Architectures for
the Next Generation
Internet
and the Future Networks

RAJ JAIN
Washington University in Saint Louis
Saint Louis, MO 63130
Jain@cse.wustl.edu
http://www.cse.wustl.edu/~jain/
A tutorial presented at ADCOM 2012
Bangalore, India, December 16, 2012
Overview

1. Why Next Gen?
2. Internet 3.0
3. Content Centric Networks
4. Software Defined Networks
5. Routing Architectures: Open Flow, ID-Locator Split Proposals
6. Next Generation Testbeds
Future Internet Projects

- In 2005 US National Science Foundation started a large research and infrastructure program on next generation Internet
- Q: How would you design Internet today? Clean slate design.
- “Future Internet Design” (FIND): 48+ projects
  - Stanford, MIT, Berkeley, CMU, …
  - “An Architecture for Diversified Internet” at WUSTL
- “Global Environment for Networking Innovations” (GENI): 29+ projects
- European Union: 7th Framework program
- Japan: AKARI (A small light in the dark pointing to the future)
- China, Korea, Australia, …20+ countries

Key Problems with Current Internet

1. **Security:**
   Fundamental architecture design issue
   Control + Data are intermixed
   Security is just one of the policies.

2. No concept of **ownership**
   (except at infrastructure level)
   Difficult to represent organizational,
   administrative hierarchies and
   relationships. Perimeter based.
   \[\Rightarrow\] Difficult to enforce organizational
   policies

![Diagram showing trusted and untrusted realms]
Problems (cont)

3. Identity and location in one (IP Address)
   Makes mobility complex.

4. Assumes live and awake end-systems
   Does not allow communication while sleeping.
   Many energy conscious systems today sleep.

5. No representation for real end system:
   the human.

Ref: R. Jain, “Internet 3.0: Ten Problems with Current Internet Architecture and
Solutions for the Next Generation,” Proceedings of Military Communications
Conference (MILCOM 2006), Washington, DC, October 23-25, 2006
Names, IDs, Locators

Name: John Smith

ID: 012-34-5678

Locator:
1234 Main Street
Big City, MO 12345
USA

Locator changes as you move, ID and Names remain the same.

Examples:
- Names: Company names, DNS names (Microsoft.com)
- IDs: Cell phone numbers, 800-numbers, Ethernet addresses, Skype ID, VOIP Phone number
- Locators: Wired phone numbers, IP addresses
Future Internet: Areas of Research

1. New architectures
2. Security
3. Content Delivery Mechanisms
4. Delay Tolerant Networking
5. Management and Control Framework
6. Service Architectures
7. Routing: New paradigms
8. Green Networking
9. Testbeds

Five Trends in Networking

1. Moore’s Law
2. User Multihoming + Mobility
3. Wireless Edge
4. Declining Revenues in Transport
5. Profusion of Services
Trend 1: Moore’s Law

- Computing Hardware is cheap
- Memory is plenty
⇒ Storage and computing (Intelligence) in the net

- Energy
- Space
- Communication in Space
- Link

- Matter
- Time
- Communication in Time
- Storage (USB, Caching, …)

Next Gen nets will use storage in networks, e.g., DTN, CCN
Trend 2: Multihoming + Mobility

- Centralized storage of info
- Anytime Anywhere computing
- Dynamically changing Locator
- User/Data/Host/Site/AS Multihoming
- User/Data/Host/Site Mobility

⇒ ID/Locator Split

Mobile Telephony already distinguishes ID vs. Locator. We need to bring this technology to IP.
Trend 3: Wireless Edge

1. Billions ⇒ Scalable
2. Heterogeneous ⇒ Customization of content
3. Slow ⇒ Bottleneck ⇒ Receiver Control
   (IP provides sender controls but no receiver controls)

Need to design from receiver’s point of view
Trend 4: Declining Revenues in Transport

- Telecom carriers' disappearing revenues in basic transport
- New opportunities in apps and Intelligent transport

Future of ISPs is to go beyond best effort trucking services
Trend 5: Profusion of Services

- Almost all top 50 Internet sites are services [Alexa]
- Smart Phones: iPhone, Android Apps
  - New globally distributed services, Games, …
  - More clouds, …

Networks need to support efficient service setup and delivery

Washington University in St. Louis http://www1.cse.wustl.edu/~jain/tutorials/ngi_adc.htm ©2012 Raj Jain
2. Internet 3.0

