FpML Change Guidelines

1. *Status of this document*

This second version of the change guidelines was approved by the FpML standards committee in June 2007. Only the FpML Standards Committee may modify or eliminate the guidelines.

2. *Introduction*

This document summarizes guidelines about the types of changes to FpML that may be introduced in different types of FpML document release processes.

- First the document describes the allowed types of FpML document release processes.
- Second the document lists the permissible effects of changes in each FpML document release process, and the types of changes that may generate these effects. It also defines which stages of the release process may introduce each type of change.
- Third, this document summarizes how these changes are identified.
- Fourth, this document describes an exception process by which normally impermissible changes can be approved.

3. *FpML document release processes*

FpML documents may be released using the following processes:

- Full major version.
- Minor version
- Fast track minor version
- Errata release

i) Major Version

A major FpML version is a complete re-release of the standard. It may incorporate any permissible change to FpML document. (Also see below under “impermissible changes.”) FpML major versions always go through the full FpML release process, including working drafts (WD), last call working draft (LCWD), trial recommendations (TR), and Recommendations (Rec), unless development is stopped prior to the specification’s reaching Recommendation status.

ii) Minor Version

A minor FpML version is a version that is intended to be fully backward compatible on the instance level with the current major version. It may add significant new functionality, such as new asset classes. FpML minor versions typically go through the same release process as major versions (see above).

iii) Fast Track Minor Version

A “fast track” minor version is a minor version, so it is intended to be fully backward compatible on the instance level with the current major version. It adds new functionality that is relatively similar to existing functionality and so does not require as thorough a review process as a major version. An example would be a new feature or enhancement to an existing asset class.

The review process for a fast track minor version is LCWD -> TR -> Rec. In other words, there will be no need for a Working Draft (WD), but instead the publication would go straight to Last Call WD. The LCWD -> TR stage for a fast track release will typically be approximately 1 month. The TR ->Rec stage for a fast track release will be similar to that for a regular release.

iv) Errata Release

An errata release is a very small change to a Recommendation that corrects minor errors or adds trivial enhancements that in the judgment of the FpML Standards Committee are small enough not to require a significant review process. An example would be an addition of a new value to an existing scheme or enumeration.

Errata releases consolidate errata items previously published under “errata”. Other than the requirement that the errata must have previously been published, there are no other steps in an errata release.

4. *Permissible Changes*

i) Effect of Change

Changes to the FpML specification can have different impact on FpML users, depending on document release process used. The following table summarizes the permissible impacts of changes, by document release process.

Effect of change	**Example**	**Allowed in Major version?**	**Allowed in full track minor version?**	**Allowed in fast track minor version?**	**Allowed in Errata version?**
Invalidates existing instance documents	Change “swap” tag to “irSwap”	√			
Invalidates user-derived types	Change name of “AdjustableDate” type	√	√		
Adds major new functionality	Add new product or asset class.	√	√		
Adds minor new functionality requiring several changes	Add new feature to existing product, requiring several additions to the schema.	√	√	√	
Requires new validation code	Addition of new optional element	√	√	√	√

ii) Impermissible Changes

There are certain aspects of FpML that we do not expect to ever change. These changes are called “impermissible changes,” and under these change guidelines they may not be made.

The following types of changes are impermissible:

- Elimination of an asset class or product.
- Elimination or name change of document-level elements (trade, party, header, portfolio).

iii) Type of Change

Following are a set of guidelines about what types of changes to the schema are permitted for each release process. These are intended to provide a more detailed set of guidelines than the general policies listed above. In addition, it for each type of change, it lists the latest stage of the release process that may introduce the change. For example, if the Latest Stage is “LCWD”, the change may be introduced in a Last Call Working Draft, but not in a Trial Recommendation.

Type of change	**Example**	**Allowed in Major version ?**	**Allowed in full track minor version?**	**Allowed in fast track minor version?**	**Allowed in Errata version?**	**Latest Release Stage for Introduction**
Major change to mechanism of schema definition	DTD -> schema; multiple namespaces	√				WD
Change to FpML root element structure	Eliminate scheme defaults	√				WD
Add mandatory elements to existing element	Add mandatory “tradeStatus” to tradeHeader	√				LCWD
Remove optional elements	Remove stubs from swapStream	√				LCWD
Make optional element mandatory	Make partyName mandatory in party.	√				LCWD
Change to existing elements (location/name/etc)	move calculationAgent from tradeHeader	√				LCWD
Changes/deletions to scheme/ enumeration	Remove “MODFOLLOWING”	√				TR
Add mandatory attributes to existing elements	Add mandatory ID attribute to tradeHeader	√				LCWD
Change modeling convention	Referencing xlink -> id/idref, scheme -> enum	√				WD

New asset class	Equities, Energies	√	√			WD
New product (member of substitution group)	swap, creditDefaultSwap	√	√			WD
New product under existing product element	credit default index, under creditDefaultSwap	√	√	√		TR
New product feature/variant	tranches in credit default index	√	√	√		TR
New top-level structure	e.g. trade, position, contract	√	√			WD
Make mandatory element optional	<openUnits> becomes optional	√	√			Rec
Change to names of types/groups	Valuation -> EquityValuation	√	√			TR
New business process/ messaging area	RFQ message area	√	√	√		WD
New message in an existing business process	CancelNovationConfirmation in novation confirmation area	√	√	√		TR
Add optional elements	add brokerPartyReference	√	√	√	√	Rec
Additions to scheme/enumeration list	Add "VECA" (Caracas) to BC list	√	√	√	√	Rec
Add optional attributes to existing elements	Add "id" attribute	√	√	√	√	Rec
Major change in schema organization	Add new subschema (.xsd file)	√	√			TR
Minor change in schema organization	Move a type definition from one file to another	√	√	√		Rec

Note that changes to schema organization have no impact on instance documents nor on derived types, but since they have an impact on users of the specification they are to be avoided unless required to resolve other issues, e.g. dependency errors, or required to address new areas of functionality (e.g. to generalize the use of a number of definitions into a new area.)

5. Identification by release process

Going forward, the following identification scheme will be used for identifying major, minor, and errata versions.

Release Process	**Version attribute**	**Namespace.**
Major version	4-0	http://www.fpml.org/2003/FpML-4-0
Minor version	4-1	http://www.fpml.org/2004/FpML-4-1
Fast Track Minor version	4-1	http://www.fpml.org/2004/FpML-4-1

Please note the following:

- There is no difference in identification between full track and fast track minor versions.
- Errata releases do not change the namespace, only the version attribute.
- In the past, a different convention was used for identifying errata versions.

6. Exception Process

There may be times, particularly when an area of the standard is relatively immature or there is a significant change to business practice, where it becomes desirable to make a change that would normally be impermissible. The process required to make this change is as follows:

- The working group requesting the change must agree unanimously that the change is required, by formal vote of active members.
- The FpML Standards Committee must formally review the requested change. It has three options:
    i. Reject the change.
    ii. Approve the change.
    iii. Request public comment on the change. In this case, the Standards Committee will determine the comment period; at the end of the comment period, the Standards Committee must either approve or reject the change.

  The usual Standards Committee voting rules will apply for approval of the exception.