

IRT Saint Exupéry LIV- S085L02-024 IRT SystemX ISX-S2C-DOC-437 Version: V6

Behavioral Scoped Review Method

DATE: 28/10/2022

Summary

This document aims to explicit and to explain the method called "behavioral Scoped Review" (BSR). This method is introduced by the LIV-S085L02-007-V6, ISX-S2C-LIV-1037-V6 document of S2C project. It is based on the concept of reviewing the behaviors delimited locally on two distinct models then status on their consistency. The method is illustrated by proofs-of-concept (PoC) based upon SE and SA models to experiment, determine and validate the method in the PoC's domain of usage.

Author(s)	Function(s) & name(s)	Research engineer IRT Saint Exupéry	S. Guilmeau
Checker(s)	Function(s) & name(s)	Head of Systems Engineering Centre of Competence IRT Saint Exupéry	J. Baclet
Approver	Function & name	Project leader IRT Saint Exupéry	J. Perrin

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRIT, CNRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MBDA.



IRT Saint Exupéry LIV- S085L02-024 IRT SystemX ISX-S2C-DOC-437 Version: V6

Table of Contents

Evolutions	4
Table of figures	5
Table of tables	
1 Introduction	
1.1 Purpose of document	
1.2 Referenced documents	
1.2.1 S2C reference documents	
1.2.2 External reference documents	
2 Objectives, constraints and context	
2.1 Objectives	
2.2 Constraints	
2.2.1 Common constraints	
2.2.2 Specific constraints	9
2.3 Context	10
2.3.1 Common context	10
2.3.2 Specific context	10
3 Developed method	12
3.1 Process	12
3.1.1 Specification SA	13
3.1.2 Expansion of SA inputs and outputs (TAIO SA)	13
3.1.3 Specification SE	13
3.1.4 Expansion of SE inputs (TAIO SE)	13
3.1.5 Apply pollution on SE Inputs	13
3.1.6 Transformation to Inputs SE-SA	14
3.1.7 Compute SE output with polluted inputs	14
3.1.8 Transformation to Outputs SE~SA	14
3.1.9 Remove same combinations or select one	14
3.1.10 Comparison	
3.2 Intra-process validation activities	
3.3 Iterations	
4 Deductible facts before PoCs are done	15
5 PoQs	16
5.1 PoC1	
5.1.1 SA Model and specification	17
5.1.2 Expansion of SA Inputs and Outputs (TAIO SA)	17
5.1.3 SE Model and specification	
5.1.4 Expansion of SE Inputs (TAIO SE)	19
5.1.5 Apply pollution on SE Inputs.	
This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRIT, ONRS, Airbus Defence & Space, Dassault Aviation, Tha Samares Engineering, DGA, ONERA, SupMeca and MBDA.	les AVS, Thales SA, Liebherr, LGM, APSYS,



5.1.6

5.1.7

5.1.8

5.1.9

5.1.10

5.1.11

5.2.1

5.2.2

5.2.3

5.2.4 5.2.5

5.2.6

5.2.7

5.2.8

5.2.9

5.2.10

5.2.11

5.2

IRT Saint Exupéry LIV- S085L02-024 IRT SystemX ISX-S2C-DOC-437 Version: V6



IRT Saint Exupéry LIV- S085L02-024 IRT SystemX ISX-S2C-DOC-437 Version: V6

Evolutions

Version	Date	Modified §	Modification summary	Modified by
1	21/02/2022	All	Creation	AGB/SG
2	17/03/2022	See technical Note	NT-S085L02T00-029	AGB/SG
3	08/04/2022	See technical Note	NT-S085L02T00-035	AGB/SG
4	25/08/2022	Add § 5.2 §2.1 §2.2.1	Add PoC2 Add point on structure aspect like BCC Remove redundant information already present in document LIV-S085L02-007-V4, ISX-S2C-LIV-1037- V4	32
5	14/09/2022	See technical Note	NT-S085L02T00-033	SG
6	28/10/2022	§1.1.2 §2.2.2	Update (minor) document references Change inter method reference to TOP reference	SG



IRT Saint Exupéry LIV- S085L02-024 IRT SystemX ISX-S2C-DOC-437 Version: V6

Table of figures

Figure 1: Different strategies of behavioral scoped consistency checking	10
Figure 2: Transformations to an intermediate level for comparison	11
Figure 3: Transformations when aligned with one of the domains (a) (b) (c) (d) (e) (f) (g)	11
Figure 4: Process underlying the method when transformation alignment is on SA domain	
Figure 5 : CL24 perimeter	16
Figure 6: AltaRica Code in CL24 SmfiaNeo function	17
Figure 7: Expansion of SA Inputs and Outputs (TAIO SA)	17
Figure 8: SE specification	18
Figure 9: Expansion of SE Inputs (TAIO SE)	19
Figure 10: Apply pollution on SE Inputs	20
Figure 11: addition of row to insert pollution	21
Figure 12: Populate row inserted with pollution	21
Figure 13: Inputs SE-SA before calculation	22
Figure 14: Inputs SE-SA after calculation.	23
Figure 15: Insertion and propagation of non-polluted output value	23
Figure 16: Insertion and calculation of polluted outputs	24
Figure 17: Insertion of implicit validity for outputs	24
Figure 18: Calculation of output SE-SA	25
Figure 19: Table SE~SA before reduction	26
Figure 20: Table SE~SA after reduction	26
Figure 21: Example of merge by SE-SA input vectors	26
Figure 22: Sde by side comparison of TAIOs	27
Figure 23: 0.41 perimeter represented with SSR tool (up SE abstracted view, bottom SA abstracted view)	29
Figure 24: AltaRica Code in QL41 SmfiaNeo function	
Figure 25: SmfiaNeo Truth Table feature acces	
Figure 26: Expansion of SA Inputs and Outputs (TAIO SA)	
Figure 27: SE synopsis	32
Figure 28: SE component in SMFIA NEO	32
Figure 29: Code patterns avoiding impossible case	33
Figure 30: PoC2 summary	



IRT Saint Exupéry LIV- S085L02-024 IRT SystemX ISX-S2C-DOC-437 Version: V6

Table of tables

Table 2: External reference documents. .7 Table 3: Sequencing of method .9 Table 4: Scoped behavioral consistency technics mitigation .10 Table 5: Versions of models. .16 Table 6: SE to SA Conversion used .22 Table 7: SE specification for QL41 .31 Table 8: SE TAIO expansion .33	Table 1: S2Creference documents	7
Table 4: Scoped behavioral consistency technics mitigation 10 Table 5: Versions of models 16 Table 6: SE to SA Conversion used 22 Table 7: SE specification for QL41 31	Table 2: External reference documents	7
Table 5: Versions of models	Table 3: Sequencing of method	9
Table 6: SE to SA Conversion used	Table 4: Scoped behavioral consistency technics mitigation	10
Table 7: SE specification for QL41	Table 5: Versions of models	16
	Table 6: SE to SA Conversion used	
Table 8: SE TAIO expansion	Table 7: SE specification for QL41	
	Table 8: SE TAIO expansion	

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRT, CNRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APS/S, Samares Engineering, DGA, ONERA, SupMeca and MBDA.

1 Introduction

1.1 *Purpose of document*

This document aims to explain the BSR method and to illustrate its use through at least one PoC. The working group dealing with BSR is the second one (called lot2) defined as per document CDP-S085-063-V0. As announced and explained in Section 2.3 of LIV-S085L02-007-V3, ISX-S2C-LIV-1037-V3, the document includes the following parts:

Section 2 exposes the context, objective and constraints that method shall consider,

Section 3 exposes the method and its adaptation regarding what the previous section exposes,

Section 4 exposes for esceable consequences of method as intermediate conclusion before Proof of Concepts are done. Section 5 exposes PoCs led during the working group.

1.2 Referenced documents

1.2.1 S2C reference documents

Title	Reference
State of the Art of the S2CProject	LIV-S085L01-001-V2, ISX-S2C-LIV-1001-V2
Method to ensure and to maintain consistency of systemic levels & Validation report MBSE/MBSA consistency	UV-S085L02-007-V6, ISX-S2C-UV-1037-V6
CONTRAT DE PROJET DE RECHERCHE EN PROPRE Pour la réalisation du PROJET S2C System & Safety Continuity	CDP-S-085-063-V0

1.2.2 External reference documents

Title	Reference
Aerospace Recommended Practice - Guidelines and Methods for Conducting the Safety Assessment Process on Gvil Airborne Systems and Equipment, 1996	ARP4761
Aerospace Recommended Practice - Guidelines For Development Of Ovil Aircraft and Systems, Revision A, 2010	ARP4754A



2 Objectives, constraints and context

2.1 Objectives

The items #id-5 and #id-6 of Table 6 of LIV-S085L02-007-V6, ISX-S2C-LIV-1037-V6 are allocated to BSR to give a sc status checks and . That means:

• #id-a Scoped BSR to consider a delimited perimeter in which the behavior , granularity and

precedence (or not) of another method. De facto, the consistency between each local perimeters is the target of this method.

- #id-b checks and art of the statement imposes BSR to (manually or numerically) exercise then compare the behavior delimited by the perimeter and to judge the discrepancies. This point relies either on quality of service of authoring or rigor of human performing the activities. The sub part: to get the data allowing complete behavior cases part: parts aims to consolidate data to judge inconsistencies.
- #id-c the execution. As the execution of the model is local information about its structure: what are (or wha during execution" is meaningless. However, information on the behavior rely on what are (or what shall be local

2.2 Constraints

2.2.1 Common constraints

refinement in Table 5 of LIV-S085L02-007-V6, ISX-S2C-LIV-1037-V6, are reused to trace compliance against this method all along the document so **#id-d**, **#id-e**, **#id-f**, **#id-g**, **#id-j**, **#id-k**, **#id-h**, **#id-m**, **#id-n**, **#id-o** are use for tracing. Note: **#id-e** does not forbid that one model can be historically derived from the other (e.g. when populating for the first version).

