Real Time Services Knocking On Your Door

Monique J. Morrow
Distinguished Consulting Engineer

mmorrow@cisco.com

SwiNOG September 29 2009

Agenda

- Attributes “Real Time”
- Example: Davos-Nagoya
- Word on “inter-cloud” somewhere in the Future
- Summary

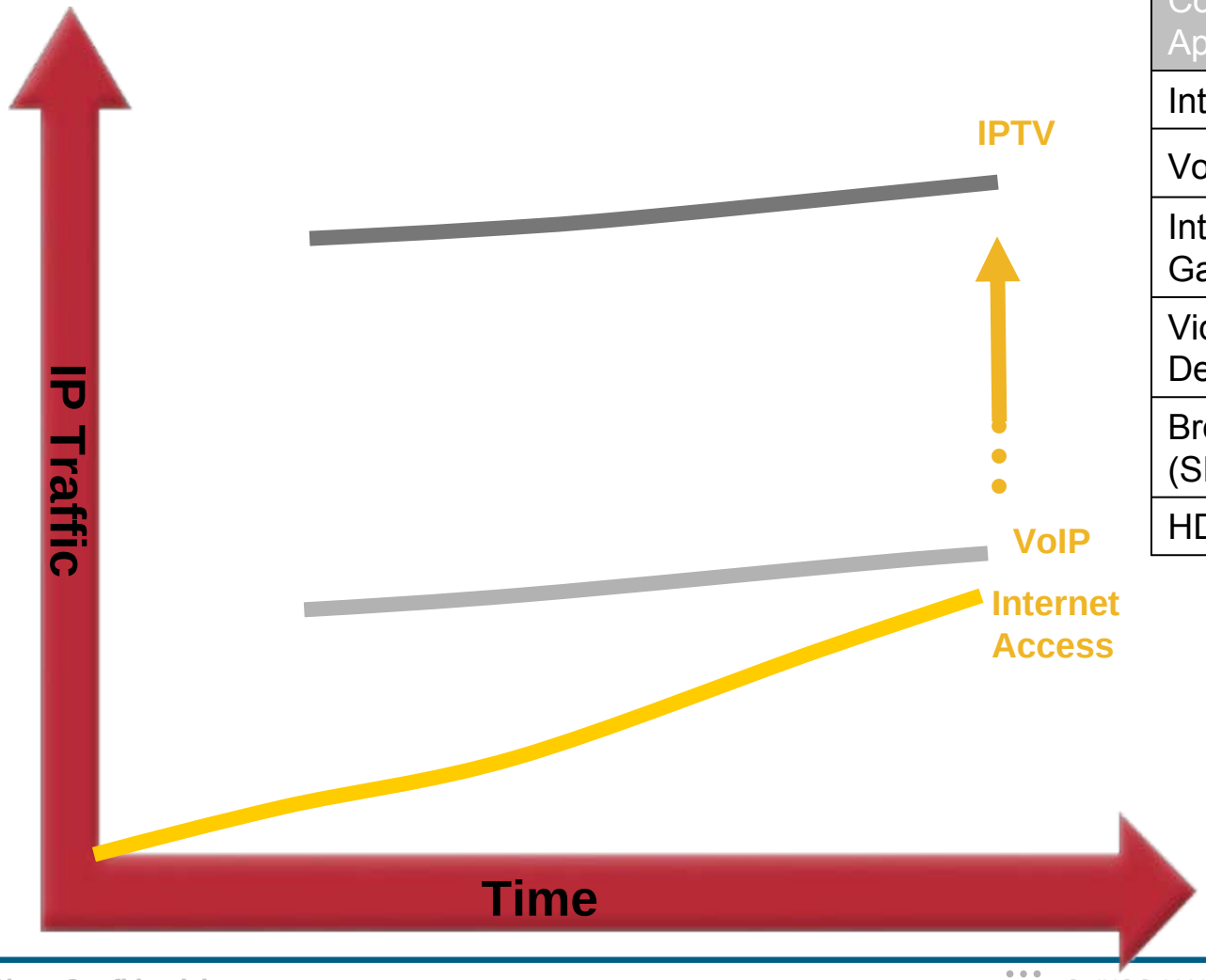
Real Time Attributes

Real Time Attributes

- **Attributes “Real Time”**
  - Require guaranteed delay and throughput for a predetermined period of time (the life time of connection).
  - The value of the delay and throughput parameters can be negotiated during the connection set up time.

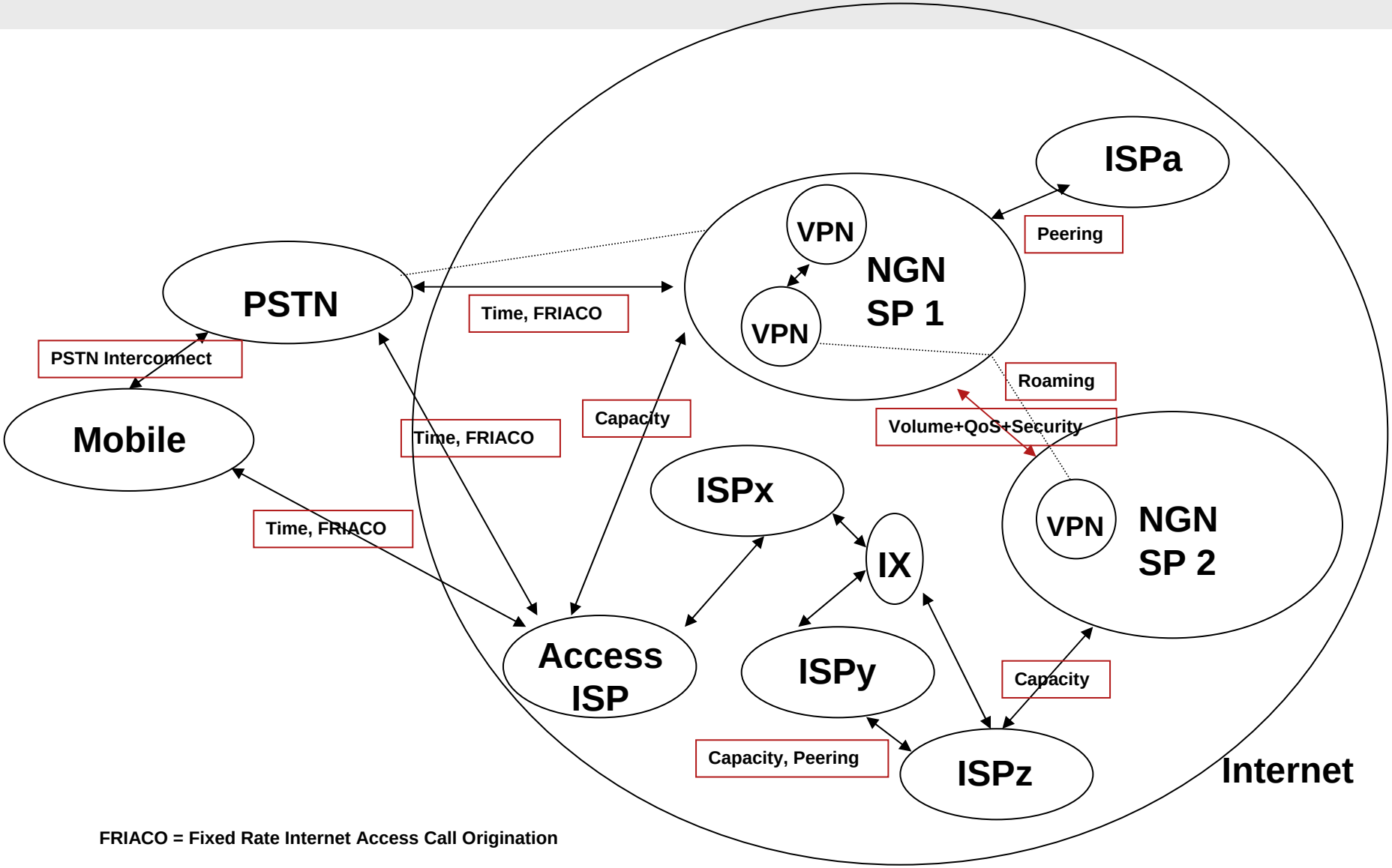
Global Traffic Growth

Real Time and Bandwidth Correlation



Consumer Applications	Bandwidth Required
Internet	.500 - 1.5 Mbps
VoIP	30Kbps-100 Kbps
Interactive Gaming	128k - 6.0 Mbps
Video on Demand	3.0 - 6.0 Mbps
Broadcast TV (SD-TV)	3.0 – 5.0 Mbps
HDTV MPEG-4	6.0 – 7.0 Mbps