- Internet 3.0: Next Generation Internet
- Internet Generations
- Organizational Representation
- User- Host- and Data Centric Models
- Policy-Based Networking Architecture
- Multi-Tier Object-Oriented View
- Virtualization
Internet 3.0: Next Generation Internet

- Internet 3.0 is the name of the Washington University project on the next generation Internet
- Goal 1: Represent the commercial reality of distributed Internet ownership and organization
- Goal 2: Develop a *clean slate architecture* to overcome limitations of the current internet
- Goal 3: Develop an *incremental approach* to implement the architecture

[Image: Diagram illustrating the incremental approach]
Internet Generations

- **Internet 1.0** (1969 – 1989)
  - Single ownership $\Rightarrow$ Trust
  - complete knowledge
  - Algorithmic optimality $\Rightarrow$ RIP

- **Internet 2.0** (1989–2009) Commerce
  - Multiple ownership of infrastructure $\Rightarrow$ Distrust, **Security**
  - No knowledge of internal topology and resources
  - **Policy based** routing $\Rightarrow$ BGP

- **Internet 3.0** (2009–2029) Commerce
  - Users, Content, Host ownership
  - Requirements, Service Negotiation
  - Mobility of users and distributed data

Washington University in St. Louis  http://www1.cse.wustl.edu/~jain/tutorials/ngi_adc.htm  ©2012 Raj Jain
Service Center Evolution

Need a distributed load balancer for globally distributed datacenters.
Globally Distributed Services

- Scale ⇒ Global ⇒ Distributed ⇒ Multihomed
- Internet 1.0 is designed for point-to-point communication
- Significant opportunities for improvement for global services
Globally Distributed Services (Cont)

- It’s the service responsibility to find the right server for the client

![Diagram of globally distributed services with Google servers in the US, India, and China connected to the internet]
**Trend: Private Smart WANs**

- Services totally avoid the Internet core ⇒ Many private WANs
- Google WAN, Akamai ⇒ Rules about how to connect users

![Diagram of Google's WAN and access points](image)

**Opportunity for ISPs to offer these types of WAN services**

Washington University in St. Louis

[http://www1.cse.wustl.edu/~jain/tutorials/ngi_adc.htm](http://www1.cse.wustl.edu/~jain/tutorials/ngi_adc.htm)

©2012 Raj Jain
OpenADN

- High-Speed WAN for Application Service Delivery.
- Allows ASPs to quickly setup services

Service A1  Service B1  Service A2  Service b2

OpenADN

Internet

Access ISP

End User Hosts

Access ISP

End User Hosts
Ten Key Features that Services Need

1. **Replication**: Multiple datacenters appear as one
2. **Fault Tolerance**: Connect to B if A is down
3. **Load Balancing**: 50% to A, 50% to B
4. **Traffic Engineering**: 80% on Path A, 20% on Path B
5. **Flow based forwarding**: Movies, Storage Backup, …
   ATMoMPLS, TDMoMPLS, FRoMPLS, EoMPLS, …
   Packets in Access, Flows in Core
6. **Security**: Provenance, Authentication, Privacy, …
7. **User Mobility**: Gaming/Video/… should not stop as the user moves
8. **Service composition**: Services using other services
9. **Customization**: Every service has different needs
10. **Dynamic Setup** ⇒ Networking as a Service
Five Arch Design Principles for Success

1. Evolution not replacement
2. Coexistence (Backward compatibility)
3. Incremental Deployment
4. Economic Incentive for first adopters
5. Customization without losing control
Networking: Failures vs Successes

- 1986: MAP/TOP (vs Ethernet)
- 1988: OSI (vs TCP/IP)
- 1991: DQDB
- 1994: CMIP (vs SNMP)
- 1995: FDDI (vs Ethernet)
- 1996: 100BASE-VG or AnyLan (vs Ethernet)
- 1997: ATM to Desktop (vs Ethernet)
- 1998: ATM Switches (vs IP routers)
- 1998: MPOA (vs MPLS)
- 1999: Token Rings (vs Ethernet)
- 2003: HomeRF (vs WiFi)
- 2007: Resilient Packet Ring (vs Carrier Ethernet)
- IntServ, DiffServ, ...

