CCSDS Mission Operations Services

Mario Merri
Head of Mission Data Systems Division (ESOC/HSO-GD)
CCSDS Mission Operations and Information Management Services (MOIMS)

Mehran Sarkarati
Head of Applications and Special Projects Data Systems Section (ESOC/HSO-GDA)
CCSDS Spacecraft M&C WG (SM&C)

Savoir Day @ ADCSS 2015
20Oct2015
ESTEC

All in One Slide

❑ The Problem

- **Terrestrial software has developed enormously in the last 2 decades whereas the software for Mission Operations has not**
- **On the <u>ground-space interface</u>: little progress**
  - **In Europe the ECSS PUS continues to serve well, but:**
    - Developed in 1994
    - Only used in Europe
    - Extends as far as the edge of the MCS
    - Fixed to CCSDS Space Packets
- **On the <u>ground-ground interfaces</u>, interoperability between organisations is still difficult**
  - **No standard set of interfaces for common operations**
  - **No standard definition of what information is exchanged**
  - **Everyone does this differently**

❑ The Solution

- **CCSDS is defining a set of MO Services that are:**
  - **Standard across all CCSDS Agencies**
  - **Distributable**
    - Designed to be used on-ground, on-board and across the spacelink
    - Extend from on-board, through the ground systems, to the end users
  - **Independent of transport and encoding technology**

Standard Phone Services

- Place call
- Receive call
- Show contacts
- …

Standard Music Services

- Stream songs
- Listen music
  - By song
  - By author
  - By album
  - …

Standard Phone Services

- Place call
- Receive call
- Show contacts
- …

Standard Music Services

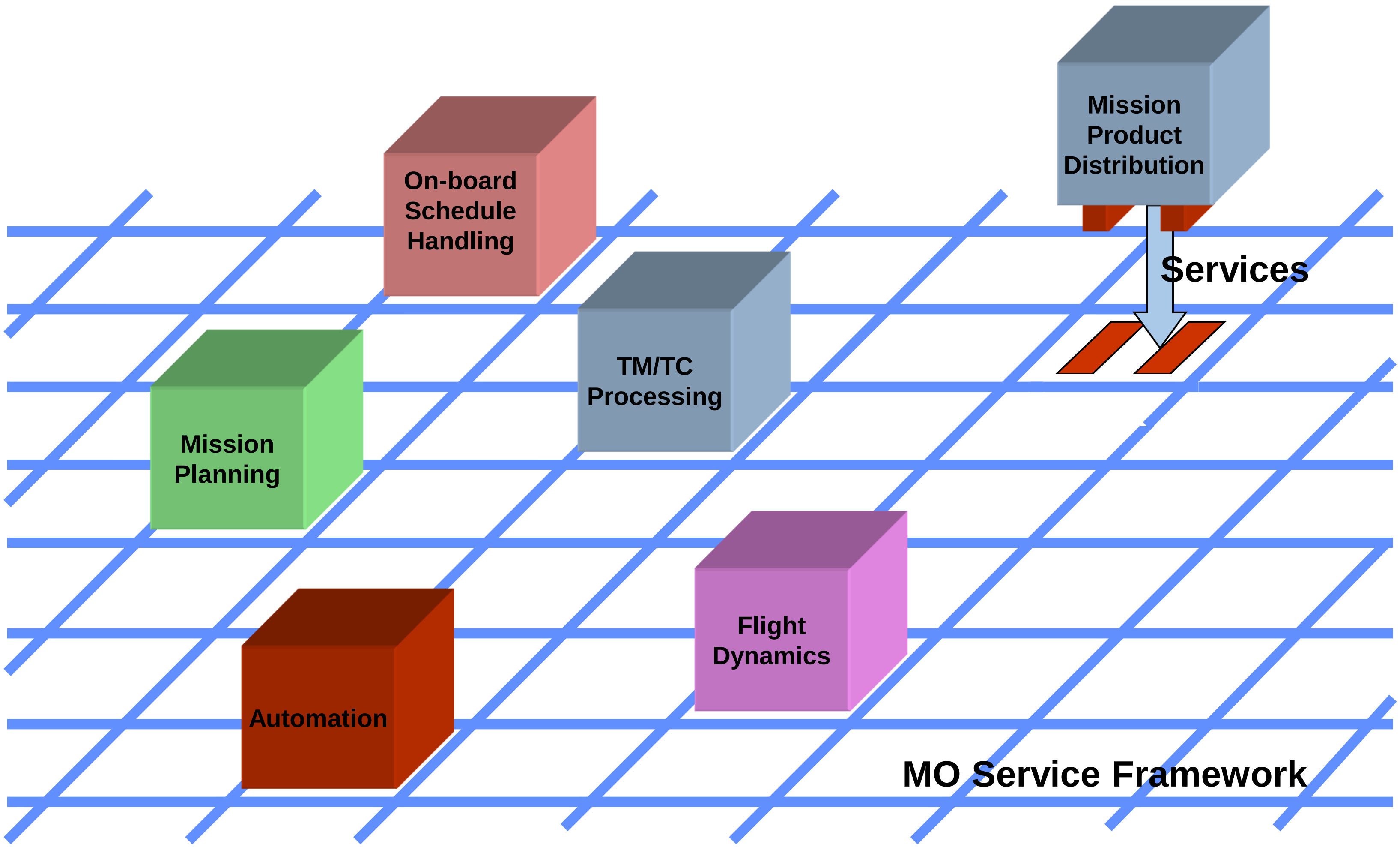
- Stream songs
- Listen music
    - By song
    - By author
    - By album
    - …

