# Trusted Computing: Introduction & Applications

## Lecture 6: The evolution of mobile businesses and the TCG MPWG Specifications


Dr. Andreas U. Schmidt

Fraunhofer Institute for Secure Information Technology SIT,
Darmstadt, Germany

# Literature


1. TCG Mobile Phone Working Group Use Case Scenarios – v 2.7. https://www.trustedcomputinggroup.org/groups/mobile/
2. TCG Mobile Reference Architecture Specification version 1.0, Revision 1.12 June 2007
3. TCG Mobile Trusted Module. Specification version 1.0, Revision 1. 12 June 2007
4. Addressing Mobile Security through Industry Standards 3GSM World Congress. https://www.trustedcomputinggroup.org/news/presentations/3GSM_presentation.pdf
5. Mobile Trusted Module Briefing CTIA Wireless IT and Entertainment 2006. https://www.trustedcomputinggroup.org/groups/mobile/CTIA_Final_Seminar_Presentation.pdf
6. Michael Kasper. Diploma Thesis
7. S. Berger, R. Caceres, K. A. Goldman, R. Perez, R. Sailer, and L. van Doorn. vTPM: Virtualizing the Trusted Platform Module. Technical report, IBM T. J. Watson Research Center, Yorktown Heights, 2006.

# Prelude: The changing role of Mobile Network Operators

**What analysts say**

- MNOs are advised to consider new business models from Web 2.0
- Mobile Entertainment is one possible opportunity, mobile network games are a new area enabled by 3G networks and convergent technologies
- It is necessary to explain to the customer these new convergent services
- Live TV and Video downloads are attractive services to commit customers to a certain service provider
- MNOs are privileged players as they already have a stable and huge customer base
- A great number of customers whishes just one service provider to deal with, offering the various services and coordinating the payment making the various service providers transparent with respect to the user recognition
- Accounting and charging competence is going to be a key ability

KPMG: Consumers and Convergence - Challenges and opportunities in meeting next generation customer need. 2006. http://www.kpmg.de/about/press_office/13611.htm

# Prelude: The changing role of Mobile Network Operators

**Threats to MNOs**

- Yahoo!, Google, etc. attack the MNO customer base and are prepare to disrupt traditional established business models by using Web 2.0 methods
- Google/Vodafone deal to establish search on mobiles is one example
- MNO think in terms of `how to construct a chargeable service‘, whereas Internet companies experiment in test markets.
- A race for innovation where MNOs are not in the pole position
- Combines with other threats, e.g., pressure on international roaming, and voice, e-mail, and IM termination revenues, SMS is at risk
- Saturated mature markets will force price competition
- In the next three to five years, MNOs‘ profit margins may drop by 25%

Annual GartnerWireless and Mobile Summit, March 2006, Detroit, MI, With some quotations of Nick Jones, Gartner vice president and distinguished analyst

# Prelude: The changing role of Mobile Network Operators

**Mobile Trends**

- 20 million **mobile e-mail** users by 2006, 100 million by 2009
- Mobile e-mail will become a commodity by 2010 pervading every part of businesses, i.e., trickle down from the executive level
- Microsoft will be the most important player in mobile e-mail
- **Mobile collaboration** is among the top four of CIOs‘ current priorities, according to a Gartner global survey emerges in response to the need for `soft‘ collaborative tasks and increasing staff mobility
- With simple automation tasks completed (CRM/ERM), RO(IT)I diminishes,
  mobile coll. offers new opportunities
- Augment the *effectiveness* of individual knowledge workers
- Geographically distributed workers are highly valuable and sophisticated enough to be aware of a wide range of consumer mobile technologies
- Employees familiar with Web based collaboration tools (Blogs, Wikis) will quickly accept their mobile analogs

# Prelude: The changing role of Mobile Network Operators

- Mobility becomes increasingly integrated in corporate strategies
- As **mobility becomes mainstream**, companies fell strategic decisions, such as who will be their mobile partner for the first time
- Mobile devices with integrated multi-access technologies pose a **security risk to corporate data**
- „Banning consumer technology from employee-owned devices is unrealistic, unverifiable, and naive“
- Prevention would stifle innovation
- Companies are to educate employees and adopt sound policies
- **Smartphone** vendors will be challenged to offer viable devices that address corporate deployment requirements
- Gartner predicts annual growth rates of 49% for 2005-2009
- One in three mobile devices will be a smartphone by 2009

# Prelude: The changing role of Mobile Network Operators

**Mobile business 2.0** is the next emerging trend

- Exact specifics not yet clear but guiding principles emerge
- old m-commerce failed because of bad value proposition and naive transplantation of Web technology to mobiles (e.g. WAP), LBS were poor
- Many Web 2.0 principles will apply to the mobile domain, including cultural behavior, business models, and interactions
- Some are different: Web users start activities by browsing, while mobile users will not want to browse on tiny screens
- Mobility 2.0 will therefore be `ambient' business driven, e.g., by consumers exploring their environment to find relevant propositions
- Data will be selectively pushed to users, matching context and specific needs, interests, mood, location, recent behaviour
- fleeting experiences arise `buy from this [physical] shop within the next ten minutes and you will get a discount'

# Device evolution

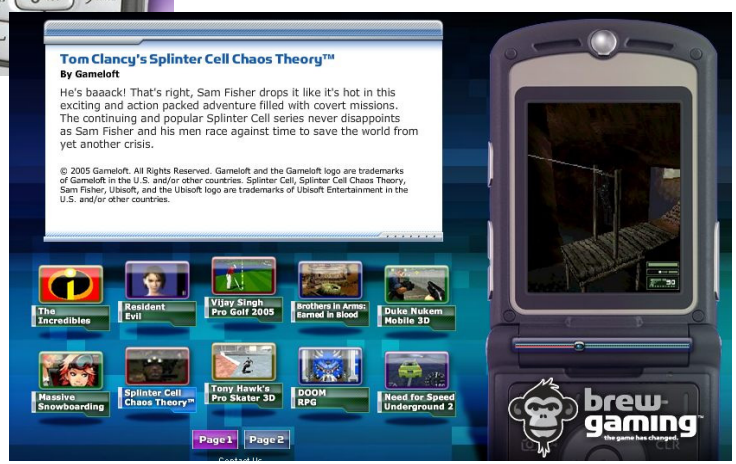
- **Integration of multimedia capabilities in mobile device**
  - Falanx offers OpenGL compatible grafic cards (now an ARM subsidary)
  - **Trend towards open source OS for mobile devices**
    - Motorola, NEC, NTT DoCoMo, Panasonic, Samsung, and Vodafone started a software platform
    - ARM, France Telekom, Montavista, and PalmSource joined in the 'Linux Phone Standards Forum' (LiPS)
    - Motorola offers opensource.motorola.com, supporting Java and Linux developer
  - **Integration of other net. access technology, e.g., Wi-Bro and WiMax**
    - Devices will be active in more networks in parallel than today
    - In device barriers are required to protect the various networks from effects caused by other networks
    - Unlicensed Mobile Access (UMA) as an interim solution

# Mobile Widgets

- Mobile Widgets are attractive because they increase usability of small-screen devices.
- The layered structure of transparent desktop items saves screen space.
- A lot of technology providers have recently emerged.
- Qualcomm's BREW has a traditional, proprietary approach, and marks a high state-of-the-art
- AJAX-based platforms
  - Nokia's WidSets, Openwave Mobile Widgets, Opera platform, Mojax Moblets, Bluepulse, mobile Open Source by Funambol, Nokia-gate5 smart2go, Vista SideSjow

# Qualcomm‘s BREW


- ´**Qualcomm’s BREW mobile application platform**
  - advanced binary API/SDK for a wide variety of mobile devices.
  - Pre-delivered to the phone by operator
  - BREW technology targets MNO as customers
  - Large amount of contracted application providers
  - Users download apps/widgets from one-stop-shop




- Large amount of contracted application providers
- Brew mainly used in US and Caribbean but enters Europe as well
- Applications range from simple to sophisticated games, LBS, to full-fledged business apps

# BREW components

- advanced platforms enabling immersive, full-color 3D gaming experiences with multiplayer functionality and high quality sound.
- LBS/enterprise applications
- deliveryOne content delivery client/server system
- User interface uiOne

- Device personalisation




- Operator branding



# BREW Business model

- Developers commercialise apps sharing revenues with MNOs and Qualcomm.
- Developers submit application for testing.
- MNO accepts applications to offer to consumers after successful testing.
- Consumers select apps to purchase and download them to their BREW devices.
- In place within the operators' infrastructures, the QUALCOMM application download and billing systems let:
  - Consumers download and pay for applications
  - Developers track the adoption of applications
  - Revenues be distributed to developers
- Before any BREW application can be made available for purchase, the developer must have an established business relationship with QUALCOMM. Authenticated BREW developers are provided with the tools and interfaces to price applications with operator services.

# Mobile Widgets

**Based on Web 2.0 standards, like mobile versions of AJAX, RSS, …**

- **Nokia's WidSets**
- **Openwave** Mobile Widgets
- **Opera Platform** – free mobile AJAX authoring environment
- **Mobidgets by mobease** (France),
- **Bling Software's** AJAX engine.
- **Quick mobile**
- **mojax** by mFoundry Mobile AJAX Application Frameworks
- **Bluepulse (AUS, CA**) open platform (development tools are free) supporting Java MIDP1, MIDP2, and Symbian
- **Mobile Open Source** by Italian Start-up **Funambol** (SyncML reference implementation, push services)

# Mojax Widget coding

Like AJAX applications, mojax applications use a combination of XML, Javascript, and CSS. The XML you use can be the specialized MJX tags developed for mojax, or Javascript.

**Layout** (MJX): When you create a mojax application, you begin by laying out the elements that will appear on the screen. You can specify layout in XML using MJX (mojax XML). Advanced JAVA developers can specify layout using Javascript .

