ProSyst®

Developing Smart Home Systems by using OSGi and Plug Computers

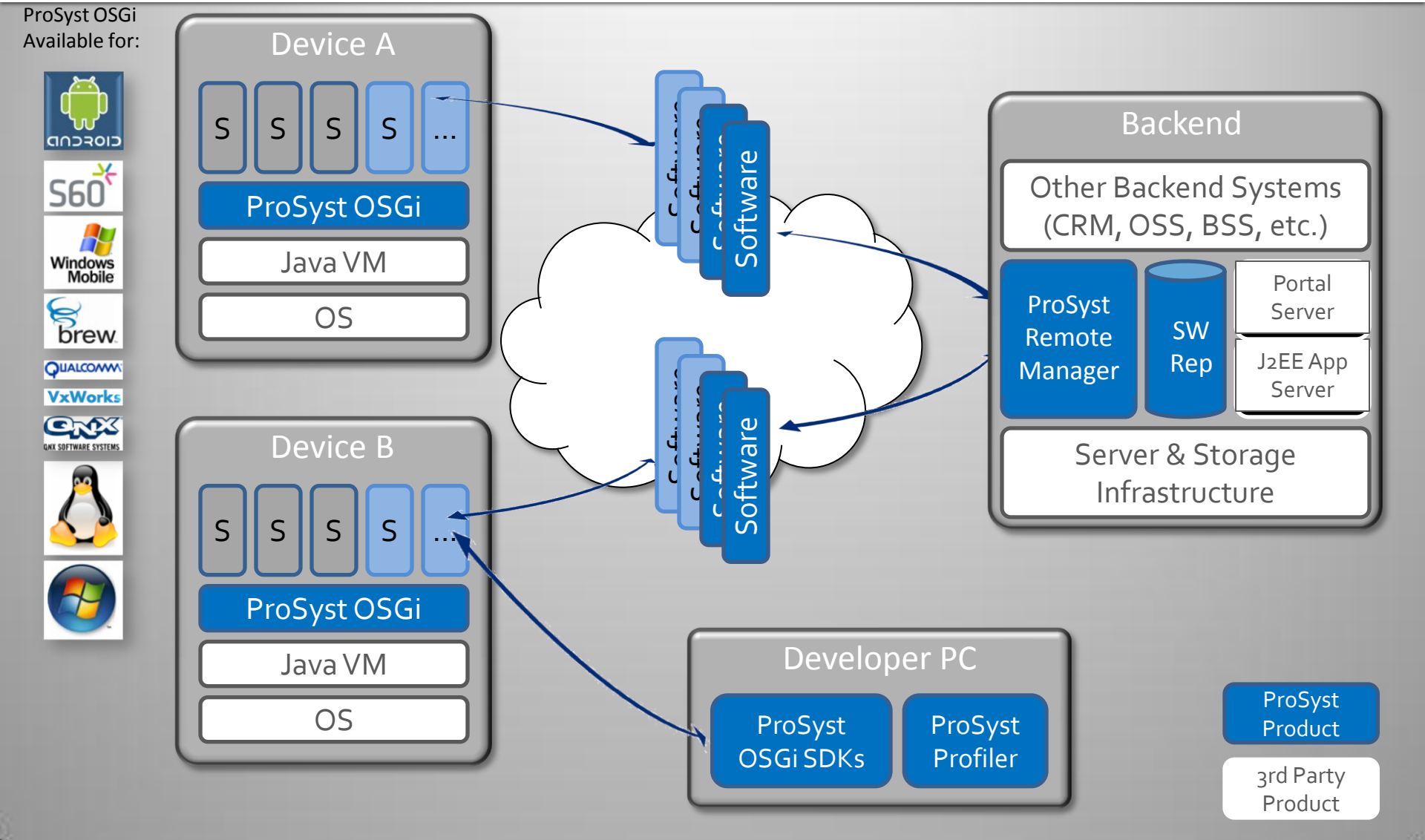
Dr. Dimitar Valtchev

24 June 2010, Stuttgart, Eclipse Embedded Day

Introduction ProSyst

- ProSyst is a leading company for end-to-end embedded software and OSGi solutions
- Founded 1997 in Cologne, Germany
- OSGi member since its foundation in 1999
- 120+ engineers employed

ProSyst E2E Products



Agenda

- Introduction OSGi
- Introduction Plug Computers
- OSGi Smart Home Architecture
- OSGi Smart Home SDK
- Live Demo with UPnP
- Conclusions

