ITU Kaleidoscope 2011
The fully networked human?
Innovations for future networks and services

SOA Driven Architectures for Service Creation Through Enablers in an IMS Testbed

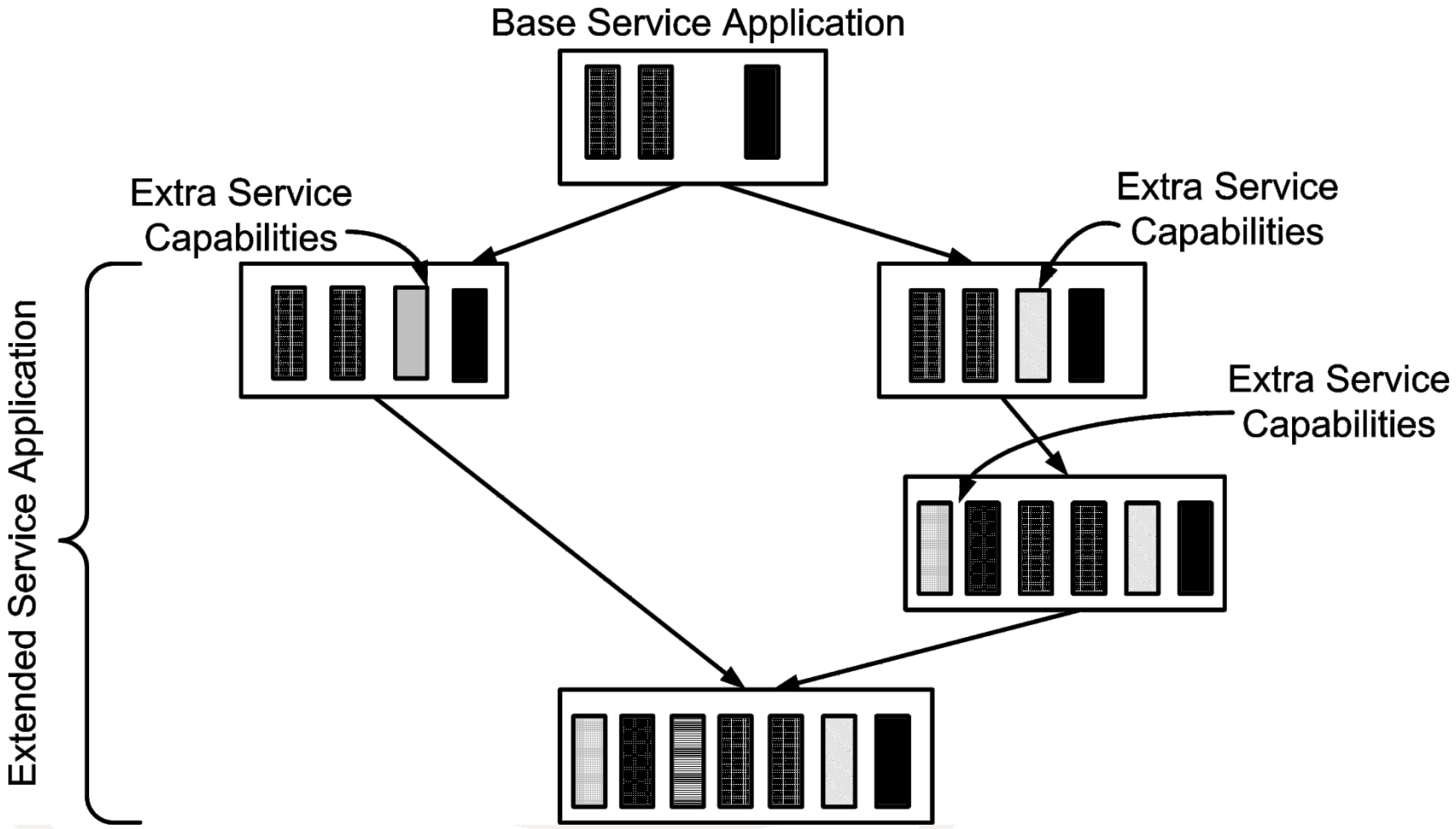
Mosiuoa Tsietsi
Rhodes University, EC, South Africa
m.tsietsi@computer.org

Cape Town, South Africa
12–14 December 2011

Outline

- Services and Service Capabilities
- SOA Development in Telecoms
- Extended IMS Service Layer (EISL)
- Implementing EISL
- Service Development in EISL
- A Case Study in EISL Development
- Conclusion

1. Services and Service Capabilities



Source: ETSI TR 181.004, 2006.

