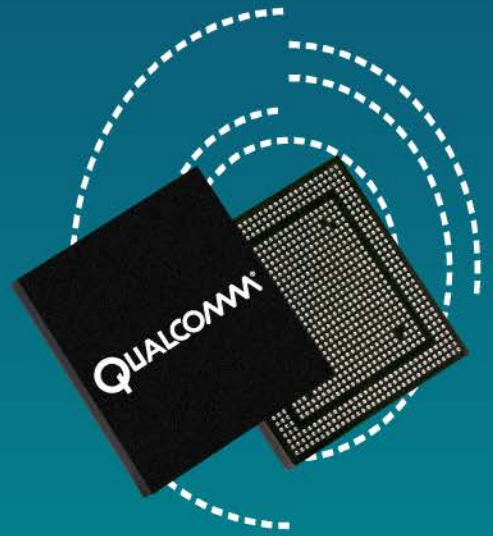
Qualcomm
TECHNOLOGIES, INC



Policy Manager Overview

80-NJ017-14 D

Qualcomm Technologies, Inc.
5775 Morehouse Drive
San Diego, CA 92121
U.S.A.

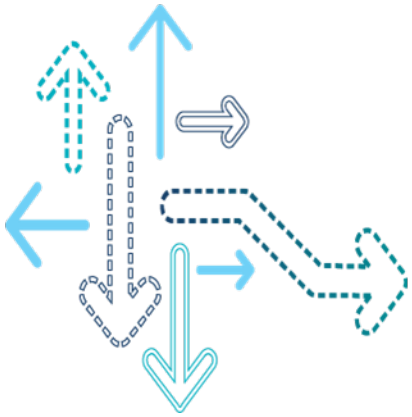
Revision History

Revision	Date	Description
A	Sep 2013	Initial release
B	Sep 2013	Updated Policyman XML Configuration slide
C	Oct 2013	Added SGLTE Policy Requirements slide; updated Configurable Elements and References slides
D	Jan 2014	Updated title and slides 14, 21-24, 26, 29, and 30

Contents

- Introduction
- Policy
- Examples
- References
- Questions?

Introduction



Objective

- At the end of this presentation, you will be able to understand:
  - What Policy Manager is
  - The high-level algorithm and use case scenarios
  - Code
  - Analyzing logs, call flows, and debug issues

Prerequisites

- Policy Manager is supported from Dime 2.0 and later.
- QXDM Professional™ Ver 03.14.557 is needed to capture Policy Manager log messages.
  - To enable, go to QXDM Pro under menu Options→Message View Config→ Message Packets→Policy Manager.
- Users should be familiar with basic Extensible Markup Language (XML) concepts.

General Overview

- Policy Manager is a new Multimode Call Processor (MMCP) component designed to manage the policy of the modem.
- It provides various configuration items based on the current state of the device and network.
- Policy Manager decides what is to be done; it is the modem's responsibility to handle how it is done.
- This allows for keeping carrier-specific requirements out of the core modem code, isolating these in the Policy Manager component.
- Carrier-specific policies are specified as a set of rules expressed in an .xml configuration file.

