This IEEE Cloud Computing tutorial has been developed by Cloud Strategy Partners, LLC. Cloud Strategy Partners, LLC is an expert consultancy firm that specializes in Technology and Strategy relating to Cloud Computing.
In this tutorial, we will discuss motivation, background and Cloud interoperability classification Multi-cloud and Intercloud. We will also review IEEE P2302/D0.2 Draft Standard for Intercloud Interoperability and Federation (SIIF) and Main architecture components. Next we will discuss Namespace, topology and trust management and Semantic resource description. We will finish with a discussion of Intercloud infrastructure federation process and Workload and Storage federation.
Outline
In this Lesson we will cover

- Motivation, Background
- Cloud interoperability classification
  - Multi-cloud and Intercloud
- IEEE P2302/D0.2 Draft Standard for Intercloud Interoperability and Federation (SIIF)
- Main architecture components
- Namespace, topology and trust management
- Semantic resource description
- Intercloud infrastructure federation process
- Workload and Storage federation
- Summary and take away
- References

Motivation – Evolve to Interoperability, like the Internet Did

Many may not remember or know this history, but there was a time where the Internet was not really used by the public, it was still a research project. For over 10 years, the on-line experience for people was with an on-line service such as Compuserve, Prodigy, or America On Line (AOL). These grew out of earlier technologies called Bulletin Board Services (BBS).

These on-line services each had their own datacenters, to which one would use a dial-up modem to connect to. A proprietary MAC or PC client would be installed, and through that client one could look at content, and use email.

Email could generally only be sent and received from people on that specific on-line service. In other words an AOL user could not email a Compuserve user. Also, one could not look at AOL content from a Prodigy or Compuserve client. AOL keyword search, only search AOL. Despite the limitations, AOL became one of the most valuable companies in the USA even buying up media conglomerate Time Warner. It was like the Facebook or Google of the early 1990’s.

Now AOL is gone. What killed it? AOL and the other on-line services died suddenly when the Internet arrived. The fact that standards for email, browsers, web site, and the emergence of Internet wide search, made these walled gardens seem proprietary and limited (which was true). Interoperability wiped out the islands of on-line services
It Really is a Déjà Vu!

Never heard of these powerful on-line services? Then the point is well taken. Yet today we are faced with a new phenomenon that is the "compute enablement" of the planet, through new "cloud" providers. Each of them offers their own "worlds" of computing, storage, social, and other services, with little interoperability.

It is a powerful analogy to the on-line services of the past. It doesn’t really take a visionary to predict a disruption in the long term. The basic services of the cloud providers in terms of compute and storage will need to interoperate. If these vendors do not accomplish this, other new vendors will. And these vendors will become marginalized. If you don’t believe your own instincts, please see the quote from Internet pioneer Vint Cerf.

Public Network Federation Trends

Let’s look even more broadly at interoperability and federation. There are many examples of global standards and interoperable systems emerging in industry, from financial markets, to airlines, to electric power, and of course in IT. Even in networks there are many examples, including plumbing such as Ethernet, or alternative Layer 3 networks such as MPLS. But to make the point so it is understood by everyone, consider the global phone system, the internet, and cloud computing. The slide contains an illustration depicting key attributes of the evolution of federation and interoperability of those systems. In telephony, direct dial and roaming took literally 100 years to develop. In the Internet, we saw the example of the first on-line services, which were in operation for almost 20 years as walled gardens, before internet interoperability took over. Cloud computing interoperability is inevitable and will follow similar architectural trends. Please see the detail in the illustration.

Interoperability Requires Deeper Mechanism

This slide contains an illustration which shows major phases of evolution in IT. One can see that applications software used to be closely linked to a set of servers and storage and was usually installed on premise, in the site of the business. Soon, computing platforms became more flexible, with virtualization and automation, so that applications could share a generalized cluster/platform, which we now call cloud.

The illustration shows that with Intercloud, businesses can run on a multiple datacenter platform, where multiple vendors clouds cooperate and interoperate to form one large distributed platform.
This illustration is courtesy of the Japan based industry association called the Global Intercloud Technology Forum. This group did pioneering work in defining the meaning and requirements for Intercloud.

