DOCUMENT

Guidelines for the use of TRLs in ESA programmes

Prepared by	**ESA TRL Working Group**
Reference	**ESSB-HB-E-002**
Issue	**1**
Revision	**0**
Date of Issue	**21 August 2013**
Status	**Approved/Applicable**
Document Type	**Handbook**
Distribution	

APPROVAL

Title Guidelines for the use of TRLs in ESA programmes	
Issue 1	**Revision** 0
Author ESA TRL Working Group	**Date 21 August 2013**
Approved by ESB on behalf of ESSB	**Date 7 November 2013**

CHANGE LOG

Reason for change	Ref/Issue	Revision	Date
First issue	1	0	21 August 2013

CHANGE RECORD

Issue 1	**Revision** 0		
Reason for change	**Date**	**Pages**	**Paragraph(s)**

Table of contents

Tables

1 Scope

This document is an addendum to ISO TRL definition document ISO 16920 providing technology readiness level definitions for space hardware and their assessment.

The document provides:

- A description of ISO scale evolution with respect to the previous scale used in ESA
- General guidelines for TRL use in ESA Programmes
- Additional elements for enabling the use of TRLs for software developments

2 References

ISO 16290:2013	Space systems - Definition of the Technology Readiness Levels (TRLs) and their criteria of assessment
ECSS-S-ST-00-01C	ECSS system - Glossary of terms
ECSS-E-ST-10C	Space engineering - System engineering general requirements
ECSS-E-ST-40C	Space engineering - Software
ECSS-Q-ST-80C	Space product assurance - Software product assurance
STM-277	ESA Technology Tree

3 Terms, definitions and abbreviated terms

3.1 Terms from other documents

For the purpose of this document, the terms and definitions from ISO 16290 apply.

NOTE Some definitions from ISO 16290 can have different wording from those in ECSS-S-ST-00-01C.

3.2 Abbreviated terms

For the purpose of this document, the following abbreviated terms apply:

Abbreviation	**Meaning**
ASIC	application specific integrated circuit
EQM	engineering qualification model
EM	engineering model
FM	flight model
FPGA	field-programmable gate array
GSTP	general support technology programme
HDL	hardware description language
HW	hardware
PC	personal computer
PFM	proto-flight model
QM	qualification model
QPL	qualified parts list
R&D	research and development
SW	software
TRL	technology readiness level
TRP	technology research programme
TSIM	target simulator (product from Gaisler Aeroflex)
V&V	verification and validation

4
ISO TRL scale

The ISO TRL Working Group successfully converged to a unified international scale endorsed by all major space actors, with a common understanding of the associated definitions. ISO TRL scale should become a standard in space business. ISO TRL scale is slightly different from the previous ESA, NASA, and US DoD respective TRL scales, which were also different one from each other and subject to various interpretations.

The evolution with respect to the “old scale” used in ESA affects TRLs 5, 6 and 7, and is further detailed in Table 4-1:

- TRL 5 (old scale) is split in two levels TRL 5 and TRL 6 in the new ISO scale.
- TRLs 6 and 7 (old scale) are merged in a single level TRL 7, representing qualification through on-ground or in-orbit validation, as needed.

Table 4-1: Comparison of ISO TRL scale and ESA old TRL scale

<table>
<tr><th colspan="2">TRLs in old ESA scale</th><th colspan="2">TRLs in new ISO scale</th></tr>
<tr><td colspan="2">TRLs 1 to 4</td><td colspan="2">TRLs 1 to 4 are basically unchanged</td></tr>
<tr><td rowspan="2">TRL 5</td><td rowspan="2">Critical functions verification in representative environment with representative scale breadboards</td><td>TRL 5</td><td>Same definition as TRL 5 old scale, but allowing reduced scale breadboard verification. Most useful for the development of large pieces (telescopes, structures) and for launcher developments.</td></tr>
<tr><td>TRL 6</td><td>Same as TRL 5 old scale</td></tr>
<tr><td>TRL 6</td><td>Qualification through on ground verifications</td><td rowspan="2">TRL 7</td><td rowspan="2">Qualification level, through validation on ground or in orbit, as needed</td></tr>
<tr><td>TRL 7</td><td>Qualification through in-orbit demonstration</td></tr>
<tr><td colspan="2">TRLs 8-9</td><td colspan="2">TRLs 8-9 are basically unchanged</td></tr>
</table>