Introduction OSGi

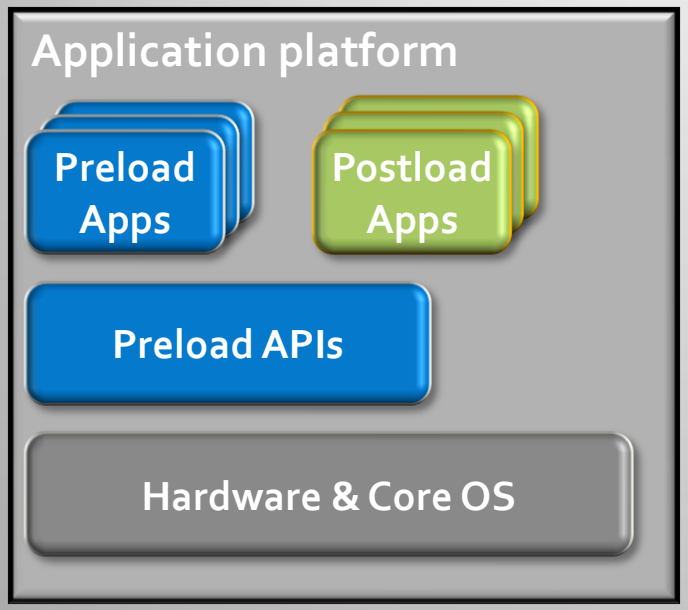
- The OSGi Alliance is an independent non-profit corporation founded in 1999
- OSGi is a module system and service platform for the Java
- It defines also the life-cycle management of applications and components
- OSGi specification is currently used in various vertical areas such as residential, mobile, automotive, enterprise, etc.
- More information at http://www.osgi.org

Technical Benefits of OSGi

- Portability of applications & services (Java)
- Dynamic discovery of services & APIs
- Modularization of the platform and apps
- Higher degree of code re-use
- Lots of off-the-shelf components available
- Platform, applications and services are remotely manageable: deployment, monitoring, diagnostics, lifecycle, configuration, etc.

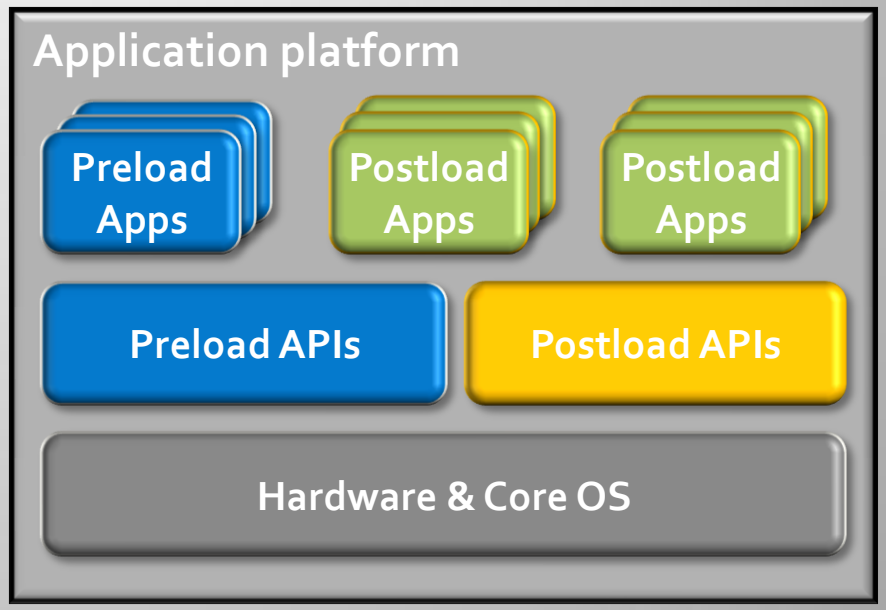
Concept: Platform Openness

- Traditional Concept



- Applications are limited to locked down set of APIs
- No platform innovation possible

- New Concept

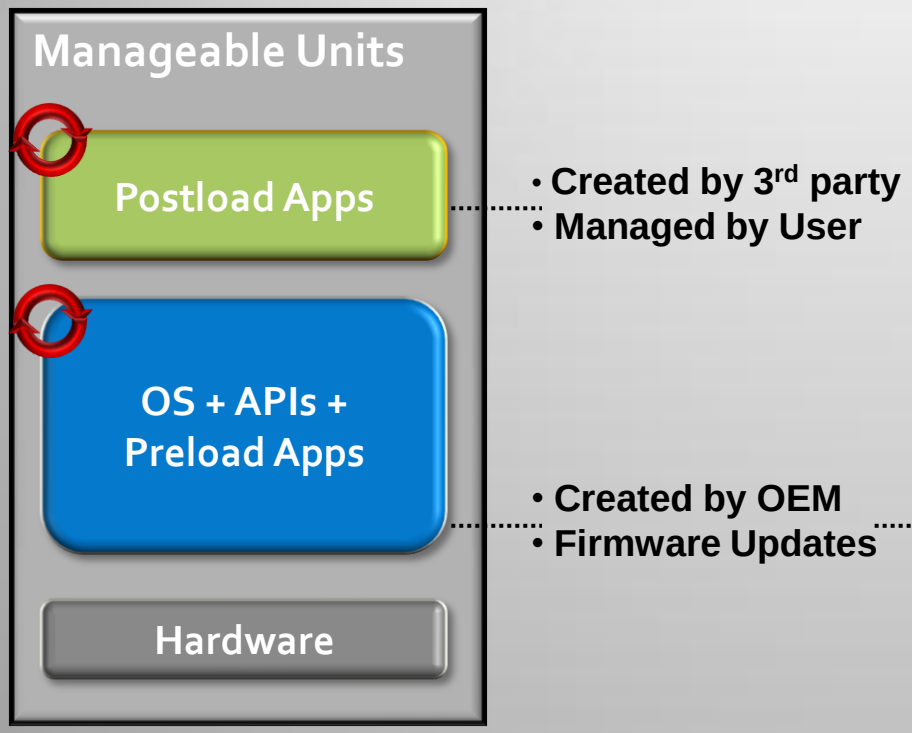


- "Soft Platform": New Features & APIs loadable at any time
- Open for Operators and 3rd parties
- Caters for rapid platform Innovation