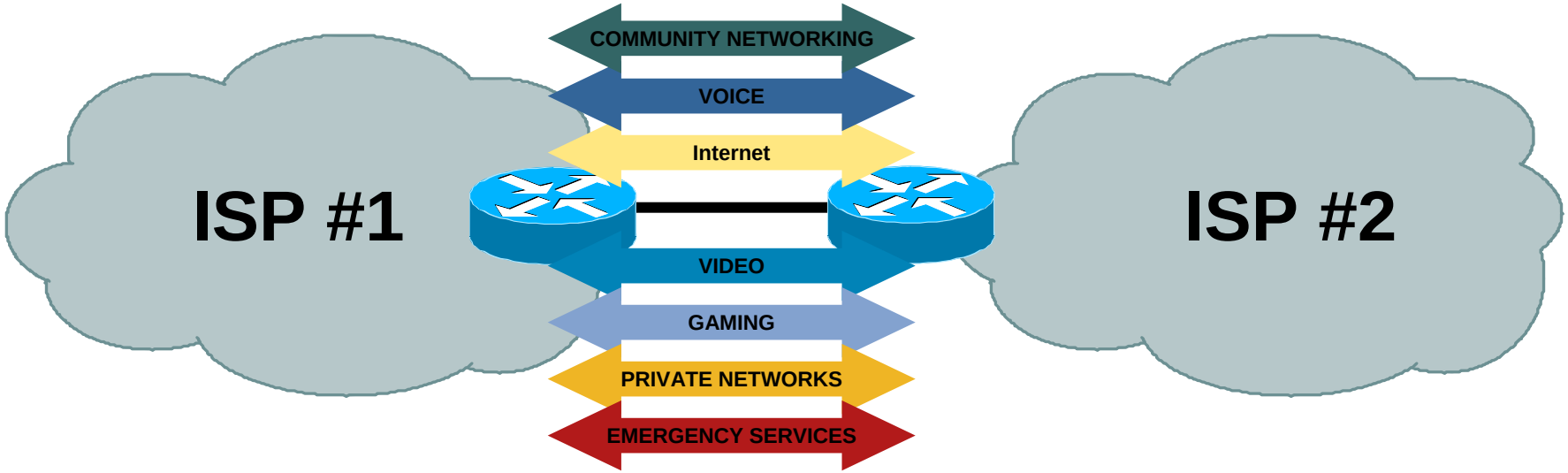
Interconnect?



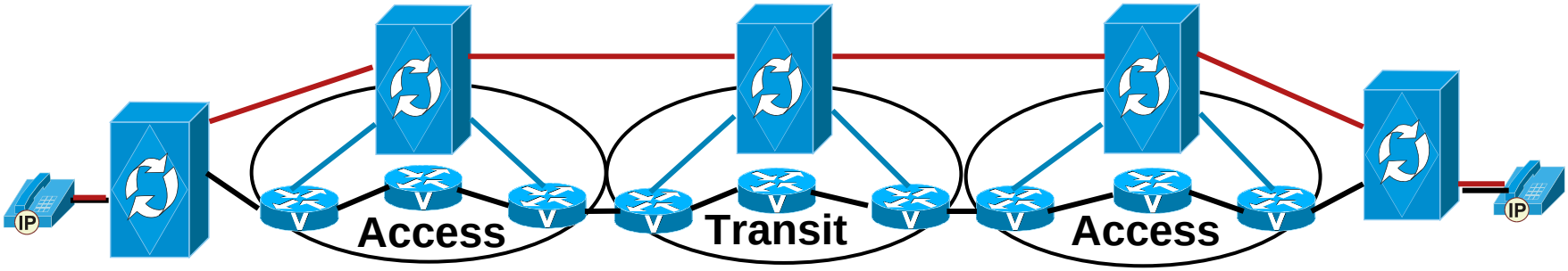
FRIACO = Fixed Rate Internet Access Call Origination

Is Convergence Realistic?

- No longer simply best effort Internet traffic
- Network Convergence and Business Relevance are mandating that new unprecedented network controls be implemented to provide consistent, predictable behavior



