

Content Management as a Service—Financial Archive Cloud

David Lebutsch · Christian Bolik · Michael Hennecke

Received: 1 October 2010 / Accepted: 11 October 2010 / Published online: 6 November 2010
© Springer-Verlag 2010

Abstract As companies are sprinting towards offering enterprise services in the cloud, they are facing the decision of either developing a service from scratch specifically for a cloud infrastructure or taking their existing products and “cloudifying” them for a quicker route to market. In this article we are elaborating on how we addressed “cloudification” challenges during the development of a Content Management as a Service pilot.

1 Introduction

Market research [8] has shown that corporate information technology officers, especially from financial institutions, are focusing on the following three key areas: storage costs, document efficiency/end user experience, and compliance responsiveness. To address these requirements new services and delivery models are being invented such as cloud services.

Cloud services use dynamic infrastructure to offer a *new consumption and delivery model* inspired by consumer internet services which enables business imperatives and supports innovation. It is important to understand that “cloud” is not primarily focused on technology—its value proposition

is that it revolutionizes how business services are delivered and consumed: as a utility and not as an investment in technology.

Cloud computing can help the IT infrastructure we know today to become more dynamic and bring together business and information technology to create new possibilities. Such a possibility is Content Management as a Service which can be provisioned within minutes in the cloud instead of a month long implementation project. Recent improvements in leveraging virtualization [24], standardization and automation [21] enabled the development of such a service.

One of the key capabilities, required for offering a cloud service at an attractive price point, is multi tenancy. Multi Tenancy in the context of Content Management as a Service is the ability to host multiple private Content Management Cloud service instances on a shared platform [23, 29, 30]. Only very few established applications have the built-in ability to separate tenants within a single instance of the application. Cloudifying these existing applications requires to overcome the challenge of servicing multiple tenants on a shared platform and still being able to isolate their namespaces, user spaces and security.

We are going to elaborate on key elements of a Content Management as a Service solution as well as how we addressed challenges during the development of the Content Management as a Service pilot.

2 Financial Archive Cloud Services

The solution to the current CIO/CTO challenges [8] is a secured, managed service for client specific content stored in a virtual private cloud hosted in IBM’s data centers. The Archive Cloud Fig. 1 provides a secure, reliable and

D. Lebutsch (✉) · C. Bolik
IBM Deutschland Research & Development GmbH, Schönaicher
Str. 220, 71032 Böblingen, Germany
e-mail: lebutsch@de.ibm.com

C. Bolik
e-mail: bolik@de.ibm.com

M. Hennecke
IBM Deutschland GmbH, Karl-Arnold-Platz 1a, 40474
Düsseldorf, Germany
e-mail: hennecke@de.ibm.com

Fig. 1 IBM financial archive cloud

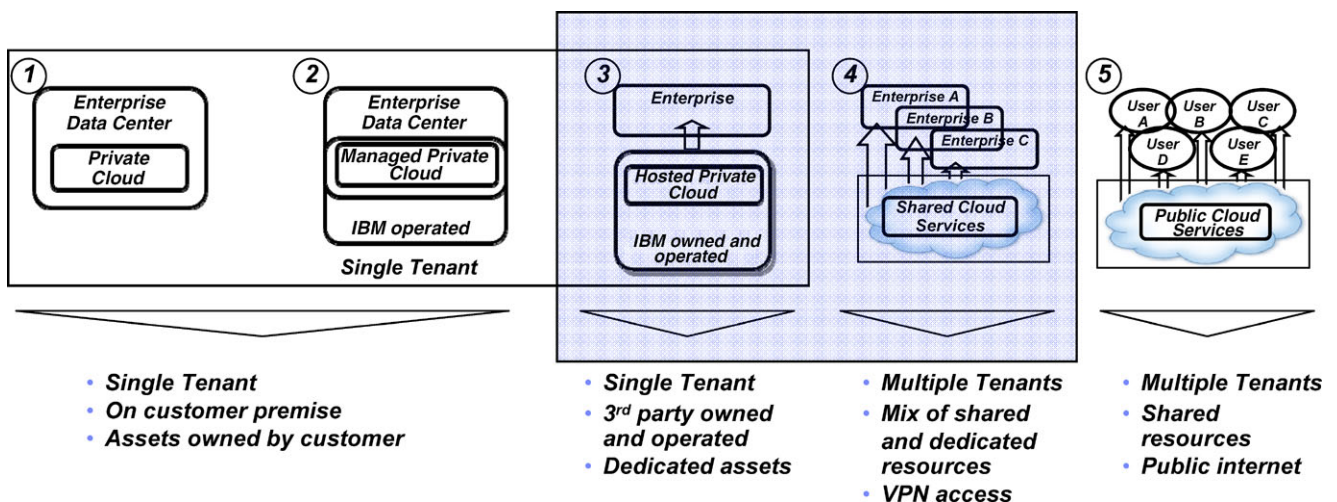
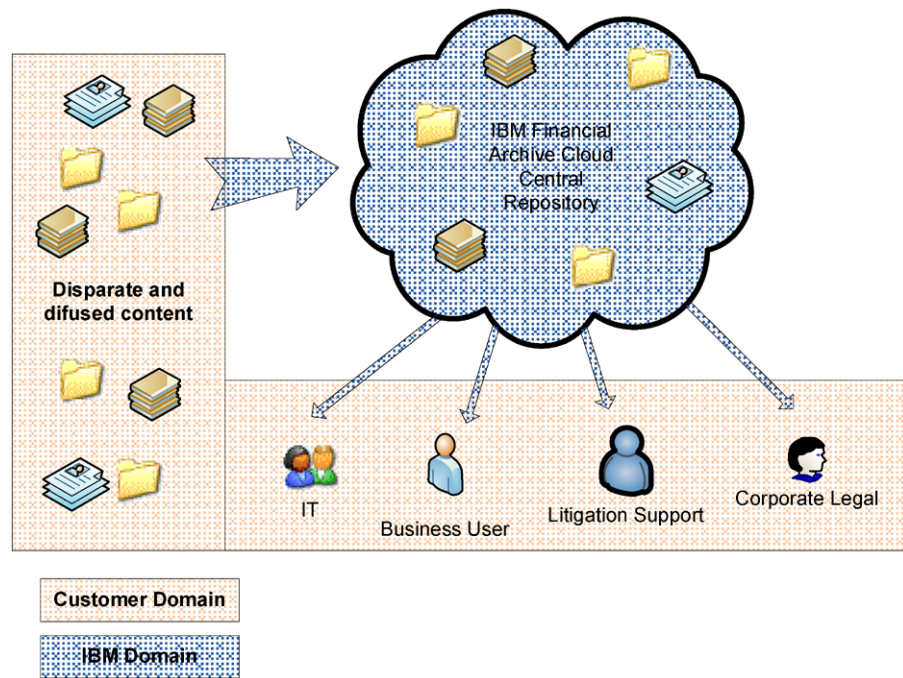


