JAIN™ Technology

Serving the Developer Community

jainteam@sun.com
http://java.sun.com/products/jain

Sun microsystems
We make the net work.

Introduction

© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN APIs: Java™ Technology for Communications

JAIN technology is:

- Java Application Servers for Communications
- Java Application Interfaces for Communications

Service Portability - *Write Once, Run Anywhere*

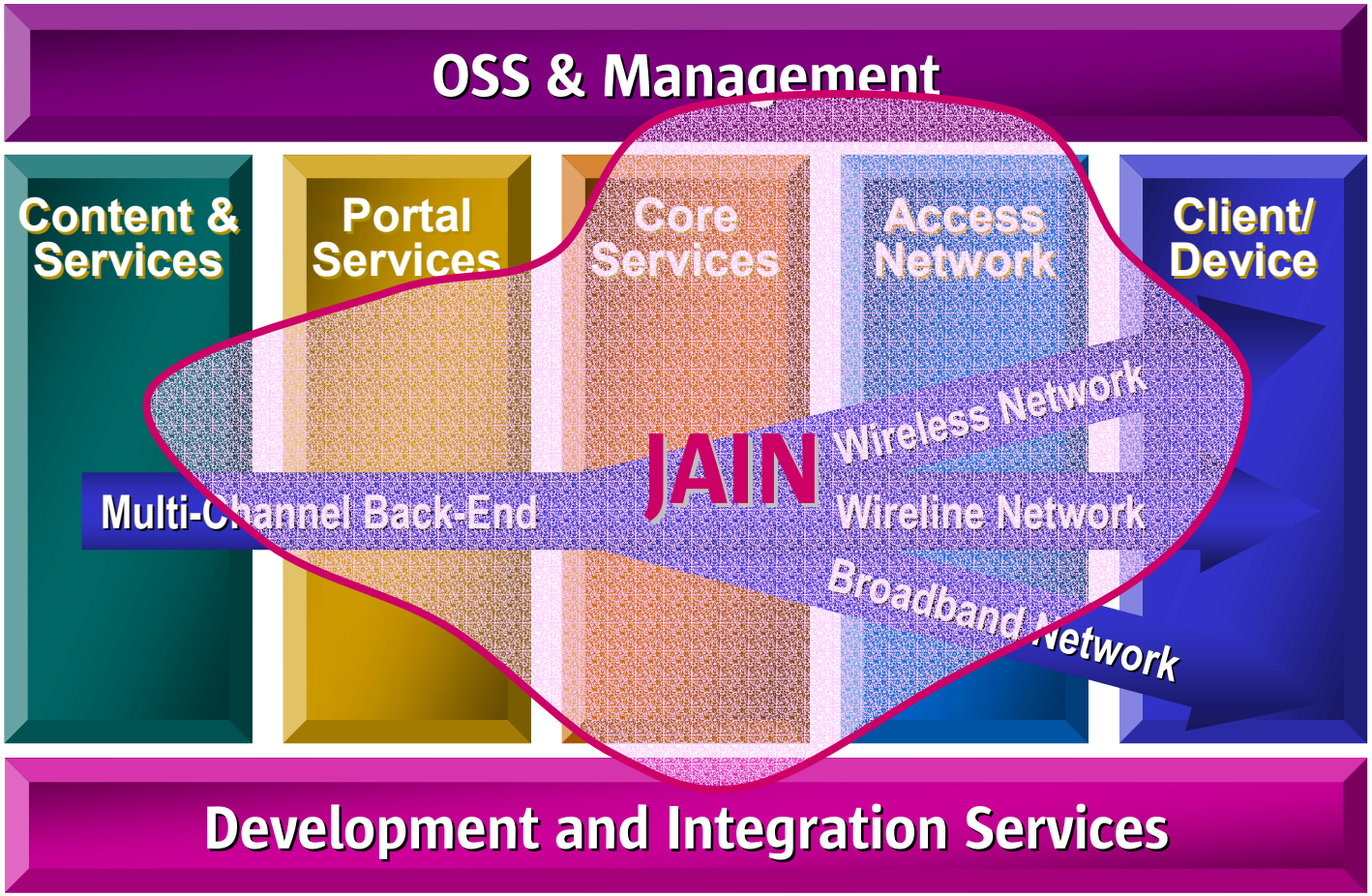
Network Independence - *Any Network*

Open Developer Interfaces - *By Anyone*

The JAIN community is an industry framework designed and specified collaboratively by groups of industry partners and experts.

© 2003 Sun Microsystems, Inc. All Rights Reserved.

New Service Driven Network



© 2003 Sun Microsystems, Inc. All Rights Reserved.

The Open API Premise

APIs that are …

- ✓ Open
- ✓ Industry standard
- ✓ Distributed
- ✓ Flexible
- ✓ Extensible

… will enable an abundance of diverse service offerings to enrich the way we communicate.

© 2003 Sun Microsystems, Inc. All Rights Reserved.

Why Standard APIs ?

- Standard APIs enable software developers to create portable services and applications
- Standard APIs provide an opportunity for abstraction
- Standard APIs enable a Write Once, Profit Often world
- Standard APIs will enable the Next Generation of Telecom Application Solutions
  - Inside the Network
  - At the Network Edge
  - The Consumer Space
  - Customer in Control of the Programmable Network

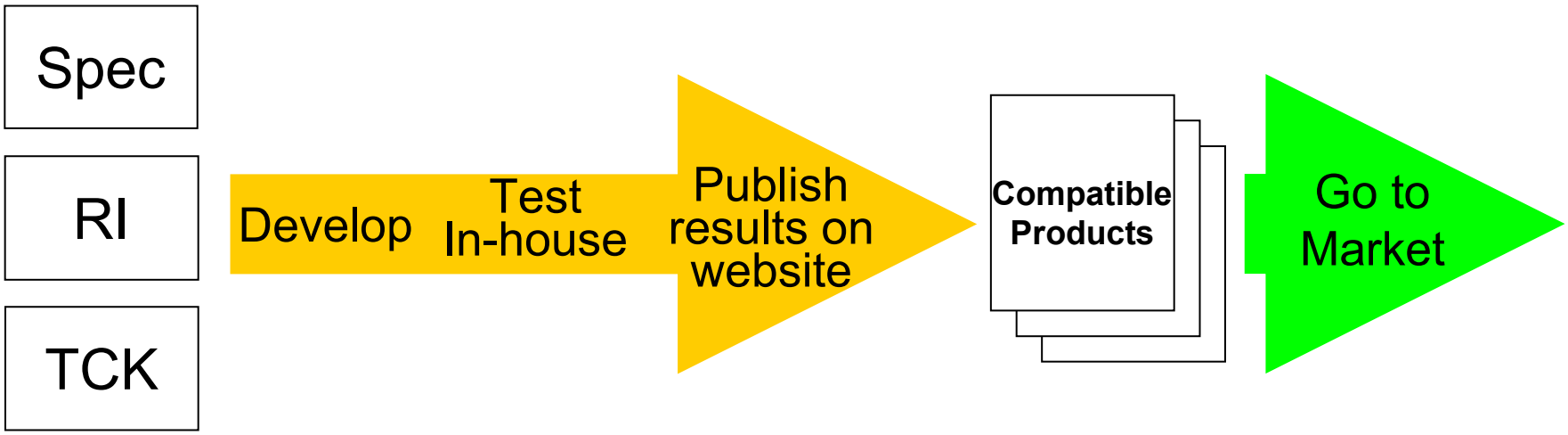
© 2003 Sun Microsystems, Inc. All Rights Reserved.

The Java Specification Process

- Java Specification Request (JSR)
  - Formed by consensus
  - By a group of industry experts
  - By a Specification Lead
- Four major steps:
  - Initiation
  - Community Draft
  - Public Draft
  - Final Release & Maintenance
- For more information
  - http://jcp.org

© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN Certification Process



1. Test implementation using TCK
2. Complete successful self assessment
3. Publish TCK results on vendor's website
4. Notify JAIN Program Manager
5. Certified product is published on JAIN website

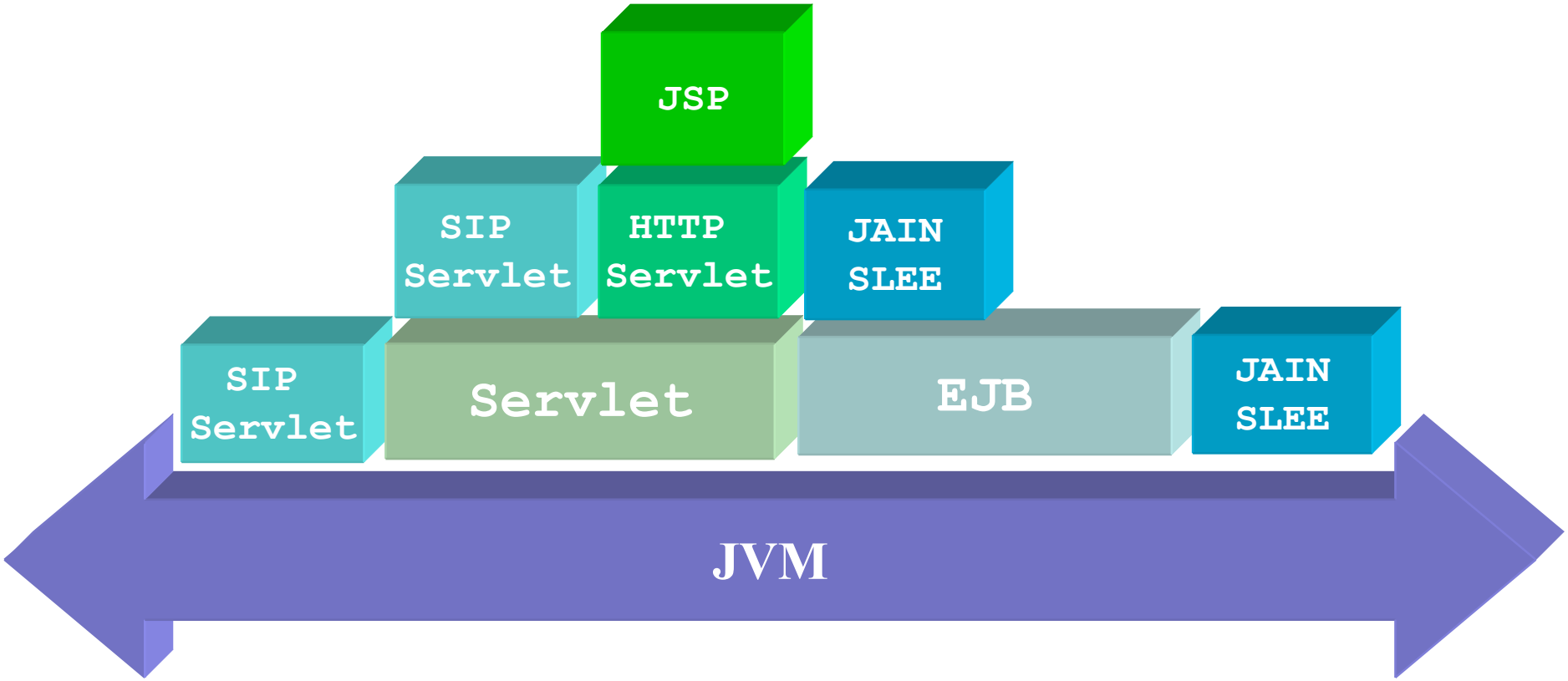
For further details on the Certification Process, visit:
http://java.sun.com/products/jain/certification.html

© 2003 Sun Microsystems, Inc. All Rights Reserved.

Execution Environments

© 2003 Sun Microsystems, Inc. All Rights Reserved.

Execution Environments



© 2003 Sun Microsystems, Inc. All Rights Reserved.

Why are Communications Applications Converging on Java Containers?

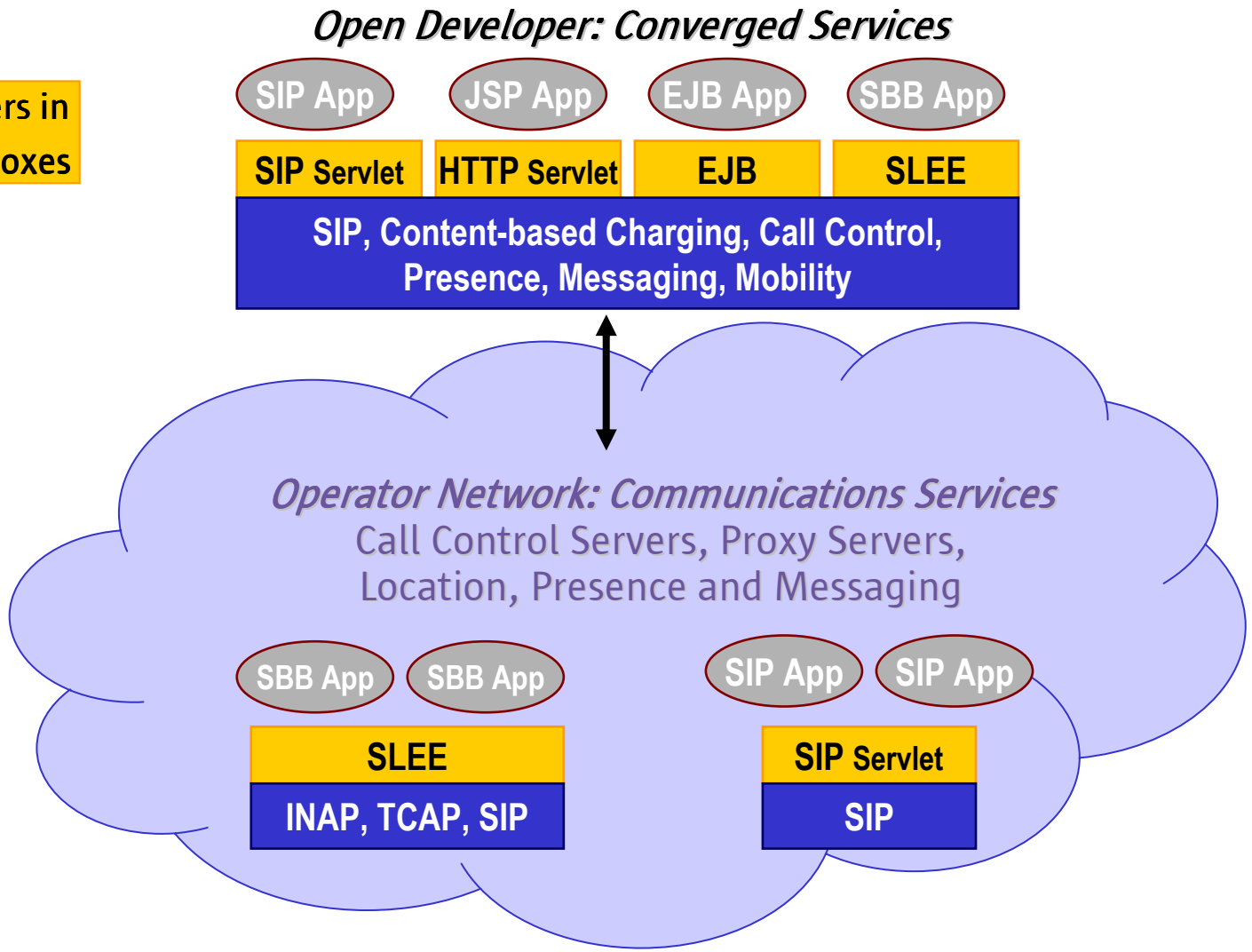
- Telco apps moving to component based architectures
- Desire to use Standard, Off-the-shelf container
  - Write-once, run-anywhere
- Container provides important infrastructure services
  - Higher level abstractions for State management, Transactions, Security, Resource pooling, …
- Focus on core value-add application logic
- Leverage large community of Java developers
- Leverage enterprise development tools, test suites, …

Time to market and reduced development cost

© 2003 Sun Microsystems, Inc. All Rights Reserved.