Locked down by OEM | Open for 3rd Parties | New Layer, Open for 3rd Parties

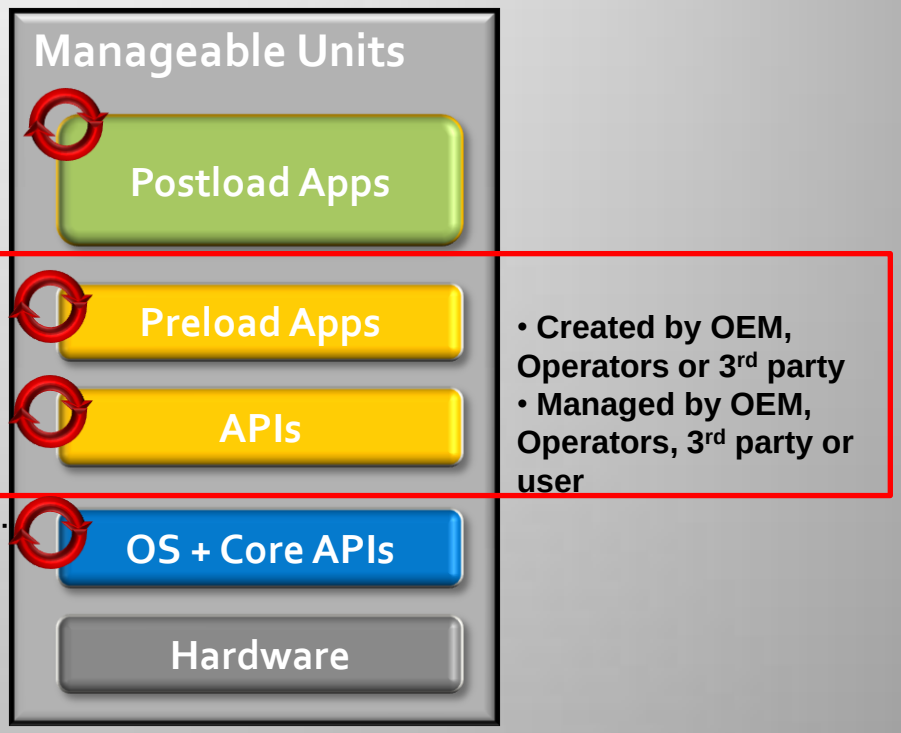
Concept: Lifecycle management

- Traditional Concept



- Coarse grained management model
- No platform innovation possible

- New Concept



- Fine grained management model
- Innovation possible on all levels
- Open for Operators and 3rd parties

Residential OSGi

- The residential area is currently one of the most promising application fields of OSGi because of:
  - There are several well synchronized specification efforts based on OSGi
  - Many useful services/protocols needed in residential boxes are well specified and implemented
  - SDKs and other convenient tools widely available

Plug Computers

- A plug computer is a small form factor network-attached server for use in the home. In effect, **a plug computer is a network appliance that is fully enclosed in an AC power plug or AC adapter**.
- Plug computers are equipped with hardware interfaces like USB 2.0, SD, Gigabit Network, JTAG mini USB, Bluetooth, Wi-Fi
- Software includes Linux, JVM and even OSGi

