



ISO/IEC JTC1/SC7
Software Engineering
Secretariat: CANADA (SCC)

ISO/IEC JTC1/SC7 N2251

2000/01/28

Doc. Type	Comment Disposition Report
Title	Comment Disposition Report of CD 15414: Information Technology - Open Distributed Processing - Reference Model - Enterprise Viewpoint.
Source	JTC1/SC7 Secretariat
Project	07.77
Status	Final
References	N2225, N2187
Action ID	FYI or ACT
Due Date	
Mailing Date	2000/01/28
Distribution	SC7_AG
Medium	Encoded Acrobat
No. of Pages	76
Note	Joint project with ITU-T (ITU-T X911).

Introduction

This document records the disposition of all comments received in response to ISO/IEC JTC1/SC7 N2187, issued for CD Ballot on 1999-07-19, and discussed at the WG17 meeting in Paris, France in November 1999.

Comments from Canada

CAN A01 Scope of the document

Comment

The issue raised by the "temporary note" in clause 5 of [1] is critical to the document. If it is not possible to clearly define the scope and rationale for the document, then it is somewhat risky to develop later parts. Resolving these issues has more impact than just editing of the clause. For instance the following text would need to be revised:

Clause 1, a)

Clause 7 first paragraph

Clause 7.1 first paragraph

Discussion

By definition (10746-1), an ODP system is an information processing system using information technology (an automated information system). An automated information system is a subset of an information system, which is a subset of an enterprise system. These form part of the context (or environment) of the ODP system. Therefore the UofD (universe of discourse) is:

1 ODP systems;

2 Context of ODP systems (enterprise systems);

3 Interfaces between ODP systems and enterprise systems.

Obviously (10746-1, clause 6.1) ODP standardization is not about "producing constraints on specifications" of enterprise systems.

It is therefore about ODP systems and/or their interfaces to enterprise systems.

A viewpoint (in 10746-1 6.2.2) is a specification of the whole system, whereas in 8.1.1 it is a subdivision of a specification of a complete system.

The enterprise viewpoint of an ODP system (and a supporting enterprise language) could be:

1. A full specification of the interface between an ODP system and its environment. This would be like the external design of the system;

2. A partial specification of the same interface, limited to components that are enterprise-driven (business rules, authorization information, etc.) and required to operate the ODP system;

3. A full specification of an ODP system, as a subset on an enterprise system, and in a manner understandable to the enterprise. This would be expressing the internal design of an ODP system in business language;

4. A partial specification of an ODP system, limited to components that are business driven (ownership information, security algorithms, location and distribution information, replication rules, ...).

It is possible to find in the current text of 10746-1 and 15414 material supporting any of the four possibilities, and other text that could be interpreted as meaning that the enterprise language is a language to specify enterprise systems.

Clause 5 should identify clearly what type of specification is proposed and what it is to be used for, and also clearly identify what it is not intended to do. Until that scoping decision is taken and clearly documented, it is difficult to discuss the rest of the document.

Disposition: Noted – the WG will consider this in detail over the next period

CAN A02 Components of a language standard

Standards for languages, and other types of formalisms (such as diagrams) have already been produced in many areas (programming languages, modelling formalisms). Given that a language or formalism is a mechanism used to describe or prescribe something else, these standards usually have the following three components:

1. The structure of the language (constructs and their semantics, associations between constructs and rules governing them, syntactical (diagrammatic) conventions to represent the language.
2. The elements of the real world the language can be used to describe or prescribe
3. The relationships between real world objects and language constructs, and rules governing them.

Since the title of the document is "Enterprise Viewpoint", and not "Enterprise Language", the document intends to define a set of standardized specification elements, or the semantic part of a language. Even if it does not go into external representation, the document itself should be structured more like a language standard.

In particular, it should be clear to the reader when a specification component is prescribed, vs. when a real world object is described.

Disposition: Accepted in principle

CAN A03 Motivation for a standardized enterprise specification for ODP system specification

The rationale for language standards is for enabling communication and consequently helping understanding.

The current text at the end of clause 5 insists on the benefits of the existence of an enterprise specification. It should emphasize the benefits of a standardized specification.

Disposition: Accepted

CAN A04 Audience of the document

It is not clear if the target audience of this document is. This can affect the style of the document.

If the target audience is those people who design languages for such type of specification, then the style has to be quite formal, and the normative part of the standard rigorous enough to enable conformance verification.

If the target audience is those who prepare or use such specification, then the nature of the document is more like a guideline, and the style is less formal.

The target audience of the document should be clearly stated in clause 1, and the requirements from this target audience stated in clause 5.

Disposition: Editor to implement – new note in Clause 5

Comments from Finland

Fin 1	Cat G	Page	Sect	Para	Line
-------	-------	------	------	------	------

The working document is not clear in the definition and the use of terms role, role type, community, community type, community template, and enterprise object. Our view is as follows:

- A community specification discusses various roles as placeholders for potentially multiple objects. An object populating a role must be of object type that is compatible with the role type for that specific role. The relationship between role and object is not a type-instance relationship.
- Role can be assigned with a cardinality. The cardinality threshold values express the limits within which the number of “object placeholders” of the given role type may vary within the community. Cardinalities could as well be expressed as indirect population criteria referencing to properties of the community instead of only properties of the objects.
- A community specification can be expressed either as a community type or as a community template. In case of community type specification, the community (instance) where enterprise objects interact is established by populating the roles based on the criteria expressed as role type and population criteria. In case of community template, the population process may be implemented as an object instantiation process, where role specifications are used as object templates. However, we consider this as a special case of the first one leading to fixed communities.
- Communities are allowed to modify themselves by deleting and adding roles of already specified types. The population process provides a mechanism for dynamic changes on the community participating enterprise objects. The obligations of the community are unchanged independent on whether a role is populated at a given time or unpopulated. Obligations can only be changed via explicit community changes.

Suggested text improvements founded on this rationale below.

Disposition: Noted

Fin-2	Cat G	Page	Sect	Para	Line
-------	-------	------	------	------	------

The working document is not clear in the ways allowed for community establishment. The population process is not adequately described, instead frequent references to instantiation of enterprise objects and community instances can be found. As stated in FIN-1: 3, we consider the population process the essential one.

Furthermore, the specification method for communities appears to confuse between roles and objects, especially putting too much emphasis on identifying enterprise objects instead of roles. The alternative ways of specifying a community need to be explained separately.

Suggested text improvements below.

Disposition: Subsumed in resolution of other comments

Fin-3	Cat G	Page	Sect	Para	Line
-------	-------	------	------	------	------

The working document is not clear in specifying the possible relationships between communities, nor in describing the possible relationships between roles. The following view to these relationships gives a rationale for text suggestions to follow.

The relationships between enterprise specifications can be divided into two essentially different categories:

- A) relationships that deal with the behaviour of the specified communities; and
- B) relationships that deal with the production of the enterprise specifications.

Both categories comprise several sub-cases where modifications to the enterprise properties (e.g., enterprise policies) are managed differently.

In addition,

C) communities may interfere with each other at the level of object instances populating roles.

A. The behaviour of two or more communities can be related via two basic mechanisms:

1. Communities can be designed to interact.

Two or more communities can be interlinked by designing how their policies, interactions, interfaces, and behaviours interleave.

The purpose of creating independent communities can be modelling of autonomous domains or modelling of independent service provision.

Examples:

the legal domains involved in international goods markets;

a company out-sources the provision of an information service.

Cooperation between communities is in this case captured at design time, and requires that the specifications of all involved communities are created, controlled, and modified by the same authority.

Any modification to the community specifications need to be taken to all involved specifications equally, to preserve consistency. This must either be done manually, or can be supported by tools. However, the result of the modifications is a new set of specifications, and a new set of communities.

A specific form of interaction can be created between communities by requiring that same object fulfils specific roles in the communities interacting through the shared object.

2. Communities can enter a cooperation state through a binding process at the system run-time.

Two or more communities can be interlinked through a match-making process that considers their policies, interacting capabilities, interfaces and behaviour descriptions.

The involved communities are in this case designed separately without a common controlling authority. Furthermore, the cooperation between the communities does not necessarily create a common authority for them.

This case is modelled with the help of community equivalent object (CEO): A community specification is considered as a property description of a class of object instances that are capable of fulfilling the joint responsibilities of all roles in the community.

A role in any community specification can thus validly be fulfilled by a CEO.

As the binding process can be late and dynamic, a single community specification can be created and modified independently from others. The consistency requirements are checked at the binding time.

Modifications to a single community (e.g., policy change) is propagated to other involved communities only if that is separately agreed as an activity between the various communities. Such an activity may be defined as a peer-to-peer negotiation or as an authority-to-subordinate relationship.

The example cases for dynamic inter-linkage are similar to the examples given in case 1.

B. In the production of enterprise specifications,

the interesting relationships include: nesting of communities, and reuse of role specifications. These cases are closely related to A.1.

Nesting of communities means, that a partial community is textually separated as a community of its own, and the textual representation can be reused (by a specification tool) as part of multiple, potentially unrelated community specifications.

Reuse of role specifications works similarly, but in this case the isolated textual representation covers only a single role.

The role and community specifications can be private to a specification and development environment, or they can be stored to a repository that can be shared by a wider audience of several development environments and groups.

C) Object instances may populate roles in more than one community simultaneous.

In such a case, the behaviour of the object is restricted by the policies of each community separately. An object should not enter a community (bound to one, designed to be part of one) when contradictory requirements occur. However, the policies of the simultaneously participated communities are not necessarily consistent, but the object may end up into a contradictory situation where it necessarily causes a failure in one of the communities it participates.

This case is closely related to the role population process, that also plays an important role in A.2

Disposition: Noted

Fin-4	Cat G	Page	Sect	Para	Line
-------	-------	------	------	------	------

The concepts presented for processes and the relationships between processes, actions and roles are unclear. We consider that the following concepts are essential and should be used.

Behaviour (of an object): A collection of actions with a set of constraints on when they may occur.

Behaviour is a combined set of all possible sequences of actions (internal and interactions) that the object is capable to participate, given the right environmental circumstances.

Composition of behaviours (Part 2, 9.1): A combination of two or more behaviours yielding a new behaviour. The characteristics of the resulting behaviour are determined by the behaviours being combined and the way they are combined (sequential composition, concurrent composition, interleaving, concealment of actions).

Cobehaviour is a composition of behaviours where the interactions of two or more objects are shown, and the internal actions hidden. The interactions visible cover both the interactions amongst those objects themselves and the interactions with the environment of the objects under composition.

Role: An identified set of activities relevant for the cobehaviour of a community.

Activity: A single-headed directed acyclic graph of actions, where occurrence of each action in the graph is made possible by the occurrence of all immediately preceding actions.

Single-headedness is a result of taking a single object in a given state as a starting point. Thus, the behaviour graph is restricted (partitioned) to separate subgraphs, each presenting a case where a certain environmental or internal prerequisite has been met.

Notes from Part 2: Action and activity are degenerate cases of behaviour. In general, several sequences of interactions are consistent with a given behaviour.

Action is anything that happens, according to Part 2.

Process: An identified subgraph of the community cobehaviour limited between interactions between the community and its environment (trigger and result).

Task: A labelled graph of actions. A task can be related either to cobehaviours (and thus, processes) or to activities (and thus, roles), and used for locating corresponding interactions presented in both role and process based approaches. The tasks need not to be identical, but some sort of equivalence class is more suitable.

Some of these are already present in Part 2 and 3 of ODP RM, and should not be repeated in this document. The suggested definitions and text changes are given below.

Disposition: Noted – concepts to be used so far as possible

Fin-5(a) Cat E	Page	Sect 5	Para 2	Line
----------------	------	--------	--------	------

Replace by

“The enterprise language provides the vocabulary and constructs to specify the purpose, scope and policies for an ODP system in terms that are meaningful for the stakeholders for that system. An enterprise specification describes the behaviour of the system within the with which it interacts. Such an environment can be a technical environment (e.g., software and hardware environment of a service component) or a social or business organisation (e.g., a group of co-operating companies, particular service inside a company). “

Disposition: Accepted - Editor to implement

Fin-5(b)	Cat E	Page	Sect 5	Para 4	Line
----------	-------	------	--------	--------	------

“An enterprise specification of an ODP system is an abstraction of the system and a larger environment in which the ODP system forms a part, describing those aspects that are relevant to specifying what the system is expected to do in the context of purpose, scope and policies of that environment (technical, organisational). It describes the behaviour assumed by those who interact with the ODP system. It explicitly includes those aspects of that environment that influence the behaviour of the ODP system – environmental constraints are captured as well as usage and management rules.”

Disposition: Accepted as re-worded

Fin-6 Cat E	Page	Sect 6.1.2	Para	Line
-------------	------	------------	------	------

Add note:

Note: A range of different kind of communities can be described. Some of them have a definite goal that can be reached and a termination condition can be captured as a objective statement. In a more general case, communities are intended to aspire an improved state without a termination condition, thus only preferences for particular outcomes of processes within the community can be stated.

Disposition: Rejected

Fin-7	Cat E	Page	Sect 6.2.4	Para	Line
-------	-------	------	------------	------	------

Keep 6.2.4.1 and the notes of 6.2.4.2; delete the rest of the text in 6.2.4.

Disposition: Subsumed in resolution of other comments

Fin-8	Cat E	Page	Sect 6.2.3.	Para	Line
-------	-------	------	-------------	------	------

Delete. Use of normal English meaning for party would be adequate. Contracting party as a term does not really reveal that the commitment has already been made, it might as well be on the way still.

Disposition: Rejected

Fin-9(a) Cat TH	Page	Sect 6.3.2.	Para	Line
-----------------	------	-------------	------	------

Replace “Role (of a community)” by “Role (in a community)” in 6.3.2. The genetive form is ambiguous.

Replace definition in 6.3.2 by

Role: An identified set of activities relevant for the cobehaviour of a community.

Disposition: Subsumed in resolution of other comments

Fin-9(b)	Cat TH	Page	Sect 6.3.1	Para	Line
----------	--------	------	------------	------	------

Replace the (empty) definition of behaviour in 6.3.1. by

Cobehaviour: A composition of behaviours where the interactions of two or more objects are shown, and the internal actions hidden. The interactions visible cover both the interactions amongst those objects themselves and the interactions with the environment of the objects under composition.

Fin-10	Cat TH	Page	Sect 6.3.3	Para	Line
--------	--------	------	------------	------	------

Replace the definition with the following and delete the three notes.

Process: An identified subgraph of the community cobehaviour limited between interactions between the community and its environment (trigger and result).

Disposition: Rejected

Fin-11	Cat TH	Page	Sect 6.3.4 , 6.3.5	Para	Line
--------	--------	------	--------------------	------	------

Replace the definition of task with the following.

Task: A labeled graph of actions. A task can be related either to cobehaviours (and thus, processes) or to activities (and thus, roles), and used for locating corresponding interactions presented in both role and process based approaches. The tasks need not to be identical, but some sort of equivalence class is more suitable.

Delete the definition of step.

Disposition: Deferred to next ballot cycle

Fin-12a	Cat E	Page	Sect 6.4.3
---------	-------	------	------------

Delete.

Disposition: Deferred to next ballot cycle

Fin-12b	Cat E	Page	Sect 6.4.4
---------	-------	------	------------

Delete.

Disposition: Deferred to next ballot cycle

Fin-12c	Cat E	Page	Sect 6.4.5.2.2 – 6.4.5.2.5
---------	-------	------	----------------------------

Delete.

Disposition: Subsumed in resolution of other comments

Fin-13	Cat E	Page	Sect 6.5, 6.7	Para	Line
--------	-------	------	---------------	------	------

Move 6.7.2 Commitment into policy concepts, making it 6.5.4.

Remove 6.7 Force concepts.

Disposition: Rejected

Fin-14	Cat TH	Page	Sect 7.1	Para	Line
--------	--------	------	----------	------	------

Replace by:

In an enterprise specification, an ODP system and the environment in which it operates are represented as a community. The objectives and scope of the ODP system are defined in terms of the roles it fulfils within a community and policy statements about those roles.

Communities can be defined either as community types or even as community templates.

An enterprise specification may thus include the specifications of

A community and the direct environment it interacts with, and

Any other communities of which the system or its parts are members.

A community specification defines the obligations, permissions, prohibitions and authorisation of each of the community members.

Notes 2 and 3 can be kept

Disposition: Accepted - covered by re-write

Fin-15	Cat E	Page	Sect 7.2	Para	Line
--------	-------	------	----------	------	------

Add opening sentences before 7.2.1:

A community is a configuration of enterprise objects interacting with each other, governed by policies committed to that community. A community is formed by populating a community specification according to its population policy. A community involves a contract, an agreement between enterprise objects, about a shared objective and shared constraints on interactions.

Disposition: Accepted

Fin-16	Cat TH	Page	Sect 7.2.1	Para	Line
--------	--------	------	------------	------	------

Replace by

A community specification is given in terms of

- Roles to be fulfilled by enterprise objects,
- Policies governing the activities related to each role,
- Processes which take place in the community,
- Policies governing the cobehaviour of objects related to each process,
- Epochs describing changes in the presence of various roles within the community and changes in the configuration of those roles,
- Population policies governing the assignment of enterprise objects to each role,
- Policies relating to environment contracts governing objects in the community.

In general, community type specifications form a type set in which various relationships between the types may be present. Corresponding cross references between the specifications concern may occur.

Roles can be described by role types. Role types can have various relationships, e.g., subtype hierarchies.

