Independent market research and competitive analysis of next-generation business and technology solutions for service providers and vendors

HEAVY READING
WHITE PAPER

BSS Cloud Transformation:
Beyond Monolithic

A Heavy Reading white paper produced for STL

AUTHOR: JIM HODGES, RESEARCH DIRECTOR, HEAVY READING

INTRODUCTION

As communications service providers (CSPs) continue to embrace cloud services, they must also look inward and assess how this shift will transform their existing operational processes.

A key component of this assessment process is documenting the operational and design impact on business support systems (BSS) and their ability to support cloud key performance indicators (KPIs) and open application programming interfaces (APIs) in an Open Digital Architecture (ODA) environment.

This white paper documents the requirements for BSS in the cloud transformation process, with a focus on the technical requirements of cloud-enabled BSS and the best practices for implementing them.

CLOUD BSS DESIGN REQUIREMENTS

While CSPs' transformation to the cloud has already commenced, it will be gradual; there are too many embedded service and provisioning capabilities that are critical to service enablement to support a flash cut. This is advantageous in several respects, including providing much needed time for CSPs to rearchitect their cloud-based BSSs.

The first step in this process is evaluating how legacy BSS differ from cloud BSS. At a high level, the differences are shaped by several factors: universal service delivery, location transparency, and open KPIs.

Universal service delivery and location transparency

Without question, the cloud will foster a universal service delivery model. In today's monolithic world, CSPs are often not able to deliver a full suite of legacy services because the cost of deploying hardware and software in remote regions with only a small customer segment to upsell is prohibitive. This is due in large part to the complex nature of how these legacy monolithic services are architected and provisioned.

In contrast, cloud compute represents a low cost service delivery model and can meet the most stringent latency requirements. The impact on BSS is that they must be able to support seamless interaction with cloud service logic platforms in a real-time environment. They must also support location transparency to assist in seamlessly moving distributed workloads between any cloud type (e.g., private, public, and edge cloud).

Cloud KPIs and open APIs

One of the key attributes of the cloud is that it drives new approaches to how services are created and where they are deployed. This includes the ability to implement a service delivery model using an underlying shared and hosted software as a service (SaaS) model.

Consequently, as cloud services are rolled out, KPIs to support these new services must be deployed. In the legacy BSS world, KPIs were well defined because the service performance requirements tended to be static and did not change much over time.

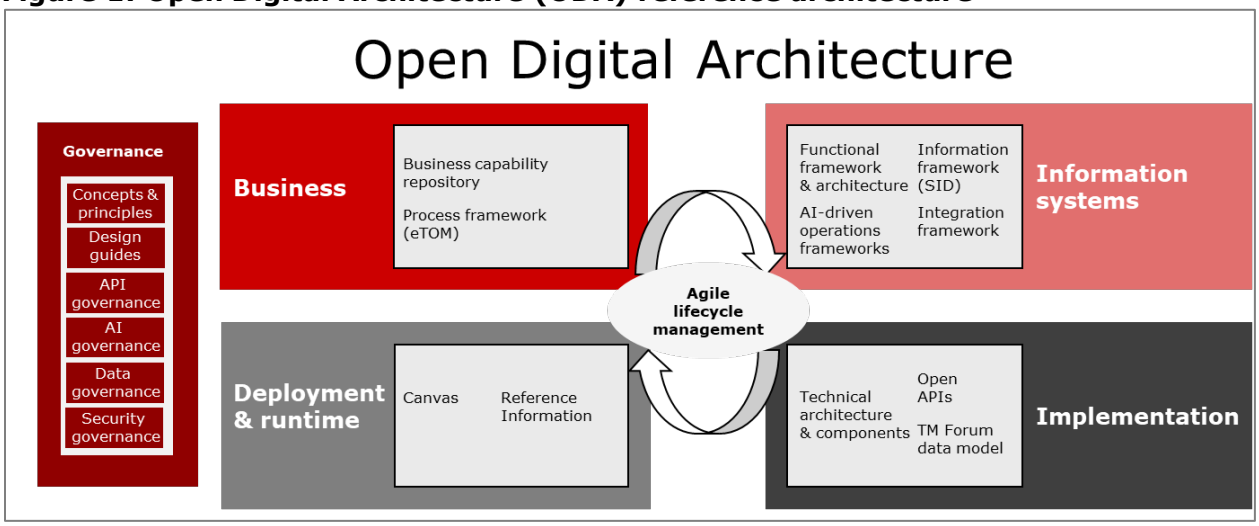
Therefore, BSS, like the services they supported, were also monolithic and based on proprietary hardware and software, which were expensive and time-consuming to update when changes were required. In contrast, cloud services are created and delivered in a much more dynamic multicloud environment, so KPI measurement must be agile and flexible to meet a much more dynamic cloud-centric service model.

