ISVV for Evolutions in SW Development Methods and Processes

Nuno Silva and Xavier Ferreira (Critical Software), Jesper Troelsen and Tomasz Kacmajor (Rovsing), Andrei-Mihai Buzgan (ESA/ESTEC)

ADCSS 2020

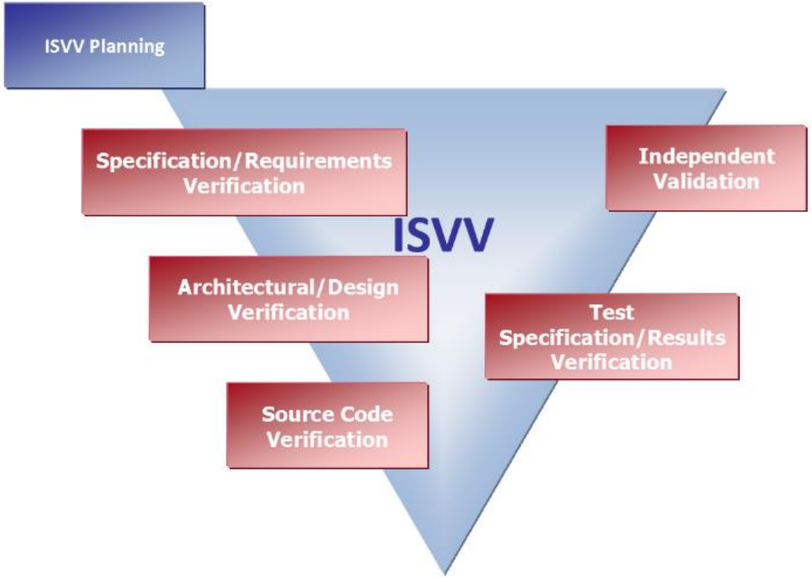
Presentation Date: 21/10/2020

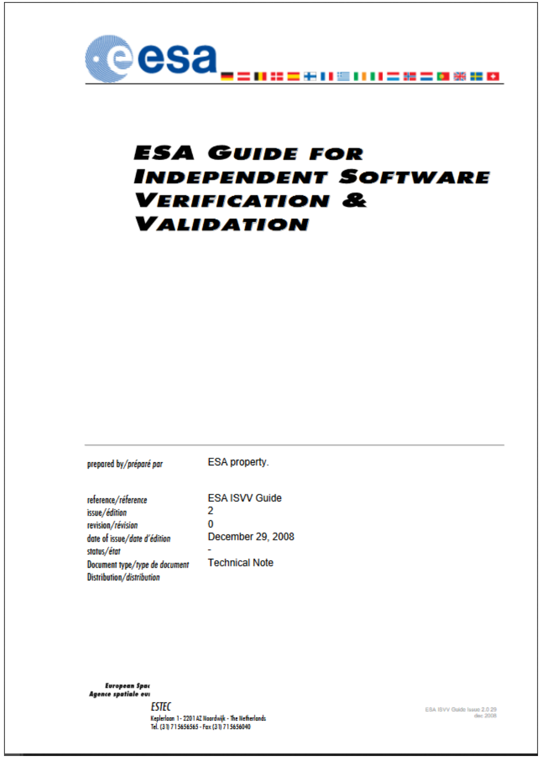
CRITICALSOFTWARE.COM

Outline

- Introduction
- Main On-Going Activities
- Roadmap
- Relevant Changes and Open Points
- Topics Workshops
- Acknowledgements
- Q&A

Introduction


ISVV Planning
Specification/Requirements Verification
Independent Validation
ISVV
Architectural/Design Verification
Test Specification/Results Verification
Source Code Verification



ESA GUIDE FOR INDEPENDENT SOFTWARE VERIFICATION & VALIDATION

prepared by/préparé par	ESA property.
reference/réference	ESA ISVV Guide
issue/édition	2
revision/révision	0
date of issue/date d'édition	December 29, 2008
status/état	-
Document type/type de document	Technical Note
Distribution/distribution	

European Spac
Agence spatiale eu

ESTEC
Keplerlaan 1 - 2201 AZ Noordwijk - The Netherlands
Tel. (31) 715656565 - Fax (31) 715656040

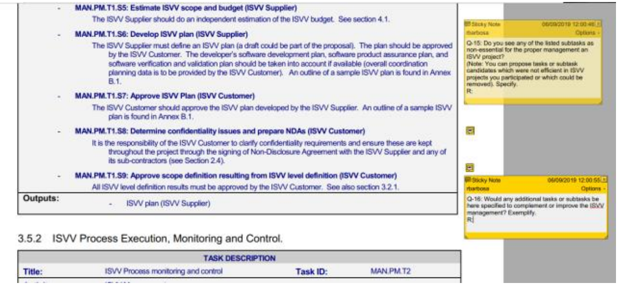
ESA ISVV Guide Issue 2.0 29
dec 2008

Introduction

ISVV Handbook Generation Process:

- 1. Collect feedback from stakeholders/participants
- 2. Analyze and extract the most relevant information from the answers to the checklists.
- 3. Prepare individual technical notes per topic.
- 4. Prepare, present and discuss this information with stakeholders in a dedicated workshop.
- 5. Produce the handbook and support ESA on its promotion.

