RIDING THE CLOUDS - BEST PRACTICES IN DATA CENTER TRANSFORMATION, INTO THE NEXT CENTURY AND BEYOND

EMC Proven Professional Knowledge Sharing 2011

Paul Brant
EMC Corporation
Paul.Brant@emc.com

Table of Contents

Table of Figures **10**

Table of Equations **12**

Abstract **13**

Introduction **15**

The need for Preservation in the Cloud/Next Generation Data Center 17

The ascent of the Computer in the Cloud/Next Generation Data Center 18

The ascent of Data Longevity in the Cloud/Next Generation Data Center 19

The ascent of Virtualization in the Cloud/Next Generation Data Center 21

The ascent of the Appliance in the Cloud/Next-Generation Data Center 25

The ascent of the Intercloud – AKA Interconnected Global Connectivity 26

The ascent of the Personal Data Store in the Cloud/Next-Generation Data Center 27

The Personal Information Management Ecosystem 31

The Drivers for the Next-Generation Data Center 32

Attributes and Technologies **35**

Massively Parallel Processing (MPP) and other architectures 37

Best Practice – Understand advantages and trade-offs of scalable systems in the NGDC 37

SMP Architectures 38

Clustered Architecture Systems 39

Best Practice – Understand Advantages in MPP (massively parallel processing) in addressing the NGDC 41

Best Practice – Understand Amdahl's and Gustafin's Laws as it relates to MPP to achieve NGDC architectures 42

Applianceization 47

Best Practice – Consider Applianceization in Big Data Architectures 48

Best Practice – Utilize Applianceization in the Financial Sector 49

Best Practice – For NGDC DBMS, consider utilizing appliances 51

Best Practice – Implement Applianceization within the infrastructure – VBLOCK Technology and Architectures 52
Best Practice – Implement a Unified Management solution for Appliance Architectures 53
Best Practice – Consider utilizing appliance structures in the cloud (IT in a box) 55
Data Transference 57
Best Practice – Implement Directory-based Cache Coherency to create efficient local and remote distributed Data Transference 59
Best Practice – Implement Directory-Based Cache Coherency for Data Federation 63
Object Affinity 65
Best Practice – Standardize Cloud Computing Interfaces that is object-based and vendor-neutral 66
Best Practice – Implement an Object-Based Open Cloud Computing Interface into NGDC designs 66
Security 70
Best Practice – Implement Identity assurance into the Personal Data Store Architecture 70
Best Practice – Implement PDS to enable the Cloud in an ever-changing social media world 71
Best Practice – Implement Federated login and password for Personal Data Stores 72
Best Practice – Embrace Identity methodologies to Enhance Security 73
Best Practice – Consider Containerization to determine who you should trust with your data 74
Virtualization 75
Best Practice – Consider Virtualization considerations when implementing a Cloud or Intercloud 75
Best Practice – Implement the NGDC with VM metadata abstraction considerations using OVF 76
Best Practice – Consider Client and Server Virtualization into the NGDC 77
Best Practice – Utilize Virtualization to address cost reduction and cost avoidance in the NGDC 78
Best Practice – Understand the benefits of Virtualized Service-Based Delivery 79
Personal Data Store Attributes 82

Best Practice – Implement Personal Data Stores into the NGDC architecture 82
Best Practice – Implement additional Data Store flows to the existing lexicon 87
Best Practice – Implement NGDC's to address the needs of Personal Data Stores as a service to the individual. 89
Archival Ecosystem 94
What is an Archive 94
Best Practice – Implement OAIS in the Next Generation Data Center Archive Architectures 97
Best Practice – Utilize Archival strategies for digital content preservation by leveraging the knowledge of the archival profession 101
Best Practice – Implement the Self-contained Information Retention Format (SIRF) for long-term retention in the NGDC 102
Best Practice – Integrate SIRF and OAIS into the NGDC 106
Best Practice – Integrate SIRF and XAM into the NGDC 109
Best Practice – Integrate SIRF and JHOVE into the NGDC 110
Best Practice – Integrate SIRF and Bagit into the NGDC 111
Best Practice – Next-Generation Data Center containerization should address specific Media and Platform requirements 111
Storage Ecosystem 114
Storage Object Affinity 114
Storage for Cloud Computing 114
Best Practice – Implement an Object-based Data Management Infrastructure to Address Interoperability and Transparency Requirements 115
Best Practice – Implement CDMI Metadata Properties Consistent with the CDMI Specification 117
Data Portability 117
Best Practice – Implement Data Portability techniques into the NGDC architecture 118
Storage Tiering 119
Best Practice – Consider Application requirements for Auto Tiering 119
Sustainable Reference Models 123

Information Empowerment 124
Historical Affinity 127
Best Practice – Define Long-Term Retention Solutions 129
Longevity 130
Best Practice – Data center retention processes need to be in place to address long-term retention of information. 130
Best Practice – Implement containerization into the Next-Generation Data Center's information management processes 131
Reliability and Resiliency 132
Best Practice – Consider long term preservation and the threats in the Next Generation Data Center 133
Best Practice – Consider "Bit Rot' in the Next Generation Data Center 135
Best Practice – Consider "Software or Format" longevity in the Next Generation Data Center 136
Best Practice – Consider encryption for long term retention in the Next Generation Data Center 136
Best Practice – Consider Exit Strategies in the long-term preservation architecture of the NGDC 137
Best Practice – Consider fault visibility as it relates to riding the cloud for the next 100 years 138
Best practice – Consider that Replication may not be the Holy Grail to data loss.. 139
Best Practice – Model data loss to achieve Reliability and Resiliency in the NGDC140
Best Practice – Consider Visible vs. Latent Faults to achieve Reliability and Resiliency in the NGDC 141
Assumptions in the best practice model 142
Best Practice – Implement Data Scrubbing of active archives 145
Best Practice – Implement Latent and Visible fault detection to address long-term data retention 145
Best Practice – Increase MV and ML to increase Reliability and Resiliency 146
Best Practice – Reduce MDL to increase Reliability and Resiliency 147
Best Practice – Reduce MRL or MRV to increase Reliability and Resiliency 147
Best Practice – Consideration and increase in using replication increasing reliability148
Best Practice – Increase independence of storage-based systems 148

Green Stability 151
Best Practice – Implement Tiered Storage with Balancing Application Response Times with PCFE in mind 153
Best Practice – Utilize Tiering Utilities to Achieve Efficient Use of Storage 154
Business and Vertical Use Case 160
Data Warehouse – Big Data 162
Best Practice – Understand business drivers and how DW will change process requirements 165
Data Management Best Practices 167
Best Practice – Implement Master Data Management and Data Quality in the NGDC or in the cloud 167
Best Practice – implement In-Memory Processing and 64-Bit Computing 168
Best Practice – Implement Open Source Software in NGDC Data Warehouse Designs 169
Best Practice – Consider implementing a “Hadoop” or Equivalent Framework in your DW Design 170
Best Practice – Implement Advanced Analytics Methodologies 170
Best Practice – Implement Data Warehouse Appliances and Similar Platforms into the Next Generation Data Center 172
Best Practice – Consider Real World Expectations in the NGDC Data Warehouse 172
Best Practice – Implement “Parallel Everywhere” into a NGDC Data Warehouse Architecture 174
Best Practice – Consider Data Partitioning as part of the NGDC Data Warehouse Designs 176
Range Partitioning 176
Round-Robin Partitioning 176
Hash Partitioning 177
Data Warehouse Appliance Vendor solution Summary 178
Archival Business Drivers in the NGDC 182
Best Practice – Understanding and Selecting a Best of Breed Migration Strategy 183
Archive Use Cases 184

Best Practice – Support for Standard Interfaces, e.g. NFS, CIFS, XAM, which are agnostic to media, platform, or vendor.....186
Best Practice – Implement a Service-Oriented Architecture for Automatic Migration193
Vertical Markets and Use Cases for Riding the Cloud.....201
Best Practice – Implement NGDC's to conform to an "Open Cloud" methodology 201
Best Practice – Consider Test and Development as a First Step into the Cloud ... 201
Web Services in the Cloud 204
Storage in the Cloud.....205
Best Practice – Implement General Content Storage with Synchronization to/from the cloud. 206
Backup to the NGDC/Cloud.....206
Best Practice – Implement Data Sharing, Native Software and the Intercloud for Server implementations 207
Best Practice – Consider Preservation Options in the Cloud.....208
Best Practice – Implement RDF into a Cloud Preservation Framework 209
Best Practice – Consider Design Objectives for Databases in the Cloud 209
Infrastructure as a Service (IaaS) for storage in the cloud 211
Content Distribution (Propagating Data Geographically).....212
Cloud Storage Peering (i.e. "Intercloud" Storage) 212
The Inter-cloud - Interconnected Global Connectivity.....214
Best Practice – Implement an "Intercloud" Strategy in Ones NGDC and/or Cloud Design 214
Best Practice - Federate Cloud Computing Environments that Facilitates Just-in-time, Opportunistic, and Scalable Provisioning.....216
Best Practice – Consider Storage Interoperability and Federation within the NGDC and Intercloud.....219
Best Practice – Conform to a Standard Protocols Profile 220
Best Practice – Understand Mobility Aspects of VMs in the Cloud in the IP Address Domain 223
Best Practice – Implement Location Identity Separation Protocol in the NGDC Intercloud Design 225
Best Practice – Consider Naming, Identity and Trust in the Creation of Intercloud Resources 226

Best Practice – Consider Presence and Messaging in the Creation of Intercloud Resources 227
Best Practice – Consider Multicast IP Architectures for Video and Audio Delivery 227
Best Practice – Consider Time Synchronization for NGDC and Intercloud implementations 228
Best Practice – Consider XMPP to Create a Dependable Application Transport Model 229
Conclusion.....230
Appendix A – Cloud Definitions and Taxonomy.....231
Appendix B – Next-Generation Data Center of the future challenges 236
Appendix C – References.....239
Author's Biography.....241
Index.....242

Table of Figures

Figure 1 The Ever Expanding Digital Universe 15
Figure 2 IT transformation Enablers 16
Figure 3 Cloud Computing Taking the Lead in IT Mindshare 18
Figure 4 Next-Generation Data Center Drivers 33
Figure 5 Top Level NGDC (Next-Generation Data Center) Ontology 34
Figure 6 Attributes and Technologies 36
Figure 7 MPP Architectures 38
Figure 8 SMP (Symmetric Multi Processing) Architecture 39
Figure 9 Cluster Architecture 40
Figure 10 Graph of Amdahl's Law 43
Figure 11 Graph of Gustafson's Law 44
Figure 12 Vblock Architectural Diagram 53
Figure 13 Ionix Unified Infrastructure Manager 54
Figure 14 IT as a Service Appliance 56
Figure 15 Centralized information Cache Coherence Model 58
Figure 16 Distributed information Cache Coherence Model 59
Figure 17 Distributed Cache Coherence Architecture 61
Figure 18 Distributed Cache Directory Coherence Architecture 62
Figure 19 EMC VPLEX: The World's First Local and Distributed Storage Federation Platform .64
Figure 20 Object-Based Cloud Computing API 67
Figure 21 Components of an OCCI URI 68
Figure 22 Object-Based Linked Resource Operations Example 68
Figure 23 Data Store Taxonomy Hierarchy and Data Flow 85
Figure 24 Best Practice Data Store Taxonomy Hierarchy and Data Flow 87
Figure 25 EMC Centera Archive Ecosystem 96
Figure 26 OAIS Consumption and Archive Model 98
Figure 27 OAIS Object Package 99
Figure 28 SIRF Container Components 103
Figure 29 OAIS Functional Model 106
Figure 30 OAIS AIP Logical Structure 109
Figure 31 CDMI Data Storage Interface 116
Figure 32 Database Tiering Data Flow 121

Figure 33 Sustainable Reference Models....123
Figure 34 Types of replica faults....141
Figure 35 Spatial overlap of faults....143
Figure 36 The Sustainable Data Footprint....152
Figure 37 Tier Advisor Analysis Data Flow....155
Figure 38 Tier Advisor Input Data....157
Figure 39 Tier Advisor Optimization Analysis....158
Figure 40 Business Vertical and NGDC Use Case Taxonomy....160
Figure 41 Vertical Market Taxonomy....161
Figure 42 Data Warehouse Architectures....164
Figure 43 Greenplum Data Computing Appliance Architecture....175
Figure 44 Preservation System Entities....185
Figure 45 Ingest and Access with Same Application....186
Figure 46 Ingest and Access with Different Applications....187
Figure 47 Ingest and Access with Different Preservation Services....187
Figure 48 Storage Format Change....188
Figure 49 Architecture for SOA Capable Archiving of Digital Data Preservation....195
Figure 50 Use Case – Cloud Test and Development....202
Figure 51 Use Case - Web Service....204
Figure 52 Use Case – Cloud Storage....205
Figure 53 Use Case – Backup and Archive....207
Figure 54 Use Case – Functional Offload....210
Figure 55 Use Case – Cloud Service Augmentation....211
Figure 56 Use Case – Storage as a Service....212
Figure 57 The Intercloud....214
Figure 58 Federated Network Mediated by Cloud Exchange....217
Figure 59 Intercloud Top Level Standards Taxonomy....221
Figure 60 Intercloud Sub Attributes Taxonomy Page A....222
Figure 61 Intercloud Sub Attributes Taxonomy Page B....223

Table of Equations

Equation 1 – Amdahl's General Law of Speedup of MPP Architectures 42
Equation 2 – Amdahl's Parallel Law of Speedup of MPP Architectures 43
Equation 3 – Amadahl's Parrallel Law of Speedup of MPP Architectures 45
Equation 4 – Probability of a fault as a function of time 142
Equation 5 – Probability of a fault as a function of time approximation 142

Disclaimer: The views, processes or methodologies published in this article are those of the author. They do not necessarily reflect EMC Corporation's views, processes or methodologies.

Abstract

There is a Digital Crisis. The digital information universe is growing at an exponential pace. Some consider it a perpetual tsunami. How do you find information? How do you handle unstructured data? How do you add structure? What are containers and how does this concept address these issues? How do you handle object affinity, distribution of assets and delve into the Storage Ecosystem, to name a few? How do we handle these issues and what will the next generation data center look like? This crisis requires that we re-think established truths and transform business and technical processes as soon as possible.

For example, there are large and even small businesses faced with retaining and preserving huge amounts of digital information for very long periods and are at the front edge of a troubling crisis. Digital information is actually easier to lose than if it were on paper or film. It is one thing to manage a domain of digital records that an archivist can personally guard and shepherd, but it is quite another to meet the archival challenges of today's enterprise data center.

These data centers can be characterized as environments with petabytes of distributed information; high data growth rates, many facilities and many departments with uncoordinated responsibilities and requirements, and lack of business-level budget, interest, and focus on its archives as well as recently created information. All these operating challenges are now compounded by high risk. Yes, risk. Risk of failure and fines from legal discovery, compliance requirements, or security threats. Add to this, the risk of losing information that may be of great value to the businesses and the picture looks daunting.

The digital crisis is exacerbated by time. In 10 years, 50 years, 200 years, which applications will still be around? What computer and storage system will be able to read old information, providing that it be not corrupted by then? Even finding a single piece of content and all the linked-objects that contain associated content amid trillions of distributed information objects is at best, a costly and complex adventure. The problems are huge and here is the dilemma. Many standards and best practices exist today documenting the practices of managing, creating, utilizing and preserving digital information. Yet, none of them addresses the core problems caused by inadequacies and inefficiencies in the supporting information storage infrastructure.

There is good news though. With the advent of new business, process and technology practices, there is hope. The private and public cloud is a burgeoning approach foreshadowing

what the next generation data center will look like. Virtualization, compliance, security, long-term retention, geographic transparency and global reach of computing resources and many more aspects to this challenge will be discussed, to try to find a way to grasp these issues and solve them.

Today's data centers and IT infrastructures face unprecedented challenges. Many are experiencing a capacity crisis as they reach the limits of older facilities and legacy, and silo's of applications infrastructures. Space is tight. Technology modernization is overdue. Energy costs are still high. As a result, many companies still spend up to 60 to 70 percent of their IT budgets on operations and maintenance, instead of innovation.

Meanwhile, your customers are demanding wider access to information, transactions and services. Pressure may be growing to evolve your IT businesses from cost center to strategic business enabler, providing a true service-based infrastructure. Moreover, you must address all of this in the face of an uncertain near term, not to mention an extended future.

What are the best practices to thrive with unpredictability and exponential information growth? With a more elastic, resilient, optimized, next-generation data centers built on attributes discussed, you will be ready for whatever comes next.

In summary, this article will describe how to ride the "cloud", how to utilize what this technology can afford, offering Best Practices that will align with the most important goal, creating a next generation Data Center, addressing the business challenges of today and tomorrow, through data business and technology transformation.

Introduction

Over the last 50 years, the cost of computing information processing has dropped a billion-fold. The first hard drive was the size of a large cupboard and held a trivial amount of data by today's standards, just 5MB. By 1979, a hard drive capable of storing 250MB of data would fill a large supermarket cart. By 2007, hard drives had reached the size of one TB (terabyte, i.e. 1,000 GB). Only two years later, the first hard drive with 2 TB of storage arrived; while it took 51 years to reach the first terabyte, it took just two years to reach the second.

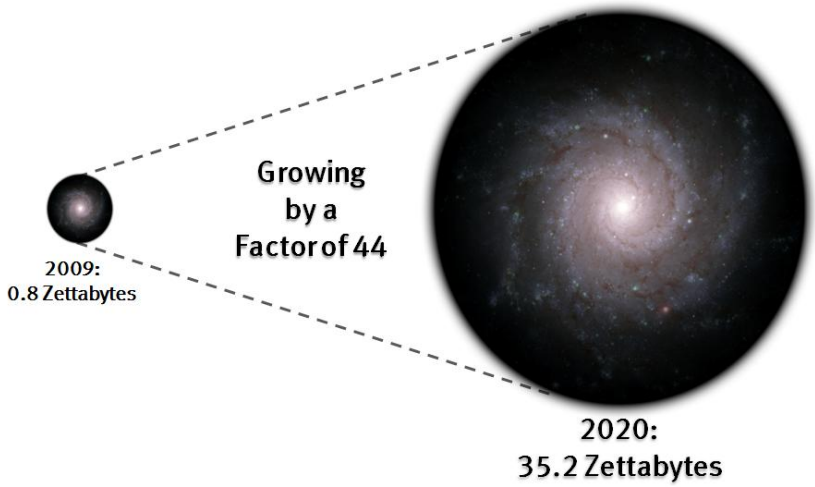


Figure 1 The Ever Expanding Digital Universe

If prices of motor vehicles had fallen as fast, you would be able to buy 4000 high-end Mercedes for just $1. This explosive growth of storage is just one example of the accelerated exponential growth of the IT world. As shown in Figure 1 above, since 2009, the Earth's digital footprint has extended by a factor of 44 times.

According to IDC, over the next ten years, storage capacity will increase sixty-seven times (67x). Additionally, IT staff will grow 1.4 times over the same period. What this tells us is this is just one example of the reason why the way IT does business must change and with it, the knowledge, skills, and tasks that occur every day have to change drastically.

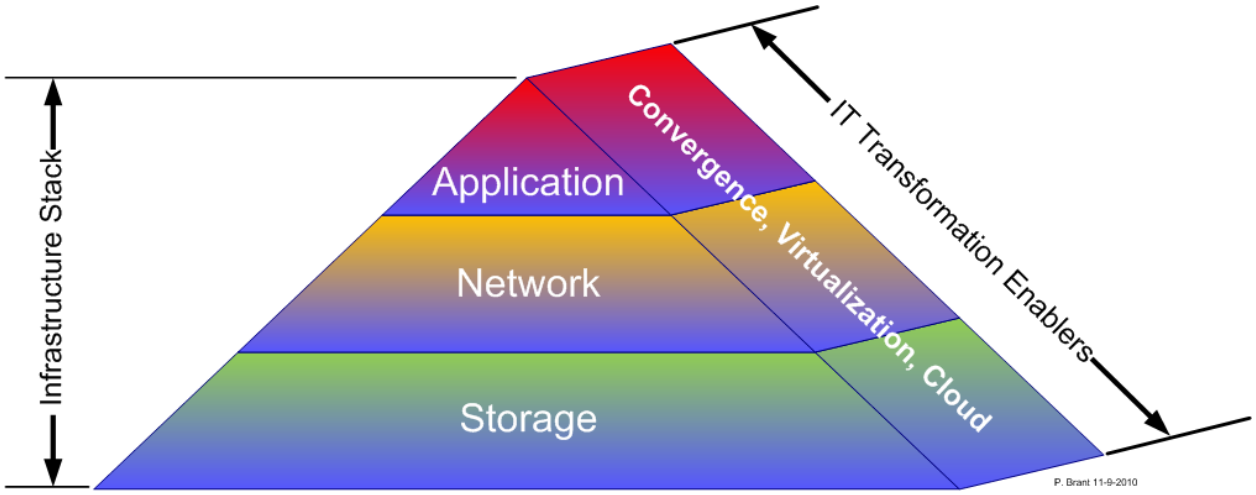


Figure 2 IT transformation Enablers

As shown in Figure 2, the infrastructure stack as we know it today is changing. Today, the focal points of applications (servers), network infrastructures, and storage are being transformed as it is related to the next generation data center. Servers that run the applications, as well as the communication medium and where the information lives is converging, virtualizing, and becoming cloud-enabled. Discussions on appliances and "Block" infrastructure architectures, to name a few, will be enumerated in the sections to follow.

For practitioners like us, this transformation offers challenges and rewards. Being EMC Proven™ is one way for all of us to gain the experience, tools and, most of all, knowledge to transform not just the data center, cloud, and infrastructure in general, but ourselves.

New IT certification programs now available through EMC Education Services—which many consider the most comprehensive of their kind—will help companies, as well as individuals, achieve the benefits of virtualization, converged infrastructure, and cloud computing, as the evolution of data centers occur for increased business efficiency.

To offer the proper skill set, EMC has taken on a substantial new education offering to address this data center transformation[1]. They include:

- EMC Cloud Architect and Data Center Architect Training and Certification
- EMC Cloud Architect (EMCCA) Virtualized Infrastructure certification

[1] https://education.emc.com/portal/

These offerings provide information architects, designers, and consultants with the knowledge and skills to address the convergence and management of storage, servers, and networking environments that are critical to building virtualized data centers and cloud infrastructures.

With all the growth in IT infrastructure and the stretching of people, process, and technology, how does one comprehend the next 100 to 1000 years with all its changes? How will this relate to the Next Generation Data Center and beyond? Let us begin.

The need for Preservation in the Cloud/Next Generation Data Center

As we ride the Clouds and as the volume of digital information continues to grow, a paradox becomes apparent. Over the years, humans can read and interpret the Dead Sea scrolls written almost 2000 years ago, but we cannot do the same with data generated 20 years ago on a 5.25-inch floppy disk. I am trying to remember when I actually saw a floppy disk last. Ironically, as the world becomes digital, it seems that we may be entering a digital "Dark Ages", in which business, public, and personal assets are in an ever greater danger of being lost. Please refer to the section titled “Best Practice – Implement containerization into the Next-Generation Data Center’s information management processes, on page 131 , which discussed more on the digital Dark Age.

To make matters worse, there is an increased need for long-lived digital information. Recent compliance legislation, such as HIPAA and the Sarbanes-Oxley Act, which require long-term data viability, have increased the need to study how to preserve myriad type of information, such as scientific, financial, healthcare, artistic, and cultural data for tens and even hundreds of years.

Preserving information is more than just storing bits of data. It involves preserving the understandability and usability of complex interrelated objects, even when technologies for computer hardware, operating systems, data management products, and applications are replaced with newer ones. Adding to the complexity, as data consumers change in demographics and lifestyles, moving into the social media stages of the Internet, continued changes in how data is used is also changing at an exceedingly rapid pace. This introduces new requirements to ensure long-term access and understandability, while enabling new interpretations of the same data.

It can be argued that at the heart of any solution to the preservation problem resides a storage component, which is the permanent location of the information. Traditional archival storage considers only bit preservation, if it considers preservation issues at all. One can argue that to take preservation considerations into account, a new type of storage structure and storage methodologies must emerge, be it local or in the cloud. In addition, one must also take into consideration how to better preserve data and deal with the "understandability" of data for long periods. The solution to this dilemma will be discussed, and a solution will be offered in detail in the coming chapters.

The ascent of the Computer in the Cloud/Next Generation Data Center

The rise of the computer (and everything it makes possible) has transformed our society, work, leisure, products, services and one could argue, the economy itself. Within this transformation, we can see some distinct phases. In the first phase, before 2000, information-processing power was embedded into products that were sold the normal way: personal computers, video games, cars, cookers, cameras, phones, and so on. This first phase will never end. Products will always get smarter, but the second phase took a new and different direction. Between 2000 and 2010, information-driven Internet services exploded onto the stage: e-commerce (Amazon, e-Bay), search (Google), price comparison sites, blogging, social networking (Facebook), and so on. The big shift had begun, albeit into the cloud as well. Individuals were beginning to use information as a tool in their own hands, to pursue their own purposes.

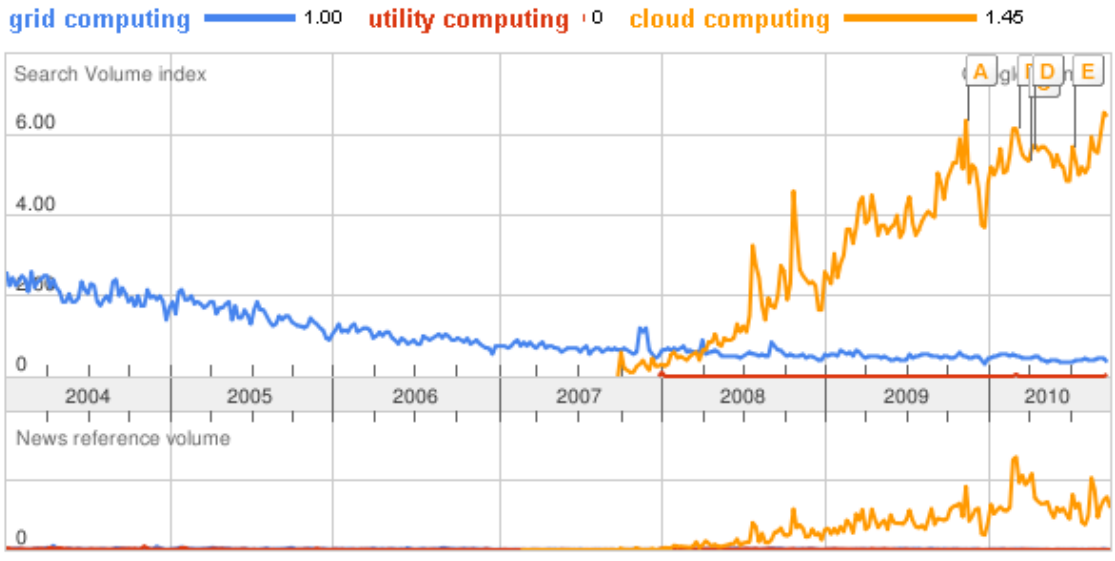


Figure 3 Cloud Computing Taking the Lead in IT Mindshare

The next phase of data center transformation is just beginning, and some claim it will be on the coattails of cloud computing. There is certainly interest and a groundswell of a need to know, as shown in Figure 3. Cloud computing takes the theme of information in a new direction, as a tool in the hands of the individual and businesses alike in a new way to a new level.[1] Up until now, even as the costs of gathering, storing, and managing information continue to fall, the job of 'information management' remained a monopoly of big businesses. Today, third party providers and even individuals are becoming managers of their own personal information in their own right. Everyone has a digital footprint and its getting bigger every day.

The ascent of Data Longevity in the Cloud/Next Generation Data Center

When the aircraft carrier USS Nimitz takes to sea, it carries more than a half-million files with diagrams of the propulsion, electrical, and other systems critical to operation. Because this is the 21st century, these engineering diagrams and drawings are not on pieces of paper, but in digital files on the ship's computers. The shift to digital technology, which enables Navy engineers and technicians anywhere in the world to access the diagrams, makes maintenance and repair more efficient. This seems to be the case, but there are some issues. Several years ago, the Navy noticed a problem when older files were opened on newer versions of computer-aided design (CAD) software.

The person trying to access the diagram looked at the drawing and noticed that the current rendering of the image or schematic did not look exactly like the drawing did before. The changes were subtle, a dotted line instead of dashes or minor dimension changes, but significant enough to worry the Navy's engineers. Even the tiniest discrepancy might be mission critical on a ship powered by two nuclear reactors and carrying up to 85 aircraft.

The challenge of retrieving and validating digital files is not an issue just for the U.S. Navy. In fact, the threat of lost or corrupted data faces anyone who relies on digital media to store documents. These days, that is practically everyone. Digital information is so simple to create and store, we naturally think it will be easily and accurately preserved for the future. Sorry to say, nothing could be further from the truth. In fact, our digital information—everything from photos of loved ones to diagrams of Navy ships—is at risk of degrading, becoming unreadable, or disappearing altogether.

The problem is both immediately apparent and invisible to the average person. The issue comes up when our hard drive crashes, or our new computer lacks a floppy disk drive, or our online e-mail service goes out of business and takes our correspondence with it. We consider these types of data loss scenarios as personal catastrophes. All of these examples are symptomatic of a growing crisis. If the software and hardware we use to create and store information are not inherently trustworthy over time, then everything we build using that information is at risk.

Large government and academic institutions began grappling with the problem of data loss years ago, with little forward progress to date. Experts in the field agree that if a solution is not found soon, we could end up leaving behind a blank spot in history. Throughout most of our past, preserving information for posterity was mostly a matter of stashing photographs, letters, and other documents in a safe place. Personal accounts from the Civil War can still be read today because people took pains to save letters, but how many of the millions of e-mails sent home by U.S. servicemen and servicewomen from the front lines in Iraq will be accessible a century from now?

One irony of the Digital Age is that archiving has become a more complex process than it was in the past. You not only have to save the physical discs, tapes, and drives that hold your data, but you also need to make sure those media are compatible with the hardware and software of the future. Digital information is encoded in some format that requires software to render it in a form that humans can perceive. As the software that knows how to render those bits becomes obsolete, there is an issue of running the software that is also becoming obsolete.

In 2004, Miami-Dade County declared that it had lost almost all the electronic voting records from a 2002 election, due to a series of computer crashes, reminding us that many of the failures of digital record-keeping are attributable to everyday equipment failure[2]. Additionally, software companies can go out of business, taking their proprietary codes with them. In 2001, the online photo storage site, “PhotoPoint”, shut down and hundreds of people lost the digital photos they stored on the site.

[2] http://www.openvotingconsortium.org/files/voting_good_bad_stupid.pdf

The other issue is data loss is not always as apparent as a malfunctioning hard drive or a disc with no machine to play it. A digital file is just a long string of binary code. Unlike a letter or a photograph, its content is not immediately apparent to the end user. To see a photograph saved as a JPEG file, or to read a letter composed in a word processing program, we need software that can translate that code for us.

Software applications are updated on average every 18 months to two years, according to the Software and Information Industry Association, and newer versions are not always backward compatible. That could be a problem on the USS Nimitz, just as it could make trouble for you if the file in question held your medical records.

Likewise, law firms find that metadata (data about the data) such as the date when a file was created, are often not transferred accurately when files are copied. For example, magnetic storage media, such as hard drives, allow for a three-part date storage system (created/accessed/modified), whereas the file architecture of optical media, such as CD-Rs, allows for only one date. This presents issues in litigation, when attorneys must build chronologies of key events in a case.

The data crisis is by no means limited to the National Archives, or to branches of the military. The Library of Congress is in the midst of its own preservation project, and many universities are scrambling to build systems that capture and retain valuable academic research.

However, the programs in development for government and academia will not help find the lost e-mail of an individual computer user. Some experts believe that this is the result of simple market forces: Consumers, as a whole, show little interest in digital preservation, and corporations are in the business of meeting consumer demand. Others say corporations are only concerned with selling more products that are new.

These are just a few examples or use cases of data center attributes and what the next generation data center needs to solve going to the cloud.

The ascent of Virtualization in the Cloud/Next Generation Data Center

Virtualization technology is no longer exclusively used as a tactical tool to drive server consolidation and higher system utilization. The use of virtualization has matured; many

businesses are now leveraging the mobility of virtual machines to improve management and operations of IT environments. This next phase of virtualization includes a host of new use cases that range from high availability and disaster recovery to hosted clients and true utility computing in a private cloud. Server virtualization is now the default approach for new server deployments at many enterprises and IT businesses, and is quickly becoming the foundational platform for cloud computing initiatives. The next phase in virtualization will require a reinvention of IT policies and procedures, as well as continued adoption of automation and management tools, as IT moves to a more agile service delivery model. Virtualization and private clouds are the latest, vitally important, ingredients in IT leaders' long-term strategy to create much more efficient and agile IT service delivery.

For the past several years, many data centers have adopted virtualization, with the initial, simple goal of driving greater efficiency around infrastructure by consolidating hardware in order to drive out costs. In addition, that has been a successful first step: IDC research shows that the IT shops that have virtualized their servers have seen savings in the range of 20–25% or greater. Although IDC research shows that we are still in the early stage of virtualization adoption worldwide, only around 15% of servers installed are virtualized; in large shops, as many as 30% of servers are virtualized on average. As some of these customers have developed more experience with virtualization, they are starting to evolve their virtualization strategy to go beyond simple hardware consolidation. They are intelligently managing their virtualized infrastructures to provide greater speed, agility, and availability into their IT service delivery. For example, they are able to more dynamically deploy additional capacity, as needed, to mission-critical workloads; more quickly and cost-effectively deploy new applications; more quickly provision (and de-provision) test and development resources; and more cost effectively provision for disaster recovery. Private clouds will bring the efficiency, simplicity, and adoption speed benefits of public clouds into data centers, and yet IT still maintains control. So, what are private clouds exactly? They build on the highly efficient and flexible foundation of virtualization and add other important elements that bring those benefits more directly to end users, such as on-demand self-service, usage-based metering and chargeback, and simplified packaging. Private clouds, in effect, package the benefits of virtualization in a way that make them easier for IT groups to provide to their internal customers and for those users to leverage for greater business value.

For CIOs, the private cloud will be able to deliver IT services at the lowest possible cost and at the highest possible speed, in order to adapt quickly to changing business requirements, including new business applications, support for mergers and acquisitions, integrating new development and distribution partners, and supporting new business configurations (e.g., outsourcing/off shoring). With virtualization and the private cloud, CIOs are much closer to that goal of efficient and dynamic IT service delivery capability. IDC research shows that customers' number 1 concern is security. For a more detailed discussion on “Cloud Security”, please refer to the 2010 Proven Professional Paper titled “How to Trust the Cloud – “Be careful up there””, by Paul Brant and Denis Guyadeen, for more information on this topic.

As IT resources are shared in a virtualized or cloud environment, customers worry whether their applications and data will be more vulnerable to tampering, theft, or loss. Performance and availability, which you could think of together as "dependability”, are also concerns; while it's cost-efficient to share resources, some IT executives worry that in a shared environment, a spike in one workload's needs may siphon resources away from other workloads. These concerns, and others, tie to an overall concern about the manageability of these environments. While the ability to virtualize IT resources has come into the market rapidly, tools that help IT executives manage those virtual resources as expertly as they manage their physical IT resources have been slower to emerge. The good news is that within the past year or so, vendors have been creating tools that address many of these issues in a virtual environment, but the industry is playing catch-up. Compared with tools to manage physical IT resources, which have decades-long records of accomplishment, tools to manage virtual IT resources are still in the early stages of development. Customers are saying "show me" to virtualization and private cloud vendors; they are demanding the same kind of sophistication and the ability to monitor and deliver appropriate service levels with this new generation of virtualization and cloud management tools.

It is also important that IT leaders look for vendors that work closely with application vendors early in the product development process, with co-development and joint testing programs in standard virtualized environments. Total solutions need to be tested together and customers should increasingly look for that testing to have been done programmatically by the vendors, long before the solutions are deployed in the customer's environment. It is also essential that a vendor understand virtualization and cloud architectures in the broader context of the CIO's IT transformation agenda. Information technology executives are not trying to just consolidate

infrastructure in order to squeeze more from their budgets; many are trying to migrate to a dynamic IT service delivery model. It is a given that many vendors, including EMC (where I am gainfully employed), is embracing it in earnest. However, vendors must also understand that virtualization has to work with the other elements of that model, including service management, automation, increasing user self-service, transparent chargeback mechanisms, and governance. Typically, vendors that understand this strategic context also have a strong understanding of, and practice in, IT service management and best practice models, such as ITIL.

Table1 shows a short list of issues that must be addressed in the next generation data center. With challenges of power, cooling, and data growth rates, something has to give. Electronic mail, with 90% being unwanted, and average daily usage off the charts, it is out of control. Please refer to Table 5 on page 236 for the full list.

Average annual digital storage demand rate (primary occurrence of data, all platforms)	35-40% (2006-2008)
Average annual disk drive real density increase	35-50% (downward trend expected as recording limits begin to appear)
Average annual disk drive performance improvement (seek, latency and data rate)	<4% (mainly with data rate, as seek time improvement is minimal)
Power usage breakdown in typical data center	Chiller – 33%, IT gear – 30%, UPS – 18%, AC – 9%
Annual average increase in electricity cost	20-40% (depending on geography)
Who gets the IT energy bill?	Facilities team – 56%, IT team 3%
Annual growth rate of unwanted e-mail message traffic	~350%
Maximum possible distance from primary data center for synchronous replication	30-50 miles
Average number of spam e-mails delivered every 30 days	>3.65 billion
Number of e-mails sent daily in 2006 (est.)	>35,000,000,000 (billion)
Percentage of customers retaining e-mail	9%

archives over 7 years	
The size of WW wireless calls in PB	2,300
Percentage of all e-mail traffic that is unwanted	~90%
Percent of companies citing employees as the most likely source of hacking	77%
Percentage of US adults with more than 200GB of storage capacity	10% (approximately 28 million)
Average digital archive content in a Fortune 1000 company in 2007	>250TB (52% cagr)
Percentage of CIOs reporting to CEO in 2006	66% (was 56% in 2005, 17% to CFOs)
Projected size of India's IT services industry in 2010	$60 B
Projected size of China's IT services industry in 2006	$8.9 B (est. cagr. 18.9%)
Percentage of businesses who take backup tapes offsite daily, weekly, and monthly?	Daily – 56%. Weekly -32%. Monthly - 4%.

Table 1 Predominant Data Center Challenges

The ascent of the Appliance in the Cloud/Next-Generation Data Center

It is interesting to note that with the transformation of data centers into the cloud, and even within existing data centers, there is a move to the ubiquitous platform, called the appliance. Appliances within the data center have come and gone, but they are back in a big way. Appliances are a perfect fit with the advent of the cloud, and especially with the need to address the requirements of the Next-Generation Data Center.

With the advent of the changing times, vendors are scrambling to make optimized hardware-software combos. There seems to be a renewed appreciation for this type of architecture, optimized software-plus-hardware systems, or appliances.

The question is, are appliances, which consist of integrated hardware-software combos truly going to be game changers for CIOs—delivering far greater performance, requiring dramatically shorter installation times, and demanding zero tuning and configuring and retuning?

One can argue that it is, given the changing times of data center architectural changes going on today with new business requirements swelling into the market because they can help companies of all sizes make the leap into the very different and demanding world of real-time business.

These optimized machines are in full ascendancy at this time, because they are the most powerful and highest-value delivery platforms for some dazzling new software applications and technologies designed to analyze, not just bigger mountains of data, but to do so in less time and with greater insights, and with almost-unlimited variations. Today's next-generation data center enterprise software is bringing alive the promise of business analytics, predictive analytics, real-time analytics, real-time OLTP, staggeringly large databases, and the soaring volumes of queries triggered by many millions of mobile business users. In doing so, it has become so powerful and so complex, that generic servers, even the biggest and powerful boxes, simply cannot exploit the full range of insights, foresights, and opportunities that today's top software can deliver. This rise leads to the concept of what is termed “Applianceization” which notes the fact that this technology, even though old is new again, but in a big way and will change how IT does business. For more detail on this subject, and the genesis of the term “Applianceization”, please see the section titled “Applianceization”, starting on page 47.

The ascent of the Intercloud – AKA Interconnected Global Connectivity

In order for the Next Generation Data Center to grow and thrive, it is important to consider not only cloud technologies, types, and interfaces, but also the fact that a “cloud” is not an island. There will be situations were one needs to address the interconnectivity of clouds.

There is a well-founded skeptical question as to whether "cloud computing" is just the 2008 re-labeling of "grid", "utility", and "network computing". As discussed by Vinton Cerf, the person most often touted as the father of the Internet (no, not Al Gore, who is touted as the inventor of the Internet), was quoted in “Cloud Computing and the Internet” that we are ripe for an Intercloud, in the same way we were once ripe for the Internet:

Cloud computing is at the same stage [pre-Internet]. Each cloud is a system unto itself. There is no way to express the idea of exchanging information between distinct computing clouds, because there is no way to express the idea of "another cloud." Nor is there any way to describe the information that is to be exchanged. Moreover, if the information contained in one computing cloud is protected from access by any but authorized users, there is no way to express how that protection is provided, and how information about it should be propagated to another cloud when the data is transferred.

There are many unanswered questions that can be posed about this new problem. How should one reference another cloud system? What functions can one ask another cloud system to perform? How can one move data from one cloud to another? Can one request that two or more cloud systems carry out a series of transactions? If a laptop is interacting with multiple clouds, does the laptop become a sort of "cloudlet"? Could the laptop become an unintended channel of information exchange between two clouds? If we implement an inter-cloud system of computing, what abuses may arise? How will information be protected within a cloud and when transferred between clouds. How will we refer to the identity of authorized users of cloud systems? What strong authentication methods will be adequate to implement data access controls?

The question is, what will the Intercloud look like? Many have weighed in with their visions of the future. We will discuss this in detail as to whom the Next-Generation Data Center and cloud, as we know it, will interact.

The ascent of the Personal Data Store in the Cloud/Next-Generation Data Center

For IT and data consumers in general, the current approach to collecting and using personal and corporate data is dysfunctional. As individuals, we have lost control over our personal data. So much so that most of us see personal data management as a threat and a risk (identity theft), a hassle and a chore (red tape), and a source of frustration and irritation (businesses taking our data and losing it).

Another issue is what the author refers to as the "Facebook Debacle"! The social networking juggernaut "Facebook" is the perfect example. Recently, the number of active Facebook users has reached over five hundred million people. Users are putting all different types of personal

data into Facebook's databases, not knowing or caring how this data will be used. This is especially true for young people between the ages of 12 to 24. The scary thing is, given their lack of interest or foresight of the potential issues of placing data on the Web; it will be very interesting to see how this data will affect their lives, potentially in a very negative way in the future.

Preservation Data Stores (PDS), that will be discussed in detail, is a novel storage and process component that supports digital preservation environments, ensuring data usability and integrity over long periods of time as well as addressing the "Facebook Debacle". PDS supports new functionalities and extensions that are specific for logical preservation. It encapsulates the raw data with its complex interrelated metadata objects, so they are inseparable during the migration processes, and when accessing the data in the future. PDS decreases data transfer between applications and storage by offloading data intensive functions, such as fixity computations and transformations to the storage. The PDS storage system simplifies the applications by transferring the responsibility for managing the storage-related events, such as provenance events, to the storage itself.

PDS can be used in a preservation solution as the storage infrastructure that manages preservation objects over time. It can be integrated in a cloud-based service for providing archive and long-term retention and preservation service. PDS can also be integrated with an enterprise content management (ECM) system to provide logical preservation support to the ECM. We must find an "OAIS" for out data (See OAIS-based long-term architectures that will be discussed in subsequent chapters). PDS maps the preservation objects to the ECM object model, utilizing its advantages as well as ensuring the preservation object encapsulation, co-location, and usability over time.

The Personal Data Store (PeDS)[3], not to be confused with the Preservation Data Store as discussed previously, is a novel idea and could not come a moment too soon. The PeDS is a container for personal information. Even though the PDS and PeDS have similarities, the primary function is on personal data management and has the following attributes:

- Data storage
- Management

[3] http://wiki.eclipse.org/Persona_Data_Model_2.0

- Sharing
- Verification
- Identity assurance
- Privacy management

The PeDS will, in the opinion of the author, become a pivotal foundational information utility of the 21st century and beyond, within the next-generation data center. Personal Data Stores will become informational equivalents of electricity supply or the plastic payment card for the 20th century consumer: rather boring and rather taken for granted, but absolutely essential. Utilities like this work because they make life easier for everyone and because they make new things possible. They are 'platform' services. They create a platform for everyone and everything else to walk on. The more taken for granted they are, the more successful they are. Like electricity supply and payment cards, they also create significant behind-the-scenes technical and eco-system challenges. We pay by plastic cards because it is easy and convenient. But behind each payment, there is a hugely sophisticated system of highly secure data 'handshakes' taking place across a complete eco-system of supporting players: credit card issuers, merchant acquirers, retailers, and so on. Behind-the-scenes complexity and robustness march hand-in-hand with the simplest possible user interface. Personal Data Stores are no different. To achieve their essential utility status, they will need to surmount many challenges including:

- Data security, both in data storage and data sharing. Individuals have to be confident that the data in the PDS is safe and will not be compromised.
- Ease of use. The biggest attribute spurring adoption of the PDS is convenience. If it fails to deliver easy, intuitive help in day-to-day chores, it won't succeed.
- Easy population of the data store, with equally easy access and use—to correct, update, link, share, and so forth. (e.g. automatic capture of electronic receipts and other transaction records).
- Easy-to-use and understand data sharing agreements, protocols, and processes. For example, 'subscribe to me' services require relatively new technologies, such as information cards, to bypass clunky first-generation password + cookie user access to business information systems online.
- The development of technical, legal, and other standards to support data sharing and data sharing agreements.
- The ability for data fields in Personal Data Stores to talk to data fields in businesses' databases without confusion or error. This requires the development of sophisticated data architectures.

- The ability to easily gather and share bespoke ‘bundles’ of data from the data store (for example, driver’s license, insurance, identity assurance, and financial information for the purposes of buying or hiring a car).

Depending on the task at hand, this might involve working with a range of different types of personal data, including data that identifies an individual, such as name, address, etc. In addition, data extracted by other parties, such as a passport number or an individual’s credit reference ratings. The list also includes:

- Information gathered by the person, such as data generated by an individual’s dealings with other parties, such as, transaction and interaction records
- Information created by the person which includes plans and preferences
- Information about a person such as mash-ups of information created by the individual, conferred by other parties, and gathered that includes financial circumstances, health, skills and learning, etc.
- The ability to analyze data in the store to identify trends, extract insights, and so on, enabling discovery of information by external parties who can access specific data in individuals’ personal data stores on a permission-only basis (e.g. seeing a purchasing or behavior profile on a personally identifiable basis).

Person-centric information sharing agreements mark a new stage in both information management and in the relationship between individuals and businesses. Some of the key attributes of information sharing agreements are:

- They are practically oriented and specific, focusing on a specific problem or information sharing need.
- They release a genuinely new class of information—‘volunteered personal information’ (VPI) that previously only the individual knew, could see, or had access to.
- They give individuals the confidence to share information they would have previously withheld—because now they know appropriate safeguards are in place.
- They are—have to be—user-friendly, based on a small number of standards, well explained and understood agreements covering the main information-sharing scenarios individuals are likely to encounter.
- They are machine readable, so their generation and consumption can be automated, including their comparison to a baseline set that are pre-approved by the individual—thereby helping the individual determine differences, extensions, and so forth.
- They operate above the level of all global privacy regulations, offering individuals and businesses a release from country-to-country regulation differences, arbitrage, etc.

- The deployment of these agreements within a broader trust framework (which explains standards, inter-operability, how liability is handled, etc.) will create a secure, efficient, and workable foundation for rich, mass scale information sharing between individuals and businesses. This is the information needed to fuel the rise of personal information management services.

The Personal Information Management Ecosystem

"Personal Data Stores' first use will be as an information utility, helping individuals manage daily informational chores, including their dealings with suppliers, public services, and other information repositories. As a storage platform, personal information management systems will also provide a secure foundation for a host of new types of service. Let's take just two examples. Reinventing marketing. Current marketing communication systems work by guesswork. Sellers don't know who is interested in what service at what time, so they have to make either an educated guess or simply spam people with messages on the chance that a certain percentage of these messages will be picked up. These processes are wasteful to sellers and irritating and/or intrusive to buyers.

Personal Data Stores will help buyers express and communicate their preferences and specifications to the marketplace, helping sellers communicate with the right people about the right things at the right time. Using Personal Data Stores will hopefully reverse a currently wasteful and often adversarial process into an efficient, win-win proposition instead.

Personal Data Stores will create "Added Value Services". It is anticipated that once the core data functionality is in place, countless new data-informed services will become possible. Imagine a new person-centric car buying service, for example. The process starts with specification-building. This spec-building service combines a number of different types of data, including: one's specific plans and preferences. Input examples include, what is the intended use of the vehicle, what are the likes and dislikes including brand, features, options, and so on? What is one's budget, including whether one can pay for it, as well as credit scores? What about previous car usage, including mileage, trade-in value, length of ownership, service history, etc?. With data such as this, the service helps the buyer build an 'ideal' specification, including potential trade-offs and compromises. The service then takes the spec to market. Sellers provide offers matched to the specification, which the buyer then considers. The buyer then

either tweaks the spec for a second round or makes a purchase decision, at which point he or she can use the PDS to share the data needed to complete the buying process.

Over the years, individuals have to manage countless different 'life events' and episodes, such as buying a home, buying a car, organizing a holiday, getting married, getting divorced, having a child, changing schools, etc. They also have to manage a whole series of ongoing life management processes, such as 'my money', 'my health', 'my communications infrastructure', 'my home', and so on. Each one involves intensive bursts of finding, sifting, sorting, and providing a wide range of different types of information.

With Personal Data Stores as a foundation, countless specialist Personal Information Management Services will be able to enrich, streamline, and enhance these processes, both for the individuals they are working for, as well as the individual's suppliers. What is being discussed is the evolution of a new and different commercial and public service ecosystem, driven by enriched, permission-based exchanges of personal data. The understanding and the ability to embrace this new technology are pivotal to the transformation of the Next Generation Data Center and Cloud to the next millennium.

The Drivers for the Next-Generation Data Center

With all the challenges discussed so far, how one grasps these long-term IT complexities in an organized manner is important.

I propose the following triad of principles for the process of transforming the Next Generation Data Center for the next 100 years and beyond. As shown in Figure 4 Next-Generation Data Center Drivers, in order to address the requirements outlined, one needs to address the following core drivers;

- A sustainable reference model
- An understanding of the attributes and technologies required and available to achieve the desired transformation results
- An understanding of the Business Drivers and Use cases for "Riding the Clouds" and the transformation of the Next Generation Data Center. This is especially important as to how it relates to the new Cloud paradigm variants—public, private, etc.—which will be a major driver in the transformation process.

Each Triad member drives the other. Business-related drivers require people, process, and technology. In addition, to achieve an efficient and profitable business, a sustainable IT model is required.

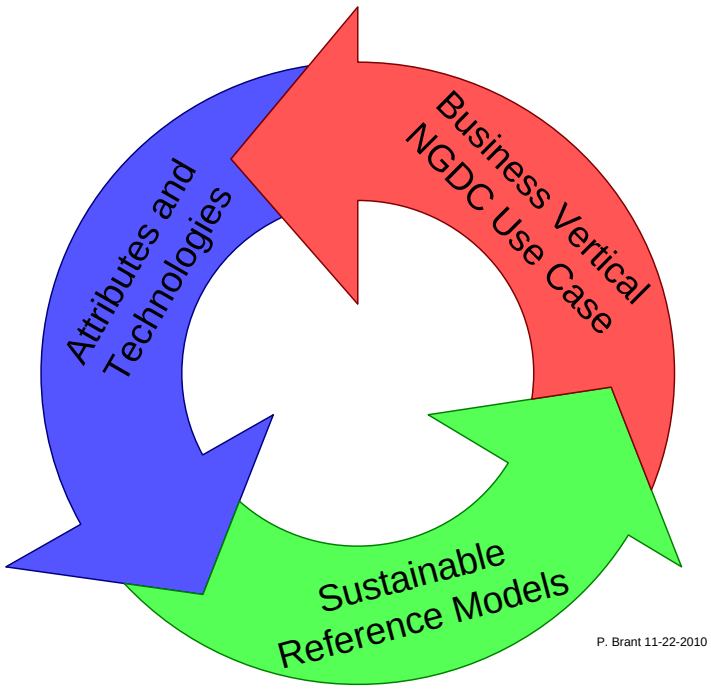


Figure 4 Next-Generation Data Center Drivers

To address all the varying challenges, as previously mentioned, understanding what the business drivers are, and how it maps to the various use cases, will give insight to proper and appropriate methodologies. How do the various Cloud models drive value? How do the various technology tools meet the desired goals?

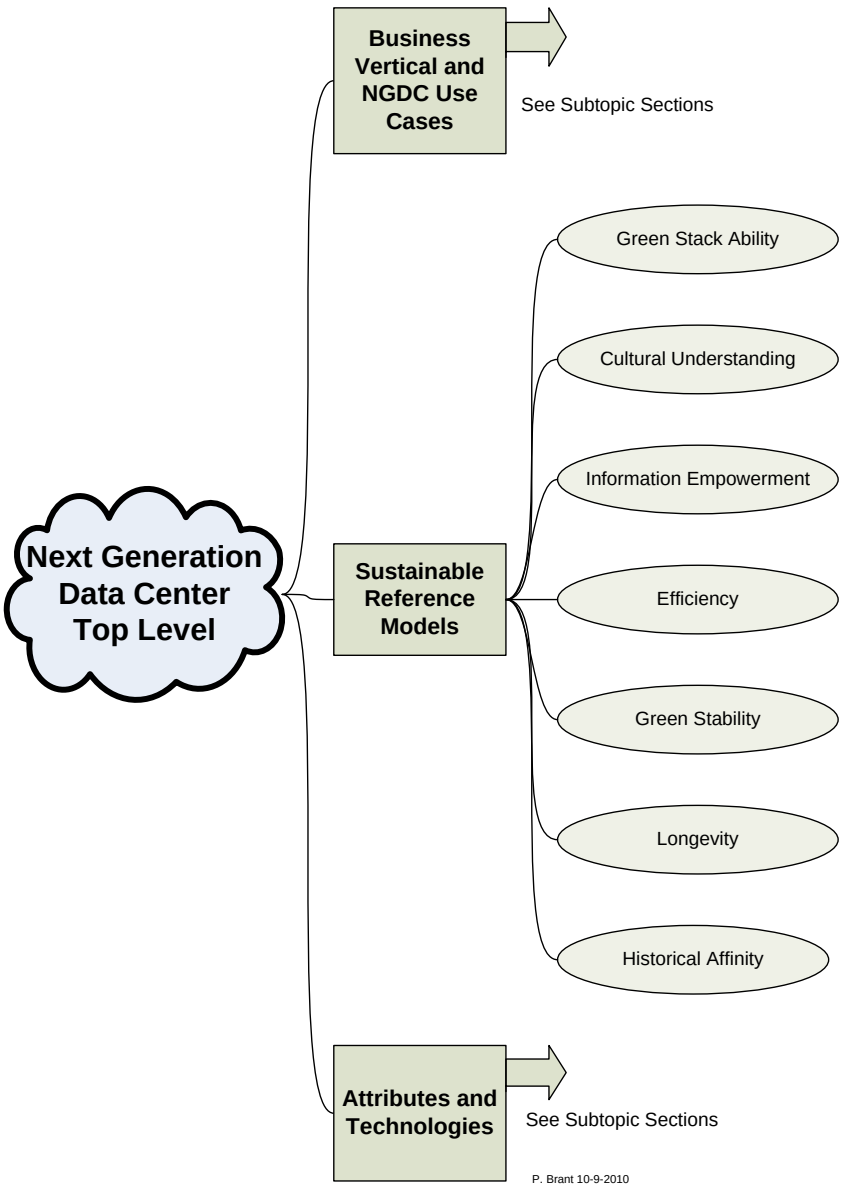


Figure 5 Top Level NGDC (Next-Generation Data Center) Ontology

How does one attain a sustainable process, allowing corporations, small to medium businesses, as well as individuals, meet the required goals? All this is outlined in the detailed ontology layout shown in Figure 5. Please see the other subtopic sections outlined in the Ontology in subsequent sections. The core drivers are all interrelated. It all comes down to IT business requirements and addressing the information consumption community.

Attributes and Technologies

The first Triad, as it relates to “Riding the cloud” and focusing on the “Next-Generation Data Center”, is to understand what technologies are available to the practitioner and what are the attributes and technologies that are important moving in this direction.

With increasing complexity of the data center, the problems of power consumption, space, management, maintenance costs, security, storage, reliability, and migration processes are gradually exposed. The critical problem that chief information officers (CIOs) face is how to improve data center efficiency, protection, price, and performance, to make the data center become a service innovation center in the information age. The path is clear, moving to a cloud architecture, be it private, public, or any of the other pluralities is the transformation approach for the next 100 years.

From the aspect of function, the data center quietly changes. The traditional data center is based on infrastructure, which is the IT platform of the service system and provides IT resources for the service system. However, the future data center integrates the information center and IT resources, which are distributed according to customers' needs and invoked by service system flexibility. In addition, the data center is based on services, manages information in a centralized manner, and provides information support for service innovation. The innovation required will come from many sources. The Storage Ecosystem (see page 114) has to change from what we know today. Storage needs to be smarter, more efficient, and more in tune to what the business drivers require, and what technology is available today and in the next 100 years, to attain sustainability.

In addition, embracing a new, but interestingly old and mature technology model—termed in this paper as “Applianceization”—will be discussed. The Appliance is back and in a big way.

The Personal Data Store also needs to be seriously addressed, given the “Facebook Debacle”. Since everyone has a digital footprint on this planet, and considering the majority of the digital storage growth will come not from the data center, but from individuals like ourselves, this is a major focal point.

It is also a given that object affinity is a huge enabler in the Next-Generation Data Center. Information, computation, and networking are becoming virtualized and as such, the fundamental methodology of business processes is becoming objectified.

