DIS TABLE OF REPLIES / 2000-10-19 / TABLEAU DES REPONSES DIS

JTC 1 — VOTING BEGAN ON/DEBUT DU VOTE:2000-04-13
ISO/IEC DIS 19501-1 — TIME LIMIT FOR REPLY/DELAI:2000-10-13

TITLE: Information technology -- Unified Modeling Language (UML) -- Part 1: Specification

TITRE: Technologies de l'information -- Langage de modélisation unifié (UML) -- Partie 1: Spécification

MEMBER BODY/COMITE MEMBRE		APPROVAL/APPROBATION	DISAPPROVAL/DESAPPROBATION	ABSTENTION		MEMBER BODY/COMITE MEMBRE		APPROVAL/APPROBATION	DISAPPROVAL/DESAPPROBATION	ABSTENTION	
Australia (SAI)	P		X			Korea, Republic of (KATS)	P	X			
Belgium (IBN)	P	X				Netherlands (NEN)	P				
Brazil (ABNT)	P	X				New Zealand (SNZ)	P				
Canada (SCC)	P	X				Norway (NSF)	P	X			
China (CSBTS)	P	X				Portugal (IPQ)	P			X	**
Czech Republic (CSNI)	O	X				Romania (ASRO)	P				
Denmark (DS)	P					Russian Federation (GOST R)	O	X			
Egypt (EOS)	P	X				Slovenia (SMIS)	P	X			
Finland (SFS)	P	X				South Africa (SABS)	P	X			
France (AFNOR)	P	X			*	Sweden (SIS)	P	X			
Germany (DIN)	P	X			*	Switzerland (SNV)	P		X		
Ireland (NSAI)	P	X				Ukraine (DSTU)	O	X			
Israel (SII)	O	X				United Kingdom (BSI)	P	X			*
Italy (UNI)	P	X				USA (ANSI)	S	X			*
Japan (JISC)	P		X								
						T O T A L		21	1	3	

* = Comments / commentaires
** = P-member having abstained and therefore not counted in the vote / Membre (P) s'abstenant de voter; n'est donc pas compté dans le vote

P-MEMBERS VOTING: / MEMBRES (P) VOTANT:		IN FAVOUR OUT OF / EN FAVEUR SUR		REQUIREMENT / CRITERE
	17	19 =	89.47%	>= 66,66%

MEMBER BODIES VOTING: / COMITES MEMBRES VOTANT:		NEGATIVE VOTES OUT OF / VOTES NEGATIFS SUR		REQUIREMENT / CRITERE
	3	24 =	12.50%	<= 25%

THIS DRAFT IS THEREFORE UNDER BALLOT
in accordance with the ISO/IEC Directives, Part 1, sub-clause 2.6.3.

CE PROJET EST DONC EN COURS DE VOTE
selon les Directives ISO/CEI, Partie 1, paragraphe 2.6.3

JTC1/SC7 ID 32620

Standards Australia

Title: Australian Comments on ISO/IEC DIS 19501-1

Date 2000-10-04

Status: National Body Comment

Australia wishes to vote against this proposal for establishing the current version (1.3) of the Unified Modeling Language (UML) as an ISO standard.
The primary reason for not accepting it as an international standard is that we believe it is premature to do so. OMG have submitted UML Version 1.3 as a PAS. However, at the same time, the OMG intends to put out a large number of versions of UML in the near future (1.4 is planned for this September, 1.5 later and 2.0 and 3.0 are being planned). Indeed, the RFP for 2.0 is in draft form and is likely to be issued in the next few months. UML is thus not stable and it is therefore premature for ISO to adopt the transient Version 1.3.

In addition to this proliferation of versions, there are a significant number of technical concerns. These include:

1. Lack of a semantically consistent and meaningful metamodel and semantics for the whole-part relationship ("aggregation"). This is seen in the contradictions in the definitions of shared and composite aggegration. It has also been noted that the rules for these in the current UML can in fact be satisfied by some cases of association which are NOT whole-part relationships. Others have written at length about the problems with UML's shared and composite aggregation, the most succinct probably being a paper presented at the <<UML>>'99 conference and published by Springer in their LNCS series. We also note that this topic is one of the ones listed on the V2.0 RFI indicating that it has been accepted by the OMG RTF as being serious.

