Mobile Middleware Course

Support Technologies

Sasu Tarkoma

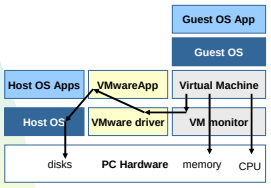
Contents

- Virtualization
- Session Initiation Protocol (SIP)
- IP Multimedia Subsystem (IMS)
- Web services
- Research systems

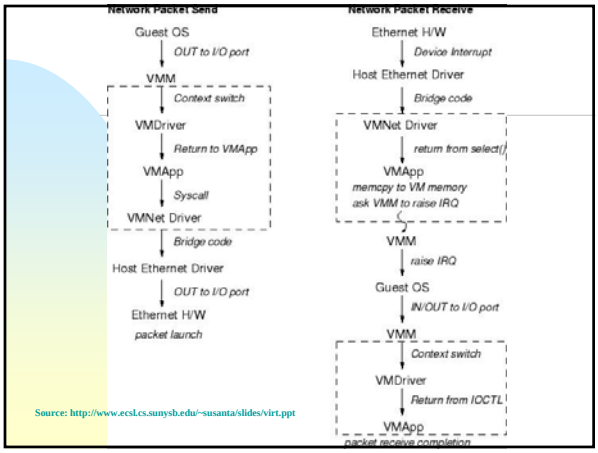
Virtualization

- Virtualization system is a framework that combines or divides computing resources to present a *transparent* view of one or more environments
  - Hardware/software partitioning (or aggregation)
  - Partial or complete machine simulation
  - Emulation (can be partial or complete)
  - Time-sharing

VMware Architecture



Source: http://www.ecsl.cs.sunysb.edu/~susanta/slides/virt.ppt



Source: http://www.ecsl.cs.sunysb.edu/~susanta/slides/virt.ppt

Mobile virtualization

- Server Based Virtualization of Desktop Infrastructure
- Client Side Solution – Assured Computing Environment
- Application Virtualization
- Moving the desktop to a virtualized image in the data center allows the complex components to be protected and componentized
- Workload isolation and migration
- Application virtualization
- Virtualization is a possible solution to the fragmentation problem
- http://www.vmware.com/technology/mobile/

SIP

- An Application-layer control (signaling) protocol for creating, modifying and terminating sessions with one or more participants.
- Sessions include Internet multimedia conferences, Internet telephone calls and multimedia distribution.
- Members in a session can communicate via multicast or via a mesh of unicast relations, or a combination of these.
- Text based, model similar to HTTP.

SIP History

- Mid-1990s, emerged from the research of Dr. Henning Schulzrinne, Columbia University, on Multi-party Multimedia Session Control (MMUSIC)
- 1996, submitted to the Internet Engineering Task Force (IETF) and developed by the SIP Working Group
- 1999, first published as IETF RFC 2543
- 2000, selected by 3GPP
- 2002, RFC 3261 (with further supplements in other RFCs)

SIP features

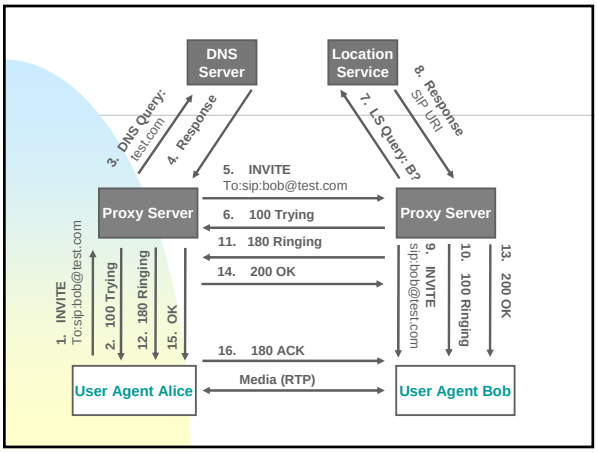
- User location: determination of the end system to be used for communication
- User capabilities: determination of the media and media parameters to be used
- User availability: determination of the willingness of the called party to engage in communications
- Call setup: "ringing", establishment of call parameters at both called and calling party
- Call handling: including transfer and termination of calls

SIP messages

- Start Line (Request Line or Status Line)
  - message type (method type & URI in requests, and response code in responses)
  - protocol version
- Headers
  - fields to convey message attributes
  - can span multiple lines, appear multiple times, take multiple comma-separated values
- Body (Content)
  - to describe the session to be initiated
  - to contain opaque textual or binary data

