IMS Client Platform and IMS End-to-End

Piotr Kessler

Ericsson AB
http://www.ericsson.com

Peter Yeung

Ericsson AB
http://www.ericsson.com

TS-3234

End-to-End IMS Services
Toolbox for Easy Development and Deployment

Why IMS
Why IMS Client Platform
IMS Services and JSR 281

Agenda

IMS—A Service Enabler

IMS Applications in Devices

JSR 281—Future for Client Development

Developing Services

Conclusions

java.sun.com/javaone/sf

Agenda

IMS—A Service Enabler

- About IMS
- An Architecture and Framework
- Rich Services
- E2e Concept

IMS Applications in Devices

JSR 281—Future for Client Development

Developing Services

Conclusions

java.sun.com/javaone/sf

It's Not Just About Technology
It's About People Sharing Experience in Their Devices

Messaging · Presence · Content Sharing · PoC · Gaming · Chat · Whatever Attractive

IMS
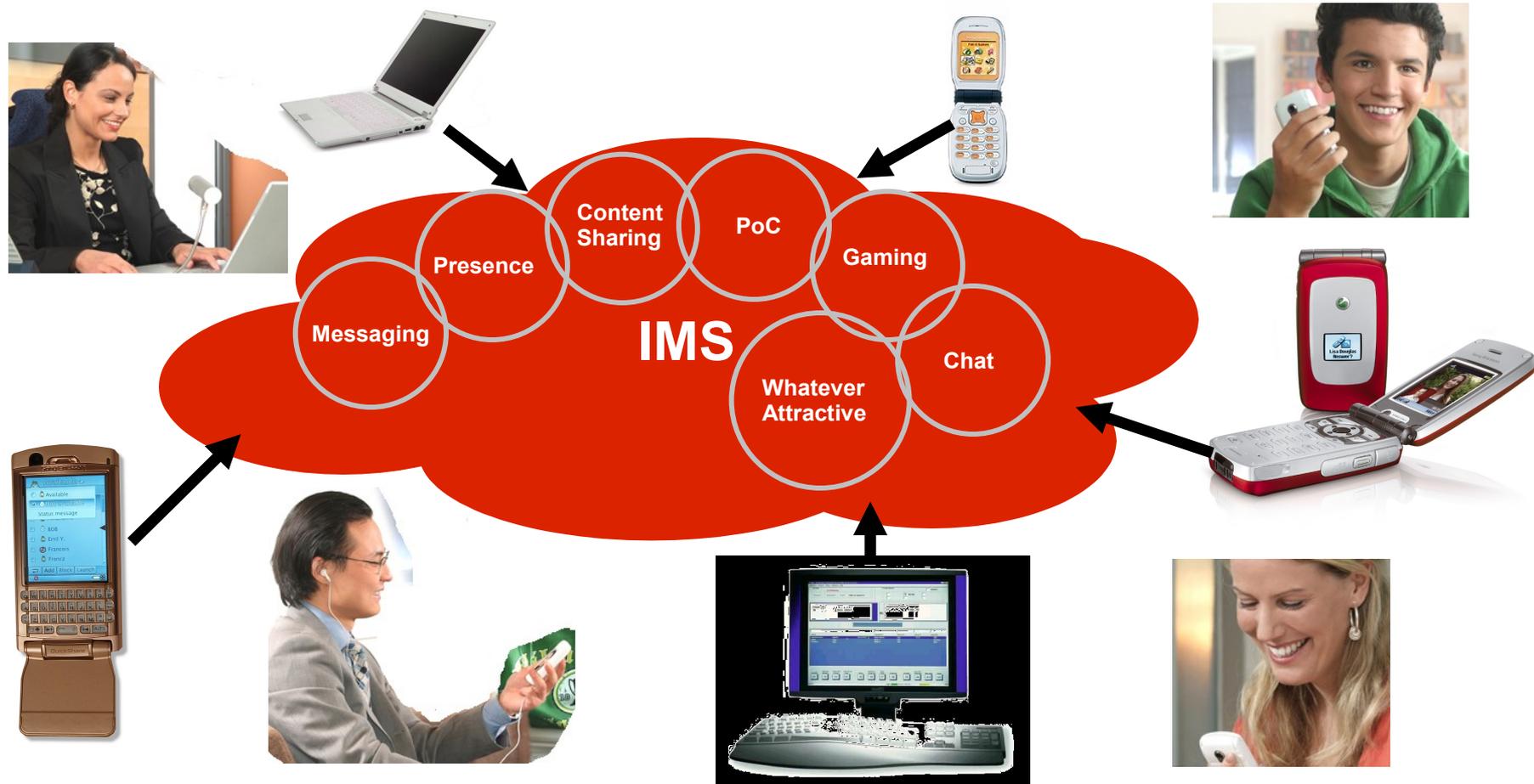
IMS...
Is About Merging Technologies to Deliver E2e Services