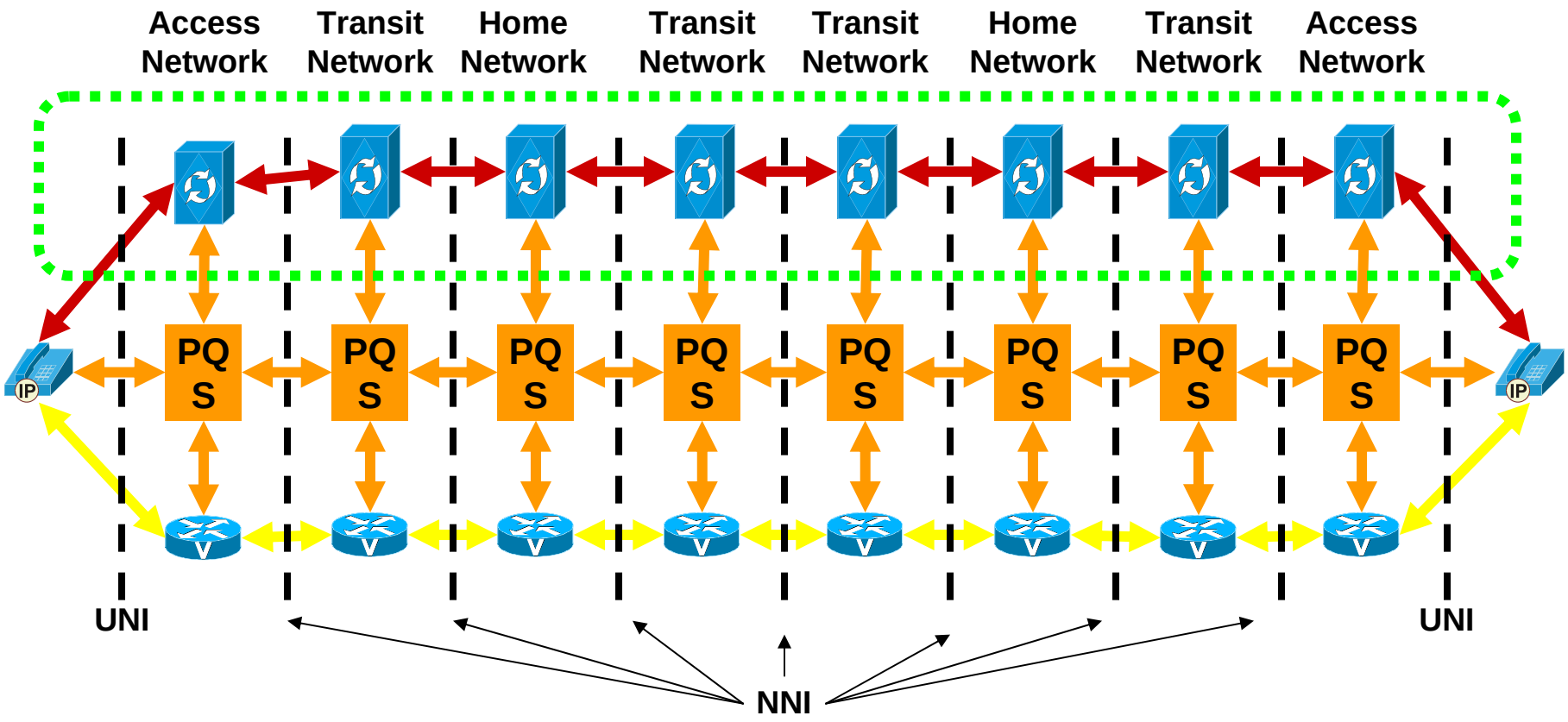
SP Interconnection - Multimedia



- SIP-based call/session control (end-to-end)
- Session border control

- Session control functions:
  - Protocol interworking
  - Call admission control
  - Address translation
  - Routing
  - Billing
  - Lawful interception
- Border element functions:
  - Firewall
  - NAT
  - NAPT
  - QoS
  - QoS monitoring
  - Transcoding
  - Encryption / decryption
  - Mid-call codec change
  - Media transformation
  - Lawful interception
  - Usage metering

SP Interconnect – VoIP Viewpoint 1: Hop-by-hop (Logical view)



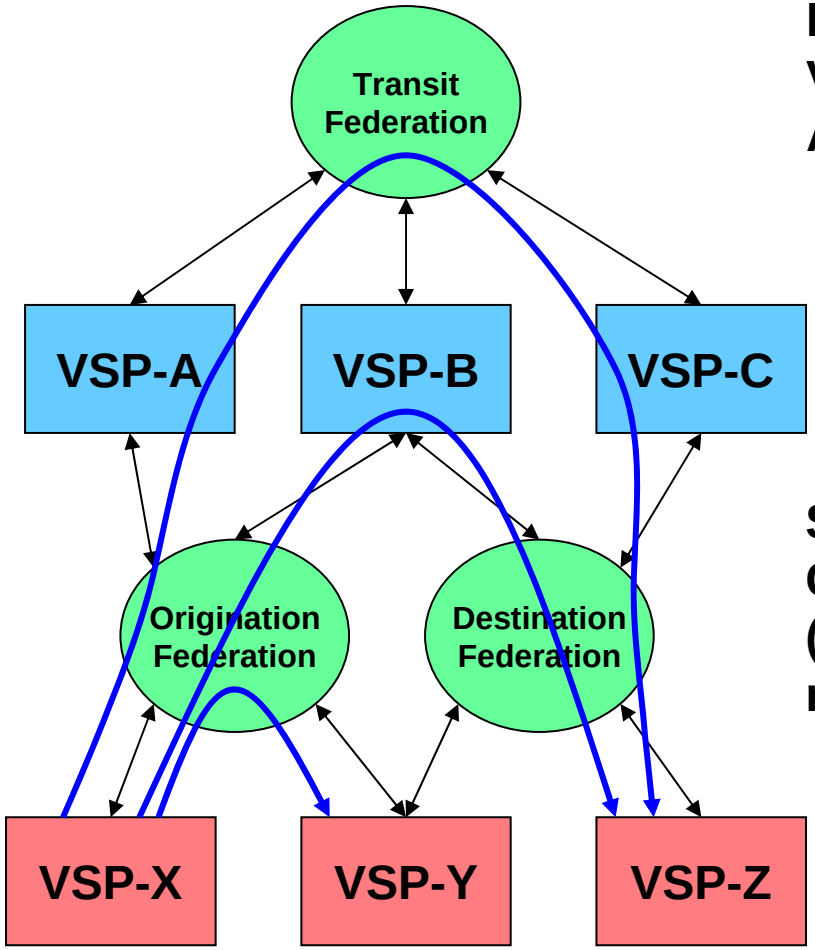
- Signaling (e.g. SIP/SDP)
- Control (e.g. Policy, QoS, Security)
- Media (e.g. RTP)

What are the odds that all VoIP ISPS agree on 381+ RFC/ID's?

SPEERMINT Federation Concept

Federation = Multilateral Rule Set including common Certificate Authority

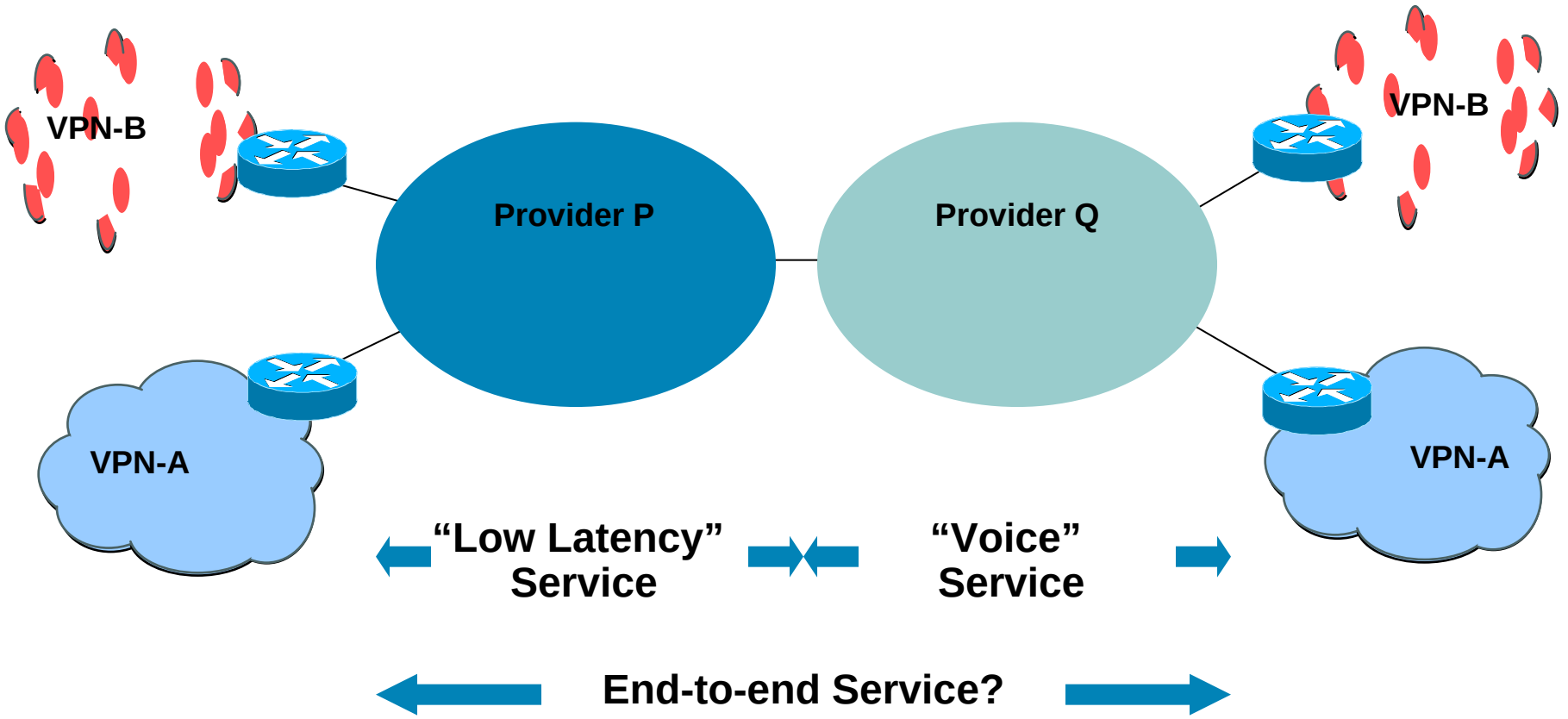
Federation could be organized and operated by a Service Provider