Basic Policy Manager Operation

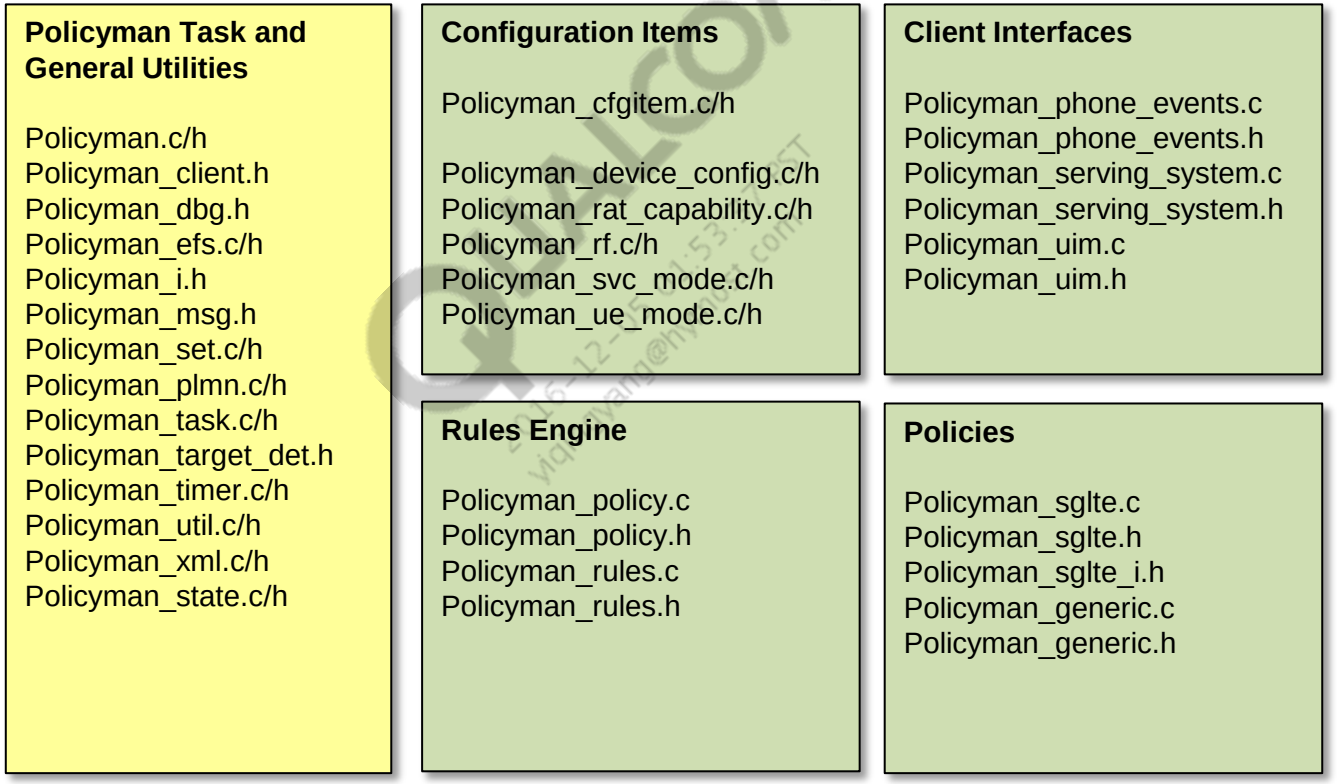
- The two major components of policyman are a database of current configuration items and a rules engine.
- When the policyman task starts, it reads and parses an .xml file (/policyman/carrier_policy.xml), to create a set of rules to be run by its rules engine.
  - The .xml file also specifies which policy code within policyman is to be used.
  - While parsing the .xml, policyman looks at the conditions used in the rules and uses these to build a set of preconditions that must be met before rules can be evaluated. Current preconditions are:
    - Serving system event received from MMODE
    - SIM type available
    - SIM IMSI available

Basic Policy Manager Operation (cont.)

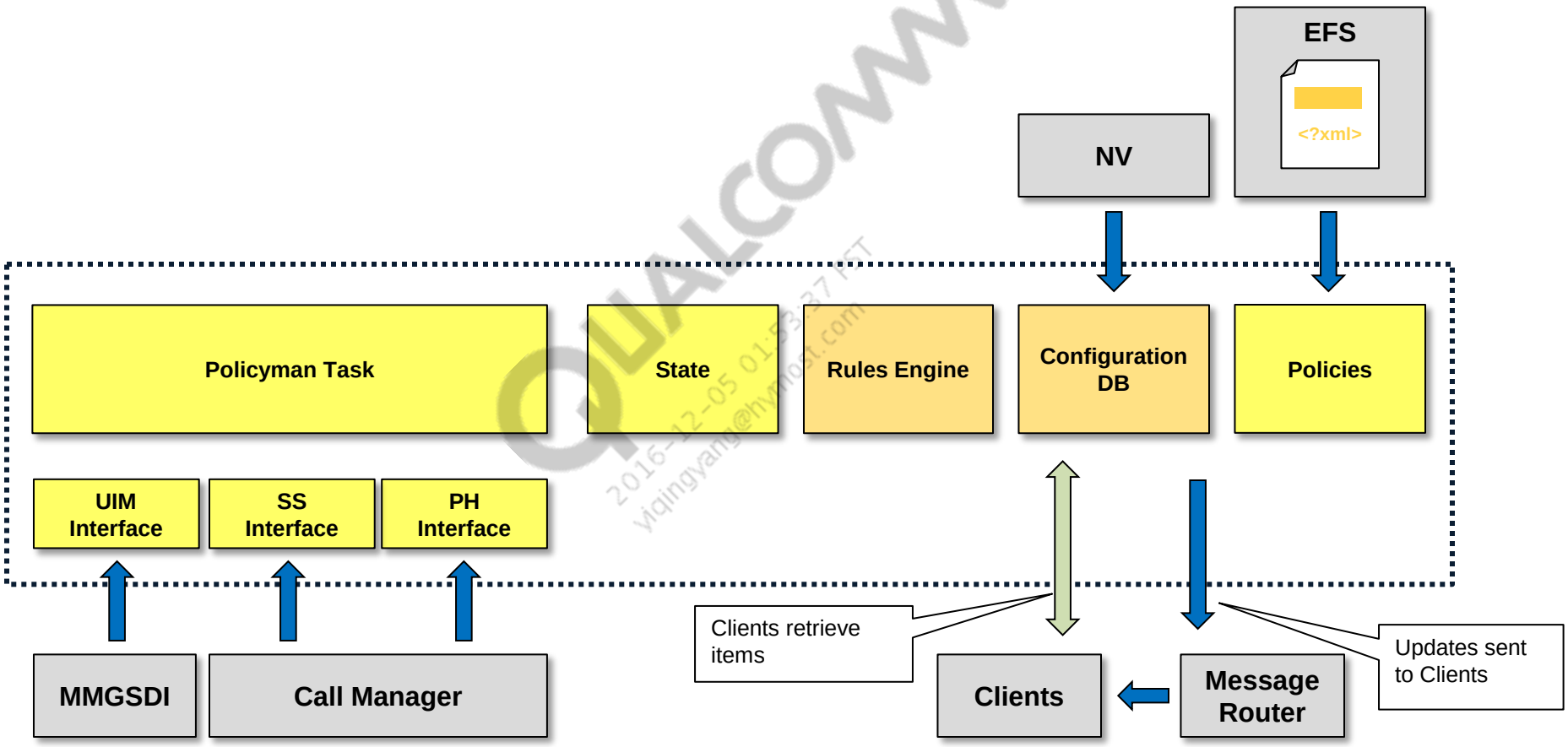
- Once the policy has been determined and any rules created, policyman registers for:
  - UIM events
  - Serving system events
  - Phone events
- At this point, policyman publishes the initial values of its configuration items (perhaps from the .xml file, perhaps just default values) and goes to sleep, waiting for events to fire.
- When an event fires, policyman determines whether the preconditions for evaluating the rules have been met. If so, policyman executes the rules engine with the policy's rules and determines the current values of the configuration items that it manages.
- Policyman updates its configuration database with the new current values of the configuration items. If any have changed, it sends a Message Router (MSGR) message that contains only the changed items.

