Presentation of OMA Open CM API

Mobile Broadband World 2012
25th September 2012

Contact: Thierry Berisot
Chairman OMA CD-OpenCMAPI SWG
Deutsche Telekom AG

Content

- Introduction
- CM is everywhere
- Challenges - Proper CM is needed
- Rationale for OpenCMAPI
- OMA OpenCMAPI – The Concept
- OMA OpenCMAPI – Features Roadmap
- OMA OpenCMAPI – Timelines
- Key aspects relevant for Mobile Broadband
- Conclusion

- Appendix

Introduction

- OpenCMAPI
  - Device API, defining & exposing Connection Management and associated services Interfaces to applications (CM applications or non CM applications).
  - OpenCMAPI was originally created for Mobile Broadband devices to address limitations due to lack of standards in the Connection Management

- Status:
  - Open Connection Manager API Work Item (WID 028) formally started in Feb 11.
  - OpenCMAPI v1.0 Enabler work completed & Approved as candidate (Jun 12)
  - First implementations of OpenCMAPI v1.0 expected by Q3/Q4 12 (Mobile Broadband Devices)
  - OpenCMAPI v1.1 New Work Item (WID 0267) – Approved July 4th – work started – target end of the 2012

Connection Manager is everywhere

- **Definition: A Connection Manager is an entity or an application that manages different network connections based on user profiles associated with these connections.**

- The Connection Manager is the center of control of the connections and the entity responsible of discovering the networks & potential available connections and establishing the appropriate connection by selecting a network and applying the right parameters.

- Basically, **any device** accessing to internet/data networks through wireless/Mobile networks is using (and is needing) **a Connection Manager**:
  - Laptop: from the OS (ex: Windows Wi-Fi) or by dedicated application
  - Smartphones: OS Wireless and Networks settings or dedicated one (vendors, operators…)
  - M2M: Automotive applications for example

  …

Problem / Opportunity Statement (1/2)

Proper CM is needed

- Challenges:
  - Multiplicity of possible mobile networks available (2G, 3G, 4G) as well as Wi-Fi
  - New/future types of devices/OS such as devices relying on the cloud emerging
  - Applications relying even more on the networks
  - Always on connection and ubiquity expected by customer
  - Need for proper management of connectivity becomes critical.
  - But there was no existing Standard or de facto Standard for Connection manager
  - This lead to negative impacts (costs, TTM, effort) for the industry.
  - Operators, OEM/ODM & users had then to develop and use different and dedicated Connection managers.

- OPEN Connection Management (CM) API was created to solve this problem

Problem / Opportunity Statement (2/2)

Information status is needed

- Services/Applications are limited by lack of information on the connection/device
  - Which network is used (Cellular vs. Wi-Fi / 4G vs. 2G)
  - Roaming situation (price impacts)
  - did the network conditions change?…
  - But there is no Standard existing functionality (within OS or else) to provide such information
  - Applications are behaving the same regardless of situation leading to huge challenges on the networks side
  - Need for generic/standardized device APIs to provide information status to applications and services becomes critical
  - Applications need to have the possibility to adapt / change the behaviour based on connection conditions.

- OPEN Connection Management (CM) API is aiming to address as well this situation by offering one generic Device API providing information status and notifications of events to applications

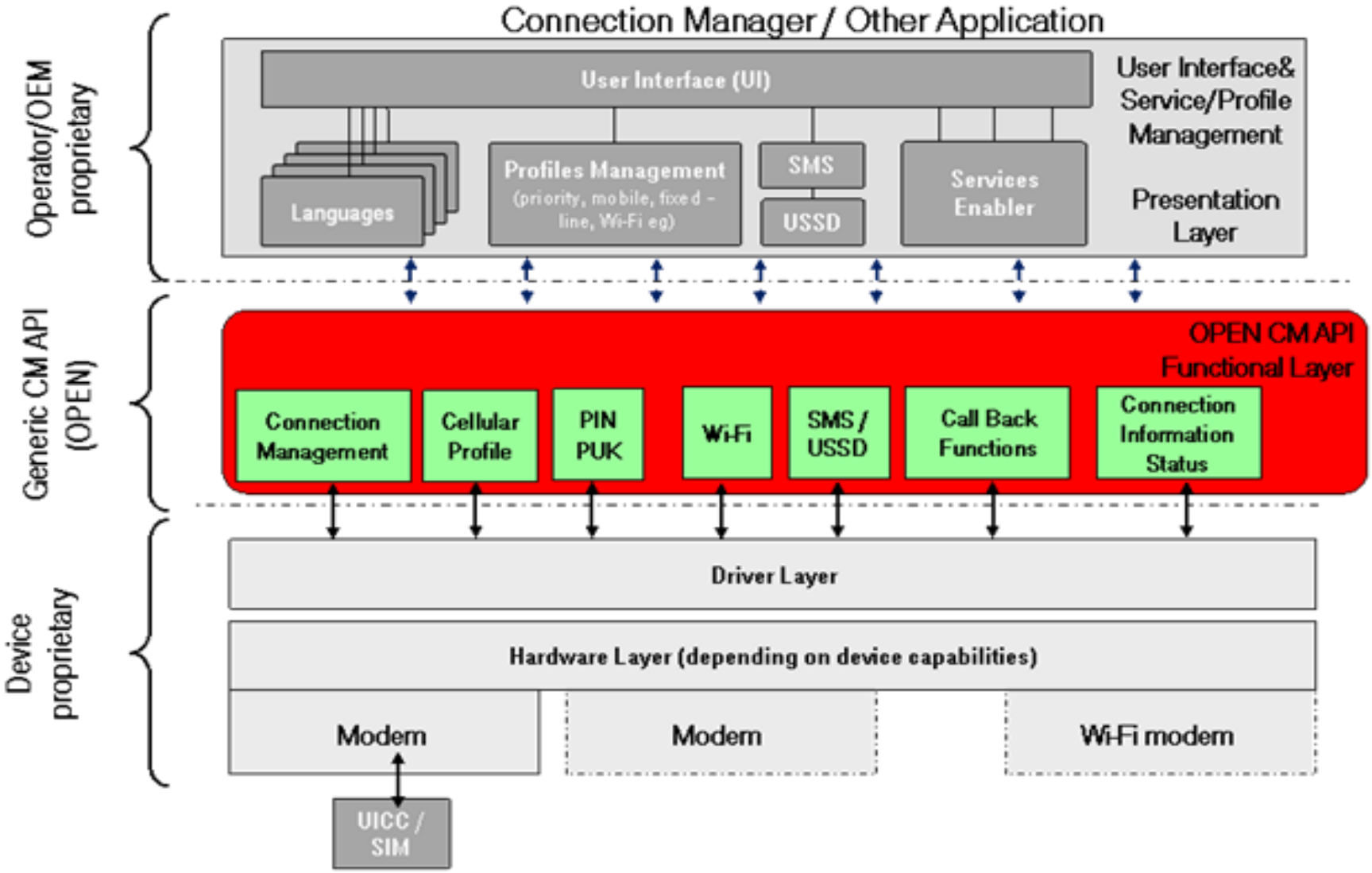
Rationale for OMA Open CMAPI

- Development and standardization of a commonly supported set of connection management APIs:
  - To allow operators, OEM/ODM or any application developers to develop Connection Manager Applications, user experience and services on top of the Connection Manager API independently of the device or the OS used.
  - To provide to any non Connection Manager Applications, dedicated information status on the connection, notifications and complementary services such as SMS, USSD, GNSS… in order to improve the quality of service and the user experience or to develop new innovative services.
  - To make the APIs applicable to different types of devices requiring to access to data connectivity such as Mobile Broadband devices, Wireless routers, M2M / Embedded Modules, Smartphones, Tablets, Cloud Devices.. in order to facilitate development of services for such devices.