2. With an increasing move to responsibility-driven design, the support in UML for responsibilities is inadequate. There is a need for a serious metamodel for this and the OMG RTF chair has invited an international group to make a submission to the forthcoming RFP to address this issue.

3. Lack of semantics. There is a group based largely in Europe called pUML (Precise UML) which has many noted members in the formal methods/formal languages community. They argue that the Semantics part of the UML document is actually syntax and that there really isn't a viable semantics component to the UML. There are many papers on this topics in recent conferences such as <<UML>>'98, <<UML>>'99, many of the TOOLS conferences etc.

4. Misuse of the UML concepts in the metamodel itself. For instance, generalization is reasonably well-defined in the UML (but the stereotype of <<implementation>> isn't). However, the use of generalization arrows in the metamodel definition (the definition of the UML itself) is in many places incorrect since it is used to represent implementation inheritance i.e. cases where the Liskov Substitution Principle fails. Black diamond aggregation (known as composition) is also used in many places in the metamodel where not only is the "strong" form of aggregation doubtful (remembering that

this definition is ambiguous and self-contradictory in the first place) but there is even doubt that some of the relationships thus labelled (with black diamond) are even whole-part of any kind (see point 1).

5. Stereotypes are contentious. Their presence at all has been questioned by not only a research group in Germany but also by members of the OMG RTF.
They are meant to be user-defined subtypes at the metamodel level but many are pre-defined not user defined and again some of these are not subtypes (e.g. Type as a stereotype of Class) for the simple reason that they don't obey the Liskov Substitution Principle for subtyping. There is also a semantic argument that, in this example, Type is NOT a special kind of Class and that they should be peers rather than being linked, as currently, in an "inheritance" hierarchy.

6. The metamodel for Type/Class/Interface is acknowledged as needing improvement. There was an attempt to create a better model between Version 1.1 and 1.3; but the 1.3 description turned out to be identical to the 1.1 description. The main problem, as noted above (point 5), is that the Liskov Substitution Principle is violated in this portion of the current 1.3 metamodel. While further suggestions have been made, none have been accepted as yet.

The consequent, widely acknowledged poor support for components is particularly important in the context of the current trend towards component-based development becoming mainstream.

7. The difference between (and need for) Association and Dependency as two separate and distinct relationships is unclear to many. Although the Relationship hierarchy was much improved in a very late draft of 1.3, there is still much that could be done.

8. It has a low quality from a semiotic and learnability viewpoint and new users are often outfaced by the sheer size of the current OMG standard (over 800 pages).

However, although voting NO now, we look forward to a subsequent PAS submission of a more stable UML version in the future.

ID 32620 ISO/IEC DIS 19501-1 JTC1/SC7 FRANCE

COMMENTS ON 19501-1
« Information technology - Unified Modeling Language (UML)
Part 1: Specification

First comment:

DESCRIPTION:
The balloted document contains a clause Restricted Rights legend that mentions that the proposed specifications are owned by Rational Software Corp. and OMG (page 5).
This is not acceptable for an International Standard.

PROPOSED SOLUTION:
Remove the clause.

Second comment:

DESCRIPTION:
The balloted document does not contain any specification of UML. In order to get them, the Preface clause provides only an address for subscription.
In addition, in this clause, the sentence "Specifications are adopted as standards only when representatives of the OMG membership accept them" (page 33) seems to mean that the opinion of ISO/IEC JTC1 experts about the UML has no importance.
This is not acceptable for an ISO standard.

PROPOSED SOLUTION:
Include the UML specifications in the document ISO/IEC 19501 and remove texts about OMG pre-eminence on UML.

DIN

General Comment on Draft international Standard ISO/IEC 19501-1

DIN supports ISO/IEC/DIS 19501-1 with the following Comment:

Introduction:

OMG and ISO/TC 184/SC 4 are in a close Liaison on Modelling Languages.

Therefor ISO/TC 184SC 4 has decided during the Melbourne meeting on the following resolution:

"RESOLUTION 445 (Melbourne, Australia-February, 2000) UML
SC4 encourages OMG to accept interoperability between UML and EXPRESS (ISO 10303-11) as a mandatory requirement in the UML 2.0 development effort.