Note: The method takes care of constraint **#id-g**, because discarding it will make it easier to constraint users to model in such manner that any method could be artificially applicable.



2.2.2 Specific constraints

Constraint #id-p: as stated in Figure 4 of LIV-S085L02-007-V6, ISX-S2C-LIV-1037-V6, the designed method can interface and intertwine with other consistency methods. That means method shall consider following combinations

before	then	BSR
SSR, LIV-S085L02-007-V6, 1037-V6	ISX-S2C-∐V-	#sq-1
BSR		#sq-0

#sq-0 means method is done standalone i.e. no other method used before it.

BSR is concerned by following sequences:

#sq-1: conciliated structures done during SSR can be re-used to foster BSR perimeter delimitation as SSR ensures a strict partitioning and coverage of the models. Conciliated flows are also useful because they will help as a basis to explicit the input/output adaptation required to ensure consistency through BSR method.

#sq-6: as BOC , it does not bother a lot with internal structuration of models, it will poorly help on the perimeter delimitation but the association of variables can be a basis for the interface adaptation between models required by BSR.

Note: the most sensible sequence is to start with SSR, then BSR then BOC to take, gradually, advantage of each previous method. Other path is possible but seems, a priori, less advantageous.



2.3 Context

2.3.1 Common context

The Section 2.2 of LIV-S085L02-007-V6, ISX-S2C-LIV-1037-V6, sets the momentum where method shall be used, so PSSA (see §2.3 of ARP4761) is kept for this method.

The Section 3.1 of LIV-S085L02-007-V6, ISX-S2C-LIV-1037-V6 about position the methods against the company processes.

Notes 1, 2 in Section 2.3.1.a and Figure 3 of LIV-S085L02-007-V6, ISX-S2C-LIV-1037-V6) about FM and SofI remains applicable here

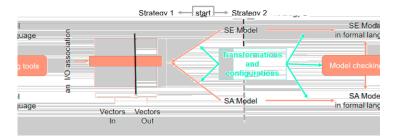
2.3.2 Specific context

The handling of the consistency on the behavior on models is more complicated than the handling of the structure, so we need to position things before to go deeper in the description of the method. The following paragraphs are intended to explicit the design method regarding the issues raised by behavior that shall be compliant with user requirements. The project has considered two strategies of comparison:

- 1) by using overlapping between tables of associations between inputs and outputs,
- 2) by using a model checking strategy that verifies properties on different models.

In both cases, each initial model (SE or SA) has to be transformed in such a way that the comparison strategy can be effective. For the first one, it is a transformation and configuration into two new models while the second one is a transformation and configuration to two tables that are compared.

These approaches are summarized as follows:



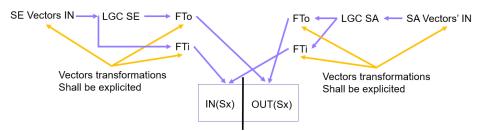
Both strategies are mitigated in the following table:

Category	Strategy 1	Strategy 2
Transformation activity	Need to transform behavior into table before passing each step of the method.	Need to pass from a model to the formal one on which property checks are applied.
Configuration activity	Need to define targeted perimeter to cover the filtering for exploitation, etc.	Need to transform a model from one technical space to another while preserving semantics.
Transformation complexity	Automatable	Automatable
Configuration complexity	Less complex but limited (as associations table cannot handle every subtleties)	More complex as projecting a semantic from one domain to another one is a hard problem.
Verbosity	More verbose as the full covering will require many lines to handle	Less verbose because infringement of check rule are more laconic. However, it requires an intellectual effort for the reader to understand the situation he/she has to correct in the original model.
Semantic	Less complex because an association table relies on a 2D matrix-perspective (no operator to know)	More complex as model is in a language hidden from user while checks are linked to this language.
Tools	Avoidable i (method modularity <mark>#id-n</mark>).	Need to handle formal language transformation and checks over each perimeter.



IRT Saint Exupéry LIV- S085L02-024 IRT SystemX ISX-S2C-DOC-437 Version: V6

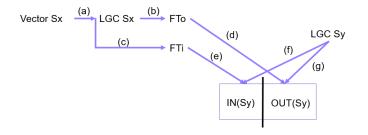
The first strategy seems convenient regarding the constraints expressed in project (**#id-j**): it is then selected. The comparison between association tables is direct (i.e. do for the same input vector I have the same output vector?) As comparison strategy is selected at this point, the problem now is the transformation of the models into tables. A generic view can be summed up as follows:



- LGC SE or LGC SA means the behavioral logic delimited by the perimeter selected. (LGC = Logic that defines the behavior).
- FTi (resp. FTo) means the transfer function that converts domain input (resp. output) into a comparable form.
- SE (resp. SA) vectors IN (resp. OUT) means the operational vector used to feed each domain logic.
- IN (Sx) (resp. OUT (Sx)) means the vectors resulting from the transformation with Sx=SE or Sx=SA.

The Elicitation of transfer functions can help during the \mathfrak{S} and \mathfrak{S} review on behavior to understand from where and how each domain gets to the compared vectors (INS (x) or OUT (\mathfrak{S} x)). This allows going further than the simple comparison of the two tables that can be correct but that may hide a domain information useful for the other one (e.g. challenge error in the reasoning of the opposite side). The OUT (\mathfrak{S} x) requires , so an execution (mental or numeric) is required, **#id-b**.

It should be noted that IN and OUT vectors of the table can be aligned on one domain to avoid a transformation for this one (at least it only remains an explicit expansion for this domain if not done). Here, the Sx or Sy can be either SE or SA depending on which one has a higher level of abstraction:



Note: the method does not forbid that the alignment is done on an intermediate level (i.e. neither SE nor SA level of abstraction). The cost will be that each domain has to do the transformation to be in the comparable domain for IN and OUT.

It should be noted that constraints **#id-f** (partially) and **#id-m** are related together as they rely on a configuration management system. Contrarily to the SSR (resp. BOC) where **Consistency Links** (CLs) (resp. scenarios/procedures and so on) do not exist in any of the domains models (because they are related to a superset of information of both of them), the intermediate artefacts of BSR can be handled

purpose). So, those artefacts can rely on configuration management tools used during development. The complement part of **#id-f** concerns the storage of capitalization (which is a superset information). A dedicated repository in previous version system tools that points the version of models can be used for that without creating a new ecosystem apart from the one of development.



3 Developed method

Section 3.3.2 of document LIV-S085L02-007-V6, ISX-S2C-LIV-1037-V6 introduces briefly the BSR. So following sections aim to detail it, to define used words and their relations against its process and to trace method artefacts against constraints exposed in Section 2.2.

The first Section exposes the activities composing the method while all following sections will describe each activity.

3.1 Process

The method focuses on the consistency between behaviors whose perimeters are delimited in both input models.

This means that this method aims at comparing SE and SA behaviors, both delimited by a scope.

So, the global behavior in one domain is partitioned into perimeters (resulting of the aggregation of structural items of model) each of which is linked to another one in the opposite domain. It is similar to SSR but it also covers the behavior and is not limited to the structure and interfaces.

In order to compare behavior, the method establishes what is

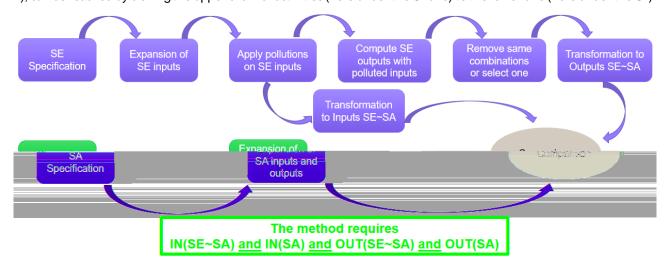
Outputs). They apply to the local leaves functions of the SE and SA, to observe possible inconsistencies between their behaviors; see Figure 1.

This method is meant to be systematic and exhaustive on each or on a given perimeter (it is not exhaustive from the whole models behaviors). Those two properties are generally not achieved while FTA are reviewed (traceability between behaviors from \mathfrak{E} and FTA is not usually performed).

Section 2.3.2 explains that, one of the main problem addressed by the method lays on the fact that domains and implemented logics for the same perimeter (defined by the structure) are usually different for \mathfrak{E} and SA specialists (due to their intrinsic needs). Thus, TAIO comparison between \mathfrak{E} and SA is not obvious, except if a coupled modelling methodology is done between \mathfrak{E} and SA specialist when they do their respective models, which is not the constraint given (see **#id-g**). In order to resolve this issue, the method proposes to perform some transformations to get both \mathfrak{E} and SA models closer to each other and to be able to compare local behaviors (limited to what is common as stated in Figure 3).

To illustrate the process, we will present the specific case of Figure 3

SA (so alignment is done on SA domain in this example). The generic situation (i.e. intermediate alignment, see Figure 2), can be reached by cloning the upper chain of activities (here under the SE one) to the lower one (here under the SA).