Policyman Modules

- Policyman services are organized into groups.



High-Level Architecture



Configuration Items Managed

- Policyman works with the following configuration items.

Item	Description
POLICYMAN_ITEM_RAT_CAPABILITY	A bitmask of the RATs to be used by the device
POLICYMAN_ITEM_RF_BANDS	Bitmasks of the RF bands (GW/LTE/TD-SCDMA) to be used by the device
POLICYMAN_ITEM_SVC_MODE	Service mode (camp_only, limited, full) to be used by the device
POLICYMAN_ITEM_UE_MODE	UE mode (normal, SGLTE, CSFB) to be used by the device
POLICYMAN_ITEM_DEVICE_CONFIGURATION	Device configuration (single-SIM, DSDS, DSDA, etc.) to be used by the device

Policyman APIs

- Policyman exposes configuration items through a query API.

```
policyman_status_t policyman_get_items(*pIds,numIds,**ppItem);
policyman_item_id_t policyman_item_get_id(*pItem);
```

- When items change, a MSGR message containing all the changed items is sent. Components can register to receive the POLICYMAN_CFG_ UPDATE_IND message to learn of changes in device configuration.
- Each updated item is an attachment to the message of type MSGR_ATTACH_REFCNT.

```
void policyman_msgr_attachments_release(pMsg)
```

- CM (Call Manager) module will be the main Client of Policyman and will be listening to any change in policy which is reported.
  - CMPMPRX (cmpmprx.c) acts as the main interface to process configuration items and propagates the changes to SD and Protocol Layers

Service Mode

- Currently supported Service modes

Mode	Description
POLICYMAN_SVC_MODE_LIMITED	Limited service will be selected with network
POLICYMAN_SVC_MODE_CAMP_ONLY	No registration or OTA will be performed with network
POLICYMAN_SVC_MODE_FULL	Full registration will be selected with network

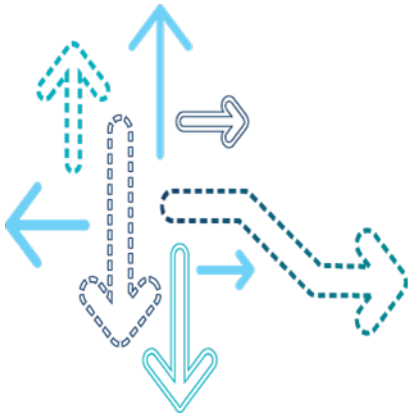
UE Mode

- Supported UE modes are:

UE mode	Description
SYS_UE_MODE_NORMAL	NORMAL is UE mode used when UE in legacy behavior. It is used before SIM is initialized, functions similarly to CSFB (single stack).
SYS_UE_MODE_SGLTE_HOME	This is SGLTE mode. The UE could function in either single stack or dual stack. No WCDMA. GSM/LTE bands are restricted to SGLTE band set.
SYS_UE_MODE_SGLTE_ROAM	This is similar to CSFB mode. However, the UE can switch back and forth between SGLTE_HOME and SGLTE_ROAM depending on UE location's Mobile Country Code (MCC).
SYS_UE_MODE_CSFB_ONLY	This is CSFB mode, and the UE is not allowed to move to SGLTE.
SYS_UE_MODE_SGLTE_TEST_ONLY	This is used for testing. It is SGLTE mode, and the UE cannot be allowed to move to CSFB regardless of UE location.

- If the item is not present and SGLTE policy is running, the default value of SGLTE_HOME is used. If the file does not exist at UE startup time, Policy Manager will set the value to 1 and write the file to EFS.
- If SGLTE policy is not in effect, the UE mode defaults to NORMAL.

Policy



Policyman XML Configuration

- Policyman rules are stored in EFS in the file /policyman/carrier_policy.xml.
- Files are write-once after they are provisioned to avoid alterations to the policy.
  - The EFS directory /policyman is made read-only when the filename locked is written to the /policyman directory; the size and contents of this file are not important.
  - Once locked is present, the only way to rewrite the file is to reflash the device.

Policyman XML Schema

- The basic structure of a policyman .xml configuration file is a policy containing an initial element followed by an ordered set of rules.
- <policy> is the single top-level element that specifies the policy.
  - The name attribute specifies the name of the policy.
  - The enabled attribute specifies whether the policy should be run by policyman.

```
<?xml version="1.0" encoding="UTF-8"?>
<policy name="rat capability" enabled="true" schema_ver="1" policy_ver="1">
  <initial>
    <actions>
      <!-- Initial actions to take when policy is initialized. -->
    </actions>
  </initial>
  <rule>
    <conditions>
      <!-- Conditions under which this rule is to be followed. -->
    </conditions>
    <actions>
      <!-- Actions to carry out if all conditions are true. -->
    </actions>
  </rule>
  <!-- More rules can follow. -->
</policy>
```


XML-Based Rules

- A policy may contain an <initial> element that can set initial values of configuration items, specify timer names and intervals, and create named lists of things to be used in the rules.
- These actions are executed when the policy is initialized during boot.
- This is where the customer-configurable parts of the policy are usually defined.

```
...
 <initial>
    <actions>
      <!-- These are the operators (IMSI PLMNs) for which SGLTE will be allowed -->
      <plmn_list name = "sglte_operators">
        460-00 460-02 460-07 460-08
      </plmn_list>

      <!-- Define the OOS timer with a 5 minute interval -->
      <!-- NOTE: Proper functioning of the SGLTE policy requires that there
           be a timer named "oos".  Do NOT rename this timer.
      -->
      <define_timer name = "oos" interval = "5" units = "min" />

      ...
    </actions>
 </initial>
```

XML-Based Rules (cont.)

- Following the <initial> block is a set of zero or more <rule> elements.
  - A rule has a set of conditions and a set of actions.
  - The policyman rules engine evaluates the conditions of the rules in order.
  - The actions of the first rule whose conditions all evaluate to TRUE are executed and rule evaluation stops unless there is a <continue> element.

```
<!-- Configuration for CMCC SIMs operating in China -->
<rule>
  <conditions>
    <imsi_plmn_in list = "CMCC_plmns" />
  </conditions>
  <actions>
    <rat_capability base = "hardware" />
  </actions>
</rule>
<!-- Configuration for other Chinese SIMS operating in China -->
<rule>
  <conditions>
    <imsi_mcc_in> 460 </imsi_mcc_in>
  </conditions>
  <actions>
    <rat_capability base = "none"> <include> GSM </include> </rat_capability>
  </actions>
</rule>
```

Configurable Elements

- Currently available elements that can be used to configure values used in rules

Element	Description
<plmn_list>	Creates a named list of PLMNs that can be used in <serving_plmn_in> and <imsi_plmn_in> conditions; certain named lists may also be used in C code
<mcc_list>	Creates a name list of MCCs that can be used in <serving_mcc_in> condition
<gw_bands> <lte_bands> <tds_bands>	Sets GW, LTE and TDS band configuration based on RF Capability (base=hardware), or manually defined using <include></include> (base=none).
<rf_band_list>	Creates a named list of RF bands (combination of <gw_bands>, <lte_bands>, and <tds_bands>) that can be used to set the RF bands to use in <rf_bands> actions; the SGLTE also accesses specific named RF band lists within C code
<define_timer>	Creates a named timer to be used within C code in the policies; the OoS timer is used in all policies
<boolean_set>	Creates a named Boolean and stores a TRUE or FALSE value. This boolean can be evaluated using conditional element <boolean_test>