Fig. 2 Cloud service delivery models

fast archiving solution, with the ability to effectively index, search, retrieve and track client specific content in a digitized form. It delivers reduced overall archiving/retrieval total cost of ownership to the Financial Enterprise, ensures adherence to privacy and archiving regulation, enables information to become a core asset and is supported by comprehensive reporting on security and access of data.

The Archive Cloud in Fig. 1 depicts the separation of domains between the service provider, IBM, and the customer. Content is transferred to the cloud from the customer domain using data upload services provided by the cloud. Users with various roles can access the content in the cloud and retrieve it if required.

3 Delivery Models for Financial Enterprise Customers

While many commercial cloud offerings for computing and storage resources provide their services on the public internet it is still rare that enterprise customers accept that their enterprise records are kept on a system accessible from the public internet. The most security cautious customers will demand a cloud service where all data is kept on their premises while the equipment will be owned and operated by a cloud service provider (option 2 in Fig. 2). However, those customers who want the cost benefits of a shared cloud platform and the security of a private service will choose option 4 of the cloud service delivery models shown in Fig. 2.

The private cloud service on a shared platform ensures that only the customer can gain access to his personal service yet the computing and storage resources are shared to participate in the cost savings brought by the economies of scale of a shared platform.

Option 1 in Fig. 2 is a cloud delivery model where the customer owns and operates the data center, equipment and application. Option 3 in Fig. 2 would be a delivery model where the service provider dedicates all service components to a single customer. Option 5 in Fig. 2 describes a public shared service where users directly access the cloud service over the public internet.

4 Cloudifying IBM Content Management Applications

Figure 3 shows at a high level the software components of an Enterprise Content Management (ECM) solution, which essentially consists of the following components:

- A relational database for managing the meta data of content
- A fulltext index for searching content not managed by the database
- An object manager receiving and serving archived objects
- An administration and end user graphical user interface
- A special client for legal discovery

Decades ago IBM's ECM software stack was not designed with cloud services in mind, so in order to use it for the archive cloud service discussed here, some design requirements needed to be validated:

1. How to host multiple customers (tenants) in the cloud while keeping operational expenses low?
2. How to scale the private cloud service as well as the shared cloud platform?

3. How to secure customer data in the cloud and enable customers to meet legal and regulatory compliance requirements?
4. How to achieve very high Quality of Service and Service Level Agreements while keeping costs down?

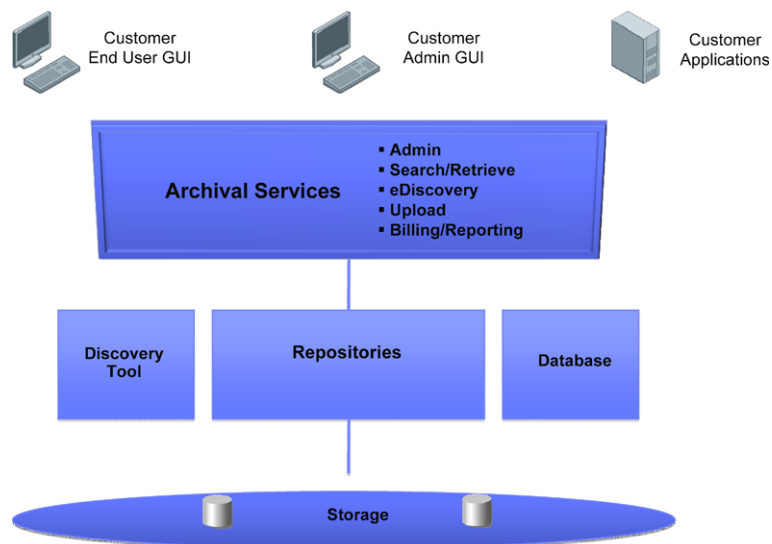
In the following we are discussing each of these four requirements in detail.

4.1 Multi Tenancy

Multi Tenancy in the context of the Archive Cloud is the ability to host multiple private Archive Cloud service instances on a shared platform [23, 29, 30]. An application which was designed to be offered as a cloud service has multi tenancy built into its core design and its data model, so that customers can cost effectively share the same application instance while their data remains secure and protected. Traditional applications for enterprise customers were designed to serve one tenant or customer, assuming a certain level of trust and a shared namespace between all users of an application. If a private namespace is required a new instance of the application would need to be instantiated which often results in a new database or even operating system instance. One can imagine that the scalability of operating and managing a cloud service is severely limited if every time a new customer signs up for an Archive Cloud service multiple operating system instances with their own complex application stack need to be provisioned and managed.

Figure 4 depicts a design which limits the operational complexity and costs while not requiring a complete rewrite of existing applications. The blue boxes depict operating system instances which we want to limit to an absolute minimum to reduce license and operations costs. These virtual machines or logical partitions live in a dynamic infrastruc-