Architectural Classification of Interoperable Clouds

There are many ways clouds can interoperate. This slide uses an illustration from a research group from University of Melbourne, who published a paper on the taxonomy of different cloud interoperability approaches. Please study the illustration. Intercloud Architecture Framework (ICAF) is based on the multi-layer Cloud Services Model (CMS) and separates control, management, federation and operational aspects in Intercloud. Developed by UvA, currently implemented in the European project GEANT.

Topologies - Different Cloud Interoperability

It is easier to understand these different cloud interoperability approaches when one visualizes the resultant topologies. The illustration in this slide shows the same four categories from the previous slide but from a topology view.

It is easy to see that the multi-cloud approach will work for a user wanting to access a small number of clouds more or less manually and must “string together” the resources from each cloud themselves. This approach is ideal for scientists who want to access simple cloud resources (like VMs) to create large clusters for Hadoop or for Grid calculations. The responsibility for understanding the variations amongst the clouds falls with the user in this case.

It is easy to see that the Federations approach is a much more scalable and interconnected architecture. This approach is ideal for systems which might span multiple cloud providers, or span private and public clouds. It also places the responsibility of understanding the variations amongst the clouds with the infrastructure, not the end user. This is an important point which we will examine more later.

The “Multicloud” Approach

This slide contains an illustration showing that in the Multicloud architecture, the responsibility for understanding the variations amongst the clouds falls with the user. Even if a Broker is used, the broker has the responsibility for understanding the variations amongst the clouds.

Finally, because as far as the clouds are concerned, they have not joined any interoperability scheme, they operate in an independent manner and can change their interfaces whenever
they want to which breaks the interfaces users have to those clouds. While users or user tools could keep up with interface changes in one cloud they will find it much harder to keep up with interface changes from many clouds.

The Intercloud Approach
This slide contains an illustration showing that in the Intercloud architecture, the responsibility for understanding the variations amongst the clouds falls with the infrastructure. On the one hand the clouds are “joining” the federation so as a result they are proactively using a gateway which contains a standardized interface. The users and the infrastructure elements all use this same standardized interface.

Cloud providers are free to innovate their interfaces so long as they provide a gateway which keeps a normalized access. This is a very scalable architecture.

IEEE Intercloud Background
Many researchers have been working on Intercloud. This slide presents a chronology of important events in the development of the Intercloud. Note that this idea emerged way back in 2007 when cloud computing was just brand new.

Since then there have been many researchers working on various aspects of the Intercloud; it has become a very hot topic of cloud research. There is at least one annual IEEE Workshop series dedicated to Intercloud work.

Intercloud Use Case
One very interesting network technology which involves voluntary federation is MPLS VPN. MPLS is an alternative Layer 3 routing system offered primarily by telecommunications carriers. MPLS is extremely complex, creating a VPN address space for the company across all of its locations. MPLS also allows a carrier to offer many dimensions of configuration to a customer including Quality of Service parameters differentiating data from voice. On top of this MPLS VPN encrypts all traffic end to end providing for a private network for the customers.

MPLS is a fully interoperable technology allowing Carriers to wholesale network infrastructure to each other, propagating all of the private address and QOS characteristics from Carrier to Carrier. The illustration in the slide shows an example of MPLS VPN wholesaling. Here, a US Carrier provides VPN to multi-location Corporation via MPLS using it’s own network.
infrastructure. In addition it shows where a US Carrier provides “US VPN” to multi-location Corporation via MPLS via Wholesale of partner network.

This interoperable, wholesale network capability is extremely profitable to all Carriers involved and is a huge business for them. It is a model for how Carriers like to do business with each other, where they can benefit from another Carriers infrastructure.

**Intercloud Use Case (Part 2)**

This slide contains an illustration which shows the very simple concept of adding cloud computing to MPLS based services.

Just as one carrier can wholesale network infrastructure from another carrier to satisfy a geographical footprint needs, that carrier could also wholesale computing or storage infrastructure from another carrier to satisfy that same geographical footprint needs.

Using MPLS to implement federated cloud services is a real example as to how VPN can enable federation.