Disposition: Accepted - covered by re-write

Fin-17	Cat TH	Page	Sect 7.2.2	Para	Line
--------	--------	------	------------	------	------

Replace by

A community specification can be populated by assigning an enterprise object to each role the specification. The object type of the object that fills the role must be consistent with the role type. However, the role/object relationship is not a type/instance relationship. Furthermore, the enterprise object selected to play a certain role must fulfil the requirements of the population policy.

Note: As a special case, an instantiation process can be used for populating the community. In such a case, the role type is interpreted as an object template. However, the created communities are fixed both in respect of members, and of configuration.

The enterprise objects assigned for roles in the community can be dynamically changed during the lifetime of the community. As a consequence, a role can temporarily be empty. Still, the community is continuously responsible for the obligations placed on that role.

An enterprise object may become a member of a community because

- The community specification provides that the community includes the object,
- The object is made part of the community at the time of community creation, or
- The object becomes a part of the community as a result of dynamic changes in the configuration of the community.

Disposition: Editor to implement to replace 7.8.1 without the note. First sentence to be “A community is populated by assigning enterprise objects to roles of the community” and then what Joaquin has.

Fin-18	Cat E	Page	Sect 7.2.2, 7.8.2, 7.8.3	Para	Line
--------	-------	------	--------------------------	------	------

Move sections 7.8.2 and 7.8.3 to form the final paragraphs in 7.2.2.

Disposition: Accepted - move 7.8 to before Objective Rules.

Fin-19	Cat TH	Page	Sect 7.2.3	Para	Line
--------	--------	------	------------	------	------

Rationale: See FIN-3. We doubt the verifiability of the transitivity properties claimed in paragraph 3; furthermore, we are not sure that the property would be always beneficial.

Replace the section with the following text but keep the final note with examples.

An enterprise specification may specify several communities. A community need not be specified in isolation; rather, it may be considered in the context of some other community or communities to which it is related.

Community specifications can be related in various ways, including relationships where

Community specifications are interlinked by desinging how their policies, interactions, interfaces and behaviours interleave.

Note: The purpose of creating independent communities can be modelling of autonomous domains or modelling of independent service provision.

Note: In case of modelling federated communities, objects and roles involved may have contradicting requirements and constraints on their cobehaviour. If such a possibility exists, the specifications should include an explicit mechanism for resolving the contradicting situation.

Community specifications can be reused and nested within specification tools.

Communities can be related in various ways, including relationships where

A community (CEO, community equivalent object) fulfils a single role in another community. Two or more communities can be thus interlinked through a match-making process that considers their policies, interacting cababilities, interfaces and behaviour descriptions. The role to be fulfilled by the CEO can be either a core role or an environment role.

The same object fulfils roles in multiple communities. If communities require that the same object fulfills specific roles, the shared object can be used as a specific form of interaction between the communities.

Note: Not all situations where an enterprise object fulfils roles in multiple communities are relevant. The situation may well be accidental and carry no intended obligations for the object. In contrary, such a case may contradict the security policy of some community.

Role specifications, role type specifications, process specifications, and policy specifications can be related in various ways, including relationships where

specifications can be reused and nested within specification tools.

Communities with a need to interact are clearly parts of a larger, outer community, thus common policies will apply.

Disposition: Rejected in fact, but certain aspects Accepted particularly removal of transitivity para.

Fin-20	Cat E	Page	Sect 7.3	Para 1	Line
--------	-------	------	----------	--------	------

Replace by

Every community has one objective, which is defined in its contract, and can be refined into a collection of sub-objectives. The objective of the community is not necessarily a reachable goal but an aspiration towards some less specific aim.

Disposition: Last sentence as a note.

FIN-22	Cat E	Page	Sect 7.4.1	Para 1	Line 1
--------	-------	------	------------	--------	--------

Replace by

An object denoted by a community specification is involved ...

Disposition: Accepted in principle

Fin-23	Cat TH	Page	Sect 7.4.2	Para	Line
--------	--------	------	------------	------	------

Replace by

Role-based approach: The cobehaviour of the community is partitioned into a set of object placeholders, roles, giving restrictions for the suitable enterprise objects to play the role, and obligations for the object within the community. For each role, only the required object behaviour is described, not the total cobehaviour of the community nor the behaviour of interacting partners.

Replace by

Process-based approach: The cobehaviour of the community is partitioned into a set of processes, each describing how the community reacts to a trigger and reaches a result, and in this way achieves some particular sub-objective in the community. For each process, the required interactions between core objects and environment objects are described, not hiding the internal communication of the community.

Delete para 3-6 (Role behaviour decomposes not be a step.)

Replace step by task

Disposition: Rejected as no consensus on cobehaviour

Fin-24	Cat E	Page	Sect 7.4.3 , 7.4.4	Para 3	Line 1
--------	-------	------	--------------------	--------	--------

Replace by

When a community is established, one or more objects is associated with ...

Add

Roles can have cardinalities assigned to them. This is a shorthand for expressing that the same type of role can be multiply populated simultaneously.

Delete 7.4.3.1, 7.4.3.2 and 7.4.3.3 , 7.4.4

Disposition: Accepted - covered by re-write

Fin-24	Cat E	Page	Sect Annex B and parking lot	Para
--------	-------	------	------------------------------	------

Delete.

Disposition: Accepted

FIN-26	Cat TH	Page	Sect 10.3	Para	Line
--------	--------	------	-----------	------	------

The correspondence rules between enterprise language and other ODP viewpoint languages are not defined in terms of ODP vocabulary. In addition, the text related to corresponding concepts has been degenerated too far by iterated corrections.

In the illustrations, for enterprise viewpoint, the following boxes should be used:

- community
- Objective
- Scoping statement
- Role
- Process
- Task
- Actor role
- Artefact role
- Policy

For information viewpoint, the following concepts/boxes should be used:

- Information object
- Dynamic schema
- Static schema
- Invariant schema

The correspondences between enterprise viewpoint concepts and information viewpoint concepts include

- One or more information objects can represent the information content of an enterprise object which fulfils any kind of role in the enterprise specification.
- Invariant, static and dynamic schemata of information objects are governed by the enterprise viewpoint policies. The policies each information object must obey are those related to the whole community, those related to the enterprise object involved, and activities of that enterprise object as far as the information object is participating the activity.
- A dynamic schema is associated with a task.

For computational viewpoint, the following concepts/boxes should be used:

- Computational object behaviour
- Computational interaction
- Computational interface
- Binding object

The correspondences between enterprise viewpoint concepts and computational viewpoint concepts include

- A computational object may include multiple computational interfaces each of which represent a computational object behaviour. One or more computational interfaces can represent the computations related to an enterprise object.
- One or more interactions of a computational object are related to a task.
- Computational object behaviour is restricted by policies.
- Binding object is related to cobehaviour of enterprise objects.

For engineering viewpoint, the following concepts/boxes should be used:

- Basic engineering object
- Node
- Interceptor

The correspondences between enterprise viewpoint concepts and engineering viewpoint concepts include

- One or more basic engineering objects may represent an enterprise object.
- An enterprise object may locate at one or more nodes.
- A domain is a specific community type in the enterprise language. One or more interceptors may be needed to implement a domain boundary.

Disposition: Editor to implement

FIN-27	Cat	TH	Page	Sect	Annex	Para	Line
--------	-----	----	------	------	-------	------	------

The BNF representation of the enterprise specification structure does not conform with the text.

Furthermore, the population constraints have been moved into a too detailed level (actions) to be useful. Population constraints are not any more clear from the text either, and should be added.

Move <constraint>*<involves statement>* into <role-object assignment>.

Figure A-1 should be updated:

Objective and process should have N:1

Step and action should be removed

SuperWho should be removed or clarified, RoleFiller as well

Behaviour and SuperWho should have no relationship

Cobehaviour should be defined

Behaviour should have a relationship with task

Disposition: Deferred to next ballot cycle

FIN-28 Cat E Page Sect 7.5, 8 Para Line

Replace section 7.5, before 7.5.1, by section 8, thus removing 8.

Keep the first paragraph of 7.5.1, but remove the rest of the section.

Disposition: Editor to implement

Fin-29 Cat E Page Sect 7.5.2 Para Line

Replace the subtitle Ownership by “Policy makers” and continue 7.5 directly. Replace the text by

The default policy maker and controller for an object is the owner of that object. Each object has at most one owner, even if refined or delegated to a set of cooperating objects. An object can be owned by itself. Object ownership becomes specified at instantiation, but can be transferred, delegated or relinquished (temporarily, permanently).

Disposition: Delete heading, replace ll42-47 of p20 with proposed text. Editor to replace object with enterprise object.

Fin-30 Cat E Page Sect 7.5.3 Para Line

Add “authorization” to the title. Add subsection Authorization

If an enterprise object is authorised to an action, the target of that action has no permission to deny the action to take place. If denial occurs, that is a failure and a violation against the community contract.

Disposition: Editor to implement

Fin-31 Cat E Page Sect 7.5.3.1 Para 3 Line 24-34

Delete the last paragraph about deontic logic, and the temporary note

Disposition: Editor to implement

Fin-32 Cat E Page Sect 7.5.3.3 Para Line

Replace “Standing obligations ... violation condition” by

Standing obligations are always obligations where enabling conditions are always true, thus any interaction must conform to the obligation in addition to other constraints on it.

Disposition: Editor to implement delete 1st always

Fin-33 Cat TL Page Sect 7.5.6.1 Para Line

Successful performance of an interaction between enterprise objects of a community requires that a set of permissions **exists**. Required permissions are either

- Associated with a particular target role in this interaction, or
- Associated with the interaction as a whole.

It is not necessary, that there is an authorization to perform the interaction. If both authorization and permission for an interaction is missing, the interaction fails and may therefore make the enterprise object to meet a failure in fulfilling its role.

Objects can pass permissions and authorizations between themselves. This passing is itself an interaction, and is subject to same permission rules.

Disposition: Editor to implement but temp note about when permissions are used in a model

Fin-34 Cat E Page Sect 7.5.6.2 Para Line

The purpose of this section is unclear. Unless the intention can be clarified, the section should be deleted.

Disposition: Editor to implement – temp note

Fin 35 Cat E Page Sect 7.7 Para Line

See comment FIN-15 – FIN-19. The proposed text for 7.2. captures the ideas of current 7.7. Therefore 7.7 can be deleted.

Disposition: Accepted and done

Fin-36 Cat E Page Sect 7.9, 7.10 Para Line

Remove.

Disposition: Rejected

Fin-37 Cat E Page Sect 7.5.4 Para Line

Delete. The idea has already been inserted into the text by comments xxx

Disposition: Deferred – no text

Comments from France

AFNOR 1 CAT MAJOR: CLAUSE 0.2 LINE 35, 36, 37:

Proposal:

Delete this sentence.

Rationale:

We think that the correspondence rules are not well defined.

Disposition: Rejected

AFNOR 2 CAT MAJOR: CLAUSE 1 LINE 11, 12

Proposal:

Delete this sentence.

Rationale:

We think that the correspondence rules are not well defined.

Disposition: Rejected

AFNOR 3 CAT MINOR: CLAUSE 5

Proposal:

Replace line 39 to 41 by:

"The enterprise language defines the concepts necessary to represent the behaviour expected of an ODP system. It defines structuring rules for using those concepts to produce an enterprise specification."

Rationale:

We prefer to not deal with community in this clause.

Disposition: Accepted

AFNOR 4 CAT MINOR: CLAUSE 6.1.1

Proposal:

Delete this definition

Rationale:

We argue that this definition is equals to the common sense.

Disposition: Deferred to next ballot cycle

AFNOR 5 CAT MAJOR: CLAUSE 6.2.1

Proposal:

We propose to not define “resource”.

Rationale:

In our point of view, a resource is kind of artefact. But, we do not need this definition.

Disposition: Rejected

AFNOR 6 CAT MAJOR: CLAUSE 6.3.1

Proposal:

Behaviour (of a community): A behaviour of a community is composed of actions that are identified by the roles of the community. Constraints on these actions must be consistent with the constraints identified by the roles of the community.

Rem: Actions identified by a role of a community may not be included in the behaviour of that community.

Rationale:

Objects live in a community to perform a co-operative work. They perform some actions to realise the objective of that community. These actions are identified by the roles of the community. The behaviour of the community expresses this co-operative work.

Disposition: Put in 7.2.1 (old numbering)

AFNOR 7 CAT MAJOR: CLAUSE 6.3.2

Proposal:

We propose to not define the "Role (of a community)"

Rationale:

We argue that it is better to explain that:

Roles belong to one community. When a role belongs to one community, objects fulfilling it must belong to this community.

Disposition: Accepted

AFNOR 8 CAT MAJOR: CLAUSE 6.3.3

Proposal:

Process:

A collection of tasks taking place in a prescribed manner and leading to the accomplishment of some result.

Rationale:

We have not understood the difference between a task and a step. If a step is just a "task" in a process, then we think that we do not need to define it.

Disposition: Deferred to next ballot cycle

AFNOR 9 CAT MAJOR: CLAUSE 6.3.4

Cat 4

Proposal:

Task:

A task is an action that belongs to a behaviour identified by a role.

Rationale:

We are not sure that the behaviour of a community includes all actions that are identified in the roles of the community. We think that there are a difference between an action that belongs to a behaviour of a community and an action that belong to a behaviour identified by a role.

Disposition: Deferred to next ballot cycle

AFNOR 10 CAT MAJOR: CLAUSE 6.3.5

Proposal

We do not need to define a "step".

Rationale:

If a step is just a task in a process, we do not think that we really need to define it.

Disposition: Deferred to next ballot cycle

AFNOR 11 CAT MAJOR: CLAUSE 6.4.1, 6.4.2, 6.4.3, 6.4.4

Proposal:

We think that we need these four definitions.

Rationale:

We think that the distinction between Actor role and Artefact role could be very useful (for example for the generation of code).

These two concepts are very well defined thanks to the concepts of Actor and Artefact.

We think that we do not need to define a resource, for us a resource is a sort of artefact.

Disposition: Deferred to next ballot cycle

AFNOR 12 CAT MAJOR: CLAUSE 6.4.5

Proposal:

We choose the fourth alternative.

We want to define:

Interface roles (in a community): Roles in a community that identify interactions identified by core roles. If the core community is represented as an object in a larger community, then this object must fulfil interface roles.

Rationale:

We think that core roles are useful to create CEO. Then, interface roles are the interface of the CEO.

Disposition: Deferred to next ballot cycle

AFNOR 13 CAT MAJOR: CLAUSE 6.6.2

Proposal:

Replace Agent by: Delegate (A delegate)

Rationale:

We think that the term "Agent" should not be defined.

We also work with agent, and this definition is not consistent with our definition.

Disposition: Editor to implement note asking for suggestions

AFNOR 14 CAT MINOR: CLAUSE 7 LINE 4

Proposal:

The model is composed of a structure of model of communities.

Rationale:

If a model is an enterprise specification (line 1), a specification of a model is a specification of a specification. We think that we just deal with enterprise specification, not with specification of specification.

Disposition: Accepted “The model is a specification in terms of...”

AFNOR 15 CAT MINOR: CLAUSE 7 LINE 4

Proposal:

In an enterprise specification, an ODP system and the environment in which it operates are represented as a community. At some level of description the ODP system is represented as an enterprise object in that community, this community is called the <S>community. The objectives and scope of the ODP system are defined in terms of the roles it fulfils within the <S>community, and policy statements about those roles.

An enterprise specification includes the specifications of:

- the <S>community
- any other communities of which the system or its parts are members, and
- other communities of which objects in the environment of the system are members.

Rationale:

We think that this community is really important; it is composed of the roles that specify the objectives and scope of the ODP system.

Disposition: Accepted in principle by re-write

AFNOR 16 CAT MINOR: CLAUSE 7.2.1

Proposal:

Remove line 3 to 9

Rationale:

These structuring rules should belong to 7.8

Disposition: Accepted - covered by re-write

AFNOR 17 CAT MINOR: CLAUSE 7.2.1

Proposal:

Add this note:

There are different levels of abstraction for the specifications of communities. A specification could be very abstract (with only of role types); or a specification could be very concrete (with very detailed description of object and very detailed description of roles).

Rationale:

We think that a community specification could be a description of a single community, but also it could be a description of a set of communities.

Disposition: Accepted - covered by re-write

AFNOR 18 CAT MINOR: CLAUSE 7.2.2

Proposal:

A community is established by a contract, an agreement between parties (enterprise objects). The objective of the community will be consistent with the objectives of all of the parties of that community contract.

Rationale:

We think that the objective of a community is more than a subset of the objectives of all of the parties of the community.

Disposition: Accepted - covered by re-write

AFNOR 19 CAT MAJOR: CLAUSE 7.2.3

Proposal:

Replace line 25 to 39 by:

These communities can be related in various ways, including relationships when:

- The enterprise specification prescribes that the object fulfilling a role in one community be part of other communities, perhaps fulfilling a certain role in each of those communities.
- A composite object is part of a community and it or some of its component objects are parts of other communities.

Rationale:

We think that there is only kind of relationship between communities.

Line 26 and 27 mean the same thing.

Line 29 is too abstract.

Line 37 and 38 mean the same thing.

Disposition: Editor to implement as re-written

AFNOR 20 CAT MAJOR: CLAUSE 7.3

Proposal:

Replace line 38 to 44 by:

Objectives of a community could be expressed by a collection of interrelated roles, processes, and policies. Objectives expressed in a role are realised by objects fulfilling the role. Objectives expressed in a process are realised by objects performing the actions of the process. Policies describe the parts of the community behaviour which are not yet prescribed, but which influence decisions within behaviour on a case-by-case basis in order to steer the behaviour towards an objective. Typically roles, processes and policy will be designed to meet different each sub-objective. A policy may influence the occurrence of a specific action in a process, which may in turn result in occurrences of other steps as well as creation (or termination) of processes.