Fig. 3 IBM enterprise content management products used in the Archive Cloud



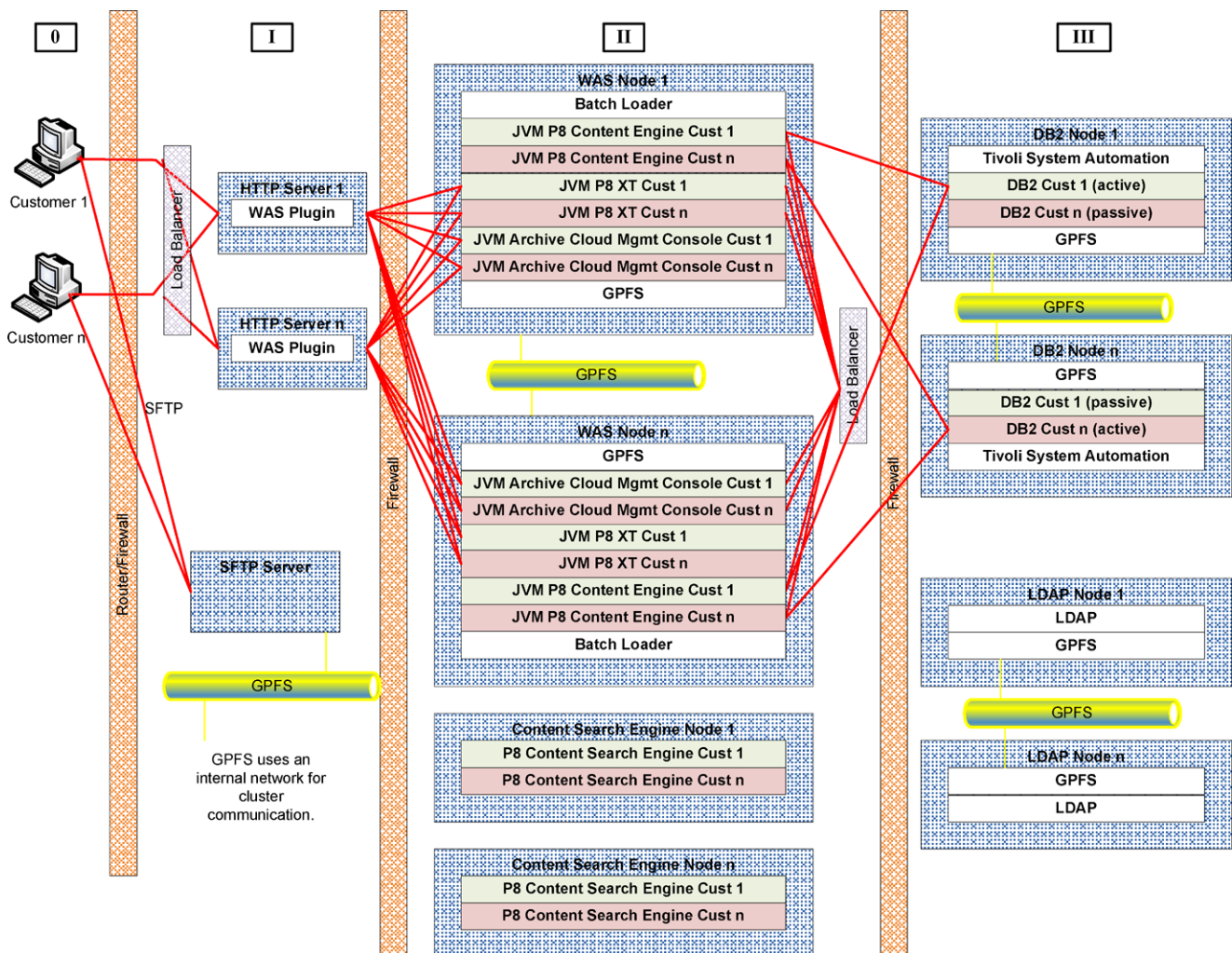


Fig. 4 Archive Cloud multi tenancy design

ture on a shared platform which provides zero downtime and on demand scalability.

Since the application itself was not developed for the cloud we still need private instances of some components: the J2EE applications and databases. The J2EE applications live on an IBM WebSphere Application Server cluster farm and the databases live on an IBM DB2 cluster sharing the same database manager instance. Shared components besides the operating systems are the HTTP Server farm and the Cloud LDAP server.

4.2 Scalability of the Archive Cloud (for a Single Customer and All Customers Combined)

The dynamic infrastructure platform allows a single operating system instance (Virtual Machine or Logical Partition) to *scale up* to 256 physical processor cores and 4 TB of memory. This is sufficient even for a large database server hosting many individual databases. Another scaling method

supported by the dynamic infrastructure is *scale out*; more Virtual Machines or Logical Partitions can be added to a cluster to increase computing resources.

In the Archive Cloud scenario the service can benefit from both, scale up and scale out. A more powerful individual Virtual Machine or Logical Partition can host more Java Virtual Machines running J2EE applications or databases. More Virtual Machines or Logical Partitions can be added to the HTTP Server Farm, IBM WebSphere Application Server or the IBM DB2 PureScale cluster to scale out the services across more instances to increase throughput and availability.

The important key word is dynamic. The dynamic shared infrastructure can only play out its strengths if the service is designed in a way to scale in every possible direction, e.g. bringing in more resources if needed or powering off resources which are temporarily not fully utilized.

The Archive Cloud design addresses all these requirements: cost effectively scaling to the largest single customer,

as well as cost effectively scaling to many medium sized customers and being able to utilize all advantages of a fully dynamic infrastructure.

4.3 Security in the Archive Cloud

No Enterprise service [3] offered in a cloud can be successful without ensuring security of the customer's data. Trusted Service and Trusted Data are the key elements.

The first security aspect is that the offered service is a trusted/private service. The customer connects the customer's network to the cloud service through a virtual private network (VPN) [2]. The service itself identifies the customer VPN connection and only allows access to components belonging to this particular customer. E.g. host names, ports and URLs are being translated and re-routed by the firewall to obfuscate the real addresses of the individual private service components.

Once the customer reaches the service the customer either provides a security token such as a SAML assertion (which includes authorization credentials), or the customer directory server has been replicated to the cloud for those users accessing the service. If the user account information has been replicated to the cloud directory server, the customer directory server would authenticate the user and send a single sign on security token to the cloud. In the cloud the user would then be authorized or denied access to the service using the cloud replica of the directory server.

Data in the cloud is also required to be trusted. Typically it is sufficient to encrypt all customer data at rest [2]. This means that customer data stored on persistent storage must be encrypted but data in memory or transferred from e.g. network card to memory can be in the clear. This is achieved by encrypting all data in the Archive Cloud service with an encryption key. This key is either managed by the customer or by IBM as the trusted key manager. Either way, administrators or other users would never be able to read the data in the clear. Data is also electronically signed by the customer and/or IBM to ensure over its lifetime that it has never been tampered with or modified. Using the digital signature and the public key of the entity which has signed the object, the originality of the object can always be proven.

