Table of Contents

List of Figures

Open Cloud Architecture—Extending an Open Narrative

Motivation: Operational Readiness

Clouds are virtual data centers. Thus, IT concern is for what can be efficiently processed through operational sourcing, whether in-sourced or out-sourced. Currently, most external cloud providers are proprietary at the end of the day, Linux and Windows Intel based servers notwithstanding.

Internally, infrastructure sourcing for many Enterprises has been against the "old" model of managed services or the more retro facilities management. Sourcing agents responding to new resource demands from dynamic business needs under these models is still in weeks to months.

In November 2008, we began seriously to consider the issues around the readiness of cloud IT Operational Sourcing. The picture below was the starting point to guide seven Proofs of Concept to answer the question, "Are Clouds ready for Enterprise prime time?"

Figure 1: Readiness Questions for Cloud IT Operational Sourcing

The answer is yes and no, but there is a critical mass of capability NOW to deploy in operations in 2011 after doing pilots in 2010 and prototypes in 2009H2.

We're just crossing the Geoffrey Moore Chasm in 2009, and by 2010 enterprises will realize the benefits of cloud computing. However before deploying into the cloud, designing the form of the cloud should be based on the functions it will perform.

Start by establishing a set of guiding principles will help to drive the solution architecture for your cloud design. The guiding principles that we used when designing the solutions described in this paper are:

1. Avoid proprietary APIs for portability for cloud to cloud federation.
2. Wrap edge SOAP/REST APIs with a security and privacy layer of abstraction.
3. Route requests to the data to comply with regional regulatory constraints.
4. Transform credentials into SAML tokens for cloud2cloud transaction routing.
5. Loosely couple business rules and security and privacy policies from the code.
6. Implement Policy Enforcement Points for Fine Grained Authorization at the edge of the cloud to secure the front doors to the cloud.
7. Implement Policy Enforcement Points for Course Grained Authorization on each cloud resource to close the back doors in the cloud.
8. Use a virtual storage namespace in the cloud to simplify access to input and output data.
9. Use distributed in-memory databases on the same resources as the compute resources in the cloud for optimal performance.
10. For optimal efficiency, suspend cloud resources when idle to avoid unnecessary charges.
11. Reduce the risk of using public cloud resources by signing all software and verifying that the build is authentic prior to using the newly instantiated resource.
12. Design and deploy custom OS and builds, uninstall unnecessary software and close open ports.
13. Integrate cloud service level event and security incident alerts with a centralized operations console using WSDM to automate event correlation.
14. As cloud resources are added and removed from an application pool update a federated CMDB to maintain an up to date view of the cloud formations.

Current economic forecast: 2009 will finish sowing the seeds of recovery. 2010 will be a year of slow global and local growth, at best. 2011 will usher in the beginnings of an accelerating global recovery.

The next section frames the question, "Are Clouds ready for Enterprise prime time?" by shaping and elaborating this question into definable and testable issues.

Multi-Tenant, Multi-Landlord Platform Evaluation 2008-9 Questions to Answer

Question 1: How secure is it?

A secure cloud provides Zones and Venues of Privacy and Trust. Issue areas include:

- Identity Management
  Who is the Authority? How do we authenticate? How many factors? Under which scenarios?

- Access Policy Enforcement
  How do we protect specific resources from unwanted use? What methods of policy definition and evaluation?

- Audit Ability
  How do we assure everything is traceable? What sovereign laws are in force?

- Integrity of communications and data storage
  How do we protect and keep private all business interactions and records?

- Non-repudiations of actions
  How do we choose methods of affirming business contracts? What are the standards for surety of action?

Question 2: How easy is it to manage?

Managing data is the necessary and most difficult management challenge. Issue areas for ease of data management include:

- Data Persistence Transparency
  What are the mechanisms of distribution and storage to hide data from applications?

- Services are delivered data on an as-needed basis
  Are application views managed by the infrastructure?

- All external references are in a single namespace
  Are the naming and identification of resources and communication points the same everywhere, modulo access rights?