SC4 directs its WGs 10 and 11 to work closely with OMG, especially the UML 2.0 WG in OMG's Analysis and Design PTF, in support of this goal.

SC4 asks its member bodies and liaison organisations to support these efforts adequately in OMG through their member companies and organisations."

To reach this goal it is necessary to have a close co-operation between these groups in the development of of the Modelling Languages EXPRESS (ISO 10303-11) and UML version 2.0. This needs to be pointed out even the German vote is yes.

DIN Germany

3

JTC1/SC7 ID 32620

The National Body of Japan disapproves ISO/IEC DIS 19501-1 and submits the comments below. If these comments are satisfactorily resolved, Japan will change its vote to approval. Comment JPN-001TH, in particular, should be thoroughly resolved before this DIS is processed further.

The following comments are numberd with the rule:
Format JPN-???[TH|TL|G|E],
JPN means the comment is from Japan,
??? is a serial number with 3 digits,
TH means high level technical comment,
TL means low level technical comment,
G means general comment, and
E means editorial comment.

[JPN-001TH] p. 7-52. 7.9: Definitions of "typeName" and "name"
In order to allow the use of multi-byte characters at "typeName" and "name", the definitions of "typeName" and "name" should be revised. The details of the proposed revision are given below.

In order to allow the free use of English words or sentences, the EBNF syntax should be extended. The sentence at p. 7-50, line 4 should be revised as shown below:

> The grammar description uses the EBNF syntax, where "|" means a choice, "?" optionally, "*" means zero or more times, "+" means one or more times, and words enclosed with "/*" and "*/" are definitions described in English words or sentences.

To modify the definitions of "typeName" and "name", the sentence at p. 7-50, line 5 should be revised as shown below:

> In the description of the string, the syntax for lexical tokens from the JavaCC parser generator is used. (See http://www.suntest.com/JavaCC.)

In order to allow the use of multi-byte characters in "typeName" and "name", the definitions of "typeName" and "name" should be revised as shown below:

```
typeName       := charForNameTop charForName*
name           := charForNameTop charForName*
charForNameTop := /* Characters except
                     inhibitedChar and
                    ["0"-"9"]; the available
                     characters
                    shall be determined by the tool
                    implementers ultimately. */
charForName    := /* Characters except
                     inhibitedChar; the
                    available characters shall be
                     determined by the tool
                     implementers ultimately. */
inhibitedChar  := " " | "¥"" | "#" | "¥" | "(" | ")" |
                  "*" | "+" | "," | "-" | "." | "/" |
                  ":" | ";" | "<" | "=" | ">" | "@" |
                  "[" | "¥¥" | "]" | "(" | "|" | ")"
```

[JPN-002TL]
Style guidelines that specify to use uppercase letters, lowercase letters or capitalization should mention that it is the case in English. Because some natural languages like Japanese don't have neither uppercase letters nor lowercase letters.

"uppercase letter" appears in 3.22.4. "lowercase letter" appears in 2.3.5, 3.22.4, 3.25.4, 3.26.4, 7.5.4.1 and 7.5.5. "capitalization" appears in 3.53.4 and 3.54.3.

[JPN-003TL]
The phrase, "The OMG XMI standard", appears in the section 6.1, but there in no reference. Specify the document that defines the OMG XMI standard in the clause "Normative reference".

[JPN-004G]
Delete copyright statements before Preface.

[JPN-005G]
Delete every contents table at the beginning of every chapter.

[JPN-006G]
Follow ISO directives (ISO standard document style guide. you can get it from
http://isotc.iso.ch/livelink/livelink/fetch/2000/2123/SDS_WEB/sds_edit.htm ("ISO/IEC Directives, Part 3").)

