

libdmclient

An Open Source implementation of OMA-DM

David Navarro
FOSDEM 2013



What is Device Management ?

- Technologies which allow authorized third-parties to remotely perform management operations on an end-user device.
- Use cases are:
 - Applications settings provisioning
 - Connectivity management
 - Software and Firmware update
 - Device lock and wipe
 - Device capabilities access control (e.g. disable camera)
 - Diagnostic and monitoring
 - and more...
- OMA-DM is the device management standard in the mobile phone industry.



OMA-DM Presentation

- Defined by the Open Mobile Alliance
- Current version is 1.2. Version 1.3 is finalized.
- Client-Server protocol.
- Mutual authentication.
- Exchange of SyncML packets over http(s)

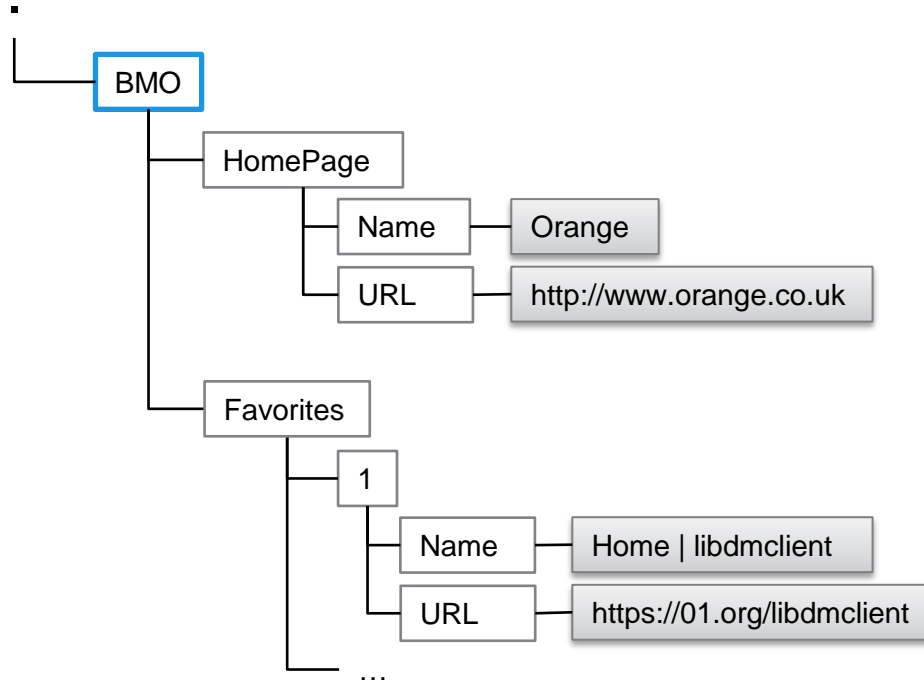


OMA-DM Protocol Overview

- Server sends SyncML commands on device's standardized node URIs. (e.g. `./DevInfo/DevId` or `./LAWMO/Operations/PartiallyLock`)
- Commands are GET, ADD, REPLACE, DELETE and EXEC.
- The nodes collection is called the DM tree.
- Access Control List