Introduction



- **MAN.PM.T1.S5: Estimate ISVV scope and budget (ISVV Supplier)**
  The ISVV Supplier should do an independent estimation of the ISVV budget. See section 4.1.
- **MAN.PM.T1.S6: Develop ISVV plan (ISVV Supplier)**
  The ISVV Supplier must define an ISVV plan (a draft could be part of the proposal). The plan should be approved by the ISVV Customer. The developer's software development plan, software product assurance plan, and software verification and validation plan should be taken into account if available (overall coordination planning data is to be provided by the ISVV Customer). An outline of a sample ISVV plan is found in Annex B.1.
- **MAN.PM.T1.S7: Approve ISVV Plan (ISVV Customer)**
  The ISVV Customer should approve the ISVV plan developed by the ISVV Supplier. An outline of a sample ISVV plan is found in Annex B.1.
- **MAN.PM.T1.S8: Determine confidentiality issues and prepare NDAs (ISVV Customer)**
  It is the responsibility of the ISVV Customer to clarify confidentiality requirements and ensure these are kept throughout the project through the signing of Non-Disclosure Agreement with the ISVV Supplier and any of its sub-contractors (see Section 2.4).
- **MAN.PM.T1.S9: Approve scope definition resulting from ISVV level definition (ISVV Customer)**
  All ISVV level definition results must be approved by the ISVV Customer. See also section 3.2.1.

Outputs:
- ISVV plan (ISVV Supplier)

Sticky Note 06/09/2019 12:00:46
rbarbosa Options
Q-15: Do you see any of the listed subtasks as non-essential for the proper management an ISVV project?
(Note: You can propose tasks or subtask candidates which were not efficient in ISVV projects you participated or which could be removed). Specify.
R:

Sticky Note 06/09/2019 12:00:55
rbarbosa Options
Q-16: Would any additional tasks or subtasks be here specified to complement or improve the ISVV management? Exemplify.
R:

3.5.2 ISVV Process Execution, Monitoring and Control.

TASK DESCRIPTION			
Title:	ISVV Process monitoring and control	**Task ID:**	MAN.PM.T2

Stakeholders/participants

- 32 invited;
- 23 (72%) answered the questionnaire;
- 85 questions provided inside the ISVV Guide:
  - 22% of the total stayed unanswered;
  - First questions answered up to 95%, last ones up to 75%;
- Classification by level of perceived importance and level of contradictoriness;

Main On-Going Activities

Activities are incorporated in three main requirement groups:

- Group 1: Incorporate experience from using the ISVV
- Group 2: Alignment to latest development standards
- Group 3: Modern software development practices and methods

- Next slides present the topics covered.

Main On-Going Activities

- Group 1: Incorporate experience from using the ISVV

Topic ID	Topic Title
R-1.1	Verification of software requirement baseline and concept documentation
R-1.2	Improvement of the Independent Validation Activity (IVA)
R-1.3	ISVV Level re-assessment
R-1.4	ISVV metrics definition and collection framework
R-1.5	ISVV statement of work template
R-1.6	Verification of the unit test specification
R-1.7	Clarification of ISVV activity outputs
R-1.8	Revisit current ISVV tasks regarding their effectiveness
R-1.9	Lessons learned collection framework
R-1.10	Independent verification and validation of software dependability and safety activities
R-1.11	Continuous ISVV process
R-1.12	Reassess the ISVV industrial context (optional)
R-1.13	Complementarity of ISVV activities (optional)
R-1.14	Miscellaneous inputs on ISVV processes
R-1.15	Level 2 description improvements (optional)

Main On-Going Activities

- Group 2: Alignment to latest development standards

Topic ID	Topic Title
R-2.1	ECSS-E40C and ECSS-Q80C impact on ISVV processes
R-2.2	Align the document structure with the ECSS Handbook documentation format
R-2.3	Traceability between ESA ISVV Handbook and International Standards (optional)
R-2.4	Independent verification and validation from other space domains (optional)
R-2.5	Independent verification and validation from non-space domains (optional)

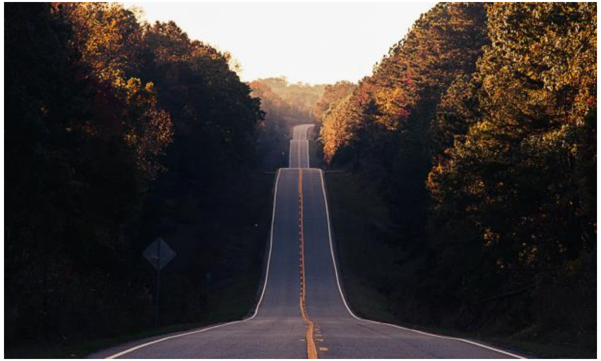
Main On-Going Activities

- Group 3: Modern software development practices and methods

Topic ID	Topic Title
R-3.1	Independent verification and validation of reused software
R-3.2	Independent verification and validation of data
R-3.3	Independent verification and validation of complex electronics (ASIC/FPGA-based designs) (optional)
R-3.4	Independent verification and validation of auto generated code
R-3.5	Independent verification and validation when using Model Based Techniques
R-3.6	Independent verification and validation of SW developed following an iterative model
R-3.7	Independent verification and validation of agile developed systems
R-3.8	Modern and alternative methods & techniques to perform independent verification and validation