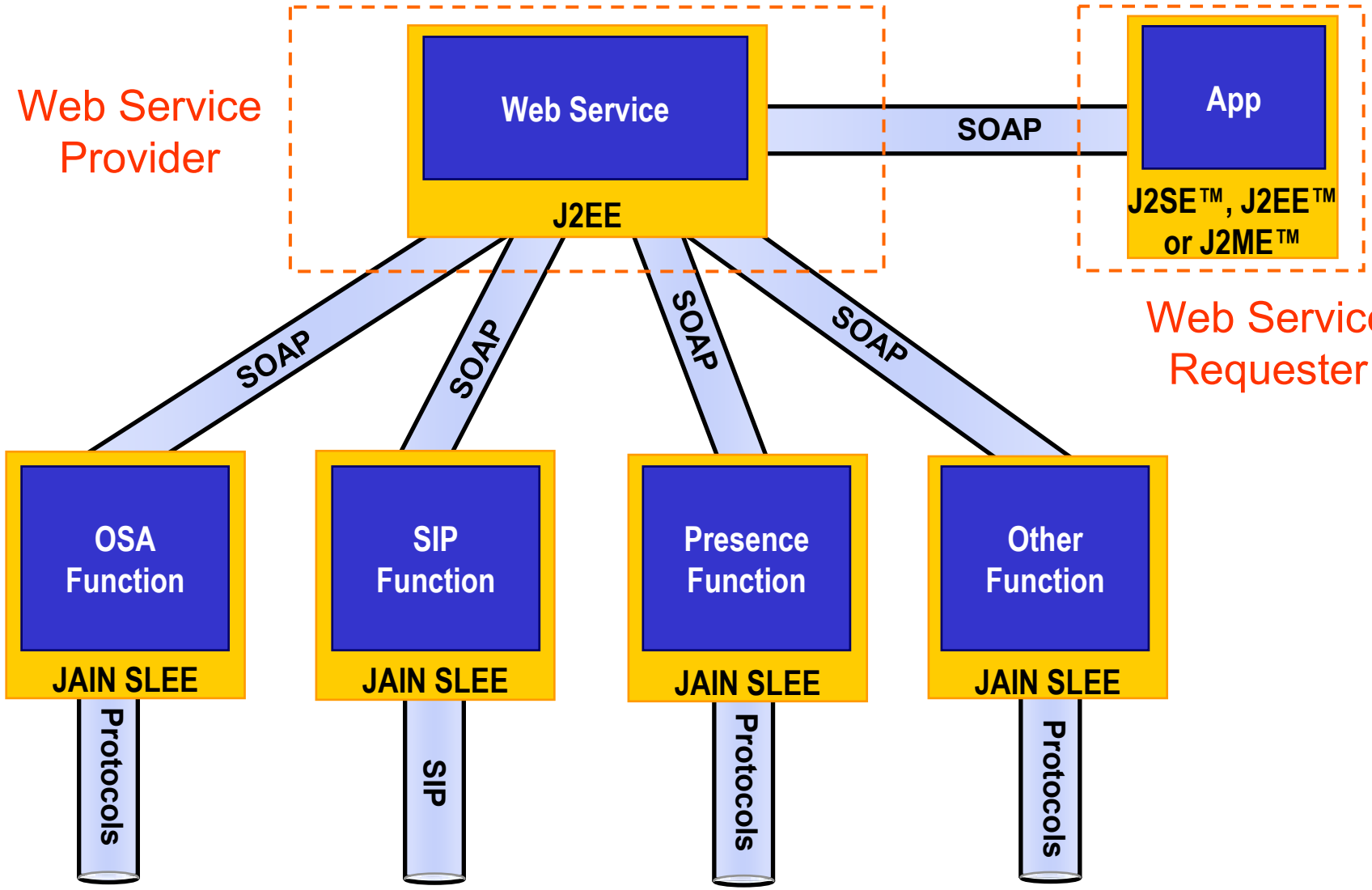
Communications Containers

Containers in orange boxes



© 2003 Sun Microsystems, Inc. All Rights Reserved.

Application Servers, OMA & OSA



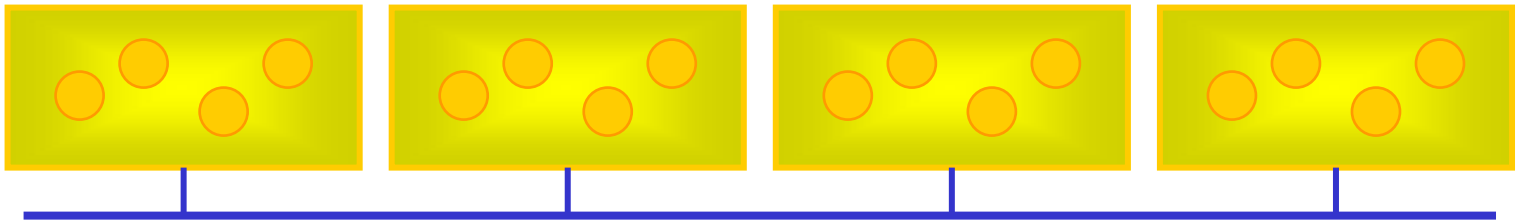
© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN Service Logic Execution Environment (SLEE)

© 2003 Sun Microsystems, Inc. All Rights Reserved.

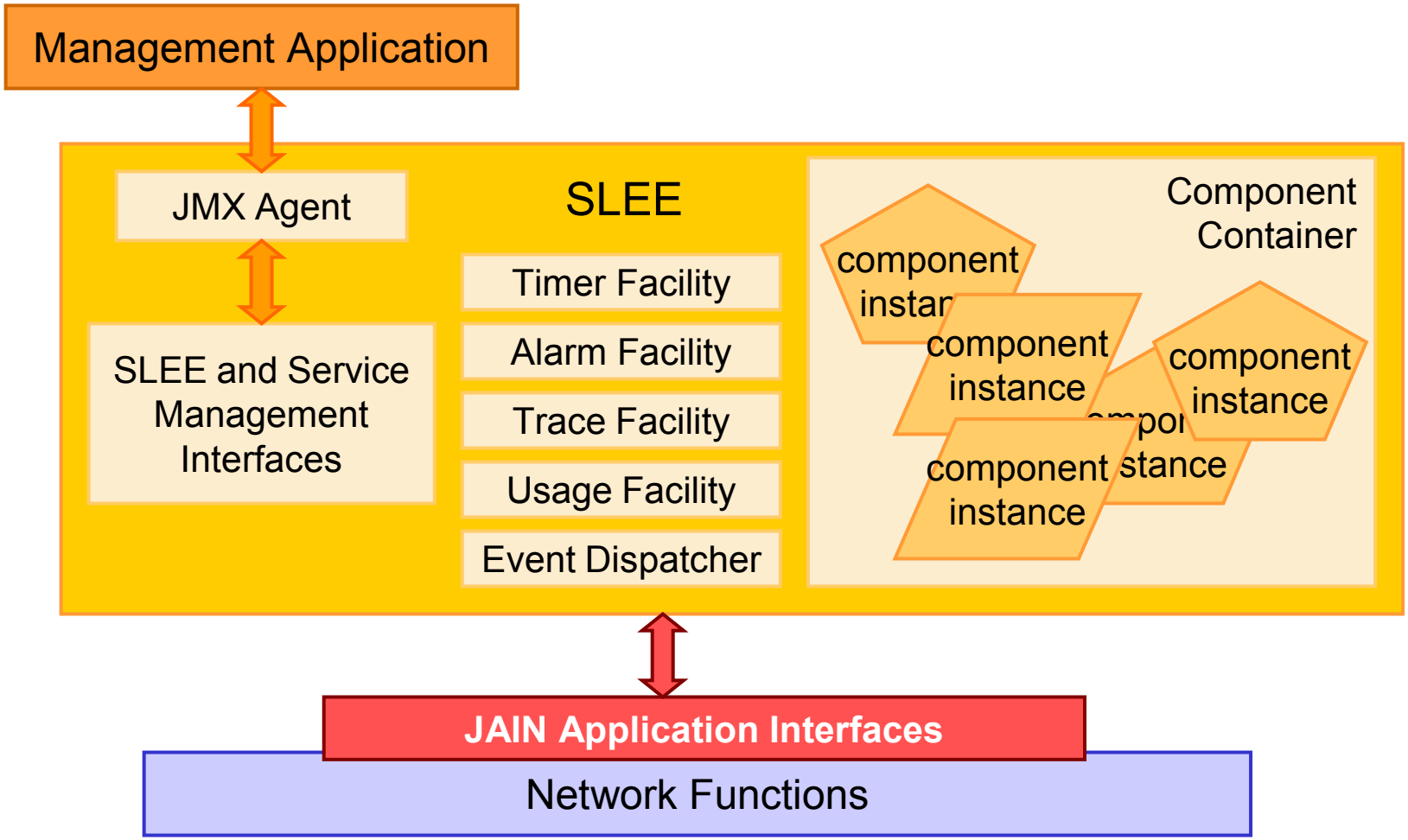
JAIN SLEE (JSR 22)

- *Low latency* and *high throughput* application server for *event processing*
  - Latency < 100 ms
  - 100's to 1000's of events per second
- Event optimized component model
- Designed for stringent requirements of event processing applications
- Distributed component model like EJB



© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN SLEE Architecture



© 2003 Sun Microsystems, Inc. All Rights Reserved.

Benefits of JAIN SLEE

- High performing platform for event driven applications
  - Supports simple and complex telecommunications applications
  - Applications deal with service logic only
    - System issues handled by container i.e. threading, transactions
    - Simplified application development through decoupled event consumers and event providers
- Standard application framework
  - Defined programming model
  - Object Orientated, asynchronous, robust and distributable
  - Event routing integrated to component model
- Independent of underlying networks
- Asynchronous support
  - Elaborate event distribution mechanism (with priority)
  - Maps events to method invocations on components
  - Creates component instances in response to initial events

SLEE reduces cost and improves time-to-market

© 2003 Sun Microsystems, Inc. All Rights Reserved.

SLEE Application Characteristics

	Communications	*Enterprise*
Invocations	Typically asynchronous - Events such as protocol triggers - Events occurrences mapped to method invocations	Typically synchronous - Database, EAI systems - RPC Calls
Event Granularity	Fine-grained events High Frequency	Course-grained events Low Frequency
Components	Light-weight fine-grained objects Short transient lifetimes - Rapid creation, deletion	Heavy weight data access objects Long persistent lifetimes
Data Sources	Multiple data sources - Location, context information - Provisioned data, cached from master copy	Database servers - Definitive master copy Back-end systems
Transactions	Light-weight transactions - For state replication demarcation - Faster completion and more frequent	Database transactions - Slower completion and less frequent
Computation	Compute-intensive - Processing is resource invocations & events	Database access intensive

© 2003 Sun Microsystems, Inc. All Rights Reserved.

SLEE Applications Characteristics

	Communications	*Enterprise*
Availability	3 to 5 9's	2 to 3 9's
Real-time	Soft real-time	
Deployment Distribution	Distributed deployment throughout network	Centralized deployment in small number of data centers
Nodes	1 to 4 CPUs	2 to 32 CPUs
Clusters	2 to 16 nodes	2 to 4 nodes

Applications characteristics drive Container Design!

© 2003 Sun Microsystems, Inc. All Rights Reserved.

Application Requirements

- Low latency
  - Time to setup a call < 500 ms
- High throughput
  - 1,000,000 Busy Hour Call Attempts (Operator specific)
- High availability, 99.999% expected
  - < 6 minutes downtime per year (planned and unplanned)
  - Different notion of availability
  - Small partial failures (< 5%) count against availability
- Graceful handling of load spikes above maximum system capacity

© 2003 Sun Microsystems, Inc. All Rights Reserved.

Low Latency Requirements

What is required to set up a call in 500 ms?