This alignment is selected because, opportunistically:

- The SA model on the analyzed perimeter is more abstract in terms of structural decomposition than SE one (best case is 1 for 1, the worst is 1 for N).
- The SA model on the analyzed perimeter is more abstract in terms of Inputs/Outputs than SE (best case is 1 flow for 1 flow).

The following paragraphs describe each one of the activities presented in Figure 4 more detailed example of these activities instantiated in a concrete case.

PoC1

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRIT, CNRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MEDA.

Licence Creative Commons Attribution-ShareAlike 4.0 International (CCBY-SA 4.0)



3.1.1 Specification SA

For the associated functions on SA side, on defined perimeter, the specialist specifies its behavior by writing AltaRica code (AR) (induced by the S2C project choice on MBSA defined in Section 2.4 of LIV-S085L02-007-V6, ISX-S2C-LIV-1037-V6). It defines the output values of the function will take depending on the internal state and values on its inputs in SA domain. This is not specific to the method, it is the SA specialist work when he/she does its model with Failure propagation strategy.

Note: Even if SA specification deals with both SofI!=OK and SofI==OK (see Figure 3), the specification taken into account for the BSR method only focuses on the nominal scenario SofI==OK (as already stated in Section 2.3.2).

3.1.2 Expansion of SA inputs and outputs (TAIO SA)

The Expansion of SA inputs and outputs aims to create desired combinations (having all combinations is the best) for the input vector (f) by taking from input the variables and their domains from their specification. Once the input vector is defined, the output vector is calculated for the nominal case (SofI!=OK are not considered because they have no meaning in the \mathfrak{E} model) for the whole input vector by applying the logic implemented by SA (g).

3.1.3 Specification SE

For the Æ-leaf function that will take part in the analysis in the model, Æ must specify its behavior. In other words, it must status the values the outputs of the function will take depending on the values on its inputs in Æ domains. This specification can be done in various ways (i.e. logic equations, text describing the function behavior, implicit truth tables, etc.). It should be noted that:

- None of these possibilities gives an explicit and exhaustive implementation of the behavior of the function.
- The SE behavior defined is functional with dysfunctional logic detection (not for all function, as only some are concerned by detection) that detects failure propagation via the system. This behavior does not consider that the logic (functional associated to some failure detection sometime) can fail (contrarily to SA which considers SofI, see Figure 3)

3.1.4 Expansion of SE inputs (TAIO SE)

As for 3.1.2 Expansion of SA inputs and outputs, all possible combinations by using the values of various ædomain input variables are created to obtain the Input vector (c).

If a variable of the Input vector does not have an associated validity, an implicit one has to be added (i.e. it is considered Valid in all combination). The existence of a validity associated to a variable is required to foster the association with SA domain of values in further steps of the method (implicitly required for SA activities for example when defining what a LOST is).

3.1.5 Apply pollution on SE Inputs

For the Input vector defined in Section 3.1.4, it is needed to determine which pattern of pollution shall be applied to transform the SE values to reach the domain of values expected from SA domain (otherwise convergence and therefore comparison cannot be done)(a)(c). Hereafter are some examples of pollution to apply regarding what SE considers that SA has done in its model (induced by **#id-e** and **#id-j**):

- Non corruptible, the value will be not polluted in the combination
- Healthy/Polluted, the value will be either keep clean or polluted (binary case, e.g. for validity if required)

In the method, the validity in SA domain cannot be corrupted because implicitly, SA converts (in his mind when creating its model) the SE validity and pollution in LOST and ERR and operational values (often domain's value is OK but not always). This activity aims to prepare and explicit (by choosing the pollution and handling validity) how the SE will be transformed into SA domain (next activity).

If the user wants to consider this validity is corruptible, then the validity must be treated as a polluted variable (with a

analysis to handle more items.

This step of determining a corruption rule for the input vector is a configuration the SE specialist must make explicit: elements are not exclusive to be able to apply transformation rules SE to SE-SA in next step. By performing this step, cases are multiplied depending on the possible number of corruptions determined for a given domain.

Note: we do not expect many patterns because they are to be reused for the different perimeters. Using the same pattern all over the models will ensure greater consistency than applying to many different patterns

combinations to treat and requires the



3.1.6 Transformation to Inputs SE~SA

This step of the process takes the previously polluted input vector expressed in SE domain and multiplex them (if needed), to get the input vector expressed in SA domain, this is called the application of FTi (e). The multiplexing is ruled as follows:

- Rule: for one single SA data traceable on just one single SE data, its associated validity must be considered for the analysis.
- Rule: for one single SA data resulting from the combination of several SE data, the following transformation rules SE-SAN to 1 must be applied:
 - \rightarrow If at least one variable among N is LOST \rightarrow result is LOST
 - \rightarrow If at least one variable among N is ERR (and there is not any LOST) \rightarrow result is ERR
 - \rightarrow If all the N variable are OK \rightarrow result is OK

Therefore, for multivariable cases, multiplexing logic to be apply must be selected. Moreover, it must be determined if the selected multiplexing logic is applicable locally (i.e. another perimeter can have different computation than this one) or also globally for the whole model (i.e. the same computation has to be done for all perimeter).

3.1.7 Compute SE output with polluted inputs

For the calculation of SE Output vector, the SE implemented logic defined in the specification is used (b). The output vector must be calculated for the entire explicit table by determining the value for the polluted and non-polluted scenarios for a given input vector from SE explicit TAIO.

The realization of this step enables to know the sensitivity of the implemented \mathfrak{E} logic towards the pollution rules defined. To illustrate, in the case of non-linearity of \mathfrak{E} logic, e.g. like presence of a switch, one single polluted input data may not have effect on the output because the input is not the one switched (this is a kind of dormant pollution).

3.1.8 Transformation to Outputs SE~SA

Once Se output vector is obtained for polluted and non-polluted scenarios for a given input vector from the Se explicit TAIO, the Output vector Se must be expressed in SA domain by applying FTo (d) in order to be able to be compared after with SA Output vector. FTo intends to compare the value obtained for polluted and non-polluted outputs for the same scenario and to status a result in SA domain.

Rules regarding FTo must be defined to correlate the values obtained in polluted and non-polluted Seoutputs and Se-SA output value, taking into account the domains available for each specialist.

3.1.9 Remove same combinations or select one

Performing the Apply pollution on E Inputs ing the cases to be treated and consequently having repeated cases in the intermediate table of the analysis (i.e. systematic application of pollution creates a large number of vectors that lead to the same result in the end). Thus, there is the need to factorize the repeated cases into a single one in order to compare them after with the SA explicit TAIO.

All \mathfrak{E} -SA input vectors sharing the same \mathfrak{E} -SA Input values are merged into a single case. After this merge, the output takes the output \mathfrak{E} -SA value or the values (since it is possible that different output values \mathfrak{E} -SA were identified for a same type of input vector \mathfrak{E} -SA). The fact that different output values are obtained is that different pollutions do not lead always to the same effect and after \mathfrak{E} -SA transformation the different applied pollutions are found in the same value (i.e. $\mathfrak{E} R$ (SA) corresponds to Pollution_1(\mathfrak{E}) and also for Pollution_2 (\mathfrak{E}). However, effects of Pollution_1 and Pollution_2 after applying \mathfrak{E} logics may not be the same on the output value. The problematic is that \mathfrak{E} logics are not systematically linear or symmetric and usually SA makes them linear and/or symmetric for simplicity reasons that will maximize the effects.

When more than a value is identified, this case must be highlighted since it is a potential source of inconsistencies in the models.



3.1.10 Comparison

At the end of the process, two tables are obtained containing aligned inputs and outputs.

- Explicit SA TAIO
- Explicit SE~SA TAIO

Cases are compared regarding their input vectors and differences in outputs may be observed and highlighted as potential sources of inconsistencies. It should be noted that, due to the \mathfrak{E} logic implemented in the specification may differ from the SA logic, some cases may be found in \mathfrak{E} -SA table that have no match in the SA table. Those cases should also to be highlighted, as some behaviors are apparently not considered by SA that at least must be discussed in a \mathfrak{E} and SA review to extract further conclusions or justify the observed difference.

3.2 Intra-process validation activities

Contrarily to BCC, which has many intermediate artefacts that can be corrupted, the BSR only has the configurations of transformations that can introduce errors (e.g. by mapping part of one domain onto inconsistent part of another domain). These configurations can be audited to detect possible errors. Such audits can be part of the cross review between SE and SA or a dedicated review by a pair of the domain. Making the audit or not is at discretion of companies regarding their needs about consistency.

3.3 Iterations

When a model changes (either SE or SA or both), the process can be, in the best case, replayed (as configuration for the transformation are kept) or updated in the worst case (to handle change in interface of content of perimeter). The process does not need to keep intermediate artefacts (if done by tool) because they are computable (or re-doable manually if no tools are considered by the company).

As comparison is based upon the overlap of IN and OUT vectors, simple tools can be used to identify which can exist in one side and cannot in the other side. It can be also simple (depending on the amount of lines in table) to check for a vector if its output is the same on the other side.

The understanding of discrepancies can be done by the observation of rules to transform one domain into IN or OUT vectors.

4 Deductible facts before PoCs are done

The structural difference weighs on the efficiency of the method as it changes greatly the interface of perimeter. Therefore, if the perimeter is too different, with interfaces too different, it will jeopardize the usability of it due to the exponential computation induced.

The SE behavior within a perimeter may not be easy, if not impossible, to transform (e.g. because of timing constraints).



5 PoCs

This section describes the Proof of Concept (PoC) assessment that was conducted to validate the method described in Section Process. This assessment also aims to identify limitations of the proposed method and further declinations leading to futur PoCs.