Example: Changing the Browser HomePage 1/3



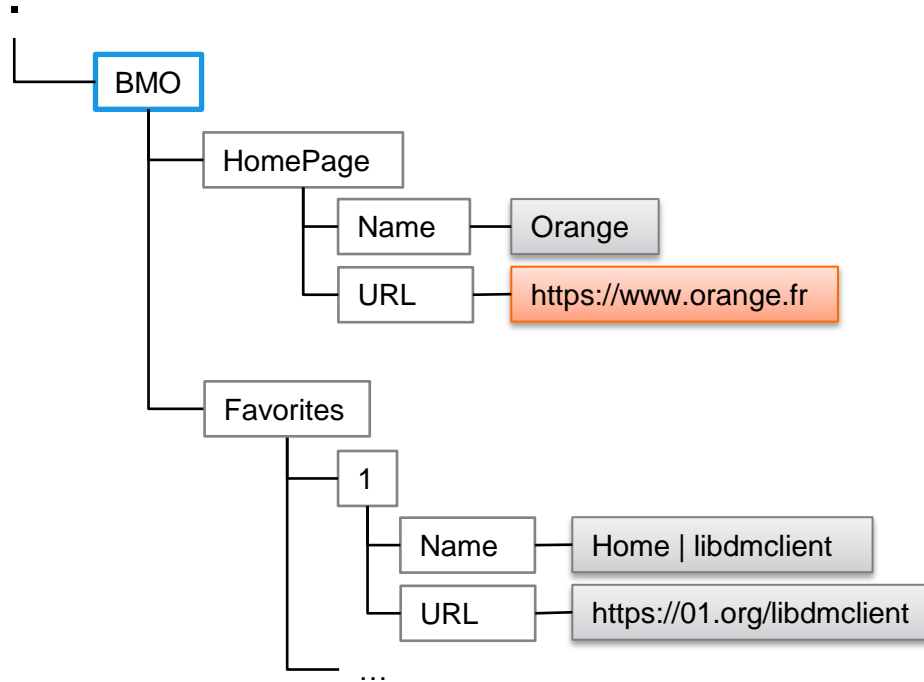
Example: Changing the Browser HomePage 2/3

```
.  
. .  
. .  
<Replace>  
  <CmdID>4</CmdID>  
  <Item>  
    <Target>  
      <LocURI>./BMO/HomePage/URL</LocURI>  
    </Target>  
    <Data>https://www.orange.fr</Data>  
  </Item>  
</Replace>  
. .  
. .
```

```
. .  
. .  
< Status>  
  <CmdRef>4</CmdRef>  
  <Cmd>Replace</Cmd>  
  <Data>200</Data>  
</Status>  
. .  
. .
```