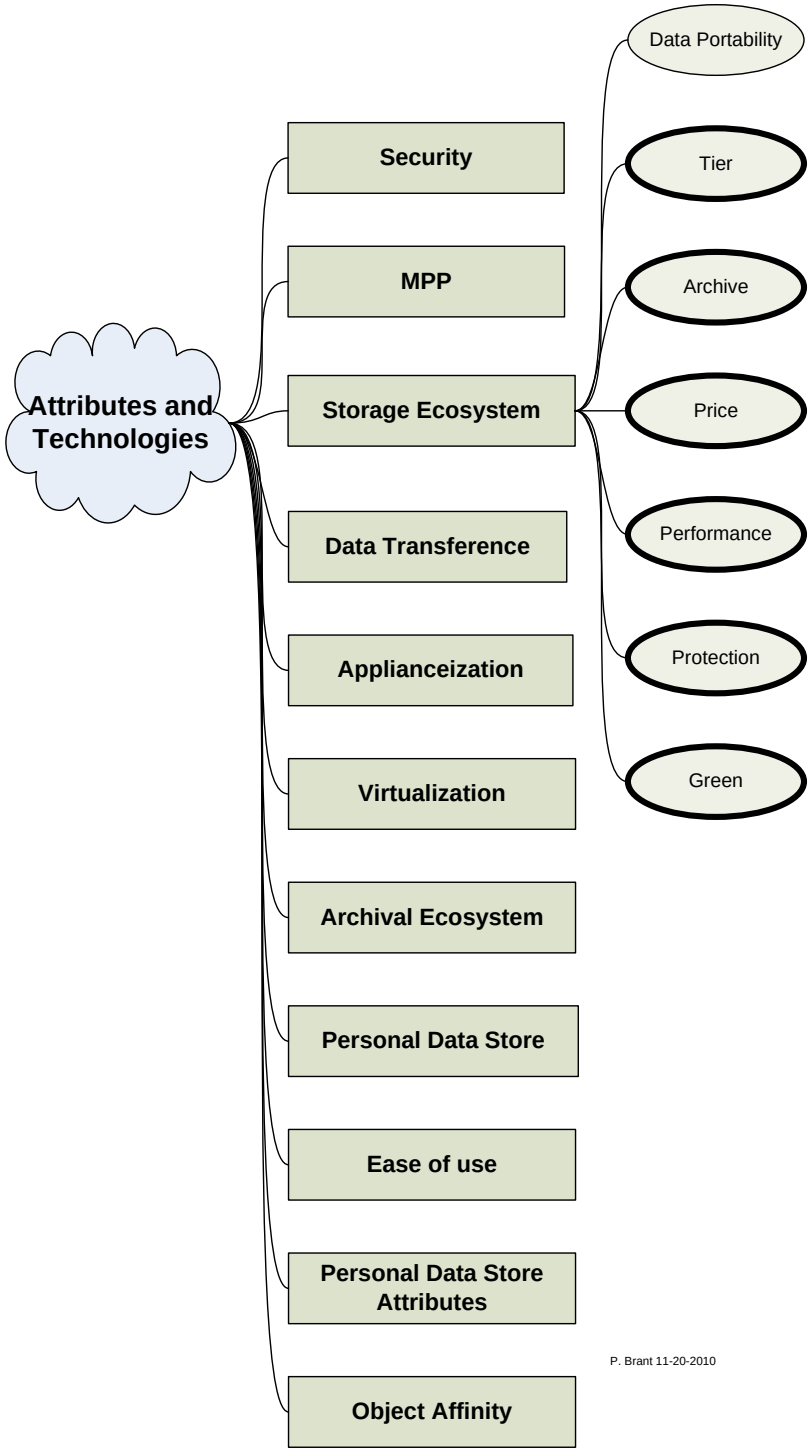


Figure 6 Attributes and Technologies

Massively Parallel Processing (MPP) and other architectures

In the world of the transformation into the Next-Generation Data Center, the need to conform to a best practice that relates to performance and scalability is a requirement. A case in point is deploying data warehouses. The Importance of Scalable Architectures is that many companies deploying data warehouses start small and grow their infrastructure, as the amount of data and the demands on the decision support systems increase. Data warehouses are often initially deployed with 16 or fewer processors, and require a growth path that can support many times the initial processing capability.

As processors are added to an SMP (Symmetric Multiprocessing), or nodes are added to MPP (Massively Parallel Processor) Systems, it is important for the system to scale. Ideally, a system will demonstrate a property called speed-up, in which a job that requires one unit of time to complete with one processor will require 1/N of the time to complete with N processors. For example, if a job that requires ten hours to complete with one processor requires only one hour to complete with ten processors, the system scales well.

Another desirable characteristic of scalability is called “scale-up”. A system with excellent scale-up offers the same level of performance as the size of the data warehouse increases through the addition of processors or nodes. For example, a batch job that takes ten hours to run when the database is one terabyte in size, will take the same length of time at two terabytes, simply by doubling the number of processors.

Best Practice – Understand advantages and trade-offs of scalable systems in the NGDC

Many database administrators look at scalability from the standpoint of whether the system has predictable behavior, as the concentration of the workload increases. A system that scales well is one that holds no surprises, as both the system/infrastructure and the workload grows.
In terms of architecture types, one can segment it into three types:

- MPP (Massively Parallel Processor Systems)
- SMP (Symmetric Multiprocessing)
- Clustered Systems

Massively parallel processor systems use a large number of nodes, each accessed using an interconnect mechanism that supports message-based data transfers (see Figure 7). This diagram illustrates that each node is a self-sufficient processor complex, consisting of CPU, memory, and disk subsystems. An “MPP” architecture is considered a “shared nothing” system

because memory and I/O resources are independent, with each node even supporting its own copy of the operating system. MPP systems promise unlimited scalability, with a growth path that allows as many processors as necessary to be added to a system.

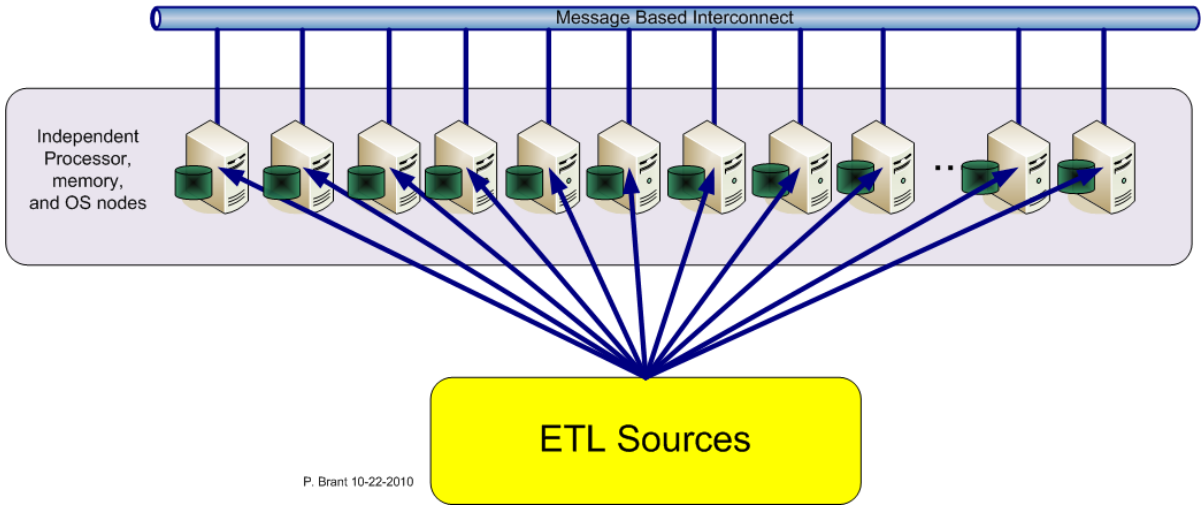


Figure 7 MPP Architectures

MPP architectures provide excellent performance in situations where a problem can be partitioned, so that all nodes can run in parallel with little or no inter-node communication. In reality, the true ad hoc queries typical of data warehouses can only rarely be so well partitioned, and thus limit the performance that MPP architectures actually deliver. When either data skew or high inter-node communication requirements prevail, the scalability of MPP architectures is severely limited. Reliability is also a concern with MPP systems because the loss of a node does not merely reduce the processing power available to the whole system; it can make any database objects that are wholly or partially located on the failed node unavailable. An interesting industry trend is for MPP vendors to augment single-processor “thin” nodes with multiprocessor “fat” nodes, using many processors in an SMP configuration within each node. If the trend continues, each MPP node will have increasing numbers of processors, fewer nodes, and the architecture begins to resemble clusters of SMP systems, discussed below.

SMP Architectures

Symmetric multiprocessors fall on the opposite end of the spectrum from MPP systems. These systems consist of, from a pair, to as many as 64 processors that share memory and disk I/O resources equally, under the control of one copy of the operating system (see Figure 8).

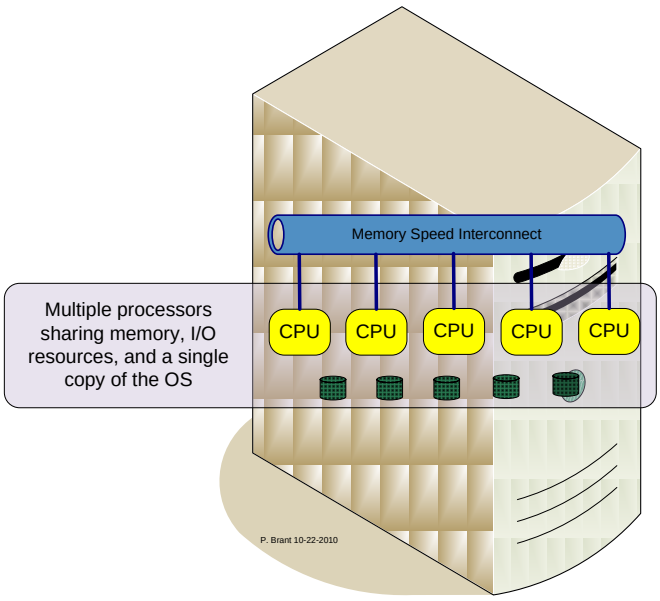


Figure 8 SMP (Symmetric Multi Processing) Architecture

Because system resources are equally shared between processors, they can be managed more effectively. SMP systems make use of high-speed interconnections to allow each processor to share memory on an equal basis. These interconnections are typically two orders (2X) of magnitude faster than those found in MPP systems. In addition to high bandwidth, low communication latency is also important if the system is to show good scalability. Latency is a very important characteristic in data warehouse database applications. The reason is that common data warehouse database operations, such as index lookups and joins, involve communication of small data packets; when the amount of data contained in each message is small, the importance of low latencies is paramount.

Clustered Architecture Systems

When data warehouses must be scaled beyond the number of processors available in current SMP configurations, or when the high availability (HA) characteristics of a multiple-system complex are desirable, clusters provide an excellent growth path. High-availability software can enable the other nodes in a cluster to take over the functions of a failed node, ensuring around-the-clock availability of enterprise-critical data warehouses. A cluster of four SMP servers is illustrated in Figure 9.

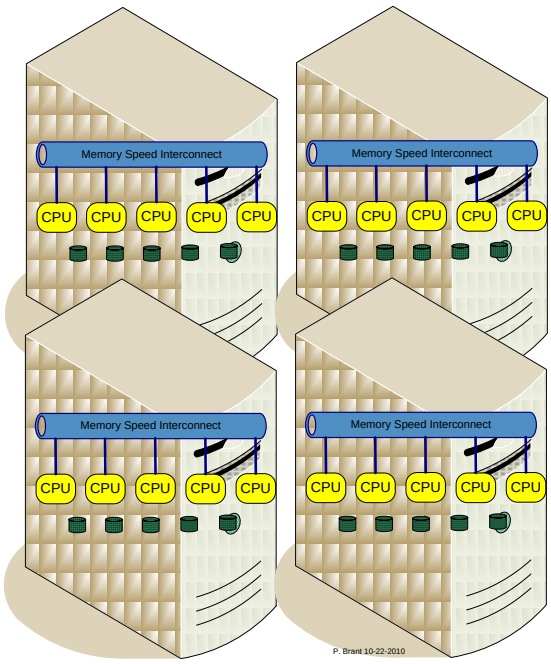


Figure 9 Cluster Architecture

The same database management software that exploits multiple processors in a SMP architecture and distinct processing nodes in a MPP architecture can create query execution plans that utilize all servers in the cluster. With their inherently superior scalability, SMP systems provide the best building blocks for clustered systems. These systems are configured with multi-ported disk arrays so that the nodes, which have direct disk access, enjoy the same disk I/O rates as stand-alone SMP systems. Nodes not having direct access to disk data must use the high-speed cluster interconnect mechanisms and methodologies. In data warehouses deployed with clusters, the database management system or load-balancing software is responsible for distributing the various threads of the DSS queries across the multiple nodes for maximum efficiency.

Decision Support Systems (DSS) are a specific class of computerized information system that support business and business decision-making activities. A properly designed DSS is an interactive software-based system intended to help decision makers compile useful information from raw data, documents, personal knowledge, and/or business models to identify and solve problems and make decisions. Typical information that a decision support application might gather and present would be:

- Accessing all of your current information assets, including legacy and relational data sources, cubes, data warehouses, and data marts

- Comparative sales figures between one week and the next
- Projected revenue figures based on new product sales assumptions
- The consequences of different decision alternatives, given past experience in a context that is described

As with MPP systems, the more effectively that the query can be partitioned across the nodes in the cluster, and the less inter-node communication that is necessary, the more scalable the solution will be. This leads to the conclusion that clusters should be configured with as few nodes as possible, with each SMP node scaled up as much as possible before additional nodes are added to the cluster. Beware that the larger the number of nodes in a cluster, the more the cluster looks like an MPP system, and the database administrator may need to deal with the issues of large numbers of nodes much sooner than they would with clusters of more powerful SMP systems. These are the familiar issues of non-uniformity of data access, partitioning of database tables across nodes, and the performance limits imposed by the high bandwidth interconnects.

A direction that many Data Warehouse applications are going with clustered systems is to develop software that allows a single system image to be executed across a clustered architecture, increasing the ease of management far beyond that of today's clusters, as is the case of the EMC recently acquired "Greenplum" product. More information on this product to follow, starting in the section titled "Applianceization", on page 47.

Best Practice – Understand Advantages in MPP (massively parallel processing) in addressing the NGDC

As previously discussed, many business solutions require parallel processing to execute on a particular application. One model that is used quite often is two laws that try to express the breadth that performance parallel processing can afford, as well as the limitations. The Amdahl Law and Gustafson's Law[4] describes the estimated speedups, as measured by parallel program potential. In 1967, Amdahl's Law[5] was used as an argument against massively parallel

[4] http://johngustafson.net/guslaw/guslaw.htm

[5] http://software.intel.com/en-us/articles/amdahls-law-gustafsons-trend-and-the-performance-limits-of-parallel-applications/

processing. Since 1988, Gustafson's Law has been used to justify massively parallel processing (MPP). Interestingly, one would argue that these two laws complement each other.

Best Practice – Understand Amdahl's and Gustafin's Laws as it relates to MPP to achieve NGDC architectures

Amdahl's law is quite well known in the area of clustered architecture analysis. In this discussion, let's define speedup as the original execution time divided by an enhanced execution time[6]. The modern version of Amdahl's law states that if you enhance a fraction f of a computation by a speedup s, the overall speedup is as shown in Equation 1 – Amdahl's General Law, below:

Equation 1 – Amdahl's General Law of Speedup of MPP Architectures

$$Speedup_{enhanced}(f,s) = \frac{1}{(1-f)+\frac{f}{s}}$$

As shown in Figure 10, as one increases the number of processors or increases the percentage of the application's code or processes that can be done at the same time or parallel, one can achieve increased processing time. Amdahl's law applies broadly and has important corollaries such as:

- Attack the common case: When f is small, optimizations will have little effect.
- The aspects you ignore also limit speedup: As s approaches infinity, speedup is bound by 1/(1 – f).

Four decades ago, Gene Amdahl defined his law for the special case of using "n" processors (cores) in parallel when he argued for the single-processor approach's validity for achieving large-scale computing capabilities[7]. He used a limit argument to assume that a fraction f of a program's execution time was infinitely parallelizable with no scheduling overhead, while the remaining fraction, 1 – f, was totally sequential.

---

[6] "From a Few Cores to Many: A Tera-scale Computing Research Overview," white paper, Intel, 2006; ftp://download.intel.com/ research/platform/terascale/terascale_overview_paper.pdf.

[7] J.L. Gustafson, "Reevaluating Amdahl's Law," Comm. ACM, May 1988, pp. 532-533.

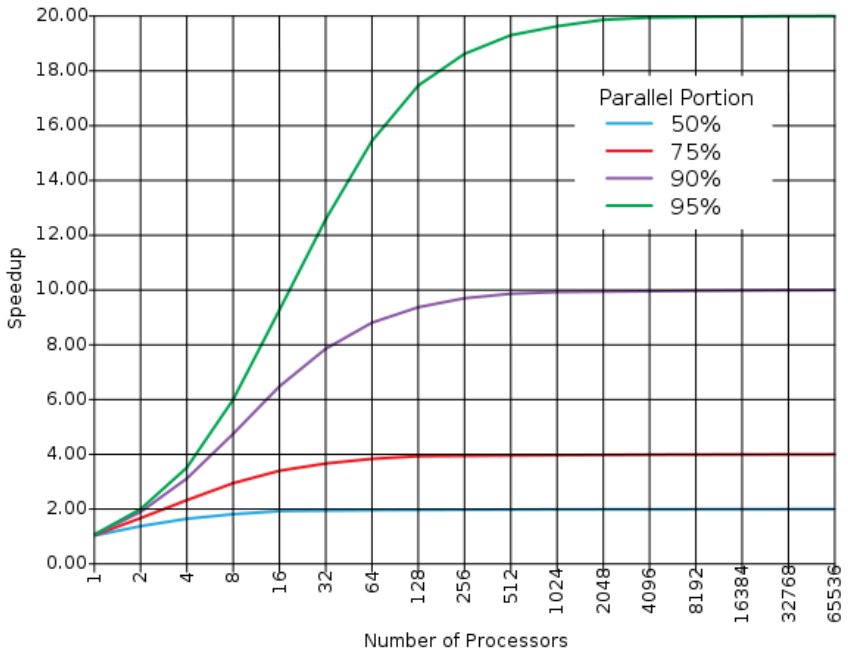


Figure 10 Graph of Amdahl's Law

Without presenting an equation, he noted that the speedup on "n" processors is governed by Equation 2 – Amdahl's Parallel Law of Speedup of MPP Architectures, shown below:

Equation 2 – Amdahl's Parallel Law of Speedup of MPP Architectures

$$Speedup_{enhanced}(f,n) = \frac{1}{(1-f)+\frac{f}{n}}$$

Amdahl argued that typical values of (1 – f) were large enough to favor single processors. Despite their simplicity, Amdahl's arguments held, and mainframes with one or a few processors dominated the computing landscape. They also largely held in the minicomputer and personal computer eras that followed. As recent technology trends shepherd us into the multicore era, Amdahl's law is still relevant. Amdahl's equations assume, however, that the computation problem size does not change when running on enhanced machines. That is, the fraction of a program that is parallelizable remains fixed.

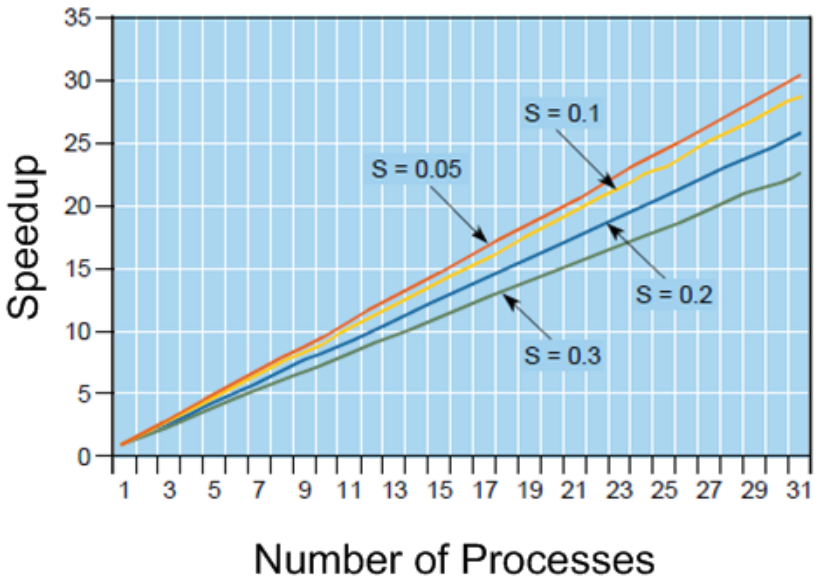


Figure 11 Graph of Gustafson's Law

There are many examples of applications that do not have a fixed problem size. For example, in a network routing application, there may be an initial configuration phase that cannot be parallelized, followed by the main task of routing and processing data packets. The number of packets is usually unknown; indeed the design goal may be to handle as many packets as possible. In such a system, it's easy to see how adding more cores could boost performance. In this case, each core can receive a new packet to process, when it has completed processing the last packet. Adding more cores means more packets processed in parallel. In such a system, the relative size of the serial portion decreases over time and the parallelized portion grows. Gustafson's Law, named after John L. Gustafson, states that the speedup for such a system, known as "Scaled Speedup", is as follows. Gustafson's Law shows that for a system where the problem size is not fixed, performance increases can continue to grow by adding more processors. The graph shown in Figure 11, describes the curves for Gustafson's Law with different values for the serial portion and number of processes. Notice how speed continues to increase with more cores or processes.

Equation 3 – Amadahl's Parrallel Law of Speedup of MPP Architectures

$$Speedup(s,n) = n + (1-n) \times s$$

Where:

- "s" is the serial portion of the process or algorithm running parallelized
- "n" is the number of processes

John Gustafson argued that Amdahl's law does not do justice to massively parallel machines because they allow computations, previously intractable in the given time constraints[8]. A machine with greater parallel computation ability lets computations operate on larger data sets in the same amount of time. When Gustafson's arguments apply, parallelism will be a far harder driver in accelerating performance. One can argue, however, that a vigorous general-purpose multicore design should also operate well under Amdahl's more pessimistic assumptions.

The underlying common stance between these two laws is that serial and parallel programs must compute the same total number of steps for the same input.

In the world of "Data Warehouse" designs and "Next Generation Scatter Gather Architectures" (See Best Practice – Implement "Parallel Everywhere" into a NGDC Data Warehouse Architecture, starting on page 174, regarding EMC's Greenplum's Scatter/Gather technology), the transformation of these fundamental laws can change. The use of a "serial percentage" concept in parallel performance evaluation is misleading. One should consider that processing times are more accurate in the NGDC argument used to formulate parallelism and process scaling.

The key to Amdahl's Law is a serial processing percentage relative to the overall program execution *time,* using a single processor. Therefore, it is independent of the number of processors. Gustafson revealed that it was indeed possible to achieve more than 1000 fold speedup using 1024 processors. This appeared to have "broken" Amdahl's Law and to have justified massively parallel processing.

An alternative formulation has been recently considered. This is often referred to as Gustafson's Law and has been widely referred to as a "scaled speedup measure". In

[8] J.L. Gustafson, "Reevaluating Amdahl's Law," Comm. ACM, May 1988, pp. 532-533.

Gustafson's formulation, a new serial percentage is defined in reference to the overall processing time using *P* processors. Therefore, it is dependent on *P*. This *P* dependent serial percentage is easier to obtain than that in Amdahl's formulation via computational experiments. But, mathematically, Gustafson's formulation cannot be directly used to observe *P*'s impact on speedup, since it contains a *P* dependent variable.

It is important to consider these equations as it relates to MPP and as will be discussed in the next section, an important attribute in the transformation of saleable appliance solutions. In order to make informative decisions as to the direction of the NGDC, understanding the theoretical relationships to performance and scalability is a given.

Applianceization

With the advent of cloud computing and the need to address the Next-Generation Data Center requirements of scalability and manageability, vendors are scrambling to make optimized hardware-software combos. The renewed appreciation of appliance architectures, and what the author proudly coined as “Applianceization”, are environments that are optimized, integrated hardware and software systems that are coming on strong and coming to your cloud neighborhood soon.

These optimized machines are in full dominance at this time, because they're the most powerful and highest-value delivery platforms for some emerging new and many existing software applications and technologies, designed to analyze not just bigger mountains of data, but to do so in less time and with greater insights, and with almost unlimited variations. With vendors all reaching to deliver an integrated solution of hardware, software, management, and infrastructure building an integrated stack, the appliance concept is a natural fit.

Today's next-generation enterprise software is bringing alive the promise of business analytics, predictive analytics, real-time analytics, real-time OLTP, resplendently large databases, and the soaring volumes of queries triggered by many millions of mobile business users. In doing so, it has become so powerful and so complex that generic servers, even the biggest and massive boxes, simply cannot exploit the full range of insights, foresights, and opportunities that today's top software can deliver.

In response, hardware vendors, software vendors, and the combination, are rushing in with their own combinations. With all this hype, however, there are some concerns.

- When used in such vital applications as analytics, business intelligence, and online transaction processing, will these engineered systems become the source of vendor lock-in? How do customers ensure that does not happen?
- As more CIOs look to slash their infrastructure costs to be able to devote more dollars to growth-oriented innovation, do these full-stack machines represent a step back in the old direction?

These are valid questions, to be sure. But one of the great things that is happening today is the list of vendors offering appliances is big and getting bigger, which means lots and lots of competition and innovations and better ideas and a huge focus on customer value. Businesses

want to transform their data centers, ride the clouds, and embrace the private cloud today and the public cloud tomorrow. All of that mitigates aggressively against sneaky lock-in tricks and hidden integration messes and, with a new and potentially huge market looming, these IT vendors should be on their very best behavior from the outset.

The Rise of the Appliance has definitely begun, and from the author's position, it looks like there are many significant advantages in being part of this powerful business-centric movement.

Best Practice – Consider Applianceization in Big Data Architectures

Big data appliances promise a more convenient way to package the database and its operating system, as well as related parts, such as encryption and compression, on hardware optimized for that job. Erroneous configurations frequently lead to underperforming data warehouses. The database encryption function, for instance, can now be linked to certain hardware assists embedded in Intel's latest chips, Xeon 7500 and 7600. Appliances are not a new concept. Security vendors have been driving this model for years, for firewalls, anti-spam, Web security, and other functions.

The continuing explosion in data sources and the abundance of data volume strains that exceed the scalability of traditional data management and analytical architectures, moving to the Next-Generation Data Center and Cloud, has its challenges. Decades' old legacy architecture for data management and analytics is inherently ailing for scaling to accommodate today's big data volumes. These systems require huge outlays of resources and technical intervention in a losing battle to keep pace with demand for faster time to intelligence and deeper insight. In today's business climate, every leading business finds itself in the data business. While at one time, it might have been acceptable for companies to capture just the obviously "important" data and throw the rest away, today leading companies with Personal Data Sources and data footprint growth, understand that the value of their data goes far deeper. The decision about which data to keep and which to throw away is becoming more and more important.

The question is which data will prove to be important in the future? This decision is very difficult to make. The result is that businesses must store and analyze data at the most detailed level possible if they want to be able to execute a wide variety of common business strategies. Combined with increased retention rates of 5 to 7 years or longer, as in the case of healthcare, it is no surprise that typical data volumes are growing by 1.5 to 2.5 times a year. Looking

forward, many businesses realize their future competitiveness will depend on new business strategies and deeper insights that might require data that isn't even being captured today. Projecting five years out, businesses should not be surprised if they want to store and analyze a rapidly growing data pool that is 100 times or more greater than the size of today's data pool. Extending beyond size, the depth of analysis and complexity of business questions raised against the data can only be expected to grow.

For example, the recently EMC Corporation-acquired technology from Greenplum, is focused on being the leading provider of database software for the next generation of data warehousing and large-scale analytic processing. Greenplum offers a new, disruptive economic model for large-scale analytics that allows customers to build warehouses that harness low-cost commodity servers, storage, and networking to economically scale to petabytes of data. This form of scalability will allow businesses to ride the cloud internally or externally as time goes on.

The Greenplum architecture leverages technology in another vector to gain performance. From a data warehouse performance standpoint, the progression of Moore's law[9] means that more and more processing cores are now being packed into each CPU. Data volumes are growing faster than expected under Moore's law, however, so companies need to plan to grow the capacity and performance of their systems over time by adding new nodes. To meet this need, the Greenplum technology makes it easy to expand and leverage the parallelism of hundreds or thousands of cores across an ever-growing pool of machines. The Greenplum technology massively parallel, shared-nothing architecture fully utilizes each core, with linear scalability and unmatched processing performance. For more information on how this technology can be used and how it can enhance the Journey to the Cloud and address the needs of the Next-Generation Data Center from a business and vertical use case perspective, please refer to the section titled "Data Warehouse – Big Data", starting on page 162 for additional information.

Best Practice – Utilize Applianceization in the Financial Sector

The financial sector is ripe for "Applianceization". This general appeal of appliances in financial services summarizes as follows:

[9] Moore's law (Gordon Moore, Intel cofounder) originally predicted in 1965 doubling the number of transistors on a piece of silicon every in 24 months. Real world was faster, every 18 months.

- Higher performance – Performance may mean lower latency, higher throughput, more consistent behavior, or a combination of the three. Appliances often include specialized hardware that eliminates performance bottlenecks to accelerate repetitive operations.
- Consolidation/cost reduction – Often a single appliance can do the work of many servers running software, resulting in a smaller data center footprint, less power consumed, fewer servers and software licenses and ultimately, lower costs. Whether trying to avoid costly collocation facility fees for front office solutions, or simply reducing data center costs in the back office, this is a focus of virtually every kind of financial firm.
- Simplified operations – Appliances offer turnkey installations that software cannot, usually shipping with everything installed and ready to work out of the box. With software, you have to install the server, install the operating system, patch the operating system to whatever level is supported by the software license, then install the software, tune, and test it. There is a multiplier effect (O/S versions X software versions X server-to-appliance ratio) that can make it 5 to 20 times harder to maintain and operate a scaled software solution than equivalent appliances.

In financial services, the degree to which each of these three areas of value matters depends on what problem is being solved, but in general, one can think of the value of appliances in terms of front-office vs. back-office requirements.

Regarding the front office, if you're building front office applications, time is money, and performance and profits are measured in microseconds. This leads buyers of electronic trading components to obsess about low latency, and just as importantly, consistent latency. The increasing uptake of hardware appliances as a means of accelerating front office operations is well documented with products such as the Solace Message Router, TIBCO Messaging Appliance, Exegy Ticker Plant, and XtremeData dBX. Plus, those shops most serious about latency install monitoring appliances (like the Tip-Off product from TS-Associates) so they can effectively monitor all their other hardware-accelerated appliances.

Most of the successful front office appliances have embraced specialized hardware, such as FPGAs or network processors, because just packaging software on a server and calling it an appliance doesn't offer a big jump in performance. Specialized hardware also minimizes jitter by eliminating the OS issues that can plague high-volume, low-latency software deployments.

As it relates to the back office, buying priorities shift. Of course, performance still matters, but back-office applications are generally more about reasonable performance with very high reliability and a guarantee of delivery. In the software world, it takes a lot of effort to achieve horizontal scaling with the redundancy it takes to ensure 24×7 operations. Well-designed appliances reduce the number of managed devices, and come with fault-tolerance and fail-over built in. Appliances often also provide operational visibility that spans both the hardware and software, making it easier to understand behavior and thresholds for operational planning.

In the back office, you will find a wider range of appliances since the application set is so much more diverse. Banking customers are some of the biggest buyers of some of the products mentioned in the Information Week article, such as the Oracle Exadata and IBM Netezza. Beyond that, IBM WebSphere Datapower and Layer7 XML Gateways provide SOA and web services solutions, F5 offers a range of security, storage, and load balancing appliances for retail banking, and Solace offers a high-performance guaranteed messaging appliance for accelerated transactions and WAN distribution. Whatever the appliance, or its role, the key to succeeding in the back office is in reducing costs, improving reliability, and making life easy for the operations teams.

Best Practice – For NGDC DBMS, consider utilizing appliances

In the world of the Next-Generation Data Center, the need to conform to a best practice to achieve a rich DBMS (Data Base Management System) is key. In the case of this specific application of processing technologies, different DBMS are best at different tasks.

A single relational database management system (RDBMS) can perform a broad variety of duties. It may even do them all pretty well. However, for some uses, a special-purpose product can greatly outperform general-purpose systems. Complex data warehousing is such a task.

“Index-light MPP appliances” excel at data warehousing. For most data warehouses, market-leading general purpose RDBMS are good enough. However, for complex queries against multi-terabyte data warehouses, index-light MPP data warehouse appliances are a much more efficient option. Offered by DATAllegro, Netezza, Teradata (if you use the term “appliance” a bit loosely), and IBM (if you use the term “appliance” very loosely), these systems beat their index-heavy SMP counterparts on several major criteria: performance, price/performance, consistency of performance, and administration costs.

The index-light MPP (Massively Parallel Processing) appliance story hinges on three technical factors:

1. Shared-nothing MPP. Loosely-coupled systems are significantly cheaper than tightly coupled ones, for the same level of raw component performance.
2. Reduced use of indices. By minimizing redundant references to information, index-light systems can store up to 7X less data than index-heavy ones. This produces enormous savings both in hardware and in administrative costs.
3. Avoidance of random disk reads. Disk rotation speeds have only improved 12.5-fold in the past 50 years, making random disk lookup the greatest constraint on conventional RDBMS performance. Index-light systems largely avoid this bottleneck.

Best Practice – Implement Applianceization within the infrastructure – VBLOCK Technology and Architectures

A best practice is to implement a converged infrastructure or appliance into the Next-Generation Data Center and riding the clouds. It comes in the form of EMC Vblock™ architecture from VMware, Cisco, and of course, EMC. The concept is simple, but effective. Take the best from industry leaders such as VMware, Cisco, and EMC, and combine it into an integrated platform based entirely around private cloud concepts, built differently, operated differently, consumed differently, with a single support model. Sell and support it as a whole and line up a wide ecosystem of partners. A block diagram is shown in Figure 12. The Vblock architecture consists of Cisco Unified Computing System that provides a next-generation compute platform that unites computer, network, storage access, and virtualization into a cohesive system. Cisco Nexus family of switches provide scalable, virtual machine–aware, 10 Gigabit Ethernet, and unified fabric networking that delivers the performance, flexibility, and policy control required for highly virtualized data centers. Cisco MDS 9000 family of director switches provide a scalable and extensible SAN fabric that delivers the throughput to make full use of server virtualization features and benefits. EMC Symmetrix® VMAX™ storage platform enables I/O to be processed across extended locations to support business operations. The EMC CLARiiON® storage platform can also be used and extends the benefits of virtualization from the operating system to the physical disk. EMC Celerra® Unified Storage extends the benefits of virtualization from the operating system to the physical disk in a NAS environment. VMware vSphere 4 manages collections of infrastructure holistically as a smooth, flexible, and dynamic operating environment.

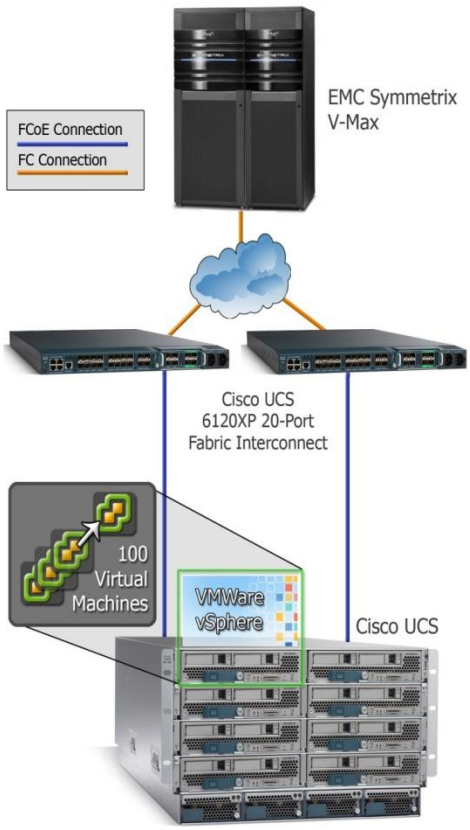


Figure 12 Vblock Architectural Diagram

A Vblock is nothing more than a new consumption option for existing advanced technologies. If you want to buy individual piece parts and assemble/integrate/support them yourself, that hasn't changed. But if you want a cohesive single solution that has been tested and configured to best practices, this is the solution that allows the path to the Next-Generation Data Center focused on private and public cloud applications. For more information on “Riding the Clouds” applied to various cloud application use cases please refer to the section titled “Vertical Markets and Use Cases for Riding the Cloud” starting on page 201.

Best Practice – Implement a Unified Management solution for Appliance Architectures

A best practice for appliance management is to have a single cohesive management architecture to address the needs to support the Next-Generation Data Center and, of course, “Riding the Cloud”. The software solution from EMC, Ionix™ Unified Infrastructure Manager (UIM), is a single point of management for Blocks. It simplifies the configuration lifecycle of Vblock resources, while ensuring resource allocation according to service requirements. UIM is

the only product that manages multiple Blocks across compute, network, and storage resources. Figure 13 shows a block diagram of the solution. As well, Figure 14 shows how one can utilize UIM as the management platform allowing IT as a service appliance. More on IT as an appliance in the next sections.

Before a Vblock is put into use, Ionox UIM can ensure it is compliant with configuration best practices and can enforce those guidelines over time. By providing an automated approach to implementing changes, it also helps enforce change management discipline. UIM can also track and report on changes to the Vblock, thereby supporting a disciplined change management process. UIM simplifies and accelerates the configuration and provisioning of Vblock network, storage, and compute resources and eliminates the need for multiple server, network and storage configuration tools, with no need for additional third-party tools to manage Cisco UCS compute resources.

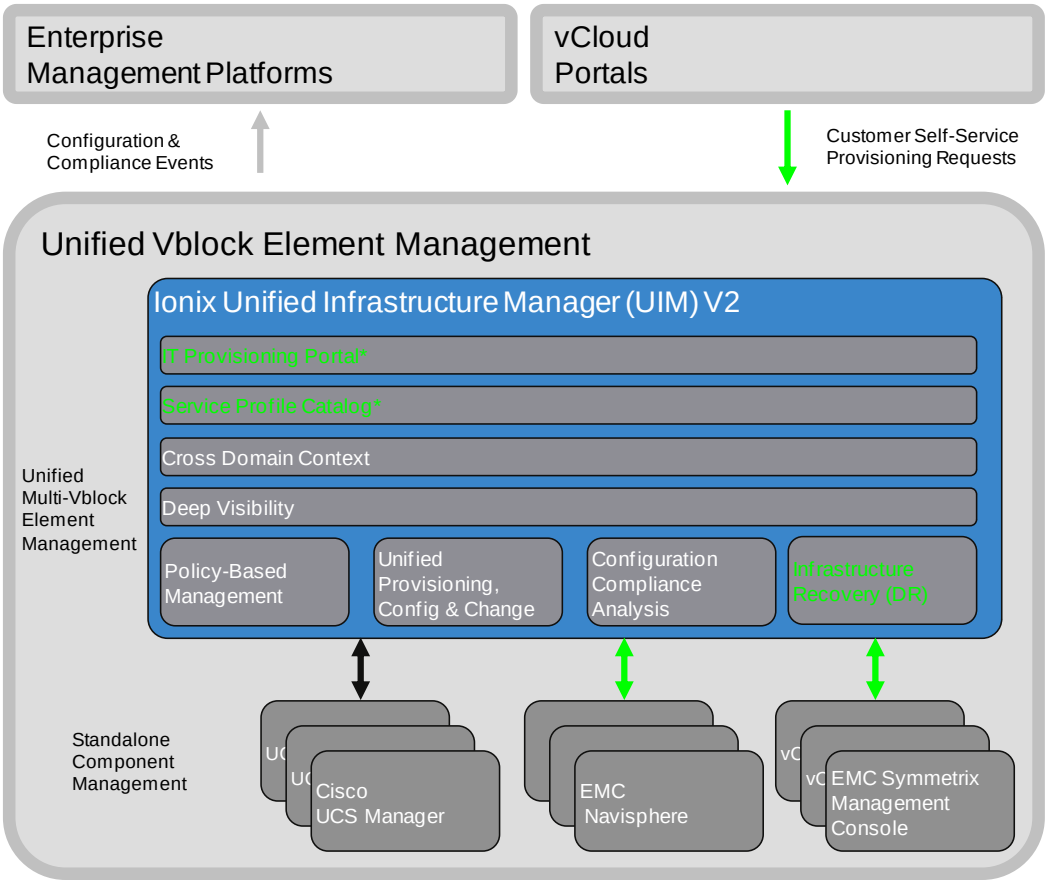


Figure 13 Ionix Unified Infrastructure Manager

Ionix UIM is not an element manager; it resides above them, interfacing with the element managers of the various domains such as XML API for Cisco UCS Manager, CLI/SNMP for the Cisco Nexus and MDS, Symmetrix Agent on the Service Processor (SMI-S), and others. This solution provides orchestration and visibility across the Vblock stack, which is needed in a cloud or NGDC architecture. Also included in Ionix Unified Infrastructure Manager 2.0 is the web-based IT infrastructure Service Catalog and Provisioning Center, and is the interface for managing the infrastructure service lifecycle. This, in turn, will create the infrastructure services, provision those services on demand, and decommission services when resources are needed for other applications.

Ionix Unified Infrastructure Manager 2.0 will also interface with EMC CLARiiON systems that are part of Vblock 1 configurations to provide storage visibility and configuration/provisioning automation. Ionix Unified Infrastructure Manager 2.0 also introduces an API that allows higher-level provisioning portals and orchestration tools to query and provision the infrastructure services managed by Ionix UIM.

Best Practice – Consider utilizing appliance structures in the cloud (IT in a box)

For businesses, faced with decreasing resources and increasing business needs, cloud computing is becoming more and more attractive, providing a more efficient, flexible, and cost-effective model for delivering IT to business: IT as a Service. To achieve this goal, many solutions are being developed that very much look like appliances or an “Infrastructure entity in a box”. While cloud computing provides the approach, VMware delivers a pragmatic path and customer-proven solutions that allow businesses to preserve existing technology investments, while achieving the goal of enabling IT as a Service.

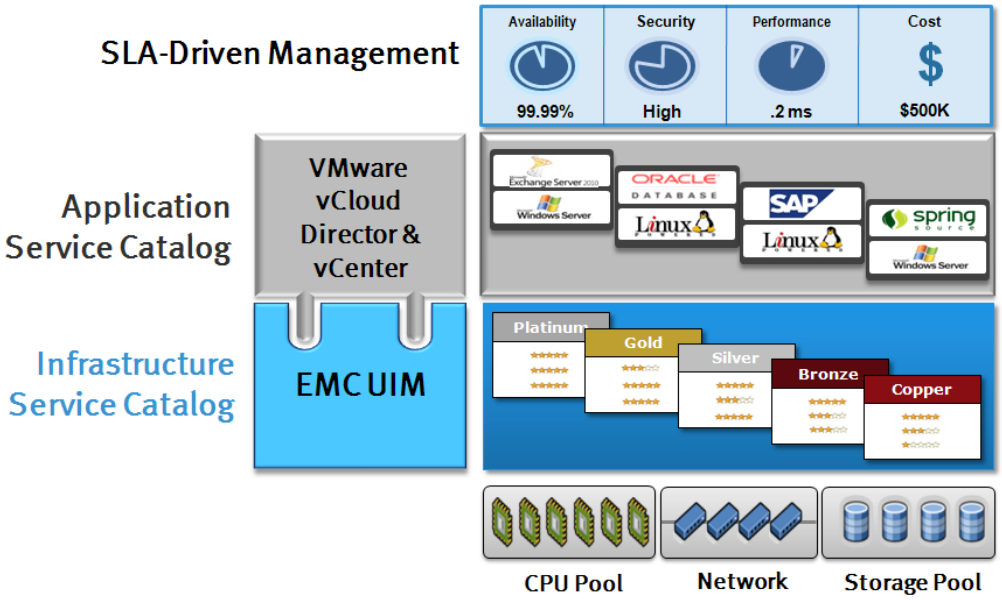


Figure 14 IT as a Service Appliance

Using appliance-based solutions from VMware as an example as shown in Figure 14, it can be shown that utilizing this type of architectural approach, one can achieve efficiency through utilization and automation. By minimizing unnecessary IT infrastructure investments, providing efficient management and maintenance tools, and preventing technology lock-in, appliance-based solutions help IT in general to adopt a more cost-effective, self-managed, dynamically optimized environment for the most efficient delivery of IT services. At the same time, Applianceization of IT assets allows IT staff to implement policies consistent with business and governance requirements, providing the control IT needs to minimize business and regulatory risk. An effective cloud strategy also means ensuring security and eliminating the need to constantly reconfigure static security infrastructure for this dynamic computing environment. Cloud computing allows IT businesses to maintain full control over the availability, reliability, scalability, security, and service level agreements (SLAs) for all workloads. After all, these attributes will enable the Next-Generation Data Center with the agility to allow business to have a sustainable growth path.

Data Transference

With the advent of Disaster Recovery requirements and the need to utilize resources from multiple locations efficiently, a continuing need to expand to Active/Active Data Centers is an attribute for the Next Generation Data Center. In addition, information and data migrations also need to be addressed. Also, a major trend is underway to enable managing the transition to the cloud.

For years, data centers have relied on “physical storage” to meet their information needs. New and evolving changes, such as virtualization and the adoption of Private Cloud computing, have placed new demands on how storage is managed and how information is accessed.

To meet these new requirements, storage must evolve to deliver capabilities that free information from a physical device to a virtualized resource that is fully automated, integrated within the infrastructure, consumed on demand, cost effective and efficient, always on, and secure. The technology enablers needed to deliver this to combine a unique set of capabilities include the ability to allow a distribution of information unbridled from the storage array. Great strides have recently been made in the venue such as EMC VPLEX technology that will be discussed in detail in the following sections.

In conjunction with VPLEX, other technology enhancements such as Fully Automated Storage Teiring (FAST), in combination with local and distributed federation lead the way towards a higher level of data transference methodologies. As it relates to data transference, the NGDC requires users to:

- Move thousands of VMs over thousands of miles
- Move workloads to batch process in low-cost energy locations
- Enable boundary-less workload balancing and relocation between sites and providers
- Aggregate big data centers from separate ones, changing the way data centers are managed and leveraged
- Deliver “24 x forever” – and run or recover applications without interruption or restart. Ever!

As previously mentioned, one technology that is a fundamental requirement to create a coherent distributed information model is the ability to distribute and disperse information over distances, allowing ease of transferring data or “Data Transference”.

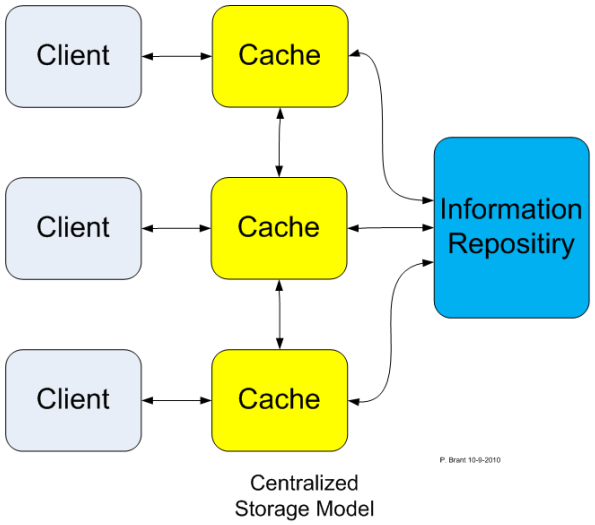


Figure 15 Centralized information Cache Coherence Model

One prerequisite is implementing some form of Cache Coherency model as shown in Figure 15. One architecture is to have the information centrally located allowing an efficient single instance of the data. Another option is to have a more distributed information structure allowing a more diverse infrastructure, achieving a better cloud architecture as shown in Figure 16. Information Repository Cache Coherence is the behavior of reads and writes to the same memory location, being consistent.

For example, a read made by a Host H to a LUN L that follows a write by the same Host H to L, with no writes of L by another host occurring between the write and the read transaction made by H, L must always return the value written by H. A read made by a processor H2 to location L, that follows a write by another processor H2 to L, must return the written value made by H2, if no other writes to L made by any host occur between the two accesses. This condition defines the concept of coherent view of storage. If hosts can read the same old value after the write made by H2, we can say that the storage is incoherent.

Writes to the same location must be sequenced. In other words, if location L received two different values A and B, in this order, by any two hosts, the hosts can never read location L as B and then read it as A. The location L must be seen with values A and B in that order.

These conditions are defined supposing that the read and write operations are made instantaneously. However, this does not happen in computer hardware given memory latency and other aspects of the architecture.

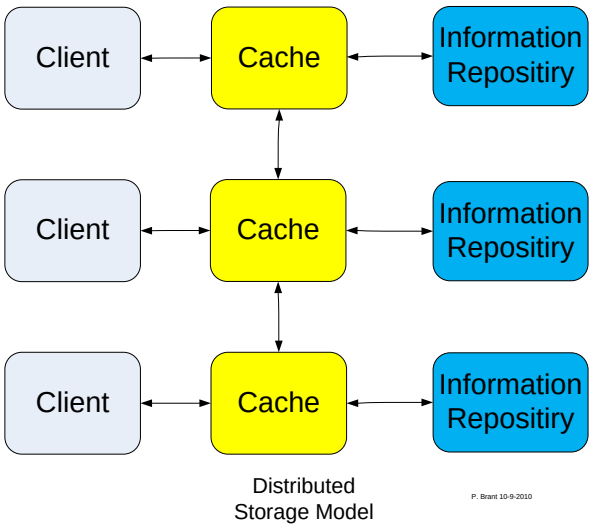


Figure 16 Distributed information Cache Coherence Model

A write by Host H1 may not be seen by a read from Host H2 if the read is made within a very short time after the write has been made. The memory consistency model defines when a written value must be seen by a following read instruction made by the other hosts. As outlined, being able to have a coherent methodology, be it host-based or storage-based, has its challenges. More discussion of this issue follows.

Best Practice – Implement Directory-based Cache Coherency to create efficient local and remote distributed Data Transference

The presence of distributed caches in current storage system designs has limitations and challenges, as previously discussed. For some time, distributed shared-memory multiprocessors have been commonplace. For servers or clusters of servers, distributed cache architectures improve performance by reducing the processor's memory access time and by decreasing the bandwidth requirements of both the local memory module and the global interconnect. The local caching of data, however, introduces a cache coherence problem. Early distributed shared-memory servers left it to the programmer to deal with the cache coherence problem, and consequently, these machines were considered difficult to implement.

Today's multiprocessors solve the cache coherence problem in hardware by implementing a cache coherence protocol. This is also true for storage system design as well, and will be discussed in subsequent sections.

There are two main classes of cache coherence protocols, snoopy protocols and directory-based protocols. Snoopy protocols require the use of a broadcast medium in the machine or cluster and therefore apply only to small-scale bus-based architectures, based on the inherent scalability issue of broadcast domains, and this limitation is valid be it server-based or storage array-based systems, or other functions. In these broadcast systems, each cache “snoops” on the bus and watches for transactions, which affect it.

Any time a cache sees a write on the bus, it invalidates that line out of its cache, if it is present. Any time a cache sees a read request on the bus, it checks to see if it has the most recent copy of the data, and if so, responds to the bus request. These snoopy bus-based systems are easy to build, but unfortunately, as the number of processors, LUNs, devices, etc., on the bus increase, the single shared bus becomes a bandwidth bottleneck, and the snoopy protocol’s reliance on a broadcast mechanism becomes a severe limitation. Simply put, it does not scale.

To address these challenges and scalability limitations, server and storage architects have adopted the Distributed Cache shared Memory (DSM) architecture. As shown in Figure 17, the DSM node contains the node function. The Node Function can be a processor or server, distributed storage arrays or other functions, a portion of the machine’s physically distributed main memory, and a node controller, which manages communication within and between nodes.

Rather than being connected by a single shared bus, the nodes are connected by a scalable interconnection network. The DSM architecture allows nodes to potentially scale to thousands of nodes, but the issue of this design is the lack of a broadcast medium that can create network storms is an inherent problem for the typical cache coherence protocols. As discussed previously, snoopy protocols are no longer appropriate, so instead, designers look elsewhere, and that is to typically use a directory-based cache coherence protocol.

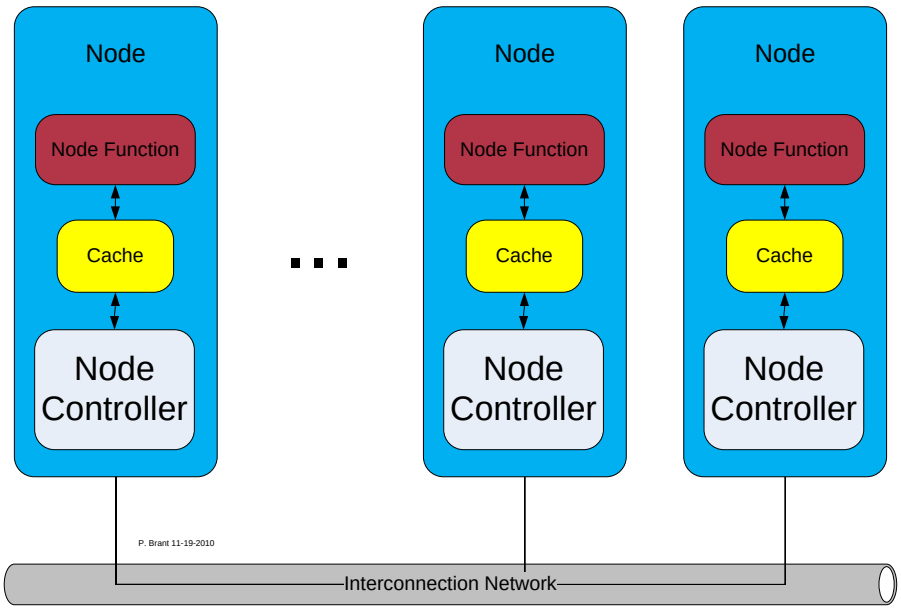


Figure 17 Distributed Cache Coherence Architecture

The directory is an auxiliary data structure that tracks the caching state of each cache line in the system. For each cache line in the system, the directory needs to track which caches, if any, have read-only copies of the line, or which cache has the latest copy of the line if the line is held exclusively. A directory-based cache-coherent node works by consulting the directory on each cache miss, and taking the appropriate action, based on the type of request and the current state of the directory. Figure 18 shows a directory-based DSM machine. The directory is distributed to eliminate the bottleneck that would be caused by a single monolithic directory. If each node's data target is divided into cache-line-sized blocks, the directory can be thought of as extra bits of state for each block of each data target. Any time a node wants to read cache data target entry D, it must send a request to the node that has the directory for line D. This node is called the home node for D. The home node receives the request, consults the directory, and takes the appropriate action. On a cache read miss, for example, if the directory shows that the line is currently un-cached or is cached read-only (the line is said to be clean), then the home node marks the requesting node as a sharing entity in the directory and replies to the requester with the copy of line L in main memory. If, however, the directory shows that a third node has the data modified in its cache (the line is dirty), the home node forwards the request to the remote third node, and that node is responsible for retrieving the line from its cache and responding with the data. The remote node must also send a message back to the home indicating the success of the transaction.

Even though the methodology is straightforward, implementing a full cache coherence protocol in a node with distributed memories and distributed directories is not trivial. Because the only serialization point is the directory itself, races and transient cases can happen at other points in the system, and the cache coherence protocol is left to deal with the complexity.

There are two major components to every directory-based cache coherence protocol; the directory matrix and the set of message types and message actions. The directory matrix refers to the data structures used to store the directory information and directly affects the number of bits used to store the sharing information for each cache line. The memory required for the directory is a concern because it is “extra” memory that is not required by non-directory-based machines. The ratio of the directory memory to the total amount of memory is called the directory memory overhead. A best practice of the system designer would like to keep the directory memory overhead as low as possible, and would like it to scale very slowly with machine size. The directory matrix also has ramifications for the performance of directory accesses, since some directory data structures may require more hardware to implement than others, have more state bits to check, or require traversal of linked lists, rather than more static data structures.

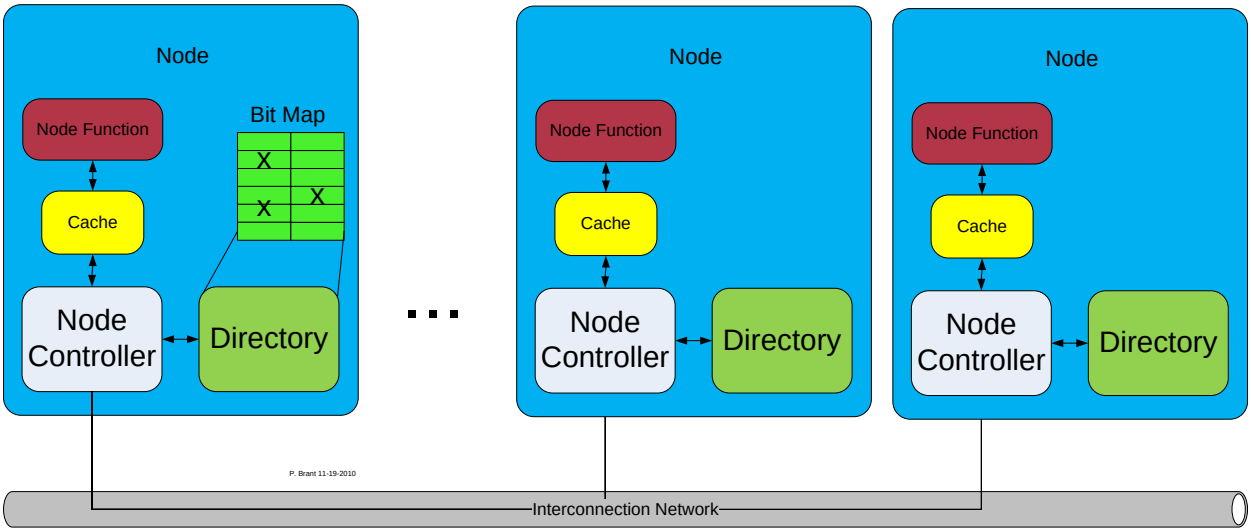


Figure 18 Distributed Cache Directory Coherence Architecture

The directory matrix holds the state of the cache coherence protocol, but the protocol must also send messages back and forth between nodes to communicate protocol state changes, data requests, and data replies. Each protocol message sent over the network has a type or opcode (Operational Code or Command Code) associated with it, and each node takes a specific action based on the type of message it receives and the current state of the system. The set of

message actions includes reading and updating the directory state as necessary, handling all possible race conditions, transient states, “corner cases” in the protocol, composing any necessary response messages, and correctly managing the central resources of the machine, such as virtual lanes in the network, in a deadlock-free manner. To do this, however, as previously indicated, is non-trivial. A best practice is to avoid “machine deadlock” scenarios, since the actions of the protocol should be focused on implementing an avoidance strategy. It is very easy to design protocols that will livelock, deadlock or livelock[10]. It is much more complicated to design and implement a high-performance protocol that is deadlock-free.

Best Practice – Implement Directory-Based Cache Coherency for Data Federation

Taking into account the architectural trade-offs, limitations, and challenges, implementing a "Directory Cache Coherence" design as discussed previously, is considered a best practice. EMC agrees with this strategy outcome and best practice and as a result, EMC implemented a design of this type. EMC’s solution to Data Federation Directory Based Cache Coherency is called “VPLEX”, shown in Figure 19, is a solution offering utilizing hardware and software that allows storage federation that extends storage beyond the boundaries of the data center. In the current release, the distributed network is a SAN and it presents hosts with a federated view of both EMC and other heterogeneous storage. It is anticipated that in future releases of this VPLEX technology, distances between storage nodes or VPLEX engines (see the definition of engines below) will be extended and many types of transport protocols will be supported.