**Example**: A simple mojax (MJX) file that renders "Hello World!"

```
<moblet default="main">
  <screen id="main" layout="vertical">
    <textbox>Hello World!</textbox>
  </screen>
</moblet>
```

**Styling** (CSS) Once you have specified mojax screen elements, you give them visual characteristics (color, padding, and so on) using CSS  Cascading Style Sheets
Example: Adding styles to the Hello World! application using CSS

```
<moblet default="main">
  <style>
  screen {
    color: #FC0000;
    background-color: #CCCCCC;
    font-size: medium;  }
  </style>
  <screen id="main" layout="vertical">
    <textbox>Hello World!</textbox>
  </screen>
</moblet>
```

**Scripting (Javascript)**

Finally, you specify how screen elements respond to user activity. You can script behavior for pre-defined events like onClick or onUpKey.

You specify application behavior using mojax Script, which is identical to Javascript except for object types. The DOM objects you can manipulate via scripting are specific to mojax applications, and are not based on the HTML document format. For a complete guide to mojax Script, see Script Language. For a reference to all mojax Script Objects, see Script Objects.

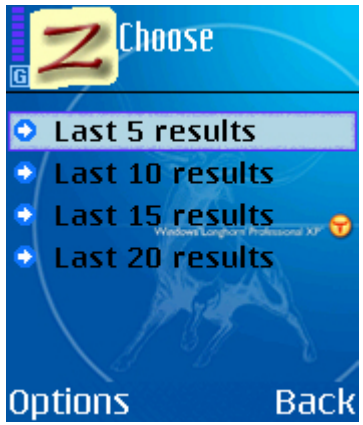
Example: Adding an "exit()" script call to the "onLeftSoftkey" event handler

```
<moblet default="main">
  <style>
  screen {
    color: #FC0000;
    background-color: #CCCCCC;
    font-size: medium;
  }
  </style>
  <screen id="main" layout="vertical" onLeftSoftkey="exit()">
    <textbox>Hello World!</textbox>
  </screen>
</moblet>
```

# Bluepulse widgets

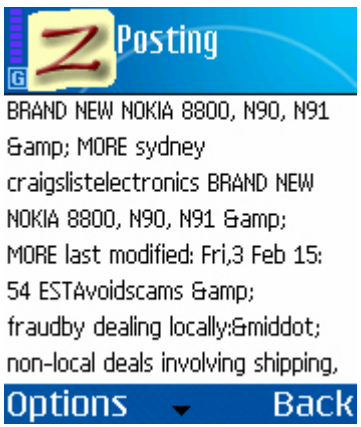

Select ZapTXT after you start bluepulse


Select ZapTask for which you want to see past alertresults




Select how many alerts/results you would like to see


Select the title of the post you're interested in




Read the post

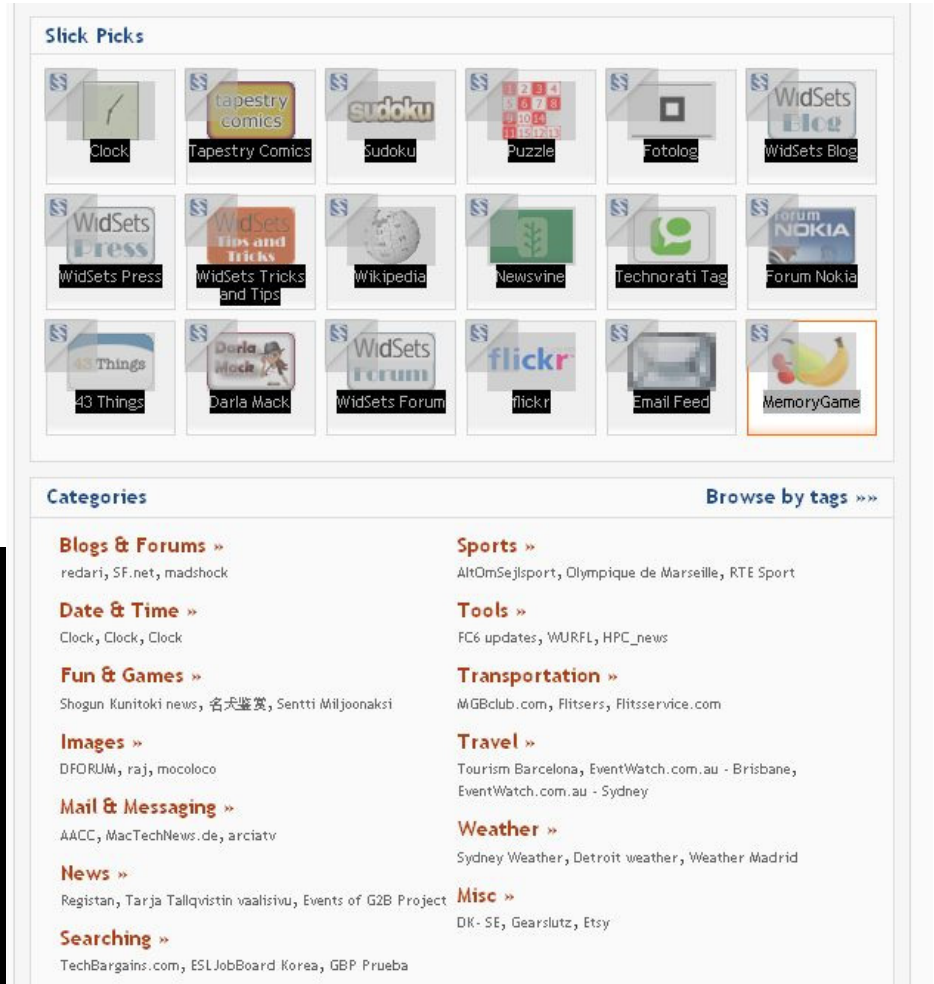

Send email to the author of the post


Type the message and send email

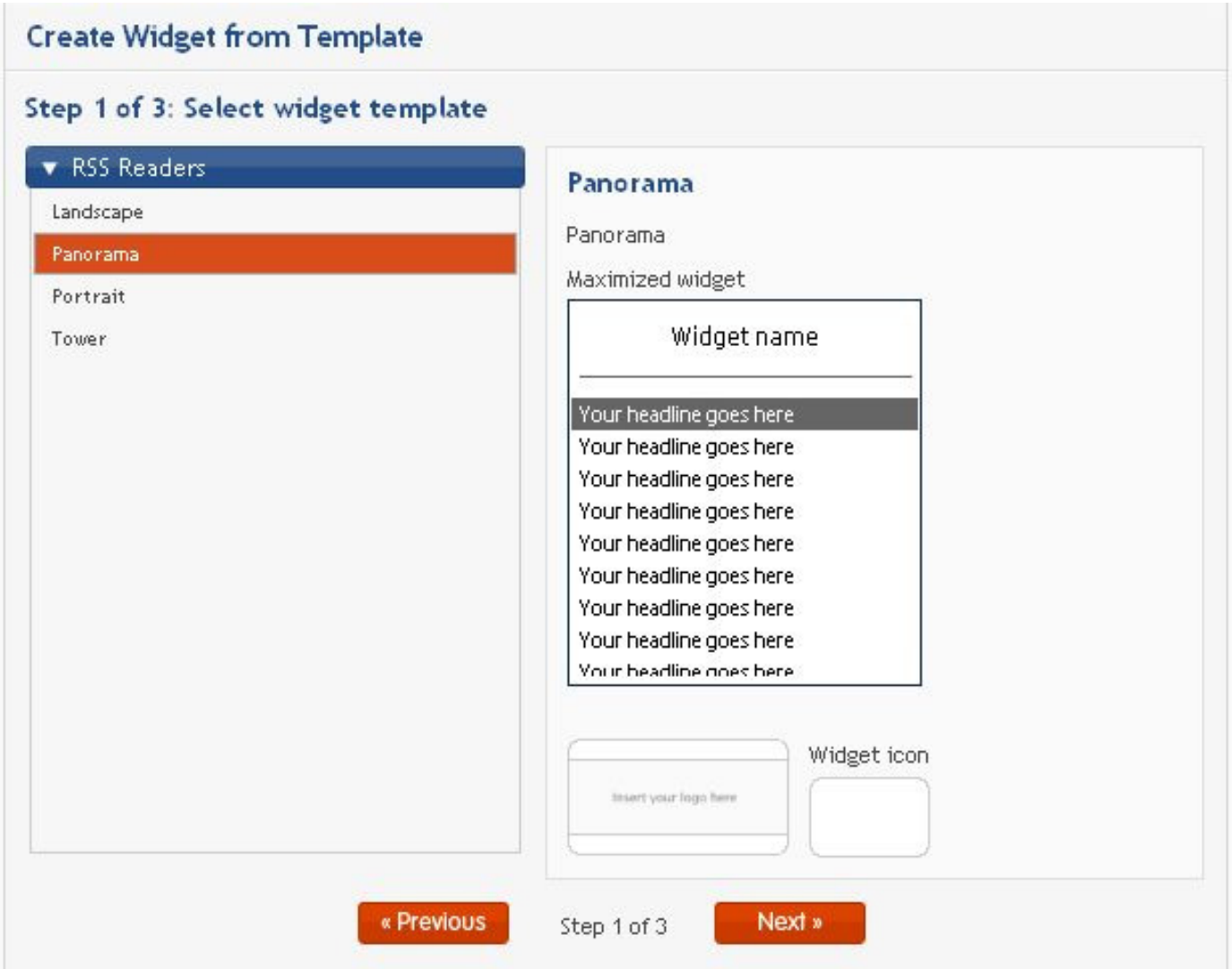
**This is an example of a ZapTask – ZapText is an RSS overlay search enginge - that was monitoring a Craigslist RSS**

# Nokia's WidSets

- From Nokia's emerging business unit
- platform openly available
- entirely based on RSS + JavaScript
- The WidSets website provides easy-to-use templates to create own Widgets
- WidSets (potential) business idea:
  - WidSets.com is a unique Widget management platform providing full control user-generated widgets. Widgets can only be downloaded from widsets.com and only be created resp. published there.