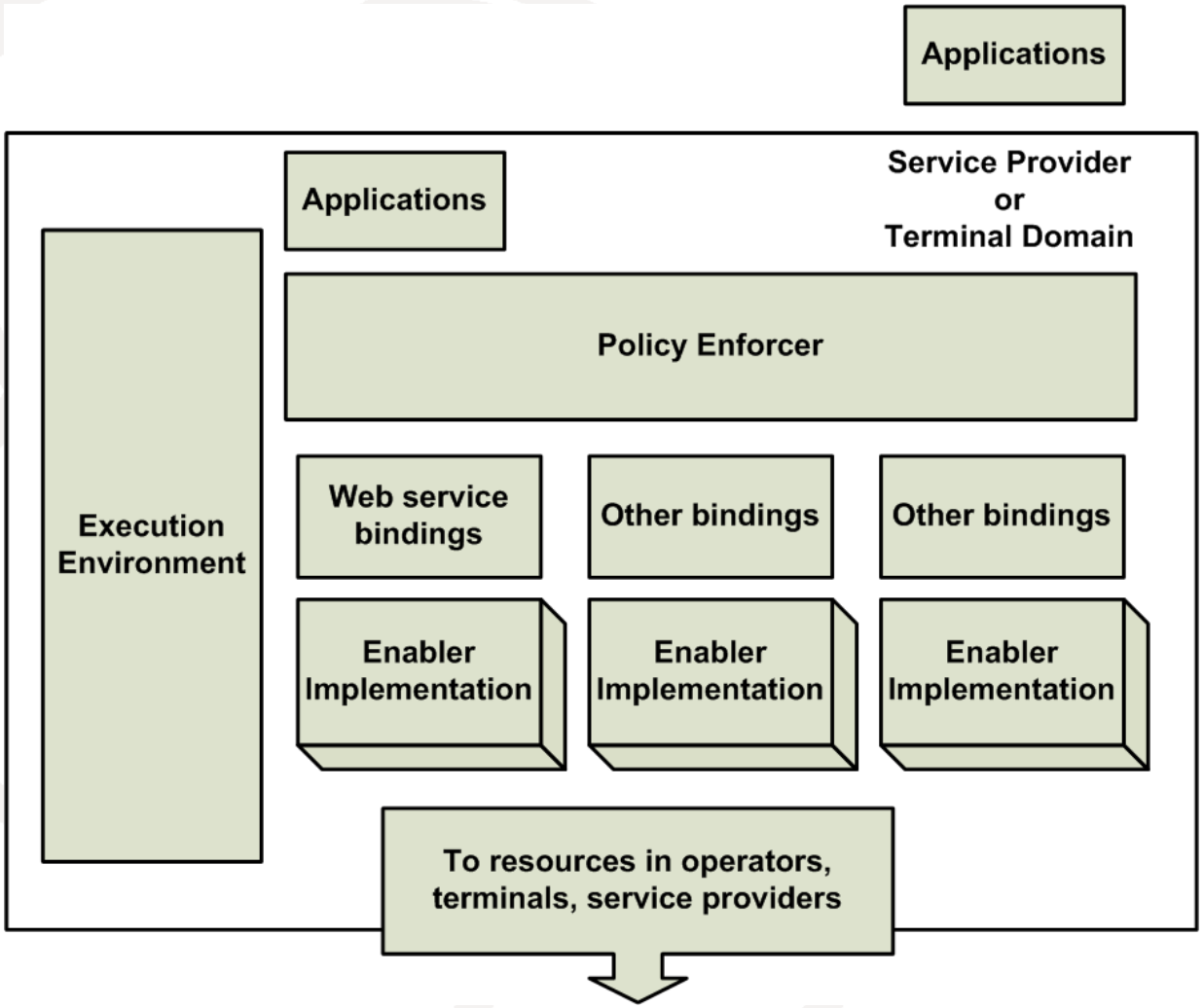
1. SOA Development in Telecoms

- Open Service Access
  - Provide access to telecommunication network functions via standard interface
  - Developed by Parlay group, now joint standardisation by 3GPP and ETSI)
  - Reliant on Parlay gateway (Service Capability Server, SCS)
  - Most development in web services

2. SOA Development in Telecoms

- OMA Service Environment (OMA SE)
  - Open Mobile Alliance is the largest developer of specifications for mobile services
  - Developed over 100 specifications for enablers such as presence, messaging..
  - Single architecture for multiple enablers
  - Utilises policy enforcer and bindings to enabler implementations