SIP

1. SIP Addressing
2. Locating a SIP Server
3. Sending SIP Requests: SIP Transactions
4. SIP Methods
5. SIP Responses
6. Subsequent Requests and Responses



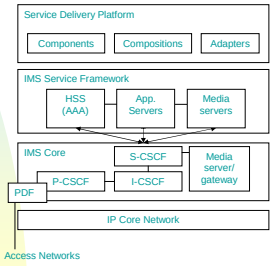
IMS

- The **IP Multimedia Subsystem** Provides Multimedia Services Across Networks (fixed & mobile), such as:
  - Instant Messaging, Video Sharing, Push-To-Talk, Gaming, Video Conferencing
- IMS Uses SIP protocol To Setup Multimedia Sessions Over IP Network
- SIP is a signalling protocol to:
  - Locate user given SIP Universal Resource Identifier (URL) (e.g., sip:jane@isp.com)
  - Set up session and negotiate its parameters

IMS Proxies

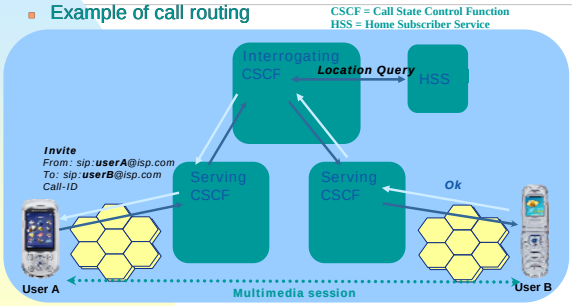
- A *Proxy-CSCF* (P-CSCF) is a SIP proxy that is the first point of contact for the IMS terminal
  - It is assigned to an IMS terminal during registration, and does not change for the duration of the registration
  - it sits on the path of all signalling messages, and can inspect every message
  - it authenticates the user and establishes an IPsec security association with the IMS terminal.
  - it can also compress and decompress SIP messages using SigComp, which reduces the round-trip over slow radio links
  - it may include a Policy Decision Function (PDF), which authorizes media plane resources e.g. quality of service (QoS) over the media plane. It's used for policy control, bandwidth management, etc. The PDF can also be a separate function.
  - it also generates charging records
- A *Serving-CSCF* (S-CSCF) is the central node of the signalling plane
  - It is always located in the home network. It uses Diameter Cx and Dx interfaces to the HSS to download and upload user profiles — it has no local storage of the user. All necessary information is loaded from the HSS
  - it handles SIP registrations, which allows it to bind the user location (e.g. the IP address of the terminal) and the SIP address
  - it sits on the path of all signaling messages, and can inspect every message
  - it decides to which application server(s) the SIP message will be forwarded, in order to provide their services
  - it enforces the policy of the network operator
- An *Interrogating-CSCF* (I-CSCF) is another SIP function located at the edge of an administrative domain

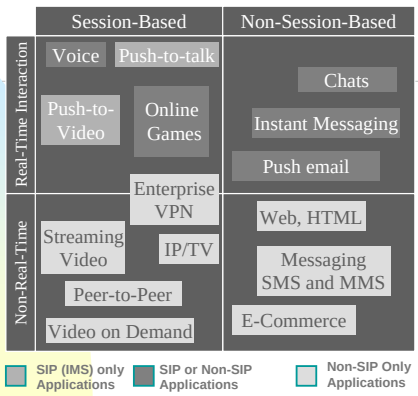
IMS



IMS

- Example of call routing





Event-based Systems and Publish/subscribe

- Event delivery from publishers to subscribers
  - Event is a message with content
  - One-to-many, many-to-many
  - Builds on messaging systems and store-and-forward
- A frequently used communication paradigm
  - Decoupling in space and time
  - Solutions from local operation to wide-area networking
  - Proposed for mobile/pervasive computing
- The event service is a logically centralized service
  - Basic primitives: subscribe, unsubscribe, publish
- Various routing topologies and semantics