Similarly, APIs must also evolve. While over the years, vendors and application developers have positioned their APIs as "open," when fully assessed, they contain some vendor-specific elements that make management more complex. The journey to the cloud is fueling a drive to openness in hardware, software, and APIs.

To truly leverage the power of the cloud, including the use of open source APIs, vendors and developers must become open to facilitating interworking and minimizing software delivery costs. Such an approach is enhanced by the availability of reusable microservices, which can shorten new service development cycles. This push to open APIs has major BSS design implications.

In response, the TM Forum has created the ODA, which documents concepts and principles associated with managing BSS requirements. **Figure 1** provides a summary of the considerations addressed by the ODA reference architecture.

Figure 1: Open Digital Architecture (ODA) reference architecture



Source: Heavy Reading interpretation of ODA reference architecture

The ODA is important on multiple levels because, as captured in **Figure 2**, it provides a framework for the end-to-end implementation of services using a shared data model based on open APIs.

Figure 2: ODA key functions description

ODA component	Key functions	Description
Business	Process framework	Delivers a flexible and programmable framework with the necessary level of agility for new service introduction.
Information systems	Information framework	Provides a common information framework. Supports an artificial intelligence (AI)-driven operations framework.
Deployment and runtime	Operations framework	Enables business agility through the deployment stack, supporting design time and runtime capabilities integration and ensuring minimization of time and cost while reducing the risk of launching a new product/service or bringing onboard a new enabling service.
Implementation	Data models and open APIs	Defines end-to-end building blocks from definition to termination, ensuring design flexibility. Supports a model-driven catalog that eliminates the need to rely on manual configuration for new service chains. Enables open integration through open APIs.

Source: Heavy Reading

BSS CLOUD TRANSFORMATION BEST PRACTICES

In addition to implementing an ODA architecture to facilitate the transition to cloud BSS, CSPs must also formulate an effective implementation strategy. This strategy must be based on business and technology best practices.

While some best practices may vary from service provider (SP) to SP based on service demands, this section of the white paper presents several common best practices that Heavy Reading believes all SPs should consider in their cloud BSS journey.

Best practice 1: Develop a transformation template

It will take many years for CSPs to fully migrate from legacy BSS platforms to cloud platforms, but planning must start immediately for CSPs that prefer to retain control of BSS versus purchasing these capabilities from a third-party provider.

Rather than focus on an aggressive cloud transformation strategy, Heavy Reading believes it is imperative that CSPs committed to building their own cloud systems should start with a very small number of basic applications. They should also put in place the vendor ecosystem and internal processes to migrate these underlying legacy services to the cloud.

The above process will create a template that can be used for migrating existing services and, perhaps more importantly, new cloud services that will unquestionably emerge over the next five to eight years.

This template will ultimately foster the creation of an environment that will shorten time to market and enable the creation of service performance metrics that can factor in application workload patterns and the desired latency characteristics.

Best practice 2: Factor in SaaS

In addition to the technical impacts, the cloud transformation also has significant business impacts. As a result, CSPs must consider all the business opportunities the cloud offers. One consideration that factors prominently into the equation is access to the SaaS-based cloud-hosted service delivery model.

5G represents the first generation of cloud architecture, so SaaS commercialization will open up a much greater range of high value upsell opportunities for CSPs. These include new services, such as the Internet of Things (IoT), slice-based, and edge services.

Flexibility is vital for success. To fully exploit the value of SaaS, CSPs will need to validate that the SaaS-based service enablement processes of their BSS are flexible enough to meet the scale and low cost delivery demands that CSPs will need to drive new revenue streams.

Best practice 3: Define cloud metrics

With so much change driven by the cloud, it is important to define new metrics that are relevant for cloud-native services that range from basic IoT device monitoring to ultra-low latency 5G private network-based services, such as autonomous vehicle monitoring (see **Figure 3**).