3. SOA Development in Telecoms

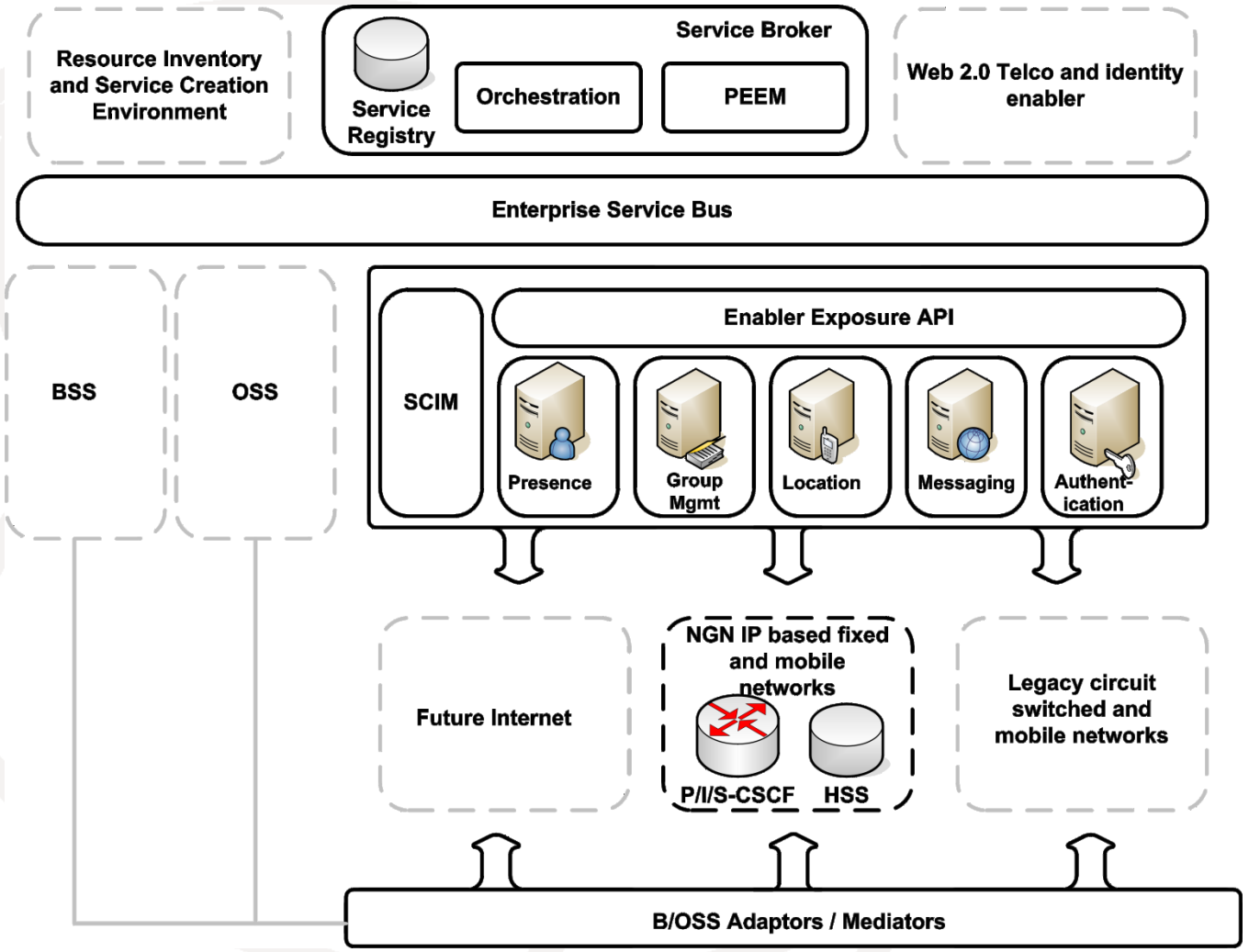


Source: OMA Service Environment v1.0.5, OMA. 2009.

4. SOA Development in Telecoms

- SOA Telco Playground
  - Open source Parlay testbed for industry and academia
  - APIs for service creation in IMS
  - Brings together OMA enablers, policies, enabler exposure and open APIs

5. SOA Development in Telecoms



Source: N Blum, T Magedanz, F Schreiner, S Wahle, 2010.