- Traverse 2–5 network nodes or application servers
- Minimum 2 protocol messages events per node (application dependant)
- 1 transaction per event
- Approx 4 – 10 events to process within 500 ms

  (excluding propagation delay, some processing overlapped with propagation)
- 50 – 125 ms per transaction

  (for simple transactions with redundancy for single failure tolerance)

© 2003 Sun Microsystems, Inc. All Rights Reserved.

High Throughput Requirements

What does 1,000,000 BHCA require?

- 2 – 10 protocol messages or events per call (application dependant)
- 1 transaction per event
- Approx 280 call attempts per second
- 560 – 2800 transactions per second

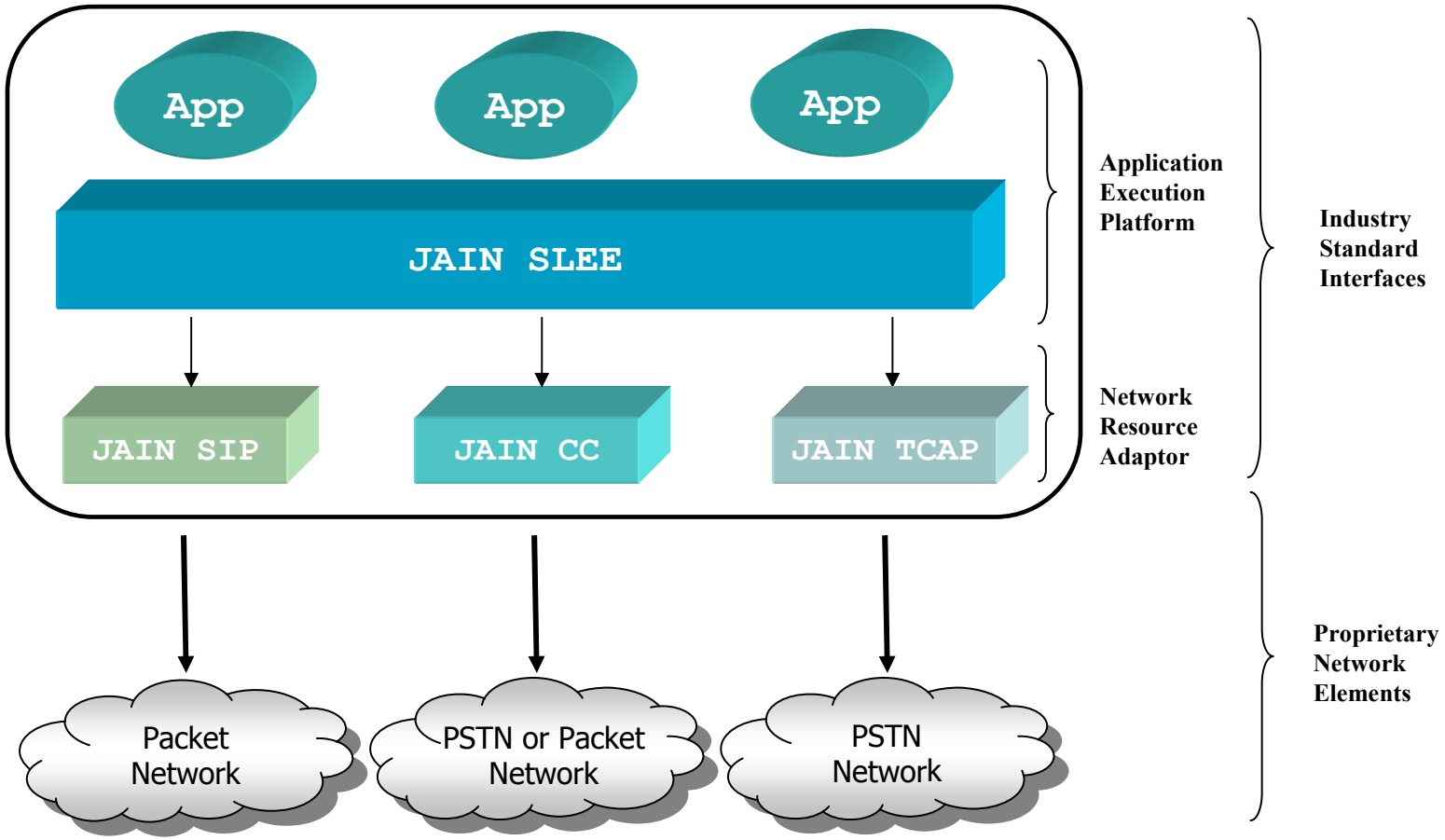
© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN SLEE in Communications Networks

- 2G and 2.5G Networks
  - Service Control Point
  - Service Node (2G to 3G using a Media Gateway)
- 3G Networks
  - Service Switching Control Point
  - Service Switching Point
  - SIP Proxy
  - 3GPP IMS CSCFs
- Convergent Networks
  - Gatekeeper
  - Common Service Delivery Platform
  - Convergent SCP, SSCP, SSP
  - OSA Gateway

© 2003 Sun Microsystems, Inc. All Rights Reserved.

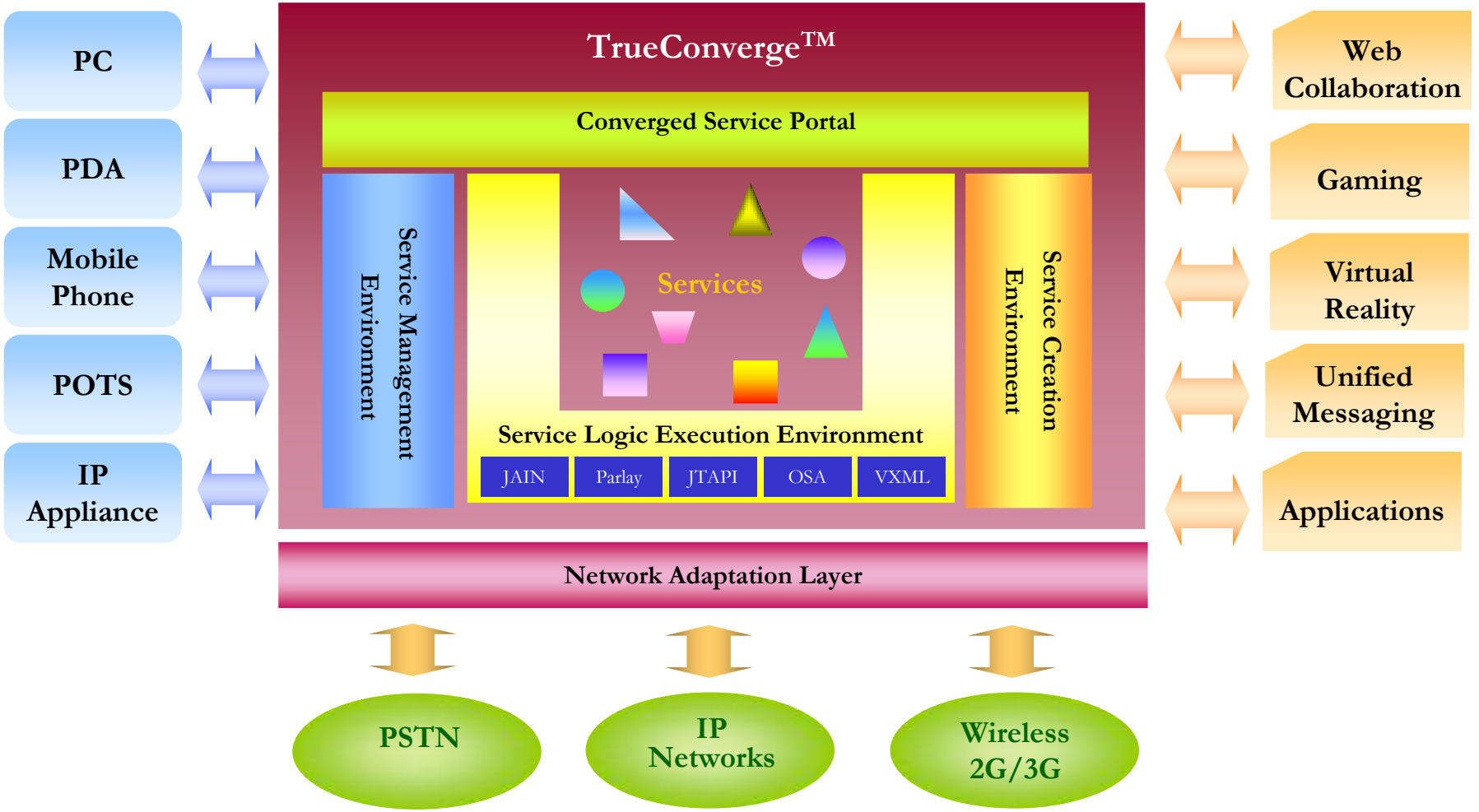
JAIN SLEE & Integrated Networks



© 2003 Sun Microsystems, Inc. All Rights Reserved.

TrueTel: First live deployment in Taiwan

– Eastern Broadband Telecom



© 2002 TrueTel Communications Inc, All Rights Reserved

SIP Servlet Execution Environment

© 2003 Sun Microsystems, Inc. All Rights Reserved.

SIP Servlet (JSR 116)

- SIP Servlets typically reside on network servers
  - make routing decisions
- Supported RFC's:
  - RFC 3261, 3262, 3265, 3428, 2976
- Utilizes http servlet model as foundation
  - builds on http generic part for SIP request and response functionality
- Applications perform a complete set of SIP signaling actions
  - User agent client (UAC)
  - User agent server (UAS)
  - Proxy server

© 2003 Sun Microsystems, Inc. All Rights Reserved.

SIP Servlet Goals

- Simplicity for the application developer
  - Containers handle “non-essential” complexity
- Containers support converged applications
  - Applications that span multiple protocols and media types
- Third party application development
  - An XML DD is used to communicate application information from the application developer to deployers
- Application composition
  - Several applications can execute on the same incoming or outgoing request or response
  - Each application has its own set of rules and executes independently

© 2003 Sun Microsystems, Inc. All Rights Reserved.

Extension to HTTP Servlet Model

- HTTP is not a peer-to-peer protocol like SIP therefore SIP Servlet extensions include:
  - Initiate requests
  - Receive responses as well as requests
  - Generate multiple Responses
    - one or more 1xx followed by a final response
  - Proxying requests, possibly to multiple destinations

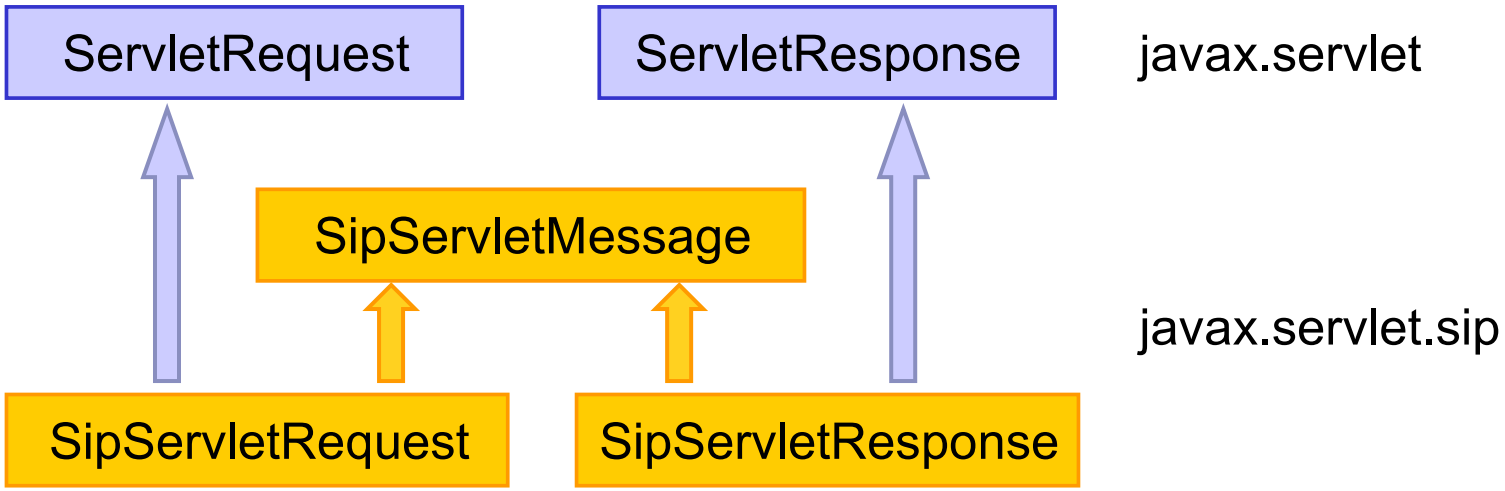
© 2003 Sun Microsystems, Inc. All Rights Reserved.

Protocol and Application Sessions

- SIP Servlet defines two types of sessions:
  - SipSession (Protocol Session)
    - Equivalent to HTTP Session and represents a Dialog in SIP
  - SipApplicationSession
    - Provides storage for application data
    - Enables different protocol sessions to share state i.e., HTTP and SIP
    - Defined by SIP Servlet but is expected to be adopted by the HTTP Servlet specification in a future release.

© 2003 Sun Microsystems, Inc. All Rights Reserved.