As models were already used in the PoC of SSR method (previously done during project and matching **#sq-1**), we opportunely took this fact into account. So CL functions resulting from SSR are used as perimeters for BSR for SE and SA domains. This does not jeopardize the method assessment, it will only reduce the amount of discrepancies detected, compared to the selection of other perimeters (case of **#sq-0** or **#sq-6**).

Perimeters used (so QL selected) for assessment will progressively be more complex by focusing on dedicated topics of assessment.

Models of domains used for the method assessment are:

SEmodel	SA model	
V4.4.3	V4.4.3	

Note: See LIV-S085L02-007-V6, ISX-S2C-LIV-1037-V6 for more detail.

5.1 PoC1

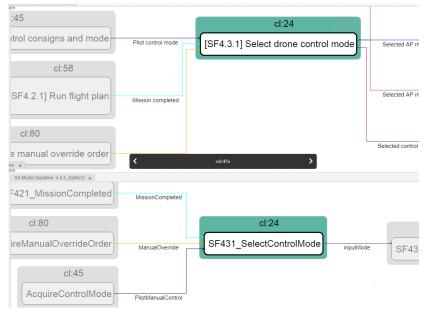
This PoC was performed onto the perimeter of CL24, based up

AIDA model. Figure 5 gives an extract from GUI of the covered perimeter for both SE and SA.

CL24 is a 1-1 CL regarding leaf functions of AIDA models (see Table 5). That means only one \mathfrak{E} function matches only one SA function in this case. Input and output flows of the functions are also 1-1 CL relation for both \mathfrak{E} and SA.

It is an assumed simple case for the method assessment because it eases its application by not overloading the inputs/outputs transformations from SE to SA as a starting point (as they are one to one).

In this PoC, activities were because the amount of interfaces and domains was judged reasonable enough. This means all TAIO were created by the user (let us say the SA specialist) using Excel file format.



On the following tables being part of PoC1, a color convention has been used that represents the following:

- Pink: input variables
- Green: output variables
- Orange: corrupted value
- Yellow: cases having at least one polluted value in its input vector

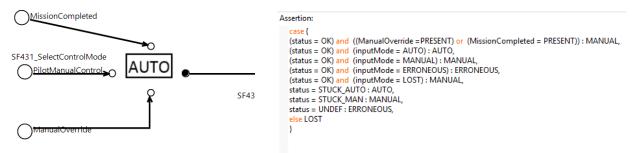
This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRIT, ONRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MBDA.



5.1.1 SA Model and specification

In Figure 6, reader will observe the modelled function in the SA tool. Inside the assertion of the function, the SmfiaNeo AR code displayed on the right side was set to show the behavioral specification of the function.

Note: the specification is formal contrarily to the SE one. This allows a way to get the SA TAIO effortlessly.



Domain values for each variable are the following:

- PilotManualControl: AUTO; MANUAL; ERRONEOUS; LOST.
- ManualOverride: PRESENT; ABSENT.
- MissionCompleted: PRESENT; ABSENT.
- OutputMode: AUTO; MANUAL; ERRONEOUS.

5.1.2 Expansion of SA Inputs and Outputs (TAIO SA)

From behavior specification of SA part of CL24 (Figure 6), all possible combinations are created with Inputs domains in Figure 7.

By using SA implemented logic, output values are calculated for each case (each line in TAIO SA).

Note: only lines having AR code in right side of Figure 6 are used to performed this step of the method because of statement in Figure 3 about Sofl.

Pilot Manual Control	Manual Override	Mission Completed	OutputMode
AUTO	PRESENT	PRESENT	MANUAL
AUTO	PRESENT	ABSENT	MANUAL
AUTO	ABSENT	PRESENT	MANUAL
AUTO	ABSENT	ABSENT	AUTO
MANUAL	PRESENT	PRESENT	MANUAL
MANUAL	PRESENT	ABSENT	MANUAL
MANUAL	ABSENT	PRESENT	MANUAL
MANUAL	ABSENT	ABSENT	MANUAL
ERRONEOUS	PRESENT	PRESENT	MANUAL
ERRONEOUS	PRESENT	ABSENT	MANUAL
ERRONEOUS	ABSENT	PRESENT	MANUAL
ERRONEOUS	ABSENT	ABSENT	ERRONEOUS
LOST	PRESENT	PRESENT	MANUAL
LOST	PRESENT	ABSENT	MANUAL
LOST	ABSENT	PRESENT	MANUAL
LOST	ABSENT	ABSENT	MANUAL

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRIT, ONRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MBDA.

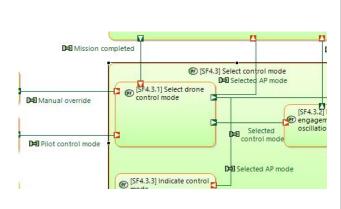


5.1.3 SE Model and specification

Figure 8 (left side) shows the modelled function in the SE tool. Inside the model description fields of this function, the table displayed on the right side serves as the specification of this function.

Note: the table is extracted from a textual specification. This extraction is more or less complex because it relies on the writing methodology and tool used by the SE specialist. Whilst the method works with table format, the cost of extraction and formatting is neither negligible nor error-free.

Note: if perimeter contains many fine grain functions with their respective behavior specifications, this extraction and formatting may not be achievable manually in a reasonable amount of time.



Pilot control mode		Manual override		Mission completed	Selected AP mode	Selected control mode
Value	Validity status	Value	Validity status	Value	Value	Value
N/A	N/A	OVERRIDE	VALID	N/A	Speed consign mode	MANUAL
N/A	N/A	N/A	INVALID	N/A	Speed consign mode	MANUAL
N/A	N/A	N/A	N/A	Completed	Speed consign mode	MANUAL
N/A	INVALID	N/A	N/A	N/A	Speed consign mode	MANUAL
MANUAL	VALID	N/A	N/A	N/A	Speed consign mode	MANUAL
Speed consign mode	VALID	NO OVERRIDE	VALID	N/A	Speed consign mode	AUTO
Flight plan mode	VALID	NO OVERRIDE	VALID	Not completed	Flight plan mode	AUTO

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRT, CNRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MBDA.



5.1.4 Expansion of SE Inputs (TAIO SE)

Like in Section 5.1.2, each SE input (columns A, C, E, G, and I from Figure 9) are mixed to create all possible combinations regarding the domains values. In the case of CL24, we have the following Inputs and their respective possible domains:

- Pilot Control Mode Value: 3 possible values (MANUAL, SOM, FPM translated into 10MD, 20MD and 30MD)
- Pilot Control Mode Validity: 2 possible values (Valid, Invalid)
- Manual Override Value: 2 possible values (Override, No Override)
- Manual Override Validity: 2 possible values (Valid, Invalid)
- Mission Completed Value: 2 possible values (Cfor computed, NCfor not computed)
- Mission Completed Implicit Validity: one possible values (Valid).

Therefore, method shall handle at this stage: 3x2x2x2x2x1 = 48 combinations (rows in the Excel file).

Note: implicit validity (always considered as Valid) must be added for the inputs not having an associated validity was complemented with an implicit validity associated

variable (column Lin Figure 9).

Note: expansion may help domain specialist to find inconsistencies (i.e. a wrong output regarding the inputs) because the specification may be factorized and some combination will not appear as explicitly as in the table.

Α	С	E	G	1	L
Pilot control	mode	Manual	override	Mission c	ompleted
Non-polluted Value 1CMD=MANUAL 2CMD=SCM 3CMD=FPM	Validity status ▼	Non-polluted Value	Validity status ▼	Non-polluted Value	Implicit Validity
1CMD	Valid	Override	Valid	C	Valid
1CMD	Valid	Override	Valid	NC	Valid
1CMD	Valid	Override	Invalid	C	Valid
1CMD	Valid	Override	Invalid	NC	Valid
1CMD	Valid	No Override	Valid	С	Valid
1CMD	Valid	No Override	Valid	NC	Valid
1CMD	Valid	No Override	Invalid	С	Valid
1CMD	Valid	No Override	Invalid	NC	Valid
1CMD	Invalid	Override	Valid	С	Valid
1CMD	Invalid	Override	Valid	NC	Valid
1CMD	Invalid	Override	Invalid	С	Valid
1CMD	Invalid	Override	Invalid	NC	Valid
1CMD	Invalid	No Override	Valid	С	Valid
1CMD	Invalid	No Override	Valid	NC	Valid
1CMD	Invalid	No Override	Invalid	С	Valid
1CMD	Invalid	No Override	Invalid	NC	Valid
2CMD	Valid	Override	Valid	С	Valid
2CMD	Valid	Override	Valid	NC	Valid
2CMD	Valid	Override	Invalid	C	Valid
2CMD	Valid	Override	Invalid	NC	Valid
2CMD	Valid	No Override	Valid	С	Valid
2CMD	Valid	No Override	Valid	NC	Valid
2CMD	Valid	No Override	Invalid	С	Valid
2CMD	Valid	No Override	Invalid	NC	Valid
2CMD	Invalid	Override	Valid	С	Valid
2CMD	Invalid	Override	Valid	NC	Valid
2CMD	Invalid	Override	Invalid	С	Valid
2CMD	Invalid	Override	Invalid	NC	Valid
2CMD	Invalid	No Override	Valid	С	Valid
2CMD	Invalid	No Override	Valid	NC	Valid
2CMD	Invalid	No Override	Invalid	С	Valid
2CMD	Invalid	No Override	Invalid	NC	Valid
3CMD	hileV	Override	Valid	C	Valid

Note: such table may be made by tools if SE specification is sufficiently formal.