5 Use of TRLs in ESA programmes

5.1 Use of TRLs in early phases

All existing space missions in the Agency have been implemented in three major steps:

1. Definition (Phases 0/A/B) where the mission is elaborated and relevant technologies are prepared if not available,
2. Industrial implementation phase (Phase C/D), where the space segment is manufactured,
3. Launch and operation in orbit for delivering the mission products to the users.

The implementation details are Programme dependent, and sometimes mission dependent. They always involve specific non-technical constraints such as industrial policy or expenditure profiles in specific countries.

However, in all Programmes, the decision to proceed with the industrial implementation phase does constitute a major decision, generally irreversible, and represents a heavy financial commitment carrying its inherent risks for ESA and for the Participating States. TRLs are useful for evaluating the technology maturity of the space segment and assessing the associated development risks before taking such decision. Note that in the current ESA procedures, this decision is taken at the end of Phase B1, where competition is still running at Prime Contractor level and sometimes at mission level. It is worth noting that the purpose of the technology readiness evaluation is not to monitor the Programme decision process, but to enable the Programme to take informed decisions. Whatever the programme decision process, it is recommended to assess the technology readiness of the space segment before the development is in full swing mode. The use of TRLs in project phases is developed in Table 5-1.

The technology readiness assessment should be made by an independent review as part of the regular project reviews, which include in addition to technology readiness the evaluation of:

- The design maturity of the space segment,
- The development plan – including the verification approach, schedule assessment, procurement approach, risk assessment and cost estimates.

Table 5-1: Potential use of TRL assessment in various project phases

Project Phase	Potential use of TRL assessment
Phase 0 (end of study)	• Consolidate technology development plan • Re-orient the design for improving technology readiness and implementation decision schedule
Phase A (PRR)	• Assess progress in the technology development plan implementation • Consolidate technology development plan • Selection criteria for competing missions
Phase B1 (SRR)	• Readiness to move to implementation phase, (B2/C/D): TRL 5 or higher for the old ESA scale, TRL 6 or higher for the new ISO scale (see Table 4-1) • Selection criteria for competing missions
Phase B2 (PDR)	• Assessment of equipment supplier proposals • Decision to place a contract: minimum TRL 6 is required • At PDR: Confirm readiness to move in phase C/D when relevant
Phases C/D/E	Low interest

NOTE 1: While TRL 5/6 (new scale) can be a suitable threshold for entering the implementation phase with acceptable risks, there is no link between lower TRLs and early study phases (e.g. Phase 0 cannot be associated or conditioned to a TRL). A good Phase A study should propose a concept that meets all technical and programmatic constraints, including technology readiness. If not possible, time should be allocated between Phase A and the start of Phase B for implementing technology developments and raising TRLs to the 5/6.

NOTE 2: The technology readiness evaluation should not be confused with the design maturity evaluation for TRLs below 7. Technology readiness and design maturity are two distinct faces of the same element providing together the complete picture for feasibility assessment. When TRL7 is reached (qualification), the design maturity is naturally fully reached and the element performance is demonstrated to meet the requirements in the relevant environment. However, for lower TRLs - in particular for the critical levels TRL 5/6 where the element is not fully built - the TRL assessment is done in parallel with the design maturity assessment, by relying on a preliminary design of the element. At this point, the element (e.g. a spacecraft) can be declared to fully rely on mature technologies, but found unfeasible (e.g. because its mass is under-estimated and not compatible with the launcher, or its thermal concept is not valid and underestimating heat leakage).