Key Information = VSP to Federation Association in DNS

Stumbling Block = Closed ENUM (Imagine if DNS was not public)

The Challenge of Inter-provider QoS



Implications of two-point promise

- Providers have a common definition of IPDV
- Providers commit to the same “low” and “high” IPDV targets and probability of meeting them
- Reporting can be done on exception basis - report only those intervals where IPDV was “high” or “extreme”
- Delivery of end-to-end SLA considerably simplified
- Providers can collect & report own measurements, but a provider may choose to verify by probing through another provider
- Loss and delay

  Can be handled with “one-point promise”

  Again, report only when promise is not met

RFC 4594* (10 Service Classes)

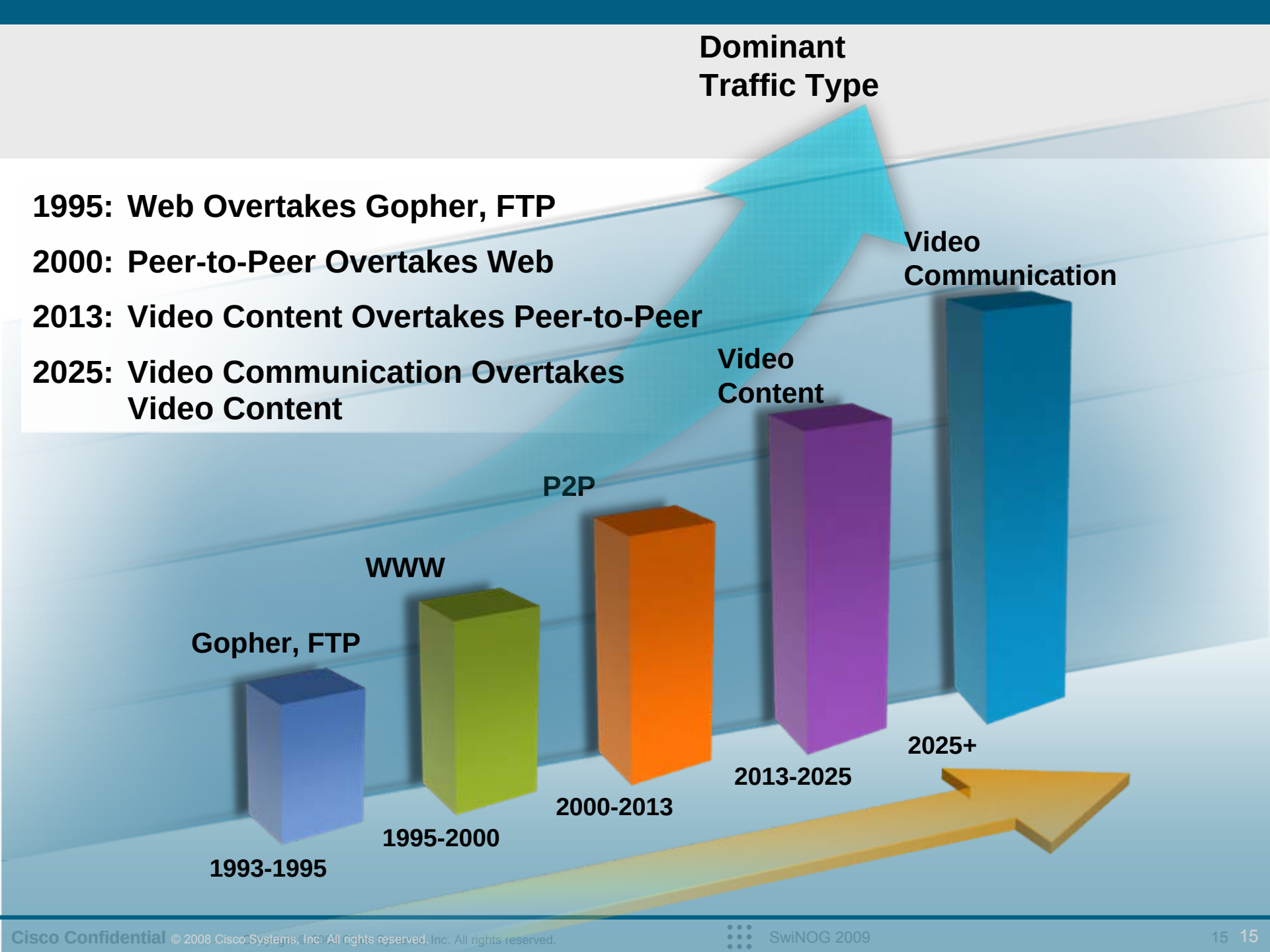
Service Class Name	Traffic Characteristics	Tolerance to Loss	Tolerance to Delay	Tolerance to Jitter
Network Control	Variable size packets, mostly inelastic short messages, but traffic can also burst (BGP)	Low	Low	Yes
Telephony	Fixed-size small packets, constant emission rate, inelastic and low-rate flows	Very Low	Very Low	Very Low
Signaling	Variable size packets, some what bursty short-lived flows	Low	Low	Yes
Multimedia Conferencing	Variable size packets, constant transmit interval, rate adaptive, reacts to loss	Low - Medium	Very Low	Low

*Configuration Guidelines for DiffServ Service Classes, August 2006, Fred Baker et al.

RFC 4594* (2)

Real-Time Interactive	RTP/UDP streams, inelastic, mostly variable rate	Low	Very Low	Low
Multimedia Streaming	Variable size packets, elastic with variable rate	Low - Medium	Medium	Yes
Broadcast Video	Constant and variable rate, inelastic, non-bursty flows	Very Low	Medium	Low
Low-Latency Data	Variable rate, bursty short-lived elastic flows	Low	Low - Medium	Yes
OAM	Variable size packets, elastic & inelastic flows	Low	Medium	Yes
High-Throughput Data	Variable rate, bursty long-lived elastic flows	Low	Medium - High	Yes
Standard	A bit of everything	Not Specified		
Low-Priority Data	Non-real-time and elastic	High	High	Yes

*Configuration Guidelines for DiffServ Service Classes, August 2006, Fred Baker et al.



© 2008 Cisco Systems, Inc. All rights reserved.

The Four Key Design Challenges

Internet Protocol

Information becomes

Agnostic to Content

Bandwidth

Information becomes

Agnostic to Time

Connectivity

Information becomes

Agnostic to Space

Quality of Service

Information becomes

Agnostic to Form

Example: Davos-Nagoya

Background

http://www.r2009.org/

- SWITCH
- See:

//www.youtube.com/ciscoswitzerland#play/user/5AD01DA13F5AA88E/3/nd08vJMQF6M

- See:
- http://www.alphagalileo.org/ViewItem.aspx?ItemId=60661&CultureCode=en

Lesson Learned

- Telepresence in combination with the (over-provisioned) research networks provide a powerful and cost-effective solution for reducing travel while creating an "almost being there" experience
- To get the most out of this set-up the attendees on both sides need to have established a community prior to the event, it took some getting used to talking to each other via TP, our feeling is that that was partly due to the fact that people didn't know each other that well to start with, socializing is still done much better over a drink ;-)
- The plenary sessions worked perfectly, and the side meetings set up proved useful both for the participants and the organizers themselves for last-minute consultation.
- Related discussions Internet 2 forthcoming….

