MEG: OSGi Based Operational Management of Mobile Devices

Vadim Draluk
Motorola, Inc.
Goals

• Creating a comprehensive and uniform mobile device management ecosystem encompassing:
  – Local device settings
  – Device monitoring
  – Over-the-air (OTA) device settings
  – OTA application lifecycle control

• Un-coupling the lifecycles of the device and the software, allowing for software installation and tuning:
  – In the factory
  – In the distribution center
  – At the point of sale
  – After the sale, with the device in user’s hands

• Scaling down to mass-market devices
Principles

• Use of OSGi’s dynamic nature in maximizing control over the lifecycles of:
  – Applications
  – Basic framework software
  – Branding elements

• Reliance on existing industry-standard protocols
  – Protocol-neutrality
  – Multiple protocol support

• Coexistence with native and legacy management systems

• Employment of industry-standard meta-data models for management

• Maximum possible compatibility with other J2ME™ configurations (JSR 246)
**Systems Management Meta-data Models**

- **SNMP**
  - The most widely deployed systems management model
  - Lacking in richness and capabilities

- **JMX (JSR 3, 160)**
  - De-facto standard in J2EE™ servers and applications
  - Lacking the mobile-specific infrastructure
  - Unknown in the mobile industry

- **CIM-WEBM (JSR 48)**
  - Rich meta-model and set of data models for systems and applications management
  - Strongly supported by Microsoft
  - No noticeable penetration into either Java™ or mobile

- **FMA (JSR 9)**
  - A generic distributed systems management standard, originating in disk storage management applications
  - Little traction in the industry
Device Management Tree (DMT) as an Industry-standard Meta-model

- Defined by OMA DM, associated with the OMA DM (SyncML DM) protocol
- Tree of nodes
- All nodes have
  - Name
  - Type
  - Version
  - Title
  - Timestamps
- Leaf nodes have simple values
- Nodes can support all or some of following operations
  - ADD
  - DELETE
  - REPLACE
  - GET
  - EXEC (with parameter)
- ACL-based authorization model per server, per operation, per sub-tree
MEG Adopting the DMT

• Advantages
  – Emerging de-facto standard in the mobile industry
  – Supported by a popular protocol

• Steps taken
  – Separation of the DMT as a meta-data model from OMA DM as a protocol
  – Expanding the notion of principals of the management process
  – Data consistency and integrity capabilities enhanced

• Compatibility fully preserved

• Future enhancements to be advanced through OMA DM
  – Rich data semantics of sibling nodes – vectors, sets
  – Enhanced data constraints
DMT Use: Applications and Agents

- Applications: units of execution activated directly by the user
  - All applications’ parameters are mapped into the DMT
    - OSGi’s Config Admin service data
    - “Native” DMT information
  - Settings applications interface the DMT
- Agents: units of execution activated due to an internal or external event
  - OMA DM
  - OMA CP
  - IOTA
  - SIM-based provisioning
Plug-ins: Data Model Extensibility

• Two data store models for the DMT
  – DMT Registry – store controlled by the DM service
  – DMT plug-ins – stores and data sources abstracted away by data plug-ins
    • OSGi framework info – installed bundles, services, applications
    • Log data
    • Monitors
    • Device readings
    • SIM config data

• Operational extensibility through EXEC plug-ins
  – Management operations associated with DMT nodes
  – Fine-grained policies for operations based on DMT ACLs
Overall Benefits

- Device management radically simplified through:
  - Uniformity of the configuration data
  - Consistent monitoring techniques
- Device management becoming highly adaptable to specifics of:
  - MAs
  - Carriers
  - Regions
- Standardization leads to cost reduction in:
  - Server-side infrastructure
  - Software development
THANK YOU