Critical software | esa

ROVSING

ISVV for Evolutions in Software Development Methods and Processes

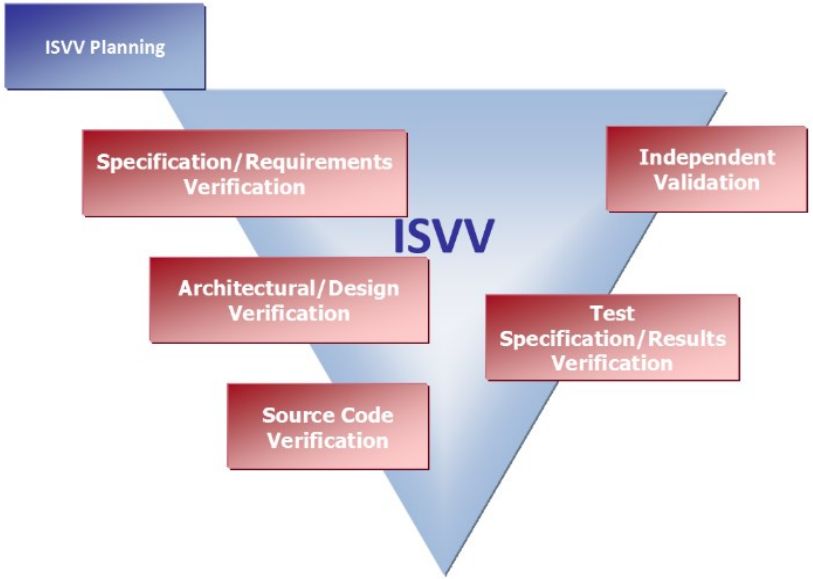
Author: Nuno Silva (Critical Software)

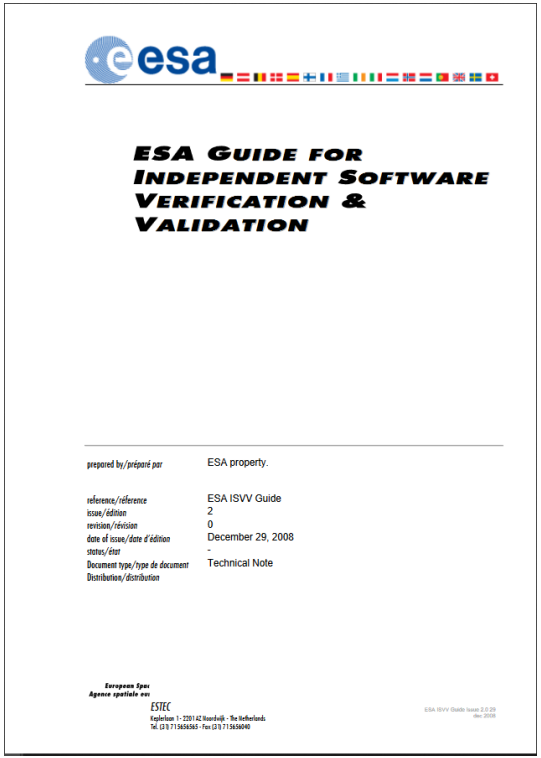
17th ESA Workshop on Avionics, Data, Control and Software Systems (ADCSS2023)
Presentation Date: 15/11/2023

Outline

- Introduction
- ISVV Handbook Generation Process
- ISVV Handbook Public Reviews
- ISVV Handbook Contents Impact
- Conclusions and Way Ahead
- Acknowledgements
- Q&A

Introduction





esa

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 Space Agency / Agence spatiale européenne
ESTEC
Keplerlaan 1 - 2201 AZ Noordwijk - The Netherlands
Tel. (31) 71 5656565 - Fax (31) 71 5656040

ESA ISVV Guide Issue 2.0 29 dec 2008

ECSS

Introduction

GSTP Project name: “**ISVV for Evolutions in Software Developments and Processes”**, ESA Contract: 4000127073/19/NL/AF

The ISVV Guide, was released in December 2008 and since then, it was the main driver for ISVV activities in European space industry.