# Nokia‘s WidSets - creation



# WidSets Widget code

## Config

Simple configuration parameters.

```
1.<config>
2.  <!-- These are used to define initial widget
      parameters. -->
3.  <parameter name="widgetname">Example
      widget</parameter>
4.  <parameter
      name="feedurlrss">http://www.yourfeed.com/ind
      ex.xml</parameter>
5.</config>
```

## Skin

The look and feel piece

```
1.<skin name="Example widget" version="0.1">
2.  <!-- Styles used by the widget -->
3.  <styles>
4.    [bg]
5.      background: grid background 15 11 12 11
6.    [feedIcon]
7.      align: left vcenter
8.    [feedTitle]
9.      font-1: small
10.      color-1: #000000
11.      align: left vcenter
```

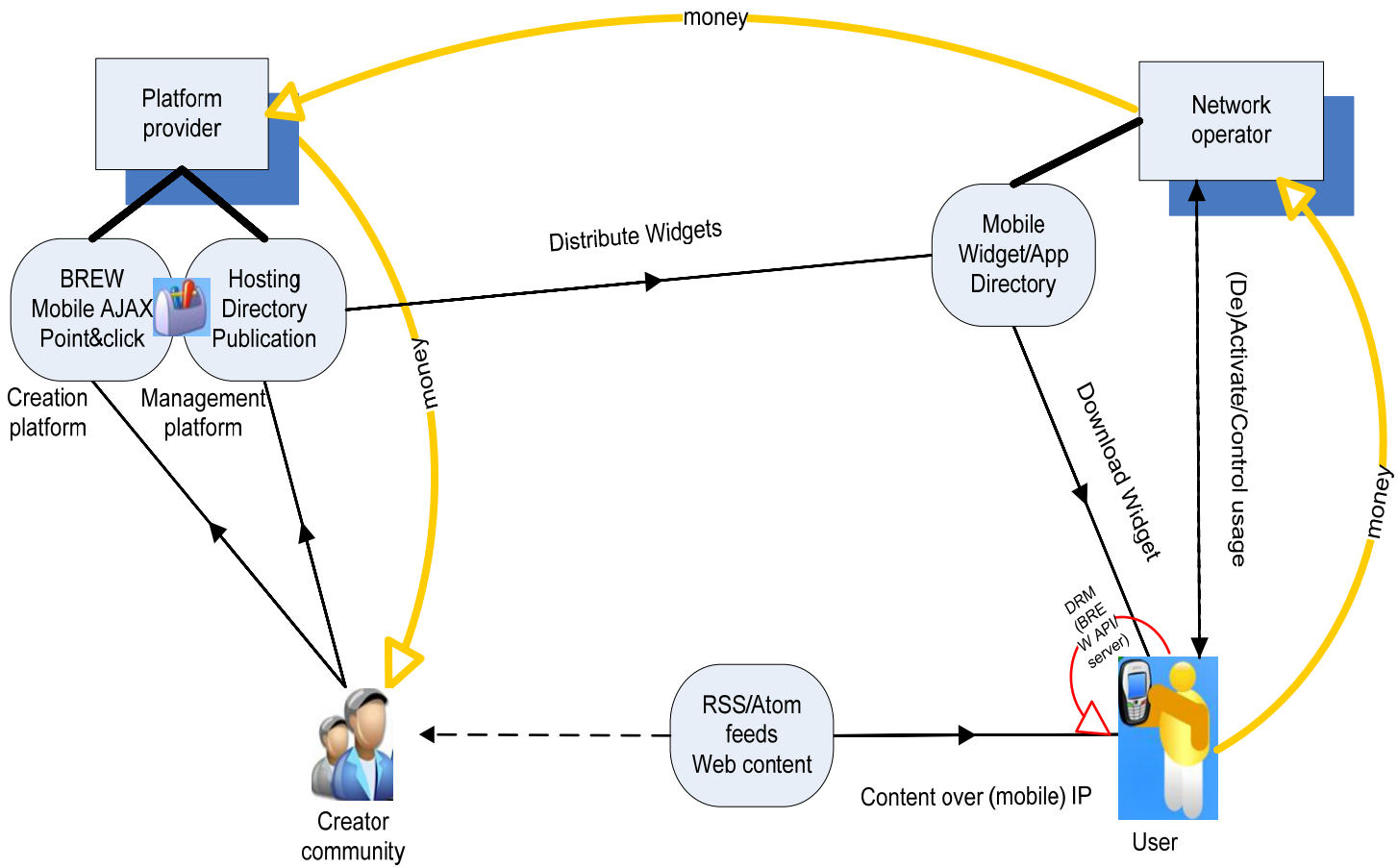
## Widget

The widget file itself:

```
1.<widget name="Example widget" version="0.1">
2.
3.  <info>
4.    <creator>
5.      <user>username</user>
6.      <date>7.6.2006 12:00</date>
7.    </creator>
8.
9.  <!-- Configuration of servicehandlers which the
      widget needs to operate -->
10.  <services>
11.    <service type="syndication" id="feed1">
12.          <reference from="feedurlrss"
      to="feedurl"/>
13.        </service>
14.  </services>
```

# Mobile Widget revenue stream

Mobile Widget/Application creation, distribution, delivery, charging and revenue sharing between creator, platform provider, MNO, and user



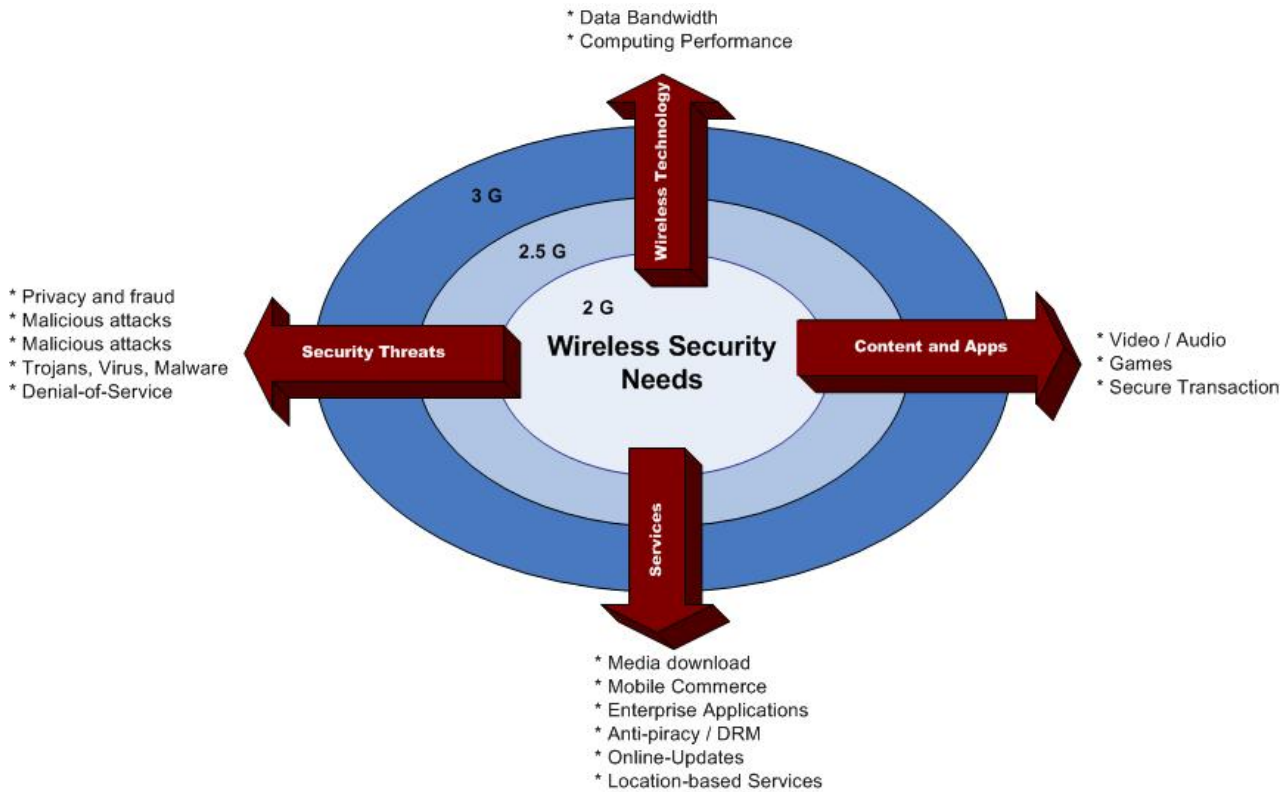
# MNOs role in Mobile Widget Business

- The MNO is only weakly coupled in the AC loop – mainly download
- Commercial value is concentrated in widgets not content
- MNO depends on
  - third-party platform provider
  - Closed technology for widget creation and operation
- When mobile widget/Web 2.0 technology becomes standard and open, this presupposition breaks down

- MNOs should (short- to medium-term)
  - syndicate and re-distribute ***content*** for mobile widgets/apps
  - exert **access control** over this content

# Wireless Security Needs for new mobile businesses

**„Location, combined with personal presence, creates transient communities which offers an opportunity for better use of time or marketing“**