Message Hierarchy

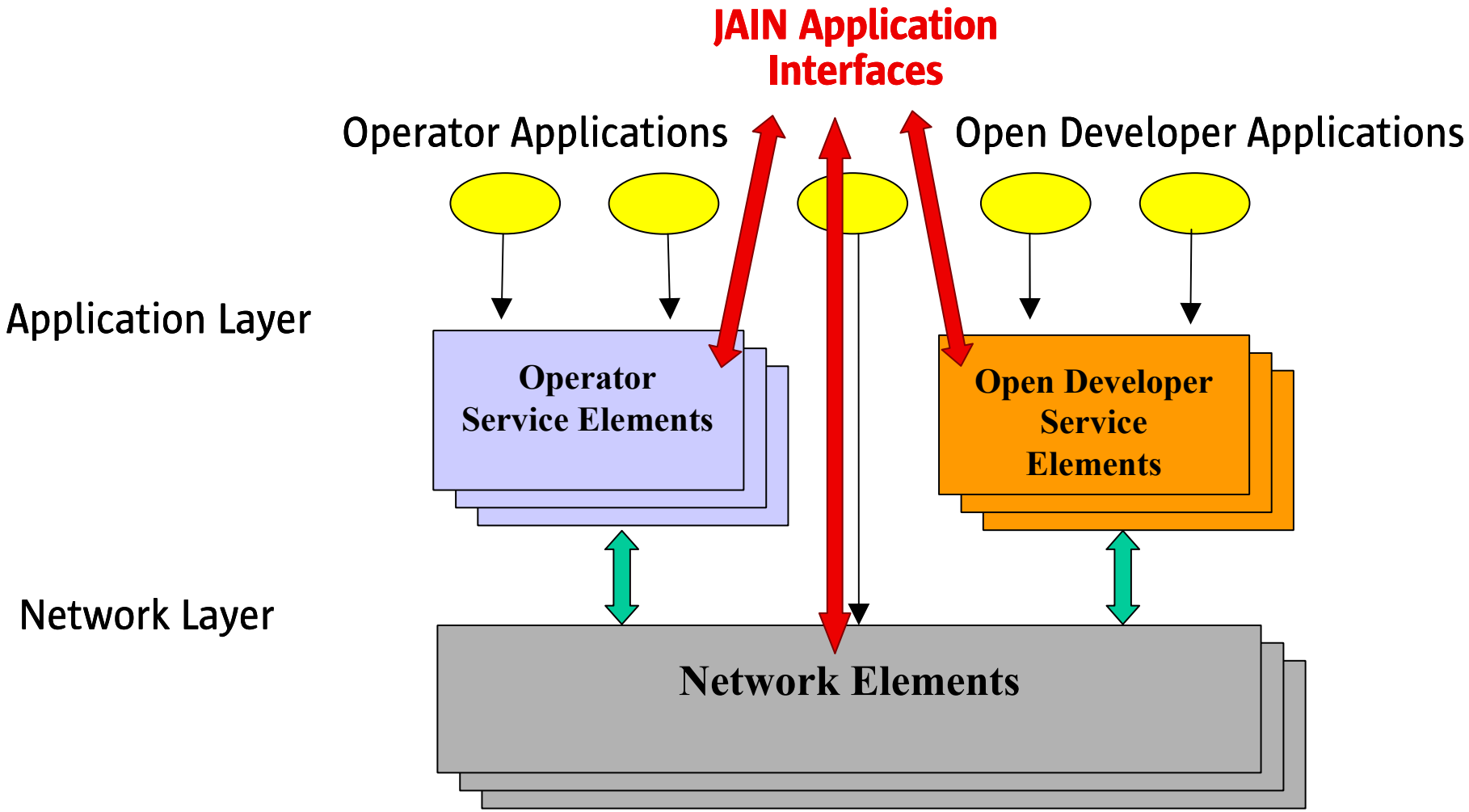


© 2003 Sun Microsystems, Inc. All Rights Reserved.

Application Interfaces

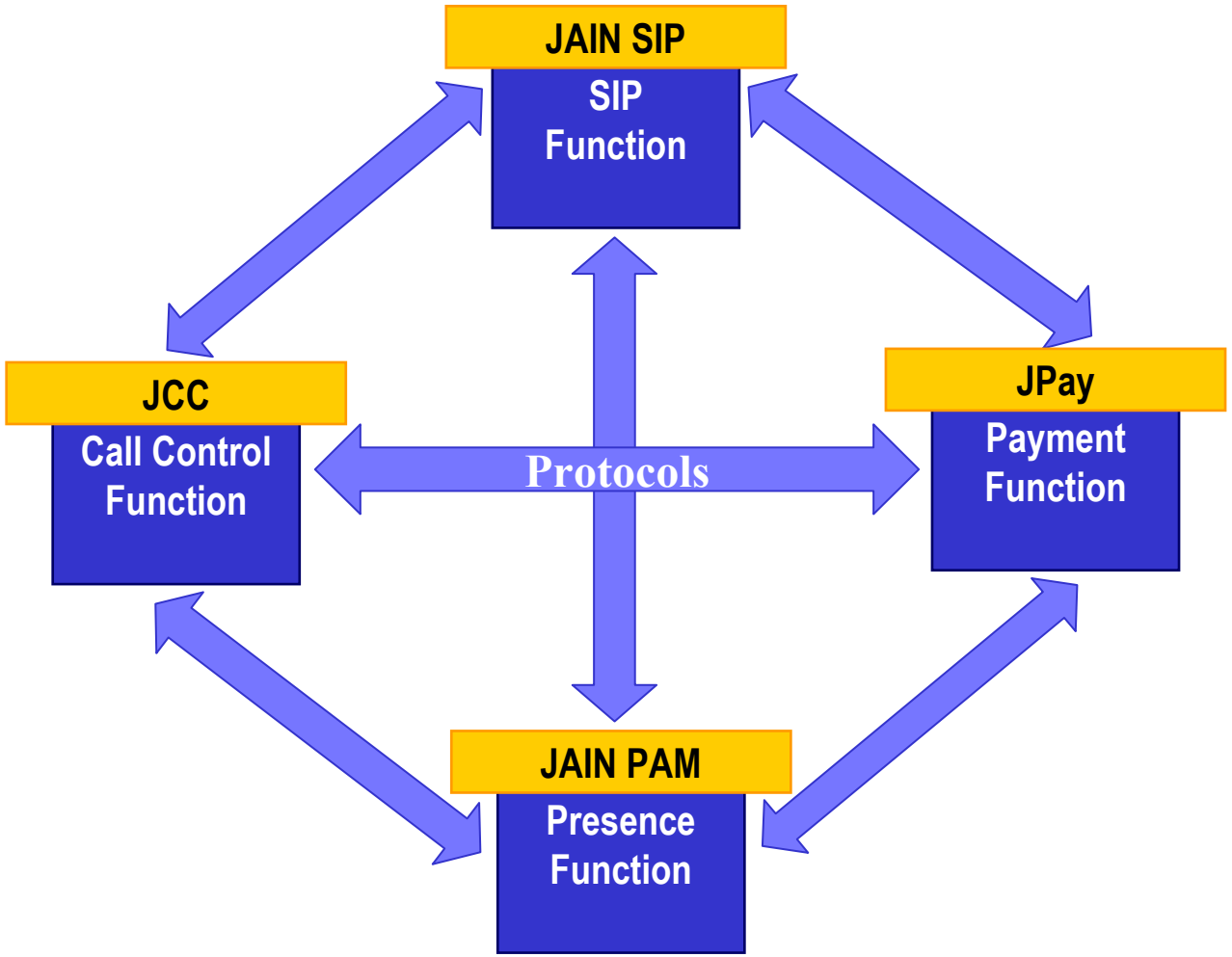
© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN Application Interfaces



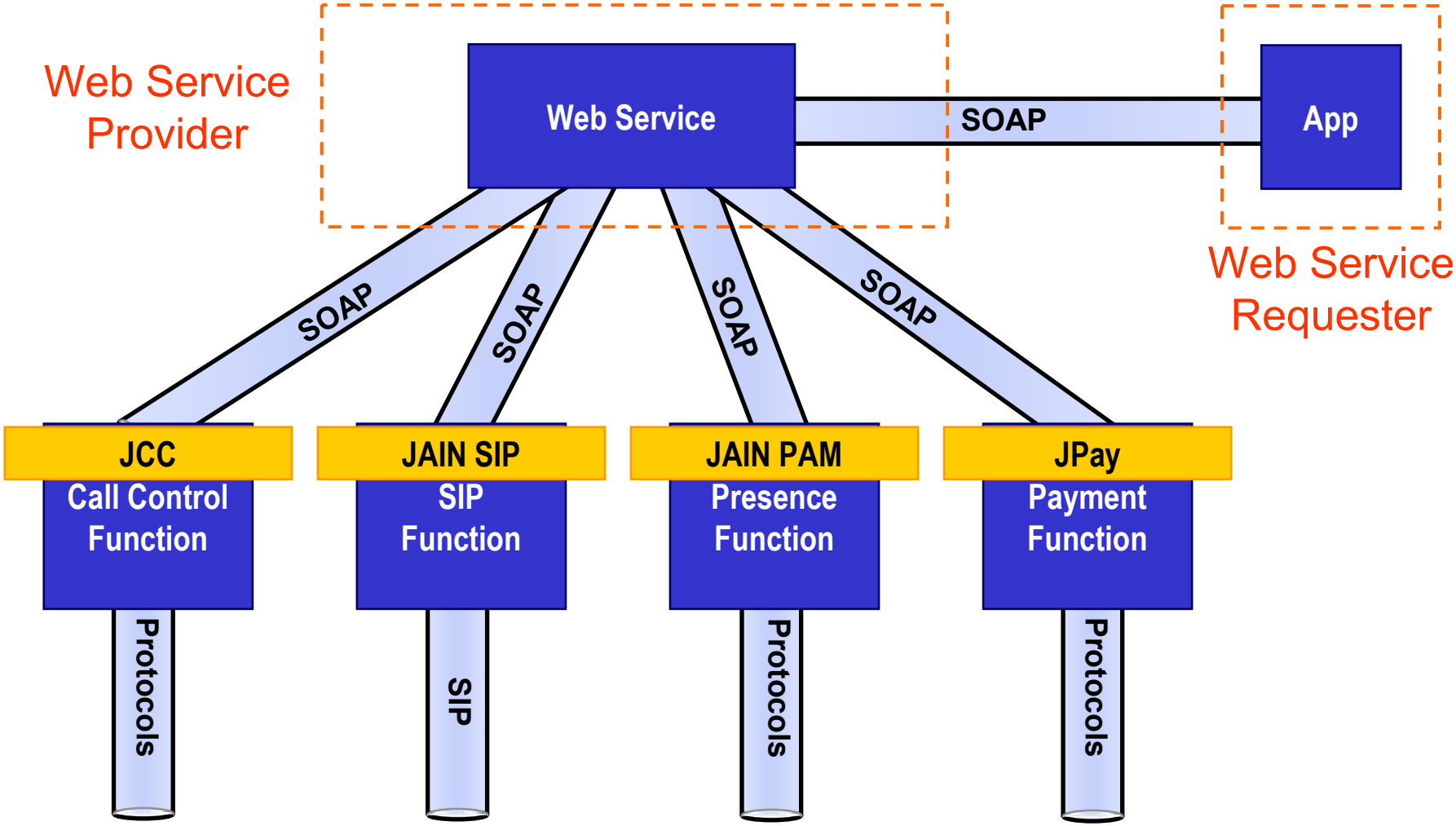
© 2003 Sun Microsystems, Inc. All Rights Reserved.

Network Functions & Application Interfaces



© 2003 Sun Microsystems, Inc. All Rights Reserved.

Application Interfaces & OMA Web Services



© 2003 Sun Microsystems, Inc. All Rights Reserved.

Session Initiation Protocol (SIP)

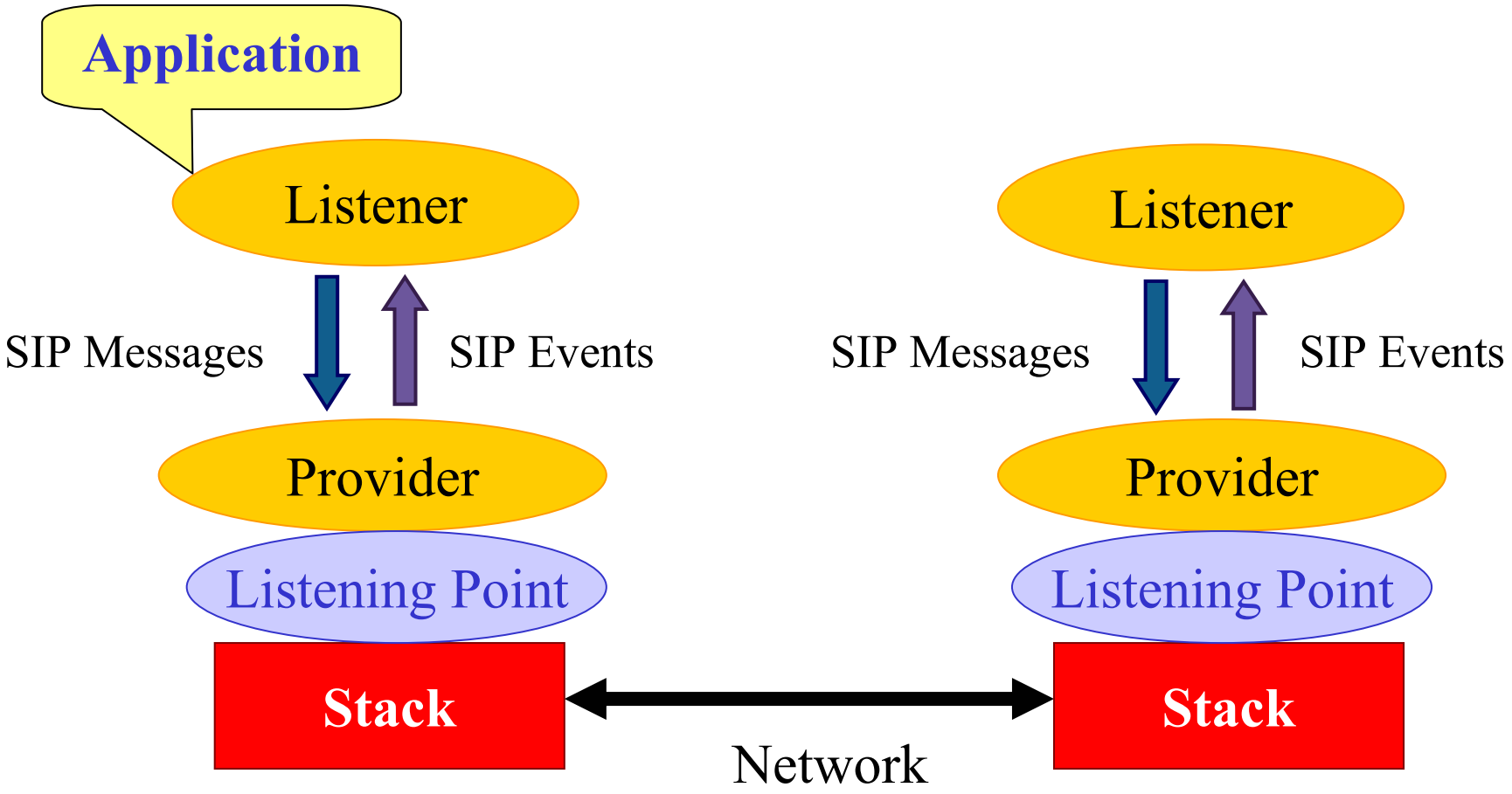
© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN SIP (JSR 32)

- Java-standard interface to a SIP signaling stack
  - Standardizes the interface to the stack
  - Standardizes the events and event semantics
  - Application portability - verified via the TCK
- Designed for the developer who requires powerful access to the SIP protocol
- JAIN SIP can be utilized in a user agent, proxy, or embedded into a service container
- Supported RFCs:
  - RFC 3261, 2976, 3262, 3265
  - RFC 3311, 3428, 3515

© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN SIP Architecture



© 2003 Sun Microsystems, Inc. All Rights Reserved.

