

FMI Development Process And Communication Policy

Revisions:

04.07.15	Version 1.0 accepted by FMI Steering Committee
03.05.15	Fixed formatting issues (used 0.90 document and copied 0.91 changes to it; included comment in [...] in italics with smaller font) + fixed issues with formatting of state machine (MO)
02.18.15	Added Martin Otter's note on backward and forward compatibility (BL)
02.10.15	Added results of discussion at FMI meeting 9./10.2.2015 v0.90 (AJ)
11.12.14	Split the document into pure rules and notes v0.10 (AJ)
11.06.14	Worked Martin Otter's proposals into the document v0.9 (AJ)
10.10.14	Added comments by two telcos with a small working group (Torsten Blochwitz, Mongi Ben Gaid, Monika Mühlbauer, Christian Bertsch, Antoine Viel, Bruno Loyer) (AJ)
09.12.14	Added comments by Mongi Ben Gaid and Monika Muehlbauer (AJ)
08.20.14	V0.8 of the FMI Development Process (Andreas Junghanns)

Contents

1. Guiding Principles of the FMI Standard Development.....	2
2. The Development Phases.....	2
2.1 Developing FMI Change Proposals (FCP).....	2
2.2 Defining a new FMI Release.....	4
2.3 Building the specification document for the new FMI release.....	4
2.4 Alpha tests.....	5
2.5 Beta tests.....	5
2.6 Release phase.....	5
2.7 Maintenance phase.....	6
3. Communication.....	6
4. Glossary.....	6

1. Guiding Principles of the FMI Standard Development

The overall goal of the development process is guided by the following fundamental tenets that all aim to facilitate widespread adoption of the standard and bridging different simulation worlds:

1. *Compatibility*: The FMI Standard shall be internally consistent and complete. The Standard documents shall be an enabler for easy exchange of simulation models.
2. *Simplicity*: The FMI Standard shall stay as simple as possible.
3. *Stability*: Backwards compatibility of future versions of the FMI Standard shall be maintained whenever possible.
4. *Neutrality*: FMI shall be neutral with respect to tools, technologies (e.g. processors, compilers, OS, file system access, solvers, real-time capability, ...) and languages.
5. *Quality*: The FMI Standard shall improve the quality and reliability of model exchange and co-simulation.
6. *Transparency*: End users and tool vendors shall be informed about the progress of standardization as early as possible.

A companion document contains design rationals and further information and arguments about the development process defined here.

2. The Development Phases

The following subsections define a number of development phases by declaring what is happening and what roles are involved, the expected decisions and results.

The development process defined here can be --- *in exceptional cases* --- overruled by the FMI Steering Committee with a qualified majority vote.

2.1 Developing FMI Change Proposals (FCP)

An FMI Change Proposal [FCP] is a collection of documents evolving through a number of stages of development and approval. An FCP has a unique number it maintains throughout its lifetime for easy reference. Anyone can work on an FCP, provided they signed the FMI CLA. An FCP has any of the following statuses:

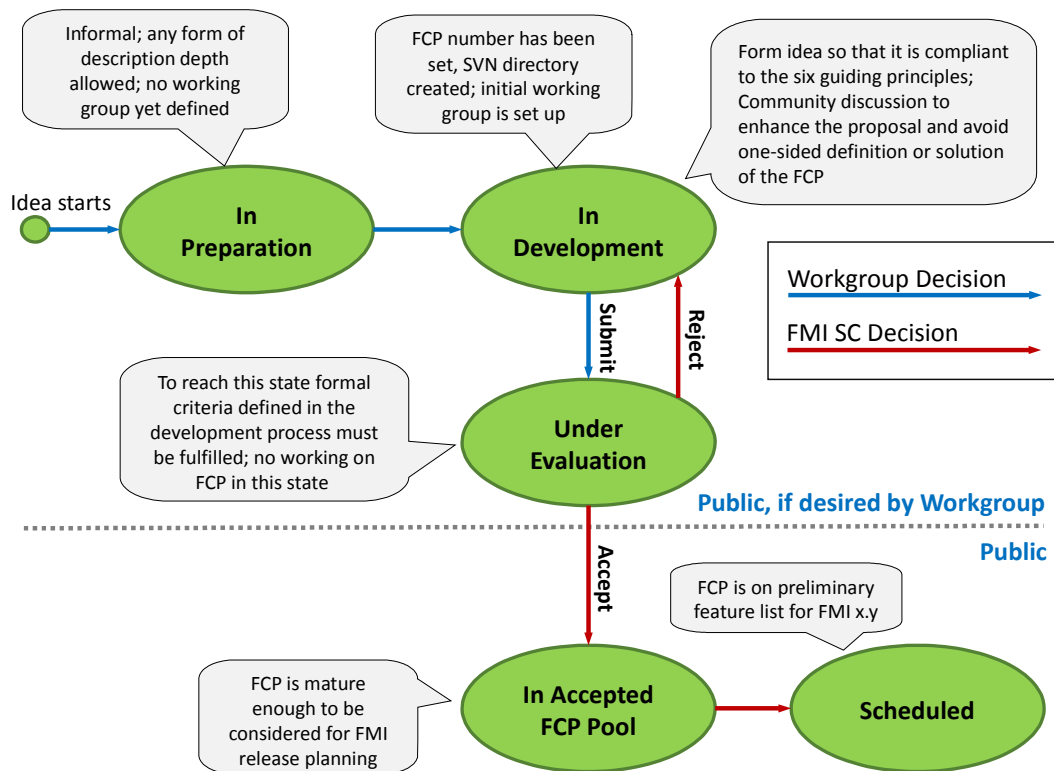
1. **In Preparation:**
In this stage, an initial idea is described in an informal way to interest potential collaborators (tool vendors and end-users) and to identify potential technical issues and their solutions.
2. **In Development:**
A number of collaborators (initial working group) work on creating a complete FMI Change Proposal ready for submission. A unique number is generated and an SVN repository directory to contain all relevant documents is created. A new ticket under <https://trac.fmi-standard.org/> is created for discussion. The (possibly changing) working group will adapt the material according to the outcome of these discussions. This phase is left when the working group deems the FCP to be complete and the phase **Under Evaluation** is entered by sending an email to the FMI group.
3. **Under Evaluation:**
The FCP might be rejected on formal grounds if the FCP does not fulfill the minimal standard (outlined below). The Steering Committee will weight the benefits (new and improved

