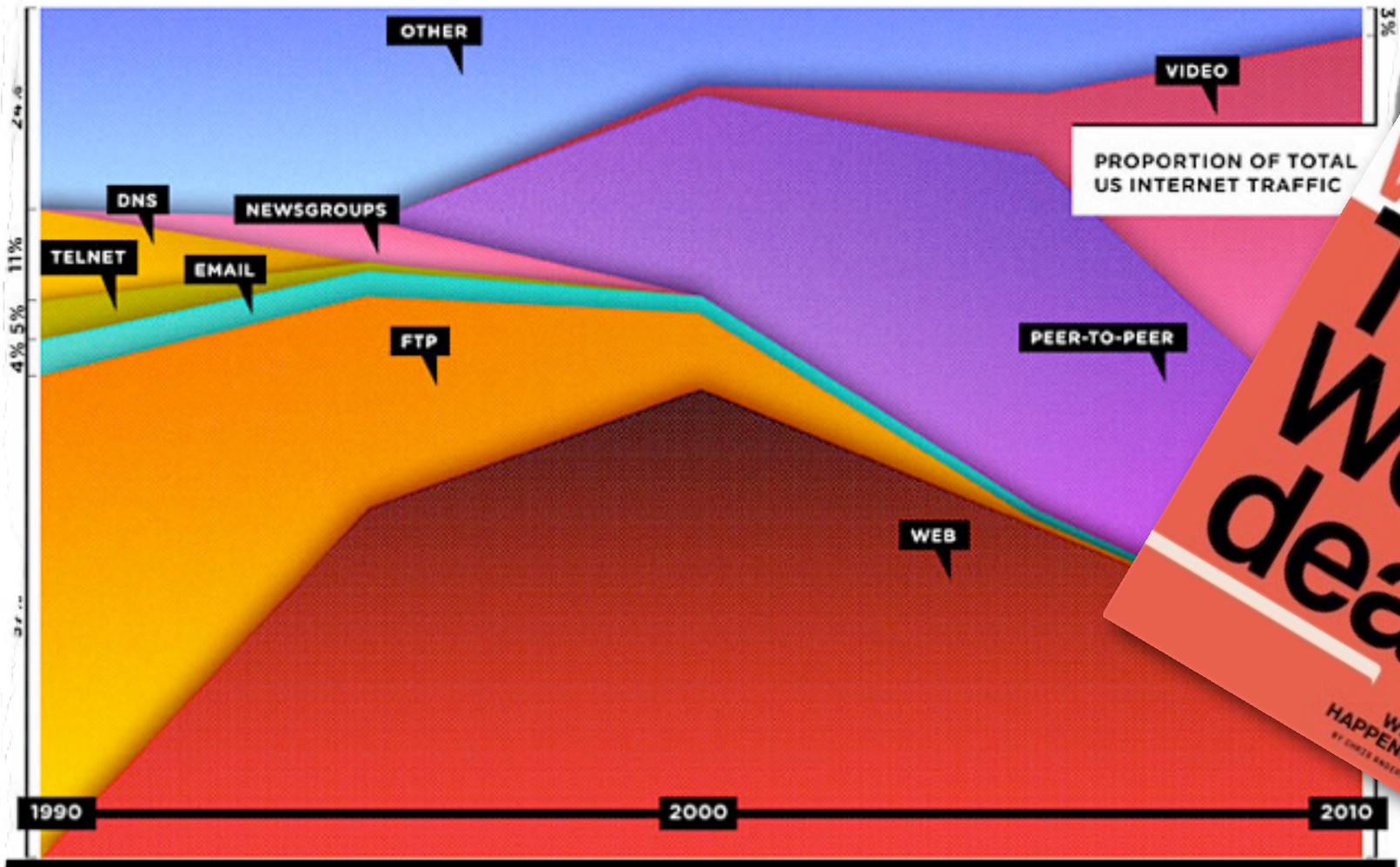
Cloudinomicon ::

Idempotent Infrastructure, Survivable Systems & The Return Of Information Centricity

v1.8

The Web Is Dead. Long Live the Internet

by Chris Anderson and Michael Wolff ✉ August 17, 2010 | 9:00 am | Wired September 2010



Sources: Cisco estimates based on CAIDA publications, Andrew Odlyzko

The Web Is Dead? A Debate

How the Web Wins

Two decades after its birth, the World Wide Web is in decline, as simpler, sleeker services — think apps — are less about the searching and more about the getting. Chris Anderson explains how this new paradigm reflects the inevitable course of capitalism. And Michael Wolff explains why the new breed of media titan is forsaking the Web for more promising (and profitable) pastures.

“The Internet Is Over*”

* Everybody knows you can’t argue with royalty

So The Cloud's Got THAT Going For It...

Fist Pump The Cloud

The Car Crash You Just Can't Stop Watching

Four Horsemen Of the Virtualization Security Apocalypse

The Frogs Who Desired a King: A Virtualization & Cloud Computing Fable Set To Interpretive Dance

Cloudifornication: Indiscriminate Information Intercourse Involving Internet Infrastructure

Key Takeaways

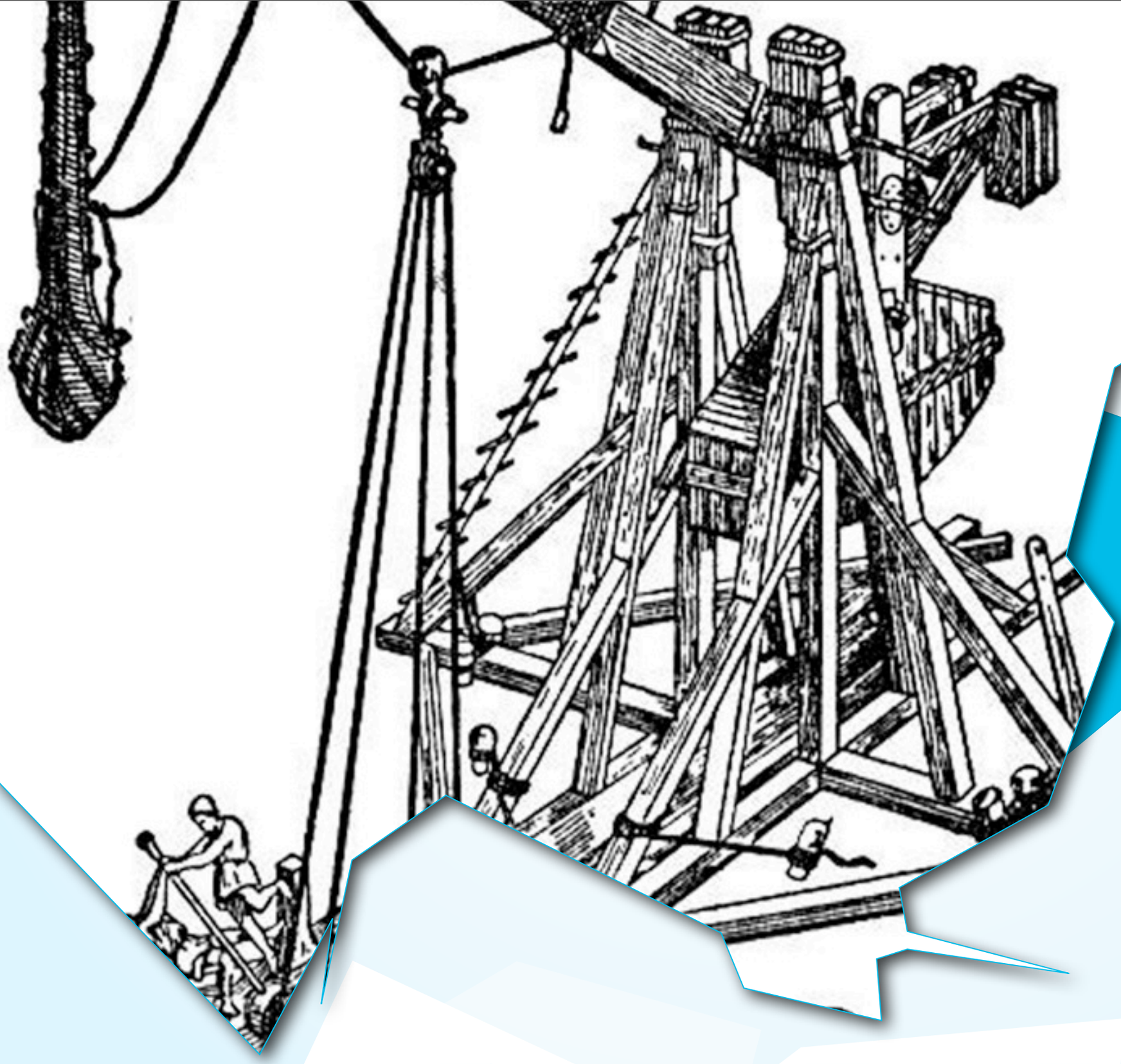
+ Not All Cloud Offerings Are Created Equal or for the same reasons.

+ Differentiation Based Upon PLATFORM: Networking, Security, Transparency/Visibility & Forensics.

+ Apps In Clouds Can Most Definitely Be Deployed As Securely Or Even More Securely Than Those In An Enterprise...

+ **...However, Often They Require Profound Architectural, Operational, Technology, Security and Compliance Model Changes**

+ What Makes Cloud Platforms Tick Matters In the Long Term



Blame The French::

Cloud, Siege Warfare & the Trebuchet...

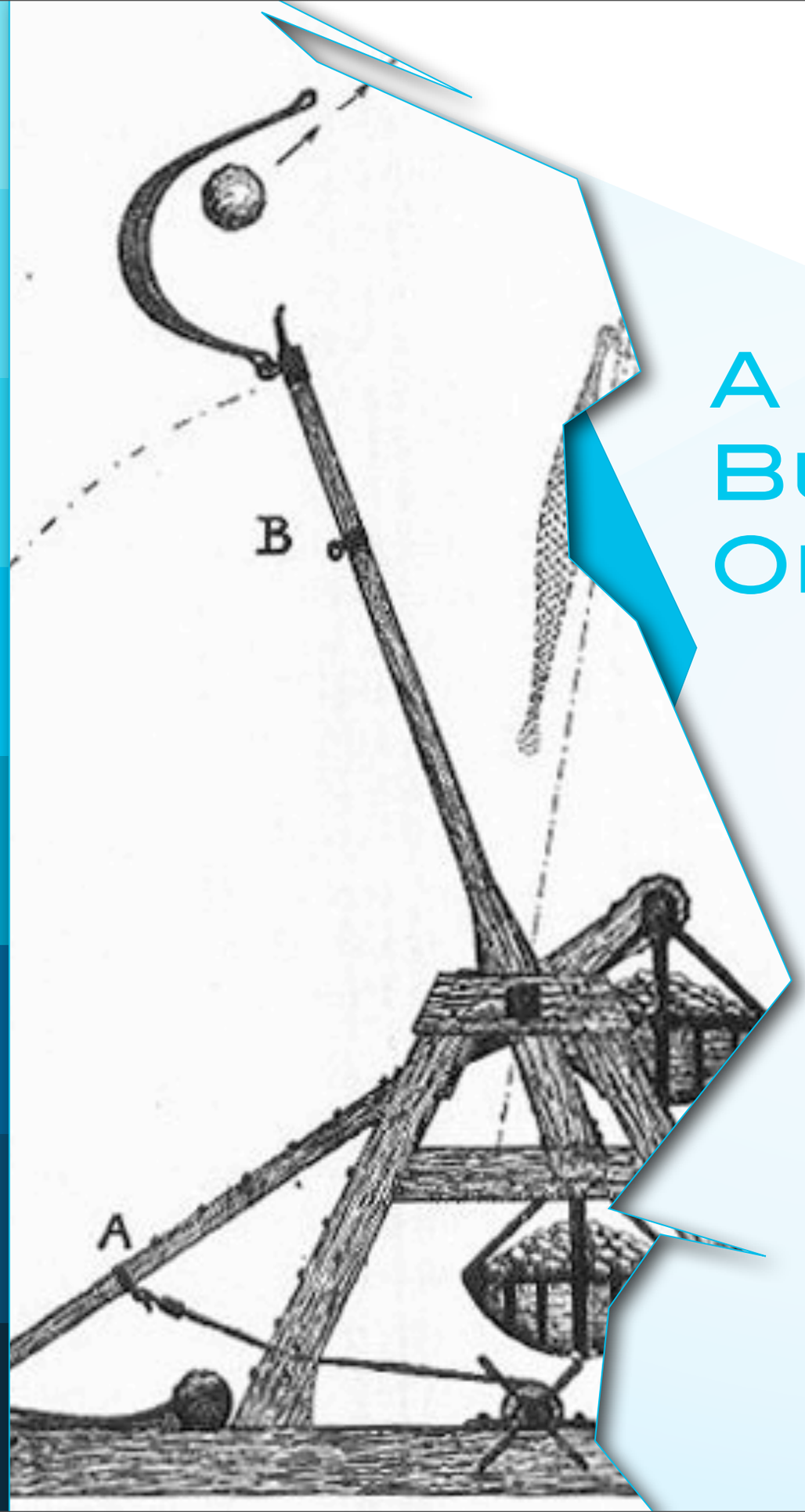
Technically Blame the Greeks & Romans...