Web Service Architecture

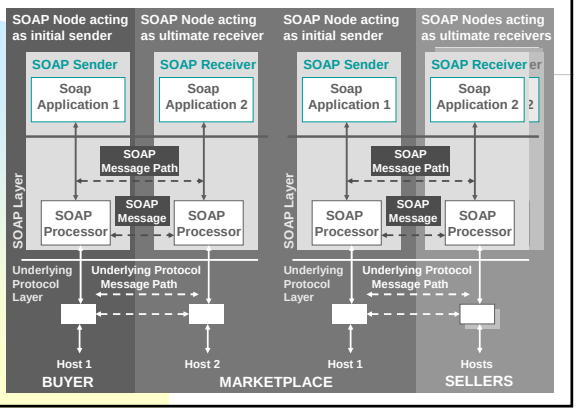
- The three major roles in web services
  - Service provider
    - Provider of the WS
  - Service Requestor
    - Any consumer / client
  - Service Registry
    - logically centralized directory of services
- A protocol stack is needed to support these roles

Web Services Protocol Stack

- Message Transport
  - Responsible for transporting messages
  - HTTP, BEEP
- XML Messaging
  - Responsible for encoding messages in common XML format
  - XML-RPC, SOAP
- Service Description
  - Responsible for describing an interface to a specific web service
  - WSDL
- Service discovery
  - Responsible for service discovery and search
  - UDDI

What is SOAP?

- Fundamentally stateless one-way message exchange paradigm
  - More complex interactions may be implemented
- Exchange of structured and typed information
  - Between peers in decentralized fashion
  - Using different mediums: HTTP, Email, ..
- Request-reply and one-way communication are supported
- Note that XML infoset is an abstract specification
  - On-the-wire representation does not have to be XML 1.0!
- SOAP 1.2 "HTTP Subset". SOAP as HTTP extension
- Specifications
  - SOAP Version 1.2 Part 0: Primer
  - SOAP Version 1.2 Part 1: Messaging Framework
  - SOAP Version 1.2 Part 2: Adjuncts
  - SOAP Version 1.2 Specification Assertions and Test Collection



REST

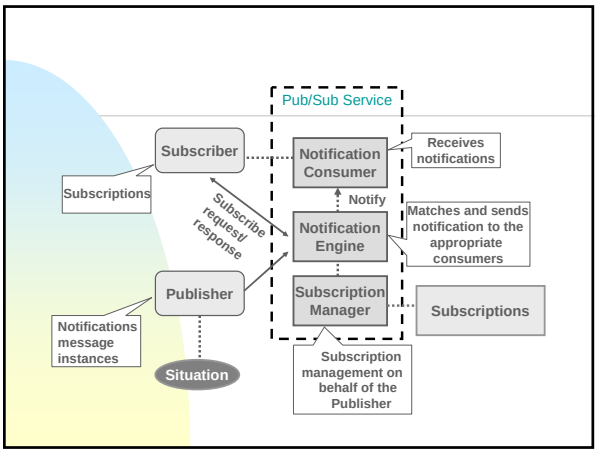
- REST (Representational State Transfer) (Roy Fielding, PhD thesis)
  - Architectural style of networked systems
  - Applications transfer state with each resource representation
    - Representations of the data are transmitted
  - State is a property of a resource
- Resources
  - Any addressable entity
  - Web site, HTML page, XML document, ..
- URLs Identify Resources
  - Every resource uniquely identifiable by a URI

REST II

- Uses standards
  - Addressing and naming: URI
  - Generic resource interface: HTTP GET, POST, PUT, DELETE
  - Resource representations: HTML, XML, GIF,..
  - Media types: MIME
- Loose coupling
- Stateless transactions
- Self-descriptive messages
- **Hypermedia is the engine of application state**
  - **Just resources and URIs**

Event Systems I

- Traditional MoM systems are message queue based (one-to-one)
- Event systems and publish/subscribe are one-to-many or many-to-many
  - One object monitors another object
  - Reacts to changes in the object
  - Multiple objects can be notified about changes
- Events address problems with synchronous operation and polling
- In distributed environments a logically centralized service mediates events
  - anonymous communication
  - expressive semantics using filtering



Event Systems II

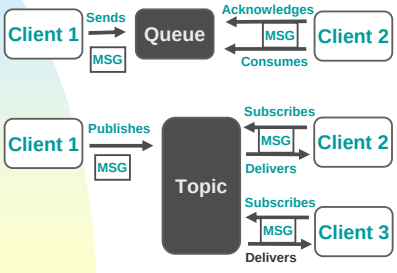
- Push versus Pull
- May be implemented using RPC, unicast, multicast, broadcast,..
- Three main patterns
  - Observer design pattern
    - Used in Java / Jini
  - Notifier architectural pattern
    - Used by many research systems
  - Event channel
    - Used in CORBA Event/Notification Service
- Filtering improves scalability / accuracy
  - Research topic: content-based routing