4.4 Quality of Service and Costs

The Archive Cloud design eliminates any single point of failure while enabling scale up/scale out characteristics and keeping the costs per service instance at a minimum. The Archive Cloud components support clustering and rolling updates of all solution components. Combined with the continuous availability of the dynamic infrastructure, this provides unprecedented service availability. The relative costs per private service instance are much lower than a customer

hosted system because the dynamic infrastructure achieves a much higher average utilization. It is possible to achieve a higher grade of utilization on larger systems, and these scale effects reduce the relative costs for each customer. Another cost advantage is the operational excellence achieved by specializing operations teams to monitor and maintain a subset of the solution and by an investment in autonomic tooling [21] which benefits all consumers of the dynamic infrastructure.

5 Adopting the Archive Cloud

5.1 Subscribing to an Archive Cloud Service

So how would a customer (enterprise or SMB) subscribe to an Archive Cloud service as described here? The first step [5] is to identify the business requirements of the customer on content management, archiving, and records management. Then, those requirements are mapped to a set of Service Level Objectives (SLOs) [19] which represent measurable availability, response time, throughput and other characteristics that the service needs to achieve [9]. In a subsequent step the data model for the archive cloud needs to be identified, i.e. the metadata needed for different categories of documents and other data to be stored in the archive as well as retention requirements and associated policies. A hierarchical file plan [15] may be constructed which maps individual record categories to specific retention policies describing for how long documents filed into the respective categories should be retained and how to handle them once their retention period has expired.

5.2 Instantiating an Archive Cloud Service

Once the customer's requirements have been discussed between the customer and the archive cloud service provider, the service provider instantiates this customer's service instance [25], by making use of the shared infrastructure described above. This instantiation ideally occurs automatically [20], by mapping as many virtual machines, storage and network resources to the instance as required to achieve the specified SLOs. One promise of a cloud infrastructure is to elastically adapt to current work load put on each service instance, meaning to assign just enough resources as are currently required for the instance's SLOs to be met [5]. As the load on an instance increases, more resources are assigned dynamically, and are unassigned again or reassigned to other service instances as it decreases.

Once the required resources have been assigned and their initial configuration has been performed, the customer is sent a URL to an administrative portal running in the provider's data center which allows for further configuration of the service and for actually putting it to use.

5.3 Configuration and Customization of the Archive Cloud Service

Content management systems which provide additional capabilities such as records management and e-discovery consist of multiple software components. Typically, many of these software components have their own different user interfaces (UIs) for administration and use. This leads to an extended learning curve for administrators and users while they are trying to get acquainted with and master all the details of the different product UIs. The Archive Cloud service we are discussing here improves on this by offering a single administrative portal for use by the customer's administrator, which allows configuration, customization, and auditing of all higher-level (i.e. non-infrastructure-related) service components [28]. The client administrator uses this portal to make modifications to the service instance's data model described above, such as adding or removing metadata attributes or updating retention policies as regulations change. Through this capability the archive cloud service allows for customers to focus on their specific use cases and associated business requirements and customizing the archive cloud service to meet their specific needs.

Another aspect of configuration is the assignment of user roles to individual client users and/or groups. The archive cloud service supports role-based authorization for restricting operations a user is able to perform. The different client user roles provided so far are the following:

- Archive administrator: Is able to configure and customize the cloud service.
- Archive operator: A read-only role, allows for monitoring the cloud service.
- Archive user: No administrative permissions, but able to use the end user interface for storing and retrieving documents (access control lists apply).
- Legal user: Is able to perform e-discovery operations such as performing legal searches via the respective user interface.

5.4 Importing Data into the Archive Cloud

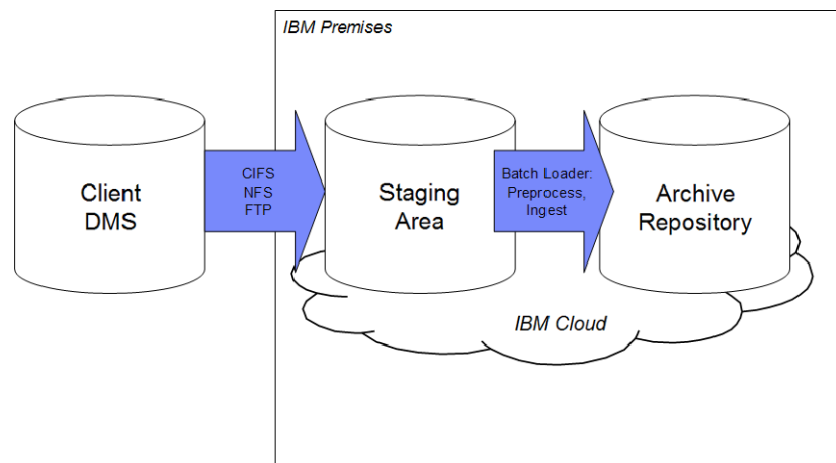
Most companies already have a content management system and most likely also a records management system in place on their premises today. When these companies want to take advantage of the cost savings, operational and administrative efficiencies offered by an Archive Cloud service, they require a way to import large amounts of data into the archive cloud efficiently. For this purpose, our archive cloud service provides a bulk import facility. The customer uploads document batch files to a staging area private to this customer inside the archive cloud, from where it is uploaded by a batch load facility running inside the archive cloud to the service's content repository. Figure 5 gives a graphical representation of this concept.

The document batch files mentioned above consist of a compressed file archive which includes an XML file. This XML file specifies, for each contained file, the associated document class and records category, and respective metadata values. The format of this XML file is compliant with the Electronic Discovery Reference Model (EDRM) schema [6]. Tooling will be provided for customers to quickly and easily create these XML files and associated document batch files.

The batch loader running inside the cloud instance imports documents from the staging area into the cloud's repository, filing them into the specified record categories and thus putting them under the control of the retention policy associated with each category. The batch loader may be configured via the administrative portal mentioned above. Multiple batch loader tasks may be configured, each one operating on a different path or names of batch files and on different schedules, as required. Also through the portal, the execution of batch loader jobs can be monitored and batch load run reports be retrieved.

In addition to the batch load facility, which allows for convenient bulk import of customer data into the archive

Fig. 5 The Archive Cloud's bulk import facility



cloud, an end user interface is provided which allows storing and retrieving individual documents into/from the cloud. Through this interface documents may also optionally be declared and filed as records.