Plug Computers by Example SheevaPlug

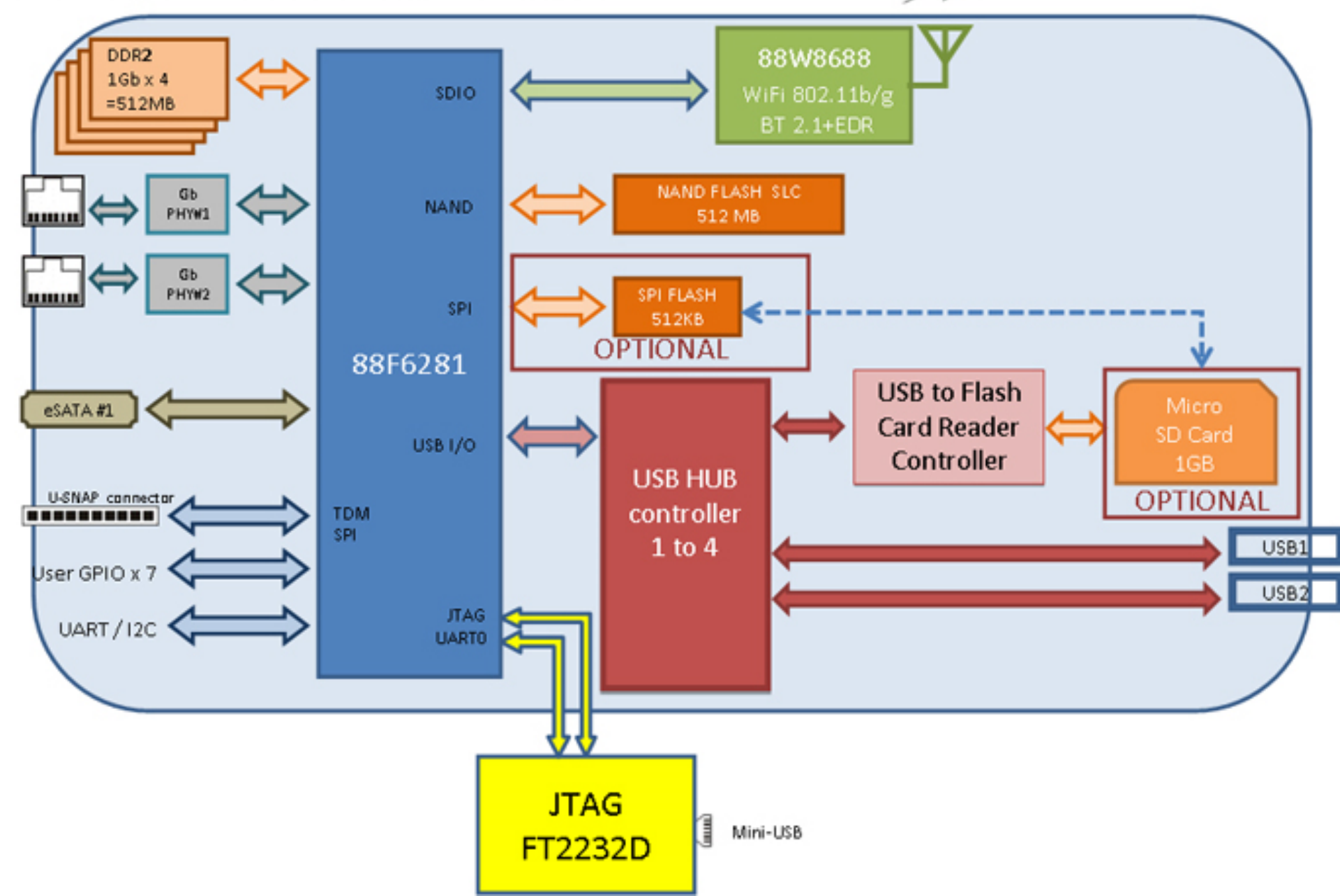
- **Manufacturer**: Marvell
- **Release date**: March 2009
- **Operating system**: Ubuntu
- **Power**: 2.3w idle no attached devices, 7.0w running at 100% CPU utilization
- **CPU**: 1.2 GHz ARM Marvell Kirkwood 88F6281
- **Storage capacity**: External hard drive/ SDIO card/flash disk
- **Memory**: 512 MB SDRAM, 512 MB Flash
- **Connectivity** : USB 2.0, SD, Gigabit Network, JTAG mini USB

Plug Computers by Example
GuruPlug Server Plus

- Successor of SheevaPlug
- Add 2x Gb Ethernet, 2 x USB 2.0, 1x eSATA @ 3Gb/s SATAII, 1x MicroSD Slot
- Wi-Fi and Bluetooth support (using external dongle)