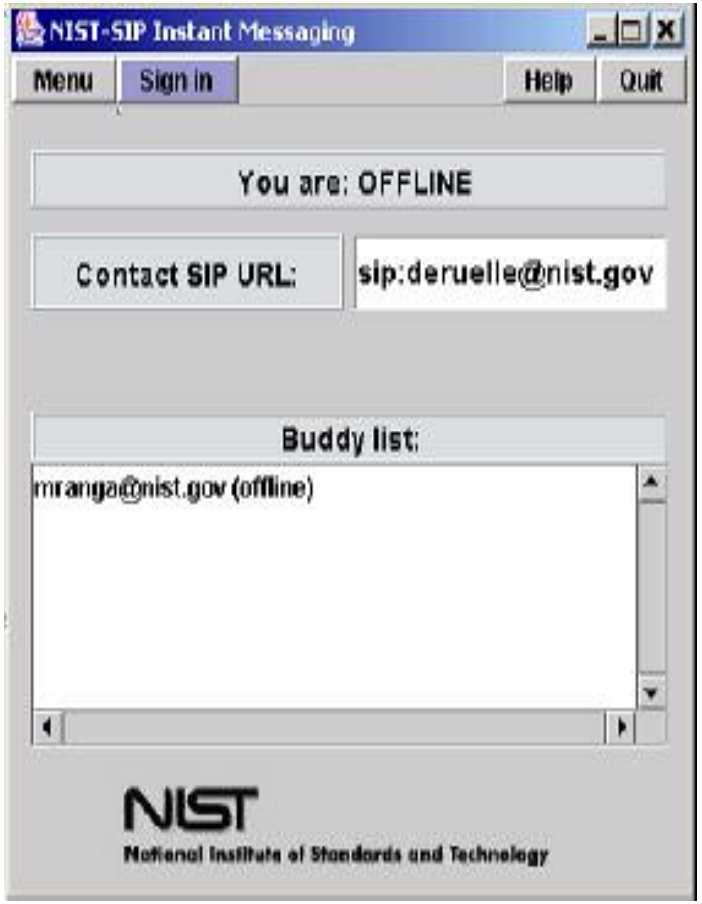
Features provided by JAIN SIP

- Provide methods to format and send SIP messages
- Parse incoming messages
  - enables application to access fields via standardized interface
- Invoke appropriate application handlers when protocol significant
  - message arrivals, Transaction time-outs
- Provide Transaction support
  - manage Transaction state and lifetime on behalf of application
- Provide Dialog support
  - manage Dialog state and lifetime on behalf of application

© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN SIP & Instant Messaging

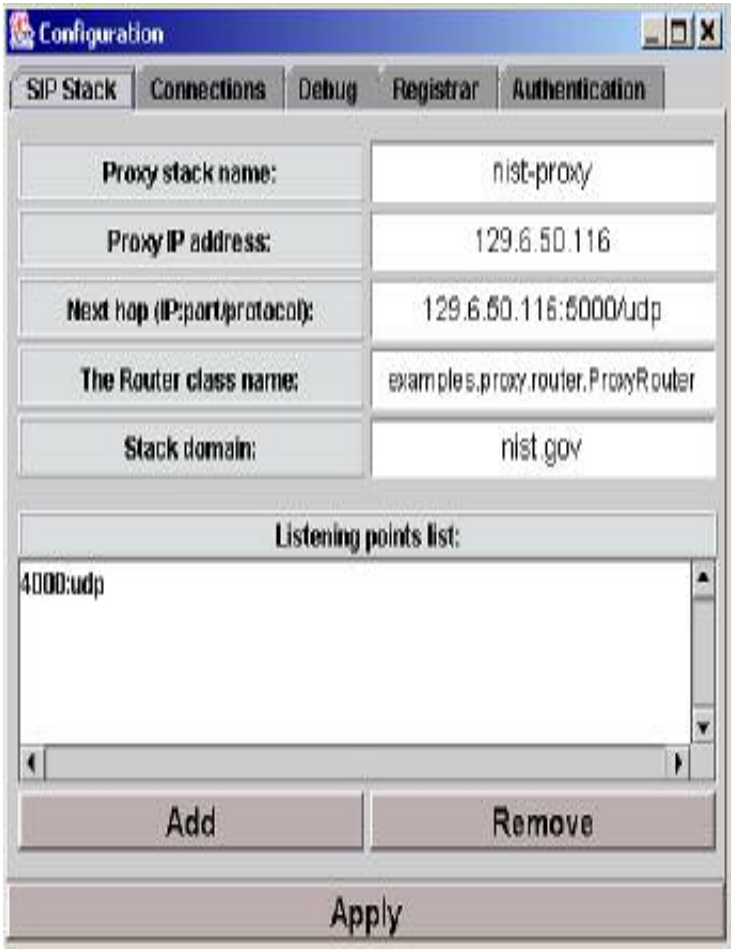
- Suitable for IM and Presence Clients and Servers
- Supports required methods and Headers
- Creates and manages Dialogs for SUBSCRIBE and MESSAGE methods
- JAIN IM Client SipListener is about 1100 LOC
- Interoperates with Microsoft IM and packaged with RI



© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN SIP in Proxy Servers

- Facilitates construction of Proxy Servers
  - Stateless, Transaction-stateful, and Dialog-stateful operation
- Access to Dialog/Transaction state and route tables
- Extensibility and application controlled Routing
- Deep copy semantics for cloning
- Proxy (including presence server) is about 3500 LOC and is included with RI



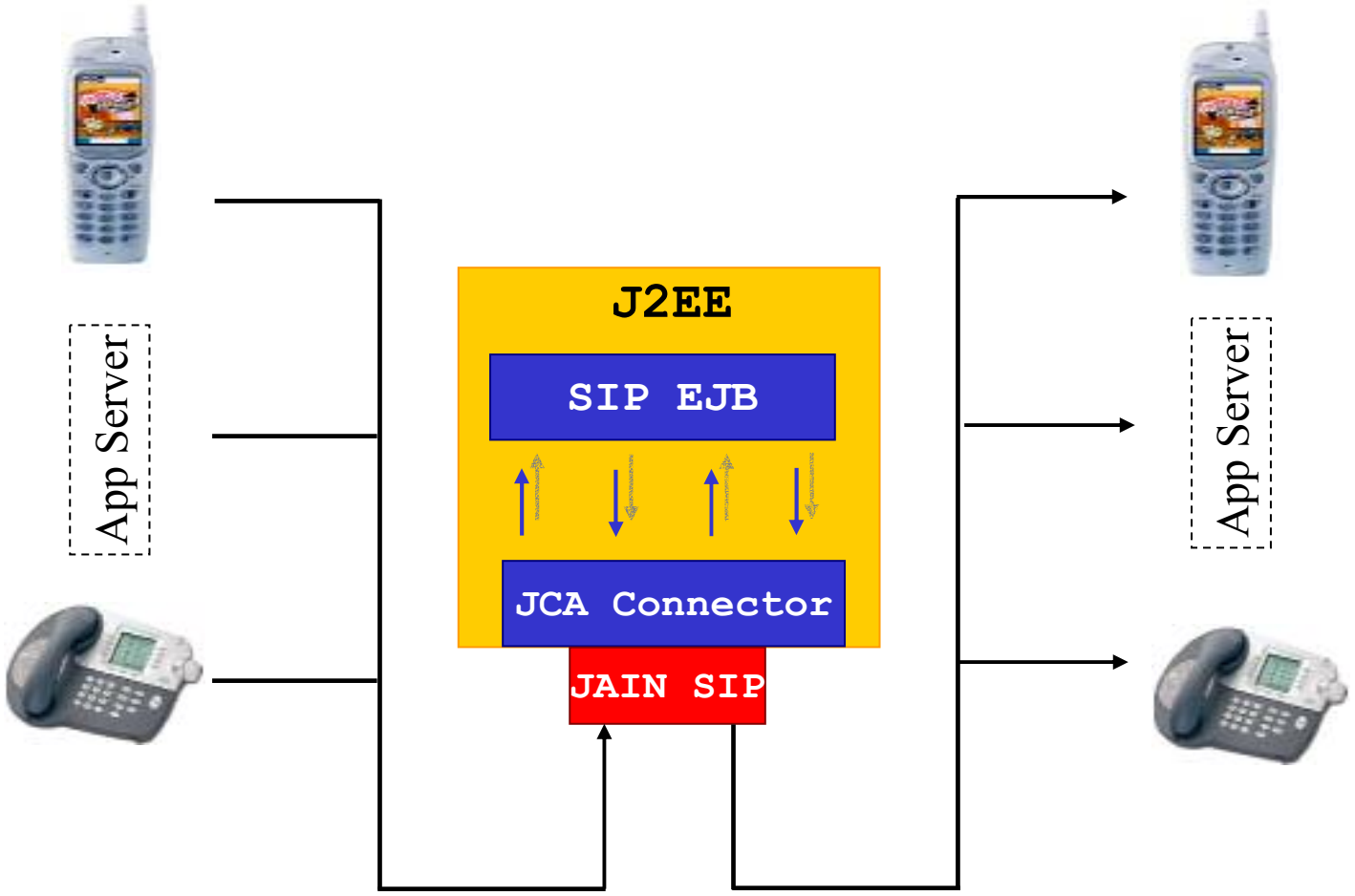
© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN SIP Support in JAIN SLEE

- JAIN SLEE represents network resources as resource adaptors and each resource adaptor has a type
  - Resource adaptor type for JAIN SIP is ‘javax.sip’
- JAIN SLEE identifies Event by Event types
  - JAIN SIP Events are classified RequestEvents, ResponseEvents and TimeoutEvents, each of these classifications contains numerous types
  - For example the event type of a Request message of type ‘INVITE’ is ‘javax.sip.RequestEvent.Request.INVITE’
- JAIN SLEE represents the flow of events as activities
  - Activity Objects in JAIN SIP are ClientTransactions (locally initiated) and ServerTransactions (remotely initiated)

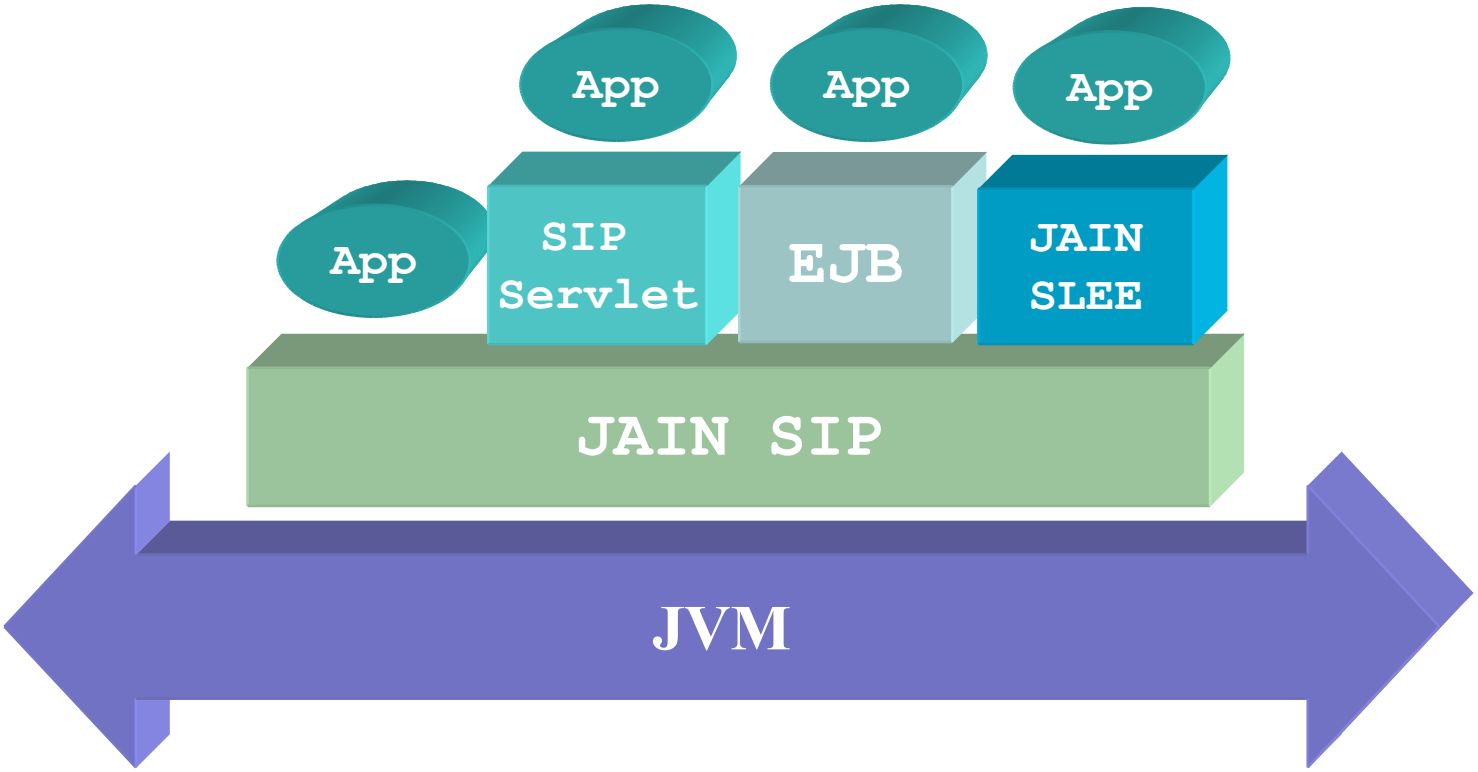
© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN SIP Support in J2EE