Messaging

Presence

Content Sharing

PoC

Gaming

Chat

Whatever Attractive
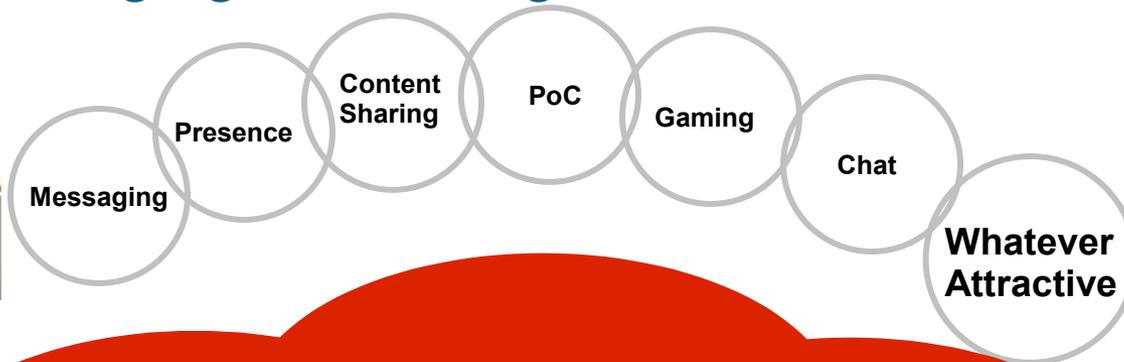
IMS

- **Openness**
- **Innovation**
- **Short TTM**
- **Multimedia**

- **Standardization (3GPP and IETF, ITU-T, ANSI,ATIS, TISPAN, OMA, GSMA)**
- **Interoperability**
- **Multiaccess**
- **Secure reachability**
- **Payment security**
- **Quality of Service**

java.sun.com/javaone/sf

IMS for Multiple Applications

ServiceNetwork
- P2P Multi-media (tispan)
- Presence (OMA)
- CSI (3GPP)
- PoC (OMA)
- Messaging (OMA)