Inter-Cloud

Inter-Cloud Encapsulation

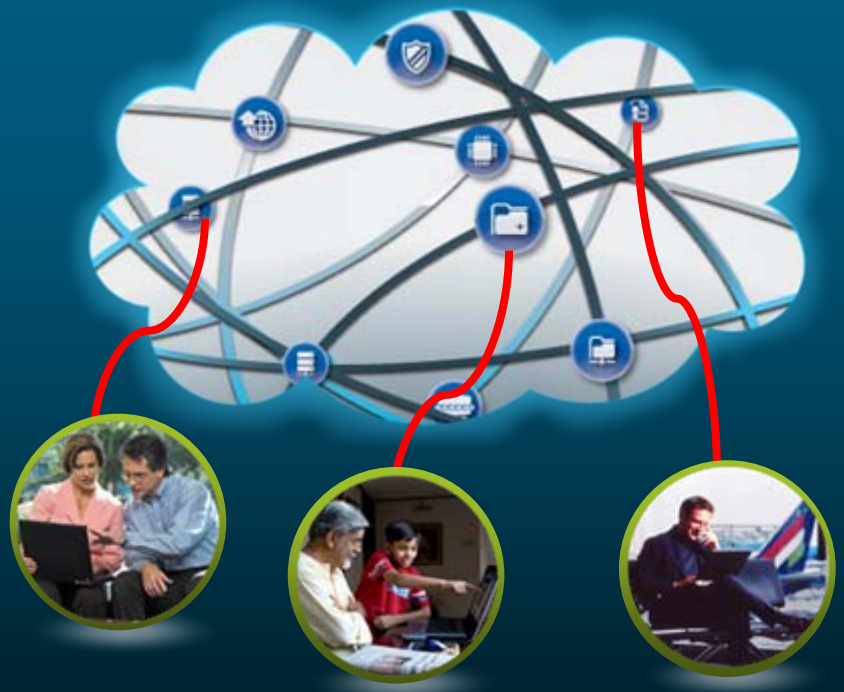
Blessing or Curse?

A Common Definition – Cloud Computing

IT Resources and Services that Are Abstracted from the Underlying Infrastructure and Provided "On Demand" and "At Scale" in a Multitenant and Elastic Environment

A Style of Computing Where Massively Scalable IT-Enabled Capabilities Are Delivered "As a Service" to Multiple External Customers Using Internet Technologies

Source: Gartner "Defining and Describing an Emerging Phenomenon" June 2008

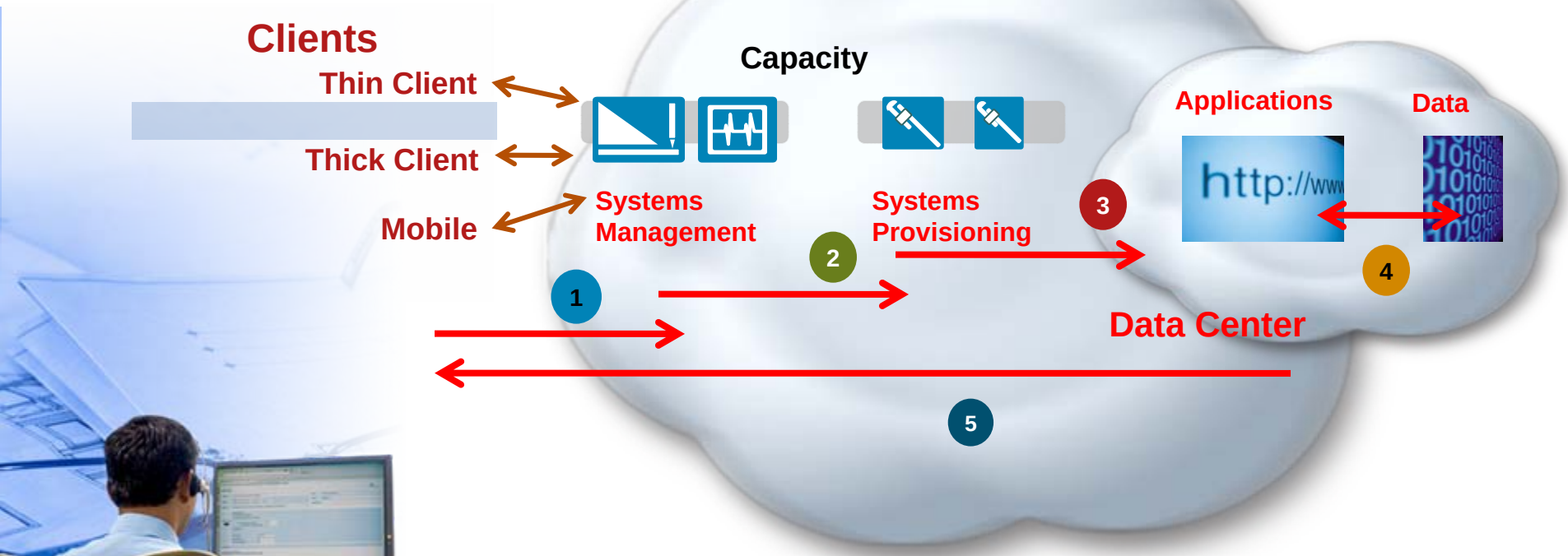


Anywhere, Anyone, Any Service

Cloud Computing – A User's Point of View

IT Service Delivery to a Business User – On Demand

Cloud Computing Architecture



1 – Client sends service requests

2 – System management finds correct resources

3 – Systems provisioning finds correct resources

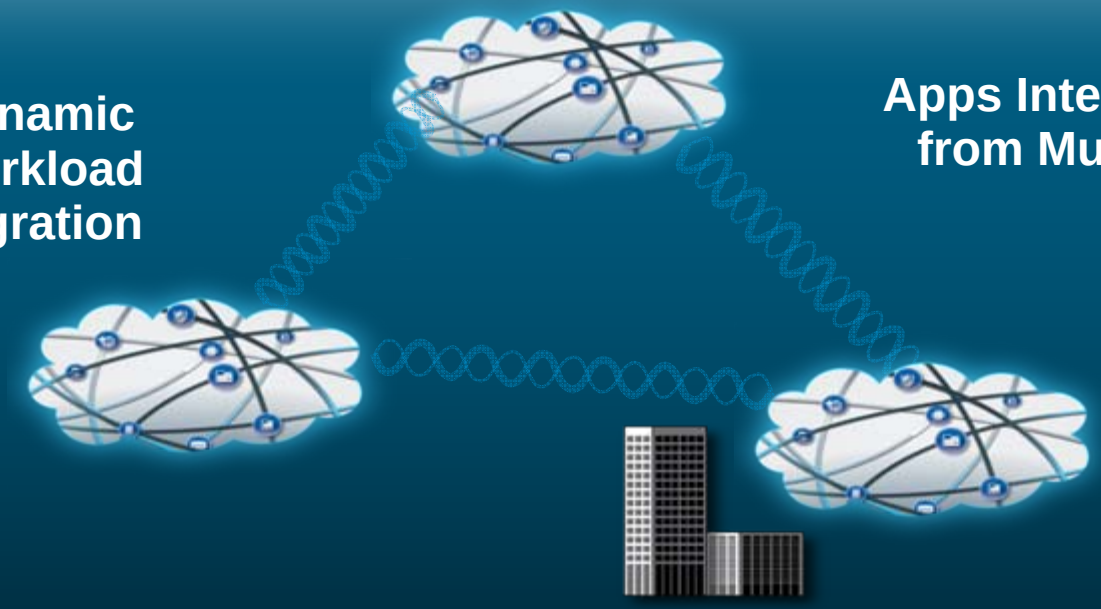
4 – Compute resources are found and service request is executed