Roadmap

- Workshop with stakeholders (per topic) to solve contradictions, reach consensus and enrich the proposed changes – Nov 2020
- Draft the ISVV Handbook – Feb 2021
- Promote Public review (according to ECSS) of the ISVV Handbook – Mar 2021
- Write down the final version



Roadmap

- Technical Note 1: IVV Collection Improvements, gathered all inputs from stakeholders and participants as well as consortium experience;
- Technical Note 2: IVV Assessment Improvements, composed by 22 topic-oriented TNs + 7 optional TNs, covers the analysis and the proposed modifications;
- A set of dedicated workshops, involving the stakeholders, will cover the proposed changes and any existing open points.

Relevant Changes and Open Points

- Some of the topics raised discussion and no obvious solutions could be found;
- In some cases, due to the novelty of the topic, there is limited experience to base upon;
- We present here some examples of the relevant changes and open points;
- Technical Notes for the assessed topics are available under request: isvv@criticalsoftware.com.

Verification of SW req. baseline and concept documentation

- Purpose:
  - Should ISVV move a step upwards, and perform IVV on the software requirement baseline and concept documentation?
- Analysis:
  - Additional survey to System Engineering participants
  - Study ECSS standards related to Software and System Engineering
- Conclusion:
  - Stakeholders mostly reject extending ISVV scope to Operations and Maintenance phase.
  - Agree that verification of requirement baseline could be relevant for ISVV.
  - List of documents have been proposed which could be reviewed in the process

Verification of SW req. baseline and concept documentation

- Add a new activity “Requirement Baseline Analysis” (IVE.RA), including:
  - correctness and completeness of the software related system requirements
  - software related system requirements externally and internally consistent
  - software related system requirements unambiguous and verifiable
  - software related system requirements related to safety and dependability correct
- Documents to review:
  - Software System Specification
  - Interface Requirement Document
  - Safety and Dependability Analysis
- Update description of the ISVV supplier competences to include that basic System Engineering knowledge is desirable.

ISVV Level re-assessment

- Purpose:
  - Assess the scope, definition and use of the ISVV level
  - Mitigate the limitations and simplify its overall process
  - Assessment of alternative ways of tailoring ISVV
- Conclusion:
  - No consensus between respondents whether to remove ISVV level definition activity
  - Many drawbacks of ISVV level definition activity identified
  - Tailoring of ISVV scope suggested to replace ISVV level

ISVV Level re-assessment

- Remove ISVV Level Definition activity (MAN.VV)
- Add ISVV Tailoring section and include tailoring in the Management activity (MAN.PM)
  - Goal: ISVV adds the most value for the given budget
  - Joint effort between the ISVV customer and supplier
  - ISVV scope should be done on SW component level
  - Based on a list of factors influencing the ISVV scope
  - List of documents which should be analysed
  - Guidelines will be made for tailoring the ISVV project

