. If it provisions a virtual machine, it is an Infrastructure as a Service. If the cloud provisions databases, it is a Database as a Service. DBaaS APIs focus on the provisioning of DBs (create, backup, delete). Database access APIs are not part of this set.

- **Database (Instance).** A database in a formal DBaaS has a slightly different meaning to that of traditional environments. A database in a DBaaS is a virtual concept that is implemented in a way that is ideally hidden to the applications group. The operations team and the DBaaS technology itself define how a single DB should be hosted and may include:
  
  1. A DB on a virtual machine
  2. A DB on several virtual machines (using something like HADR)
  3. DBs on bare metal
  4. DBs on a multi-tenancy DB architecture

The definition of “database” in a DBaaS is the virtual representation of a database to the provisioning user and not necessarily a physical database from traditional environments. It is also worth noting that a DB is owned by the application group, not the operations group with a DBaaS.

- **DB Images.** Like an IaaS which has VM images, a DBaaS has DB images and the concepts are almost the same. With an IaaS, the virtual machine image can be captured from a live virtual machine after configuration and setup. With a DBaaS, a DB image can be captured from a live DB after data has been loaded and the DB has been tuned accordingly.

---

\(^{10}\) At this time, there does not appear to be a strong interoperability requirement for defining a formal standard for this API.
A DB image should include all of the underlying data and configuration information from the live DB instance. In other words, a DB instance created from a DB image would be a “clone” of the original live DB that the image came from. This includes the QoS, HA configuration, DB users, DB configuration, data, etc. Under the covers, a DB image is implemented as a database backup plus metadata that includes information not captured in a DB image. A DB backup is equivalent to a DB image.

**Desired Cloud Implementation**

On premise IT shops have to consider the best place to remove cost in their organization. The large or mission critical databases are too big, too complex, too few in number or too important. However, the smaller departmental databases are a prime candidate for cost reduction for the following reasons:

- They are less important
- They are easier to standardize (do one thing right, many times)
- There are many of them

It is estimated that approximately 80% of the databases in IT shops are less than 200GB in size. These are the primary candidate databases for an on-premise DBaaS.

Database customers (especially the DB administrators (DBA)) tend to be risk adverse. They will (typically) not use unproven technology for production workloads. Customers will also not tolerate solutions that put their database at risk. DBAs are the ones who will get paged if a DBaaS does not manage a DB as it should. Trust will need to be earned by taking small steps and proving the technology.

The private cloud implementation should allow customers to provide their own hardware or use vendor provided hardware to provision and automate the management of the databases for these departmental applications. Additionally, this provisioning must use best practices upon deployment such as tuning and securing the database and operating system. To be successful, the implementation must automate the high availability, backup and restore tasks necessary to support the potentially large number of smaller databases within an enterprise.

A DBaaS applies equally to the public cloud as well as the private cloud. In both cases, the application group has an interest in focusing more on their application and business goals and less on their database and/or other middleware. Because privacy and security are major concerns, a private cloud is often desired.

**Business Goal**

The main goals of a DBaaS are to reduce cost for the operations team and provide self-service databases to the applications group.

**Necessary Conditions**

The following conditions must be met for the application on-boarding process to be implemented successfully.
- **Security.** There is a strong requirement to leverage existing user directories and authentication systems within the current infrastructure.

- **Privacy.** Private cloud implementations will oftentimes be chosen to keep data on premise without incurring additional privacy concerns. In some cases, it may be illegal for the data to cross jurisdictional boundaries.

- **Interoperability.** An open standards-based provisioning interface will ensure consistency across private and public database deployments (i.e., hybrid environments).

- **Portability.** At a minimum, an interoperable import/export procedure must be provided to transport DB images from one cloud service provider to another.

- **Other.** Business usage of cloud resources must be properly accounted for through commonly understood usage terminology: # of transactions, # of users, # of database etc. In addition, the solution needs to automate database administration as much as possible.

**4.5 Big Data Analytics**

A financial services company is looking to perform fraud analysis on all credit card transactions. The company utilizes a proprietary self-learning algorithm that enhances its capability by processing very large volumes of transaction histories. The company is looking to utilize an Apache Hadoop system to process internet scale data sets using large scale compute clusters.\(^\text{11}\)

**Desired Cloud Implementation**

Creating large scale Hadoop clusters requires very significant capital investment to procure and deploy compute and storage resources. This IT infrastructure is lightly utilized most of the time as Hadoop jobs are akin to batch processing in nature. Creating a dedicated Hadoop cluster to handle periodic large scale jobs is an expensive proposition with low utilization of capital.