+ Introduced in ~12th century by the French who bettered the design elements of the catapult & ballista

+ The trebuchet utilized a sling to double the power of the engine and throw its projectile twice as far

+ Catapults were efficient mechanisms for lobbing loads of 50-60 pounds

+ Trebuchets could throw stones of up to 300 pounds and at great distance



A Better Mousetrap But A More Complex Operational Model

+ The sling trebuchet was a marriage of previous catapult design, application of better physics & advanced physical science.

+ It works on a simple principle, but there was nothing simple about making sure a sling trebuchet was built or operated with precision...*

*http://www.medieval-castle-siege-weapons.com/sling-trebuchet.html

What Does That Have To Do With Cloud?

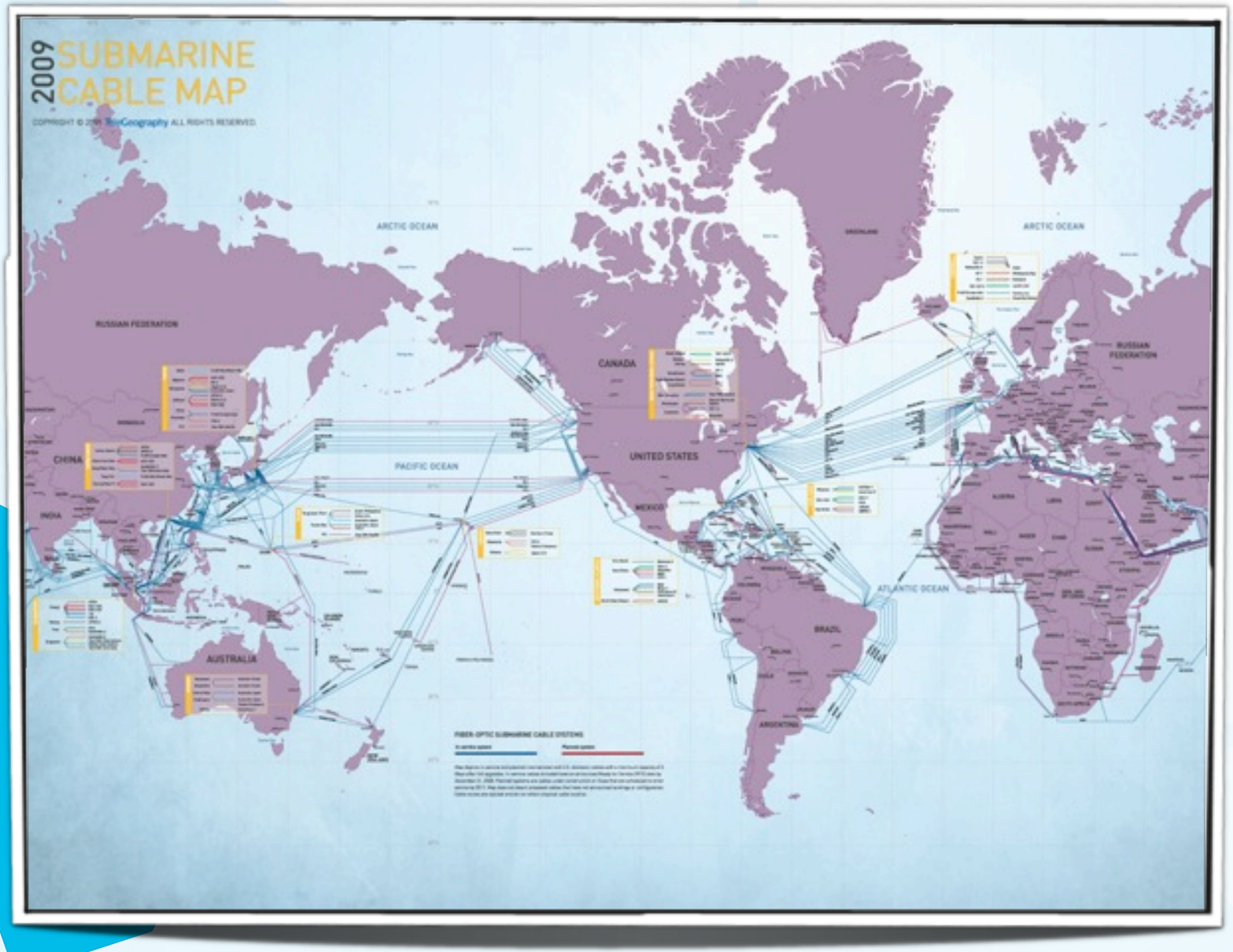
+ Evolutionary application of revolutionary ideas*
+ Caused quite a stir and a wholesale shift in strategy
+ Laid the foundation for even more innovation
+ Set the stage for automation!

* Gravity. Who Knew?

Shifts In Thinking*

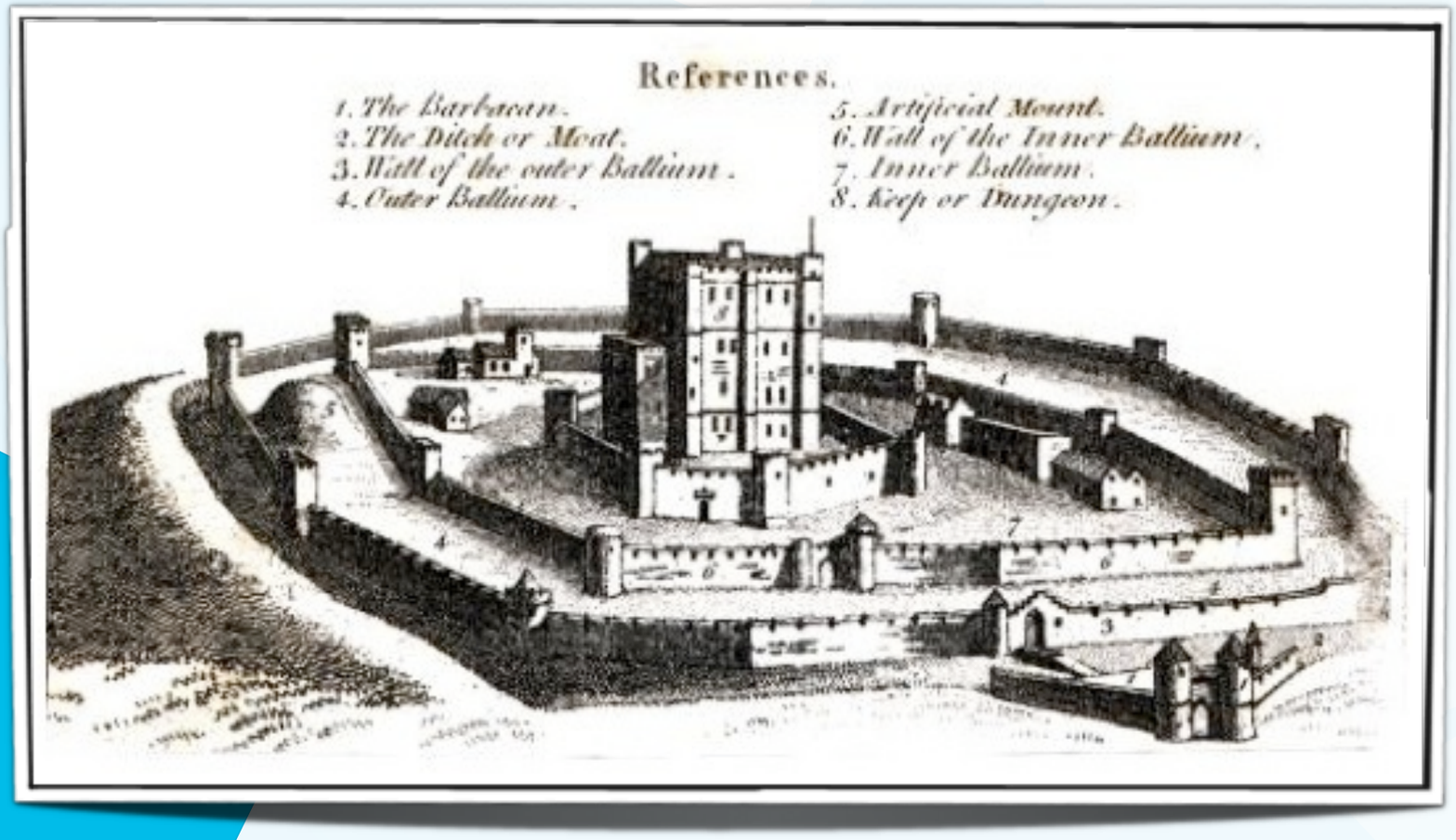
IT Evolves

*2001 - Carnegie Mellon University. Information Survivability: A New Executive Perspective



Centralized To Global

Bounded To Unbounded



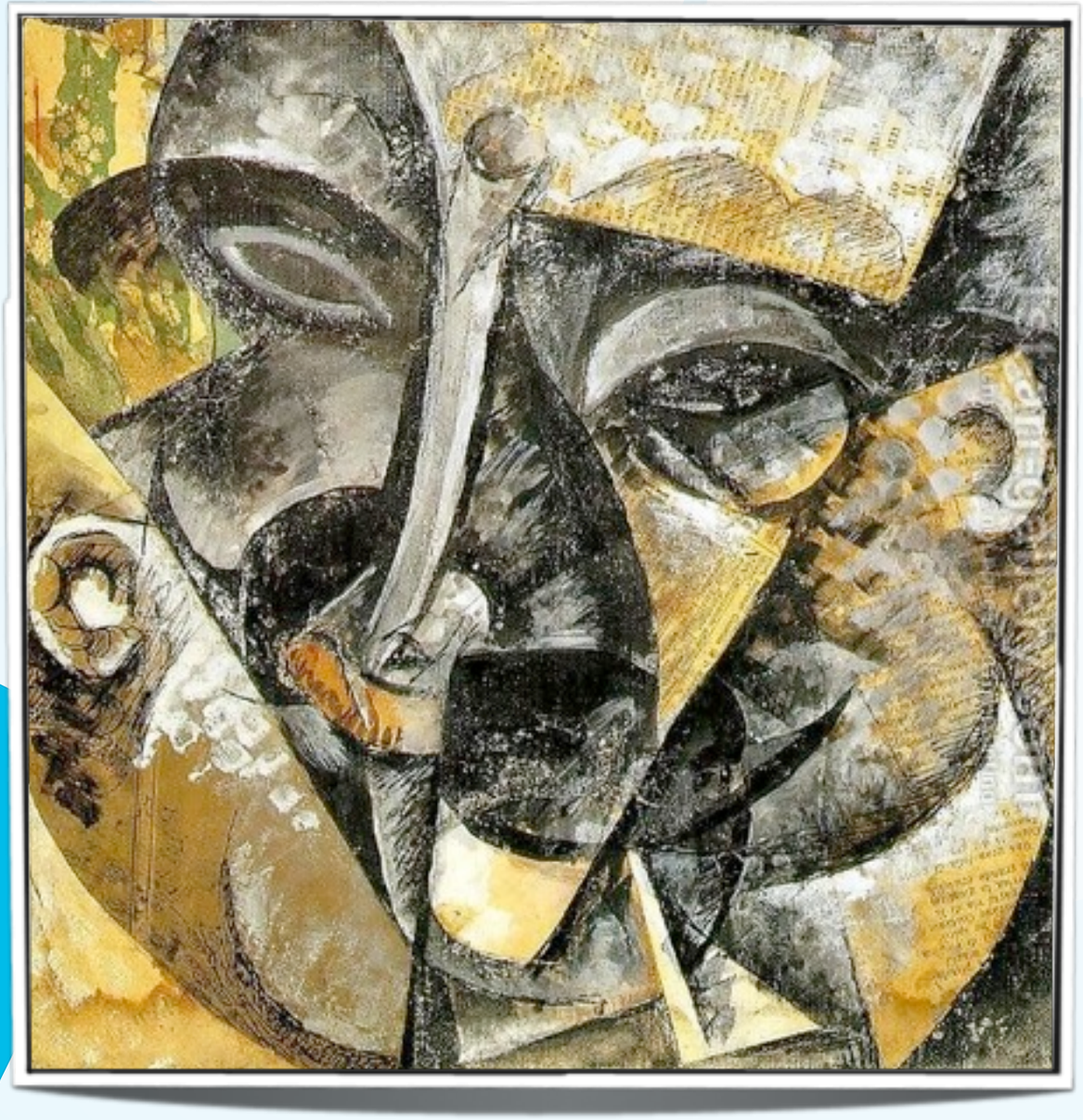
Insular To Networked

Predictable To Asynchronous

Single To Shared Responsibility

Overhead To Essential

Security To Survivability



Static To Dynamic*

*I Added This One

Manual To Automated*

*I Added This One

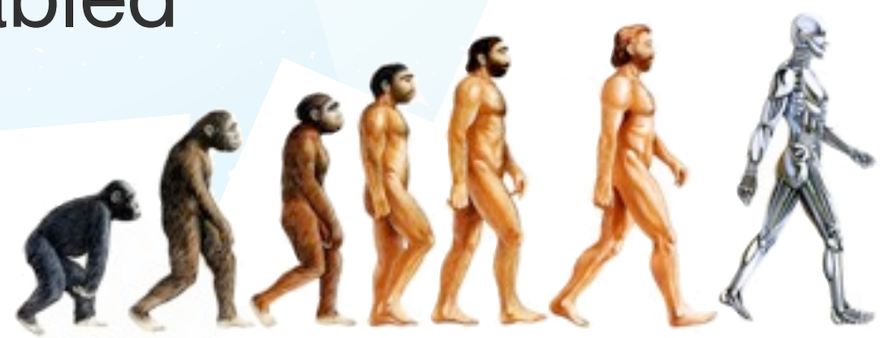
Shifts In Thinking: IT Infrastructure Evolves

+ Consolidating From Servers To Pooled Resources Of Compute

+ Network & Storage Moving From Dedicated Switches & Local Disk To Clusters And Fabrics, Implemented In Both Hardware *And* Software

+ Escaping From Tightly-Coupled Hardware/Software Affinity To Distributed Computing Often Enabled By Virtualization

+ Transitioning From Infrastructure To Composeable Service Layers



Something Wicked This Way Comes...

