Annex E – EXPLANATION OF FUNDAMENTAL CONCEPTS FOR CONCEPTUAL MODEL FRAME-OF-REFERENCE

In order to better understand and incorporate into the best-practice guidance specified in detail in following annexes, the Task Group undertook an analysis of fundamental concepts for conceptual model composition and specification. This analysis addressed particularly the differences and similarities between a variety of commonly used conceptual models, formalisms and ontologies. This analysis had as its objective the following:

- Give appropriate and necessary definitions. Discuss the differences and the benefits of *natural languages, artificial languages, and machine languages.*
- Discuss the possibility to verify the models by rules and constraints – if they have formal representations.
- Explain how to transform an informal conceptual model into a formal or semi-formal model.
- Discuss Machine Readability – e.g., to make an ontology 'machine-readable' we need to select a formal, machine-processable implementation language.
- Show the steps towards making the human-readable information also machine-readable. (One of the accepted ways of expressing such machine readable conceptual models is UML (Unified Modeling Language). UML is being used for far more than simply conceptual modeling. Additionally, UML class diagrams (conceptual models) can be categorized as *semi-formal* ontologies themselves. (Semi-formal because they are not directly machine-readable.) However, tools are being developed that enable automatic transformation from UML class diagrams to ontology formalisms like DARPA Agent Markup Language (DAML), OWL, etc.).

E.1 OVERVIEW

To determine what can be a good conceptual frame for a set of elements for capturing the semantic contents necessary and sufficient for a conceptual model, we have reviewed a number of methods known to the Task Group to be relevant and representative for use for conceptual modeling. We identified and studied sixteen (16) different methods/templates and have noted their properties, their most important information elements, parameters, etc. The results of this analysis revealed, as expected, the common factors among them from which we could select attributes of the consequent conceptual model specification prescriptive guidance to follow.

Given this analysis, a synthesis step followed. Having found the denotative name for information elements, their information content, and justification of their necessity; we allocated those elements to our nascent conceptual modeling process and product. Taken together with the options are their representation/specification, and the degree their quality should be certified; we proceeded to successfully generate a table of semantic contents necessary and sufficient for a conceptual model, along with their qualifying attributes and other necessary information.

For the purpose of this analysis/synthesis, the following terminology is relevant:

- A 'Behavior' is the means an Actor uses to actuate Events.
- An 'Actor' is the subject of action.
- An 'Event' is an action composed of Activities.

- A ‘Control’ defines what can occur within an Activity.
- ‘Information’ details the capabilities of a Behavior, Actor, Event, or Control.

Using this vernacular, a process is specified (in the present context), thus:

Using <Behaviors>, an <Actor> actuates <Events> composed of <Activities> that are directed by <Controls> and richly described by <Information>.

… where the text indicated within carets as reserved words are used below in tabulating the model schema characteristics in tabular form for each model schema analyzed.

E.2 ANALYSIS

Analysis of conceptual schema alternative styles follows, given the primitives established above. In each case, we identify the schema, and characterize it systematically in order both to:

- Appreciate its intrinsic nature and the systematic relationships among alternative typical relevant schema; and
- Provide evidence to the effect that the use of the normative precept above is sufficient for any conceptual model specification.

E.2.1 Method: BOM [1]

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Patterns of interplay:
   A. Identifies a sequence of actions.
II. State machines:
   A. Identifies the behavior states expected to be exhibited by one or more conceptual entities.
III. Event type (Conceptual events):
   A. Describes the types of events that are the result of, or triggered from, actions relevant in a pattern of interplay.
   B. Two types:
      1. Triggers.
      2. Messages.
IV. Entity type (Conceptual entities):
   A. Types of conceptual entities that represent senders and receivers of information in a pattern of interplay.
   B. Object classes.
   C. Interaction classes.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by *Information***
State machines	Entity type	Event type	Patterns of interplay		
	(Object class)	(Triggers)			
	(Interaction class)	(Messages)			

E.2.2 Method: BOM++ [2]