OMA OpenCMAPI – The Concept 1/2

- A Connection manager is basically composed of 2 parts:
  - The hardware & connectivity engine part to manage the device with the necessary functions relevant for the user/customer of the connection manager
  - The user experience presented to the customer and composed mainly of the UI, the profiles and the services offered to the user based on actions and answers from the hardware engine part.
- The Main concept from Open CM API is to separate UI/User Experience from connectivity engine.
- Open Connection Manager API relies on hardware and connectivity engine part of the device.

OMA OpenCMAPI – The Concept 2/2



OMA OpenCMAPI Features Roadmap (1/3)

Connectivity & Connections

Feature	OpenCMAPI v 1.0	OpenCMAPI v 1.1*	OpenCMAPI Future Releases*
Connectivity to Cellular	2G, CDMA2000, 3G, LTE		Extension to new networks
Connectivity to WLAN	WLAN including EAP	Hotspot 2.0 Full Support	
Additional Connectivity			WIMAX?
Direct Communication Device to Device		Wi-Fi Direct	LTE Direct
Connection Management	All CM functions incl. profiles mgt	More APIs	Improvement., ANDSF…
Tethering	Supported		Extension?
Mobile IP	Partial for CDMA2000		Extension to WDCMA?

*** Scope currently under consideration**

OMA OpenCMAPI Features Roadmap (2/3)

Security & various devices related functions

Feature	OpenCMAPI v 1.0	OpenCMAPI v 1.1*	OpenCMAPI Future Releases*
UICC	PIN/PUK UICC APIs		Extension – More APIS
Security	Access Control, Authentication		Extension
Power Management	Basic functions	More APIs	More APIs
Device Management (Configuration / Settings)	Partial for CDMA2000		Full support (Objects Mgmt)
Device Management (Provisioning)	Partial for CDMA2000		Full support
Device Management (Fw / Apps update)	Partial for CDMA2000		Full support

*** Scope currently under consideration**

OMA OpenCMAPI Features Roadmap (3/3)

Additional Value Added Services

Feature	OpenCMAPI v 1.0	OpenCMAPI v 1.1*	OpenCMAPI Future Releases*
Information Status	Device / Connection	More APIs	More APIs
Notifications/Event	Callbacks Device / Connection	More APIs Event Mgt	More APIs
Complementary services (SMS, USSD…)	SMS, USSD (Send, Receive)		Others?
Location / GNSS	GNSS API		Extensions of GNSS APIs
Statistics	Connection Stats		Extension?
Contact Management		Basic	Extension?
Push Service handling	Basic		Extension?

*** Scope currently under consideration**

OMA OpenCMAPI – Potential Timelines*/**

Releases Development & Implementation per device type

	2011				2012				2013			
	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4	Q1	Q2	Q3	Q4
OMA OpenCMAPI Releases	v 1.0				v 1.1*				likely to be v2.0*			
1st Commercial Implementations per device type**												
MBB (USB)												
Laptops with embedded module												
Wireless Routers												
M2M												
Smartphones / Tablets									 			

* Beyond 1.0, timelines subject to change

** Depending on device vendors' roadmap

OMA Open CMAPI

Important aspects - Relevant for Mobile Broadband

- Key OpenCMAPI aspects relevant for Mobile Broadband:
  - Supporting management of different types of network (2G, 3G CDMA2000, 4G, WLAN)
  - Support all features relevant for Mobile Broadband context
  - Ready for Wi-Fi offload
  - OS independent (potential to support any OS)
  - Modem independent – develop once
  - User Interface independent – Customization of User experience easy
  - Supporting Multi-Instances (several applications/services can use it in parallel if necessary)
  - Potential to support multiple connections in parallel (dependency to device/modem capabilities)
  - Enabling development of services/application

- OpenCMAPI: the "HMTL 5 of the connection management"

Conclusion

- There is a strong need to have Device APIs developed and supported by the Devices (or the OS) to allow development of better applications and innovative services.
- These APIs should be standardized to enable the whole ecosystem/industry to focus on what matter the most (differentiation, user experience, services…).
- OMA OpenCMAPI is one of the first OMA Enablers pointing and aiming in this direction and has the merit to be now existing.
- The industry should support and encourage further development of such initiatives/Device APIs.

Thank you for your time

Contact & Links

Contact: thierry.berisot@telekom.de

Open CM API v1.0:

http://www.openmobilealliance.org/Technical/release_program/OpenCMAPI_v1_0.aspx

Open CM API v1.1 WID:

http://member.openmobilealliance.org/ftp/Public_documents/TP/Permanent_documents/OMA-WID_0267-OpenCMAPI-V1_1-20120704-A.zip

Live demos of OpenCMAPI at OMA Demo day:

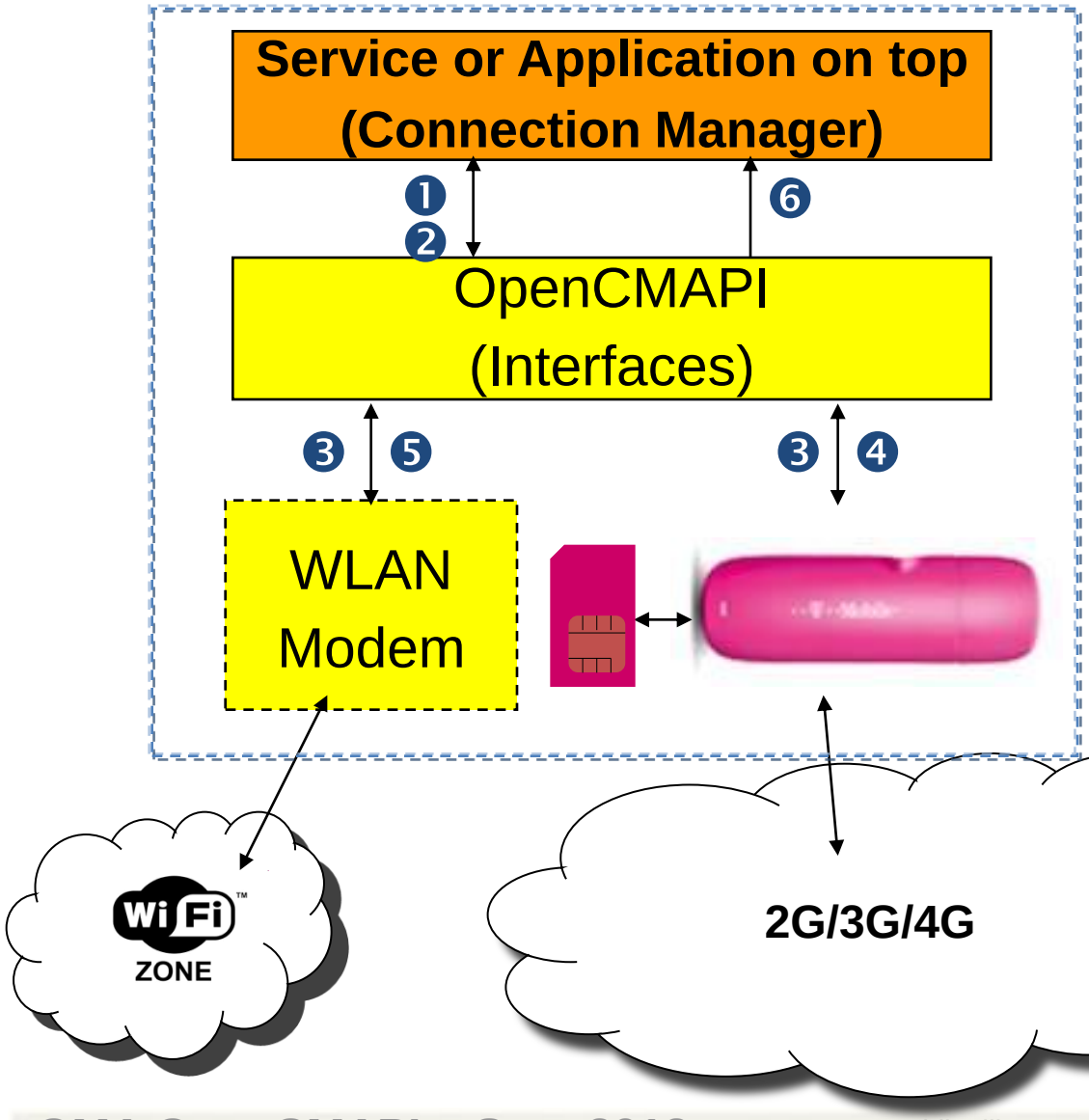
http://www.openmobilealliance.org/comms/pages/OMA_brooklyn_demo_day.htm

Appendix

Description of existing OpenCMAPI functions

OpenCMAPI – Step 1

Registration, Callbacks & Device Discovery



❶ Open API

❷ Registration Call backs (various)

❸ Discovery Devices

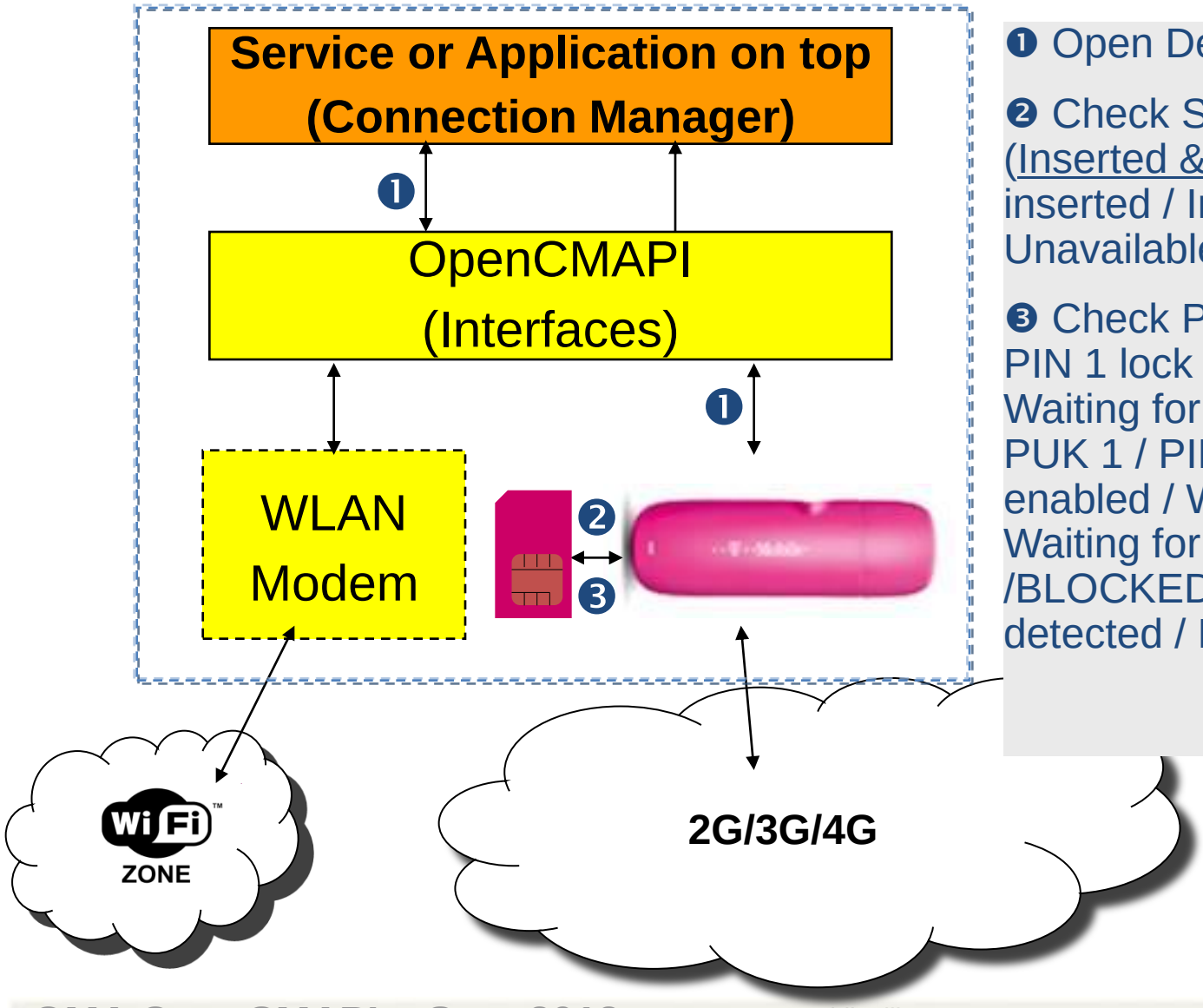
❹ Cellular Modem (Device 1)

❺ WLAN Modem (Device 2)

❻Call back Devices available (1 & 2)