5 – Results of the service requests are sent to the clients

Vision—The Inter-Cloud

Flexible Infrastructure and a New Application Platform

Dynamic Workload Migration

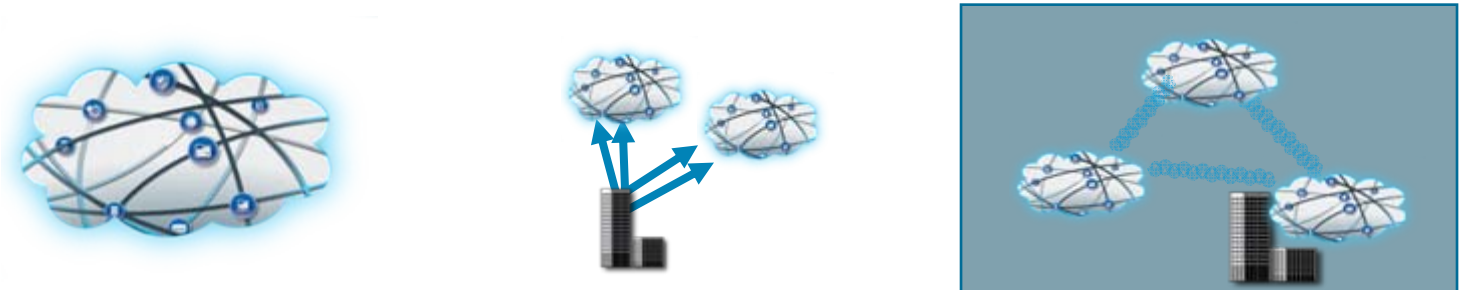


Apps Integrate Services from Multiple Clouds

A Federation of Clouds Based on Open Standards:

- Naming/Discovery
- Trust
- Exchange/Peering

Evolution of the Cloud Elements



	Enterprise-Class		Inter-Cloud
Cloud OS		Application APIs	Application APIs
		Storage Services	Integration Services
	▪ Orchestration	▪ Select Workload Mobility	▪ Load Balancing
	▪ App Deployment		▪ Fault Tolerance
	Billing/Charge-back	Data Access/Mobility	▪ Workload Exchange
	SLA Monitoring		
Extended IP Network	Traffic Isolation	Address Mobility	▪ Identity/Presence
	Security	Monitoring/ Event Processing	▪ Discovery
	▪ QoS		▪ Network Search
	Virtual Switching	Intelligent Caching	▪ TBDs
	Cluster Communications	Policy-Based Selection	

Cloud Security Threats and Issues

- Where is my data?
  - Geographical location of data
  - Who is accessing it on the physical and virtual servers?
  - Is it segregated from others?
  - Can I recover it?
- What is the threat vector for cloud services?
  - Will it be heavily targeted? I don't hear about the cloud-attacks
- How do I identify the the weakest link in cloud services security chain?
- Would centralization of data bring more security?
  - Federated trust and identity issues
- Who would manage risk for my business assets?
- And, can I comply with regulatory requirements set by <choose your standards body>

TOC

- Intro
- Overview
- Problems
- Alternative Ideas
- Summary

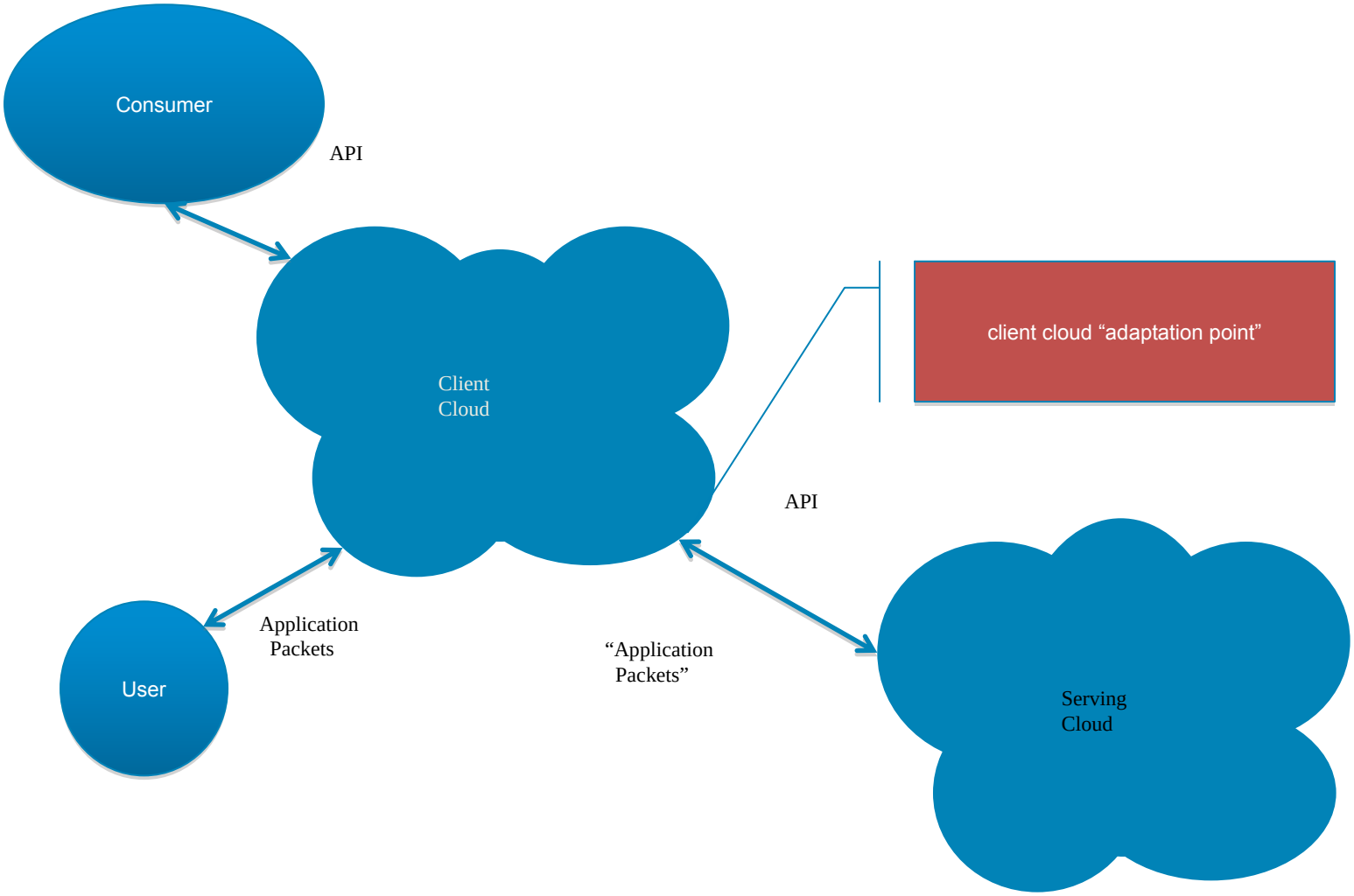
Your Mission Today

- Encapsulation is the Best We Have … so far.
- This is the first approach we came up with that looks like it probably works.
- There may be problems. Can we do better?
- Your job is to attack it and either make it stronger or replace it with something better.