The VPLEX environment is very dynamic and uses a hierarchy to keep track of where I/O goes, as defined in the architecture discussions previously. An I/O request can come in from anywhere and it will be serviced by any available engine in the VPLEX cluster. VPLEX abstracts the ownership model into a high level directory that’s updated for every I/O and shared across all engines. An engine, which is based on the VMAX Intel hardware platform, implements the functionality analogous to the node controller, node function, cache, and directory outlined in Figure 18.

[10] Deadlock, Livelock, and Starvation, describe undesirable situations involving blocking or not making progress of processes in a concurrent program or multi threaded system.

The directory uses a small amount of metadata and tells all other engines in the cluster, in 4k blocks, which block of data is owned by which node or engine, and at what time. The communication that actually occurs is much less than the 4k blocks that are actually being updated.

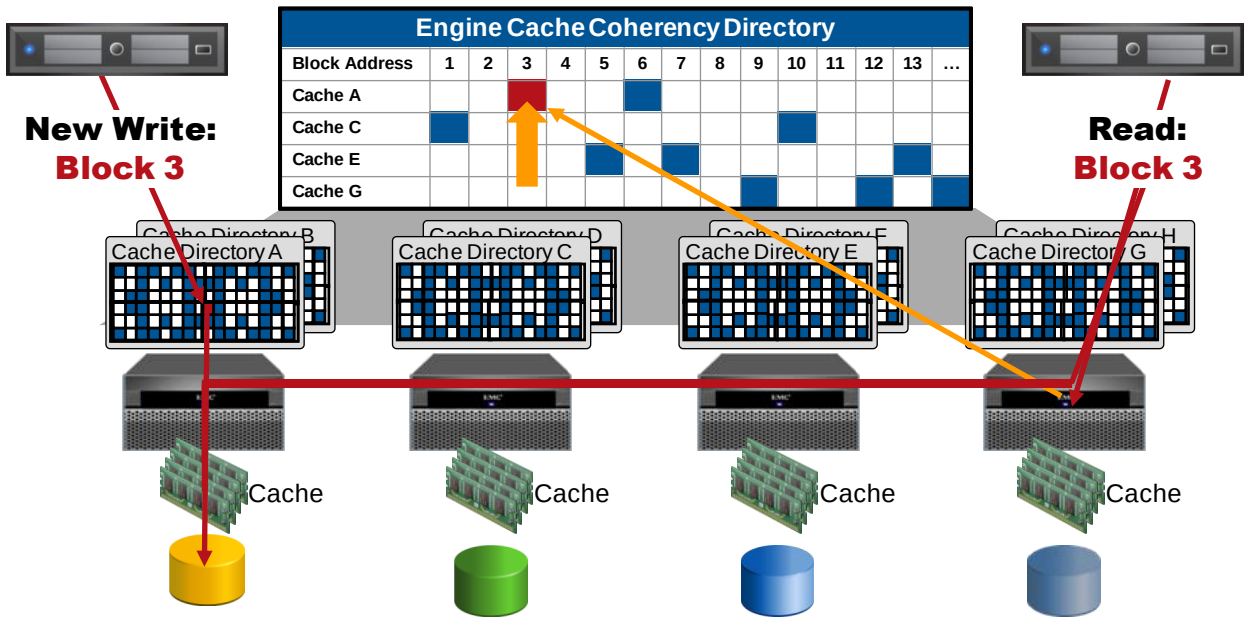


Figure 19 EMC VPLEX: The World's First Local and Distributed Storage Federation Platform

If a read request comes in, VPLEX automatically checks the directory for an owner. Once the owner is located, the read request goes directly to that node or engine.

Once a write is done, and the table is modified, if another read request comes in from another engine, it checks the table and then can pull the read directly from that engine's cache. If it's still in cache, there is no need to go to the disk to satisfy the read. This model also enables VPLEX to stretch the cluster. As a result, one can distribute this directory between clusters and, therefore, between multiple sites. Given its directory-based cache architecture, and its high performance distributed network, VPLEX has minimal overhead, is very efficient, very scalable, and enables communications to be routed simply over distance.

Technologies such as VPLEX and FAST, allowing enhanced levels of data transference functionality, enables the Next-Generation Data Center and other cloud-related solutions to meet the growing demand of achieving true transformation practices for the next 100 to 1000 years.

Object Affinity

In order to address the needs of the data center for the next 100 years, one must address the new technologies and architectures being driven by the IT community recently. Cloud computing is one of these technologies. Cloud computing has been driven by new offerings of computing resources that are attractive due to per-use pricing and elastic scalability, providing a significant advantage over the typical acquisition and deployment of equipment that was previously required. The effect has been a shift to outsourcing of not only equipment setup, but also the ongoing IT administration of the resources as well. The needed changes to applications, in order to take advantage of this model, are the same as those required for server consolidation. These changes have already been taking place for several years prior to the advent of the Cloud. The increased resource utilization and reduction in power and cooling requirements achieved by server consolidation are now being expanded into the cloud.

The new technologies underlying this is the system level virtual machine, that allows multiple instances of an operating system and associated applications to run on a single physical machine. Delivering this over the network, on demand, is termed Infrastructure as a Service (IaaS). The IaaS offerings on the market today allow quick provisioning and deployment of applications and their underlying operating systems onto an infrastructure that expands and contracts, as needed, to handle the load. The resources that are used can be better matched to the demand on the applications.

One method of enhanced manageability is leveraging IaaS offerings that typically provide an interface that allows the deployment and management of virtual images onto their infrastructure. The lifecycle of these image instances, the amount of resources allocated to these instances, and the storage they use can all be managed through these interfaces. In many cases, this interface is based on REST (REpresentational State Transfer) HTTP operations. Without the overhead of many similar protocols, the REST approach allows users to easily access their services. Every resource is uniquely addressed using a Uniform Resource Identifier (URI).

Based on a set of operations that are object-based include commands such as – create, retrieve, update and delete – all of which can be managed. Currently, three types of resources are considered: storage, network, and compute resources. Those resources can be linked together to form a virtual machine with assigned attributes.

Best Practice – Standardize Cloud Computing Interfaces that is object-based and vendor-neutral

Having a programmable object-based interface to the IaaS infrastructure means that you can write client software that uses this interface to manage your use of the cloud. Many cloud providers have licensed their proprietary APIs freely, allowing anyone to implement a similar cloud infrastructure. Despite the accessibility of open APIs, cloud community members have been slow to uniformly adopt any proprietary interface controlled by a single company. The Open Source community has attempted responses, but this has done little to stem the tide of API proliferation. In fact, Open Source projects have increased the tally of interfaces to navigate in a torrent of proprietary APIs.

What is needed, instead, is a vendor-neutral, standard API for cloud computing that all vendors can implement with minimal risk and assured stability. This will allow customers to move their application stacks from one cloud vendor to another, avoiding lock-in and reducing costs.

Best Practice – Implement an Object-Based Open Cloud Computing Interface into NGDC designs

The Open Grid Forum[11] has created a standardized interface to the cloud. The Open Cloud Computing Interface (OCCI) is a royalty-free, open, community consensus-driven API, targeting cloud infrastructure services. The API shields IT data centers and cloud partners from the disparities existing between the lineup of proprietary and open cloud API's, as shown in Figure 20.

[11] http://www.gridforum.org/

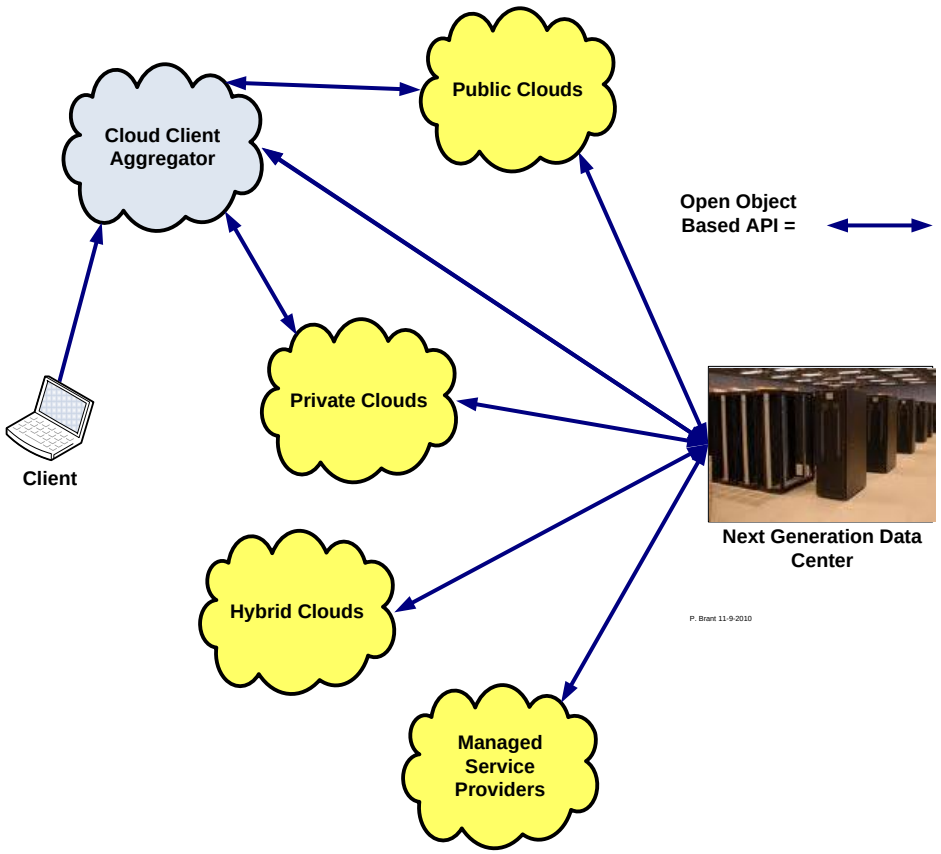


Figure 20 Object-Based Cloud Computing API

A Resource Oriented Architecture (ROA) has been defined as representing key components, comprising cloud infrastructure services. Each resource (identified by a canonical URI) can have multiple representations that may or may not be hypertext (e.g. HTML) planning mappings of the API to several formats, including Atom/Pub, JSON, and Plain Text. A single URI entry point defines an OCCI interface.[2] Interfaces expose "nouns" which have "attributes" and on which "verbs" can be performed. Figure 21 shows how the components of an OCCI URI align to IaaS resources:

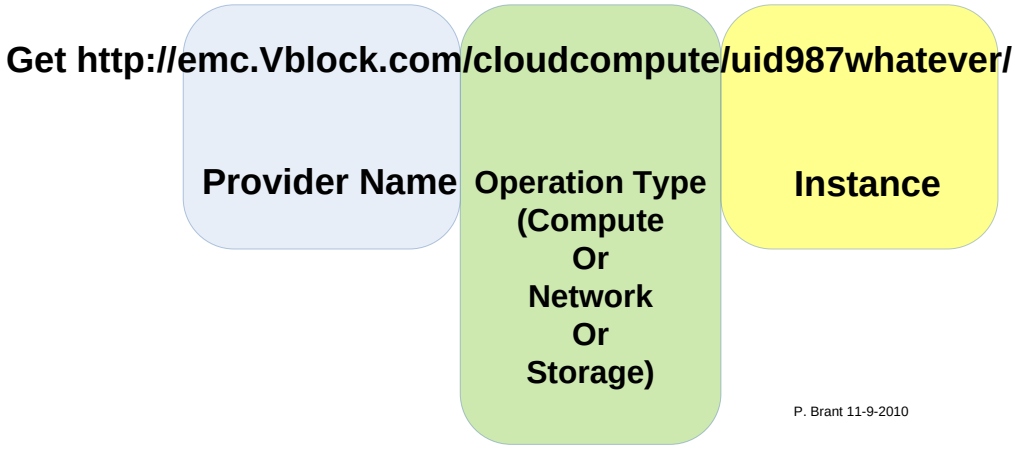


Figure 21 Components of an OCCI URI

Attributes are exposed as key-value pairs and the appropriate verbs as links. The attributes may be described as a URI. Adopting URI support affords the convenience of referencing (linking to) other interfaces, including SNIA's Cloud Data Management Interface (CDMI), for example. The API implements CRUD operations: Create, Retrieve, Update, and Delete. The operations can also be linked to address the three levels of abstraction of the IaaS model, as shown in Figure 22.

Each is mapped to HTTP verbs POST, GET, PUT, and DELETE, respectively. HEAD and OPTIONS verbs may be used to retrieve metadata and valid operations without the entity body to improve performance. All HTTP functionality can take full advantage of existing Internet infrastructure including caches, proxies, gateways, and other advanced functionality. All metadata, including associations between resources is exposed via HTTP headers (i.e. the Link: header). The interface, natively expressed as Atom, executes as close as possible to the underlying HTTP.

Figure 22 Object-Based Linked Resource Operations Example

OCCI provides the capabilities to govern the definition, creation, deployment, operation, and retirement of infrastructures services. Using a simplified service life cycle model, it supports the most common life cycle states offered by cloud providers. In the event providers do not support or report service life cycle states, OCCI does not mandate compliance, defining the life cycle model as only a recommendation. Cloud providers, wishing to do so, can comply with the OCCI service life cycle recommendations.

With OCCI, cloud computing clients can invoke a new application stack, manage its lifecycle, and manage the resources that it uses. The OCCI interface can also be used to assign storage to a virtual machine, in order to run the application stack such as that exported by SNIA's CDMI interface.

Security

With advances in IT deployment such as cloud architectures, as well as initiatives about the Next-Generation Data Center for the next 1000 years (the author is getting aggressive), security is and has always been, a major contributor to the adoption and longevity of various IT initiatives. For a more detailed discussion on “Cloud Security”, please refer to the 2010 EMC Proven Professional Paper Knowledge Sharing article, “How to Trust the Cloud – “Be careful up there””, by Paul Brant and Denis Guyadeen.

Extending beyond the cloud security issues, there is a much broader issue, as it relates to not only infrastructure security, but also personal security[3]. With the advent of the eventual migration to a more “Personal Data Store”-centric approach on how to manage and protect one’s individual information, implementing a strong authentication process should be put in place. For more information on the “Personal Data Store”, please refer to the section titled “Personal Data Store Attributes”, starting on page 82.

Best Practice – Implement Identity assurance into the Personal Data Store Architecture

As part of the same process of interacting and transacting, businesses in both the private and public sector need to be confident that the person they are dealing with is who they say they are. Usually, this assurance is given by an agreed ‘gold standard’ piece of identification such as a passport or bank account. Personal Data Stores (PDS) can help streamline these identity assurance processes by linking verifications to such data. But PDS can take identity assurance much further. With a PDS, the individual can provide a wide range of identifiers in addition to traditional ‘gold standard’ pieces of identity assurance. For example, as well as the passport/driving license, individuals could also provide evidence that they have banked with this bank for 15 years, using this address for six years, and received home delivery groceries and Amazon shipments at this address for five years, have been a member of the following online communities for three years, and so on. The greater the combination of different such data sources, the harder it is for fraudsters to succeed.

In terms of privacy management, much of the current debate about privacy is misconceived because it is based on organization-centric assumptions, i.e. what ‘privacy policies’ businesses should or should not confer on individuals whose data they collect. PDS transform this debate by recognizing privacy as a personal setting, where the individual is empowered to choose what

information he or she wishes to share with what other party, for what purposes, in what context. It recognizes the contextual, contingent nature of all privacy concerns.

Best Practice – Implement PDS to enable the Cloud in an ever-changing social media world

The trend toward PDS is now evident in countless ways already, most of them only suggestive of the future, or displaying some flaws or failings. For example, it's now commonplace for e-commerce companies to provide customers access to "My Account" facilities where they can update records, manage communication preferences, access transaction histories, and so on. Such facilities involve individuals as managers of their own information, but in an organization-centric way – on the organization's systems, in ways that require individuals to spend time and hassle, logging into and jumping through the security and identity hoops of each different organization. The next step is for such "My Account" information from all "my" suppliers to be held on an individual's own PDS, so that I can manage them all together, from one place, in a time-efficient, safe, integrated way.

With the advent of social services like Facebook, we all are creating a database about ourselves and selectively or, even more frightening, unknowingly disclosing what information we permit others to see. From the PDS perspective, the drawbacks of the Facebook approach are obvious; the data is held by Facebook, not the individual, and Facebook determines (and keeps changing) 'the privacy policy'. But, it underlines the value of the data and educates the public on the concept of personal data management and sharing (including the pitfalls!). Meanwhile, in specific industry sectors such as healthcare, proto-personal data stores are being developed by many companies. Microsoft's "Healthvault", for example, looks forward to a day when individuals manage their own personal health records.

This pace of change is accelerating and rapidly gaining critical mass. Cloud computing refers to when data and applications that are normally stored and operated either on personal computers or corporate servers are instead stored and operated on third party servers "in the cloud", i.e. accessible from any location or device over the Internet. This has many benefits in terms of cost savings, reduced complexity, simplified administration, and sharing of data across domains and applications. It is also a necessary step toward making information services less 'device-centric' and more 'person-centric'. Business Week predicts, for example, that cloud computing will

trigger a proliferation of 'personal virtual assistant services'[12], which are perfect vehicles for the generation of rich, structured volunteered information.

Best Practice – Implement Federated login and password for Personal Data Stores

One solution is OpenID[13] that turns the current businesses-centric approach to identity on its head. Currently, every business assigns its own identifier to every individual it deals with, which means that individuals dealing with many different businesses have to remember to use the multiple different identifiers allocated to them. With OpenID, individuals have one, single identifier that they can 'carry' around with them whenever they go online so, for example, they no longer have to bother with multiple usernames and passwords. This is much more than a compelling consumer proposition. It makes the individual, not the organization, the 'pivot point' of data sharing.

Information Cards (I-Cards)[14] build on these technologies to make information sharing even easier. I-Cards are the digital equivalent of the cards you hold in your wallet: they contain information (from yourself, or from other websites) that you can use to prove your identity or share information about you. Instead of sitting in your wallet, however, I-Cards sit in a digital wallet, which is accessible from all your devices. If a website accepts I-Cards for login, you just click their I-Card icon and then select the I-Card you want to use. One click and you are logged in: no typing usernames or passwords at all! Cards are always transmitted with strong encryption and disclose only the personal information needed for any particular transaction, so they protect both security and privacy. As the non-profit Information Card Foundation (ICF) explains, I-Cards "make routine Internet transactions—logins, form-filling, purchases, reference checking—as easy as swiping a credit card". There is no pre-defined limit to the amount of information that can be shared this way.

---

[12] Tech Beat Hey YouOS! – BusinessWeek". www.businessweek.com. http://www.businessweek.com/the_thread/techbeat/archives/2006/03/hey_youos.html. Retrieved 2007-12-12.

[13] http://openid.net/

[14] http://informationcard.net/

Best Practice – Embrace Identity methodologies to Enhance Security

I-Cards can also be combined with XDI data sharing to create what are known as Relationship Cards (R-Cards). An R-Card is an I-Card that is shared in order to create an ongoing data sharing relationship. For example, you could use an R-Card to share your mailing address with a magazine site so that it was always delivered to your home no matter where you live. For example, when you move from one house to another, you only need to update your mailing address once, and your R-Card will automatically send the updated address to each subscriber you have approved.

The Open Identity Exchange (OIX)[15] is about trust at Internet scale. Open identity technologies like OpenID and Information Cards, reduce the friction of using the Web, much like credit cards reduce the friction of paying for goods and services. However, they also introduce a new problem. The question is who do you trust? How does a relying party know it can trust credentials from an identity service provider without knowing if that provider's security, privacy, and operational policies are strong enough to protect the relying party's interests? OIX is working on this, not just as a technology problem, but also as a business, legal, and social problem.

XRI (often known as i-names) is a new type of Internet identifier that enables both people and machines to 'tag' pieces of information, so that it can be located, described, and understood in a way that works across different domains and applications. For example, with a URL, the most common form of Web address today, you can only tell (at most) that it represents a file of a certain type (a Web page, a Word document, spreadsheet, a PDF file, an image, etc.) With an XRI, you can determine if the address represents a person, a company, a concept, etc. If the XRI represents a file, such as an Excel spreadsheet, it can tell you, for example, that it is the third version of a budget produced by a company's Human Resources department.

XDI is a new data sharing protocol based on XRIs that makes it possible to securely share, link, and synchronize data between any two devices or applications anywhere on the Internet. A key feature of XDI is 'link contracts' that enable control over the authority, security, privacy, and rights of shared data to be expressed in a standard machine-readable format. In the context of

[15] http://openidentityexchange.org/

VPI, this means that we now have scalable, worldwide infrastructure for individuals to share their personal information on terms and conditions to which they agree, and these terms and conditions will always travel 'with' the data in a secure, auditable way. Drawing on these developments, other bodies have emerged to advance work in particular areas.

Best Practice – Consider Containerization to determine who you should trust with your data

It is difficult these days to know whom to trust. This is true in the short and medium term but is even more risky in the long term. Of particular interest to a project, such as CASPAR, is the question of long-term preservation of the digitally encoded information on which we increasingly rely and yet which is inherently fragile. For more than a decade there have been demands for some way to certify digital repositories. OAIS included this in its roadmap of follow on standards. As a result, RLG and NARA organized a Task Force which produced the Trusted Repository Audit Checklist (TRAC). TRAC has been used as the basis of the new draft standard, which will be submitted to ISO soon. For more technical information on "Containerization", please refer to the section titled "Best Practice – Implement containerization into the Next-Generation Data Center's information management processes, starting on page 131.

Virtualization

Virtualization within the Next-Generation Data Center is a very important enabler required to achieve the efficiency, scalability, and performance transforming the data center and riding the clouds. This section will outline use cases that cover the two basic natures of cloud computing delivered resources and services. They are use cases involving physical allegories (servers, disks, network segments, and so on) and a use case involving abstract allegories (blob storage functions, message queue, email functions, multicast functions, and so on), as it relates to Virtual Machine Instantiation and Mobility.

One important virtualization use case to note in the NGDC is the “Intercloud”. Please refer to the section titled “The Inter-cloud - Interconnected Global Connectivity, starting on page 214 and the following section for more details on this topic.

Best Practice – Consider Virtualization considerations when implementing a Cloud or Intercloud

One of the most basic resources, which cloud computing delivers, is the virtual machine (VM), a physical allegory type of resource. One way or another, a subscriber requests the provisioning of a particularly configured VM with certain quantities of resources, such as memory processor speeds and quantities. The format of this request varies widely by cloud computing platform and is somewhat specific to the type of hypervisor (the virtualization layer of the operating system inside the cloud computing platform). In a few seconds, they receive pointers and credentials with which to access it. The pointers are usually the MAC and IP addresses and sometimes, a DNS name given to the VM. The credentials are usually a pair of RSA keys (a public key and a private key, which one uses in the API to speak with the VM). Most often, the VM presents an x86 PC machine architecture. On that VM, one boots a system image yielding a running system, and uses it in a similar manner as one would use a running system in your own data center.

VM Mobility is a feature in some hypervisor’s, such as VMware, which allows a running system to be “moved” from one VM to another VM. As far as the running system is concerned, it does not need to be reconfigured; all of the elements such as MAC and IP address and DNS name stay the same; any of the ways storage may be referenced (such as a World Wide Name in a SAN stay the same. Whatever needs to happen to make this work is not the concern of the running system.

VM Mobility has been implemented with several hypervisors but there are limitations. Usually these limitations are a result of the "scope" of applicability of the network and storage addressing. Typically, VM Mobility is restricted to a Layer 3 subnet and a Layer 2 domain (for VLANs), because the underlying network will support the VM operating outside of the local scope of those addresses. In contrast, the network-addressing scheme in a cloud operated by an entirely different service provider is not only a different subnet but also a different class B or class A network altogether. Routers and switches simply would not know how to cope with the "rogue" running system.

Another aspect is that the instantiation instructions of the VM for the running system are very specific to that cloud computing platform and the hypervisor it uses. We would want to re-issue some of these instructions to the new cloud, so that the VM it delivered onto which the VM would move, was as suitable as the first VM, which was provisioned for us. If the new cloud takes an entirely different set of instructions, this is another barrier to VM Mobility.

All of this assumes that in the universe of cloud computing systems out there, one is able to find another cloud ready, willing, and able to accept a VM mobility transaction with the other cloud. In addition, establishing an appropriate medium and being able to have a reliable conversation with that cloud, perhaps exchanging whatever subscription or usage related information which might have been needed as a pre-cursor to the transaction. Finally, having a reliable transport on which to move the VM itself is required.

Best Practice – Implement the NGDC with VM metadata abstraction considerations using OVF

Most cloud computing implementations have a capability to deliver a VM "on demand" to a subscriber, who requests the provisioning of a particularly configured virtual machine with certain quantities of resources. At that point, the VM is "booted" with an image (or via instructions) to result in a running system.

The metadata, which specifies the image or the system, is a crucial abstraction, which is at the center of VM interoperability, a key feature for Intercloud. One would like to see an open, secure, portable, efficient, and flexible format for the packaging and distribution of one or more virtual machines to this end.

A best practice is to implement virtual image, which relies on an XML descriptor to create virtual machines from virtual machine images. In general, a virtual machine image consists of the XML descriptor (usually in a file image.xml) and a number of files for the virtual machine's disks. The virt-image tool defines a simple XML format, which can be used to describe a virtual appliance. It specifies things such as minimum recommended RAM and VCPUs, the disks associated with the appliance, and the hypervisor requirements for booting it.

The resultant XML format is describing a specific deployment of a virtual machine on a specific hypervisor. For more general interoperability, a best practice is to embrace another proposed standard.

Open Virtualization Format (OVF) is a platform independent, efficient, extensible, and open packaging and distribution format for virtual machines. OVF is virtualization platform-neutral, while also enabling platform-specific enhancements to be captured. Even though VMware was the original creator of OVF, there is also an Open-source library and tools to support it.
There is much work to do in this area. AWS for example, support their own format called an Amazon Machine Image (AMI), and although the Xen community has worked on OVF, the KVM community is just starting to. We are encouraged by the possibility of convergence of this space on OVF by recent open source conversion utilities such as “Thincrust virt-convert” which are a proof point that VM meta-data for instantiation and mobility, can be solved eventually.

Once a VM is packaged for deployment, one must be able to talk to the VM to control them (for Mobility, for example). Most virtualization systems do not allow for direct communication to the VM rather, they provide API’s to their management toolsets. For example, this is the case with VMware.

Best Practice – Consider Client and Server Virtualization into the NGDC

With costs at the forefront of the discussion for nearly all CIO’s, IT management will seek to leverage virtualization to optimize their IT infrastructure transforming into the Next-Generation Data Center. Client and server virtualization technology provides proven cost savings and demonstrated improvements to the performance, availability, and security of applications and is a key enabling technology for most organizations.

IT organizations need to embrace both client and server virtualization in their data centers to make more efficient use of resources, improve availability, assist in security and disaster recovery measures, and centralize support and administration. Virtualization allows the abstraction of physical infrastructure from operating systems, applications, and services and has changed the approach of organizations to data center design and operation. Client or desktop virtualization borrows from the traditional thin-client model, but is designed to give system administrators and end users the best of both worlds, enabling system administrators to host and centrally manage virtual and/or physical desktop machines in the data center, while giving clinical end users the traditional PC desktop experience to which they have become accustomed.

Virtual desktop infrastructure (VDI) is a variation of the client/server model where individualized desktops are maintained on a central machine, thus reducing the complexity of managing multiple applications running on numerous workstations and providing end-user support. User provisioning is also simplified, making it easier to add new users. VDI can support increased service level demands with fewer resources by centralizing management, security, and control. Within the healthcare environment, VDI enables single sign-on (SSO) and the ability for a user session to follow users as they move from individual work areas, thus streamlining secure access to business critical information in a highly mobile way. Since data is stored on the centrally managed server and not local devices, the risk of a security or privacy breach of protected health information as the result of a lost or stolen laptop, tablet, or other mobile device is essentially eliminated.

Best Practice – Utilize Virtualization to address cost reduction and cost avoidance in the NGDC

Another IT benefit associated with virtualization in transforming into the Next-Generation Data Center and cloud technologies can be summarized and outlined as follows. A best practice is to address cost reduction and cost avoidance. Virtualization significantly reduces IT infrastructure, operational costs, and provides opportunities for energy savings by addressing the following.

- A best practice is to address capital costs management. Minimizing the number of physical servers, which lowers hardware acquisition and maintenance costs, saves space in the data center and results in a clear return on IT's operational investment.
- Operational costs. Additional operational cost savings are derived from the ability to easily update/upgrade applications and add new users.

- A best practice is to address energy savings scenarios. Energy savings from virtualization can come from decreased energy consumed by idle servers as well as reduced cooling needs and space requirements, with fewer servers in the data center. Additionally, public perception associated with organizations that are trying to be more "green" adds an intangible plus.
- A best practice is to address security. Virtual environments are easier to secure. Sensitive data is not resident on the client machine as it relates to VDI; it resides instead in a single location in the data center. This reduces the vulnerability to intrusion or unauthorized copying of information. Security, compliance, and control of information is also enhanced.
- A best practice is to embrace performance enhancements, resulting in virtualization. When peak demands are encountered, the ability to dynamically add processing power with virtualized clients enables significant reduction in processing time. The adoption of virtualized desktop infrastructure as a horizontal solution has clear utility advantages for various use cases and applications.
- A best practice for the Next-Generation Data Center is to enhance availability beyond current standards. Virtualization mitigates unplanned outages and improves business continuity by enabling automatic switchover to working resources in the case of an outage, which is critical in demanding real-time application delivery models. This approach enables many more options for automating business continuity strategies.
- A best practice for the Next-Generation Data Center is to enhance accessibility. For end users, the ability to dynamically address mobile requirements can ease the integration of new tools into their workflow, support more choices of endpoint devices, and help accelerate formal and standard access.
- A best practice for the Next-Generation Data Center is to consider transparency and visibility. Virtualization provides a comprehensive view across all the physical and virtual layers and into infrastructure components, such as storage arrays, routers, switches, firewalls, and hypervisors, simplifying compliance, resource monitoring, and troubleshooting.

Best Practice – Understand the benefits of Virtualized Service-Based Delivery

Time-sensitive IT implementations have stressed the physical and human resources of IT departments. IT professionals are under pressure to deliver complex applications with high performance and security, while tight budgets demand cost reduction initiatives. Consequently,

IT professionals are seeking service-based offerings that reduce the infrastructure burden on their organizations and at the same time reduce operating costs and the associated capital investments.

As a result, several important benefits are helping to drive interest in service-based delivery of applications and storage that offer built-in security and cloud-based implementation models. They include:

- A best practice for the Next-Generation Data Center is transitioning from capital to operating expenses. Cloud services typically require minimal up-front investment, demand lower start-up costs, and have regular monthly subscription fees that are usage-based. This shift from capital to operational expenditures frees up capital budgets for investing in meaningful use technologies and innovation.
- A best practice for the Next-Generation Data Center is to consider agility and scalability in a virtualized environment. The provision of computing, networking, and storage services in a utility-style manner provides a complete set of integrated resources that can be quickly deployed, made immediately available, provide a robust and reliable level of responsiveness, and deliver both cost-effectiveness and the ability to rapidly scale.
- A best practice for the Next-Generation Data Center is virtualization system manageability. Cloud providers usually offer system and application management software that supports rapid self-service provisioning and configuration and usage monitoring. Often this includes the ability to automatically fix software faults and "spin up" replacements, which means that the user experience does not change even if overloading, hardware problems, or mis-configurations are detected in existing systems. Human intervention is not typically needed for these events, which keeps operations flowing consistently.
- A best practice for the Next-Generation Data Center is understanding the issues of virtualization and how it relates to cloud Security. While security is often cited as a challenge, new SaaS or SaaS-based applications have stronger security models than many older legacy applications. Today, many cloud providers offer multiple types of predefined service level agreements and compliance policies to ensure that data security concerns are addressed. For additional information on virtualization and how it relates to cloud security, please refer to the 2010 EMC Proven Professional Knowledge Sharing article “How to Trust the Cloud – “Be careful up there””, by Paul Brant and Denis Guyadeen.

- A best practice for the Next-Generation Data Center is considering availability and stability within the cloud. Cloud computing architectures provide for dynamic provisioning of resources, which enables information migration to other points in the cloud on demand. The benefit is that one event or anomaly will not take down an entire system and will improve information flow and operational stability. Cloud implementations must also include backup and data recovery where information is backed up automatically by the primary system to the cloud environment. Information can be sourced from multiple locations, but stored centrally in the cloud.
- A best practice for the Next-Generation Data Center is considering cost reduction and cost avoidance. In a cloud environment, applications and services can safely run on commodity servers, which gives business the ability to retire and/or repurpose some of their most powerful (and expensive-to-maintain) servers and substantially reduce overhead costs.

Personal Data Store Attributes

In terms of the next generation data center, especially moving to and riding the clouds, one needs to understand the requirements as it relates to personal information. All aspects of information management need to be considered when data is located within the NGDC and external to it.

Best Practice – Implement Personal Data Stores into the NGDC architecture

Why do we need to address personal data? Individuals as well as businesses have a complex need to manage personal information over a lifetime and beyond, and one can argue that the tools we have at our disposal today to do so are inadequate. Existing tools include the brain. I have one, but it does not have enough RAM, onboard storage, or an Ethernet socket. As for storage, we all have stand-alone data stores such as paper, spreadsheets, and phones, which are good, but not connected in secure ways that enable user-driven data aggregation and sharing.

With respect to supplier-based data stores such as cloud SaaS services, which can be tactically a great solution but execute under supplier-provided terms and conditions, has limitations and GRC (Governance, Risk Management and Compliance) issues. For a more detailed discussion on “Cloud Security” as it relates to GRC, please refer to the 2010 EMC Proven Professional Knowledge Sharing article “How to Trust the Cloud – “Be careful up there”. Our current perception of ‘personal data stores’ is shaped by the good ones that are out there such as an individual’s online bank, ones online health database, and others. What we all need is that functionality and a whole lot more.

Personal Data Store is an evolving term and anthology to describe a complex set of functions that will be discussed, but it is a start. This is a somewhat simplistic term that masks a lot of complex activity. One analogy is as used in previous sections as the ‘data warehouse’. For more information on Data Warehouse architectures, please refer to the section titled “Data Warehouse – Big Data, starting on page 162.

A Personal Data Store can take two basic forms:

- Operational Data Stores that gets things done, and only needs to store sufficient breadth and depth of data to fulfill the operation for which they are built (i.e. pay a credit card bill, book a doctor’s appointment, and order my groceries).

- Analytical Data Stores underpin and enable decision-making, and typically needs a more tightly defined, but much deeper data set that includes data from a range of aspects of life rather than just that from one specific operation (i.e. plans a home move, buys a car, organizes an overseas trip).

A sub-set of the individual's overall data set requirement can include both of the above. In both cases, the functionality required is to source, gather, manage, enhance, and selectively disclose data (to presentation layers, interfaces, or applications). Figure 23 depicts the relationship of who has what data on your personal information as well as subsequent diagrams showing current state and target states as best practices using defined standards. The Venn diagram defines:

- My Data – This data is exclusively within, and only within, the domain of an individual's data. It is a defining characteristic specifically made available to any other party under a signed, binding agreement. This space has been increasingly encroached upon by technology, governments, and businesses, especially in the health care industry in recent times (i.e. behavioral tracking tools like Phorm[16]) and this encroachment will continue. Phorm is a global personalization technology company that makes content and advertising more relevant. Phorm's platform is marketed as if it preserves user privacy and delivers a more interesting online experience, but that is in question. Indeed, a general comment can be made that 'my data' equates to privacy in the context of personal data, so the rise of the surveillance society and state is a direct assault on 'My Data'. Management of 'My Data' can be run by the individual, or outsourced to a 'fourth party service'.
- Your Data – This is the data that is within the domain of businesses; either private, public, or third sector. Proxy views of this data may exist elsewhere but are only that. This data would include, for example, the businesses or businesses own master records of their product and/or service range, their pricing, their costs, their sales outlets, and channels. Customer-facing views of much of "Your Data" are available for reproduction in the 'Our Data' intersection of the diagram.
- Our Data – This is the data that is jointly accessible to buyer and seller/service provider, and potentially to any other parties to an interaction, transaction, or relationship. It is the data generated through engaging in interactions and transactions in and around a

[16] http://www.phorm.com/about_us/index.html

customer or supplier relationship. Despite being ‘our’ data, it is probably technically owned and dispersed, or at least provided under terms of service designed by the seller/ service provider; in practical terms this also means that the seller/service provider dictates the formats in which this data exists and is made available.

- Their Data – This is the data built, owned, and sold by third party data aggregators. This includes entities such as credit agencies and marketing data providers in all their forms. Its defining characteristic is that it is only available or accessible by buying/licensing it from the owner.
- Everybody’s Data – This is public domain data, typically developed or run by large, public sector entities including local government, such as electoral rolls, Post Offices including postal address files, and others. Typically, this data is accessed under a legal contract or government access policy such as the Patriot Act in the United States, but the barriers to accessing this data are progressively more set to lower levels.
- The Basic Identifier Set/Bit in the Middle – This is the core personal identity data, which, like it or not, exists largely in the public domain. This is most typically, but not exclusively, a result of electoral rolls being made available publicly, and specifically to service providers who wish to build things from them. This characteristic is what enables the completely personal ecosystem and its impact on data privacy to exist.

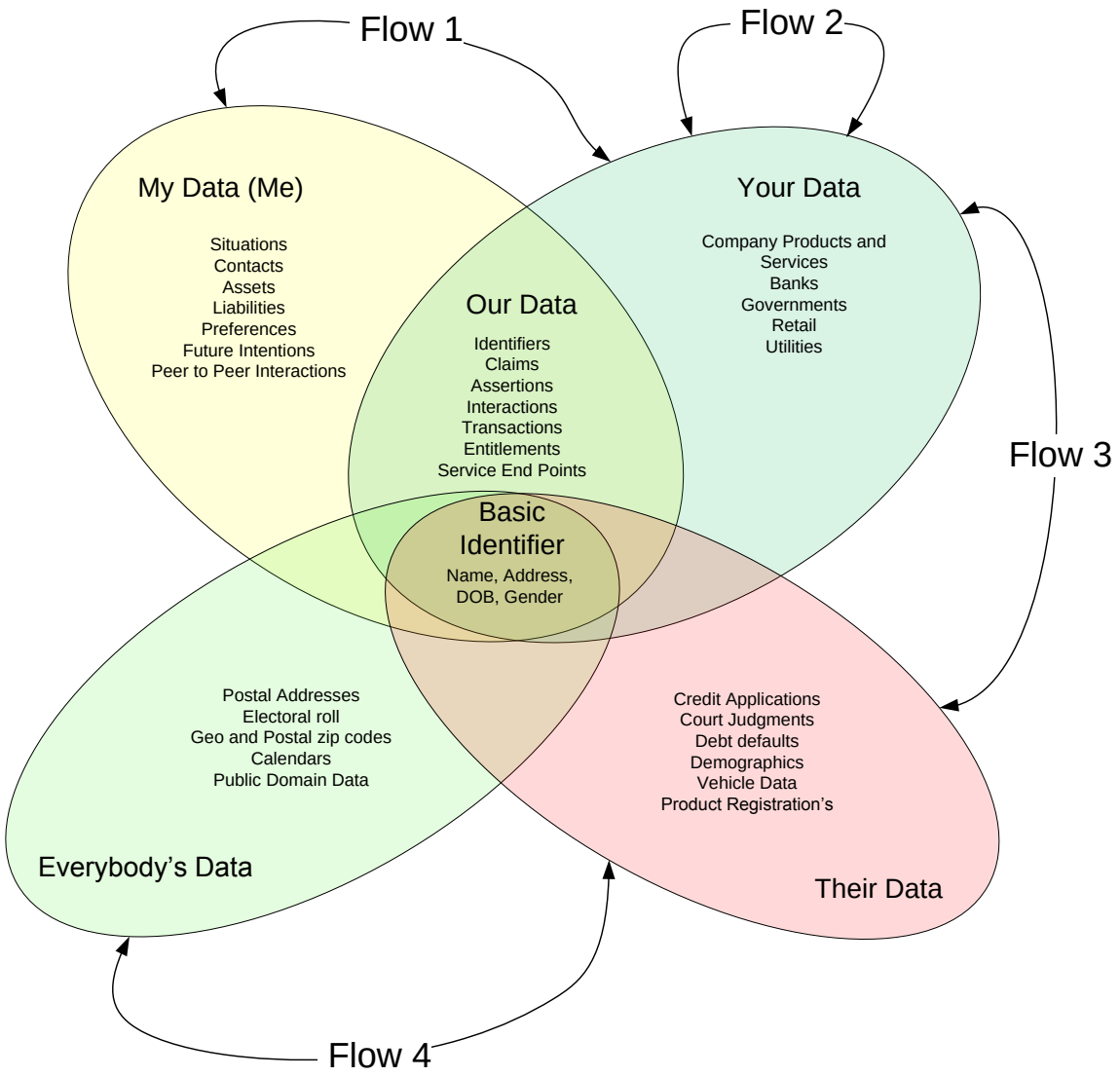


Figure 23 Data Store Taxonomy Hierarchy and Data Flow

The ovals in the Venn diagram represent the static state. This is where data lives at a point in time. The flow arrows show where data flows to and from in this personal data store ecosystem;

Flow 1 (My Data to Your Data, and My Data to Our Data) – Individuals provide data to businesses under terms and conditions set by the businesses, the individual being offered a 'take it or leave it' set of options. Some granularity is often offered around choices for onward data sharing and use.

Flow 2 (Your Data to Your Data, including Our Data) – Businesses share data with other businesses, usually through a back-channel, i.e. the details of the sharing relationship are typically not known to the data subject.

Flow 3 (Your Data, including Our Data to Their Data) – Businesses share data with a specific type of other organization, data aggregators, under terms and conditions that enable onward sale. Typically, the sharer is paid for this data or has a stake in the re-sale value.

Flow 4 (Everybody's Data to Their Data) – Data Aggregators use public domain data sources to initiate and extend their commercial data assets.

The target state shown in Figure 24 shows a different scenario altogether. Given the current path to the private and public cloud, personal data stores will evolve incrementally over the next ten years or so. This will occur data attribute by data attribute, customer and supplier management process by customer and supplier management process, industry sector by industry sector, and so on. In this scenario, the individual and the associated 'My Data' becomes the dominant source of many valuable data types (e.g. buying intentions, verified changes of circumstance), and in doing so eliminates vast amounts of guesswork and waste from existing customer/citizen management processes.

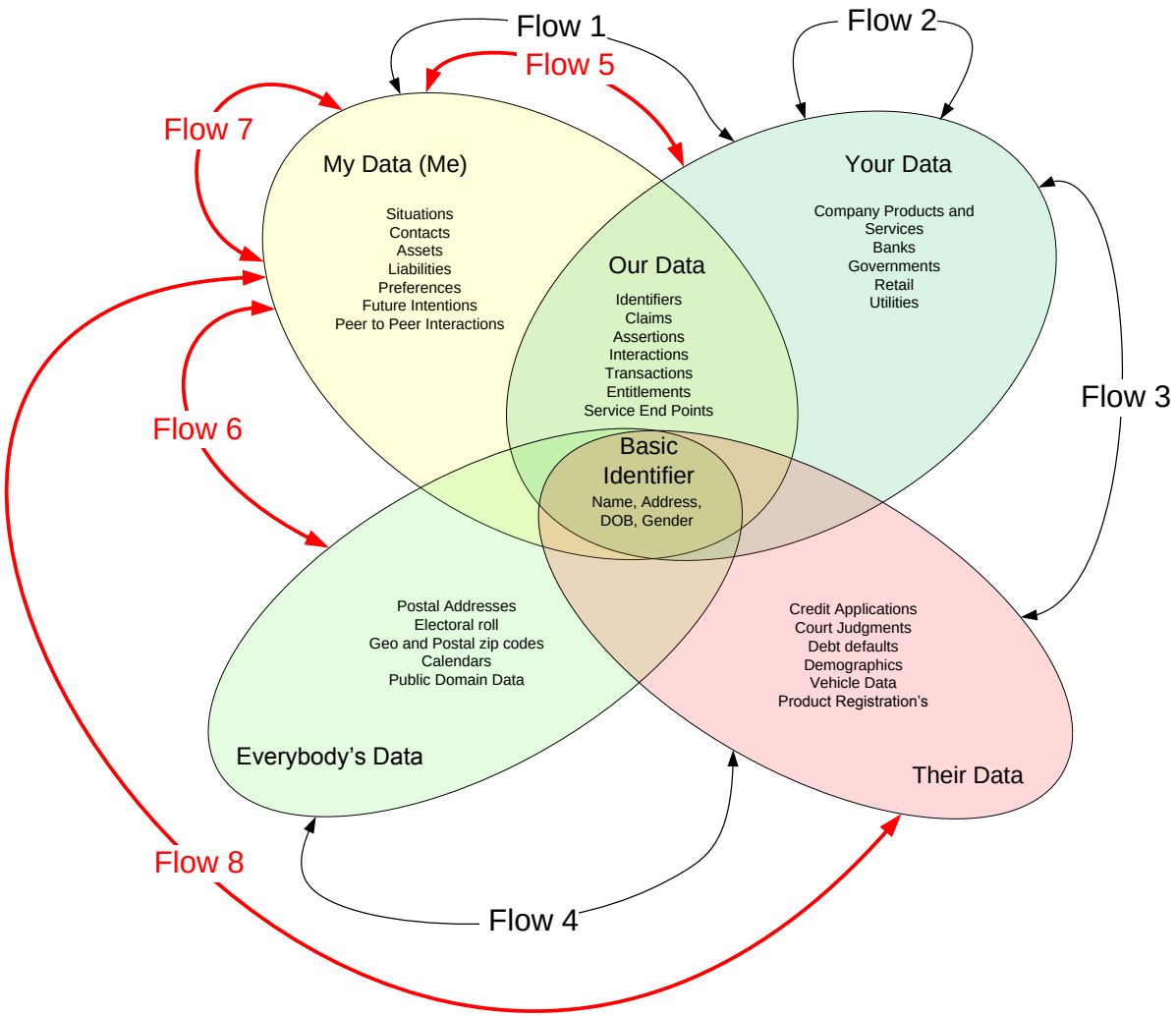


Figure 24 Best Practice Data Store Taxonomy Hierarchy and Data Flow

Best Practice – Implement additional Data Store flows to the existing lexicon

The key new capabilities required to enable this to happen are those being worked on in the User Driven and Volunteered Personal Information work groups at Kantara (one tech group, one policy/commerce), and elsewhere within and around Project VRM. The new capabilities will consist of:

- Personal data store(s), both operational and analytical
- Data and technical standards around the sharing of volunteered personal information
- Volunteered personal information sharing agreements (i.e. contracts driven by the individual perspective, creative commons-like icons for VPI sharing scenarios)
- Audit and compliance mechanics

Around those capabilities, there is a need to build a compelling and clearly articulating, in a shared lexicon the benefits of this approach—for both individuals and businesses.

Given best practices, the targeted state will emerge once these capabilities begin to impact implementations. This impact would also include the four additional individual-driven information flows over and above the current ones defined below. The defining characteristic of these new flows is that they can only be initiated by the data subjects themselves, and most will only occur when the receiving entity has 'signed' the terms and conditions asserted by the individual/data subject. The new flows are:

- Flow 5 (My Data to Your Data (including Our Data) – Individuals will share more high value, volunteered information with their existing and potential suppliers, eliminating guesswork and waste from many customer management processes. In turn, businesses will share their own expertise and data with individuals, adding value to the relationship.
- Flow 6 (Everybody's Data to My Data) – With their new, more sophisticated personal information management tools, individuals will be able to take direct feeds from public domain sources for use on their own mashups and applications (e.g. crime maps covering where one lives or travels).
- Flow 7 (My Data to (someone else's) My Data) – An enhanced version of 'peer to peer' information sharing.
- Flow 8 (My Data to Their Data) – The (currently) unlikely concept of the individual making his/her volunteered information available to/through the data aggregators. Indeed, we are already starting to see the plumbing for this new flow being put in place with the launch of the Acxiom Identity Card.

The implications in terms of the next generation data center of the above flows are substantial. It is anticipated that over time customer management processes will be driven from 'My Data'. There are two reasons for this; a) the information market is already seeing the beginning of the change in the current rush for 'user generated content' (VPI without the contract), and b) because the economics will stack up. Businesses need data to run their operations. They do not really mind where it comes from. So, if a new source emerges that is richer, deeper, more accurate, less data-toxic, and all at lower cost than existing sources, businesses will use this source.

This will not happen overnight. Obviously, as mentioned above, specific tools, processes, and commercial approaches need to emerge before this information begins to flow. Even then, the

shift will be slow but steady, probably beginning with Buying Intention data, as it is the most obvious entry point with enough impact to trigger the change.

Best Practice – Implement NGDC's to address the needs of Personal Data Stores as a service to the individual.

The first thing to remember is that Personal Data Stores (PDS) are a service to the individual. With a PDS, the data sits on the side of the individual under the individual's control; data is collected and stored in the individual's own database to be managed and controlled by that individual for the individual's purposes. This is a central, critical departure point. PDS are primarily a 'person-centric' service. They are not designed or implemented with any organization's interests or agenda in mind. They exist to serve the individual. This needs to be emphasized to offset the prevailing mindset that sees the organization as the manager of personal information and therefore assumes that anything to do with the management of personal information has to be designed to fit the organization's agenda.

Personal Data Stores are designed as services to the individual that businesses' best interests will be served by them. First, how do Personal Data Stores work for individuals?

Data storage: The first thing Personal Data Stores do is help individuals store, access, and use the information they need to manage their daily lives. Take something boring but essential, such as an insurance policy. We need to keep a record of the policy number, the certificate, the terms of the policy, how to claim, the details of correspondence relating to a claim, and so on. We need similar information relating to everyone we deal with. Most of us don't know it (because we don't stop to think about it and make a list), but at any one time we have a commercial relationship with about 200 different suppliers, all of them generating some information which needs to be stored, accessed, and used at some point in time. In real life what tends to happen is that a) this information gets scattered across many places (a filing cabinet here, an email there), and b) when we need it, we can't find it. Personal Data Stores create a single, secure, easy-to-access store for such information so that when we need the information, it is at our fingertips. Personal Data Stores will therefore be a godsend when little disasters happen, such as losing a wallet. Wallets are packed full of vital, useful, valuable information. Your wallet could include your credit cards, driving license, passport, membership cards, loyalty scheme cards, and so on.

With a Personal Data Store you simply retrieve a list of your current active cards, license and passport details, etc., along with necessary contact points. Instead of having to phone a call center for each business (waiting in frustrating queues for each one, giving the same painful details again and again), the PDS can create one single message informing them of the fact that the card has been lost. It can then be sent securely, direct to their systems bypassing traditional call centers, log-ins, and passwords and so on. It can be done in a matter of minutes—a 'one-click' hassle free process rather than a bureaucratic nightmare. In other words, Personal Data Stores are convenient. They help people do things they already have to do, only much easier and much better.

Data management: The second thing Personal Data Stores do is help us manage information better. Figure 23 lists many of the common information processing activities we undertake every day. We do them so automatically we do not realize how many different processes there are, or how sophisticated and complex they can become. Personal Data Stores help us translate these information processing activities to an increasingly online digital world, giving us tools to undertake a wide range of information management tasks including: Gather, Store, Authenticate, Verify, Share, Protect, Transfer, Dispose, Combine, Sort, Manipulate, Correlate, Personalize, Duplicate, Deduplicate (or remove duplicates), Audit, Record, Provide identification, Authorize or Give permission/s. (For more on some of these tasks, see references below).

Data sharing: The third thing Personal Data Stores do is provide individuals with better, more sophisticated tools for information sharing. This can work in many different ways, so let us look at a few examples. Currently, when conducting transactions online, the typical process is as follows. We choose what we want to buy. Then, to complete the transaction we have to tick a box declaring that we have read and agreed to the organization's terms and conditions, privacy policy, and so forth.

Personal Data Stores reverse the process. If the businesses wants to gain access to information in the individual's Personal Data Store, before it does anything else, the businesses has to tick the individual's terms and conditions. (For more detail, see the box: Information Sharing Agreements). This allows individuals to specify what information they wish to share with which businesses, for what purposes, under what terms and conditions; what we call Selective Disclosure. Selective Disclosure works in two main ways: bespoke and automatic.

Bespoke information sharing happens on a one-on-one basis. It is negotiated individually for each new situation. For example, a charitable medical research foundation might want to access an individual's health records for research purposes. The individual may say "Yes, you can have access to this information for free on the condition that a) it is not passed on to anybody else and b) it remains anonymous—so no personally identifiable details travel with it". However, if it's a pharmaceutical company doing research, the individual may set the same terms except charge a sum of money instead of handing it over free. (In practice, over time, many standard agreements will emerge allowing for many different such information sharing negotiations to take place quickly and easily.)

The second type of selective disclosure is an automatic 'subscribe to me' service. Here, businesses subscribe to updates from specific fields within the individual's Personal Data Store. To gain access, they must sign the individual's terms and conditions. The individual can choose which organization he or she wishes to accept or reject as a subscriber. Once the subscription is in place, every time the individual changes the relevant field in his data store, the subscribing organization is alerted to this fact. Take the simple example of address change. Currently, if we move to a new house, we have to remember all the businesses we have a relationship with, getting in touch with them via the processes they have dictated, wasting hours of time and hassle in call center queues, putting in passwords and usernames on web sites, and so on. With a fully populated Personal Data Store the list of current relationship is up-to-date and complete (because the store is used to manage the relationship), and the same updated information is shared with all these suppliers with just one click. The other side of the house-moving example is the businesses we forget to inform. This creates relationship management headaches for both the individual and the organization, which now has to invest significant amounts of time and money simply trying to make sure its existing database is not out-of-date. With the 'subscribe to me' service, subscribing businesses benefit from getting the right information at the right time, not six months after the event.

These three functions of 'store', 'manage' and 'share' form the heart of the Personal Data Store. However, they are just a beginning. Once they are in place, they become a foundation and springboard for a host of additional, more sophisticated services.

Data collection, data hand-backs, and personal profiles: Currently, when we buy something in a shop we are given a paper receipt, which we are told to keep as proof of purchase but

which most of us promptly throw away or lose. With a Personal Data Store the data can be 'squirted' from the retailer's system to the PDS in the form of a digital receipt. With a PDS, the individual can build a richer transaction history—a profile—of his or her purchases receipt by receipt. In addition, businesses that have collected data about the individual for their own purposes can pass this data back to the individual. Why should businesses do this? First, because individuals will ask for it (after all, transaction data is as much the individual's as the organization's). Second, because if the customer can combine data from many different suppliers, the resulting picture of the customer's behaviors and preferences is much richer and much more accurate. Take a simple example. Currently, Amazon has a detailed record of all the books I buy from Amazon. However, it doesn't know what books I buy from other booksellers. Therefore, at best, it has a partial picture of my book purchases and when it uses this data to generate 'if you bought this you might want to buy that' recommendations, it invariably gets things wrong. However, if Amazon helped me build a picture of all my purchases (by combining its transactions with transactions from other booksellers) then the picture becomes much more accurate. Of course, it's up to the individual whether or not he or she wants to share this profile with Amazon. However, if Amazon agrees to the individual's data sharing terms, there may be mutual benefits. The customer gets better recommendations, and Amazon gets more sales. In fact, Amazon might even be prepared to pay for the right to access such personally enriched data.

This gives the lie to the current organization-centric quest for a 'single view' of the customer, which, in reality, is nothing of the sort. It is just a single view of that particular organization's dealings with the customer. In fact, the only entity capable of building a genuinely comprehensive 'single' view of customer is the customer—using his Personal Data Store. In fact, as individuals build comprehensive pictures of their activities across aspects of their lives such as 'my money', 'my health', 'my home', and so on, the data held by individuals in their personal data stores will grow to be richer, more rounded and comprehensive, more accurate, and generally more valuable than any individual organization's customer data. This has two implications:

1. Looking forward, the critical 'master data' that's essential to the efficient management of customer/company relationships will shift from its current position as 'owned', controlled, and managed by the businesses to 'owned', controlled, and managed by the individual. The individual will become the primary data manager.

2. The more this information accrues, the more it will fuel new ‘added value services’ that analyze and act upon it on behalf of the individual.

One of the problems is the individual may get things wrong, or may lie. As such, verification services will be required.

Currently, if you want to make a significant purchase (such as a car), the seller conducts many credit referencing and other checks ‘behind your back’—to make sure that you are capable of paying for it, that you have a good credit history, and so on. If you are negotiating with three different sellers at the same time, they each have to organize their own checks—thereby duplicating the process three times over. With Personal Data Stores, the individual can store this data in his PDS with the verification attached to it, so that when he or she sends his details to the car seller it arrives already verified: ‘these are my credit scores as defined by Experian, Acxiom, etc.’. Any information which requires verification by a third party—driving license, endorsements, passport, educational qualifications, employment history, testimonials, insurances, transaction receipts, and so on—can be treated in the same way. In this way, it becomes cheaper and safer rather than more expensive and risky for businesses to use data from the individual’s PDS. The PDS helps streamline the process of doing business.

Archival Ecosystem

Among the many archival requirements of Next-Generation Data Centers moving into the next Millennium, is the ability to preserve and maintain access to large volumes of digital content indefinitely into the future.

Regulatory compliance and legal issues require preservation of email archives, medical records, and information about intellectual property. Web, public and private cloud services, and applications will compete for consumers, providing storage, businesses and sharing of photos, movies, and other creations. In addition, many other fixed-content repositories are charged with collecting and providing access to scientific data, intelligence, libraries, movies, and music.

Unfortunately, preserving and maintaining access to large amounts of digital information is still difficult, error-prone, and expensive. Long-term digital content suffers from many threats, including corruption of the digital content, attack, businesses changes, and obsolescence of hardware and software. For affordability and efficiency, any processing to address these threats must be performed at scale.

For the same reason, archivists and records managers of physical items avoid processing individual items (e.g. documents, objects, records). Instead, they gather a group of items that are related in some manner—by usage, by association with a specific event, by timing, etc.—and then perform all of their processing on that group as a unit. The group itself may be known as a series, a collection, or even in some cases as a record or a record group. Once assembled, an archivist will place the series in a physical container (e.g. a file folder or a filing box of standard dimensions), and that container will be marked with a name and a reference number and placed in a known location. Information about the series will be included in a "finding aid" such as an online catalog that conforms to a defined schema, which gives the name and location of the series, its size, and an overview of its contents.

The ecosystem involved in archiving is expansive, but it can be managed. The following sections will discuss this in detail.

What is an Archive?

The term ‘archive’ has come to be used to refer to a wide variety of storage and preservation functions and systems. Traditional archives are understood as physical facilities or Cloud SaaS

(Storage as a Service) or other architectures that can preserve records. The sources of this data that typically generate archive information are government agencies, public and private institutions, corporations, and individuals. The “archive” accomplishes this task by taking ownership of the records, validating that they are understandable to the accessing community, and managing them to preserve the information content and authenticity for long periods of time.