Japan proposes the following table of contents:

1 Scope
See [JPN-007G].
2 Normative references
See [JPN-008G].
3 Terms and definitions
See [JPN-009G].
4 Abbreviated terms
See [JPN-010G].
5 UML semantics
6 UML notation guide
7 UML CORBAfacility interface definition
8 UML XMI DTD specification
9 Object constraint language specification
Annex A (normative) UML summary
First chapter of original document, except "1.5 Scope of the UML".
Annex B (normative) UML standard elements
Annex C (normative) IDL modules
"5.4 IDL Modules" of original document.
Annex D (normative) UML XMI DTD
"6.3 UML XMI DTD" of original document.
Annex E (informative) About OMG
First section and 6th section (Acknowledgements) of Preface. pp. i-ii and pp. xv-xvi.
Annex F (informative) Compliance to the UML
5th and 6th section of Preface. pp. xi-xv.
Annex G (informative) Introduction to OMG modeling
2nd sections of Preface. pp. ii-iii.
Annex H (informative) Architectural alignment of UML, MOF and CORBA
3rd section of Preface. pp. iii-ix.
Annex I (informative) References
Last section of Preface. pp. xvii-xviii.
Annex J (informative) Index

The mapping from original document to Japan's proposal is followings:

UML 1.3 document	ISO standard
Preface About OMG	Annex E About OMG
Preface Introduction to OMG Modeling	Annex G Introduction to OMG modeling

Preface Architectural Alignment of UML, MOF, and CORBA	Annex H Architectural alignment of UML, MOF and CORBA
Preface Document Summary	deleted
Preface Compliance to the UML	Annex F Compliance to the UML
Preface Acknowledgments	Annex E About OMG (the rest of "About OMG")
Preface References	Annex I References
chapter 1 UML summary 1.1 - 1.4	Annex A UML summary
chapter 1 UML summary 1.5 Scope of the UML 1.5.1 outside the scope of UML	1 Scope
chapter 1 UML summary 1.5.2 comparing UML to other modeling languages 1.5.3 Features of the UML 1.6 - 1.7 UML summary	Annex A UML summary
chapter 2 UML semantics	5 UML semantics
chapter 3 UML notation guide	6 UML notation guide
chapter 4 UML Extensions	7 UML Extensions
chapter 5 UML CORBAfacility interface definition 5.1 - 5.3	8 UML CORBAfacility interface definition
5.4 IDL Modules	Annex C IDL modules
chapter 6 UML XMI DTD specification 6.1 - 6.2	9 UML XMI DTD specification
6.3 UML XMI DTD	Annex D UML XMI DTD
chapter 7 Object Constraint Language Specification	10 Object Constraint Language Specification
Glossary	3 Terms and definitions
appendix A - UML standard elements	Annex B UML standard elements
Index	Annex J Index

[JPN-007G]
Change "1.5 Scope of the UML" to conform with Directives as follows:

1 Scope

The Unified Modeling Language (UML) is a language for specifying, constructing, visualizing, and documenting the

artifacts of a software-intensive system.

First and foremost, the Unified Modeling Language fuses the concepts of Booch, OMT, and OOSE. The result is a single, common, and widely usable modeling language for users of these and other methods.

Second, the Unified Modeling Language pushes the envelope of what can be done with existing methods. As an example, the UML authors targeted the modeling of concurrent, distributed systems to assure the UML adequately addresses these domains.

Third, the Unified Modeling Language focuses on a standard modeling language, not a standard process. Although the UML must be applied in the context of a process, it is our experience that different organizations and problem domains require different processes. (For example, the development process for shrink-wrapped software is an interesting one, but building shrink-wrapped software is vastly different from building hard-real-time avionics systems upon which lives depend.) Therefore, the efforts concentrated first on a common metamodel (which unifies semantics) and second on a common notation (which provides a human rendering of these semantics). The UML authors promote a development process that is use-case driven, architecture centric, and iterative and incremental.

The UML specifies a modeling language that incorporates the object-oriented community's consensus on core modeling concepts. It allows deviations to be expressed in terms of its extension mechanisms. The Unified Modeling Language provides the followings:

- Semantics and notation to address a wide variety of contemporary modeling issues in a direct and economical fashion.
- Semantics to address certain expected future modeling issues, specifically related to component technology, distributed computing, frameworks, and executability.
- Extensibility mechanisms so individual projects can extend the metamodel for their application at low cost. We don't want users to directly change the UML metamodel.
- Extensibility mechanisms so that future modeling approaches could be grown on top of the UML.
- Semantics to facilitate model interchange among a variety of tools.
- Semantics to specify the interface to repositories for the sharing and storage of model artifacts.