Rationale:

We think that it is to confusing to say that a role has an objective.

Disposition: Agreed as re-worded

AFNOR 21 CAT MAJOR: CLAUSE 7.4.1 LINE 17

Proposal:

Remove line 17, 20 and 21

Rationale:

We think that a resource is a subtype of an artefact. We think that we do not need to define a resource.

Example of action: Take a car.

Does the car is a resource or an artefact?

We do not know, that is why we prefer to deal only with artefact.

Disposition: Withdrawn

AFNOR 22 CAT MAJOR: CLAUSE 7.4.2

Proposal:

Replace all occurrence of “step” by “task”.

Rationale:

We think that we do not need to define “step”.

Disposition: Noted but no further action. Editor to bring out as major issue.

AFNOR 23 CAT MINOR: CLAUSE 7.4.3

Proposal:

Remove line 35 to 37

Rationale:

These structuring rules should belong to 7.8

Disposition: Withdrawn

AFNOR 24 CAT MAJOR: CLAUSE 7.4.3.1

Proposal:

We prefer:

7.4.3.1 Interface Roles

7.4.3.2 Community Structures

Rationale:

We use the interface roles in the “community structures”.

Disposition: Accepted - covered by re-write

AFNOR 25 CAT MAJOR: CLAUSE 7.4.3.1

Proposal:

Objects of a community may interact with objects outside of that community. If these interactions are identified by roles. Then:

- Either, the objects that interact must belong to a community. Then, the roles that identify these interactions must belong to that community.
- *or*, the objects outside of that community interact with a CEO that represents *some of the objects* that community. Then, the roles that identify these interactions and that are fulfilled by the objects of that community are interfaces of the CEO.

In the first case, this interaction is reduced to interaction of objects inside a community. The interactions between these objects are identified by the roles of this community. In the second case, we consider that the CEO may be a member of a larger community. Then:

- The CEO fulfils roles in the larger community
- The other objects of the larger community interact with the CEO through its interfaces.

These structuring rules prescribes the relation between the community and the larger community:

- The interfaces of the CEO in the larger community *correspond to* the interface roles of the community
- The roles that the CEO fulfils in the larger community identify only actions that are performed by objects of the core community
- Actions identified by other roles of the larger community that are interactions with the CEO are also identified by environment roles of the community.

Rationale:

When an object in a community interacts with an object outside of that community, either these objects belong to a community either the object outside the community interacts with a CEO that represents that community. In the first case, the roles of this “outer” community identify these interactions. In the second case, these interactions belong to the interface of the CEO. The CEO is an object in a larger community. Its roles identify only actions that are identified in the roles of the core community. In the larger community, interactions between the CEO and other objects are also identified by environment roles of the community.

Disposition: Accepted - covered by re-write

AFNOR 26 CAT MINOR: CLAUSE 7.8

Proposal:

Remove line 10 (see annotation)

Rationale:

We think that roles have no objectives, they may express an objective.

Disposition: Subsumed in resolution of other comments

AFNOR 27 CAT MAJOR: CLAUSE 7.8

Proposal:

Replace line 11 to 15 by:

The starting point is the objectives of objects. The participation of an object in a community enables the object to meet an objective that would not be possible if the object acts individually. This impossibility could be due to physical or temporal constraints, limitations in the capability of the object, or lack of resources.

Rationale:

We do not think that objective of a community is a shared sub-objective of the objects of the community.

Disposition: Subsumed in resolution of other comments

AFNOR 28 CAT MAJOR: CLAUSE 7.8

Proposal:

Replace line 18 to 20 by:

Objects filling roles must take on the objectives expressed by those roles. Selecting an object to fill a role will typically include some assessment of the conflict between the objectives expressed by the roles and the objectives of the object ...

Rationale:

We think that a role has no objectives.

Disposition: Subsumed in resolution of other comments

AFNOR 29 CAT MAJOR: CLAUSE 7.8.1

Proposal:

Replace line 38 to 40 by:

A community is established by a contract, an agreement between parties (enterprise objects). The objective of the community will be consistent with the objectives of all of the parties of that community contract.

Rationale:

We think that the objective of a community is more than a subset of the objectives of all of the parties of the community.

Disposition: Subsumed in resolution of other comments

AFNOR 30 CAT MAJOR: CLAUSE 10

Proposal:

Remove this clause

Rationale:

Not well defined.

Disposition: Noted See AFNOR 1 and AFNOR 2.

AFNOR 31 CAT MAJOR: ANNEXE A

Proposal:

Remove the entity Step.

Rationale:

We think that we do not need the definition of step

AFNOR 32 CAT MINOR: ANNEXE A

Proposal:

Add this note:

We use the UML notation in this model.

Disposition: Deferred to next ballot cycle

Comments from Japan

JPN001

The description of TEMPORARY NOTE is important, and it should be rewritten as NOTE in the following:

NOTE - In actual specifications, it is important to clarify a distinction between prescriptive postconditions for actions and those postconditions which include preferences about particular outcome of an action. The former are not objectives and the latter are. The details of the clarification may depend on actual specifications.

Disposition: Accepted

JPN-002TL:

"Resource", "Party" and "Owner" should be in Role concepts because these represent some characteristics of behaviours of objects and because the characteristics should be identified as a role.

"Service" should be in Basic concepts because it is elementary for ODP systems' design.

Disposition: Yes for Resource and Party but not Owner

JPN-003TL:

Because "Contracting party" is essential in Enterprise Viewpoint, "Party" should mean "Contracting party" and not "Natural party". When referring to "Natural party", the full terms should be used.

Disposition: Agreed

JPN-004TL:

The second alternative, 6.2.4.2, is better because an owner should be a party, that is, a contracting party (see JPN-004TL), which is subject to some contracts.

Disposition: Accepted

JPN-005TL:

The following definition is proposed: A collection of behaviours of objects in that community which are subject to contracts of that community.

Disposition: Accepted in new structuring rule

Disposition: Rejected

JPN-006TL:

The following definitions are proposed:

Core role (in a community): A role to be fulfilled by an object that performs actions for the objective of the community.

Support role: A role to be fulfilled by an object that supports actions of objects in a core role of a community. An object in a support role may and may not be in that community, and may also be in an environment role or in an interface role.

Environment role (with respect to a community): A role to be fulfilled by an object that interacts with objects of the community but is not part of the community.

Interface role (of a community): A role to be fulfilled by an object that interacts with objects outside the community. An object in an interface role is in the community.

Disposition: Rejected – no other NB support

JPN-007TL:

The current text is acceptable.

Disposition: Accepted

JPN-008TL:

The description of "Editor's note" is true, and it should be inserted to the main text as presented.

Disposition: Make a Note, and say “as stated in Pt2”

JPN-009TL:

"Role" is an identifier according to RM-ODP Part 2, and is also a name. It is strange to create and destroy a name. It is better that "Role may be validated and invalidated".

Disposition: Accepted - covered by re-write

JPN-010TL: p.18, line 29, 7.4.3.2 Interface Roles

The abbreviation CEO is unclear. Its full spelling should be given.

Disposition: Accepted

JPN-011TL: p.19, line 61, TEMPORARY NOTE (is this in the right place? No line 61)

The text after the TEMPORARY NOTE is preferred because the terms in the text are ones defined in RM-ODP Part2. The consistency between this standard and RM-ODP Part2 should be kept.

Disposition: Accepted - Re-write will take account

JPN-0012TL: p.22, line 38, 7.5.4 Nesting of policy frameworks

The text in 8 may be used as a candidate draft of "policy framework". Therefore,

- Move the text in 8 to 7.5.4 and remove the current clause 8.
- Change the title of 7.5.4 to "Policy Framework".
- Insert a new subclause 7.5.4.1 entitled by "Nesting of policy framework" with the text in "Text parking lot".

Disposition: 1, 2 Subsumed in resolution of other comments, 3 Rejected and leave temp note. Delete parking lot

JPN-0013TL: p.29, 10. Consistency rules

Clause 10 should exist for "the" design instruction of ODP systems.

Even if it is difficult to get a consensus for correctness of the content of clause 10 among NBs, clause 10 should be left as a guide for ODP systems' design in an informative annex.

Disposition: Accepted

JPN-0014TL: p.31, Figure 10-1, 10-2 and 10-3

Roles defined in 6.4 (and possibly 6.2) should be added in the figures.

If modification of the figures is difficult, it should be noted, at least, that there may be other roles than ones described in the figures and that mapping of these roles into other viewpoints may depend on details of each ODP system.

Disposition: Accepted

JPN-0015TL: p.32, Figure 10-2 and its explanations

There is no coincidence between "E-CX" in the figure and its explanation. Numbering in the figure should be "E-C1", "E-C2", "E-C3", "E-C4" and "E-C5" from the top to the bottom.

The current explanation of "E-C4" is the same as that of "E-C5" and is redundant. It should be replaced with the following:

E-C4 A computational interface should conform to the type of an enterprise interaction.

Disposition: Accepted

JPN-0016TL: p.33, Figure 10-3

The important staffs in Engineering viewpoint such as "stub", "binder" and "protocol" should be added.

The proposal is the following:

- Draw a line with a label "E-N6" from "enterprise interaction" and "policy" to "stub", "binder" and "protocol".
- Add an explanation of "E-N6" as follows:

E-N6 An enterprise interaction with a policy should provide some constraints for behaviours of an engineering interaction based on "stub", "binder" and "protocol".

Disposition: Accepted

JPN-0017TL: p.35, Annex A

The relationship between figure A-1 and figure A-2, that is, the usage of the two notations for representation of the normative part of this standard, is unclear. If the definite relationship is not given, the possible inconsistency may cause a confusion to users of this standard.

Annex A should be informative at least. And, if the definite usage of these notations is not given, this annex should be removed.

Disposition: Deferred to next ballot cycle

JPN-0018TL: p.39, Annex B

The necessity of this annex is doubtful because the description is given in the normative part of this standard and is redundant. This annex should be removed.

Disposition: Accepted

JPN-019E: p.4, line 56, 3.1.2 Viewpoint language definitions

"ITU-T Recommendation X.902 | ISO/IEC 10746-2" should be "ITU-T Recommendation X.903 | ISO/IEC 10746-3".

Disposition: Editor to implement

JPN-020E: p.16, line 22, 7.4.2 Roles and processes

The sentence in line 22 is the same as that in line 12, and is redundant. It should be removed.

Disposition: Editor to implement

Comments from Norway

NOR-1 TL Page 12 Section 7.1

Rationale

In an enterprise specification, an ODP system and the environment it operates are normally represented as a community. However, there exist situations where other modelling concepts (e.g. domain) are more useful (see [1] for a discussion on this), but 7.1 excludes the use of any other modelling concept when representing an ODP system and the environment it operates. Even if an outer-community can be identified, it is not always useful to identify and represent this.

Recommendation

Moderate 7.1 to open up to the use of alternative modelling concepts (such as domain). This can be done by adding the sentences: "Even if an ODP system and the environment it operates always can be represented by a community at some level of description, an enterprise specification may use other modelling at the outer-most level. For instance, domain can be used to represent a configuration of objects that do not share a common objective, but are nevertheless restricted by policies within that domain (such as a jurisdiction)."

[1] Agedal, J. Ø., Milosevic, Z., "ODP Enterprise Language: UML Perspective", EDOC '99.

Disposition: Accepted in principle - covered by re-write

NOR-2 TH Page 16 Section 7.4.3

Rationale

The role rules do not reflect the fact that if a role is fulfilled, it is fulfilled by exactly one object at any one time. Role is an identity-preserving abstraction of object, and can as such not be associated with several objects at a time (unless they are grouped in a configuration object, of course). Line 26 reflects this, but line 28 does not. Also, line 29-30 can be read as there may be several objects fulfilling a role at a specific point in time, but by being specific on this in line 28, this interpretation is eliminated.

Recommendation

Delete the last part of sentence on line 28 (from comma and out). Alternatively, one can change to “or by a configuration of objects at the same time” after the comma, (then the configuration is actually the object that fulfils the role).

Agreed in principle - Covered by re-write

Comments from UK

UK-1 Cat G Whole document

Proposal:

Use the form “enterprise object” throughout.

Rationale:

This is consistent with Part 3 usage which always uses the term “object” qualified by the viewpoint name in the specific viewpoint clauses, and avoids possible misunderstandings about the scope of modelling statements.

Disposition: Disposition: Accepted

UK-2 Cat: E Clause: 2.2 and 2.3

Rationale:

There are no references either to Paired ITU-T Recommendations | International Standards or to other International Standards.

Proposal:

Delete these clauses.

Disposition: Editor to implement

UK-3 Cat: E Clause: 3.1.2

Rationale:

- a) The reference is wrong.
- b) The referenced document does not contain definitions for “community template” and “enterprise object” and definitions are not required.

Proposal:

- a) Change “ITU-T Recommendation X.902 | ISO/IEC 10746-2” to “ITU-T Recommendation X.903 | ISO/IEC 10746-3”.
- b) Delete references to “community template” and “enterprise object”.

UK-4 Cat: E Clause: 3.2 [see also UK-18]

Rationale:

The definition of “policy” is neither refined nor extended by clause 6.5.1.

Proposal:

Delete the clause.

Disposition: Withdrawn

UK-6 Cat: TH Clause: 5 Whole clause

Rationale:

a) The text of the Temporary Note does not apply to this text which matches text submitted by the UK to the Curitiba meeting.

The text matches that agreed at the Brisbane and Washington meetings, in particular:

it more clearly states the motivation of the standard;

it removes the use of the word “enterprise” used as a noun, as was agreed at Brisbane.

b) Page 5, lines 26 and 27. The phrase “which will apply across a variety of notations and modelling methods” should be amplified.

Proposal

a) Retain the existing text.

b) Delete the phrase on page 5, lines 26 and 27 “which will apply across a variety of notations and modelling methods”, and add a new paragraph:

There are many modelling methods and approaches used for understanding, agreeing and specifying systems in the context of the businesses of which they form a part. Many of these approaches fall into the categories often referred to as analysis or requirements specification. They can provide useful insights into both the business under consideration and the requirements for systems to support it, however many lack the rigour, consistency and completeness needed for formal specification. It is a key objective of this Recommendation | International Standard to provide a way of relating the commonly used concepts and underlying principles of such methods to the modelling framework of the RM-ODP.

Disposition: Accepted

UK-7 Cat E Clause: 6 Whole clause

Rationale:

The grouping of concepts with associated subheadings helps understanding, however some of the headings are superfluous and confusing.

Proposal:

Accept the grouping of concepts with the following changes:

change subheading 6.1 to “General Concepts”;

delete subheading 6.2 (“As yet unclassified concepts”) – the associated concepts become part of the “General Concepts” grouping;

delete subheading 6.4 (“Role concepts”) – the associated concepts become part of the “Behaviour Concepts” grouping.

Disposition: Accepted

UK-8 Cat: TL Clause: 6.1.1 Objective (of a <Y>)

Proposal:

Replace "...towards the preferred states." by "...towards the achievement of the preferred states."

Rationale:

Clarification.

Disposition: Accepted

UK-9 Cat: E Clause: 6.1.3 Scope (of a system)

Proposal:

Replace "The behaviour named by the set of roles..." by "The behaviour identified by the set of roles..."

Rationale:

Consistency with Part 2 definition of "role".

Disposition: Accepted

UK-10 Cat: TL Clause: 6.2.3 Party

Rationale:

"Contracting party" appears to be a special case of "natural party"

Proposal:

Identify "contracting party" as a special case of "natural party".

Disposition: Accepted - Editor to implement

UK-11 Cat: TL Clause: 6.2.4 Owner

Rationale:

"Owner (of a system)" seems to be a special case of "owner".

Proposal:

Identifier "owner (of a system)" as a special case of "owner".

Disposition: Make 2nd definition owner of an <x> and use second definition. Make note state that it's an enterprise object.

UK-12 Cat: TL Clause: 6.2.4.1 Owner

Rationale:

Editor's Note

Proposal:

Replace "the default controller of an object" with "the default controlling object of an enterprise object."

Disposition: Disposition: Subsumed in resolution of other comments

UK-13 Cat: TH Clause: 6.3.1 Behaviour (of a community)

Rationale:

The definition follows directly from Part 2 concepts.

Proposal:

Behaviour (of a community): the behaviour of a community viewed as a composite object.

NOTE – This definition of behaviour covers both interactions and internal actions of the community.

Disposition: Subsumed in resolution of other comments

UK-14 Cat: E Clause: 6.4.4 Artefact role

Rationale:

The NOTE duplicates the NOTE on 6.4.2.

Proposal:

Delete NOTE.

Disposition: Accepted

UK-15 Cat: TH Clause: 6.4.1-4

Rationale:

Taking into account the statements about resource in clause 7.4.1 (Objects and actions) and UK-31 there is a need to introduce a definition of resource role similar to that for artefact.

Proposal:

Add a clause 6.4.5:

6.4.5 Resource role: A role in which the object fulfilling the role is involved in any actions of the role as a resource.

Disposition: Editor to implement as re-worded

UK-16 Cat: TH Clause: 6.4.5.1 Objects defined in terms of roles

Rationale:

The defining of objects in terms of roles is neither desirable nor necessary.

There should, however, be a rule about the naming of objects in terms of their roles e.g. an enterprise object in a manager role may be referred to as a manager..