5.5 Monitoring and Troubleshooting the Archive Cloud

Since costing of the Archive Cloud service is based on the amount of data stored in the archive and on the SLOs committed to by the Archive Cloud service provider, the customer has an acute interest in monitoring their current consumption of Archive Cloud storage space and achievement [18] of the committed SLOs. For this purpose the administrative portal provides a dashboard, allowing the client’s archive administrator and operator to monitor the associated key performance indicators (KPIs), such as storage space used, average response time, average throughput, etc. Historical reporting for these metrics will be provided as well, allowing the customer to identify trends in their use of the service.

While the Archive Cloud service relieves the client from having to trouble-shoot infrastructure-related issues, there still needs to be some amount of trouble-shooting capabilities provided for customers. For instance, a document batch file’s XML file specifying document classes may reference a document class which does not exist, causing the batch load job to fail. Here, the batch load report will specify exactly which document class could not be found.

Through these facilities the archive cloud enables companies making use of this service to always have an up-to-date view on their subscription costs and SLO attainment of their service instance, as well as to be able to quickly resolve any errors in their use of the service.

6 The Archive Cloud Storage Factory

Reducing the storage costs of Archive Cloud services is a key objective which needs to be addressed by the underlying dynamic infrastructure. As ECM solutions are typically I/O bound [22], this cost reduction has to be realized without negatively impacting performance. To enable the utility model of service delivery, it is also critical that the cloud storage is easy to deploy and maintain, and that it is robust and supports the various high availability mechanisms of the higher level components.

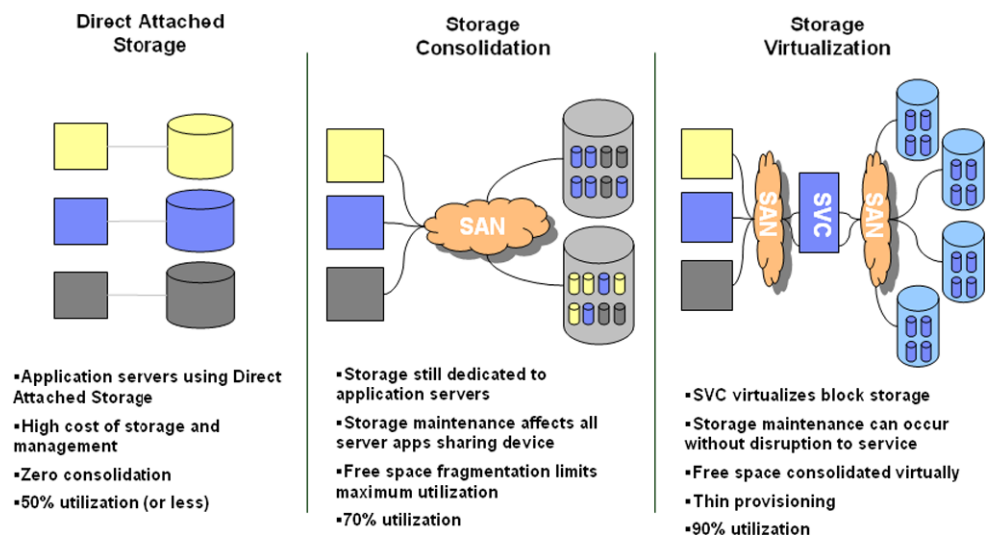
6.1 Block Storage Virtualization

The classical approach to achieve cost reductions in the IT infrastructure is to increase the utilization of the hardware through consolidation and virtualization.

Today, server consolidation and virtualization is business as usual, and methods for thin provisioning of virtual machines or logical partitions are well established using various hypervisor technologies. On the storage side, the IT industry has gone through a number of steps to increase the utilization of block-level storage, as depicted in Fig. 6:

- With local disks directly attached to different servers, there is no consolidation effect and relatively poor utilization of the disk storage hardware is the result. Storage management is inflexible: free space has to be provisioned on each server, and changes like moving a disk from one server to another are typically not possible without interrupting the service.
- Storage consolidation is achieved by replacing the directly attached physical local disks with logical disks (LUNs) in a Storage Area Network (SAN) attached external storage subsystem. A SAN offers switched point to point network connections much like Ethernet does

Fig. 6 Block storage consolidation and virtualization



but uses a lossless protocol (Fibre Channel Protocol) and has been optimized for storage workloads. The two most common end points on a SAN are a storage subsystem offering a LUN or logical disk and a server consuming the LUN or logical disk.

- Storage utilization is usually increased, although it is still limited by the fact that individual LUNs are dedicated to the different servers. Storage management is more flexible, but changes may still cause disruptions to the service.
- Products like the IBM System Storage SAN Volume Controller (SVC) [10, 12, 17] implement storage virtualization at the storage network layer. LUNs that are provided by the storage subsystems are not made directly accessible to the servers. Instead, virtual disks are exported from the SVC to the servers, and the mapping of virtual disks to LUNs on the underlying storage hardware can be transparently managed by the SVC without disrupting the service. This virtualization layer also allows thin provisioning of block-level storage, and generally results in much higher storage utilization.

However, for our purposes the block-level storage virtualization also has a number of limitations:

- While it is possible to map a virtual disk to multiple servers/partitions, SVC itself does not contain mechanisms to coordinate concurrent access to the virtual disks.
- No file system layer is provided. While most databases may directly use block storage devices, a file system is needed to store the archived objects and the full text index.
- The virtualization mechanism by which storage thin provisioning is implemented may cause performance degradation. While this may be acceptable for the archived objects, it will hurt the overall system performance when block-level thin provisioning is used for database storage or for the full text index (which are both heavily I/O bound).

So while block storage virtualization is useful, especially for manageability and to unify access to block storage residing on a number of heterogeneous storage subsystems, storage virtualization at the file system level is better suited for the Archive Cloud as discussed in the following section.

6.2 Storage Virtualization and the GPFS File System Layer

The alternative to storage virtualization at the block storage level is to present the storage to the application layer through a file system. In this case all the block storage is consolidated within one or more file systems. This raises the following questions: Which file system protocol should be used for optimum scalability, performance and manageability; and how will the multi-tenant requirements of the Archive Cloud services be addressed within the file system when it is not based on separate (virtual) block storage devices?

Traditional ECM solutions use tier 1 storage for databases and full text indexes (typically SAN attached Fibre Channel disks), and tier 3 storage for the archived objects (typically a NAS filer attached through a TCP/IP network). Storage tiers in this context characterize classes of storage: tier 1 being the fastest most expensive class of storage. Tier 3 is still on-line storage (as opposed to tape) but with low performance at low costs.