Historically, these records or data have been in many forms such as books, papers, maps, photographs, and film, which can be read directly by humans, or read with the aid of simple optical magnification and scanning aids. The major focus for preserving this information has been to make sure that the information is on media with long-term stability and access to the media is carefully controlled and monitored. The explosive growth of information in a digital format has created a major challenge, not only for traditional archives and their information providers, but also for many other data sources including businesses in the commercial and non-profit sectors. What many institutions and corporations are finding, or will find out very soon, is that one needs to take on the information preservation functions, typically associated with traditional archives, since digital information is easily lost or corrupted. Please refer to the section titled “Reliability and Resiliency”, starting on page 132 for a deep dive on this topic.

The pace of IT technology innovation and varied incremental revisions is causing many hardware and software systems to become obsolete in a matter of a few years. These changes can put severe pressure on the ability of the related data structures or formats to continue and maintain an effective representational structure and model of the information to be archived.

Long-term digital retention and preservation is the ability to sustain the understandability and usability of digital objects in the distant future regardless of changes in technologies and in the "designated communities" that use these digital objects (that is, the data consumers). Specialized preservation systems and processes are needed to enable and support long-term retention. A key component in those preservation systems is the storage subsystem where the preservation objects are located for most of their lifecycle. Storage subsystem examples include EMC Centera® and Atmos™ archiving and cloud platforms respectively. EMC’s archive ecosystem is shown in Figure 25. As shown, with over one hundred partner integrated solutions as well as utilizing standard file system interfaces such as CIFS, NFS, and HTTP, archiving,

utilizing the technologies that will be discussed in the next few sections will allow Data Center Archive Transformation for the NGDC.

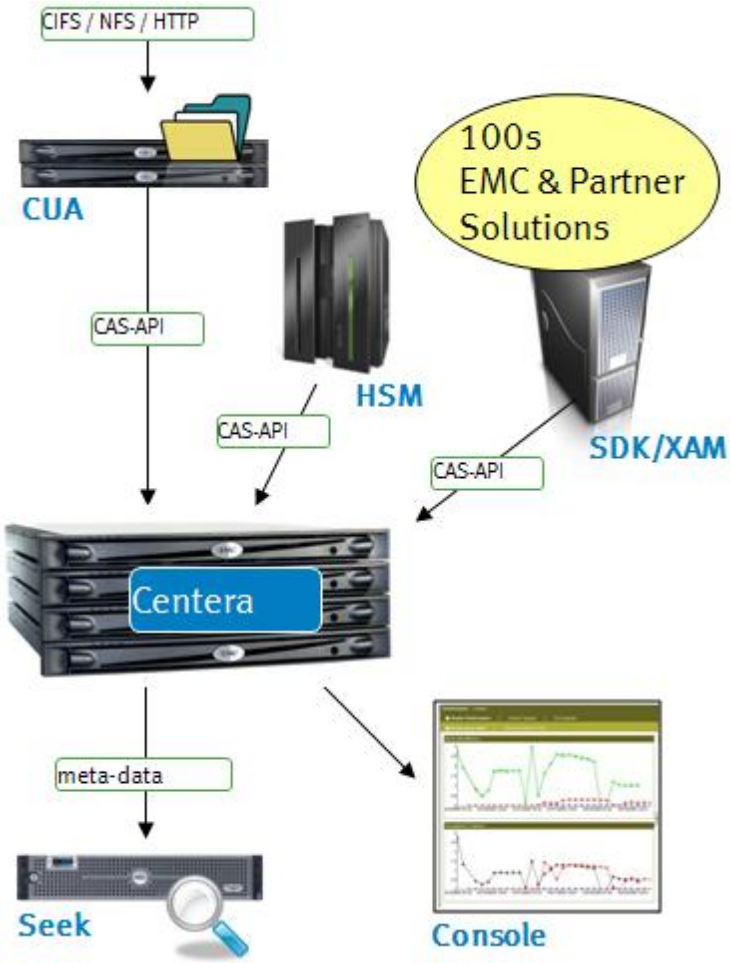


Figure 25 EMC Centera Archive Ecosystem

Because much of the supporting information necessary to preserve this information is more easily available or only available at the time when the original information is or was produced, it is important to focus on dealing with a long-term preservation solution today and not as an afterthought as it relates to Next-Generation Data Center and cloud architectures..

The explosion of computer processing power and digital media has resulted in many systems where the “Producer” role and the “Archive” role are the responsibility of the same system. A producer is the entity creating the information or data and the archive is the entity that makes

sure the data is sound for long periods of time. These systems, sometimes known as “Active Archives”, subscribe to the goals of Long Term Preservation as a best practice.

An Active Archive is a process in which the producer or administrator of the information is dynamically looking at the data to see if the data is still correct. It should be noted that the process of determining if the data is still “good” is not a trivial problem to solve and it typically comes at a cost. Refer to the section titled “Reliability and Resiliency” starting on page 132, for additional details on how to determine if data is still “good”. Given the cost attribute, challenges in efficient and optimized archiving is in direct contradiction to the consumer space in that low cost is a very important constraint in that market segment. Typically, personal data is a good example of a use case for consumer archive solutions and given the lack of an optimized low cost archive solution, individual data is at a high risk of data loss or corruption as will be discussed. One example of an active archive solution addressing many of the long term issues that will be discussed is EMC Mozy™ cloud archive solution. As a result, developing a facility, process, and standardizing to address preservation and access of information for the long term, is a necessity.

Best Practice – Implement OAIS in the Next Generation Data Center Archive Architectures

An OAIS (Open Archival Information System)[17] is a best practice that Next-Generation Data Centers and cloud providers should consider as a formal long term preservation interface or API when the IT community wants to archive and preserve information for subsequent access. The API should be robust to keep up with steady input streams of information as well as those that experience primarily non-periodic inputs. It includes archives that provide a wide variety of sophisticated access services as well as those that support only the simplest types of requests.

The OAIS API and underlying architecture recognizes the already highly distributed (locally and geographically) nature of digital information assets and the need for local implementations of effective policies and procedures supporting information preservation.

[17] http://en.wikipedia.org/wiki/Open_Archival_Information_System

An important attribute of the OAIS model includes the ability for archives to keep up with steady input streams of information. It includes archives that provide a wide variety of sophisticated access services as well as those that support only the simplest types of requests.

The OAIS model shown in Figure 26 recognizes the already highly cloud-focused distributed nature of digital information holdings and the need for the more traditional data center local implementations of effective policies and procedures supporting information preservation. This allows, in principle, a wide variety of businesses arrangements, including various roles for traditional archives, in achieving this preservation.

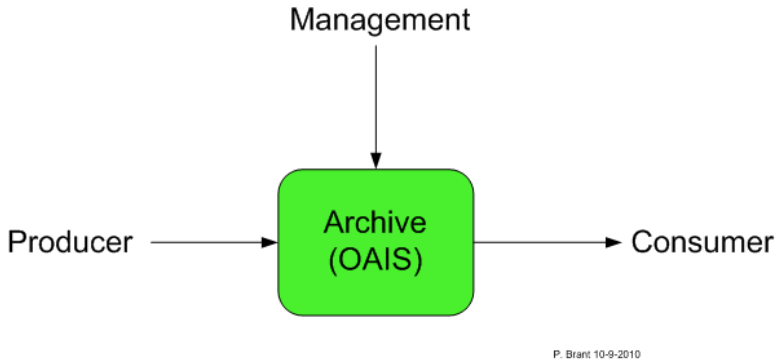


Figure 26 OAIS Consumption and Archive Model

The OAIS model, which is an open archive based on a standard API, defines Producers, Management, and Consumers. As previously discussed, the producer is the entity providing the information to be preserved. The management entity sets the overall OAIS policy as one component in a broader policy domain. Management control of the OAIS is only one of management's responsibilities. Management is not involved in day-to-day archive operations. Consumers are defined as the entity that acquires preserved information of interest and must interpret the information to acquire the content. A typical consumer is a cloud provider.

Even though the open archival concept is a sound one, there are numerous obstacles in its way. A clear definition of information is central to the ability of an OAIS to preserve it. A "system", can have a knowledge base, which allows it to understand received information. For example, a person who understands English will be able to read and understand English text. Information is defined as any type of knowledge that can be exchanged, and this information is always expressed (i.e. represented) by some type of data. For example, the information in a hardcopy book is typically expressed by the observable characters (the data) which, when they are combined with a knowledge of the language used (the knowledge base), are converted to more

meaningful information. If the recipient does not already include English in its knowledge base, then the English text (the data) needs to be accompanied by an English dictionary and grammar information (i.e. Representation Information) in a form that is understandable using the recipient's knowledge base.

Similarly, the information stored within a CD-ROM file is expressed by the bits (the data) it contains which, when combined with the Representation Information for those bits, are converted into more meaningful information as long as the Representation Information is understandable using the recipient's knowledge base.

As a result, defining an "Information Object" is an approach that OAIS takes to define the content information, which is the original target of preservation. For more information on "Objects" refer to the section titled "Object Affinity, starting on page 65, discussing object clarity as it relates to riding the cloud.

The "Information Object" consists of the Content Data Object (Physical Object or Digital Object, i.e. bits) and its associated Representation Information needed to make the Content Data Object understandable to the Designated Community as shown in Figure 27.

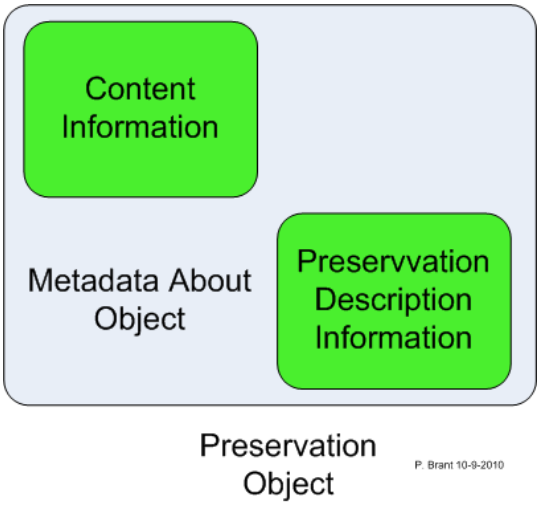


Figure 27 OAIS Object Package

Every submission of information to an OAIS by a Producer, and every distribution of information to a Consumer, occurs as one or more discrete transmissions. Therefore, it is convenient to define the concept of an "Information Package". An "Information Package" is a conceptual container of two types of information called Content Information and Preservation Description

Information (PDI). The Content Information and PDI are viewed as being encapsulated and identifiable by the Packaging Information. The resulting package is viewed as being discoverable by virtue of the Descriptive Information.

The Content Information is that information which is the original target of preservation. It consists of the Content Data Object (Physical Object or Digital Object, i.e. bits) and its associated Representation Information needed to make the Content Data Object understandable to the Designated Community. For example, the Content Data Object may be an image that is provided as the bit content of one CD-ROM file together with other files, on the same CD-ROM, that contains Representation Information.

Only after the Content Information has been clearly defined can an assessment of the Preservation Description Information be made. The Preservation Description Information applies to the Content Information and is needed to preserve the Content Information, to ensure it is clearly identified, and to understand the environment in which the Content Information was created. The PDI is divided into four types of preserving information—Provenance, Context, Reference, and Fixity. Briefly, they are defined as follows:

- Provenance describes the source of the Content Information, who has had custody of it since its origination, and its history (including processing history).
- Context describes how the Content Information relates to other information outside the Information Package. For example, it would describe why the Content Information was produced, and it may include a description of how it relates to another Content Information object that is available.
- Reference provides one or more identifiers, or systems of identifiers, by which the Content Information may be uniquely identified. Examples include an ISBN number for a book, or a set of attributes that distinguish one instance of Content Information from another.
- Fixity provides a wrapper, or protective shield, that protects the Content Information from undocumented alteration. For example, it may involve a check sum over the Content Information of a digital Information Package.

The Packaging Information is that information which, either actually or logically, binds, identifies, or relates the Content Information and PDI. For example, if the Content Information and PDI are identified as being the content of specific files on a CD-ROM, then the Packaging Information

would include the ISO 9660 volume/file structure on the CD-ROM, as well as the names and directory information of the files on CD-ROM disk. The Descriptive Information is that information which is used to discover which package has the Content Information of interest. Depending on the setting and configuration, this may be no more than a descriptive title of the Information Package that appears in some message, or it may be a full set of attributes that are searchable in a catalog service.

Best Practice – Utilize Archival strategies for digital content preservation by leveraging the knowledge of the archival profession

One of the major needs to make this preservation strategy possible is a digital equivalent to the physical container—the archival box or file folder—that defines a series, and that can be labeled with standard information in a defined format to allow retrieval when needed, utilizing the OASIS model described previously. The Self-contained Information Retention Format (SIRF)[18] is defined to be that equivalent—a logical container for a set of (digital) preservation objects that also contains catalogs and metadata related to the entire contents of the container as well as to the individual objects. This logical container makes it easier and more efficient to provide many of the processes needed to address threats to digital content.

The solution uses graphical symbols and text to specify how users or applications in specific roles use SIRF. The architecture also defines use cases allowing practitioners the tool set to create archives in an appropriate way.

A preservation object is a digital information object that includes the raw data to be preserved and additional embedded or linked metadata needed to enable the sustainability of the information encoded in the raw data for decades to come. The preservation object is the basic unit in a preservation system, and may be subject to physical and logical migrations, making it an updateable object over time. An updated preservation object is a new version of the original and its audit log records the changes that have occurred so authenticity may be verified. The OAIS Archival Information Package (AIP) standard is an example of a preservation object. OAIS provides a reference model and describes the elements that should be within an AIP, without specifying their format or how they are packaged together. Some standards are emerging for

[18]http://www.sresearch.com/Digital_Preservation_in_the_Datacenter_files/SNIA%20DMF_Digital-Preservation-Datacenter_.pdf

specific designated communities that provide specification for the actual format and packaging of a preservation object. Examples of such standards are the XML Formatted Data Unit (XFDU) for space data, the VERS Encapsulated Object (VEO) for electronic records, the Metadata Encoding and Transmission Standard (METS) for digital libraries, PREservation Metadata: Implementation Strategies (PREMIS), and Long Term Archiving and Retrieval (LOTAR) for aerospace data.

Best Practice – Implement the Self-contained Information Retention Format (SIRF) for long-term retention in the NGDC

SIRF is a logical container format for the storage subsystem appropriate for the long-term storage of digital information. It is a logical data format of a mountable unit, e.g. a file system, a block device, a stream device, an object store, a tape, etc. It assumes the mountable unit includes an object interface layer that constructs objects out of the sectors and blocks. Some advanced storage subsystems provide a built-in object interface, as in the case of Object storage, Cloud storage and XAM storage. Other, more lower level storage subsystems, have specialized media dependent standards to expose object interfaces as in the case of UDF (Universal Disk Format-ISO/IEC 13346) for DVDs, CDFS (Compact Disc File System-ISO 9660) for CDs, FAT (File Allocation Table) for HDDs, and LTFS (Long Term File System) for tapes.

It is also important to note that SIRF is unique in that the format preserves collections of objects and their relationships, and Includes generic metadata that can be extended with domain-specific information for fast access. In addition, SIRF can be mapped to and physically migrated between a wide variety of underlying storage systems. The diagram in Figure 28 schematically depicts a SIRF container that includes:

- A magic object that identifies if this is a SIRF container and its version. The magic object is independent of the media and has an agreed defined name and a fixed size. It includes means to access the SIRF catalog.
- Numerous preservation objects that are immutable. The container may include multiple versions of a preservation object and multiple copies of each version. See next section for a detailed definition and description of preservation objects.
- A catalog that is updateable and contains metadata needed to make the container and its preservation objects portable into the future, without relying on functions external to the storage subsystem.

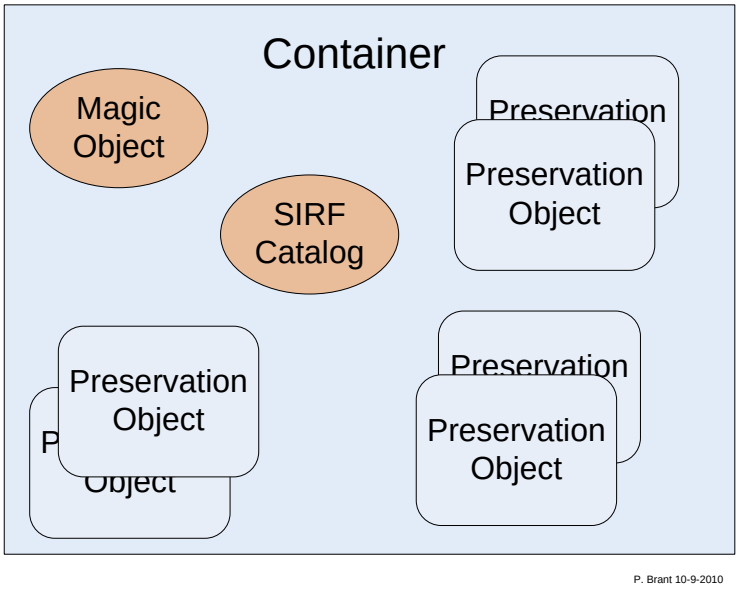


Figure 28 SIRF Container Components

SIRF is defined as using a layered approach with two levels. The SIRF level 1 catalog contains unique metadata that is not included within the preservation objects, but is mandatory to make those preservation objects portable into the future. Examples of such metadata include retention hold, reference counts, preservation object fixity algorithms, fixity values and fixity calculation dates, etc. The SIRF level 2 catalog includes information that may also be included within the preservation objects but is needed for fast access to the preservation objects. Examples of such metadata are links to representation information needed to assure referential integrity, metadata about the relationship among the preservation objects, packaging format, and so on.

SIRF does not specify the preservation object format. Preservation objects are generally created by applications and services defined outside of the storage subsystem, and their formats tend to be domain-specific. Please refer to the section titled “Storage Object Affinity”, starting on page 114, for additional information on best practices in object abstraction as it relates to cloud infrastructure, specifically relating to storage.

The storage subsystem will include multiple formats of preservation objects and this will be supported by SIRF. Specifically, SIRF is scoped to define the metadata and format in its catalog, which includes information about the preservation objects, the relationship among these objects, and information to support implementation of preservation processes.

One of the processes performed upon a preservation object is migration, which is essential for long-term digital retention and preservation. The migration process includes the act of moving data from one system to another because of a change. The nature of the change may include (but is not limited to) one of the following:

- Possible decay of storage media
- Obsolete hardware or software (encompasses obsolete file formats)
- Change in availability of software or documentation (copyright issues)
- Change in external environment, e.g., businesses, staff, moving to the private or public cloud

Migration is a major component in preservation environments. The OAIS reference model identifies four primary digital migration types:

- **Repackaging** - copying data while changing the placement of the components within the preservation object. This changes the bits of the packaging information but not the content information object itself.
- **Transformation** - copying data while performing format change on the data. This may change the bit sequence of both the packaging and content information object. Data that is transformed runs the risk of losing some of the original functionality, since newer formats may be incapable of capturing all the functionality of the original format, or the converter itself may be unable to interpret all the nuances of the original format. The latter is often a concern with proprietary data formats.
- **Refreshment** - bit-to-bit copy of the entire media's contents onto newer media of the same type, without changing the bit sequence of either the packaging or content information, or the placement of the data objects. As a result, the existing archival storage-mapping infrastructure, without alteration, is able to continue to locate and access the preservation object.
- **Replication** - copying data onto newer media that is not necessarily of the same type, but without changing the bit sequence of either the packaging or content information. Note that refreshment is also replication, but replication may require changes to the archival storage-mapping infrastructure.

Once created, the preservation objects are generally immutable[19], but new versions may be created over time. SIRF needs to support these immutable objects and migration processes. SIRF is self-describing. In other words, it can be interpreted by different systems and in different points in time. SIRF is also self-contained, meaning that all data needed for the preservation objects interpretation is contained within the container. This facilitates containment of any information losses. Any loss of a single mountable unit does not affect other mountable units.

SIRF facilitates transparent logical and physical migration and movement in order to support long-term retention and preservation where:

- Media, subsystem, or bit stream movement may include removing the mountable unit from one system and attaching it to a new system.
- Transparent migration and movement means that the original system is not involved. All the information needed for the new system to understand the mountable unit is self-described and self-contained within the mountable unit.
- Long-term may include several years and extend beyond that.
- Preservation includes sustaining the understandability and usability of the data and not just the bits.

SIRF makes it possible to reduce the cost of preservation, as the preservation processes can be done in a lower level of the system stack and can be performed close to the data in more robust, efficient, and automatic methods. Additionally, with the advent of new storage media with longer life expectancy, such as holographic versatile disk (HVD), SIRF enables reducing the number of migrations by moving the media to future preservation systems without depending on today's systems to extract, interpret, and export the preservation objects.

Table 2 summarizes the behavior and benefits when using SIRF for long-term retention and preservation:

Without SIRF	With SIRF
Sets of linked preservation objects are moved individually between systems; thus referential integrity and context may be lost	Sets of linked preservation objects are moved between systems while maintaining referential integrity and full context

[19] In this context, immutable means that the data is not subject or susceptible to change or variation in form or quality. It is static non-changing data.

Only the original application that created the preservation objects can read and interpret them	Any SIRF-compliant application can read and interpret the preservation objects
Export and import processes are needed to migrate objects	Objects can be migrated without export and import processes
Hard to sustain Preservation Objects for long-term	Preservation Objects can survive longer

Table 2 SIRF Benefits

Within the Archival Storage Ecosystem, there are several specifications that are related to SIRF, and in order to consider all of the technical interactions for the NGDC, it is important to address them and consider them as best practices for data center transformation into the foreseeable future.

Best Practice – Integrate SIRF and OAIS into the NGDC

The current reference model for long-term digital preservation is the Open Archival Information System (OAIS) ISO standard[20]. OAIS includes a functional model that describes the entities as well as their functions and process flows in a preservation system as shown in Figure 29.

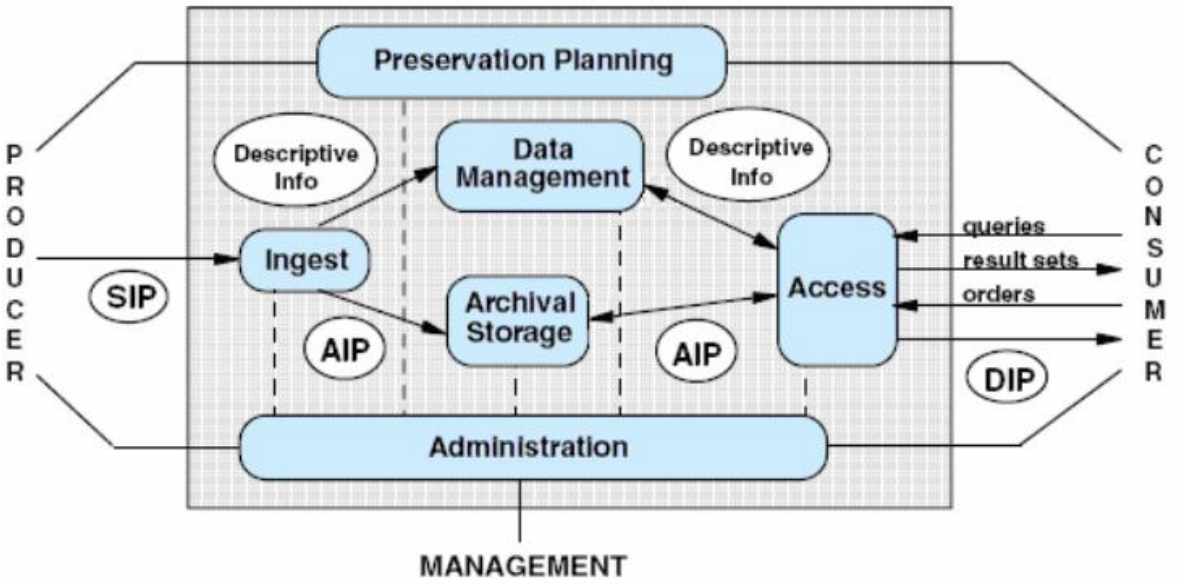


Figure 29 OAIS Functional Model

[20] nost.gsfc.nasa.gov/isoas/

The *Ingest* entity is responsible for accepting information submitted by producers and preparing it for inclusion in the archival storage, while the *Access* entity manages the services and processes by which consumers locate, request, and receive delivery of items from the archive.

The *Archival Storage* entity is responsible for verifying that archived content resides in appropriate forms of storage and remains complete and able to read and render over the long-term. This is done by periodic media refreshment or format migration, as well as implementation of safeguard mechanisms such as error-checking procedures and disaster recovery (DR) policies. The *data management* component maintains databases of descriptive metadata, identifying and describing the archived information. OAIS supports search and retrieval of the "OAIS" archived content.

The functionality described as "Preservation Planning" is responsible for monitoring the environment and developing recommendations for updating the OAIS policies and procedures to accommodate these changes.

The element described as "Data Management", provides the services and functions for populating, maintaining, and accessing both Descriptive Information, which identifies and documents archive holdings and administrative data used to manage the archive. Data Management functions include administering the archive database functions (maintaining schema and view definitions, and referential integrity), performing database updates (loading new descriptive information or archive administrative data), performing queries on the data, management data to generate result sets, and producing reports from these result sets.

The element described as "Administration", provides the services and functions for the overall operation of the archive system. Administration functions include soliciting and negotiating submission agreements with Producers, auditing submissions to ensure that they meet archive standards, and maintaining configuration management of system hardware and software. It also provides system-engineering functions to monitor and improve archive operations, and to inventory, report on, and migrate/update the contents of the archive. It is also responsible for establishing and maintaining archive standards and policies, providing customer support, and activating stored requests.

The element described as “Preservation Planning”, provides the services and functions for monitoring the environment of the OAIS and providing recommendations to ensure that the information stored in the OAIS remains accessible to the Designated User Community over the long term, even if the original computing environment becomes obsolete, which is a critical element going to the cloud. Preservation Planning functions include evaluating the contents of the archive and periodically recommending archival information updates to migrate current archive holdings, developing recommendations for archive standards and policies, and monitoring changes in the technology environment and in the Designated Community’s service requirements and Knowledge Base. Preservation Planning also designs IP templates and provides design assistance and review to specialize these templates into SIPs and AIPs for specific submissions. Preservation Planning also develops detailed Migration plans, software prototypes, and test plans to enable implementation of Administration migration goals.

The element described as “Access” in Figure 29 provides the services and functions that support Consumers in determining the existence, description, location, and availability of information stored in the OAIS, and allowing consumers to request and receive information products. Access functions include communicating with consumers to receive requests, applying controls to limit access to specially protected information, coordinating the execution of requests to successful completion, generating responses (Dissemination Information Packages, result sets, reports), and delivering the responses to Consumers.

OAIS also includes an information model. One of the main concepts in the information model is the Archival Information Package (AIP), which is the basic object stored in a preservation system. AIP serves as an example of a preservation object. As depicted in Figure 30, an AIP contains zero or one Content Information compartments and one or more Preservation Description Information (PDI) compartments. More specifically, Content Information contains the Content Data Object (raw data) that is the focus of the preservation, plus the Representation Information (RepInfo) which is needed to render the object intelligible to its designated community. This may include information regarding the hardware and software environment needed to view the content data object. The PDI compartments include additional metadata focused on describing the past and present states of the Content Information, validating it is uniquely identifiable, and determining it has not been altered in an undocumented manner. The PDI contains the following five sections:

- **Reference** – The reference portion of the AIP contains identifiers for the content information. At least one of these identifiers should be globally unique and persistent.
- **Provenance** – The provenance portion of the AIP documents the history and the origin of the content information and any changes that may have taken place since it was originated. Provenance information also documents who has had custody of the content information since it was originated.
- **Context** – The Context portion of the AIP documents the reasons for the creation of the content information and relationships to its environment.
- **Fixity** – The Fixity portion of the AIP defines an integrity check that demonstrates that the particular content information has not been altered in an undocumented manner.
- **Access Rights** - The access rights portion of the AIP defines the information that identifies the access restrictions pertaining to the content information, including the legal framework, licensing terms, and access control.

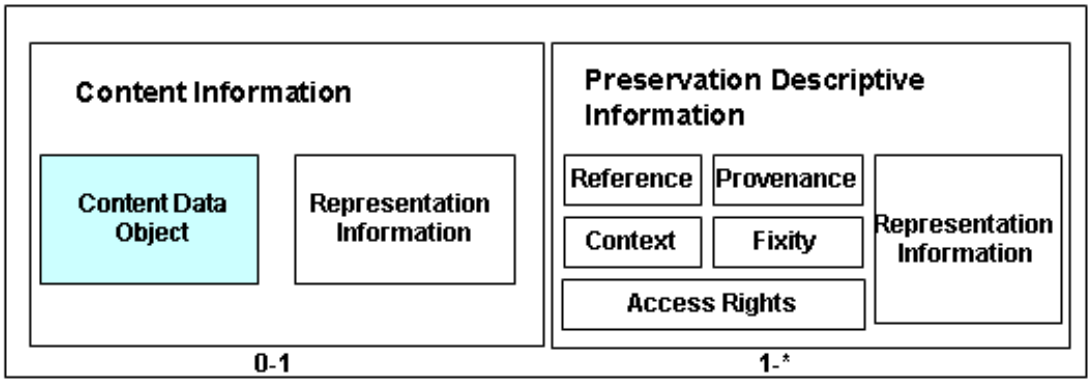


Figure 30 OAIS AIP Logical Structure

Preservation Objects within SIRF may utilize OAIS AIP. In such cases, a SIRF implementation that is OAIS-aware can enable the access to the various AIP parts including CDO, RepInfo, reference, provenance, context, fixity, and access rights.

Best Practice – Integrate SIRF and XAM into the NGDC

Some SIRF implementations may utilize Extensible Access Method (XAM)[21]. XAM is a SNIA initiative to define a standard interface between consumers (application and management software) and providers (storage systems). A XAM storage system includes one or more XSystems, with each XSystem being a logical container of XSet records. Note that EMC

[21] http://www.snia.org/forums/xam/

Centera® archive platform supports the XAM API. An XSet, which is the basic artifact in XAM, is a data structure that is a package of multiple pieces of data and metadata bundled together for access under a common globally unique external name, called an XUID. An XSet is a collection of XSet Fields. There are two types of XSet Fields: *Properties* and XStreams. A property holds contents of a simple data-type (Boolean, int64, uint64, float64, string, date time, or xuid), checked and enforced by the storage system. *XStreams*, on the other hand, includes unbounded byte streams. These can be of any valid MIME[22]-type, but the data type is not checked or enforced by the storage system.

As mentioned earlier, XAM can be used to provide an object interface for SIRF, and the XAM interface can be used to access the SIRF container and the contained preservation objects. Moreover, in some implementations of SIRF, a preservation object may be implemented as an XSet object with properties for short typed metadata and XStreams for the actual content to be preserved. Also note that the EMC Centera Object-based Archive Platform supports the XAM interface.

Best Practice – Integrate SIRF and JHOVE into the NGDC

The open source JHOVE characterization tool has proven to be an important component of many digital repositories and preservation workflows. The Library of Congress, under its National Digital Information Infrastructure Preservation Program (NDIIPP) initiative, is now funding the development of next-generation JHOVE2 architecture for format-aware characterization. JHOVE2 is based on DROID (and PRONOM), which perform automatic format identification of a file.

JHOVE is orthogonal to SIRF and the combination of the two can be very powerful. Given a SIRF-compliant storage subsystem, the application can read the preservation objects (e.g. OAIS APIs) included in that stream. Then, for each preservation object, the application can read its Content Data Object (CDO - actual data to be preserved), and its external-to-CDO metadata e.g. representation information, provenance, and fixity. However, SIRF is agnostic regarding the content inside the CDO.

---

[22] Multipurpose Internet Mail Extensions (MIME) is an Internet standard that extends the format of e-mail, to support text in character sets other than ASCII, Non-text attachments, Message bodies with multiple parts, etc.

JHOVE2 is a tool to be used upon the CDO to identify the characterization of this CDO including its format. This characterization can be used as additional representation information to enrich the preservation object of that CDO stored in a SIRF-compliant storage subsystem. Alternatively, it can be used to identify the format of the CDO after it was read from SIRF-compliant storage.

Best Practice – Integrate SIRF and Bagit into the NGDC

BagIt is a hierarchical file packaging format developed by the Library of Congress and published as an internet draft of the Internet Engineering Task Force (IETF). A bag consists of a payload that is the custodial focus of the bag and is treated as semantically opaque. The bag also includes tags that are metadata files intended to facilitate and document the storage and transfer of the bag. The tags include information such as the listing of payload files and corresponding checksums, the businesses transferring the content, and the date that the content was prepared for delivery.

SIRF is a logical container format of a mountable unit, while BagIt is more intended for a single preservation object, SIRF is focused on a container of multiple preservation objects. It includes metadata in its catalog and numerous preservation objects. The catalog metadata includes much broader information than that provided in BagIt to help interpret the preservation objects as well as the interrelationship among those preservation objects in the container. Yet, once this SIRF catalog metadata is defined, we may choose to format it in a way similar to the BagIt format.

Best Practice – Next-Generation Data Center containerization should address specific Media and Platform requirements

The containerization protocol should address the following list of the derived requirements divided into categories.

General Requirements:

- Media agnostic
  - Tape, disk, future media
  - Direct random access and serial access
  - Support mixture of storage technologies

- Vendor and Platform agnostic
- Support different standard storage technologies and interfaces, e.g. NFS, CIFS, XAM
- Extensible
    - Support additional information which may be added in the future

Format Requirements:

- Self-describing
    - The amount of "a priori" information is small and can be acquired in stages
    - Interpretable by both physical users and machines
    - Ability to do offline inspection
- Support self-contained data
    - Include means to represent internal links and cross references
- Support different SIRF formats and versions preserved in a way independent of SIRF itself, e.g. preserve the SIRF formats in an external registry
- Interoperability
    - Ability to migrate data between different systems without loss of information – data should be interpretable after migrations
    - Can be interpreted in the future
- Support methodology for verification of completeness and correctness

Preservation Object Data Model Requirements:

- Allow different data models for preservation objects
    - Allow different object data models at one time
    - Allow complex data structures such as collections of objects
    - Allow migrating objects from one data model to an alternative data model
- Can handle any proper data format for the raw data
    - No restrictions on file formats
- Enable keeping various versions of the same preservation object with their relations
    - References from new to existing preservation objects of the same version series
- Support a persistent identifier for each preservation object
    - Include additional external identifiers

- Support for retention holds
- Verification of document provenance and authenticity, regardless of migrations, whether logical or physical
- Support for storing audits. The audits can include records about modification, possibly records about access, and so forth.
- Support for “special” (secondary catalog) preservation objects
- Support for auditable time stamps that are immutable and created by known authority
- Support for managing identifiers over time
- Support secured access to the data

Performance Requirements:

- Performance
    - Need to have good performance, even for data that includes text and binaries
    - Support large objects, e.g. web archiving objects, database archiving objects, movies
    - Do not require complete scanning for access
- Enable parallel data migration
    - Enable parallel reads and writes

Storage Ecosystem

Understanding the storage ecosystem and how all of the pieces fit together is critical to helping companies achieve business agility and efficiency focusing on the Next-Generation Data Center and the alignment to the cloud. The ecosystem includes Object affinity, Archiving, Data Management, Virtualization, Long-term retention, and Data portability. All of this will be discussed in this section.

Storage Object Affinity

Cloud storage has been increasing in popularity recently due to many of the same reasons as cloud computing. Cloud storage delivers virtualized storage on demand, over a network based on a request for a given quality of service (QoS). There is no need to purchase storage or in some cases even provision it before storing data. You only pay for the amount of storage your data is actually consuming.

Some of the Use Cases

Cloud storage is used in many different ways. For example, local data (such as on a laptop) can be backed up to cloud storage; a virtual disk can be "synched" to the cloud and distributed to other computers; and the cloud can be used as an archive to retain (under policy) data for regulatory or other purposes. For applications that provide data directly to their clients via the network, cloud storage can be used to store that data and the client can be redirected to a location at the cloud storage provider for the data. Media such as audio and video files are an example of this, and the network requirements for streaming data files can be made to scale in order to meet the demand without affecting the application. The type of interface used for this is HTTP. Files can be fetched from a browser without having to do any special coding, and the correct application is invoked automatically. However, how do you get the file there in the first place and how do you make sure the storage you use is of the right type and QoS? Again, many offerings expose an interface for these operations, and it's not surprising that many of these interfaces use REST principals as well. This is typically a data object interface with operations for creating, reading, updating, and deleting the individual data objects via HTTP operations.

Storage for Cloud Computing

For cloud computing boot images, cloud storage is almost always offered via traditional block and file interfaces such as iSCSI or NFS. These are then mounted by the virtual machine and attached to a guest for use by cloud computing. Additional drives and file systems can be

similarly provisioned. Of course, cloud-computing applications can use the data object interface as well, once they are running.

What makes Cloud Storage unique is the purchase of a dedicated appliance or storage array and that of cloud storage is not the functional interface, but merely the fact that the storage is delivered on demand. The customer pays either for what they actually use or, in other cases, what they have allocated for use. In the case of block storage, a LUN or virtual volume is the granularity of allocation. For file protocols, a file system is the unit of granularity. In either case, the actual storage space can be thin provisioned and billed based on actual usage. Data services such as compression and deduplication can be used to further reduce the actual space consumed. The management of this storage is typically done out of band of these standard data storage interfaces, either through an API, or more commonly, though an administrative browser-based user interface. This interface may be used to invoke other data services as well, such as snapshot and cloning.

Best Practice – Implement an Object-based Data Management Infrastructure to Address Interoperability and Transparency Requirements

A best practice is to implement a broad object-based transparent interface to address storage operations within the Next-Generation Data Center. A Cloud Data Management Interface (CDMI)[23] has been defined and is meant to enable interoperable cloud storage and data management. In CDMI, the underlying storage space exposed by the above interfaces is abstracted using the notion of a container. A container is not only a useful abstraction for storage space, but also serves as a grouping of the data stored in it, and a point of control for applying data services in the aggregate. Please refer to the section titled “Best Practice – Implement containerization into the Next-Generation Data Center’s information management processes”, starting on page 131 for additional information on Containerization.

CDMI provides not only a data object interface with CRUD semantics[24]; it also can be used to manage containers exported for use by cloud computing infrastructures as shown in Figure 31.

---

[23] http://cloud-standards.org/wiki/index.php?title=SNIA_Cloud_Data_Management_Interface_(CDMI)

[24] http://msdn.microsoft.com/en-us/library/ms978509.aspx

With a common cloud computing management infrastructure using CDMI and OCCI for a Cloud Computing Infrastructure, CDMI Containers are accessible not only via CDMI as a data path, but other protocols as well. This is especially useful for using CDMI as the storage interface for a cloud computing environment as shown:

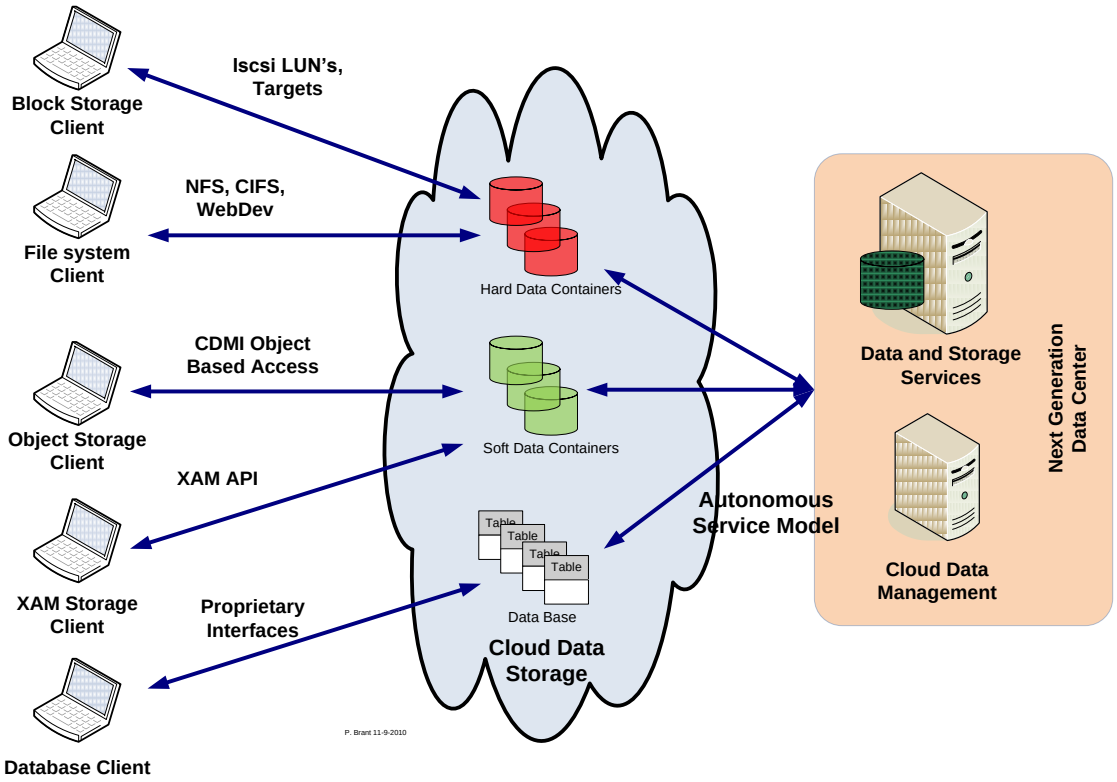


Figure 31 CDMI Data Storage Interface

The exported CDMI containers can be used by the virtual machines in the cloud computing environment as virtual disks on each guest as shown. With the internal knowledge of the network and the virtual machine, the cloud infrastructure management application can attach exported CDMI containers to the virtual machines.

The cloud computing infrastructure management shown in Figure 31 supports both OCCI and CDMI interfaces. To achieve interoperability, CDMI provides a type of export that contains information obtained via the OCCI interface. In addition, OCCI provides a type of storage that corresponds to exported CDMI containers. OCCI and CDMI can achieve interoperability initiating storage export configurations from either OCCI or CDMI interfaces as starting points. Although the outcome is the same, there are differences between the procedures using CDMI's interface over the OCCI's as a starting point. Below, we present examples of interoperability

initiating storage export from both CDMI and OCCI approaches. A client of both interfaces would perform the following operations as an example:

- The Client creates a CDMI Container through the CDMI interface and exports it as an OCCI export type. The CDMI Container ObjectID is returned as a result.
- The Client then creates a Virtual Machine through the OCCI interface and attaches a storage volume of type CDMI using the ObjectID. The OCCI Virtual Machine ID is returned as a result.
- The Client then updates the CDMI Container object export information with the OCCI Virtual Machine ID to allow the Virtual Machine access to the container.
- The Client then starts the Virtual Machine through the OCCI interface.

Best Practice – Implement CDMI Metadata Properties Consistent with the CDMI Specification

CDMI uses many different types of metadata, including HTTP metadata, data system metadata, user metadata, and storage system metadata. HTTP metadata is metadata that is related to the use of the HTTP protocol, such as content-size, content type, and so on. This type of metadata is not specifically related to the CDMI standard, but needs to be discussed to explain how CDMI uses the HTTP standard. Data system metadata is metadata that is specified by a CDMI client and attached to a container or data object, abstractly specifying the data requirements that are then supplied by data services that are deployed in the cloud storage system. The data system metadata settings are treated as goals. In some cases, actual measurements toward these goals are specified.

User metadata is arbitrarily defined metadata that is specified by the CDMI client and attached to objects. The namespace used for user metadata is self-administered (such as using the reverse domain name) and restricted to not beginning with the prefix "cdmi_". Storage system metadata is read-only metadata that is generated by the storage services in the system to provide useful information to a CDMI client.

Data Portability

The Next-Generation Data Center is changing in many ways. One way is the concept of data or information being moved seamlessly, without disruption (i.e. non-disreputably), with no performance or reliability degradation.

Best Practice – Implement Data Portability techniques into the NGDC architecture

Data portability enables a borderless environment, where one can move information easily between network services, reusing data they provide, while controlling their privacy and respecting the privacy of others.

With data portability, for the end user or consumer, one can bring your identity, friends, conversations, files, and histories with you, without having to manually add them to each new service; a major plus. Each of the services one can use can draw on this information relevant to the context. As one's or a process experiences accumulate, and you add or change data, this information will update on other sites and services if you permit it, without having to revisit others to re-enter it.

With cross-system data access, interoperability, and portability, people can bring their identities, friends, conversations, files, and histories with them to your service, cutting down on the need for form filling, which can drive people away. With minimal effort on the part of new customers, you can tailor services to suit them. When your customers browse networked services and accumulate experiences, this information can update on your service, if people permit it. Your relationship remains up-to-date and you can adapt your services in response, even when they do not visit. With mutual control and mutual benefit, your relationships remain relevant, encouraging continued usage.

Data portability is a new approach, where it is easier to use and deliver services. This frictionless movement through the network of services fosters stronger relationships between people and services providers, and helps build a healthy networked ecosystem.

The goal of Data Portability is to assist producers of data, be it cooperative or personal data, to use and protect the data one creates on networked services, and to advocate for compliance with the values of Data Portability. This is an optional extra function, and is complementary to the goal of the personal data store architecture discussed in the section titled "Personal Data Store Attributes", starting on page 82.

Storage Tiering

The concept of Storage Tiering is the use of virtual or physical storage devices with different I/O performance, data availability, and relative cost characteristics to provide differentiated online storage for computer systems.

The advent of Serial ATA (SATA), as well as EFD (Enterprise Flash Drive) based disk arrays has accelerated interest in multi-tier storage. However, multiple tiers of storage have been available to users since there have been disk arrays and volume managers. Software-based virtualization technology in the form of volume managers implements multi-tier storage in the form of virtual volumes configured from the hardware devices at hand. Multiple tiers of storage hardware simply add to the number of options available to users. Any type of software-based virtual volume can be created from any type of storage device or LUN presented by disk arrays. One can argue, however, that storage hardware and software can implement a more efficient and dynamic methodology as it relates to tiering using autonomic self-healing system theory[25].

Best Practice – Consider Application requirements for Auto Tiering

One attribute of the storage ecosystem for the Next-Generation Data Center is the ability to place application data on the most efficient level of physical hardware consistent with the application's performance and availability requirements. IT organizations, tasked with deploying a tiered storage strategy, for example, databases that can reside on EMC Symmetrix or CLARiiON storage arrays have the capability of leveraging EMC Fully Automated Storage Tiering (FAST)[26]. EMC FAST supports the ability to dynamically move data across tiers of storage without interruption to applications. Access to data and corresponding I/O thresholds are typically used to classify data with the appropriate tier of storage. The challenge with defining the tiers of storage for data sets within database objects, based on I/O alone, is the lack of business process and user context. Without this dimension of input, there is a potential for misaligning information and inefficiently deploying storage tiers.

---

[25] See Section on Autonomic Systems in the 2010 Proven Professional Paper titled "Above The Clouds - Best practices to Create a Sustainable Computing Infrastructure to Achieve Business Value and Growth".

[26] See Section on FAST in the 2010 Proven Professional Paper titled "Above The Clouds - Best practices to Create a Sustainable Computing Infrastructure to Achieve Business Value and Growth".

Typically, there has not been an efficient way for database administrators to coordinate with storage administrators on how to appropriately classify and tier data. How does the storage subsystem identify the highly active database objects that should be moved to higher performing storage if the storage does not have any inkling of what the application is doing? A best practice and the most optimum solution is understanding what the application is doing via some form of feedback control that would have the most impact on efficiency and performance for the business.

A solution such as Zettapoint, using the product “DBClassify”, is one solution, “DBClassify” is a Policy Engine for EMC FAST that addresses this challenge. This is done by considering data I/O in context with business users and processes that access that data. Additionally, because DBClassify can track time-based usage patterns, the right information can be placed proactively on the right tier at the right time. Placement of data on the appropriate class of storage is optimized, maximizing the efficient use of higher performing storage. DBAs are now able to work more efficiently with their storage team counterparts and control proper data placement. Because IT can work more efficiently, DBClassify quickly enables effective storage tiering with EMC FAST, accelerating time to an efficient I/O workload.

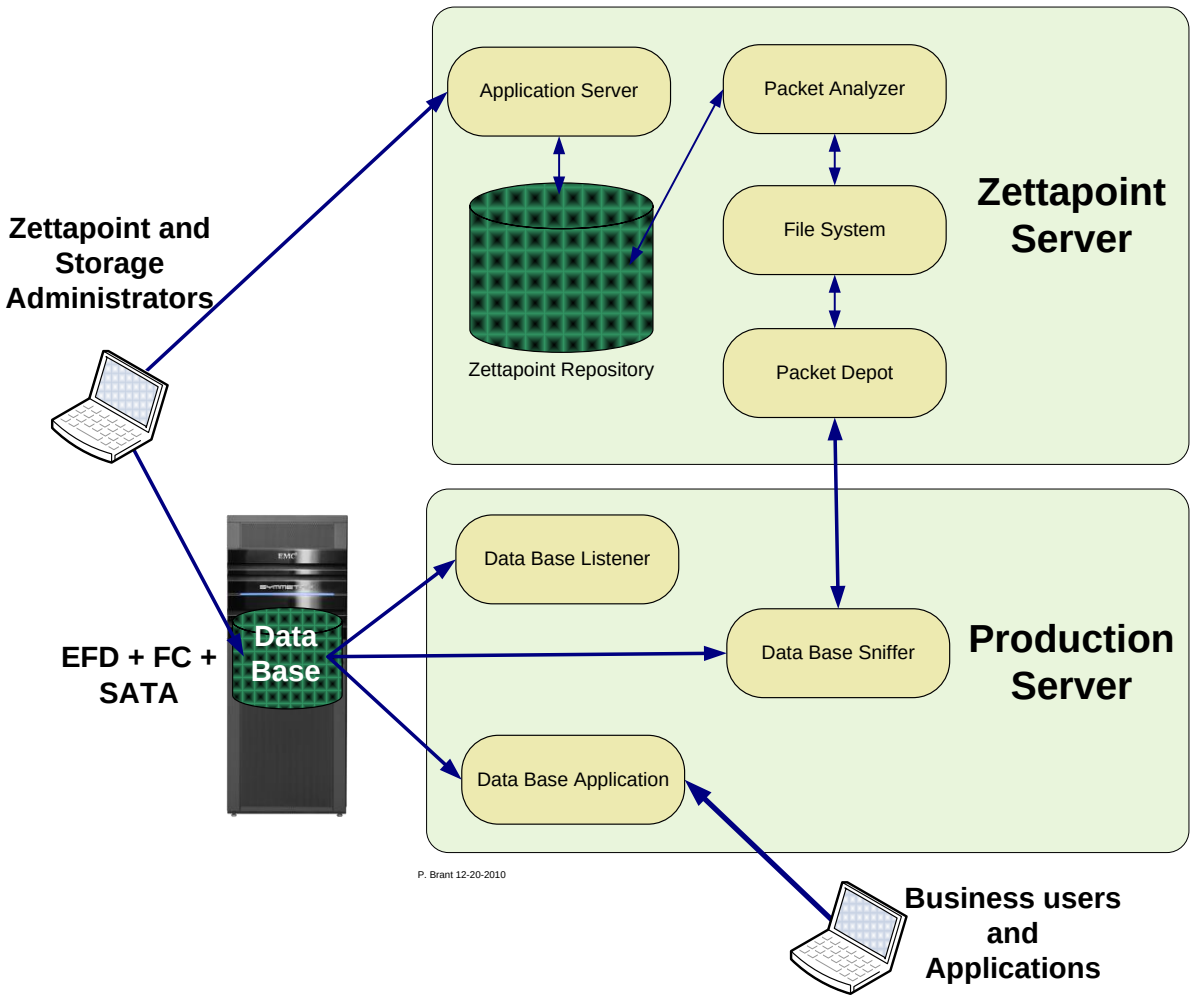


Figure 32 Database Tiering Data Flow

The process of precisely identifying what application object or portion of the database can be tiered is done by using FAST Technology and DBClassify. The architecture is shown in Figure 32.

The packet analyzer extracts the captured raw database and processes the logical session structure. The packet analyzer then uses database-specific parsing to process the raw messages. The packet analyzer then assembles the model of the database from the parsed data base objects, which then creates a data dictionary formed by the database schema.

The database sniffer listens to database transactions running on the database server, and then captures the packets and places them into the packet depot. The advantage of this architecture

is the sniffer has to only monitor the database statements and I/O wait data and does not have to do full scans of the shared global data.

With this type of architecture, one can easily scale to larger databases, allowing growth and expansion for the NGDC. It is also important to note that there are other tools available to create an efficient tiered solution approach. DBClassify, in conjunction with EMC's FAST, is a more dynamic approach to tiering. To create a more baseline tiering model based on generic I/O modeling methods, taking cost and efficiency into place and not focusing as much on specific applications, please refer to the section titled "Green Stability", starting on page 151, for a more sustainable "Green" approach to tiering.

Sustainable Reference Models

The second triad as it relates to riding the cloud and focusing on the transformation to the Next-Generation Data Center, is to understand what reference models are available in order to achieve sustainability. Technologies are available to the practitioner. The questions are; what are the attributes that are important moving in this direction? Let us enumerate the enablers and go over the details.

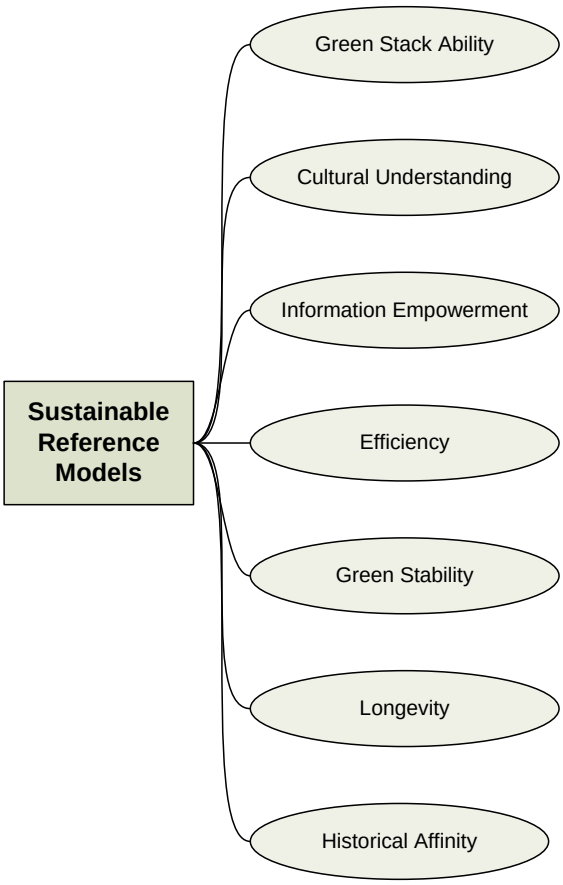


Figure 33 Sustainable Reference Models

Let us first define what is meant by "Sustainability". Sustainability is a pro-active approach to ensure the long-term viability and integrity of the business by optimizing IT resource needs, reducing environmental, energy, and/or social impacts, and managing resources while not compromising profitability to the business[27]. One corollary to this definition would be that not

[27] 2010 Proven Professional Paper titled "Above The Clouds - Best practices to Create a Sustainable Computing Infrastructure to Achieve Business Value and Growth", by Paul Brant.

only would developing a sustainable IT model not compromise profitability, but also, by conforming to best practices, would actually increase it. To achieve a sustainable model, the factors that relate to this goal are outlined in Figure 33 Sustainable Reference Models, on page 123, above.

In order to ride the cloud effectively and address the NGDC requirements, historical issues of whether the data will be there 100 years from now, or empowering business and individuals alike in controlling the destiny of their critical information need to be addressed. This and more will be discussed in the following sections.

Information Empowerment

Our current approach to collecting and using personal and corporate data is dysfunctional. As previously mentioned, the social networking juggernaut, “Facebook”, is the perfect example of personal data being manipulated and used with the average user not knowing what the long-term negative ramifications are, in terms of the disclosing personal information. As a result, as individuals, we have lost control over our personal data!

So much so that most of us see personal data management as a threat and a risk (identity theft), a hassle and a chore (red tape), and a source of frustration and irritation (businesses taking our data and losing it.

To add to the problem, abusing personal data by using it to spam us with junk messages, or handing it over to third parties, who we do not know and have no control over). Meanwhile, businesses are discovering that the collection and use of personal data is:

- Increasingly expensive, with high levels of waste from excessive duplication, error and inaccuracy, and guesswork
- Corrosive to trust, turning off customers, which in turn undermines rather than strengthens relationships
- A brake on innovation and growth

A new personal information ecosystem is emerging. It is organized around individuals collecting, storing, managing, using, and sharing their own personal data for their own purposes. Building on the technological revolutions of the last few decades (data warehousing, ubiquitous personal and mobile computing and internet connectivity, social networking), its core purpose is to help

individuals manage personal data as a personal asset and resource, which they can use to organize and manage their lives better.

The new personal information ecosystem will trigger and support a new wave of innovation and economic growth from a wide range of new services. These services will help individuals use their own personal information to research, make, and implement better decisions in countless ways. They will help people when they lose their wallet, move their home or buy a car, and help them manage their health or finances over long periods.

Many of the components of this new information ecosystem have emerged over the last ten years (e.g. online search, comparison sites, P2P product reviews, social networking, identity management, privacy enhancing technologies). However, the catalyst around which the new ecosystem will crystallize is only beginning to emerge. The Personal Data Store is a new type of personal information management service that helps individuals gather, store, update, correct, analyze, and share their own data in ways that they can control. Personal Data Stores represent a fundamental breakthrough in personal information management. They will:

- Give individuals the tools they need to realize the rapidly growing value of their own data in an increasingly online world.
- Unleash a new wave of economic growth around new types of information-driven personal services.
- Help businesses reduce the high levels of cost and waste inherent in their current businesses-centric approaches to the management of personal data, while helping them gain richer, more timely insight into their customers' needs and preferences.
- Help to create a climate of trust, leading to more information being used more responsibly, to add greater value for both businesses and their customers.
- Benefit society: the more confident and empowered individuals become in the management and use of their own personal data, the more they will contribute positively to social, civic, and economic affairs.

Businesses need to work with Personal Data Store providers such as Mydex to test, develop, and prove Personal Data Store technologies, infrastructure, information sharing processes and mechanisms, legal, and business compliance models.

Governments and regulators need to pave the way for the new ecosystem by addressing online identity policy and being ready to work with structured authenticated data from verified

individuals. This needs standards. It may need minor policy revisions. It does not need significant new legislation or major infrastructural investment.

For a detailed discussion on what technologies and processes are available or will be available allowing for the transformation of information empowerment back to the users riding the clouds, please refer to the previous sections titled “Personal Data Store Attributes”, starting on page 82.

Historical Affinity

A 350-year-old copy of Shakespeare is about as readable as a new one. However, a 35-year-old floppy is about as readable as my 8-track tape. I guess the author is showing his age ☺. Preserving data is essential to the digital civilization of which we are all part. However, the question is how? Here is a new, proposed approach.