ISVV Level re-assessment

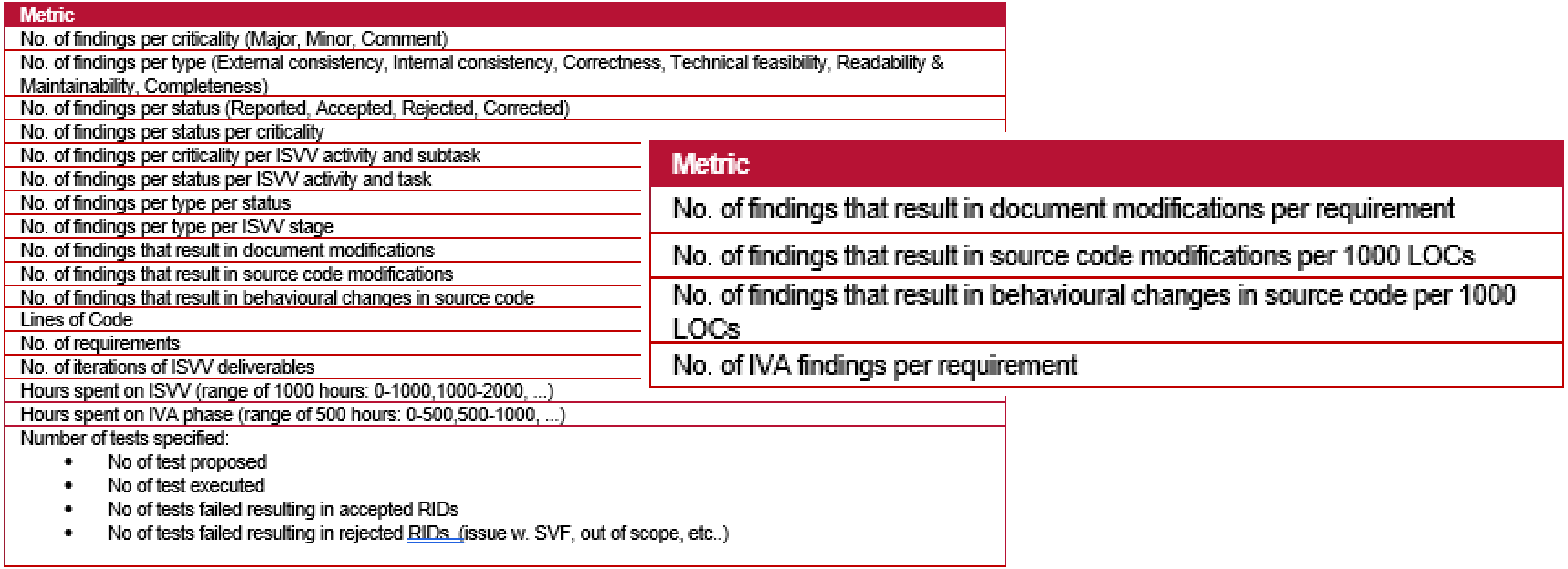
- Input to tailoring
  - ISVV budget
  - SW and mission criticality
  - SW characteristics (size, complexity, reuse)
  - External dependencies (e.g. availability of the SVF),
  - Lessons Learned from previous projects
- A questionnaire will be made asking about influencing factors
- Guidelines for tailoring the ISVV project
  - ISVV tasks are categorised with priority (Recommended/optional).
  - When to perform certain activities
  - Prioritization between activities.
  - Methods to be used/prioritized.
- How to guide the tailoring process?

ISVV metrics definition and collection framework

- Purpose:
  - Measure the efficiency of ISVV
  - Help improving the ISVV process
- Metrics categorized by purpose:
  - Quality of the SW deliverables
  - Efficiency of the ISVV process
  - ISVV process improvement
- Metrics quality measures:
  - Objectivity.
  - Ease of access

ISVV metrics definition and collection framework

- Proposed List of Metrics:



Metric
No. of findings per criticality (Major, Minor, Comment)
No. of findings per type (External consistency, Internal consistency, Correctness, Technical feasibility, Readability & Maintainability, Completeness)
No. of findings per status (Reported, Accepted, Rejected, Corrected)
No. of findings per status per criticality
No. of findings per criticality per ISVV activity and subtask
No. of findings per status per ISVV activity and task
No. of findings per type per status
No. of findings per type per ISVV stage
No. of findings that result in document modifications
No. of findings that result in source code modifications
No. of findings that result in behavioural changes in source code
Lines of Code
No. of requirements
No. of iterations of ISVV deliverables
Hours spent on ISVV (range of 1000 hours: 0-1000,1000-2000, ...)
Hours spent on IVA phase (range of 500 hours: 0-500,500-1000, ...)
Number of tests specified: • No of test proposed • No of test executed • No of tests failed resulting in accepted RIDs • No of tests failed resulting in rejected RIDs (issue w. SVF, out of scope, etc..)

Metric
No. of findings that result in document modifications per requirement
No. of findings that result in source code modifications per 1000 LOCs
No. of findings that result in behavioural changes in source code per 1000 LOCs
No. of IVA findings per requirement

Lessons Learned Collection Framework

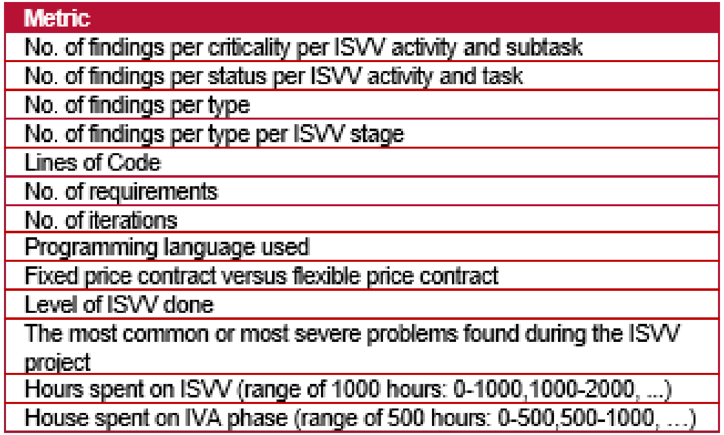
- Purpose:
  - Define a consistent and documented procedure to capture lessons learned.
- Definitions:
  - Data (quantitative and qualitative) which that can help ISVV **process improvements**
  - **Best practices** (or knowledge) other ISVV projects can benefit from.
  - **Project meta-data** for categorizing, sorting and evaluating the Lessons Learned
- Same Quality Measures as for Metrics Definition