The followings are outside the scope of the UML:

- Programming Languages
  The UML, a visual modeling language, is not intended to be a visual programming language, in the sense of having all the necessary visual and semantic support to replace programming languages.
- Tools
  The UML defines a semantic metamodel, not a tool interface, storage, or run-time model, although these should be fairly close to one another.
- Process
  Many organizations will use the UML as a common language for its project artifacts, but will use the same UML diagram types in the context of different processes. The UML is intentionally process independent, and defining a standard process was not a goal of the UML.

[JPN-008G]
Standards for XML, MOF, CORBA, and UTF-8 should be referred to in the 2nd clause "Normative reference".

[JPN-009G]

All words in glossary should be declared in the 3rd clause "Terms and definitions".

For example,
3.1
abstract class (Bold face)
A class that cannot be directly instantiated. Contrast: concrete class.

[JPN-010G]
All abbreviations in this document should be explain in the clause "Abbreviated terms" (alphabetical order).

The following is example, complete it.

The following abbreviations are used in this international standard:

DCOM Distributed Common Object Model?
OCL Object Constraint Language
OOFRam (what's this?)
One of the software development methodologies.
OMT Object Modeling Technique
One of the software development methodologies.
OOSE Object-Oriented Software Engineering.
One of the software development methodologies.
MOF Meta-Object Facility
ROOM Real-Time Object-Oriented Modeling
One of the software development methodologies.

[JPN-011G]
Change the word "chapter" into "clause" to conform directives.

[JPN-012E]
Following double quotation marks should read as a left double quotation mark and a right double quotation mark.

p. 2-8, 2.3.1, line 27: "lightweight"
p. 2-10, 2.3.4, line 8: "instance.", "a Class instance" and "an Association instance,"
line 9: "a Class", "an Association", "a" and "an"
line 10: "an interface of"
line 11: "Elements," and "a set (or the set) of instances of the metaclass Element"
p. 2-68, 2.6.1, line 10: "virtual"
line 11: "virtual", pseudo" and "base"
p. 2-69, 2.6.1, line 2: "pseudoattributes"
p. 2-71, 2.6.2.2, in Second table: "pseudo metaclass"
p. 2-72, 2.6.2.3, line 2: "virtual"
p. 2-76, 2.6.4, line 2: "stable"
p. 2-10, 2.3.3, line 2: "Object Constraint Language Specification"

About the character names, see http://charts.unicode.org/ (Unicode 3.0 Character Chart). and so on.

[JPN-013E] p. 2-20, 2.5.2.2. First table
"implicit" in the left box in the table of "Stereotypes" should read as "<<implicit>>".

[JPN-014E] p. 2-22, 2.5.2.4. Fourth row in the table
"instance. An instance value is..." and "classifier. A classifier itself is..." should read as "instance- An instance

8

value is..." and "classifier- A classifier itself is...", respectively.

[JPN-015E]
Make the font of "and", "or", "not" or "implies" with boldface.

They are at following places:
p. 2-52, 2.5.3.10: The example of [2]
p. 2-53, 2.5.3.10: The example of "Additional operations" [1]
p. 2-59, 2.5.3.31: The example of [1]
p. 2-99, 2.9.3.7: The example of [1]
p. 2-99, 2.9.3.10: The example of [1]
p. 2-100, 2.9.3.12: The example of [3]
p. 2-100, 2.9.3.12: The example of [4]
p. 2-102, 2.9.3.16: The example of [1]
p. 2-102, 2.9.3.18: The example of [1]
p. 2-111, 2.10.3.1: The example of [1]
p. 2-112, 2.10.3.3: The example of [2]
p. 2-112, 2.10.3.4: The example of [1]
p. 2-113, 2.10.3.4: The example of [3]
p. 2-114, 2.10.3.6: The example of [1], [2] and [6]
p. 2-123, 2.11.3.1: The example of [1]
p. 2-141, 2.12.3.4: The example of [1]
p. 2-142, 2.12.3.4: The example of [3]
p. 2-166, 2.13.3.4: The example of [1]
p. 2-167, 2.13.3.4: The example of [2] and [3]

[JPN-016E] p. 2-94, 2.9.2.13: Title of the second table
The title of the table should be "Standard Constraints", because the constraints are explained in the second table.