SheevaPlug2 (GuruPlug Server) Luxury version
Hardware Functional block diagram

GTI-100105

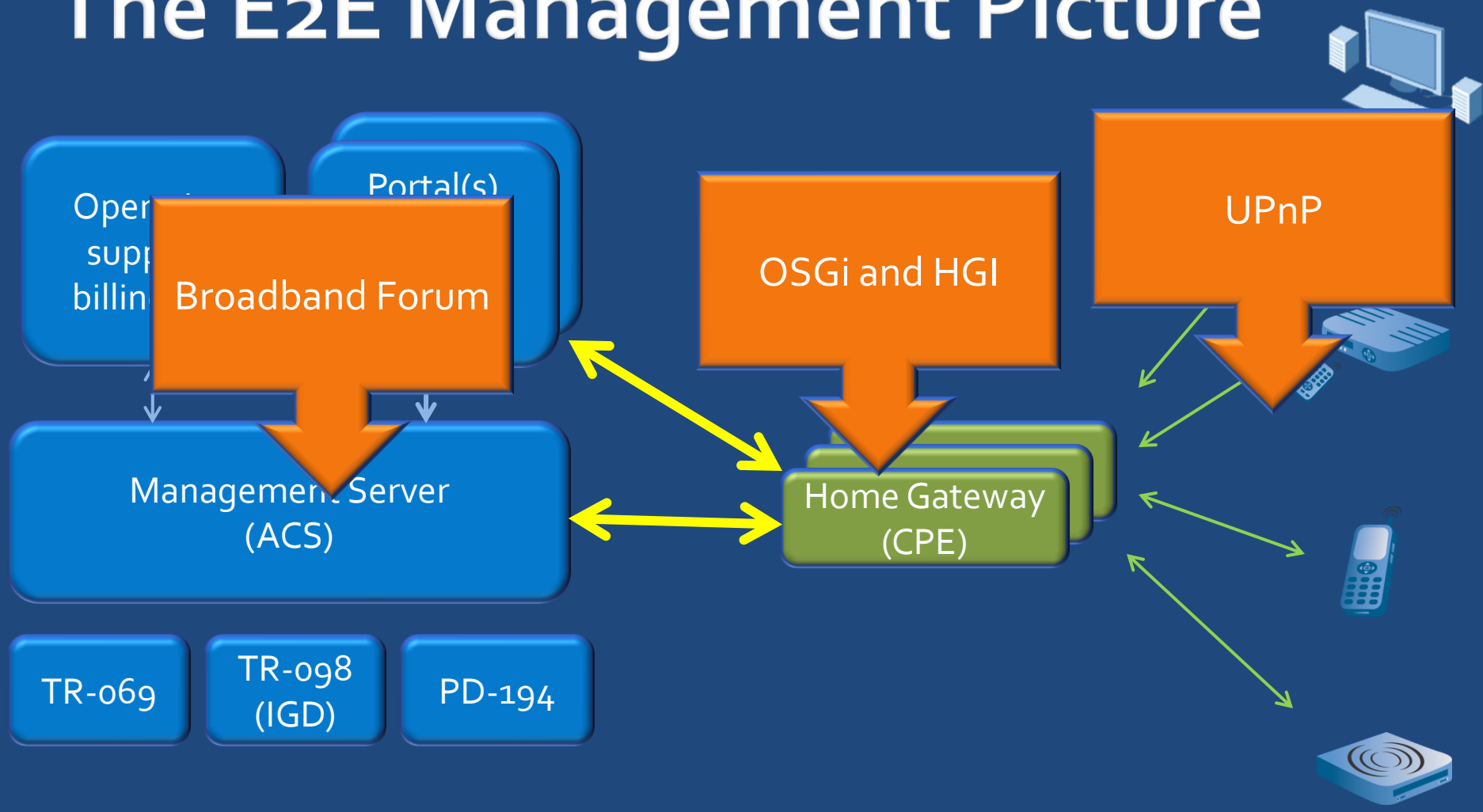


Benefits of Plug Computers

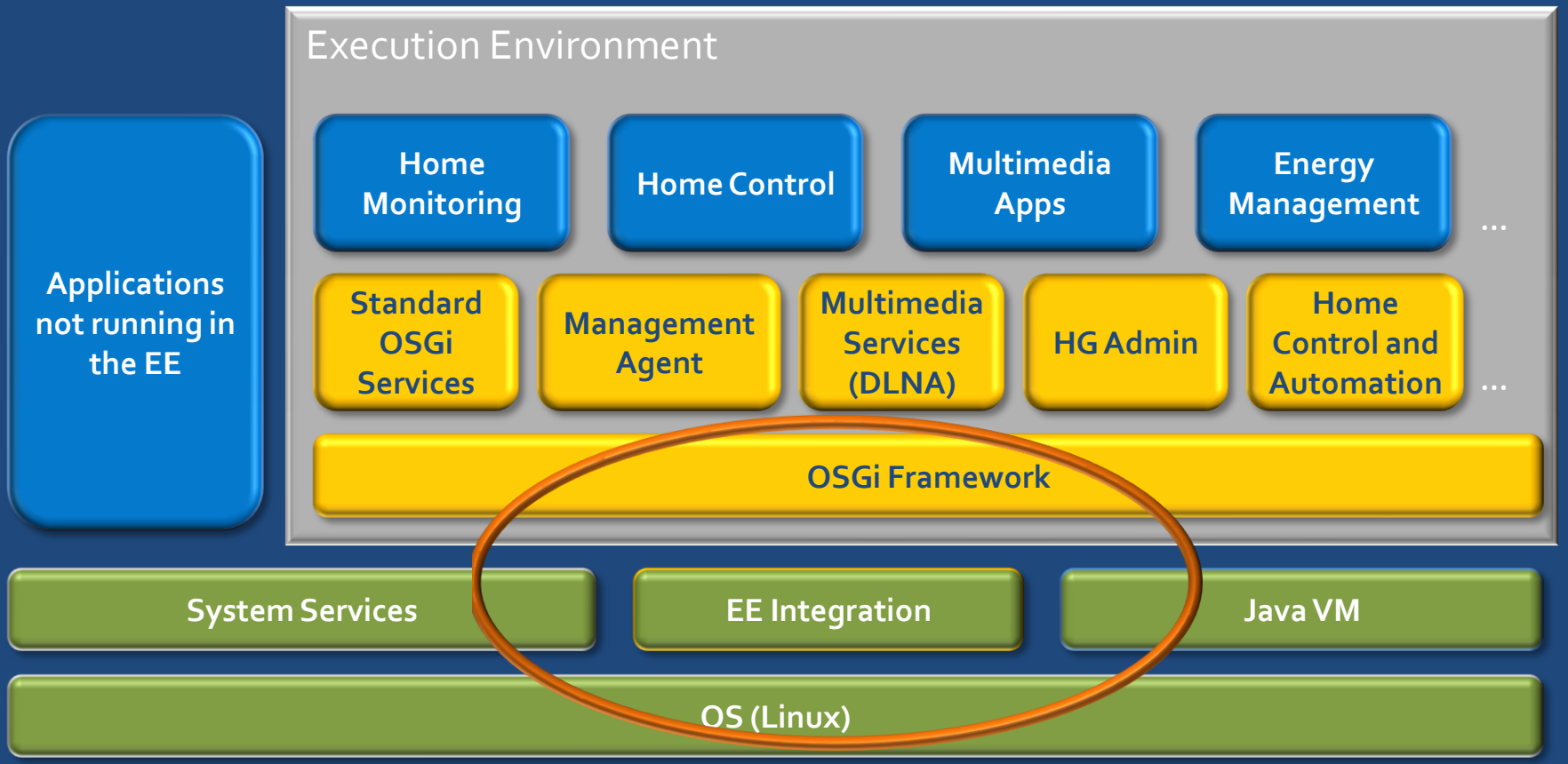
- Power consumption. - drawing under 5 Watts of power, this powerhouse can handle all your biggest tasks while still saving about 96% on energy costs when compared to the average 175 Watt desktop computer.
- Customization - you can customize your Plug to work in almost any industry - Home Automation, Security/Survellance, Medical Monitoring, Industrial Automation, Smart Grid Electrical, Mesh and Grid Computing.
- Connectivity.