Lessons Learned Collection Framework

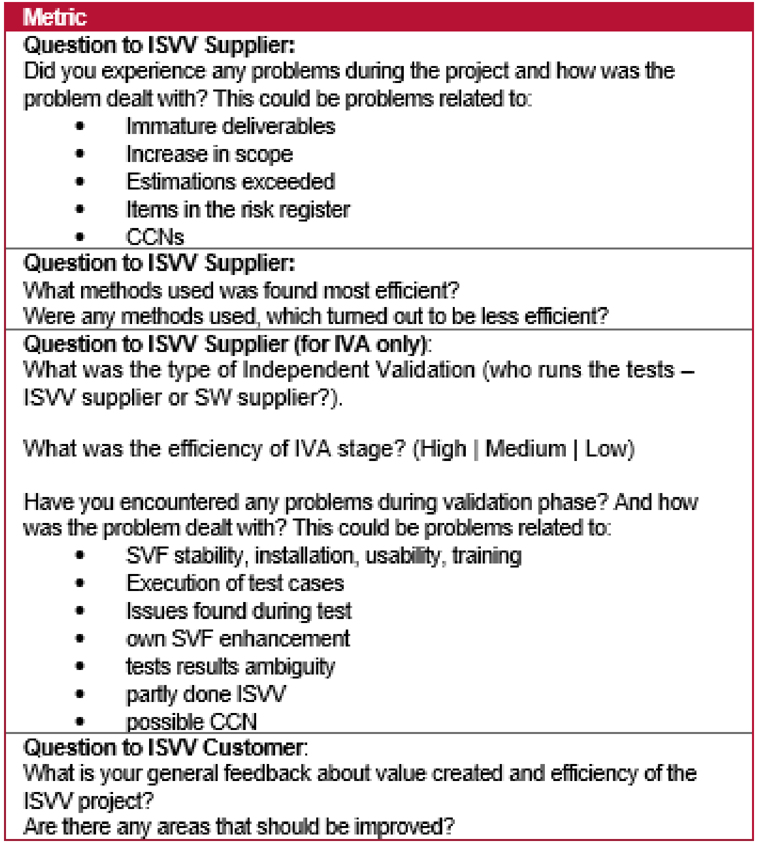
- Conclusion:
  - Consensus among the stakeholders for having a clearly defined Lessons Learned collection framework.
  - List of metrics (qualitative and quantitative) proposed
  - Still open which system should be chosen as a platform for knowledge exchange?
- Proposed changes:
  - Add two structured Lessons Learned templates: one per ISVV activity, one for more general Lessons Learned
  - Add tasks: ISVV customer/supplier to review Lessons Learned
  - Add task: Put structured Lessons Learned into the database

Lessons Learned Collection Framework

- Proposed Lessons Learned:



Metric
No. of findings per criticality per ISVV activity and subtask
No. of findings per status per ISVV activity and task
No. of findings per type
No. of findings per type per ISVV stage
Lines of Code
No. of requirements
No. of iterations
Programming language used
Fixed price contract versus flexible price contract
Level of ISVV done
The most common or most severe problems found during the ISVV project
Hours spent on ISVV (range of 1000 hours: 0-1000,1000-2000, ...)
House spent on IVA phase (range of 500 hours: 0-500,500-1000, …)



Metric
Question to ISVV Supplier: Did you experience any problems during the project and how was the problem dealt with? This could be problems related to: • Immature deliverables • Increase in scope • Estimations exceeded • Items in the risk register • CCNs
Question to ISVV Supplier: What methods used was found most efficient? Were any methods used, which turned out to be less efficient?
Question to ISVV Supplier (for IVA only): What was the type of Independent Validation (who runs the tests – ISVV supplier or SW supplier?). What was the efficiency of IVA stage? (High | Medium | Low) Have you encountered any problems during validation phase? And how was the problem dealt with? This could be problems related to: • SVF stability, installation, usability, training • Execution of test cases • Issues found during test • own SVF enhancement • tests results ambiguity • partly done ISVV • possible CCN
Question to ISVV Customer: What is your general feedback about value created and efficiency of the ISVV project? Are there any areas that should be improved?

Improvement of the Independent Validation Activity

- Purpose:
  - Assess evaluation of Test Case Specifications and/or test reports
  - Make more explicit the scope of the IVA activity
  - Assess possibilities for the software validation facilities (SVF)
  - Extend IVA activity beyond the pure software life cycle
  - Extend IVA activity scope to cover operational end-to-end validation and to validate the software using operational test scenarios

Improvement of the Independent Validation Activity

- Conclusions:
  - Add optional task to perform verification of validation testing specification
  - Suggest initial step to discuss test approach with ISVV customer
  - Create a new list of techniques for creating test cases for IVA activity
  - Existing sections should be modified to put more emphasis on involvement of the SVF supplier
  - Add optional task to identify supplementary test cases for system independent validation

Revisit current ISVV tasks regarding their effectiveness

- Purpose:
  - Assess if any current tasks of ISVV guide should be removed or explained better
- Conclusions:
  - Delete 3 subtasks:
    - Verify software requirements/architectural design/detailed design conformance with applicable standards
  - Shift start date of Software User Manual Verification from DDR to CDR
  - Several minor updates to wordings to make tasks more clear
  - Merge of 3 sub tasks.