[JPN-017E]
The right mark of the quotation shown below is a left single quotation mark. A right single quotation mark should be used.

p. 3-51, 3.30.2, line 3: '<',
p. 3-58, 3.38.2, line 15: '[',
p. 3-133, 3.76.2, line 2: '('

[JPN-018E]
The quotation shown below is enclosed with two apostrophes. It should be enclosed with a left single quotation mark and a right single quotation mark.

p. 3-144, 3.82.2, line 2: '*',
p. 3-145, 3.82.3, line 3: '*',

[JPN-019E] p. 3-133, 3.76.2, line 2: "... comma-separated-parameter-list ')"
There is not a right single quotation mark at the end.
"...comma-separated-parameter-list ')" should read as "... comma-separated-parameter-list ')'".

[JPN-020E] p. 4-2, 4.2, Table 4-1
Delete the second to Fourth rows from the bottom, because the second to fourth rows from the bottom are same as the three rows above.

[JPN-021E] p. 7-3, 7.2.1, line 9: "In the example. the keyword of..."
'.' is used. "In the example. the keyword of..." should read as "In the example, the keyword of...".

[JPN-022E] p. 7-3, 7.2.1, line 11:
Delete the sentence "OCL expressions are written using ASCII characters only." according to the comment: TH001.

[JPN-023E]
p. 7-5, 7.3.3, line 14: "The label inv: declares the constraint to be an <<invariant>> constraint."
p. 7-6, 7.3.4, line 3: "The stereotype of constraint is shown by putting the labels 'pre:' and 'post:' before actual Precondition and Postcondition."

The expression of the first sentence and the second one should be same, in order that the contents of the two sentences explain about the same kind of constraints. The second sentence should be revised as shown below:

> The labels pre: and post: declare the constraints to be a <<precondition>> constraint and a <<postcondition>> constraint, respectively.

[JPN-024E] p. 7-10, 7.4.7, line 1:
"The operators '+', '-', '*', '/', '<', '>', '◇', '<=', '>=' are used as infix operators."

The last sentence (p. 7-10, 7.4.7, line 8) says: the operators 'and', 'or' and 'xor' are infix operators. However, these logical operators are not included in the sentence as shown above. So, the sentence should be revised as shown below:

> The operators '+', '-', '*', '/', '<', '>', '◇', '<=', '>=', 'and', 'or' and 'xor' are used as infix operators.

[JPN-025E] p. 7-51~53, 7.9: All definition including "<name>", "<string>" and "<number>"

Delete enclosing marks of <name>, <string> and <number>, because they are non-terminal symbols.

[JPN-026E] p. 7-51, 7.9: Definition of "literal"
"STRING" should read as "string", because the non-terminal symbol "STRING" is not defined.

[JPN-027E] p. 7-52, 7.9: Definition of "string"
The indent of ')', ')*' and '"¥'"' are different from other rules. This definition should be revised as shown below:

```
string  := "'" ( ~["'", "¥¥", "¥n", "¥r"])
                 | ("¥¥"
            ( ["n", "t", "b", "r", "f", "¥¥", "'", "¥""]
                | ["0"-"7"] ( ["0"-"7"] )?
                | ["0"-"3"] ["0"-"7"] ["0"-"7"] )) )* "'"
```

[JPN-028E]
Following duplications should be eliminated.

p. A-1, Two rows from the bottom are same: "destroyed / Association / Constraint"
p. A-2, row 9 and 10 from the bottom are same: "new / Association / Constraint"
p. A-3, row 4 and 5 from the bottom are same: "transient / Association / Constraint"

[JPN-029E] p. A-2, row 14: implicit
The Standard Element Name "implicit" is not enclosing with a left-pointing double angle quotation mark ('<<') and a right-pointing double angle quotation mark ('>>'), although it is a stereotype. "implicit" should be enclosed with a left-pointing double angle quotation mark ('<<') and a right-pointing double angle quotation mark ('>>').