The key construct for these metrics is that they emulate the flexibility of the microservices of the services they support. To accomplish this, KPIs must be open and must be able to capture performance for any stateless services.

In addition, these cloud-native metrics must align with a continuous integration (CI)/ continuous delivery (CD) model and be able to support zero message loss and easy rollbacks with no backward compatibility defects when problems during code upgrades are encountered.

Best practice 4: APIs and application health

While microservices offer advantages in terms of reuse, scale, and even service cost, they also introduce additional complexity. This is due, in large part, to the number of distinct service functions that must run simultaneously to create a single application. If one function starts to fail, it can impact the service instance.

Even in a failure scenario, if implemented correctly, microservices should be more resilient than a monolithic architecture where a single component failure can have a greater overall service impact. However, detection is more complex because of how these service interactions are coupled and interact.

To address this concern, applications must integrate health check APIs to support auto-healing for real-time applications. In this role, the API monitors network connectivity and ensures that even underlying capabilities, such as application server availability, are not compromised.

Again, openness is key to ensuring that APIs have unrestricted access to all network functions to have the "big picture" of application health.

Best practice 5: Include multi-domain orchestration in the transformation journey

The cloud will support new services, such as the Industrial Internet of Things (IIoT) and Wi-Fi 6, which will provide a suite of new enterprise service opportunities. This will drive the need to manage and orchestrate services on a much greater scale than before.

In turn, cloud BSSs must be designed to be able to flexibly support orchestrated services. This includes the ability to support orchestration from conception to retirement and across all phases of service enablement, including planning, deployment, customer onboarding, and service assurance.

The challenge, of course, is the expansion of where orchestration needs to be performed. Applications will be running in private, public, hybrid, and edge cloud domains, so orchestration must be flexible enough to orchestrate and manage the application lifecycles in this multi-domain environment.

Best practice 6: Commit to DevOps with CI/CD integration

Although DevOps and CI/CD are often positioned as optional to enhance service delivery processes, the reality is that CSPs' successful cloud transformations will demand some level of commitment to such an implementation.

DevOps is important because it exploits the flexibility of software to streamline delivery to reduce costs associated with new service delivery. The value of DevOps is that it is complementary to microservices adoption in that it simplifies the process of extending new services or delivery of new services by creating a more responsive DevOps developer environment.

CI/CD also plays a key role in the services process. CI/CD makes it possible to build new applications, integrate automation-based testing, and enforce new policies in real time.

Heavy Reading believes that committing to implementing and integrating CI/CD in a DevOps environment is no longer simply an option; rather, it now represents a strategic imperative.

Best practice 7: Using automation

Starting with 5G, ultra-low latency services will become the norm in the cloud. In response, there must be a shift from relying on human intervention to automated policy enforcement for service execution.

This means a BSS transformation strategy must address how and where automation can be used in a cloud BSS.

One area that is emerging is leveraging AI for a broad range of functions, including network alerts, zero-touch provisioning, and support for CI/CD-based processes. In all of these scenarios, AI will enhance the real-time monitoring and processing of application data.

This will facilitate better insights into the performance of intelligent digital services wherever they are deployed, especially when integrated with analytics that have access to application-level personalized service consumption models.

Best practice 8: Reassess ecosystem and vendor selection processes

Transitioning to a cloud BSS will disrupt current BSS models, in part due to SaaS adoption. As a result, CSPs need to consider their vendor and ecosystem partner relationships. The bottom line is there is no incumbency in the cloud. This is because there are so many third-party application developers, each with its own unique strengths.

However, to leverage these third-party products, open hardware platforms and open APIs are important because they are both integral to implementing the ODA, which will deliver the flexibility and agility to be a success in the cloud domain.

In other words, CSPs must focus on agility and flexibility when selecting BSS partners and ecosystem partners because the pace of service innovation will demand that BSS become more programable and extensible than in the past.

Best practices summary

A successful cloud BSS transformation will hinge on the implementation of the best practices documented above. This is because all, in some way, have a positive benefit on service execution by fostering the creation of a flexible and agile BSS environment optimally positioned for managing and monitoring myriad use cases. **Figure 3** provides additional insight by mapping the benefits of each of the eight defined best practices to a specific use case.