Example: Changing the Browser HomePage 3/3

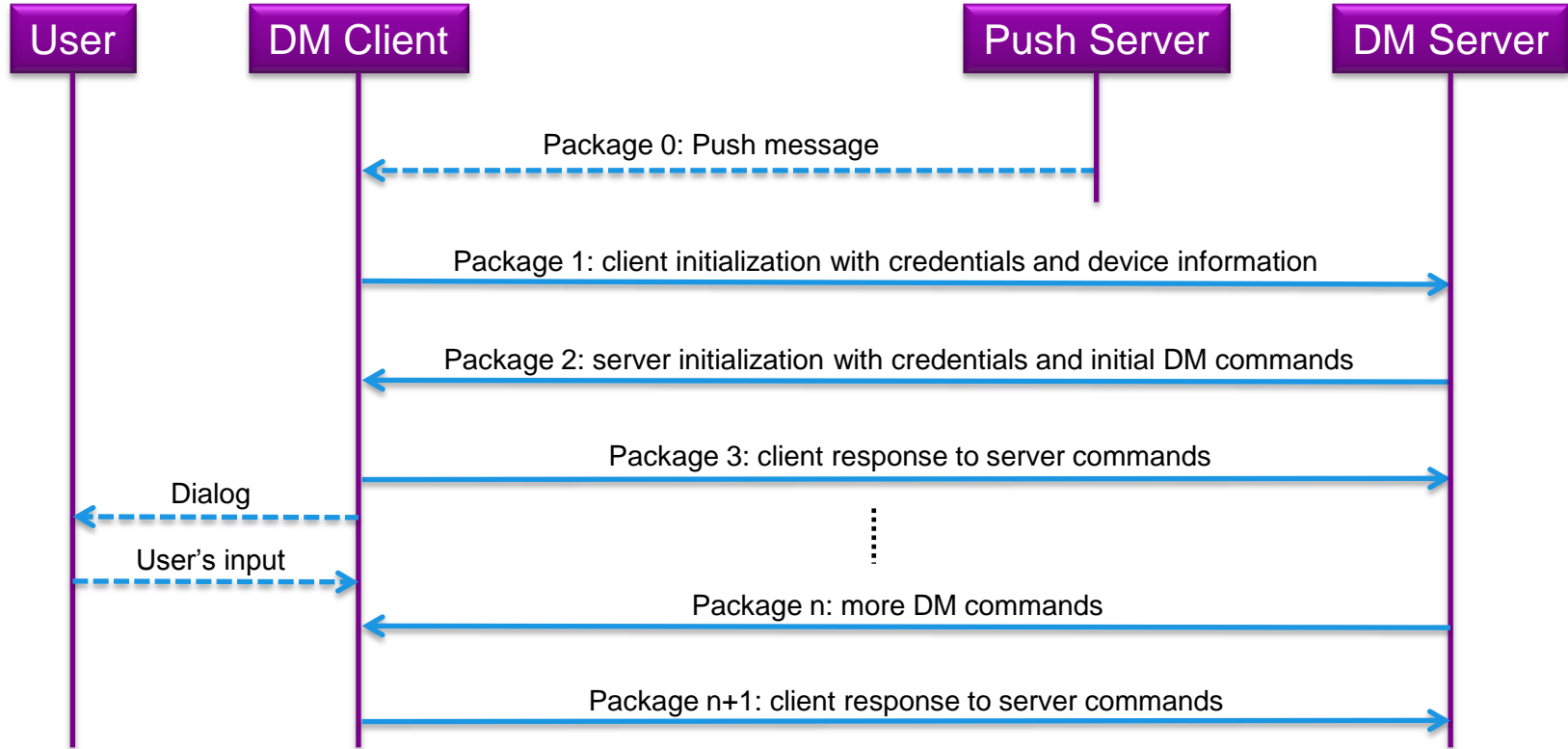


OMA-DM Management Objects

- Nodes are grouped in Management Objects.
- Each MO is an interface for a functionality.
- Three MOs are mandatory:
 - DevInfo
 - DevDetail
 - DmAcc



OMA DM Session Overview



Introducing libdmclient

- Implementation of the client-side of OMA-DM 1.2
- Encodes and decodes OMA-DM packets.
- Dispatches DM commands to plugins.

<https://01.org/libdmclient>



Characteristics

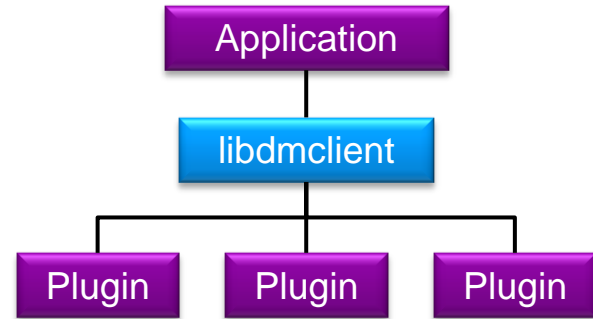
- Designed for Linux
- Written in C
- Single threaded
- No file system access
- Dependencies:
 - libxml2
 - libwbxml