5.2 Technology readiness threshold for implementation phase

It is conceptually easy to require a very high level of technology maturity for all the components of the space segment of a given mission, for example by systematically asking for a QM or EQM (e.g. TRL 7 in the ISO scale). This approach was actually followed in some early space developments and would obviously drop the development risks close to zero.

Today, this approach is no more affordable and is neither desirable in terms of implementation cost efficiency. It would basically mean building a model of the entire space segment prior to taking the implementation decision, which is financially meaningless and technically not justified.

With ISO new scale, TRL 6 corresponds to the appropriate technology readiness level for supporting the decision to go for implementation with acceptable risks. TRL 6 requires that the critical functions of the element are demonstrated in the relevant environment. Reaching this level for the space segment means the following:

- The critical elements of the space segment, not relying on mature technologies, have been identified,

- The associated uncertainties and the minimum set of verifications have been defined,
- Test vehicles or breadboard models for removing the unknowns have been developed and successfully verified in the relevant environment that is applicable to the mission.

A reliable development schedule for a spacecraft can be established only when TRL 6 is reached. Conversely, when a spacecraft element is not at TRL 6, the development schedule still has uncertainties, since the element can fail in performing in the mission relevant environment. In some cases, this can lead to a major re-design of the space segment at all levels and jeopardize the whole mission concept.

TRL 5 is an intermediate readiness level that could also be considered to go for implementation with acceptable risks. There are however remaining risks related to scaling effects. The effective risk in starting a Project implementation at TRL 5 will have to be discussed on a case by case basis. As an illustrative example, Herschel telescope development was decided at TRL 5. Reaching TRL 6 for this particular case would have required building the full scale primary mirror, which is not far from the entire telescope in the specific case of Herschel.

NOTE Technology Readiness Levels below TRL 6 are not necessarily implying the element cannot be developed – a number of things can be done with unlimited funding and schedule - but imply major uncertainties in the development schedule.

5.3 Guidelines for the technology readiness assessment

Table 1 in ISO 16290 details the activities to be achieved and documented for each TRL. Additional recommendations are provided here below, addressing specifically the technology readiness evaluation:

- A crucial step is the identification of critical technology elements and of the associated unknowns and critical verifications to be performed. The number of critical elements is expected to be very low for a well-defined mission. Elements can be gradually removed from the list of critical technology elements by using pre-development results or by similarity with elements used in previous missions.
- As input to the review process, a specific document addressing technology readiness should be requested. This document should include the identification of critical elements, the definition of related uncertainties, and technology plan addressing how these critical elements are or have been verified/validated: definition of the test vehicles and of their verification/validation, with the status of resulting tests. Annex A provides the correspondence between the development models (e.g. EM, QM, FM) and the ISO TRL scale and should be considered in producing the dedicated input documentation for the reviews.
- It is recommended that the review panel starts the technology readiness evaluation by reviewing the identified critical elements, the related unknowns and the test vehicle definition for removing the uncertainties. Experience indicates that critical elements are generally correctly identified at early stages of the mission definition, while failures often occur either in the proper identification of the unknowns (example: Bepi-Colombo solar cells qualification for the mission environment) or in the lack of representativeness of the test vehicle resulting from the underestimation of coupling effects between the element components (example:

Aeolus/Aladin laser pre-development model representativeness). In case of disagreement, complementary information will be requested from the study/project team.

- The review panel should assess the test or verification/validation results associated to each critical element. In case of non-conclusive results, the review panel shall identify at the lowest possible level the origin of the problem.
- For each critical element that cannot be declared at TRL6 at the time of the evaluation, the project team should be requested to either provide a back-up scenario or explain why the critical validations can likely be concluded in the course of the development phase, with mastered schedule and cost impact. The latter exercise is by nature difficult and controversial, but is mandatory for enabling a Programme management decision.
- The back-up scenario should rely on alternative mature technologies and the degradation impact on the mission requirements shall be evaluated and acknowledged.