5.1.5 **Apply pollution on SE Inputs.**

Each of the 48 combinations is a nominal case that can be corrupted (independently or jointly with other). Corruptible three possible values regarding its domain two possible values err regarding its domain). Therefore, 48x3x2x2 = 576

regarding its domain),

combinations have to be considered during the assessment.

If validity was considered corruptible, the total number of cases to be treated for CL24 would be: 576x2x2x2 = 4608cases. However, here, the associated validities to the polluted values considered are not corruptible at all (implicit hypothesis from SA way of working as validity is melted with the value leading to the value LOST). If the validity was finally decided to be considered as corruptible, the previous given amount of extra combination shall be considered.

In practice, we arbitrarily choose to add Overr

Polluted Val

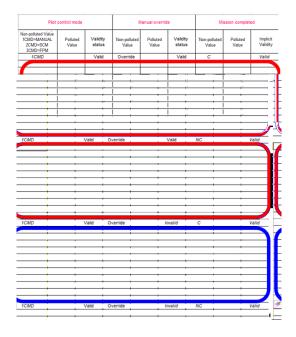
F, and Jin Figure 10) while associated validity remains unchanged.

A	в	C	Ł	F	G	I	J	L
Pilot	control mode		М	anual overrie	ie	Mi	ssion complet	ed
Non-polluted Value 1CMD=MANUAL 2CMD=SCM 3CMD=FPM	Polluted Value	Validity status	Non-polluted Value	Polluted Value	Validity status	Non-polluted Value	Polluted Value	Implicit Validity
1CMD		Valid	Override		Valid	С		Valid
1CMD		Valid	Override		Valid	NC		Valid
1CMD		Valid	Override		Invalid	С		Valid
1CMD		Valid	Override		Invalid	NC		Valid
d 1.CMD			alid : 3:5 /Nó (9/			alidi≣ IIII (2	Ve
id 1.CMD	2222		álið- <u></u> No:Ov		V		67799	Va
id 1CMD			alid No Ov		Im			Ve
id 1GMD			alid No Qu			alid N		Va
d = 1CMD		<u></u>	alid Over	ride		alid- C		Va
id 1CMD		(inv	alid Over	ride-	V	alid N		Ve
id1CMD		Inv	alid Over			alid C		Ve
id 1.CMD		Inv		ride —		alid N	•	Va
id 1CMD		inv	alid — No Ov			alid C		Ve
id 1CMD		Inv	alid No Ov	erride.		alid N		Ve
id_==== 1CMD			alid No Ovi			alid C		Va
id: 1.GMD			alid No.Ov			alid N		Ϋŧ
id 2CMD			alid Over	ride				V.e
ið 2CMD			alid Qvér				<u> </u>	Væ
id2CMD		V	alid Over	ride		ali <u>d</u> C		Ve
id 📃 🔣 2CMD				ride 🗠			<u>د نے</u>	Va
id 2CMD		V	alid No:Ov			alid C		V,t
d 2GMD			alid No Ov			alid No		Va
d 2CMD			alid No:Ove		Inv	alid C		Va
d 2CMD			alid No Ove			alid—N		Va
d 2CMD			alid Over			alid C		Vε
d 2CMD			alid Over			alidN		Va
id- 2CMD			alid Over			alid C		V٤
d 2CMD		(=	alid Over			alid N		Va
d 2CMD			alid No Ove			alid C		Va
id – 2CMD			alid No Ov			alid No		Va
d: 2CMD		(alid No Ove			alid C		Va
d 2CMD		(=	alid No Ove			alid No		Va
d = 3CMD				ride				Va
id === 3CMD			alid Over			,	Q:::=	Va
d 3CMD		(E	alid Over					Va
d 3CMD	(22.22 North		alid Over	ride	Inv	alid N		Va Va

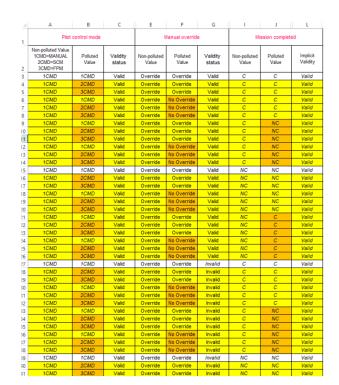
Additional rows are created to treat all the 576 combinations identified for the Input vector caused by pollution rules established in the analysis. Each added group of rows keeps the non-polluted case as the header of the group. Note: added rows can be done automatically if the domain of each variable is known.



IRT Saint Exupéry LIV- S085L02-024 IRT SystemX ISX-S2C-DOC-437 Version: V6



Combinations are done for the input vector. In Figure 12, nominal cases (first 48 identified) corrupted ones and polluted values are represented regarding color convention discussed in Section 5.1.



This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRT, CNRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MBDA.



5.1.6 Transformation Inputs SE~SA

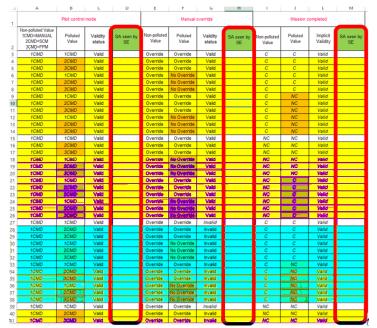
For the calculation of Input SE-SA, a transformation is needed to express the SE input variable in SA domain. In this case, variable value and validity on SE side are merged in a single data SE-SA (validity for SA is always implicit). For the calculation, - are

discarded at this stage.

Hypothesis considered in the transformation are:

If Validity is	nd Value is	hen the SA value is
	any	LOST
	polluted (orange cell)	ERR
	is not polluted (no orange cell)	value according to SA domain value

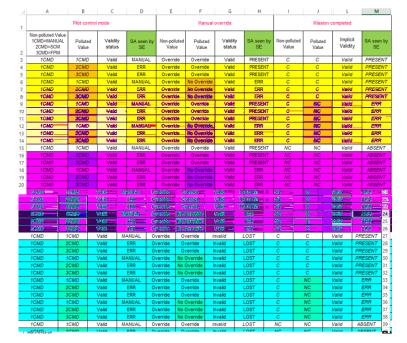
In practice, as seen in Figure 13, it is needed to add three columns in the case of CL24 (D, H, and M).



With applied rules of Table 6, columns « SA seen by SE » are completed with SA domains. For example, for QL24 values from column D are calculated by using values from columns B and C in order to determine them (see Figure 14).



IRT Saint Exupéry LIV- S085L02-024 IRT SystemX ISX-S2C-DOC-437 Version: V6



5.1.7 Compute Output SE non-polluted and polluted values

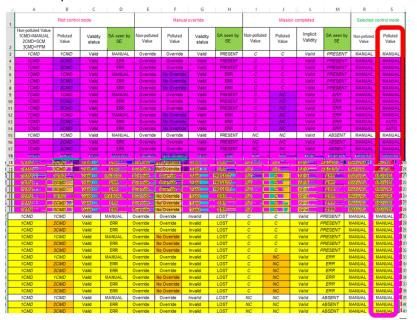
The calculation of \mathfrak{E} « Non-polluted » Output (Rcolumn in Figure 15) is done by using the « Non-polluted » Input vector (A, E, and I columns) with its associated validities (columns C, G and L non-corruptible by hypothesis). In order to determine the output value, \mathfrak{E} logic expressed in the specification is used.

.4	A	В	с	D	E	F	G	н	1	J	L	M	R
1		Pilot control	mode			Manual	override			Mission o	completed		Selected control mode
2	Non-polluted Value 1CMD=MANUAL 2CMD=SCM 3CMD=FPM	Polluted Value	Validity status	SA seen by SE	Non-polluted Value	Polluted Value	Validity status	SA seen by SE	Non-polluted Value	Polluted Value	Implicit Validity	SA seen by SE	Non-polluted Value
1.	1968	19000	. ``valo	HMANVAL	fl qverhae	0 uverhoe	· Váio	ROFAESEN	. Nv :	0	vaha	IPRESEN	II MANUAL
4	1CMD	2CMD	Valid	ERR	Override	Override	Valid	PRESENT		С	Valid	PRESENT	
5	1CMD	3CMD	Valid	FRR	Override	Override	Valid	PRESENT	с –	C	Valid	PRESENT	MANUAL
Г													
Г													
Γ.													

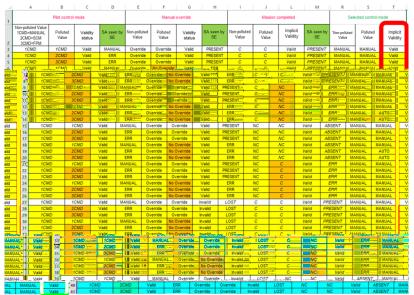
This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRIT, CNRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MBDA.



The calculation of \mathfrak{E} « polluted » Output (Scolumn in Figure 16) is done by using the « polluted » Input vector (columns B, F, J) with its associated validities (columns C, G and L non corruptible by hypothesis). In order to determine the output value, \mathfrak{E} logic expressed in the specification is used.



Hypothesis taken for the output value is that its associated validity is also valid. An implicit validity is set for the SE output vector (T column in Figure 17).



This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRIT, CNRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MBDA.



5.1.8 Transformation Outputs SE~SA

As it has been done in Section 5.1.6, same rules are applied for the computation of the output vector. Comparison is utput SE-SA always

expressed in SA domain (U). If both value are consistent, between column (R) and (S), the output will be the same. If values differ, the value will be ERR at least or LOST if validity is invalid.