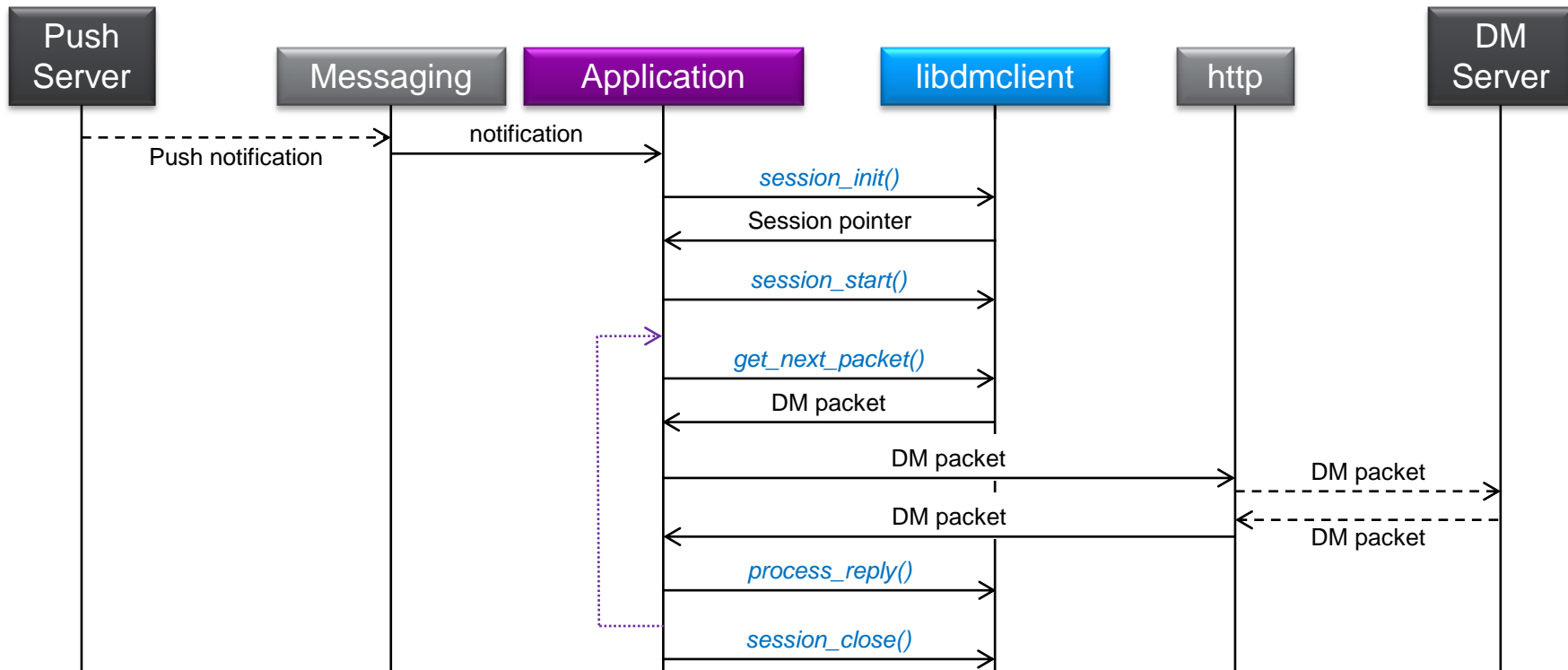
libdmclient Usage Overview

The application takes care of:

- communication with the DM server
- reception of the push notification
- User Interface

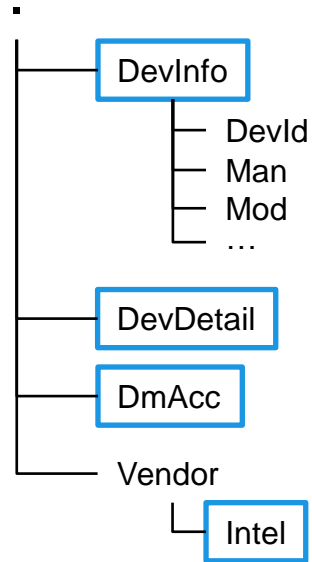


Usage Flow



libdmclient Plugins

- Handle operations on a subpart of the DM tree.