Technology alone does not mean success.
Five Architecture Design Principles

1. Evolution not replacement.
2. Coexistence (Backward compatibility): Old on New. New on Old
3. Incremental Deployment
4. Economic Incentive for first adopters
5. Customization without loosing control (No active networks)

Most versions of Ethernet followed these principles. Many versions of IP did not.
The Narrow Waist

- Everything as a service over service delivery narrow waist
- IP, HTTP, Content, Service delivery, …
3. Content Centric Networks

- Content-Centric Networks (CCN)
- CCN Packets
- CCN Capable Routers Operation
- CCN Security
Content-Centric Networks

- IP cares about "Where": forward packets from A to B
- Users care about "What": Movie X
- Replace “packets” with “Data Objects” or “Interests” (requests)
- Replace “Addresses” with “Names of Objects”

CCN Packets

- Interest Packets: Request for Data
- Data Packets: Signed Data
- Longest prefix match is used as in IP addresses
  http://www.cse.wustl.edu/~jain/talks/ftp/in3_video matches
  http://www.cse.wustl.edu/~jain/talks/ftp/in3_video/V00/S00

**Interest packet**

| Content Name | Selector (order preference, publisher filter, scope, ...) | Nonce |

**Data packet**

| Content Name | Signature (digest algorithm, witness, ...) | Signed Info (publisher ID, key locator, stale time, ...) | Data |

[JAC09]
CCN Capable Routers Operation

- **Content Store**: Local cache of data
- **Pending Interest Table (PIT)**: Recent requests forwarded
- **Forwarding Information Base (FIB)**: Known data locations
- **Faces**: Requesting processes and hardware interfaces

---

[JAC09]
Routers Operation (Cont)

- Applications send “Interest” in data X
- Router looks up in local store and sends if found
- Router looks up in PIT, if entry already exists (someone requested it recently), adds the interest, face to the same entry
- Router looks up in FIB, if entry exists (data location is known), a PIT entry is made and the interest is multicasted to all faces in the FIB entry
- If there is no FIB entry, interest is discarded (router does not know how to get the data)
- When data arrives, Content Store match ⇒ duplicate, discard
  PIT match ⇒ Forward to all faces
  FIB match ⇒ No PIT ⇒ Unsolicited ⇒ Discard
- Data providers register their data ⇒ Creates FIB entries
CCN Security

- Data-Centric Security ⇒ Protections travel with the data
- All data is signed
- Data can be replicated or moved
- All data is versioned and is immutable once in the system
- IP and CCN routers can coexist. Public domain code available.
VOIP over CCN

- On-demand publishing: Data is produced only when some wants to connect
- Callee’s phone registers a service
- Caller looks for the service
- Issue: Complexity/State proportional to # of flows/users

4. + 5. Routing Architectures

- OpenFlow
- Software Defined Networking
- ID-Locator Split
  - Host Identity Protocol: HIP
OpenFlow

- Originally designed to allow researchers to run experimental protocols on production networks
- Each router/switch has a flow forwarding table
- Forwarding table is prepared by a central controller
- Vendors do not need to expose internal workings of their switches
- No need to program switches. Just program the central controller.


Washington University in St. Louis  http://www1.cse.wustl.edu/~jain/tutorials/ngi_adc.htm  ©2012 Raj Jain
OpenFlow (Cont)

- Three Components:
  - Flow table: How to identify and process a flow
  - Secure Channel: Between controller and the switch
  - Open Flow Protocol: Standard way for a controller to communicate with a switch

<table>
<thead>
<tr>
<th>Rule</th>
<th>Action</th>
<th>Stats</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Forward to Port n</td>
<td>Packet + Byte Counters</td>
</tr>
<tr>
<td></td>
<td>Encapsulate and forward to controller</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Drop</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Send to normal processing pipeline</td>
<td></td>
</tr>
<tr>
<td></td>
<td>Modify fields</td>
<td></td>
</tr>
</tbody>
</table>

- Mask
OpenFlow (Cont)

- TCAMs are used to match the fields
- Controller forwards the packets correctly as the mobile clients move
- Can handle non-IP networks
- OpenFlow Consortium is developing OpenFlow Switch Specification.
- Reference designs for Linux, Access points (OpenWRT), and NetFPGA (hardware)
- Combined packet and circuit switching
- Multiple controllers to avoid single point of failure: Rule Partitioning, Authority Partitioning

Ref: [MCK08], OpenFlowSwitch.org
Reactive and Proactive Operation

Proactive

- Switch flow tables pre-populated by the controller
- No flow setup time
- Loss of control connection does not affect operation
- Many entries never triggered