- Data are specifically and generally Locatable
  Is access to data location transparent? Can data be confined to specific geography and/or sovereign jurisdiction?

  - Accommodate sovereign regulations
    What are the relevant privacy and transaction rules for data?

  - Replication
    What mechanisms are used to replicate data amongst virtual data store nodes?

  - Caching
    What mechanisms and strategies for caching exist to locate data near its consumption?

  - Segregation
    How are data contained and shielded from the risk of accidental disclosure?

Question 3: How hard is it?

The major impedance to deploying processes in separating concerns and identifying interaction protocols. Issue areas include:

- Process Segregation
  What are the requirements and mechanisms of separation of duties to reduce the risk of process compromise?

  - SOX, HIPAA, EU Privacy, Industry Regulatory Oversight
    What needs to be done to satisfy Sarbanes Oxley, the Health Insurance Portability and Accountability Act, European Union and Industry best practices on separation of the roles within business process work flows?

- Standard Interaction Messaging Protocols
  What are the General, Industry and Proprietary Protocols used to exchange information, context state and data within and among clouds?

- Service Invocation
  What are the topics of interest under which service request can be made? How are Services advertised and located? How are Service Level Agreements negotiated?

- Event Dispatch
  What are the business, application and infrastructure events of interest for which state is shared amongst decoupled processes? How are processes instrumented to report these events?

- Data Distribution
  What are the mechanisms for streaming data within the virtual data store and amongst decoupled application processes running on infrastructure nodes?

Question 4: How is it monitored?

What is involved in governing the full life cycle of cloud based assets? Issue areas include:

- Specification
  What are the processes of writing application and infrastructure development requirements?

- Realization
  How are services composed and conditioned to provide secure, compliant business services?

- Deployment
  What are the environments and platforms within which configurations of resources are provisioned and exposed for use?

- Instrumentation
  What are the mechanisms of capture, storage, reporting and control of key business, application and infrastructure environment events? What are the Business and Technical Metrics?

  - Infrastructure Environment
    What are the system components and their respective emitted events of interest?

  - Application Work Flow
    What are the process tasks that comprise applications with their respective emitted events of interest?

- Business Process
  What are the key business capabilities (suppliers of significant revenues) and their respective emitted events of interest?

- Testing/QA
  What are the methods of promotion of functionality from unit development to system integration to staging area to production?

- Production
  What are the key operational indices and performance metrics for operational facilities and their respective network and security operation centers? How are incident cases managed for both operational interruptions and security events?

- Retirement
  How are assets measured in terms of return? What are the processes for retiring low performing assets?

Question 5: How safe is it?

What is the ability to detect damage from insiders and outsiders, safety being the other side of the security coin? Issue areas include:

- Misfeasance
  What controls and reporting mechanisms are in place to detect errors and inadvertent failure to follow standard operating procedures?

- Malfeasance
  What controls and reporting mechanisms are in place to detect willful, malicious subversion of standard operating procedures?

- Vandalism
  What are the mechanisms and methods to detect and repel attempts to interrupt operations?

- Piracy
  What are the methods of fraud detection, reporting, intervention and remediation?

The Case for Multi-Landlord Focus

The heavy hype around clouds is its relatively low-cost, multi-tenant aspects. The most troubling question is, however, the security models within both external and internal manifestations of clouds.

Moreover, cloud/data center providers (Amazon Web Services, Rackspace, GoGrid, Google AppEngine, Microsoft Azure) worry mostly about accommodating multiple

consumers, each shielded from the others within their properties, both real and virtual. Not surprising, providers desire some level of stickiness (read lock-in) of clients. It was decided immediately, that a dynamically competitive anything-as-a-Service model is most advantageous for consumers.