What Can We Learn From This Analogy?

- **Communication is an enabling technology**
  - **Necessary condition (must have it)**

- **… but the real breakthrough is in the definition of semantically rich application-level services**
  - **Are independent of the communication technology (Bluetooth, wire/USB, …)**
  - **Allow independent developments at the two ends of the interface**
    - **Any Bluetooth telephone works with any Bluetooth-enabled car stereo**
    - **Does not prevent innovation**
    - **Increases the availability of commercial solutions → boost competition → cost reduction**
    - **Increases long term maintainability**
      - One can replace the phone w/o replacing the car!

How Does This Apply to Space?

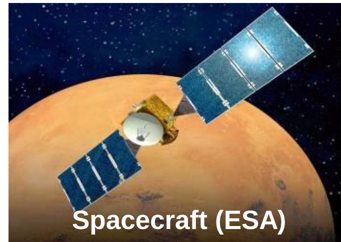


What Are MO Services?

- Coherent set of <u>application-level</u> services needed to operate a space mission, such as:
  - Classical Monitoring & Control (TM, TC, Events)
  - Navigation (e.g. orbit, attitude, events)
  - Mission planning, scheduling and automation
  - Mission product data distribution
  - File management
  - Time management
  - Software management
  - … and more (extensible)

- Key technical proprieties of MO Services:
  - Follow <u>SOA</u> approach
  - Standardise exchange of <u>semantically rich data</u>
  - Are rigorously defined in a common way via the <u>MO Service Framework</u>
  - Are <u>technology and location independent</u>
  - Are compatible with <u>model driven development</u> and <u>auto-coding</u>

MO Services at Work
(an over simplified deployment example)



M&C Services
File Management Service
…

M&C Services
Planning Services
Navigation Services
…

Data Distribution Services
Planning Services
…

M&C Service
Data Distribution Services
Planning Services
…

M&C Services
SW Management Service
…


Spacecraft Manufacturer
(Astrium)

Re-use of same MO Services across space and ground segments!

Re-use of same MO Services over different technologies!