IMS
- SIP
- MRF
- HSS
- SIP

IP
- tispan
- 3GPP

Multiaccess

Residential   **Enterprise**   **Mobile**
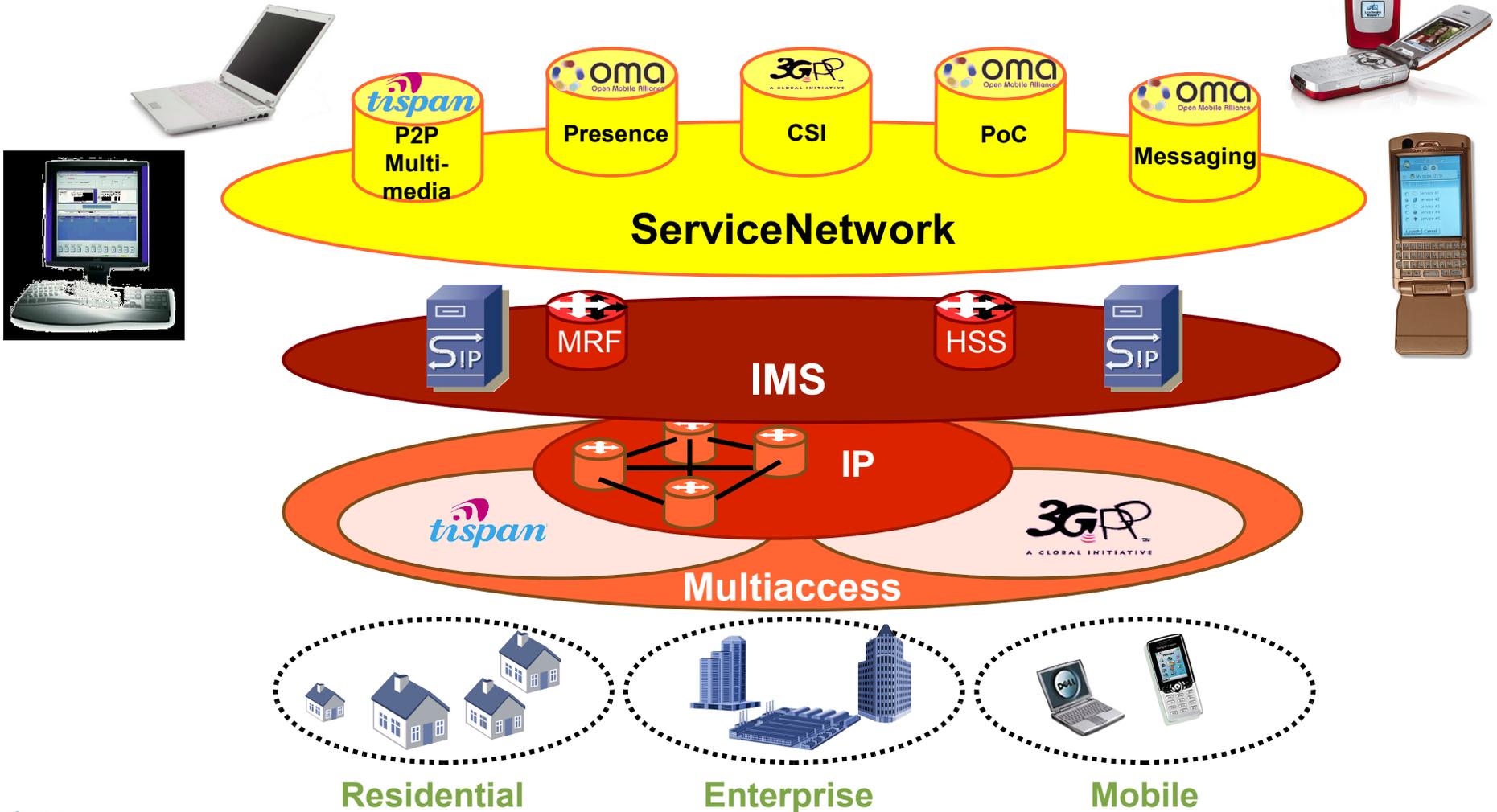
IMS Is the Framework

E2E

Reachability

Mobility

Interoperability

Convergence

Quality of Service

MultiMedia connections

Security

Charging

java.sun.com/javaone/sf

Agenda

IMS—A Service Enabler

IMS Applications in Devices

JSR 281—Future for Client Development

- Developer's Perspectives
- Developer-friendly IMS Terminal
- IMS Client Platform
- DEMO

Developing Services

Conclusions

java.sun.com/javaone/sf

Developer's Perspectives—A Challenge

Technology

- A number of communication protocols to use SIP/SDP/RTP/RTCP/MSRP/XML/XCAP
- A number of standards to follow
- A number of Real Time requirements to full fill
- A number of low-level tasks to manage
- A number of service enablers to implement

End-user rich application

- A number of services to aggregate using service enablers
- Attractive and competitive GUI for the user
- Simple and engaging user interaction

But...