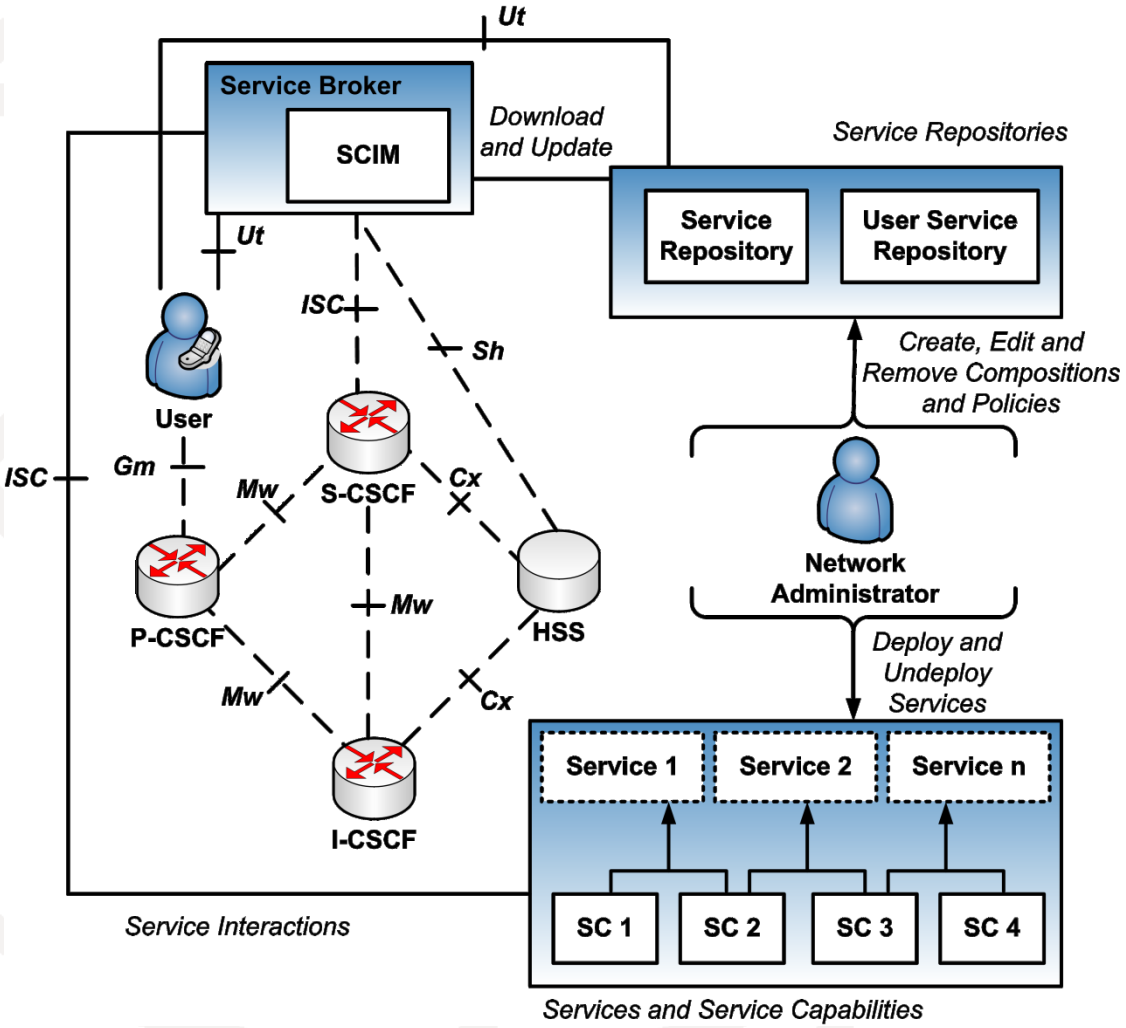
1. Discussion

- IMS/telecom developers need to "enablerise" their networks/testbeds
- Web services have long history, but there are alternatives
  - JAIN - Java APIs for Intelligent Networks
  - JAIN SLEE –Service Logic and Execution Environment
  - Mobicents – prominent SLEE implementation

1. Extended IMS Service Layer (EISL)

- Extended view of standard IMS service layer
- Consists of:
  - Network personnel
  - Service broker
    - Service Capability Interaction Manager
  - Data Repositories
    - Service Repository
    - User Service Repository
  - Third party application servers

2. Extended IMS Service Layer (EISL)



Source: M. Tsietsi, A Structural and Functional Specification for a SCIM for IMS, PhD Thesis. August 2011.

