

MQP Level 2

Software Qualification Policy

Any software used for R&D, design, manufacturing, installation, construction, commissioning, and operation of ITER facility shall be appropriately qualified prior to approval for use. The qualification shall be done to demonstrate that the software adequately and correctly performs all intended functions and user needs.

Approval Process			
	<i>Name</i>	<i>Action</i>	<i>Affiliation</i>
<i>Author</i>	Bourgue R.	06 Aug 2021:signed	IO/DG/CORP/IT
<i>Co-Authors</i>	Stepanov D.	11 Aug 2021:signed	IO/DG/SCOP/SCOD/CD/DCS
<i>Reviewers</i>	Dumarcher V.	05 Sep 2021:recommended	IO/DG/SCOP/SCOD/OPD
	Elbez-Uzan J.	24 Aug 2021:recommended	IO/DG/SQD/EPNS
	Khomutnikov A.	25 Aug 2021:recommended	IO/DG/SQD/QMD
	Klora J.	16 Aug 2021:recommended	IO/DG/CORP/IT
	O'Connor N. *	16 Aug 2021:recommended	IO/DG/CNST/CMO/SPC
	Pinches S.	07 Sep 2021:recommended	IO/DG/SCOP/SCOD/SCD/PMA
	Salamon B.	01 Sep 2021:recommended	IO/DG/ENGN/CIO/CMD/DCC
	Schioler T.	11 Aug 2021:recommended	IO/DG/ENGN/CIO/IEA
	Stepanov D.	12 Aug 2021:recommended	IO/DG/SCOP/SCOD/CD/DCS
<i>Approver</i>	Wallander A.	20 Aug 2021:recommended	IO/DG/SCOP/SCOD/CD
	Tada E.	08 Sep 2021:approved	IO/DG/CORP
Document Security: Internal Use			
RO: Khomutnikov Aleksei			
<i>Read Access</i>	GG: MAC Members and Experts, AD: ITER, AD: External Collaborators, AD: External Management Advisory Board, AD: OBS - Quality Management Division (QMD), AD: DA, AD: Auditors, AD: ITER Management Assessor, project administrator, RO, AD: IO_Director-General, AD: OBS - Integrated Engineering Analyses Se...		

<i>Change Log</i>			
Software Qualification Policy (KTU8HH)			
<i>Version</i>	<i>Latest Status</i>	<i>Issue Date</i>	<i>Description of Change</i>
v0.0	In Work	07 Oct 2013	
v1.0	Signed	07 Oct 2013	First issue. Document created as per MQP doc Request C2L5NL.
v1.1	Revision Required	15 Nov 2013	Verbal and written comments of reviewers were incorporated.
v1.2	Approved	06 Feb 2014	Incorporated reviewers' comments; improvement of descriptions of purpose, scope, and definition for users of document.
v2.0	Approved	06 Aug 2021	As per approved MQP doc request https://user.iter.org/?uid=428VUG the changes are: After discussion with ENGN, current implementation calls for process simplification: remove the need for verification of each installation of already validated design software. Change impacts chapters 5 and 6.2.

Contents

1	Purpose	2
2	Scope	2
3	Definitions and Acronyms.....	2
3.1	Definitions	2
3.2	Acronyms	4
4	References.....	4
5	Basic Principle.....	4
6	Software Qualification Methodology	5
6.1	Embedded Software	5
6.2	Software used for Design or Technical Justification.....	5
6.3	External Software	5
7	Software Qualification Process.....	6
7.1	Planning.....	6
7.1.1	<i>Assignment</i>	<i>6</i>
7.1.2	<i>Qualification Plans / Procedures</i>	<i>6</i>
7.2	Execution.....	6
7.3	Records.....	6

1 Purpose

Any software used for R&D, design, manufacturing, installation, construction, commissioning, or operation of the ITER facility shall be appropriately qualified to ensure its quality and integrity prior to approval for use. The qualification shall be undertaken to demonstrate that the software adequately and correctly performs all intended functions and meets all user requirements.

The purpose of this document is to establish a common framework for qualification of software being developed, acquired (procured), operated, maintained, and/or reused for the ITER Project.

2 Scope

This document shall apply to any software which affects public safety, health and sanitation, the protection of nature and of the environment, and the reliability and performance of the ITER Facility.

This document does not apply to software utilized for the management and administration of the ITER Organization, system software for computer operation, or business tools, such as word processing software, communication tools, virus protection, etc. Neither does it apply to physics research software, except that used within the Plant Operation Zone (POZ) or for pulse validation.