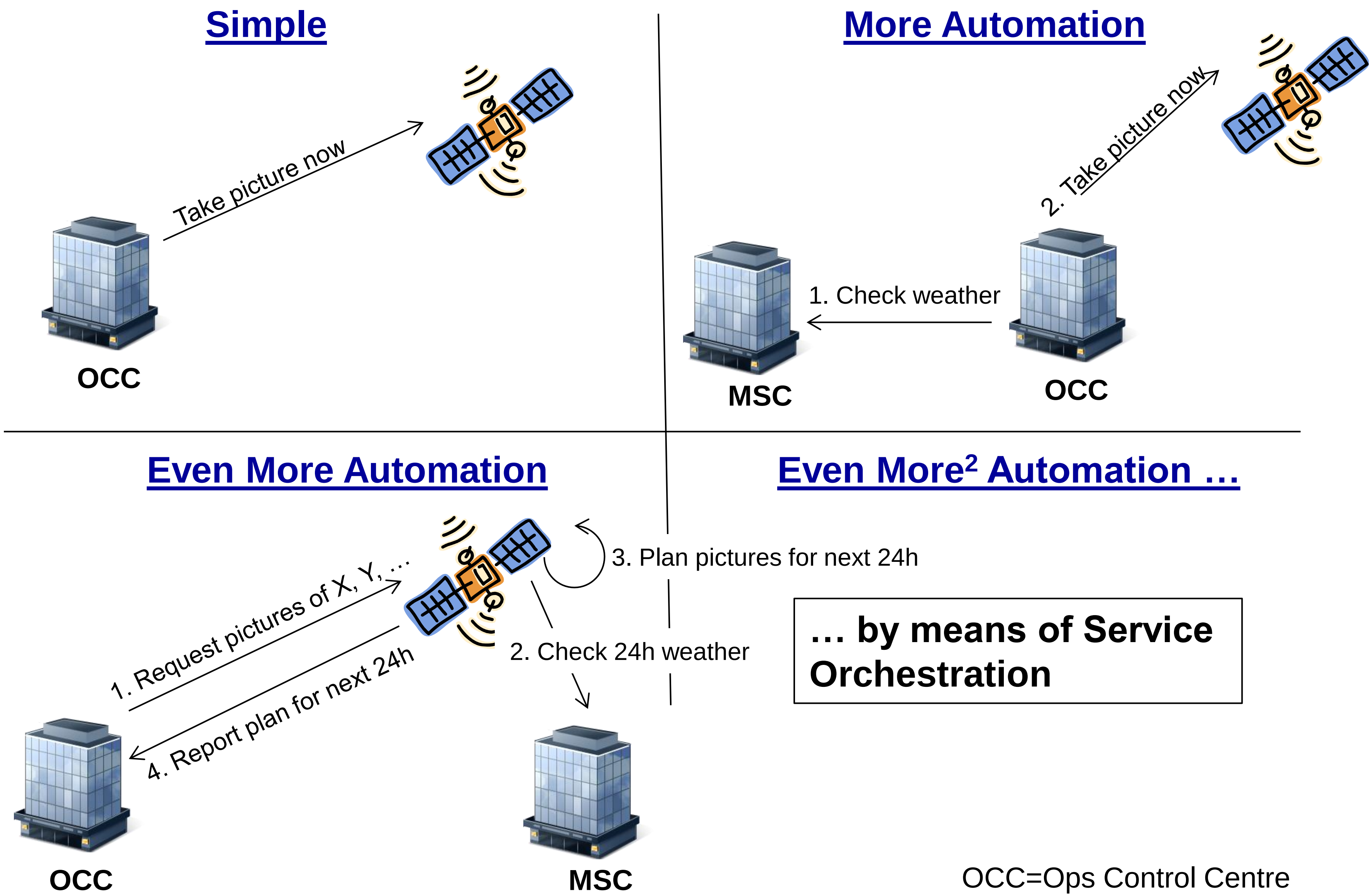
Benefits of MO Services

- ❑ **Higher re-use and lower cost:**
  - Reuse across missions of
    - components (ground/on-board)
    - code (Open Source Code already available)
    - ops concept (even from different Primes),
    - people (minimal training)
    - → shorter schedules, less risks, higher quality …
  - Ability to establish common multi-mission infrastructure (ground/on-board)
  - Boost the availability of commercial components
    - increased industrial competition
    - ability to select the best product from a range of compatible components
    - vendor independence
  - Support code auto-generation

- ❑ **Higher flexibility:**
  - More interoperability between agencies (10 agencies today involved in MO)
  - Flexible deployment boundaries (ground/on-board)
  - Capability of "bridging" between technologies
  - Improved long-term maintainability (both for components and infrastructure)

- ❑ **Higher Mission Data Return:**
  - Focus resources on field-specific innovation (not in reinventing the I/Fs)
  - Increased mission automation and specialisation by service orchestration

More Mission Automation = More Science



… by means of Service Orchestration

OCC=Ops Control Centre
MSC=Meteo Service Centre

MO Services are Getting Real!

Because they are being used more and more!

CCSDS Prototypes

CCSDS requires that each new standard is validated via 2 independently-developed and interoperable prototypes

Test Bed available that simplifies this work and might also be used in the future to "certify" compliant MO applications

So far 3 space agencies (CNES, DLR, ESA) are involved in formal prototyping

Other Prototypes/ Studies

Since 2006, about 29 (known) prototypes:

- Astrium-D -> 1
- CNES -> 6
- DLR -> 1
- ESA -> 4
- EUMETSAT -> 1
- NASA/GSFC -> 2
- NASA/KSC -> 1
- NASA/JPL -> 2
- NASA/JSC -> 7
- NASA/MSFC -> 2
- UKSA -> 2