OSGi Smart Home Architecture

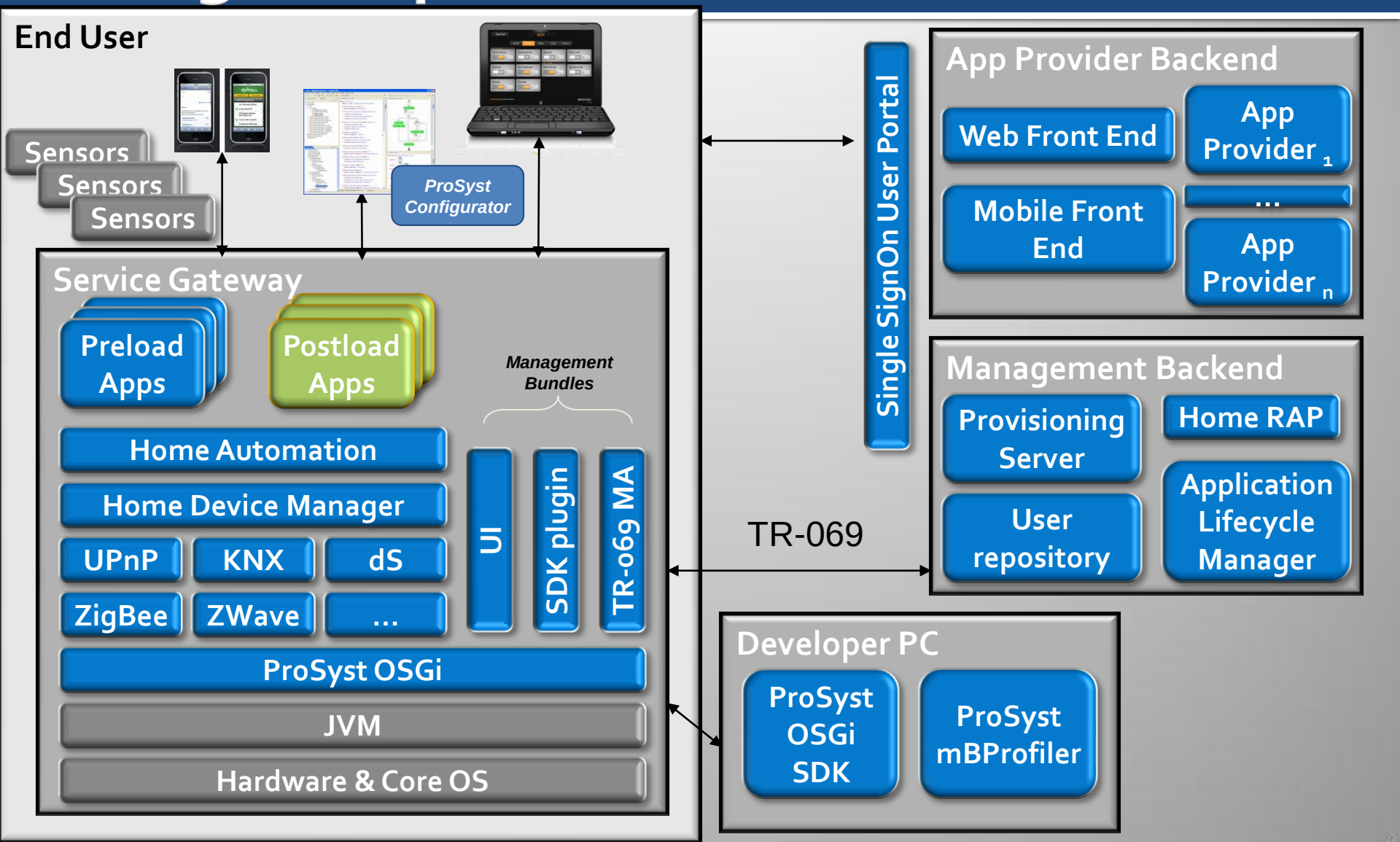
The E2E Management Picture



Using OSGi as EE in Home Gateways

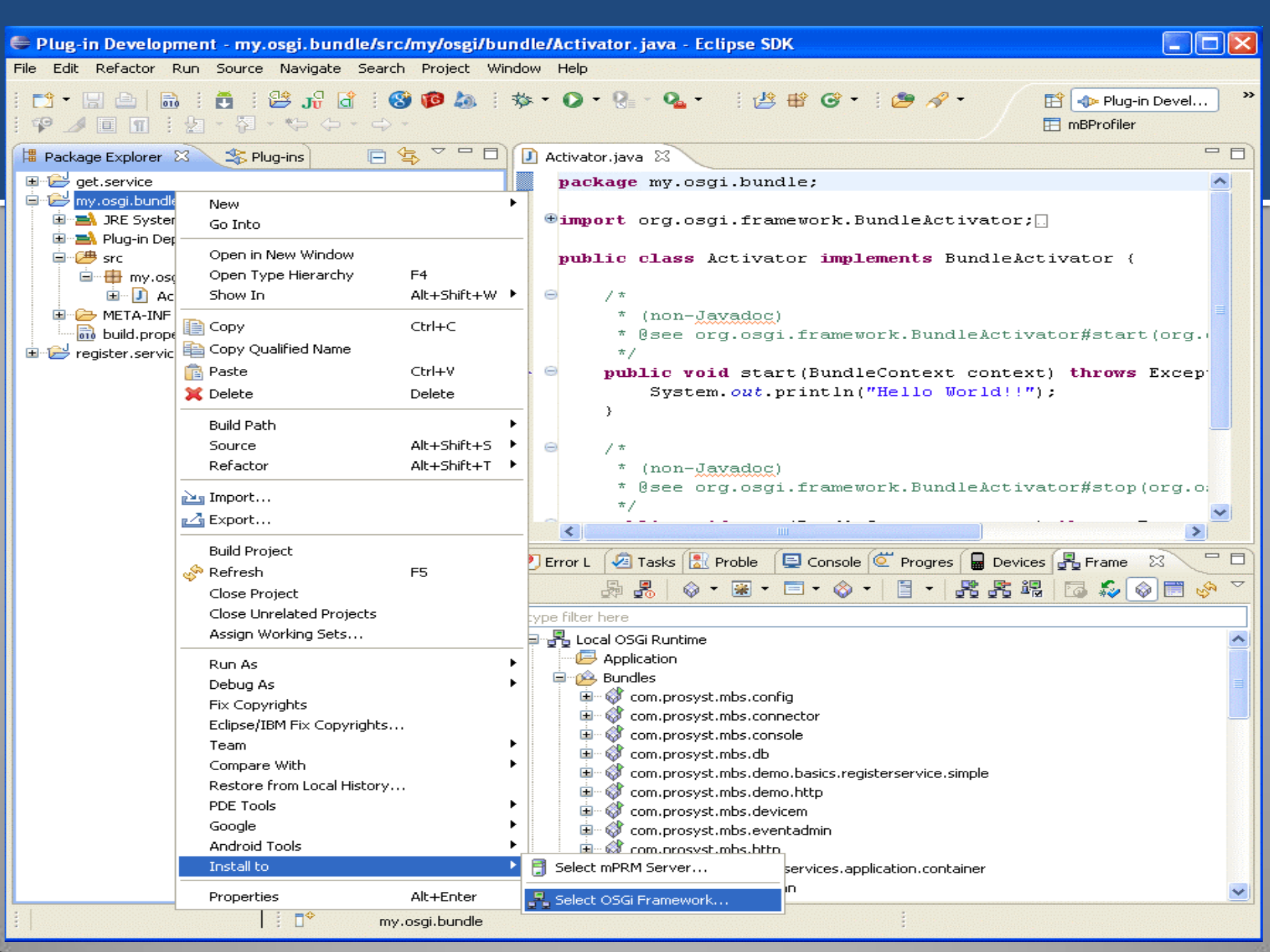


Home Automation System with Plug Computer



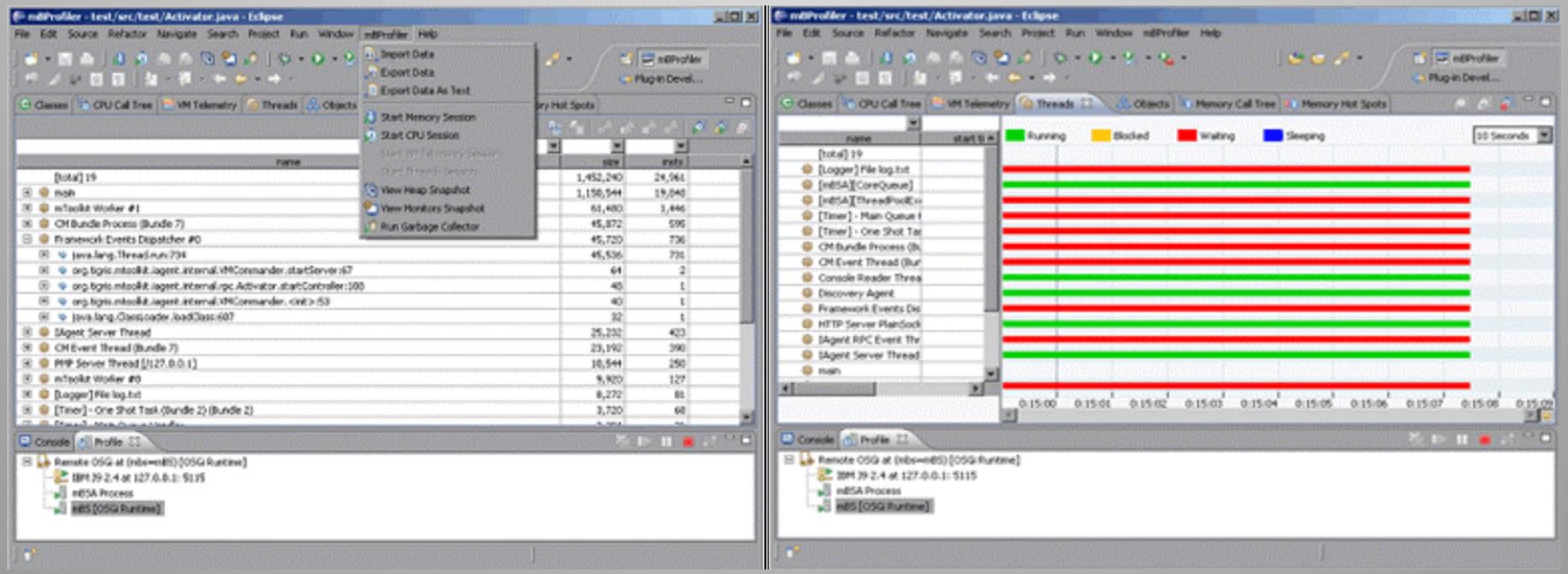
OSGi Development for Plug Computer

- mBS SH SDK is a collection of convenient tools for deployment and management of OSGi-compliant bundles on OSGi Runtimes straight from within the Eclipse Workbench.
- Developers can use a set of Eclipse plug-ins to perform the following tasks:
  - Model and build OSGi Runtime images that best fit the requirements of the target device platform.
  - Debug or profile a remote OSGi Runtime in order to test its performance or the behavior of specific bundles being developed.


Plug-in Development - my.osgi.bundle/src/my/osgi/bundle/Activator.java - Eclipse SDK
File Edit Refactor Run Source Navigate Search Project Window Help
Plug-in Devel...
mBProfiler
Package Explorer
Plug-ins
get.service
my.osgi.bundle
JRE Syster
Plug-in Dep
src
META-INF
build.prope
register.servic
New
Go Into
Open in New Window
Open Type Hierarchy F4
Show In Alt+Shift+W
Copy Ctrl+C
Copy Qualified Name
Paste Ctrl+V
Delete Delete
Build Path
Source Alt+Shift+S
Refactor Alt+Shift+T
Import...
Export...
Build Project
Refresh F5
Close Project
Close Unrelated Projects
Assign Working Sets...
Run As
Debug As
Fix Copyrights
Eclipse/IBM Fix Copyrights...
Team
