YAN: A Framework for Universal Active Service Access In IP Networks

Tein-Yaw David Chung
Active Network Research Group
Department of Computer Science and Engineering
Yuan-Ze University
Intelligent Internet

- Services can be deployed dynamically over Internet as demanded.
  - Network devices can be heterogeneous.
  - User devices can be very different, such as PDA, Notebook, IA, …
  - Personal communications are fully supported.
  - Service components can be built, composed, and loaded distributedly to enrich services resources.
What Services in II

• Similar services in Intelligent Network (IN).
  – Call forwarding, Conference Calls, Call recording, Call monitoring ……

• More services such as tele-immersion, combining teleconferencing, telepresence, virtual reality.
Design Approach

• Component based
  – Each service components can be provided by different vendors and software engineers can use these off shelf components to compose a new service

• Cross Middleware
  – Service components may be written over Java Beans, Corba, Dcom, or Jini.

• Automatic Service Deployment
  – Once a service is offered, users can access it from a Web page. The service, if new, is automatically deployed and provides seemingless service to users.
Two development Directions

- **Horizontal Plane:**
  - Develop middleware that uses II basic building blocks to support new applications, such as CORBA-enable systems, mobile agents.
  - Develop programmable platform on IP-based core network to deploy new services, such as P1520.
  - Develop protocols on IP, e.g., Megaco, …

- **Vertical Plane:**
  - Introduce new protocols that are independent of IP planes, such as ANTs.
Active Networking

- A programmable paradigm that enables users, network operators, or third party service providers to dynamically deploy de facto services.
- Routers, Servers, and End Stations all need to support the programmable environment.
- On demand download of protocols, or programs that implement the protocols to the network active nodes.
Active Network Architecture
End to End Service in a Heterogeneous Environment

Relay Model
How Does It Work?

End node

Active node

server

Active node

End node

New services set up

proceeding

ok

New services set up

proceeding

ok

Data capsules

New services set up

ok
How Does It Work

- End node
  - New services set up
  - proceeding
  - ok
- Active node
  - New services set up
  - Data capsules
- Server
  - Download is needed
  - Here is the code
  - ok
- Active node
- End node
  - New services set up
  - ok
How Does It Works?

- **End node**
  - New services set up
  - proceeding
  - ok

- **Active node**
  - Download is needed
  - Here is the code
  - ok
  - Data capsules

- **Active node**
  - New services set up
  - Download is needed
  - Here is the code
  - New services set up
  - Download is needed
  - Here is the code
  - ok

- **End node**
  - ok
ANTs Structure
ANTs Element

- Capsule is a transmission unit.
- A protocol is defined for Capsule transmission, including code distribution and active channel set up.
- Programs that implement service protocols are encapsulated in capsules for transmission.
Protocol Structure

(P) protocol (unit of protection)

(G) code group (unit of deployment)

(F) forwarding routine (unit of processing)

(C) capsule (unit of transmission)
ANTs Element (cont’)

- Channel creates an active pipe for capsule transmission.
- In a heterogeneous environment, a channel can cross several non-active routers.
- Channels commonly create an Active Network topology.
ANTs Element (cont’)

- Nodes include basic objects that create a required execution environment for service deployment.
- Extension allows users or network operators to extend existing Nodes for more sophisticated services. That is, frequently used objects can be incorporated into Nodes and resides in the active router.
ANTs Element (cont’)

• Applications that access the active services using protocol objects, which can inherit from existing protocols or be built upon Nodes and extension.
ANTs Architecture

End system -> application -> node -> channel -> Extension -> Active router

End system -> application -> node -> channel -> Extension -> capsule

End system -> application -> node -> channel -> Extension -> capsule

End system -> application -> node -> channel -> Extension -> capsule
ANTs Characteristics

- Connection-oriented model
  - Active network topology must be set up using channels.

- Overlay model over Internet, i.e., ANTs over IP.

- A generic platform for new protocols development based on an ANTs protocol, capsule.