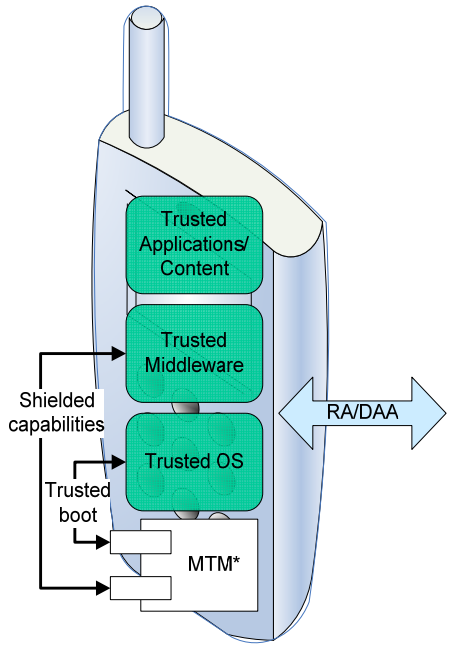


# Trust in mobile service access

- **Mobile access to applications & content is becoming network-agnostic**
  Customers attracted by attractive applications & content
  Diversity of technologies (2G, 3G, WLAN, WiMAX, MobileIP)
  Customers interested in optimizing price/performance ratio
- **Mobile devices are becoming very smart, multi-purpose devices**
  More than voice comm., both consuming and providing applications, data and media Network access is a commodity, customers expect additional features. Next step for MNOs (business models): providing customised/customisable services
- **Novel requirements for trust across domains – even technological boundaries**
- **Trusted computing can become the enabler for service provisioning**
  - Enables network- and device-agnostic trust relations on application-level
    Uniform trusted platform for service provisioning
- **Credentials from various domains of trust, carried, managed and transmitted by TC-enabled devices can yield *transitive trust relationships***

# TCG MPWG

- **Mobile Phone Work Group (TCG MPWG)**
  - Subgroup of the TCG
  - Specifies the TCG MPWG Reference Architecture and TCG MPWG Specification
  - Members: Nokia Corp., Wave Systems, Infineon,Gemalto, Samsung Electronics Co., France Telecom SA or Ericsson, HP Labs, Nokia Siemens Networks, …
- Document History
  - MPWG Use cases – 09/2005
  - MPWG Requirements – 03/2006 (internal doc)
  - MPWG Reference Architecture – 06/2007
  - MTM Specification – 06/2007
  - … Use case Analysis



# Specification Backgrounds

- Baseline was TCG TPM v1.2 specifications
  - Optimized for PC platform
- Standard HW, Standard Boot Cycle (BIOS, EFI), Standard OS
  - Rich feature set
- Basically unlimited by size/power constraints
- TCG MPWG set out with two tasks
  - Enable implementation in a mobile phone
  - Enable implementation of published use cases
- Enabling implementation in mobile phones
  - Heterogenous hardware and OS environment
  - Proprietary boot cycle
  - Suitable for implementation inside a SoC
  - Enable implementation as SW in a separate trusted on-chip execution environment
- Enable implementation of published use cases
  - Local Verification
  - e.g. Platform Integrity, IMEI protection

# MPWG Use cases

- **User Data Protection and Privacy** - Enable the protection of user's personal information, such as identity and address books, from access or copying by unauthorized parties.
- **Platform Integrity** – Ensure device operation occurs with only authorized operating system(s) and hardware.
- **Device Authentication** – Ensure that 1) device authentication may be used to assist in end user authentication, and 2) that it may prove the identity of the device itself.
- **Robust DRM Implementation** – Ensure any implementation of a DRM specification can be trusted to protect the data that users acquire and the content and service providers require .
- **SIMLock / Device Personalization** – Ensure that subsidized mobile devices remain locked to the appropriate network until unlocked in an authorized manner.
- **Secure Software Download** – Enable the secure download of application software or updates, firmware updates or patches to protect against attacks.
- **Secure channel between device and UICC** – Provide shared functioning for security sensitive applications (e.g., an m-commerce application) that must implemented partly in the UICC and partly in the device.
- **Mobile Ticketing** – Enable new services based on a user purchase of an electronic ticket which is downloaded to the mobile device and used for entry to an event or access to a service.
- **Mobile Payments** – Enable the mobile device to serve as a user's wallet or purse for electronic payments to point of sale devices. Support for a variety of payment sources including credit cards, debit cards, pre-paid funds, and online accounts.
- **Software Usage** – Assure that software applications retain their integrity against attacks, adhere to device user policies, and cannot interfere with other device functions.

# MPWG MTM Specification Highlights

- Maintains passive TPM/MTM presence
- Introduces a secured Root Verification Authority for engine policy
  - Adds PKI public key verification for authorization authentication
- Deliberately implementation flexible
  - Minimum mandatory requirements
  - Does not mandate any MTM internal re-writable NV storage
  - Conducive to virtual MTM implementations
- Introduces second MTM monotonic counter
  - Low modulus “Global” counter for image replay protection
  - Local counter for platform and application specific credential revocations
- Defines a subset of TPM V1.2 commands as mandatory
  - Leverages pre-existing TPM functionality and Roots of Trust
- Introduces new MTM functionality to assist in integrity credential verification

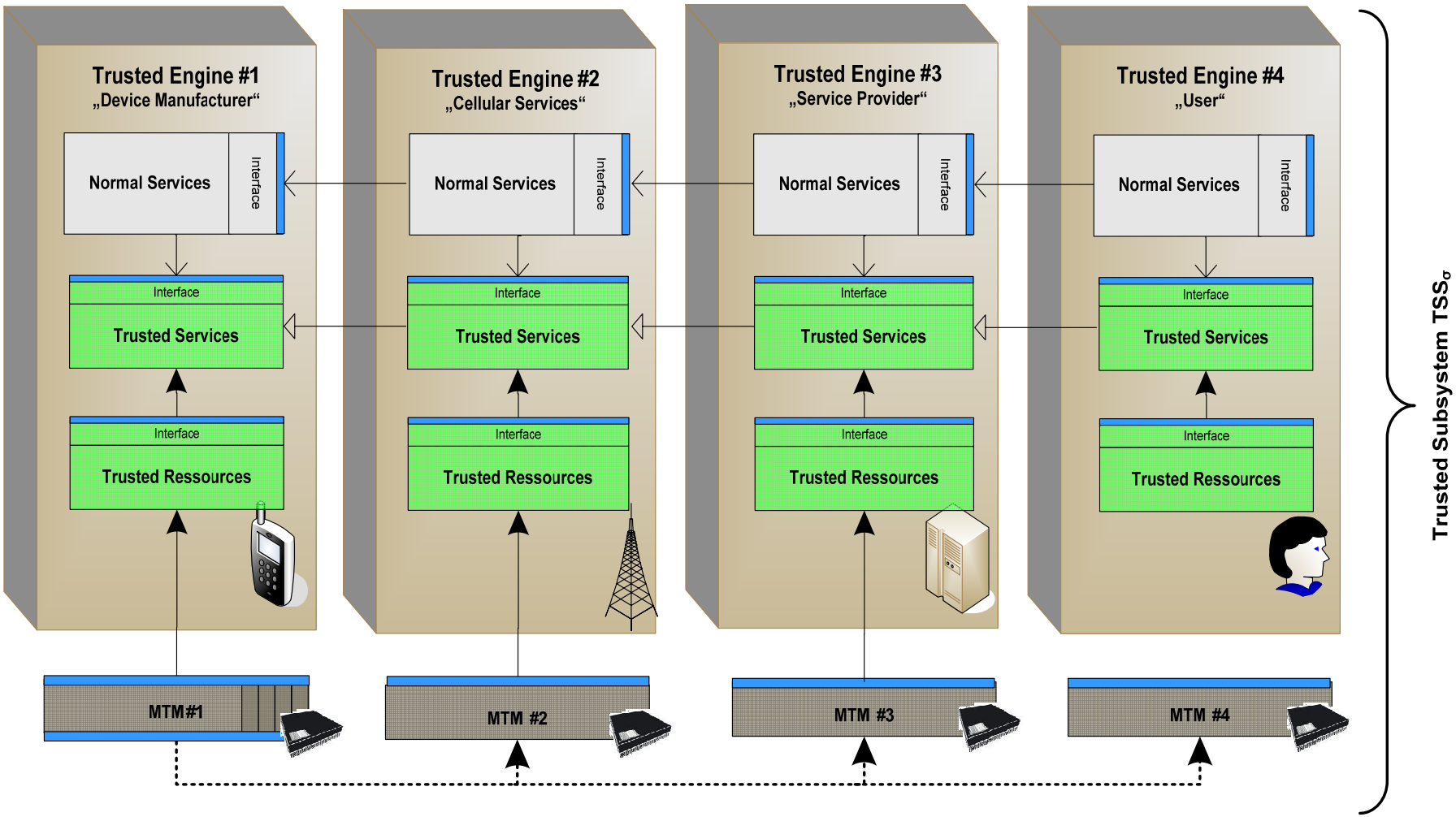
# MPWG Platform Highlights

- Introduces RTV as a root verification agent under the control of a secured Root Verification Authority identity
- Controlled mutability PKI public key RVAI
- Elaborates
  - An abstract architecture for platform/device description
  - A verified and controlled transitive trust verified boot mechanism
  - A reference integrity framework to validate platform integrity
- Reference Integrity Metrics - RIMs
  - An authentication mechanism to assure the validity of all integrity control information
- RIM Certificates
  - An authorisation mechanism to validate the authority of all authorisers traceable to two kinds of Mobile Trusted Modules
    - MRTM – Mobile Remote Owner TM
      - For remotely configured security policy
      - Allows for local verification
    - MLTM – Mobile Local Owner TM
      - For locally configured security policy

# Trusted Subsystems

- A subsystem with designated TBB, acting on behalf of a single stakeholder
- Consists of
  - Trusted Engines (TE)
  - Trusted Services (TS)
  - Trusted Resources (TR)
  - Measured Services
  - Normal Services
  - Normal Resources
  - Security Policies (SP)
  - System Configurations (SC)
- Supported by one or more dedicated MTM / vMTM's
- Principal entities (MPWG spec):
  - local stakeholders Device Owner (DO) and User (U); and the
  - remote stakeholders Device Manufacturer (DM), and more general Remote Owners (RO) (e.g. Communication Carrier, Service Provider).