...The Solution Is There—Two Domains

IMS Framework

- Domain of the device platform provider

IMS Applications

- Domain of the application developers

A Developer—Friendly IMS Terminal

- Layered architecture
    - IMS Client Platform—Realization of IMS Framework
        - Focus on IMS technology
        - Focus on service logic
        - Focus on co-location of Applications
    - High-level API
        - Hides all technology details
        - Offers straight-forward application-level methods
    - Applications
        - Focus on usability and user interaction
        - Focus on attractive GUI
        - Focus on attractive service behaviour

API

IMS Applications

API

IMS Client Platform (ICP)
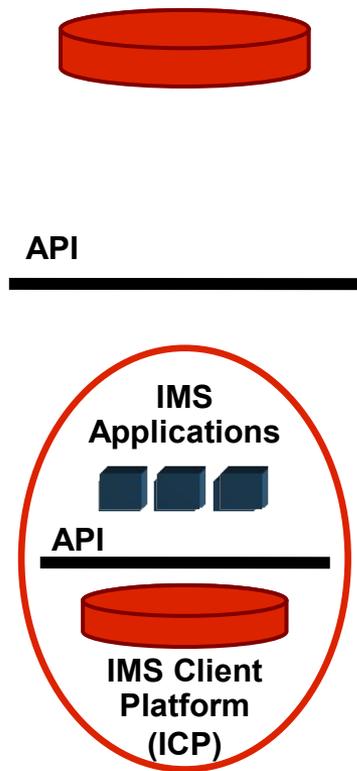
Why IMS Client Platform

- Enables easy development, deployment and co-existence of IMS Clients for different services
  - Creates a generic horizontal IMS platform for co-executing IMS Clients
  - Secures conformance to applicable IETF, 3GPP and OMA standards
  - Manages correlation between IMS enablers
  - Manages low-level tasks for the Clients
  - Exposes IMS enablers through high level Java™ technology API
- Offers Convergence
- Secures IOT

java.sun.com/javaone/sf
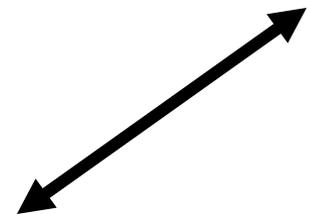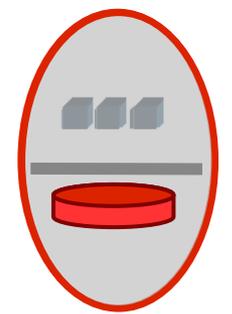
IMS Client Platform Architecture

