Embarking to end-to-end system modelling for Galileo Second Generation

presented by Catherine Morlet
ESA/ESTEC – NAV Directorate
Galileo system evolution architect engineer

24/11/2022

Outline of the presentation

1. Galileo Second Generation (G2G) in context
2. What has been done:
    - Development organisation
    - Achievements – model envelope
    - Documentation
3. Findings for future development and usage of MBSE:
    - (upcoming) use cases
    - Findings
4. Concluding remarks

G2G in context – program level view

Galileo program

- Europe's initiative started in the 1990's for a state of the art global satellite navigation system
- First satellite launched in 2005 and operational since end 2016
- Currently 28 satellites in orbit (24 providing service worldwide), 2 ground centres, 15 remote sites worldwide
- More than 3.5bn users around the world (meaning navigation receivers with Galileo embedded)
- … and best-in-class navigation system today

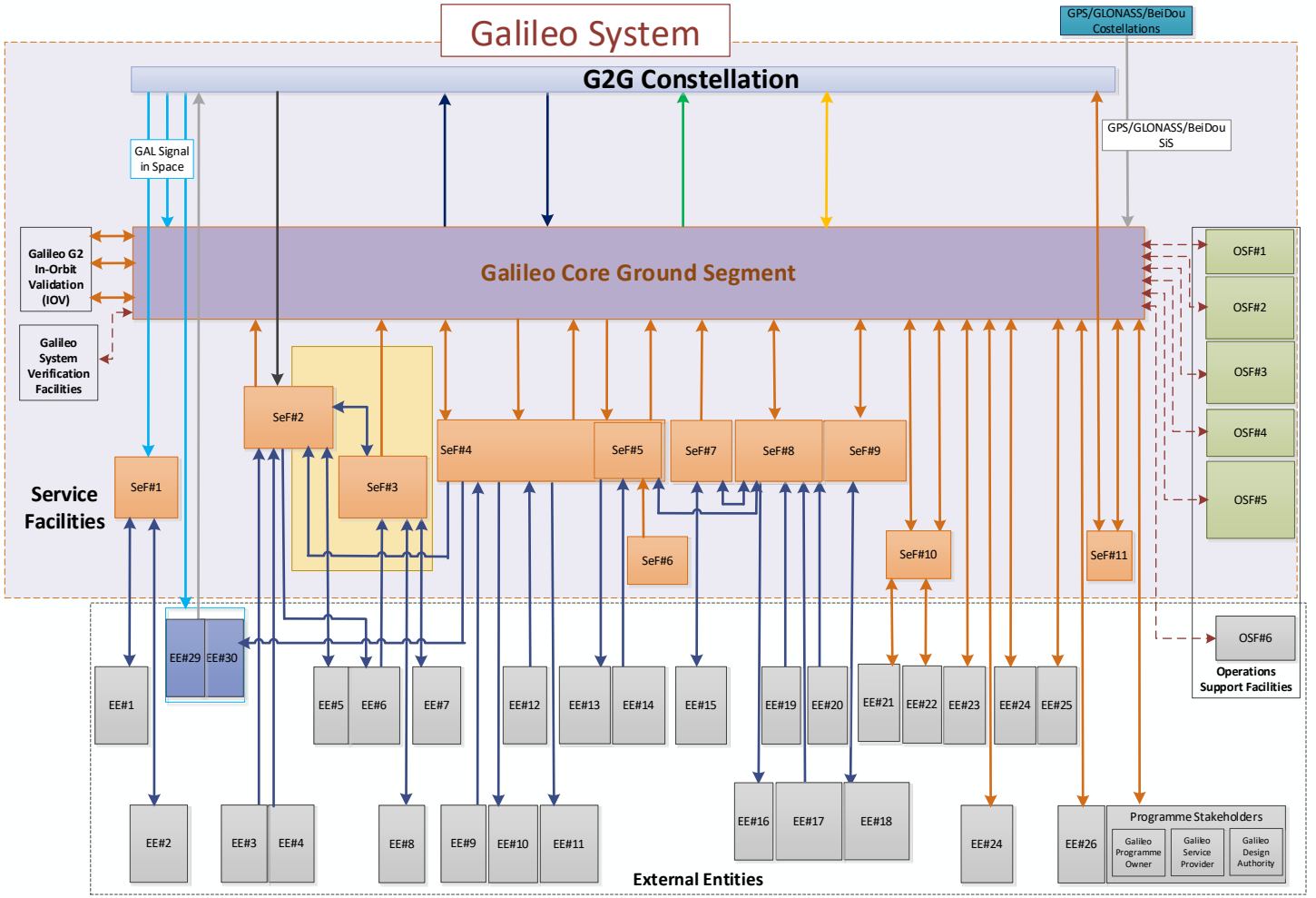
Studies for Galileo Second Generation (G2G) started in 2013 with the aim to define what comes after the full completion of the first generation as an evolution of the system in operation. Today we have:

- G2 System PDR on-going
- 2 contracts running for the first batch of the next generation satellites
- 3 parallel ground segment contracts in Phase B with some pre-development

G2G in context – design complexity overview

In terms of architecture of the system:

- A satellite constellation
- Distributed core ground segment with sites worldwide in order to collect data that contribute to the generation of navigation products (closed loop system), to perform the monitoring and control (including security aspects)
- Several services facilities on ground (in Europe)
- And Interfaces with many external entities (worldwide)



G2G in context – modelling need

The design complexity is to be mastered all through the design cycle, from specification to development, qualification and operations.

Galileo is also a long-living programme with successive versions of the system in operations

➔ model-based methodology is a must to ensure consistency and completeness of the design, including interfaces between segments and system operations.

➔ Use of MBSE was experimented in late Phase A of G2G and started in full in Phase B1.

But the idea of MBSE was not new to Galileo program:

- An MBSE approach had been experimented in Galileo until 2009 but had been abandoned as too complex to maintain. (trend to over-specify by going deep inside the procured component black boxes and the inadequate tooling available at that time).
- Enterprise Architect (EA) has been used for some years to support activities related to security accreditation. However, the development made was more bottom-up trying to represent with the model the reality of the implemented system (thus more a kind of physical model).

What has been done in G2G system activities

Choice of a tool: In 2018 we chose Team4Capella as the most promising and quicker to use for non-MBSE experts (T4C installed on VM with dedicated server remotely accessible)

Training of the team and set-up of first projects: set of engineers to become the developers trained as a group and who developed first attempts of the system model together sharing their experience and their methodology in weekly progress meetings

Verification by an MBSE expert of our initial project (in Capella at system analysis level before transition to the logical analysis): consistency of the approach was checked, readiness to go to the next level confirmed but we were going sometimes already too deep in our system analysis.

Production of a first (unclassified) model for our SRR and some additional separated models for the classified aspects

Decision to rebuild in full the model in Phase B2 towards system PDR:

- To simplify the system analysis and move the details to the logical analysis
- To extend our system perimeter and define new interfaces based on SRR decisions
- To construct a fully coordinated and synchronised view of the end-to-end system with its 4 security classification branches (security branches are developed on dedicated IT infrastructure)

What has been done in G2G system activities

Result today is a 5-branches model fully synchronised to the first branch that represents our unclassified model, the 4 other branches being the 4 classified modules (different security classifications and need-to-know).

Reference model: the unclassified part

Note that all system functions are included in the reference as well as all the logical components;

Each classified model is derived from the unclassified reference

Each SEC.i branch is developed concurrently starting from the unclassified

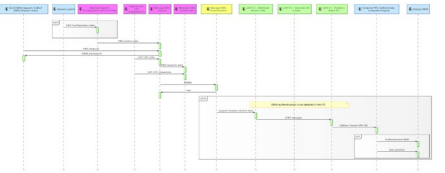
Additional logical functions, functional scenarios, exchanges are defined but no more system function or logical component

The unclassified reference has been modified and expanded while each SEC.i was developed so there is a need to resynchronise all SEC.i branches to the reference (process on-going – models are not on the same server and need to respect classification/NTK rules for each SEC.i).

What has been done in G2G system activities

Documentation is necessary for review especially when the source of the model cannot be shared or people are not familiar with MBSE.

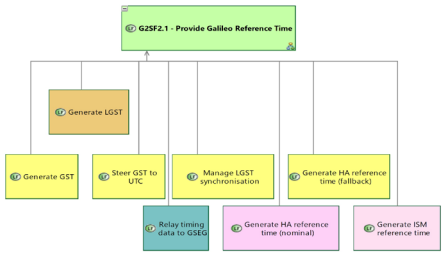
M2Doc is not easy at start but gives very good results. Scripts allow to export from the model on a systematic basis some information (diagrams or text or summary tables) for all functions or capabilities.