capabilities) given the provided business cases against the costs (increase in complexity, breaking/keeping backward compatibility,...) to either accept or reject the FCP according to the **6 Guiding Principles**. The vote on an FCP can be held no earlier than 1 month after submission. The decision will be published in the corresponding ticket. This simple majority vote has two outcomes:

1. Accept the FCP: The FCP will be labeled **In Accepted FCP Pool** and can then be considered for the next phase of **Defining a New FMI Release**.
2. Reject the FCP: The FCP will be send back to the authors with comments and the status will be changed to **In Development** again.

4. In Accepted FCP Pool:

Accepted FCPs form a pool of potential features to be included in one of the following releases. There is no guarantee for an accepted FCP to be included into any future release.



State machine describing FCP statuses and transitions

5. Scheduled:

Once an FCP was chosen to be implemented in a specific FMI release (see section **Defining a New FMI Release**), it will be marked as **Scheduled**. This is only relevant for documentation and sorting purposes.

An FCP is a document together with additional material that is based on the provided template, where all fields of this template must be filled. In particular, the following information must be given:

- Business case: why should this feature be included? What problems can be solved (better) that cannot be solved (as easily) now?
- The precise text that shall be included in the latest standard document reduced to the relevant parts (but keeping the chapter headings to not disturb the numbering).

- An example implementation, for both an export and an import tool, simple test cases to profoundly illustrate the solution of the problem, and at least one description of an industry-size example to prove scalability of the solution. FMUs complying to a pre-release version of the standard have to be marked with the semantic version number (see <http://semver.org/spec/v2.0.0.html>, e.g. 2.1.0-alpha.2).

The visibility of FCPs is decided by the FCP authors, but each publication must be clearly marked with something like: "This proposal is waiting for approval of the FMI Steering Committee and might not be scheduled for any future FMI release."

Result: A backlog of FMI Change Proposals that wait for either further improvements or the definition of a new FMI release.

2.2 Defining a New FMI Release

The FMI Steering Committee can decide at any time on a new FMI release and selects which FCPs to include in that release. This implies a positive vote with qualified majority on the following proposal:

- Version number: MajorVersion.MinorVersion.MaintenanceVersion
 - Changes introduced by Major versions don't have to be backward nor forward compatible.
 - Minor versions within the same Major version have to be backward compatible.
 - Maintenance versions must not introduce new features and have to be forward and backward compatible.
 - All versions will have semantic version numbers.

[Note that backward and forward compatibility is with respect to the FMI specification, not with respect to a tool. For example, if FMI 2.1 is backward compatible to FMI 2.0, then every 2.0 FMU must be also a valid 2.1 FMU. As a consequence, a minor version implies restrictions with respect to the previous minor version, especially, but not limited to: (a) new XML elements and attributes, as well as new C API functions must be optional, (b) the argument lists of existing C API functions cannot change (only the meaning of existing arguments can be enhanced, if this is signaled in the modelDescription.xml file with new, optional, XML elements or attributes).]