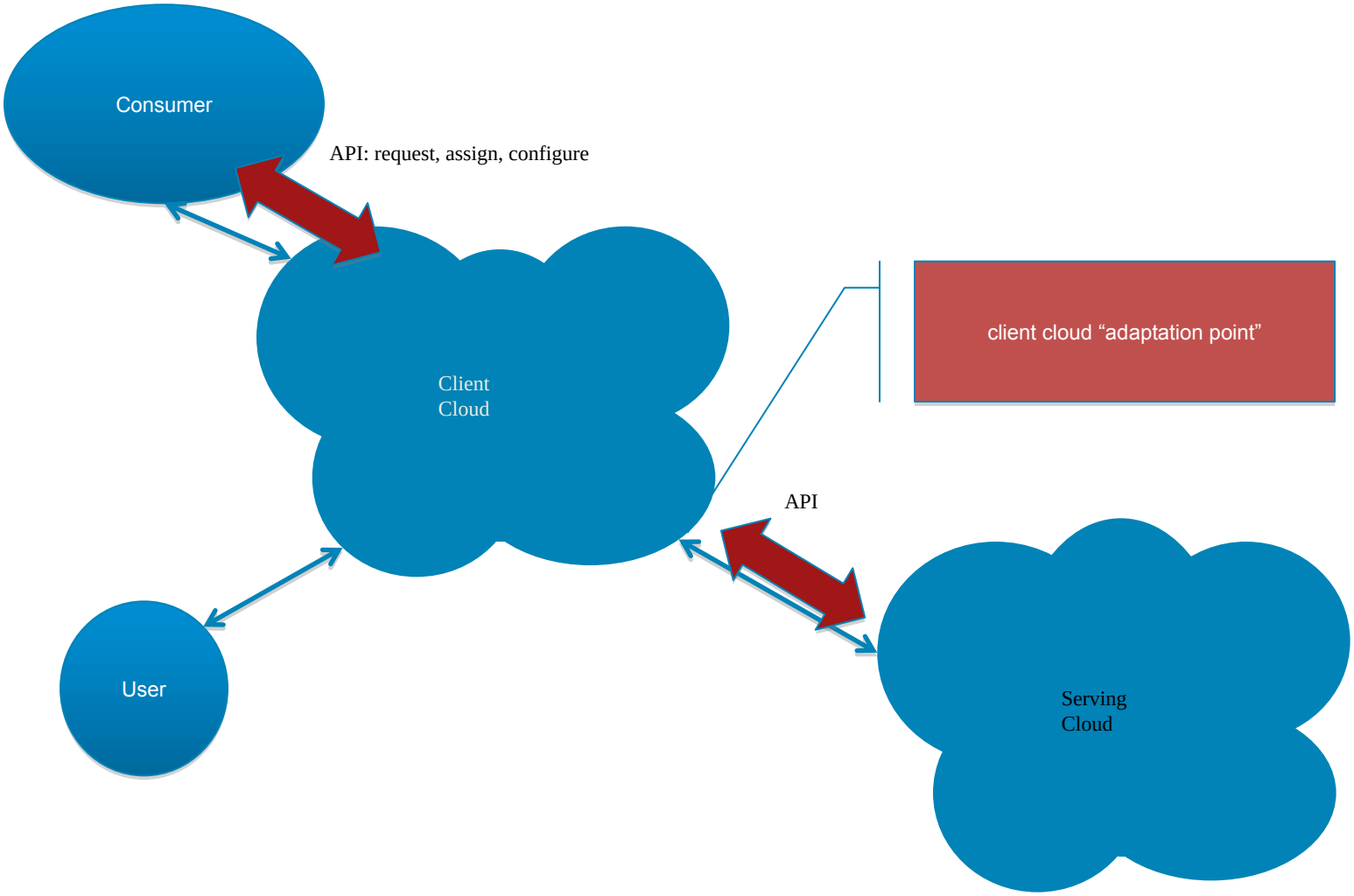
Requirements

- Consumer only sees one provider and one API
- Separation of APIs
  - Decouple API capabilities
  - Allow client cloud API advanced features
- Resource mobility
- Client cloud has complete control of consumer and user experiences
- Client cloud can provide connectivity at L2 and/or L3
- Client cloud-based mashups