Note: some pollutions are not effective immediately. They can be masked by a non-linearity in the SE logic (e.g. via a switch set on a sound input while the pollution is on the not selected input).



5.1.9 Remove same combinations or select one

From the whole SE-SA table obtained (Figure 18 ed and isolated. The resulting four columns table is the equivalent table SA but coming from SE specification (including its logic), but they still cannot be compared yet.





Table in Figure 19 still has 576 rows and for the input vector (D, H, and M columns), there are some cases that are repeated. Number of rows are reduced by merging the cases that are repeated into the same input vector. After this merge, the output vector will inherit the value (or values if different output for the same input vector) as its pre-merger equivalents. When more than a value for the output has been detected, the cell in question is highlighted in red. In that case, there may be potential incoherencies to be treated (see Figure 20) or maximization/minimization choices to deal

complementary analyses shall be done to check if this exceeding is induced by the maximization done here.

Note: row #25 from Figure 20 corresponds to the merge of the 8 rows from Figure 21. All cases, share the same input vector after SE-SA transformation (AUTO, ERR, ERR ERR ERR

case and consequently leads to different repercussions to its output value after \mathfrak{E} logic application. This is the reason why for a given \mathfrak{E} -SA input vector, the output values may take more than one possible value.

	A	В	С	D	E	F	G	Н	1	J	K	L	0	P	Q	R
1		Pilot control I	node			Manual	override			Mission c	ompleted			Selected co	ntrol mode	
2	Non-polluted Value 1CMD=MANUAL 2CMD=SCM 3CMD=FPM	Polluted Value	Validity status ▼	SA seen by SE	Non-polluted Value	Polluted Value	Validity status ▼	SA seen by SE	Non-polluted Value	Polluted Value	SA seen by SE	Implicit validity	Non-polluted Value	Polluted Value	Implicit validity	SA seen by SE
204	2CMD	2CMD	Valid	AUTO	Override	No Override	Valid	ERR	С	NC	ERR	Valid	MANUAL	AUTO	Valid	ERR
216	2CMD	2CMD	Valid	AUTO	Override	No Override	Valid	ERR	NC	С	ERR	Valid	MANUAL	MANUAL	Valid	MANUAL
252	2CMD	2CMD	Valid	AUTO	No Override	Override	Valid	ERR	С	NC	ERR	Valid	MANUAL	MANUAL	Valid	MANUAL
264	2CMD	2CMD	Valid	AUTO	No Override	Override	Valid	ERR	NC	С	ERR	Valid	AUTO	MANUAL	Valid	ERR
396	3CMD	3CMD	Valid	AUTO	Override	No Override	Valid	ERR	С	NC	ERR	Valid	MANUAL	AUTO	Valid	ERR
408	3CMD	3CMD	Valid	AUTO	Override	No Override	Valid	ERR	NC	С	ERR	Valid	MANUAL	MANUAL	Valid	MANUAL
444	3CMD	3CMD	Valid	AUTO	No Override	Override	Valid	ERR	C	NC	ERR	Valid	MANUAL	MANUAL	Valid	MANUAL
456	3CMD	3CMD	Valid	AUTO	No Override	Override	Valid	ERR	NC	С	ERR	Valid	AUTO	MANUAL	Valid	ERR

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRIT, CNRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MEDA.



5.1.10 Comparison

Finally, last step consists in comparing the tables obtained SA (Figure 7) and SE-SA (Figure 20). Rows highlighted in grey are the cells found in SA TAIO mapped in the SE-SA resulting TAIO. White rows behaviors are a priori not in the scope of SA specification and need to be discussed because it means both domains have not taken the same hypothesis regarding either the logic or the association between values.

d by

a SEU) whilst SA specialist does not consider it at all. The issue must be discussed during SE-SA reviews and further responsibilities and design guarantees may be extracted from that review.

Note: column (A) for left side and (E) for right side serves for traceability between tables.

	А	В	С	D	E
1		Pilot Manual Control	Manual Override	Mission Completed	OutputMode
2	А	AUTO	PRESENT	PRESENT	MANUAL
3	В	AUTO	PRESENT	ABSENT	MANUAL
4	С	AUTO	ABSENT	PRESENT	MANUAL
5	D	AUTO	ABSENT	ABSENT	AUTO
6	E	MANUAL	PRESENT	PRESENT	MANUAL
7	F	MANUAL	PRESENT	ABSENT	MANUAL
8	G	MANUAL	ABSENT	PRESENT	MANUAL
9	Н	MANUAL	ABSENT	ABSENT	MANUAL
10	1	ERRONEOUS	PRESENT	PRESENT	MANUAL
11	J	ERRONEOUS	PRESENT	ABSENT	MANUAL
12	К	ERRONEOUS	ABSENT	PRESENT	MANUAL
13	L	ERRONEOUS	ABSENT	ABSENT	ERRONEOUS
14	М	LOST	PRESENT	PRESENT	MANUAL
15	N	LOST	PRESENT	ABSENT	MANUAL
16	0	LOST	ABSENT	PRESENT	MANUAL
17	Р	LOST	ABSENT	ABSENT	MANUAL

	A	в	С	D	Е
1	Pilet Manual Centrel	Manual Override	Mirzian Completed	OutputMade	Traçabilité w Tay SA AB
2	MANUAL	PRESENT	PRESENT	MANUAL	E
3	MANUAL	PRESENT	ABSENT	MANUAL	F
4	MANUAL	PRESENT	ERR	MANUAL	
5	MANUAL	ABSENT	PRESENT	MANUAL	G
6	MANUAL	ABSENT	ABSENT	MANUAL	н
7	MANUAL	ABSENT	ERR	MANUAL	
8	MANUAL	LOST	PRESENT	MANUAL	
9	MANUAL	LOST	ABSENT	MANUAL	
10	MANUAL	LOST	ERR	MANUAL	
11	MANUAL	ERR	PRESENT	MANUAL	
12	MANUAL	ERR	ABSENT	MANUAL	
13	MANUAL	ERR	ERR	MANUAL	
14	AUTO	PRESENT	PRESENT	MANUAL	A
15	AUTO	PRESENT	ABSENT	MANUAL	B
16	AUTO	PRESENT	ERR	MANUAL	
17	AUTO	ABSENT	PRESENT	MANUAL	c
18	AUTO	ABSENT	ABSENT	AUTO	D
19	AUTO	ABSENT	ERR	ERR	
20	AUTO	LOST	PRESENT	MANUAL	
21	AUTO	LOST	ABSENT	MANUAL	
22	AUTO	LOST	ERR	MANUAL	
23	AUTO	ERR	PRESENT	MANUAL	
24	AUTO	EBB	ABSENT	EBB	
25	AUTO	ERR	ERR	MANUAL/ERR	
26	ERR	PRESENT	PRESENT	MANUAL	1
27	ERR	PRESENT	ABSENT	MANUAL	J
28	ERR	PRESENT	ERR	MANUAL	
29	ERR	ABSENT	PRESENT	MANUAL	к
30	ERR	ABSENT	ABSENT	AUTO/ERR	L
31	ERR	ABSENT	ERR	MANUAL/ERR	
32	ERR	LOST	PRESENT	MANUAL	
33	ERR	LOST	ABSENT	MANUAL	
34	ERR	LOST	ERR	MANUAL	
35	ERR	ERR	PRESENT	MANUAL	
36	ERR	ERR	ABSENT	MANUAL/ERR	
37	ERR	ERR	ERR	MANUAL/ERR	
38	LOST	PRESENT	PRESENT	MANUAL	м
39	LOST	PRESENT	ABSENT	MANUAL	N
\$0	LOST	PRESENT	ERR	MANUAL	
н	LOST	ABSENT	PRESENT	MANUAL	0
£2	LOST	ABSENT	ABSENT	MANUAL	P
\$3	LOST	ABSENT	ERR	MANUAL	
14	LOST	LOST	PRESENT	MANUAL	
1 5	LOST	LOST	ABSENT	MANUAL	
46	LOST	LOST	ERR	MANUAL	
17	LOST	EBB	PRESENT	MANUAL	
48	LOST	EBR	ABSENT	MANUAL	
49	LOST	ERR	ERR	MANUAL	

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRT, CNRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MBDA.



5.1.11 Conclusions

This PoC done on a trivial perimeter points out:

The confirmation that explicitness of table will collide with verbosity (amount of rows and lines to handle).

A future track to follow can be the automation of some activities.

The rules for transformation will become more complex when interfaces will be more numerous and intertwined.

A future track to follow can be on the assessment of the method with a more complicated interface.

The free specification of behavior of SE is a brake to the method.

A future track to follow can be the use of a Domain Specific Language that will help in the production of tables. The difference of cardinality of functions in the perimeter (so a different grain).

A future track to follow can be the assessment of method with a non 1-1 cardinality of CL

The method is a kind of elicitation of how SA specialist thinks implicitly when he/she does his/her SA model. This elicitation allow reviewers (Safety pair or the Especialist) to challenge the choice done.

This method detects discrepancies (all white lines and red cells) for this simple case to be discussed between SE and SA only for the Sofl=OK, the other case of Sofl are normal work between SE and SA.



5.2 PoC2

Previous PoC was a setup for the method using a one for one mapping between SE and SA. However, this situation is not representative when models are ruled by a relative freedom in their realization regarding each jobs. That is why another PoC must be done and it shall include:

- SE function using continuous variables while SA is using enumerated ones. This is different from previous PoC that have only enumerated types in both sides and this has an impact on the domain matching between SE and SA.
- SE behavior specification are not composed of a monolithic table as previous PoC iables go in the game,
- Some discrepancies in the interfaces for the representation of a same function between the SE and SA models.