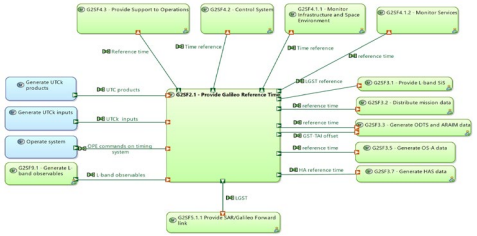


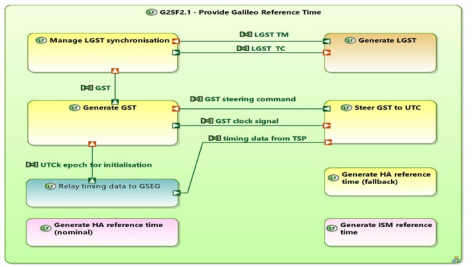
Benefits observed for the developers:

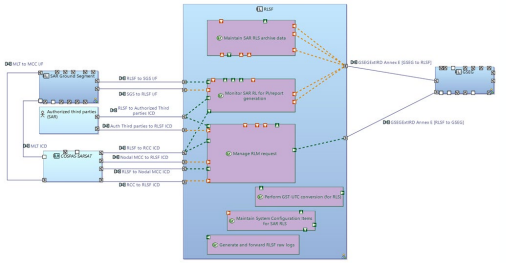
- Harmonisation of key elements developed in the model (e.g. figures LFBD, SDFB, LDFB, LAB, FS)
- Text description of what we are representing in the different figures
- Detailed description of each exchange between functions (see example table)
- Ease the co-development thanks to the descriptive text
- Partial exports

Benefits for the readers: systematic structure and set of information









System Exchange Item	Description
EWM	The EWM is a sequence of 122 bits disseminated by the Galileo System in E1b and E5b frequencies in one of the I-NAV pages that can be overwritten as defined in the G2G SiS ICD [RD-SISICD]. The EWM contains information about an event to warn the population and provide guidance. It is disseminated over a large area and is displayed only in the target area coded in the EWM message. Each EWM is disseminated via at least two satellites and is repeated [RD-GSREQ] with a repetition rate between 1 and 60 minutes assigned by the EWSPC. The EWMs are authenticated either on board or on ground, in line with the system configuration.
EWM transmission status	The EWM transmission status [RD-GSEGExtIRD] is transmitted from the ground segment to the EWSPC for each ongoing EWM dissemination and contains the following information: • EWM request identifier • Uplink status of the EWM: time of release to the ULS or TTC stations for uplink of all EWM messages at each satellite and the target Satellite [Distribute mission data] • Collection status by the ground segment: times, SV ID, GSS ID of all EWM collections **[Generate L-band observables]** and EWM dissemination monitoring status (content and latency check outcome) **[Monitor Message Dissemination]** • termination status indicating that the dissemination has finished and the trigger for its termination
EWMCR	The EWM Cancellation Request is a real time dataflow generated at the EWSPC and sent to the ground segment to trigger the cancellation of the dissemination of an on going event. It contains: • EWMCR ID • EWMR ID to which the cancellation refers to
EWMR	The Emergency Warning Request for Dissemination is the dataflow generated at the EWSPC and sent in real-time to the ground segment to trigger the dissemination of the EWM. The EWMR contains: • EWMR identifier (ID) • EWM • Type of request • Support data for dissemination (event priority, event target area, criteria for satellite selection for EWM dissemination, event dissemination duration, EWM repetition rate)

G2G (upcoming) use cases

- ✓ Use case 1 (starting point): establish and maintain the end-to-end system view as a supporting tool for ESA being system design authority for Galileo Second Generation; [first version completed including synchronisation of the branches with the reference; concatenation of all branches on-going]
- ❑ Use case 2: improve the interactions between system and segment contracts (and suppliers) beyond the traditional requirements baseline; [not started but high-level process defined]
- ❑ Use case 3: perform impact assessments related to changes requested at mission level or stemming from segments feedback; [not started]
- ❑ Use case 4: establish and maintain several system configurations along the time line of development, deployment and operations (as specified, as built, as operated); [not started]
- ➢ Use case 5: establish and maintain several system versions to be specified for the successive system releases; [on-going]
- ➢ Use Case 6: maintain traceability between model(s) and requirements baseline at all stages of the design for all versions. [on-going]

Finding#1: Tools

A system end-to-end view has been modelled for G2G and ensures having a full coherent functional design (T4C).