Condition Elements

- Currently available conditions

Element	Description
<true>	Always evaluates to TRUE; used for the last rule in a policy to specify what to do if none of the other rules fires
<not>	TRUE if encapsulated conditional element evaluates to FALSE and vice versa. Equivalent to a logical “!” (not) operation.
<serving_mcc_in>	TRUE if the MCC of the serving system is in the inline list specified in this element
<serving_plmn_in>	TRUE if the PLMN of the serving system is in the list (named or inline) specified in this element
<imsi_plmn_in>	TRUE if the PLMN of the operator contained in the SIM card is in the list (named or inline) specified in this element
<imsi_mcc_in>	TRUE if the MCC contained in the PLMN of the operator contained in the SIM card is in the inline list specified in this element
<sim_type>	TRUE if the SIM Type matches with the current loaded SIM type
<user_domain_pref>	TRUE if the user domain preference (CS, PS, CSPS) is equal to the preference specified in this element
<user_mode_pref_contains>	TRUE if the user mode preference contains the RATs specified in this element
<ue_mode_is>	TRUE if the current UE mode (NORMAL, SGLTE, CSFB, etc.) is equal to the mode specified in this element
<any_of>	TRUE if any of the encapsulated condition elements evaluate to TRUE. Equivalent of an logical “||” (or) operation.
<boolean_test>	TRUE or FALSE depending on the value set through configuration element <boolean_set>

Actions Elements

- Action elements update the configuration items that policyman manages.

UE mode	Description
<device_configuration>	Specifies the device configuration, i.e., the maximum number of subscriptions handled, how many subscriptions may be simultaneously active, which RATs may be used (this is for the HLOS), and which features (SGLTE, SVLTE, etc.) are enabled
<rat_capability>	Sets the RATs on which the device may attach; may <include> or <exclude> RATs from the base, which may be hardware or none
<ue_mode>	Sets the UE mode (NORMAL, SGLTE, CSFB, CSFB_ONLY, SGLTE_TEST_ONLY) in which the device should operate
<rf_bands>	Sets the RF bands (<gw_bands>, <lte_bands>, <tds_bands>) to use; may use named or inline lists
<svc_mode>	Sets the service mode (FULL, CAMP_ONLY, LIMITED) to use on the device
<continue>	Indicates that rule evaluation can continue after rule execution

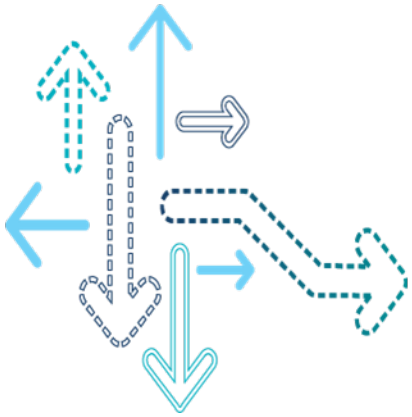
SGLTE Policy Requirements

- In order for policyman to operate in SGLTE mode, the UE needs to have the following:
  - 3G SIM card
  - Mode preference (NV 10) should include LTE and GSM
  - Domain preference (NV 850) should be set to CS PS
- For additional SGLTE settings, see [Q2].