Figure 3: Best practices – Use case benefits mapping

Best practice	Service use case	Benefits mapping
Develop a transformation template	Private cloud application—5G slice-based service	5G slices are all unique. The template will allow a CSP to create a low risk slice, such as a private cloud application, and use the template to ensure that BSS can validate conformance to runtime KPI requirements.
Factor in SaaS	Security as a service (SECaaS)	The integration of security services into their telco cloud is enabling CSPs to deliver a broad portfolio of scalable and lower cost security services to small and medium business customers. While SECaaS simplifies the delivery of security services for these customers because each has unique security requirements, CSPs must ensure that cloud BSS are agile and scalable enough to support the provisioning and monitoring of tailored services in a cloud domain.

Best practice	Service use case	Benefits mapping
Define cloud metrics	Mission-critical microservice-based applications	Cloud microservices differ from traditional services in that they use a microservices design model. Unlike legacy services, microservices decompose traditional services into single-function reusable service blocks. This is highly desirable as it simplifies new service design, and it is much more scalable and aligns with service design models of hyperscalers. Moreover, it represents a must more flexible approach for enabling new mission-critical applications, such as private network-based autonomous vehicles that are extremely latency-sensitive. In a design context, to support these microservices, BSS must be software configurable to support real-time provisioning while maintaining the ability to support legacy monolithic designed services.
APIs and application health	IoT edge-based health services	Edge-based IoT services, such as a connected car that integrates third-party applications, require open APIs to ensure that performance metrics are met as new IoT service capabilities and devices are deployed at the edge. Some of these IoT services, such as health monitoring, are mission-critical, so monitoring application performance and stability is crucial.
Include multi-domain orchestration in the transformation journey	Wireless wireline convergence (WWC)	The implementation of the cloud will harmonize the fixed and mobile services domains and enable the delivery of a common suite of WWC services. This will be a major improvement for customers currently limited by the underlying access network used for service execution. For CSPs, this translates into new revenue opportunities associated with selling a harmonized service model to *all* subscribers based on a WWC architecture model. The challenge is that it means that orchestration must be supported on a multi-domain model compared to the current individual fixed or mobile domains. Again, openness, automation, and programmability are key because they will enable BSS to support seamless orchestration among both multi-domains and multi-vendor implementations within these domains.

Best practice	Service use case	Benefits mapping
Commit to DevOps with CI/CD integration	5G core network exposure of third-party services	The 5G core is based on a Services-Based Architecture (SBA), which supports the exposure of API-based microservices. This approach, in turn, opens the door for CSPs to integrate and expose not just specific vendor services, but any third-party service that complies with the API framework. This broadens the spectrum of service innovation and also aligns with the concept of CI/CD, which supports a constant DevOps-fueled software stream compared to the inflexible and vendor-specific limited software release cycles of monolithic networks. To support CI/CD in a DevOps environment, BSSs must be completely programmable to enable software updates without restriction, while supporting vigilant audit processes with the ability to restore to the previous software framework if problems are encountered.
Use automation	Operations and maintenance	Automation supports real-time, in-depth service-level agreement (SLA) and quality of experience (QoE) monitoring and policy enforcement whether BSS elements are deployed in public or private clouds. This now represents an essential capability to enable seamless monitoring of disaggregated services deployed anywhere in the cloud (e.g., edge, private or public cloud).
Reassess ecosystem and vendor selection processes	Running telco workloads in the public cloud	In response to customer demand and cost pressures, CSPs are now starting to run telco workloads in the public cloud. This has numerous benefits from a scale and compute cost perspective, but it also means that BSS must be programmable and agile enough to interwork with many more vendor solutions, including those of hyperscalers and smaller third-party developers. As a result, Heavy Reading believes that CSPs must now place greater emphasis on vendor and ecosystem partners' ability to support scalable and open public cloud services when selecting BSS partners.

Source: Heavy Reading

CONCLUSION

BSS must evolve to keep pace with CSPs' cloud transformation journey. Therefore, CSPs must create BSS transformation strategies that empower them to deploy cloud-enabled BSS successfully and gradually while maintaining existing services.

As this white paper documents, an effective transformation strategy must be based on open architectures that integrate automation, agility, and flexibility on both the design and implementation levels to address the challenges associated with managing legacy monolithic BSS implementations.