OpenCMAPI – Step 2

Check SIM



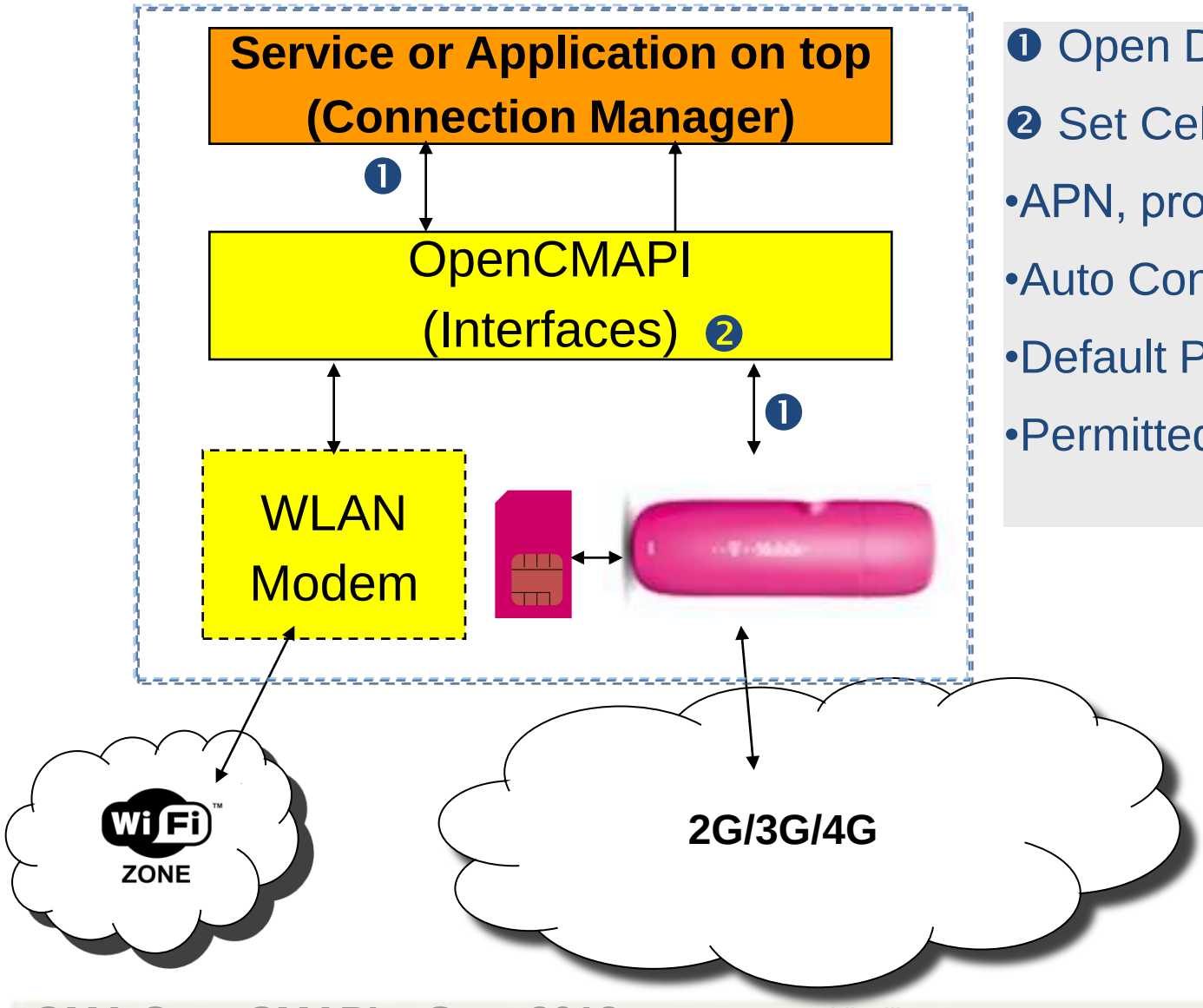
❶ Open Device 1

❷ Check SIM card status (Inserted & Available / Not inserted / Invalid / Unavailable / Unknown)

❸ Check PIN status (Ready / PIN 1 lock feature enabled / Waiting for PIN 1 / Waiting for PUK 1 / PIN 2 lock feature enabled / Waiting for PIN 2 / Waiting for PUK 2 /BLOCKED / No SIM detected / Error)

OpenCMAPI – Step 3a

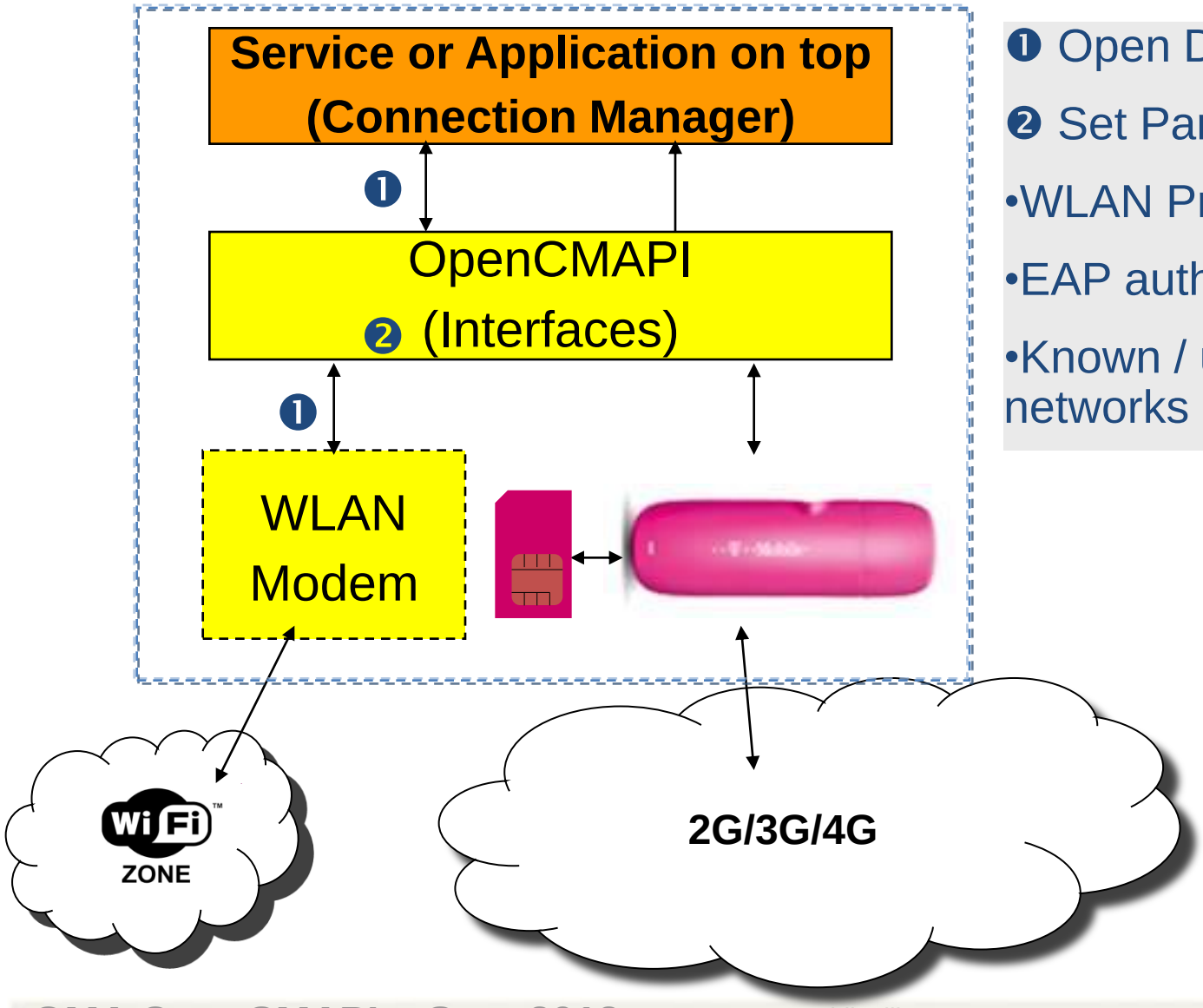
Set profile for cellular



1. Open Device 1
2. Set Cellular Profile:
   - APN, profile name…
   - Auto Connect Mode
   - Default Profile
   - Permitted Bearers