Timers

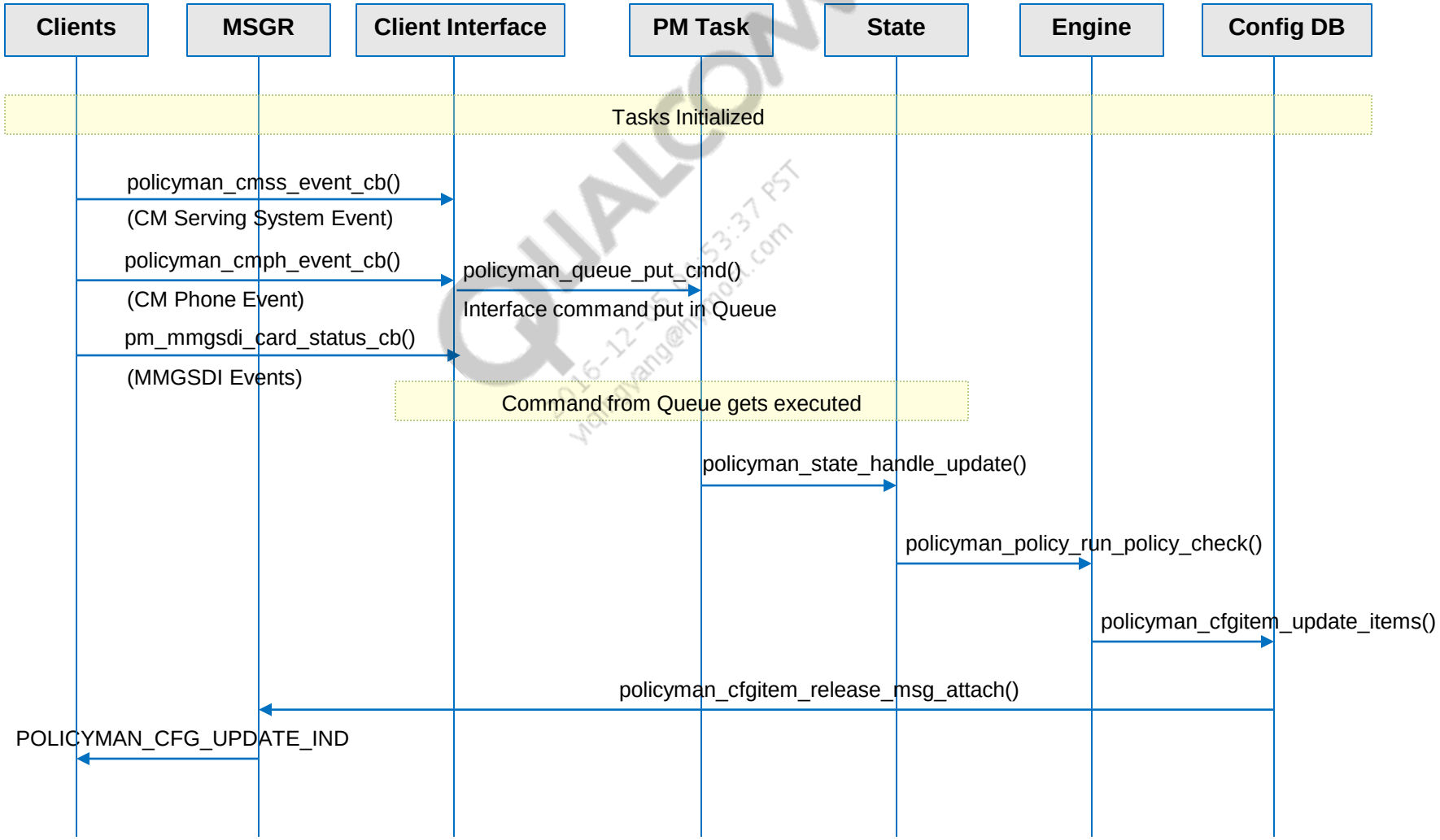
- OoS timer
  - PM starts timer when UE loses service
  - When timer expires, PM will enable all RATs so that UE has a chance to find any type of service
  - Acquiring service on any RAT will cause PM to disable the timer
- LTE Band 39 Forbid Timer
  - Transmission in B39 from LTE or TD-SCDMA will desense GSM in 1850 to 1880 MHz due to Receive Band Noise
  - PM removes B39 upon B39Forbid indication from GERAN (GERAN_GRR_UPDATE_FORBIDDEN_BANDS_IND) and starts the Ttmer.
  - PM restores B39 upon Timer Expiration or B39 Restore indication.