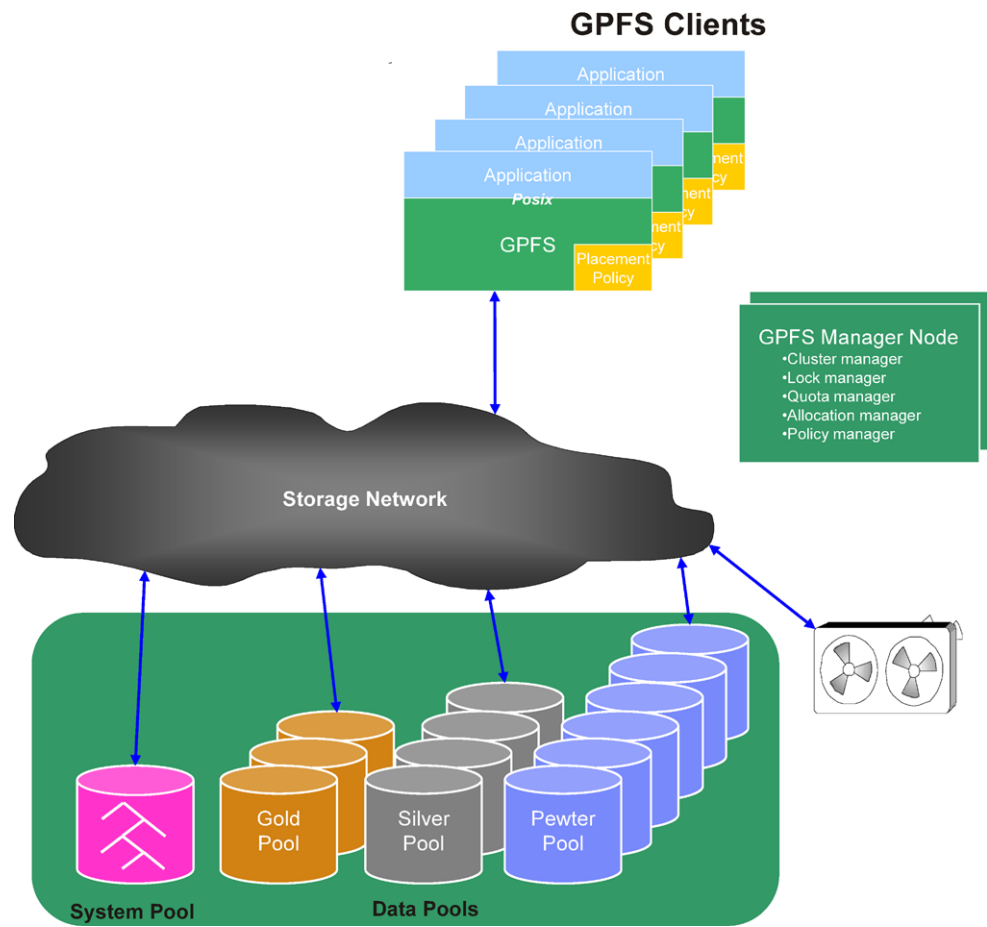
For large Archive Cloud solutions that are providing services to several large enterprise customers, a single monolithic NAS filer may not be sufficient to support the required number of files or I/O data rates. Deploying multiple NAS filers would add significant complexity, and typically also introduces multiple name spaces. In such cases the IBM Scale Out Network Attached Storage (SONAS) [13, 16, 26] could provide a scalable NAS solution with a single namespace and sufficient capacity and aggregate bandwidth. This solution uses the IBM General Parallel File System (GPFS™) [14, 27] internally, and exploits the GPFS wide striping capabilities to scale out across multiple midrange storage subsystems and achieve a large aggregate I/O performance.

However, neither the classical NAS filers nor SONAS are a good fit for the ECM tier 1 storage requirements from a performance perspective: As clients are connecting through a TCP/IP network using the NFS or CIFS protocols, the single-client performance is limited by this TCP/IP connection which is typically inferior to a direct Fibre Channel connection to block storage devices. For this reason, we are implementing the storage infrastructure for the Archive Cloud by using GPFS natively rather than in the back-end of a scale out NAS solution.

The General Parallel File system is a cluster file system with superior scalability and performance; it supports extremely large files as well as large numbers of files, scales to thousands of nodes, and provides consistency in fully parallel operation within a single shared namespace. High availability is ensured through multi-pathing to the physical storage, journaling and snapshots, and the option of additionally replicating file system data and/or metadata at the GPFS level. Figure 7 shows the GPFS SAN attached storage model, in which all GPFS nodes/partitions have a direct SAN connection to all LUNs which are used within GPFS.

Figure 7 also shows how the GPFS Information Lifecycle Management (ILM) features can be used to organize different types of LUNs into different GPFS *storage pools*. For example, the GPFS file system metadata (directory entries, inodes and indirect blocks) will always reside in the system pool which is typically implemented with Fibre Channel disks or SSD drives. Data pools can be used to hold different types of disks, or to distinguish between LUNs with different RAID levels. GPFS file placement and migration rules control which storage pool is used for which files within a file system.

Fig. 7 GPFS SAN attached access model and storage pools



In a cloud environment, not only the servers and the storage need to be virtualized: The physical Fibre Channel connections between the servers and storage need to be virtualized as well. In [11] we describe how to use the industry standard N_Port ID Virtualization (NPIV) to virtualize the FC adapters in the physical servers, and how to implement multiple isolated GPFS clusters over a shared set of physical Fibre Channel connections.

6.3 GPFS File System and Fileset Layout

Using GPFS, all the physical storage can be provided to the various service instances within a single global namespace, with concurrent access from all OS instances. This dramatically simplifies provisioning, and as the data is available on all nodes this also provides shared storage for the high availability functions of the higher software layers, without the need for an HA takeover of block storage devices in the case of node failures.

In addition to the performance advantage that is achieved by using virtualized Fibre Channel connections to the OS instances, using the native GPFS layer also enables us to fine-tune the file systems for different usage scenarios. Typically

in ECM solutions there are four different types of storage areas with different access patterns:

- Database tables with random read/write I/O, at a tunable database table extent size
- Database logs, which exhibit sequential write I/O
- Full text indexes, with highly random read I/O when using the index and both random and large sequential read and write I/O when creating or extending the index
- Archive objects, which are typically written once and are only very seldom read. These objects in principle could be staged to tape storage. However, to satisfy response time requirements for e-discovery it is advisable to keep them on disk storage.