Multi-landlord was added to the concept. So the question of what are the design patterns with respect to Security issues was asked as illustrated by the classis ER diagram below:

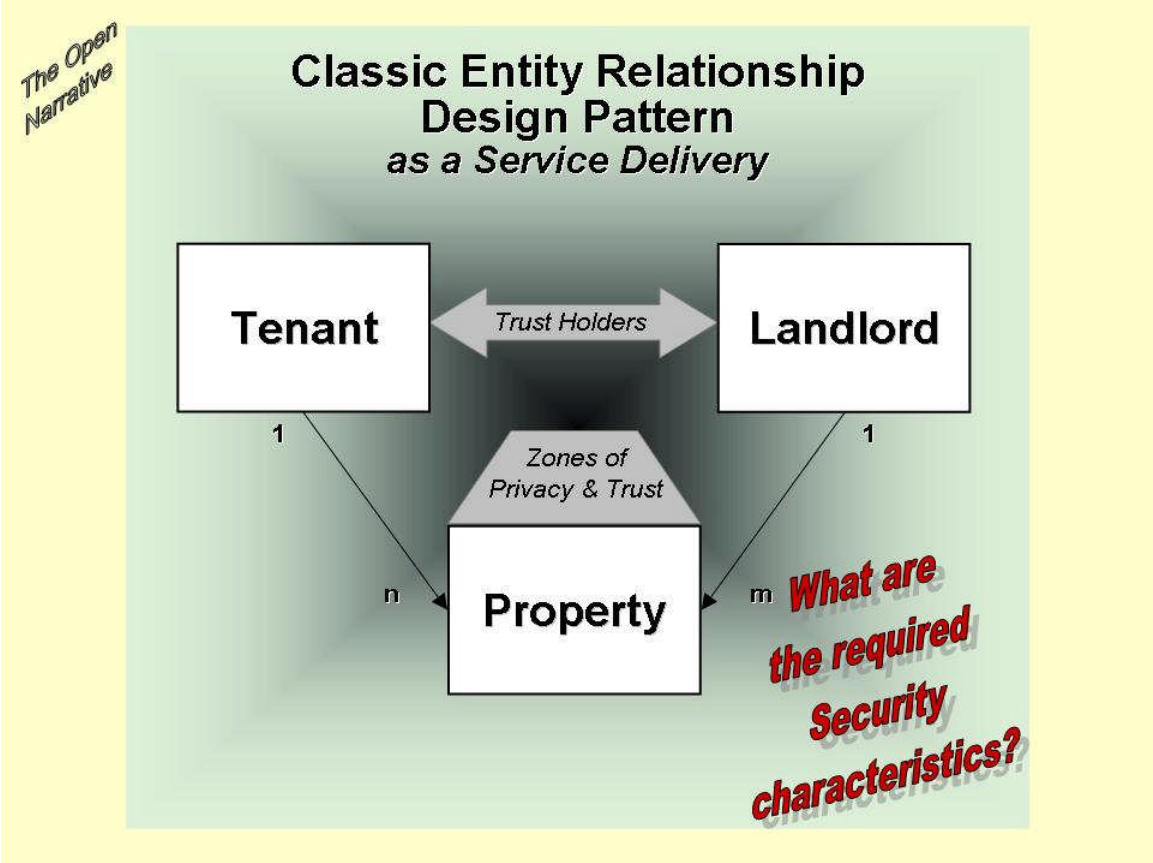

Figure 2: Central Enterprise View of Cloud as a Service Design Pattern

"As a Service" (*aaS) delivery is the operational model embraced by cloud providers. Otherwise, we revert to the old ways of managed services or facilities management. AaS is, for the most part, a recycling and update to the time-sharing models so popular in the '60s and '70s before the commercial advent of workstations and PCs in the '80s.

This dynamic multi-landlord sourcing for capabilities allows ready switching of providers for most advantageous resource pricing in addition to the elastic quality of resource provisioning. It places a requirement for being able to federate domains of trust amongst resource providers.

For security, the above Figure 2 captures the multi-tenant/multi-landlord relationship with respect to the locales of operational facilities, namely, the Zones of Privacy and

Trust. The main issues are around engineering the establishment of Trust Domains amongst Trust Holders, Tenants and Landlords, within the Zones of Privacy & Trust.