Examples



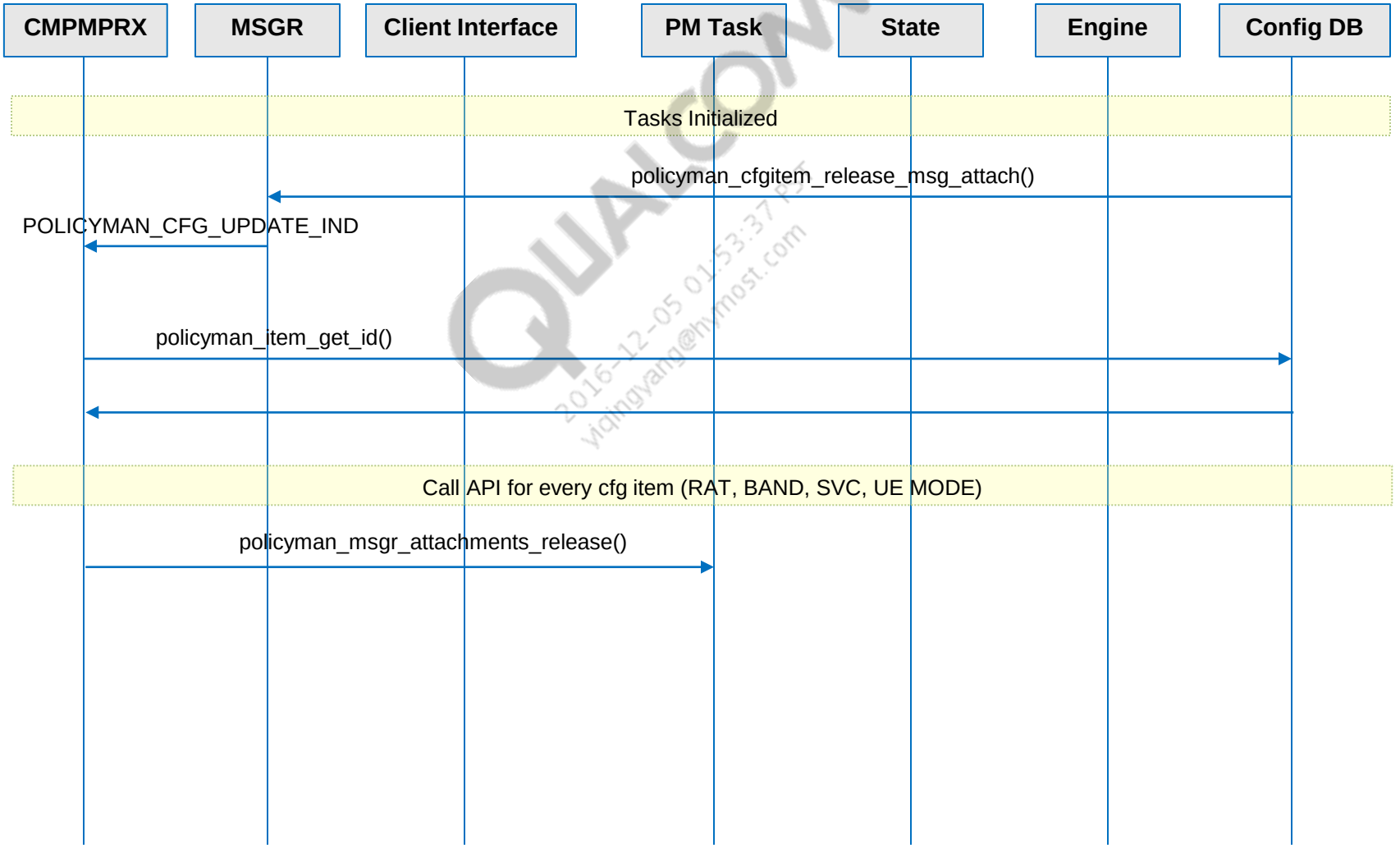
Call Flow

- Event update scenario



Call Flow (cont.)

- Client fetches configuration item from PM



Log Analysis – Preconditions

// PM first needs to meet al preconditions to start evaluating rules
// First 2 preconditions are provided by SIM events
00:00:02.525 mmgsdi.c 8811 H MMGSDI_SUBSCRIPTION_READY_EVT, app: 0x3, slot: 0x1
00:00:02.525 policyman_uim.c 560 H PM got MMGSDI evt 12
00:00:02.525 policyman_state.c 99 H Preconditions changed from 0 to 4
00:00:02.583 policyman_uim.c 386 H PM IMSI 0x0 0xf1 0x10
00:00:02.583 policyman_state.c 99 H Preconditions changed from 4 to 5

// Last precondition is met when UE finally camps on a system
00:00:10.828 policyman_serving_system.c 1005 H received cm ss evt 0
00:00:10.828 policyman_serving_system.c 1921 H cm_serving_info: stack 0, received sys_mode 9, srv_status 1
00:00:10.828 policyman_plmn.c 190 H MCC = 1, MNC = 1, mnc_includes_pcs_digit = 0
00:00:10.828 policyman_state.c 99 H Preconditions changed from 5 to 7

// PM starts with the first rule, these should map with carrier_policy.xml in UE's EFS /policyman/
00:00:10.828 policyman_rules.c 456 H evaluating conditions for rule #1

// If all conditions return TRUE, then the rule is executed
00:00:10.828 policyman_lang.c 127 H condition <true> returns 1
00:00:10.828 policyman_rules.c 460 H executing actions for rule #1

// Variables can be modified during rule execution
00:00:10.828 policyman_lang.c 746 H <boolean_set>: boolean set to 0