- End-to-end service model.
Challenges to ANTs

- Capsule processing is at application level and leads to low performance.
- Useful for few applications.
- Motivated by bypassing standard process, but ANTs itself must be a standard first.
- Involves active routers, and hence uses a hard-state model, a problem for scaling.
- Capsule model makes security a tough challenge.
P1520

- IEEE working standard
- Develop APIs for a router or switch.
- Current sub-group
  - Active IP router
  - Active ATM switch
- Purpose:
  - Facilitate the resource sharing of a router or switch through standard APIs.
P1520 Approach

- Standardize different level APIs for router programmability.
- Some level of API can either be loaded on demand or by operators. Some can only be loaded by operators.
- Dynamic service deployment is not the targeted goal, but the flexibility in router function upgrade is.
P1520 Architecture

Value-added Application
Value added Service
New network functions
New switching functions
Switch resources

V interface
U interface
L interface
CCM interface
Difficulty of P1520

- Tradeoff between layered API and performances.
  - Sequenced processing: A problem that is addressed by application layer framing (ALF) and integrated layered implementation.
  - User or software processing: A serious problem in routers in which hardwired performance is expected.

- A plug-and-play in Kernel level is difficult with current model.
Difficult of P1520 (cont’)

- Current development focuses on existing new protocol implementation, such as DiffSev.
- Difficult to define API comprehensive enough to facilitate new upcoming protocols development.
Yen Active Network (YAN)

- A framework that allows users to access active services from Web servers.
- Profile-based network service deployment
  - Application profiling
  - Network service profiling
- Active signaling and active data transmission.
Application Service Profile
Brick 1: Application Configuration
Application Configuration

• Node:
  – Type of devices, user, server, gateway or transcoder. This must be known to the application.
  – URI of the devices
  – Access media type

• Session:
  – Session description
    • Port
    • Transport protocol
    • QoS

• Application Configuration
  – Topology of the nodes and sessions.
  – URIs of application objects or modules for each node type.
Brick 2: Active Signaling

- SIP signaling
  - Location tracking of involved parties
  - Device profile finding

- Active signaling
  - SIP signaling extension. Default is SIP.
  - Execution at a signaling path, instead of data transmission path.
Brick 3: Execution Environment

- EE types
  - such as ANTs, or …
- EE version
- Platform
  - such as LINUX, or …
- Multiple EE’s may be required.
Brick 4: Capsule

- The service routines associated with messages used in the applications.
- Service routines appeared as URI’s.
- Capsule can be defined
  - different service routines for different user devices and access medium type.
  - Different service routines for different node types.
Two steps service deployment

- **Step 1: Application deployment**
  - SIP redirect mode that contains the URI of application.
  - Once SIP resolves the URI, it returns the object reference to user terminal.
  - User terminal downloads the software objects from application repository if local is not available.

- **Step 2: Network control deployment**
  - deploy required software at the active routers between each end devices using deployment protocol.
Working Principle

1. ASDI
2. Session Initialization
3. Session profile distribution
4. ASDI to NSDI mapping
5. Service deployment
3-step application service profiling

ASDTD → ASP → Default Profile → User entry → Customized Profile → SIP → Exact Profile
2-step network service profiling

Session Profile → Path Profile → Transmission route → Nodal Profile
Session Profile

• Type of transmission
  – bidirectional unicasting
  – Uni-directed multicasting
  – uni-directional broadcasting

• QoS of a session
  – delay
  – jitter
  – bandwidth
  – error probability
Path Profile

- **Basic Components**
  - QoS requirement
    - Loss probability
    - Bandwidth
  - Type of transmission
    - unicast
    - multicast
    - broadcast
  - Reference of routing components in Active nodes
  - Capsule
Network Service Profile

- Protocol graph for the required service.
  - Network control services
  - Network service objects
  - Initialization parameters
  - Network service capsules
Service in YAN

V interface
- Connection management
- Load balance
- multicast
- security
- Feedback reduction
- UDP
- TCP

U interface
- Resource Manager
- CR
- RSVP
- OSPF
- QoS
- DS

L interface
- ICMP
- IP
- CBQ

Proxy service
Application filter
Agent
ASDP
Advantage of YAN

- Profile-based service deployment encourages software reuses.
- Profile-based approach allows separation of service provisioning.
- SIP-based active signaling allows separation of control signal path and data transmission path.
- Universal active services access from Web servers.
Current & Future Works

- Network security on YAN.
- Application filter on YAN.
- Agent framework at YAN.
- Active signaling at YAN.
- Resource management on active operating system.
- The framework of YAN and its detail design.
- Routing services in YAN.
- Detail design of YAN components.
- Profile DTD refinement.
- Design of U and V interfaces.
- Applications development of YAN.