The Archive Cloud deploys the underlying block storage into five different GPFS file systems (the four areas described above plus a small file system for internal utilities), as shown in the five vertical blocks of Fig. 8. The file systems are tuned at the GPFS level and at the storage subsystem and LUN level to best fit the above access patterns.

To provide support for multiple tenants within each of these file systems, the GPFS *fileset* mechanism is used. Filesets are a subset of the directory tree which can be managed very much like a completely separate file system. In particu-

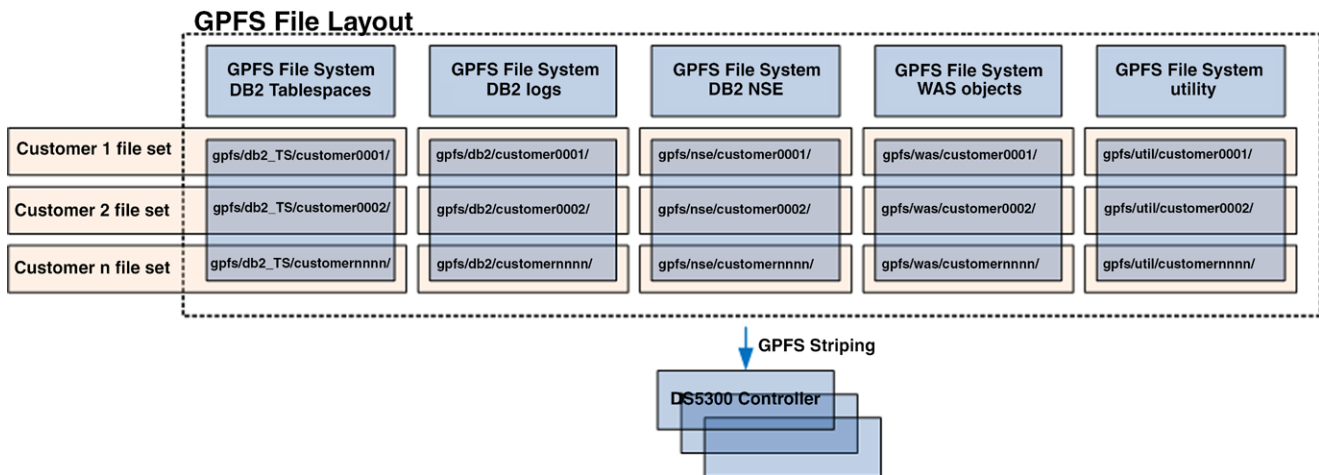


Fig. 8 GPFS file system and fileset layout for multi-tenant support

lar they can be mounted into the root fileset or at other places in the directory tree; their capacity can be controlled by file-set quotas, and it is possible to take per-fileset GPFS snapshots. The fileset dimension is shown in Fig. 8 in the three horizontal stripes. In the file system, they are manifested as per-customer directories. As all filesets within a file system share the free space of the file system, this avoids fragmentation of free space which would arise when using separate file systems instead of filesets within a file system.

So by using the GPFS SAN attached model with NPIV storage virtualization, we can provide excellent performance to each OS instance in the cloud. The GPFS robustness, scalability and advanced manageability and ILM features enable us to implement a file system and fileset architecture that easily scales to huge server and storage configurations and hundreds of tenants. Included in the life cycle management features (of the block storage devices, not to be confused with the document life cycle management which is performed at the higher software layers) is the capability of GPFS to *remove* disks from GPFS file systems while the data is online, as well as adding new storage to existing file systems. This is an essential manageability feature to ensure continuous operation within a single namespace for the many years that objects live in the Archive Cloud [4].

GPFS can also be used to implement disaster recovery. It supports replication at the GPFS layer, and multiple GPFS clusters can cross-mount GPFS file systems using the native GPFS protocol. So far this has been primarily used for coupling within the datacenter, but some organizations also use GPFS multi-cluster in a Wide Area Network (WAN) setup. Recent work on wide-area caching [1, 7] also makes this technology much more feasible on the WAN and can deal with disconnects in the network without disruption to service on either side of the connection.

7 Conclusion

With our Archive Cloud pilot we have demonstrated that current “state of the art” technology makes it possible to take an existing Content Management application and cost effectively offer it as a cloud service.

Internal research has shown that for a small Archive Cloud (<0.5 Petabytes) the operational people cost are the main cost factor followed by storage and software license costs. The more storage the Archive Cloud system manages the bigger the cost factor of storage becomes relative to other cost factors. Resolving the storage cost issue was our primary concern.

We have demonstrated that with our approach of the storage factory we can not only bring down storage costs by only requiring one class of slower but cheaper disks but also decrease operational costs by using only one file system and storage technology. GPFS enables us to address many solution requirements with a single file system instead of many different storage technologies. Because we are able to combine GPFS with other components of dynamic infrastructure such as virtualization and automation we are able to implement multi dimensional scalability as well as dynamically provision and de-provision resources. This elastic resource management is the main factor in reducing the storage and computing costs.

Multi tenancy was the second biggest cost factor. The challenge was to find a design which cost effectively allowed for multiple private services to live on a shared infrastructure. We knew that without solving the multi tenancy problem the Archive Cloud would not be possible—it would not have been possible to re-write an application which was the product of hundreds or even thousands of developer years. We also knew that if we would have required e.g. eight logical partitions running complex software for every customer we wouldn’t be able to contain operational complexity with

hundreds of customers. The model we have developed is a workable compromise for medium and large customers and ensures the lowest operational costs possible with today's technology. It remains to be seen if improvements in automation make this approach cost efficient even for very small customers.

The automation brought by the dynamic infrastructure allows us to elastically provision our Archive Cloud service and meet any customers SLOs. All of this results in much reduced operational costs and together with scale effects creates a business model where the service gets cheaper per unit the larger it grows.

For us the next step is to focus on further integration and consolidation of solution resources to drive down costs as well as offer completely new services on top of the Archive Cloud.