Cloud computing provides a perfect solution to the demands of Hadoop processing. Using virtualized cloud systems, Hadoop clusters of almost any size can be created on-demand and for a fraction of a cost of dedicated infrastructure. For example, a Hadoop cluster of 100 servers would require an initial capital investment of half a million dollars and significant effort and cost to deploy and maintain. On the other hand, creating such a cluster and using it for an hour on a public cloud can cost less than $10 with zero up front expenditure. The world’s largest Hadoop cluster (Yahoo.com) has 40,000 servers. Building such a cluster would require capital expenditure in the hundreds of millions of dollars, but the cost of renting such a cluster (if sufficient cloud capacity is available) can be as low as $4000.

Furthermore, creation of such clusters and running of Hadoop jobs can be tailored to the spot market price for compute resources. For example, jobs requiring significant resources can be scheduled to run on public cloud at the times when spot prices drop

\(^{11}\) Refer to http://hadoop.apache.org/ for details on Apache Hadoop.
below a predefined threshold. This allows companies to conduct analysis that would otherwise be unaffordable and would not be cost justified.

Moving dedicated Hadoop cluster infrastructures to take advantage of virtualized automated environments of private clouds offer similar benefits. IT managers can schedule allocation of resources in the private cloud to Hadoop jobs at the time of low activity for other workloads. The virtualization and automation of private clouds enables rapid on-demand provisioning of IT resources required to run Hadoop jobs and enables companies to derive value from their data that would not be affordable or economically justifiable in traditional IT environments.

Hybrid clouds provide additional flexibility. A company may create a modest size private cloud to accommodate steady-state workload. At the times of peek workloads additional IT capacity can be provisioned from one or more public cloud providers. The cost of provisioning can be further enhanced by taking in to account public cloud provider price structure such as availability of spot pricing.

**Business Goal**

Leveraging IaaS, companies can greatly reduce, or eliminate all together, large capital expenditures associated with big data analytics. Cloud computing can be used to enable analysis of internet scale data sets that would not be cost justified using traditional IT methods. Public clouds enable even the smallest companies to derive value from big data. Private clouds enable larger organizations to dramatically reduce resources and costs yet deliver secure, managed environments for analyzing sensitive data.

**Necessary Conditions**

The following conditions must be met for the application on-boarding process to be implemented successfully.

- **Security.** When utilizing public cloud providers or transferring data over public networks, special care must be taken to ensure security. Public cloud providers must have documented and audited processes in place to ensure data security. A VPN connection should be used for all of the interactions with the Hadoop cluster in the public cloud.

- **Privacy.** Typical analytical jobs do not require access to personally identifiable customer information. Therefore, data loaded for analysis should be masked as to obscure any and all information that can jeopardize privacy and cause failure to comply with various industry and government regulations.

- **Interoperability.** Open standard interfaces are required that provide provisioning through the full life cycle of off-premise images. In addition, the system requires open standard interfaces to manage the use of external cloud resources including monitoring usage and performance. Automation tools for workload optimization need to work with any cloud service provider. Lastly, open standard formats and interfaces are required to request pricing and QoS attributes offered by different cloud service providers.

- **Portability.** Virtual machine and application images need to work with any cloud service provider.
• *Other.* Hadoop analytics involve terabyte or even petabyte data sets. Ability to transfer such large data sets into cloud-based Hadoop clusters is paramount.

### 5.0 Conclusion

The use cases described in this paper are not all encompassing; however, they do represent the scenarios that are important to the members of the CSCC. Consequently, these are the areas where the IaaS and PaaS working groups of the CSCC will focus their near term efforts. Both working groups will leverage these real world use cases to develop specific standards requirements which can be brought forward to the appropriate standards development organizations (SDO) for consideration.

If you believe there are additional use cases that should be considered or if you’re interested in contributing to the detailed standards analysis of the current use cases, please consider joining the CSCC and working with your peers on this important initiative. Membership is free and can be completed on the web at [http://www.cloud-council.org/membership-application.htm](http://www.cloud-council.org/membership-application.htm).