In the past 15 years, the software development evolved dramatically and so did the space software industry.

The project was guided based on an initial set of requirements reflecting the experience gathered in over a decade of application of the ISVV Guide along with observations upon the evolution of software engineering in general.

Introduction

- Feedback was collected from a large set of stakeholders representing the industry, agencies and other users, and the change proposals were later discussed and agreed with the same stakeholders.
- After redaction, the ISVV Guide became the “ISVV Handbook”, prepared for ECSS adoption as an official Handbook.
- It was later reviewed by many representatives from industry and ESA.
- Since Dec. 2022 the new “ISVV Handbook” is published and recommended to be used for software in European space projects.
- This presentation highlights the main novelties of the new “ISVV Handbook”.

Introduction :: Modifications

Independent Verification

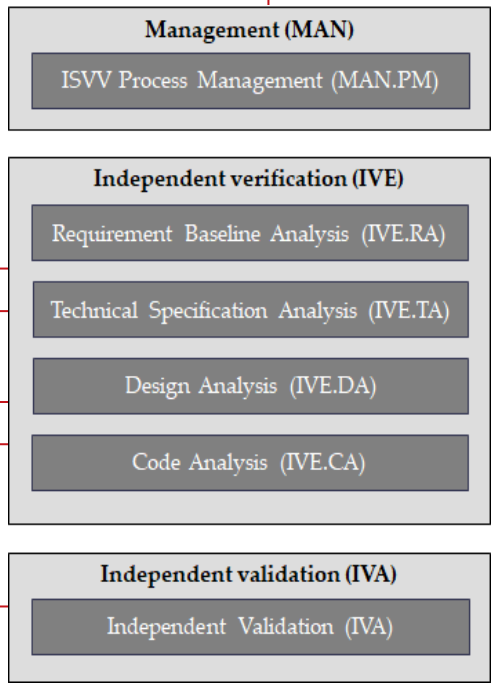
- New activity **Requirements Baseline Verification (IVE.RA)**: 1 task and 8 subtasks (e.g. SSS & IRD, FDIR, SCAR)
- Added IVV of **Configuration Data** (on all activities: TS, DA, CA and Test specifications)
- Removed **ISVV level definition**, added **Scope and Budget definition**, **ISVV Tailoring**, **ISVV Prioritization**
- Improved Verification of **reused Software** (IVE.TS, IVE.DA, IVE.CA ...)
- Added **UT test specs and test procedures** verification
- Added **Validation test specs, test data and test reports** (new) Verification whithin TS and RB (verification of validation environment, HSIA)

Independent Validation

- Added subtasks (e.g. IVA approach, System Validation ...)

Management

- Proposed **ISVV metrics** to be collected
- Added guidance on **ISVV Lessons Learned** collection
- Added guidance to **support different lifecycles** (mainly iterative) while performing ISVV



Introduction :: Modifications

Old Annexes

- Annex A. Definitions and Acronyms
- Annex B. ISVV activity outputs
- Annex C. Review Item Discrepancy Form Example
- Annex D. Summary of ISVV tasks, activities and methods and techniques
- Annex E. ISVV Levels and SW Criticality Categories
- Annex F. Methods
- Annex G. Checklists
- Annex H. Software Validation Facility
- Annex I. References

New Annexes

- Annex A. ISVV activity outputs & DRDs
  - A.1. SoW
  - A.2. ISVV implementation plan
  - A.3. ISVV implementation report
  - A.4. Final ISVV implementation report
  - A.5. Progress reports
  - A.6. Request for clarification
  - A.7. IVV findings resolution report
  - A.8. ISVV metrics
  - A.9. Lessons Learned
- Annex B. Review Item Discrepancy (RID) from example
- Annex C. Summary of ISVV tasks, methods and techniques
- Annex D. ISVV adaptation to different development life cycle models
- Annex E. Software Validation Facility
- Annex F. Methods and techniques description
- Annex G. Checklists
- Annex H. Configuration data types
- Annex I. ISVV tailoring