But some other aspects need to be analysed through the system activities:

- Many segments (space, ground but also service facilities) that may need to be more detailed/in-depth developed;
- End-to-end service behaviours that make use of 2 or more segments
- Handover/switch-over of functional behaviours depending on operational circumstances
- Data dissemination spanning across several segments
- Traceability between the functions and the requirements at system and segment levels

What we already did so far:

- Some fast-track model development is identified as necessary to get information into the decision-making process. Some modelling with **Innoslate** was done e.g. for one specific means of data dissemination across the system.
- Dynamic traceability of the functions with the requirements (system, space, ground done and service facilities to come) using **Reqtify** (establishing links between **T4C** and **DOORS**)

The one source-of-truth of the model in T4C but other tools are considered. Most likely the best way forward is to keep this **multiple-tools organisation**.

Finding#2: Impact assessment

G2G is a long-living system.

Requirements, even at mission level, evolve along the design, development and deployment phases.

Several versions of the system may be deployed and thus maintained at engineering level.

Process expected ultimately:

Mission evolution → system evolution analysis with the model and definition of requirements changes → flow to the impacted segments in terms of models (at segment level (at high-level/system analysis of the segment) and its requirements

Means to facilitate the exchange of models between the system prime and the segments suppliers/manufacturer without imposing a common tool is deemed necessary.

But we want to keep benefits of the modelling at the same time.

Details of level of exchanges and data types need to be defined

Vice versa injecting results from the segments may be injected at system level model to reflect the “as designed” and compare with the “as specified” model.

Finding#3: Challenging configuration management

Today the tools we have show limitations wrt our configuration management needs (see next slides)

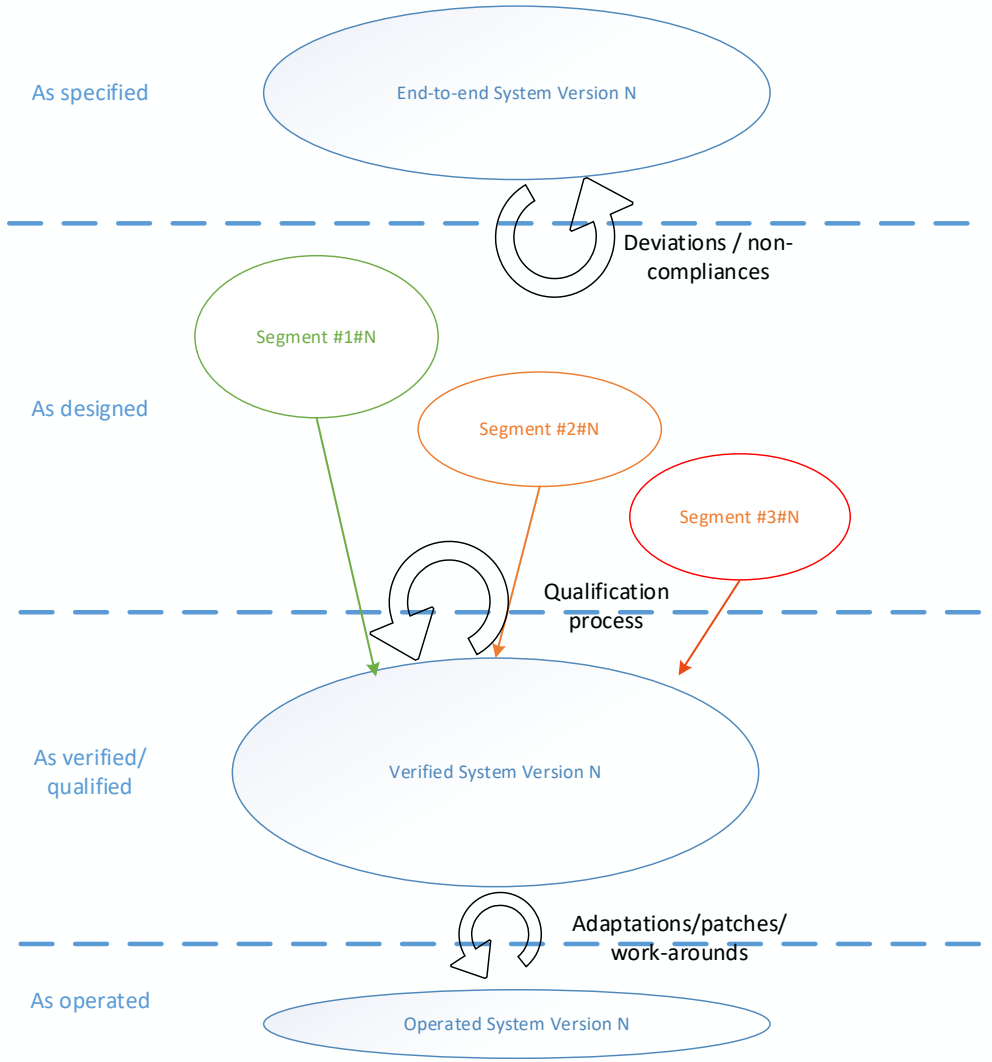
The system changes to handle may not be a functional level, but in terms of behaviours of functions, performance, or data content.