Log Analysis – Rule Execution and Client Notification

// Once a rule executes, rule evaluation will end unless there is <continue> action element
00:00:10.828 policyman_lang.c 539 H <continue> encountered - continuing rule evaluation
00:00:10.828 policyman_rules.c 456 H evaluating conditions for rule #2

// If a condition returns FALSE, PM will move to the next rule
00:00:10.828 policyman_uim.c 987 H condition <imsi_mcc_in> with mcc 1 returns 0

// Rule execution can change several configuration elements
00:00:10.829 policyman_rules.c 456 H evaluating conditions for rule #7
00:00:10.829 policyman_rules.c 460 H executing actions for rule #7
00:00:10.829 policyman_ue_mode.c 187 H action <ue_mode> set mode to 1
00:00:10.829 policyman_rat_capability.c 341 H action <rat_capability> set mask to 0xa08
00:00:10.829 policyman_rf.c 424 H action <rf_bands> set:
00:00:10.829 policyman_svc_mode.c 198 H action <svc_mode> set mode to 2

// If configuration items are set the same as last time, PM will not notify clients
00:00:10.829 policyman_cfgitem.c 590 H No configuration items need updating

// If configuration items change, then PM will notify Clients and settings will be propagated
00:07:49.206 policyman_rules.c 460 H executing actions for rule #2
00:07:49.207 policyman_rules.c 480 H clients notified for 1 items
00:07:49.207 cmpmprx.c 00394 =CM= PMPRX: RATs supported = 552, Service scan mode = 2, UE mode = 0
00:07:49.207 cmpmprx.c 00399 =CM= PMPRX: BANDs(MSB) GW = 0x00000000, LTE = 0x000001e0, TDS = 0x00000000
00:07:49.207 cmpmprx.c 00404 =CM= PMPRX: BANDs(LSB) GW = 0x04e80380, LTE = 0x00000000, TDS = 0x00000000

Log Analysis – B39forbid Timer

// Received Indication from RR

17:47:04.050 rr_gprs_debug.c 3704 X OMsg: GERAN_GRR_UPDATE_FORBIDDEN_BANDS_IND sent to Message Router

17:47:04.050 policyman_sglte.c 299 H SGLTE B39: SGLTE policy received GERAN_GRR_UPDATE_FORBIDDEN_BANDS_IND

// LTE Band 39 removed and CM notified

17:47:04.050 policyman_timer.c 398 H Starting timer 2 for policy 2

17:47:04.050 policyman_rf.c 753 H state RF bands GW-32-MSB 0x0, GW-32-LSB 0x280380

17:47:04.050 policyman_rf.c 754 H state RF bands LTE-32-MSB 0x1a0, LTE-32-LSB 0x0

17:47:04.050 policyman_rf.c 755 H state RF bands TDS-32-MSB 0x0, TDS-32-LSB 0x1

17:47:04.050 cmpmprx.c 401 H =CM= PMPRX: BANDs(MSB) GW = 0x00000000, LTE = 0x000001a0, TDS = 0x00000000

17:47:04.050 cmpmprx.c 406 H =CM= PMPRX: BANDs(LSB) GW = 0x00280380, LTE = 0x00000000, TDS = 0x00000001

// After 90 sec, timer expires and LTE band 39 is restored

17:48:34.050 policyman_timer.c 116 H policyman_timer_cb called: timer 2 for policy 2 expired

17:48:34.050 policyman_sglte.c 363 H SGLTE B39: forbid timer expired

17:48:34.050 policyman_sglte.c 372 H SGLTE B39: queueing restore bands command

17:48:34.050 policyman_cfgitem.c 480 H clients notified for 1 items

17:48:34.050 policyman_rf.c 753 H state RF bands GW-32-MSB 0x0, GW-32-LSB 0x280380

17:48:34.050 policyman_rf.c 754 H state RF bands LTE-32-MSB 0x1e0, LTE-32-LSB 0x0

17:48:34.050 policyman_rf.c 755 H state RF bands TDS-32-MSB 0x0, TDS-32-LSB 0x21

17:48:34.050 cmpmprx.c 401 H =CM= PMPRX: BANDs(MSB) GW = 0x00000000, LTE = 0x000001e0, TDS = 0x00000000

17:48:34.050 cmpmprx.c 406 H =CM= PMPRX: BANDs(LSB) GW = 0x00280380, LTE = 0x00000000, TDS = 0x00000021

References

Ref.	Document	
Qualcomm Technologies		
Q1	*Application Note: Software Glossary for Customers*	CL93-V3077-1
Q2	*Application Note: MPSS.DI.2.0 SGLTE Device Configuration*	80-NJ017-11

Questions?

https://support.cdmatech.com