java.sun.com/javaone/sf
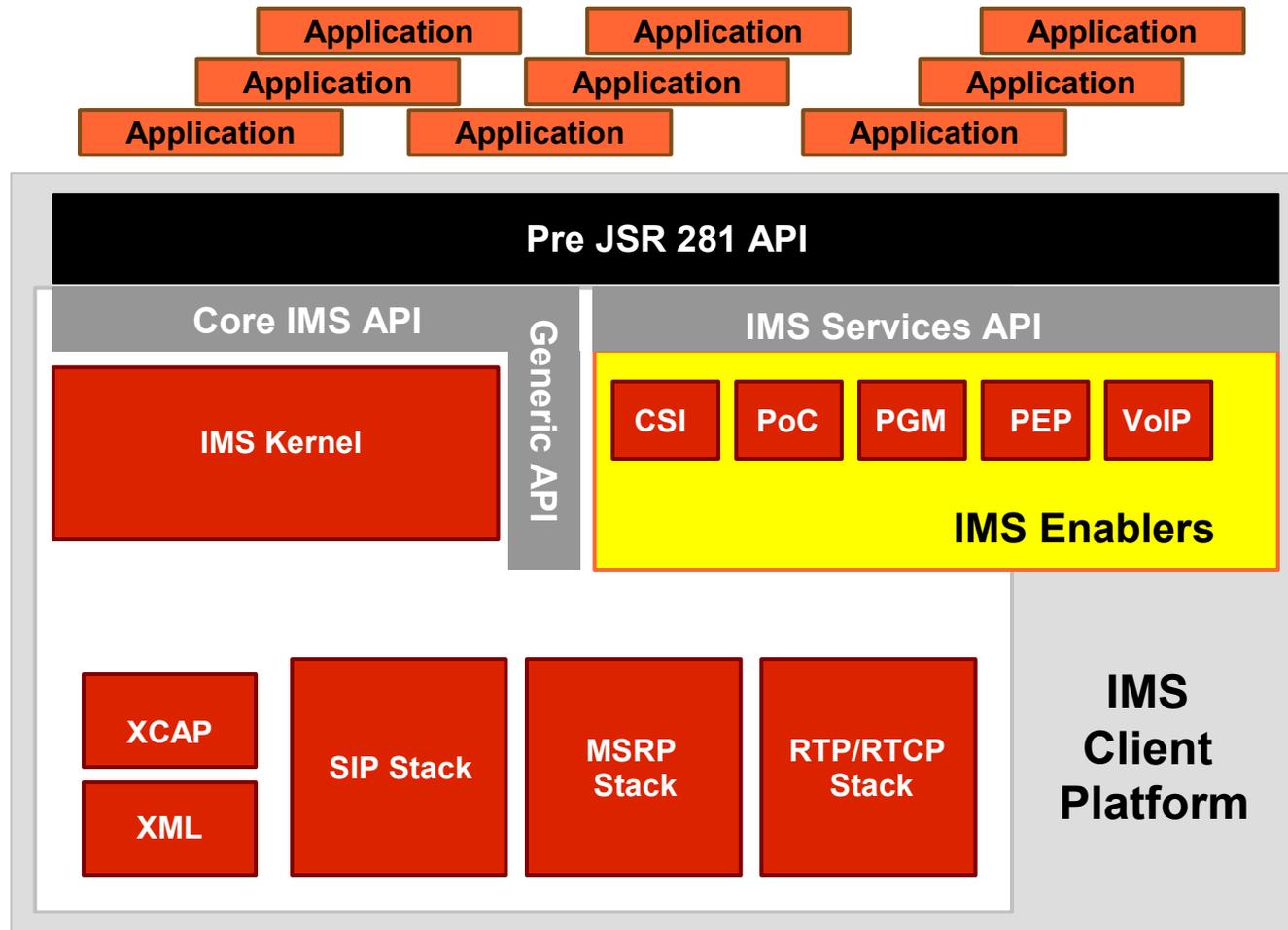
Benefits
Ease Of Application Deployment

- Java technology API

- Core IMS API—for new non-standardized services
  - Generic service sessions
  - Multimedia connections

- Service API—for standardized IMS services
  - Presence and Group List Management
  - PoC—Push to Talk over Cellular
  - VoIP—Voice over IP
  - CSI—Circuit Switch IMS Integrated Services

- Combine both to create fancy Java technology-based applications

Benefits

Ease Of Application Deployment

- IMS technology encapsulation in the platform
  - Multi-vendor interoperability
  - Fixed and mobile convergence

- Broad terminal support
  - Feature phones
  - Smart phones
  - Desk phones
  - PC
  - Home Gateways

- Java technology—fast application delivery

DEMO

A Fancy IMS e2e Service

java.sun.com/javaone/sf

Agenda

IMS—A Service Enabler

IMS Applications in Devices

JSR 281—Future for Client Development

Developing Services

Conclusions

java.sun.com/javaone/sf

Standardized Way Forward
JSR 281

- API for Client application development for Java ME devices: CDC/CLDC

- Abstracts IMS technology through API
  - Generic IMS API
  - IMS Services API

- Brings standardized IMS Client Service Creation toolbox for Java technology development community

- Lead by:
  - Ericsson (Piotr Kessler and Stefan Svenberg)
  - BenQ (Volker Bauche and Mirko Nauman)

JSR 281 Architectural Concept

Application Layer

- Non standard — e.g. Game
- .....
- Non standard — Gaming, Messaging, PoC
- Standard — PoC

API Layer

Core API
- Event Framework
- PacketMedia
- StreamMedia
- Session
- Basic Messaging
- Player
- Recorder
- Network

Service API
- PoC
- IMS Presence
- GLM
- XDM

} JSR 281

Implementation Layer

IMS Core
- IMS Session
- PacketMedia
- StreamMedia
- Basic Messaging
- Player
- Recorder
- Event framework
- Registration/Auth
- Network