Proposal:

Delete definitions and add to clause 7.4.3 (Role rules) a statement of the form:

Where the context makes the usage clear, an enterprise object in a community specification can be referred to by the name of the role that it is fulfilling.

Disposition: Accepted

UK-17 Cat: TH Clause: 6.4.5.2 Roles defined

Rationale:

Page 8, line 29 to page 9, line 21. Since these are definitions of roles:

a) they should be formulated in terms of the behaviour identified by the role, not in terms of the objects that fulfil them;

b) they should be consistent with relevant text on roles in clause 7.4.2 (e.g. Page 15, line 43: “Role behaviour decomposes the behaviour of the community into roles...”).

Proposal:

Replace page 8, line 29 to page 9, line 21 with:

6.4.5.2.1 Core role (with respect to a community): A role identifying part of the behaviour of a community viewed as a configuration of objects.

NOTE – Each enterprise object in the configuration fulfils a core role.

6.4.5.2.2 Environment role (with respect to a community): A role identifying all or part of the behaviour of the environment of the community when that community is viewed as a composite object.

6.4.5.2.3 Interface role: A core role identifying behaviour that involves at least one interaction that is also part of behaviour identified by an environment role.

Disposition: Subsumed in resolution of other comments

UK-18 Cat: TL Clause: 6.5.1 Policy

Rationale:

This definition neither refines nor extends the Part 2 definition.

Proposal:

Delete the definition.

Disposition: Deferred to next ballot cycle

UK-19 Cat: TH Clause: 6.7 Force concepts

Rationale:

Without related structuring rules it is unclear how these concepts are to be applied in an enterprise specification.

Proposal:

Delete unless adequate structuring rules are introduced that clarify the application of the concepts.

Disposition: Subsumed in resolution of other comments

UK-20 Cat: TL Clause: 7 Lines 3 and 4, and line 19

Proposal:

Replace “in which it operates” by “of which it forms a part”.

Rationale:

The revised phrasing is more consistent with the fact that the system may have a significant role in the business processes.

Disposition: Accepted

UK-21 Cat: TL Clause: 7 Lines 14 and 15

Rationale:

Any activity of the system will be part of a process in which it participates.

Proposal:

Delete Line 14: “activities undertaken by the system”

Agreed – editor to provide note explaining why it’s in Part 3

UK-22 Cat: TL Clause: 7.1 Line 23

Proposal:

Insert “of an ODP system” after “An enterprise specification”.

Rationale:

Clarification.

Disposition: Editor to implement

UK-23 Cat: TL Clause: 7.1 Line 26

Proposal:

Insert “where relevant” at the start of the bullet text.

Rationale:

Clarification.

Disposition: Editor to implement

UK-24 Cat: TH Clause: 7 Lines 2-31

Rationale:

The text should be reordered and modified to avoid repetition and redundancy, and to present the concepts in a more logical order.

Proposal:

Replace the paragraphs before clause 7.1 and all of clause 7.1 itself by the following reordered text (note that this text includes the changes proposed in UK-20 to UK-23):

7.1 Overall structure of an enterprise specification

An enterprise specification for an ODP system is a model of that system and its environment that focuses on the scope and purpose of that system and the policies that apply to it in the context of the organisation of which it forms a part. The model is specified in terms of a structure of communities. A community is a configuration of enterprise objects representing a group of entities in the organisation (e.g. human beings, IT systems, resources of various kinds and groupings of these) that are subject to some implicit or explicit agreement governing their behaviour in the organisation.

Note: The term “organisation” is not limited to business organisation.

In an enterprise specification, an ODP system and the environment of which it forms a part are represented as a community. At some level of description the ODP system is represented as an enterprise object in that community. An enterprise specification of an ODP system defines the purpose, scope and policies of that ODP system in terms of:

- roles played by the system;
- processes in which the system participates;
- policy statements about the system, including those relating to environment contracts.

An enterprise specification of an ODP system includes the specifications of:

- the community in which the ODP system is represented as a single enterprise object interacting with its environment,
- any other communities of which the ODP system or its parts are members, and
- where necessary for clarity or completeness, other communities of which enterprise objects in the environment of the ODP system are members.

NOTES –

1 - Types of communities or community templates may be used in the specification of these communities.

2 - Types of communities may be related by refinement.

3 - An enterprise specification can define communities that include objects that are in the environment of the system and not part of the ODP system, when this is necessary or will clarify the specification.

This clause defines how the concepts identified in clause 3 or defined in clause 6 of this Recommendation | International Standard are used in an enterprise specification.

Disposition: Accepted - covered by re-write

UK-25 Cat: TH Clause: 7 and Annex B ODP system rules

Rationale:

The UK believes that Annex B does not contain structuring rules in the sense of statements about the use of the term “ODP system” in an enterprise specification. However, the statements are about the nature of an enterprise specification and, while some of the statements duplicate text in the introductory paragraphs of clause 7 and subclause 7.1, paragraphs 3 and 4, with the associated Note contain material the sense of which needs to be included in clause 7.

Proposal:

Insert the following text before the last paragraph of the text proposed in UK-24:

(NOTES ...)

4. The purpose of an ODP system is expressed by the objective of the community in which that system appears as a single enterprise object interacting with its environment. However, other communities may subsequently be defined in which an existing system is represented as an enterprise object, and these may have objectives that diverge from, or contradict, the original purpose of the system.

The scope of the system, where the term scope has the meaning defined in clause 6 above, will be the necessary and sufficient set of statements about the behaviour of the system such that information, computational, engineering and technology specifications can be developed.

NOTE – In order to understand that behaviour, it may be necessary to model at both more abstract and more detailed levels of description than that at which “the ODP system is represented as a single enterprise object interacting with its environment.” Thus this Recommendation | International Standard makes no prescriptions about either the most detailed or the most abstract levels of any enterprise specification, nor does it make any recommendations about the relative merits of modelling from ‘top-down’ or ‘bottom-up’. The approach taken will be a modelling choice based on the system being specified and the purpose of the modelling.

Disposition: Subsumed in resolution of other comments

UK-26 Cat: TH Clause 7.11 Scoping Statement

Rationale:

Given the nature of a Scoping Statement, as specified in clause 7.11, the need for such a statement should be identified early in clause 7.

Proposal:

Move clause 7.11 to be clause 7.2 and renumber subsequent clauses.

Disposition: Editor to implement

UK-27 Cat: TH Clause 7.2.1 Specification of a community

Rationale:

- a) The text of Page 12, lines 34-39 and Page 13, lines 1-2 cannot be easily related to the clauses that follow it.
- b) The text of Page 13, lines 3-7 states enterprise object rules rather than rules about the specification of a community.
- c) The text of Page 13, lines 8-9 is explanatory material that is more appropriate to clause 7.2.3 (Relationships between communities).

Proposal:

- a) Replace Page 12, lines 34-39 and Page 13, lines 1-2 by:

A community is defined in terms of the following elements:

- the objective of the community;
- the behaviour of the community;
- the configuration of enterprise objects that comprises the community.

The behaviour of the community is defined in terms of the following elements:

- the interactions of the community with its environment;
- the processes that take place in the community;
- the relationships between the interactions and processes;
- policies that apply to the interactions and processes.

The configuration of enterprise objects is defined in terms of the following elements:

- the roles fulfilled by enterprise objects in the community;
- the assignment of enterprise objects to roles;
- the relationship of roles to processes;
- the policies that apply to the roles and to the assignment of enterprise objects to roles.

- b) Move text of Page 13, lines 3-7 to clause 7.6, Enterprise object rules.

- c) Move text of Page 13, lines 8-9 to clause 7.2.3 (Relationships between communities). A community is defined in terms of each of the following elements:

Disposition: a) Accepted - covered by re-write - proposals b) and c) Disposition: Editor to implement

UK-28 Cat: TH Clause: 7.2.2 and 7.7 Establishment of a community; Contract

Rationale:

- a) The concepts of clause 7.2.2 and 7.7 are closely related to each other and to clause 7.2.1. The concepts discussed would be more easily understood, and related to each other and to other concepts on community, if the clauses were merged.
- b) The third paragraph of clause 7.7 (page 25, lines 27 to 34) duplicates (more or less) what is said in clause 7.1.
- c) The statements on ownership of the community in clause 7.2.2 (page 13, lines 12-13 and lines 17-18) should be covered in clause 7.5.2 (Ownership) or 7.8 (Lifecycle of a community).
- d) In paragraph 1 of clause 7.7 (page 25, lines 18-23) the term “agreement” is used in relation to entities (things being modelled), in paragraph 2 (page 25, lines 24-26) the term is used in relation to “community” (a model concept). This needs clarification.

Proposal:

- a) Merge clauses 7.2.1, 7.2.2 and 7.7. A possible merged text taking into account UK-27 and (b), (c) and (d) of the Rationale is:

7.2.2 Specification of a community

A community exists by virtue of a contract, an agreement between parties (enterprise objects). The objective of the community is a subset of the objectives of all of the parties of that community contract.

NOTES – There is no requirement that the parties fill other roles in the community. Indeed, the lifetime of a community can exceed the lifetime of the parties to the community contract.

The concept of contract, as defined in ITU-T X.903 | ISO/IEC 10746-3, provides the means to specify actual communities. For a group of entities to be modelled as a community, there must be some implicit or explicit agreement about the group covering three things:

- the objective for which the group exists;
- the structure, policies and behaviour of its members;
- the entities comprising its members.

This agreement is modelled as the contract for the community. This contract specifies the community in terms of the following elements:

- the objective of the community;
- the behaviour of the community;
- the configuration of enterprise objects that comprises the community.

The behaviour of the community is defined in terms of the following elements:

the interactions of the community with its environment (where the community is viewed as a composite object);

- the processes that take place in the community;
- the relationships between the interactions and processes;
- policies that apply to the interactions and processes.

The structure is defined by the configuration of objects that comprise the community, expressed in terms of the following elements:

- the roles fulfilled by enterprise objects in the community;
- the assignment of enterprise to roles;
- the relationship of roles to processes;

- the policies that apply to the roles and to the assignment of enterprise objects to roles.

NOTE – A family of related contracts may generated from a contract template. Some aspects of the contract template (e.g. membership) may only apply to particular contract instances, while other aspects may apply to all contract instances.

b) Add text to clause 7.5.2 (Ownership) or 7.8 (Lifecycle of a community) to cover the intent of the text from page 13, lines 12-13 and lines 17-18.

Disposition: a, b Accepted - covered by re-write; c, d Withdrawn

UK-29 Cat: TH Clause: 7.2.3 Relationships between communities

Rationale:

The revised text of clause 7.2.3:

- a) does not fully represent the sense of the text in the WD;
- b) introduces apparent duplication (e.g. page 13, line 26 and lines 38-39);
- c) introduces concepts that are not relevant to an enterprise specification and are of doubtful validity (e.g. page 14, lines 3-5 text on transivity of relationships).

Proposal:

Develop revised text based more directly on the text of the Working Draft (JTC1/SC7/WG3 3N65). A possible text taking into account UK-28 is:

An enterprise specification can specify more than one community. This occurs where a community cannot be specified in isolation but must be considered in the context of some other community or communities to which it is related.

The communities can be related in various ways, including relationships when:

- One community, viewed as a composite object, is an expression of an enterprise object fulfilling a role in a community of wider scope and in a model at a different level of abstraction;
- One community re-uses role specifications from another community.
- One community is populated with one or more enterprise objects that represent the same entities as enterprise objects in another community;

NOTES

1 The first case is where it is necessary to consider the community of interest (perhaps one in which the ODP system appears as a single object) in the context of some larger (outer) community, in which it is modelled as an enterprise object fulfilling a role. For example, if a community interacts with its environment (i.e. with objects that are not members of that community) then it does so as a part of (i.e. as an enterprise object subject to the policies of) a community of greater scope.

Alternatively, company may be defined as operating within a particular legal system, for example under English Law. This is equivalent to saying that there exists a community whose members are those subject to English Law. The laws and decision procedures can then be codified as the behaviour of this outer, English Legal community. By being defined within this context, the community representing the company inherits from the outer community a corresponding set of obligations on its members, representing the requirement that they operate in accordance with the law.

2. The second case occurs where broad communities may be defined to represent aspects of commercial activity, such as buying and selling, or even a specific framework for ownership of resources. A specific commercial undertaking or organisation may be modelled in terms of a community specification identifying its constituent members, the roles they fulfil and the internal procedures under which they operate. Some of these roles may be required to operate on behalf of this organisation as, for example, buyers for the organisation, and this part of their behaviour can be expressed by requiring that they play a buyer role within the definition of the

(outer) commercial community. Making this association opens up the ability to define "suppliers" to the organisation as the set of things having seller roles in the (outer) commercial community definition. The commercial obligations on these roles then follow directly without re-specification.

Another example occurs where an outer community requires an auditor role to police the behaviour of core communities. Depending on the size and complexity of the organisation being described, this may lead to the definition of a further core community defining the procedures of an audit office to fulfil this role.

There can also be peer-to-peer interaction between communities. For this interaction between the communities to be meaningful, there must be some element of shared objective, which itself implies a higher level of community of which both communities will be members, and a common set of policies will apply.

Disposition: Accepted in principle. Re-word of notes Accepted.

UK-30 Cat: E Clause: 7.3 Objective rules

Rationale:

- a) Page 14, lines 35-37. The relation between sub-objectives of the community and objectives of the roles, processes and policies of the community should be made clearer.
- b) Page 14, lines 37-38. The second sentence is ambiguous.
- c) Page 14, lines 37-47. This text should be restated in a clearer rule form.

Proposal:

Replace 7.3 by:

A community has exactly one objective. The behaviour of the community realises its objective.

This objective of a community can be refined into sub-objectives that apply to roles, processes and policies of the community.

The behaviour of the enterprise object fulfilling a role realises the objective of the role.

The performance of the part of the community behaviour prescribed by a process realises the objective of the process.

A policy defines how decisions on behaviour are made on a case-by-case basis where part of the behaviour of the community is not fully prescribed by the specification of roles and processes. The objective of a policy is realised by steering behaviour to meet it.

Typically, roles, processes and policies have different (sub-)objectives. However, a process can involve policy-influenced decisions that can result in the creation (or termination) of processes; hence, processes and policies can be intertwined.

An enterprise object can have objectives where the entity being modelled has objectives.

An enterprise object fulfilling a role has the objective of the role. Hence, where a community, viewed as a composite object, fulfils a role in a community of wider scope, the objective of the first community is consistent with the objective of the role.

Disposition: Accepted – in particular last 2 paras.

UK-31 Cat: TH Clause: 7.4.1 Objects and actions

Rationale:

- a) Page 15, line 4. The first statement should relate an enterprise object to a role as well as an action
- b) Page 15, 6 and 7. There should be a parallel statement about resource to those about actor and artefact..

c) Page 15, 15-16, and 19-20. There should be statements for resource parallel to those for artefact.

d) Since the clause relates to the behaviour of enterprise objects it will follow more logically after clause 7.4.3 on Role rules.

Proposal:

a) Replace the text by the following:

An enterprise object fulfils at least one role in at least one community and can be involved in actions in the following ways:

- the enterprise object can participate in carrying out the action;
- the enterprise object can be mentioned in an action;
- the enterprise object may be used by the action.

NOTE – For every action there is at least one participating enterprise object. Where two or more enterprise objects participate in an action, it is an interaction.

An enterprise object that participates in an action is said to be an "actor" with respect to that action.

An enterprise object that is referenced in an action is said to be an "artefact" with respect to that action.

An enterprise object that is used in an action is said to be a "resource" with respect to that action.

In the special case where an enterprise object is referenced in an action in which it also participates (e.g. an enterprise object reporting its state) it is both an actor and an artefact with respect to that action.

In the special case where an enterprise object is used in an action in which it also participates it is both an actor and a resource with respect to that action.

Resources can be used in an action, and the action is constrained by the availability of those resources. There can be zero or more resources used in each action.

Where a role in a community involves an enterprise object in actions only as an artefact, then it is an artefact role in that community.

Where a role in a community involves an enterprise object in any action as a resource, then it is a resource role in that community.

Where a role in a community involves an enterprise object in any action as an actor, then it is an actor role in that community.

NOTE - Therefore, roles in a community can be partitioned into actor roles, artefact roles and resource roles with respect to that community.

b) Move the clause to follow clause 7.4.3 (Role rules).

Disposition: Accepted as amended. Editor to implement editor has the words

UK-32 Cat: TH Clause: 7.4.2 Roles and processes

Rationale:

a) Page 15, lines 28 and 29. The description of the behaviour of a community should also include the interactions of the community, considered as a composite object, with its environment (see UK-27)

b) Page 15, lines 32-41. The text implies that processes and roles provide alternative but equivalent approaches to the specification of the behaviour of a community. In fact, a process-based description and a role-based description address different aspects of a community specification, namely:

(Roles) The partitioning of tasks into roles and the assignment of roles to enterprise objects.

(Processes) Interactions of the community, considered as a composite object, with its environment, and tasks and the ordering of tasks

c) Page 16, lines 1-3. The second sentence “The behaviour of a role will be a subset of the behaviour of the object that fulfils that role” is more appropriate as a role rule. The last sentence duplicates a statement in clause 7.4.3 (page 16, line 27)

d) Page 16, lines 10 and 11. There is no explanation of the circumstances in which a task will not be a step in a process. The UK believes that a task can always be related to a process.

e) Page 16, lines 12-19. The second sentence and the associated bullet items duplicate material in 7.4.4.2.