References

- Ananthanarayanan R, Eshel M, Haskin R, Naik M, Schmuck F, Tewari R (2008) Panache: a parallel WAN cache for clustered file systems. *SIGOPS Oper Syst Rev* 42:48–53
- Berger S et al (2009) Security for the cloud infrastructure: trusted virtual data center implementation. *IBM J Res Dev* 53(4). Paper 6
- Black J et al (2009) Architectures and technologies for the globally integrated enterprise. *IBM J Res Dev* 53(6). Paper 1
- Bradshaw PL et al (2008) Archive storage system design for long-term storage of massive amounts of data. *IBM J Res Dev* 52(4/5):379–388
- Breiter G, Behrendt M (2009) Life cycle and characteristics of services in the world of cloud computing. *IBM J Res Dev* 53(4). Paper 3
- EDRM Electronic discovery reference model. Online at <http://edrm.net/>
- Eshel M, Haskin R, Hildebrand D, Naik M, Schmuck F, Tewari R (2010) Panache: a parallel file system cache for global file access. In: *Proceedings of the 8th USENIX conference on file and storage technologies*. USENIX Association, Berkeley
- Gartner EXP (2009) Meeting the challenge: the 2009 CIO agenda; January 2009
- Gelb JP (1989) System managed storage. *IBM Syst J* 28(1)
- Glider JS, Fuente CF, Scales WJ (2003) The architecture of a SAN storage control system. *IBM Syst J* 42(2)
- Hennecke M, Lebutsch D, Schleipen S (2010) GPFS in the cloud: storage virtualization with NPIV on IBM system p and IBM system storage DS5300. *IBM Redpaper publication REDP-4682*. Online at <http://www.redbooks.ibm.com/abstracts/redp4682.html?Open>
- IBM Corporation IBM system storage SAN volume controller. Online at <http://www-03.ibm.com/systems/storage/software/virtualization/svc/>
- IBM Corporation IBM scale out network attached storage. Online at <http://www-03.ibm.com/systems/storage/network/sonas/>
- IBM Corporation IBM general parallel file system. Online at <http://www-03.ibm.com/systems/software/gpfs/>
- IBM Redbooks Quick reference: records management 101. Online at <http://www.redbooks.ibm.com/abstracts/tips0595.html>
- IBM scale out network attached storage architecture and implementation. *IBM Redbooks publication SG24-7875*. Online at <http://www.redbooks.ibm.com/abstracts/sg247875.html?Open>
- Implementing the IBM system storage SAN volume controller V5.1. *IBM Redbooks publication SG24-6423*. Online at <http://www.redbooks.ibm.com/abstracts/sg246423.html?Open>
- Isom PK et al (2010) Intelligent enterprise architecture. *IBM J Res Dev* 54(4). Paper 6
- ITIL IT infrastructure library, OGC. Online at <http://www.itil-officialsite.com/>
- Jin X et al (2009) Reinventing virtual appliances. *IBM J Res Dev* 53(4). Paper 7
- Kephart JO, Walsh WE (2004) An artificial intelligence perspective on autonomic computing policies. In: *Proceedings of the fifth IEEE international workshop on policies for distributed systems and networks (POLICY'04)*
- Lovelace M, Buchanan N, Cameron G, de Rezende F, Tarella J (2008) IBM enterprise content management and system storage solutions: working together. *IBM Redbooks publication SG24-7558*. Online at <http://www.redbooks.ibm.com/abstracts/sg247558.html?Open>
- Mega C, Krebs K, Wagner F, Ritter N, Mitschang B (2008) Content-Management-Systeme der nächsten Generation. In: *Keuper F, Neumann F (eds) Wissens- und Informationsmanagement*, pp 539–567. ISBN: 978-3-8349-0937-4
- Molloy C, Iqbal M (2010) Improving data-center efficiency for a smarter planet. *IBM J Res Dev* 54(4)
- Naghshineh M et al (2009) IBM research division cloud computing initiative. *IBM J Res Dev* 53(4). Paper 1
- Oehme S, Deicke J, Akelbein J-P, Sahlberg R, Tridgell A, Haskin RL (2008) IBM scale out file services: reinventing network-attached storage. *IBM J Res Dev* 52(4–5):319–328
- Schmuck F, Haskin R (2002) GPFS: a shared-disk file system for large computing clusters. In: *Proceedings of the first USENIX conference on file and storage technologies*, Monterey, CA, January 28–30, pp 231–244
- Tagg BS (2009) The IBM services connection: service delivery through self-service portal technologies. *IBM J Res Dev* 53(6). Paper 3
- Wagner F, Krebs K, Mega C, Mitschang B, Ritter N (2008) Towards the design of a scalable email archiving and discovery solution. In: *Proceedings of the 12th East-European conference on advances in databases and information systems*, Pori 2008. pp 305–320
- Wagner F, Krebs K, Mega C, Mitschang B, Ritter N (2008) Email archiving and discovery as a server. In: *Intelligent distributed computing, systems and applications; Proceedings of the 2nd international symposium on intelligent distributed computing*, Catania 2008, pp 197–206



David Lebutsch is a ‘The Open Group’ Master Certified IT Specialist and Enterprise Content Management (ECM) Solution Architect with 12 years experience in designing complex Content Management and compliance solutions for large global companies. Mr. Lebutsch is a leader in the ECM Architecture & Solutions team of the IBM Research and Development Lab Germany. He is responsible for the design and implementation of first of their kind and complex ECM solutions. Most recently he has been focussing on the development of IBM’s ECM cloud offerings. Before joining the IBM Germany Research and Development Lab he worked for IBM SWG Lab Services in the USA and Austria.



Christian Bolik is an Advisory Software Engineer and IT Architect working with IBM since 1996. Previously he was the lead architect for the Management Console of IBM Information Archive, a combined software/hardware appliance for general purpose archiving. In his current role Mr. Bolik leads the user interface development team for the Content Management as a Service offering described here. Mr. Bolik holds a Dipl. Ing. degree in Computer Systems Engineering from the University of Hanover, Germany.



Michael Hennecke is a ‘The Open Group’ Master Certified IT Specialist and HPC Systems Architect in the Deep Computing technical sales team at IBM Germany. Mr. Hennecke has 19 years of experience in architecting and implementing High Performance Computing (HPC) and high performance storage solutions. He is currently focusing on deployments of the IBM General Parallel File System (GPFS) in non-HPC usage areas which can benefit from the GPFS performance, scalability, availability and manageability features.