Reactive

- First packet of the flow triggers new flow entries
- Flow setup time
- Limited operation if control connection lost
- Efficient use of flow table entries

OpenFlow allows both models
Flow-based vs. Aggregated

**Flow Based**
- Every flow is individually setup
- Too many entries for large networks
- Good for fine-grained control

**Aggregated**
- Mostly wild card entries
- One entry per flow group
- Good for large networks, e.g., backbone networks

OpenFlow allows both options.
Current Limitations of OpenFlow

- Millions of flows in the backbone networks
  ⇒ Solved by using aggregated (wildcard) switching rather than per-flow switching

- Hardware is Openflow version specific
  New packet formats (non-IP, non-Ethernet, …)

- Non-flow based applications
  Stream of UDP packets can overwhelm the controller

- Use all switch features (vary with products)

- Security: 802.1X

- DHCP
Software Defined Networks

- Initial idea from Martin Casado (Stanford U/Nicira)
- Enhanced by Scott Shenker (UC Berkeley)
- Significant industry interest ⇒ Open Networking Foundation, https://www.opennetworking.org/
Problem: Complex Routers

- The routers are expensive because there is no standard implementation.
- Every vendor has its own hardware, operating/management system, and proprietary protocol implementations.
- Similar to Mainframe era computers. No cross platform operating systems (e.g., Windows) or cross platform applications (java programs).

<table>
<thead>
<tr>
<th>OSPF</th>
<th>BGP</th>
<th>DHCP</th>
<th>Cisco IOS</th>
<th>Juniper JUNOS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Network Operating System</td>
<td>Proprietary fast forwarding hardware</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Washington University in St. Louis  http://www1.cse.wustl.edu/~jain/tutorials/ngi_adc.htm  ©2012 Raj Jain
Solution: Divide, Simplify and Standardize

- Computing became cheaper because of clear division of hardware, operating system, and application boundaries with well defined APIs between them.

- Virtualization ⇒ simple management + multi-tenant isolation

<table>
<thead>
<tr>
<th>Scientific</th>
<th>Business</th>
<th>Batch</th>
</tr>
</thead>
<tbody>
<tr>
<td>OS360 Operating System</td>
<td></td>
<td></td>
</tr>
<tr>
<td>IBM 360 HW, Storage, …</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Physical HW</th>
</tr>
</thead>
<tbody>
<tr>
<td>Hypervisor</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>VM1</th>
<th>VM2</th>
<th>VM3</th>
</tr>
</thead>
</table>

<table>
<thead>
<tr>
<th>MSOffice</th>
<th>OpenOffice</th>
</tr>
</thead>
<tbody>
<tr>
<td>Windows</td>
<td>OS X</td>
</tr>
<tr>
<td>Chrome</td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Intel</th>
<th>AMD</th>
<th>ARM</th>
</tr>
</thead>
</table>
Multi-Tenant SDN Architecture

Enterprise 1
- Multicasting
- Network OS1

Enterprise 2
- Mobility
- Network OS2

Enterprise 3
- App1
- App2
- Network OS3

Applications
Network OS
Virtualization
Forwarding

Network Virtualization
# SDN Architecture Component Examples

<table>
<thead>
<tr>
<th>Forwarding</th>
<th>Monitoring/Debugging</th>
<th>Virtualization/Slicing</th>
<th>Applications</th>
<th>Network OS/Controller</th>
</tr>
</thead>
<tbody>
<tr>
<td>HP</td>
<td>oftrace</td>
<td>FlowVisor</td>
<td>HP</td>
<td></td>
</tr>
<tr>
<td>NEC</td>
<td>openenseer</td>
<td></td>
<td>Pronto</td>
<td></td>
</tr>
<tr>
<td>Ciena</td>
<td>oflops</td>
<td></td>
<td>Netgear</td>
<td></td>
</tr>
<tr>
<td>Juniper</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Pronto</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Ref:** [https://courses.soe.ucsc.edu/courses/cmpe259/Fall11/01/pages/lectures/srini-sdn.pdf](https://courses.soe.ucsc.edu/courses/cmpe259/Fall11/01/pages/lectures/srini-sdn.pdf)

Washington University in St. Louis  [http://www1.cse.wustl.edu/~jain/tutorials/ngi_adc.htm](http://www1.cse.wustl.edu/~jain/tutorials/ngi_adc.htm)  ©2012 Raj Jain
SDN Abstractions

- Distribution State Abstraction: No longer design a distributed control protocol. Design only centralized control.