Summary of Seven Proofs of Concepts and Lessons Learned

Proofs of concept were carried out in 6-week or less cycles. From December 2008-August 2009, we managed seven substantive proofs of concept. This resulted in much knowledge gained and information on what works and what does not at this moment in time.

The questions in each PoC are elaborated below with summary of efforts and lessons learned.

1. Can we build a prototype Monte Carlo Simulation calculation engine grid quickly and deploy within an internal or external cloud?

Description

We conducted a Proof of Concept for integration of Enterprise Internal and External System Clouds for the purposes of operating annuities hedging applications. External platforms to study include Salesforce.com, Amazon.com (Elastic Cloud Computing, Simple Storage System) and/or Microsoft Azure. Settling in on Amazon Web Services, we used an existing body of Windows based C++ code from current hedging applications to demonstrate the low operational cost structure feasibility of using such external services. Additionally, we developed a prototype Java grid application on a Linux image to do performance testing of speed up in dong large parallel calculations.

Effort

Duration/effort level: 4 weeks/2 person months
Vision Coordination: 15 days
Microsoft C++ Conversion: 15 days
Linux Java Grid Prototype: 10 days

Lessons Learned

- **Getting started:** a credit card and a plan
  - Stood up development environment in less than an hour for less than $10
  - Support is good from Amazon
- **Infrastructure cost:** incidental
  - Choice of images to particularize is huge

- Easy to share private images

- **Amazon Cloud is easy for the knowledgeable and skilled**
  - Not always the case in other infrastructure development centers
- Can use Salesforce.com Cloud as front-end for Amazon Cloud
  - Able to invoke calc engine ai Amazon Cloud and asynchronously return results

2. Can we build a security domain with policy enforcement?

Description

We devised a Security Architecture for Data Centers that anticipates establishing domains of Trust amongst multiple external systems organized as Clouds. The work built on and deepened the security diagrams delivered as part of first PoC. We showed scaling Sonoa Policy Suites controlling invocation of calculations in the Amazon cloud.

Effort

Duration/effort level: 6 weeks/1.5 person months
Vision Coordination: 10 days
Write Policy Suite for Sonoa-based DMZ within Amazon Cloud: 15 days
Assistance from Sonoa, Amazon Web Services: 5 days

Lessons Learned

- **Policy enforcement infrastructure:** low-cost option is available
  - DMZ in a cloud operates as usual

Policies are easy to write and apply

- **Monitoring for policy control panel:** effective
  - Sonoa ServiceNet control panel provides useful functionality
  - Needs to integrate across heterogeneous clouds

3. How do we store data in the cloud using a dbms and unstructured files-Part 1: Build Terabyte Test Data Base?

Description

Build a Terabyte Oracle Release 10g 2.0.4 data base of Obfuscated Test Corporate Data in Amazon ec2/EBS, a fully loaded Oracle Release 10g 2.0.4 data base instance. Secured a depersonalize business data set of 25MB that was replicated into a Terabyte.

Effort

Duration/effort level: 6 weeks/1 person months
Vision Coordination: 5 days
Oracle DBA Tasks: 12 days
Vendor Support: 3 days

Lessons Learned

- **Setup:** straightforward, but detailed
  - Using Oracle pre-bundled images
  - Only ASM is still complicated because it needs a known IP address
  - Contains automated features for first time db build and mount

4. How do we store data in the cloud using a dbms and unstructured files-Part 2: Benchmark Oracle Cloud Configuraton and Queries?

Description

Using Terabyte Oracle Release 10g 2.0.4 data base of Obfuscated (depersonalized) Test Corporate Data in Amazon ec2/S3/EBS built in Part 1, benchmark a set of illustrative queries to gauge performance characteristics. Ran queries and produced benchmarks.