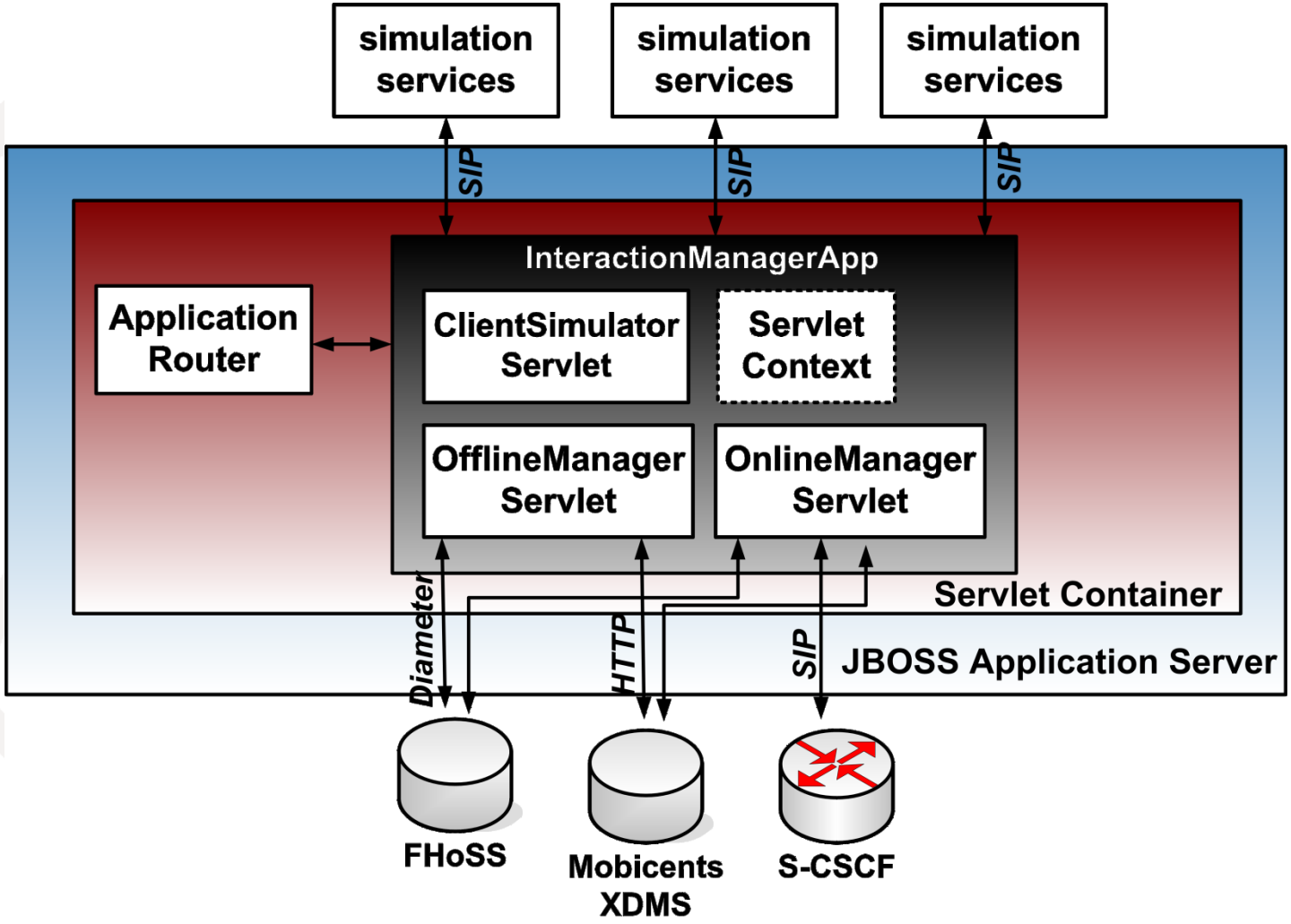
1. Implementing EISL

- Emphasis on open standards and open source implementation
- Components:
  - Open IMS Core and JSLEE Mobicents
    - Open IMS Core (SVN checkout)
    - Mobicents SIP Presence Service
      - SIP Presence and XDMS (IETF/OMA presence)
      - ETSI/3GPP simservs appusage (TS 24.173)
      - IETF/OMA common policy (RFC 4745, OMA-TS-XDM_Core)
    - Mobicents SIP Servlet Container (JSR 289)

2. Implementing EISL

```
<?xml version="1.0" encoding="UTF-8"?>
<simservs xmlns="http://uri.etsi.org/ngn/params/xml/simservs/xcap"
xmlns:cp="urn:ietf:params:xml:ns:common-policy"
xmlns:ocp="urn:oma:xml:xdm:common-policy">
  <originating-identity-presentation-restriction   active="true" priority="1">
    <default-behaviour>presentation-not-restricted</default-behaviour>
  </originating-identity-presentation-restriction>
  <outgoing-communication-barring active="true" priority="2">
    <cp:ruleset>
      <cp:rule id="rule66">
        <cp:conditions>
          <cp:identity>
            <cp:one id="sip:mallory@open-ims.test"/>
          </cp:identity>
        </cp:conditions>
        <cp:actions>
          <cp:allow>false</cp:allow>
        </cp:actions>
      </cp:rule>
    </cp:ruleset>
  </outgoing-communication-barring>
</simservs>
```