- Specification Abstraction: Control program should specify “What” and not “how” ⇒ Virtualization.

- Forwarding Abstraction: Map global view to physical forwarding elements ⇒ OpenFlow.

SDN Impact

- Why so much industry interest?
  - Commodity hardware
    ⇒ Lots of cheap forwarding engines ⇒ Low cost
  - Programmability ⇒ Customization
  - Sharing with Isolation ⇒ Networking utility
  - Those who buy routers, e.g., Google, Amazon, Docomo, DT will benefit significantly

- Opens up ways for new innovations
  - Dynamic topology control: Turn switches on/off depending upon the load and traffic locality ⇒ “Energy proportional networking”
ID-Locator Split

1 and 2.
Host A obtains the identifier of host B from DNS.

3 and 4.
Host A gets host B’s ID resolved to host B’s locator through ID-locator mapping system.

5.
Host A sends packets to B.

6.
Routing based on locator.
ID-Locator Split (Cont)

- Allows hosts to move
- Allows entire organizations to move
  - Allows organizations to change providers
- No need to use “Provider Independent (PI)” addresses
- Provider Aggregatable (PA) addresses are preferred since they result in shorter BGP tables
  - Scalable
- Several proposals for host-based ID-locator split:
  - HIP, Shim6, I3, and HI3
- All hosts have ID and global locators
- Allow mobility, multihoming, renumbering
HIP

- Host Identity Protocol
- 128-bit Host ID tag (HIT)
- TCP is bound to HIT. HIT is bound to IP address in the kernel
- Uses flat cryptographic based identifier
- Two Methods:
  - Locator registered using Update packets to DNS
    ⇒ Does not allow fast mobility
  - Use rendezvous servers
    ⇒ Does not adhere to organizational boundary
- Requires changes to end hosts

6. Next Generation Testbeds

- Past: PlanetLab, Emulab
- Federation
- GENI, Requirements, Subsystems
- GENI Prototype Clusters
- Supercharged PlanetLab Platform (SPP)
- FIRE
- AKARI
PlanetLab

- Global networking research testbed
- 1055 nodes at 490 sites [Nov 2009]
- Researchers use it to experiment with new ideas on distributed storage, network mapping, peer-to-peer systems, distributed hash tables, and query processing

Ref: http://www.planet-lab.org/
PlanetLab (Cont)

- Linux virtual server software on Interned nodes
- **Slivers** = Piece of a resource
- Node manager (NM) manages the node's virtual servers
- Planet Lab Control (PLC) interacts with NM
- Experimenters request a "Slice" = slivers in various sites
Emulab

- Networking research testbed at University of Utah
- Available for public use for research and education
- Software implemented at two dozen sites around the world
- Allows simulated links and nodes in slices
  ⇒ Allows fault studies
- Provides repeatability

Ref: http://www.emulab.net/
Federation

- Larger testbeds
- Testbeds for specialized resources such as access technologies
- Specialized research communities and cross-discipline
- Challenges:
  - Homogenization of diverse context
  - Interoperability of security protocols
  - Political or social-economic issues
  - Intellectual Property rights
  - Commercial and non-commercial interests

GENI

- Global Environment for Network Innovations
- Dedicated shared substrate facility for large-scale experiments
- US National Science Foundation project
- Dedicated backbone links through LambdaRail and Internet2
- Diverse and extensible set of technologies

Refs: [GENI01, ON410]
GENI Requirements

- Sliceability: Sharing with isolation.
- Programmability: All components should be programmable
- Virtualization: Slicing via virtualization or space/time sharing.
- Federation: Combination of independently owned testbeds
- Observability: Allow specifiable measurement framework
- Security: Should not harm production Internet

Refs: [AND052, SHA05, CLA05, RAY05, BLU05, BELL05, KAA05]
GENI Subsystems

GENI Admin and Ops. Org
- Administrator
- Operator

Administrative and Accounting Tools
Operations and Management Tools

GENI Aggregates
- Aggregate Manager
- Hosts
- Host X

Slice A
Sliver

Experimental Plane
Measurement Plane
Control Plane

GENI Clearinghouse
- Principal Registry
- Principal Registry Authentication Query
- Slice Registry
- Slice A Record
- Slice B Record