VOTE ON ISO/IEC/DIS 19501-1	
date 2000-10-13	ISO/IEC/JTC 1
national body PORTUGAL	

FORM 12

iD 32620 SC7

To cast a vote on a draft International Standard, national bodies shall complete and sign this ballot paper, and return same with any comments to the ISO Central Secretariat.

All national bodies are invited to vote. *P-members of the joint technical committee concerned have an obligation to vote.*

☐ We approve the technical content of the draft as presented (editorial or other comments may be appended)

2000-10-16

☐ We disapprove for the technical reasons stated at annex

☐ Acceptance of specified technical modifications will change our vote to approval

☒ We abstain (~~for reasons below~~)

Remarks:

Harmonization of UML and EXPRESS (ISO 10303-11) is important and necessary in the future.

Instituto Português da Qualidade

signature

Texte français au verso

JTC1/SC7
ID032620

SNV

Swiss comment on ISO/IEC DIS 19501-1

Comment justifying the negative Swiss vote on this DIS:

Dans la lettre d'accompagnement du vote sur UML, il est écrit:

"RM-ODP Part 2 (ISO/IEC 10746-2) defines the foundational concepts and modelling framework for describing distributed systems. This includes modelling abstractions such as "system" "object", "interface", "class", etc. Since both RM-ODP and UML are based on the object paradigm, each of the core concepts in Part 2 can be mapped directly into corresponding UML modelling concepts. Note that because it has a broader scope than RM-ODP, UML has additional modelling abstractions not found in RM-ODP. However, these additional modelling capabilities do not change or contradict the basic provisions of RM-ODP. Consequently, UML conforms to Part 2 of the RM-ODP standard, i.e., it represents a concrete realisation of Part 2."

Tout ceci n'est malheureusement pas correct. ODP a été conçu pour écrire des normes (spécifications) précises sur des systèmes répartis ou des composants de ceux-ci. UML a été conçu pour concevoir et documenter des implantations en SW. Ceci introduit une importante différence sur le sens (et les contraintes) que les 2 normes donnent au concept d'objet. La définition d'objet en ODP est beaucoup plus générale que celle de UML. Elle ne se limite pas à un objet de programmation, contrairement à UML.

Il est vrai qu'on peut plus ou moins utiliser la notation UML pour ODP si l'on ignore la sémantique d'UML, mais cela est ennuyeux... Et ce n'est pas seulement la notation d'UML qui devrait être acceptée comme une norme par l'ISO.

Concrêtement, voici un problème précis où UML 1.3 est incompatible avec le RM-ODP. En ODP, un objet peut avoir un nombre arbitraire d'interfaces, et peut même créer dynamiquement de nouveaux interfaces sur lui-même (correspondant, par exemple, à l'ouverture de nouvelles connections sur un serveur). Ceci permet à un objet ODP de modéliser un système complexe, tel que par exemple un serveur de noms X.500. UML ne voit derrière objet que le concept d'objet de programmation dans un langage à objets. Un objet UML n'a donc qu'un interface serveur, et est donc incapable de représenter un système dans un modèle.

Il est à noter que la technologie CORBA, de l'OMG, permet de créer dynamiquement de nouveaux interfaces sur un objet (CORBA, pour des raisons historiques, a une mauvaise terminologie: un object CORBA étant en fait un interface. Cependant, ce point est maintenant bien compris par la communauté CORBA) . Il est dès lors étonnant et d'autant plus regrettable que UML ne soit pas compatible avec CORBA sur ce point. L'ISO pourrait demander à l'OMG de clarifier la relation entre ses propres spécifications avant de les soumettre à l'ISO.