+ Not All Public Clouds Are Created Equal Or For The Same Purpose
+ Scale Enabled By Abstracted & Idempotent Infrastructure
+ Massive Data Centers With Hundreds Of Thousands Of Cores, Huge Storage And Bandwidth
+ Extremely Agile, Heavily Virtualized, Mostly Automated & Hugely Software Driven
+ Management Via API

Across The Great Divide...

Therein lies the problem...

+ Huge monocultures of custom hardware and software layers abstracted for your pleasure

+ It's the functional equivalent of Siebel: don't fit the software to the business, change the business to fit the software.

+ ...not necessarily a bad thing, but cultural, operational, security, and compliance issues are daunting.

Cloud:

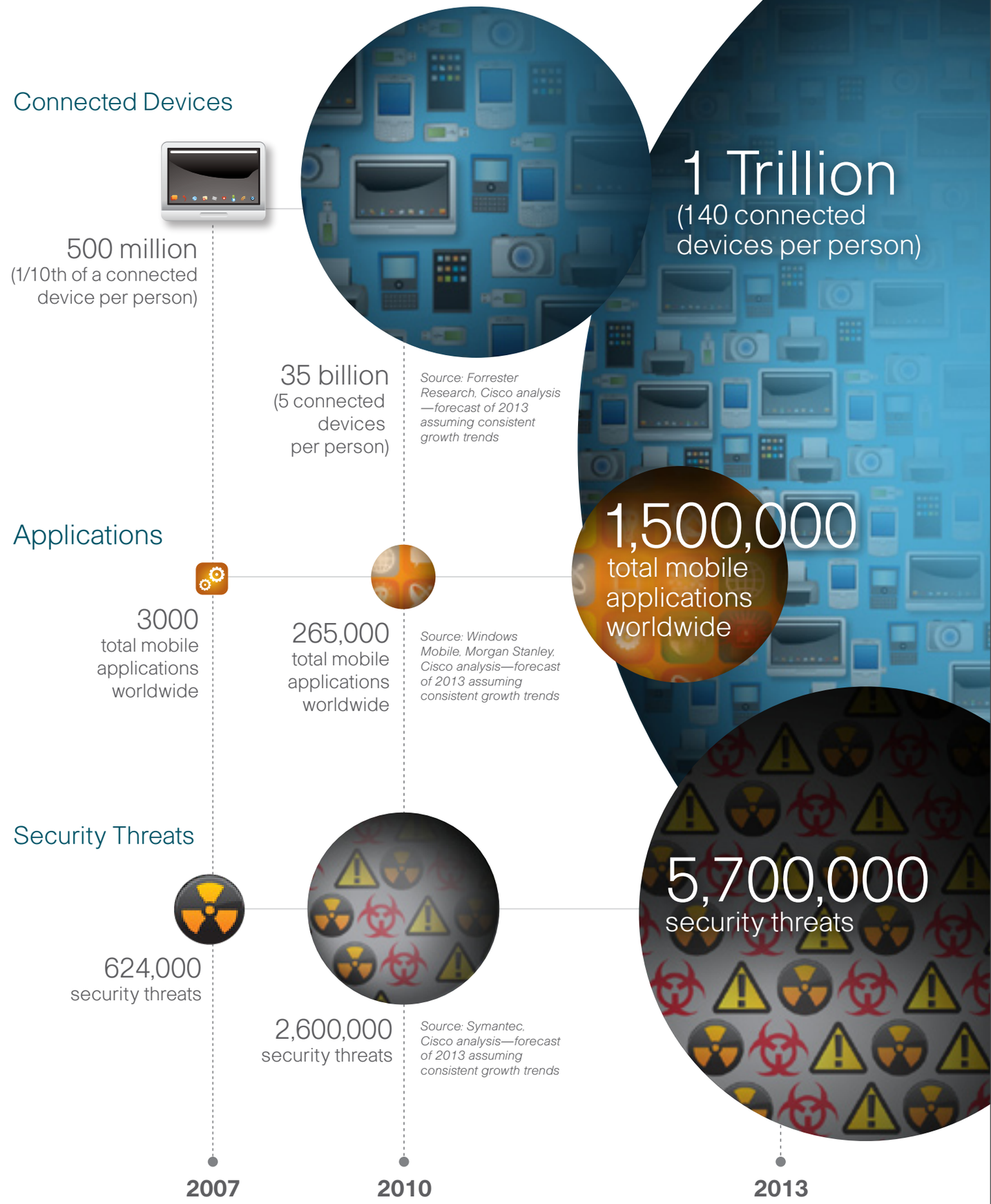
All About Gracefully Giving Up Direct Operational Control Over Infrastructure

We Need Risk Ritalin

There Be Monsters Here...

+ Suffering From Security Attention Deficit Disorder & Lack Of Holistic Approach
+ Threat Model Velocity And Innovation Of Attacker > Defender
+ Security Doesn't Scale
+ Economic Model Does Not Incentivize Solutions That Solve Long Term Problems

Happy Happy, JoyJoy



Cisco 2010 Mid-Year Security Report

‘Round & ‘Round We Go…

Mobility

Mainframes

Achtung! Divergent Models

The Cloud

Web2.0

Client/Server

Web1.0

Display

Compute

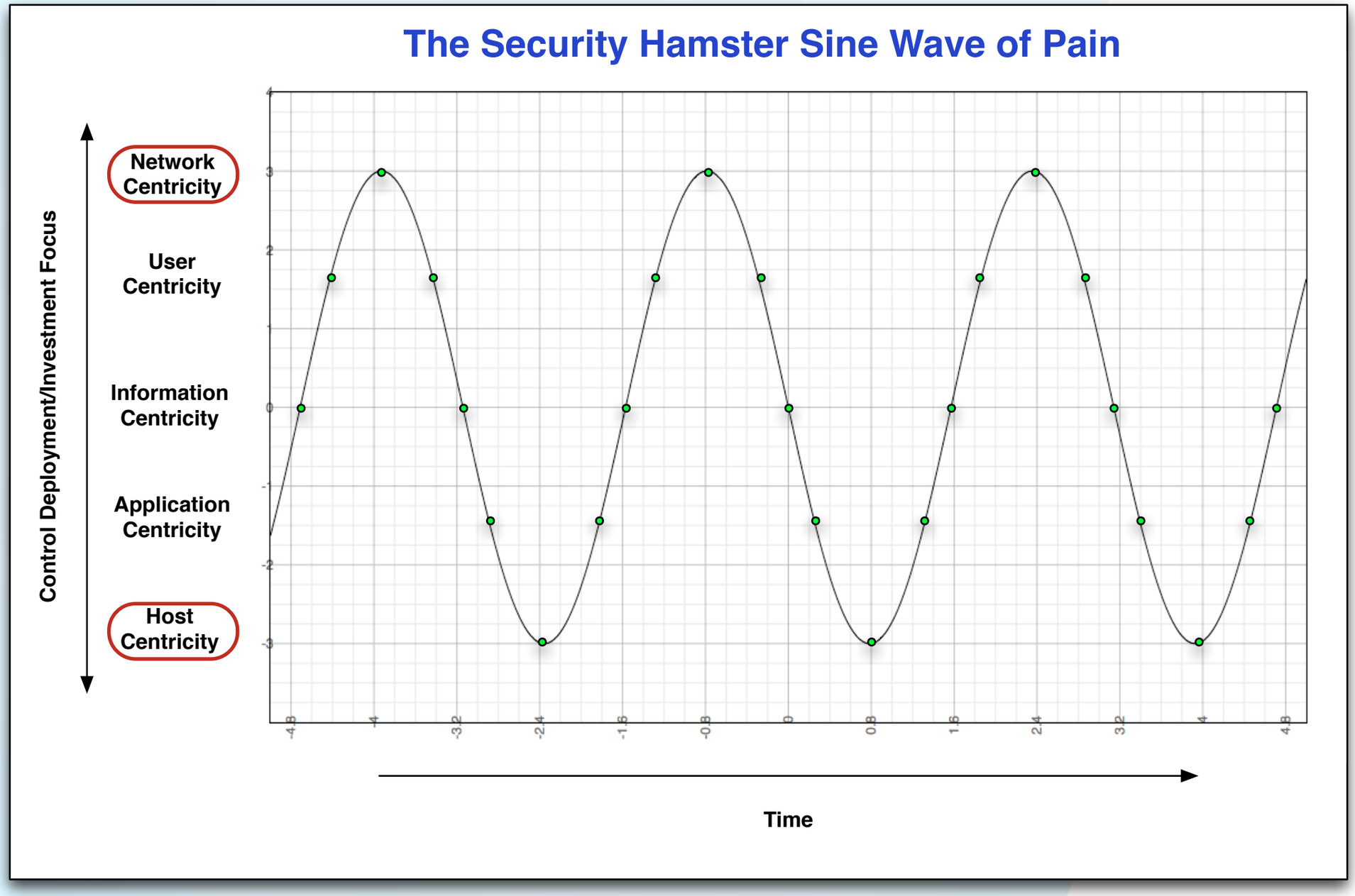
Data

Bandwidth

	Developers	Security
1995	CGI, PERL	Network firewalls, SSL
1997	ASP, JSP	Network firewalls, SSL
1998	EJB, J2EE, DCOM	Network firewalls, SSL
1999	SOAP, XML	Network firewalls, SSL
2001	Rest, SOA	Network firewalls, SSL
2003	Web 2.0	Network firewalls, SSL
2007	Cloud Computing	Network firewalls, SSL