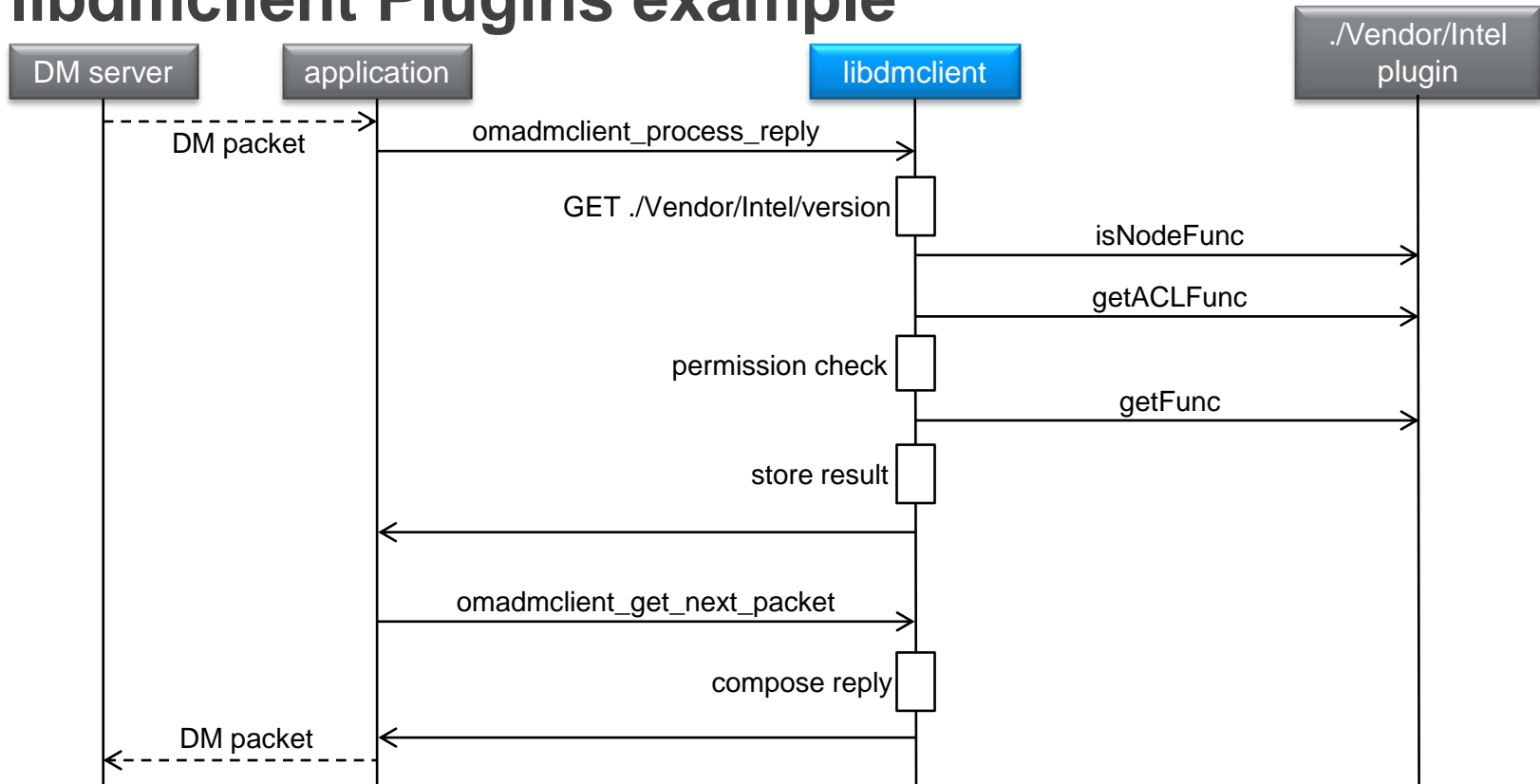
libdmclient Plugins

- Defined by a base URI and several callbacks.
- Loading mechanisms:
 - API
 - shared libraries

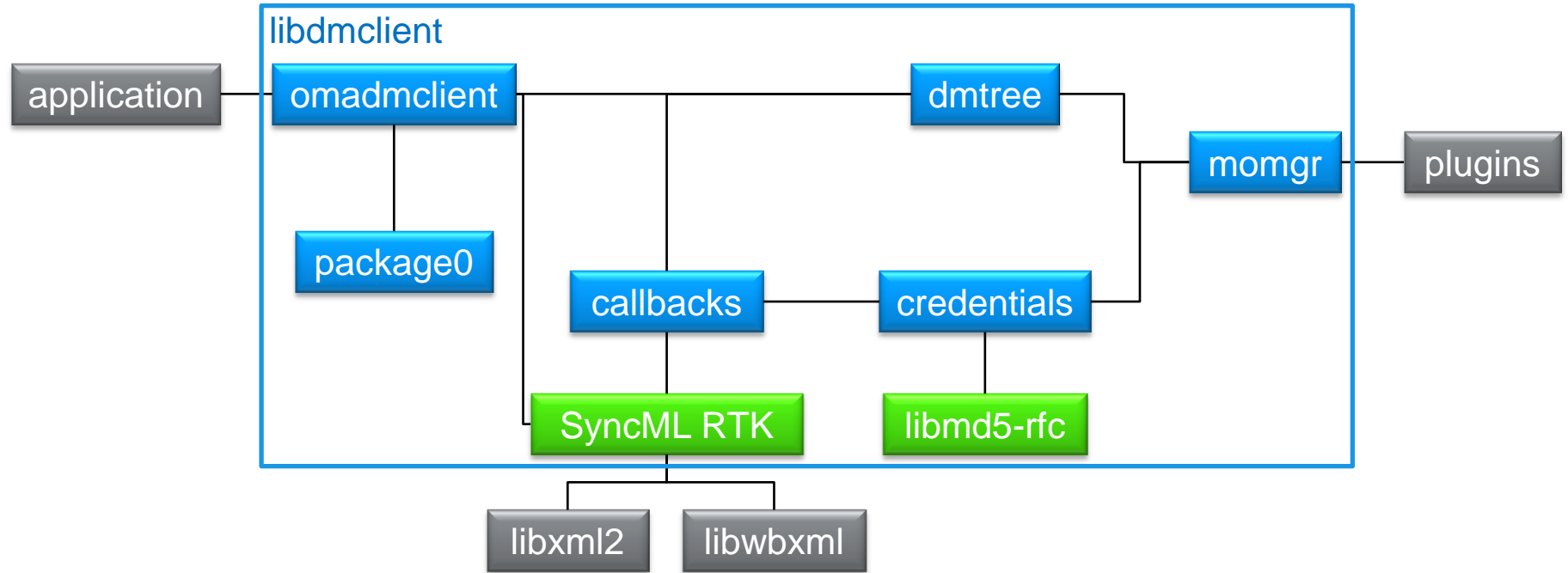
```
typedef struct
{
    char * base_uri;
    omadm_mo_init_fn      initFunc;
    omadm_mo_close_fn    closeFunc;
    omadm_mo_is_node_fn  isNodeFunc;
    omadm_mo_find_urn_fn findURNFunc;
    omadm_mo_get_fn      getFunc;
    omadm_mo_set_fn      setFunc;
    omadm_mo_get_ACL_fn  getACLFunc;
    omadm_mo_set_ACL_fn  setACLFunc;
    omadm_mo_rename_fn   renameFunc;
    omadm_mo_delete_fn   deleteFunc;
    omadm_mo_exec_fn     execFunc;
} omadm_mo_interface_t;
```