Several studies

A number of Ph.D.'s

Actual Projects

- ISIS: CNES new ground segment infrastructure (fully MO based internally)
- EGS-CC: new European Ground System Common Core (external I/F)
- METERON: ISS experiment (MO over DTN)
- OPS-SAT: ESOC cube-sat (space-ground link fully MO based)

Status in 2012

MO Services Status

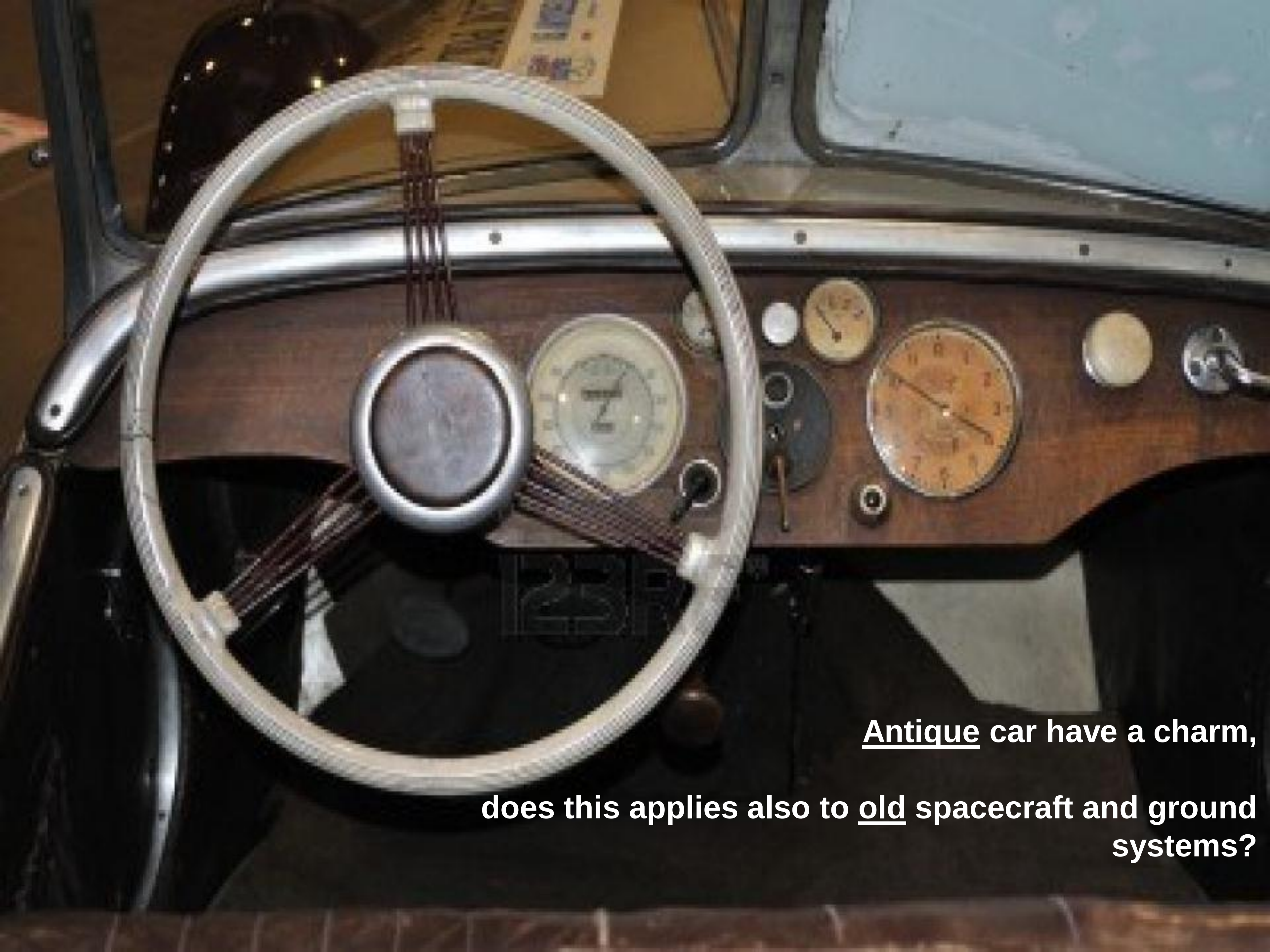
- **MO Services Framework is available (ESA + CNES)**
  - Standards are published (MAL and COM)
  - Java implementation available as Open Source Software from ESA and CNES
  - C++ draft implementation available as OSS from NASA/JSC
- **Language mappings**
  - Java API published (CNES)
  - C++ API in preparation (NASA)
- **Technology (encoding and transport) mappings**
  - Space Packet Transport Binding and Binary Encoding published (CNES + DLR)
  - ZeroMQ Transport Binding and CNES Binary Encoding in preparation (CNES)
  - HTTP Transport Binding and XML Encoding in preparation (ESA)
  - TCP/IP Transport Binding and Split Binary Encoding in preparation (ESA)
- **MO Services**
  - M&C Service under final prototyping (ESA + DLR)
  - Common Services in preparation (ESA + CNES)
  - Mission Data Product Distribution Services in preparation (ESA)
  - Mission Planning & Scheduling Service WG approved
- **Check out: www.ccsds.org**