ISVV Handbook Generation Process

Stakeholders Questionnaire:	TN1: IVV Collection Improvements Technical Note
Handbook Contents Drafted:	TN2: IVV Assessment Improvement Technical Notes (16)
Handbook related Workshops:	ADCSS2020 (October 2020) + ESA/ESTEC Workshop (November 2020)
Handbook Drafting and Public Review	

ISVV Handbook Generation Process

- Most Active Participants (stakeholders and participants)
  - ESA / Critical Software / Rovsing
  - Airbus DS
  - OHB
  - Thales Alenia Space
  - TERMA
  - HULD
  - CAPTEC
  - GMV
  - Everis
  - Leonardo
  - NASA
  - JAXA
  - DLR
  - INPE

→ 86 Specific Proposed Changes stated as questions inside the former ISVV Guide.

→ Almost 1500 questions answered!

ISVV Handbook Public Reviews

- Implementation status
  - Rejected: 66 (22.8%)
  - Agreed: 202 (69.9%)
  - Postponed: 21 (7.3%)

- Total comments: 289
  - Trivial: 70 (24.2%)
  - Simple: 176 (60.9%)
  - Complex: 43 (14.9%)

- Postponed issues are the ones that require significant changes or feedback from the handbook application (to be covered during ECSS process)

- Reviewers Severity
  - Major: 45 (15.6%)
  - Minor: 211 (73.0%)
  - Comment/Editorial: 33 (11.4%)

ISVV Handbook Contents Impact

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

The groups, areas of improvement/topics, were requirements from the SoW meant for guidance. They drove the categories of changes/updates

- Solved contradictions
- Proposed Handbook Specific Contents
- Drafted it and got it publicly reviewed

ISVV Handbook Contents Impact

- 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)

ISVV Handbook Contents Impact

- 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)

ISVV Handbook Contents Impact

- 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

ISVV Handbook Contents Impact

- All TNs included:
  - ISVV Handbook Topic Assessment → From SoW + Questionnaire → **16 Technical Notes**
  - Summary of Improvements → **120 Changes in total**
  - Impact on ISVV Processes & Methods
  - Open Points → **20 Open Points in total**
  - Proposed Updates

  - Delivered for 16 dedicated workshops

ISVV Handbook Contents Impact

- Examples of most significant changes
  - ISVV level re-assessment / tailoring
  - ISVV Metrics / Lessons Learned
  - ISVV of Requirements Baseline
  - ISVV Life Cycle definition

ISVV level re-assessment / tailoring

I.4 Influencing factors

This section lists the main factors, which might influence the tailoring process and scope definition.

Software factors:

a. Software criticality and mission criticality.
   1. The ISVV supplier may also perform the criticality analysis by using a simplified FMECA method – see annex F.2.27.2.

b. Software characteristics:
   1. Complexity (language, models, algorithms).
   2. Size (lines of codes, number of requirements).
   3. Reusability (is the software component reused from previous project, will it be reused for future projects).
   4. Possibility to update in flight.
   5. Performance requirements.
   6. Type of code to be analysed: E.g., the programming language in which the code is developed, if code is auto generated; particularly, the inspection of auto generated code using custom code generators is an important factor.

c. The software development life cycle.

Experience and lessons learned:

a. Lessons learned from previous ISVV projects.

b. Experience and knowledge gained from this ISVV project (relevant for update of tailoring throughout the ISVV project).