3 Definitions and Acronyms

3.1 Definitions

- **acceptance testing:** (A) Testing conducted to establish whether software satisfies its acceptance criteria and to enable the customer to determine whether to accept the system. [2]
- **acquirer:** stakeholder that acquires or procures a product or service from a supplier. The acquirer could be one of the following: buyer, customer, owner, purchaser. [1]
- **acquisition:** process of obtaining software product or software service [1]
- **application software:** software designed to fulfil specific needs of a user; for example, software for navigation, payroll, or process control. Contrast with: support software; system software. [4]
- **computer program:** a combination of computer instructions and data definitions that enables computer hardware to perform computational or control functions. [4]
- **data:** A representation of facts, concepts, or instructions in a suitable manner for communication, interpretation, or processing by humans or by automatic means. [4]
- **developer:** organization that performs development tasks (including requirements analysis, design, testing through acceptance) during a life cycle process [1]
- **embedded software:** software that is a part of a larger system and performs some of the requirements of that system. [4]
- **external software:** software obtained from any source outside, such as nationally recognized institutions and laboratories, and commercial software vendors.
- **firmware:** The combination of a hardware device and computer instructions and data that reside as read-only software on that device. [1] [4]

- **qualification**: process of demonstrating whether an entity is capable of fulfilling specified requirements [1][9]
- **qualification requirement**: set of criteria or conditions that have to be met in order to qualify a software product as complying with its specifications and being ready for use in its target environment or integration with its containing system [1]
- **qualification testing**: testing, conducted by the developer and witnessed by the acquirer (as appropriate), to demonstrate that a software product meets its specifications and is ready for use in its target environment or integration with its containing system [1]
- **software**: computer programs and associated documentation and data pertaining to the operation of a computer system. Contrast with: hardware [2][4]
- **software tool**: A computer program used in the development, testing, analysis, or maintenance of a computer program or its documentation. Examples include comparator, cross-reference generator, decompilers, driver, editor, flowcharter, monitor, test case generator, timing analyser. [4]
- **support software**: software that aids in the development or maintenance of other software; for example, compilers, loaders, and other utilities. Contrast with: application software. Support software includes software tools and system software. [4]
- **system software**: software designed to facilitate the operation and maintenance of a computer system and its associated programs. For example, operating systems, assemblers, system utilities [4]
- **test**: determination of one or more characteristics according to a procedure. [9]
- **test case**: A set of test inputs, execution conditions, and expected results developed for a particular objective, such as to exercise a particular program path or to verify compliance with a specific requirement. [2]
- **testing**: (A) An activity in which a system or component is executed under specified conditions, the results are observed or recorded, and an evaluation is made of some aspect of the system or component. (B) To conduct an activity as in (A). [2]
- **test plan**: A document describing the scope, approach, resources, and schedule of intended test activities. It identifies test items, the features to be tested, the testing tasks, who will do each task, and any risks requiring contingency planning. [2] [4]
- **test procedure**: (A) Detailed instructions for the setup, execution, and evaluation of results for a given test case. (B) A document containing a set of associated instructions as in (A). (C) Documentation that specifies a sequence of actions for the execution of a test. [2] [4]
- **user manual**: A document that presents the information necessary to employ a system or component to obtain desired results. Typically described are system or component capabilities, limitations, options, permitted inputs, expected outputs, possible error messages, and special instructions. [4]
- **validation**: confirmation, through the provision of objective evidence, that the requirements for a specific intended use or application have been fulfilled. Note: Validation in a life cycle context is the set of activities ensuring and gaining confidence that a system is able to accomplish its intended use, goals and objectives. [1][9]
- **verification**: confirmation, through the provision of objective evidence, that specified requirements have been fulfilled. Note: Verification in a life cycle context is a set of activities that compares a product of the life cycle against the required characteristics

for that product. This may include, but is not limited to, specified requirements, design description and the system itself. [1][9]

3.2 Acronyms

COTS	Commercial-Off-the-Shelf
FAT	Factory Acceptance Test
I&C	Instrumentation and Control
IAEA	International Atomic Energy Agency
IEC	International Electrotechnical Commission
IEEE	Institute of Electrical and Electronics Engineers
IO	ITER Organization
ISO	International Organization for Standards
POZ	Plant Operation Zone
QA	Quality Assurance
RO	Responsible Officer
SAT	Site Acceptance Test
WBS	Work Breakdown Structure

4 References

- [1] ISO/IEC 12207-2008 System and software engineering software lifecycle process
- [2] IEEE 829-2008 Software and System Test Documentation
- [3] IEC 61508-3-2010 Functional Safety of Electrical/Electronic/Programmable Electronic Safety-Related Systems – Part 3: Software Requirements
- [4] IEEE 610.12-1990 Glossary of Software Engineering Terminology
- [5] IEEE 730-2002, IEEE Standard for Software Quality Assurance Plans
- [6] IEEE 1012-2012, IEEE Standard for Software Verification and Validation Plans
- [7] IAEA NS-G-1.1 Software for Computer Based Systems Important to Safety in Nuclear Power Plants
- [8] Order dated 7 February 2012 relating to the general technical regulations applicable to BNI - EN (ITER_D_7M2YKF)
- [9] ISO 9000-2008 Quality Management Systems – Fundamentals and Vocabulary
- [10] ITER Quality Assurance Program (ITER_D_22K4QX)
- [11] Quality Classification Determination (ITER_D_24VQES)

5 Basic Principle

Software shall be qualified based on the purpose of the software in accordance with applicable standards, procedures and/or instruction prior to approval for use. According to Quality Classification Determination (ITER_D_24VQES), software used for Quality Class 3 Non Safety Related (NSR) items or services do not have to follow the qualification process laid out in this document.