3. Implementing EISL

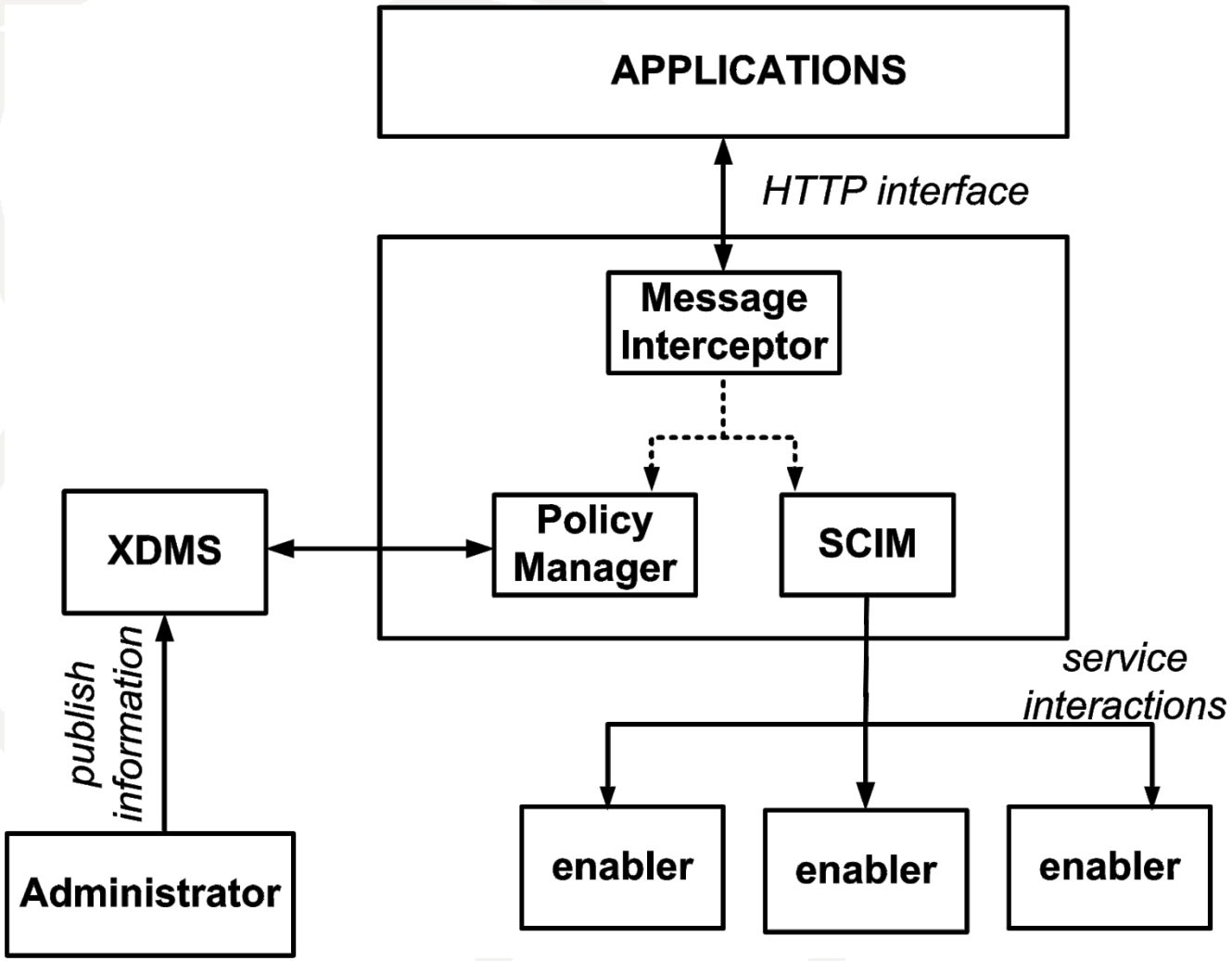


Source: M. Tsietsi, A Structural and Functional Specification for a SCIM for IMS, PhD Thesis. August 2011.

1. Service Development in EISL

- Interfacing with the SCIM:
  - SIP ✗
  - HTTP ✓
- Choose an API that is well understood and widely used
- Converged Servlet container can do both SIP/HTTP