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Patterns of interplay:
   A. Identifies a sequence of actions.
   B. Identifies the behavior states expected to be exhibited by one or more conceptual entities.
II. Event type (Conceptual events):
   A. Describes the types of events that are the result of, or triggered from, actions relevant in a pattern of interplay.
   B. Two types:
      1. Triggers.
      2. Messages.
III. Entity type (Conceptual entities):
   A. Types of conceptual entities that represent senders and receivers of information in a pattern of interplay.
   B. Extended through use of Namespace Qualified Extensions.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by *Information***
State machines	Entity type	Event type	Patterns of interplay		
	(Object class)	(Triggers)			
	(Interaction class)	(Messages)			

E.2.3 Method: BPMN [3]

BPMN is particularly conceived to meet the following criteria:

- Designed to be usable to business community; and
- Must generate executable processes through a BPMN model, coupled with some additional information.

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Activities:
   A. Work that is performed within a business process.
   B. Can be atomic or non-atomic (compound).
   C. Activities that are a part of a Process Model are:
      1. Sub-Processes:
         a) A compound activity included within a Process:
            (1) Can be broken down into a finer level of detail (a Process) through a set of sub-activities.
            (2) Enables hierarchical process development.
            (3) Two types:
               (a) Embedded.
               (b) Independent (reusable).
II. Task:
   A. An atomic activity that is included within a Process.
   B. Used when the work in the Process is not broken down to finer level of Process Model detail.
   C. Can be looped.
III. Events:
   A. Something that "happens" during the course of a business process.
   B. Affect the flow of the Process.
   C. Usually have a trigger or a result.
   D. Can start, interrupt, or end the flow:
      1. Start:
         a) Indicates where a Process will begin.
      2. Intermediate:
         a) Occur after a process has been started and before a process is ended.
      3. End:
         a) Indicates where a process will end.
IV. Gateways:
   A. Modeling elements that are used to control how Sequence Flows interact as they converge and diverge within a Process:
      1. Exclusive Gateways (decisions):
         a) Locations within a business process where the Sequence Flow can take two or more alternative paths.
         b) Only one of the possible outgoing paths can be taken when the Process is performed:
            (1) Two types decision mechanism:
               (a) Data.
               (b) Events.
               (c) A branching point in the process where the alternatives are based on events that occurs at that point in the Process, rather than conditions.
      2. Inclusive Gateways:
         a) Where there is more than one possible outcome.
      3. Complex Gateways:
         a) Where there more advanced definitions of behavior can be defined.
      4. Parallel Gateways:
         a) Places in the Process where multiple parallel paths are defined.
V. Connectors:
   A. Sequence Flow:
      1. Used to show the order that activities will be performed in a Process.

2. Source and target must be Events, Activities, and Gateways.

B. Conditional.

C. Default:

1. Exits an Exclusive or Inclusive Gateway may be defined as being the default path.

D. Message Flow:

1. Used to show the flow of messages between two entities that are prepared to send and receive them.

E. Association:

1. Used to associate data, information and artefacts with flow objects.

VI. Swim-lanes:

A. Used to partition and organize activities.

B. Pool:

1. Represents Participants in an interactive (B2B) Business Process Diagram.

C. Lane:

1. Represents sub-partitions for the objects within a Pool.

VII. Artefacts:

A. Provide the capability to show information beyond the basic flow-chart structure of the Process:

1. Data Objects:

a) Used to show how data and documents are used within a Process.

2. Groups:

a) Used to highlight certain sections of a Diagram without adding additional constraints for performance, as a Sub-Process would.