Tuple Spaces

- Tuple-based model of coordination
- The shared tuple space is global and persistent
- Communication is
  - decoupled in space and time
  - implicit and content-based
- Primitives
  - **In**, atomically read and removes a tuple
  - **Rd**, non-destructive read
  - **Out**, produce a tuple
  - **Eval**, creates a process to evaluate tuples
- Examples: Linda, Lime, JavaSpaces, TSpaces

Java Message Service (JMS)

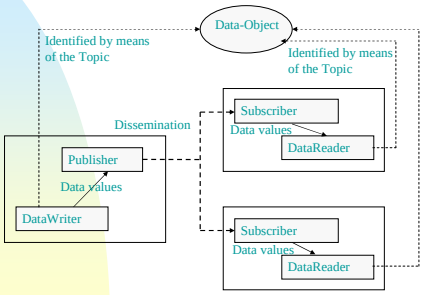
- Asynchronous messaging support for Java
- Point-to-point messaging
  - One-to-one
- Topic-based publish/subscribe
  - SQL for filtering messages at the topic event queue
  - One-to-many
- Message types:
  - Map, Object, Stream, Text, and Bytes
- Durable subscribers
  - Event stored at server if not deliverable
- Transactions with rollback



OMG Distributed Data Service I

- The Data Distribution Service for Real-Time Systems (DDS)
- The specification defines an API for data-centric publish/subscribe communication for distributed real-time systems.
- DDS is a middleware service that provides a global data space that is accessible to all interested applications.
- DDS uses the combination of a Topic object and a key to uniquely identify instances of data-objects.
- Content filtering and QoS negotiation are supported
- DDS is suitable for signal, data, and event propagation.

DDS



Pervasive computing middleware

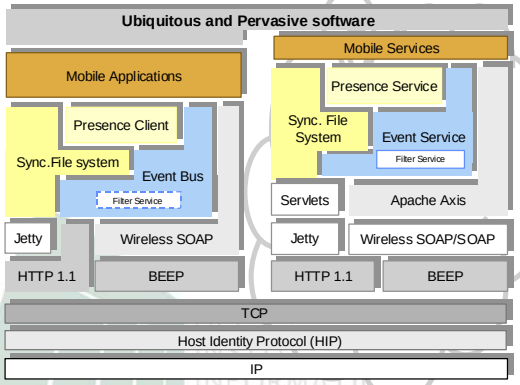
Projects	*Key Issues*
UIC	**Heterogeneity of devices and networks:** It helps users to specialize to the particular properties of different devices and network environments
X-Middle	**Disconnected operations in mobile applications:** It allows mobile users to share data when they are connected, or replicate the data and perform operations on them off-line when they are disconnected; data reconciliation takes place when user gets reconnected
Gaia	**Dynamic adaptation to the context of mobile applications:** It supports the development and execution of portable applications in active spaces
Lime	**Programming constructs which are sensitive to the mobility constraints:** It explores the idea by providing programmers with a global virtual data structure and a tuple space (Tspace), whose content is determined by the connectivity among mobile hosts
Tspaces	**Asynchronous messaging-based communication facilities without any explicit support for context-awareness:** It explores the idea of combination of tuple space (Tspace) and a database that is implemented in Java. Tspace targets nomadic environment where server contains tuple databases, reachable by mobile devices roaming around
L2imbo	**QoS monitoring and control by adapting applications in mobile computing environment**: It provides the facilities of multiple spaces, tuple hierarchy, and QoS attributes
Aura	**Distraction-free pervasive computing:** It develops the system architecture, algorithms, interfaces and evaluation techniques to meet the goal of pervasive computing

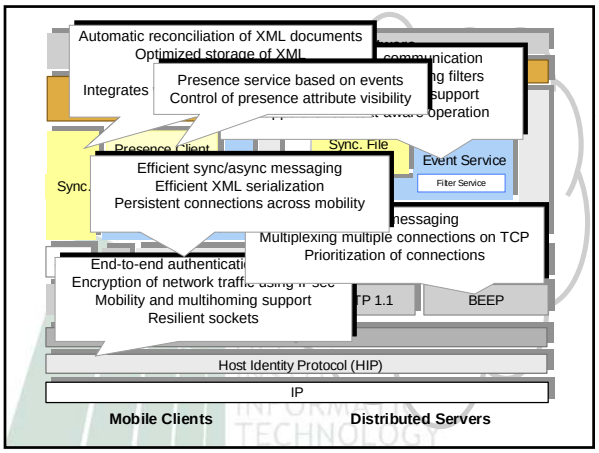
Fuego Core (HIIT)