OpenCMAPI – Step 3b

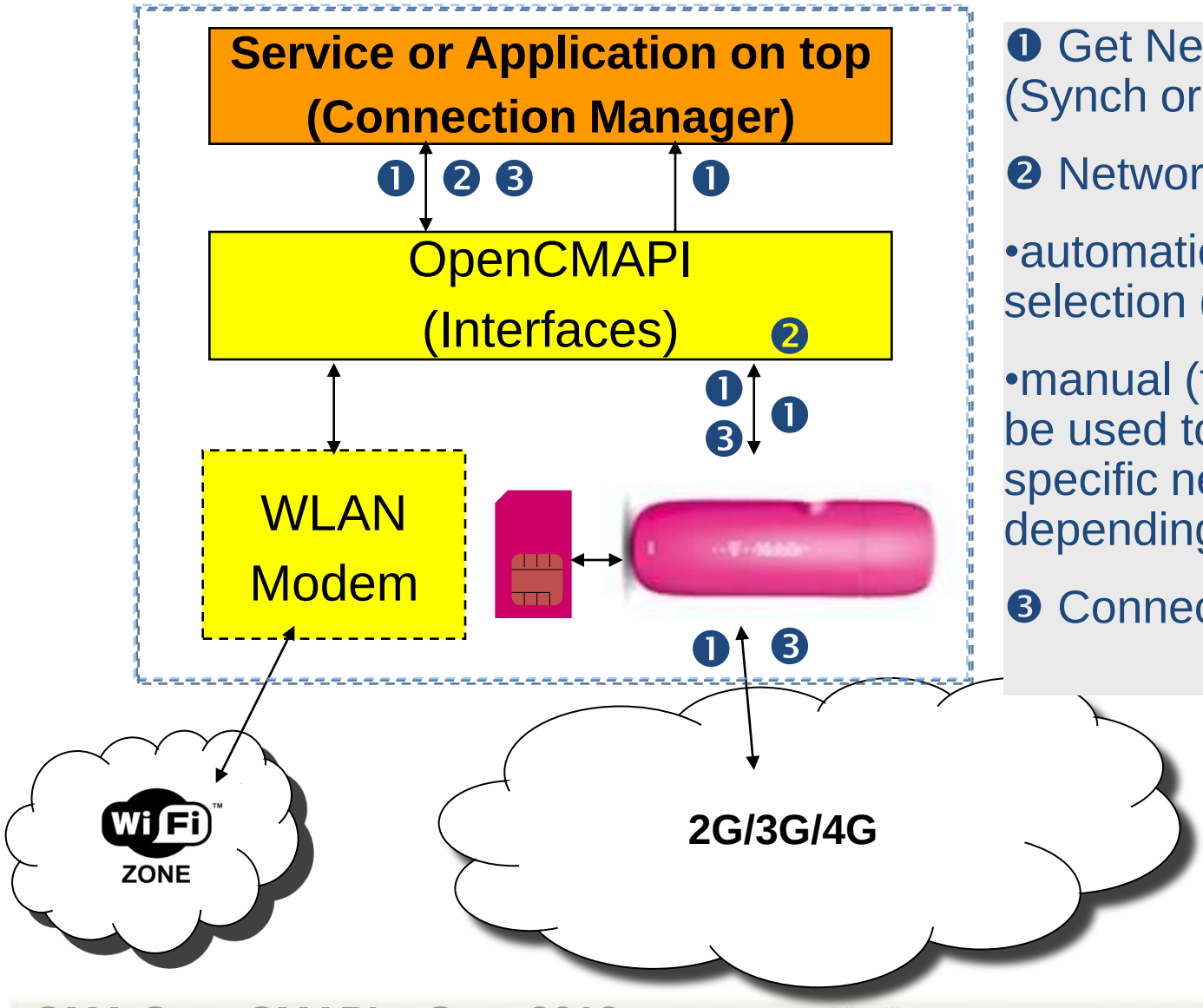
Set parameters for WLAN



1. Open Device 2
2. Set Parameters
   - WLAN Profile
   - EAP authentication
   - Known / unknown networks list

OpenCMAPI – Step 4a

Connection to Cellular



❶ Get Network available (Synch or Asynch.)

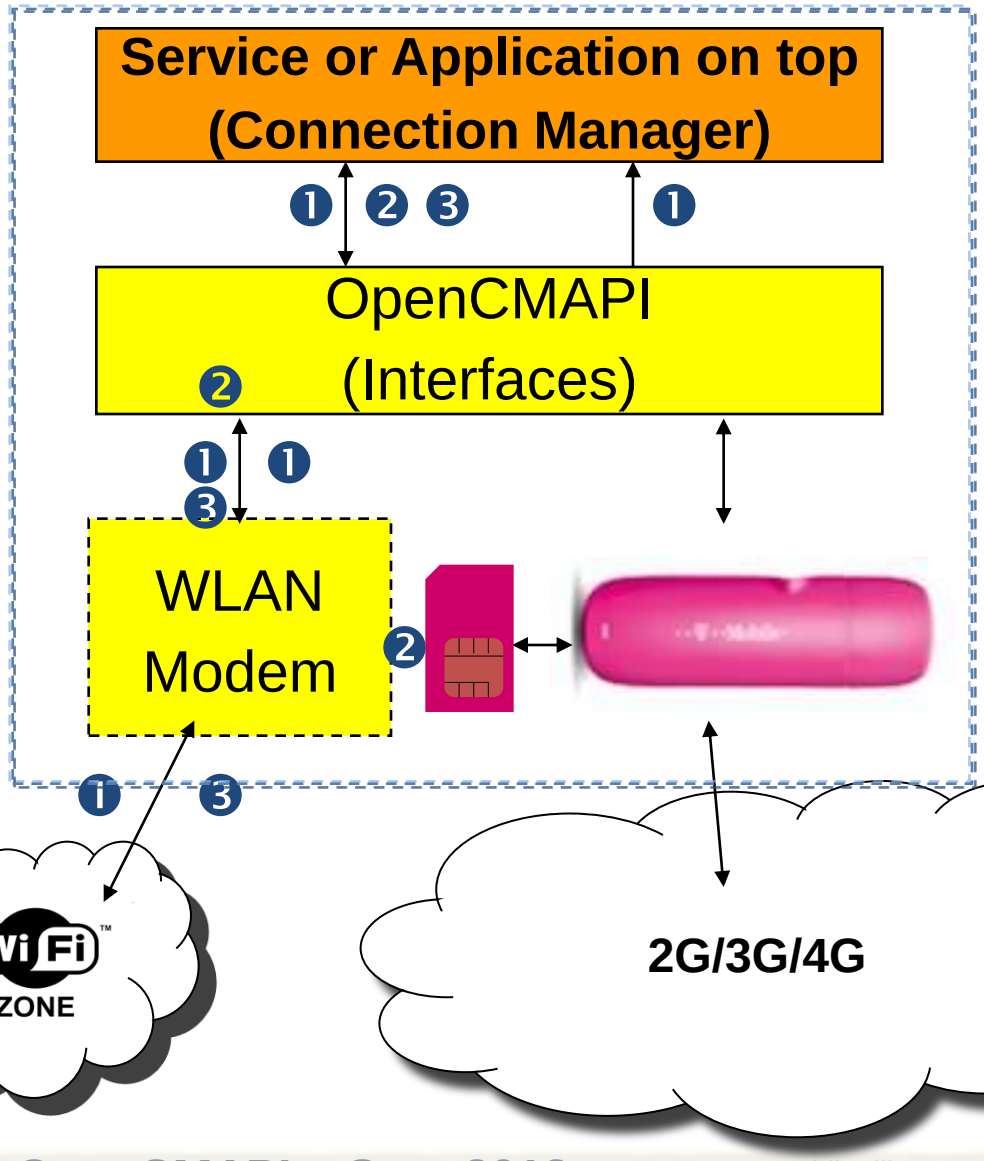
❷ Network Selection:

- automatic network selection (default)
- manual (for M2M, could be used to connect to specific network(s) depending of the need)

❸ Connect / Disconnect

OpenCMAPI – Step 4b

Connection to WLAN



❶ Scan Network available (Synch or Asynch.) or Search for specific Network (Asynch.)

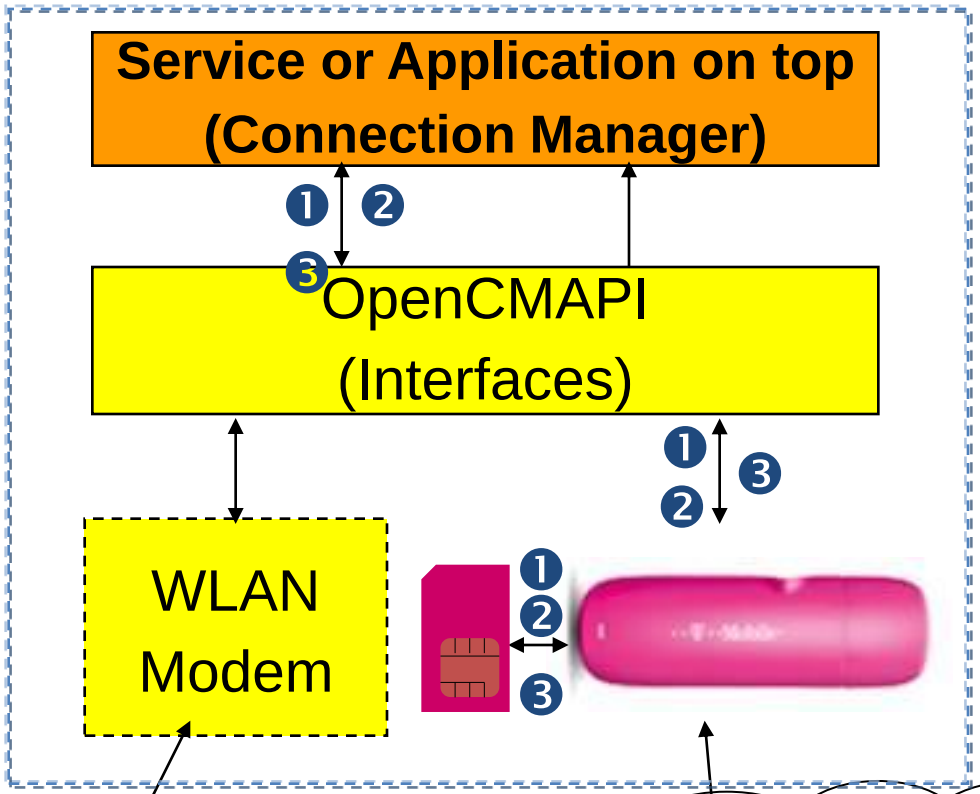
❷ Select Network – could be a specific Network (for M2M) - could be interacting with smart card:

- Preferred WLAN networks list
- EAP Authentication

❸ Connect or Connect known networks/ Disconnect

OpenCMAPI – Step 5

UICC functions



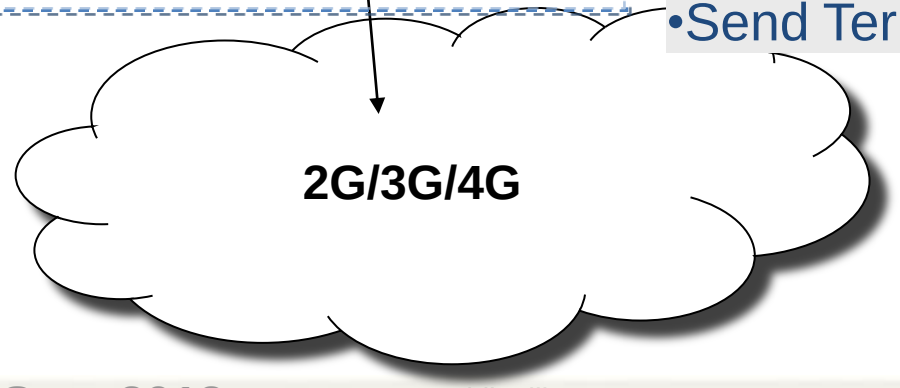
❶ Access Control check

❷ PIN/PUK management:

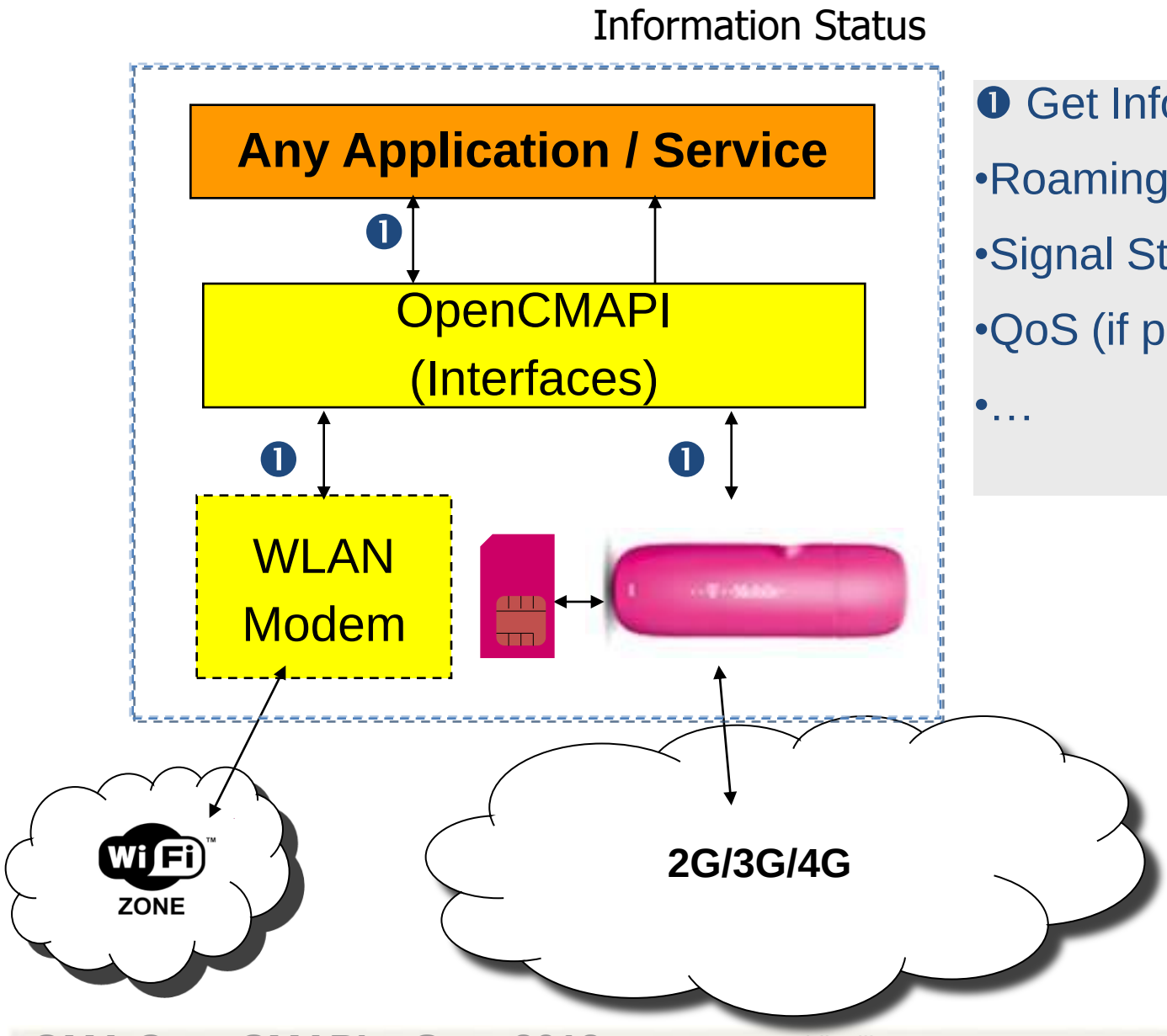
- Get NAA available,
- Enable, Disable, Verify, Unblock, Change PIN