6 Guidelines for the use of TRLs in ESA technology development activities

R&D programmes can be seen as technology providers with the goal of delivering technologies at a given level of maturity. The TRP nominally provides Technology up to TRL level 3, although in some individual cases this may go up to TRL 4 or even higher. Target TRLs in GSTP technology activities can vary between TRL 3 and TRL 7.

The evaluation of the results achieved by ESA R&D programmes is one of the key elements of the ESA End-to-End process. In particular, the key issue is the evaluation of the actual TRL achieved and the comparison with the target TRL envisaged at the beginning of the R&D activity.

It is proposed to maintain a simple procedure in the TRL assessment for basic technology activities achieved through industrial contracts using TRP or GSTP funding.

For TRL ≤ 4, the Technical Officer involved in the conduct of the R&D activity should lead the collection, review and verification/validation of the achieved TRL, in accordance with the Table 1 of ISO 16290.

For elements not targeting a specific Project (e.g. technology activity aiming at an equipment for possible use in several missions) and for TRL ≥ 5, the TRL evaluation should involve the relevant Technology Domain Responsible (as defined in document STM-277).

7
Use of TRLs for ESA software developments

7.1 ISO TRL scale and software developments

ISO TRL definition does not address the use of TRLs for software and there is no international or even European uniform approach for using TRLs for software developments. For convenience and for avoiding the introduction of a specific scale for software, it is proposed to use the same ISO scale for software developments by providing a clear definition of the expected development state at each TRL.

For that purpose, software specific definitions are provided in section 7.2, and basic principles underlying software developments are exposed in sections 7.3. Annex B provides further details and examples.

7.2 Software specific definitions for the purpose of this document

7.2.1 software building blocks

software element that has an identifiable function within a more complex (software) system, and that can potentially be reused for a range of applications

7.2.2 software tool

software element that performs a function in a stand-alone mode

7.3 Basic principles

Software TRL (SW TRL) shall be applied to assess the maturity of technologies implemented in software which may be part of the flight segment (flight software), ground segment (ground software) or engineering tools (software tools).

Due to their very different development and application characteristics, three types of software need to be identified for the purpose of TRL definition:

1. Software tool. SW that runs in a stand-alone mode, i.e. without requiring a specific input/output simulator
2. Embedded Software. SW that necessarily interacts with other software and possibly also with HW. Two categories exists as follows:
   (a) Building block: embedded SW conceived to be reused in a range of missions, either flight or ground software. This software is executed as part of a larger software application.

      NOTE This category includes IP cores for micro-electronics (e.g. FPGA, ASICs).

(b) Specific embedded Software: SW that is targeting a specific application and that is not conceived to be reused in another domain of application (e.g. equipment embedded software).

NOTE This category includes Hardware Description Language for micro-electronics (e.g. FPGA, ASICs).

This section clarifies the notion of SW TRL for the SW types 1 and 2.a, For the SW type 2.b, the TRL ISO classification is applicable as is (the SW is part of the HW TRL assessment).

As for HW TRL, the SW TRL levels are not meant to be applied to the management of a software development project, for which typically the software engineering standard (e.g. ECSS-E-ST-40 and ECSS-Q-ST-80) is applied. The SW TRL is then simply a tool for the evaluation of the maturity of a given software technology (building blocks, tools) within the context of its intended application.

The underlying principles are summarized below:

- **TRL 1 to 4** cover the beginning of the development by increasing the level of implemented functionality, from the mathematical formulation and through prototyping and incremental enhancement:
  - For a SW tool, TRL 4 corresponds to the “alpha” version.
  - For a building block, TRL 4 corresponds to “pre-product prototype”.

- **TRL 5 and 6** cover the transformation of the prototype into a product with frozen requirements. A pre-qualification data package is produced by making use of the ECSS-E-ST-40 and ECSS-Q-ST-80 standards, giving confidence that the product will perform as expected in the final environment:
  - For a SW tool, TRL 5 corresponds to the “beta” version and TRL 6 corresponds to the first release version.
  - For a building block, TRL 6 corresponds to the released product verified with a simulated environment.