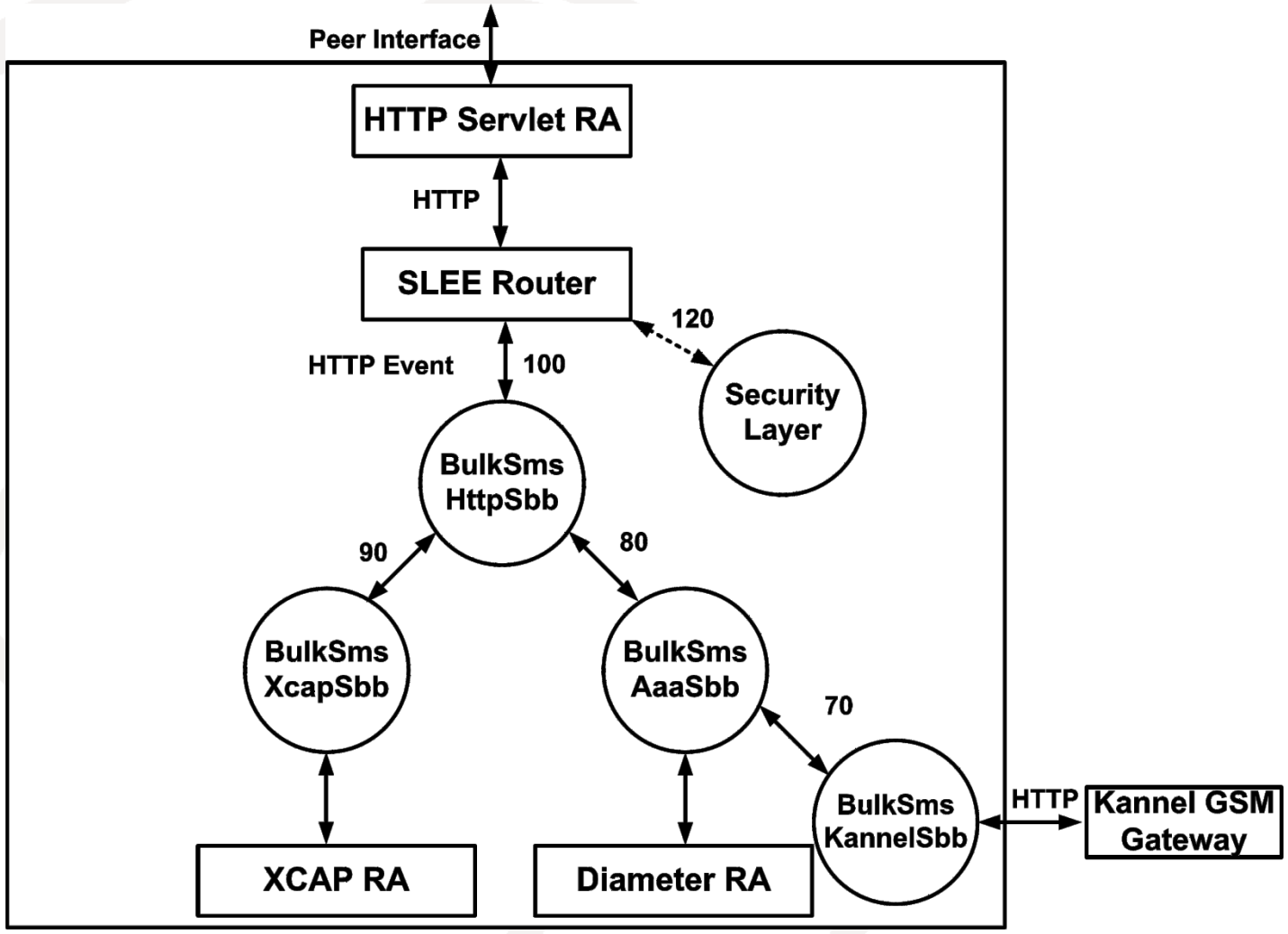
2. Service Development in EISL



1. Case Study in EISL Development

- Exposes an API to service developers
- Comprises of:
  - SMS Enabler
  - Resource List Enabler
  - Policy filter

2. Case Study in EISL Development



3. Case Study in EISL Development

- Create contact list

http://ip:port/server/username=usr&password=pass&type=document&op=add

- Add contact to a group list

http://ip:port/server/username=usr&password=pass&type=contact&op=add&group=Close%20Friends

- Send sms to contact

http://ip:port/server/username=usr&password=pass&type=sms&to=+27786346926&group=Close%20Friends&report=true

Conclusion

- EISL marries new paradigm in service creation with exposing capabilities in telco networks
- Service broker (SCIM) is central to this and must be multi-protocol
- Rapid service development is aided by using well known protocols (HTTP)