Part of the overall strategy of the Next-Generation Data Center is the concept of historical retention of information. Long-term digital retention and preservation is the ability to sustain the understandability and usability of digital objects in the distant future, regardless of changes in technologies, and in the "designated communities" that use these digital objects (that is, the data consumers). Specialized preservation systems and processes are needed to enable and support long-term retention and a requirement in the NextGeneration Data Center and any cloud implementation. A key component in those preservation systems is the storage subsystem, where the preservation objects are located for most of their lifecycle.

One cannot predict what features future storage subsystems will provide, so the most practical way to solve this problem is to make sure the content itself provides the means to be migrated, without losing either its metadata or our ability to identify its format. To make it easier to move content between systems and technologies, while ensuring it remains complete and interpretable, we need a standard way to store that information that is self-contained, self-described, and extensible. The key properties of a long-term storage container format are:

- **Self-contained:** Long-term retention requires the preservation of both data and its surrounding metadata, which can become disaggregated. To prevent this from happening, the unit of storage for an object should include, to the extent possible both the data and its metadata, so that they are treated and moved together as a single storable unit that will be kept intact for the life of the object. Similarly, the unit of storage for the objects' container should include to the extent possible both the objects and the metadata about the objects and their interrelationship. The metaphor we use here is a closed bottle that includes all the information needed to understand the bottle's content in another point in time (see SIRF visual identifier).
- **Self-described:** It should be possible to look at a data package and determine what it is, so that we can interpret it correctly. For example, it should be possible to determine the objects within the container and their associated metadata. One problem is that the self-

description of the container must also be interpretable. If it is complex, then it too must be self-describing. Because of this recurring problem, a completely self-describing format is impossible to achieve. However, self-describing formats remain useful if at the root of the recurrence, they use only very widely used formats, such as ASCII, and the self-description itself can be updated over time. While it is possible to create self-describing proprietary formats, widely used industry standard formats are more likely to have a long life.

- **Extensible:** It is impossible to predict all the changes likely to be needed for information retained for decades. As these changes occur, we want to preserve information about what changes we made and when. For example, we need to record information about format migrations, and may want to keep the original container tied to its rendition in a new format. As another example, we may want to add information about changes in custody of the container, or be able to add new types of contact information to existing information about customers. A good long-term storage container format must allow for additions and extensions, while preserving the integrity of the original data.

The 'two grand technical challenges' of long-term digital information retention are logical and physical migration. Logical migration is the practice of updating the format of the information into a newer format that can be read and properly interpreted by future applications or readers without losing the authenticity of the original. Physical migration means to copy the information from newer storage media to preserve the ability to access it, and to protect it from media corruption. Best practices today require logical and physical migration every 3-5 years. Based on these practice standards, the real underlying challenge is how to scale migration capabilities while controlling cost. A business that has 1,000 TB (a petabyte, PB) in its digital archive repository will have 50 percent more next year. In three years, they will need to migrate that first petabyte. In five years, one will need to migrate 2.25 PB. How do businesses expect to do that and keep up with the growth, the cost, and the complexity? The answer is businesses cannot and it is not sustainable!

It can be argued that today's migration practices do not scale cost-effectively and changes will not happen until the long term archive issue is addressed. This means that today's reliance on migration is not sustainable. The world's digital information is at great risk! New technological approaches are required that meet the legal, business, cost, and scalability requirements of the 'digital age' for long-term digital information retention.

Best Practice – Define Long-Term Retention Solutions

There are four categories corresponding to the classes of needs that solutions or best practices must address in the Next-Generation Data Center (NGDC).They include;

Accommodate the requirements of the critical business drivers behind long-term retention by mitigating legal, compliance, business, and security risk, as well as preserving the history of the businesses forever.

- Overcome the barriers inhibiting adoption of best practices that range from the cost-effectiveness of solutions to stimulating collaborative efforts within the businesses. Many of these requirements are business issues and fit the profile of best practices. The most alarming barriers are indications and warnings that executive management does not really care and that there is no prestige in archive practices within the IT businesses.
- Improve operating practices by providing better management tools, best practices, job visibility, and education.
- Solve the technology challenges by:
  - Solving logical and physical migration
  - Solving the ability to scale the volume of information
  - Incorporating metadata into the archival repository
  - Including databases, email, and legacy information
  - Providing a full spectrum of information and data services core to the digital information repository that provide for classification, control, discovery, availability, protection, security, integrity, audit, forensics, non-repudiation, preservation, and permanent deletion

Longevity

The Next-Generation Data Center needs to address a very important aspect of information management. The issue is long-term retention of information.

Best Practice – Data center retention processes need to be in place to address long-term retention of information.

A SNIA survey relating to how businesses feel about their data retention requirements shows that 68 percent of businesses needed to preserve data for 100 years or longer[28]. Data is fragile. Threats include:

- Media/hardware obsolescence. Even if you have an 8-inch floppy drive, there may not be hardware capable of running the software required to read it, let alone the application to open the files on the floppy.
- Software/format obsolescence. Remember WordStar?
- Lost context/metadata. A document's contents may appear mundane, but if it is from the President to the Secretary of State, its context makes it important.
- Disaster
- Physical user error
- Media fault
- Attack

Preserving bits is difficult. Saving one PB for 50 years, with a 50 percent chance of damage gives a bit a half-life of 1017 years. That is not achievable for large data sets. There is no simple technical fix: we cannot predict change but know it will occur. Processes are key! Processes for data preservation must evolve to get us to the next step. Standards make it easier, but are not the whole answer. The other questions are what to preserve? Bits? Applications? Context? Is it even possible to preserve everything? For example, with an old book: the content? Paper wear? Political context? Bookplate? Where it falls open?

We will lose information moving from physical to digital. We cannot know what future generations will consider valuable. For example, scientists collect old hollow metal buttons because they contain air samples from when the buttons were made. Who dreamed 150 years

---

28

http://www.snia.org/about/news/newsroom/pr/view?item_key=b5122ec64ecf57fdc6b95d2e80f27b1540ee08f1

ago, that would be valuable? Preservation must facilitate the storage of objects and map to a wide variety of devices and technologies.

Best Practice – Implement containerization into the Next-Generation Data Center's information management processes

A best practice is to implement a "Self-contained Information Retention Format" also known as "SIRF". This is the digital equivalent of a physical container that archivists already know how to manage. SIRF containers hold preservation objects, a catalog, and an object that labels the SIRF container. SIRF maintains referential integrity, links between objects, and context. Any SIRF-compliant app can read and interpret the objects. Objects are migrated easily. A couple of use cases show some of the problems:

- **Legal holds and e-discovery.** In civil suits, the parties are required to preserve all requested documents—legal hold—under threat of severe penalties. However, not all documents are included, such as client-attorney emails. How can all documents be preserved and the right ones selected for disclosure?
- **Biomedical information.** Medical images are needed for patient history. However, what if the patient was 12 years old and now is an adult? How do we protect their privacy and ensure that only the "right" adults now get access to it?

Massive data loss can threaten civilization. The burning of the ancient Library of Alexandria, destroying hundreds of thousands of handwritten books, contributed to Europe's Dark Ages, as knowledge of ancient art, science, and math were lost. The little recovered through Muslim scholars helped create the Enlightenment, but how much more was lost?

However, the threat of digital data loss is far larger. Cheap storage and sophisticated data mining allow us to derive value from datasets that once we could not even afford to collect, let alone analyze. This is critical.

Reliability and Resiliency

With the idea of riding the clouds, emerging cloud web services, such as email, photo sharing, and web site archives must preserve large volumes of quickly accessible data indefinitely into the future. The costs of doing so often determine whether the service is economically viable. Please refer to the section titled “Vertical Markets and Use Cases”, starting on page 201 for additional information on riding the clouds on emerging cloud services.

One can make the case that these applications' demands on large-scale storage systems over long time horizons require us to reevaluate traditional system designs. One can examine threats to long-lived data from an end-to-end perspective, taking into account not just hardware and software faults, but also faults due to humans and businesses. This paper presents a simple model of long-term storage failures that helps us reason about various strategies for addressing some of these threats. Using this model, the anticipation is that the result is that the most important strategies for increasing the reliability of long-term storage are detecting latent faults quickly, automating fault repair to make it cheaper and faster, and increasing the independence of data replicas.

Long-term reliable storage presents challenges that differ from traditional problems in the storage literature, warranting increased interest from systems and storage researchers. Frequent headlines remind us that bits, even bits stored in expensive, professionally administered data centers, are vulnerable to loss and damage. The vulnerabilities grow when large volumes of data must be stored indefinitely into the future, as required by emerging cloud enabled web services, such as e-mail (e.g. Gmail), photo sharing (e.g. Ofoto), and archives (e.g. The Internet Archive). The economic viability of these services depends on storing data at low cost. This is also predicated by the acceptance by customers, depending on their keeping data unaltered and accessible with low latency, which are very big assumptions. Doing so over long periods would be easy if fast, cheap, reliable disks were available, and if threats to the data were contained to the storage subsystem. Unfortunately, one can argue that neither is true. The economics of high-volume manufacturing provide a choice between consumer-grade drives, which are low cost, fast, and reliable, and enterprise-grade drives, which are vastly more expensive, much faster, but only a little more reliable. For short-lived data, current levels of drive reliability might not pose a problem, but for long-lived data, faults are unpreventable. In addition, long-term storage faces many threats beyond the storage system. The issues include obsolescence of data formats, malicious attacks, and economic and structural volatility of the

host businesses. One use case is the need for digital preservation, storing static data over long periods.

One can list the threats to data survival, using examples from real systems, and examine mismatches between the design philosophy of many current storage systems and these threats. One can introduce a simple reliability model of long-term replicated storage systems.

Many analysis models base reliability and protection models based on the RAID protection architecture. A best practice is to take a more end-to-end, rather than device-oriented approach and address a wider range of faults. The best practice model proposed in this section explicitly incorporates both latent faults, which occur long before they are detected, and correlated faults, when one fault causes others or when one error causes multiple faults. For example, one would like to be able to answer questions such as, do latent faults occur frequently enough that we need to worry about them? How often should we search for latent faults, if doing so can itself cause damage? Note that the increase in fault rate must be balanced by a quadratic reduction in the time to detect and repair the faults. Is it better to increase the mean time between visible faults or between latent faults? Perhaps neither if it significantly decreases the other. Is it better to increase replication in the system or increase the independence of existing replicas? One can argue that both are applicable, but replication without increasing independence does not help much. Some of these strategies have been proposed before, but one can argue that both are worth revisiting in the context of long-term storage.

One can conclude that the most important strategies for increasing the reliability of long-term storage are detecting latent faults quickly, automating repair to be faster and cheaper, and increasing the independence of replicas at all levels.

Best Practice – Consider long term preservation and the threats in the Next Generation Data Center

Preserving information for decades, or even centuries has proved important. One example is the Shang dynasty in the 12th century B.C. Chinese astronomers inscribed eclipse observations on animal bones and tortoise shells. About 3200 years later, researchers used these records together with one from 1302 B.C. to estimate that the accumulated clock error was just over 7

hours, and from this derived a value for the viscosity of the Earth's mantle, as it rebounds from the weight of the glaciers[29].

Longitudinal medical studies depend upon accurate preservation of detailed patient records for decades. In 1948, scientists began to study the residents of Framingham, Massachusetts[30] to understand the large increase in heart disease victims throughout the 1930s and 40s. Using data collected over decades of research, scientists discovered the major risk factors that modern medicine now knows contribute to heart disease. In 1975, the former USSR sent probes Venire 9 and 10 to the surface of Venus to collect data and imagery. The low quality images attracted little interest. About 28 years later, an American scientist used modern image processing algorithms on the diligently preserved data to reveal much more detail[31]. These timescales of many decades, even centuries, contrast with the typical 5-year lifetime for computing hardware and digital media. Beyond just scientific data, legislation such as Sarbanes-Oxley and HIPAA require many businesses to keep electronic records over decades. Consumers used to analog assets such as mail and photographs, which persist over many decades, and are now happily entrusting their digital versions to online services. The associated marketing literature encourages them to expect similar longevity.

There are many threats to preservation. While some apply also to short-term storage, others are unique to long-term preservation (e.g. media obsolescence). Large-scale disaster is an example. During the life of archival data, we must expect large-scale disasters (earthquakes, acts of war). Such disasters typically trigger other types of threats, such as media, hardware, and businesses faults, as was the case with many data centers affected by the 9/11 attack. Human error is another exampe. Users or operators may accidentally delete content they still need, or purposefully delete data for which they later discover a need.

Sometimes, the errors affect preservation hardware (losing tapes in transit), software (uninstalling a required driver), or infrastructure (turning off the air conditioning system in the

---

[29] T. Dawber, G. Meadors, and F. Moore. Epidemiological Approaches to Heart Disease: the Framingham Study. American Journal of Public Health, 41(3):279{81, Mar. 1951.

[30] T. Dawber, G. Meadors, and F. Moore. Epidemiological Approaches to Heart Disease: the Framingham Study. American Journal of Public Health, 41(3):279{81, Mar. 1951.

[31] D. Whitehouse. Reworked Images Reveal Hot Venus.News, Jan. 2004.

server room or swapping out the wrong disk in an array that has suffered a disk failure). Human error is increasingly the cause of system failures. Component faults can, and do, occur. Taking an end-to-end view of a system, any component may fail. Hardware components suffer transient, recoverable faults, such as temporary power loss, and catastrophic irrecoverable faults (e.g. a power surge destroys a controller card). Software components, including firmware in disks, suffer from bugs affecting stored data. Ingestion of data into a preservation system over the network may itself fail. External license servers or the companies that run them might no longer exist decades after an application and its data are archived. Domain names will vanish or be reassigned if the registrant fails to pay the registrar, and a persistent URL will not resolve if the resolver service fails to preserve its data with as much care as the storage system client.

Best Practice – Consider "Bit Rot' in the Next Generation Data Center

The storage medium is a vital component. No affordable digital storage media are completely reliable over long periods of time. They are subject to gradual accumulation of irrecoverable bit errors, often called bit rot, and to sudden irrecoverable loss of bulk data, such as disk crashes[32]. Bit rot is particularly troublesome, because it occurs without warning, and might not be detected until it is too late to make repairs. A familiar example might be CD-Rs. Manufacturers claim lifetimes up to 100 years, but even when properly stored, actual lifetimes may be only 2 to 5 years. Similarly, a previously readable disk sector can become unreadable, or may be readable, but contain the wrong information, due to firmware bugs or misplaced sector writes.

Media/hardware obsolescence. Over time, media and hardware components can become obsolete, no longer able to communicate with other system components, or irreplaceable. This problem is particularly acute for removable media, which though readable, may have outlived any suitable reader device. 9-track tape and 12-inch video laser discs are typical examples. The recently ubiquitous PC floppy is no longer part of industry specifications and will soon share their fate.

[32] N. Talagala. Characterizing Large Storage Systems: Error Behavior and Performance Benchmarks. PhD thesis, CS Div., Univ. of California at Berkeley, Berkeley, CA, USA, Oct. 1999.

Best Practice – Consider "Software or Format" longevity in the Next Generation Data Center

Another best practice is to consider software and/or format obsolescence. Software obsolescence is similar, often characterized as format obsolescence. The digital bits in which the data were encoded remain accessible, but the information can no longer be correctly interpreted. Proprietary formats, even popular ones, are equally vulnerable. For example, digital camera companies have proprietary, often undocumented "RAW" formats for recording camera data. When a company ceases to exist or to support its format, photographers can lose valuable data.

Loss of context. Metadata, or more general context, includes information about the subject and origin of content, the layout, location, and inter-relationships among stored objects, and the processes, algorithms, and software needed to manipulate them. Preserving context is as important as preserving the actual data, and it can even be hard to recognize all required context in time to collect it.

Best Practice – Consider encryption for long term retention in the Next Generation Data Center

Encrypted information is a particularly challenging example, since the decryption keys must be preserved, as well as the encrypted data. Unfortunately, over long periods of time, secrets (such as keys) can get lost, leak, or break. Short-term storage applications are less vulnerable; their assets rarely live long enough for the context to be lost or the information to become uninterruptable.

The other issue is attack of the digital data. Traditional repositories are subject to long-term malicious attack, and there is no reason to expect their digital equivalents to be exempt. Attacks include destruction, censorship, modification, and theft of repositories' contents and disruption of their services. The attacks can be short-term or long-term, legal or illegal, internal or external. The attacks can be motivated by ideological, political, financial, or legal factors, by bragging rights, or by employee dissatisfaction; you name it. Short-term storage is also vulnerable, but typically, abrupt, intense attacks are more noticeable and, in theory, one can better defend against them than slowly subversive attacks. Because much abuse of computer systems involves insiders, a digital preservation system must anticipate attack, even if it is completely isolated from external networks.

A Next-Generation Data Center system view of long-term storage must include not only the technology but also the businesses in which it is included. These businesses can die out, perhaps through bankruptcy or change missions, but the content needs to continue!

Best Practice – Consider Exit Strategies in the long-term preservation architecture of the NGDC

A best practice is to consider the planning and analysis of data retention “Exit Strategies". This methodology must address the possibility of the asset or the data being retained and preserved as well as the process of the content being transferred to a successor business.

During business changes, a large IT company closed a research lab and requested the lab's research projects be copied to tape and sent to another of its labs. Unfortunately, the tapes languished without documentation of their contents, because few knew about them. When it became clear that some of the project data would be useful to current researchers, enough time had passed that nobody could identify what would be on which tape, and the volume of data was too huge to reconstruct an index.

Storage services can also make mistakes, and assets dependent on a single service can be lost. An Ofoto user's digital photographs of her wedding were deleted when she did not make a purchase in the required interval. She had not updated her e-mail address, so did not receive the warning. Even if she had, at that time Ofoto provided no “data exit strategy" by which she could have retrieved her high-resolution originals.

Economic faults can also occur. Businesses often stretch their limited budgets, simply to get their collections online, leaving little or nothing to ensure continued accessibility. There are ongoing costs for power, cooling, bandwidth, system administration, equipment space, domain registration, renewal of equipment, and so on. Information in digital form is much more vulnerable to interruptions in the money supply than information on paper, and budgets for digital preservation must be expected to vary up and down, possibly even to zero, over time. These budget issues affect our ability to preserve as many collections as desired. Many libraries now subscribe to fewer serials and monographs. The lack of tools to predict these on-going costs makes it difficult to motivate an investment in preservation, especially if the future target audience does not exist at the time decisions are made. While budget is an issue in the purchase of any storage system, a shorter lifespan makes planning easier.

These threats are not new, so why are archives still losing data? A recent study for the National Archives noted the lack of explicit threat models for archival storage systems. Add complexity to that systematic design and an end-to-end perspective is lacking. These include visibility of faults, independence of faults, and unlimited budgets, as described below.

Best Practice – Consider fault visibility as it relates to riding the cloud for the next 100 years

While many faults are detected at the time an error causes them, some occur silently. For example, media errors are the best known of many sources of these latent faults. A disk sector might become unreadable, or bits might rot (rot is a technical term ☺), but this will not be detected until an attempt is made to read them. Silent media errors and faults occur more frequently than most realize. Silent block faults occur five (5) times as often as whole disk faults[33]. One study done of data from the Internet Archive (IA)[34] shows a smaller, but still significant rate. The Internet Archive is a 501(c)(3) non-profit that was founded to build an Internet library. Its purposes include offering permanent access for researchers, historians, scholars, people with disabilities, and the general public to historical collections that exist in digital format.

Overall, systems like the IA might supply users with data items at a high rate, but the average data item is accessed infrequently. Detecting loss or corruption only at user access times renders the average data item vulnerable to an accumulation of latent faults. Beyond media faults, the threats of fault visibility can lead to many types of latent faults:

Human error: Accidental deletion or overwrite might not be discovered until the affected material is needed.

Component failure: The reliance on a failed system component or a third-party component that is no longer available might not be discovered until data depending on that component are accessed.

Media/hardware obsolescence. Failure of an obsolete, seldom-used reader might not be discovered until information on its medium needs to be read. It might then be impossible or too costly to repair or replace the reader.

---

[33] T. Schwarz, Q. Xin, E. Miller, D. Long, A. Hospodor, and S. Ng. Disk Scrubbing in Large Archival Storage Systems. In MASCOTS, 2004.

[34] http://www.archive.org/about/about.php

Software/format obsolescence. Upon accessing old information, one might discover it is in a format belonging to an application one can no longer run. The loss of context or metadata can also cause issues for the extended archive. One might not discover we are missing crucial metadata about saved data until we try to make sense of them. For example, one might not have preserved an encryption key. Please see the section titled “Archival Ecosystem”, starting on page 94, for additional information on metadata encapsulation.
Deliberate attack: Results of a successful censorship or corruption attack on a data repository might never be discovered, or might only become apparent upon accessing the data long after the attack.

Best practice – Consider that Replication may not be the Holy Grail to data loss

Data replication is necessary for preventing data loss, but one can argue that it is not sufficient. Analysis of replication schemes is often based on the assumption that replicas fail independently. In practice, faults are not as independent as we might hope. Taking an end-to-end perspective, the threats of assumed independence provide many sources of fault correlation:

The first consideration is understanding the issues of a large-scale disaster: A single large disaster might destroy all replicas of the data. Geographic replication clearly helps, but care must be taken to ensure it provides sufficient independence. For example, the 9/11 disaster in New York City destroyed a data center. The system correctly failed over to a replica data center on the other side of the Hudson River, but unfortunately, the replicas were not independent enough. The chaos in the streets prevented users from getting to the backup. Eventually, it was unable to continue unattended.

The second consideration is human error. System administrators are human and fallible. Unfortunately, in most systems, they are also powerful, able to destroy or modify data without restriction. If all replicas are under unified administrative control, a single human error can cause faults at all of them.

The third consideration is Component and Media. If all replicas of the information depend on the same external component, the loss of that component causes correlated faults at every replica. With regard to media faults, temperature and vibrations of tightly packed devices in machine room racks are sources of correlated media faults.

The fourth consideration is business faults. Correlated business faults, which is the simultaneous corruption of data through various business units, including parent companies or subsidiaries that can cause multiple copies or all copies of an archive to fail at the same time.

Lastly, one might consider the biggest threats to digital preservation are economic, especially in the consumer space. Most of the information people would like to see live forever is not in the hands of businesses with unlimited budgets; it cannot be preserved by optimal, but expensive techniques such as synchronous mirroring of RAIDs across widely dispersed geographic replicas.

Best Practice – Model data loss to achieve Reliability and Resiliency in the NGDC

Abstract protection models, such as those for RAID, are useful for reasoning about the reliability of different replicated storage system designs. It is important to consider and focus on issues important to long-term storage, by incorporating the effect of latent and correlated faults on the overall reliability of an arbitrary unit of replicated data. The model proposed is agnostic to the unit of replication; it can be a bit, a sector, a file, a disk, a collection of objects, or an entire storage site. The goal is to attempt to develop a more abstract model that can be interpreted in a more general, holistic fashion. The model provides a conceptual methodology for reasoning about the relative impact of a broad range of faults, their detection times, their repair times, and their correlation. The model helps point out what strategies are most likely to increase reliability, and what data we need to measure to resolve trade-offs between these strategies. The model will consider the differences between the size of the unit of replication and the size of the fault. The damaged data may be bigger or smaller than the replication unit.

For example, while one might replicate data at the file level, a fault might only affect a few bytes of the file. Traditionally, in looking at block-level or disk-level replication strategies, faults have sometimes been assumed to affect a whole disk, even if some of the information is salvageable from the disk. This model separates replication size and fault size to make it possible to identify more generally data which is actually damaged.

The disk has traditionally been the standard unit of measure to work with, because manufacturers report MTTFs (Mean Time To Failure) for disks. In contrast, this model allows one to work at any granularity, incorporating more specific information about the likelihood of failures for different units of replication, as the information becomes available. Determining the

most effective, lowest cost method of replication for a system might require parallel analysis at several different choices of replication unit.

Best Practice – Consider Visible vs. Latent Faults to achieve Reliability and Resiliency in the NGDC

One can distinguish between immediately visible and latent faults as shown in Figure 34. Visible faults are those for which the time between their occurrence and detection is negligible. Examples of such faults include entire-disk or controller failures. We denote the mean time to a visible fault by MV and the associated mean time to repair by MRV (see symbol key in Table 3. Latent faults are those for which the time between occurrence and detection is significant. Examples include misdirected writes, bit rot, un-readable sectors, and obsolete data formats. We denote the mean time to a latent fault by ML, and mean time to repair by MRL. We only consider latent faults that are detectable; meaning they have a finite mean time between occurrence and detection, denoted by MDL.

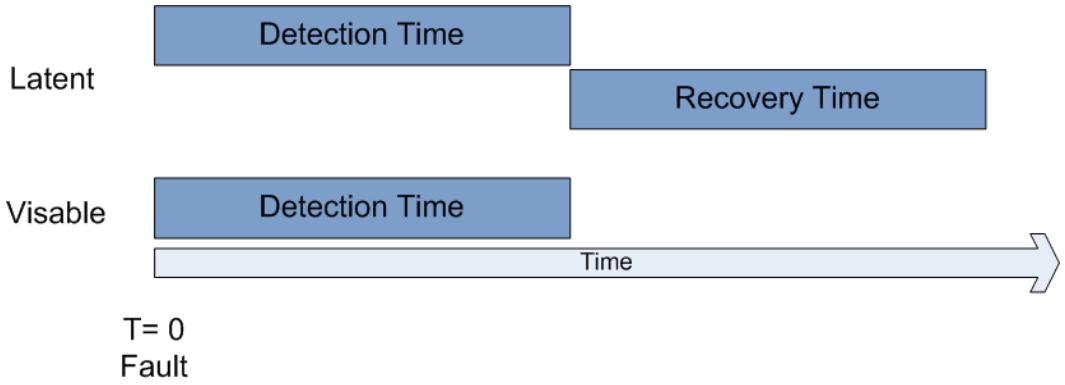


Figure 34 Types of replica faults

A	Temporal correlation factor
β_{ab}	Probability that a and b overlap spatially
MDL	Mean time to detect latent fault
ML	Mean time to latent fault
MRL	Mean time to repair latent fault
MRV	Mean time to repair visible fault
MTTDL	Mean time to data loss
MTTF	Mean time to fault
MV	Mean time to visible fault
WOV	Window of vulnerability
$a \cap_T b$	a coincides in time with b

Table 3 Key to Symbols and acronyms in the model

Assumptions in the best practice model

The mathematical representation of the model is initiated by starting with simple assumptions and adding complexity in stages. Following RAID, we start by assuming that the processes generating faults is "memory-less", meaning that there is no pre existing knowledge or data, as in the case of a feedback system. That is, as shown in Equation 4 – Probability of a fault as a function of time, the probability, P(t), of a fault occurring within time *t*, is independent of age. This assumption leads to the exponential distribution where MTTF is the "Mean Time To the Fault". For many aspects of this derivation, one can consider the case where t << MTTF, so the following approximation holds as shown in Equation 5 – Probability of a fault as a function of time approximation, below:

Equation 4 – Probability of a fault as a function of time

$$P(t) = 1 - e^{-t/MTTF}$$

Equation 5 – Probability of a fault as a function of time approximation

$$1 - e^{-t/MTTF} \approx 1 - \left(1 - \frac{t}{MTTF}\right) \cong \frac{t}{MTTF}$$

This approximation is used only to simplify the expression for the exponential in the probability and is not fundamental to the model. Initially, one can assume that all faults occur independently of one another, both temporally and spatially. Subsequently, we introduce correlated faults that are also exponentially distributed, but with an increased rate of occurrence. For simplicity, we model this increase by a multiplicative correlation factor, the same for both latent and visible faults. We also accommodate faults with an increased likelihood that resulting damage affects the same data in both replicas. This approach accounts for faults correlated either temporally or spatially, but does not account for cross correlations across space and time. These equations provide a mean-value analysis of reliability. If the distributions of faults, recovery times, or detection times are not exponential (i.e. bimodal), our use of means is inexact. Finally, assumptions are made about detection and repair of latent faults. Latent errors manifest themselves in two basic ways; as inaccessible data or as corrupted data. If data is inaccessible, then noticing the fault upon access is straightforward. We can repair this fault by creating a new copy from the remaining redundant copies. If the data are corrupted, then noticing the fault requires further work, such as comparing the data to a copy. To decide which copy is corrupted, one needs additional information. For instance, one might use the information that one copy decrypts or decompresses to something sensible, while the other produces gibberish. We might

replicate content more aggressively and use majority consensus among the replicas. One can also utilize collision-resistant checksums. However, checksums cannot be used to repair a damaged copy, they can only provide validation of correctness.

This model considers redundancy schemes that replicate the entire body of data. For most of this section, the focus is concentrated on mirroring technology, the simplest form of replication.

Mirrored data become irrecoverable when there are two successive faults, one in each copy, such that;

1. The second fault occurs before the initial fault can be repaired temporal overlap, and
2. The damaged portions of the copies intersect (spatial overlap).

MTTDL is the mean time to data loss, and 1/MTTDL is equal to the rate of double faults that lead to data loss. To evaluate the reliability of mirrored data, it is important to understand how mirroring is affected by visible, latent, and correlated faults. One needs to estimate the probability of temporal overlap, and subsequently, account for the probability of spatial overlap. Note that a double fault may occur without a user being aware of the data loss. Nevertheless, it still counts toward the double-fault rate.

One experiences temporal overlap when a fault occurs at a second replica during the “Window of Vulnerability” (WOV) after a fault at the first replica. The WOV is the time during which the first fault remains unrepaired. We denote this intersection in time by $\cap_T$. Since faults may be visible or latent, we need to consider the WOV after each type.

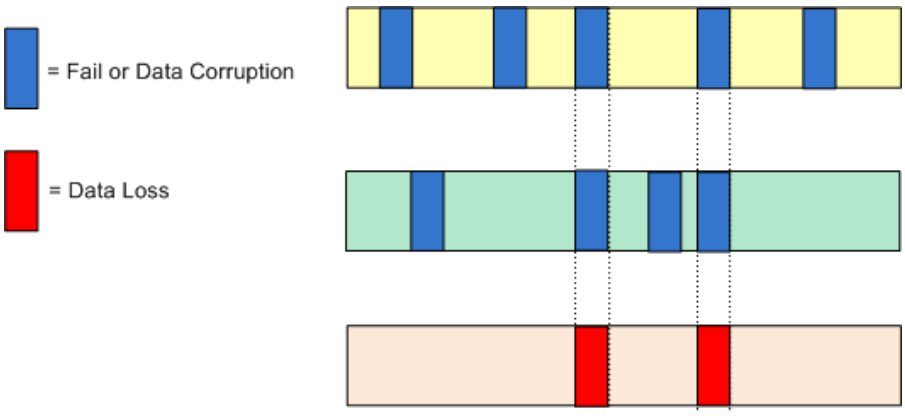


Figure 35 Spatial overlap of faults

First, consider the WOV after a visible fault, which on average is MRV. During this WOV, both latent and visible faults can occur. The probability is that another visible fault, overlaps in time with the WOV of the first fault,

To obtain MTTDL, we need the overall probability that a second fault overlaps with a first fault; hence, we next account for spatial overlap of the damaged areas, as illustrated in Figure 35. For instance, the likelihood that two faults overlapping in time would also damage the same materials in two replicas depends on their size and the areas they affect. Assuming that failures across replicas are not cross-correlated with space and time, we obtain the overall probability by multiplying each term for temporal overlap with a term, representing the probability that the faults overlap spatially. For example, for two successive visible faults, spatial overlap may be either physical or logical.

	Name	Mirrored Disk	Mirrored Archive
A	Temporal correlation factor	1	1
$B_{VV,LV,VL}$	Probability that. a and b overlap spatially	1	1/1795
ML	Mean time to latent fault	9.7 years	1531hrs
MRV	Mean time to repair visible fault	1.4hrs	4.4hrs

Table 4 Parameters used for Reliability estimation

An example of physical overlap is where the same sectors on mirrored disks suffer faults. An example of logical overlap is where the same files hosted at two sites suffer faults; the same materials, regardless of their physical layout, are damaged at both replicas. Moreover, β can be a function of time. The amount of damage in a replica will accrue over time, and the longer the detection and recovery times, the more likely it becomes that damage in the second replica overlaps with the damage in the first replica. To estimate the total double-fault failure rate, we multiply the rate of the first fault by the probability that a second fault overlaps with the first, and then sum the products for each fault type.

To account for temporally correlated faults, we assume that the probability of the second fault (conditioned on the occurrence of the first) is also exponentially distributed, but with a faster rate parameter.

To understand the implications of the spatial overlap of faults, we investigate the behavior of two systems, a single pair of mirrored disks and a large, geographically replicated archive. To do so, we parameterize the model with real data to explore the differences in reliability under three scenarios:

1) The effect of latent errors are ignored
2) Latent errors exist, but the system does not try to detect or repair them
3) Latent errors exist, and the system detects and repairs them.

Best Practice – Implement Data Scrubbing of active archives

Proactively searching for latent disk errors is often referred to as “scrubbing." Another term used is “auditing" to apply to all system levels and avoid confusion with deleting data. The parameters to calculate reliability are shown in Table 4, for each of the two systems. In a first example system, we consider replicated consumer ATA drives and take into account only faults from media errors. Assuming an exponential failure distribution and a 7 percent per year disk failure rate, we calculate an MV of 120K. We derive ML from one of several quoted worst-case unrecoverable error rates of 1 bit in 10^{15}. This is a read error rate, but for lack of additional information, one can assume this rate is the rate of latent errors. It can be shown that this bit error rate results in a .164 percent probability of an unrecoverable read of a 200 GB disk. Assuming the disk is 99 percent idle and supports a 40 MB/s transfer rate, we would read a disk capacity worth of data approximately 63 times a year. This results in an unrecoverable read once every 9.7 years, which we use for ML. The mean time to recovery (MRV) for mirrored disks (reading the whole of the surviving replica and in parallel writing a hot standby) takes about 1.4 hours. To calculate the spatial overlap parameters for this system, one can assume that all visible faults affect a whole disk, which is an entire replica.

A second example system consists of two geographically separate, just-a-bunch-of-disks (JBOD) replicas networked together, each with 1795 200GB drives. One can derive the parameters from aggregate statistics computed over a subset of this failure data.

Best Practice – Implement Latent and Visible fault detection to address long-term data retention

This simple model described previously reveals a number of strategies for reducing the probability of irrecoverable data loss:

1. Increase MV by, for example, using storage media less subject to catastrophic data loss.

2. Increase ML by, for example, using storage media less subject to data corruption.
3. Reduce MDL by, for example, auditing the data more frequently to detect latent data faults.
4. Reduce MRL by, for example, automatically repairing latent data faults rather than alerting a human operator to do so.
5. Reduce MRV by, for example, providing hot spare drives so that recovery can start immediately, rather than once a human operator has replaced a drive.
6. Increase the number of replicas enough to survive more simultaneous faults.
7. Increase α and decrease β by increasing the independence of the replicas.

While one generally describes these strategies in terms of the more familiar hardware and media faults, they are also applicable to other kinds of faults. For instance, in addition to detecting faults due to media errors, auditing can detect corruption and data loss due to attack. As another example, we can use a similar process of cycling through the data, albeit at a reduced frequency, to detect data in endangered formats and convert to new formats before we can no longer interpret the old formats. In the rest of this section, we examine the practicality and costs of some techniques for implementing these strategies.

Best Practice – Increase MV and ML to increase Reliability and Resiliency

A best practice is that increasing MV and ML will provide higher reliability, but the cost trade-offs must be considered in each case. For example, should we build an archive using more reliable, but more expensive, enterprise drives, or use lower-cost consumer drives and more replication? Based on Seagate's specifications[35], a 200GB consumer Barracuda drive has a 7 percent visible fault probability in a 5-year service life, whereas a 146GB enterprise Cheetah has a 3 percent fault probability. However, the Cheetah costs about 14 times as much per byte ($8.20/GB versus $0.57/GB, prices from 8/13/10). The Barracuda has a quoted irrecoverable bit error rate of $< 10^{14}$ and the Cheetah of $< 10^{15}$. Instead of using Cheetahs, we could implement the archive, using RAID5 arrays of Barracudas, increasing the archive's MV, and replicate the entire archive, increasing MTTDL given the MV.

[35] Seagate. ST3200822A Configuration and Specifications. http://www.seagate.com/support/disc/specs/ata/st3200822a.html, Sept. 2003.

Best Practice – Reduce MDL to increase Reliability and Resiliency

A potentially less expensive approach to addressing latent faults is to detect the faults as soon as possible and repair them. The only way to detect these faults is to audit the replicas by reading the data, and either computing checksums or comparing against other replicas. Assuming (unrealistically) that the detection process is perfect and the latent faults occur randomly, MDL will be half the interval between audits, so the way to reduce it is to audit more frequently. In other words, one can reduce MDL by devoting more disk read bandwidth to auditing, and less to reading the data.

Recent work suggests that in many systems, a reasonable balance of auditing versus normal system usage can be achieved. Online replicas, such as disk copies, have two substantial advantages over offline copies, such as tape backups. First, the cost of auditing an offline copy includes the cost of retrieving it from storage, mounting it in a reader, dismounting it, and returning it to storage. This can be considerable, especially if the offline copy is in secure off-site storage. Second, online media are designed to be accessed frequently and automatically. Auditing offline copies, on the other hand, is a significant cause of highly correlated faults, from the error-prone human handling of media to the media degradation caused by the reading process. The audit strategy is particularly important in the case of digital preservation systems, where the probability that an individual data item will ever be accessed by a user during a disk lifetime is astonishingly small. The system cannot depend on user access to trigger fault detection and recovery, because during the long time between accesses latent faults will build up enough to swamp recovery mechanisms. A system must therefore proactively audit its replicas to minimize MDL.

Note that relying on offline replicas for security is not foolproof. Offline storage may reduce the chances of some attacks, but it may still be vulnerable to insider attacks. Because it is harder to audit, the damage due to such attacks may persist for longer.

Best Practice – Reduce MRL or MRV to increase Reliability and Resiliency

Reducing the mean time to repair a visible fault is important in reducing the window of vulnerability, and therefore, in improving reliability. Although the mean time to repair a latent media fault will normally be far less than the mean time to detect it, it is important that we do not inadvertently increase MRL to be as big as MDL. Again, online replicas have the major advantage that repair times for media faults can be short and can be as quick as a few media

access times. No human intervention is needed, and the process of repair is in itself less likely to cause additional correlated faults. Repairing from o_-line media incurs the same high costs, long delays, and potential correlated faults as auditing off®line media.

Best Practice – Consideration and increase in using replication increasing reliability

Offline media are the most common approaches to increasing replication. However, the processes of auditing and recovering from faults using offline backup copies can be slow, expensive, and error-prone. Some options for disk-based replication strategies include replication within RAID systems, across RAID systems, and across simple mirrored replicas. Examples of this type of replication technology include EMC MirrorView™, SnapView™ and TimeFinder® products. Replication within RAID systems does not provide geographical or administrative independence of the replicas. If we require this type of independence, a global view of system reliability and costs must weigh the costs of increasing the reliability of individual sites versus increasing the level of geographic replication, using cheaper components. EMC Atmos™ is an example of using a large number of replicas on cheap disks.

Best Practice – Increase independence of storage-based systems

Based on a study done by Talagala[36], in just the six months of data analysis, many correlated faults were observed due to disks sharing power, cooling, and SCSI controllers, and systems sharing network resources. This analysis model suggests that, in most cases, even with far lower rates of correlated faults, increasing the independence of replicas is critical to increasing the reliability of long-term storage. Long-term storage systems can reduce the probability of correlated faults by striving for diversity in hardware, software, geographic location, and administration, and by avoiding dependence on third-party components and single businesses. Examples include hardware and disks in an array often coming from a single manufacturing batch, exhibiting similar fault characteristics. However, the increased cost incurred by giving up supply chain efficiencies of bulk purchase might make hardware diversity difficult. Note, though, that replacing all components of a large archival system at once is likely to be impossible. If new

[36] N. Talagala. Characterizing Large Storage Systems: Error Behavior and Performance Benchmarks. PhD thesis, CS Div., Univ. of California at Berkeley, Berkeley, CA, USA, Oct. 1999.

storage is added in “rolling procurements" over time[37], then differences in storage technologies and vendors over time naturally provide hardware heterogeneity.

Similar to hardware issues, software dependencies are another issue. Systems with the same software are vulnerable to epidemic failure. There have been studies that have shown that the natural diversity of systems within a data center or cloud can be used to reduce this vulnerability[38]. However, the increased costs caused by encouraging such diversity, in terms not merely of purchasing, but also of training and administration, might again make this a difficult option for some businesses.

Given the speed with which malware can find all networked systems sharing a vulnerability, increasing the diversity of both platform and application software is an effective strategy for increasing α or the “Temporal correlation factor”.

Another factor is geographic location. Many systems use offline backup store replicas offsite, despite the additional storage and handling charges that implies. A best practice for digital preservation systems is to establish each of their online replicas in a different location, again despite the possible increased operational costs of doing so. The EMC Atmos architecture, with the geographic attributes with multiple replicas, is a perfect fit addressing this increased independence solution.

Another factor is administration issues. Human error is a common cause of correlated faults among replicas. It is a best practice in ensuring that no single administrator will be able to affect more than one replica or copy. This is probably more effective and more cost-effective than attempts to implement “dual-key" administration, in which more than one administrator has to approve each potentially dangerous action. In a crisis, shared pre-conceptions are likely to cause both operators to make the same mistake[39].

---

[37] M. Baker, K. Keeton, and S. Martin. Why Traditional Storage Systems Don't Help Us Save Stu

Forever. In Proc. 1st IEEE Workshop on Hot Topics in System Dependability, 2005

[38] F. Junqueira, R. Bhagwan, A. Hevia, K. Marzullo, and G. M. Voelker. Surviving Internet Catastrophes. In Usenix Annual Technical Conference, 2005.

[39] J. Reason. Human Error. Cambridge University Press,1990.

Addressing component level designs should also be considered. System designs should avoid dependence on third-party components that might not themselves be preserved over time. Determining all sources of such dependence can be tricky, but some sources can be detected by running systems in isolation to see what breaks. For example, running a system in a network without a domain name service or certificate authority can determine whether the system is dependent on those services.

Business issues should also be considered. Offering increased independence is a best practice. Taking an end-to-end view of preservation systems, it is also important to support business independence. For example, if the importance of a collection extends beyond its current business, then there must be an easy and cost-effective “exit strategy" for the collection if the business ceases to exist. For example, quarrelling couples might not want to store their children’s photos on a home server that requires their continued cohabitation for access. This is a perfect use case for EMC Mozy. Not the best use case, but a practical one.

Unfortunately, these strategies are not necessarily orthogonal, and some can have adverse effects on reliability. Auditing media to detect latent faults increases reliability, but can introduce other channels for data corruption. An example could be attacking a distributed system through the audit protocol itself[40], which therefore, must be designed as carefully as any other distributed protocol. Automated recovery can reduce costs and speed up recovery times, but if buggy or compromised by an attacker, it can also introduce latent faults. This can be dangerous because even visible faults can now (though seemingly having been recovered) turn into latent ones. Strategies such as increasing the independence of administrative domains and diversity of hardware and software also come with increased costs and business overhead.

In summary, the main techniques for the Next-Generation Data Center and riding the cloud for long term reliability and resiliency within the cloud for increasing the reliability of long-term storage are; replication, independence of replicas, auditing replicas to detect latent faults, and automation to reduce repair time and cost.

---

[40] P. Maniatis, M. Roussopoulos, T. Giuli, D. S. H. Rosenthal, and M. Baker. The LOCKSS Peer-to-Peer Digital Preservation System. ACM TOCS, 23(1), Feb. 2005.

Green Stability

To say that there is no such thing as a data or computer recession would be an understatement. Demand to store more data for longer periods is driving the need for more data storage, which is second to servers in energy consumption and subsequent cooling demands. This section will also discuss storage and other related technologies building on recent EMC Proven papers describing IT sustainability, addressing IT data infrastructure efficiency, optimization along with power, cooling, floor-space and other related topics[41].

All of these factors aid in sustaining business growth while building on infrastructure resource management functions, including data protection, business continuance and disaster recovery (BC/DR), storage allocation, data movement, and migration, along with server, storage, and networking virtualization topics.

After facilities cooling for all IT equipment and server energy usage, external data storage has the next largest impact on power, cooling, floor space, and environmental (PCFE) considerations in most environments. In addition to being one of the large users of electrical power and floor space, with corresponding environmental impact, the amount of data being stored and the size of its data footprint continue to expand.

Though more data can be stored in the same or smaller physical footprint than in the past, thus requiring less power and cooling, data growth rates necessary to sustain business growth, enhanced IT service delivery, and new applications are placing continued demands on available PCFE resources.

A key driver for the increase in demand for data storage is that more data is being generated and stored for longer periods of time as well as more copies of data in multiple locations as shown in Figure 36. This trend toward increasing data storage will likely not slow anytime soon for businesses of all sizes.

The data footprint is the total data storage needed to support application and information needs and provides a perspective of where and how storage is used in a typical data center. Your data

[41] 2010 Proven Professional Paper titled “Above The Clouds - Best practices to Create a Sustainable Computing Infrastructure to Achieve Business Value and Growth”

footprint may, in fact, be larger than the actual amount of data storage you have, or you may have more aggregated data storage capacity than actual data. A general approach to determine your data footprint is simply to add up all of your online, near-line, and offline data storage (disk and tape) capacity. For example, consider all the data being stored at home on personal computers and laptops, PDAs, digital cameras and video recorders, TiVo sets and DVRs, USB fixed and removable disk drives, among other media that support various data and information needs.

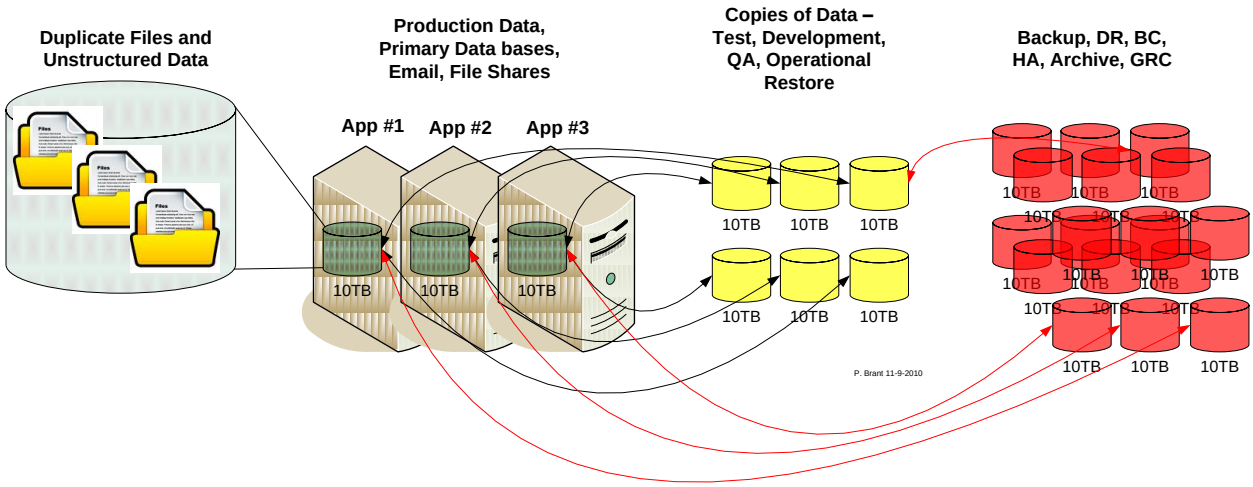


Figure 36 The Sustainable Data Footprint

Suppose that a business has 20 TB of data storage space that is allocated and being used for databases, email, home directories, shared documents, engineering documents, financial, and other data in different formats, both structured and unstructured. For these 20 TB of data, the storage space is probably not 100 percent used; database tables may be sparsely allocated, and there is likely duplicate data in email and shared document folders. However, to keep the example straightforward, assume that of the 20 TB, two complete copies are required for BC/DR purposes, and 10 TB are duplicated to three different areas on a regular basis for application testing, training, and business analysis and reporting.

The overall data footprint, (see Figure 36) is the total amount of data, including all copies plus the additional storage required to support that data, such as extra disks for redundant array of independent disks (RAID) protection or remote mirroring. In this overly simplified example, the data footprint and subsequent storage requirement amount to several times the 20 TB of data. And the larger the data footprint, the more data storage capacity and performance bandwidth

are needed and that have to be powered, cooled, and housed in a rack or cabinet on a floor somewhere.

In the past, it was debatable as to how much energy in a typical data center is actually consumed by storage (internal to servers and external) as well as how much data is active or inactive. Today, with advanced tools such as EMC's Power Calculator[42], the ability to have specific power and cooling requirements have become much easier.

The major power draws for common storage systems are usually spinning hard disk drives (HDDs) and their enclosures, which account for, on average, 66–75 percent; controllers and related I/O connectivity components generally account for most of the balance of electrical power consumption. Consequently, data storage is an important area for energy optimization and efficiency improvements.

Best Practice – Implement Tiered Storage with Balancing Application Response Times with PCFE in mind

Given the different characteristics and application service requirements for data, different types of storage supporting online active data along with inactive idle data that vary in performance should be considered as a best practice. Availability, capacity, and PCFE (power, cooling, floor space, and environmental) and in general energy consumption per price point and category of storage are all variables in the efficiency equation. To address the different data access and activity patterns and points in the data life cycle, virtualization provides a means to abstract the different tiers and categories of storage to simplify management and enable the most efficient type of storage to be used for the task at hand.

Tiered storage is an umbrella term and is often referred to by type of HDD, price band, or architecture. Tiered storage embraces tiered media, including different types and classes of HDDs, which vary in performance, availability, capacity, and energy usage. Other storage media such as SSDs, magnetic tape, and optical and holographic storage devices are also used in tiered storage.

[42] http://powercalculator.emc.com/Main.aspx, Note that one needs a EMC Power Link Account to obtain access to this tool

Tiered storage—various types of storage media configured for different levels of performance, availability, capacity, and energy (PACE)—is a means to align the appropriate type of IT resources to a given set of application service requirements. Price bands are a way of categorizing disk storage systems based on price to align with various markets and usage scenarios. (e.g. consumer, small office/home office, and low-end small to medium-size business (SMB) in a price band of under $6,000; mid- to high-end SMB in middle price bands into the low $100,000 range; and small to large enterprise systems ranging from a few hundred thousand dollars to millions of dollars).

Best Practice – Utilize Tiering Utilities to Achieve Efficient Use of Storage

A best practice to achieve green stability is to utilize tiering methodologies into the next generation data center. EMC offers a utility that provides best of breed performance optimization and efficiency with a utility called Tier Advisor.

Tier Advisor is a utility that estimates the performance and cost of mixing different types of disk drive technology within EMC storage arrays. It helps one determine whether to add storage, upgrade storage, or replace existing storage to achieve optimal performance and cost effectiveness within a storage environment. This utility is particularly useful when planning to implement the Fully Automated Storage Tiering (FAST) solution. FAST optimizes the use of different disk types, or storage tiers, in a Symmetrix array by placing the right data in the right tier at the right time.

Tier Advisor models an optimal storage array configuration by enabling one to interactively experiment with different storage tiers and storage policies until achieving your desired cost and performance preferences. The Tier Advisor helps you define the number of disk drives to use for each disk drive technology when configuring a tiered storage solution.

After the storage array configuration is defined, a recommend best practice is to use the EMC "SymmMerge" utility to create a detailed configuration plan. This assigns logical devices to physical disks to prepare for the migration process.

The term tier can have different meaning when used in information management vs. other IT environments. Fundamentally, the term “Tier” as it relates to information storage is the SLA’s (Service Level Agreements) as it is associated with protection, performance, and price (the

three P's). From a business perspective, the term tier refers to an application classification that reflects the desired service levels for that application. For example, for an Oracle Financial application, a #1 or highest tier is considered. For an email archiving application, a #3 or #4 Tier may be used.

Given an application's tier, the application and its desired service levels for performance and availability may be targeted by matching each of the application's data volumes to the appropriate storage resources.

In the historical storage view, a tier is a storage platform such as Symmetrix and CLARiiON. In the contemporary storage view, a tier represents a set of storage resources with the same technology residing in a storage platform. Simply put, a tier can be a set of SATA (Serial Advanced Technology Attachment), FC (Fiber Channel), or EFD (Enterprise Flash Disk) disks within the same storage array.

To avoid confusion with the historical view of a tier, a new term called storage type is used for Tier Advisor. In this document, the term tier and the term storage type are used interchangeably to refer to the contemporary storage view of a tier.

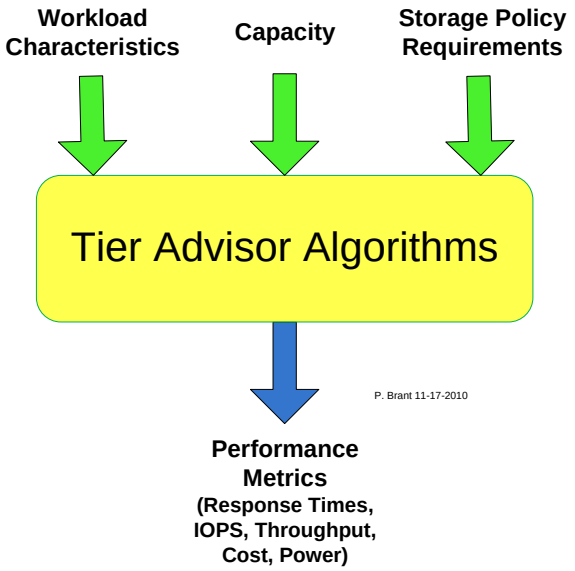


Figure 37 Tier Advisor Analysis Data Flow

As shown in Figure 37, Tier Advisor is an analytical performance model that uses the following three categories of input parameters: workload characteristics, capacity, and user-defined storage policies.

Figure 37 depicts that the workload characteristics of the existing storage array is used as one of the inputs and is based on the average number of I/Os per second, the write percentage, the average number of MB per second and the Skew percentage (distribution of I/Os over used capacity).

In addition, the capacity is calculated in terabytes by the number of logical devices and the addressable size of these devices. A custom mode that enables manual input workload characteristics and capacity information, and a data mode that processes performance statistical data from other sources, such as files from EMC Workload Analyzer (WLA) and Symmetrix CLI statistics collection daemon (STP) is also available. For both modes, Tier Advisor provides immediate feedback on the different values entered for the workload input parameters.

Lastly, user-defined storage policies are also considered as part of the input parameters that define the physical characteristics of a storage array to be modeled and the usage constraints. These include the parameter values for disks, tiers, and tier usage policies in the storage system. The storage designer defines these parameters manually and then adjusts them dynamically during the Tier Advisor modeling exercise.

Tier Advisor provides a list of factory disks and other available disks for purchase along with a price list. The disks are categorized by type (EFD, FC, and SATA) and capacity. Price information is embedded in the Tier Advisor application and is used for ‘what-if’ financial decisions. The EMC pre-sales group can provide this information to the customer as required.

The output of the Tier Advisor utility are various what-if’s as they relate to different options to make a storage array more efficient and able to meet the business SLA in a more cost-effective manner outlined by the following parameters;

1. Performance
2. Cost
3. Power consumption
4. Raw capacity
5. Disk count
6. Relative Disk Service Time vs. I/O graph

All output results are presented as relative values, indicating the relative position of each policy result to the reference policy, except for the number of disks. For example, if policy A is set as a reference policy, all of its relative values are set to reference point 1. All other policies will have results that are relative to the reference point. These relative results can be displayed as a numeric ratio or as a percent difference, up to double the reference amount.

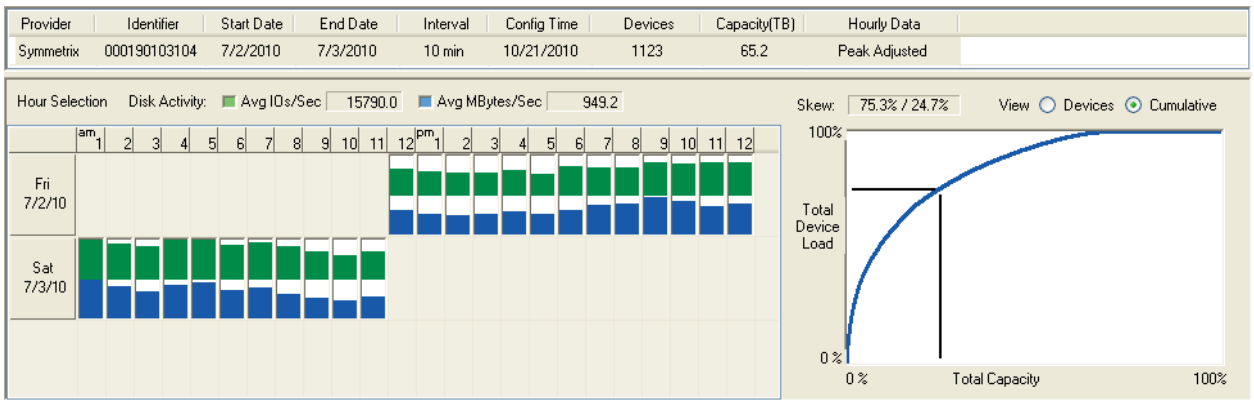


Figure 38 Tier Advisor Input Data

Figure 38 shows the STP data input from the existing Symmetrix storage array. Notice that the sample period is over a two day period, but the length of samples can be extended or reduced as required. Also note that the diagram also shows the I/O skew. In this case, only 25 percent of the devices are doing most of the work, showing a very typical workload. This is indicative of a tiered optimized solution.

The diagram in Figure 39 indicates a sample output generated by Tier Advisor. The dotted line in this figure is associated with a storage policy named Baseline, indicating this policy is a reference policy and all the relative values of the Baseline policy are set to one. The Baseline policy has 100 percent of the storage capacity allocated in the storage tier named Fiber 3R5. The Drive Optimization policy has 3 percent of the capacity allocated to the Flash 3R5tier, 40 percent in Fiber 3R5, and 57 percent to the SATA tier. After the algorithms process the data and match the devices to the disk drives in the storage tiers, the following is calculated and shown in Figure 9. The relative Disk Service Time in the Drive Optimization policy is 0.85. This means that the estimated response time of the described drive mix can be 15 percent faster in comparison to the Baseline policy, which is a good thing.

Note that the Relative Service Time is the only relative value affected by the position of the I/O Rate marker in the Relative Disk Service Time vs. I/O Rate Chart. One can move the I/O Rate marker to the right or left, increasing or decreasing the relative value. The initial position of the I/O Rate value indicates the average I/O Rate of the intervals analyzed.

The relative Cost of Acquisition of the disk drives in the Drive Optimization policy has a value of 1.3, which is 30 percent higher than defined in the Baseline policy. The Relative Cost refers to the cost of acquisition of disk drives and software. The cost of software is affected by tiered storage, that is, the software customized for the SATA tier is discounted. The Total Cost of Ownership is also affected by other operating expenses such as software maintenance and power consumption. Operating expenses are not considered by the application.