Do We Really Need MO Services?

Of course not,

we could live without them,

but …



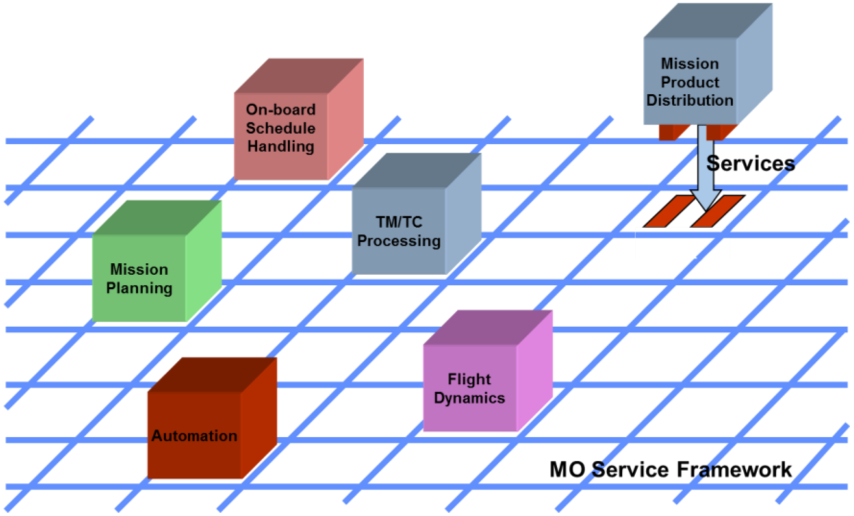
Antique car have a charm,

does this applies also to **old** spacecraft and ground systems?

CCSDS MO Services Benefits Overview

Open Architecture

MO Approach	Current Solutions
MO Services standardise the main interactions between the high-level components of a space system ►Components can be independently developed	Often specific component-component interfaces are defined (message syntax and full protocol stack)

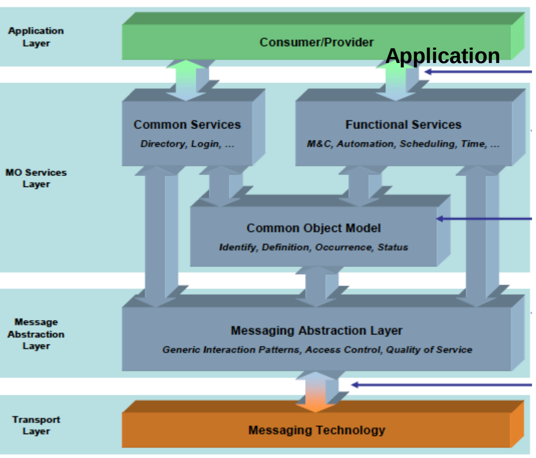


Generic Services

MO Approach	Current Solutions
Similar interactions between dissimilar components can use the same generic Service specification ►Service implementations can be common to multiple service deployments. Service consumers (client functions) can work with any compliant service provider	Often dissimilar interfaces are used in different contexts for similar information exchanges (e.g. S/C M&C, GS M&C, Rover M&C)

Clean Protocol Layering

MO Approach	Current Solutions
The MO Framework identifies clear conceptual layers that allow the definition of abstract services ► Service implementations can be common to multiple service deployments. Service consumers (client functions) can work with any compliant Service provider	Often interfaces are defined explicitly in terms of a low-level message transport technology, such as packet TM/TC or file-based data formats. This can make it difficult to support the same interaction in different deployment contexts – e.g. across space and ground interfaces.
Clear separation of engineering knowledge ► Easier to change the "business logic" of mission operations without impact on infrastructure layers and vice versa	Often engineering knowledge and its implementation in data formats and technology are inextricably linked



Technology and Location Independence

MO Approach	Current Solutions
The MO Services are defined independently of programming language, message encoding and transport technology ► Facilitate legacy integration, service extensibility, interoperability between different technology. Mitigate technology obsolescence	Most interfaces are defined explicitly in terms of specific data encoding and transport technologies or middleware. APIs, where they exist, are defined in terms of a specific language.
MO service can be deployed as most appropriate for the mission and independent of the location of the service provider and service consumer. ► New mission configuration (e.g. orbit determination on-board) can be considered with minimal impact on the rest of the system	Often the location of service provider and service consumer is fixed (e.g. in PUS the provider of Housekeeping Monitoring service is always on-board).