CL41 (see Figure 23), based up matches all criteria of previous paragraph:

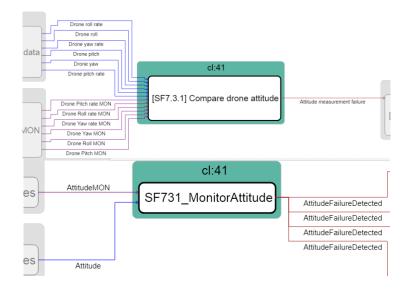
31 Compare Drone Attitude

t

- SE focus on the continuous real values of the attitude, while SA considers the enumeration of the attitude "failure state.

- SE specification of behavior is composed of 7 sub-tables linked by intermediate/hidden variable
- Interface are 12 variables (for SE) against 2 (for SA)

However, QL41 remains a 1-1 QL function (see Rounded green square in Figure 23) regarding leaf functions of AIDA models versioned in Table 5. That means only one SE function matches only one SA function.



Note: there is some divergence between interface and

specifications. Specifications of the function must use First

discrepancy: the specification can need an interface lacking in the model. Second discrepancy: the model can have an interface not required by the specification. Third discrepancy: word used in specification may carried semantic that the label of the interface does not (e.

that semantic is not aligned between specification and model and may lead to wrong integration or wrong spec).

Note: the modeling and specification activities must be ruled by guide in companies. Rules comes from:

- the standardization of way of working in a company (e.g. whatever the project, a worker shall be able to find same pattern between projects, e.g. where artefacts (model or doc etc) are stored, where just

- the standardization of jobs, so that each specialist can be pervasive between project, e.g. any SE (whatever its project shall have a model containing these and another one for those) etc

- The standardization of tools usage so that fragmentation is reduced, e.g. if any \mathfrak{E} use differently the feature of a tool, or attributes any semantic to a representation, the understanding of model can be jeopardized

In this PoC, some activities may require automation because of the amount of interfaces and domains that increase the number of combinations for expansion and pollution activities.

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRIT, ONRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MEDA.



5.2.1 SA Model and specification

QL41 SE model is already represented via Figure 23.

SA specification is an implementation expressed in the SA tool format (Smfia Neo Altarica code) that is formal contrarily to the \mathfrak{E} one (see 5.2.3). This allows a way to get the SA TAIO effortlessly.

Assertion:	^
case { input1 = LOST : TRU, input2 = LOST : TRU, input1 = input2 : FALS,	=
else TRU }	

Domain values are enumerated ones and are the following for each variables:

- Attitude: OK, LOST and ERRONEOUS
- Attitude MON: OK, LOST and ERRONEOUS

Note: SA implementation is source driven that means each individual attitudes are merged into a single variable always justified despite CL flows make formal the link grouping.

Implementation like this is similar to what is called early-optimized code in software development. That means, SA arrived to a reduced result that does not how the transformation from the reference specification (i.e. SE spec) was done. So some hypothesis are lost or implicit and may jeopardize work to check this choices.

5.2.2 Expansion of SA Inputs and Outputs (TAIO SA)

From behavior implementation of SA (Figure 24), all possible combinations are created with Inputs domains in Figure 26 by using feature (rounded square in red in Figure 25) of SIMFIA NEO tool (available since 1.4.2 version).

Properties ×	🗐 Images 🔒 AltaRica			
Comparato	r			
Identification			_	
Behavior Propagation	$\bigcirc \bullet \odot \times$			Assertion
😤 Brick Style	Name	Domain	Direction	case i input1 = LOST : TRU,
🧟 User data	O input1	BasicFunctionStatus	In	input2 = LOST : TRU,
	O input2	BasicFunctionStatus	In	input1 = input2 : FALS,
	output	MyBool	Out	else TRU
				N

The TAIO SA is generated automatically by the feature (based upon the Altarica code) as follow:

	○ input1	O input2	output	
	SELECT_ALL	SELECT_ALL	SELECT_ALL	Ø
1	ERRONEOUS	ERRONEOUS	FALS	
2	OK	ERRONEOUS	TRU	
	SBRONEOUS		7011	
	SPROLICOUR.	411 :	TPH LO	ST
	REPORTIONS	500 LOS	T 🗠	
	FALS	655 OK	OK	
	TRU	788 OK	LO	ST
			т	
		 nn	Т	hΤ

Note: As method align on SA TAOI there is no more to do.

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRT, ONRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MBDA.



5.2.3 SE Model and specification

OL41 SE model is already represented via Figure 23.

Q.41 SE behavior specifications are defined through several table see Table 7.

A B	C		D		E	: [F		G	Н	
		Result of dron			Drone yaw						
1 Ref Drone yaw status	Drone yaw MON status				measureme						
		1A 3. 1							14-11 J.		
 	3	83/01				P	•		an Usara		
	4	C [53/59E=		N/A		Me			WHUE ADD		
	3	D N/A		INVAU D		N/A ···			TAILURE DR	39130	
	6	·									
				_			one yaw rate di				
				Drone yaw r			ie yaw rate -dro	one yaw rate	Drone yaw i		
			w rate status			MON <t_yav< td=""><td>v_rate)</td><td></td><td>measureme</td><td></td><td></td></t_yav<>	v_rate)		measureme		
		E VALID		VALID		OK KO			NO FAILURE FAILURE DE		
	3	GINVALID		N/A		N/A			FAILURE DE		
	1	H N/A		INVALID		N/A			FAILURE DE		
	1	1 1 1 1 1 1 1							I AILONE DE	LETED	·
		1				Decult of dr	one pitch discre	an a			
				Drone pitch			h -drone pitch	cpancy check	Drone pitch		
	13	Drone pit	ch status	status		MON <t_pit< td=""><td></td><td></td><td>measureme</td><td></td><td></td></t_pit<>			measureme		
	14	VALID		VALID		OK			NO FAILURE		
	10			VALID		KO			FAILURE DE		
	16			N/A		N/A			FAILURE DE		
	1	L N/A		INVALID		N/A			FAILURE DE		
	18					Í III					
·· · · · · · · · · · · · · · · · · · ·						Result of dr	one pitch rate o	liscrepancy			
and the second sec		1 I 1	Bracer:	These in the			mexiciante		ale grand "	summer's	anet i i i
		19	status		ratatus.		Irane (1014 ×7 g			mausurem	
		20	M VALD		V/N.D		dĸ			NO FALLUR	ε
		21	N VALD		VM D		ck KO				етестер
		22	DINVALID		N/A		N/A			FAILURE DI	
-		23	P N/A		INVAL D		N/A			FAILURE D	ETECTED
		24									
							Result of dron	e roll discrep	ancy check	Drone roll	
e		25	Drone rol	l status	Drone roll N	/ON status	(drone roll -d			measurem	nent failur
		26	Q VALID		VALID		фк			NO FAILUR	(E
		27	R VALID		VALID		ко			FAILURE D	ETECTED
		26 27 28 29	S INVALID		N/A		N/A			FAILURE D	ETECTED
		29	T N/A		INVALID		N/A			FAILURE D	ETECTED
		30									
							Result of dron	e roll rate dis	crepancy		
					Drone roll r	ate MON	check (drone		ne roll rate	Drone roll	rate
e		31	Drone rol	l rate status	status		MON <t_roll_< td=""><td>rate)</td><td></td><td>measurem</td><td>ient failur</td></t_roll_<>	rate)		measurem	ient failur
		32 33 34 35	U.VALID		VALID		OK .			NO FAILUR	E
		33	V VALID		VALID		ко			FAILURE D	
		34	WINVALID		N/A		N/A			FAILURE D	
			K N/A		INVALID		N/A			FAILURE D	ETECTED
		36									
Drone roll Drone ro										11	
measurement measur			Drone yav		Drone yaw i		1			Drone pito	
e failure failure	failure	37		ment failure	measureme		Drone pitch m	easurement f	ailure	measurem	
NO FAILURE NO FAIL			1 NO FAILU		NO FAILURE		NO FAILURE			NO FAILUR	E
N/A N/A	FAILURE DE		2 FAILURE D	DETECTED	N/A		N/A			N/A	
N/A N/A	FAILURE DE		3 N/A		FAILURE DE	TECTED	N/A			N/A	
N/A N/A	FAILURE DE		4 N/A		N/A		FAILURE DETEC	TED		N/A	
N/A N/A	FAILURE DE		5 N/A		N/A		N/A			FAILURE D	ETECTED
FAILURE DETECTED N/A	FAILURE DE		6 N/A		N/A		N/A			N/A	
N/A FAILURE	DETECTED FAILURE DE	TECTED 44	7 N/A		N/A		N/A			N/A	

SE chooses to organize the function behavior by cross comparing each attitude of a source (Pitch, Roll Yaw, time derivate of Pitch of Roll and of Yaw) to its corresponding attitude in the other source (green column: Ref A to X). Then he consolidate

Note: such a function is not atomic as it can be sub-divided in more grained-functions (each attitude monitoring and then the consolidation). That means the \mathfrak{E} stops its structural breakdown on not terminal functions.

Note: Se behavior implementation is driven by breakdown of function and flows between them while the SA specification is driven by sources of data see 5.2.1. (i.e. data coming from same origin can be merge) and behavior inside grouped functions (i.e. consecutive functions having same propagation can be merge into a single function to reduce cut-sets). The usage of different modeling philosophies (abstraction at function and flow level) may make it harder to prove the consistency.