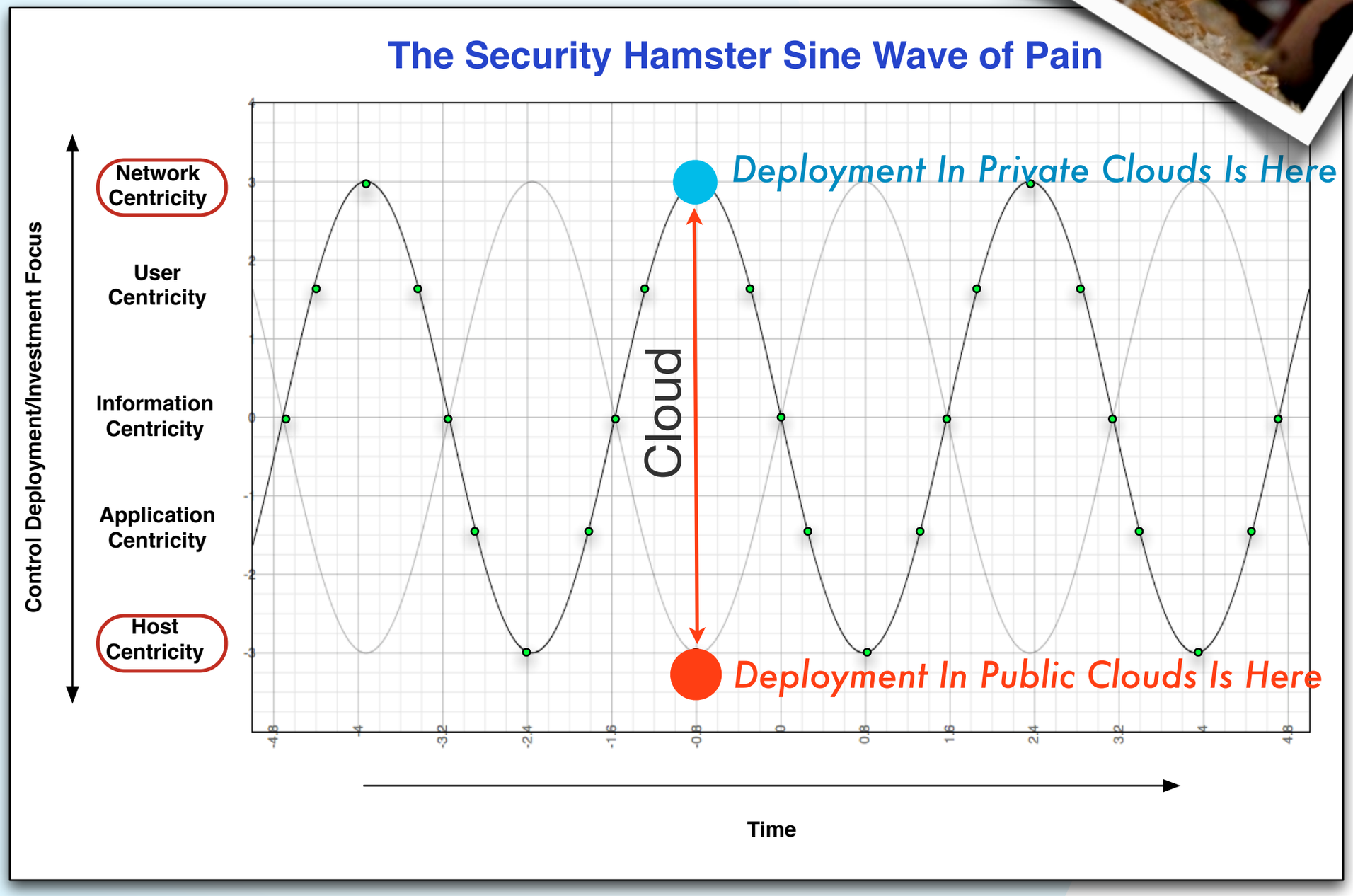
** Credit: Gunnar Peterson*

Revenge Of The Hamsters...



** With Apologies to Andy Jaquith & His Hamster...*

Revenge Of The Hamsters...



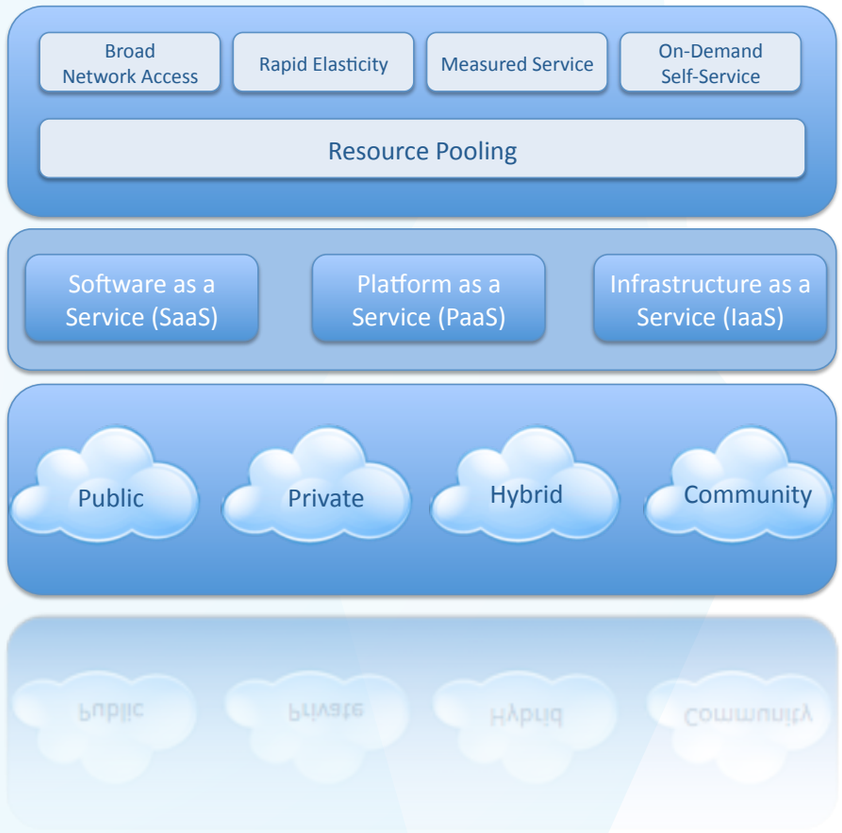
** With Apologies to Andy Jaquith & His Hamster...*

Our Focus In Contrast

+ **Private cloud.** The cloud infrastructure is operated **solely for an organization.** It may be managed by the organization or a third party and may exist on premise or off premise.

+ **Public cloud.** The cloud infrastructure is **made available to the general public** or a large industry group and is owned by an organization selling cloud services.

Visual Model Of NIST Working Definition Of Cloud Computing
http://www.csrc.nist.gov/groups/SNS/cloud-computing/index.html



Private OR "Enterprise" Cloud To Boldly Go...

Private: Leverage Virtualization To Yield Higher Efficiency In Service Delivery, Agility And Meet Existing Security And Compliance. Infrastructure Exposed.

General Preservation Of Existing Architectural Blueprints But With Virtualized/Converged Infrastructure. Primarily Hardware Infrastructure Enabled & Enterprise-Class Virtualization Layers

Public: Fundamental Re-Architecture Of Application, Operations & Service Delivery Leveraging Virtualization & Automation. Massive Abstraction.

Focuses On Scale, Lower Costs And Homogeneity At Infrastructure Layers. Primarily Software Enabled

Deconstruction

Infostructure

Applistructure

Metastructure

Infrastructure

- **Content & Context** -
  Data & Information
- **Apps & Widgets -**
  Applications & Services
- **Glue & Guts** -
  IPAM, IAM, BGP, DNS, SSL, PKI
- **Sprockets & Moving Parts** -
  Compute, Network, Storage

Idempotent Infrastructure

Idempotent?

idempotent |ˈīdemˌpōtənt| Mathematics

adjective
denoting an element of a set that is unchanged in value when multiplied or otherwise operated on by itself.

noun
an element of this type.
ORIGIN late 19th cent.:
from Latin ***idem 'same'*** + **potent**

*In computer science, the term **idempotent** is used to describe methods or subroutine calls that can **safely be called multiple times**, as invoking the procedure a single time or multiple times **has the same result**;*

Infrastructure

* wikipedia: http://en.wikipedia.org/wiki/Idempotence

Idempotency & Cloud

+ Homogeneity Provides Foundation For Scale [out]
+ Often Implies Commodity Hardware
+ Maximize Density & Modularity Of Resources
+ Constant Deployment Model (Agile) Of Software Driven “Infrastructure”
+ Code As Infrastructure (and vice versa) yields automation

Infrastructure

Attack Of the Stack

- Some Examples Of The Growing Number Of Available Cloud “Operating Systems” and resultant APIs:
  - OpenStack.org
  - Cloud.com CloudStack
  - Citrix XenCloud
  - VMware vCloud
  - Enomaly ECP
  - RedHat Cloud Foundations
  - Nimbula Director
  - Eucalyptus Enterprise Edition
- **Open Source vs. Open Core vs. Proprietary...The business models matter!**

Compute Architecture - Cores & Memory

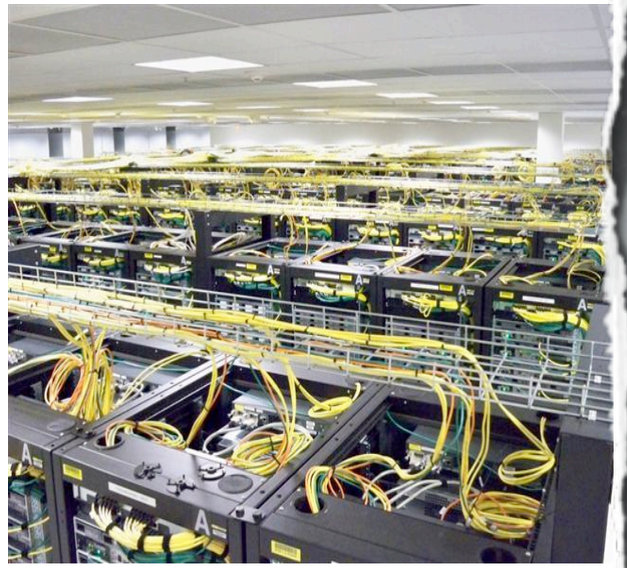
- Compute Fabrics:
  - Commodity, “Engineered” or Proprietary/Specialized
  - Lots of CPUs vs Fewer CPUs With Lots Of Cores
  - CPU vs GPU (or otherwise)
- Low Power, Low BTU, Highly Dense
- Dedicated vs Huge Shared Memory
- Management via RESTful HTTP API

Infrastructure

"Datacenter Networks Are In My Way*"

Agenda

- Where Does the Money Go?
  - Is net gear really the problem?
- Workload Placement Restrictions
- Hierarchical & Over-Subscribed
- Net Gear: SUV of the Data Center
- Mainframe Business Model
- Manually Configured & Fragile at Scale
- Problems on the Border
- Summary



* James Hamilton, AWS

He's Right...

+Hadoop has different requirements than a LAMP stack than does Facebook than does Google...

+The application and use case are extremely relevant

+AWS has Hadoop clusters, GPU clusters and "general" CPU clusters

+The networks that power them aren't the same, either...

Network Architecture

- **Where's the Network? Software vs Hardware: VMM-Integrated, Nexus 1000v, Open vSwitch, OpenFlow, Nicira, Or Hybrid Models...**

- **Hugely Abstracted Networks Create Challenges With:**

  - **Topology** - L2/L3 Design, Multicast/Broadcast, STP, etc. New protocols: FabricPath, OTV, TRILL, HIP, LISP...

  - **Security** - Presentation Of Hooks To Interconnect/Segment, Visibility, Management

  - **Mobility** - Dynamism Stresses Resource: Naming, Location, Addressing, etc.

  - **Performance** - I/O, Packet Ping Pong, QoS

- **Revenge Of The Meshed Overlay VPN & PKI**

  - Need for PKI, Link (Physical & Logical) Encryption, Authentication, Tagging

Infrastructure

Network Architecture (Cont.)

- **Big, "Dumb," Flat, Fast L2 Networks Vs. Next Generation Of Classical Core|Distribution| Access**
  - Heavily Virtualized High Density, Low Latency, Non-Blocking, Line Rate, 10+Gb/s
  - Full Bisection Bandwidth vs. Statistical Multiplexed & Over-subscribed
  - Segmentation, Multi-tenancy & Scale: Abstracted Into VMM or (p)VLANs/VRF or by separating data/control/flow planes
  - Programmable & Open vs. Fixed & Proprietary
  - Data-Only Versus Converged Data & Storage
- **RESTful HTTP APIs and Exposed Interfaces for Automation/Provisioning/Orchestration/ Instrumentation**

Infrastructure

VMs : The New DMZ?

+ Practically, The VM Boundary **Is** The Emerging Atomic Unit And Therefore Is The Logical Perimeter. For Now.
+ Ultimately We'll See A New Measure As VM's & The Servers They Replaced Give Way To PaaS/Language Abstractions & "Workload" Gets More Defined*
+ The Cloud OS Platform Networking Can Clearly Be A Fantastic Differentiator Or A Huge Limiter For Delineating Policy Boundaries

Infrastructure

*E.g. Heroku uses "Slugs" & "Dynos"

Storage

- Local disk vs NAS/SAN/Object-Based
- Storage Can Be Exposed via RESTful/SOAP APIs Or Volumes
- Persistent vs Non-Persistent
- Volume/Bucket/File Size Limits
- Converged (FCoE) Storage & Networking
- I/O and Performance
- Impacts On Application Architecture
- Storage “Mobility”
- BCP/DR/Resilience/Recovery
- Isolation/Security/Forensics in Multi-Tenant Environments

Protocols: That Pesky TCP/IP

- Van Jacobson Summed It Up Well (and I Paraphrase)*:
  - The Cloud Today Is Like The ARPAnet Of Yesterday: "...At The Outset The New Network Looked Like An Inefficient Way To Use The Old Network."
  - Things that people are doing with cloud and what the cloud does inside are different; use cases are the side effect.
  - Ubiquitous "Any-To-Any" Communication Is Not What TCP/IP Was Created For
  - Today's Networking [Protocol] Problems Don't Reflect An Architectural Failing, But Rather TCP/IP's Success In Creating A World Rich In Information & Communication; **TCP/IP Is A "Success Disaster"**

Metastructure

* *Van Jacobson: A New Way To Look At Networking | 2006 | http://bit.ly/5jGte*

The Solutions Are Not Flawed, The Problems Have Changed

"Developers Are New Kingmakers"