Effort

Duration/effort level: 6 weeks/1.25 person months
Vision Coordination: 5 days
Oracle DBA Tasks: 15 days
Vendor Support: 5 days

Lessons Learned

- **Performance:** scales well with added instances and suitable data partitioning
- **Data Sharing:** not supported on AWS
  - Oracle RAC not supported
  - Data is stored persistently in EBS volumes, which cannot be shared between compute instances
- **Replication:** cumbersome process
  - Data has to go through S3 to replicate to new volumes
  - S3 takes whole data file at a time – cannot update changed data only
  - Transfers between S3 and EBS are fast, but large datasets are still a problem

5. How do we store data in the cloud using a dbms and unstructured files-Part 3: Optimize Oracle Cloud Parallel Query Infrastructure?

Description

Using results of Part 2, we created a segmented, partitioned version of the test data base.

Effort

Duration/effort level: 4 weeks/1 person month
Vision Coordination: 5 days
Oracle DBA Tasks: 15 days
Vendor Support: 5 days

Lessons Learned

- **Dynamic scale-out:** difficult to automate
  - Six volumes per instance are recommended by Amazon for performance
  - Scale-out is by adding instances, but data is replicated by volumes
  - Data in tables must have partitions aligned to volumes if you want to divide it across instances
  - ASM hides the volumes for load balancing and easier management of data growth – but then the partitions can't be aligned to volumes

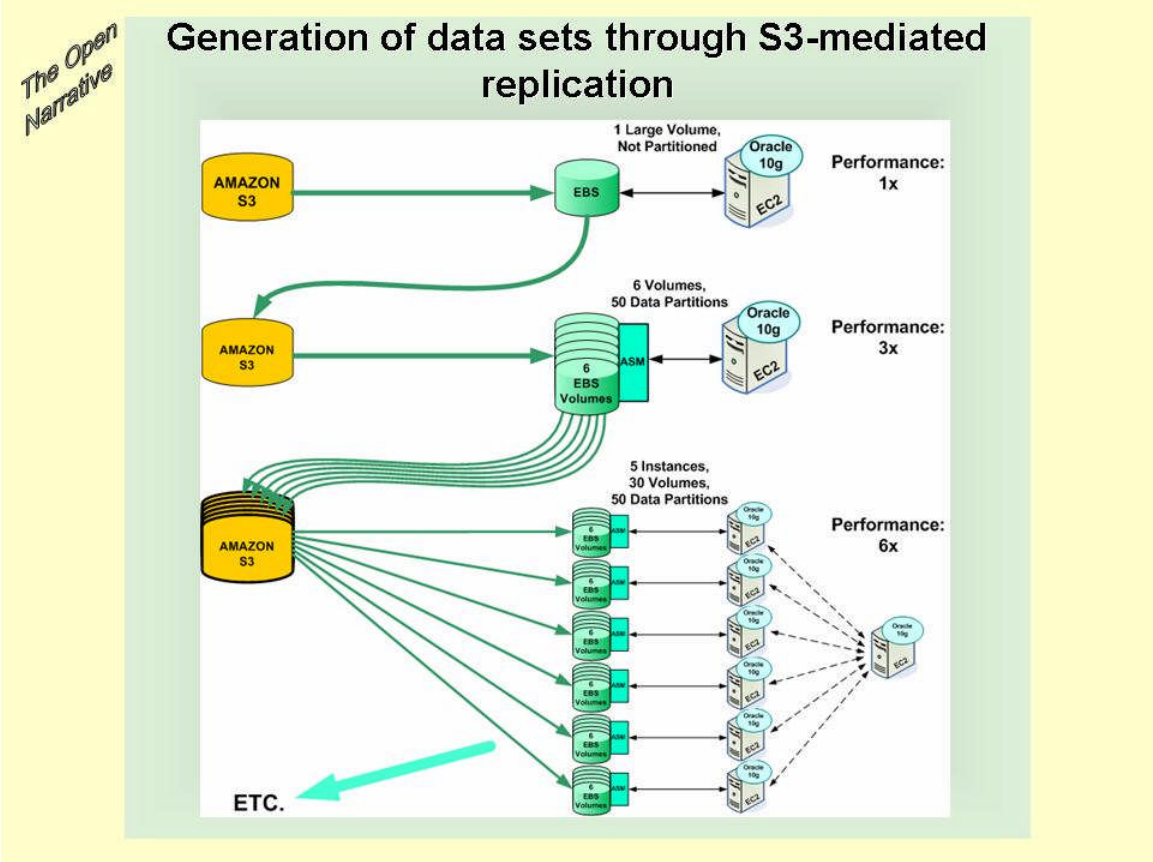

Figure 3: Oracle Performance Enhancement in the Amazon Cloud PoC Structures (Courtesy Enterprise Architecture, ING Americas)