- **TRL 7** corresponds to the SW qualification for the foreseen application verifying the SW performance in its intended environment.
  - For a SW tool, this corresponds to full validation on a representative pilot case.
  - For a building block, this corresponds to successful qualification as part of the foreseen application.

- **TRL 8** corresponds to the final product acceptance for operation.
  - For a SW tool, it corresponds to the readiness for the full deployment in operation.
  - For a building block, this corresponds to a successful final acceptance of the product embedding the SW.

- **TRL 9** corresponds to successful operations and performance achievement in the foreseen application.

Annex B provides further details for SW TRL understanding and use.

NOTE 1 The terms “alpha” and “beta” versions are used by analogy with names used out of space domain, but does not intend to carry any of these non-space meanings. Instead, the TRL 4 and 5 are defined in Annex B.1.

NOTE 2 Software criticality, as defined in relationship with dependability and safety according to the consequences of failures, is not linked with the maturity of a piece of software described by the TRL. They are independent.

Annex A - Models associated to ISO TRLs

TRL	ISO definition	Associated model	Performance requirements	Representativity of environment & test	Comment and practical use
1	Basic principles observed and reported	Not applicable	In elaboration	No	Seldom considered in ESA developments
2	Technology concept and/or application formulated	Not applicable	In elaboration	No	Seldom considered in ESA developments
3	Analytical and experimental critical function and/or characteristic proof-of-concept	Mathematical models, supported e.g. by sample tests	Partly defined	No	Use in technology developments for monitoring progress
4	Component and/or breadboard functional verification in laboratory environment	Breadboard	Partly defined	No	Use in technology developments for monitoring progress
5	Component and/or breadboard critical function verification in a relevant environment	Scaled EM for the critical functions	Fully defined	Yes, for critical functions subject to scaling effect	Could be used as threshold for enabling the start of implementation phase (C/D), subject to risks related to scaling effects
6	Model demonstrating the critical functions of the element in a relevant environment	Full scale EM, representative for critical functions	Fully defined	Yes, for critical functions	Critical threshold for enabling the start of implementation phase (C/D) with mastered schedule
7	Model demonstrating the element performance for the operational environment	QM	Fully defined	Yes	QM validated, generally during the implementation phase C/D Note: project may allow EQM or PFM instead of QM
8	Actual system completed and "flight qualified" through test and demonstration	FM acceptance tested, integrated in the final system	Fully defined	Yes	Qualification of ground achieved, last step before launch of most of space developments
9	Actual system completed and accepted for flight ("flight qualified")	FM, flight proven	Fully defined	Yes	Corresponds to mature technology

Annex B - Correspondences for the use of TRLs for software in ESA programmes

B.1 Correspondence table for the use of TRL for software in ESA programmes

TRL	Engineering terms relevant to SW	Additional explanation to cover software	Description	Requirements	Verification	Viability
1	MATHEMATICAL FORMULATION	Scientific knowledge	Detailed mathematical formulation description. Publication of research results.	Expression of a problem and of a concept of solution.	Proven mathematical formulation.	Feasibility to be implemented in software with available computing facilities demonstrated.
2	ALGORITHM	Individual algorithms or functions are prototyped	Algorithm implementation documented. Results documented.	Practical application identified. A concrete specification of a part of the problem.	Single algorithms are tested resulting in their characterization and feasibility demonstration.	feasibility to build important functions in a system architecture demonstrated.
3	PROTOTYPE	Prototype of the integrated main system	Architectural design of important functions is documented. Depending on size and complexity of the implementation.	Some solutions to a range of problem. Main use cases implemented.	A subset of the overall functionality is implemented and tested to allow the demonstration of performance. V&V in a simulated laboratory environment.	Feasibility to build an operational system taking into account performance and usability aspects demonstrated.