libdmclient Plugins example



Component View

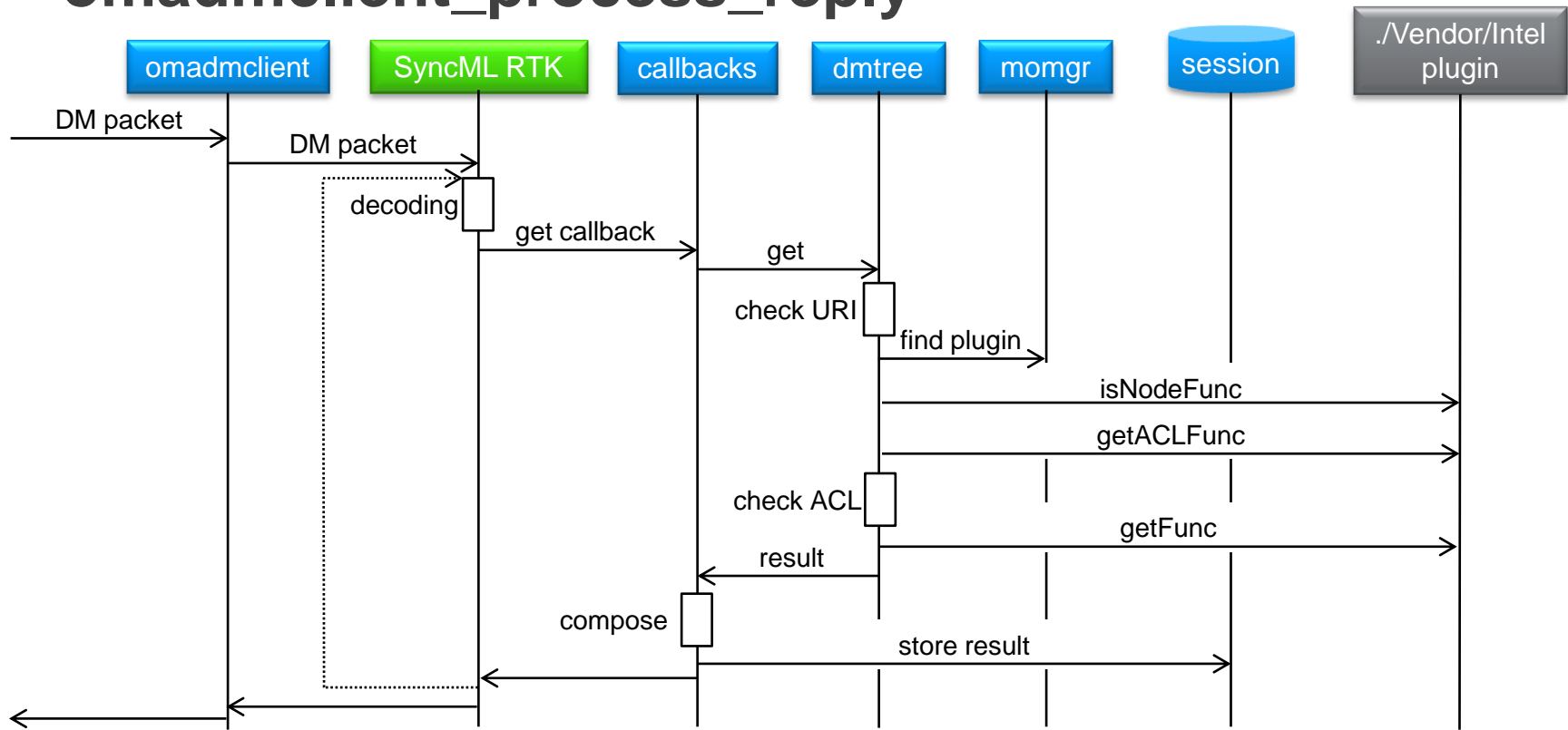


Session Data Structure

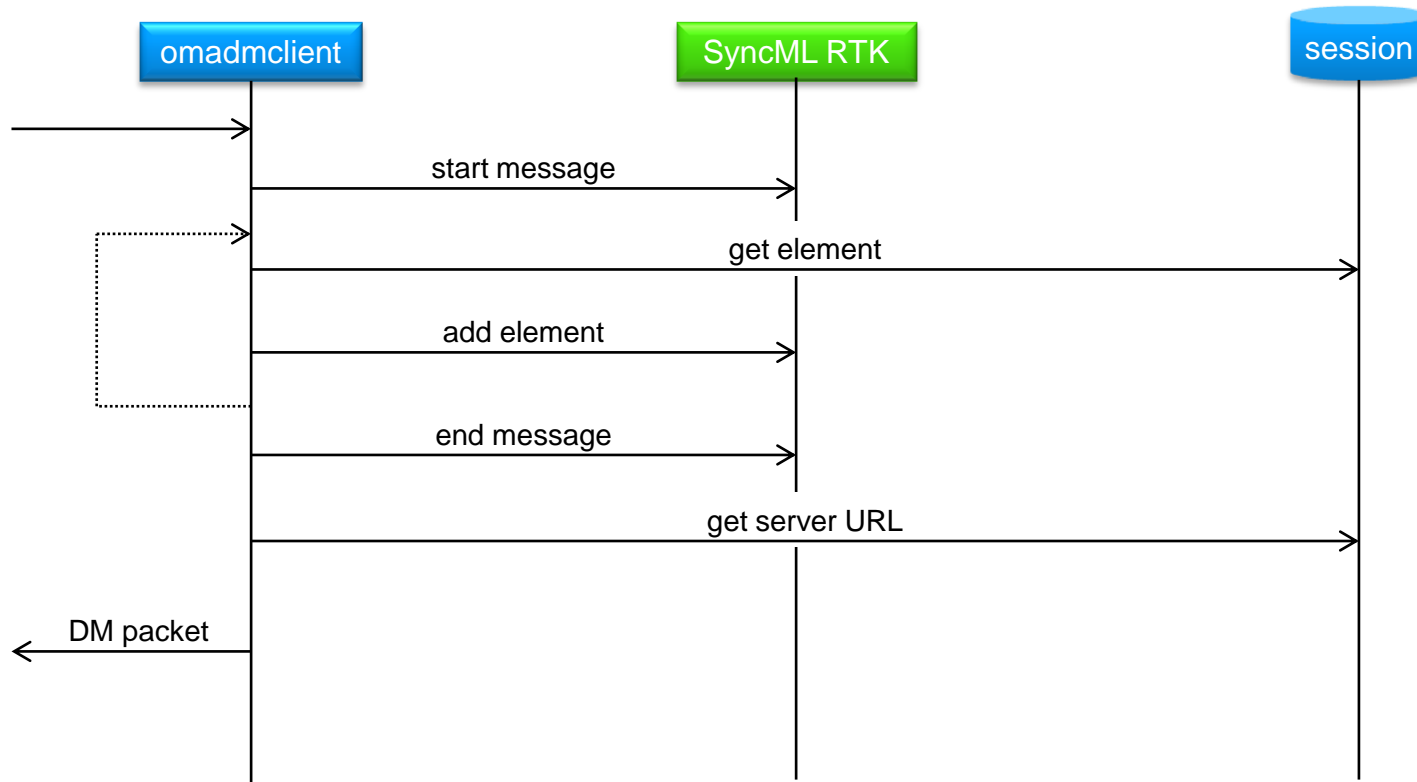
- In memory
- Opaque pointer used in APIs
- Contains all required data
 - state variables
 - plugins tree
 - list of reply elements
 - SyncML RTK handle