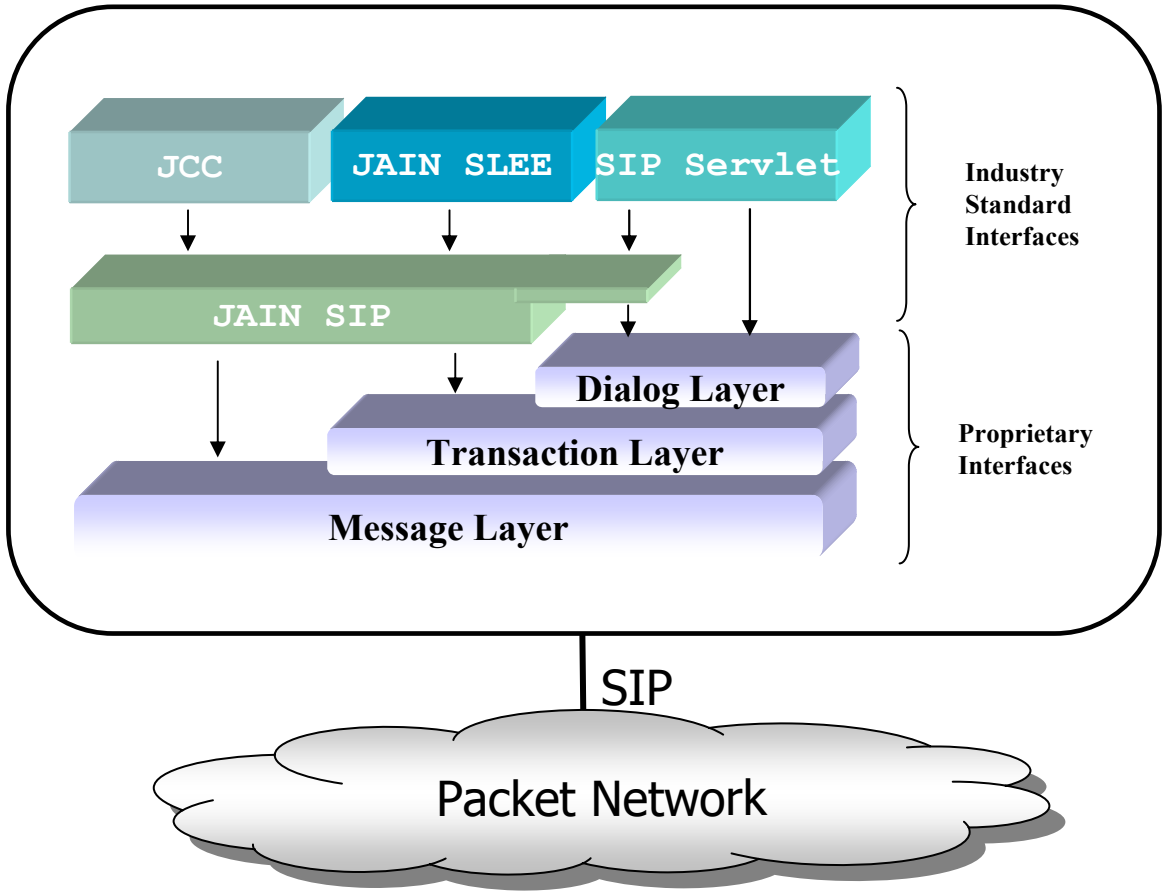
© 2003 Sun Microsystems, Inc. All Rights Reserved.

SIP Execution Environments



© 2003 Sun Microsystems, Inc. All Rights Reserved.

Relative SIP Network Interfaces



© 2003 Sun Microsystems, Inc. All Rights Reserved.

SIP for J2ME (JSR 180)

- SIP for J2ME is the standardized SIP interface for mobile handsets
- SIP for J2ME is an optional package for the J2ME platform
  - enables resource limited devices to send and receive SIP messages
- Designed for the CLDC profile, however can be used with the CDC profile
- Gives transactional control over the SIP protocol
- Client devices must support SIP for Rel5.0 of the UMTS architecture.
  - SIP for J2ME is the perfect platform for these client devices

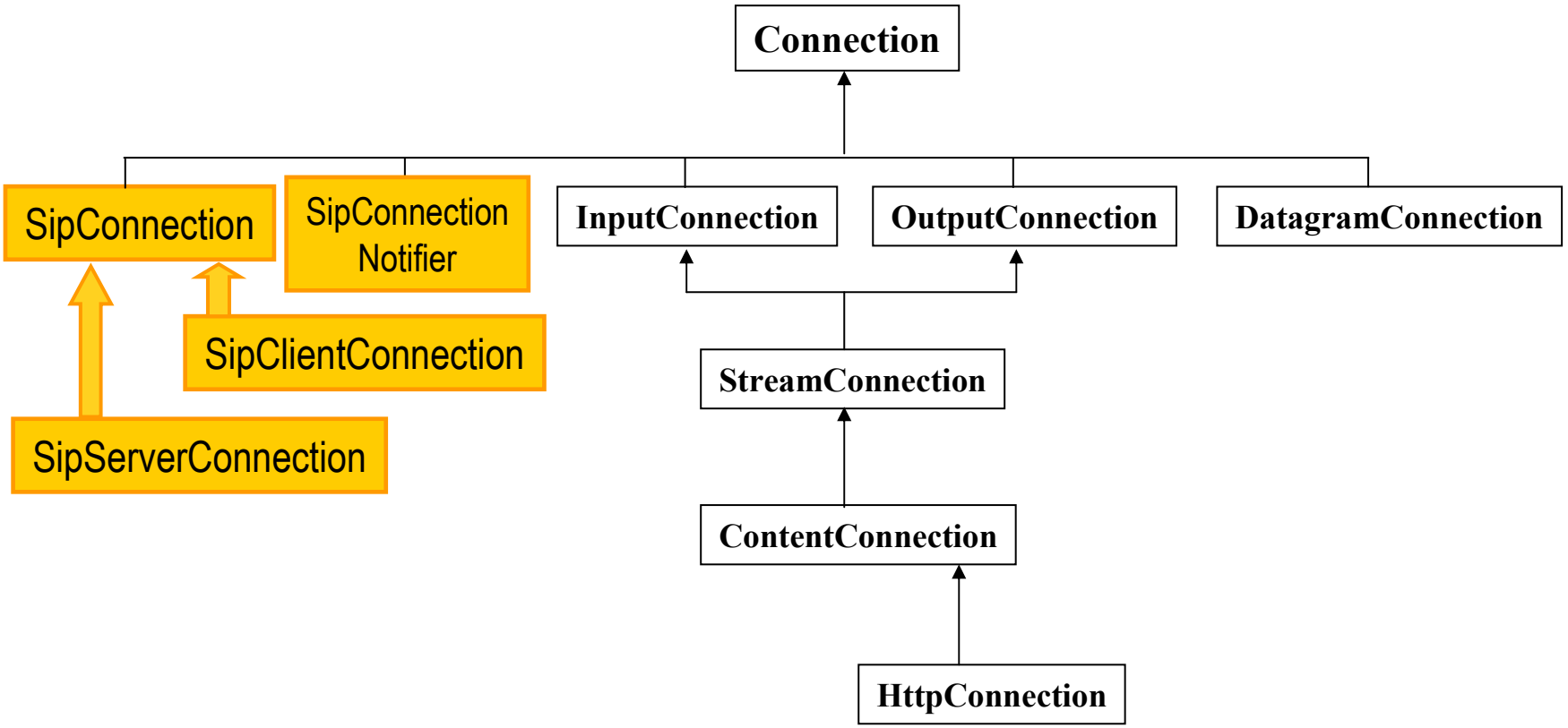
© 2003 Sun Microsystems, Inc. All Rights Reserved.

SIP for J2ME Goals

- Enables terminals supporting CLDC to run SIP enabled MIDlets
- Builds upon CLDC Generic Connection framework
- Specifically targeted at mobile phone handsets
- Retains the “look and feel” of the HTTP API
- Ensures small API size and keeps the number of created objects low
- Provides developers with helper functions
  - RefreshHelper for Register and Subscribe

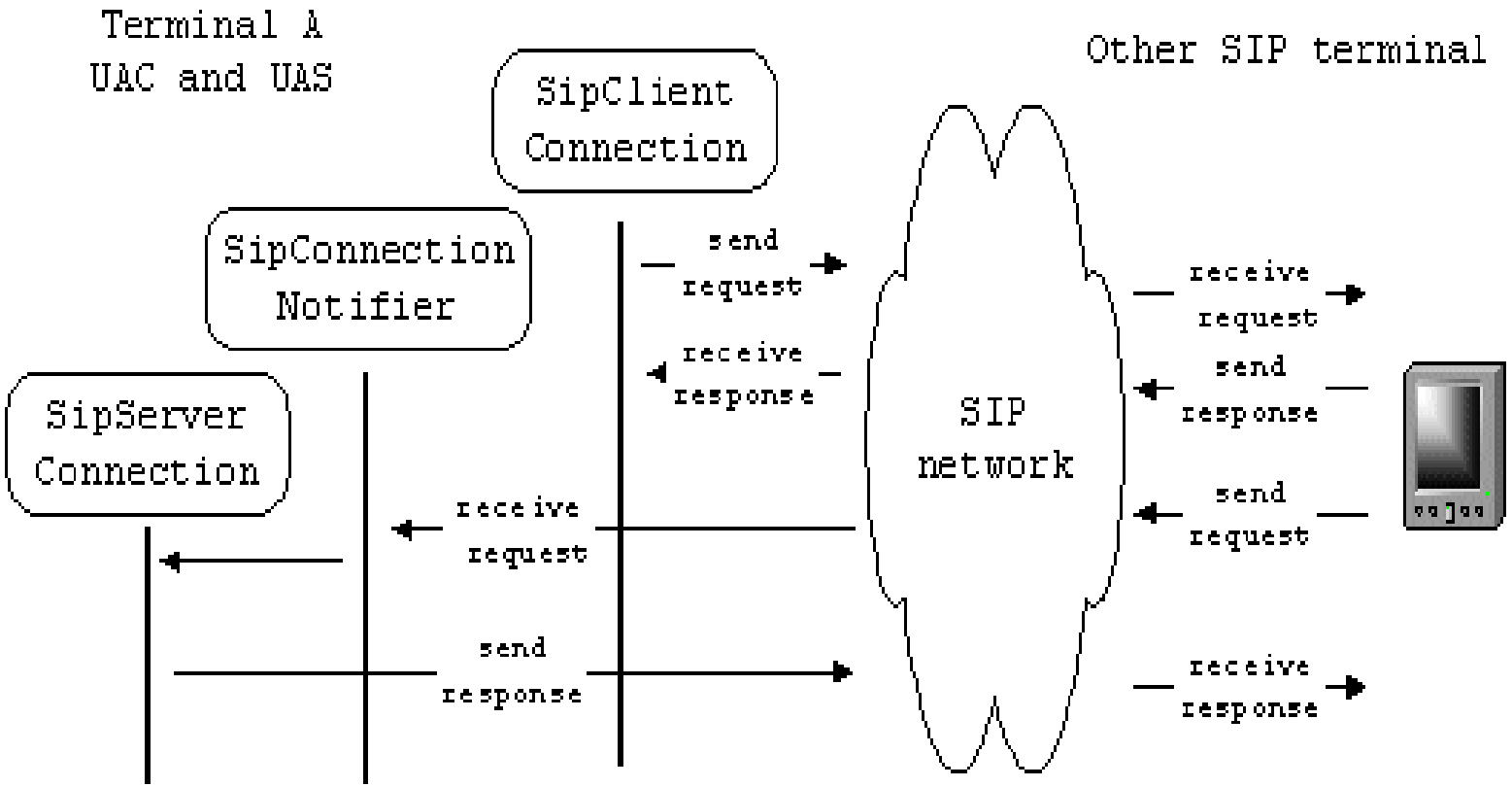
© 2003 Sun Microsystems, Inc. All Rights Reserved.

SIP for J2ME & the Generic Connection Framework



© 2003 Sun Microsystems, Inc. All Rights Reserved.

SIP for J2ME Architecture



© 2003 Sun Microsystems, Inc. All Rights Reserved.

Application Development Options

Specification	Developer Community	Target Java Platform
JAIN SIP	Desktop	J2SE
JAIN SIP & J2EE™ Connector	Enterprise	J2EE Business Tier
JAIN SIP & JAIN SLEE RA	Telecom	J2EE Business Tier
SIP Servlet	Enterprise	J2EE Web Tier
SIP for J2ME	Device	J2ME

© 2003 Sun Microsystems, Inc. All Rights Reserved.