3. Annotations:

a) Provide additional information about a Process.

b) A modeler or tool can extend BPMN by defining new Artefacts.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by Information**
		Event	Process		Gateways
			– Activity		Connector
			– Sub-process		– Sequence flow
			(embedded)		– Conditional
			(reusable)		– Default
			– Task		– Message flow
					– Association
					Swim-lanes
					– Pool
					– Lane
					Artefacts
					– Data objects
					– Groups
					– Annotations

E.2.4 Method: CML (OneSAF) [4]

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Elements:
   A. Real/abstract things making up and acting in the battle-space.
   B. Belong to a Category.
   C. Have Characteristics.
   D. Issue Events.
   E. Stimulate Behaviors.
   F. Behaviors change state of Elements.
   G. Two sub-classes of Elements:
      1. Piece.
      2. Players.
      3. Those things explicitly represented:
         a) Tanks, infantry, etc.
         b) Markers.
      4. Things implicitly represented:
         a) Tactical positions, Psyops, etc.
      5. Game-space:
         a) Environment.
         b) Establish physical battle-space whereas Elements have a location.
      6. Zone:
         a) Abstract spaces where inhabitants do not have a location:
            1) Satellites providing intelligence.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by *Information***
Behaviors	Elements	Events		Game-space	Category
	– Piece			Zone	Characteristics
	– Player				

E.2.5 Method: CommonKADS [5]

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Communication Model:
   A. Needs and desires with regards to other agents (e.g., a user interface or interfaces with other systems).
II. Knowledge Model:
   A. Knowledge and reasoning requirements.

B. Specifies data and knowledge structures for an application:
   1. Domain knowledge.
   2. Task/Inference Knowledge:
      a) The objectives of an app, together with how to achieve these.
      b) Tasks>Sub-tasks>Elementary inferences.
      c) Therefore, a task is composed of a number of combined inferences, documented in an inference diagram.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by *Information***
Communication model	Agent	Task	Sub-task	Communication model	
			Elementary inferences	Knowledge model	
				– Domain	
				– Task/ inference	

E.2.6 Method: DCMF (KM3) [6]

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Model element:
   A. Primary active object, as well as objects that are part of actions.
   B. Can be:
      1. EntityType.
      2. RoleinAction.
      3. RoleInOrganizationType.
      4. Action Type.
II. Attribute:
   A. Describes an optional, measurable characteristic of a model element.
III. State:
   A. Set of attributes with values associated with a model element.
   B. Specifies conditions under which an activity starts and ends.
IV. Rules:
   A. Description of changes to model elements.
   B. Rules are pairs:
      1. Activity role.
      2. Atomic formula:
         a) Statement about the state or attributes of a role.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	An *Actor*	Actuates *Events*	Composed of *Activities*	Directed by *Controls*	Richly Described by *Information*
	Element		Activity	State	Attribute
				Rules	

E.2.7 Method: DCMF (M2CM) [7]

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Meta data:
   A. Provides information regarding their potential for reuse in extending and creating conceptual models:
      1. POC (Point Of Contact):
         a) Holds information about an organization or a person having a particular role with respect to Conceptual Modeling.
      2. Model Identification:
         a) Accommodates information related to the identification of Conceptual Modeling:
            (1) Name.
            (2) Type.
            (3) Version.
            (4) Modification Date.
            (5) Security Classification.
            (6) Release Restriction.
            (7) Purpose.
            (8) Application Domain.
            (9) Description.
            (10) Use Limitation.
      3. Use History:
         a) Describes how Conceptual Modeling was used.
      4. Reference:
         a) Pointer to additional sources of info:
            (1) T. ex., Locations in XML documents.
            (2) References to ontologies.
      5. Implementation Dependencies:
         a) A log of all dependencies determined during Conceptual Modeling development:
            (1) T. ex., Domain ontologies.
      6. Key Word:
         a) Contains information about keywords (used for searching).
      7. Glyph:
         a) Contains an image of Conceptual Model.
         b) Sub-parts:
            (1) Notation.
            (2) Views.
            (3) Dynamic.
            (4) Static.

II. Static:
   A. Role could possibly be filled using DCMF-O, in particular JC3EIDM:
      1. Action:

a) Describes military actions on the lowest granular level or higher level aggregation.
2. Context:
   a) Context in which action occurs.
3. Reporting:
   a) Contains all data and information.
4. Rule of Engagement:
   a) Associated with actions.
5. Candidate-Target List:
   a) Associated with actions.
6. Objects:
   a) Type:
      (1) Static and persistent.
   b) Item:
      (1) Dynamic and likely to change over time.
7. Action Capability:
   a) Capability of an object to perform a function or achieve an end.
8. Location:
   a) Specific place for any item in the sphere of operations.
   b) Geometric shapes needed to plan, direct, and monitor operations:
      (1) Related to object-item concept.
      (2) T. ex, Boundaries.
      (3) Corridors.
      (4) Restricted Area.
      (5) Minefields.
9. Affiliation:
   a) All objects possess an affiliation:
      (1) T. ex Political Nation.
      (2) Ethnic group.