The utility also has the ability to create storage policies allowing a more dynamic 'what if' ability to define the best mix for the individual use case.

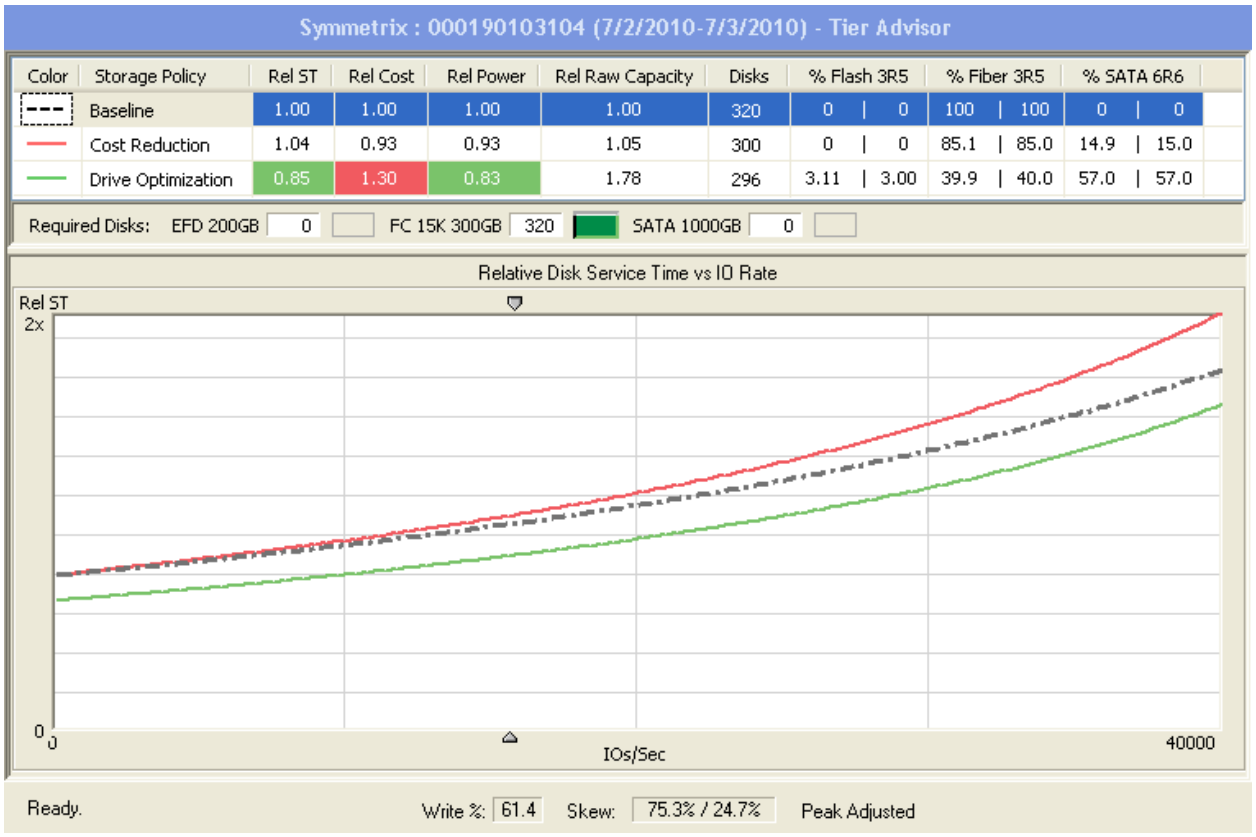


Figure 39 Tier Advisor Optimization Analysis

Relative Power Consumption, related to the disk drives, is reduced by .83 or 17 percent in the Drive Optimization Policy. The Relative Raw Disk capacity in the Drive Optimization policy is 10 percent higher. The increase in Raw Disk capacity can happen as a result of a change of RAID protection. For example, moving from RAID 5 7+1 to RAID 6 6+2 consumes more raw space. However, the increase in raw capacity can also happen when Tier Advisor must short stroke RAID groups to accommodate the I/O workload.

An increase in Raw Capacity can have a significant effect on the relative cost of the configuration. Using a tool such as this, bringing a storage array in line to best practices in terms of efficiency, power, and cooling while maintaining the same or better service levels and response times is a given for the Next Generation Data Center.

Business and Vertical Use Case

The third and final Triad as it relates to Riding the Cloud and focusing on transformation of the Next-Generation Data Center is to understand what specific business models need to be addressed and to understand how the sustainable reference models and attributes and technologies previously discussed, can drive value to the CIO or upper management in general.

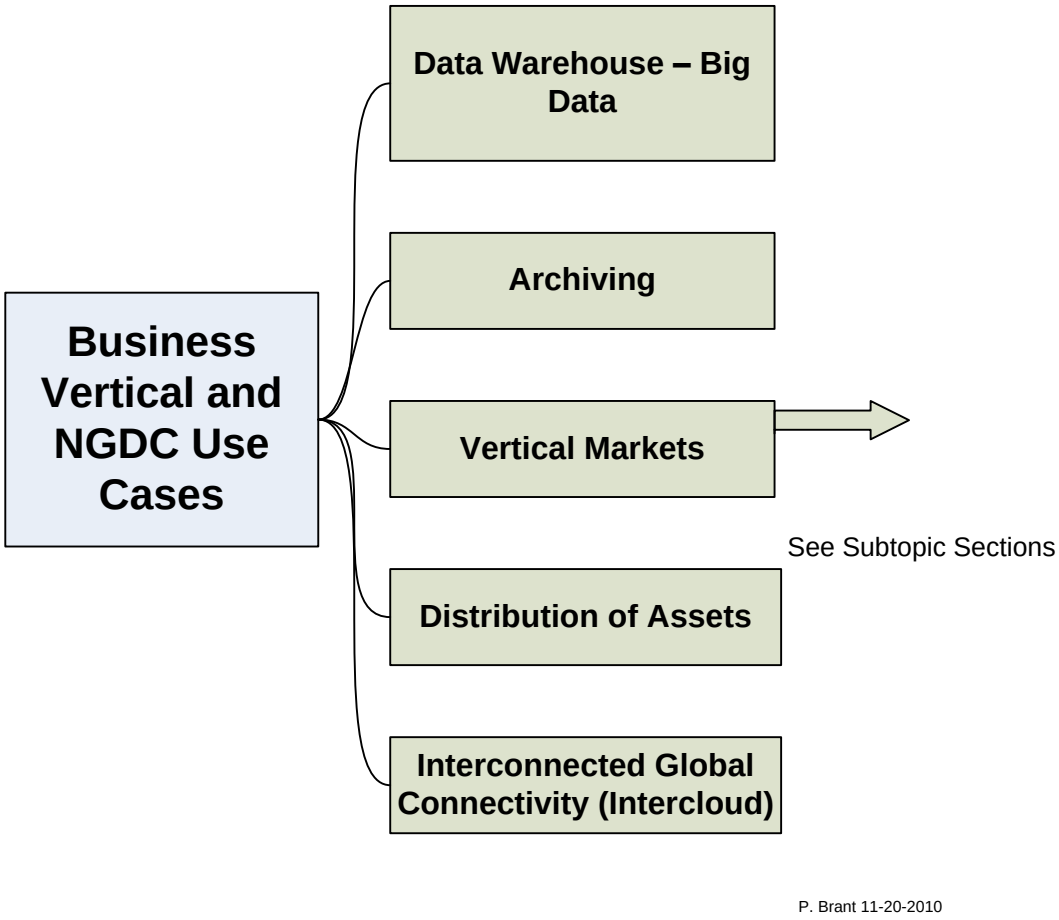


Figure 40 Business Vertical and NGDC Use Case Taxonomy

As shown in Figure 40, many vertical markets and specific use cases need to be addressed. Varying IT business drivers dictate the Data Center Transformation path. Each business vertical as shown in the Taxonomy diagram shown in Figure 41 has different and in many cases, unique requirements based on which market the cloud and/or transformation is needed.

Riding the cloud will require an understanding of the challenges of dealing with the Data Warehouse (DW) juggernaut with “Big Data” requirements. How does one distribute ones IT assets? Distribution of assets describes the challenges and benefits of geographically or locally

allocating applications and infrastructure utilizing the cloud and/or other game-changing data transmission technology in a new way. How does one archive appropriately and ensure that security and compliance are considered? In addition, how does a business get ready for the third Internet wave called the “Inter-Cloud”? All this and more will be discussed in the following sections.

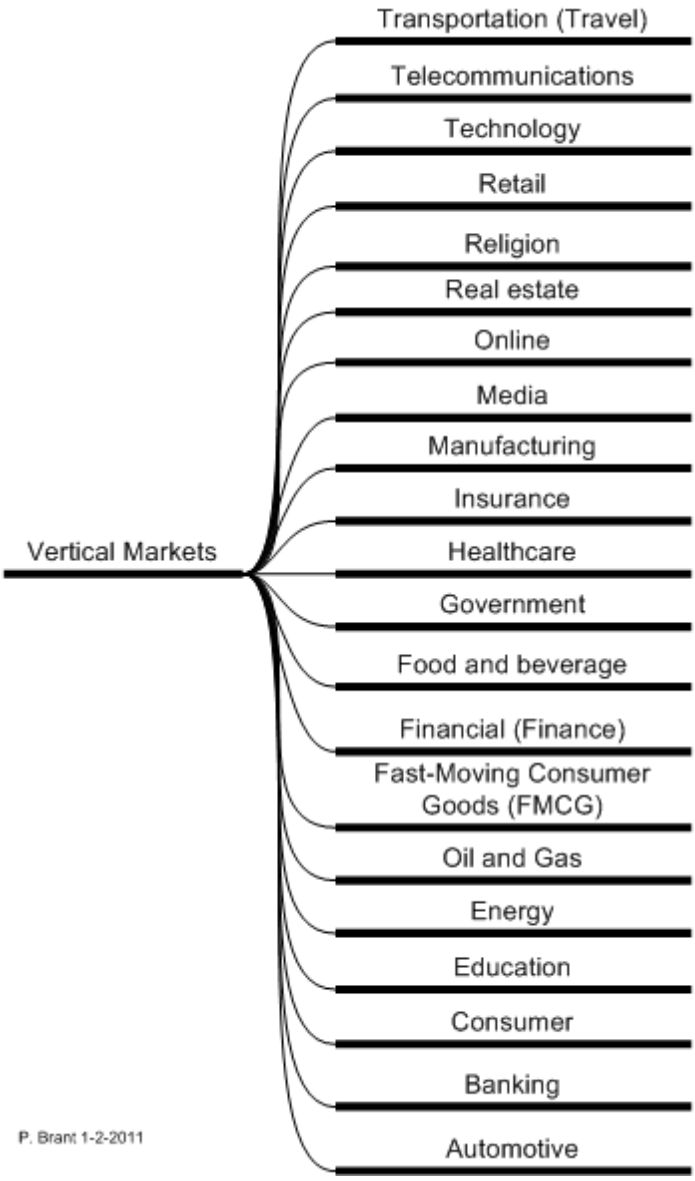


Figure 41 Vertical market Taxonomy

Data Warehouse – Big Data

With the rise of both the Internet-driven global economy and wireless communications, and increasing demands for regulatory compliance, data is exploding in the enterprise. A terabyte (TB) is not a lot of data anymore and definitely not in the Next-Generation Data Center. As data warehouses in the hundreds of terabytes, even petabytes, become more common, companies are realizing the value and sheer depth of this data, and need to develop a strategy to quickly and easily analyze it. This way companies can use this data to leverage customer interactions, optimize business processes, and sustain a competitive advantage.

Multi-terabyte data warehouses, which can cost millions of dollars to implement and tend to be available only to small groups now must be accessible to everyone. This is true from top-level management to frontline employees. The challenge is to make the data available in the most cost-effective and open manner possible. As a result, today's data warehouse systems need to be high performance, simple to implement, scalable, flexible, standards-based, and easy to maintain.

In addition, with the proliferation of various input entities such as SaaS applications, networked sensors, handheld devices, and other technological advances, data is being ingested by data centers from a rapidly-increasing number of endpoints. The result has been a "big data" log jam, where the sheer amount of data is making it difficult and awkward to capture, store, search, share, analyze, and visualize.

The Next-Generation Data Center needs to embrace current and future Data Warehouse architectures. One may have noticed that there are many new options for data warehouse platforms that have appeared this decade. We have seen the emergence of new categories of data warehouse platforms, such as data warehouse appliances and software appliances. Please refer to the section titled "Applianceization", starting on page 47 for additional information on this topic. A new interest in columnar databases has led to several new vendor products and renewed interest in older ones. Open source Linux is now common in data warehousing, and open source databases, data integration tools, and reporting platforms have come out of nowhere to establish a firm foothold. In the hardware realm, 64-bit computing has enabled larger in-memory data caches, and more vendors now offer MPP architectures. Please refer to the section titled "Massively Parallel Processing (MPP)", starting on page 37 for additional information on this topic.

Leading database vendors have added more features and products conducive to data warehousing. Those are mostly features within the data warehouse platform, especially its database. There are also growing practices that are demanding support from the platform, including real-time integration between the data warehouse platform and operational applications, various types of advanced analytics, and reusable interfaces exposed through Web services or service-oriented architecture (SOA). In addition, a number of data warehouse platforms and other business intelligence platforms are now readily available through software-as-a-service (SaaS) and cloud computing focused on the Next-Generation Data Center.

The good news is that the options for data warehouse platforms have recently become far more numerous. The bad news is that it is difficult for data warehouse professionals and their business sponsors to keep track of these advancements and select those appropriate for their needs.

To help understand how data warehouse or “Big Data” architectures can address NGDC’s needs, it is important to understand the many new options available. As will be discussed, many businesses are planning the next generation of their data warehouse, and this section provides information and best practices that can be instrumental for such planning. The focus from previous sections was on technology, but this section also explains how technology’s adoption in next-generation data warehouse data centers and platforms is driven by real-world business needs and requirements.

A data warehouse platform consists of one or more hardware servers, an operating system, a database management system (DBMS), and data storage. These communicate via a LAN or WAN, although a multi-node data warehouse platform may have its own specialized network. Note that a data warehouse platform manages a data warehouse, defined as a collection of metadata, data model, and data content, designed for the purposes of reporting, analyzing information, and making decisions. However, the data warehouse is not part of the platform per se. Please refer to Figure 42 Data Warehouse Architectures, below. All these components and more have seen generational advances in recent years.

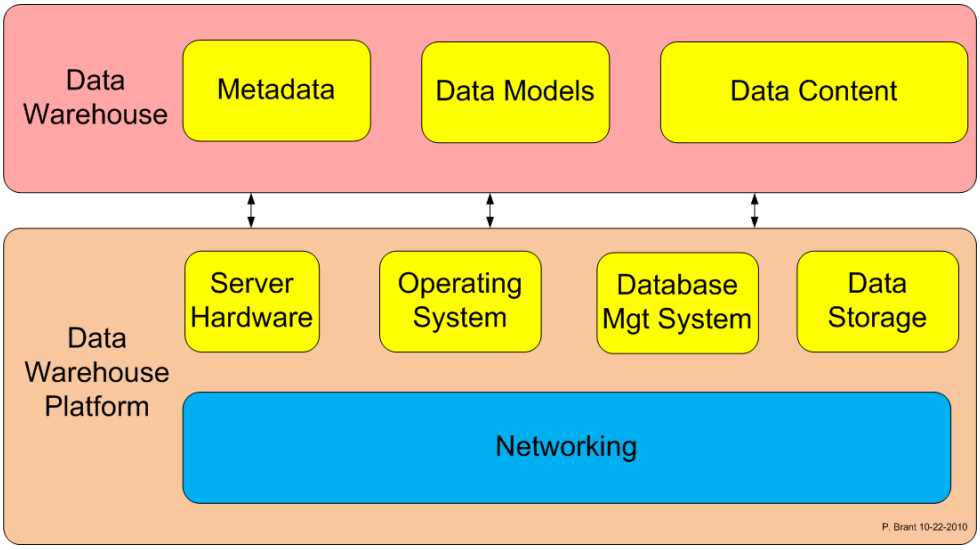


Figure 42 Data Warehouse Architectures

It is important to consider that relatively new technologies, techniques, and business practices are driving the majority of data warehouses and their platforms toward a redesign, major retrofit, or even replacement that we can recognize as a generation. The current generation of a data warehouse will bring about the next generation "Big Data" solution as we advance transforming how we all analyze data. In many cases, generational change is an evolutionary process that adapts the resulting data warehouse to changing business and technology requirements. In fact, generational change is often driven by these requirements. In other cases, generational change is more of a maturation process that steps a data warehouse through multiple stages of a lifecycle.

What is next for a given businesses' data warehouse platform can vary tremendously. For example, a next-generation data warehouse platform may tap into leading-edge features, such as appliances, open source, and cloud computing. It may simply get you caught up with somewhat more established practices for real-time operation, advanced analytics, and services. Sometimes, the next generation addresses administrative issues, such as hardware upgrades (from 32-bit to 64-bit), data migrations (from one DBMS to another), or architectural changes from SMP (Symmetrical Multi processing) to MPP "Massively Parallel Processing". Some argue that MPP has limitations and constraints. This was discussed in the section titled "Best Practice – Understand Advantages in MPP (massively parallel processing) in addressing the NGDC. Keep in mind though, a next generation data warehouse platform is a relative concept. It

depends on where you are starting, what new requirements you must address, and how many resources you have.

Best Practice – Understand business drivers and how DW will change process requirements

Businesses face change more often than ever before. Recent history has seen businesses repeatedly adjusting to boom-and-bust economies, a recession, financial crises, and shifts in global dynamics or competitive pressures. Increasingly, businesses rely on the data warehouse and related business intelligence infrastructure to understand change and react appropriately.

DW platforms need updating to support changing business requirements. In fact, many of the technologies associated with the next-generation DW relate to change in some way, such as advanced analytics, scalable architectures, virtualization methods, reusable services, real-time integration with operational applications, and so on. Successful DWs mature through multiple lifecycle stages. This usually provokes changes in the underlying DW platform and elsewhere in the business intelligence (BI) infrastructure.

There is more than one path to the next-generation data warehouse platform. One option is to retain the current platform, but do more with it. Many users still have not tapped all the capabilities of their current platforms. In situations of early-to-mid-life project phases of designs, it was time (defined by business readiness) to embrace the platform's more sophisticated capabilities, especially real-time functions, data federation, in-memory processing, and analytics (whether based on OLAP or data mining). Since there is a difference between a data warehouse and the platform that manages it, one can remodel the DW significantly to add value without replacing the platform. Some new generations involve tools that are peripheral to the platform, such as solutions for data integration, quality, master data, reporting, and so on. Incremental additions to hardware are common (to add more CPUs, memory, or storage), and these satisfy next generation requirements (fast queries, in-memory databases, and scalability) by doing more with the current platform.

Another option is to replace the current platform, then build out the new one. Compared to other approaches, ripping out and replacing a data warehouse platform is rather expensive for IT budgets and intrusive for business users. Therefore, this path to the next generation should be avoided, in general.

The Primary business drivers for the Next-Generation Data Center as it relates to data warehouses are;

- Analytics of various types help the business cope with change and discover opportunities. The increasing use of advanced analytics is driven by businesses' need to understand constantly changing business environments, as well as to discover opportunities for cost reductions and new sales targets.
- Real-time and related technologies are enablers of operational excellence. Because of economic, competitive, and quality issues, many businesses are under pressure to achieve unprecedented levels of business excellence. Many are pursuing this goal by embedding data and functionality available from their BI and data warehouse infrastructure into their operational and transactional applications. This enables time-sensitive business practices such as operational BI, on-demand dashboards and performance management, just-in-time inventory and manufacturing, and so on. Technology people talk about integrating BI and operational systems through real-time data warehousing, which they may call by other names such as active, dynamic, or on-demand data warehousing. However, these are just enabling technologies that help satisfy more fundamental business requirements for operational excellence.
- Scalability, in many senses, enables business growth. As businesses grow, as they automate more business processes with software, and as they depend more on BI and data warehousing, they generate more data that needs processing for BI and to run the business. Likewise, there is growth in user communities, reports, analyses, and so on. With the enterprise data warehouse at the heart of BI, and more and more at the heart of operational excellence, warehouse scalability has become a critical success factor across the board.
- Addressable memory space automates new time-sensitive business practices. In particular, various types of in-memory databases can now be far larger than before, and data operations in memory are far faster than those that involve input/output with disks. Imagine putting an entire data mart or warehouse in memory; reporting and analysis functions are now so fast that they are more easily embedded in operational applications. This speedy intelligence takes business practices to a new level, such as up-sell/cross-sell guidance in a telemarketing scenario, automated recommendations in

an e-commerce situation, improved service in call center and similar online applications, and fraud detection in a variety of contexts.

- Warehouse architecture and related practices affect a DW's ability to support a business. Most of the technology and business drivers mentioned here involve creative adjustments to the logical architecture of the data warehouse's data model and the systems architecture of its hardware configurations. When moving to a next-generation data warehouse platform, expect to make architectural changes to accommodate new technical functions and their related business requirements.

Data Management Best Practices

Some of the most pressing needs for Next-Generation Data Center data warehouse platforms involve technologies and practices that we generally do not think of as part of the platform. In particular, many users need to update the data management tools that process data for use through the data warehouse.

Best Practice – Implement Master Data Management and Data Quality in the NGDC or in the cloud

Master Data Management (MDM) is one of the highest priorities for data warehousing. A similar, but slightly less urgent, data management practice is data quality (DQ), which likewise should see strong growth, bolstered by committed users (69 percent plan to use it). There are good reasons why MDM and DQ are high priorities for Next-Generation Data Centers utilizing data warehouses.

MDM comprises a set of processes and tools which consistently define and manage the non-transactional and transactional data entities of an organization (also called reference data). MDM has the objective of providing processes for collecting, aggregating, matching, consolidating, quality-assuring, persisting, and distributing such data throughout an organization in such a way as to ensure consistency and control in the ongoing maintenance and application use of this information.[43] Data Quality is the process of making sure the data is complete, concise, and secure.

[43] http://en.wikipedia.org/wiki/Master_data_management

Most DWs still lack MDM and DQ functions. MDM and DQ certainly are not new, but they are still underutilized by data warehousing professionals. For many, extending data integration and other data management practices around the DW to include MDM and DQ is an act of catching up, not stepping out on the leading edge. The goal is to achieve quality decisions based on quality data. Tighter integration with operational systems demands MDM. In these situations, MDM ensures better integration by identifying data properly, so that the best sources are found for a specific use and "apples to apples" data exchanges are made. Likewise, some businesses appoint the enterprise data warehouse (EDW) as the "system of record" for master and reference data, which obviously requires a hefty MDM solution for the EDW.

Regulatory and financial reports must be squeaky clean and accurate. More and more, businesses produce these reports through their BI and DW infrastructure. These reports must be as accurate and credible as possible. Data management technologies such as DQ and MDM help achieve that end, though most DWs lack these today, forcing a generational change. Given the current economic and political environment, regulatory requirements are set to increase rapidly in the next few years.

Best Practice – implement In-Memory Processing and 64-Bit Computing

Sixty-four-bit systems typically offer faster CPUs and more power-efficient hardware than older systems. However, for DW professionals, the most compelling benefit of 64-bit systems is the large space of addressable memory. For example, a leading reason for upgrading to 64-bit systems is to deploy an in-memory database for reporting or analytic applications that need very fast query response. In-memory databases provide such speed because they do not have disk input/output (I/O) to slow them down, even though EFD's (Enterprise Flash Drives) are helping considerably in this area. Also note that EMC, as an example, has proven solutions in this area. The in-memory database is usually a function of a DBMS, but some BI platforms for reporting and analysis also support in-memory data stores and related processing. Tools for extract, transform, and load (ETL) commonly support in-memory processing in a 64- bit environment, so that complex joins and transformations are executed in a large memory space without the need to land data to disk in temporary tables. This makes an ETL data flow a true "pipe", which means the ETL tool can scale up to large data volumes that are processed in relatively short time periods.

Disks are not going away; they are needed for storage, and in-memory data structures are typically loaded from disk. One of the barriers is that main memory is still rather expensive. As the price comes down, however, data warehousing should experience an upswing for in-memory processing, and DW platforms of the future will commonly support multi-terabyte memory spaces. Furthermore, as previously mentioned, solid-state disks (which have many of the performance characteristics of memory) now have a foothold in the IT market, and they are also coming down in price.

In many ways, MPP architectures are an alternative to 64-bit-based, in-memory processing, because MPP pools memory resources from many servers (whether 32- or 64-bit) to create a large virtual space. In-memory databases and the 64-bit hardware and software that enable them are high priorities for next-generation data warehouse platforms. Whenever BI/DW teams plan a next-generation data warehouse platform, 64-bit systems and in-memory processing should be a required part of the design. As an example of a relevant vendor product, the Oracle Database Machine is a 64-bit platform that leverages the extended in-memory capabilities of Oracle Database 11g R2.

Best Practice – Implement Open Source Software in NGDC Data Warehouse Designs

Open source tools are being used at an unprecedented level in business intelligence, data integration, and data warehousing. There are good reasons for the upswing of open source software used in data warehousing. They are:

1. The recent recession has driven up interest in low-cost open source software
2. Open source tools are coming into a new level of maturity
3. Open source software augments traditional enterprise software without replacing it

However, there are three leading barriers to using open source software. They are:

1. getting adequate support and maintenance, which typically occurs when selling the software to peers
2. management
3. learning a new tool

The three leading benefits of open source are:

1. low price
2. the ability to download and try the software before buying
3. a community of collaborative developers

Best Practice – Consider implementing a "Hadoop" or Equivalent Framework in your DW Design

One open source framework that has had much traction is called "Hadoop". Hadoop[44] is software that implements a reliable, scalable, distributed computing solution that includes the following functionality:

- A High Performance Distributed File system (HDFS) that provides high throughput access to application data.
- A software framework (MapReduce) for distributed processing of large data sets on compute clusters.
- A high-performance coordination service (ZooKeeper) for distributed applications.
- A data serialization system (Avro).
- A data collection system (Chukwa) for managing large distributed systems.
- A scalable, distributed database (HBase) that supports structured data storage for large tables.
- A data warehouse infrastructure (Hive) that provides data summarization and ad hoc querying.
- A Scalable machine learning and data mining library (Mahout).

Best Practice – Implement Advanced Analytics Methodologies

In the next few years, advanced analytics will experience growth, bolstered by the strongest commitment yet seen in usage of advanced analytics driven by businesses' need to understand constantly changing business environments, as well as to discover opportunities for cost reductions and new sales targets. There are different analytic methods users can choose as they move beyond basic online analytic processing (OLAP)-based methods and into advanced analytics. Some users choose advanced analytic methods based on data mining, predictive analytics, statistics, artificial intelligence, and so on.

The majority of users, however, seem to be choosing SQL-based methods. This is probably because they know and trust SQL, and can leverage the SQL-based tools and skills they already have. This trend is pushing SQL-based analytics to a new extreme. With "load and go" methods, users quickly load a few terabytes of raw operational data and go at it with ad hoc queries, until the data reveals the answers they need. The ad hoc queries get more complex

[44] http://hadoop.apache.org/

with each iteration by a business analyst or similar power user. This method does not allow time and resources for data transformation, cleansing, or remodeling, so users compensate with lots of WHERE clauses, table joins, and temporary tables (when necessary).

SQL-based analysis at this advanced level is powerful, but it succeeds only when supported by a DBMS that can quickly execute extremely complex SQL statements run against multi-terabyte volumes of raw data, in a schema-neutral fashion that supports "load and go" practices. Many businesses depend on an enterprise data warehouse (EDW) to fulfill most of their analytic requirements. The problem is that most EDWs are optimized for standard reports and recurring analytic questions, based on OLAP. It may be that EDWs have lost their analytic prowess as users have evolved them into reporting and OLAP databases. As a result, there is a need for DBMSs that can support free-form, ad hoc analysis against multi-terabyte data stores of mostly source data in simple data models, but still handle simpler workloads, like standard reports and OLAP. Whether to store analytic data in the EDW or in a separate analytic database is one of the most critical design and architecture decisions adopters of next generation DW platforms must make. The following are some issues that need to be considered for NGDC data warehouse transformation:

- Analytics processed within the EDW. Many of the analytic tools based on data mining technologies require users to dump analytic data into flat files with a specific record structure, because that is the data structure. In recent years, data mining and predictive analytic tools have gotten better at processing data while it is stored in a DBMS. This is called "in-database analytics." It is anticipated that the trend toward in-database analytics will continue, because most users would rather manage data with an EDW or similar database and leave the data in place when analyzing it. SQL-based analytics demand that data be managed by a SQL-compliant DBMS.
- Analytic databases outside the EDW. This, too, is a well-established practice This takes many forms. At one extreme, data marts proliferate outside the EDW until IT and DW teams are forced to rein them in through time-consuming data mart consolidation projects. This may well be what survey respondents were thinking of when they said there will be a decline in analytic databases outside the EDW. Note that EMC's data warehouse Greenplum® solution excels in this area. Please refer to the section titled "Applianceization" for technical details on this solution.

- A best practice is to isolate disruptive analytic workloads on data warehouse appliances and other analytic databases outside the EDW. Again, please refer to the section titled "Applianceization" for additional information on this best practice.

Support for advanced analytics has been an area of great activity for software vendors in recent years. For example, many new software firms offering DBMSs built specifically for data warehousing and analytics have sprung up in this decade, including 1010data, Aster Data Systems, Greenplum (purchased by EMC), Illuminate, Infobright, Kognitio, ParAccel, and Vertica. Neoview is a new EDW and analytics platform from established vendor HP. Many established DBMSs have updated their support for query optimization and SQL standards (partially for better SQL-based analytics), including IBM, Oracle, Microsoft, Sybase, and Teradata. Likewise, some of the established DBMS vendors have improved their in-database analytic capabilities, as seen in the partnership between SAS and Teradata.

Other vendors have taken analytic processing to where data lives in storage. For example, the Oracle Database 11g R2—when used in combination with Oracle Exadata—pushes the scoring of data mining models down into the storage tier (Exadata). This produces data mining results faster because far less data is moved over the network to central CPUs. The Dataupia Satori Server and the Netezza Performance Server provide similar storage-level processing for queries and advanced analytics. Note that this approach requires specialized hardware for the storage tier, which Dataupia, Netezza, and Oracle Exadata all provide.

Best Practice – Implement Data Warehouse Appliances and Similar Platforms into the Next Generation Data Center

The most widely discussed and argued new DW option of the decade has to be the data warehouse appliance (DWA). It has undergone considerable evolution since first appearing early this decade. DWAs now include diverse product types that any definition of data warehouse appliance should encompass.

Best Practice – Consider Real World Expectations in the NGDC Data Warehouse

There seems to be considerable evidence that data warehouse appliances will be in your future in the Next-Generation Data Center. It is important to consider that next generation technology drivers are really business drivers. For example, real-time data warehousing (RTDW) is a high priority for the next generation's business, so the NGDC must keep pace. However, a best

practice is that a DW design should never implement hefty technology without proper consultative services. A RTDW should support time-sensitive business methods like operational Business Intelligence and just-in-time manufacturing or inventory management.

It is also important to consider avoiding assembling your own data warehouse platform, if at all possible. Today, most DW platforms are assembled by in-house personnel. This is time-consuming, can be a distraction and, in most cases, is not the major deliverable in your IT portfolio. Be open to pre-assembled and integrated DW appliances and similar DW bundles, or outsource the work to a system integrator or consultants.

It is important to plan for Big Data. The number of businesses in the “10-terabyte club” will double within three years. Many of the businesses moving into the club will probably need to replace their data warehouse platform to make the move possible and sustainable.

A best practice is to consider embracing low-cost DW platform options. If budget constraints are blocking your BI projects, look into open source software (especially DBMSs) and data warehouse appliances, plus creative approaches to licensing (SaaS) and low-investment deployments (clouds).Technologies outside the DW platform can be generational, as well, especially data management practices like master data management, data quality, and data integration.

Analytics will be a major factor in next generation DW platforms. Your peers in other firms report they will step up analytics, so perhaps you should, too. Many are moving data for advanced analytics off the EDW onto next generation platforms such as clouds, appliances, analytic databases, and columnar databases. This might also make sense for you.

Note that some next generation options are a critical path to others. For example, you may need to upgrade to 64-bit hardware and software before you can achieve goals with in-memory databases, faster query performance, and data scalability. Likewise, you may need to implement Web services and SOA before you get the broad access and real-time functionality needed to tightly integrate the EDW with operational systems.

Typically, advanced analytics and real-time business methods often lead many businesses to deploy separate platforms for these to off-load and complement an EDW. Be open to alternative

DBMSs. Thanks to the next generation, you now have more alternatives to consider, including open source, analytic, appliance, and columnar DBMSs. A best practice is to carefully decide which of these are appropriate to secondary platforms versus the primary EDW.

Best Practice – Implement "Parallel Everywhere" into a NGDC Data Warehouse Architecture

It is important to consider scaling and other MPP performance characteristics into the Next-Generation Data Center. As defined by Amdahl's Law and Gustafson's Law describing the estimated speedups as measured by parallel program potential, implementing parallel data flows and minimizing serial bottlenecks is imperative.

As an example, EMC's Greenplum database architecture is a state-of-the-art parallel processing engine (shown in Figure 43).

When users run SQL queries and MapReduce programs, the processes are executed in this engine which coordinates processing and data movement between 10s or 100s of servers to achieve the best performance possible. This eliminates every sequential bottleneck to make sure that everything is running in parallel as efficiently as possible. A similar approach was done as it relates to loading. EMC's Greenplum database creates full parallelism from multiple sources (files, database, applications, ETL/DI systems, and so on) to simultaneously stream them to all nodes of a Greenplum database cluster in parallel. This technology is called "'MPP" or Massively Parallel Processing.

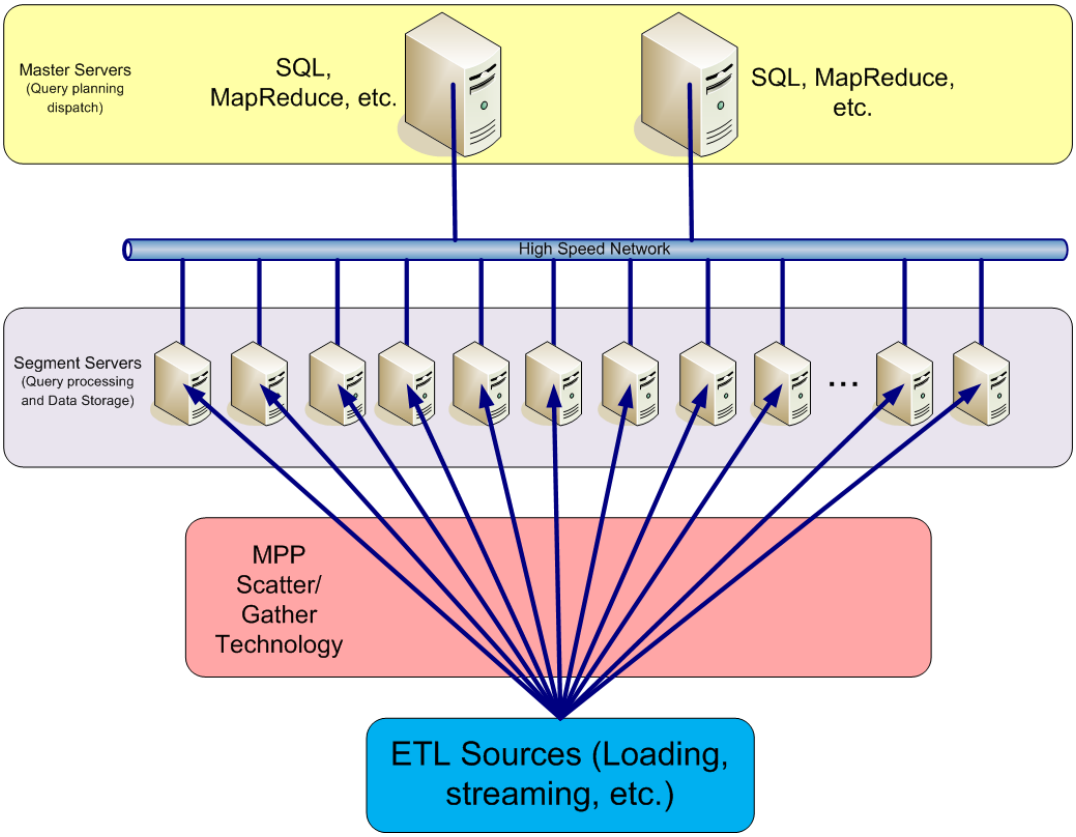


Figure 43 Greenplum Data Computing Appliance Architecture

This technology allows sources to be split into pieces that one can 'scatter' across 100s or 1000s of streams that flow directly to all database nodes. This is an example of one of Gustafson's Law's caveats that breaking the associations of serial processes allows for scalability without limitations. Please refer to the section titled "Massively Parallel Processing (MPP)", starting on page 37 for additional information on how this relates to Gustafson's Law MPP fundamental theories.

As the data arrives, the appliance applies any in-flight transformation and cleansing before writing it to disk (in parallel, automatically partitioned across nodes, with optional on-disk compression). There are no sequential bottlenecks, and performance increases linearly with the number of nodes in the cluster (with no theoretical maximum). This architecture powers an 'external table' interface for loading and streaming external data sources, as well as Greenplum's 'gpload' command line loading utility.

Best Practice – Consider Data Partitioning as part of the NGDC Data Warehouse Designs

The range of architectural choices from MPP to SMP offers a complex decision space for businesses deploying data warehouses. Please refer to the section titled “Massively Parallel Processing (MPP) and other architectures”, starting on page 37, for a more detailed discussion on this topic. Given that companies tend to make architectural choices early, and then invest up to hundreds of millions of dollars as they grow, the result of choosing an architecture that presents increasingly intractable partitioning problems and the likelihood of idle nodes can have consequences that measure in the millions of dollars. A system that scales well exploits all of its processors and makes best use of the investment in computing infrastructure. Regardless of environment, the single largest factor influencing the scalability of a decision support system is how the data is partitioned across the disk subsystem. Systems that do not scale well may have entire batch queries waiting for a single node to complete an operation. On the other hand, for throughput environments with multiple concurrent jobs running, these underutilized nodes may be exploited to accomplish other work. There are typically three approaches to partitioning database records:

Range Partitioning

Range partitioning places specific ranges of table entries on different disks. For example, records having “name” as a key may have names beginning with A-B in one partition, C-D in the next, and so on. Likewise, a DSS managing monthly operations might partition each month onto a different set of disks. In cases where only a portion of the data is used in a query, (the C-D range, for example), the database can avoid examining the other sets of data in what is known as partition elimination. This can dramatically reduce the time to complete a query. The difficulty with range partitioning is that the quantity of data may vary significantly from one partition to another, and the frequency of data access may vary, as well. For example, as the data accumulates, it may turn out that a larger number of customer names fall into the M-N range than the A-B range. Likewise, mail-order catalogs find their December sales far outweigh the sales in any other month.

Round-Robin Partitioning

Round-robin partitioning evenly distributes records across all disks that compose a logical space for the table, without regard to the data values being stored. This permits even workload distribution for subsequent table scans. Disk striping accomplishes the same result—spreading read operations across multiple spindles—but with the logical volume manager, not the DBMS,

managing the striping. One difficulty with round-robin partitioning is that, if appropriate for the query, performance cannot be enhanced with partition elimination.

Hash Partitioning

Hash partitioning is a third method of distributing DBMS data evenly across the set of disk spindles. A hash function is applied to one or more database keys, and the records are distributed across the disk subsystem, accordingly. Again, a drawback of hash partitioning is that partition elimination may not be possible for those queries whose performance could be improved with this technique. For symmetric multiprocessors, the main reason for data partitioning is to avoid “hot spots” on the disks, where records on one spindle may be frequently accessed, causing the disk to become a bottleneck. These problems can usually be avoided by combinations of database partitioning and the use of disk arrays with striping. Because all processors have equal access to memory and disks, the layout of data does not significantly affect processor utilization. For massively parallel processors, improper data partitioning can degrade performance by an order of magnitude or more. Because all processors do not share memory and disk resources equally, choosing which node on which to place data, has a significant impact on query performance.

The choice of partition key is a critical, fixed decision that has extreme consequences over the life of a MPP-based data warehouse. Each database object can be partitioned once, and only once, without re-distributing the data for that object. This decision determines long into the future whether MPP processors are evenly utilized, or whether many nodes sit idle, while only a few are able to efficiently process database records. Unfortunately, because the very purpose of data warehouses is to answer ad hoc queries that have never been foreseen, a correct choice of partition key is one that is, by its very definition, impossible to make. This is a key reason why database administrators who wish to minimize risks tend to recommend SMP architectures where the choice of partition strategy and keys have significantly less impact.

The choice between symmetric multiprocessors and massively parallel processors for decision support systems is one of the most critical decisions faced by today’s Information Technology businesses. Because most enterprises start small and enlarge their data warehouses as their data and their processing needs grow, a large financial investment will ultimately be made in computing infrastructures. The most important factors determining whether the benefit of this financial outlay will be realized in decision support system performance is whether the database

servers can provide reliable performance without the use of highly-specialized short-cuts, and whether the database can scale with rapidly-increasing use and data storage.

Massively parallel processors are highly sensitive to whether database administrators are able to partition the database uniformly across all MPP nodes. This is a risky proposition where the ultimate result is that some queries will perform very well, and some will perform poorly. Both the distribution of data and the communication costs for inter-node transfers are difficult to optimize, and have a tremendous influence on the cost effectiveness of an enormous investment. Data warehouses based on MPP architectures work best when queries are highly predictable, have little skew, and where the update activity is minimal. MPP performance depends on highly skilled database administrators, who dynamically monitor queries, add and subtract indices as needed, and who are patient enough to monitor systems having a large number of nodes.

Symmetric multiprocessors give each processor equal access to memory and I/O resources, giving consistently superior performance over MPP architectures. Given that no database administrator can foresee all DSS queries that a server is to process, symmetric multiprocessing represents a choice that can scale and grow with the demands of today's data warehouses, while minimizing risks to the IT businesses supporting them. Because symmetric multiprocessors share data, I/O, and processing resources, SMP performance is most efficient, and provides the most reliable performance for true ad hoc queries.

Clustered architectures provide a growth path when a greater number of processors are needed than is available on SMP systems, and when the business needs require the high availability that comes through the use of multiple independent servers. For workloads that require fewer than 64 processors, a single SMP system will always provide superior performance. When more processors are needed, use a cluster based on a small number of SMP systems. It is always easier to manage a small number of powerful nodes in a cluster than to manage a large number of nodes in an MPP configuration.

Data Warehouse Appliance Vendor solution Summary

This section will outline the current and future Data Warehouse Appliance (DWA) vendor offerings. Netezza was the first vendor to offer a data warehouse appliance (introduced around 2002), so early DWA definitions were based upon Netezza products, which provide a whole-

technology stack for data warehousing. That is, the Netezza Performance Server combines database and operating system software with server and storage hardware in a complete data warehouse platform. Before Netezza, Teradata, Sequent, and WhiteCross (now Kognitio) had for years offered similar single-vendor combinations of hardware and software purpose-built for data warehousing, though not necessarily in an appliance package or described as an appliance.

DATAllegro launched in mid-2005 with a whole-technology stack solution involving proprietary hardware. DATAllegro soon left its proprietary hardware in favor of commodity hardware from other vendors, before being acquired by Microsoft in 2008. Teradata announced in 2008 a new family of data warehouse packages that includes DWAs, some running on commodity hardware. In 2003, Kognitio moved from proprietary to commodity hardware, after offering a data warehouse platform on proprietary hardware since 1989. The movement from proprietary to commodity hardware has good reasons behind it. Commodity hardware is relatively inexpensive and, subsequently, helps keep down the price of DWAs. This is important, since DWAs compete largely on their low cost.

There have been various Software Appliance Solutions. Starting in 2006, a new wave of vendors emerged with database management systems (DBMSs) purpose-built for data warehousing. These include DBMSs based on the relational model (Greenplum, Aster Data Systems, and Kognitio) and the columnar model (Infobright, ParAccel, and Vertica). Most of these DBMSs are sold and licensed stand-alone (like any DBMS software), or embedded in an appliance (usually with a certified or recommended hardware configuration). In the context of the embedded license model, most of the new DBMS vendors call their product a “software appliance”.

This somewhat oxymoronic term refers to a software component (namely a DBMS), that may be embedded in a full data warehouse appliance. Hence, each of these vendors offers a partial-stack appliance, called a software appliance. The software appliance has proved to be a good starting point for the new DBMS vendors. It allows them to focus on database software (not designing and building hardware), which is their point of greatest innovation and, therefore, their value proposition. The software appliance product enables the new, small DBMS vendors to collaborate with commodity hardware vendors and to benefit from these larger firms’ resources.

There have been multiple bundled DW platforms from multiple vendors recently, As DWAs entered the marketplace, relational database vendors (IBM, Microsoft, Oracle, Sybase, and HP) stepped up their offerings of hardware and software bundles that assemble a whole-technology stack for data warehousing. Most of these bundles are not DWAs per se, yet they offer many of the benefits of a DWA. In particular, a preconfigured technology stack reduces system integration work, reduces time to use, and comes from a single vendor that supports the whole stack. Furthermore, vendor size matters in that some user businesses avoid start-up vendors. For these users, the hardware/software bundles are significant, because they come from large, stable vendors and include familiar, mature DBMSs.

Examples of bundles from leading database vendors include HP Neoview, IBM Smart Analytic System, and IBM InfoSphere Balanced Warehouse. Most of the hardware and software components in the IBM bundles come from IBM. In the case of HP Neoview, all the hardware and software components come from HP. Launched in late 2008, the HP Oracle Database Machine and the HP Oracle Exadata Storage Server are both based on Intel processors, hardware from HP, and software from Oracle.[4] Announced in May 2008, the Sybase's Analytic Appliance combines pSeries hardware from IBM with Sybase IQ (a columnar database, purpose-built for data warehousing). In February 2009, Microsoft launched SQL Server Fast Track Data Warehouse, which accelerates DW deployments through reference configurations. Fast Track Data Warehouse is available on inexpensive hardware from EMC, Bull, Dell, and HP.

However, the Data Warehouse Appliance is being redefined. Due to the trend among vendor offerings moving from whole-technology stacks to partial ones, a newly revised definition must encompass DWAs that complies with the original definition (from Netezza and DATAllegro), as well as the newer software appliances (from Aster Data Systems, Infobright, Greenplum, Kognitio, ParAccel, Vertica, and so on). Furthermore, hardware/software bundles assembled for data warehousing (from HP, IBM, Microsoft, Oracle, Sybase, Teradata, and so on) share many characteristics and benefits with DWAs, so these should be mentioned whenever DWAs come up. Prospects of growth and adoption vary across the three definitions of appliances:

The user community and industry verticals continue to redefine how it uses data warehouse appliances. As discovery-driven advanced analytics, based on SQL become more common and more mission critical, users are in dire need of a data warehouse platform that can respond

quickly (with little or no tuning) to ad hoc and/or complex queries against multi-terabyte data sets of less-than-ideal structure and quality. DWAs fulfill this need, as do the analytic DBMSs discussed elsewhere in this report. For this reason, the vast majority of DWAs and software appliances manage multi-terabyte data marts and other analytic databases. EDWs are rare on DWAs and software appliances today (though common on DW bundles), but this should change over time, as user confidence grows and appliance capabilities increase.

Archival Business Drivers in the NGDC

As it relates to the Next-Generation Data Center, archiving is and always has been difficult to address. In many ways, the way one archives and the technology variations of how one archives has many challenges. Archiving has many facets and many interdependencies with other common IT processes, such as migration. In previous sections, archive technology has been discussed in terms of what tools and methodologies are available. This section will outline the business and use cases for archiving.

Efforts to archive large amounts of digital data are being developed by many cultural, heritage, and business institutions. Numerous initiatives are being developed aiming to extract the Web's data[45], together with institutional repositories[46]. However, getting the material inside the archive is just the beginning for any initiative concerned with the long-term preservation of digital materials.

Digital preservation can best be described as the activity or set of activities that enable digital information to be intelligible for long periods of time. In general, digital information kept in an archival environment is expected to be readable and interpretable for periods of time much longer than the expected lifetime of the individual hardware and software components that comprise the repository system, as well as the formats in which the items of information are encoded. A good example is the health care sector, where CT scans and X-rays need to be preserved for more than twenty years. As a result, digital preservation has many use cases. For more on digital preservation and how to keep the data reliable and achieve longevity, please refer to the section titled "Reliability and Resiliency" starting on page 132, and the section titled "Longevity", starting on page 130.

Over the past decade, a vast number of preservation strategies have emerged from the various preservation projects developed, literally, all over the world. Not surprisingly, the most used continues to be migration, especially as it relates to fixed content digital objects, such as images, databases, or text documents, which typically are the focus of preservation artifacts.

---

[45] http://www.archive.org/

[46] Growth of Institutional Archives over Time
Http://archives.eprints.org/index.php?action=analysis

The major drawback for using a migration strategy is that whenever an object is converted to a new format, some of its original properties may not be adequately transferred to the target format. This may occur due to incompatibilities between the source and target formats, or because the application used to perform the conversion is not capable of carrying out its tasks correctly.

Best Practice – Understanding and Selecting a Best of Breed Migration Strategy

Best practice states that two decisions have to be made prior to any object migration:

1. which format should be used to accommodate the properties of the original object
2. which application should be used to carry out that migration

This decision-making activity constitutes the first stage of any migration process. The goal is to focus on the optimal combination of target format and conversion software, one that preserves the maximum number of properties of the original object at the minimum cost.

Cost should be regarded as a multi-dimensional variable. Factors such as throughput, application charges, format openness, or prevalence should be considered collectively during this decision-making activity. Objective tools or frameworks, especially designed to help institutions in the selection of appropriate options, would greatly simplify this exceptionally complicated task.

The conversion work consists of the reorganization of the information elements that comprise the digital object into the logical structures as defined by a different format. From the preserver's point-of-view, carrying out a conversion usually consists of setting up a conversion application and executing it against a collection of digital objects. Some scripts may have to be developed in order to automate the whole procedure.

After the conversion process, the resulting objects should be evaluated to determine the amount of data loss incurred during migration. This is accomplished by comparing the properties that comprise the source object with the properties of its converted counterparts. If the evaluation results are below expectations, i.e. the object's properties have degraded to an unacceptable level, a best practice is a different migration alternative should be selected, and the whole process reinitiated. In addition, note that if the source objects utilize containerization, this is less of an issue. Please refer to the section titled “Longevity”, starting on page 130 for additional details on containerization.

In most cases, the evaluation process still requires a considerable amount of manual labor. Certain subjective properties such as the disposition of graphic elements in a text document or the presence of compression artifacts in an image file are generally inspected manually and typically require human intervention. As a result, this type of rendering activity is onerous, time-consuming, and potentially error-prone.

Archive Use Cases

This following contains descriptions of the entities that are involved in the preservation and archive aspects being discussed. The framework defined in the section titled "Archival Ecosystem", starting on page 94, outlines what are the best practice technologies that will do the job efficiently and economically with the Cloud and Next-Generation Data Center Transformation process in mind. Following are descriptions of typical use cases and the requirements derived from each use case. The use cases are divided into generic use cases and workload-based use cases. The former are not specific to a type of data or application, while the latter are specialized for concrete workloads.

The physical user entities typically are, Archive Employee, Consumer, Preservation Manager, Producer, System Administrator and Auditors, as shown in Figure 44. The non-physical user entities or attributes in a preservation and archive system that relate to long term archiving are:

- **Storage** – Storage subsystem that maintains numerous preservation objects. Examples of storage subsystem solutions include EMC Centera® Archive Platform. Please refer to the section titled Archival Ecosystem, starting on page 94, for additional details.
- **TPA-Service** – Today's preservation and archiving service, e.g. OAIS ingest service, transformation service.
- **NG-Service** – Next Generation preservation service, which may be unknown today.
- **T-App** – Today's application that generates digital data, e.g. a word processor, eMail application.
- **NG-App** – Future application (Next Generation) which may be unknown today.
- **Reg** – Registry that stores representation information of the used storage formats, e.g. the specification documents of the used formats.

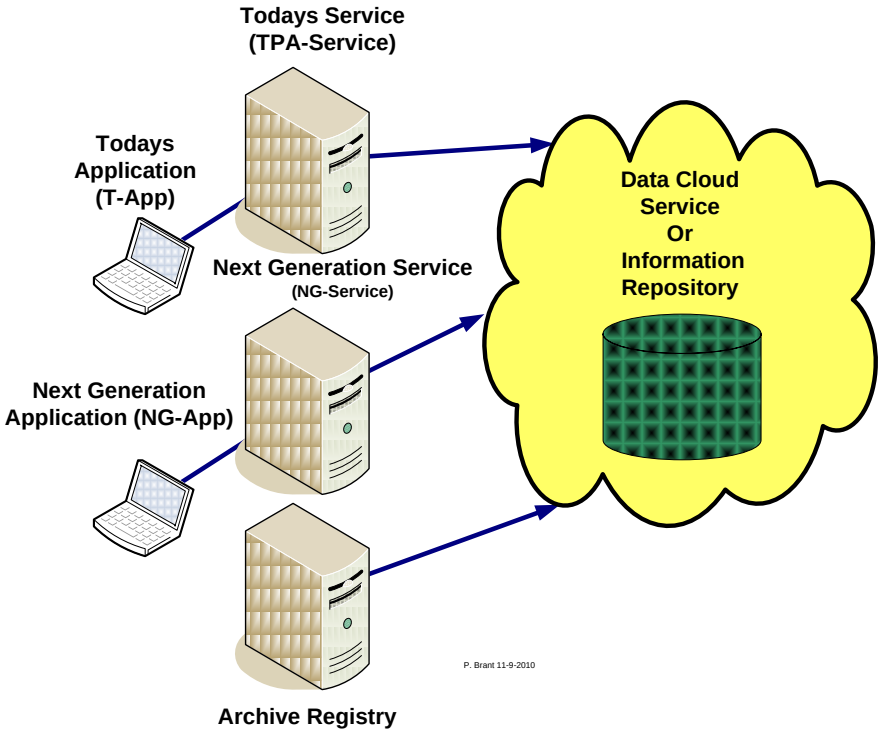


Figure 44 Preservation System Entities

The following describes the generic use cases that appear with any application or type of data, because of changes in technology (and thus the environment) over time. The first use case is where there is no change in the environment, and subsequent cases add more changes in the system, due to the passage of time and the application remains the same as shown in Figure 45. For each use case, a flow is given and a set of requirements derived. The use-case flow is:

I. T-App ingests a Preservation Object at 10:00 using a standard interface. The operation is agnostic to media, platform, and vendor.

II. An hour passed, and there is no change in environment.

III. T-App accesses the Preservation Object at 11:00, using a standard interface.

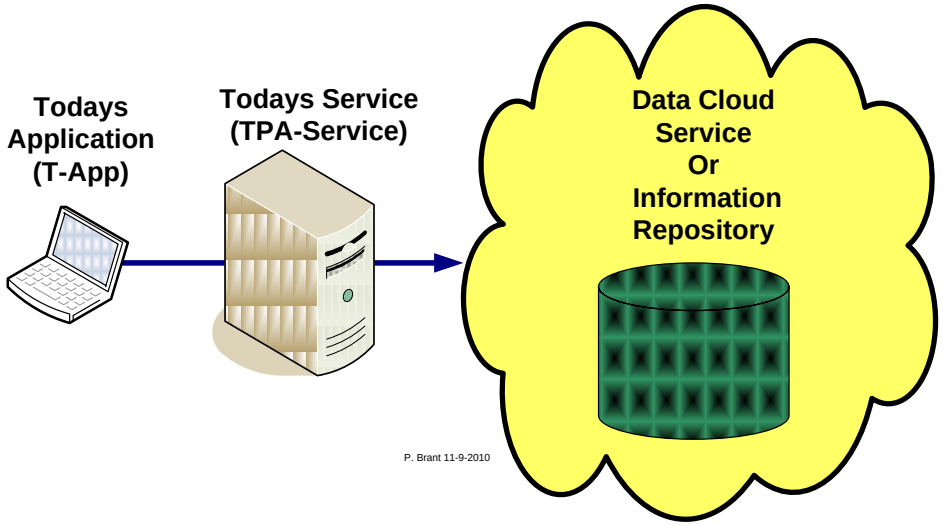


Figure 45 Ingest and Access with Same Application

Best Practice – Support for Standard Interfaces, e.g. NFS, CIFS, XAM, which are agnostic to media, platform, or vendor.

The main requirements derived from this use case are support for standard interfaces, e.g. NFS, CIFS, and XAM that are agnostic to media, platform, or vendor as shown in Figure 46. The following shows an example of ingest and access with different applications. The use-case flow is:

I. NG-App ingests a Preservation Object today, e.g. an object with meteorological data from a specific satellite.

II. Time passes and a newer application called NG-App is developed for the same type of data, e.g. for meteorological data from satellite. Note that although it is the same type of data, it may now be in a different format.

III. NG-App accesses the Preservation Object in the future using one of TPA-Service's supported interfaces.

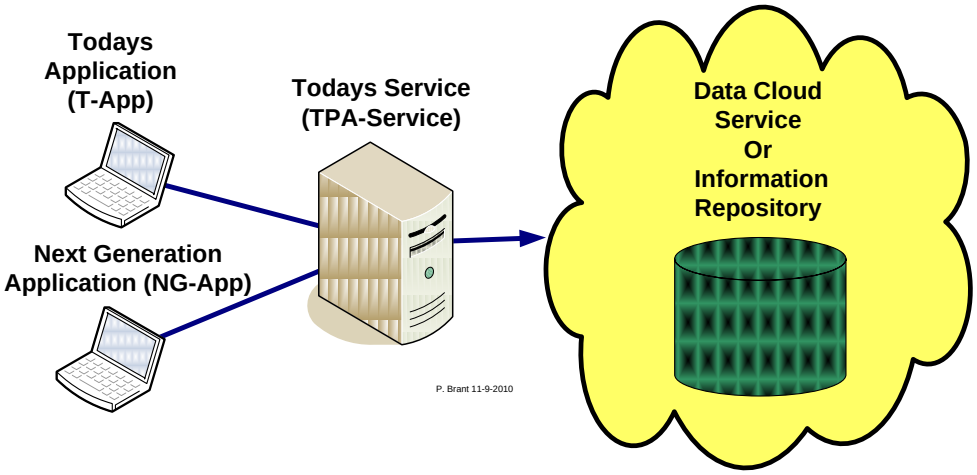


Figure 46 Ingest and Access with Different Applications

The main requirements derived from this use case are support for multiple versions of preservation objects and support for multiple data models and multiple formats for the raw data. The following is an example of Ingest and Access with Different Preservation Services as shown in Figure 47. The use-case flow is:

I. NG-App ingests a Preservation Object today via TPA-Service.

II. Time passes and the preservation services changed. New preservation services called NG-Service were developed.

III. NG-App accesses the Preservation Object in the future via NG-Service.

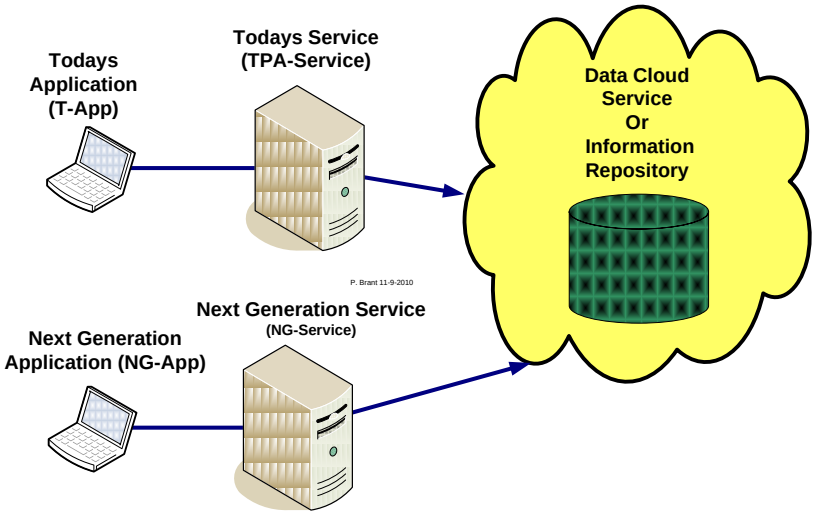


Figure 47 Ingest and Access with Different Preservation Services

The main requirements derived from this use case are persistent globally unique identifiers for the preservation objects so the object identifiers and references continue to work and self-contained data is maintained so nothing is lost when moving from TPA-Service to NG-Service.