TRL	Engineering terms relevant to SW	Additional explanation to cover software	Description	Requirements	Verification	Viability
4	ALPHA version	Most functionality implemented.	Documentation as for TRL 3 plus: • User Manual • Design File	Clear identification of the domain of applicability. Requirements for solutions to a range of problems specified. All use cases implemented.	Verification & Validation process is partially completed, or completed for only a subset of the functionality or problem domain. V&V in a representative simulated laboratory environment.	Feasibility to complete missing functionality and reach a product level quality demonstrated.
5	BETA version	Implementation of the complete software functionality.	Full documentation according to the applicable software standards, including test reports and application examples.	Formal definition of the domain of (re)use and associated variability features of the implementation. All use cases and error handling specified.	Validated against the requirements of the complete domain of applicability including robustness. Quality assurance aspects taken into account. V&V in an End-to-end representative laboratory environment including real target.	Feasibility to fix all the reported problems (e.g. all open SPRs) within available resources demonstrated. User support organization in place.

TRL	Engineering terms relevant to SW	Additional explanation to cover software	Description	Requirements	Verification	Viability
6	Product RELEASE	Ready for use in an operational/ production context, including user support.	Documentation according to the applicable SW eng and Quality standards for a software product.	Building block: Process for reuse, for instantiation in the domain of the implementation and its test environment. Tools: All use cases and error handling implemented. User friendliness validated.	Building block: Validated against the requirements of the complete domain, validation environment also reusable, reuse file available. Tools: Verification and Validation process is complete for the intended scope, (including robustness. Configuration control and Quality assurance processes fully deployed. V&V in an End-to-end fully representative laboratory environment including real target.	Feasibility to be applied in an operational project demonstrated. Availability of a data package suitable to support future qualification.
7	Early adopter version	Building block: qualified for a particular purpose Tool: ready for market deployment	In addition to TRL 6 Documentation, updates to documentation and qualification files SPR database Lessons learnt report	Requirements traced to mission requirements. Validity of solution confirmed within intended application. Requirements specification validated by the users.	Building block: Integrated in the spacecraft following the applicable software standards Tools: The tool has been successfully validated in a pilot case, representative of the intended project application	Engineering support and maintenance organization in place, including helpdesk

TRL	Engineering terms relevant to SW	Additional explanation to cover software	Description	Requirements	Verification	Viability
8	General product	Ready to be applied in the execution of a real space mission	Full documentation including specifications, design definition, design justification, verification & validation (qualification file), users and installation manuals and software problem reports and non-compliances. Including also qualification files, SPR database. Lessons learnt report.	Requirements traced to mission requirements. Validity of solution confirmed within intended application. Requirements specification validated by the users.	Building block: Integrated in the spacecraft/ground segment and completed successfully system qualification campaign. Tool: the tool has been successfully applied in an operational project but has not yet been validated against the in-flight experience	Engineering support and maintenance organization in place, including helpdesk. Capability for in-orbit data exploitation and post flight analysis. Capabilities.
9	Live product	Has been applied in the execution of a real space mission	In addition to TRL 8 Updates to documentation and qualification file SPR database Lessons learnt report Track record of application in space projects	Building block: Maintained Tools: Full process implemented, Maintenance, updates, etc.	Building block: fully validated for the mission and qualified for intended range of applicability. Tool: the tool has been successfully validated in one or several space missions, including exploitation of in-orbit data. All anomalies encountered have been analyzed and resolved.	Sustaining engineering, including maintenance and upgrades in place

B.2 Comments for the case of SW building block

The range of targeted application is translated into the domain of reuse of the building block.

- For low TRL levels, requirements for the SW application are not frozen. The differences between the 4 first levels relates to the:
  - Range of problems and the variability of the solutions,
  - Amount of functionality implemented,
  - Level of V&V.
- At TRL 5, the requirements are frozen. Then, the building block becomes mature enough and its domain of reuse has been established.
- TRL 5 and 6 are to reach pre-qualification level by verifying the SW performances in a simulated environment.
- In TRL 7, the SW is qualified for the foreseen application in its intended environment.