**Multiple VPC (Virtual Private Cloud) Federation**

This slide contains an illustration showing how VPN can enable Virtual Private Clouds across Carrier infrastructure.

Note that in the illustration, there are two cloud sites, each run by a different Carrier. Note that there are two Enterprises, one “green” and one “black”.

In the first case, “black” company wants 3 VMs. As the illustration shows, “black” company can obtain all 3 VMs from the lower cloud, and so has it’s Carrier provision a VPN to the lower cloud and secures a VPC to it containing the 3 VMs needed.

In the second case, “green” company wants 9 VMs. As the illustration shows, “green” company needs to obtain these using a mix of VMs from both the lower cloud and the upper cloud (this could be because of lack of inventory on the lower cloud, or because of a distributed geographic requirement by “green” company, this topic will be detailed later in this lesson).

Continuing the second case, “green” company has it’s Carrier provision a VPN to both cloud sites, joining together the 9 VMs and supplying them via a VPC to “green” company. The detail of the MPLS VPC provisioning is shown on the next slide.
Multiple VPC Federation Mechanism

This slide contains an illustration of the underlying network machinery for the VPN. In the center you can see the intra-carrier MPLS interconnect router. On the right you can see where the MPLS edge routers for the cloud connect.

Multi Carrier MPLS/VPN Federation

This slide contains an illustration showing how, using MPLS, clouds and networks from different Carriers can interconnect. Using intra-Carrier MPLS, Cloud Service Providers can wholesale their resources into the Carrier networks, or the Carriers can supply their own Cloud Services. Consumers using MPLS can see the Cloud Resources on their VPN networks in either case, if provisioned by the Carrier properly. Through these examples we can see how a dynamically provisioned VPC network can be a key to federation.

IEEE P2302 WG Cloud Interoperability Federation

To solve this problem several researchers have started two related efforts. The IEEE P2302 Working Group for Intercloud Interoperability and Federation (SIIF) is the part of the effort dedicated to create a standard reflecting a working system, which can be tested and developed with the companion effort of the IEEE Global Intercloud Testbed effort.

The Main components of the architecture are:
- Intercloud root
- Intercloud Exchange
- Intercloud gateway

IEEE Intercloud Elements

The IEEE Intercloud project has several elements to it, some of them technical elements such as software and network research and many of them other activities. This slide contains an illustration showing all of these related elements.

As to the technical elements, the architecture begins with core infrastructure including an element called an “Intercloud Root” as well as an “Intercloud Exchange”. A Cloud connects to this infrastructure by using an “Intercloud Gateway”. These Gateways are integrated with several Cloud OS systems. These elements of Root, Exchange, and Gateway are topology parallels to the topology used by Internet Routing and Peering, where the Gateway is the equivalent of an access router. All of these technical elements will agree on protocols, formats, and processes.
As to the non-technical elements, in addition to best practices, and governance, Standards, Industry Associations, University Funded work and Partnerships, and a public testbed come into play.

**Intercloud Protocols Taxonomy**

This slide contains an illustration showing the Taxonomy of protocols used within the Intercloud topology.

The components of the system utilize XMPP for presence and messaging. Once the conversational protocol is established, a services framework is established over Web Sockets. Over this services framework the supplier/requester federations are determined (more detail is provided on that later in this Lesson).

A VPC is provisioned over a bearer network, upon which the federation actually occurs. An example bearer network protocol is indicated on the right where UDT is listed as an example storage federation bearer network protocol.

Other protocols used by Intercloud components include DNS for namespace, and Gnutella for P2P based horizontal (root to root and exchange to exchange) scale-out.

**Intercloud Gateway**

Let us take a closer look at each of the Intercloud components. The first is the Gateway. This is a software module or an appliance which has to be specially adapted to each cloud platform.

On the outside of the Gateway, the “Common Channel Signaling” profile of Intercloud protocols and standards including Naming, Identity and Trust, Conversation Substrate, and Services Transport are implemented. On the inside, the interfaces to the Cloud OS specific Federation API’s and Bearer Network “Drivers” are implemented.