6. How difficult is it to port from one cloud to another?

Description

Using the code, configurations and images developed in the first Proof of Concept, we ported the prototype grid calc engine from Amazon to Rackspace. We attempted to engage with IBM before Rackspace. Additionally we set up three prototype clouds reflecting an End to End Business Architecture.

Effort

Duration/effort level: 2 weeks/0.75 person months
Vision Coordination: 8 days
Porting effort: 5 days
Rackspace Support: 2 days

Lessons Learned

- **Rackspace is an effective choice for doing clouds:** slightly behind Amazon

- Useful and easy provisioning, auto-provisioning api
- Amazon has more higher level convenience capabilities like Hadoop service
- Rackspace catching up quickly
- Rackspace has variable provisioning based on availability of idle resources
- Rackspace will provision more than one core to an image, unlike Amazon
- Pricing differentials still unclear, requires n-depth analysis

- **IBM not yet ready for true cloud:** computing on-demand is too manually oriented, research cloud is not open to public

  - Have well articulated marketing strategy
  - Current offerings still have long provisioning lead times
  - Will be there as a fast follower in 2010

7. Can we build and control two interacting external clouds with policy federation-Front End and Offer Clouds?

Description

To do Business Alignment in the wool, we structured the policy federation question in terms of the primary usage of the clouded resources: Front End—Channel Delivery, Offer—Product and Service Construction. A many-many possibility of interaction mediated control by Enterprise Cloud. W built out two different clouds with our configuration.

The Figures below show:

- A demonstration of federated policy enforcement across hybrid clouds, and,

- An information control architecture for hybrid clouds, as well.