Slice Manager
Component Registry Query

Component Registry

GENI Components
- Component Manager
- Components
- Component A

Services Manager
- Services
- Service S

GENI Services
- Slivers

Geni End Users
Geni Access Network

Washington University in St. Louis
http://www1.cse.wustl.edu/~jain/tutorials/ngi_adc.htm
©2012 Raj Jain

59
GENI Prototype Clusters

Five Clusters in Spiral 1:

1. Trial Integration Environment with DETER (TIAD): Emulab based security experiments testbed
2. PlanetLab: Federate all slice-based substrates PlanetLab, Emulab, VINI, and GENI
3. ProtoGENI: Federation of Emulab testbeds, Enhanced Emulab Control
4. Open Resource Control Architecture (ORCA): Resource manager runs under the host operating system Uses virtualization to allocate containers
5. Open Access Research Testbed (ORBIT): Wireless testbed with emulated and real nodes

Spiral 2: Improved instrumentation, tools for integration
Spiral 3: Integration. Experimentation across clusters.

Ref: GENI Spiral 1, http://groups.geni.net/geni/wiki/
Virtualizable Network Concept

Slide taken from Jon Turner’s presentation at Cisco Routing Research Symposium
Virtualization

- Allows multiple overlays on a single substrate
- Allows nodes to treat an overlay as a native network
- Provides isolation \(\Rightarrow\) multiple architectures, Partitioned Control
- Allow testing diverse routing protocols and service paradigms
- Better architectures will attract more users and become main line
- Allows diversified services while utilizing economies of scale in the substrate components
- Virtualization over IP networks \(\Rightarrow\) Not suitable for experiments at lower layers

Supercharged PlanetLab Platform (SPP)

- Allows multiple virtual routers with different stacks
- Fast path for line speed packet forwarding
- Slow path for application specific processing
- Multiple meta-networks (routers, links) on a substrate
- 3 Components: Line cards, switching fabric, control proc
- Virtualizing line cards is difficult
- Processing Pool Architecture:
  No processing in line cards
  Simply switch to proc engines

Refs: [TUR06, TUR107, TUR207]
FP7/ICT Program 2011/12

Challenge 1 - “Pervasive and Trusted Network and Service Infrastructures”

Ref: European Framework Programme for Research and Innovation (FP7),
Washington University in St. Louis http://www1.cse.wustl.edu/~jain/tutorials/ngi_adc.htm ©2012 Raj Jain
Future Internet Research and Experimentation
- Federate multiple existing testbeds in Europe
  ⇒ Provide a large multi-context research testbed
Japan

- Next Generation (Incremental): NXGN
  - Add QoS and authentication to IP
- New Generation (Clean slate): NWGN for 2015+
  1. National Institute for Information and Communications Technology (NiCT) is leading the research on NWGN
     - AKARI= A Small light pointing to the future
  2. Testbeds:
     - JGN2plus testbed for Network Virtualization
     - JGN X testbed for NWGN services and operations
  3. NWGN Promotion Forum (Japan Wide, Industry and Academic)

Ref: http://akar-project.nict.go.jp
AKARI Components

1. Parallel Optical Packet Transmission
2. All-Optical path/packet switching
3. Packet division multiple access
4. ID/Locator separation
5. Overlay network/Virtualization
6. Self-Organizing control
Top 10 Features of Next Generation Internet

1. Security
2. Mobility
3. User/Data-Centric: Network support of data objects
4. Easy to use: Self-organizing, better user control
5. Disruption Tolerant
6. Green: Proxy, Sleep Modes,
7. Services: Storage, Translation, Monitoring
8. Organizational Representation
9. Virtualizable to create Application Specific Context
10. Policy Enforcement
NSF FIA Winners

- **Named Data Networking**: CCN
  - Routing scalability, Fast forwarding, Trust models, Network security, Content protection and privacy
- **Mobility First**: Generalized Delay Tolerant Networking with self-certifying public key addresses
- **Nebula** (Latin for Cloud): Trustworthy data, control and core networking for cloud computing
- **eXpressive Internet Architecture (XIA)**: Application programming interface (API) for communication, flexible context-dependent mechanisms for establishing trust

XIA

- Partners: CMU, BU, UWisc
- Security, x-centric
- Principals: Hosts, Domain, Contents, Services, Users
- Secure identifiers for all principals: Hash of the public key
- Content naming based on cryptographic hash of the content
  ⇒ Receiver can verify correct content