Practical ways to show

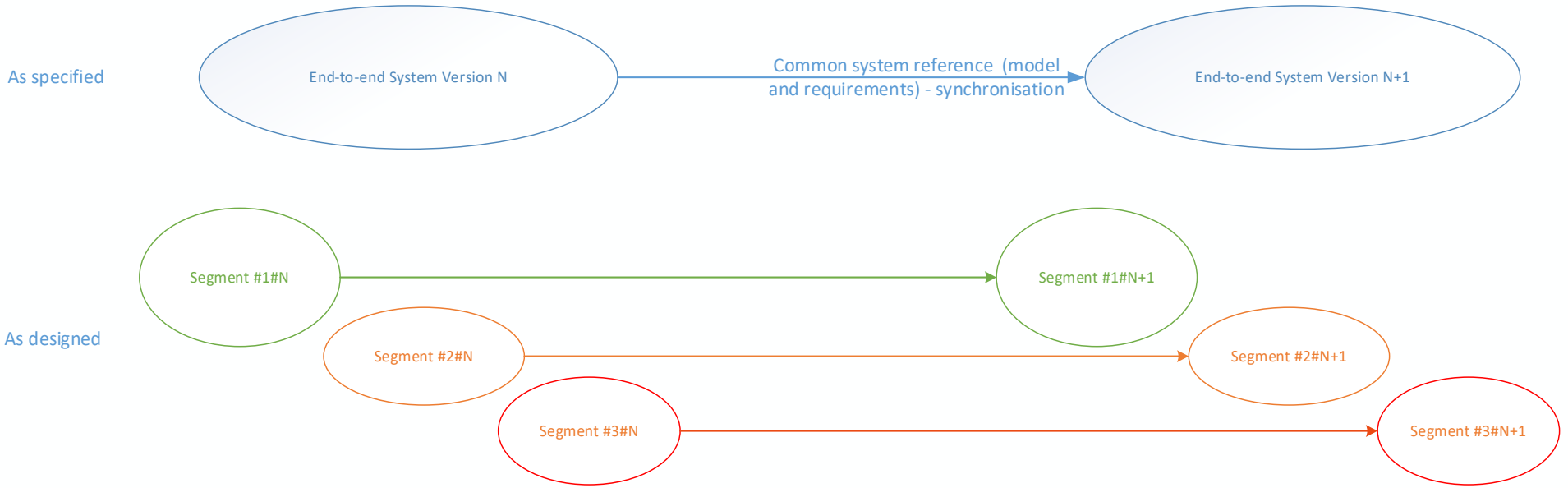
- What is unchanged
- What is new
- What is modified