# Trusted Mobile Platforms



# TSSys functions

- The functionality of a *TSSys* either is based on dedicated resources of an embedded engine or may provided by trusted services of external engines.
- Each subsystem is able to enforce its SP and SC
- Available to a TSSys are functions of
  - Its TR
  - The derived TS
  - External TS
  - Constrained by
    - SP and
    - SC of the stakeholder
- TSSys are functionally isolated from each other
  - Accessible to each other only if interfaces are described and exported
  - Depends on trust between stakeholders
- Stakeholders can establish trust by issuing SP
  - A set of credentials
  - Reference measurements, quality assertions, security-critical requirements, …

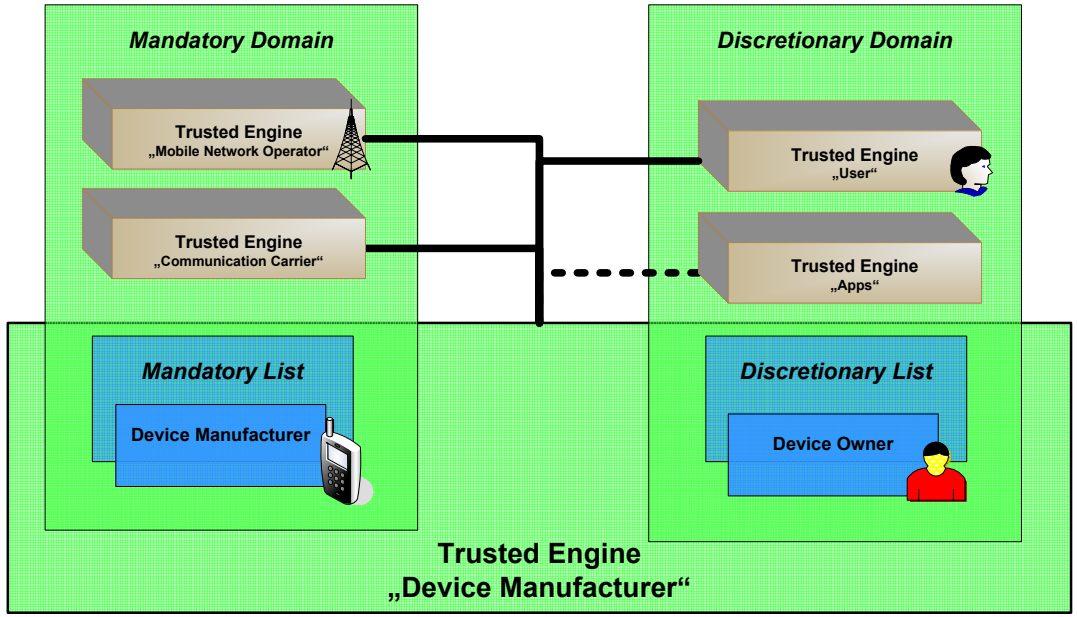
# Kinds of services

Services customise platform ressources and make them available for computation and storage within one engine and communication with external entities

Services of a trusted engine come in three flavours

- Trusted services
  - Customise trusted resources
  - Are implicitly supplied with EK or AIK to be able to attest to their trustworthiness
  - Are capable of providing trustworthy measurements of their current state and
  - To provide evidence thereof to normal services or other enitites
- Normal services
  - Customise normal (unmeasured) resources
  - Can access trusted resources via a TS
- Measured services
  - Are normal services which additionally
  - Are provided with integrity metrics of themselves by a TS

# Domains

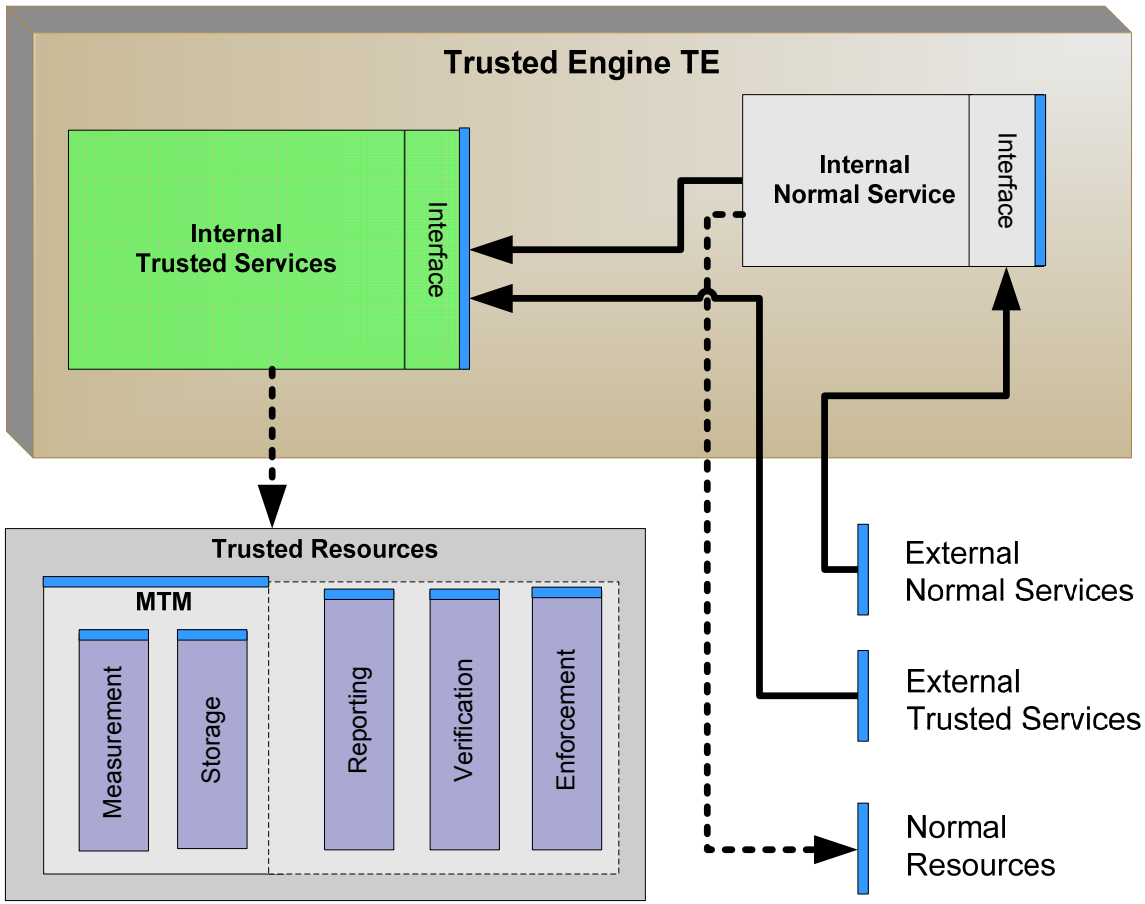



- Extensibility requirements led to the distinction between mandatory and discretionary domains
- Mandatory domains
  - Belong to DM or DO
  - Engines inside a mandatory domain are permanently located on a trusted platform and hold security-critical and essential functionality
  - essential services of a trusted mobile platform should be located inside the mandatory domain, which does not permit a local stakeholder to remove a remote owner from the engine
  - Mandatory engines have access to a MRTM to guarantee that a valid and trustworthy engine' state is always present
- Discretionary domains
  - Contain services which are replaceable by the device owner *DO*
  - Discretionary engines are required to be supported by a MLTM

# Trusted engines

The TCG MPWG abstracts a trusted mobile platform as a set of tamper-resistant trusted engines.

Each Trusted Engine is able to
- implement arbitrary software functionalities as trusted and/or normal services provide evidence for its trustworthiness, and report its current state, access a set of trusted resources, and import and/or export services, shielded capabilities and protected functionality.



# Trusted engines functionality

**Minimal capabilities**

- implement arbitrary software functionalities as trusted and/or normal services,
- provide the evidence for its trustworthiness,
- report the evidence of its current state,
- obtaining and using Endorsement Keys (EK) and/or Attestation Identity Keys (AIK),
- access a set of trusted resources, and
- import and/or export services, shielded capabilities and protected functionality.