III. Dynamic:
   A. Role could possibly be filled using DCMF-O, in particular BOM++:
      1. Captures activities, actions, and decisions performed by the atomic knowledge components of the Conceptual Model.
   B. Pattern of Interplay:
      1. Represented by one or more pattern actions needed to accomplish a specific purpose or capability.
   C. State Machine:
      1. Describes the possible states of the conceptual entities as well as the transitions between them.
   D. Entity Type:
      1. Provides a mechanism for describing the types of conceptual entities used to represent senders and receivers identified within a Pattern of Interplay and to carry out the role of conceptual entities identified within the state machine.
   E. Event Type:
      1. Provides a mechanism for describing the types of conceptual events used to represent and carry out:
         a) Pattern actions.
         b) Variations.
         c) Exceptions.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by *Information***
State machine	Entity type	Event type	Pattern of interplay	Ontology stack	Meta data
				(Action) (Context) (Reporting) (Rules of Engagement) (Candidate-Target List) (Objects) (Action Capability) (Location) (Affiliation)	(POC) (Use history) Reference (Implementation Dependencies) (Key word) (Glyph) (Model ID) – Name – Type – Version – Modification date – Security Classification – Release restriction – Purpose – Application domain – Description – Use limitation

E.2.8 Method: DRDC [8]

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Elements:
   A. Entities.
   B. Entity Attributes.
   C. Functions/Actions.
   D. Relationships/Interactions.
II. Assumptions.
III. Functions/Actions.
IV. Constraints (restraints) / Bounds / Limitations (restrictions).
V. Domain Specific Algorithms.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	An *Actor*	Actuates *Events*	Composed of *Activities*	Directed by *Controls*	Richly Described by *Information*
Functions/ actions	Elements			Entity attributes	Attributes
				Relationships/ interactions	Assumptions
				Constraints/ bounds/ limitations	Domain specific algorithm

E.2.9 Method: Entity Relationship [9]

Entity Relationship perspectives are characterized by the following:

- Role of an entity in a relationship is the function that it performs in the relationship.
- The information about an entity or a relationship is obtained by observation or measurement, and is expressed by a set of attribute-value pairs.
- An attribute can be formally defined as a function which maps from an entity set or a relationship set into a value set or a Cartesian product of value sets.
- Another conceptual approach is provided by Entity-Relationship (ER) Modeling (ERM) Although ER models can be useful once the design process is finished, they are less suitable for formulating, transforming or evolving a design. ER diagrams are further removed from natural language, cannot be populated with fact instances, require complex design choices about attributes, lack the expressability and simplicity of a role-based notation for constraints, hide information about the semantic domains which glue the model together, and lack adequate support for formal transformations. Many different ER notations exist that differ in the concepts they can express and the symbols used to express these concepts. For such reasons we prefer ORM for conceptual modeling. In addition to ORM, VEA supports IDEF1X (a hybrid of ER and relational modeling) as a view of ORM.

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Entity:
   A. A “thing” which can be distinctly identified.
II. Relationship:
   A. An association among entities.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	An *Actor*	Actuates *Events*	Composed of *Activities*	Directed by *Controls*	Richly Described by *Information*
	Entity			Attribute/ value pairs	Attribute/value pairs
				Relationships	

E.2.10 Method: KAMA [10]

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Core meta-model (KAMA Foundation):
   A. KAMA Behavior:
      1. Missions.
      2. Tasks.
      3. Activities.
      4. Events.
      5. States.
      6. Data Items.
      7. Command/Control Units.
   B. KAMA Relationships.
   C. KAMA State Machine:
      1. Expresses the behavior of a conceptual model entity.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by *Information***
KAMA state machine			KAMA behavior	KAMA behavior	Attribute/value pairs
				KAMA relationships	KAMA relationships