Washington University in St. Louis  http://www1.cse.wustl.edu/~jain/tutorials/ngi_adc.htm  ©2012 Raj Jain
eXpressive Internet Protocol (XIP)

- Allows multiple destinations
- Allows multiple paths to a destination
- XIP addresses are directed acyclic graphs (DAGs)
## XIP Packet Header

<table>
<thead>
<tr>
<th>Ver</th>
<th>NxtHdr</th>
<th>PayLen</th>
<th>HopLimit</th>
<th>NS</th>
<th>ND</th>
<th>NxtDAGptr</th>
</tr>
</thead>
<tbody>
<tr>
<td>0</td>
<td>XidType</td>
<td>ID</td>
<td>P[N]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>XidType</td>
<td>ID</td>
<td>P[N]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>ND-1</td>
<td>XidType</td>
<td>ID</td>
<td>P[N]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>0</td>
<td>XidType</td>
<td>ID</td>
<td>P[N]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>...</td>
<td>XidType</td>
<td>ID</td>
<td>P[N]</td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>NS-1</td>
<td>XidType</td>
<td>ID</td>
<td>P[N]</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

- Variable length DAG fields
- 28B per DAG (4B type, 20B address, 4 1B edge pointers)
With CID, clients can get the content from other servers, replicas, and caches
Services on XIA

- Services are identified as: \( AD_{ID}:Host_{ID}:Service_{ID} \)
- Resolvers may resolve the \( Service_{ID} \) to \( AD_{ID}:Service_{ID} \) (Host is not specified)
- AD can select any host with that service
  \( \Rightarrow AD_{ID}:Host_{ID}:Service_{ID} \)
- If the service moves, new client is notified of the new hostID via a signed message from the previous host
MobilityFirst

- Partners: Rutgers, UMass, Duke, UMichigan, UNC, MIT, UNebraska, UWisconsin
- Designed for mobile devices: 4B cell phones
  1. Separation of naming and addressing
  2. Self-certifying public key network addresses
  3. Generalized Delay-tolerant networking
  4. Hop-by-hop transport protocol over path segments
  5. Flat-label internet routing with public key addresses
  6. Separate network management plane
  7. Privacy features for user and location data
  8. Programmability of routers for evolution

Ref: http://mobilityfirst.winlab.rutgers.edu/
NEBULA

- Trustworthy cloud computing
- Multiple stakeholders: Sender, receiver, transit providers, middle boxes, … Each has its own policy
- A packet is forwarded if the path meets all policies
  2. Nebula Data plane (NDP): Uses PoC and generates Proof of Path (PoP) – Route followed
  3. Nebula Core (NCORE): Provides high availability paths

Ref: http://nebula.cis.upenn.edu/NEBULA_brief.pdf
## Summary: NGI Research

- Clean-slate Internet architecture program started with NSF FIND program in 2005. Now extensive research in Europe, Japan, China, Korea, Taiwan, ...

<table>
<thead>
<tr>
<th></th>
<th>USA</th>
<th>Europe</th>
<th>Japan</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Architecture</strong></td>
<td>1. FIND 40+ projects</td>
<td>FP7:</td>
<td>AKARI</td>
</tr>
<tr>
<td></td>
<td>2. FIA</td>
<td>1. Network of the future</td>
<td></td>
</tr>
<tr>
<td></td>
<td>a. NDN</td>
<td>2. Service and software</td>
<td></td>
</tr>
<tr>
<td></td>
<td>b. XIA</td>
<td>architectures, Infrastructures</td>
<td></td>
</tr>
<tr>
<td></td>
<td>c. MobilityFirst</td>
<td>and Engineering</td>
<td></td>
</tr>
<tr>
<td></td>
<td>d. Nebula</td>
<td>3. Secure, Dependable and</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>Trusted Infrastructure</td>
<td></td>
</tr>
<tr>
<td></td>
<td></td>
<td>4. Networked Media</td>
<td></td>
</tr>
<tr>
<td><strong>Testbed</strong></td>
<td>GENI</td>
<td>GEANT2 (34 NRENs)</td>
<td>JGN2,</td>
</tr>
<tr>
<td></td>
<td></td>
<td>FIRE</td>
<td>JGN2plus,</td>
</tr>
<tr>
<td></td>
<td></td>
<td></td>
<td>JGN2 X</td>
</tr>
</tbody>
</table>