❸ UICC Management:

- Get/Set Terminal Profile
- Send ToolKit Envelope Command
- Send Terminal Response



OpenCMAPI – Step 6

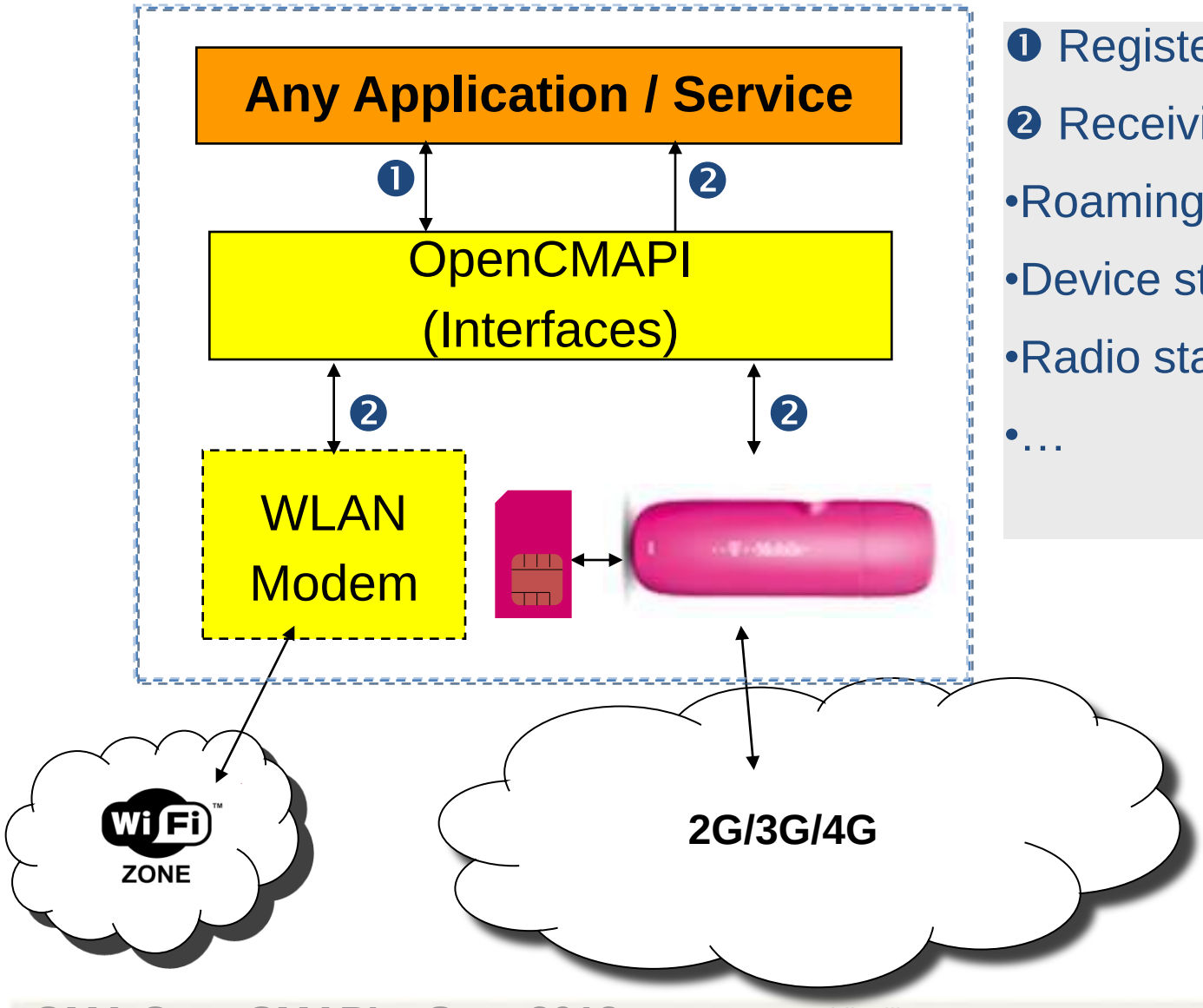


❶ Get Information Status:

- Roaming
- Signal Strength
- QoS (if provided)
- …

OpenCMAPI – Step 7

Notifications/event – call-backs



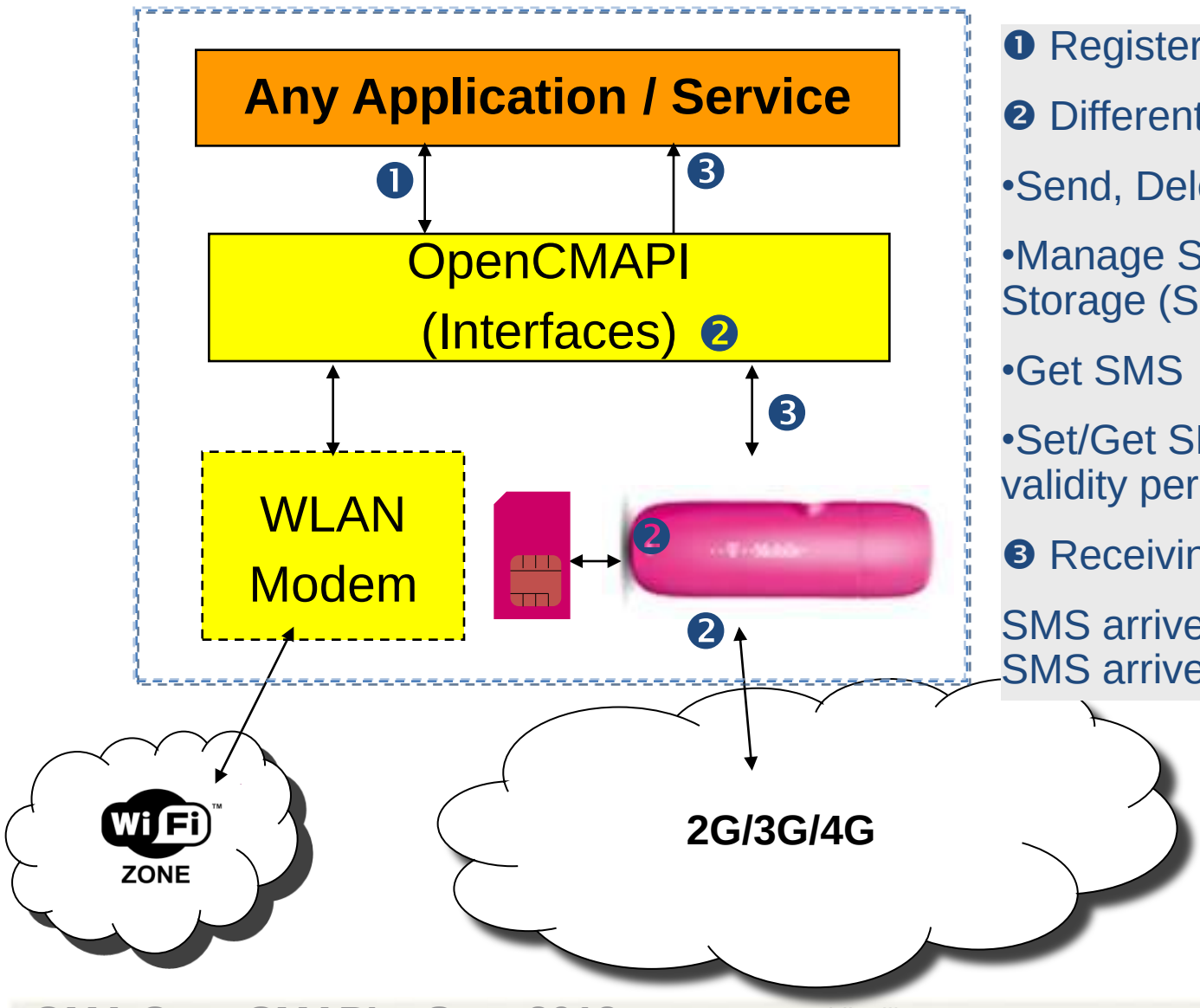
❶ Register to call-backs

❷ Receiving call-backs:

- Roaming status changed
- Device status changed
- Radio state changed
- …

OpenCMAPI – Step 8

SMS



❶ Register to SMS call-backs

❷ Different SMS functions:

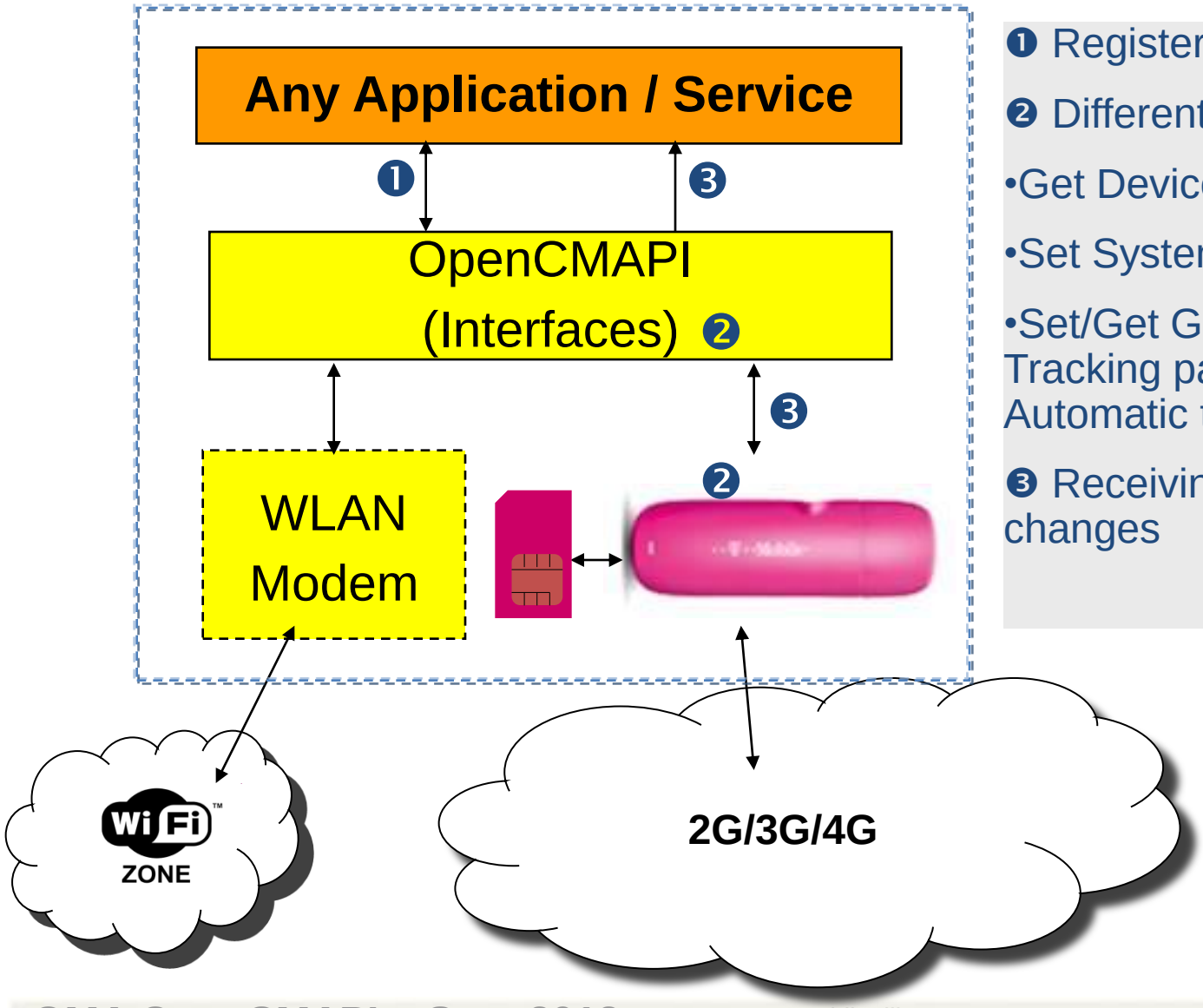
- Send, Delete SMS
- Manage SMS including Storage (SIM or Device)
- Get SMS
- Set/Get SMSC address, validity period, delivery report

❸ Receiving Call-backs

SMS arrived indication or SMS arrived details

OpenCMAPI – Step 9

GNSS functions



❶ Register to GNSS callback

❷ Different GNSS functions:

- Get Device position
- Set System Time
- Set/Get GNSS State, AGPS, Tracking parameters, Automatic tracking

❸ Receiving Callback GNSS changes