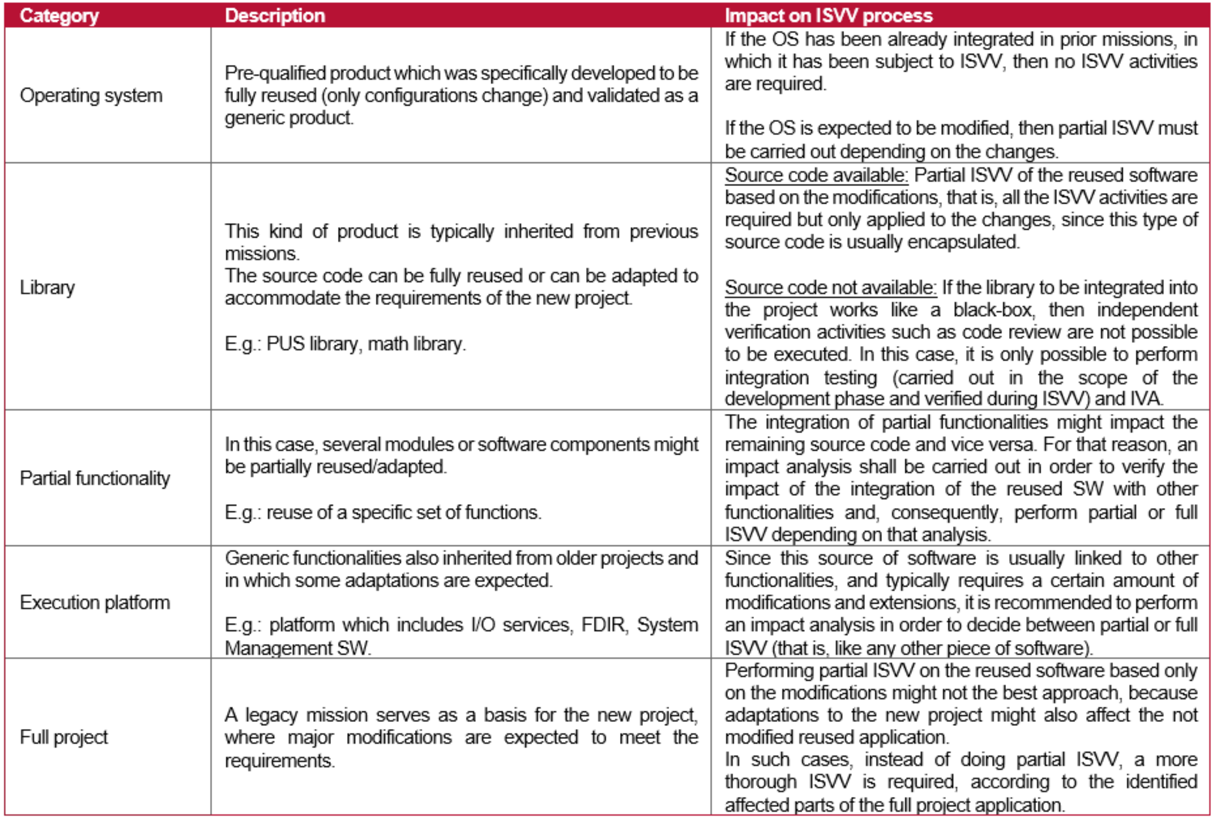
ECSS-E40C and Q80C impact on ISVV processes

- Purpose:
  - To update the ESA ISVV Guide according to the ECSS evolutions.
- Analysis:
  - Delta Analysis of ECSS standards E40C and Q80C;
  - Surveyed stakeholders to gather identifiable changes.
- Conclusion:
  - Update required on definitions and severity classification;
  - New upcoming security requirements and security relevant topics will influence the ECSS standards, but since they are not yet included in the ECSS, the impact on ISVV is still uncertain. Do you have a view on this?

Independent V&V of reused software

- Purpose:
  - To provide guidelines on how and when to perform ISVV on different types of reused software.
- Analysis:
  - Surveyed stakeholders to gather expertise and recommendations;
  - Consortium/ESA expertise gathering and discussions.
- Conclusion:
  - ISVV shall be performed according to predefined criteria, then main being the type of reused SW and the level of modifications / adaptations;
  - There's no silver bullet when defining categories of reused SW and level of containment of such components;
  - Suggestions are welcome to reach an agreement on how to approach this.