Several levels of validation environment are used in the table in B.1:

- Simulated (in TRL 3): the software building block is executed in a target simulator, which is a piece of software that emulates the behaviour of the actual processor (e.g. TSIM).
- Representative simulated (in TRL 4): this simulator is representative in terms of
  - actual process functions,
  - the real-time behaviour performance and accuracy,
  - the simulation of the hardware elements connected to the actual processor, e.g. the device drivers, bus drivers.
- End to end representative including real target (in TRL 5 and 6): the software building block is executed on a hardware board with the actual processor chip. The end-to-end aspect denotes the fact that the interface behaviour is fully representative, possibly implying the use of HW. For flight SW, the mission environment requirements are fully taken into account (e.g. radiation impacts).

B.3 Examples

B.3.1 Example for software building block

A typical software building block is an on-board operating system.

- **TRL 1**: the mathematical formulation of the scheduling theory is done.
- **TRL 2**: there is a prototype of the scheduling algorithm itself, without any hardware or application context, but scheduling with this particular algorithm is feasible.

- **TRL 3**: there is an architecture that shows not only the scheduling algorithm, but also the interrupt management, the semaphores, the relation to hardware timers, etc. Integrating the algorithm into an operating system is feasible.
- **TRL 4**: the operating system has a specified interface for application software users, all expected functions of an operating system are implemented, but not all are tested (e.g. the priority inversion protection). The operating system has been validated by running in a simulator of the target processor, itself being a software running on a standard PC.
- **TRL 5**: the domain of use of the operating system is defined, in terms of target processors (e.g. ERC32, Leon, PowerPC), communication capabilities (e.g. SpaceWire driver, 1553 drivers) or operational capability (e.g. maximum number of priority, tasks, semaphores). The operating system is validated for all the parameters values and hardware environment that are foreseen in the domain of reuse. The validation environment is a hardware board with a representative target processor and hardware communication drivers.
- **TRL 6**: a formal qualification data package (in the meaning of the software standards applied in the foreseen criticality level) is available and approved by software product assurance. It constitutes a qualification credit that can be invoked in a project. The process to be used in order to delta-qualify the operating system in the user project is defined. Support to users is organized (helpdesk). The operating system can be called a “product” and can be proposed to users.
- **TRL 7**: a spacecraft project has selected the operating system for its flight software. Therefore, the parameters for the specific use have been selected: target processor, communication drivers and maximum sizes and ranges are set. There is a successful qualification of the operating system with these values, in the intended environment, The validation environment includes the actual hardware of the project.
- **TRL 8**: the SW is integrated in the final spacecraft that has been accepted and is ready for flight.
- **TRL 9**: the spacecraft has been launched and the operating system is nominally functioning.

B.3.2 Example for software tool

A typical software tool is a software compiler (or a HDL compiler in microelectronics).

- **TRL 1**: the algorithm related to parsing source code to generate machine code or gates, in one or several passes, exists.
- **TRL 2**: there is a set of prototypes that reads a selection of the source code syntax and generates machine code using part of the instruction set.
- **TRL 3**: the architecture of the compiler is defined, and the complete source code syntax and semantics is covered.
- **TRL 4**: the Alpha version of the compiler has a primitive man machine interface, generates non optimized machine code, and the execution time is slow. It is validated using typical examples of source code.

- **TRL 5**: the Beta version of the compilers has optimized the machine code generation, the performance and the ergonomy of the man machine interface. A reference test suite of source code has been established to validate the compiler, and the generated object code runs on the hardware processor.
- **TRL 6**: the compiler is a Product with a good documentation and acceptable performances. It produce error messages which are complete and user friendly. The support is organized, as well as the product packaging and delivery.
- **TRL 7**: the compiler is delivered to early adopters for extensive testing. Then, the compiler robustness is improved following user feedbacks.
- **TRL 8 and 9**: the compiler is deployed to the complete user community.