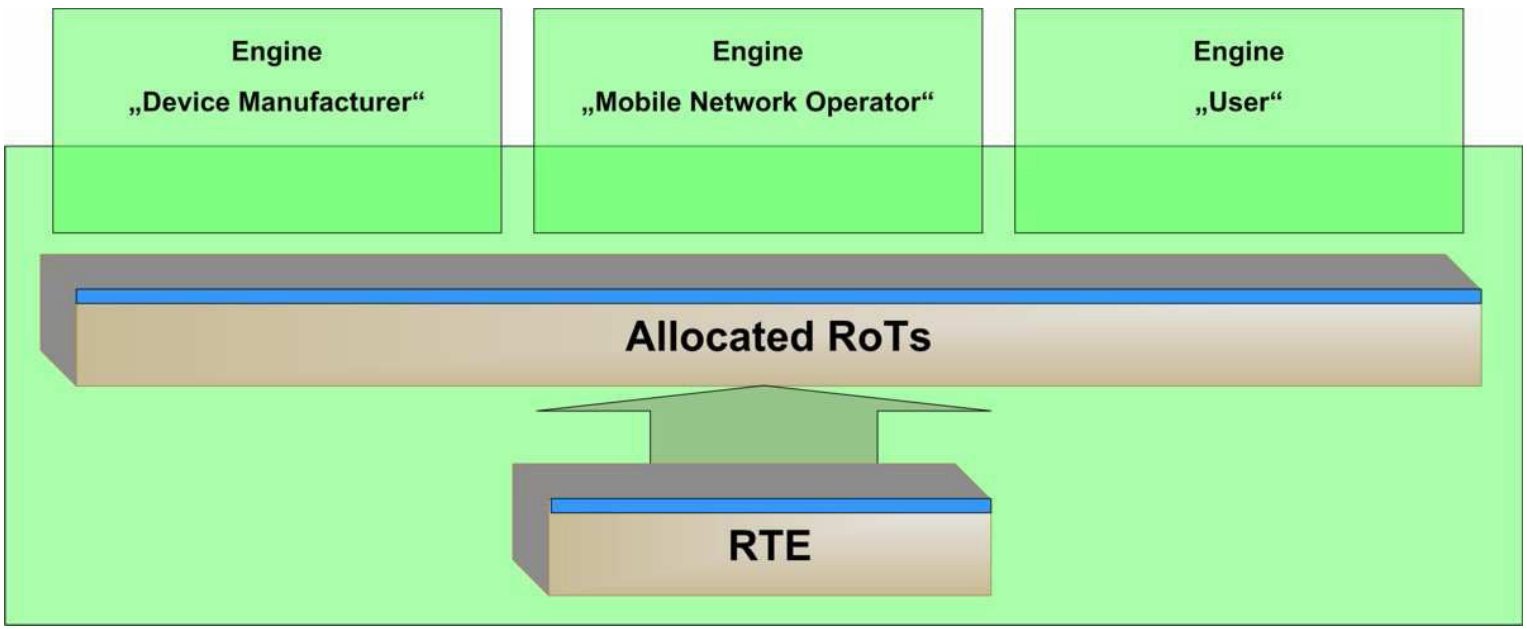
**RoTs as Trusted Resources**

- The TCG MPWG defines Roots-of-Trust (RoT) as Trusted Resources
  - Root-of-Trust-for-Reporting (RTR)
  - Root-of-Trust-for-Storage (RTS)
  - Root-of-Trust-for-Measurement (RTM)
  - Root-of-Trust-for-Verification (RTV)
  - Root-of-Trust-for-Enforcement (RTE)

- Typically, a Mobile Trusted Module (MTM) consists of a RTR & RTS (with a subset of TPM v1.2 functionality plus a set of new Mobile-specific commands)
- Each RoT vouches its trustworthiness either directly by supplied secrets (EK, AIK) and associated credentials, which are only accessible by authenticated subjects of the stakeholder, or indirectly by measurements of other trusted resources.

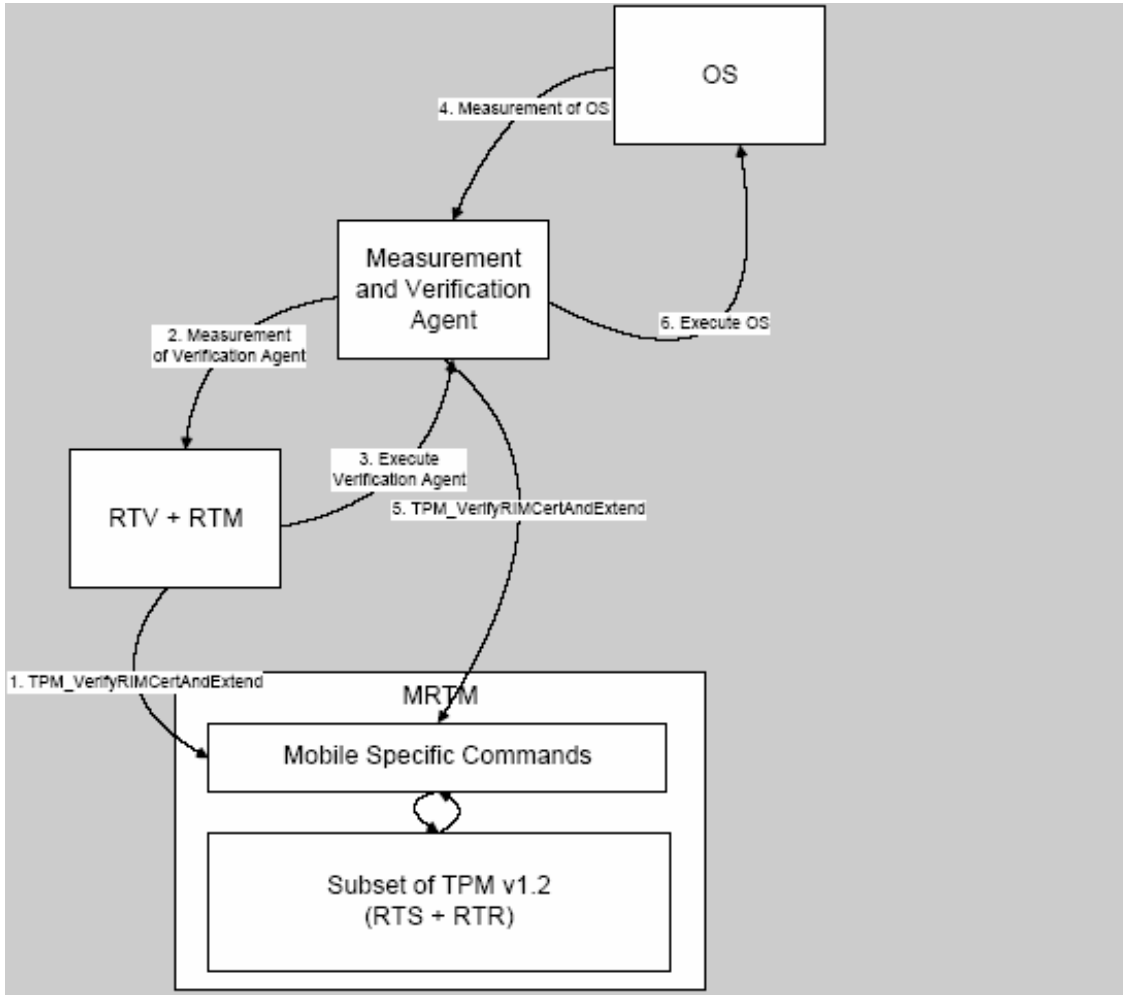
# RoT for Enforcement

- Enforcement is required if an engine uses *allocated* resources
- The RTE then functions as a trusted boot loader
- ensures the availability of all allocated trusted resources and services within aTSSys



# Trusted boot process

- **Trusted / Authenticated Boot** Measurement of the initial device state and confirms integrity of the underlying system
- *Measure -> Verify -> Extend Process*
- MRTM MUST support secure boot, MLTM can

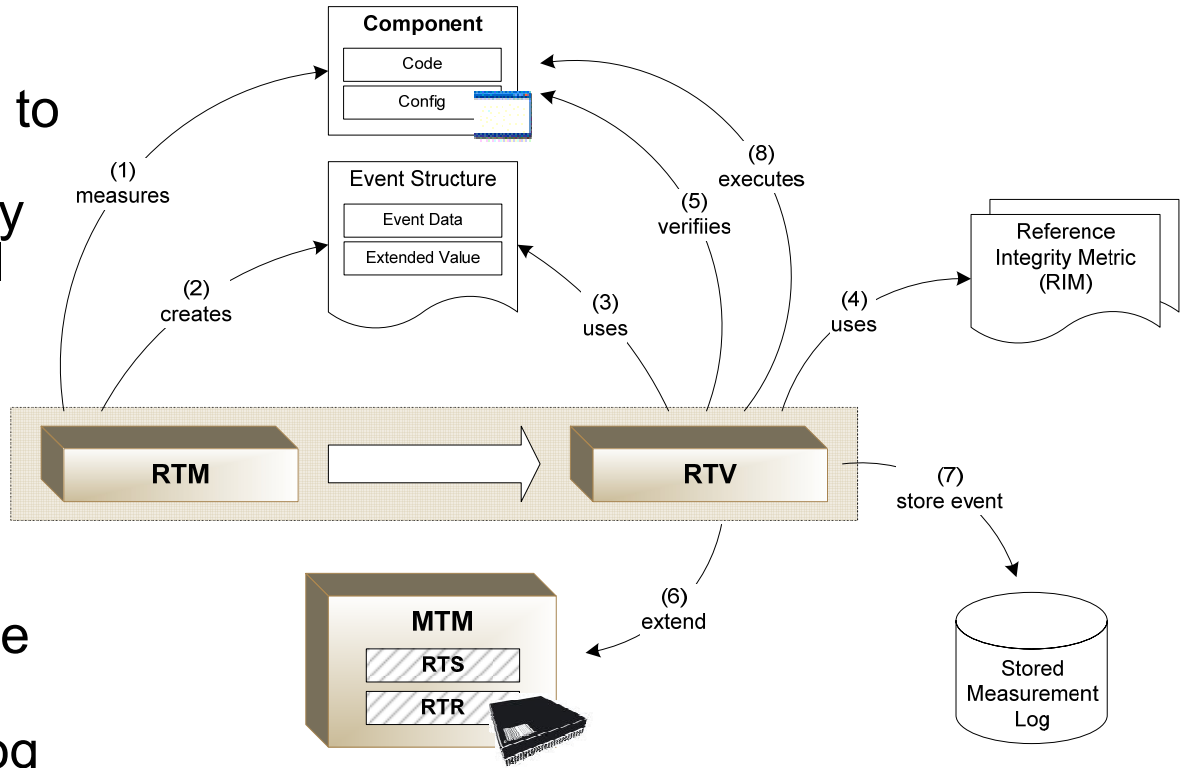


# Key concepts of local verification

- **Reference Integrity Metrics (RIM)**
  A value used to validate the result of a measurement taken before software or hardware is loaded or initialized (for execution).
- **Target Integrity Metric (TIM)**
  - Integrity Metric of a target object or component as measured by the measurement agent of that object. Typically a digest of a software image and/or configuraton data.
- **RIM Auth**
  - An actor that signs the RIM_Certs and delegation RIM_Auth_Certs under its authority, i.e. the source providing and authorizing the external RIM_Certs
- **RIM Cert**
  - A means of securely authentication RIM information for a given target object by an authorized RIM_Auth. Typically this is a data structure that is signed by the RIM_Auth.
- **RVAI**
  - The root public key of a hierarchy of RIM_Auth public keys, i.e. the public key that will be used to verify the RIM_Auths
- Each engine has its own set of RIMs and RIM_Auths. The stakeholder owner approves the RIM_Auths for that Engine