Proposal:

a) Replace page 15, lines 28 and 29 with:

The behaviour of a community defines what the community should be observed to do and consists of

- expected actions of the community, called tasks, and the possible ordering of the tasks;
- Interactions of the community, viewed as a composite object, with its environment.

Add a subclause after 7.4.4 (Process rules):

7.4.5 Interactions

The specification of a community can include the specification of interactions of the community, viewed as a composite object, with its environment. Within an enterprise description these interactions may be defined as part of the specification of a community at a higher level of abstraction in which the first community is represented as an enterprise object.

Each interaction initiates and/or terminates one or more processes.

b) Replace page 15, lines 32-41 with:

The assignment of tasks to the enterprise objects that comprise a community is defined in terms of roles. A role identifies an abstraction of the community behaviour that comprises a set of tasks that is assigned to a single enterprise object within the community. Each abstraction is labelled as a role. The emphasis is on which enterprise objects participate in the particular behaviour.

The tasks and their ordering are defined in terms of processes. A process is a collection of abstractions of the community behaviour in which each abstraction includes only those tasks that are related to achieving some particular result/purpose/sub-objective within the community. Each abstraction is labelled with a process name. The emphasis is on what the behaviour achieves.

c) Page 16, lines 1-3. Move the second sentence “The behaviour of a role will be a subset of the behaviour of the object that fulfils that role” to follow the first paragraph of clause 7.4.3 (Page 16, line 26).

Delete the last sentence.

d) Delete Page 16, lines 10 and 11.

e) Delete the second sentence and the associated bullet items of page 16, lines 12-19.

Disposition:

- a) **Unbulletise the first bullet – make the second bullet a note. 7.4.5 Disposition: Withdrawn**
- b) **Accepted**
- c) **delete both sentences**

- d) **deferred**
- e) **Accepted**

UK-33 Cat: TH Clause:7.4.3 Role rules – introductory text

Rationale:

- a) Page 16, line 26. A role is an identifier for behaviour as well as a placeholder for an object that will exhibit that behaviour.
- b) Page 16, line 28. The phrase “or by several objects at the same time” does not make sense.
- c) Page 16, line 29. The first sentence implies:
 - i) that all roles are assigned at the time the community is instantiated: this is not necessarily true;
 - ii) that more than one enterprise object can be associated with a given role at one time: this does not make sense except in the context of an enterprise object considered as a composite object, and contradicts line 26.
- d) Page 16, line 32. This text has left out significant rule related material from the Working Draft (SC7/WG3 3N65). This stated:

“The roles in a community may vary during its lifetime, since its behaviour may evolve. Roles may be created or destroyed, so that the role lifetime is contained within the community lifetime, and the period for which a particular enterprise object fulfils a given role is contained within the lifetime of that role.”

The deleted material should be restored in an appropriate rule form.
- e) Page 16, lines 37-41. This paragraph is more about the nature of the specification of a community than about the nature of roles.

Proposal:

- a) Replace Page 16, line 26 by:

In the specification of a community, each role stands as a placeholder for some enterprise object that exhibits the behaviour identified by the role.”
- b) Delete the phrase “or by several objects at the same time” in page 16, line 28.
- c) Replace the first sentence of page 16, line 28 by:

During the lifetime of the community, enterprise objects are associated with roles of the community. A role can be empty for periods of time during the lifetime of a community.
- d) Replace Page 16, line 32 with:

Roles may be created or deleted during the lifetime of a community. The role lifetime is contained within the community lifetime, and the period for which a particular enterprise object fulfils a given role is contained within the lifetime of that role.
- e) Move the text from Page 16, lines 37-41 to clause 7.2, Community rules.

Disposition: a) Accepted

b), c) and d) Accepted - covered by re-write

e) deleted, not moved

UK-34 Cat: TH Clause: 7.4.3.1 Community Structure

Rationale:

This clause is obscure and, by avoiding the terms CEO, core role and environment role, invalidates text elsewhere (e.g. clause 7.4.3.2).

It is suggested that the issues of concern can be addressed more directly by using the fact that a community specification can include interactions of a community (viewed as an enterprise object) with its environment. These are the interactions of the community when considered as enterprise object fulfilling a role in a community appearing in a model at a higher level of abstraction - which is already talked about in 7.2.

Proposal:

Replace the text of clause 7.4.3.1 with:

7.4.3.1 Environment and core roles

The specification of a community can include the specification of interactions between enterprise objects in that community and enterprise objects outside it. The configuration of enterprise objects that comprises a community may be composed into a single enterprise object equivalent to that community, referred to as the community equivalent object (CEO). The situation where enterprise objects in a community interact with enterprise objects outside it is equivalent to there being interactions between the corresponding CEO and its environment. Thus the interactions between a community and enterprise objects outside it may be modelled at two levels of detail.

At one level of detail, an enterprise specification defines the interactions between the CEO and its environment in terms of the roles of the CEO and the roles of the external enterprise object(s) with which it interacts within a community of wider scope. At another, more detailed level, an enterprise specification views the community as a configuration of enterprise objects and defines core roles and environment roles for it. Each core role identifies part of the behaviour of the configuration of enterprise objects and each environment role defines part of the behaviour of the environment of the CEO. Some core roles in the community include interactions with enterprise objects in environment roles. See Figure 1.

Such specifications of roles at different levels of abstraction must be mutually consistent. Consistency is maintained by observing the following structuring rules:

- the behaviour identified by each of the roles of the CEO (at the less detailed level of description) is identified by a set of core roles interacting with one or more enterprise objects fulfilling environment roles.(at the more detailed level of description).
- the environment roles (at the more detailed level of description) identify all aspects of the behaviour of enterprise objects in the environment of the CEO (at the less detailed level of description) so far as that behaviour concerns the interactions of those enterprise objects with the CEO.

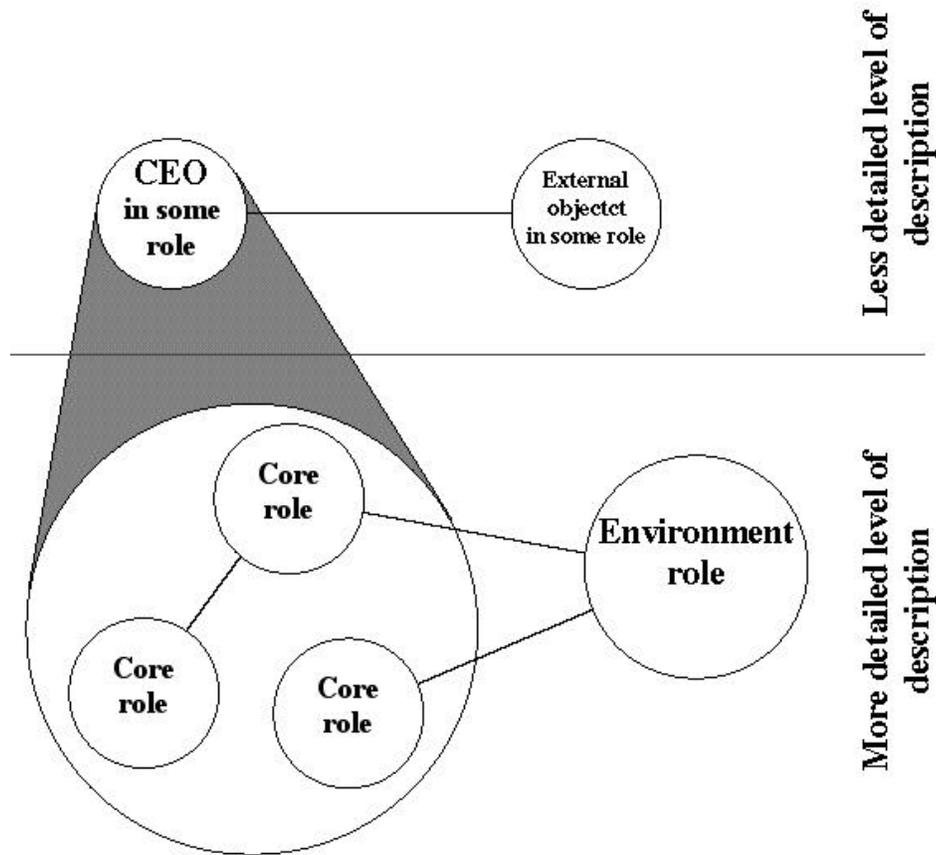


Figure 1

Disposition: Resolved by AFNOR 25

UK-35 Cat: TL Clause: 7.4.3.2 Interface roles

Rationale:

This text needs to be consistent with the text for clause 7.4.3.1. The existing text is also repetitive.

Proposal:

Taking into account UK-34 replace the existing text with:

It is sometimes necessary to consider the interactions of a community as a composite enterprise object with its environment as the provision one or more services. This can be modelled by considering each service as an interface of the CEO. Then, in specifying the community it is necessary to establish the mapping between the interfaces (services) of the CEO and the core roles of the enterprise objects in the community that are directly associated with the interactions that provide the service(s). These are interface roles. This mapping emphasises the fact that in the community specification these interactions must be modelled as being interactions involving enterprise objects of the community rather than as interactions of the CEO.

Disposition: Accepted in re-write

UK-36 Cat: TH Clause: 7.4.4 Process rules

Proposals:

Page 18, line 47 to page 19, line 22. Replace with:

A process is modelled as a directed acyclic graph of steps, in which each step:

- is made possible by state changes of one or more enterprise objects, or the completion of one or more previous steps in the process;
- results in state changes, some of which make possible the occurrence of subsequent steps in the same process.

NOTES –

1 – The use of ‘acyclic’ indicates that the trace, or history, of actions does not contain cycles of cause and effect. This does not prevent the use of notations with a concept of iteration; such looping concepts generate a sequence of distinct action occurrences.

2 – In an enterprise specification, a process is an abstraction of the behaviour of some community in which the identities of the objects fulfilling roles in the community have been hidden as a result of the abstraction.

A step in a process may itself be modelled as a (sub-)process.

Disposition: Accepted as amended

UK-37 Clause: 7.4.4 Process rules

Rationale:

- a) Page 19, line 24 Steps should be associated with the tasks of actor roles.
- b) Page 19, line 25 Steps are only associated with enterprise objects in the context of the roles that they fulfil.

Proposals:

- a) Page 19, line 24 Replace “associated with some actor role” with “associated with a task of some actor role”.
- b) Page 19, line 25 Delete “or different enterprise objects”.

Disposition:

- a) **Deferred until step/process distinction is resolved**
- b) **Disposition: Accepted**

UK-38 Cat: G Cat: TH Clauses: 7.4.2-4

Rationale:

Clause 7.4.2 implies that processes and roles provide alternative but equivalent approaches to the specification of the behaviour of a community. These two approaches address different issues in relation to behaviour [see UK-27]:

Community behaviour is defined in terms the processes that it carries out and its interactions with its environment.

Roles provide the means to describe how the behaviour is partitioned among the enterprise objects that comprise the community.

The remainder of the clause 7.4 (except 7.4.1) combines material on behaviour (7.4, Process rules, and parts of 7.4.3.1, Community structure) with material on the partitioning of behaviour in terms of roles (7.4.3, Role rules, and the remainder of 7.4.3.1).

Combining discussion of these two is confusing.

Proposal:

The proposal is too complex to insert here; it is anyway a compilation of many of the other UK comments – see full UK comments.

Disposition: Withdrawn

UK-39 Cat: TH Clause: 7.5 Policy rules

Rationale:

The text in 7.4.5 needs to be reviewed and the following issues addressed:

Material from clause 8 and the “Text parking lot” on delegation needs to be incorporated;

Note that, where there is duplication, text in clause 8 provides a better formulation.

Text should be put more uniformly in rule-like form;

The text in 7.5.3 should be kept at the current level of detail but it should be revised to deal first with obligations and then with permissions and prohibitions as special cases.

Examples in current text give examples of policy related to both population and behaviour but existing rules all relate to behaviour.

Proposal:

The UK will propose text for the Editing Meeting.

Disposition: Deferred to next ballot cycle

UK-40 Cat: TL Clause: 7.6 Enterprise object rules

Rationale:

a) Page 24, line 40. An entity is part of the organisation that is modelled by the community, not of the community itself.

b) Page 25, lines 8-11. Examples should not be part of the main text.

Proposal:

a) Page 24, line 40. Replace the first sentence by:

An enterprise object is any object in an enterprise specification that is significant to the achievement of the objective of any community in that specification.

b) Page 25, lines 7-11. Replace these two sentences by:

NOTE – Thus, in considering the trading of grain, the grain itself need not be modelled if the general stock level or capacity is relevant. However, individual batches of grain may need to be modelled as enterprise objects if the test results of individual batches are the subject of policy.

Disposition: Editor to implement – delete note

UK-41 Cat: TH Clause 7.9 Force rules

Proposal:

This clause should be deleted if no text is provided.

Disposition: Subsumed in resolution of other comments

UK-42 Cat: E Clause 7.10 Common community types

Rationale:

This clause is out of place.

Proposal:

Move the clause to the end of clause 7.2.

Disposition: Editor to implement

UK-43 Cat: E Clause: 8

Proposal:

Merge with material in clause 7 (seeUK-39)

Disposition: Accepted

UK-44 Cat: TL Clause: 9

Rationale:

a) Page 29, line 14. “Reference points” are not discussed in the text.

b) Page 29, line 15 The term “conformance” is used in standards for conditions for which tests can be applied. The term “compliance” is the appropriate term to be used for consistency of a specification with a standard (see RM-ODP Part 2, clause 15.1).

c) Page 29, line 25. The second reference in page.

Proposal:

a) Page 29, line 14. Change title to “Compliance”.

b) Page 29, line 15. Replace “conformant with” by “compliant with”.

c) Page 29, line 25. Replace “or ITU-T Rec. X.902 | ISO/IEC 10746-2” with “or ITU-T Rec. X.903 | ISO/IEC 10746-3”

Disposition: Accepted

UK-45 Cat: E Clause: 10 Consistency rules

Proposal:

This clause should be revised to align with the remainder of the text when that text has been agreed

Disposition: Noted

UK-46 Cat: TH Annex A

Rationale:

The UK considers that this annex provides important tutorial material that should remain in the standard, however:

the diagram and the EBN text need review for consistency with the body of the text;

there is a need to provide better explanatory text;

the Annex should be referenced from the body of the text.

Proposal:

The UK will provide draft text for the Editing Meeting. This should be revised to align with the output text.

Disposition: Deferred to next ballot cycle

Comments from USA

US-1.

Location: General

Qualifier: M

Rationale: Throughout the document, it becomes clear that some constructs have types (and templates), and some constructs may be composed. Part 2 explicitly mentions types (and templates) for objects, interfaces and actions. In addition, types are needed (and this is sometimes mentioned in the enterprise language) for roles, communities, relationships and processes. Similarly, Part 2 mentions compositions of objects and of behaviours; it becomes clear that compositions are needed for processes, communities and relationships. There is a need to describe these additions in one Clause, especially taking into account that these considerations are not specific to the enterprise language and may be used for other viewpoint specifications. [Note that, in accordance with Part 3-4.2.2, these constructs may be used in other viewpoint specifications even if they are concepts of the enterprise language.]

Proposal: Include a Clause about extending the use of the Part 2 concepts, type and composition.

Disposition: Deferred to next ballot cycle

US-2.

Location: General

Qualifier: m

Rationale: It is not clear whether a definition in terms of actions is preferable to a definition in terms of roles, for example. A collection of roles, together with the invariants that define relationships between roles, already determine substantial properties of actions that may happen. These invariants define properties that may be more stable than properties of actions. Since at the same time the invariants that define roles and relationships between roles may be deduced from the properties of actions, it may be a modeler's choice which approach to modeling should be preferred. The standard should not impose a specific modeling approach.

Disposition: Accepted

US-3.

Noted

Location: General

Qualifier: m

Rationale: Consider the duality of state and behaviour. A customer is a role within a community described in the final note to 7.4.3.1. This role may be defined, on the one hand, by the specification of a collection of actions in which the customer participates, with the set of constraints (within the context of the said community), and on the other hand, by the specification of the properties of the collective state of the community in which this role participates (for example, using invariants). The specifier makes the choice. In many situations both approaches will be used.

Similarly, we don't need to restrict such examples to "customers": the same considerations apply to a posting ledger system within the community of an accounting system and related objects. Here an object that models an existing legacy system may fulfill the role of a posting ledger system, or by an

object that models a system-to-be-built, or even by a collection of human accountants that satisfies the same specification.

US-4.

Location: 0.2, third paragraph

Qualifier: m

Rationale: RM-ODP may be used for different purposes, as stated in Part 2. More specifically, it does not have to be used for specifying an IT system. Therefore the statement about object-based application architectures is too restrictive and may be misleading.

Proposal: Delete the fragment “,so as to enable...application architectures”.

Disposition: Accepted

US-5.

Location: 3.1.2

Qualifier: e

Rationale: The terms in this Clause are taken from Part 3 rather than Part 2.

Proposal: Replace “X.902” with “X.903” and “10746-2” with “10746-3”.

Disposition: Editor to implement

US-6.

Location: 5. p. 5 ll. 17 ff.

Qualifier: M, E

Rationale: The presentation ought to be expressed in a more precise manner. Some of the fragments of the second and third paragraphs are semantically very close. The enterprise language provides not only the vocabulary but, more importantly, the concepts. The distinction between stakeholders and users is not made clear (and may not be needed in this context). The size of organization (“very large”, “more limited”, or “much smaller” need not be related to the complexity of the specification of this organization and of the system within this organization. The duality of state and behaviour emphasized in Part 2 (e.g., in 2-8.1) is not used. There is no mention of the configuration of the system. Finally, there may be different communities within which the ODP system may be considered, and therefore the concept of “within its community” is unclear. It will be clearer to make and consistently use the distinction between organization – a construct from the universe of discourse (UoD), and community – a construct from the model.