E.2.11 Method: Mind Maps [11]

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Central focus of an image or graphic representation of the problem or information being mapped is placed in the center of a page (BOI's or Basic Ordering Ideas):
   A. Associate and connect ideas with lines, arrows, and symbols:
      1. Ideas are allowed to flow freely without judgment.
      2. Key words are used to represent ideas.
      3. One key word is printed per line.
      4. Key word ideas are connected to the central focus with lines.
      5. Color is used to highlight and emphasize ideas.
      6. Images and symbols are used to highlight ideas and stimulate the mind to make other connections.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	An *Actor*	Actuates *Events*	Composed of *Activities*	Directed by *Controls*	Richly Described by *Information*
N/A	N/A	N/A	N/A	N/A	N/A

E.2.12 Method: Topic Maps [12]

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Topics:
   A. Are related to each other by associations, which are typed n-ary combinations of topics:
      1. Associations are the general form for the representation of relationships between topics in a topic map. An association can be thought of as an n-ary aggregate of topics. That is, an association is a grouping of topics with no implied direction or order, and there is no restriction on the number of topics that can be grouped together.
   B. May also be related to any number of resources by its occurrences.
   C. Represents information using:
      1. Associations (representing relationships between topics):
         a) Each topic involved in an association has a role type.
      2. Occurrences (representing information resources relevant to a particular topic):
         a) Used to represent or refer to information about a concept represented by a topic.
         b) Can be used either to store string data within the topic map, or to reference any kind of Web-addressable resource external to the topic map.

II. Topics (representing any concept):
   A. A topic is a machine-processable representation of a concept:
      1. Topics have:
         a) Four principal forms of identity.
         b) Can have zero or more of each of these forms of identity, and thus can be identified within a topic map system by a number of different ways:
            (1) Identity as a topic resource in a serialized topic map:
               (a) When a topic map is represented in a serialized form for interchange, each topic is assigned a URI identifier that is unique across that topic map.
               (b) These URIs are used principally for deserializing references between topics.
               (c) Such identifiers are referred to as source locators.
            (2) Identity as a human-readable label:
               (a) Names act as labels for human consumption:
                  (i) Can be either text or a reference to some non-textual representation (for example, an icon, a sound clip, an animation clip, and so on.).
                     (a) Can have any number of topic names.
                  (ii) The scope mechanism allows for the case of homonyms (where a single word is used to refer to two or more different concepts).
            (3) Identity by reference:
               (a) When a topic is used to represent a resource that already has its own unique URI, that URI can be used as part of the identity of the topic.
               (b) Known as a subject locator in the topic map standard.
            (4) Identity by description:
               (a) Topics can be used to represent a concept that does not have its own unique URI:

(i) Many of the things that a topic can represent could never have a unique URI because they are not things that a computer can resolve a reference to:
(a) For example, a person may have any number of database records about himself or online biographies or pictures, but none of those addressable resources are the person – they are merely some form of descriptor for the person.
(ii) The resource that the subject identifier resolves to is known as a subject indicator.
(b) Subject identifiers (URIs) identify the subject the topic is about:
(i) The key difference between a subject identifier and subject locator is that a subject identifier requires human interpretation of a resource to determine the concept that a topic represents, whereas a subject locator simply points to the concept that the topic represents.

III. Scope:
A. Used in the topic map standard to refer to a constraint or a context in which something is said about a topic. The way in which such statements about topics are made is by adding a name to the topic; specifying an occurrence for a topic; or creating an association between topics (in which case the statement applies to all of the topics in the association):
1. Can be attached to any name, occurrence, or association in a topic map to qualify a statement, but still express it.
2. Is defined by a collection of topics that can be assigned to a name, an occurrence, or an association. The default scope (where no set is assigned) is known as the unconstrained scope and simply means that the name, occurrence, or association is always valid.