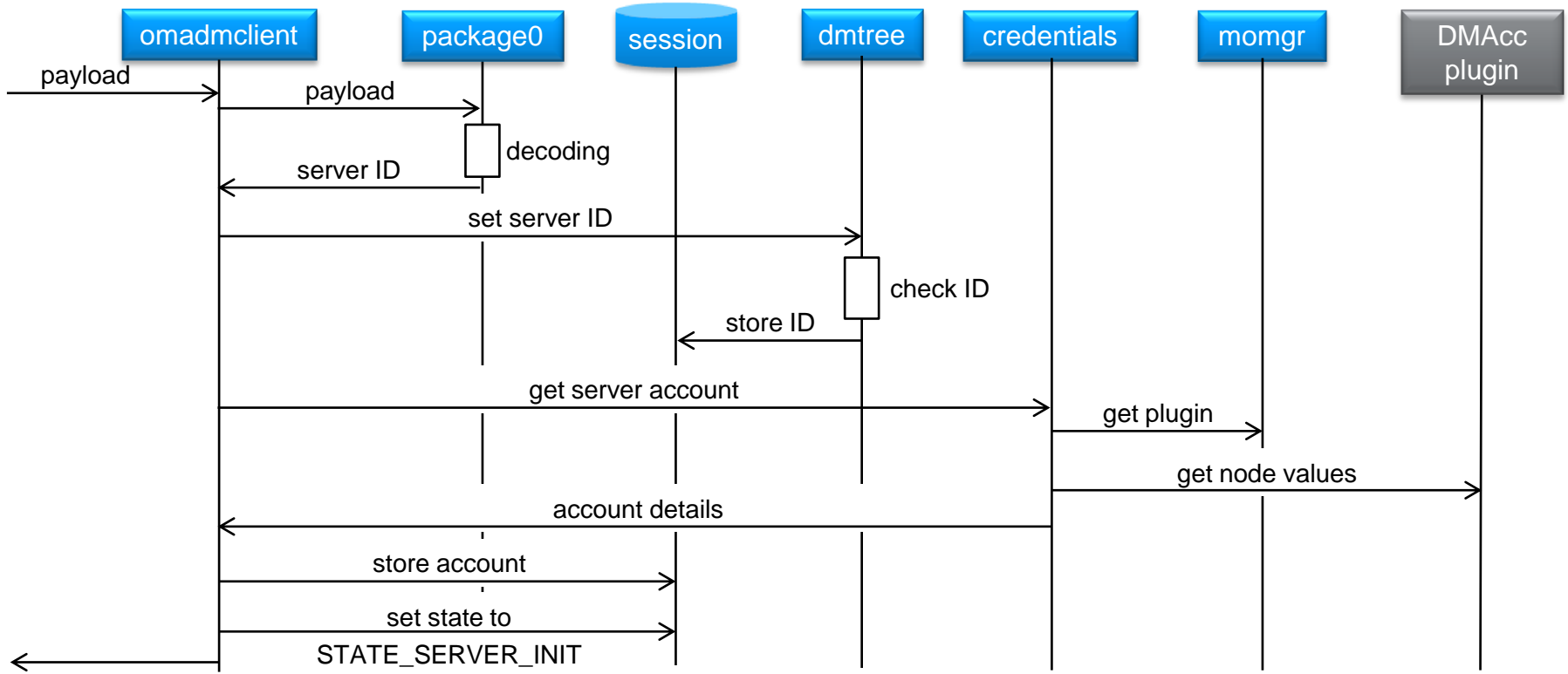
omadmclient_process_reply



omadmclient_get_next_packet



omadmclient_session_start_on alert



Current Status

- IOP with Funambol DM server
- Support of GET, ADD, REPLACE and DELETE
- test material includes:
 - command-line application
 - hard-coded DevDetail, DevInfo and DMAcc
 - storage plugins (in memory and SQLite)



TODOs

- Support OMA-DM 1.3
- EXEC support
- Plugin management rework
- Bootstrapping
- SyncML RTK strip
- Logs
- Yocto integration

<https://01.org/libdmclient>



OP CS YUPTURE INTEL LINUX WIRELESS
YOCTO CONNMANXEN GUPNP KVM POKY
SYNCEVOLUTION SIMPLE FIRMWARE INTERFACE (SFI) OFONO LINUX KERNEL
ENTERPRISE SECURITY INFRASTRUCTURE



INTEL OPEN SOURCE
TECHNOLOGY CENTER