Applistructure

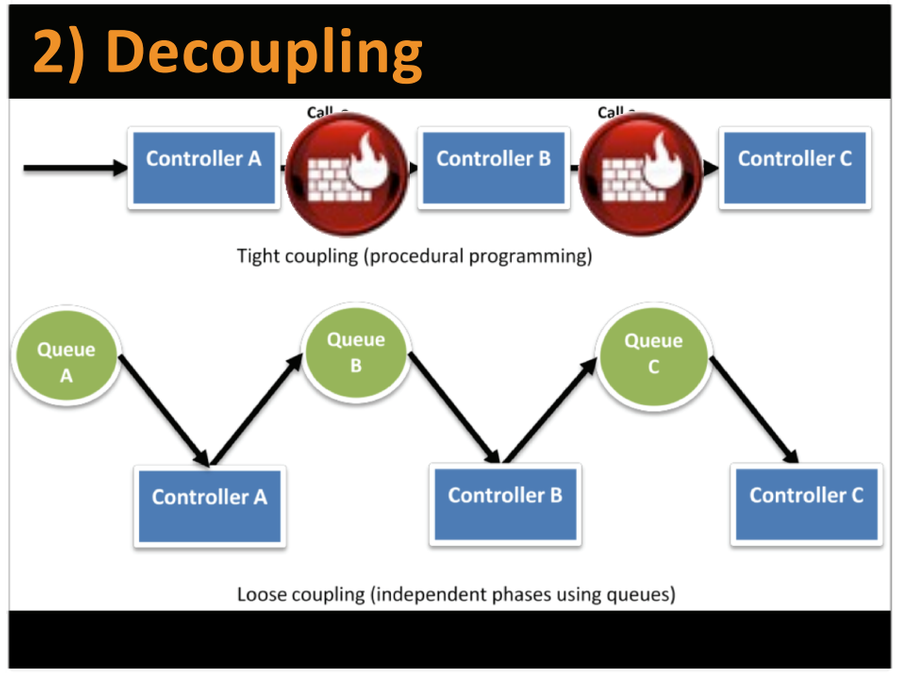
"The high level of usage (and its somewhat clandestine nature) was illustrated at a conference earlier this year when one CIO noted that he had ***gone through the expense reports turned into him for reimbursement and found 50 different cloud computing accounts being used by developers*** in his organisation. The reason? It's ***a lot easier for a developer to obtain computing resources from a public cloud provider*** than to undergo the extended waits typical of the existing compute environments."*

* *B. Golden "Cloud computing: The truth about what runs on Amazon"*

Application Architecture

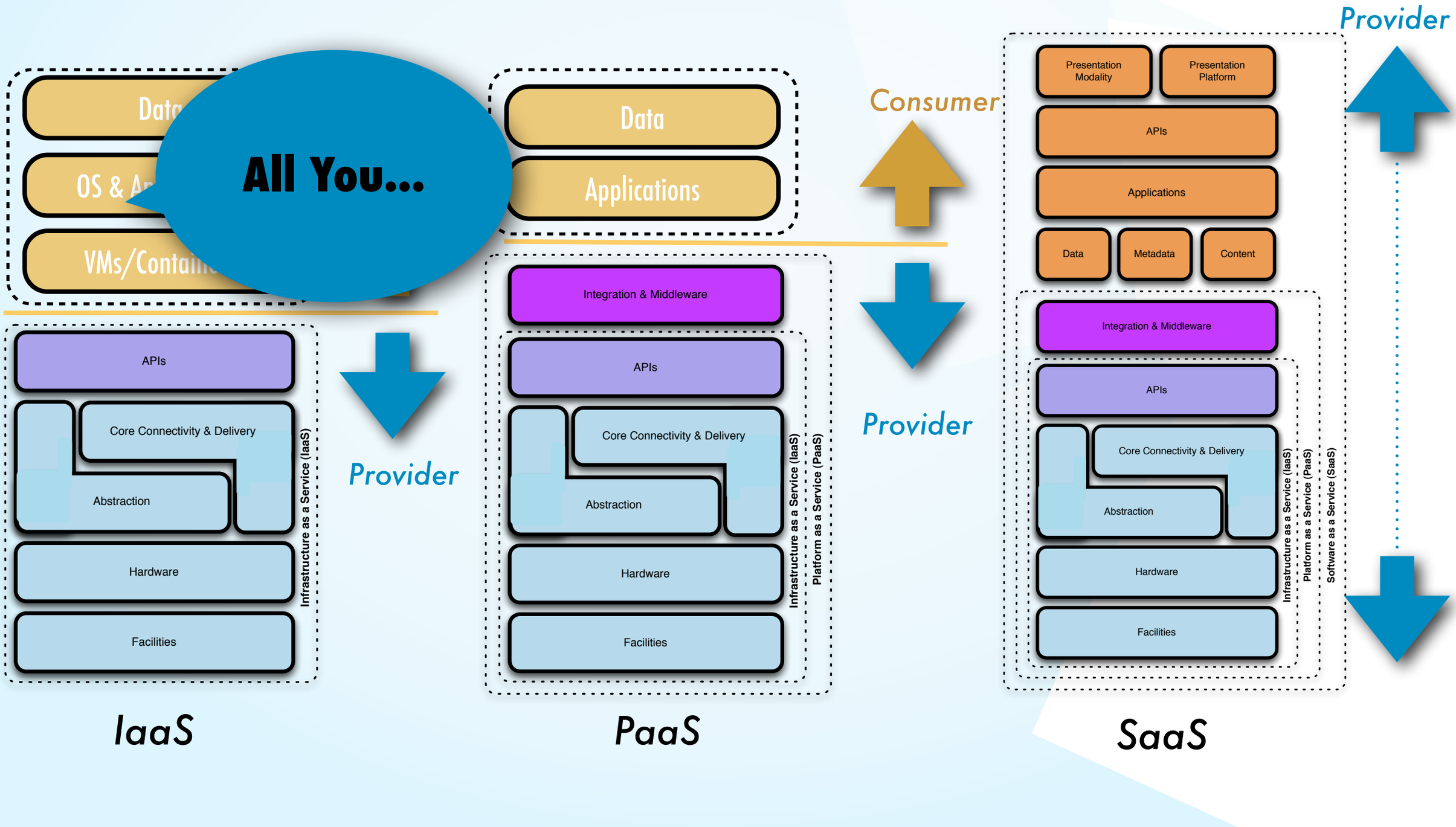
Applistructure

- Architecturally, The Classical n-Tier/DMZ Segment & Asset Grouping Methodology May No Longer Apply
- Applications Are Likely Not Composable In This Manner As They Are More Topologically & Infrastructure-Insensitive Than Ever (*See Flat L2 designs*)
- **In Many Cases, Applications Must Be Completely Rewritten To Leverage Public Cloud & The Security Models Must Be Adjusted**
- Dumb Networks Equal Dumber Security Or At Least Less Options/Capabilities; Workload Sensitivity vs. Network Capability Is Critical
- People Still Write Crappy Code, No Matter How Good, Abstracted Or Elastic The Infrastructure Is
- Continuous Deployment Models (Agile & DevOps) Are Taking Root
- **Application Security Is Even More Important Than Ever**



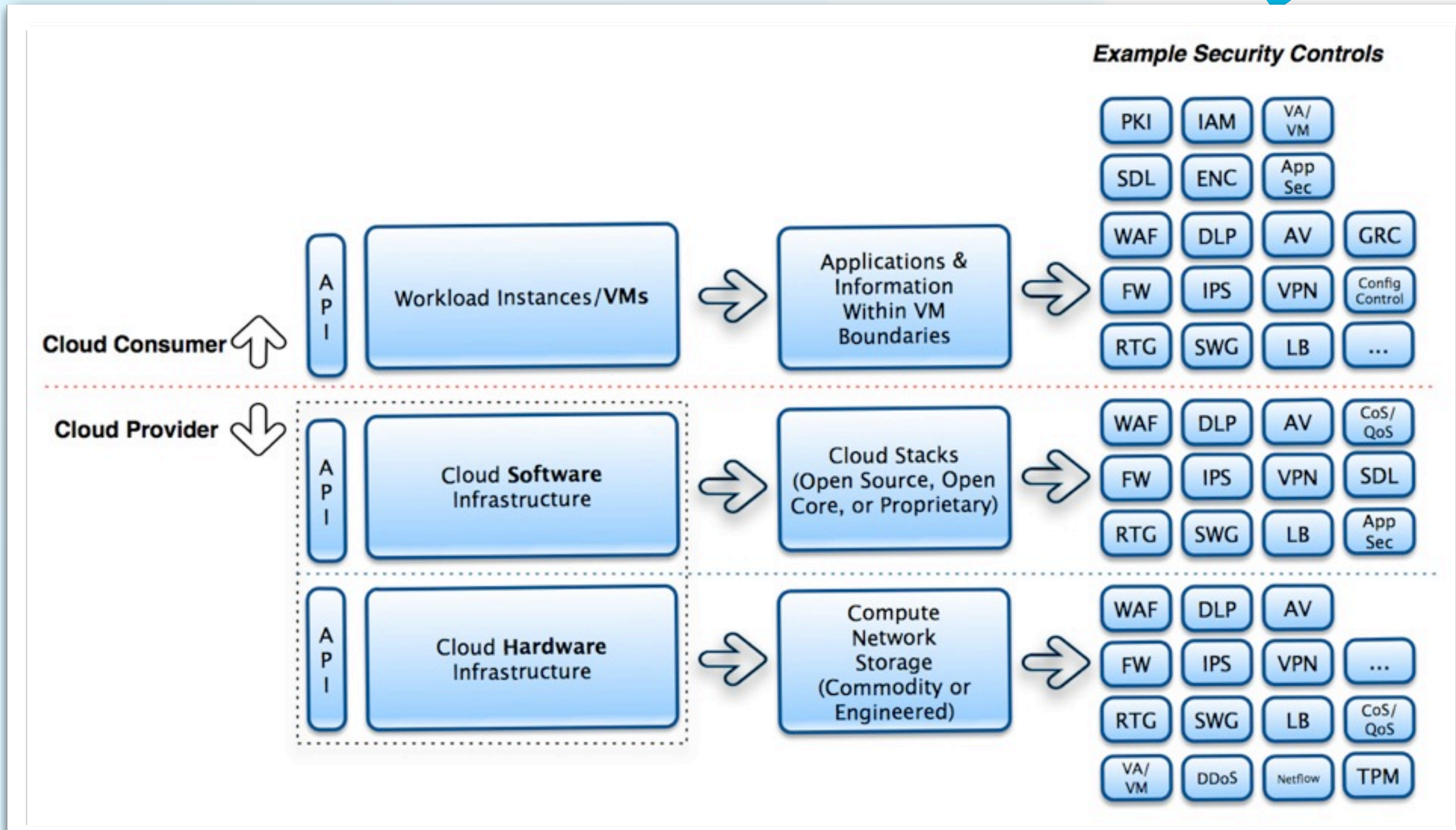
Simone Brunozzi - http://bit.ly/aws_architect-cloud

Cloud Security Model



So Is The Problem That We Don't Have Enough Security...

...Or That We Have Too MUCH Security?