# Local verification

- **Local verification Process**
  In the mobile domain, to avoid communication costs, this functionality is extended by a local verifier, which checks the measurements against a given Reference Integrity Metrics (RIM).
- Device-side verifier offers assertions to the integrity values.
- verifier receives the log and a signed PCR value as well as the certicates to verify the signature



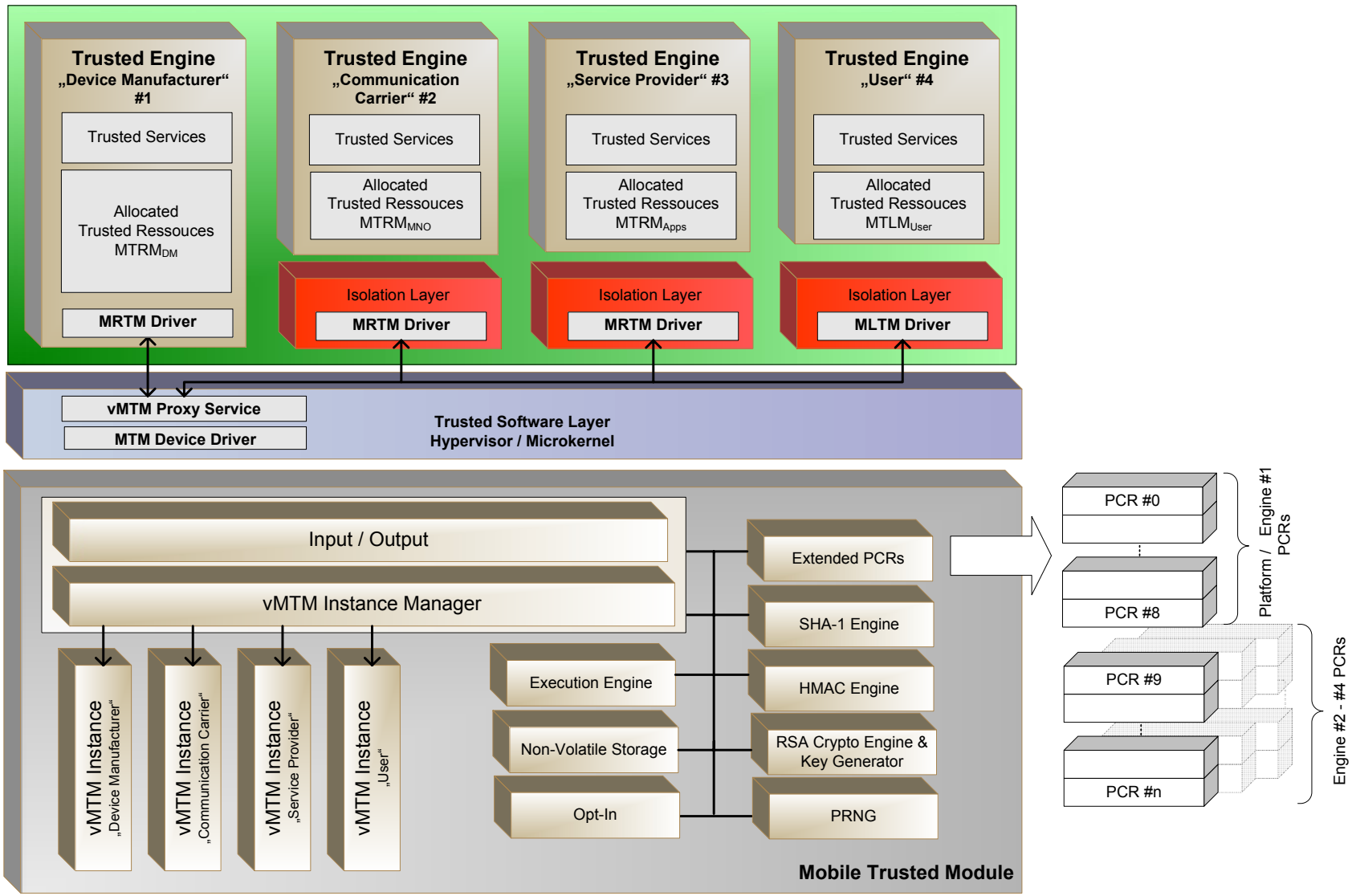
# Architecture variants for MTM & TSSys

- Standard TPM based
  - uses a non-modied standard TPM to build the TCB of this system.
  - The secret keys are stored into a single key-hierarchy on behalf of *DO*
  - an adversary or malicious local owner may be able to access the secret keys of a remote stakeholder and take control of a remote owner compartment
  - The user can disable the MTM or corrupt engines of remote stakeholders
- Software-based MTM-Emulation Model
  - Software-based allocated *MTM*-emulation with an isolated key-hierarchy per vMTM instance
  - Security critical data, e.g. *EK* or *SRK*, are only protected by software mechanisms outside the tamper-resistant env.
  - Advantage high performance – good for simulation and testing
- Generic MTM-based Model supporting multiple Stakeholders and virtual MTMs

# Multiple-stakeholder model

- Adaption of secure coprocessor  architecture for virtual TPMs proposed in [7]
- Single genuine hardware MTM and several virtual software MTMs
- One MTM for each trusted engine
- A Trusted Software Layer oers a vMTM Proxy Service to all embedded trusted engines *TE*
  - routes MTM commands from a *TE* to its dedicated instance *vMTM*
- Requires some additional functionality to separate vMTM instances

# Multiple stakeholders in virtual TEs



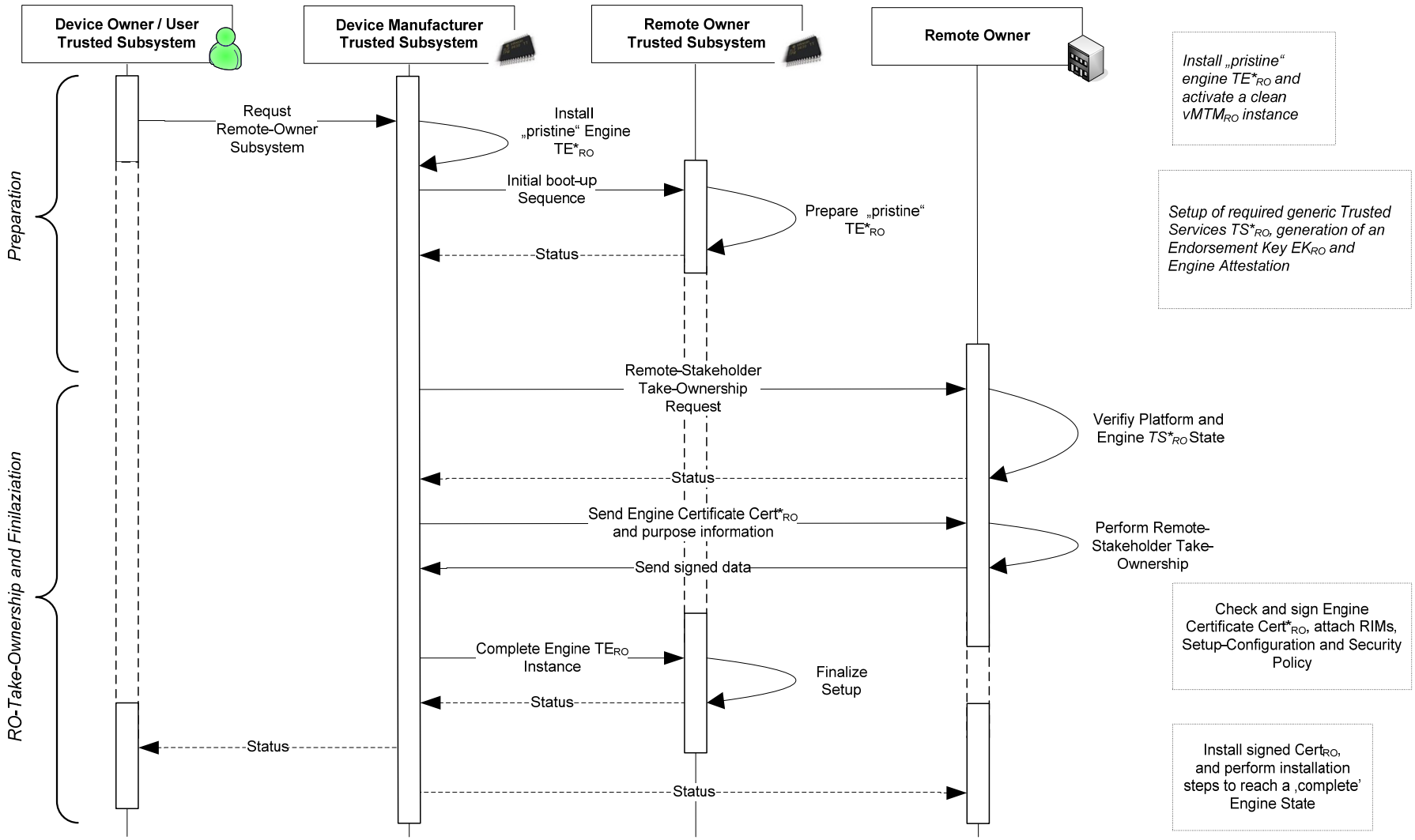
# Remote owner take ownership

- Unspecified in MPWG reference architecture
- Minimal requirements
  - the remote owner MUST be protected from a User attempting to remove the remote owners ownership of the engine, or attempting to disable or de-activate the engine's MRTM
  - (eninge might be removed entirely, however, if on a DO-conrolled list in the discretionary domain)
  - remotely owned engine MUST therefore support secure boot, to ensure that the engine loads the way the remote owner is expecting.
- a general model is that the engine's MRTM is already enabled and activated, and already has an owner set when the User takes possession of the device.
  - This MUST be true of the Device Manufacturer's Engine, for example.
  - However, a remote owner MAY be able to take ownership at a later date if not already set