Feasibility factors:

a. Experience and capabilities of the ISVV supplier

b. Access to tools and experience using tools

c. ISVV budget

d. Project risk register

e. External dependencies (e.g. access to SVF)

I.6 Prioritisation score

…

The following score can be applied to these elements:

a. Software Criticality (Crit), Score: A->10, B->7, C->4, D->0

b. Software Complexity (Comp), Score: 1-3

c. Software Reusability (Reus), Score: 1-3

The Prioritisation Score for the software component is calculated by:

Prioritisation Score = (Crit) + (Comp) + (Reus)

Some general rules based on the Prioritisation Score:

Prioritisation Score of >= 10: This software component has **"High"** priority for ISVV. All the tasks should be performed, and a broad variety of methods and tools should be used for verification and validation.

Prioritisation Score of 7-9: This software component has **"Medium"** priority for ISVV. Prioritisation between the tasks can be applied, but a broad variety of methods and tools should be used for verification and validation.

Prioritisation Score of <= 6: This software component has **"Low"** priority for ISVV. Certain task might be skipped for this component or only a few methods and tools could be selected for the verification and validation. The complete ISVV could also be skipped for this component in favour of software components with a higher prioritisation score.

ISVV Metrics / Lessons Learned

Table A-2: Catalogue of simple metrics

1. Metric
1.1. # RIDs per criticality (major, minor, comment).
1.2. # RIDs per type (external consistency, internal consistency, correctness, technical feasibility, readability & maintainability, completeness).
1.3. # RIDs per status (reported, accepted, rejected, corrected).
1.4. # RIDs per status per criticality.
1.5. # RIDs per criticality per ISVV subtask and task.
1.6. # RIDs per status per ISVV subtask and task.
1.7. # RIDs per type per status.
1.8. # RIDs per type per ISVV activity.
1.9. # RIDs that result in document modifications.
1.10. # RIDs that result in source code modifications.
1.11. # RIDs that result in behavioural changes in source code.
1.12. # physical lines of code, excluding comments (LOC).
1.13. # requirements.
1.14. # iterations of ISVV deliverables.
1.15. Range of hours spent on ISVV.
1.16. Range of hours spent on IVA activity.
1.17. IVA proposed tests metrics: • # tests proposed. • # tests executed. • # tests failed resulting in accepted RIDs. • # tests failed resulting in rejected RIDs (issue with SVF, out of scope, etc.).

Table A-5: Lessons learned questionnaire

4. Lessons learned
4.1. **Questions to the ISVV supplier:** • What was the type of the software under ISVV (AOCS, CSW, etc.)? • Did you experience any problems during the project and how were the problems dealt with? This could be problems related to: - Immature deliverables - Increase in scope - Estimations exceeded - Items in the risk register - CCNs - RIDs handling process • Did you apply any good practices that brought visible benefits to the project? • What methods used were found most efficient? • Were any methods used, which turned out to be less efficient than planned? • What lessons learned can be derived from the applied tailoring for the project?
4.2. **Questions to the 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 activity? (High | Medium | Low) • Have you encountered any problems during validation activity? And how were the problems dealt with? This could be problems related to: - SVF stability, installation and usability - SVF training - Execution of test cases - Issues found during test - Own SVF enhancement - Tests results ambiguity - Partly done ISVV - Possible CCN • Did you apply any good practices that brought visible benefits to the project?
4.3. **Questions to the end customer/ISVV customer/SW supplier:** • What is your general feedback about value created and efficiency of the ISVV project? • What good practices, tasks, improvements have brought a significant added value to the ISVV project? • Are there any areas that should be improved? • Was the tailoring appropriate for the ISVV project?