IMS Service Enablers
- PoC
- IMS Presence
- GLM
- XDM

Device SW Platform Layer

Stacks: SIP/SDP/RTP/RTCP/MSRP/XML/HTTP/XCAP

Please visit a TS#3319 to see JSR-281 in detail

Agenda

IMS—A Service Enabler

IMS Applications in Devices

JSR 281—Future for Client Development

Developing Services

- Service Development Studio
- E2E Service Creation
- Demo

Conclusions

java.sun.com/javaone/sf

IMS Service Creation Solution

- Eclipse based Service Development Studio
  - Use of existing tools and skills

- Target Developer
  - Internet-savvy developer with Java ME, Java SE and Java EE platform experience

- Server side development
  - SIP Servlet engine with JSR 116 API

- Client side development
  - IMS Client Platform with pre-JSR 281 API for Java ME and Java SE

java.sun.com/javaone/sf

E2E Service Creation

IMS Services

Client
Service Components

Network
Service Components

Service Layer
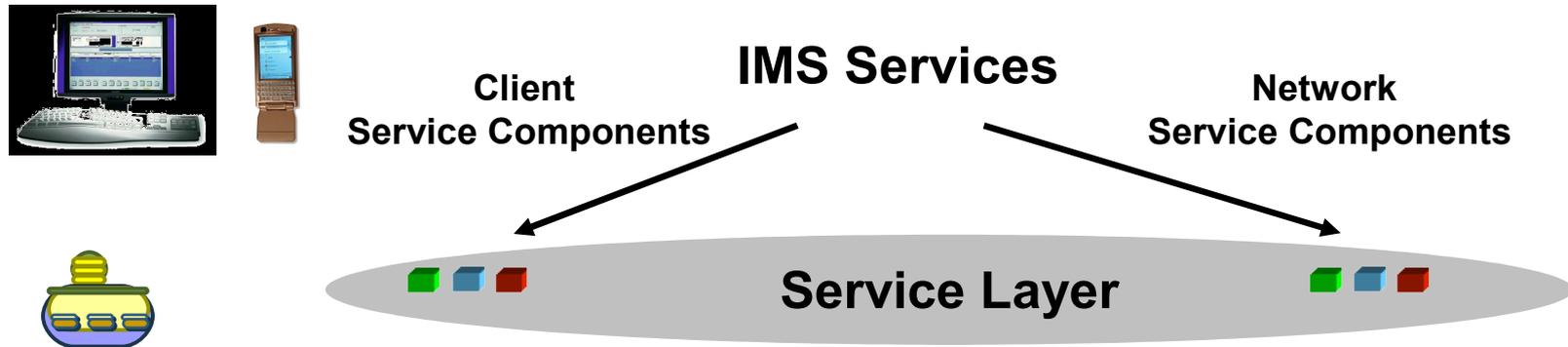
- Initial concerns
  - **TTM**: Efficient development and deployment of e2e services
    - High abstraction level of API
    - Verification on PC without access to real network
  - **IOT**: Interoperable e2e services
   - Standardized interactions secured by the client platform
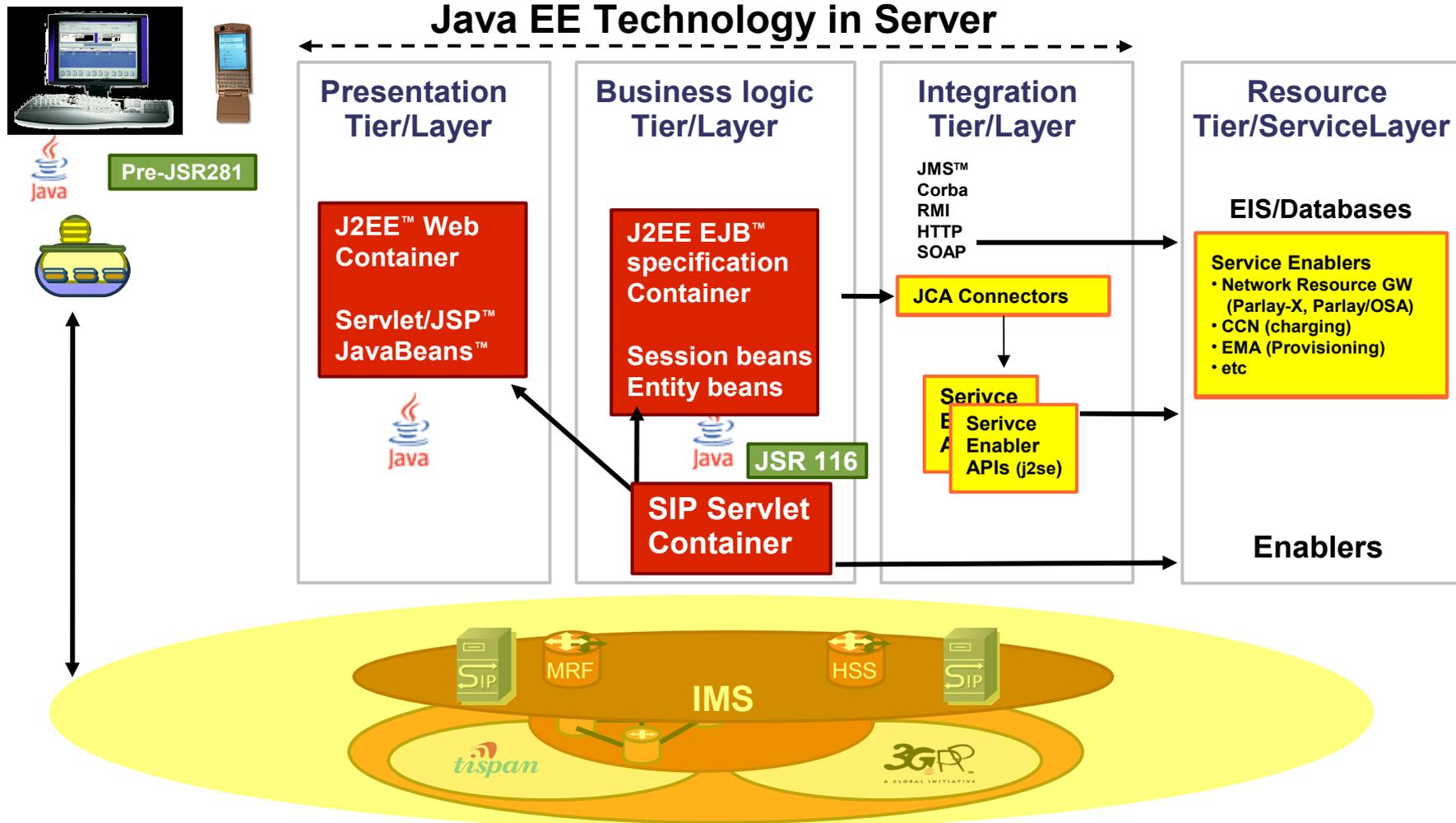
Abstracted View on IMS Network

IMS e2e Services

Pre-JSR281 — IMS Client Platform (ICP)