Auto Code Generation

MO Approach	Current Solutions
MO Language and Technology bindings are essentially transformation algorithms from the abstract service model to the physical implementation. ►Standardisation of the bindings allows automatic code generation for all services defined in terms of the MO Framework	Protocol encoder/decoder components are typically developed by hand. On top of being manpower intensive, it is also error prone.

Increased Automation & Science Return

MO Approach	Current Solutions
MO Services capture also the dynamic behaviour of an interface. While this is not new for live data stream interfaces, it has often not been applied for off-line information exchange (typically via file). ► Allow software components to access information without human intervention thus increasing reporting, automation, and science return	Many off-line interactions are file based. The method for file exchange is typically ad-hoc and may use: • Telephone coordination • E-mail • Manual file transfer. Often, multiple file formats are used to support similar information exchanges within the same system.

Multi-hop Activity Verification

MO Approach	Current Solutions
The MO Framework provides a single end-to-end approach for tracking the progress of activities (a command, a remote procedure, a schedule, etc). ►Allow end-to-end verification (e.g. of a lander via its orbiter or an automated on-board procedure that triggers the execution of other activities on-board). ►Allow "external monitoring": one component is able to monitor the activities in the system without requiring knowledge of what components are active.	The PUS is only able to support single hop verification and no activity chaining verification.

Interoperability & Business Opportunities

MO Approach	Current Solutions
By formalising and standardising the semantic information exchange at service level the potential for interoperability between systems is greatly improved. Moreover, the MO Services are being standardised in CCSDS (consisting of the 11 most important space agencies of the world). ►It will be the interoperability platform for cooperative missions ►European Industry will be able to compete more easily on the world market.	Current world-wide success-story in space interoperability relate mainly to mere data transfer (e.g. CCSDS TM/TC and SLE standards). PUS is a success-story at service level, but only European.

PUS and MO Services

Relationship to PUS

- **MO Services are fundamentally based on PUS**
  - Refactored to make them self-consistent and transport independent
- **MO Services expand PUS**
  - MO Services cover more functions than PUS
- **MO Services improve PUS**
  - The specifications are independent of transport and encoding technology
  - They are designed to be used on-ground, on-board and across the spacelink

MO Services are based on PUS

Service Type	PUS Service	MO Service
1	**Telecommand Verification**	**COM / Activity**
2	Device Command Distribution	Software management
3	**Housekeeping and Diagnostic Data Reporting**	**M&C / Aggregation**
4	**Parameter Statistics Reporting**	**M&C / Statistic**
5	**Event Reporting**	**M&C / Alert**
6	Memory Management	Software management
8	Function Management	Automation
9	Time Management	Time
11	On-board Scheduling	Scheduling
12	**On-board Monitoring**	**M&C / Check**
13	Large Data Transfer	Data product management
14	Packet Forwarding Control	Remote buffer management
15	On-board Storage and Retrieval	Remote buffer management
17	Test Service	
18	On-board Operations Procedure	Automation
19	Event-Action Service	Automation
20	**Parameter Management**	**M&C / Parameter**
21	Request Sequencing	Scheduling || Automation
22	Position-based scheduling	Scheduling
23	File Management	File Management