High-Level Overview Figure



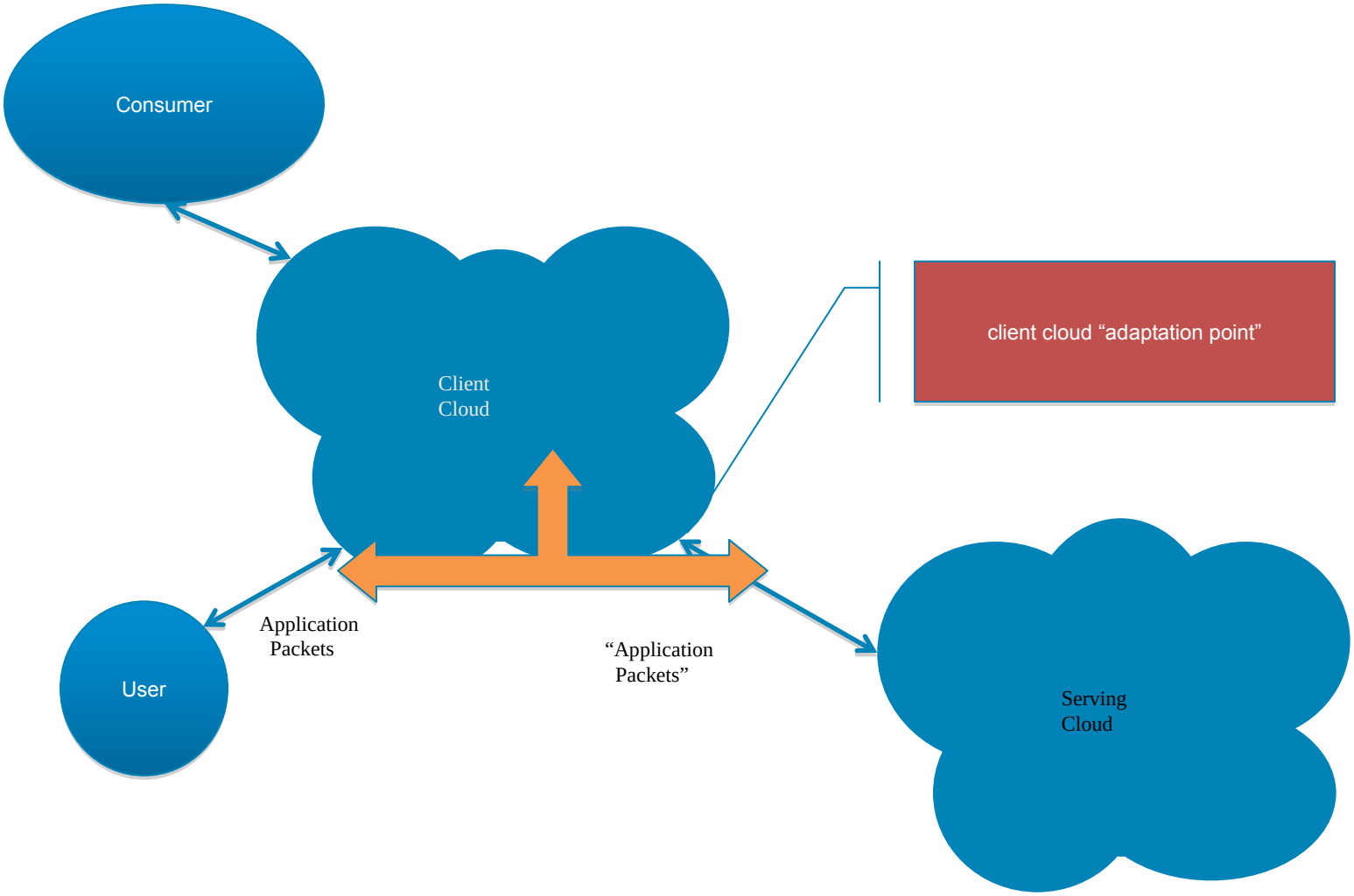
API / Control Traffic



Control Traffic

- Consumer ⇔ Client Cloud
  - Client cloud API for resources in client cloud
- Client Cloud ⇔ Serving Cloud
  - Serving cloud API for resources in serving cloud
  - *Client cloud is consumer to serving cloud*
- Exactly the same relationship. No new capabilities needed (maybe some semantics for new requirements).

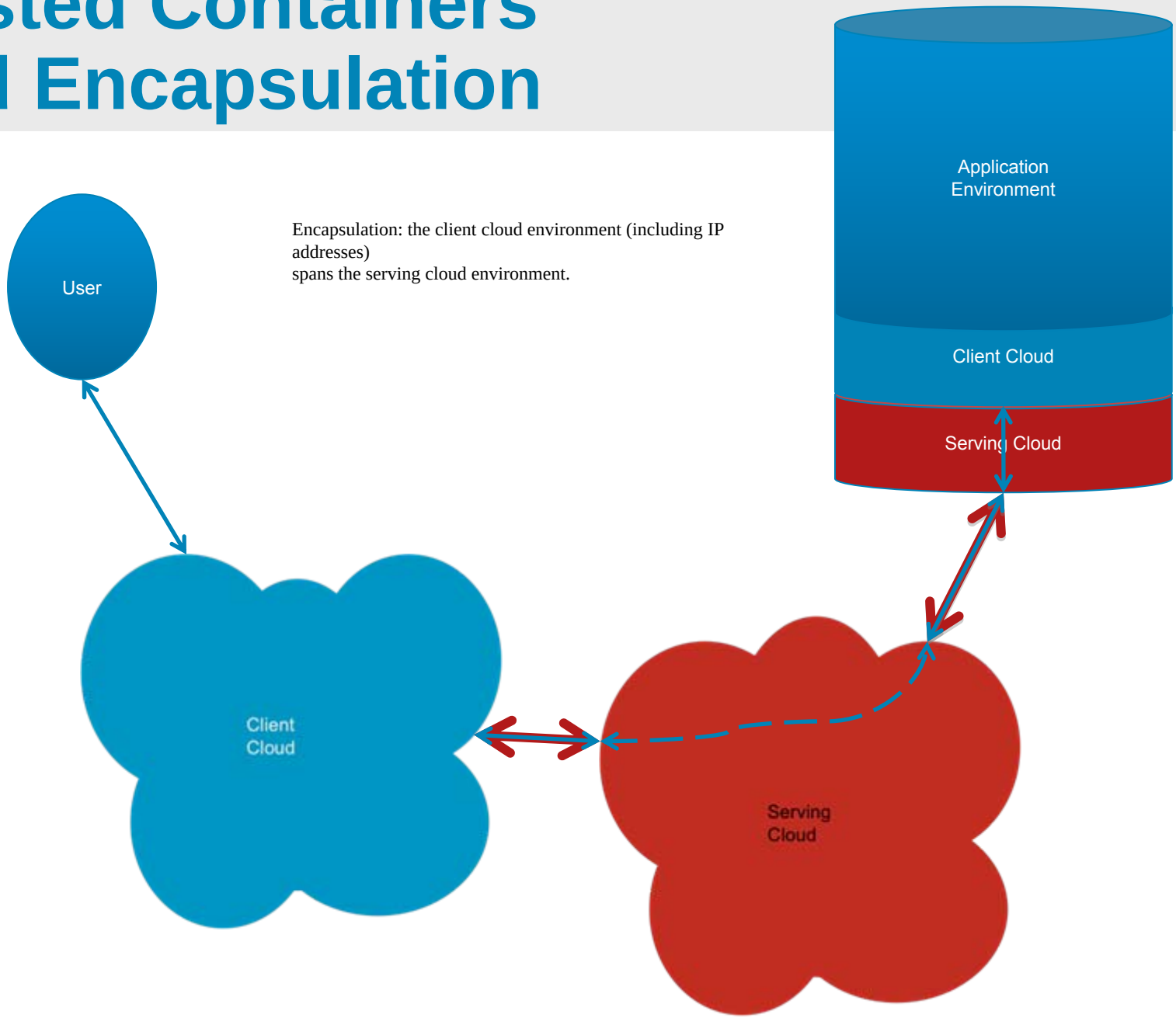
“Application” Traffic



“Application” Traffic

- Actions the client cloud takes on its resources in the serving cloud are seen by the serving cloud as “user” activity (it is not API-based).
  - client cloud is also “user” to serving cloud.
- Application user packets are seen by both client cloud and serving cloud as “application” activity.
- A VM has an IP address assigned by whoever controls it. In the serving cloud, containers are *nested*.

Nested Containers and Encapsulation



Resource Mobility

- With encapsulation, a cloud's resources are all under its control regardless of where they are.
- Use the owning cloud's internal mechanisms.
- Non-VMs: client cloud resources can move in and out of serving cloud resources.
- VMs: internal L2, LISP, MIP, whatever.

Problems?

- MTU
  - If a VM moves from intra- to inter-cloud and MTU changes, will there be problems?
- Path length due to “trombone” effect
  - (see also later slide)
- Performance questions (VM in VM, in VM?)
  - Does a VM support emulation of VM accelerators?

When is Path Length a Problem?

- If the client cloud is widespread, it may be close to most app users. Packets take good path across client cloud to serving cloud.
- If the serving cloud is also widespread, resources can be moved near customers (just as with a single cloud).
- It depends on the customer topology.
- We need to check with customers and their account teams about this.

Avoid Encapsulation?

- Can we get avoid encapsulation and meet the requirements?
  - Consumer only sees one provider and one API
  - Separation of APIs
    - Decouple API capabilities
    - Allow client cloud API advanced features
  - Resource mobility
  - Client cloud has complete control of consumer and user experiences
  - Client cloud can provide connectivity at L2 and/or L3
  - Client cloud-based mashups
  - Client cloud has complete performance/usage info

Alternative Ideas

- If serving cloud can add special capabilities …
  - External LISP, internal handling:
    - Serving cloud announces a LISP mapping to a client cloud prefix.
    - Serving cloud container enhanced to deliver that address.
    - LISP being considered for intra-cloud use anyway.
  - Special address:
    - URI -> special serving cloud address. Both containers know it goes to the client cloud app (not client cloud container).
  - External MIPv6, special address:
    - Client cloud VM uses special serving cloud address as a MIPv6 care-of address. (v6 only)

Summary

- It looks like encapsulation is hard to beat.
- If not required, can we ask a cloud to add container features in order to be a serving cloud? What about adding LISP?
- Need to find out about
  - MTU if VM moves between inter- and intra-cloud
  - VM-in-VM performance
  - Customer topologies re tromboning

Thank you