**Intercloud Exchanges**

The next important component in the Intercloud architecture is the Intercloud Exchange. The Exchange is actually Software on a Cloud Platform. Same as all the other components, at the edge of the Exchange is the Gateway, the “Common Channel Signaling” profile of Intercloud protocols and standards including Naming, Identity and Trust, Conversation Substrate, and Services Transport are implemented.
The main function of the Exchange is to solve for resource requests, that is, to use a semantic resource request as a semantic query against the pool of semantic resource inventory. As this is a computationally demanding task, the Exchange is a cloud itself. Exchange algorithms will likely be varied and a point of differentiation. The Exchange also contains a number of other convenient servers (auditing)

Intercloud Roots

The next important component in the Intercloud architecture is the Intercloud Root. The Root too is actually Software on a Cloud Platform. Same as all the other components, at the edge of the Root is the Gateway, the “Common Channel Signaling” profile of Intercloud protocols and standards including Naming, Identity and Trust, Conversation Substrate, and Services Transport are implemented.

The main function of the Root is to provide servers for all functions, for example DNS for Naming, Certificate Authority for Identity and Trust, XMPP servers for Conversational Substrate, and so on.

Roots replicate horizontally using Gnutella P2P protocol and they extend their trust domains as well. These aspects are explained more later in the Lesson.

Namespace Details

Now let us focus on Intercloud Namespace. Clouds provide services via URN designated API endpoints. Behind that service delivery, they have implementations, which are largely opaque to the consumer. As long as the cloud does what it should it’s internal construction is not of the user’s concerns. This is a situation which is identical to networks. Networks have edge routers which are gateways responsible for routing within their opaque network. In the IP networking space, we call individual networks Autonomous Systems and we give them an AS number. Therefore in the Intercloud domain we utilize this same design, calling the autonomous systems CSPs for Cloud Service Provider. A CSP is a number just like RFC 5396 but with some extensions. The IEEE will be the numbering authority for CSPs. DNS will be used to map CSP numbers to URNs. The NAPTR query record of DNS will be used for this mapping, as illustrated by the examples detailed in the slide.

PKI Certificates

The Intercloud topology relies on a “Root of Trust” architecture. The a root certificate is a ITU-T X.509 standard self-signed certificate that identifies the Root Certificate Authority and
includes a digital signature. The root certificate is part of a PKI (public key infrastructure) scheme where the private key of which is used to "sign" other certificates. Exchanges downstream participate in the PKI scheme as shown in the illustration in the slide. In this way, identity and trust is propagated through the system. Note, there can be multiple domains of trust hierarchies, managed by multiple roots. This is envisioned to solve governance concerns of multiple communities and countries.

Trust Management – Extended PKI
This slide illustrates the Extended Trust Management system through the PKI mechanism. From Intercloud topology perspectives, Intercloud Roots will provide static PKI CA root like functionality and Intercloud exchanges will be responsible for the dynamic "Trust Level" model layered on top of the PKI certificate based trust model. Exchanges are the custodians/brokers of "Domain based Trust" systems environment for their affiliated cloud providers. Cloud providers rely on the Intercloud exchanges to manage trust.

Reference Intercloud Components
This slide summarizes the software components of each of the Reference Intercloud Components.

Note that they all have the Intercloud protocol suite: CSP Namespace, Federated Identity, Conversational Substrate (XMPP), and Transport/Services (Web Sockets).
Each component contains modules for its functionality. These are listed on the slide.

Reference Intercloud Topology
The Reference Intercloud Topology is illustrated in this slide. The components which have been detailed are put together here to see the whole topology. Note the purple network, is the signaling (Intercloud protocols) network (also called Common Channel Signaling).

Roots and Exchanges Topology
This slide sets up an example for Intercloud Roots and Exchanges Topology.

In this example root.intercloud.org is the Root of Trust, is the System of Record CSP Numbering feeds NAPTR DNS function, is the XMPP directory and Root Gnutella node for Roots & Exchanges, and has an Aged Semantic Resource Directory.
In this example exchange.intercloud.org acts as a Trust Proxy, is the XMPP directory and Gnutella node for Exchanges & Clouds, holds a Working Set Resource Directory, and mainly serves a Resource Solver/Matching Function.