from one version to another is what we need

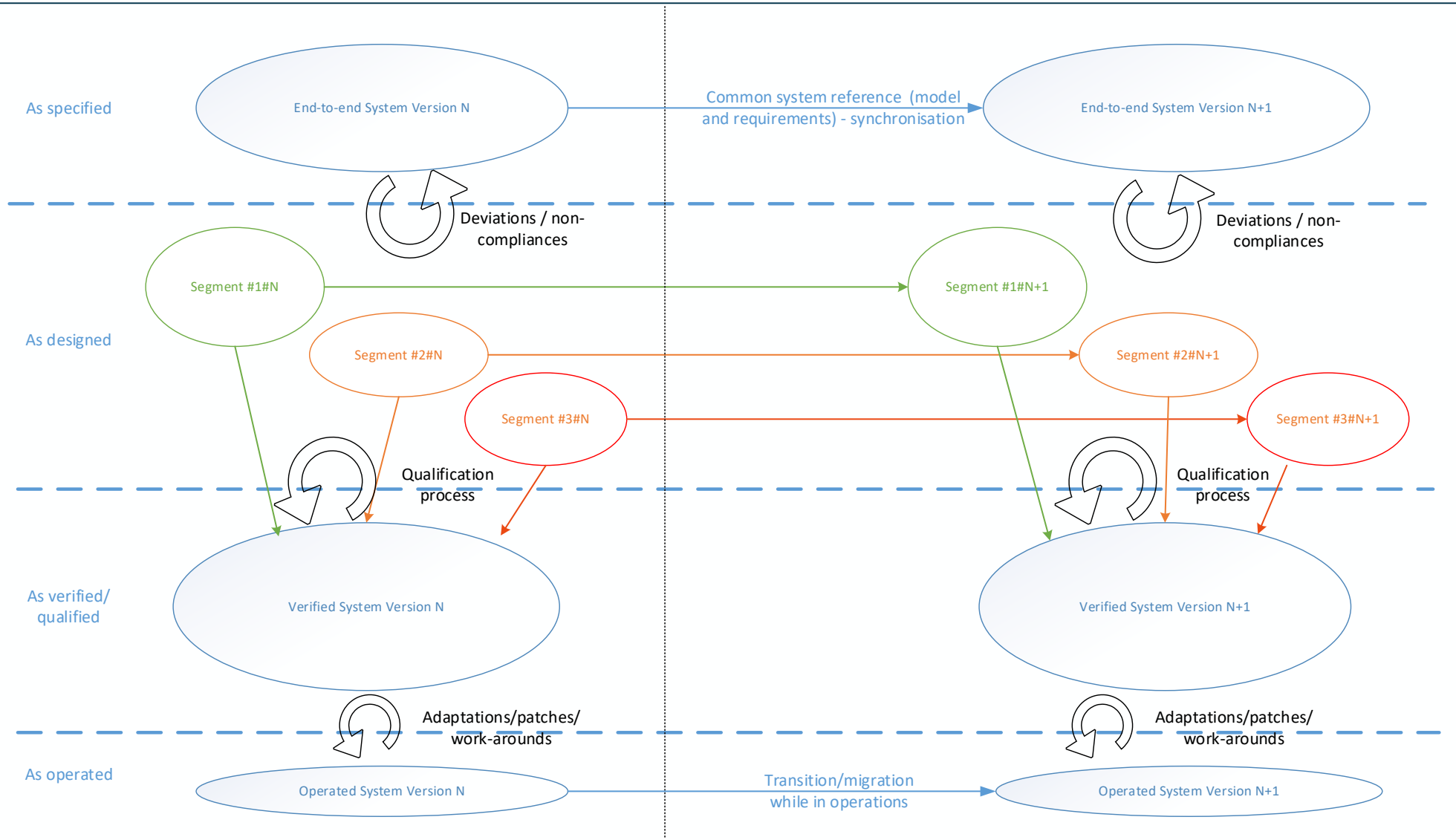
Finding#3 cont'd: the lifetime of Version N



Finding#3 cont'd: the evolution of a given design level



Finding#3 cont'd: the overall landscape of system versions



Concluding remarks and way forward

Finalising Phase B for G2G with

- A model of the specified system for a time horizon post 2030
- A model being derived for the system to be in operations in 3-4 years from now
- Dynamic requirements tracing between the functional model and the requirements baseline being implemented

We want to go further with

- Several gradual end-to-end system versions with good configuration management
- Traceability between model and ICDs/IRDs
- Links between model and performance models
- Foster model-interactions rather than requirements interactions between system and sub-systems

… and we need more from the **tools** (plural !) in the future

Thank you !

Any questions?

Paper prepared with:

C.Ruta, R.Dall'Ora (Thales Alenia Space Italia)

T.Bey (Airbus Defence and Space)

M.Chattou (Thales SIX)

R.Dellago, G.Lopez Risueno, S.Bouchired, M. Manteiga Bautista (ESA/ESTEC – NAV Directorate)

A.Gonzalez Fernandez (ESA/ESTEC – TEC Directorate)