Presence and Instant Messaging

© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN Presence & JAIN IM

- JAIN Presence (JSR 186)
  - A client-side API for Presence
    - Watcher and Publisher modes
    - J2ME (thin clients) and J2SE (thick clients) based
  - Receives and publishes presence information to presence (and availability management) servers
- JAIN IM (JSR 187)
  - A client-side API for Instant Messaging
    - J2ME (thin clients) and J2SE (thick clients) based
  - Receives and publishes instant messages to IM servers
  - Orthogonal yet complementary to JAIN Presence

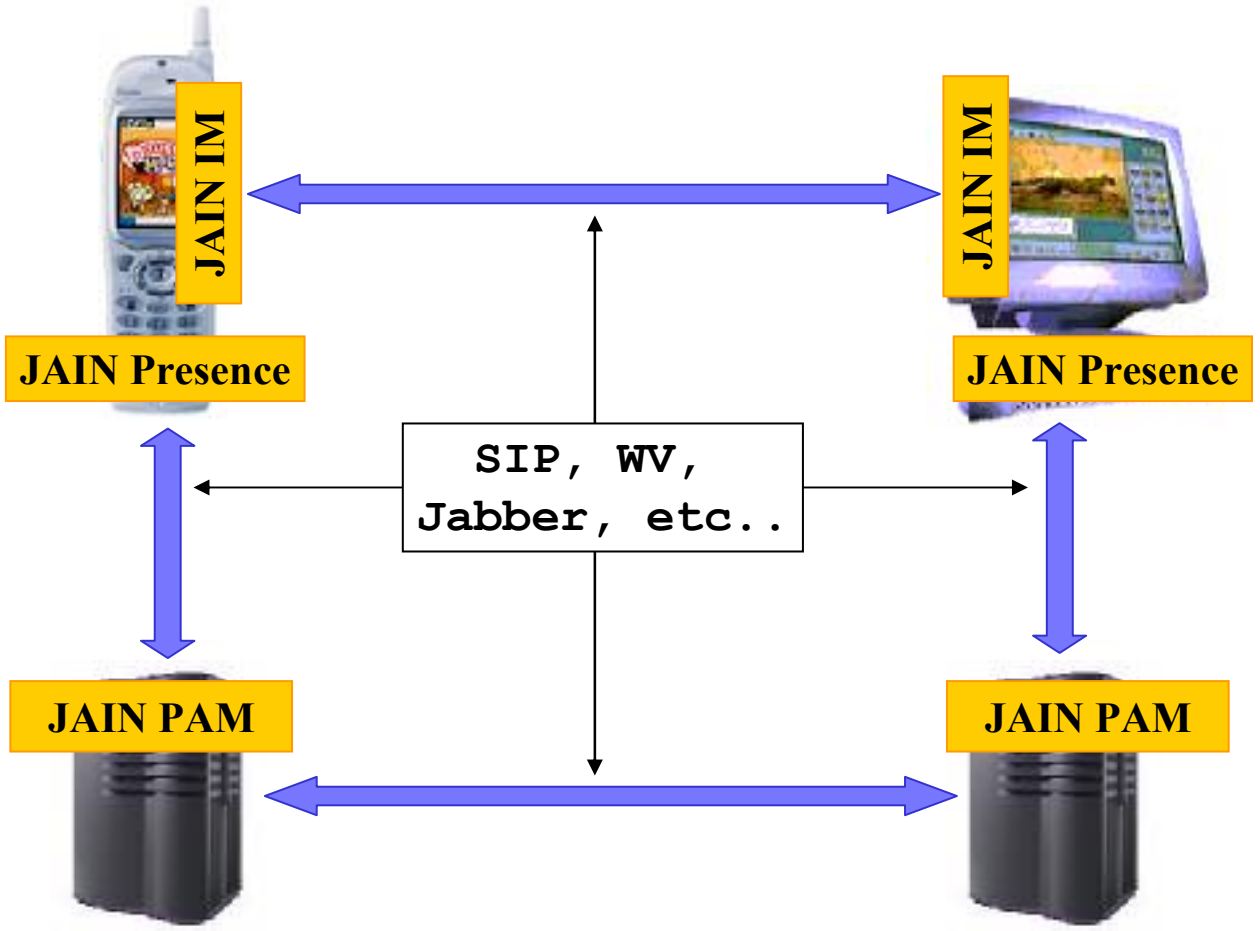
© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN PAM (JSR 123)

- A server-side API for Presence and Availability Management
  - J2EE (network servers) based
- Receives presence information from a variety of devices
  - location servers, presence servers, mobile terminals, etc.
- Publishes presence information to a variety of devices
  - presence servers, mobile terminals, etc.
- Manages the availability of users based on the identity of the presence requestor and the communications mode

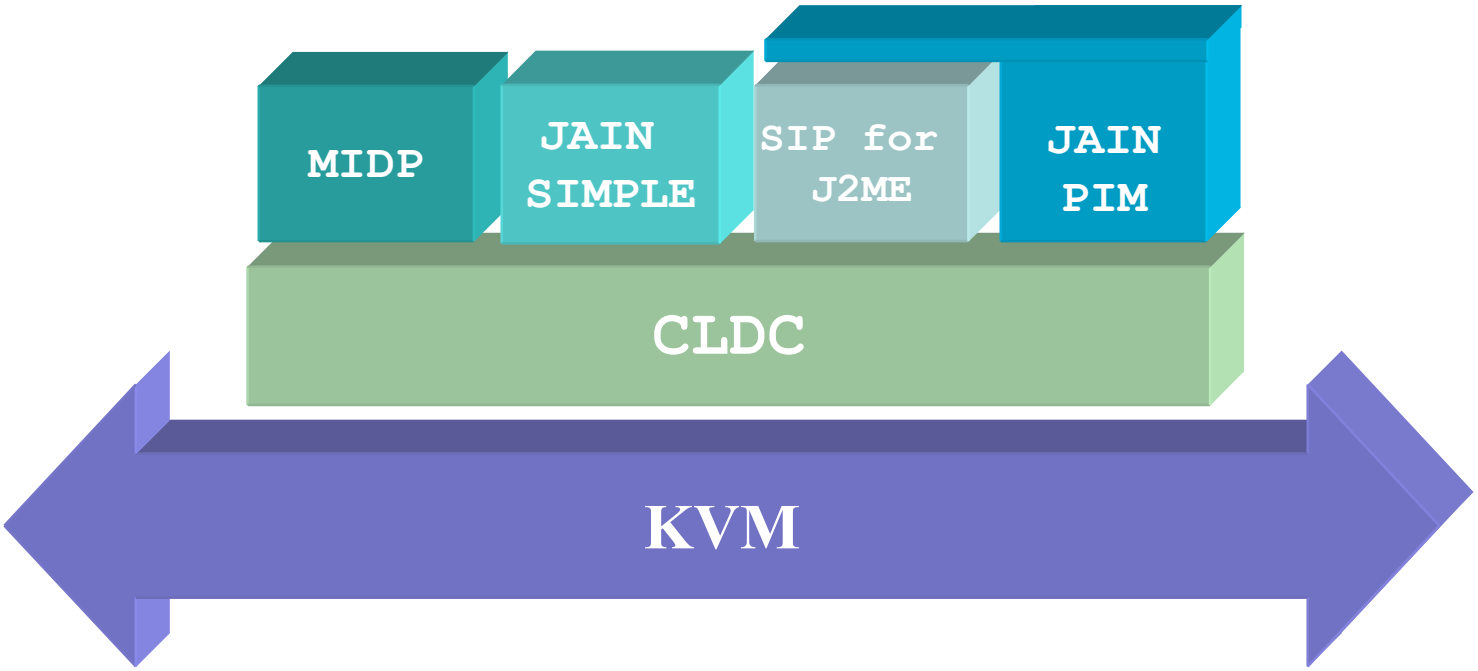
© 2003 Sun Microsystems, Inc. All Rights Reserved.

Presence & Instant Messaging



© 2003 Sun Microsystems, Inc. All Rights Reserved.

SIP and IM Interfaces for J2ME



© 2003 Sun Microsystems, Inc. All Rights Reserved.

Call Control

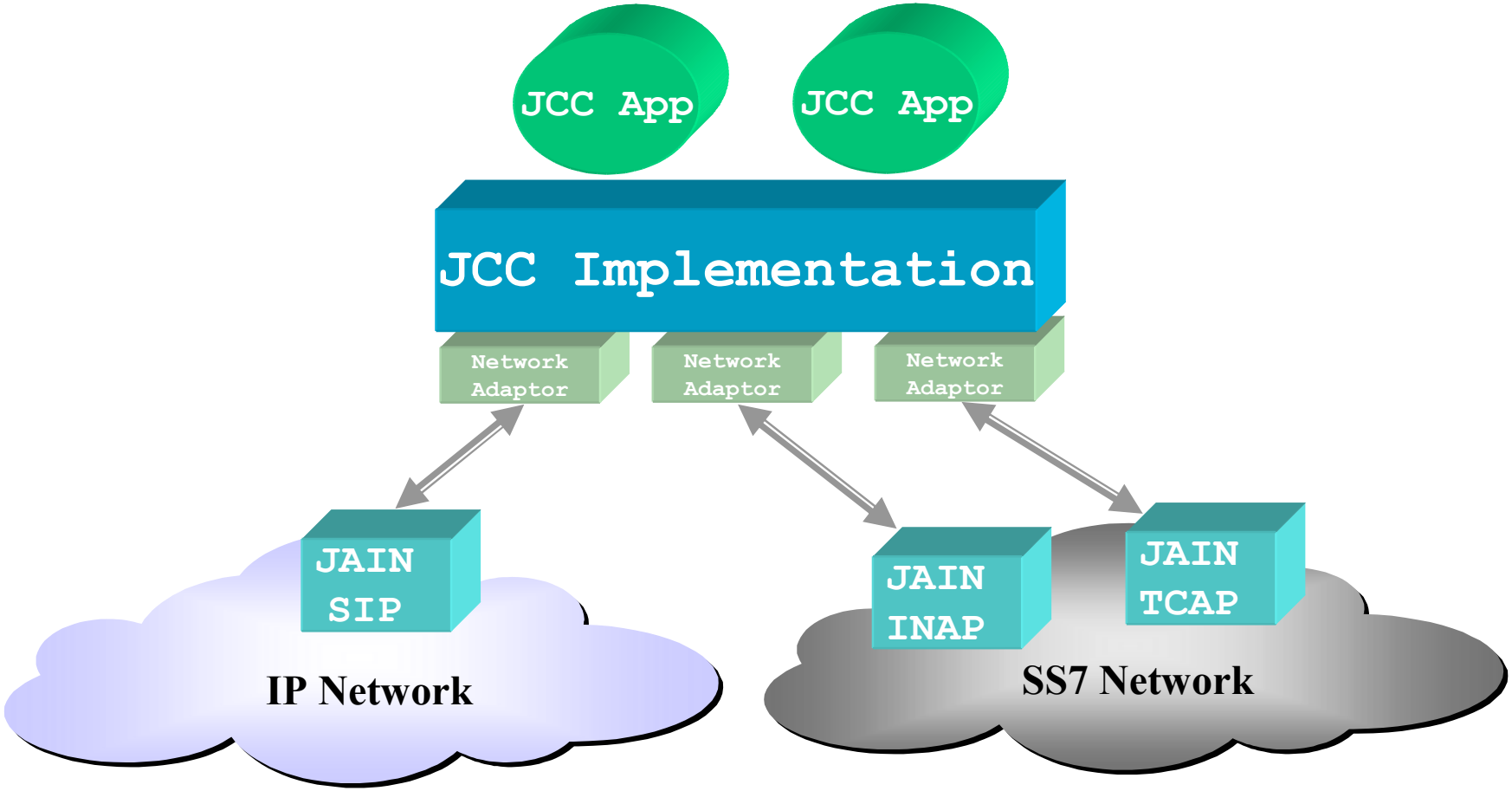
© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN Call Control – JCC (JSR 21)

- Call manipulation
  - observing, initiating, answering and processing
- Enables unified call control for multimedia, multiparty, multi-protocol sessions over underlying integrated networks
- **Simple** yet provides a reasonably rich set of functions
- Supports first-party as well as third-party calls
- Service drivers include:
  - 800/900 Number Translation, Wake-up Call
  - Voice Activated Dialing, Click to Dial

Copyright ©2001 Telcordia Technologies. All Rights Reserved.

JAIN Call Control Realization



© 2003 Sun Microsystems, Inc. All Rights Reserved.