PUS-A

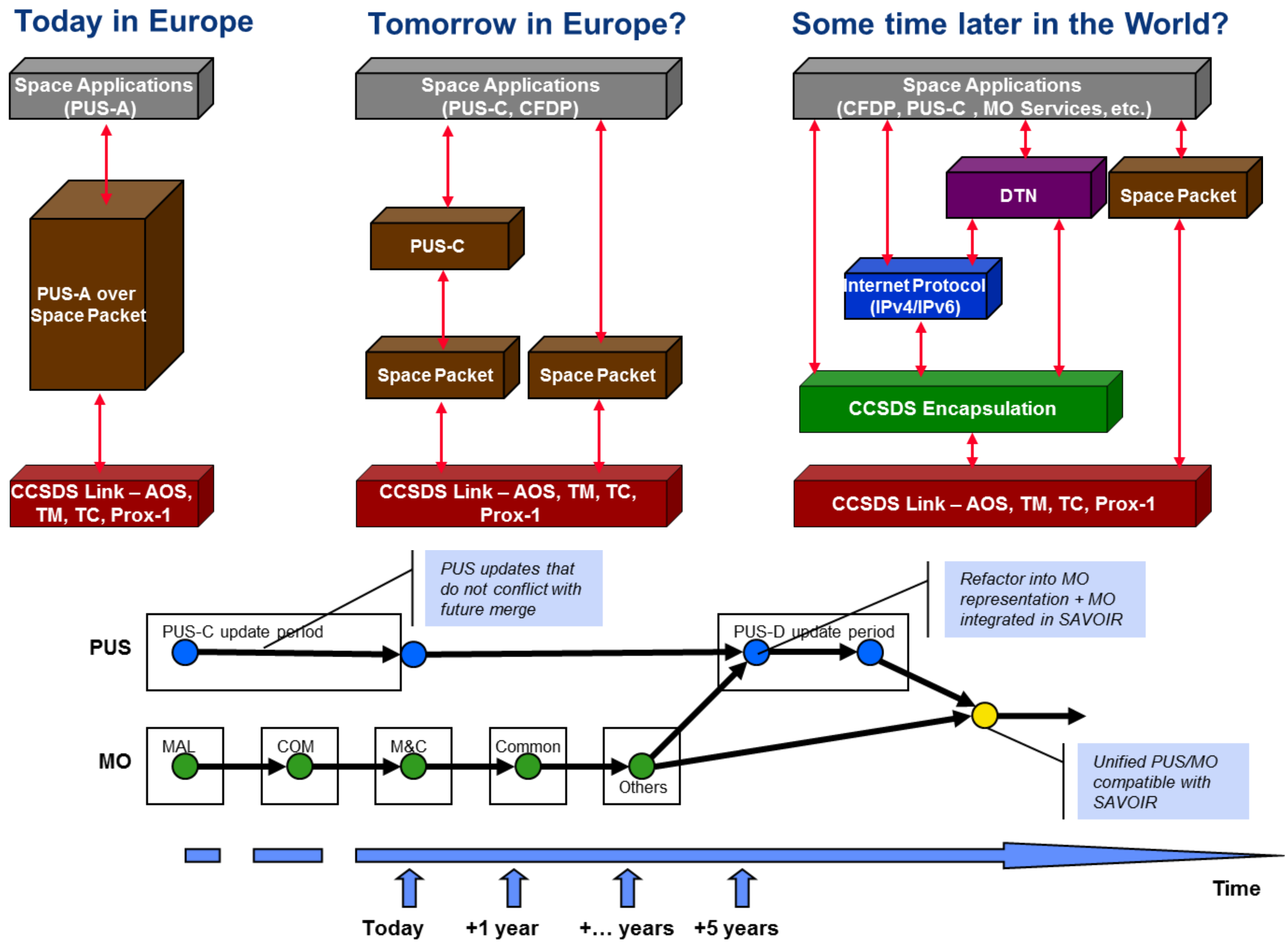
PUS-C

MO Services expand PUS

❑ Additional services planned in MO

- Navigation Services
- Planning Services
- Mission Data Product Distribution Services
- …