Proposal: Merge the second and third paragraphs. Add “concepts” to the first sentence of the second paragraph. Explicitly mention state and duality of state and behaviour instead of mentioning just the behaviour of the system. Explicitly mention the configuration of the system. Delete “social or business” as attributes of organization (in the second and fourth paragraphs) and instead mention in a note that an organization is not limited to a social or business organization. Move the sentence “Such an organization may be very large...” to a note and mention the specification complexity, which is independent of the size of the organization. In the second paragraph, replace “represent” with “specify.” Replace “by other entities within its community” with “by the entities with which it interacts” In the third paragraph replace “within its community” with “within its communities”. Replace “what the system is expected to do” with “what the system is expected to be”.

Consider replacing “organization” throughout with “community”.

Disposition: Deferred to next ballot cycle

US-7.

Qualifier: M, E

Rationale: The fourth paragraph is not normative.

Proposal: Remove the fourth paragraph or move it to an annex. If not, then replace “enterprise” in the last sentence of the fourth paragraph with “organization”.

Disposition: Rejected

US-8.

Location: 6.1.2

Qualifier: M

Rationale: Purpose and objective are synonymous (or very nearly so).

Proposal: Consider removing this definition.

Disposition: Deferred to next ballot cycle

US-9.

Location: 6.1.2

Qualifier: e

Rationale: The choice of ‘<Y>’ is not explained.

Proposal: Change ‘<Y>’ to ‘<X>’ as is normally the case, or explain in a note why ‘Y’ is used instead of ‘X’.

Disposition: Accepted

US-10.

Location: 6.1.3

Qualifier: m

Rationale: Since a system can fulfil different roles in different communities (and these roles may even be mutually inconsistent), the definition of the scope of the system ought to mention this explicitly, at least in a Note.

Proposal: Add a note about the scope of the system being a composition of the behaviours named by the roles the system can fulfil in the contexts in different communities.

Disposition: Withdrawn because covered in structuring rules

US-11. .

Location: 6.1.4

Qualifier: m

Rationale: As described in 7.11, the enterprise specification may be used in some situation that is defined, generally, by its invariants. Therefore the “preconditions” in 6.1.4 may just refer to satisfying the invariants defined by the context in which an enterprise specification may be used. It is much more interesting to refer in the scoping statement to these invariants directly.

Proposal: Replace “preconditions” with “invariants”.

Disposition: Rejected

US-12.

Location: 6.2.1 p. 6 l. 36

Qualifier: m

Rationale: A resource need not be an enterprise object. (It might be, for example, an engineering object. That's because resource is a Part 2 concept.)

Proposal: Delete the word 'enterprise.' Consider putting this term in a separate subclause near the beginning of the clause.

Disposition: Withdrawn

US-13. .

Location: 6.2.2

Qualifier: m

Rationale: Service is described in some detail in Part 2. The definition of a service, as provided here, does not seem to be needed and seems to contradict with the descriptions provided in Part 2-8.1.

Proposal: Delete 6.2.2.

Disposition: Accepted

US-14.

Location: 6.2.2

Qualifier: M

Rationale: If the clause is kept, this expression of the meaning is inconsistent with Part 2: " an object is informally said to perform functions and offer services (an object which makes a function available is said to offer a service). For modelling purposes, these functions and services are specified in terms of the behaviour of the object and of its interfaces (see 8.4)".

Proposal: Reword to eliminate operational interface types (client/server). Consider putting this term in a separate subclause at the beginning of the clause.

Disposition: Subsumed in resolution of other comments

US-15.

Location: 6.2.3 p. 7 ll. 5-9

Qualifier: M

Rationale: "Natural" is awkward at best, as the concept names models of both natural and artificial persons. If the concept of 6.2.3.1 is named 'party,' this standard will be consistent with the OMG work on party.

Proposal: Use 'party' for 6.2.3.1 Consider for 6.2.3.2 these possibilities from Roget: participant, assenter.

Disposition: Accepted

US-16. 6.2.3.1

Qualifier: m

Rationale: In accordance with Part 2, an object is a model of an entity. Therefore a "natural" party, if the definition will be used, is an object modeling a natural person, etc.

Proposal: Replace “representing” with “modeling”.

Disposition: Accepted

US-17.

Location: 6.2.4.1

Qualifier: m

Rationale: The terminology used is unclear. Specifically, “default” and “exploit the object” do not have a clear meaning. It may be the case that the concept of an owner ought to be subtyped into appropriate (possibly overlapping) subtypes along the lines suggested in the Note to {2}. Some roles include owner, that pays for the system; steward that manages the system and is responsible for it; user that uses the system; stakeholder that has a vested interest in the system

Proposal: Use the term ‘owner’ with the second definition. Consider defining various subtypes of the owner distinguished by their control over various aspects of the system specification, instantiation, and use. Consider the term ‘controller’ for the first definition.

Disposition: Disposition: Subsumed in resolution of other comments

US-18.

Location: 6.3.1 p. 71. 32

Qualifier: m

Rationale: Perhaps we can simply take the definition of ‘behaviour’ and change it as needed to suit behaviour (of a community). The objection has been raised that, given the rules for use of T (of a Q), it does not make sense to have a term, plain T, and another term, T (of a Q), in the same vocabulary.

Proposal: Consider removing this concept.

Disposition: Accepted

US-19.

Location: 6.3.1 p. 71. 32

Qualifier: m

Rationale: Behaviour is generally a collection of actions with a set of constraints (Part 2). Behaviour of a community may be defined in a similar manner, with a reference to the objects participating in the community. In other words, the behaviour of the community is the collective behaviour of the objects in the community. At the same time, it may make sense to consider the collective behaviour of some subsets of the community. Therefore it is not clear why a specific situation in collective behaviour (of the whole community) is defined while other such situations are not.

Proposal: Consider the definition of collective behaviour in general and replace “Behaviour of the community” with “collective behaviour” in the definition.

Disposition: Disposition: Subsumed in resolution of other comments

US-20.

Location: 6.3.2 p. 7 ll. 34-35

Qualifier: M

Rationale: The definition does not consider the use of role. In the enterprise language the concept should be used as it is defined in Part 2. Note the objection that has been raised at earlier meetings

that, given the rules for the use of terms of the form T (of a Q), it does not make sense to have two terms in the same language, one T and the other T (of an Q).

Proposal: Consider deleting the term. If the term is kept, change the definition to: Identifier for a behaviour, which may appear as a parameter in a template for a community, and which is associated with one of the objects of the community."

Disposition: Accepted

US-21.

Location: 6.3.3

Qualifier: e

Rationale: The third note may be ambiguous.

Proposal: Change to "A specification may define types of processes, and may define process templates."

Disposition: Accepted

US-22.

Location: 6.3.3

Qualifier: e

Rationale: This concept may be useful in other viewpoint languages.

Proposal: Consider putting this term in a separate subclause near the beginning of the clause.

Disposition: Rejected

US-23.

Location: 6.3.4 p. 7 l. 44

Qualifier: m

Rationale: Do we need a term for this? What is wrong with action? On the other hand, this definition fits well with the ordinary meaning of task, and may cater to those who want to do process modeling.

Proposal: If the term is kept, do the research needed to find out what terms are used to specify process in other standards, for example OMG workflow.

Disposition: Deferred to next ballot cycle

US-24.

Location: 6.3.4

Qualifier: m

Rationale: If the concept of a task is used to distinguish between actions associated with objects and tasks associated with roles then it is useful to make this distinction explicit.

Proposal: Include a Note about this distinction, if this is the intended use.

Disposition: Deferred to next ballot cycle

US-25.

Location: 6.3.5

Qualifier: M

Rationale: This definition limits the use of the term 'step' to processes in communities.

Proposal: Define step as an action in a process. Consider putting this term in a separate subclause near the beginning of the clause.

Disposition: Deferred to next ballot cycle

US-26.

Location: 6.4.5.2

Qualifier: m

Rationale: These are (in accordance with the Part 2 definition of type) different context-dependent types of roles. If such definitions are included then the context -- community -- ought to be made explicit.

Proposal: Refer explicitly to the community in these definitions.

Disposition: Deferred to next ballot cycle

US-28. 6.4.5.2

Qualifier: m

Rationale: It is not clear whether these definitions ought to be included. If they are to be included then, at least, some definitions are unclear and should not therefore be used. For example, the concept of "central to the objective" is unclear. The concept of an object agreeing to a contract is anthropomorphic. In the second alternative, it is not made clear why a composite object should fulfill the core role.

Proposal: Do not accept the third and fourth alternatives.

{1.1} and {1.3}

Rationale: This says that interface roles are core roles. It's ok, but we should be well aware of, and all agreed to that, if this first alternative is chosen.

{1.1}

Rationale: Any enterprise object in the community can be a composite object.

Proposal: Do not accept this definition.

{1.2}

Rationale: "Outside of" would need definition.

Proposal: Do not accept this definition as worded. [See comment to 6.4.5.2.3.2 below.]

{2.1}

Rationale: This says that there could be several such composite objects in the same community. That is fine, as the proponents of the concept, core role, often have several core roles in examples. It also means that an object that is not decomposed can not fill a core role.

{2.3}

Proposal: Consider "A role to be fulfilled by an object that interacts with objects in the environment of the community."

{3.1}

Rationale: Central does not have a clear meaning. What if there are no distinguished "central" objects?

{3.2} This definition aligns better with that of 'environment'.

{4.1}

Rationale: Same as {4.1}

{4.2}

Rationale: "Not central to" has the same problem.

{5.1}

Rationale: This is based on the concept that object 'agree.' A different meaning of agree is used elsewhere.

R: If objects agree, then every object in the community has agreed to the community contract, which does not distinguish 'core'.

{5.2}

Rationale: For an object to not agree, the contract must have first been offered, and this does not appear to be the intent of the definition.

Disposition: All removed except Interface left as placeholder only.

US-27. 6.4.5.2.1, Editor's note, second paragraph

Qualifier: m

Rationale: Not all enterprise objects may be decomposed into configurations represented as communities, etc. This happens only for those enterprise objects for which the specifier decides to do so.

Proposal: Replace "Any" with "An".

Disposition: Subsumed in resolution of other comments

US-29.

Location: 6.5.3

Qualifier: m

Rationale-1: It is not clear whether a violation is an action or a state. If violation is an action then it may be unimportant, when using the concept of a failure, which specific action led to the failure. In some situations it may be impossible to determine such an action: although a rule may be violated, it may not be clear what specific action led to such a violation. It may be of interest to distinguish between, and provide terminology for, the situation when it is possible or desirable to determine such an action and the situation when it is much more important to rectify the violation.

Proposal: Distinguish explicitly between the two situations above.

Rationale-2: It seems that if a rule says an action may not occur (or must occur), then a failure is an action (or the absence of an action). If a rule says that a state must not happen (or must happen) then a failure is a state. So at most we need a note to that effect.

Disposition: Deferred to next ballot cycle

US-30.

Location: 6.6

Qualifier: e

Rationale: The clause needs a brief introduction.

Proposal: Add “These concepts may be used delegation in the universe of discourse, including delegation to the ODP system.”

Disposition: Deferred to next ballot cycle

US-31.

Location: 6.6.2

Qualifier: e

Rationale: The definition has editorial errors.

Proposal: Change to:

6.6.2 Agent: An object ~~which~~ that has been delegated (authority, a function, etc.) by and acts for another (in exercising the authority~~sation~~, performing the function, etc.).

Disposition: Editor to implement

US-32.

Location: 6.6.2

Qualifier: e

Rationale: The first note is incomplete, and therefore wrong.

Proposal: Change to:

1) An agent may be a party or may be the ODP system or one or its components. Another system in the environment of the ODP system may also be an agent.

Disposition: Editor to implement

US-33.

Location: 6.6.3

Qualifier: m

Rationale: Since delegation may be transitive, the concept of a principal may be context-dependent. Is this the intent?

Proposal: If the concept of a principal is context-dependent it is useful to say so..

Disposition: Deferred to next ballot cycle

US-34.

Location: 6.7

Qualifier: M

Rationale: It appears that some sentences in this Clause describe the universe of discourse while other sentences describe its model. Specifically, for example, an act is defined in terms of the model (so that there are no acts in the universe of discourse), while a declaration is defined as if it were an act in the

universe of discourse. The definition of an instruction as an act is also unclear, for the same reasons (there are no persons or machines in the model; and it is possible to instruct a legal entity, not just a person). It is not clear what can be said about the definition of prescription in this context.

Proposal: Distinguish between the universe of discourse, its state of affairs, etc., and their models, in an explicit and consistent manner.

Disposition: Deferred to next ballot cycle

US-37.

Location: 6.7

Qualifier: e, m

Rationale: The clause needs a brief introduction. Some of the definitions have editorial errors. Some of the definitions can be clarified by rewriting.

Proposal: Make the following changes. [The changes shown also include the changes mentioned in the preceding items.]

6.7 Force concepts

These concepts may be used to model changes in that part of the universe of discourse modeled by the environment of the ODP system, including such changes when caused by changes in the ODP system itself.

6.7.1 Act: An action in which ~~the initiating~~ one or more objects is a party or agent and which action is a model of something done by an entity modelled by a party or its agent.

6.7.2 Commitment: An act by which ~~an object~~ the entity represented by a party is bound by a rule or by a contract. A commitment creates an obligation to comply with the rule or perform the contract.

NOTE - The object(s) participating in an act of commitment may be parties or agents acting on behalf of a party. In the case of acts of an agent, it is the entity modelled by the party which is bound.

6.7.3 Declaration: An act that ~~constitutes~~ creates a state in the universe of discourse modelled by the environment of the ODP system.

NOTE – The essence of a declaration is that, by virtue of the act of declaration itself, it causes a state of affairs to come into existence in the universe of discourse.

6.7.4 Delegation: An act ~~which~~ that delegates (authority, a function, etc.)

6.7.5 Description: An action ~~which~~ that communicates information about the representation within the system of a state in the universe of discourse modelled by the environment of the system.

6.7.6 Evaluation: An act ~~which~~ that assigns a value to something.

NOTE – For example, the act by which an ODP system assigns a relative status to some thing, according to an estimation by the system of its worth, usefulness, or importance.

6.7.7 Instruction: An act ~~which~~ that is intended to cause a person or machine to do something.

6.7.8 Prescription: An act ~~which~~ that establishes a rule.

Disposition: Editor to implement

US-35.

Location: 6.7.1

Qualifier: m

Rationale: It is not clear whether the construct of an initiating object is needed in the definition of an act. The cause-and-effect relationship may be of no importance in some contexts: for example, it may not be important who initiated a purchase-and-sale contract to which the parties committed (or, similarly, there may be several parties that jointly acted in a declaration such as creation of a new international body). The definition of an act, as it stands, may require unnecessary overspecification. It appears that an act may be accomplished by a collection of objects (parties or agents).

Proposal: Delete the reference to an initiating object. Refer to a collection of objects instead.

Disposition: Deferred to next ballot cycle

US-36.

Location: 6.7.3

Qualifier: m

Rationale: In addition to the Item about 6.7, the definition here is unclear. An act does not constitute a state; an act may result in a state (or establish a state)

Proposal: Rephrase the definition accordingly, also taking into account the Item about 6.7.

Disposition: Deferred to next ballot cycle

US-38.

Location: 7 p. 12 ll. 2-8

Qualifier: m

Rationale: An ODP system is not necessarily in an organization.

Proposal: Change “the context of the organization in which it operates” to “the context of the environment in which it operates.” Change ‘organization’ to ‘environment’ elsewhere, or remove ‘in the organisation,’ as appropriate.’

Disposition: Accepted - covered by re-write

US-39.

Location: 7

Qualifier: M

Rationale: It seems that organization, here and elsewhere, is a concept of the universe of discourse, and community is a modeling concept. Also, an object is a model of an entity (Part 2), so that a configuration of enterprise objects is a model of a set of entities. Furthermore, if the agreement governing the behaviour of entities in the organization is implicit then it is difficult or impossible to determine whether the agreement is, for example, violated: therefore, the model of such an agreement has to be explicit. Finally, it appears from this Clause that the ODP system is considered as a single enterprise object.

Proposal: Differentiate explicitly between the universe of discourse concept of an organization and the modeling concept of a community. Replace “representing” with “modeling” in the third sentence of the first paragraph. State the necessity of being explicit in specifying the agreement in the model. If the intent is to describe an ODP system as a single enterprise object then this also ought to be explicit.

Disposition: 2nd sentence yes, otherwise specific

US-40. 7.1

Qualifier: m

Rationale: A system considered as a single object at some abstraction level can, at another abstraction level, be decomposed into components, in accordance with Part 2. The terminology used in descriptions of this kind should be consistent with the one used in Part 2, and therefore “parts” should be replaced with “components”.

Proposal: Replace “parts” with “components” in the second bullet item.

Disposition: Editor to implement

US-41.

Location: 7.2.1

Qualifier: M

Rationale: A community is defined not only in terms of the elements enumerated in the bullet items, but also in terms of relationships between the roles of the community as well as relationships between the objects of the community (see, e.g., 7.4.3.3). Clause 7.2.3 already deals with relationships between communities, and therefore it would be unreasonable to keep default the relationships between elements of the community. Policies governing various actions and, especially, policies governing the configuration of objects and assignment of roles to objects (as well as configuration of roles) may be formulated in terms of these relationships.