The following is the example of a storage format change as shown in Figure 48. The difference here is an Archive Registry Data Base server is required to store the archive metadata. The use-case flow is:

I. NG-App ingests a Preservation Object today via TPA-Service.

II. Time passes and the storage subsystem migrates to a new one with a new container format standard that replaced SIRF.

III. NG-App accesses the Preservation Object in the future via NG-Service.

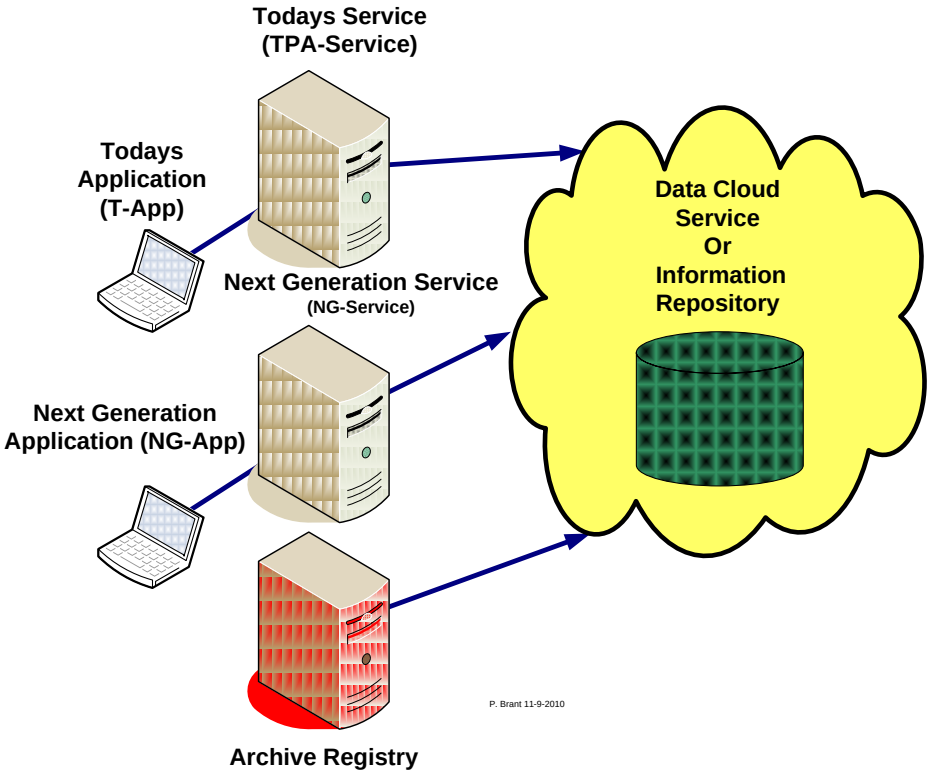


Figure 48 Storage Format Change

The main requirements derived from this use case are:

I. Self-describing via a simple formalized meta-language that itself should be changeable to support SIRF format migration.

II. SIRF Representation Information should be preserved in an external registry. A best practice is the registry should be recursively able to be preserved. The recursion ends when the Representation Information is described in a simple format that can be preserved by the community, e.g. a simple text file.

The following will be of the form of specific workload-based use cases that include data, which needs to be accessed and used in the future in spite of technology changes. For each use case, a flow is given and a set of requirements derived.

A typical use case for archiving is Discovery or “eDiscovery”, The event of “eDiscovery” is the formal legal process of finding information relevant to a legal matter and delivering it to opposing council. More loosely defined, discovery can include formal legal requests as well as internal investigations that may never reach a court. eDiscovery is discovery as applied to electronically stored information. Preservation objects, like any other electronic information, can be subject to eDiscovery.

The following eDiscovery Terminology is defined:

- Case – a legal matter, i.e. lawsuit or investigation
- Responsive – information that is related to a specific case is “responsive” to it
- Legal hold – a means for ensuring that responsive information is not deleted or modified while a case is pending. A specific preservation object may be subject to any number of holds and must be maintained until all of them have been released.
- Identification – determining what data is potentially relevant to a legal inquiry
- Collection – the process of gathering all identified information
- Preservation – ensuring that potentially relevant information cannot be destroyed or altered
- Processing, Review, and Analysis – the process of sifting through collected information either electronically or manually to identify which objects are responsive and which are not
- Retention Policy – A policy governing when and for how long an object must be retained by a storage system
- Disposition Policy – A policy that defines what actions to perform at the end of an object’s lifecycle.

The Discovery or “eDiscovery” use case flow is as follows:

I. NG-App ingests a Preservation Object today via TPA-Service.
II. Time passes and the data becomes subject to eDiscovery.
III. Potentially responsive preservation objects are identified using provenance, context, and content information stored with preservation objects.
IV. Identified objects are put on “legal hold,” preventing deletion or modification.

V. Identified objects are copied from the preservation system and collected to a case repository for processing, review, and analysis.

VI. At some future date, the “legal hold” is removed. The object may become subject to other legal holds or retention/disposition policies at any time.

The main requirements derived from this use case are:

- Support for retention holds on preservation objects that prevent their deletion or modification
- Support for verification of document provenance and authenticity, regardless of migrations whether logical or physical
- Support methodology for verification of completeness and correctness
- Support for storing audits. The audits can include records about modification, possibly records about access, and so on.
- Must be possible to identify, collect, and preserve Preservation Objects that are relevant to a legal matter.

Email data may include interrelated objects and numerous repetitions. An email thread includes one or more messages where each message is an email by itself and can contain zero or more attachments. The following flow is one method of preserving emails used to derive SIRF requirements, but other methods may exist. The use case flow is:

1. NG-App ingests an email thread today via TPA-Service. This includes ingesting a collection of several interrelated Preservation Objects (POs) as follows:
2. Ingest a new PO for the thread. The PO metadata should include all mail header information, auditable date information, keywords, and so forth, including allowance for business-unique metadata.
3. For each message within the thread, check if a PO already exists for that message. If it does, create a link from the thread PO to the message existing PO. If not, ingest a new PO for the message and a link from the thread PO to the newly created message PO.
4. For each file attachment within the message, ingest a PO for that attachment and a link from the message PO to the attachment PO.
5. Ingest one or more POs for information upon which the thread depends, such as a PO for the address book, POs for business processes, POs for data leakage policies, and so on.

6. Time passes and the businesses changes scope, name, undergoes a merger, etc. As a result, NG-Service creates a set of new version POs. These include a new version PO for the address book, new version POs for the new business processes, new version POs for data leakage policies. Note that the thread, message, and attachment POs created in Step 1 are not affected.
7. More time passes and NG-App searches the metadata of threads, messages, and attachments in parallel to find relevant POs. NG-App creates POs for the search results to raise performance of future searches and ingests them to the preservation system via NG-Service. Those new POs may contain links to the thread, messages, and attachments created in Step 1.

The main requirements derived from this use case are:

- Support for links between objects that are as immutable as the objects. The links can be either "hard links" that require the existence of the linked objects within SIRF or "soft links" that can reference objects external to SIRF.
- Support for auditable time stamps that are immutable and created by known authority.
- Support for "special" POs such as a PO that includes address book, or a PO that includes search results.
- Generic support for business unique metadata (perhaps extended TLV such as OrgID/Type/PrimitiveType/Length/Value).

The following is an example of a consumer archive into the public or private cloud. An individual wants to preserve his family photos and documents on a cloud that provides preservation services, so that future generations will be able to access that data and study their roots. The use case flow is:

- A user creates a genealogy container for his genealogy-related documents on a cloud that provides SLAs for preservation.
- The user uses NG-App to ingest a genealogy-related document via TPA-Service on the cloud.
- TPA-Service on the cloud ingests the PO with the original document as well as transforms the document to a standardized format believed to be more sustainable such as pdf/a and ingests the resulting PO version to the same genealogy container.
- Time passes and the grandchildren would like to get that document.
- NG-Service will validate the grandchildren’s identity and will provide appropriate credentials to access the genealogy container and the document.

- NG-App, which is a future application executing on technology used at that future time, access via NG-Service the latest version of the document and renders the pdf/a document.

The main requirements derived from this use case are:

- Support for transformations of preservation objects, e.g. support various versions of the PO and the tree structure they create.
- Support for managing identifiers over time.
- Support secured access to the data that is updateable over time e.g. when a security mechanism becomes weak.
- Support cloud containers to be SIRF-compliant, so containers can be migrated to other clouds with all the required preservation information.
- Verification of document provenance and authenticity, regardless of migrations whether logical or physical.

The following is an example of a biomedical bank. A large hospital, which has an adjacent academic medical research center, stores the patients' biomedical data in a biomedical bank in which data is preserved for decades. The data is used at the point of care as well as for biomedical research by the adjacent research center. The use case flow is:

1. NG-App ingests via TPA-Service a PO that includes a standardized Digital Imaging and Communications in Medicine (DICOM) image of the leg of a patient who is a minor.
2. Time passes and the patient, who is now an adult, schedules an appointment to check a new problem he has encountered in his leg.
3. NG-Service will identify the data needed for the scheduled appointment using reference, context, and provenance information.
4. The identified Preservation Objects will be extracted from an offline archive media to an online media to be timely accessible for the appointment.
5. NG-App at point of care accesses the identified POs for the patient via NG-Service.
6. More time passes and a researcher from the adjacent academic medical research center wants to access that image for research purposes. According to HIPAA regulations, the researcher can get just an unidentified image.
7. NG-App accesses the unidentified PO via NG-Service.

The main requirements derived from this use case are:

- Support hierarchical storage management, e.g. support unique IDs for the POs regardless of the storage tier, support online and offline storage.
- Support masking of sensitive data, e.g. support storing POs for un-identification modules within the SIRF container.
- Verification of document's provenance and authenticity, regardless of logical or physical migrations.

The following is another cloud-related use case as in the situation of merging cloud repositories. The use case flow is:

1. NG-App ingests via TPA-Service a PO in a cloud that is provided by company “FirstCloud”.
2. NG-App also ingests via TPA-Service a second PO in a second cloud provided by company “SecondCloud”.
3. Time passes and the two companies, “FirstCloud” and “SecondCloud”, merge and their two cloud repositories are combined. This is possible as the POs identifiers are globally unique.
4. NG-App access via NG-Service the two POs in the combined cloud provided by the merged company.

The main requirements derived from this use case are:

- Support cloud containers to be SIRF-compliant, so containers can be interpreted by other clouds.
- Persistent **globally** unique identifiers for the preservation objects.

Best Practice – Implement a Service-Oriented Architecture for Automatic Migration

In order to implement a valid migration, the three outlined activities (i.e. selection of migration options, conversion, and evaluation) should be performed in an automated fashion. A best practice is to implement a Service-Oriented Architecture (SOA) by combining input from different distributed applications, and enable client institutions to preserve collections of digital material automatically.

Many Institutions already possess a digital repository system capable of storing, managing, and providing access to the digital objects they hold. The repository system acts as the client application that benefit from the services provided by the SOA.

It is expected that advantages of this SOA architecture will be that a repository will experience objects in formats previously not encountered and which will need at the very least to undergo a process of normalization before being deposited or archived. In the presence of an unrecognized format, a best practice is that the archive repository system will invoke a format identification service provided by the SOA in order to obtain information about the object's format, in addition to checking its integrity. After this operation, the repository will interrogate the SOA to obtain a list of formats to which the object “could be converted”. The repository will inform the SOA of its preservation preferences and requirements, i.e. a list of preservation-oriented requirements derived from the policies created by the senior management of the archive. A list of requirements as per best practices is:

1. Preservation interventions should be affordable and quick.
2. Interventions should preserve the maximum number of significant properties of the original object.
3. The interventions should not route to specific formats that are dependent on the payment of royalties.

The SOA would then address all of these criteria with information previously acquired about the behavior and quality of all accessible conversion applications and would then produce a ranked list of optimal migration options. The repository system could then select the most suitable one from this list and request the SOA to carry out the corresponding migration.

After the conversion process, the repository system would receive a new digital object (better yet, a new digital representation of the source digital object) and a migration report stating the amount of data lost in that migration. This report could then be merged with the preservation metadata already maintained by the repository in order to document the preservation intervention and sustain the object's authenticity. On a regular basis, the repository would consult with a notification service to determine if any of the formats it holds are at risk of becoming obsolete. When a format falls into that condition, a new migration process is triggered.

As per best practices, the outlined scenario enables one to identify the following services that would be required:

1. A format identification service that also checks the integrity of digital objects.
2. A service that produces recommendations of optimal migration options (selection of a migration option).
3. A service to carry out format migrations (the conversion).

4. A service to determine the amount of data loss resulting from a migration (evaluation of results).
5. A service that provides information about the formats at risk of becoming obsolete.

The general architecture of the SOA Archive best practice is shown in Figure 49.

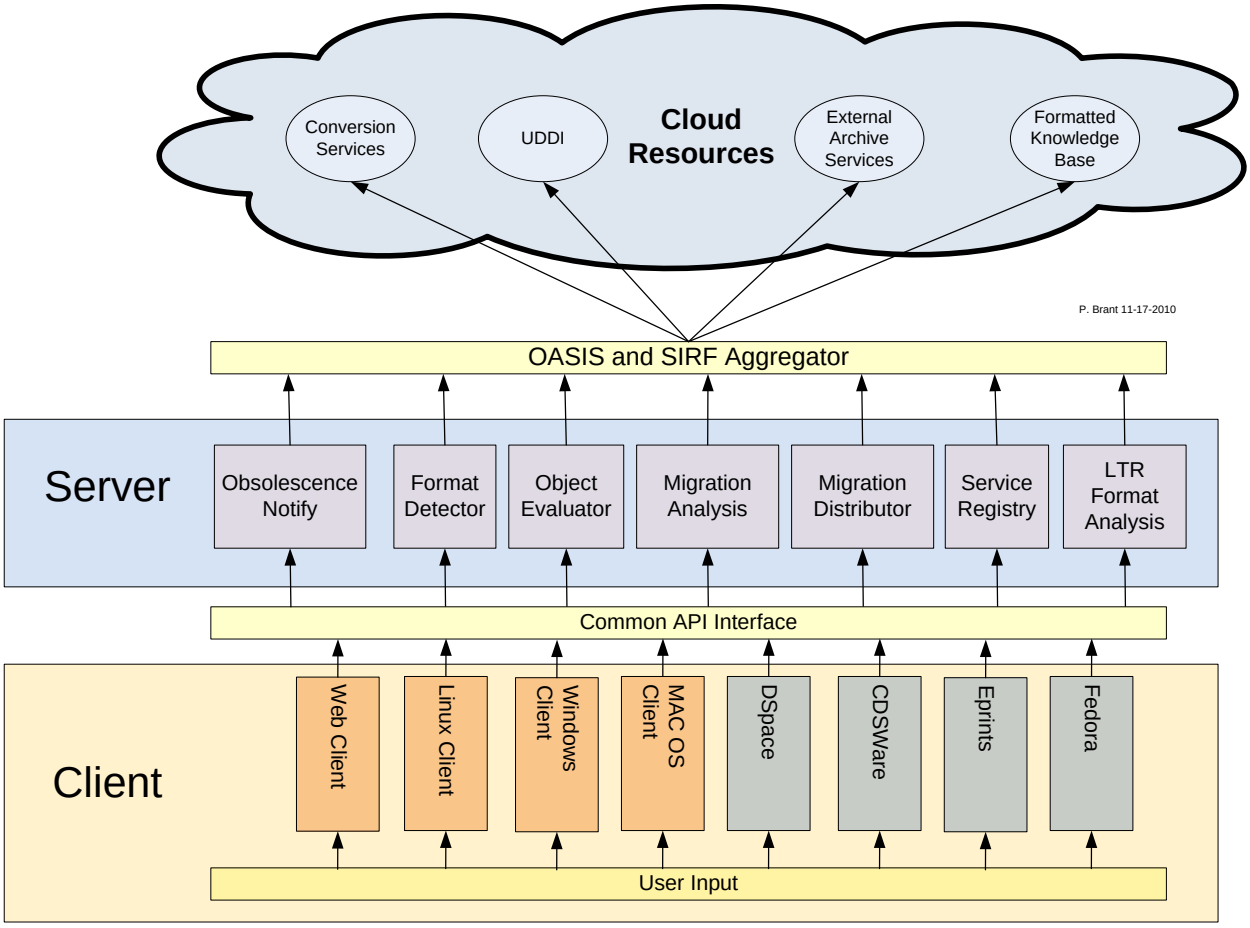


Figure 49 Architecture for SOA Capable Archiving of Digital Data Preservation

The clients extract user input that is typically archive data of various types, and it shows typical applications that may use the services provided by the SOA Archive solution. Among these are: digital repository systems such as DSpace, Fedora, or Eprints, and custom applications developed by individual users.

It is important to point out that any application capable of invoking a Web service may make use of the proposed SOA. On the server-side are depicted the chief components comprising this framework. Each of these components is actually an independent application with distinctive roles and responsibilities that cooperate with each other by exchanging messages. This

approach makes it possible for each component to be governed by a different organization and facilitates the distribution of workload.

The first of these components is the Obsolescence Notifier, a service responsible for raising awareness among clients of the file formats that are at risk of becoming obsolete. This function should utilize the “INFORM” methodology.

Developed at OCLC (Online Computer Library Center)[47], INFORM (Investigation of Formats based on Risk Management) is a process for investigating and measuring the risk factors of digital formats and providing guidelines for preservation action plans. INFORM attempts to discover specific threats to preservation and measure their possible impact on preservation decisions. By repeating the process, changes in the threat model over time can be detected, to which one can act accordingly.

A comprehensive approach to digital format assessment must include the following considerations: (1) risk assessment; (2) significant properties of the format under consideration; (3) the features of the format as defined in the format specification. The method of incorporating the latter two aspects in a preservation decision will be detailed at a later time. Meanwhile, there are six classes of risk that must be assessed:

1. Digital object format – risks introduced by the specification itself, but also including compression algorithms, proprietary (closed) vs. open formats, DRM (copy protection), encryption, digital signatures.
2. Software – risks introduced by necessary software components such as operating systems, applications, library dependencies, archive implementations, migration programs, implementations of compression algorithms, encryption, and digital signatures.
3. Hardware – risks introduced by necessary hardware components including type of media (CD, DVD, magnetic disk, tape, WORM), CPU, I/O cards, peripherals.
4. Associated businesses – risks related to the businesses supporting in some fashion the classes identified above, including the archive, beneficiary community, content owners, vendors, and open source community.

[47] http://www.oclc.org/us/en/default.htm

5. Digital archive – risks introduced by the digital archive itself (i.e. architecture, processes, business structures).
6. Migration and derivative-based preservation plans – risks introduced by the migration process itself, not covered in any other category.

It is very important to analyze the digital format specification in the proper context. Each specification has dependencies on software and hardware, some more than others. The INFORM method described here takes the guesswork out of the decision-making process and records the logic behind why, for example, TIFF would be a preferred choice. The documentation resulting from this methodology becomes part of the record of the preservation actions undertaken over time.

Sometimes, specific software or hardware is required to render an object in a given digital format. Newer technologies—such as DRM (Digital Rights Management), encryption, and digital signatures rely on proprietary, secret software (often protected by law) and hardware components. Other times, the specification may have lost its currency among commercial and open source developers, with only one or two products still supporting the aging format. In this case, the fact that its software and community dependencies are unique in their respective classes poses significant risk to the format's longevity.

As shown in Figure 49, the Format Detector is a service capable of identifying the underlying encoding of a digital object. The client institution should be able to monitor, migrate, and validate the integrity of digital objects without human intervention and this service. It enables digital formats to be identified according to the naming scheme used by other components that comprise the SOA architecture via the Migration Broker.

Any conventional application may also be used as a service if an appropriate application wrapper is developed, i.e. a small piece of software that acts as the intermediary between the application and the migration communication protocol. In this type of approach, a client application is used to send out a digital object to a remote procedure that, after decompressing (utilizing a data compression package such as PKZIP) the received message, converts the embedded object and returns the result to the client. Standard protocols, such as those that accompany Web services technology, may play an important role in this domain due to their open-standard and platform-independent characteristics.

The Service Registry is responsible for managing information about existing conversion services. It stores metadata about its producer/developer (e.g. name, description, and contact), about the service itself (e.g. name, description, the source/target formats that it is capable of handling, cost, and so on) and information on how the service should be invoked by a client application (i.e. its access point).

It is best practice that the Service Registry is populated with rich metadata. Much of the information delivered to end users after a conversion will be obtained from this data source. This information can be used to document the preservation intervention as it outlines all the components that took part in the migration process and describes the outcome of the event in terms of data loss and object degradation. For more information on this topic, please refer to the section titled “Reliability and Resiliency” starting on page 132, and the section titled “Longevity”, starting on page 130.

The Migration Broker is responsible for carrying out object migrations. This component is responsible for making sure that composite conversions are performed without manual intervention from the point of view of the client application and the rest of the SOA components. Additionally, this component is responsible for recording the performance of each migration service. The results of these measurements are stored in the Evaluations Repository, a knowledge base that supports the recommendation system.

The Format Evaluator provides information about the current status of file formats. This information enables the Migration Advisor to determine which formats are better suited to accommodate the properties of source objects by looking at the characteristics of each pair of formats. This service is supported by a data store containing facts about formats (i.e. Format Knowledge Base), but could also exploit external sources of information such as the PRONOM registry or Google Trends, to determine automatically a format's prevalence and usage.
The current prototype is capable of determining the potential gain (in terms of preservation) that one might obtain in converting an object from its original format to a new one by considering the following set of criteria:

- Market share or adoption – Whether the format is widely accepted or simply a niche format.
- Support level – Does the format provider offer technical support on the format?
- Is it a Standard? – Has format been published by a standards organization?

- Is it an open specification? – Can the specification be independently inspected?
- Does the format support compression, or lossy compression?
- Supports transparency – Does the format support transparency features?
- Embedded metadata – Does the format contain embedded metadata?
- Royalty-free – Are there royalties or license fees?
- Open source – Are there decoders whose source can be independently inspected?
- Backward compatibility – Do revisions support previous versions?
- Documentation level – Is the format specification well-documented?
- Competing formats available – Do competing or similar formats exist?
- DRM support – Does the format support DRM, encryption, or digital signatures?
- Update frequency – Are there frequent revisions which can mitigate against the format being archived?
- Custom extensions – Does the format support executable sections or narrowly supported features, which can be added to the format?
- Maturity – How many years have passed since the format was officially released?
- Transparent decoding – Degree to which the digital representation is open to direct analysis with basic tools, including human readability using a text-only editor.

The Object Evaluator determines the quality of the migration outcome. This is accomplished by comparing objects submitted for migration with their converted counterparts. The evaluations will be performed according to a range of criteria. These criteria, known in this context as significant properties, constitute the set of attributes of an object that should be kept intact during a preservation intervention. This includes the array of attributes that characterize an object as a unique intellectual entity, independent of the encoding used to represent it. The Bible, for example, may exist in many different formats and media, e.g. ASCII text, Portable Document Format, written on paper, or carved on stone, and still be regarded as the Holy Bible. Considering text documents as an example, some significant properties could be: the number of characters, the order of those characters, the page size, the number of pages, the graphical layout, and the font type and size.

The Migration Advisor is responsible for producing suggestions of migration alternatives. In reality, this component acts as a decision support center for client institutions and is capable of determining the best possible choice within a wide range of options. It accomplishes this by

confronting the preservation requirements outlined by client institutions with the accumulated knowledge about the behavior of each accessible migration path. The behavior of each migration path is determined by taking into consideration the sets of criteria previously described: conversion performance, status of the formats involved, and data loss (handled respectively by the Migration Broker, Format Evaluator and Object Evaluator).

In order to archive, one must, in many cases migrate the data. Achieving the right mix of tools and methodologies is paramount to achieve the goal of archiving data for the next 100 to 1000 years in the Next-Generation Data Center design.

Vertical Markets and Use Cases for Riding the Cloud

This section will highlight the capabilities and requirements that should be considered when addressing the Next-Generation Data Center and Cloud implementations (riding the cloud). Many of the following best practices should be considered to be standardized in a cloud environment to ensure interoperability, ease of integration, and portability. A best practice is to consider one's specific implementation and map it to one or all of the use cases described in this paper without using closed, proprietary technologies.

Cloud computing must evolve as an open environment, minimizing vendor lock-in and increasing customer choice. Cloud providers must work together to ensure that the challenges to cloud adoption are addressed through open collaboration and the appropriate use of standards. A best practice is cloud providers should use and adopt existing standards wherever appropriate. The IT industry has invested heavily in existing standards as well as standards businesses methodologies, so there is no need to duplicate or reinvent them.

Best Practice – Implement NGDC's to conform to an "Open Cloud" methodology

As will be outlined, a best practice is to define specific "Use Cases" that provide a practical, customer experience-based context for discussions on interoperability and where existing standards should be used. Another best practice is to emphasize the importance for the IT industry to pay attention to Open Cloud Computing and to also make it clear that standards work remains to be done.

If a particular use case cannot be built today, or if it can only be built with proprietary APIs and products, the industry needs to define standards to make that use case possible. Any community effort around the open cloud should be driven by customer needs, not merely the technical needs of cloud providers, and should be tested or verified against real customer requirements. Cloud computing standards businesses, advocacy groups, and communities should work together and stay coordinated, making sure that efforts do not conflict or overlap. Cloud providers must not use their market position to lock customers into their particular platforms and limit their choice of providers.

Best Practice – Consider Test and Development as a First Step into the Cloud

Many companies hesitate to explore cloud computing because of concerns relating to security, reliability, and cost of an always-on cloud-based application versus one hosted internally. An

ideal cloud application that is suited for cloud evaluation is dev/test (Development and Test). Development and test environments are suited very well for the cloud because of its unique characteristic; a cloud environment can actually address dev/test requirements better than the internal option. Figure 50 shows an example of the interfaces and API's between a cloud provider such as Amazon and the traditional data center and/or hosting provider. Therefore, if you have been waiting to explore cloud computing, it is a best practice to consider using dev/test as one's initial toe in the water. Many dev/test efforts are poorly served by existing infrastructure operations.

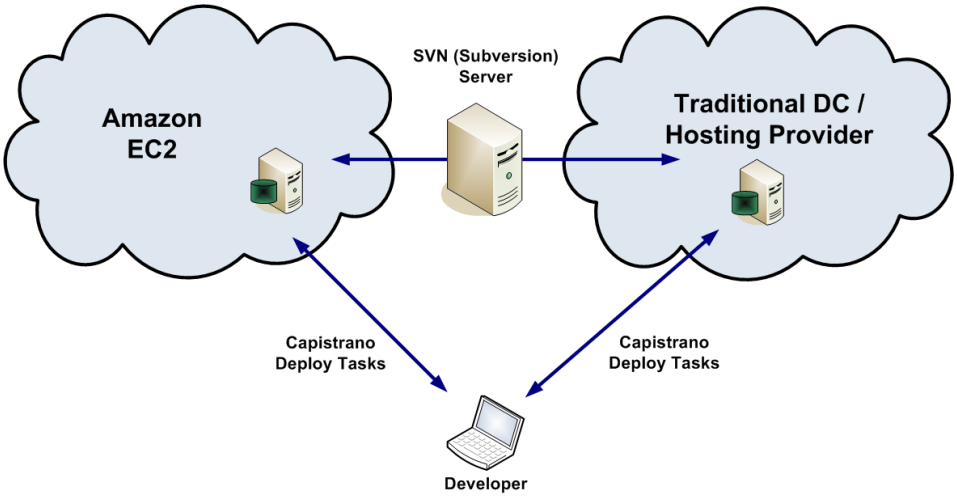


Figure 50 Use Case – Cloud Test and Development

Typically, dev/test environments are underfunded with respect to hardware. Operations get a higher budgeted priority. Companies naturally devote the highest percentage of their IT budget to keeping vital applications up and running. Unfortunately, that means dev/test is usually underfunded and cannot get enough equipment to do its job.

Typically, infrastructure objectives differ from dev/test and other applications. Dev/test is dynamic in nature versus operational applications which are more thought-out. Dev/test environments have the requirement to provision faster. Operations, however, if well managed, have very deliberate, documented, and tracked processes in place, to ensure nothing changes too fast and anything that does change can be audited.

Infrastructure use patterns differ as well. Dev/test use is spiky, while operations seeks smooth utilization to increase hardware use efficiency. A developer will write code, test it out, and then tear it down while doing design reviews, whiteboard discussions, and so on. By its very nature, development is a spiky use of resources. Operations, of course, is charged with efficiency with an aim of lowest total cost of operations.

Operations do not want dev/test to affect production systems. Putting development and test into the production infrastructure, even if quarantined via VLANs, holds the potential of affecting production application throughput, which is anathema to operations groups. Consequently, dev/test groups are often hindered in their attempts to access a production-like environment.

In many cases, dev/test scalability and load testing affect production systems. If putting dev/test in a production environment holds the potential of affecting production applications, what about when dev/test wants to test out how well the application under development responds to load testing or to variable demand? This means that some of the most necessary tasks of development—assessing how well an application holds up under pressure—is difficult or impossible to assess in many environments. Many of the most important bugs only surface under high system load. If the bugs are not found during development, that only means they will surface when in production. In addition, in resource constrained environments, it is difficult to reproduce a production environment topology, which means it is hard to assess, prior to going into production, the impact of network latency, storage throughput, and so forth.

All of these reasons are examples of how current infrastructure management practices optimize toward supporting existing production applications, and not dev/test. Since the focus is always on production, not focusing on dev/test will quite often eventually cause more problems for production systems, where it is most expensive for them to be fixed.

One major advantage of the cloud, as it relates to a test/dev environment, is that there is no contention for resources. Development and QA can each get as much computing resource as they need. Another characteristic that adds value to this model is the lack of long-term commitment necessary for users of a cloud, with respect to individual compute resources. One can use just what is needed at one point in time, and then release the resources with no further commitment. The short-duration, variable usage patterns typical of dev/test are well-aligned with this characteristic. Lastly, another characteristic of this architecture is scale and load that can

easily be tested. Infinite resources or an ability to achieve a superior elastic mode means that setting up a stress test is easy.

Web Services in the Cloud

As previously discussed, there are numerous use cases for the Next-Generation Data Center and moving to the cloud. One use case is shown in Figure 51. This use case is called “Web Facing Applications”, which utilizes web application servers back ended by a data base server. Examples of this scenario include Amazon, Salesforce.com, and others.

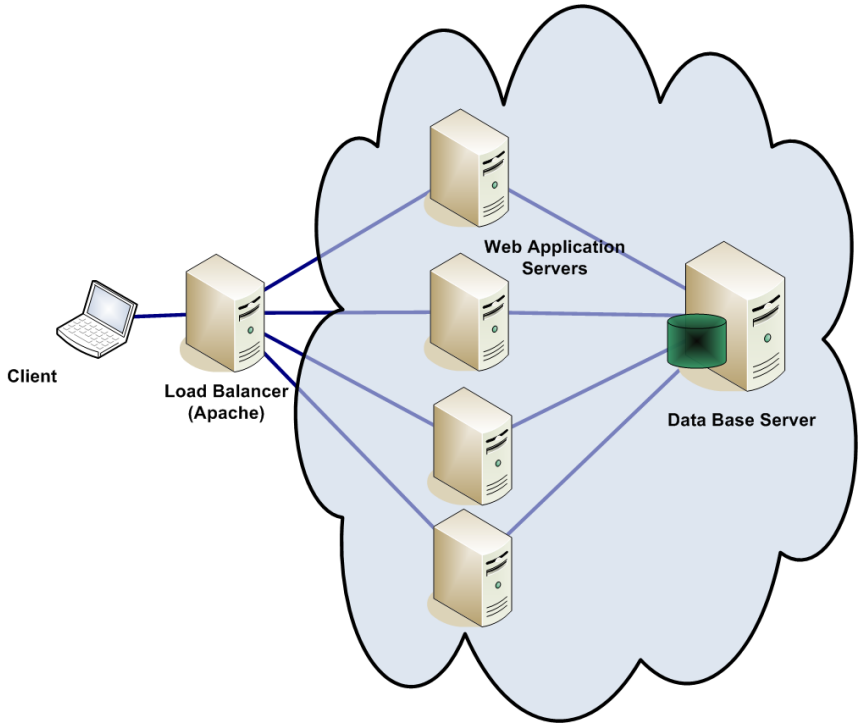


Figure 51 Use Case - Web Service

Web facing applications will typically use a cloud storage offering that provides the data directly to the user’s browser using a URL. The data is typed (MIME) and the browser invokes the appropriate application to view the data. Another use case in Web Services is Media Streaming. Media (audio, video) files are served as a stream of data, allowing use of parts of the data within the file without requiring all data in the file to have been received by the client. Other examples include “YouTube” and *Social Media Sites.* Social media sites include Myspace, Facebook, Twitter, Blogs, etc.

Storage in the Cloud

Another use case is cloud storage as shown in Figure 52. Cloud storage is a typical use case focused on delivering an auxiliary storage space augmenting the web facing social application. Examples include “Joyent” and “Smugmug” which support storing pictures and content within the cloud and are typically URL-based interfaces. A content management system is typically used to keep track of additional metadata associated with the data.

Unstructured data storage within the cloud is another use case under the cloud storage model. This is a pre-allocated storage space (LUN, file system) that is exported via standard client protocols (e.g. WebDAV, NFS, CIFS), and “mounted” on a local machine. Normal POSIX semantics are available at that point for creating/reading/writing/deleting the files.

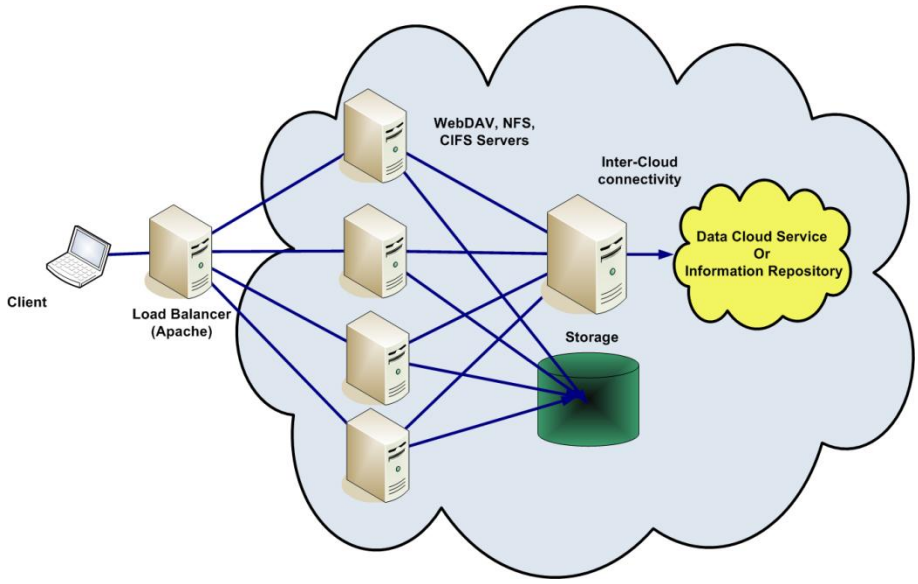


Figure 52 Use Case – Cloud Storage

A number of vendors have offerings in this space. A sub case is cloud desktop, which includes, for example, “iCloud” and “ThinkGrid”. As shown in Figure 52, typical architectures include a load balancer server back-ended by one or more NFS, CIFS, or WebDAV servers. The back end storage can be local storage on SATA or Fiber Channel drives, but many implementations are choosing low-cost direct attached drives. Another possibility is utilizing another data cloud service, using the “Intercloud model”. For more information on the “Intercloud” and other information repositories, please refer to the section titled “The Inter-cloud - Interconnected Global Connectivity”, starting on page 214.

Best Practice – Implement General Content Storage with Synchronization to/from the cloud.

This use case is the ability to synchronize local client data from multiple clients with a cloud storage image replica/version. Changes are detected and then synchronization is done asynchronously and opportunistically. Access may or may not be through standard file protocols and URIs. Clients and servers have a way of sharing state information describing what has/needs to be synced. File sync management may be done through:

- List of exclude/include files
- A dedicated “folder” to synchronize between machines

There are generally three elements to such syncing:

1. Synchronize client-side interface for file input/output
2. Synchronize server-side method for data presentation to clients
3. Create a policy for the synchronization and visibility control

Examples of implementations include MobileMe, Windows Live Mesh, and a combination of Google Docs and Google Gears. Application frameworks include ”Adobe Air”.

Backup to the NGDC/Cloud

Another use case for the Next-Generation Data Center is addressing backing up and archiving clients. This is done by some form of backup software running on local machines where the destination is Cloud Storage as shown in Figure 53.

This is local backup software or backup server, using cloud storage as the destination of backup data. Some implementations of backup software are done on a machine-to-machine basis, such as EMC Mozy. This is a backup application that only backs up a single machine. Some solutions are backup server-based for multiple local machines. This is a local, central backup server that aggregates the use of the cloud storage for one location. An example of this use case is EMC Avamar. It generally takes the form of an appliance, giving the user an interface to manage the appliance. In addition, the appliance would back up its own metadata.

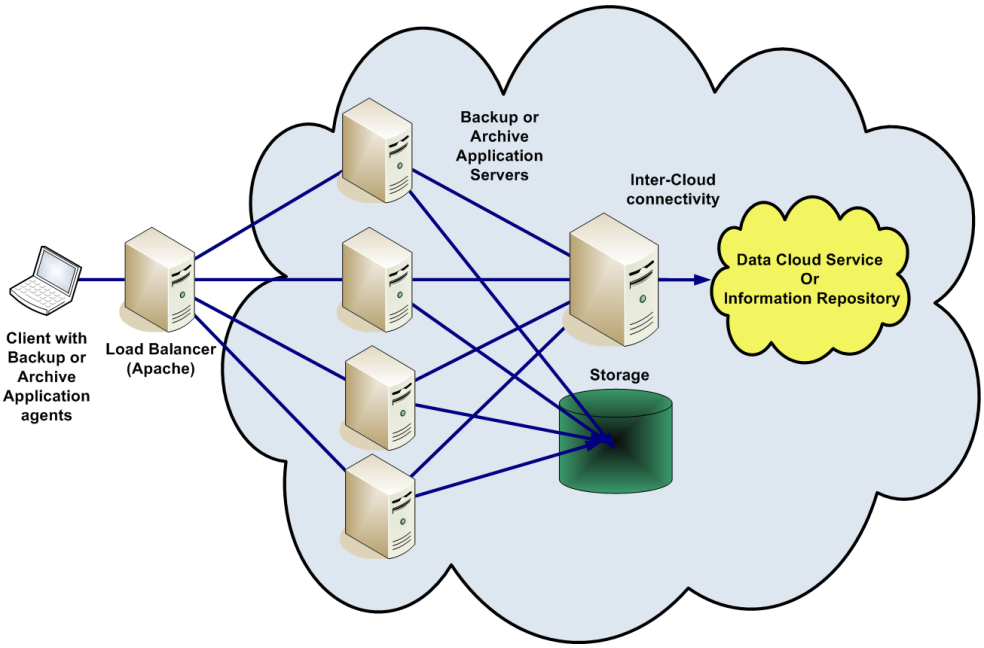


Figure 53 Use Case – Backup and Archive

Another use case is a File Server Appliance locally with embedded backup to cloud. This is a NAS server with integrated backup to the cloud. Examples include Datto, Seagate Free Agent, Iomega's StoreCenter IX2, HP's MediaSmart Server, and Cachengo.

Best Practice – Implement Data Sharing, Native Software and the Intercloud for Server implementations

For server implementations, a best practice and also a common technique used by some local servers is to have client computers turn on data sharing (i.e. becoming a CIFS server), then having the local backup server become a CIFS client of the backup client and then backing up the data in that manner. This is an elegant way to circumvent having to install third party backup software on the backup clients. In terms of backup of cloud computing data or backup of the data used in cloud computing (IaaS), a best practice is to utilize the existing native backup software to increase performance. Examples include VMware's vSphere backup, which includes deduplication as well.

Another best practice is to back up from one cloud provider to the other, as in the case of the Intercloud shown in Figure 53. This is the case of using a second cloud provider as the target of backup data from the first cloud provider.

In terms of restore, this is the obvious reason why you are doing the backups in the first place, needing to restore. Most solutions allow for both online restores as well as physical shipment of media to the customer. Examples include Mozy, which ships DVDs of data. R1Soft is doing bare metal restore.

In terms of Archive/Retention to the cloud, this is the use case of using the cloud storage for archiving of data. Theoretically, XAM should be an ideal interface for this use case. Considerations and best practices for the use case of Archive/Retention to the Cloud are:

- The user should consider maintaining a local copy.
- The cloud provider should do virus scans or other operations on your behalf.
- It is not useful to have to pull the files back over the wire.
- Specify and create a retention period: For example, "Keep my files for X amount of time". This is the case where you define the period of time that you guarantee files will be retained.
- Implement a Secure Deletion process. A best practice is to validate that "When it has gone, it is REALLY gone". This is the case where the service provider provides a means of deleting data in such a way that it has truly gone, i.e. not recoverable by any means. A common alternative method for this is encrypting the data at rest and then shredding the encryption keys.

Another use case is eDiscovery. This use case provides a service, so that when a certain document or documents are required to be produced for a court case, the appropriate documents are produced without undue time or costs. Note that email archiving may be a specialization of this case.

Best Practice – Consider Preservation Options in the Cloud

Preservation is distinguished from Archive/Retention in that the goal of preservation is to actively maintain the upkeep of information, most likely for long periods of time. Libraries and university archives/repositories are also included. Many of these repositories leverage Resource Description Framework (RDF) as a way to describe the data and its relation to other data.

Best Practice – Implement RDF into a Cloud Preservation Framework

The RDF API[48], is a proposed standard for encoding metadata and other knowledge on the Semantic Web. In the Semantic Web, computer applications make use of structured information spread in a distributed and decentralized way throughout the current web. RDF is an abstract model, a way to break down knowledge into discrete pieces, and while it is most popularly known for its RDF/XML syntax, RDF can be stored in a variety of formats. Fedora Commons and other content managers are used to keep track of the metadata. Preserving of machine images along with the data so that a user can at a minimum display the information using the application that generated the data is necessary. Examples include “Fedorazon”. For a more detailed discussion, please refer to the section titled “Longevity”, starting on page 130.

Best Practice – Consider Design Objectives for Databases in the Cloud

Cloud Table Storage use cases are a very important part of the Next-Generation Data Center. Cloud Table Server, storage and infrastructure falls into the following categories:

- Horizontally Scalable, Object-Relational: Examples: Microsoft Azure Tables, Google BigTable, (hyperTable), SimpleDB
- Vertically Scalable, Traditional Relational: Example(s): SQL services
- Document Model: Example(s): CouchDB

Figure 54 describes a use case where a portion of the database is located in the cloud interfacing either directly or indirectly to the application servers via cloud or application server API’s.

[48] www.w3.org/RDF/

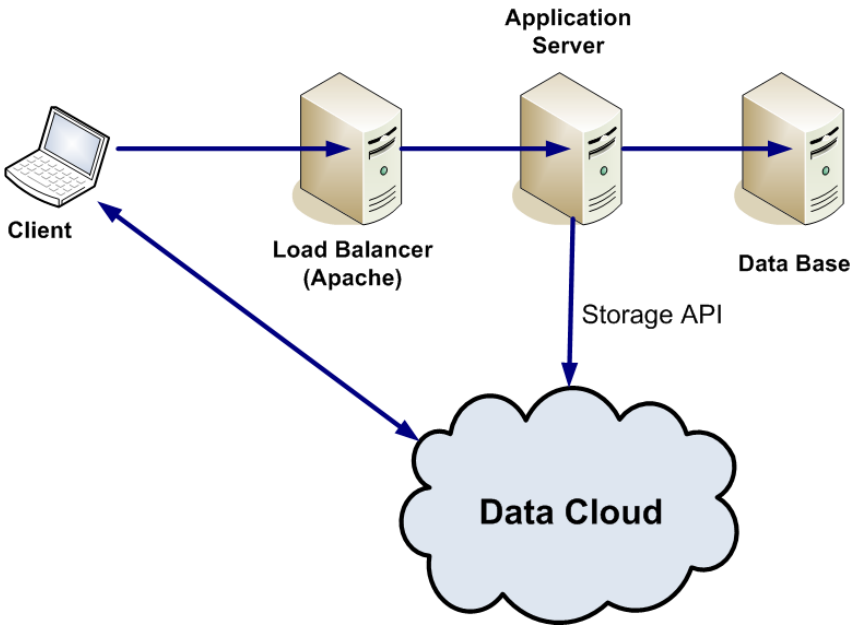


Figure 54 Use Case – Functional Offload

Another use case for database Next-Generation Data Centers is the ability to augment specific cloud-based application servers, utilizing the cloud as shown in Figure 55.

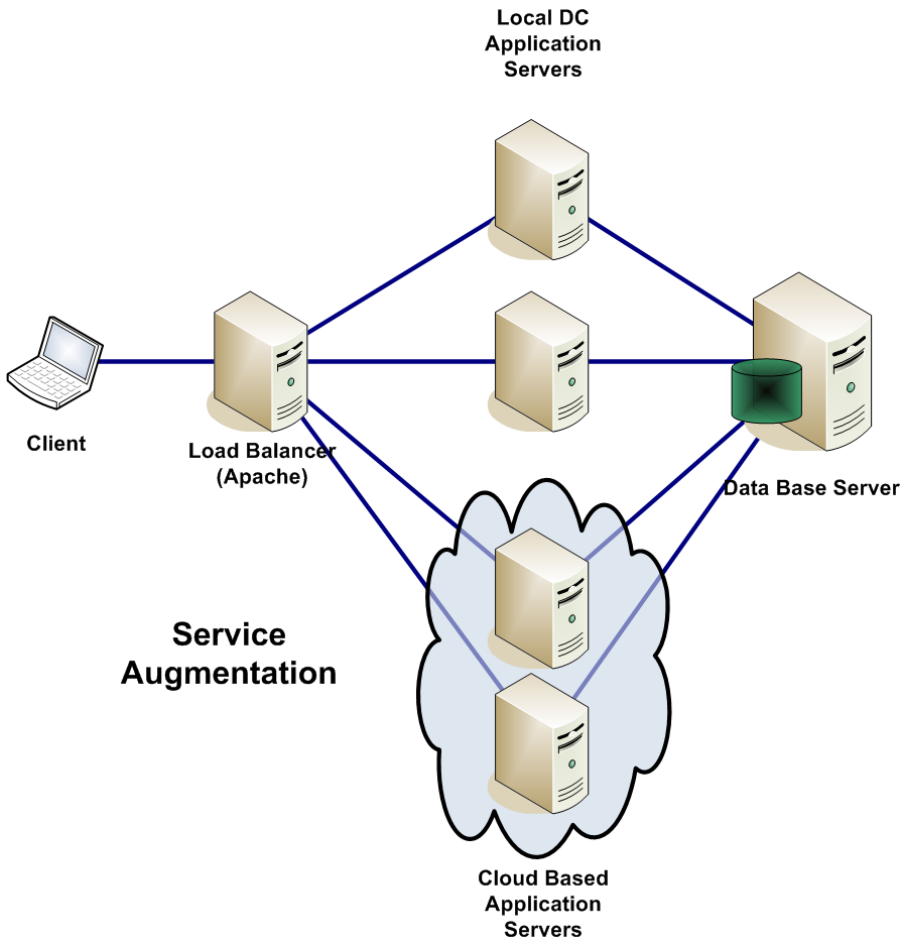


Figure 55 Use Case – Cloud Service Augmentation

Infrastructure as a Service (IaaS) for storage in the cloud

Traditional data storage, which is accessible as part of the computing infrastructure, is changing as we know it. Examples include “EC2” which leverages S3 as if it were a private cloud.

Image Storage, the image of the Guest OS which is made available to hypervisors for starting a VM (Virtual Machine), is another example. The Guest Auxiliary Storage use case is when the Guest provisions the storage space, at a given QoS, which the guest needs beyond the boot storage.

The application Image Storage use case is when the application is maintained in the cloud and invoked locally. Maintenance of the application is done centrally and streamed in, using a statistically probable execution order so that not all bits need be present to start executing. This is a function of the distribution network and can be layered on top of the interfaces.

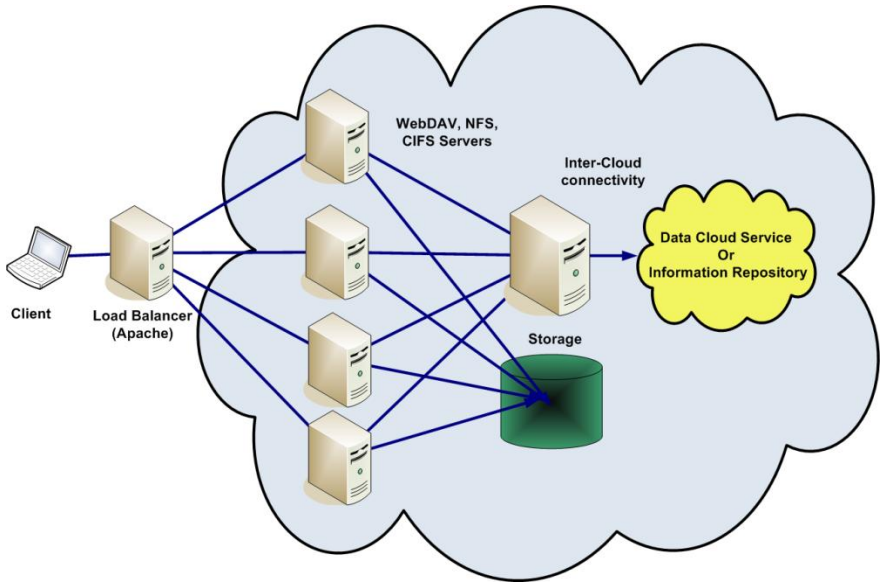


Figure 56 Use Case – Storage as a Service

Content Distribution (Propagating Data Geographically)

Another use case is distributing data globally for the purposes of decreasing latency and increasing scalability. Examples include:

- "Hot" media serving – move to point of presence, replicate out to caches, then recover the resources when unused
- Data transformation in route (i.e. localization, NTSC->PAL in the case of video and audio content streaming)

Cloud Storage Peering (i.e. "Intercloud" Storage)

This is the concept of having the storage clouds of different cloud storage providers able to interoperate between each other (in other words doing for storage clouds what the Internet did for separate, proprietary networks). Possible characteristics include shared storage and replication between cloud storage offerings that can;

- Distribute the data across cloud storage providers (possibly via a storage broker that provides a blended rate).
- Enable data to be erasure encoded, as well as replicated.
- Enable caching and distribution between the client and "dumb storage" provider, geographic staging and replication.

- Activate and de-activate relative to some trust model. Activation requires assembly from the erasure coded blocks, decoding, and decryption. De-activation involves encryption, encoding, and distribution.
- Allows network topology to be more nuanced than the typical two tier processing model and more dynamic as well.

Typical practical examples include “Federated” Cloud Storage”, “A Cloud Exchange”, “Cloud Bursting”, offloading, and Hybrid Internal/External Clouds.

The Inter-cloud - Interconnected Global Connectivity

In order for the Next-Generation Data Center to grow and thrive, it is important to consider not only cloud technologies, types, and interfaces, but the fact that a “cloud” is not an island. There will be situations where one needs to address the interconnectivity of clouds.

Best Practice – Implement an “Intercloud” Strategy in Ones NGDC and/or Cloud Design

The “Intercloud” is an interconnected global "cloud of clouds” and an extension of the Internet "network of networks" on which it is based. The term was first used in the context of cloud computing in 2007 when Kevin Kelly indicated, "eventually we'll have the Intercloud, the cloud of clouds. [49]This Intercloud will have the dimensions of one machine comprising all servers and attendant “cloudbooks” on the planet." It became popular in 2009 and has been used to describe the data center of the future as shown in Figure 57.

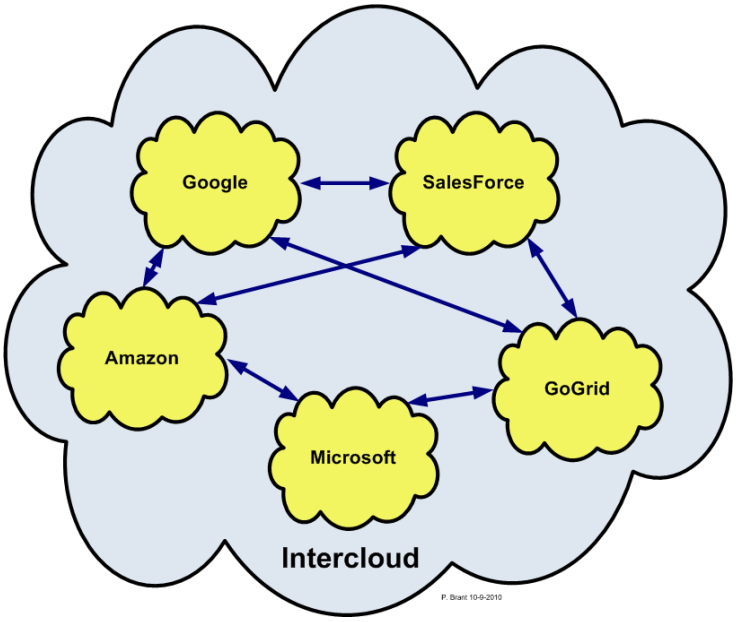


Figure 57 The Intercloud

The Internet is a global network of networks, so it logically follows that the “Intercloud” is a global “cloud of clouds”. It's amazing to think that the Internet has occupied the state of mind of the world for many years, and all it has done is to deliver the functionality to pass messages between any two (or more) clients. To contemplate all the interactions we have utilized with this seemingly simple advance is amazing. Now, with the advent of the Next-Generation Data

[49] http://cloudcomputing.sys-con.com/node/1331133

Center and achieving cloud architecture dominance, it is time to take the Internet to the next level for the next 100 years.

While the servers scaled up as the masses poured in, it wasn't long before we reached a glass ceiling—clearly vertical scalability wasn't the way forward. One can build big machines; after all, for those of us over the age of 40, mainframes and minicomputers will always be a common remembrance. The question becomes, is the cost of big iron vs. commodity white boxes an issue anymore?

Enter Google, Amazon, and others and the entire grid community, who worked out how to make horizontal scalability work properly with systems such as BigTable (A Distributed Storage System for Structured Data) and MapReduce (Simplified Data Processing on Large Clusters).

It can be argued that the Intercloud concept is analogous to the power grid with which we are all familiar. Using the electricity grid analogy, the Internet is like the grid itself, the network of wires and power stations that connect everything together. Now, cloud computing with various cloud providers (and the underlying Internet) is forming the Intercloud. So who invented the term?

Although vendors talk as though there is only one Internet cloud, each vendor will be running its own set of data centers that customers can use to access Internet-based information and resources. This invariably may and will complicate matters.

If one is to fully contemplate the full value of cloud computing, one can consider that this architecture is considered as a loosely coupled "aggregation" of various offerings rather than putting all our eggs in one basket with a single provider.

The Intercloud scenario is based on the key concept that each single cloud does not have infinite physical resources. If a cloud saturates the computational and storage resources of its virtualization infrastructure, it would not be able to satisfy further requests for service allocations sent from its clients. The Intercloud scenario aims to address such a situation, and each cloud can use the computational and storage resources of the virtualization infrastructures of other clouds.

Such use cases as pay-for-use may introduce new business opportunities among cloud providers if they manage to go beyond the theorectical framework. Nevertheless, the Intercloud raises many more challenges than solutions concerning federation, security, interoperability, vendor's lock-ins, trust, legal issues, QoS, monitoring, and billing.

Best Practice - Federate Cloud Computing Environments that Facilitates Just-in-time, Opportunistic, and Scalable Provisioning

Cloud computing providers have set up several data centers at different geographical locations over the Internet to optimally serve the needs of their customers around the world. However, existing systems do not support mechanisms and policies for dynamically coordinating load distribution among different cloud-based data centers in order to determine optimal location for hosting application services to achieve reasonable QoS (Quality of Service) levels. Cloud computing providers are challenged to predict geographic distribution of users consuming their services. As a result, the load coordination must happen automatically, and distribution of services must change in response to changes in the load.

To counter this problem, a proposed best practice is the creation of a federated cloud computing environment or “InterCloud” that facilitates just-in-time, opportunistic, and scalable provisioning of application services, consistently achieving QoS targets under variable workload, resource, and network conditions. The goal is to create a computing environment that supports dynamic expansion or contraction of capabilities (VMs, services, storage, and database) for handling sudden variations in service demands as shown in Figure 58. The parts of the system include:

- *Cloud Coordinator* – Exports cloud services and their management driven by market-based trading and negotiation protocols for optimal QoS delivery at minimal cost and energy.
- *Cloud Broker* – Responsible for mediating between service consumers and cloud coordinators.
- *Cloud Exchange* – A market maker enabling capability sharing across multiple cloud domains through its match-making services.