# Remote take ownership proposal

- Idea
  - install and instantiate a ’blank’ trusted subsystem containing
    - a pristine engine
    - a set of generic trusted services (such as a trusted boot agent)
  - Certify it by the remote owner on condition that
    - the platform is able to provide evidence of the TSS’ pristine configuration and policy conformance with respect to the RO’s policy

# Remote take ownership protocol (proposed)

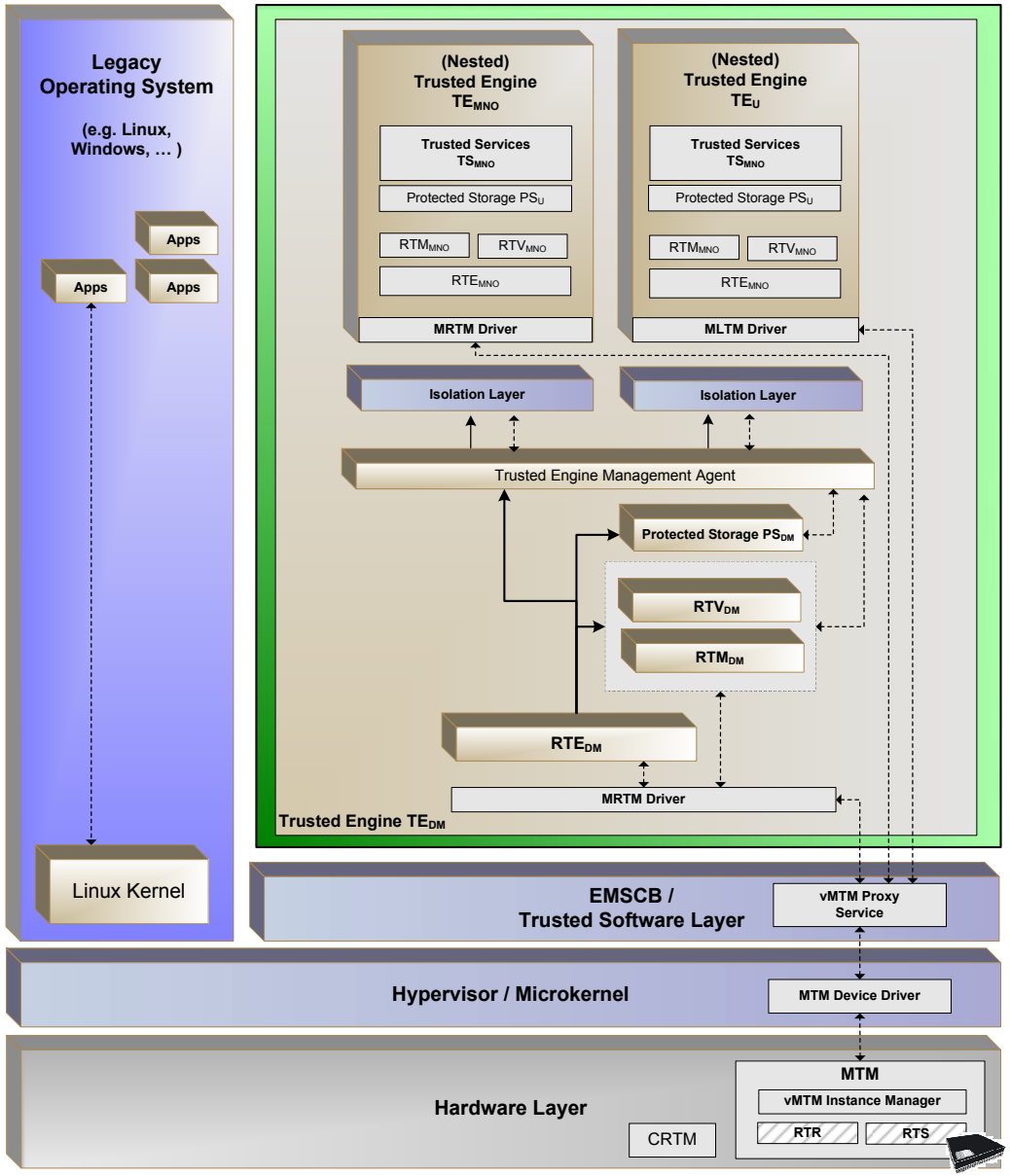


# Remote take ownership protocol (proposed)

- Precondition: DM engine started in trusted boot, containing ability to install pristine engine
- Protocol:
- Platform preparation
  - Install pristine engine e.g. from dedicated ROM
  - Booted under control of RTE_DM
  - Instantiate vMTM_RO in this engine
  - Create EK pair locally in this engine, and according certificate
  - Local or remote attestation of the pristine engine
  - Using RIM Cert of RO
- Take ownership execution
  - Platform requests RO take ownership by sending generated EK, certificate, and attestation data – *encrypted channel!*
  - RO checks attestation data and validity of intended purpose of the engine on that particular platform
  - Signs the engine‘s certificate
  - Creates RIM certificates for local verification
  - Sends it to the platform, encrypted with EKpub_RO to TSS_DM
  - TSS_DM triggers completion of TSS_RO instance, i.e.
    - Installation of signed EK certificate
    - Installation of RIM certificates
  - Completion is confirmed to RO

# Prototype

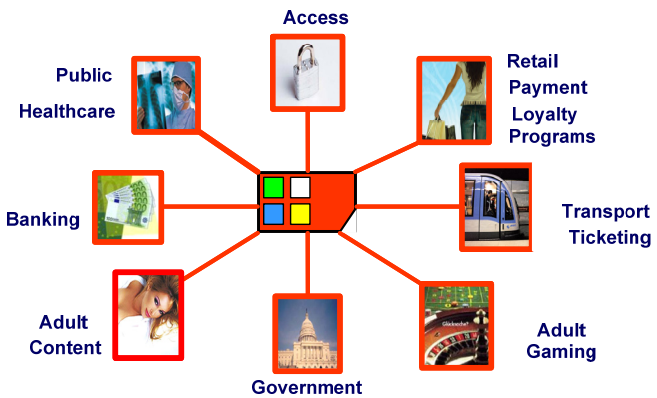
- A prototypical implementation of a Trusted mobile platform based on EMSCB/Turaya
- Work in progress

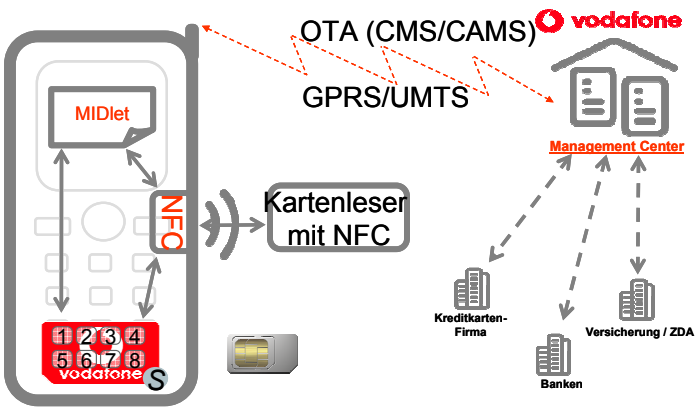


# MegaSIM

- USIM functionality, security and high-density flash
  - fully functional SIM cards
  - SIM card form factor
  - offer secure personal storage in a two-chip solution
  - Secure smart card controller with additional crypto engines, crypto libraries and MPU provide the highest level of secure operations
  - A secure, high-performance controller, with embedded mask ROM, is based on a 32-bit ARM SC100 secure core. The controller empowers mSIM cards with multi-tasking capabilities.
  - NAND flash memory and advanced crypto engines offer high-performance, low-power and secure personal storage.

# MegaSIM Use Cases




- **propose megaSIM-enabled handsets as platforms for qualified electronic signatures**
- **Common handsets, Multi- application- SIM card as certified platform for service access, enables novel services**
- **Ongoing projects in Germany and Europe**
  - Ticketing, Payment, access control, Elektr. Gesundheitskarte

**Integration of VAS in the health sector**

- Telemedizin / Telemonitoring
- Individualised care offerings, prevention, outpatient care, Fitness, Wellness, home care, using the elektr. Patienten Akte
- Assistance services based on LBS and GPS

**Quote: ‚MNOs do not want to stand in for higher costs of megaSIMs‘**

# MegaSIM vs. MTM

## Some arguments in favour of the MTM

- Better **cost/efficiency ratio**
  - Through the unique feature of trusted boot, the MTM can cannibalise the full resources of the device for security features and services
  - While a megaSIM is a monolith – admittedly more powerful than saingle mTPM
  - For this sole reason, the MTM is likely to even be cheaper than a megaSIM
- **Virtualisation** is a powerful TC concept that can be used in conjunction with an mTPM to realise the concept of **secure compartments**
  - Various vendor, MNO, and service provider specific trust and AAA concepts can be realised with no physical limits
  - Even xSIM can be virtualised in TC-enabled devices
  - While a megaSIM offers only limited (security service) scalability
- The MNO-centric business model to sell or rent megaSIM compartments to third party SPs, does as well apply to MTM – only better for the mentioned reasons
- Discussion
  - ‚Do You think that MTMs will soon be incorporated in a megaSIM?‘.
  - Answer: ‚Such a device would perhaps not be removable anymore, since otherwise the trusted boot concept would be void. If it is not removable, why would anyone have two physical security anchors in one device (or are we building hybrid cars)?‘
  - IMHO: Rather virtual xSIMs will run in an MTM protected environment – as legacy applications, alongside with virtual smart cards, etc.