ISVV of Requirements Baseline

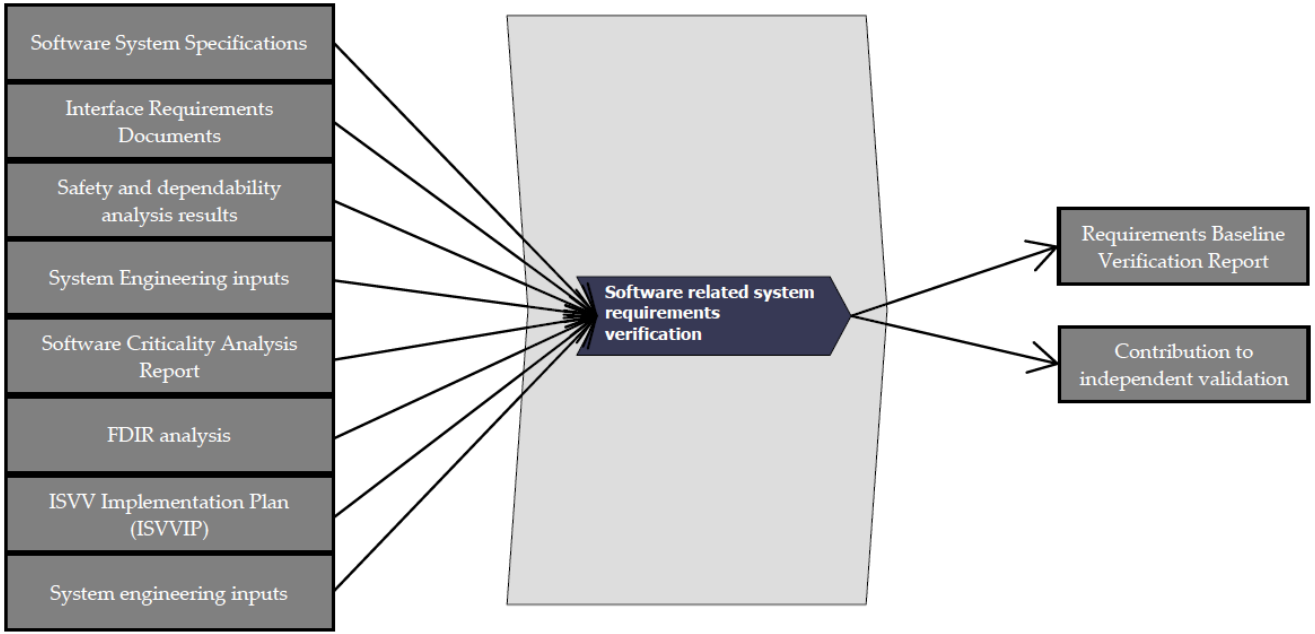


Figure 6-1: Requirements Baseline Analysis

ISVV life cycle definition

- Annex D
  - Life cycle models and concepts
  - Life cycle definition (development stages, reviews, data packs, adaption to iterative developments)
  - Initiating and end events
  - Recommendations of ISVV life cycles (sequential vs iterative)

Conclusions and Way Ahead

- Handbook available (not ECSS approved yet)
- List of Postponed issues (some might lead to changes, some not)
- Feedback on HBK applicability is requested/collected, namely for the more significant changes (RB ISVV, LL and Metrics collection, applicability to new lifecycles, ISVV tailoring…)
- Gathering statistics in a more consistent way through means of metrics and lessons learned.

Conclusions and Way Ahead

Future View on ISVV

- Collection of Lessons Learned and Metrics → Feedback
- New Methods and Tools for ISVV → Adaptation, Efficiency
- ISVV Process Evolution → As SW Engineering also evolves
- ISVV Laboratory / Central Repository → Relevant Data/Resources access
- RB and TS verification effectiveness → Bring added value to the whole process
- Look into postponed public review comments → Together with ECSS process
- Etc.

Conclusions and Way Ahead

- ECSS Secretariat will include the new ISVV Handbook in the upcoming ECSS updates
- HBK available for download at (registration needed):
  - https://essr.esa.int/project/independent-software-verification-and-validation-handbook
  - ISVV contact address: isvv@esa.int

Aknowledgements

- All stakeholders participating in the initial questionnaire, the ISVV Workshops and the public review
- ESA Reviewers

Thank You!

Q&A

Nuno Silva, PhD
nsilva@criticalsoftware.com

Andrei-Mihai Buzgan
andrei-mihai.buzgan@esa.int

Pedro Barrios
pedro.barrios@esa.int