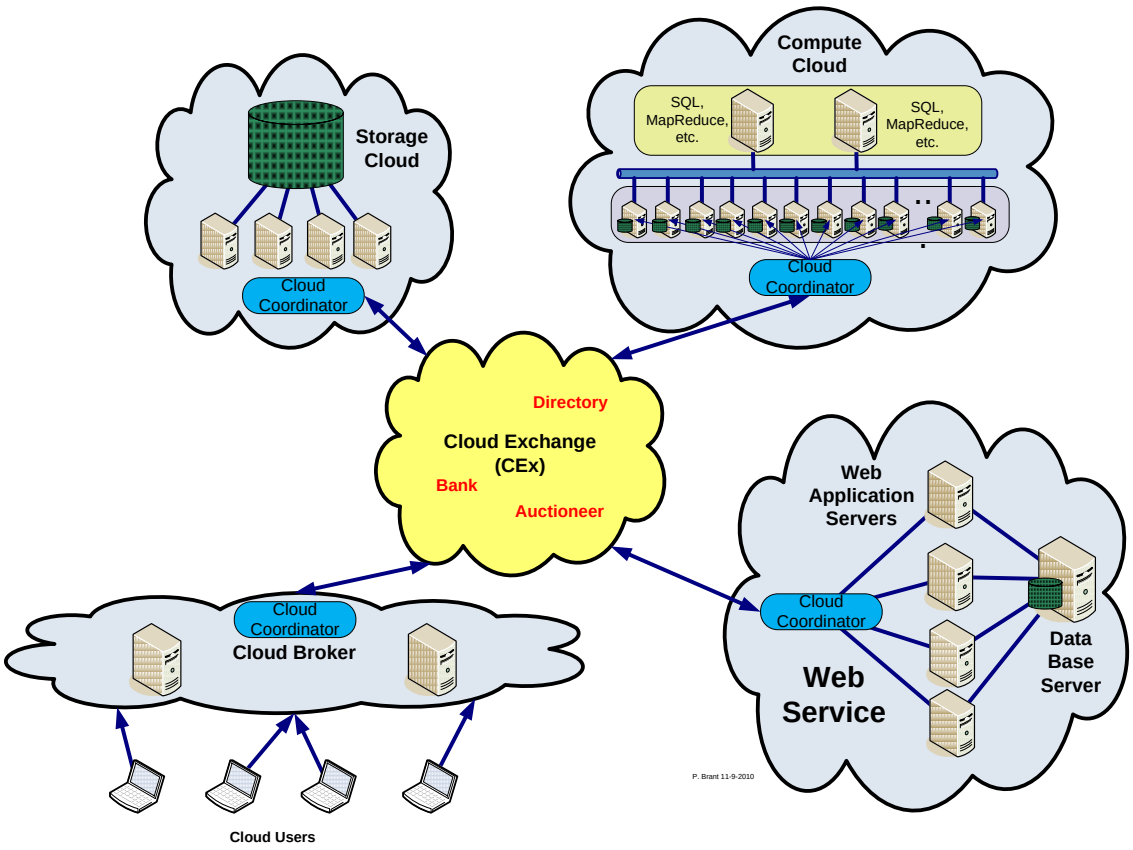


Figure 58 Federated Network Mediated by Cloud Exchange

The Cloud Exchange (CEx) acts as a market maker, bringing together service producers and consumers. It aggregates the infrastructure demands from the application brokers, and evaluates them against the available supply currently published by the Cloud Coordinators. It supports trading of cloud services based on competitive economic models[50], such as commodity markets and auctions. CEx allows the participants (Cloud Coordinators and Cloud Brokers) to locate providers and consumers with fitting offers. Such markets enable services to be commoditized and thus, would pave the way for creation of dynamic market infrastructure for trading based on SLAs. An SLA specifies the details of the service to be provided in terms of metrics agreed upon by all parties, with incentives and penalties for meeting and violating the expectations, respectively. The availability of a banking system within the market ensures that

[50] R. Buyya, D. Abramson, J. Giddy, and H. Stockinger. Economic Models for Resource Management and Scheduling in Grid Computing. Concurrency and Computation: Practice and Experience, 14(13-15): 1507-1542, Wiley Press, New York, USA, Nov.-Dec. 2002.

financial transactions pertaining to SLAs between participants are carried out in a secure and dependable environment.

Every client in the federated platform needs to institute a Cloud Brokering service that can dynamically establish service contracts with Cloud Coordinators via the trading functions espoused (i.e. create a relationship) by the Cloud Exchange.

The Cloud Coordinator service is responsible for the management of domain-specific enterprise clouds and their membership, to the overall federation driven by market-based trading and negotiation protocols. It provides a programming, management, and deployment environment for applications in a federation of clouds. The Cloud Coordinator then exports the services of a cloud to the federation by implementing basic functions for resource management such as scheduling, allocation, (workload and performance) models, market enabling, virtualization, dynamic sensing/monitoring, discovery, and application composition, as discussed below:

In terms of the Cloud Exchange (CEx), as a market maker, the CEx acts as an information registry that stores the cloud's current usage costs and demand patterns. Cloud Coordinators periodically update their availability, pricing, and SLA policies with the CEx. Cloud Brokers query the registry to learn information about existing SLA offers and resource availability of member clouds in the federation. Furthermore, it provides match-making services that map user requests to suitable service providers. Mapping functions will be implemented by leveraging various economic models, such as Continuous Double Auction (CDA) as proposed in earlier works. As a market maker, the Cloud Exchange provides directory, dynamic bidding-based service clearance, and payment management services as discussed below.

The "Market Directory" allows the global CEx participant to match providers or consumers with the appropriate bids/offers. Cloud providers can publish the available supply of resources and their offered prices. Cloud consumers can then search for suitable providers and submit their bids for required resources. Standard interfaces need to be provided, so that both providers and consumers can access resource information from one another, readily and seamlessly.

Auctioneers periodically clear bids and requests received from the global CEx participants. Auctioneers are third party controllers that do not represent any providers or consumers. Since

the auctioneers are in total control of the entire trading process, they need to be trusted by participants.

The banking system of the CEx enforces the financial transactions pertaining to agreements between the global CEx participants. The banks of the CEx are also independent and not controlled by any providers and consumers; thus facilitating impartiality and trust among all cloud market participants that the financial transactions are conducted correctly without any bias. This should be realized by integrating with online payment management services, such as PayPal, with clouds providing accounting services.

Best Practice – Consider Storage Interoperability and Federation within the NGDC and Intercloud

Now let us consider an interoperability use case involving an abstract allegory; for example, running a script or code in a NGDC or in the cloud, which is utilizing cloud-based storage functions. In cloud computing, storage is not like disk access. There are several parameters around the storage which are inherent to the system, and one decides if they meet your needs or not. For example, object storage is typically replicated to several places in the cloud, In AWS and in Azure it is replicated in three places. The storage API is such that a write will return as successful when one replicate of the storage has been effected, and then a “lazy” internal algorithm is used to replicate the object to two additional places. If one or two of the object replicates are lost, the cloud platform will replicate it to another place or two so that it is now in three places. A user has some control over where the storage is physically. For example, one can restrict the storage to replicate entirely in North America or in Europe. There is no ability to vary from these parameters; that is what the storage system provides. We do envision other providers implementing say, five replicates, or a deterministic replication algorithm, or a replicated (DR) write, which doesn’t return until and unless all its replicates are persisted (i.e. validated captured correctly). One can create a large number of variations around “quality of storage” for cloud implementations.

In the interoperability scenario, suppose AWS is running short of storage, or wants to provide a geographic storage location for an AWS customer. Since AWS does not have a data center, it would sub-contract the storage to another service provider. In either of these scenarios, AWS would need to find another cloud which was ready, willing, and able to accept a storage subcontracting transaction with them. AWS would have to be able to have a reliable

conversation with that cloud, again exchanging whatever subscription or usage related information which might have been needed as a precursor to the transaction, and finally have a reliable transport on which to move the storage itself. Note, the S3 storage API is not guaranteed to succeed. If there is a failed write operation from AWS to a subscriber request, the subscriber code is supposed to deal with that (perhaps via an application code level retry). However, cloud to cloud, a target cloud write failing is not something the subscriber code can take care of. A best practice is that in this scenario, the write needs to be reliable.

Although the addressing issues are not as severe in this case where abstract allegories are used, the naming, discovery, and conversation setup items challenges all remain.

Best Practice – Conform to a Standard Protocols Profile

To address interoperability use cases such as these, certain commonalities between peer cloud entities must be adopted. With the Internet, interoperability foundations were set with the basics of IP addressing, DNS, exchange and routing protocols, such as BGP, OSPF, and peering conventions using AS (Autonomous System[51]) numbering.

It is important to consider implementing an Intercloud that can support various kinds of virtualized infrastructures; one example is using hypervisors from VMware and the associated tooling and conventions that are associated with that set of products. Another consideration is using open source hypervisors, such as Xen and KVM from RedHat, and the associated tooling and conventions (Linux, and AWS-like) that are associated with that set of products.

Figures 59, 60, and 61 describe the domain of cloud standards that should be considered for the transformation of the existing cloud architectures supporting the Next-Generation Data Center.

[51] Guidelines for creation, selection, and registration of an Autonomous System (AS), and related other RFCs at http://tools.ietf.org/html/rfc1930

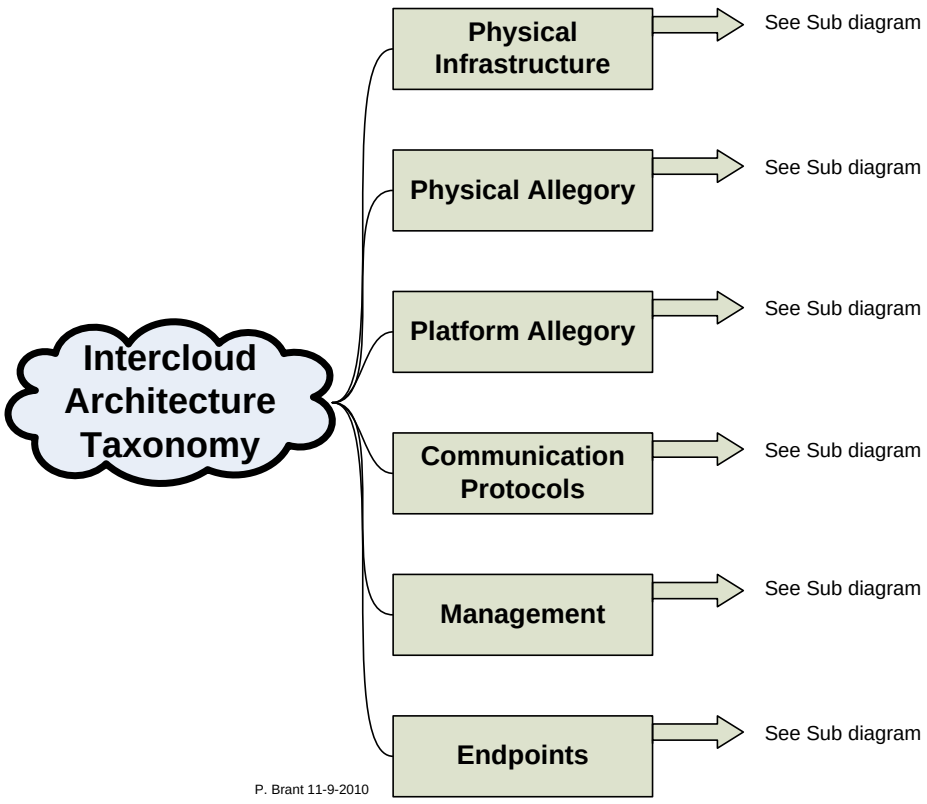


Figure 59 Intercloud Top Level Standards Taxonomy

The way the "Power Grid" is ubiquitous in today's society, the Intercloud will most likely become the IT Grid of the future. As with the power grid, standards are required. Aspects of the physical infrastructure (such as the network pipe architectures such as Fiber Channel, FCOE (Fiber Channel Over Ethernet) and other technologies) must be enumerated. Communication protocols must be standardized as well as management elements. What will the endpoints look like and how do we secure them?

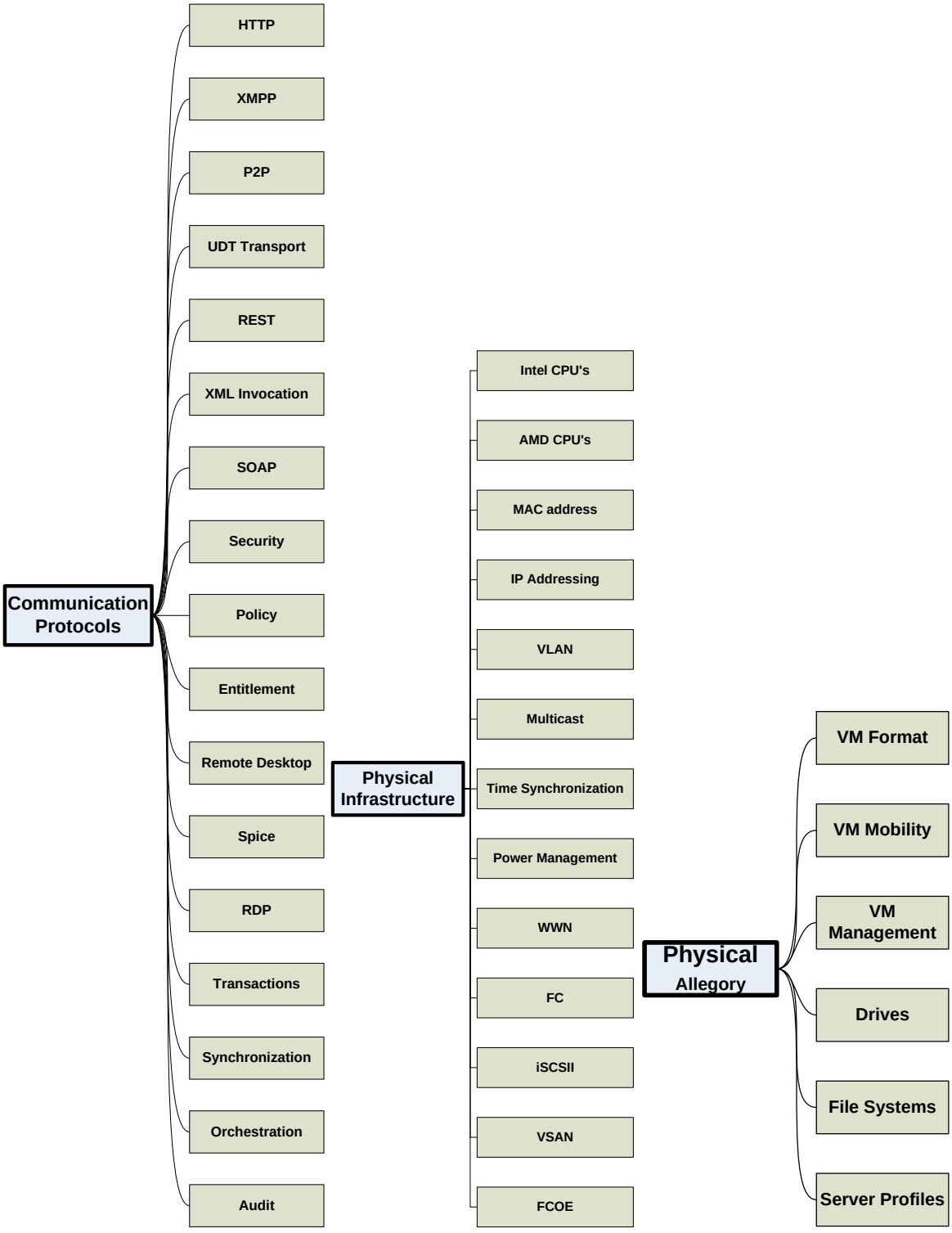


Figure 60 Intercloud Sub Attributes Taxonomy Page A

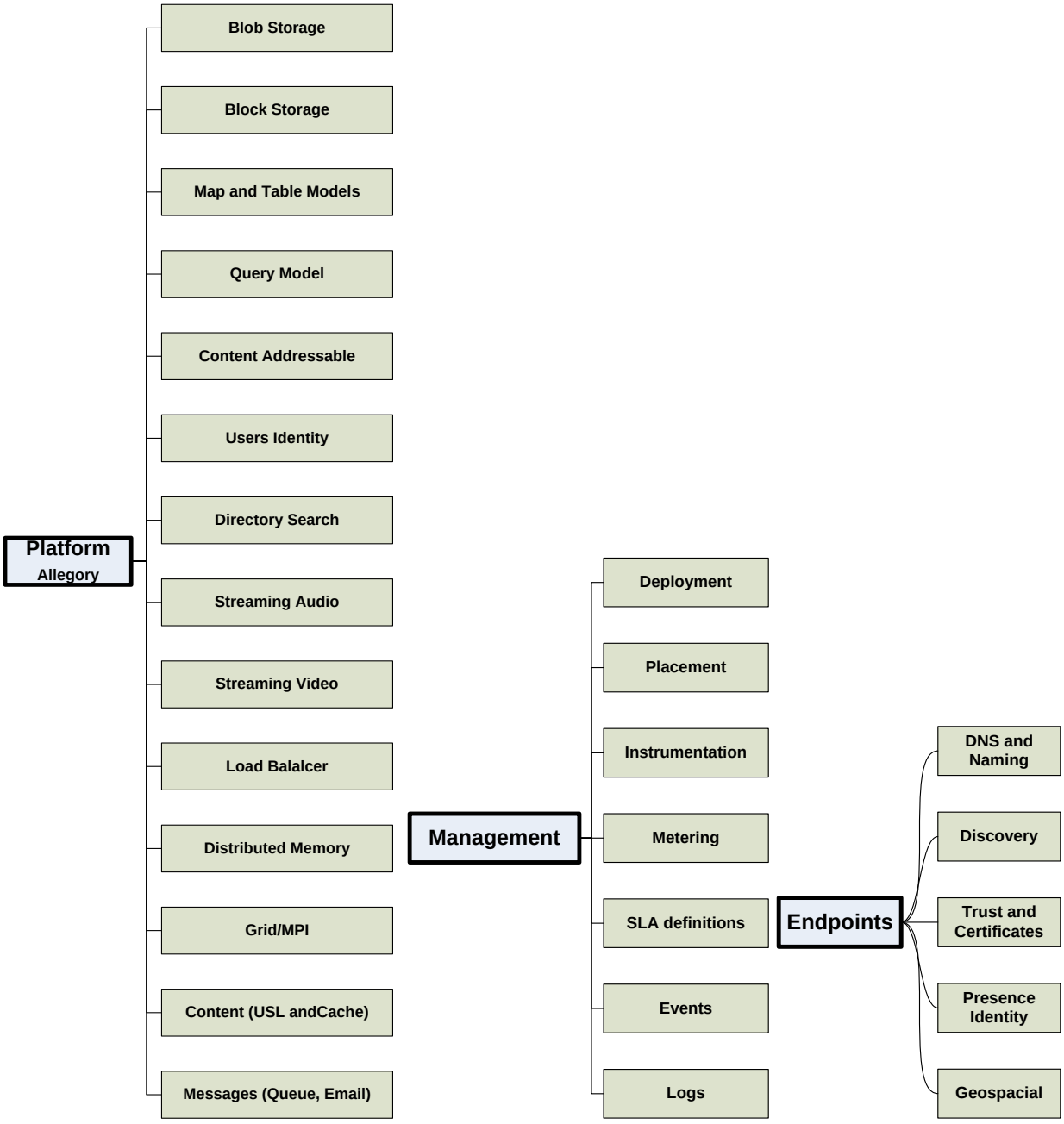


Figure 61 Intercloud Sub Attributes Taxonomy Page B

Interoperability is a major challenge in each taxonomy sub-area.

Best Practice – Understand Mobility Aspects of VMs in the Cloud in the IP Address Domain

One area that imposes major challenges is network addressing. IP address space explodes in highly virtualized environments. Everything has multiple IP addresses; servers have IP

addresses for management, for the physical NICs, for all of the virtual machines and the associated virtual NIC's, and if any virtual appliances are installed, they have multiple IP addresses as well.

Several areas are of concern. One concern is the IPv4 address space simply starts to run out. Consider an environment inside the cloud which has 1M actual servers. As explained above, assuming a 16 core server, each server could have 32 VM's, and each VM could have a handful of IP addresses associated with it (virtual NICs, etc). That could easily explode to a cloud with well over 32M IP addresses. Even using Network Address Translation (NAT), the 24-bit Class A reserved Private Network Range provides a total address space of only 16M unique IP addresses.

As a result, many cloud operators are considering switching to IPv6, which provides for a much larger local address space, providing unique IP addresses in the trillions. Switching to IPv6 is quite an undertaking, and some believe that switching from one static addressing scheme to another static addressing scheme (i.e. IPv4 to IPv6) might not be the right approach in a large, highly virtualized environment, such as cloud computing. When reconsidering addressing, one should consider the mobility aspects of VMs in cloud.

VM Mobility can create new challenges in any static addressing scheme. When one moves a running VM from one location to another, the IP address goes with the running VM and any application runtimes hosted by the VM. IP addresses (of either traditional type) embody both Location and Identity in the IP address, i.e. routers and switches use the IP address not only to uniquely identify the endpoint, but by virtue of decoding the address, there is an inference of the Location of the endpoint and how to reach that endpoint using switching and routing protocols. So, while an addressing scheme is being reconsidered, one can consider two schemes which embody mobility as discussed below.

One possible solution and a best practice is utilizing Mobile IPv4 and Mobile IPv6 mechanisms. The only problem is these two standards are not interoperable. Because we are trying to solve the problem from one cloud to another, we need a protocol that has a common, interoperable mobility scheme which can be mapped/encapsulated in both IPv4 and IPv6.

Best Practice – Implement Location Identity Separation Protocol in the NGDC Intercloud Design

A completely dynamic scheme where Location and Identification have been separated has been created in an attempt to completely generalize the addressing solution in a way that interoperates with both IPv4 and IPv6, This new scheme is called Location Identity Separation Protocol (LISP). LISP-based systems can interoperate with both IPv4- and IPv6-based networks, through protocol support on edge routers. However, internal to a cloud, which may in itself span several geographies, LISP addressing may be used.

The basic idea behind the Loc/ID split is that the current Internet routing and addressing architecture combines two functions: Routing Locators (RLOCs), which describe how a device is attached to the network, and Endpoint Identifiers (EIDs), which define "who" the device is, in a single numbering space, the IP address. Proponents of the Loc/ID split argue that this "overloading" of functions places the constraints on end-system use of addresses that we detailed. Splitting these functions by using different numbering spaces for EIDs and RLOCs yields several advantages, including improved scalability of the routing system through greater aggregation of RLOCs. To achieve this aggregation, we must allocate RLOCs in a way that is congruent with the topology of the network. EIDs, on the other hand, are typically allocated along business boundaries.

Because the network topology and business hierarchies are rarely congruent, it is difficult (if not impossible) to make a single numbering space efficiently serve both purposes without imposing unacceptable constraints (such as requiring renumbering upon provider changes) on the use of that space. LISP, as a specific instance of the Loc/ID split, aims to decouple location and identity. This decoupling will facilitate improved aggregation of the RLOC space, so, a best practice is to implement persistent identity in the EID space. This will hopefully increase the security and efficiency of network mobility.

As a result, current experimentation is being done to assess the viability of using this protocol in conjunction with virtualization and, in particular, with VM Mobility. Of course, if and when LISP becomes a proven solution for the cloud scenario, it must propagate into many forms of networking equipment, which will take some time. For more information on LISP, please refer to the "Networking section" in the 2010 Proven Professional Paper titled "Above The Clouds - Best

practices to Create a Sustainable Computing Infrastructure to Achieve Business Value and Growth", by Paul Brant.

Best Practice – Consider Naming, Identity and Trust in the Creation of Intercloud Resources

One should consider that public or private clouds are not endpoints, in the way that servers or clients are. They are resources, and as such, are typically identified using a uniform resource identifier or URI[52]. However, a simple name lookup, allowing one to access a URI over the Internet is not sufficient for cloud computing. It is a best practice to make sure that cloud providers provide assurances that this is indeed the service it says it is, make transparent more detail about what service levels, capabilities, and requirements this service may offer and, since we are using something outside of our local trust domain, perhaps have some audit capabilities.

Using DNS can be part of an Intercloud Root, and can also be part of a cloud computing instance. In addition to DNS-like capabilities, we would like a rich capability for expressing names and services, like a directory service, such as LDAP or Active Directory. It is not a best practice which allows clouds communicating over a non-secure networks to prove their identity to one another in a secure manner, such as Kerberos. A system which can supply trusted security certificates, such as the X.509 is a better methodology. This provides for a public key infrastructure (PKI) for single sign-on and Privilege Management Infrastructure (PMI). X.509 specifies, among other things, standard formats for public key certificates, certificate revocation lists, attribute certificates, and a certification path validation algorithm.

Utilizing IPA[53] , which is an integrated security information management solution, offers an integrated security information management solution combining an open LDAP directory server, MIT Kerberos, and a X.509 Certificate Authority. IPA provides the functions of:

- Identity (machine, user, virtual machines, groups, authentication credentials)
- Policy (configuration settings, access control)
- Audit (events, logs, analysis, and so on)

---

[52] Uniform Resource Identifiers (URI): Generic Syntax, and related other RFCs, at http://www.ietf.org/rfc/rfc2396.txt

[53] The FreeIPA Project at http://freeipa.org

In IPA, one user ID is shared between LDAP and Kerberos, and Kerberos gets the benefit of the directory server's multimaster replication. IPA provides an XML over RPC interface, to allow for automation and self service with cloud infrastructure. IPA is a centralized authentication point, which tracks what persons or services logged onto what and when. Services mutually authenticate and encrypt with Kerberos. DNS and Certificate Authority are currently being integrated into IPA.

For additional information on identity, management, and trust, please refer to the 2010 Proven Professional Paper titled "How to Trust the Cloud – "Be careful up there"", by Paul Brant and Denis Guyadeen for a more detailed discussion on this topic.

Best Practice – Consider Presence and Messaging in the Creation of Intercloud Resources

Part of interoperability is that cloud instances must be able to dialog with each other. As the use cases explained, one cloud must be able to find another cloud, which for a particular interoperability scenario, is ready, willing, and able to accept an interoperability transaction with and furthermore, exchange whatever subscription- or usage-related information which might have been needed, as a precursor to the transaction. Therefore, a best practice is an Intercloud Protocol for presence and messaging needs to exist.

A best practice is to utilize "Extensible Messaging and Presence Protocol" (XMPP)[54] in the Intercloud. XMPP is a set of open XML technologies for presence and real-time communication developed by the Jabber open-source community in 1999, formalized by the IETF in 2002-2004, and continuously extended through the standards process of the XMPP Standards Foundation. XMPP supports presence, structured conversation, lightweight middleware, content syndication, and generalized routing of XML data.

Best Practice – Consider Multicast IP Architectures for Video and Audio Delivery

An area of particular interest is where applications running on clouds are rich media enabled, or are collaboration applications. Application-enabling large numbers of people to work together who are audio and video enabled are exciting applications for cloud computing. Augmentation of

[54] Extensible Messaging and Presence Protocol (XMPP): Core, and related other RFCs at http://xmpp.org/rfcs/rfc3920.html and XMPP Standards Foundation at http://xmpp.org/

social applications such as Facebook and MySpace with rich media, multi-point collaboration is a challenge to the infrastructure which supports them. It is well known that massive scale, real-time, multi-point applications are well served by IP Multicast.

IP Multicast is a well understood technology. However, most service provider infrastructures do not currently allow one to transit IP Multicast on their networks, as it is very demanding on their routers. Within a cloud computing environment, we see this as a crucial element for Intercloud, in that applications which want to use API's which ultimately will use IP Multicast for implementation must be supported. More importantly, for these types of applications to work in an Intercloud context, IP Multicast must work in between and stratified for cloud enablement. This requires Interdomain IP Multicast routes.

Another challenge is if a LISP addressing scheme has been adopted, as discussed above, a LISP-enabled multicasting architecture would need to be implemented.

For more information on Multicast Next Generation video and audio streaming considerations, please refer to the 2007 Knowledge Sharing Paper titled "Challenges and Best Practices in the Deployment and Management of IPTV Networks", by Paul Brant.

Best Practice – Consider Time Synchronization for NGDC and Intercloud implementations

Depending on the applications, time synchronization may not be very important. Network Time Protocol (NTP) may be sufficient for cloud computing instances, in terms of keeping accurate time, and in accurately synchronizing the distributed computing elements in the Cloud. However, our research has shown considerable clock drift in a distributed system and applications which depend on accurate time will not return correct results or in some cases, function incorrectly.

Precision time synchronization will likely be an important aspect of cloud computing. We have spent considerable time on a precision time capability, called IEEE 1588. In the context of cloud computing, there is nothing additional for industry to do here, except perhaps to realize that Intercloud Protocol capability may rely on having precision timing in the cloud. It is a consideration of ours that the Intercloud Root may be a source for this time synchronization in the IEEE 1588 format as well.

Best Practice – Consider XMPP to Create a Dependable Application Transport Model

Using XMPP for control plane information is sufficient. However, when services need to move payloads in a transactional manner, like exchanging business records, customer data, critical storage blocks, or anything demanding a reliable, transactional application transport, or a different mechanism, is required.

Applications needing this functionality have traditionally turned to MQseries from IBM, JMS from BEA Weblogic, or other J2EE provider, or the The Information Bus from TIBCO. In the cloud computing world, AWS includes a service called SQS. Unfortunately, as of the time of this paper, none of the applications message protocols interoperate since the on-the-wire formats are all different.

One best practice is to consider inplementing an interoperable message queue standard, called Advanced Message Queuing Protocol (AMQP)[55]. AMQP is an open standard application layer protocol for Message Oriented Middleware. The defining features of AMQP are message orientation, queuing, routing (including point-to-point and publish-and-subscribe), reliability, and security. AMQP requires the behavior of the messaging provider and client to be such that implementations from different vendors are truly interoperable, in the same way as SMTP, HTTP, FTP, and so forth have created interoperable systems. Previous attempts to standardize middleware have happened at the API level (e.g. JMS) and this did not create interoperability. Unlike JMS, which merely defines an API, AMQP is a wire-level protocol.

[55] Advanced Message Queuing Protocol, at http://jira.amqp.org

Conclusion

The Digital Crisis is upon us. The digital information universe is growing at an exponential pace. How you find information you want in a reasonable timeframe was discussed. Data centers can be characterized as environments with petabytes of distributed information, high data growth rates, many facilities and many departments with uncoordinated responsibilities and requirements, and lack of business-level budget, interest, and focus on its archives, as well as recently created information. All these operating challenges were discussed, and solutions were defined as to how to reduce risk. New business process and technology practices were discussed, to help one ride the private and public cloud with a burgeoning approach, foreshadowing the transformation process leading to what the Next-Generation Data Center will look like.

In summary, this article described how to ride the “cloud”, how to utilize what technology can afford, offering Best Practices that will align with the most important goal, creating a Next - Generation Data Center, addressing the business challenges of today and tomorrow through data business and technology transformation.

Appendix A – Cloud Definitions and Taxonomy

The following definitions and taxonomy are included to provide an overview of cloud computing concepts.

Cloud computing is a model for enabling ubiquitous, convenient, on-demand network access to a shared pool of configurable computing resources (e.g. networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction. (This definition is from the latest draft of the NIST Working Definition of Cloud Computing published by the U.S. Government's National Institute of Standards and Technology.)

Cloud computing defines three delivery models:
Software as a Service (SaaS): The consumer uses an application, but does not control the operating system, hardware or network infrastructure on which it is running.
Platform as a Service (PaaS): The consumer uses a hosting environment for their applications. The consumer controls the applications that run in the environment (and possibly has some control over the hosting environment), but does not control the operating system, hardware, or network infrastructure on which they are running. The platform is typically an application framework.
Infrastructure as a Service (IaaS): The consumer uses "fundamental computing resources" such as processing power, storage, networking components, or middleware. The consumer can control the operating system, storage, deployed applications, and possibly networking components such as firewalls and load balancers, but not the cloud infrastructure beneath them.

The NIST definition defines four deployment models:

- Public Cloud: In simple terms, public cloud services are characterized as being available to clients from a third party service provider via the Internet. The term “public” does not always mean free, even though it can be free or fairly inexpensive to use. A public cloud does not mean that a user’s data is publically visible; public cloud vendors typically provide an access control mechanism for their users. Public clouds provide an elastic, cost effective means to deploy solutions.
- Private Cloud: A private cloud offers many of the benefits of a public cloud computing environment, such as being elastic and service based. The difference between a private

cloud and a public cloud is that in a private cloud-based service, data and processes are managed within the businesses without the restrictions of network bandwidth, security exposures, and legal requirements that using public cloud services might entail. In addition, private cloud services offer the provider and the user greater control of the cloud infrastructure, improving security and resiliency because user access and the networks used are restricted and designated.

- Community Cloud: A community cloud is controlled and used by a group of businesses that have shared interests, such as specific security requirements or a common mission. The members of the community share access to the data and applications in the cloud.
- Hybrid Cloud: A hybrid cloud is a combination of a public and private cloud that interoperates. In this model, users typically outsource nonbusiness-critical information and processing to the public cloud, while keeping business-critical services and data in their control.

There are five essential characteristics of cloud computing. A private cloud can be managed by a third party and can be physically located off premises. It is not necessarily managed and hosted by the businesses that use it. A Hybrid Cloud is a superset of the technology used in a Community Cloud.

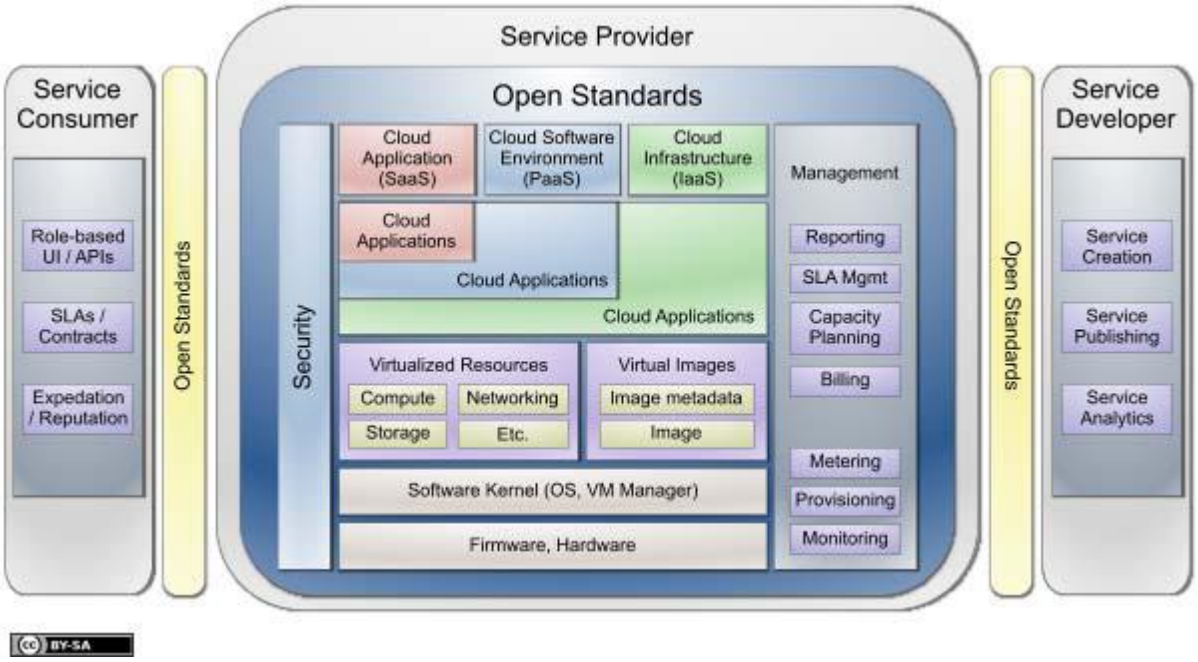
Other definitions include;

- Rapid Elasticity: Elasticity is defined as the ability to scale resources both up and down as needed. To the consumer, the cloud appears to be infinite, and the consumer can purchase as much or as little computing power as they need. This is one of the essential characteristics of cloud computing in the NIST definition.
- Measured Service: In a measured service, aspects of the cloud service are controlled and monitored by the cloud provider. This is crucial for billing, access control, resource optimization, capacity planning, and other tasks.
- On-Demand Self-Service: The on-demand and self-service aspects of cloud computing mean that a consumer can use cloud services as needed without any physical user interaction with the cloud provider.
- Ubiquitous Network Access: Ubiquitous network access means that the cloud provider's capabilities are available over the network and can be accessed through standard mechanisms by both thick and thin clients.
- Resource Pooling: Resource pooling allows a cloud provider to serve its consumers via a multi-tenant model. Physical and virtual resources are assigned and reassigned

according to consumer demand. There is a sense of location independence in that the customer generally has no control or knowledge over the exact location of the provided resources but may be able to specify location at a higher level of abstraction (e.g. country, state, or data center).

- Multi-Tenancy: Multi-tenancy is the property of multiple systems, applications, or data from different enterprises hosted on the same physical hardware. Multi-tenancy is common to most cloud-based systems.
- Cloud bursting: Cloud bursting is a technique used by hybrid clouds to provide additional resources to private clouds on an as-needed basis. If the private cloud has the processing power to handle its workloads, the hybrid cloud is not used. When workloads exceed the private cloud's capacity, the hybrid cloud automatically allocates additional resources to the private cloud.
- Policy: A policy is a general term for an operating procedure. For example, a security policy might specify that all requests to a particular cloud service must be encrypted.
- Governance: Governance refers to the controls and processes that make sure policies are enforced.

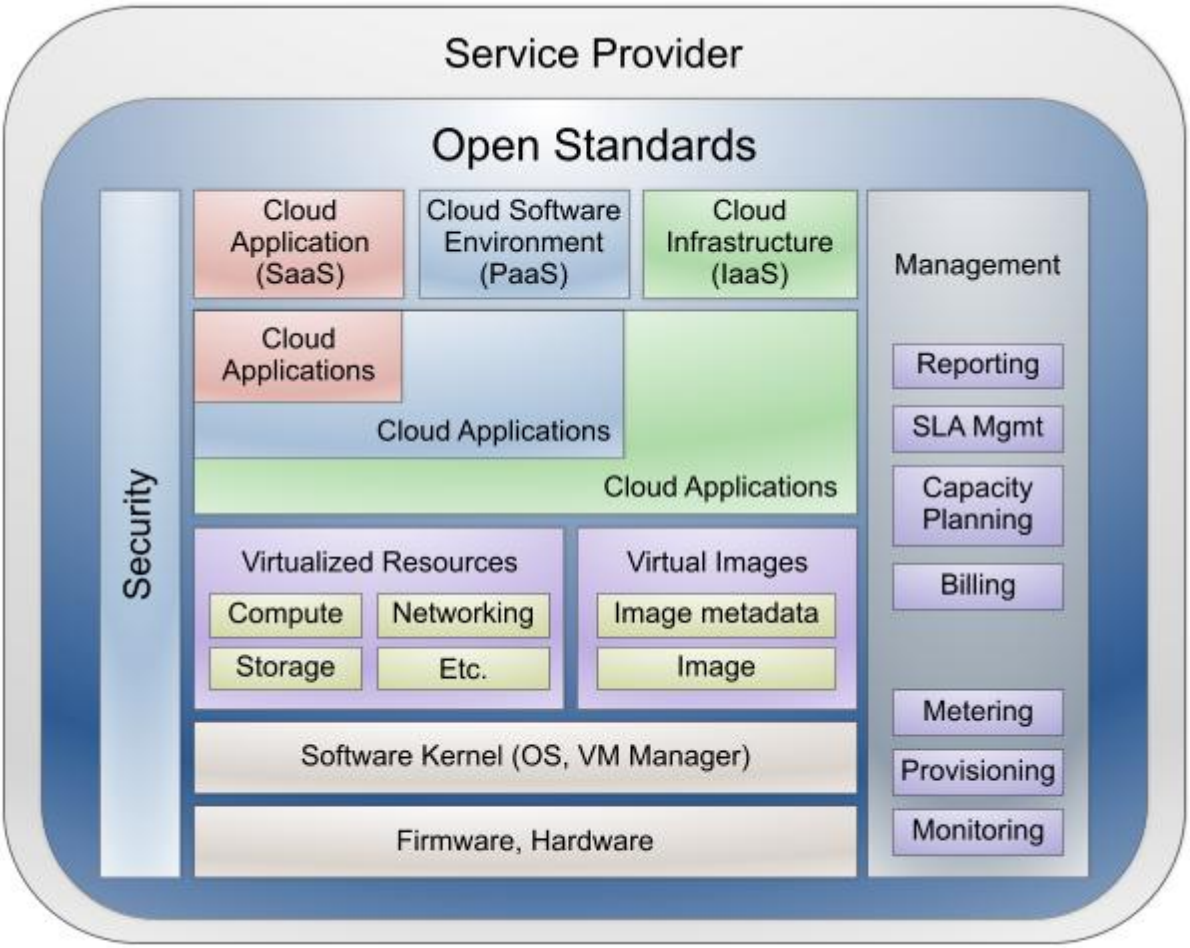
Taxonomy - This diagram defines a taxonomy for cloud computing:



In this diagram, Service Consumers use the services provided through the cloud, Service Providers manage the cloud infrastructure and Service Developers create the services

themselves. (Notice that open standards are needed for the interactions between these roles.) Each role is discussed in more detail in the following sections.

The Service Consumer is the end user or enterprise that actually uses the service, whether it is Software, Platform or Infrastructure as a Service. Depending on the type of service and their role, the consumer works with different user interfaces and programming interfaces. Some user interfaces look like any other application; the consumer does not need to know about cloud computing as they use the application. Other user interfaces provide administrative functions such as starting and stopping virtual machines or managing cloud storage. Consumers writing application code use different programming interfaces depending on the application they are writing. Consumers work with SLAs and contracts as well. Typically, these are negotiated via physical user intervention between the consumer and the provider. The expectations of the consumer and the reputation of the provider are a key part of those negotiations.



The Service Provider delivers the service to the consumer. The actual task of the provider varies depending on the type of service:

- For Software as a Service, the provider installs, manages, and maintains the software. The provider does not necessarily own the physical infrastructure in which the software is running. Regardless, the consumer does not have access to the infrastructure; they can access only the application.
- For Platform as a Service, the provider manages the cloud infrastructure for the platform, typically a framework for a particular type of application. The consumer's application cannot access the infrastructure underneath the platform.
- For Infrastructure as a Service, the provider maintains the storage, database, message queue or other middleware, or the hosting environment for virtual machines. The consumer uses that service as if it was a disk drive, database, message queue, or machine, but they cannot access the infrastructure that hosts it.

In the service provider diagram, the lowest layer of the stack is the firmware and hardware on which everything else is based. Above that is the software kernel, either the operating system or virtual machine manager that hosts the infrastructure beneath the cloud. The virtualized resources and images include the basic cloud computing services such as processing power, storage, and middleware. The virtual images controlled by the VM manager include both the images themselves and the metadata required to manage them. Crucial to the service provider's operations is the management layer. At a low level, management requires metering to determine who uses the services and to what extent, provisioning to determine how resources are allocated to consumers, and monitoring to track the status of the system and its resources.

At a higher level, management involves billing to recover costs, capacity planning to ensure that consumer demands will be met, SLA management to ensure that the terms of service agreed to by the provider and consumer are adhered to, and reporting for administrators. Security applies to all aspects of the service provider's operations. (The many levels of security requirements are beyond the scope of this paper.) Open standards apply to the provider's operations as well. A well-rounded set of standards simplify operations within the provider and interoperability with other providers.

Appendix B – Next-Generation Data Center of the future challenges

This table is compiled from a wide variety of sources and is intended to serve as consideration and planning guidelines for storage and data management planning for the Next-Generation Data Center of the future.

Table 5 - Data Center of the future challenges

Challenge	Value
Average annual digital storage demand rate (primary occurrence of data, all platforms)	35-40% (2006-2008)
Amount of magnetic disk data stored on Unix, Windows, and Linux systems WW (est.)	>90%
Average disk allocation levels for z/OS (eSeries mainframes using DFSMS suite)	60-80%
Average disk allocation levels for iSeries (AS/400 servers)	60-80%
Average disk allocation levels for Unix/Linux systems	30-45%
Average magnetic disk allocation levels for Windows systems	25-40%
Average annual disk drive areal density increase	35-50% (downward trend expected as recording limits begin to appear)
Average annual disk drive performance improvement (seek, latency and data rate)	<4% (mainly with data rate, as seek time improvement is minimal)
Increase in disk drive capacity per actuator since the first disk drive in 1956	200,000x (5MB to 1000 GB)
Increase in native tape cartridge capacity since the first tape cartridge in 1984	4,000x (200MB to 800 GB = 1.6TB compressed)
Recommended data center power	50-100 watts per square foot

consumption level	
Power usage breakdown in typical data center	Chiller – 33%, IT gear – 30%, UPS – 18%, AC – 9%
Average cost to build a Tier 3 data center	$480 per square foot
Electricity consumed by hi-density blade servers	>7kW/rack and > 30kW/enclosure
Annual average increase in electricity cost	20-40% (depending on geography)
Who gets the IT energy bill?	Facilities team – 56%, IT team 3%
Average tape cartridge utilization levels for integrated virtual tape systems	60-80%
Typical range of disk data managed per administrator (for non-mainframe systems –Windows, Unix, Linux)	500GB – 10TB
Typical amount of disk data managed per administrator (z/OS, mainframe)	>50TB
Estimated range of automated tape data managed per administrator (all platforms)	40TB to >1EB (varies widely based on library size)
Annual growth rate of unwanted e-mail message traffic	~350%
Estimated percentage of SANs that are homogeneous (the same operating system)	70-75% (Mainframes, Unix and Windows systems only)
Percent of NAS deployed databases greater than 500GB (est.)	~10%
Percentage of SMBs implementing iSCSI SANs in 2006	37% (was 20% in 2006)
Maximum possible distance from primary data center for synchronous replication	30-50 miles
Average number of spam e-mails delivered every 30 days	>3.65 billion
Number of e-mails sent daily in 2006 (est.)	>35,000,000,000 (billion)
Percentage of customers retaining e-mail	9%

archives over 7 years	
The size of WW wireless calls in PB	2,300
Percentage of all e-mail traffic that is unwanted	~90%
Percent of companies citing employees as the most likely source of hacking	77%
Percentage of US adults with more than 200GB of storage capacity	10% (approximately 28 million)
Percentage of digital data stored on removable media (primarily magnetic tape)	~75%
Percentage of digital data stored on mobile (portable) technologies	50-60%
Number of new (1st round) data storage companies funded in 2000	92
Number of new (1st round) data storage companies funded in 2006	2
Total storage companies acquired in 2006	100
Number of new small businesses created in the US in 2005	550,000
Average revenues of InformationWeek 500 companies in 2007	$12.03 B (was $9.0B in 2004 and $12.4B in 2001)
Average IT budget of Information Week 500 as a percentage of revenue in 2007	2.76% (was 3.88% in 2001)
Average digital archive content in a Fortune 1000 company in 2007	>250TB (52% cagr)
Percentage of IT businesses managing more than 10TB of disk data	39%
Average percentage of IT budget in the US spent on IT salaries and benefits	32%
Average percentage of IT budget in the US spent on compliance	5%
Average percentage of IT budget in the US spent on hardware in 2006	16.3 % (projected to be 13.9% in 2011)

Average percentage of IT budget in the US spent on services in 2006	23.3% (projected to be 29.7% in 2011)
Average percentage of IT budget in the US spent on personnel and salaries in 2006	19.8% (projected to be 12.7% in 2011 - only if health-care costs are not covered)
Average percentage of IT budget in the US spent on support and maintenance in 2006	4.3% (projected to be 4.8% in 2011 - includes energy)
Number of mid-market firms in the US in 2005 (100-999 employees)	93,876
Percentage of all IT jobs in businesses with fewer than 99 employees	72%
Percentage of WW IT workers considered mobile in 2007	20%
Percentage of CIOs reporting to CEO in 2006	66% (was 56% in 2005, 17% to CFOs)
Projected size of India's IT services industry in 2010	$60 B
Projected size of China's IT services industry in 2006	$8.9 B (est. cagr. 18.9%)
Percentage of businesses who take backup tapes offsite daily, weekly, and monthly?	Daily – 56%. Weekly -32%. Monthly - 4%.

Horison Information Strategies http://www.horison.com

Appendix C – References

[1] Cloud Computing Value Chains:Understanding Businesses and Value Creation in the Cloud, Ashraf Bany Mohammed, Jorn Altmann and Junseok Hwang, Dec 2009

[2] Cloud Data Management Interface Specification, Version 0.80, Jan 2009

[3] ObjectSecurity, Plug & Play Tech Center, 530 University Ave, Palo Alto, CA 94301, USA

[4] St John's Innovation Centre, Cowley Road, Cambridge CB4 0WS, UK, ulrich.lang@objectsecurity.com

[5] The Institute of Internal Auditors: Managing and Auditing Privacy Risks,

[6] http://www.theiia.org/download.cfm?file=33917

[7] Casassa Mont, M.: Dealing with Privacy Obligations, Important Aspects and Technical Approaches. TrustBus 2004 (2004)

[8] Casassa Mont, M., Pearson S., Bramhall, P.: Towards Accountable Management of Identity and Privacy: Sticky Policies and Enforceable Tracing Services. In: DEXA 2003, pp. 377-

[9] 382. IEEE Computer Society (2003)

[10] IBM, The Enterprise Privacy Authorization Language (EPAL), EPAL specification, v1.2,

[11] http://www.zurich.ibm.com/security/enterprise-privacy/epal/ (2004)

[12] OASIS, eXtensible Access Control Markup Language (XACML), http://www.oasisopen.

[13] org/committees/tc_home.php?wg_abbrev=xacml29. Cranor, L.: Web Privacy with

[14] P3P, O'Reilly & Associates (2002) ISBN 0-59600-371-4

[15] Damianou, N., Dulay, N., Lupu, E., Sloman, M.: The Ponder Policy Specification

[16] Language, http://wwwdse.doc.ic.ac.uk/research/policies/index.shtml (2001)

[17] Tang, Q.: On Using Encryption Techniques to Enhance Sticky Policies Enforcement,

[18] Technical Report TR-CTIT-08-64, Centre for Telematics and Information Technology, University of Twente, Enschede (2008)

[19] Pöhls, H. C.: Verifiable and Revocable Expression of Consent to Processing of Aggregated Personal Data, ICICS 2008 (2008)

[20] Schunter, M., Waidner, M.: Simplified privacy controls for aggregated services – suspend and resume of personal data, Privacy Enhancing Technologies, 7th International Symposium, pp. 218–232. Springer (2007)

[21] Clarke, I. and Miller, S.G.: Protecting Free Expression Online with Freenet, IEEE Computing (2002)

[22] Rhea, Eaton, Geels, Weatherspoon, Zhao, and Kubiatowicz: Pond: the OceanStore

[23] Prototype. In: FAST ’03 (2003)

[24] Huang, C.D. and Goo, J.: Rescuing IT Outsourcing- Strategic Use of Service Level

[25] Agreements, IT Pro (2009)

[26] Yearworth, M., Monahan, B. and Pym, D.: Predictive Modelling for Security Operations

[27] Economics, HPL-2006-125 (2006)

[28] EU FP7 Network of Excellence: http://www.coregrid.net/ (2009)

[29] EU FP7 Project SLA Aware Infrastructure: http://sla-at-soi.eu/ (2009)

[30] EnCoRe: Ensuring Consent and Revocation project, http://www.encore-project.info (2008)

Author's Biography

Paul Brant is a Senior Technology Consultant at EMC in the Global Technology Solutions Group located in New York City. He has over 29 years experience in semiconductor VLSI design, board level hardware and software design, and IT solutions in various roles, including engineering, marketing, and technical sales. He also holds a number of patents in the data communication and semiconductor fields. Paul has a Bachelor (BSEE) and Masters Degree (MSEE) in Electrical Engineering from New York University (NYU), located in downtown Manhattan as well as a Masters in Business Administration (MBA), from Dowling College, located in Suffolk County, Long Island, NY. In his spare time, he enjoys his family of five, bicycling, and other various endurance sports.

Index

Al Gore 26
Amazon..... 18, 70, 77, 92, 202, 204, 215
Amdahl's Law..... 41, 42, 43, 45, 174
AMQP 229
Appliance ... 25, 35, 48, 50, 53, 56, 175, 178, 179, 180, 207
Applianceization..... 26, 35, 47, 48, 49, 52
archive . 25, 28, 94, 95, 96, 97, 98, 107, 108, 110, 114, 128, 129, 138, 139, 140, 145, 146, 161, 182, 184, 188, 191, 192, 194, 195, 196, 197, 238
ATA..... 119, 145
AWS 77, 219, 220, 229
BGP 220
Big data..... 48
Big Data..... 48, 160, 162, 163
cache coherence..... 59, 60, 62, 63
CAD 19
CASPAR 74
CDMI 68, 69, 115, 116, 117
CEx..... 217, 218, 219
CIFS 95, 112, 186, 205, 207
CIO 24, 77, 160
Cisco..... 52, 54, 55
CLARiiON 52, 55, 119, 155
Cloud Computing 239
cloud of clouds 214
Cloud Storage 114, 115, 204, 205, 206, 208, 212, 213
Clustered Systems 37
compliance..... 13, 14, 17, 69, 79, 80, 87, 94, 118, 125, 129, 161, 162, 238
Containerization 74
CRUD 68, 115
Dark Ages 17, 131
data portability..... 118
Data Quality 167
Data security 29
data transference 57, 64
data warehouse..... 37, 39, 51, 82, 162, 163, 164, 165, 166, 167, 169, 170, 171, 172, 173, 177, 178, 179, 180
database applications..... 39
DATAllegro..... 51, 179, 180
DBMS 51, 163, 164, 168, 171, 172, 176, 177, 179
deadlock..... 63
Digital Crisis 13, 230
distributed caches 59
DNS 75, 220, 226, 227
Drive Optimization..... 157, 158, 159
DSM..... 60, 61
DSS 40, 176, 178
DWA 172, 178, 180
e-Bay 18
ECM..... 28
EDW 168, 171, 172, 173
EFD..... 119, 155, 156
EMC..... 1, 12, 16, 24, 41, 49, 52, 53, 55, 57, 63, 64, 95, 96, 97, 109, 110, 119, 120, 122, 148, 149, 150, 151, 153, 154, 156, 172, 174, 180, 184, 206, 241
EMCCA..... 16
ETL 168, 174

Exadata 51, 172, 180
F5 51
Facebook 18, 27, 28, 71, 124, 204, 228
Facebook Debacle 27, 35
firewalls 48, 79, 231
Google 18, 198, 206, 209, 215
GRC 82
Greenplum 41, 49, 172, 174, 175, 179, 180
grid 26, 215
Gustafson's Law 41, 44, 45, 174, 175
HIPAA 17, 134, 192
HP 172, 180, 207
IAAS 68
IBM 51, 172, 180, 229
I-Cards 72, 73
IDC 15, 22, 23
INFORM 196, 197
Information Object 99
Intercloud 26, 27, 75, 205, 207, 212, 214, 215, 216, 219, 220, 221, 222, 223, 225, 226, 227, 228
Ionix 55
IPv4 ..... 224, 225
IT241
JHOVE 110
Kerberos 226, 227
Kognitio 172, 179, 180
Latency 39
Latent faults 141
LDAP 226, 227
Library of Congress 21, 110, 111
LISP 225, 228
livelock 63
LOTAR 102
LUN 58, 115, 119, 205
MapReduce 170, 174, 215
Massively Parallel Processor Systems 37
MDM 167, 168
MDS 52, 55
MediaSmart 207
Migration 104, 108, 183, 193, 197, 198, 199, 200
millennium 32
Mobility 75, 76, 77, 223, 224, 225
Moore's law 49
MPP 37, 38, 39, 40, 41, 42, 43, 45, 46, 51, 52, 162, 164, 169, 174, 175, 176, 177, 178
MTTDL 141, 143, 144, 146
National Archives 21, 138
NDIIPP 110
Netezza 51, 172, 178, 180
network computing 26
next generation data center 13, 14, 16, 21, 24, 29, 82, 88, 119, 150, 154, 163, 166, 167, 182, 220, 230, 236
NFS 95, 112, 114, 186, 205
NGDC 34, 37, 41, 42, 45, 46, 51, 55, 57, 66, 75, 76, 77, 78, 82, 89, 96, 102, 106, 109, 110, 111, 118, 122, 124, 129, 137, 140, 141, 160, 163, 167, 169, 172, 174, 176, 182, 201, 206, 214, 219, 225, 228
OAIS 28, 74, 97, 98, 99, 101, 104, 106, 107, 108, 109, 110, 184
OCCI 66, 67, 68, 69, 116, 117
OCLC 196
OLAP 165, 170
OLTP 26, 47

Open Cloud Computing Interface 66
Open Identity Exchange 73
Oracle 51, 155, 169, 172, 180
OSPF 220
PACE 154
Patriot Act 84
PCFE 151, 153
PDS 28, 29, 32, 70, 71, 90, 92, 93
PeDS 28, 29
PhotoPoint 21
preservation ... 18, 21, 28, 74, 94, 95, 96, 97, 98, 99, 100, 101, 102, 103, 104, 105, 106, 108, 110, 111, 112, 113, 127, 129, 130, 131, 133, 134, 136, 137, 140, 147, 149, 150, 182, 184, 187, 189, 190, 191, 192, 193, 194, 196, 197, 198, 199, 200, 208
Preservation Data Stores 28
QoS 114, 211, 216
RAID 133, 140, 142, 148, 152, 159
Raw Capacity 159
RDBMS 51, 52
Relative Service Time 158
reliability 35, 51, 56, 117, 132, 133, 140, 142, 143, 145, 146, 147, 148, 150, 201, 229
Replication 104, 139, 148
resiliency 150, 232
REST 65, 114
SAAS 82, 94
SAN 52, 63, 75
Sarbanes-Oxley 17, 134
Scalable Architectures 37
scale-up 37
Scatter Gather 45
Scatter/Gather 45, 246
scrubbing 145
SIRF 101, 102, 103, 105, 106, 109, 110, 111, 112, 127, 131, 188, 190, 191, 192, 193
SMP ... 37, 38, 39, 40, 41, 51, 164, 176, 177, 178
SMTP 229
Snoopy protocols 60
SOA .. 51, 163, 173, 193, 194, 195, 197, 198
spam 24, 31, 124, 237
SQL 170, 171, 172, 174, 180, 209
SSO 78
Storage Ecosystem 13, 35, 106, 114
StoreCenter 207
Sybase 172, 180
Symmetric Multiprocessing 37
Symmetrix 52, 55, 119, 154, 155, 156
TIBCO 50, 229
Tier Advisor 154, 155, 156, 157, 158, 159
tiering 119, 120, 122, 154
TPA 184, 186, 187, 188, 189, 190, 191, 192, 193
TRAC 74
transformation 14, 16, 18, 19, 24, 25, 32, 45, 46, 64, 106, 123, 171, 175, 184, 212, 220, 230
Transformation 1, 96, 104, 184
UIM 54
Vblock 52, 53, 54, 55
VDI 78, 79
Vinton Cerf 26
VM 75, 76, 77, 211, 224, 225, 235
VMAX 52, 63

VmWare 52, 56, 75, 207
VPLEX 57, 63, 64
vSphere 52, 207
WAN 51, 163
Web security 48
WebSphere 51
WOV 141, 143, 144
XAM 102, 109, 110, 112, 186, 208
XFDU 102
XML 51, 55, 77, 102, 209, 227
XMPP 227, 229
XRI 73
XStreams 110
XtremeData 50
XUID 110
ZooKeeper 170