JSR 116 — IMS Server Platform

IMS

MRF HSS

IP

tispan 3GPP A GLOBAL INITIATIVE

Multi-Access
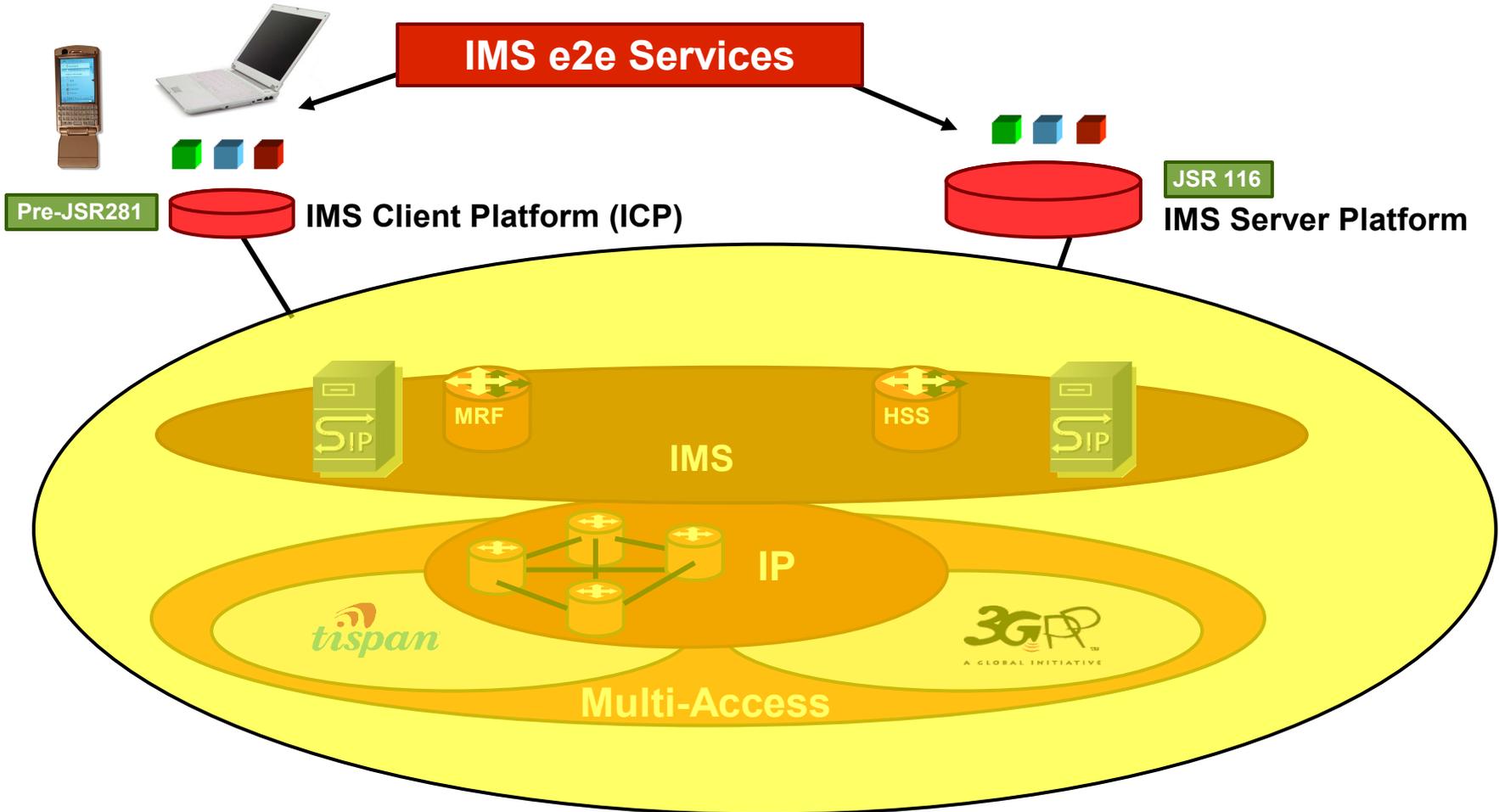
DEMO

Step-by-Step Development of a Simple
Service: Sharing Pictures

java.sun.com/javaone/sf

Summary

- IMS provides framework for pervasive IP multimedia services

- IMS Client Platform enables simple client development

- And secures convergence and interoperability

- JSR-281 brings everything together through standardized Java technology API

For More Information

Piotr.Kessler@ericsson.com

- www.ericsson.com/mobilityworld

- TS-3319 PoC and Beyond—The IMS Services API (JSR-281)

- JSR-281 www.jcp.org/en/jsr/detail?id=281

- Exhibition case 502 and Mobility Village case

Q&A

IMS Client Platform and IMS End-to-End

Piotr Kessler

Ericsson AB
http://www.ericsson.com

Peter Yeung

Ericsson AB
http://www.ericsson.com

TS-3234