IV. Topic Merging:
A. In any given topic map, each subject described by the topic map must be represented by one and only one topic in the topic map. This means that it is the responsibility of the topic map processor to attempt to identify the situation in which two topics represent the same subject and to process them so that only one topic remains. This is the process of merging.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by *Information***
N/A	N/A	N/A	N/A	N/A	N/A

E.2.13 Method: Operational Conceptual Modeling Language (OCML) [13]

Operational Conceptual Modeling Language has the following attributes:

- Description at: http://technologies.kmi.open.ac.uk/ocml/;
- Allows the specification and operationalization of functions, relations, classes, instances and rules; and
- Includes mechanisms for defining ontologies and problem solving methods, the main technologies developed in the knowledge modeling area.

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Relations:
   A. Allow the OCML user to define labeled n-ary relationships between OCML entities.
II. Functions:
   A. Defines a mapping between a list of input arguments and its output argument.
III. Classes.
IV. Instances.
V. Rules:
   A. Forward: forward rule comprises zero or more antecedents and one or more consequents.
   B. Backward.
VI. Procedures:
   A. Define actions or sequences of actions which cannot be characterized as functions between input and output arguments.
VII. Constructs:
   A. Functional:
      1. Specifies an object in the current domain of investigation:
         a) Can be:
            (1) Constant.
            (2) Variable.
            (3) String.
            (4) Function application.
            (5) Can also be constructed by means of a special term constructor.
   B. Control terms:
      1. Specify actions.
      2. Describe the order in which actions are executed.
      3. Can be expressed as:
         a) Sequential.
         b) Iterative.
         c) Conditional control structures.
   C. Logical expressions.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by *Information***
	Classes		Procedures	Relations	
	Instances			Functions	
	Functional construct			Rules	
				Control terms	

E.2.14 Method: Object Process Methodology (OPM) [14]

See for reference: http://www.opcat.com/products_opm.htm.

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Object Process Diagrams (OPD):
   A. Formal graphic model.
II. Object Process Language (OPL):
   A. Natural language (English sub-set/controlled vocabulary) generated in RT in response to human input in OPD.
   B. OPD are developed and then OPL is generated from it.
III. Three building blocks:
   A. Objects:
      1. Things that exist.
      2. Physical or informational things:
         a) What are the states of the object?
         b) Which processes create, destroy, and modify the object?
         c) Which processes does it instrument?
   B. Processes:
      1. Things that transform objects by changing their states or creating or consuming objects.
      2. Processes are peers of Objects.
      3. Objects may contain processes and processes may contain objects:
         a) Which objects instrument and initiate the process?
         b) Which of the three functionalities does it fulfill: create, destroy, or modify the states of an object?
   C. States:
      1. Lower level entities since states are expressed inside of objects:
         a) What object does it belong to?
         b) Which process triggered the change of the state?
         c) What are the source and destination states?
IV. Two parts of OPM ontology:
   A. Entities.
   B. Links, which can be:
      1. Structural – express static, time-independent relations between pairs of entities.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by *Information***
	Object	Processes	Processes	States	

E.2.15 Method: Object-Role Modeling (ORM) [15]

Object Role Modeling may be characterized as follows:

- Early versions of object role modeling were developed in Europe in the mid-1970s (for example, binary relationship modeling and Natural Language Information Analysis Method (NLIAM)). The version discussed here is based on the author's formalization of the method, and incorporates extensions and refinements arising from research conducted in Australia and the United States. The associated language Formal Object-Role Modeling Language (FORML) is supported in Microsoft® Visio® for Enterprise Architects (VEA), part of Visual Studio® .NET Enterprise Architect.

- The information system's life-cycle typically involves several stages: feasibility study; requirements analysis; conceptual design of data and operations; logical design; external design; prototyping; internal design and implementation; testing and validation; and maintenance. ORM's Conceptual Schema Design Procedure (CSDP) focuses on the analysis and design of data. The conceptual schema specifies the information structure of the application: the types of fact that are of interest; constraints on these facts; and perhaps the derivation rules for deriving some facts from others.