Proposal: Mention explicitly relationships between roles of the community and between the objects of the community. Consider reusing some of the definitions from the beginning of 7.2.3, especially about compositions.

Agree on a definition of the concept of relationships between roles. For example, something along the lines of: “Role-1 is related to role-2” is shorthand for “When object-a fulfills role-1 and object-b fulfills role-2, then object-a is related to object-b.” Or, maybe, that sentence with ‘when – is’ replaced with ‘if – must be.’

Noted, US to offer text for next ballot

US-42.

Location: 7.2.1

Qualifier: e

Rationale: Part 3 says “— policies relating to environment contracts governing the system.” The last item in the first list says “— policies relating to environment contracts governing the community.” It is not clear why this change is made. This, of course, opens the often-avoided questions about environment contracts.

Proposal: Consider a rewording. Perhaps “--policies relating to environment contracts governing objects in the system, and the set of these policies that relate to the environment in which the system operates.”

Disposition: Withdrawn

US-43.

Location: 7.2.2

Qualifier: M

Rationale: Is a community established by a contract? It is possible that the contract that specifies how the objective can be met was put in place in an earlier epoch than that in which the community is created. Unless ‘establish’ has a different meaning than ‘create,’ the community is not established by the contract. A community may also be introduced.

Proposal: Rewrite to take into account epochs and introduction.

Disposition: Accepted - covered by re-write

US-44.

Location: 7.2.2

Qualifier: m

Rationale: Objectives of the parties to the community contract may be mutually inconsistent. Therefore these inconsistencies ought to be explicitly determined and resolved. This can be referred to, at least, by using the composition concept of Part 2.

Proposal: Replace “a subset of the objectives of all the parties” in the second sentence with “a composition of subsets of the objectives of all the parties”. Include a Note about the need to determine and resolve possible inconsistencies between the relevant subsets of the objectives of the parties. [This is a different meaning of party than proposed above for adoption.]

Disposition: Accepted - covered by re-write

US-45.

Location: 7.2.2

Qualifier: m

Rationale: Note 1 mentions “other roles.” Other than which roles is not clear.

Proposal: Consider adding a note that says: Not all objects that participate in the community need be parties. Some roles in the definition of the community may require objects that fulfil these roles to be parties, some other roles may require objects that fulfil these roles to be non-parties; and some roles may be fulfilled by objects that are either parties or non-parties.

Disposition: Deferred to next ballot cycle

US-46.

Location: 7.2.3

Qualifier: m

Rationale: The definition or description of relationships should include the definition of constraints (invariants) that may be used to define these relationships. These constraints may refer to specific roles or objects.

Proposal: Include a bullet item, in addition to the first three: “- the specification includes a constraint (invariant) referring to specific roles, or specific objects, in different communities”. In Note 2 in the second set of Notes, insert “, a constraint” after “such as a role template”.

Disposition: Subsumed in resolution of other comments

US-47.

Location: 7.2.3

Qualifier: M

Rationale: In the final Note, some expressions used are not well defined and appear to be borrowed from the legacy OO terminology. The Note itself illustrates important issues, and therefore should use terminology consistent with the rest of the standard. Specifically, the concept of inheritance in Part 2 is used in the context of derived class/base class, and therefore is not appropriate in the context of this Note. It is also useful to mention here that the properties of the role in an English Legal community

may be inconsistent with the existing properties of the community representing the company (as with companies in some of the countries that will join the EU), and therefore conflict determination and resolution mechanisms are essential.

Proposal: In the second paragraph of the last Note, insert in the third sentence “the components of” before “this outer, English Legal community”. Replace in the fourth sentence of the second paragraph “inherits from the outer community” with “, by virtue of fulfilling a specific role in the English Legal community, acquires a new type, i.e.,”. Insert a statement about conflict determination and resolution mechanisms needed for acquiring such new types.

Disposition: Accepted - Editor to implement but apply to re-worded note in UK-29

US-48.

Qualifier: m

Rationale: The terminology used here ought to be more precise. Specifically, there is no need to determine the “direction” of the composition relationship between objective and its sub-objectives. A step, in accordance with its definition, is an action, and not a decision. Finally, in the same manner as elsewhere, inconsistencies between objects and their acquired roles ought to be explicitly mentioned.

Proposal: Replace the second sentence of the first paragraph with “An objective can be a composition of sub-objectives.” Insert “interrelated” in the first sentence of the second paragraph between “a collection of” and “roles”. Replace the last sentence of the second paragraph with “A policy may influence the occurrence of a specific step in a process, which may in turn result in occurrences of other steps as well as creation (or termination) of processes.” Insert a statement about conflict determination and resolution mechanisms in the context of the last two paragraphs.

Disposition: Accepted Location: 7.3

US-49.

Location: 7.4.1

Qualifier: m

Rationale: Substantial fragments of this Clause repeat material already available in Part 2 (e.g., about interactions) or elsewhere in the standard.

Proposal: Consider the need to drastically reduce the size of this Clause.

Disposition: Noted

US-50.

Location: 7.4.1 p. 15 l. 13

Qualifier: e

Rationale: There is not support in OED for this use of ‘to reference.’ Of course, it is good computer software jargon. For ‘to mention’ OED has “to make mention of; to refer to or remark upon incidentally; to specify by name or otherwise.”

Proposal: Replace ‘referenced’ with ‘mentioned’.

Disposition: Accepted

US-51.

Location: 7.4.1 p. 15 l. 14

Qualifier: e, m

Rationale: Can we do better than “used?” We still are having trouble with defining ‘resource.’ But the definition we currently have at 6.2.1 is usable. The problem with this paragraph is that an object that participates in an interaction and in that interaction provides a service to another object can very well be said to be used by that other object. In particular, “The expression “use of a function” is a shorthand for the interaction with an object which performs the function.” [2-8.1]

Proposal: Delete the paragraph.

Disposition: Deferred to next ballot cycle

US-53.

Location: 7.4.2

Qualifier: E

Rationale: Large parts of this clause are not normative.

Proposal: Move the non-normative parts to an annex. Rewrite the normative rules in the style of, for example, subclause 7.2.3.

Disposition: Deferred to next ballot cycle

US-54.

Location: 7.4.2

Qualifier: m

Rationale: The terminology used here ought to be made more precise. Some recommendations appear to be too strong or incorrect. For example, the use of abstraction is determined by human understanding, and so “must be hidden by abstraction” is a recommendation applicable only in some contexts. It also appears that collective behaviour may be representing using interactions (as well as in other ways).

Proposal: In the second paragraph (ignoring Notes), delete the first sentence. In the paragraph about “tasks”, replace “part of a process” with “component of a process”. In the (b) bullet item, replace “for each role” with “for appropriate collections of roles”. Delete the first sentence of the last paragraph (repetition).

Disposition: Deferred to next ballot cycle

US-55.

Location: 7.4.2

Qualifier: M

Rationale: As noted, this text uses “participate in an action” rather than “perform an action.”

Proposal: Consider the expressions that are used in Part 2: actions are associated with an object; actions take place with the participation of an object; actions in which object take part. Use “participate in an interaction.”

Disposition: Deferred to next ballot cycle

US-52.

Location: 7.4.2 p. 16 ll. 20-21

Qualifier: E

Rationale: The USA agrees that a single word is needed here.

Proposal: At this time the USA is unable to propose “a single word for ‘scope/purpose/objective’ and ‘result/purpose/sub-objective’ in this clause.”

Agreed – Remove temp note

US-56.

Location: 7.4.3

Qualifier: M

Rationale: This Clause seems to make the incorrect assumption about the existence of a single type of an object (and of a role). Both an object and a role may have several types. Some terminology used in the Note is inconsistent with the rest of the standard. In accordance with the duality of state and behaviour, it is possible and desirable to define communities in terms of invariants about relationships between their roles (and objects), see also 7.4.3.3. There is a need to specify explicitly different epochs in the lifecycle of the community (perhaps also referring to Clause 7.8). There is no need to determine what comes first -- the community behaviour or the set of constraints that define this behaviour -- since these would be modeling recommendations. The Note says "the behaviour is seen as continuing throughout the lifetime of the community." Is it not the case that the behavior may continue for some specified time in the lifetime of the community? The sixth paragraph of the Note is incorrect since it is based on the idea of a single type of an object and of a role. With reference to the first Editor's Note, it may indeed be possible for a role to have no objects associated with it; this is determined by the invariant of the community and by the definition of the instantiation action of the community. With reference to the second Editor's Note, it appears that a term cannot be destroyed but its context can.

Proposal: In the first sentence of the third paragraph, replace “one or more objects is” with “one or more objects may be”. At end of this sentence, add the following: “, depending upon the constraints of the community and upon the definition of the instantiation action of the community”. Add a Note about the need for conflict resolution.

In the first paragraph of the Note, insert at end of paragraph: “The constraints (invariants) of the community should be satisfied during its lifetime. However, this invariant may change thus determining different epochs in this lifetime. Specifically, this may lead to the changes in the sets of roles of the community.”

In the third paragraph of the Note, delete “type-safe” in the first sentence. Delete the second sentence of this paragraph. In the third sentence, replace “community type specification” with “community template specification”.

Add a new paragraph: “When an object is associated with a role, it acquires a new type corresponding to the role type. The predicates which are the types of the role should not contradict with any of the object types of an object that is to fulfil this role. If the predicates of the object types and the role type(s) may nevertheless be inconsistent the specification should include mechanisms to determine and resolve such inconsistencies, for example, by changing some of these types.”

Rephrase the fifth paragraph without any reference to “what comes first”.

Delete the sixth paragraph (it is incorrect).

Clarify the duration of the continuing behaviour.

Disposition: Accepted - covered by re-write

US-59.

Location: 7.4.3

Qualifier: e

Rationale: The term, 'sub-classification' is introduced here for the first time.

Proposal: Replace with a standard term, such as ‘subclasses.’

Disposition: Editor to implement Remove last 2 paras of Note

US-60.

Location: 7.4.3

Qualifier: m

Rationale: The text says one class of interest is environment roles and core roles. Discussions of environment and core roles often include interface roles.

Proposal: Add interface roles to the second bullet.

Resolved by US-59

US-57.

Location: 7.4.3 p. 16 l. 27

Qualifier: e

Rationale: This is redundant with text at lines 2-3 of page 16.

Proposal: Delete this text or the other.

Disposition: Delete ll2-3 of p16

US-58.

Location: 7.4.3 p. 16 ll. 27-28

Qualifier: M

Rationale: A role can be fulfilled by only one object at a time.

Proposal: delete “or by several objects at the same time.”

Disposition: Agreed

US-61.

Location: 7.4.3 p. 16 l. 32

Qualifier: M

Rationale: A community has identity. It remains the same community when objects fulfilling roles change, when roles are added or removed, and across epochs.

Proposal: If the working group agrees that this is so, make this clear when this text is rewritten.

Disposition: Deferred to next ballot cycle

US-62.

Location: 7.4.3 pp.16-17 ll. 37-45, 1-25

Qualifier: E

Rationale: This is explanatory material.

Proposal: Move this long note to an informative annex.

Disposition: Rejected

US-65.

Location: 7.4.3.1 p. 17 ll. 26ff.

Qualifier: m

Rationale: The structuring rules in Part 3 prescribe that an enterprise specification starts with a specification of a community that consists of the system being specified and the objects in its environment. There are no objects outside this community.

Proposal: Add text at the beginning of this paragraph or a note after to make this clear.

Disposition: Editor to implement – as a Note

US-63.

Location: 7.4.3.1

Qualifier: m

Rationale: Not all interactions of the objects of a community with objects outside of that community are relevant to the specification of the community.

Proposal: Change the first sentence to “Objects of a community may interact with objects outside of that community to achieve the objective of the community.”

Disposition: Editor to implement as written

US-64.

Location: 7.4.3.1

Qualifier: E

Rationale: The sentence ‘A description of a system in an enterprise specification may identify roles to be fulfilled by those objects outside that community that interact with objects of the community’ is redundant or unclear and possibly incorrect.

Proposal: Delete this sentence.

Disposition: Withdrawn

US-66.

Location: 7.4.3.1 p. 17 ll. 38-40

Qualifier: m

Rationale: Reenskaug describes a procedure for composing role models which is also suitable for composing community templates.

Proposal: Consider rewriting this text in light of that approach.

Resolved by AFNOR 25

US-67.

Location: 7.4.3.1 p. 18 ll. 11-14

Qualifier: M

Rationale: The USA considers that a role may be fulfilled only by one object at a time or by no object, not by more than one object at a time.

Proposal: Delete language that indicates that several objects can fulfill the same role at the same time.

Disposition: Accepted

US-68.

Location: 7.4.3.1 p. 18 ll. 11-20

Qualifier: e

Rationale: Most of the examples in the current draft use persons and organizations rather than information processing systems. Given the primary purpose of the ODP standards, examples using distributed information processing systems will be useful to readers.

Proposal: Replace the example. Consider examples such as the one of the posting ledger system.

Disposition: Accepted

US-69.

Location: 7.4.3.1 p. 18 ll. 21-23

Qualifier: M

Rationale: This does not seem artificial to some. In the example, it appears that customers have obligations in commercial activity.

Proposal: Delete the paragraph.

Disposition: Noted

US-70.

Location: 7.4.3.1 p. 18 ll. 11-26

Qualifier: E

Rationale: This note is informative.

Proposal: Move this note to an informative annex.

Disposition: majority view is that informative notes should be in the body but that they should be concise and relevant. Agree remove 2nd 2 paras.

US-71.

Location: 7.4.3.1

Qualifier: m

Rationale: Constraints on configurations of roles of the community are determined in the community specification and usually include more interesting semantics than cardinalities. Also, a role may have several types.

Proposal: In the final Note, add after the first sentence: "Constraints on configurations of roles of the community are not restricted to cardinalities and are specified in the community specification. For example, only some members of a community may be members of a specific committee of that community." In the second paragraph of the final note, in the second sentence, replace "role types" with "role templates".

Disposition: Subsumed in resolution of other comments

US-72.

Location: 7.4.3.2

Qualifier: m

Rationale: The terminology used here is defined later (e.g., CEO). Also, the concept of the core community seems to differ from the concepts in the previous Clauses. Finally, the CEO probably cannot fulfil the interface roles since different abstraction levels appear to be used in the same sentence.

Proposal: Rewrite the text without using the term CEO and to rectify the problems.

Disposition: Subsumed in resolution of other comments

US-73. 7.4.3.3 p. 18 ll. 43-44

Qualifier: E

Rationale: As written, this is advice.

Proposal: Delete or move to an annex.

Disposition: Editor to implement. – make a note

US-74.

Location: 7.4.3.3

Qualifier: m

Rationale: Policies also govern the actions in roles, but are not included in this list.

Proposal: Add mention of policies.

Disposition: Editor to implement

US-75.

Location: 7.4.3.3 p. 18 l. 41

Qualifier: e

Rationale: Do we want to say “constraints on behaviour” or “constraints of behaviour?” The constraints, after all, are part of the behaviour. The sense in which role is used in this paragraph is unclear.

Disposition: Remove constraints on behaviour.

US-76. 7.4.4.1 p. 18-19 l. 47, ll. 1-22

Qualifier: m

Rationale: The second alternative is simpler and as good a model.

Proposal: Adopt the second alternative.

Disposition: Accepted

US-77.

Location: 7.4.4.1

Qualifier: m

Rationale: The state change described in this Clause may refer to the change of states of several enterprise objects (i.e., collective state that should satisfy the properties defined in the appropriate

invariants). More generally, the text before the Temporary Note appears to be substantially weaker than the text after.

Proposal: Use the second alternative mentioned in the Temporary Note. If a reference to the state is used, formulate it in terms of the collective state of several objects.

Disposition: Accepted

US-78.

Location: 7.4.4.2

Qualifier: m

Rationale: An action may be associated with more than one actor role (for example, in contract negotiation).

Proposal: Replace “some actor role” with “some actor roles” in the first sentence.

Disposition: Editor to implement using “is” and “at least one actor role”

US-79.

Location: 7.4.4.2

Qualifier: E

Rationale: Both paragraphs are methodology.

Proposal: Move to an annex.

Disposition: Deferred to next ballot cycle

US-80.

Location: 7.5

Qualifier: m

Rationale: The behaviour subject to policies need not be more restricted than it was before since some policies represent an empowerment (in some countries, for example, after the change of policies it became possible to choose one of several candidates during an election).

Proposal: Rephrase the first bullet item in accordance with the Rationale.

Disposition: Editor to implement noting that constraints are different

US-81.

Location: 7.5

Qualifier: e

Proposal: Rewrite or remove as suggested in the temporary note.

Disposition: Deferred to next ballot cycle

US-82.

Location: 7.5.1

Qualifier: M

Rationale: A policy may apply to a configuration of roles within a community (and thus, for example, define a relationship between roles of the community). Therefore the first paragraph is overly

restrictive and even misleading. Also, a community does not inherit from its environment, but rather acquires new type(s) in accordance with its role(s) in outer communities.

Proposal: Include configuration of roles of the community in the second sentence of the first paragraph. Rephrase the second paragraph in the same manner as proposed above for the Clause 7.2.3.

Disposition: Editor to implement as a set of roles – but not 2nd bit

US-83.

Location: 7.5.1

Qualifier: m

Rationale: “Within a community, policies are used to express constraints on the behaviour of objects fulfilling actor roles.” Is it not possible that policies also constrain the behaviour of objects fulfilling artefact and resource roles?