Note: Se structure (Figure 23.) hide the validity information that are required by its behavioral specifications (presence of status in tables). That means Se interface is not 12 variables (6 attitudes by 2 sources) but 24 variables (6 attitudes and their respective validity by 2 sources).



5.2.4 Expansion of SE Inputs (TAIO SE)

We decide to reuse the SIMFIA NEO thruth table feature (see 5.2.2) to produce the SE TAIO automatically. However, to do this, we had to implement a SIMFIA NEO model to represent the SE behavior as defined. For this:

- Firstly, we build a SMFIA NEO 24-inputs and 1-output component, see left part of Figure 28
- Secondly, we do the ALTARICA code that map the SE specification: 6 tables for monitoring (see central part of Figure 28) and 1 table for consolidation (see right part of Figure 28). As attitudes are not enumerated values and our specialization of method require an SA alignment (see Figure 3) which is enumerated, we have to implement a trick see in the central part of Figure 28. It consists to divide the continuous domain into 3 ranges that are:
 - When an attitude of source 1 is near from source 2 by a threshold of T (which is reciprocal)
 - o When an attitude-

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRIT, CNRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MEDA.



Note: abbreviation by single character is to reduce the size of the generated TAIO as it contains all combinations amongst 24 variables.

Note: despite the amount of interface, SMFIA NEO feature produces a TAIO in less than five minutes.

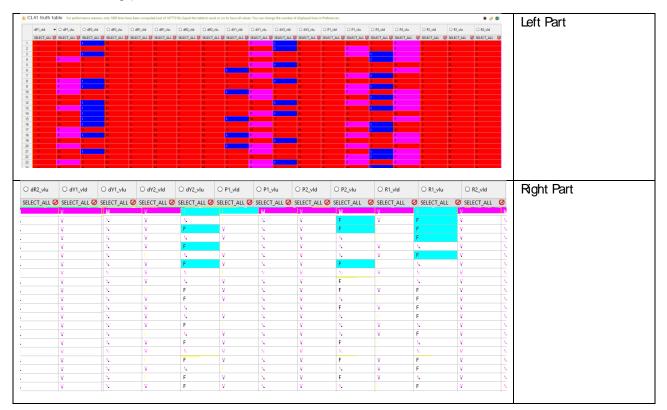
Note: The explicit impossible case while not reachable was a modelling slight error because we can merge this case with the failure (has it is impossible it will change nothing qualitatively or quantitatively

because the impossible cases represent a slight part (730 combinations) of the overall combinations (2^24) produced.

(resp. consolidation) should be written with the following patterns (see upward part of Figure 29 (resp downward part)

X_status = <u>case</u> {
$(X1_vld = V)$ and $(X2_vld = V)$ and $(P1_vlu = N)$ and $(P2_vlu = N) : n$,
Else
f
}
attitude_failure = case {
(P_status = n) and (dP_status = n) and (R_status = n) and (dR_status = n) and (Y_status = n) and (dY_status = n) : n,
else
f
}

Note: Doing implementation of Se specification using SMFIA NEO, the truth table feature can be used to challenge the Se specification and allow to detect by review unexpected local behavior (i.e. a wrong output regarding the input and state). This implementation can be therefore be also reused for method after its usage for specification review (one artefact several usage).



Note: as width of TAIO is important, it is split onto left and right part of Table 8, and only the first 20 lines are considered but there are is 16777216 (=2^24).

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRT, CNRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MBDA.



5.2.5 Apply pollution on SE Inputs

Whereas this activity was done by hand in PoC1, a coding is required here; regarding the amount of combinations induced by SE design of its functions. This occurs because each combinations (2^24) will need to mutate, as method required it. Mutations are one or several pollutions regarding the 12 pollutable columns (6 attitudes per source while their validity are not considered as pollutable) constituting the case.

The coding efficiency rely on the formatting of the data organisation. That means the SE TAIO export is important for this activity to have a performant result while respecting the memory capability of the computer. But as SIMFIA NEO truth table only export CSV or excel file, that are suitable for human reading but not performance when 2^24 combinations are in game and will generate 2^12 polluted combinations each one.

Note: Despite this activity is purely computational (except the choice of pollutable columns that rely on explicating the SA criteria given to the tool doing this activity), this investment has no meaning if not all activities of the method cannot be achieved (i.e. all activities shall succeed otherwise coding this one is useless). So we have to go further to status this point.

5.2.6 Transformation Inputs SE~SA

This activity relies on each generated combination issued from Section 5.2.5 to compute the resulting polluted inputs. This activity require to code (because of the amount of combination) the SA knowhow about the transformations of \mathfrak{E} inputs to its SA ones. This transformation is specific from each perimeter, because \mathfrak{E} inputs are different from one to another \mathfrak{E} perimeter (except the case when a \mathfrak{E} pattern is instantiated several time in its design). Therefore, if there is 100 distinct perimeters, 100 transformations are to be done to map the \mathfrak{E} inputs into the SA ones (despite of SA reuse the same domains in his side). SA specialist does these transformations in his mind when he implements the model but no usable traces remain of this knowhow for consistency except it seems consistent (but no proof of that). This elicitation is the aim of this activity to explicit the transformation done.

Coding the transformation does not meet technical impossibility (except performances when amount of combination is huge). However, we have to wonder if the SA specialists are ready to do such coding as they are always specific because of the SE specificity interface from where they derive when they do their SA model.

It shall be noticed that even if a GUI is designed and implemented to reduce the SA efforts while keeping the choice he did, it will do not match all the cases. The Poc2 is such a case where 2 orthogonal philosophies (driven by source and by data) are taken. This dilemma occurs because the SA modeled a different philosophy from the reference one (because he is free to do so) while if he keeps the SE philosophy no such transformation are required (we go back in PoC1 with a one for one mapping).

Note: The same discurse can be applied to the Section 5.2.8.



5.2.7 Compute Output SE non-polluted and polluted values

Technically this activity rely on the retrieving of SE outputs associated to the SE inputs (polluted and not)

For this, the buffering of the 2^24 combinations has be used because the SE polluted inputs can be retrieved from one of the nominal combination (as all combination are produced).

Note: Smilarly to note of Section 5.2.5, this phase is purely computational but doing this code depends on the potential confidence of success of other activities especially the Section 5.2.6 and Section 5.2.8 otherwise is meaningless to do it.

5.2.8 Transformation Outputs SE~SA

The note of Section 5.2.6 is applicable here, SA has to explicit how SE outputs map its SA enumerated outputs. This situation may be less painful as there is often less output than inputs. However, the wondering remains here because outputs of SE perimeter are specific so are the transformations if he choose to derive from the reference.

5.2.9 Remove same combinations or select one

This activity is purely computational, so note of Section 5.2.5 and Section 5.2.7 are applicable about the investment in coding.

5.2.10 Comparison

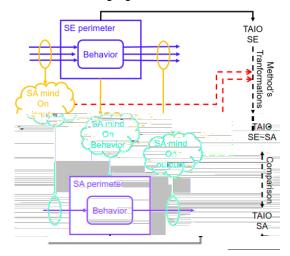
This activity is partially computational for the presentation of the data to the user, but not for the interpretation of the data that remains the prerogative of humans.

This activity (for the preparation part) is subject to same remark carried by note of Sections 5.2.5, 5.2.7 and 5.2.9.



5.2.11 Conclusions

The situation for Poc2 summarizes into the following Figure 30:



Synopsis has following legend:

Green (resp. dark blue) box and arrows are the SA (resp. SE) perimeter, behavior, flows present in models, and are the ones we want to check constancy across.

Orange clouds, rounds, and arrows represents the selection and transformations that SA does to make its model regarding SE model reference.

Black plain arrow are automated works done of the method

Black dashed arrow are potentially doable automated work of the method activities but that are jeopardized by implicit decision and manual works (i.e. coding of SA rules specific for perimeter)

Red dashed arrows are questioning, as it requires eliciting formally (in code for automation) the knowhow of SA, used when he creates by hand his model before using method.

From here, there is two strategies can be imagined:

- 1- Forcing SA to map the interface of SE
- 2- Avoiding SA to create by hand its model

First strategy ensures the better consistency reachable. An advantage of this strategy is, the \mathfrak{E} can review the SA model as it is similar to its own model (this reduce his efforts on the reviewing of the events considered by SA and the analysis produced). Another advantage is that the SA model represents the \mathfrak{E} design system (and not an abstraction from it) so there is less doubts on analysis produced. The possible drawback is the computational part of the SA job may be jeopardized

having same interface will ease also the correct understanding between specialists.

Second strategy, require tools (with QoS) that SA configures to do the transformation and so the tool capture the expertise, and can generate the code for the method. That means SA specialist shall not have to create his model by his own hands but help and survey the transformation gateway tools to do the job as expected. So SA has to change an in the interface setting parameters of transformation to get the box and

arrows he wishes.

In both strategy, the SA has to fill the dysfunctional behavior (what he does already) but lose a part of his freedom to make the model as he wishes and not as it is referenced.



IRT Saint Exupéry LIV- S085L02-024 IRT SystemX ISX-S2C-DOC-437 Version: V6

End of document

This document is the property of the S2C Project Participants: IRT Saint Exupéry, IRT SystemX, IRIT, ONRS, Airbus Defence & Space, Dassault Aviation, Thales AVS, Thales SA, Liebherr, LGM, APSYS, Samares Engineering, DGA, ONERA, SupMeca and MBDA.