Multi-service, Multi-protocol Management for Residential Gateways

Yvan Royon
Pierre Parrend
Stéphane Frénot
Humberto Abdelnur

Serafeim Papastefanos
Dirk Van de Poel

BB Europe Antwerp
December 6th, 2007

Multi Service Access Everywhere
> Multiple providers for VVD
> The home user chooses on-demand
> Drawbacks of multi-play
  • More and more dedicated set-top boxes
    – Complexity
    – Manufacturing and maintenance costs
  • Problems with non-standardisation
    – Many closed / proprietary solutions
  • Multimedia VV services only
Other types of services exist! MUSE I

> Management/supervision services at home
  • Domotics
    – Heating, lights
  • Health care devices supervision
    – Pacemaker, elder people help, quality of life improvement
  • Whitegoods supervision
    – fridge, dishwasher...

> Applicative services
  • Games
  • Graphical user interfaces for management

> ==> Multi-service/multi-provider/multi-device
The service gateway model

> Hosts software services (any service)
Evolution of Home Gateways

> Need for a single platform
  • Open and extensible (dynamic market)

> Execution Environment
  • Dynamic and collaborative aspects of services

> Management infrastructure
  • Local and remote management
  • Deployment

> Move user-related intelligence to the border of the network
Management Realms

User Management

Service Management

Manage Services

Manage the Gateway

Set Preferences

Access Management
Management Realms

User Management

DHCP
UPnP
OSGi
User Friendly

Service Management

JMX
Isolation
Security

Access Management

TR-69
ACS
QoS
Management Realms

User Management
- DHCP
- UPnP
- OSGi
- User Friendly

Access Management

Service Management
- JMX
- Isolation
- Security

TR-069
- ACS
- QoS
How to collapse the management realms?

> **Management issue:**

> **protocol convergence**

  - Multi-provider integration
    - Initial provisioning of ACS,
    - Ping diagnostic in multi-provider environment

  - TR-69 vs Netconf
    - IETF netconf management protocol is used to manage TR-69 CPE
    - A Netconf/TR-69 proxy is designed

  - TR-69 vs UPnP
    - How to handle UPnP devices
    - A TR-69/UPnP proxy

> **Run-time issue:**

> **service oriented architecture**

  - Remote access to services
    - Global management MOSGi
    - JMX (YAMF)

  - Service provider isolation
    - Namespace isolation VOSGi
    - Resource control

  - Service security
    - Bundle signature
    - Method execution control
Most home devices have a UPnP interface but not a CWMP interface.
The ACS needs to identify and control some of these UPnP devices.
By using a UPnP/TR69 proxy, the ACS can do it without the need to add a TR-69 interface to the UPnP devices.
Each UPnP device will be represented as a TR-106 compatible Service by the proxy.
The MUSE network gains access to thousands of UPnP home devices.

IEEE Communications Magazine, October 2007 issue
MOSGi

> An end-to-end management framework for OSGi Gateways
> Based on JMX management architecture
  • Java Standard
  • Easy to implement
  • Open-source initial distributions

> End-to-End
  • Probes: low level data from the system, linux, OSGi, installation
  • Agent and services: notifications, rmi and xml connector
  • Dedicated Management Console: On-Demand Management Panels

> Embeddable
  • SOP approach to the implementation
  • Lightweight embedded JMX agent
Console before connexion

OSGi GUI Remote Manager

Servers
- 127.0.0.1:1099/core
- 134.214.144.106:1099/t1

Remote Logger

<table>
<thead>
<tr>
<th>Src</th>
<th>Level</th>
<th>Id</th>
<th>Text</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>
Once connected new user interface
General overview of MOSGi

Integrated in Felix Apache project, 2005
Per-provider isolation of services
- Explicit service sharing on demand
- Isolation of management agents
Virtual OSGi

- Runs OSGi within OSGi
- Patch to Apache Felix and Concierge
- Namespace isolation
- Service sharing

- Published in CBSE'06, Stockholm
Securing OSGi

- INSTALLATION
- RESOLVING
- UNINSTALLATION
- STARTING
- ACTIVE
- STOPPING
Securing OSGi

1. Download
2. Validate
   - Valid: Install
   - Invalid: Reject
3. Authorize
4. Install
5. Resolve
6. Refresh
7. Start
8. Uninstall
9. Stop
10. Pre-install validation by SOSGi
11. Standard OSGi bundle life cycle
An overall demonstrator

IEEE Communications Magazine, October 2007
Stéphane Frénot
INRIA
stephane.frenot@insa-lyon.fr

Questions ?