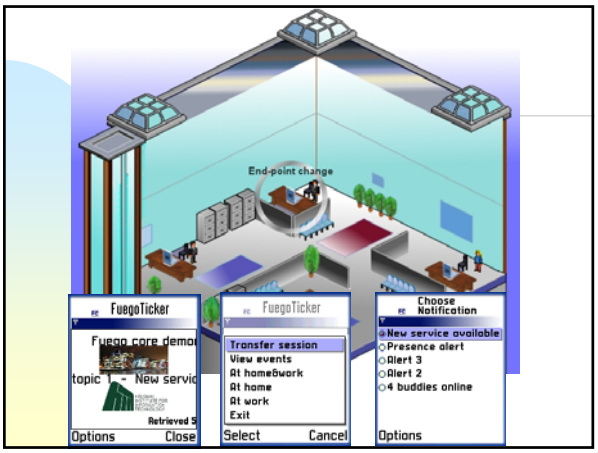
- Mobile and wireless environments have different requirements than desktop systems
  - User mobility, terminal mobility, connectivity, device characteristics, dynamic environments
- Vision: A service application is distributed among various application servers, network elements and terminals
- Three year Tekes project (2002-2004)
  - Industrial partners: Nokia, TeliaSonera, Elisa, Ericsson, Movial
  - Open source software
- Demonstration at WMCSA 2004

Fuego Architecture

- The Fuego middleware service set for mobile computing
  - Data communication
    - Efficient wireless SOAP
    - Efficient content-based routing (asynchronous events)
  - Data synchronization
    - 3-way XML document merging
    - XML-aware distributed file system
  - Applications
    - Presence service, mobile ticker, image-album



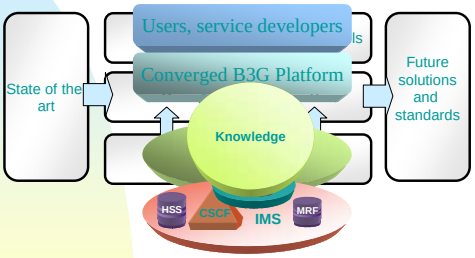




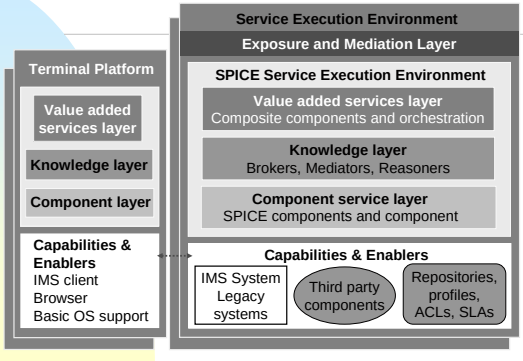
SPICE

- Provides service and component development and deployment infrastructure
  - For network operators and 3rd party service providers
  - IMS as enabling technology. Focus on IMS evolution
  - Combining IMS with Web services
  - Support expected business models
- Goals
  - 3rd party services can be easily created
  - Converged services - combining telecommunications and IT services
  - Service provider can focus on the core business
  - Platform support:
    - identity, charging, context awareness interface, service roaming between platforms, technology abstractions, …
  - Generate operator revenues jointly with 3rd parties (revenue sharing)

Architecture



SPICE Platform



SPICE: Key Principles

- SPICE users are essentially IMS subscribers, not necessarily registered in IMS
  - Rationale: loose coupling of non-IMS-based services with IMS through the use of IMS credentials and identity management
- SPICE Application Servers are primarily but not exclusively IMS AS
  - SIP-based mechanisms are the default choice for access using conversational or multimedia sessions
  - For converged network services, we are also examining HTTP and Web services based access
- SIP-based data models should be used for interaction between the users and the SPICE platform
  - IETF-defined presence data model and filtering/authorizing mechanisms, the OMA-defined UAProf model
  - For example, in Knowledge Management Framework, HTTP or Web services or SIP-based mechanisms may be used
- OMA/IMS network enablers should be used in priority
  - Presence, IM, voice and video calls, Messaging