Finally, in this example intercloud.oneplanet.com (a participating cloud) Uses Gateway to Join Intercloud, Publishes Resource Availability, Makes Resource Requests, Allows network management by Exchange to provision Federations.

**Topology Initialization and Scale Out**

This slide details the Intercloud Topology Initialization and Scale Out.
In this example the Intercloud Root “Zero” is operated by intercloud.org Also called xmpp://root.intercloud.org and http://root.intercloud.org for Gateway XMPP and Web Sockets
Some of the specific characteristics of the Intercloud Root “Zero” are listed in the Slide
In this example the Intercloud Exchange “Zero” operated by intercloud.org
Some of the specific characteristics of the Intercloud Exchange “Zero” are listed in the Slide.

**Exchange & Supplier Clouds Initialization Scenario Flow**

This slide details the Exchange & Supplier Clouds Initialization Scenario Flow
Please consult the steps in the slide as illustrated in the sequence diagram.

**Additional Roots and Exchanges Operated by intercloud.org - Trust Hierarchy**

This slide illustrates the example of a Trust Hierarchy of Additional roots and exchanges operated by intercloud.org
It shows root.intercloud.org also known as root0.intercloud.org
Creating a Trust Hierarchy sideways to exchange.intercloud.org also known as exchange0.intercloud.org
And also Creating a Trust Hierarchy downwards to root1.intercloud.org
And so on.
Additional Roots and Exchanges Operated by intercloud.org – Gnutella Groupings

This slide illustrates the example of Gnutella Groupings for Scale Out of Additional roots and exchanges operated by intercloud.org

It shows root.intercloud.org also known as root0.intercloud.org
Replicating sideways to exchange.intercloud.org also known as exchange0.intercloud.org
And also Replicating downwards to root1.intercloud.org
And so on

Semantic Resource Directory

This slide contains an illustration which details how the Semantic Resource Directory works.
Please note the different relationships as indicated.
Provider Clouds register “inventory” with the Semantic Resource Directory. The Resource

Declarations include several elements, as detailed later in this Lesson

The Root keeps the Semantic Resource Directory. In some implementations the Exchanges may cache the Semantic Resource Directory Consuming Clouds ask the Exchanges to find them needed resources, which in turn check with the Semantic Resource Directory

Example Properties Describing CPU Concepts in the Ontology

This slide contains a table showing aCPU example of some Properties describing Concepts in the Ontology. Please consult the table

Description/Match of A Compute Resource

Resources Declarations are created in a tool which emits OWL descriptions. The slide depicts such a tool showing editing of several example CPU profiles.

When an Exchange seeks to find a resource, it uses the requesting Resource declaration to drive a query in the Exchange. In some implementations, this query will use SPARQL. An example SPARQL query is shown.
An example SPARQL result is show, where several matching resources have been found.
Semantic Resource Definitions have Resource, SLA, and Bearer Network Declarations

Let us examine the content of the Semantic Resource Definitions. Semantic Resource Definitions also have Resource and Bearer Network Declarations. Semantic Resource Definition Uses OWL Specification against a common “Ontology Schema” for Resources (Incl SLA, Pricing) Uses OWL Specification against a common “Ontology Schema” for Bearer Network Suppliers of Resources declare the Offered Resources and the Bearer Network within which they can be Supplied Requesters of Resources declare the Desired Resources and the Bearer Network within which they can be Consumed

Examples of Bearer Network Declarations

Bearer Network Declarations have many different options

This slide details options in the areas of
Network Performance,
Type of Network,
Security of Network,
Special Network Connection,
Metric

Bearer Network Considerations

On the subject of Bearer Network Considerations Clouds may have several options for Bearer Network Exchange finds best match of not only the resource, but also of the Bearer Network

Federation Steps - 1

The next series of slides step through the steps of Federation in detail.