The software qualification process shall be depend upon the characteristics, importance, and complexity of the software, and be applied to demonstrate that the software adequately and correctly performs all intended functions and meets all user requirements. The process shall also demonstrate, as appropriate, that the software

- 1) properly handles abnormal conditions and events as well as credible failures;

- 2) does not perform adverse unintended functions;
- 3) does not degrade the system either by itself, or in combination with other functions or configuration items.

Where changes to previously qualified software are made, re-qualification shall be required for the change, including evaluation of the effects of these changes. In the case the verification of installations is required, such verification shall also be performed whenever the software is installed on a different computer, or when significant hardware or operating system configuration changes are made, the necessary validation shall be performed.

The qualification records shall be documented and maintained as quality assurance records.

6 Software Qualification Methodology

6.1 Embedded Software

Software within an embedded structure, system, or component (SSC) of the ITER facility shall be qualified in accordance with the applicable SSC requirements before use. Where applicable, the qualification tests for the software shall be part of the hardware integration tests, factory acceptance tests (FAT), or site acceptance tests (SAT), and shall be documented in the test results' documentation.

In particular, I&C software embedded in systems that perform protection and/or safety functions shall be subject to the software qualification process defined in the ITER I&C standards. If not provided, software qualification shall be undertaken as described in Section 7 of this document.

6.2 Software used for Design or Technical Justification

Software used for design and technical justification to ensure an existing, modified, or proposed structure, system, and component (SSC) design meet technical requirements shall be validated before use.

The verification of installations of validated software used for design or technical justification is not required unless it is mandated by regulatory requirement, the developer or publisher of the software.

In cases where software is developed or modified, the validation perimeter of this type of software shall be compliant with the intended use of the software, and the validation record must define clearly the validation limit.

In cases where commercial software (e.g. ANSYS) is utilized without modification, the software acquirer or user shall ensure that the software is qualified for the intended purpose.

6.3 External Software

Where procured software is developed or modified by external sources, procurement documents shall provide appropriate software qualification requirements so that users can validate the procured software prior to use.

In cases of COTS, freeware, or shareware software acquired or procured without any modification, validation shall be performed to permit confirmation of acceptable performance of the software in the operating system. If a user manual or a test procedure for validation does not exist, software qualification shall be undertaken as described in Section 7 of this document.

7 Software Qualification Process

The WBS Responsible Officer (RO), who is responsible for the development, acquisition, or use of software, is responsible for the selection of standards and establishing qualification requirements based upon the applicable technical requirements. In cases where the applicable standards are not specified, the following process shall be applied for the qualification.

7.1 Planning

7.1.1 Assignment

The WBS RO shall be the software manager unless otherwise designated. The software manager shall assign a competent person(s) for software qualification. The person(s) shall be other than those that performed the original design and development of the software, and should be given appropriate training in the use of test tools, procedures and techniques.

7.1.2 Qualification Plans / Procedures

The person(s) shall prepare software qualification plans or technical procedures based upon the software requirements. The plans or procedures shall include test cases encompassing the range of intended use for the new or revised software. Qualification testing should be taken into consideration to demonstrate that software meets its specifications and is ready for use in its target environment or integration with its containing system.

Where necessary to evaluate technical adequacy for verification, the plan should indicate how the results are to be evaluated. For example, the results may be compared to results from alternative methods such as:

- 1) analysis without computer assistance;
- 2) other qualified software;
- 3) experiments and tests;
- 4) standard problems with known solutions; or 5) confirmed publications or correlations.

7.2 Execution

According to the established plans and/or procedures, the person(s) carrying out software qualification shall do so under the direction of the software manager. The tools or equipment used to conduct the qualification should be uniquely identified in the plans or procedures to ensure that qualification is repeatable and to assist with the investigation of anomalies.

7.3 Records

Software qualification results shall be documented by the person(s) carrying out the qualification, reviewed by other competent person(s) assigned by the software manager, and approved by the software manager (WBS RO). The documentation shall identify the following:

- 1) Software name, number, and version;
- 2) Computer hardware, software tools (including version), and equipment used, if any;
- 3) Person(s) who conducts qualification;
- 4) Simulation models used, where applicable;
- 5) Results and acceptability including range of applicability of the software;