Deconstruction

Infostructure

Applistructure

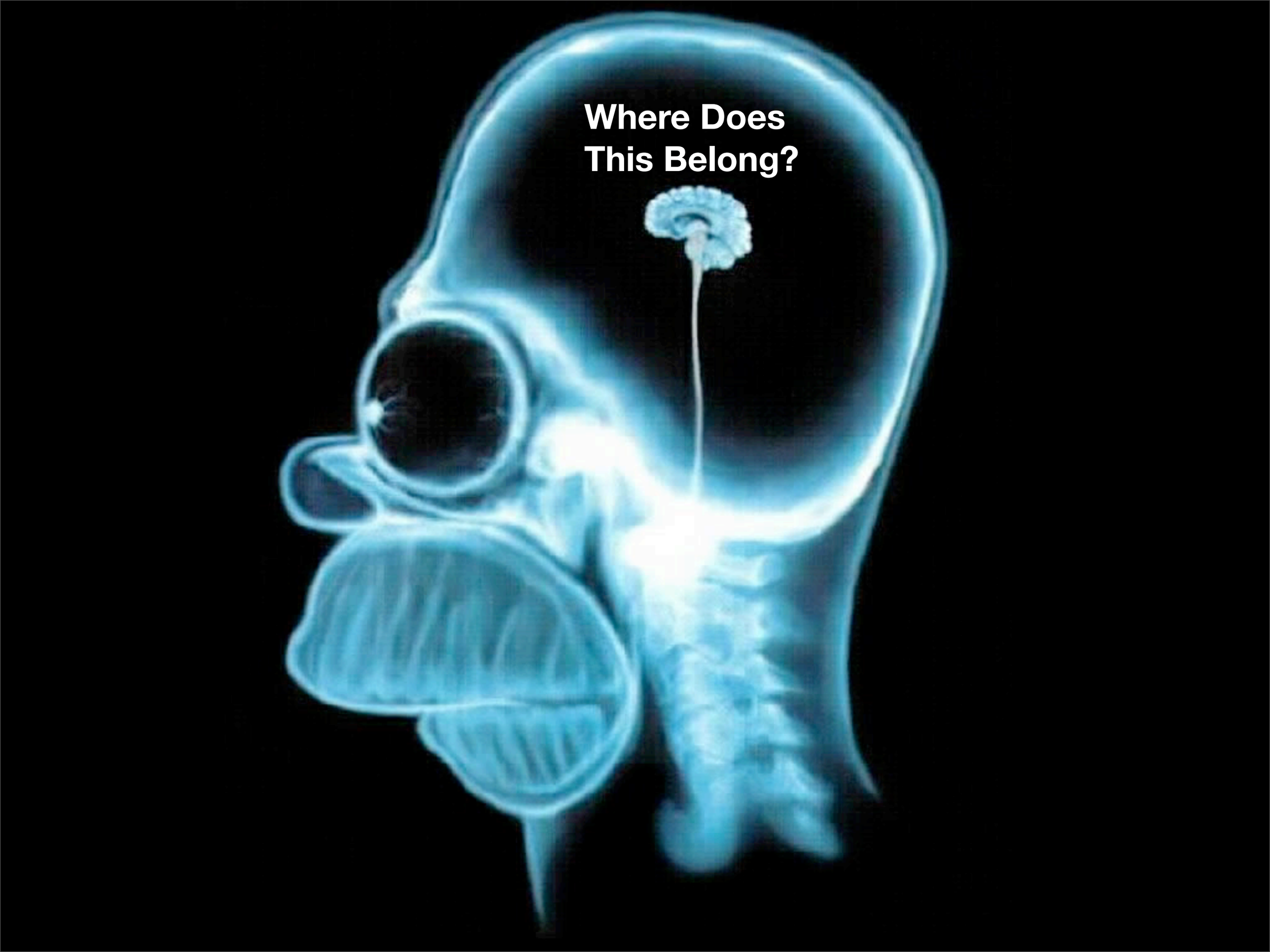
Metastructure

Infrastructure

→ Information Security

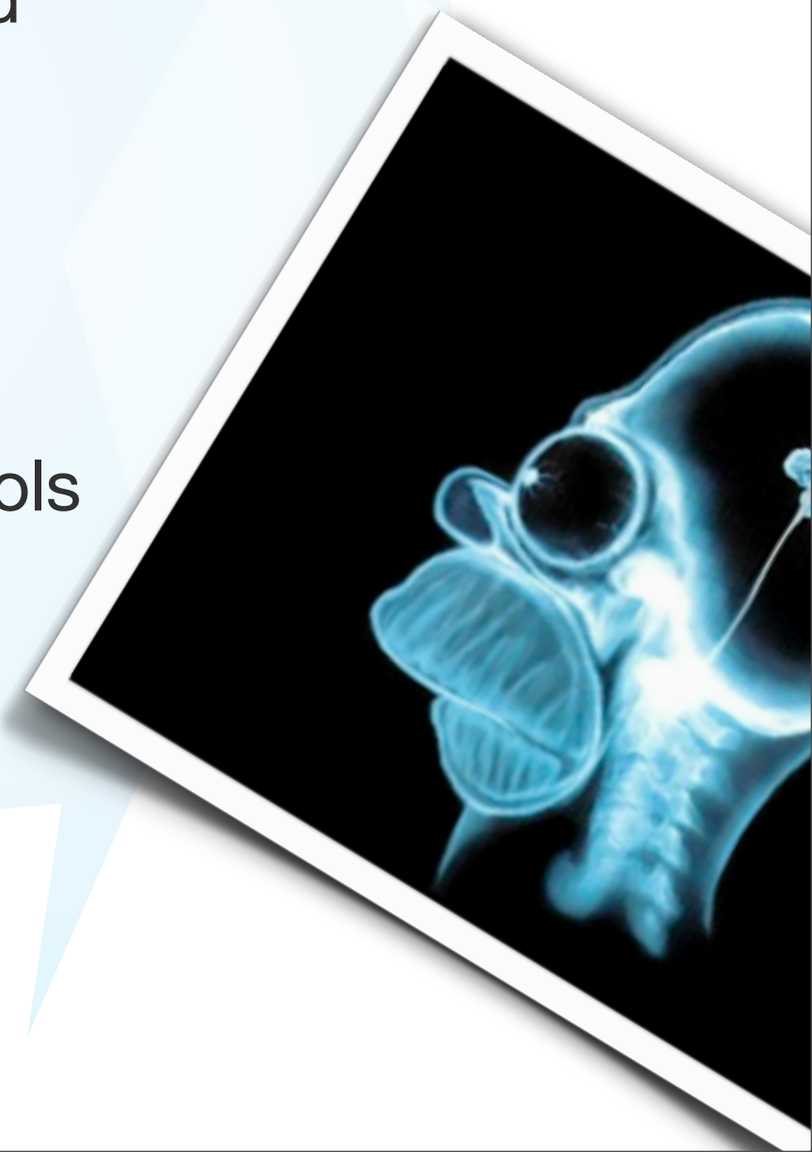
→ Application Security

→ Network Security
Host-based Security
Storage Security


Where Does
This Belong?

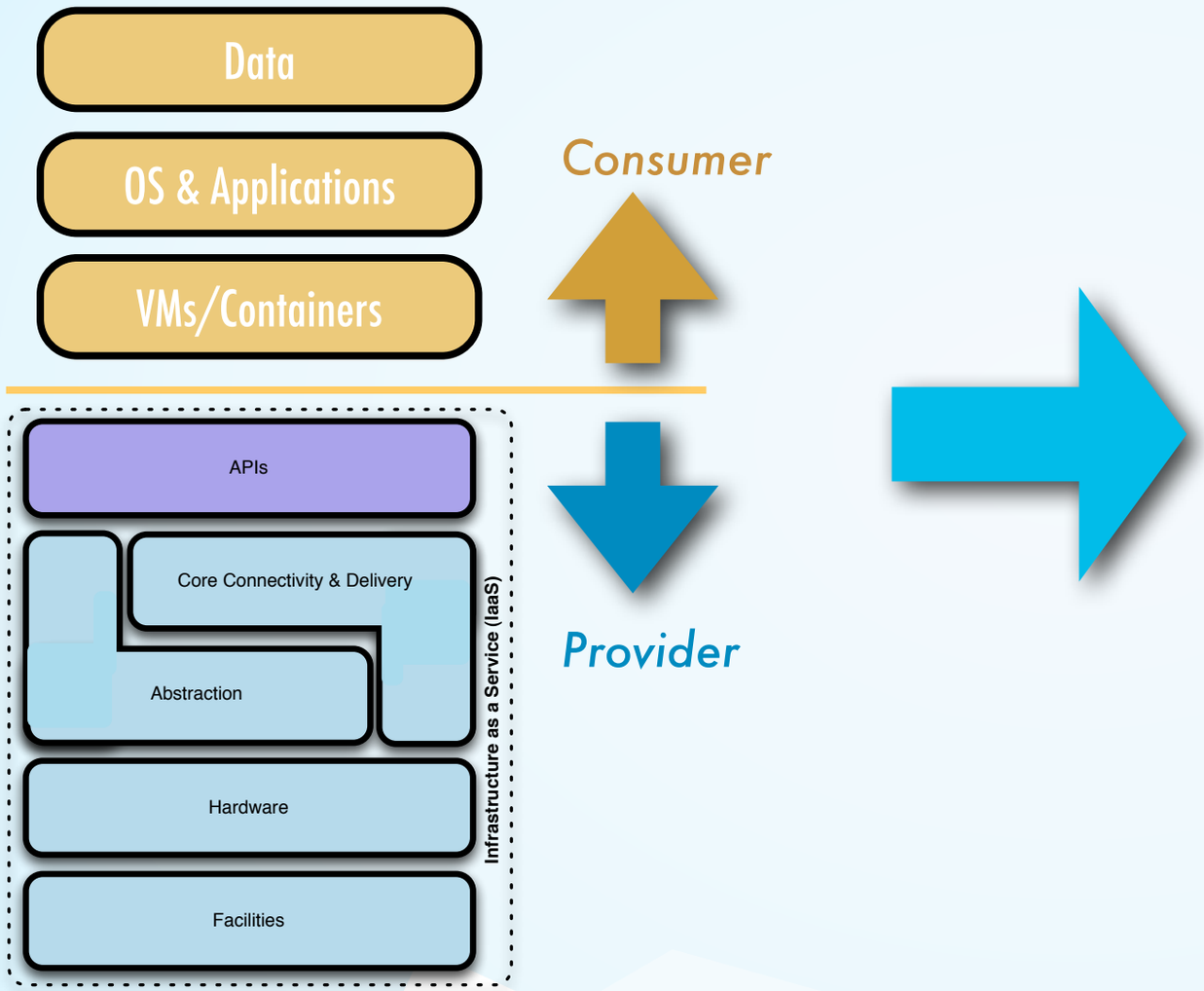
We Need Hybrid Security To Deal With Hybrid Computing

+ Cloud Blurs The Lines Between Network, Application and Information Security
+ We Need To Opportunistically Leverage Cloud Models Based On Use Case (Public, Private, Virtual Private...) SECURELY
+ ...Which Requires Simplified and Consistent Policies & Management Without Regard To Distinction Between Physical Or Virtual Controls
+ Solutions Across the Stack Must Leverage Common Telemetry
+ We MUST Engineer Automation Via APIs Into Security Controls To Enable Scale



It's The Integrator's Dilemma...

Abstraction Has Become a Distraction



Not Much You Can Do Below the Line...

The Focus Is Here:

- Building Survivable Systems
- Building Secure Apps
- Securing Data

Infostructure

Survivable Systems

http://ingesidee.de/resourceviewer.php?pgid=63&lang=en&subpage=1&module=content

Survivability?

+ Delivery Of Essential Services and Preservation Of Essential Assets Capable Of Fulfilling Mission Objectives

What Cloud Means To ~~Security~~ Survivability

+ Focus on sustaining the business/mission in the face of an ongoing attack; requires a **holistic perspective** (not siloed)
+ Depends on the ability of networked systems to **provide continuity of essential services, albeit degraded**, in the presence of attacks, failures, or accidents
+ Requires that only the **critical assets need the highest level of protection**
+ **Complements current risk management** approaches that are part of an organization's business practices
+ Includes (but is broader than) traditional "security"

Architecting for the Cloud

1) Design for Failure

2) Decoupling

3) Elasticity

4) Security

5) Break Costraints

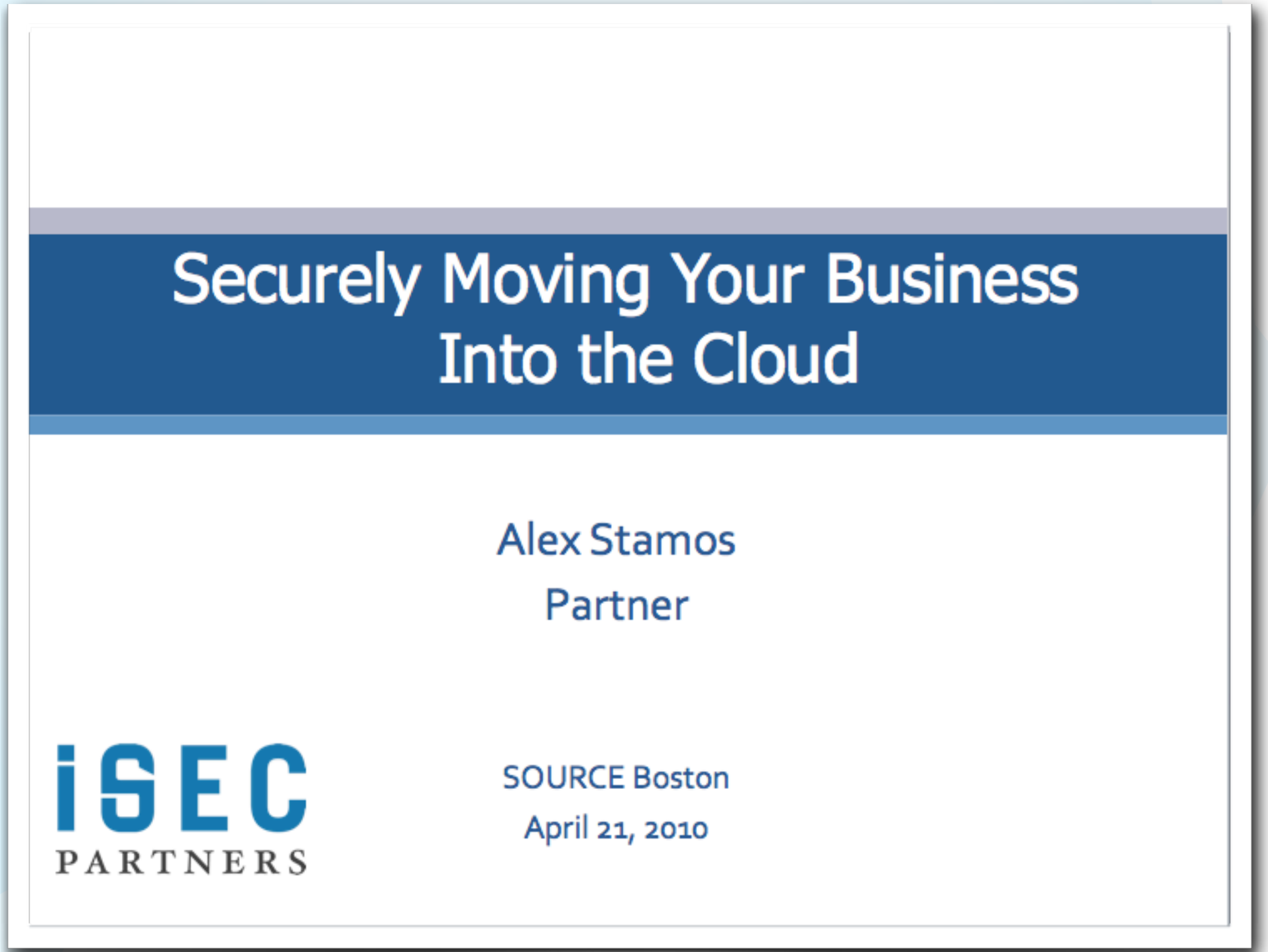
6) Think Parallel

7) Different Storage Options

1) Design for Failure

- Backup/Restore Strategy
- Be impervious to Reboot/Relaunch
- Move in-memory sessions to Data Store
- Use Availability Zones, Distribute EC2
- Use Elastic Load Balancer
- Use Relational Database Service (RDS)
- Use Elastic IP

Or In Alex Stamos' Words...