1(a). User Requests Cloud Resources by issuing existing cloud specific UNI API call (EC2/S3, Nova/Swift, OCCI, MS SC 2012) to a Home Cloud.
1(b). If the Home Cloud is able/configured to fulfil the entire User Request locally, it does so and returns the User Request API. The flow stops here as there is no need for rest of the Intercloud federation steps.
Federation Steps - 2
Federation Steps, cont 2(a). Where the Home Cloud wishes to use Intercloud federation to fulfill certain of the User Request, the Home Cloud uses a Cloud OS specific interface to it's associated Gateway communicating the User Request. User Request API has not returned yet. 2(b) The Gateway construct the canonical Resource Description Declaration Including Resource, SLA, and Bearer Network sections. 2(c). The Gateway serializes these into the Federation API format including and invokes them onto the implicit Gateway at the Exchange.

Federation Steps - 3
Federation Steps, cont 3(a). The implicit Gateway at the Exchange and forms appropriate Exchange-Internal API's for the Solver/Matching process. 3(b). The Resource Description Declaration General Form is used to create a SPARQL/SWRL query to the Solver/Matcher; the timestamp/deadline requirements 3(c). The Solver/Matcher seeks qualifying inventory (here this example from Supplier Cloud) solving the constraints/quantifiers and if found, constructs a canonical Resource Description Declaration Including the specific Resource, SLA, and Bearer Network which the Supplier Cloud would deliver.

Federation Steps - 4
Federation Steps, cont 4(a). If the Exchange did not solve for inventory it returns an error to the Home Cloud Gateway, which unwinds this to return the User Request API with an error.
4(b). If the Exchange found inventory it produces a Resource Description Declaration Supplier Cloud Form onto the Gateway at the Home Cloud.
4(c). In this case the Exchange will also construct a Federation API for the Supplier Cloud, The Supplier cloud may or may not choose to optimistically reserve or begin to provision those resources. The User Request API has not returned yet.

Federation Steps - 5
Federation Steps, cont 5(a). The Network Management (SDN Controllers) in the Exchange reach out to the Network Management points in the Home Cloud and Supplier Cloud Gateways. 5(b). The Home Cloud and the Supplier Cloud Gateways may have self-provisioned based on the bearer network information in the Federation API supplied in Step 4. 5(c). If not, the Network Management system (SDN Controller) provisions the connectivity to the bearer network specified in the Resource Description Declaration 5(d). If the bearer network cannot be provisioned, the Exchange returns an error to the Home Cloud Gateway, which returns to the User Request API.
Federation Steps – 6, 7

Federation Steps, cont 6(a). The Home Cloud Gateway serializes the Resource Description Declaration Supplier Cloud Form onto the Supplier Cloud Gateway. 6(b). The Supplier Cloud provisions the resources, if not done optimistically earlier. 6(c). If the Supplier Cloud resources cannot be provisioned, it returns an error to the Home Cloud Gateway, which unwinds this to return the User Request API 7(a). The Resources are provided over the Bearer network via VPN based VPC, to the User as if they were supplied directly by the Home Cloud.

Federation of Workloads

Now let us look at what a resultant topology look like while Federation of Workloads is occurring The slide contains an illustration of a Requesting Cloud and a Supplying Cloud This example is really just the classic VPN based Virtual Private Cloud (VPC) model.

Federation of Storage

This slide contains an illustration of how Federation of Storage looks.

In this case, the object storage cluster in the Requesting Cloud is federating with an extending object storage cluster in the Fulfilling cloud. Under the covers, a VPN based VPC is also used.

Wrap up and Take away

Think about the great ICT systems of the World Phone System Internet MPLS
There are many architectural approaches for Interoperability Portability is for applications Interoperability is for networks Federation is for whole systems
For the Intercloud, key architectural concepts
  Voluntary
  Cloud Decide to Join
  Obtain membership through a numbering authority
  Must adapt their cloud operating system software

Transparent Clouds act as proxies to users Users do not know the machinery which is federating for them
Generalized
Independent of any cloud operating system
Semantic Resources based

Signaling Network Based Like SS7 for the phone network, routing protocols for the Internet
Internet Protocols over XMPP/Web Sockets to find and set up matching resources
SDN to organize Bearer Network Multi-Domain Capable
Technically and Governance wise