Independent V&V of reused software



Category	Description	Impact on ISVV process
Operating system	Pre-qualified product which was specifically developed to be fully reused (only configurations change) and validated as a generic product.	If the OS has been already integrated in prior missions, in which it has been subject to ISVV, then no ISVV activities are required. If the OS is expected to be modified, then partial ISVV must be carried out depending on the changes.
Library	This kind of product is typically inherited from previous missions. The source code can be fully reused or can be adapted to accommodate the requirements of the new project. E.g.: PUS library, math library.	Source code available: Partial ISVV of the reused software based on the modifications, that is, all the ISVV activities are required but only applied to the changes, since this type of source code is usually encapsulated. Source code not available: If the library to be integrated into the project works like a black-box, then independent verification activities such as code review are not possible to be executed. In this case, it is only possible to perform integration testing (carried out in the scope of the development phase and verified during ISVV) and IVA.
Partial functionality	In this case, several modules or software components might be partially reused/adapted. E.g.: reuse of a specific set of functions.	The integration of partial functionalities might impact the remaining source code and vice versa. For that reason, an impact analysis shall be carried out in order to verify the impact of the integration of the reused SW with other functionalities and, consequently, perform partial or full ISVV depending on that analysis.
Execution platform	Generic functionalities also inherited from older projects and in which some adaptations are expected. E.g.: platform which includes I/O services, FDIR, System Management SW.	Since this source of software is usually linked to other functionalities, and typically requires a certain amount of modifications and extensions, it is recommended to perform an impact analysis in order to decide between partial or full ISVV (that is, like any other piece of software).
Full project	A legacy mission serves as a basis for the new project, where major modifications are expected to meet the requirements.	Performing partial ISVV on the reused software based only on the modifications might not the best approach, because adaptations to the new project might also affect the not modified reused application. In such cases, instead of doing partial ISVV, a more thorough ISVV is required, according to the identified affected parts of the full project application.

Table 12: Reused software categorisation

Alternative and simplified categorization:

1. Component/System (black-box) – ISVV interfaces
2. Component/System (white-box) – partial to full ISVV
3. Component modified – partial to full ISVV

Where 1 can correspond to OS or libraries reuse, 2 can be mapped to partial functionality, execution platform or full project reuse, and 3 can correspond to partial functionality or execution platform reuse.

Independent V&V of auto generated code

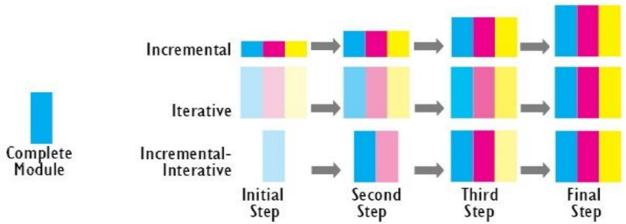
- Purpose:
  - How (if) shall ISVV assess autogenerated code?
- Analysis:
  - Surveyed stakeholders to gather expertise and recommendations;
  - Consortium/ESA expertise gathering and discussions.
- Conclusion:
  - When needed both models and readable code generated from models are provided for ISVV;
  - How to proceed when only the models/model language code are provided (auto generated code not provided)? (Is this a real use case?)
  - What ISVV tasks shall be performed on the auto generated code for this case?
  - Are there other use cases that will require ISVV adaptation?

Independent V&V when using Model Based Techniques

- Purpose:
  - How to perform ISVV when the inputs come from Model Based development?
- Analysis:
  - Surveyed stakeholders to gather expertise and recommendations;
  - Consortium/ESA expertise gathering and discussions.
- Conclusion:
  - There is a lack of lessons learned and expertise on this topic in what concerns applying ISVV;
  - Practical/Common use cases involving model-based techniques are needed;
  - Existing ISVV tasks shall be updated/generalized to cover also the Mode-Based development inputs.

Independent V&V of SW developed following an iterative model

- Purpose:
  - How to adapt/integrate ISVV when the development follows an iterative model?
- Analysis:
  - Surveyed stakeholders to gather expertise and recommendations;
  - Consortium/ESA expertise gathering and discussions.
- Conclusion:
  - ISVV plan and activities shall be adapted according to typical cases;
  - Use cases of iterative development and relevant lessons learned are needed;
  - Iterative vs incremental?
  - Proposed solution in the next slide.