Un autre problème concret est le fait que les intéractions entre objets en UML sont limitées à 2 sortes: stimulus et signal (sans que personne ne comprenne pourquoi UML fait une différence entre ces 2 concepts d'intéraction). Le modèle objet générique de la partie 2 du RM-ODP n'impose aucunement une telle restriction sur les intéractions entre objet. Il est donc possible de spécifier que plusieurs objets participent dans une même action jointe et atomique, dans laquelle plusieurs objets

Alors que l'ISO est en général intéressé à spécifier comment des systèmes normalisés se comportent, tels que ce comportement est observé depuis l'extérieur du système. L'ISO se garde bien de dire comment le système doit être implanté. Dans ce sens, l'ISO s'intéresse à des spécifications abstraites. UML, dans sa sémantique, est incompatible avec spécifications abstraites. Officiellement, UML et un langage pour écrire et documenter des applications SW (donc, au niveau de l'implantation). Dans ce sens, une spécification UML est une spécification concrête (une implantation partielle voire complète). A cause de cette vision trop étroite de la modélisation, UML n'est pas un bon langage pour écrire des spécifications abstraites. Il est possible que UML 2.0 améliore considérablement cette situation, mais cela reste loin d'être certain. Le but officiel de UML reste en effet le même:
écrire et documenter des applications SW.

Extrait de la future norme ISO:

"3.1 Recommendations | International Standards
This Recommendation | International Standard makes use of the following terms defined in ITU-T Rec. X.902 | ISO/IEC 10746-2:

object
interface,"

Comme indiqué ci-dessus, la définition d'objet en RM-ODP est beaucoup plus générale que celle d'UML. Il est douteux que la définition d'objet de ODP puisse être substituée à celle d'UML sans compromettre UML.

Soit le lien entre UML et ODP devra être rompu, soit l'ISO devra demander de nombreux amendements à UML pour le rendre consistent. En particulier, permettre à des objets d'avoir de multiples interfaces, et introduire un concept généralisé d'action dans UML.

ID 32620 SC7 UNITED KINGDOM

Editorial Comments accompanying the UK Vote on DIS 19501-1, Information Technology- Unified Modelling Language (UML) – Part 1: Specification

1. Format of standard

It is acceptable, given the complexity of the UML specification, that the format of the existing OMG specification is not changed for its publication as a standard. Nevertheless, there introductory text is required that states the scope of the standard and explains its relation to other ISO standards (cf ISO 19500-2). Such text should also explain what other Parts of 19501 are to be expected.

In order to not to disturb existing numbering, such text should be included as an Introduction and Clause 0 of the standard, replacing the existing Preface which could be merged with the Introduction.

A suitable basis for such text is provided in the Explanatory Report submitted with the text for Ballot, subject to the comment in 2 below.

The approach to be taken needs to worked out with OMG experts but the UK will provide proposals for consideration by the Editing Meeting.

2. UML relation to ODP

1.3.2 of the Explanatory Report states that

"...each of the core concepts in Part 2 can be mapped directly into corresponding UML modelling concepts"

and that

"Consequently, UML conforms to Part 2 of the RM-ODP standard, i.e., it represents a concrete realisation of Part 2."

These statements are too strong since:

- the scopes and objectives of the RM-ODP Part 2 and the UML, while related, are not the same;
- in a number of cases the RM-ODP Part 2 and the UML specification use the same term for concepts which are related but not identical (e.g. interface).

The essential relationship is that a specification using the Part 2 modelling concepts can be expressed using UML with appropriate extensions (using stereotypes, tags and constraints).

This should be taken into account in using text from the Explanatory Report as Introductory material for the standard, as proposed in 1.

JTC 1 SC 7
id 32620

Subject: US Vote on ISO/IEC DIS 19501-1
Date: Mon, 2 Oct 2000 14:11:11 -0400
From: Susan Bose <SBOSE@ANSI.org>
To: 'DIS Votes' <votes@iso.ch>
CC: Barbara Bennett <bbennett@itic.org>, ISOT <ISOT@ANSI.ORG>

2000 -10- 03

Please accept this transmission as official notification of the U.S. vote for ISO/IEC DIS 19501-1, Information technology - Unified Modeling Language (UML) - Part 1: Specification.
The U.S. vote is to APPROVE WITH COMMENTS. The comments are:
"The US votes Yes with the following comments on Fast Track PAS DIS 19501-1:
Item: 1
Qualifier: editorial
Location: References
Rationale: The references should be updated to point at the ISO standard for those OMG specs which are ISO standards at the time of publication of 19501-1."
Please do not hesitate to contact me if I may be of further assistance.
Sincerely,
Susan Bose
For the US P-member JTC 1/.SC 7