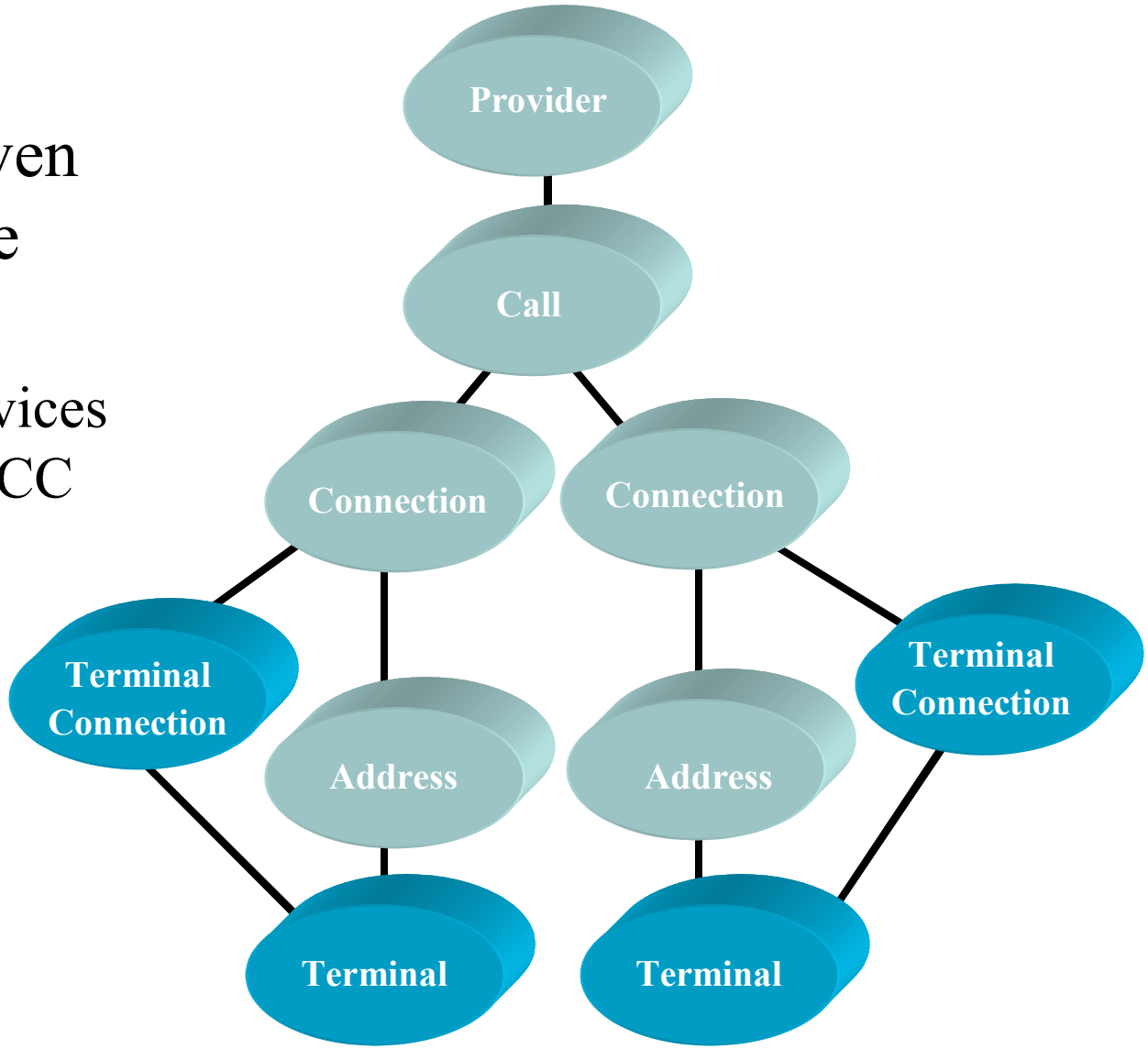
JAIN CAll control exTensions – JCAT (JSR 122)

- A rich call model to address extended call control
  - Extends JAIN Call Control v1.1 with concepts to model and control terminal capabilities
- Addresses end-office environment
  - Class 5, local exchange
- Supports first-party as well as third-party calls
- Introduces richer FSM, Terminal and Terminal Connection objects
  - Explicitly models terminals (unlike JCC)
  - JCC's state transitions models are enriched and more control is provided over its processing

Copyright ©2001 Telcordia Technologies, All Rights Reserved.

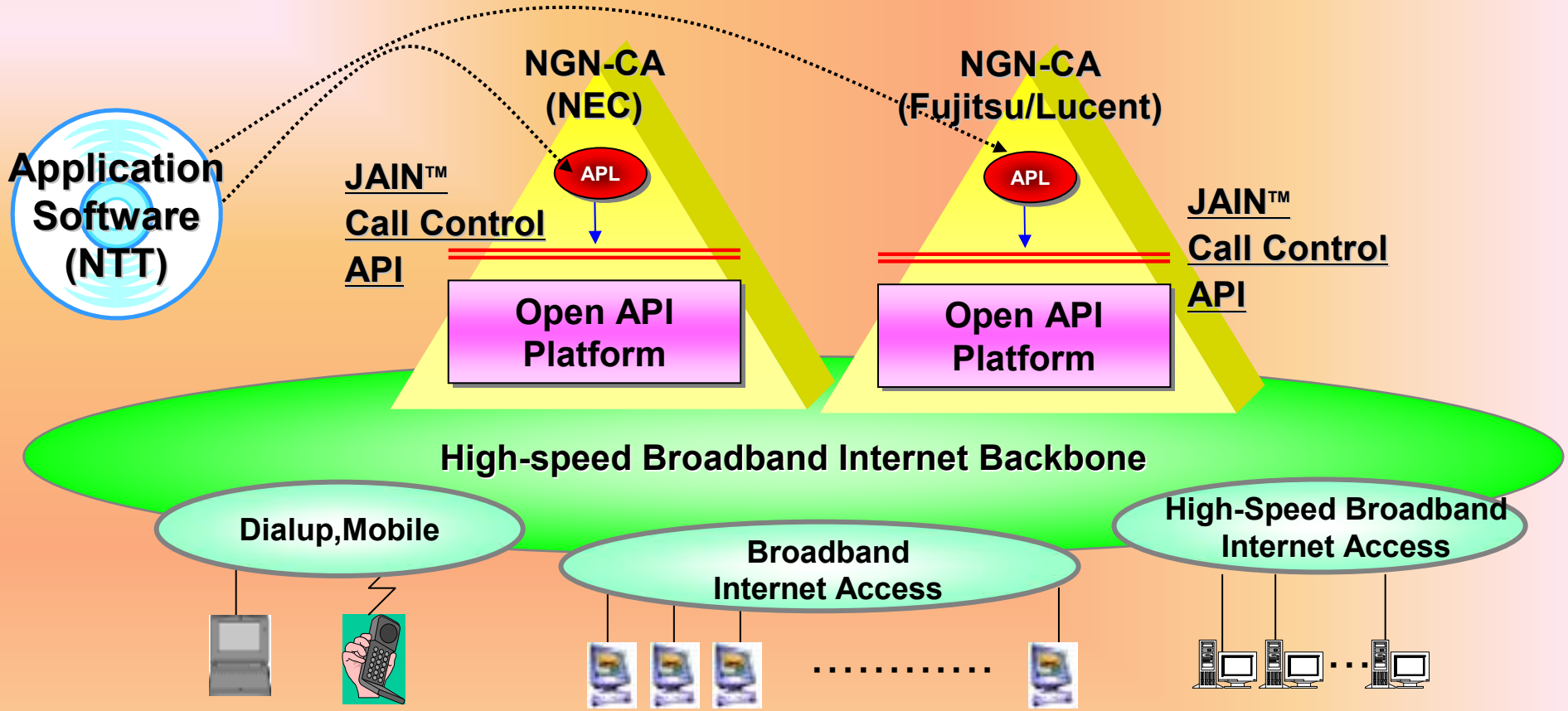
JCAT Architecture

- Requirements driven by a list of Service Drivers
  - Address those Services not supported by JCC



© 2003 Sun Microsystems, Inc. All Rights Reserved.

Application Portability across NGN-CAs using JAIN™



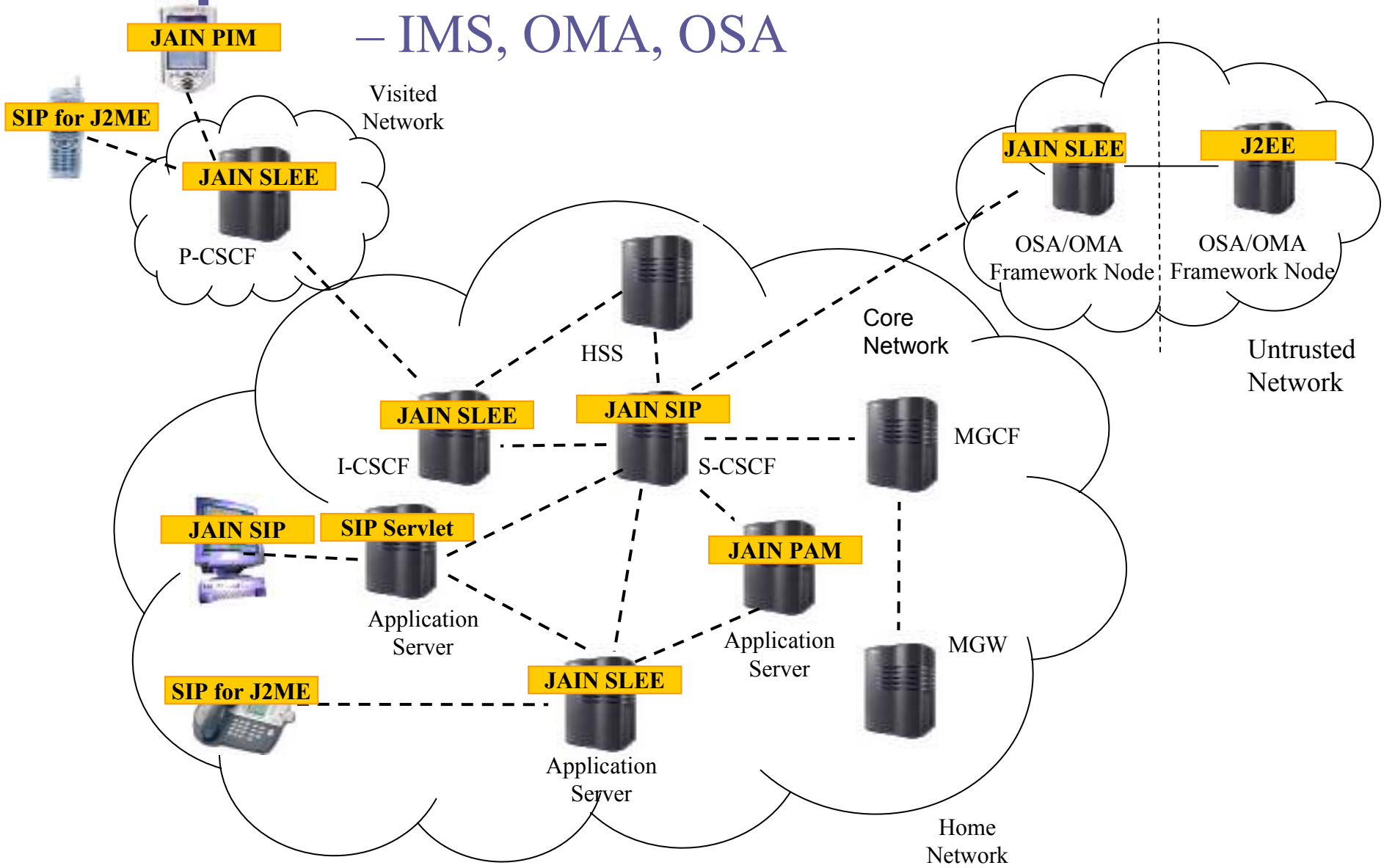
NGN-CA realizes
the efficient development and deployment of new services and stimulate the network service market.

 © NTT 2001

Summary

© 2003 Sun Microsystems, Inc. All Rights Reserved.

Sample Network Architecture
– IMS, OMA, OSA



© 2003 Sun Microsystems, Inc. All Rights Reserved.

Sample vendor solutions

- SIP network
  - NIST: JAIN SIP stack, JsPhone, IM UA & Proxy
  - Open Cloud: JAIN SLEE 3GPP IMS network elements
  - Siemens: SIP Servlet Application Server
- OSA network
  - jNETx: JAIN SLEE OSA Platform
- Presence network
  - Teltier: JAIN PAM Presence Server
- See further JAIN vendor solutions at:
  http://java.sun.com/products/jain/certified_products.html

© 2003 Sun Microsystems, Inc. All Rights Reserved.

Feedback from the Operators

"Unless we talk about what we want, we're not going to get it."

- "Telco-specific development will be facilitated by using off-the-shelf middleware platforms, along with component-centric software platforms and architectures, such as J2EE and JAIN SLEE. With this we can see a reduction in infrastructure costs and opportunities for new revenue-generating services."

  Elmar Weber, Vodafone

- "Open network interface is the thrust of our whole Next Generation Services environment…. While we may choose to go with a vendor's proprietary API that meets an internal business need, … JAIN will be the minimal entry requirement."

  Chris Shaw, Orange

© 2003 Sun Microsystems, Inc. All Rights Reserved.

Key JAIN JSRs…

- JAIN SIP (JSR 32) → Maintenance Review
- SIP Servlet (JSR 116) → Final Release
- JAIN SLEE (JSR 22) → Proposed Final Draft 2
- JAIN PAM (JSR 123) → Completed Public Review
- SIP for J2ME (JSR 180) → Public Review
- JAIN Call Control (JSR 21) → 2nd Maintenance Review
- JAIN JCAT (JSR 122) → Community Review
- JAIN Presence (JSR 186) → Community Review soon
- JAIN IM (JSR 187) → Community Review soon
- SAMS (JSR 212) → New JSR approved
- For a complete listing of JAIN JSRs, visit: http://java.sun.com/products/jain/api_specs.html

© 2003 Sun Microsystems, Inc. All Rights Reserved.

How to Get Involved ...

- JAIN™ technology is being specified as a community extension to the Java™ 2 Platforms
- Development is being carried out under the terms of the Java Community Process (JCP)
  - Visit http://jcp.org/en/participation/overview
- JCP Participants must sign either:
  - the Java Specification Participation Agreement (JSPA)
  - Individual Expert Participation Agreement (IEPA)

© 2003 Sun Microsystems, Inc. All Rights Reserved.

For More Information...

The JAIN APIs:

http://java.sun.com/products/jain

Certification:

http://java.sun.com/products/jain

JAIN Products:

http://java.sun.com/products/jain

JAIN Discussion Lists:

http://archives.java.sun.com

Java Community Process:

http://jcp.org

Contact:

JAINteam@sun.com

© 2003 Sun Microsystems, Inc. All Rights Reserved.

JAIN™ Technology

Serving the Developer Community

jainteam@sun.com
http://java.sun.com/products/jain