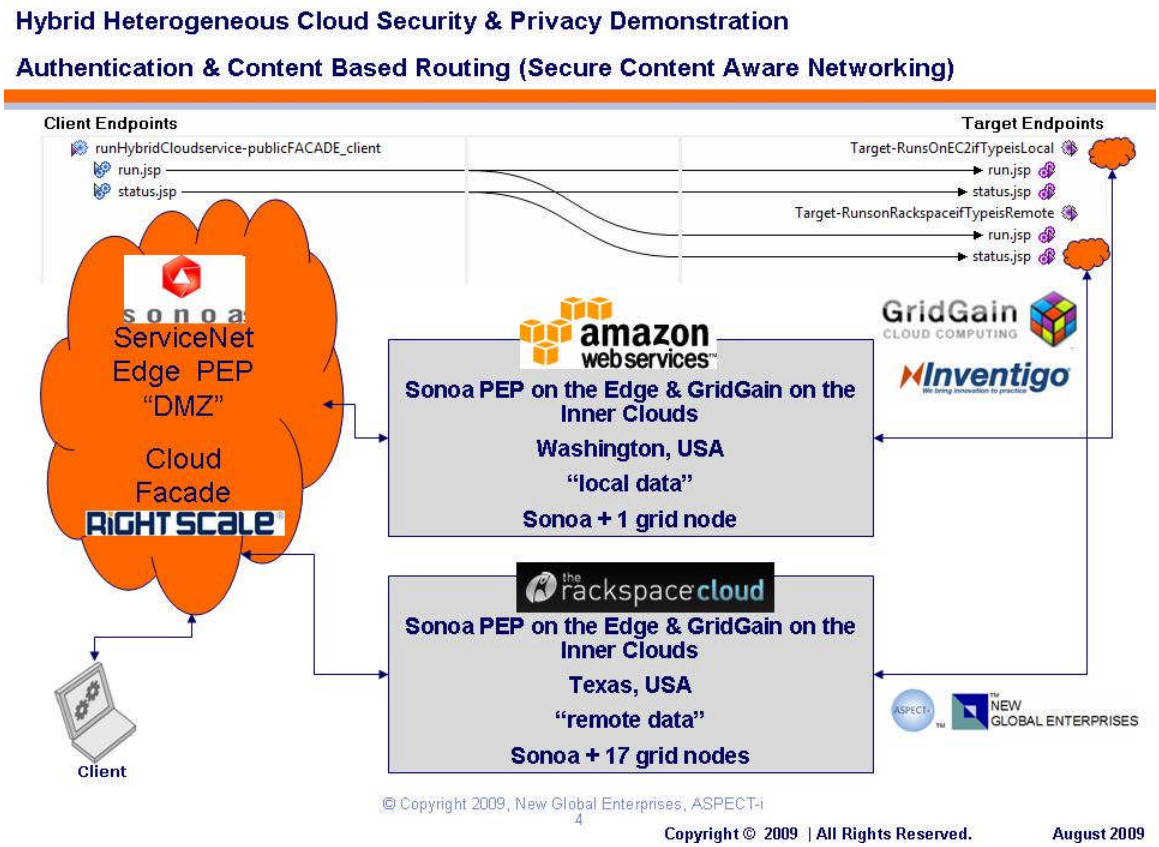

Figure 4: The Demonstration of Hybrid Cloud Security Federation

The Demo

Figure 4 depicts the two XACML policy shielded clouds, the "Inner" clouds, Amazon Web Services and Rackspace, in which two identical REST services exist:

- run.jsp
  The prototype calc engine service that runs on a grid

- status.jsp
  The results delivery service

The difference is that the data (a year of daily price ticks for 100 different symbols), about 800MB for the calc service are local to Amazon and remote to Rackspace.

The "DMZ" cloud has the service facades which are invoked by an authenticated Client. The purpose of Sonoa ServiceNet/RightScale is to provide scalability of service request mediation to either Service Cloud.

The Amazon Cloud contains the Calc Engine Service running on a 1-node grid and the Rackspace Cloud contains the Calc Engine Service running on a 16-node grid. The GridGain/fastXML images which compose the grids are exactly the same. The prototype

calc engine discovers other like images and forms the grid, in this case, a 1-node grid by default in the Amazon Cloud and a 16-node grid in the Rackspace Cloud.

As another note, the images n each Cloud are exactly the same, the porting being done in less than a day. The major time was in transferring the image which is about 1GB.

Each Cloud Calc service policy shield "knows" what runs in its respective Inner Cloud and will dispatch a request for the other cloud service seamlessly to the Client Requestor.

The prototype runs the Calc Service on the year of daily price ticks to produce Open, High, Low, Average, Close and Next Day Prediction (a linear interpolation of the previous Closing prices). In the case of the Amazon 1-node calc engine, the timing is about 60 seconds. In the case of the Rackspace 16-node calc engine, the timing is about 6 seconds.

The Hybrid Cloud Demo Context: Extranet Cloud Trust Creation

Figure 5 depicts the logical design for Information Control across a distributed data cloud. It handles identity federation via SAML 2.0 tokens to exchange among the federated clouds and their respective security policy shields.