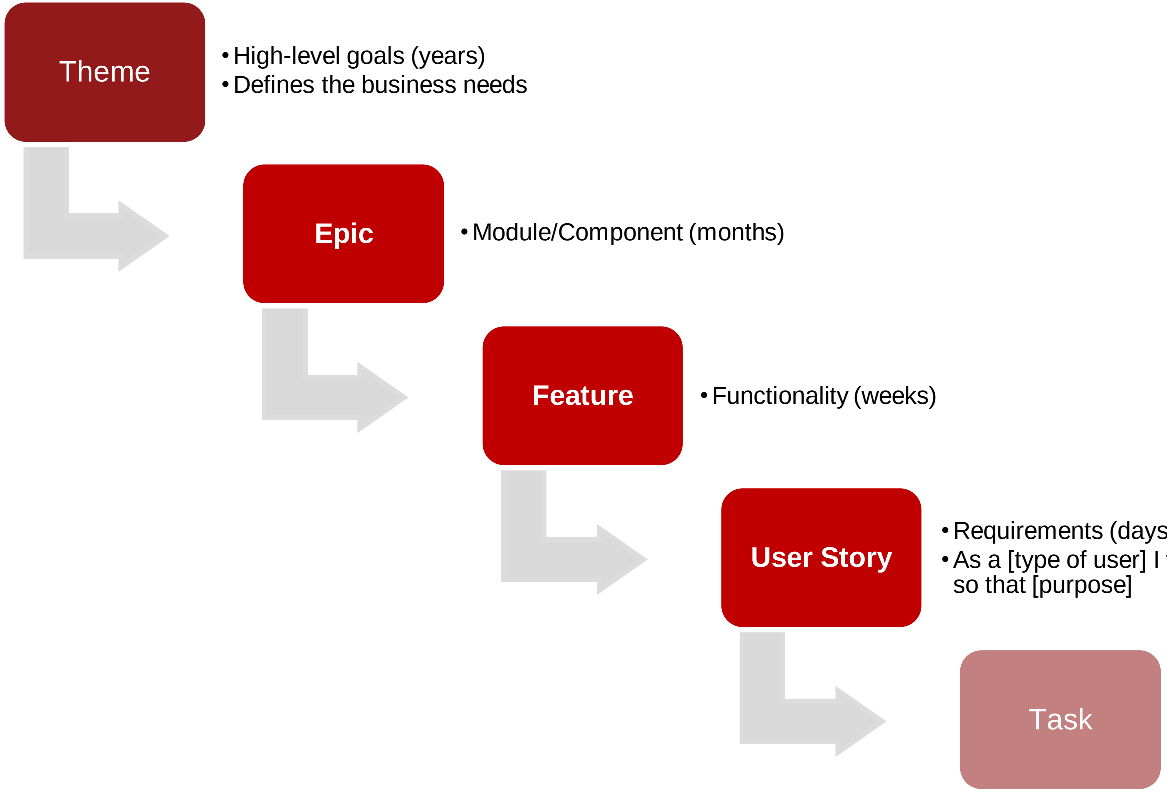
Independent V&V of SW developed following an iterative model

- a. Main ISVV activity planned (for instance when 80% of requirements implemented) and then a delta ISVV at the end (when all requirements are implemented);
- b. Normally the independent verification activities take place for each individual component as they become ready for ISVV. IVA activities are normally targeted for the overall SW components subject to ISVV at once and after verifications activities are completed or almost completed for each of the components;
- c. Delays and additional iterations shall be communicated between SW suppliers, ISVV costumer and ISVV supplier. It might be better to pause activities rather than working over unstable items;
- d. ISVV contract shall include a set of considerations, namely a basic ISVV effort for the tasks at hand, possibility of additional iterations, delays in iterations, architecture modifications at later iterations, evaluation of the iteration outputs maturity and decision on go-ahead with the ISVV activities.

Independent V&V of agile developed systems

- Purpose:
  - How to adapt/integrate ISVV when the development follows an agile methodology?
- Analysis:
  - Surveyed stakeholders to gather expertise and recommendations;
  - Consortium/ESA expertise gathering and discussions.
- Conclusion:
  - ECSS Agile Handbook is relatively new, there is no experience in applying it yet;
  - Use cases and lessons learned from agile-based projects are required to re-adapt the ISVV;
  - Where in the ISVV handbook shall the agile recommendations be included, together with the iterative?

Independent V&V of agile developed systems



- High-level goals (years)
- Defines the business needs

- Module/Component (months)

- Functionality (weeks)

- Requirements (days)
- As a [type of user] I want [an action] so that [purpose]

Independent V&V of agile developed systems

- ISVV to analyze the product backlog whenever independent verification activities are performed;
- ISVV performed when deliveries of a major SW datapack or during/after formal reviews in the case of hybrid methodologies that still consider ECSS reviews;
- ISVV performed during or after delta-reviews (if too many delta-reviews are expected, propose to perform ISVV only at a final stage);
- ISVV performed for the release of SW deliverables produced in every sprint (not recommended due to probable low ISVV efficiency and high impact on agile teams);
- Proposed types of ISVV performed in a continuous manner:
  - Full ISVV for each SW delivery;
  - Basic ISVV for intermediate deliveries and full for milestone review;
  - Delta ISVV for subsequent SW deliveries.

Topics Workshops

- Dedicated Workshops on the weeks of Nov 16th and 23rd
- Tue, Nov 17th, ISVV for Agile Workshop, 10AM
- Every day a set of topics are discussed with the stakeholders (online)
- No more than 4 hours of meetings / day (2 meetings)
- One or more (related) topics can be discussed per meeting
- A way forward shall be reached taking into account all presented arguments.

Acknowledgements

Q&A & Thank You!

Nuno Silva, PhD

nsilva@criticalsoftware.com

Jesper Troelsen

jtr@rovsing.dk

Andrei-Mihai Buzgan

andrei-mihai.buzgan@esa.int

ADCSS 2020