Or In Alex Stamos' Words...

Takeaways

- Current conventional wisdom on cloud computing is missing the point
- You cannot securely move into the cloud without re-writing your software
- Secure cloud applications "collapse the perimeter"
- Properly going through this process should leave you more secure than before

iSEC PARTNERS

2

Or In Alex Stamos' Words...

What is the alternative?

- Go Flat
- Collapse the Perimeter
- Use cloud glue services
- Enforce access control via cryptography

iSEC PARTNERS

Sounds Easy Enough, But...

- There's A Big Difference Between Greenfield and Existing Applications
- The Software Architecture Changes Dramatically, So The Security Models & Solutions Need To Also
- As We Move Greenfield Applications To Public Cloud, We're Forced To Build More Survivable Systems Because We Can't Depend On The Provider

IaaS & VM's Represent the Problem...

+ The reason we have virtualization in the first place is because the OS failed at things like process isolation & user/kernel mode separation

+ Now we've squeezed the balloon and moved that duty to a third party and with another layer of indirection (the VMM)

That Means...

+ Assume All Environments Are Hostile And Isolation in Multi-Tenancy Will Fail
+ Prepare for the move to JEOS (Just Enough O.S.) or NoOS (No O.S., i.e. PaaS and SaaS)
+ Choose Open Protocols and Stacks & Leverage Introspection and Ensure Robust Instrumentation & Telemetry At All Layers
+ Enable Robust Monitoring And Forensics
+ Utilize the Three R's and a **SYSTEM-FOCUSED** Design Philosophy

The 3 R's: Resistance, Recognition & Recovery

- **Resistance**: Capability Of System To Repel Attack
- **Recognition**: Capability Of System To Detect Attack and Evaluate Extent Of Damage/ Compromise
- **Recovery**: Capability To Maintain Essential Services and Assets During Attack, Limit Extent Of Damage and Restore Services

If Our Systems Aren't Survivable Why Do We Expect Our Information To Be?

Infostructure

Information Centricity

Distilled Principles of Information-Centric Security

+ Information (audio/video/ data) must be self describing and defending

+ ...Structured Or Unstructured

+ Policies and controls must account for business context.

+ Information must be protected as it moves between silos, between locations, and changing business contexts.

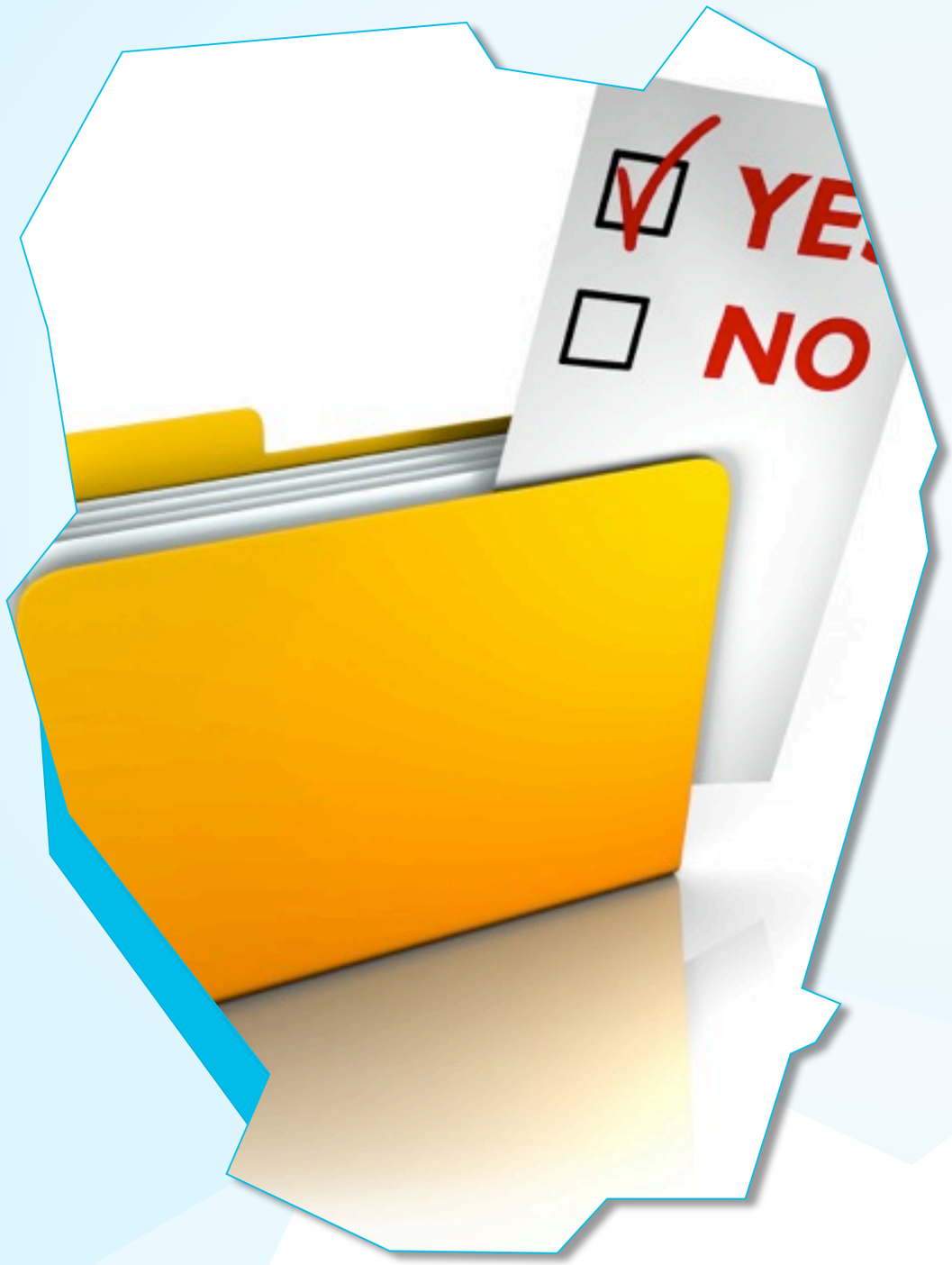
+ Policies must work consistently through the different defensive layers and technologies we implement.

**Rich Mogull - Securosis*

Risk Assessment & Threat Modeling

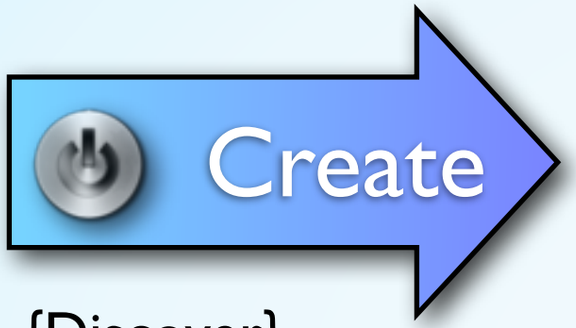
- Risk Assessment and Threat Modeling are NOT Information Alchemy, But Do Require Lots Of Work
- Some Frameworks To Choose From:
  - Microsoft **STRIDE/DREAD**
  - Carnegie-Mellon **OCTAVE**
  - RMI **FAIR**
- *"How Can You Have Your Pudding If You Don't Eat Your Meat?"*

The 6 Key Questions



1. How would we be harmed if the asset became public and widely distributed?
2. How would we be harmed if an employee of our cloud provider accessed the asset?
3. How would we be harmed if the process or function was manipulated by an outsider?
4. How would we be harmed if the process or function failed to provide expected results?
5. How would we be harmed if the information/data was unexpectedly changed?
6. How would we be harmed if the asset was unavailable for a period of time?

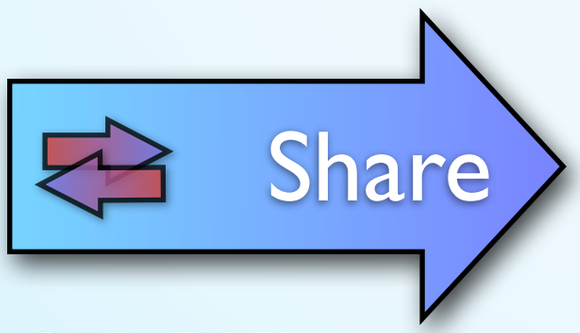
Managing Information Across Its Lifecycle*



{Discover}
Classify
Assign Rights

Access Controls
Encryption
Rights Management
Content Discovery

Activity Monitoring
and Enforcement
Rights Management
Logical Controls
Application Security