Compare With
Restore from Local History...
PDE Tools
Google
Android Tools
Install to
Properties Alt+Enter
Select mPRM Server...
Select OSGi Framework...
Activator.java
package my.osgi.bundle;
import org.osgi.framework.BundleActivator;
public class Activator implements BundleActivator {
/*
* (non-Javadoc)
* @see org.osgi.framework.BundleActivator#start(org.
*/
public void start(BundleContext context) throws Excep
System.out.println("Hello World!!");
}
/*
* (non-Javadoc)
* @see org.osgi.framework.BundleActivator#stop(org.o
*/
Error L
Tasks
Proble
Console
Progres
Devices
Frame
type filter here
Local OSGi Runtime
Application
Bundles
com.prosyst.mbs.config
com.prosyst.mbs.connector
com.prosyst.mbs.console
com.prosyst.mbs.db
com.prosyst.mbs.demo.basics.registerservice.simple
com.prosyst.mbs.demo.http
com.prosyst.mbs.devicem
com.prosyst.mbs.eventadmin
com.prosyst.mbs.http
services.application.container
my.osgi.bundle

mBProfiler

- **mBProfiler** assists developers in improving the efficiency of applications. It is focused on testing and exploring different aspects of the performance of a Java program, associated with JVM's consumption of the available platform resources (CPU, memory and threads).



mBS SH SDK Specific Plug-ins

- **J9 JRE Plug-in** – Installs in Eclipse a J9 2.4 JVM fully compliant with the J2ME CDC Personal Profile 1.1
- **Target Platforms Store** - Adds to the Eclipse PDE a bunch of target platforms holding the APIs and services available in the OSGi Runtime.
- **Target Image Descriptors** – Represent a set of pre-defined OSGi Runtime images containing the functional components for the most typical production use cases. Developers can use the image descriptors to generate a ready runtime and deploy it on devices or to design own images.

Live Demo with UPnP

- Demonstrate a simple OSGi service registered in UPnP network
- The service can be personalized
- Demo can be debugged and profiled directly on the device

OSGi + JVM Add-on for SheevaPlug Dev Kit on SD-Card

- The Add-on contains everything that is needed to get an OSGi-based setup started.
- It includes:
  - OSGi Framework
  - JVM
  - Development tools
  - Useful demo applications running on top of the OSGi layer.
- The Add-on is supplied on a SD-card (fits into card slot of SheevaPlug)
- Source: http://www.globalscaletechnologies.com

Conclusions

- OSGi and Plug Computers fits perfectly for use in Smart Home solutions
- The existing OSGi based COTS products can accelerate and facilitate the development of residential products enormously
- ProSyst offers ready OSGi runtimes for many hardware platforms including Plug Computers
- ProSyst provides SDKs for box manufactures, system integrators, operators and application developers
- For more info visit: **http://www.prosyst.com**

Thanks!

Dr. Dimitar Valtchev
d.valtchev@prosyst.com
www.prosyst.com