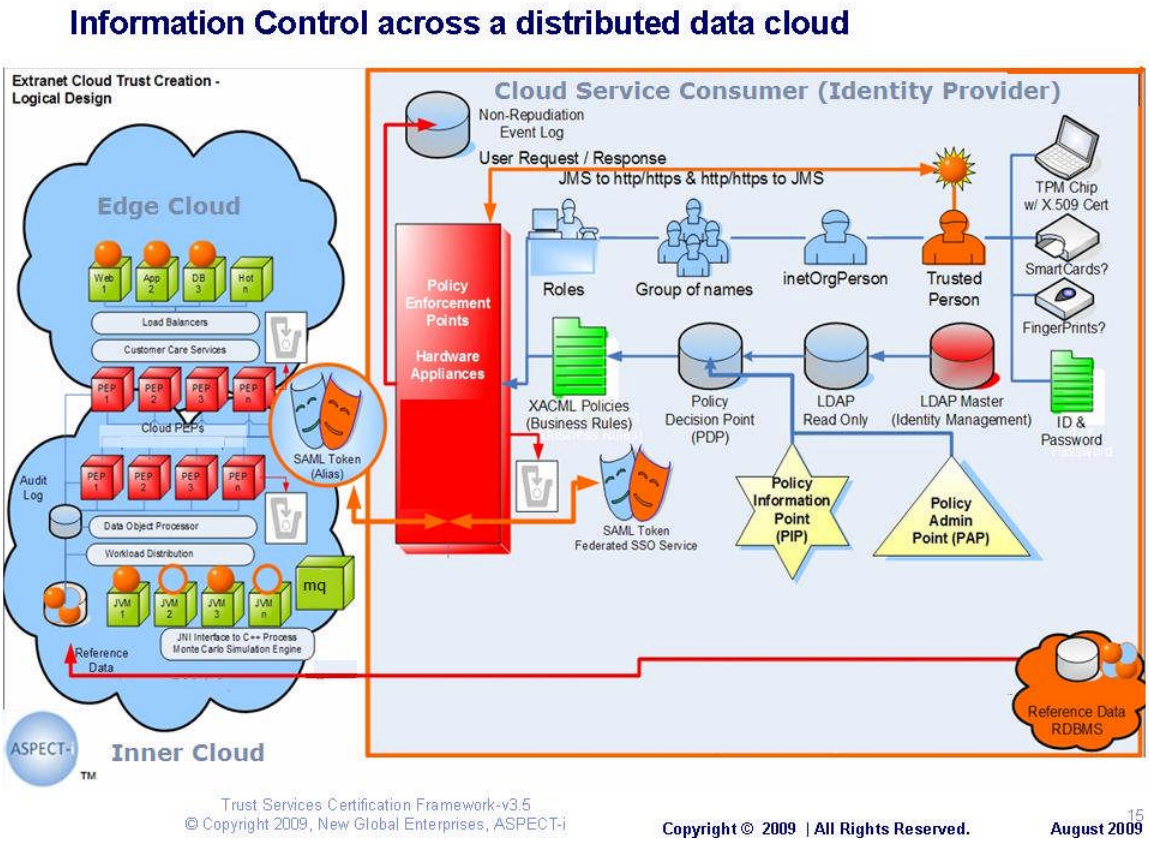

Figure 5: Information Control in Hybrid Clouds

Effort

Duration/effort level: 3 weeks/2 person months
Vision Coordination: 15 days
System Configuration Assurance: 20 days
Rackspace, Amazon Vendor Assistance: 5 days

Lessons Learned

- **Each app port has its configuration:** the Operational Master is ignition
  - Finally, Java promise "use everywhere" defined and made usable
  - Hacks are easy, engineered is hard, so tools, tools, tools.

- **Amazon and Rackspace are viable contenders:** GoGid and others following quickly

    - IBM, eBay, HP, Google, Yahoo are followers within two years
    - The basic capital cost of being a fully enabled enterprise is less than $100K and can start for less than $10K

Next Steps

The next steps are to move on to Prototyping as the Proofs of Concept have borne sufficient information on degree of readiness.

2009H2: Prototyping

The year-end deliverables are twofold:

1. A Cloud Implementation Strategy with Two-year Roadmap and Agenda

2. Part II of the Enterprise Security Framework to cover Legacy, Virtualized Platforms and Cloud Environments as business aligned fitness-for-business-purpose operational facilities.

A 2010 Pilot Implementation Plan will derive from both these deliverable and carry into the following year.

2010: Pilots

The strategy is to pick system remediation efforts that are usually very large and use the Proven Prototyped

2011: Large Production Roll-Out

By 2011, all the big players will be in the game. Thus, the risk can be managed at a price.