CMP (DLP)
Encryption
Logical Controls
Application Security

Encryption
Asset Management

Crypto-Shredding
Secure Deletion
Content Discovery

*Rich Mogull - Securosis

Information Centric Solutions For Cloud

Labels

**Rich Mogull - Securosis*

Information Centric Solutions For Cloud

DLP

**Rich Mogull - Securosis*

Information Centric Solutions For Cloud

EDRM

**Rich Mogull - Securosis*

Information Centric Solutions For Cloud

Encryption

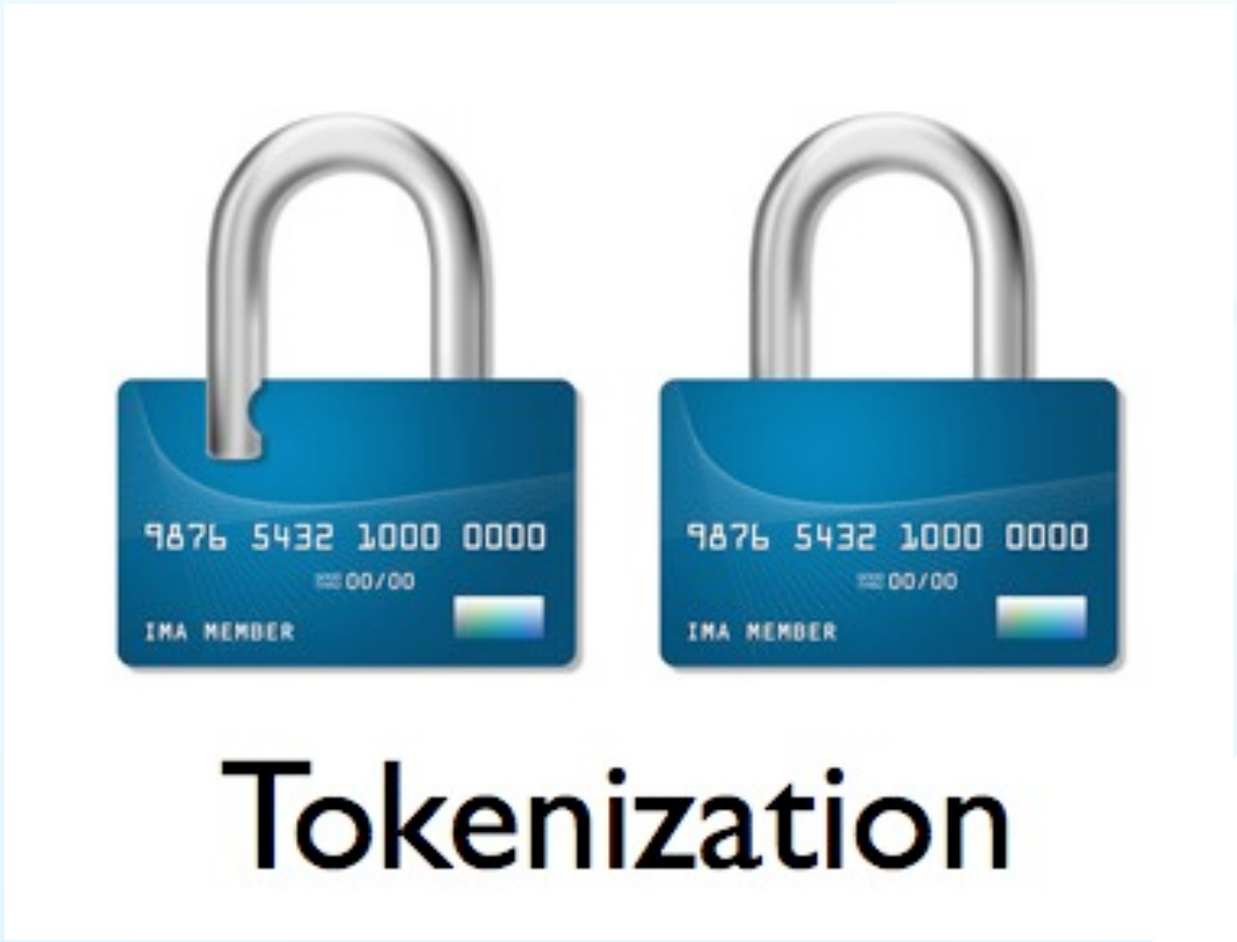
**Rich Mogull - Securosis*

Information Centric Solutions For Cloud

IAM

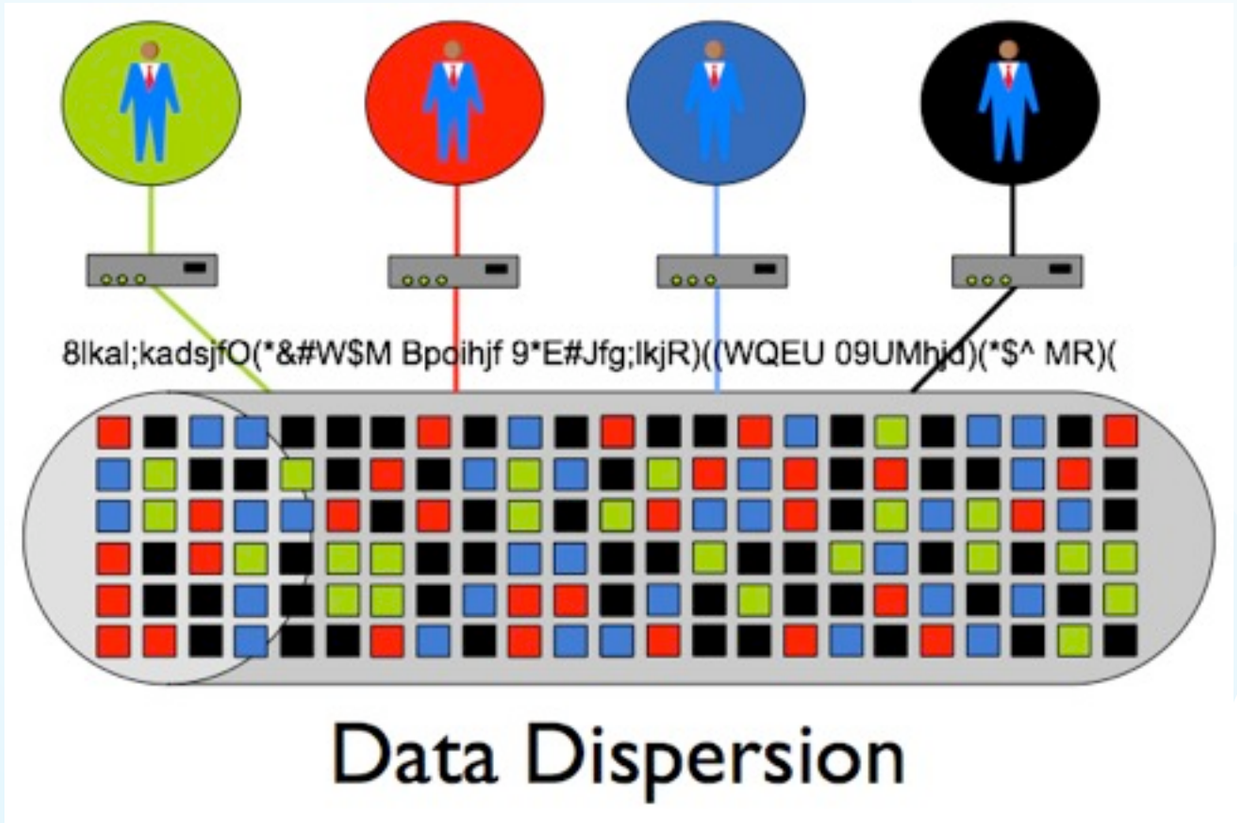
*Rich Mogull - Securosis

Information Centric Solutions For Cloud



*Rich Mogull - Securosis

Information Centric Solutions For Cloud



**Rich Mogull - Securosis*

Where This Take Us

Infostructure

+ Content Analysis Fully Integrated Into Both Productivity And Transaction Applications As Well As Datastores

+ Rights (And Thus Enforcement) Applied At The Point Of Creation (Or Discovery,) At The Data-Element Level

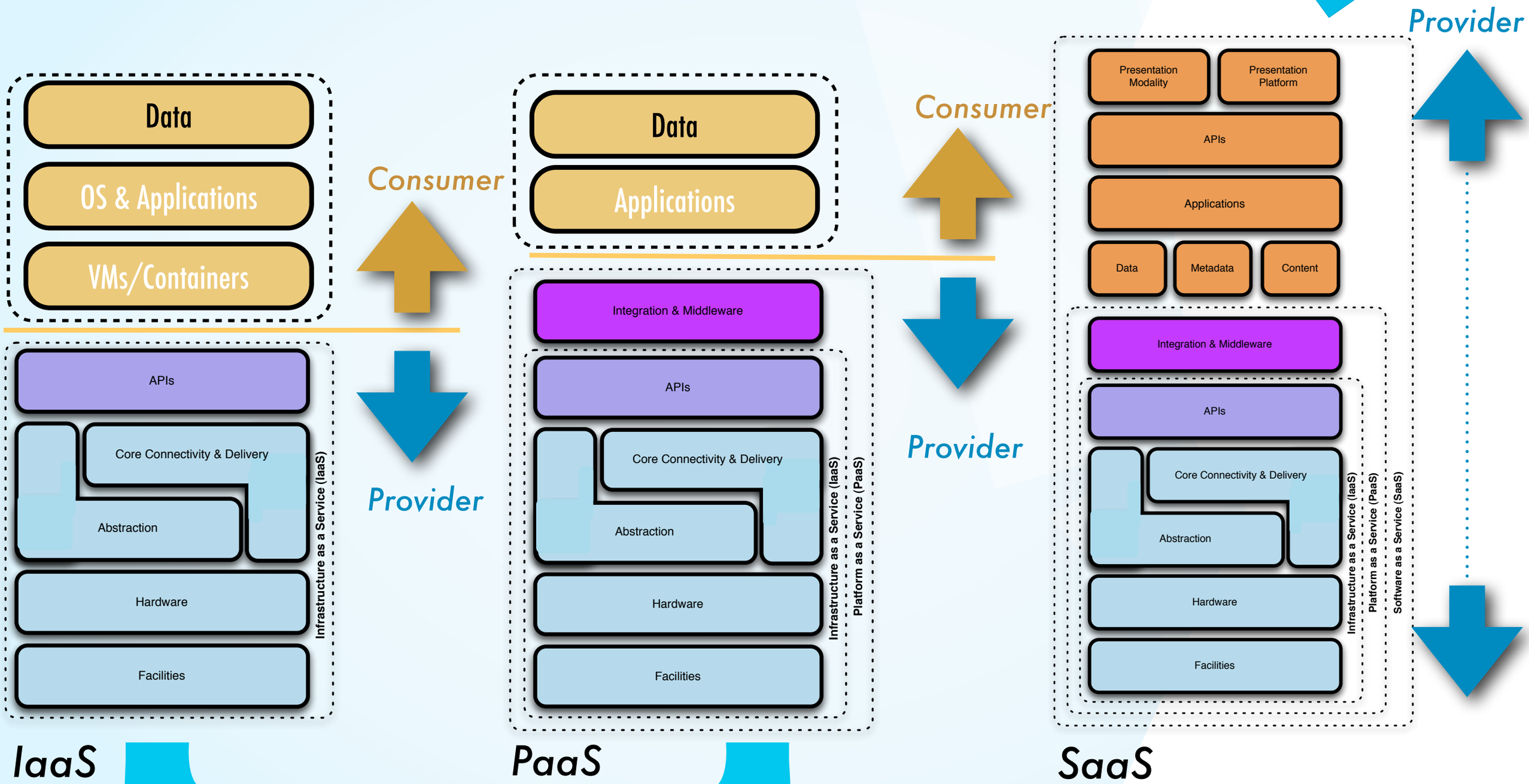
+ Choke Points Between On-premise, Off-premise, And Between Cloud Services Enforce Policies At The Data Level, Enforced By Encryption/DRM

+ Rights Transfer And Enforcement Are Maintained Between State Changes

+ Don't Expect Your IaaS Provider To Do Any Of This For You; They Are Blind (By Design) To Most Of Your Data

**Rich Mogull - Securosis*

Delivery Model Mapping Revisited



Security Is Shared

Securing the "Centers Of Data" of the Future

+ Change Is Hard; **Requires Fundamental Changes In Infrastructure, Programmatic, Management, Network, Application & Security Architectures**

+ The **Security Policies** That Govern Information And Assets **Need To Travel With Them**; We Need Consistency In Metadata

+ **Infrastructure Must Gain Intelligence & Context of Info- & Applistructure & Vice-Versa** Through Consistent, Standards-Based Telemetry, Correlation & Disposition

+ Consumers & Data/Application Owners Must **Make Centers Of Data "Survivable" & Focus On Information-Centric Security**

The Punchline

+ In The Simplest Of Terms, Using Cloud Means Imagining Applications & Information Across All Tiers Have The Potential To Be Connected Directly To The Internet...
+ We Can't Trust The Provider, So We Must Engineer Security Into Design Patterns Across The Entire Stack
+ Any "Dumb" Component In The Stack Compromises The Integrity Of the Entire Stack...
+ APIs, Intelligence and Automation EVERYWHERE

Contact...

- Christofer Hoff
- www.rationalsurvivability.com/blog
- choff@packetfilter.com [not work]
- hoffc@cisco.com [work]
- +1.978.631.0302
- @beaker [The Twitters]

Image Attribution

- Skull Cloud: http://www.clippingimages.com/blog/tutorial-making-a-skull-shaped-cloud/
- Prince: http://www.virginmedia.com/images/prince-lovesexy-gal.jpg
- Star Trek Tribbles: http://i84.photobucket.com/albums/k3/NonStopPop/TheTroubleWithTribbles.jpg
- Jersey Shore: http://images.huffingtonpost.com/2010-03-01-shore.jpg
- Trebuchet:
- Trebuchet, Catapult, ballista: http://bit.ly/bCQTL2
- Trebuchet:
- Chinese Army: http://www.life.com/image/75878629
- Potemkin Village: http://www.flickr.com/photos/wili/274828493/
- Copernican: http://justanapprentice.files.wordpress.com/2009/11/copernicus-universe.jpg
- Floppy Disk: *http://www.flickr.com/photos/yaal/162100723/*
- Switchboard Operator: http://afeatheradrift.files.wordpress.com/2009/06/lily-tomlin-telephone-operator.jpg?w=300
- USB Hub: http://technabob.com/blog/wp-content/uploads/2009/05/usb_lego_hub.jpg
- Hall Of Mirrors: http://historyofeconomics.files.wordpress.com/2008/08/hall-of-mirrors.jpg
- Ritalin: http://commons.wikimedia.org/wiki/File:Ritalin-SR-20mg-1000x1000.png
- Obama Vulcan:http://www.flickr.com/photos/alexisalex/3519670022/ - Shan Carter
- Apple 1984: http://upload.wikimedia.org/wikipedia/en/2/22/Ad_apple_1984_2.png
- Gangsta Chimp: http://www.mattcioffi.com/samples/gangstaChimp24.jpg - Matt Cioffi
- Nostradamus: http://omarab.files.wordpress.com/2009/12/nostradamus-1503-15663.jpg
- Castle: http://www.jokerjitsu.com/images/medieval-castle.jpg
- Apathy: http://www.flickr.com/photos/comiccharacters/3792479746/sizes/l/
- W.O.P.R. http://www.hexkey.co.uk/lee/log/media/2009/08/wargames-wopr-med.jpeg
- Train: http://cache.gawker.com/assets/images/jalopnik/2009/02/Balla-train.jpg
- Cloud Umbrella: http://www.likecool.com/Gear/Design/Cloud%20Umbrella/Cloud-Umbrella.jpg