Proposal: Rewrite to include these other possibilities.

Disposition: Editor to implement

US-84.

Location: 7.5.1

Qualifier: m

Rationale: “No community can operate in isolation from its context; it cannot arbitrarily place or relax policies, but is itself constrained by what it inherits from the environment in which it is created and the obligations taken or responsibilities devolved...” But there is at least one community in each enterprise specification for which there are no objects in its context. This community necessarily operates in isolation. Other communities will get obligations and responsibilities as provided by the specification. A general rule is not needed. (This paragraph may be explanatory, and not normative, in which case this may not be intended as a general rule.) The phrase ‘what it inherits from the environment’ does not have a clear meaning.

Proposal: Remove this statement. If it is kept: Replace ‘what it inherits from the environment.’ Consider using a different word than ‘devolved.’

Resolved by re-wording agreed

US-85.

Location: 7.5.2

Qualifier: m

Rationale: The Clause contradicts with the rest of the standard since it does not mention the constraints specified in the roles to be fulfilled by the object in communities in which it will participate. It is also not clear whether the constraints of the outer communities can be overridden.

Proposal: Rephrase the Clause in accordance with the Rationale. State that the specification should determine whether and under what conditions the or some constraints of the outer community can be overridden by the owner.

Disposition: Accepted - Editor to implement

US-86.

Location: 7.5.3

Qualifier: M

Rationale: This text is overly prescriptive. It is material for a possible ODP policy language standard.

Proposal: Remove the subclause, or move it to an Annex.

Disposition: Editor to implement – temporary note saying 1st sentence and asking for material that is not overly ...

US-87.

Location: 7.5.3

Qualifier: m

Rationale: The first paragraph does not mention violations: “no choice” appears to imply that violations are impossible. Also, policies are not necessarily constraints.

Proposal: Provide additional text in accordance with the Rationale.

Disposition: Rationale agreed – no text

US-88.

Location: 7.5.3

Qualifier: m

Rationale: This model of choice does not feel correct. Roughly speaking, and for our purposes, humans have complete freedom of choice. Computer programs, on the other hand, only have choice in a very special sense of the word. To conflate these two meanings of ‘choice’ is wrong.

Proposal: Revise the text in accordance with the Rationale.

Rationale agreed – no text

US-89.

Location: 7.5.3.1 and 7.5.3.2

Qualifier: m

Rationale: Collective behaviour is not treated properly. Permission may be defined for a set of roles instead of by a single role: a permission for parties to enter a contractual relationship is an example.

Proposal: In the second bullet item of 7.5.3.1, replace “a role” with “a non-empty set of roles”. In the third paragraph, replace “If an enterprise object” with “If a non-empty set of enterprise objects”. Similarly, replace “the enterprise object” with “this set of enterprise objects”, and “the role” with “the role(s)”. Similarly, replace the old text with the same kind of new text in 7.5.3.2.

Disposition: Editor to implement

US-90.

Location: 7.5.3.3

Qualifier: m

Rationale: Standing obligation may not be fulfilled by a single occurrence of the prescribed behaviour, so that the first paragraph appears to be inconsistent with the end of the Clause. It is possible to include a reference to composition in the second paragraph. The non-normative text ought to be deleted. Triggering conditions may be a very useful construct in describing policies, and can be described separately if so desired.

Proposal: In the second sentence, state that some (non-standing) obligations may be fulfilled by the occurrence of the prescribed behaviour. In the second paragraph, state that new policies are composed

with the existing (contextual) ones, and that potential conflict id to be recognized and resolved in accordance with the specific composition rules provided in the specification. Rewrite the items (1)-(3) by deleting non-normative text such as examples (or including it in Notes). Consider a separate Clause for triggering conditions. In the last sentence, consider mentioning deactivation of the standing obligation; also, use one of the terms “satisfied” (this paragraph) or “fulfilled” (first paragraph).

Disposition: Deferred – no text

US-91.

Location: 7.5.5

Qualifier: m

Rationale: Most of the text of this Clause is a narrative and not a normative text.

Proposal: Remove or rewrite the Clause.

Disposition: Acceptance in principle but text remains as place holder for normative, appropriate text.

US-92.

Location: 7.5.6

Qualifier: M

Proposal: Unless some national body or the editor proposes language that suits the invitation, remove this clause.

Disposition: Rejected

US-93.

Location: 7.5.6

Qualifier: m

Rationale: The arguments in the second paragraph apply not only to the organization of policy, and even not only to the enterprise viewpoint. These arguments appear to follow from Part 2. In addition, it is necessary to state that the submodels may be mutually inconsistent and that, therefore, such inconsistencies should be acknowledged and resolved in the same manner as expressed earlier, within the framework of composition.

Proposal: Consider deleting the second paragraph. Include considerations analogous to the ones specified above (e.g., for Clause 7.5.3.3) about possible inconsistencies and their resolution.

Disposition: Editor to implement as shown

US-94.

Location: 7.5.6.1

Qualifier: m

Rationale: Collective behaviour is not used in this Clause, but should be. Also, the concept of the “general permission” is not clear.

Proposal: In the first bullet item in the first list, replace “a particular role” with “a particular set of roles”. In the second paragraph, replace “with an object” with “with a set of objects”. Clarify the second bullet item in the second list.

Disposition: Subsumed in resolution of other comments - text has gone

US-95.

Location: 7.5.6.2

Qualifier: M

Rationale: The clause needs to be rewritten as suggested in the temporary note.

Proposal: If such text is not proposed, remove the clause.

Disposition: Accepted

US-96.

Location: 7.5.6.2

Qualifier: m

Rationale: The term 'operation' in the first paragraph is a computational viewpoint term. It is not clear whether this was the intent. Should the 'any' in the last bullet item be replaced with 'all'?

Proposal: Clarify the usage of 'operation' (or replace the term).

Disposition: Subsumed in resolution of other comments

US-98.

Location: 7.6

Qualifier: m

Rationale: Substantial fragments of this clause are narratives about modeling methods; they do not belong to the normative clauses of the standard. The definition of CEO provided here is used earlier.

Proposal: Move the clause earlier. Delete the second and third paragraphs. In the first paragraph, include legal entities, etc. in the enumeration in the second sentence.

Disposition: Editor to implement but don't move

US-99.

Location: 7.6

Qualifier: m

Rationale: The second sentence is too limiting, even with the additions proposed in the previous item. An enterprise object may represent anything the specifier feels is necessary or desirable to specify the system from the enterprise viewpoint or to understand the enterprise specification.

Proposal: Remove the second sentence.

Disposition: Editor to implement – make a note

US-97.

Location: 7.6

Qualifier: M

Rationale: The first sentence is false. An enterprise object is any object in an enterprise specification. Those will be exactly the object the specifier feels are necessary or desirable to specify the system from the enterprise viewpoint or to understand the enterprise specification.

Proposal: Remove the first sentence.

Disposition: Accepted – use rationale 2nd two sentences to replace existing 1st sentence.

US-100.

Location: 7.7

Qualifier: m

Rationale: The concept of a contract is defined in Part 2. If the agreement about the “group of entities” is implicit then its model still ought to be explicit since otherwise checking whether the agreement constraints have been satisfied or whether a violation occurred is impossible. Furthermore, in accordance with UCC a contract ought to include consideration, and although the existing text refers to the objective for which the group exists this need not include consideration. Also, a community can have several types, and therefore “the community type” is an incorrect expression.

Proposal: Replace references to Part 3 with references to Part 2. State that the agreement, even if implicit, should be modeled explicitly. Consider inclusion of a consideration in enumerating the items covered by an agreement. Replace “things” with something else. Replace “community type” with ‘community template’ throughout the Clause. Replace “the relationship between the roles” with “the relationships between the roles”.

Disposition: Accepted - covered by re-write of 7.1 with the exception of consideration, which is Rejected

US-101.

Location: 7.7

Qualifier: e

Rationale: The phrase “—the structure, policies, and behavior of its members” intends the structure of the group, the policies for the group and the behavior of its members. As mentioned elsewhere, it is the collective behaviour or the members that is most interesting.

Proposal: Change the sentence accordingly.

Disposition: Accepted - covered by re-write

US-102.

Location: 7.7

Qualifier: M

Rationale: The term ‘group’ can not be used here with the intended meaning. Group is a foundation language concept from Part 2.

Proposal: Choose another word.

Disposition: Collection to be used in re-write

US-103.

Remove all of 7.8 to 7.8.1.

Location: 7.8

Qualifier: M

Rationale: The clause is based on theories on which there is not consensus in the working group.

Proposal: Remove the clause or move it to an Annex.

US-104.

Location: 7.8

Qualifier: m

Rationale: Not only communities, single roles and single objects may have objectives, but also collections of objects may have (joint) objectives (for example, parties in a contract may want to be able, if necessary, to enforce it through legal means). It should also be noted that objectives may change, and such changes usually mean changes of epochs.

Proposal: In the third paragraph, remove references to “The starting point”. Add statements about joint objectives of collections of objects. In the fourth paragraph, insert “relationships between roles” after “by specifying the roles”. In Note 1, replace “if the object has a plan to manage” with “if the specification provides for managing”; mention also collective behaviour of objects in this sentence. In Note 2, replace “because the role’s objectives may be a subset of” with “because the role’s objective may be a component of”.

Add a statement about epochs in the context of possible changes of objectives.

Disposition: Subsumed in resolution of other comments

US-105.

Location: 7.8.1

Qualifier: m

Rationale: In the same manner as in 7.2.2, objectives of the parties to the community contract may be mutually inconsistent. Therefore these inconsistencies ought to be explicitly determined and resolved. This can be referred to, at least, by using the composition concept of Part 2.

Proposal: Replace “a subset of the objectives of all the parties” in the second sentence with “a composition of subsets of the objectives of all the parties”. Include a Note about the need to determine and resolve possible inconsistencies between the relevant subsets of the objectives of the parties.

Disposition: Subsumed in resolution of other comments

US-106.

Location: 7.8.1

See the comment to 7.2.1.

Disposition: Subsumed in resolution of other comments

US-107.

Location: 7.8.1

Qualifier: M

Rationale: The working group still does not share a clear understanding of the concept, ‘agreement between objects.’ There is no clear, terse text to tell us what it means. The term, ‘agree’ in the sense intended in ‘agreement between objects’ can not mean the same thing as ‘agree’ in ‘agreement between people.’

Proposal: Discuss this an attempt to form a consensus.

Disposition: Subsumed in resolution of other comments

US-108.

Location: 7.8.1

Qualifier: e

Rationale: The second sentence about subsets of objectives is either tautological or overly prescriptive.

Proposal: Remove the sentence

Disposition: Subsumed in resolution of other comments

US-109.

Location: 7.8.2

Qualifier: m

Rationale: A community may change its objectives, constraints, or roles, due to changes in the outer communities in which this community fulfils some roles. For example, a community of law-abiding citizens enjoying certain beverages became a community of law violators (with possibly different roles) with the introduction of Prohibition in the outer community. This meant a change of epoch. Moreover, the objectives, constraints and roles of a community may change by decision of the owners of the community.

Proposal: Add a reference to changes in community objectives, constraints or roles due to changes in the outer communities in which this community fulfils some roles, or due to the decisions of the owners of the community. Mention the changes of epochs in such situations.

Accommodated by replacement text in 7.1

US-110.

Location: 7.8.2

Qualifier: m

Rationale: It may be that in “creating and deleting roles (as instances),” ‘(as instances)’ is meant to distinguish type of role from role.

Proposal: If so, rewrite to make this clear.

Disposition: Editor to implement – remove (as instances) and add new words that say that creation of roles can only happen if the community specification includes definition of the relevant role type.

US-111.

Location: 7.8.3

Qualifier: m

Rationale: A community may be terminated also due to changes in the outer communities in which this community fulfils some roles. For example, some religious communities ceased to exist in some countries when their government (system) changed. This also meant a change of epoch.

Proposal: In the third bullet item, insert “or due to the changes in outer communities in which this community fulfils some roles”. Mention the change of epochs in such situations.

Or rethink this whole business discussed in 7.8.2 and 7.8.3.

Disposition: Deferred to next ballot cycle

US-112.

Location: 7.9

Qualifier: M

Rationale: The working group has invited national bodies to propose structuring rules for the force concepts of subclause 6.7

Proposal: Add these rules as a first cut:

7.9 Force rules

TEMPORARY NOTE – These are a first attempt at rules for the use of the force concepts. National bodies are invited to comment on and suggest improvements to this text before the November 1999 meeting of SC 7/WG 17.

The environment of an ODP system includes natural persons or other entities considered to have some of the rights, powers and duties of a natural person; these entities are modeled by parties. The environment also may include other information processing systems in addition to that ODP system. This Reference Model provides rules for the specification of an ODP system in terms of the force that actions of that system have in the universe of entities modeled by parties in its environment.

An enterprise specification describes the authority delegated to an ODP system in terms of:

- the parties that have delegated authority to the system;
- the authority that each party has delegated;
- the duration and conditions of the delegation;
- provisions for additional delegation and withdrawal of delegation during the operation of the system.

By each such delegation, the system becomes an agent of the party delegating, and the party becomes a principal of the system.

Actions of an ODP system that are an exercise of delegated authority are acts. Acts cause states of affairs in that part of the universe of discourse modeled by the parties in the environment of that system. The force of an act specifies which kinds of states of affairs the act causes.

7.9.1 Authority rules

For each authority delegated, an enterprise specification describes the force of actions of the system in exercising that authority, including authority:

- to make commitments, these bind the principal
- to issue declarations, these create a state of affairs just as if the principal had made the declaration;
- to delegate an authority to a component of the system, to another system or to a party, these cause that agent to have the authority;
- to communicate descriptions behalf of the principal;
- to make evaluations on behalf of the principal;
- to give instructions to a person or machine on behalf of the principal;
- to make a prescription which establishes a rule with the same force as if the principal had made the prescription.

TEMPORARY NOTE - We are short of terms here. This should probably not be ‘the principal’ (an object), but ‘the entity modelled by the principal.’ Lacking a succinct term for that concept, it will be necessary to correct this clause by spelling it out every time.

7.9.2 Delegation rules

When an ODP system is an agent with authority to delegate, and it may (subject to the provisions of that authority) delegate (authority, a function, ...) to a party or other system. That party or other system shall then be an agent of the principal and have the same authority as if delegated directly by the principal.

In general, a principal is responsible for the acts of its agents.

Disposition: Accepted as Editor to implement – note to explain why the term “force” is used and requesting a name for the clause.

US-113.

Location: 7.10.1 and 7.10.2

Qualifier: E

Rationale: These are not intended to be redefinitions of <X>-domain and <X>-federation.

Proposal: In the first sentence of each clause, replace “<X>-domain” with “<X>-domain community type” and “<X>-federation” with “<X>-federation community type”

Disposition: Deferred to next ballot cycle

US-114.

Location: 7.10.3

Qualifier: m

Rationale: The term ‘subtype-by-specialization’ is not defined in Part 2. The term ‘non-subordinate’ is not defined. Also, the last sentence (“Note”) should mention conflict detection and resolution.

Proposal: Delete the first sentence. Rewrite to show the meaning of ‘non-subordinate.’ Refer to conflict detection and resolution in the last sentence.

Disposition: Accepted – delete non-subordinate – reject final comment

US-115.

Location: 8

Qualifier: M

Rationale: There is not consensus in the working group on the idea of an assumed environment in terms of outer communities. It appears to some that many communities can be specified in glorious isolation. At least, and certainly, the idea that an outer legal system must be specified is not correct.

Proposal: Remove this clause.

Disposition: Subsumed in resolution of other comments

US-116.

Location: 8

Qualifier: m

Rationale: In the same manner as in 7.2.3 and 7.5.1, a community does not inherit from its environment, but rather acquires new type(s) in accordance with its role(s) in outer communities. The outer communities need not be explicit, but at times can be considered “self-evident”; in such cases their constraints should still be specified explicitly.

Proposal: Replace in the first paragraph “which it inherits from the environment” by “which exists due to the fulfilling by this community of specific roles in the communities that constitute the environment”. In the third paragraph, note that the outer communities need not be specified, but that the constraints (and policies) should be explicit.

Disposition: Subsumed in resolution of other comments

US-117.

Location: 8

Qualifier: M

Rationale: Items a) thru d) are too prescriptive.

Proposal: Move this to an Annex as an example of how it might be done.

Disposition: Subsumed in resolution of other comments

US-118.

Location: 9

Qualifier: E

Rationale: This material is not about conformance and reference points.

Proposal: Move the material to Clause 7.

Disposition: Editor to implement note that Conformance clause may be needed

US-119.

Location: 9

Qualifier: e

Rationale: This use of ‘conformant’ does not correspond to the definition of conformance [3-4.6].

Proposal: Change “An enterprise specification conformant with this Recommendation | International Standard” to “An ODP enterprise specification.”

Disposition: Subsumed in resolution of other comments

US-120.

Location: 9

Qualifier: M

Rationale: The meaning, “unambiguous explanations of the relationships between the concepts” in the two rules is ambiguous.

Proposal: Delete the first rule. Remove “unambiguous explanations of the relationships between the concepts” from the second rule.

Disposition: Delete “unambiguous” but otherwise Deferred to next ballot cycle.

Comment of Unknown source

Unknown NB Cat: E Clause: 3.2

Rationale:

Not all abbreviations and acronyms in the text are identified here, for example “UML” is referenced in clause 11 and Annex A.

Proposal:

The Editor should carry out a search through the text to find all abbreviations and acronyms

Disposition: Accepted