- Preliminary feature list:
A **public** list of FCPs (links to the FCPs on the svn and to the corresponding tickets) clearly stating its preliminary nature.
- Preliminary release schedule:
A preliminary, **internal** release schedule for the next release. Whenever necessary, the Steering Committee can adapt that internal schedule. The Steering Committee decides if and when to publicly announce release dates and **preliminary** feature lists. Without a decision of the Steering Committee the status is "under development, no release date available".

Result: An FMI Steering Committee approved **preliminary feature list** to be implemented in the proposed FMI release. This **preliminary** feature list is **public** outside FMI Steering Committee and FMI Advisory Committee.

The FMI Steering Committee announces publicly its plans for new versions of the standards together with a preliminary feature list. The preliminary release schedule is **not public**.

All of the next (building the specification document, alpha, beta and release) phase changes will be announced publicly by the Steering Committee with an expected duration of the phase entered.

2.3 Building the Specification Document for the New FMI Release

The FCPs from the preliminary feature list are merged into the last released version of FMI to arrive at a single, consistent document. The FCP authors are responsible for this development phase and are required to check if the merge kept their feature proposals intact.

This phase does not require implementations.

Result: After all FCP authors approved the merged standard document, the FMI Steering Committee declares this document version as **Alpha.1**. Alpha versions of the standard document are **not public** outside the FMI Steering Committee and the FMI Advisory Committee.

2.4 Alpha Tests

During this phase, tool vendors start implementing the specification and file tickets with problems they encounter, such as inconsistencies, clarifications needed etc. Further Alpha versions result from these tickets.

The FMI Steering Committee must approve each new Alpha version with simple majority. The number of Alpha versions should be kept to a minimum to avoid unnecessary implementation effort for the participating tool vendors.

Changes to Alpha versions of the standard document don't have to be backward compatible with previous alpha versions. In fact, it might even be possible to change the preliminary feature list. In contrast: Beta versions should only change in a backward compatible way.

All major changes to the document will be tracked at the beginning in a change history with references to the ticket system, whenever appropriate.

The cross check infrastructure will be (non-publicly) available to facilitate exchange of FMUs and reporting of the current state of implementations. FCP authors are required to supply and maintain the test cases (FMUs and cross checks) for their FCP.

Result: Once sufficient proof-by-implementation is given – as judged by the cross check results – the FMI Steering Committee may vote with simple majority on promoting the latest Alpha version to a **publicly available Beta.1**. As a byproduct, the preliminary feature list will turn into the **publicly available, final feature list** for this version of the standard.

2.5 Beta Tests

With a fixed feature list, it is expected that all tool vendors will start implementing this version of the standard. Tickets will be resolved in mostly backward compatible ways and lead to more (but few) Beta versions – exceptions to backward compatibility must be well supported.

The Cross-Check infrastructure will be used to track progress in implementations and interoperability.

Plug fests are encouraged, but not mandatory. These plug fests help vendors to speed up their development efforts and improve vendor-to-vendor communication for issue resolution. The participants of a plug fest will contribute using the trac-system and progress will be documented in the Cross-Check infrastructure. Finally the participants may issue a recommendation to the FMI Steering Committee whether and when to close Beta testing.

Result: The FMI Steering Committee will decide which Beta version will be promoted to a **publicly available release candidate (RC.1)** or, when absolutely necessary, to revert back to an earlier development phase. This decision has to be approved by a qualified majority vote. The FMI Steering

Committee will especially take into account whether sufficient test implementations have been proving this version of the standard.

2.6 Release Phase

The release phase can be seen as a final tidying of the standard document, the implementation notes, examples and Cross-Check infrastructure before the standard is released. Only clarifications, spelling and formatting corrections are allowed in this phase to produce further **RCs**.

Result: The FMI Steering Committee will set a waiting period for feedback after each new RC. The FMI Steering Committee will decide when to stop waiting for more feedback and promote the latest RC to the the **Final Release** of the FMI standard documents with qualified majority.

Final release of the standard will lead to publishing the standard on the FMI web pages for public download.

2.7 Maintenance phase

After the final release of the standard documents, further clarifications and cleanup might be needed. Such backward compatible changes may result in **maintenance versions** being released.

The ticket system will have a milestone for such maintenance revisions. Issues are collected there that (may) need fixing.

The steering committee can decide

- if an apparent bug fix is effectively a feature request and needs to be dealt with as such with due process,
- if the tickets will be delayed to a later version.

Result: Once all tickets for such a maintenance version are fixed, a release phase is started by the FMI Steering Committee (see point above) - all the way to finally releasing this maintenance version.

3. Communication

Public decisions of the FMI Steering Committee are announced by the FMI project leader on the FMI web page. Statements of opinion and (time/release) estimates by members of the FMI community should be avoided. If done, they must be denoted as personal opinions.

4. Glossary

Term	Definition
simple majority	Proposal must receive more than 50 per cent of the valid votes submitted
qualified majority	Proposal must receive more than 66 per cent of the valid votes submitted