MO Services Roadmap



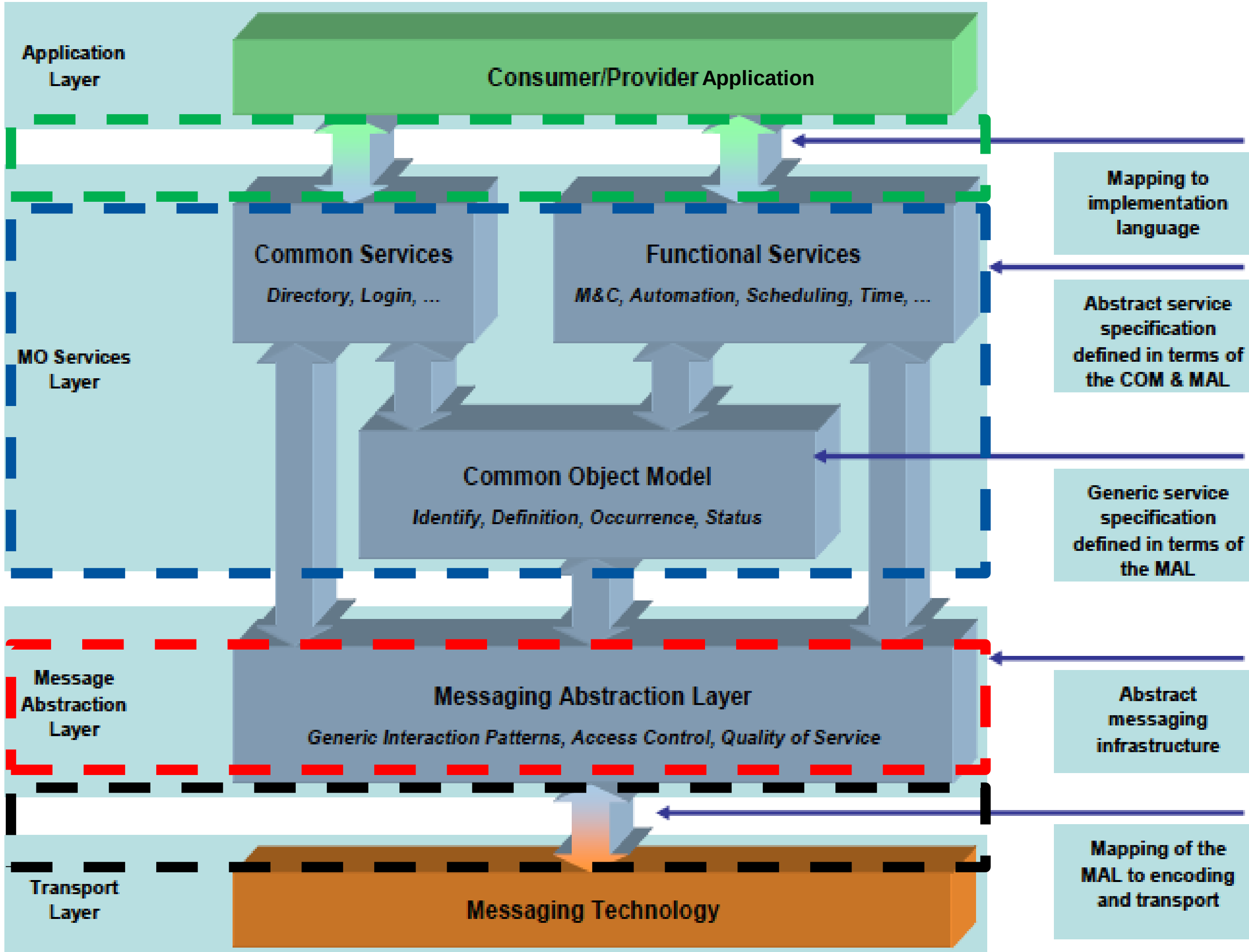
Thank you

Mario Merri: Mario.Merri@esa.int

Mehran Sarkarati: Mehran.Sarkarati@esa.int

Back-up Slides

MO Services Framework Layers



MO Services may be used by Applications coded in different languages

MO Services are expressed in terms of the MAL

Message Abstraction Layer (MAL) provides technology and location independence

MO Services may "travel" over different encoding/transport technologies

Do You Want to Know More?

- **High Level**
  - **Green Book:** http://public.ccsds.org/publications/archive/520x0g3.pdf
  - **Reference Model:** http://public.ccsds.org/publications/archive/520x1m1.pdf
  - **Videos:** http://public.ccsds.org/outreach/overview.aspx#CCSDS_Overview_Video
  - **Wikipedia:** http://en.wikipedia.org/wiki/CCSDS_MO_Services
  - **MO Wiki** https://github.com/esa/CCSDS_MO/wiki

- **MO Service Framework**
  - **MAL:** http://public.ccsds.org/publications/archive/521x0b2e1.pdf
  - **COM:** http://public.ccsds.org/publications/archive/521x1b1.pdf

- **Language mappings**
  - **Java API:** http://public.ccsds.org/publications/archive/523x1m1.pdf

- **Technology (encoding and transport) mappings**
  - **MAL Space Packet Transport Binding and Binary Encoding:** http://public.ccsds.org/publications/archive/524x1b1.pdf

- **MO Services**
  - **M&C Service (under review RIDs due 31May14):** http://public.ccsds.org/sites/cwe/rids/Lists/CCSDS%205221R3/Overview.aspx

- **Open Source Software**
  - **ESA:** https://github.com/esa
  - **CNES/NASA: upon request**