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by *Information***
					Conceptual schema

E.2.16 Method: UML [16]

Properties educed from examination of the standard in accordance with the strategy outlined in Section E.1 above include the following:

I. Structure Diagrams:
   A. Class Diagram:
      1. Shows how the different entities (people, things, and data) relate to each other.
   B. Object Diagram.
   C. Component Diagram:
      1. A physical view of the system, meant to show the dependencies that the software has on the other software components.
   D. Composite Structure Diagram.
   E. Package Diagram.
   F. Deployment Diagram:
      1. Shows how a system will be physically deployed in the hardware environment.
II. Behavior Diagrams:
   A. Use Case Diagram:
      1. Illustrates a unit of functionality provided by the system.
   B. Activity Diagram:
      1. Show the procedural flow of control between two or more class objects while processing an activity.
   C. State Machine Diagram:
      1. Models the different states that a class can be in and how that class transitions from state to state.
III. Interaction Diagrams:
   A. Sequence Diagram:
      1. Shows a detailed flow for a specific use case or even just part of a specific use case.
   B. Communication Diagram.
   C. Timing Diagram.
   D. Interaction Overview Diagram.
IV. Other important concepts:
   A. A UML profile is a specification that does one or more of the following:
      1. Identifies a sub-set of the UML meta-model.

2. Specifies “well-formedness rules” beyond those specified by the identified sub-set of the UML meta-model.
3. “Well-formedness rule” is a term used in the normative UML meta-model specification to describe a set of constraints written in UML’s Object Constraint Language (OCL) that contributes to the definition of a meta-model element.
4. Specifies “standard elements” beyond those specified by the identified sub-set of the UML met-model.
5. “Standard element” is a term used in the UML meta-model specification to describe a standard instance of a UML stereotype, tagged value or constraint.
6. Specifies semantics, expressed in natural language, beyond those specified by the identified sub-set of the UML meta-model.
7. Specifies common model elements, expressed in terms of the profile.

B. Stereotype:
1. A stereotype defines how an existing meta-class may be extended, and enables the use of platform or domain specific terminology or notation in place of, or in addition to, the ones used for the extended meta-class.
2. A stereotype denotes a variation on an existing modeling element with the same form but with a modified intent. Stereotypes are effectively used to extend the UML in a consistent manner.

Language Unit	Purpose
Actions	(Foundation) modeling of fine-grained actions
Activities	Data and control flow behavior modeling
Classes	(Foundation) modeling of basic structures
Components	Complex structure modeling for component technologies
Deployments	Deployment modeling
General Behaviors	(Foundation) common behavioral semantic base and time modeling
Information Flows	Abstract data flow modeling
Interactions	Inter-object behavior modeling
Models	Model organization
Profiles	Language customization
State Machines	Event-driven behavior modeling
Structures	Complex structure modeling
Templates	Pattern modeling
Use Cases	Informal behavioral requirements modeling

A table compiling the canonical allocation of conceptual components evident in the standard to the vernacular conceptual model characteristics reserved words, therefore, is:

Using *Behaviors*	**An *Actor***	**Actuates *Events***	**Composed of *Activities***	**Directed by *Controls***	**Richly Described by *Information***
General behaviors	Classes		Actions	Behavior diagrams – Use case diagram Activity diagram State machine diagram	Structure diagram – Class diagram
Interactions			Activities	Interaction diagrams – Sequence diagrams – Communications diagrams – Timing diagram – Interaction overview diagram	
State machines				Profiles	
				Stereotypes	
				Use cases	

E.3 REFERENCES

[1] Simulation Interoperability Standards Organization, Guide for Base Object Model (BOM) Use and Implementation, SISO-STD-003.1-2006, 31 March 2006. http://www.boms.info/standards.htm, Retrieved 18 September 2009.

[2] Mojtahed, V., Andersson, B., Kabilan, V. and Zdravkovic, J., "BOM++, a Semantically Enriched BOM", Spring Simulation Interoperability Workshop, 2008.

[3] White, S., "Introduction to BPMN", IBM Corporation, 2006.

[4] Karr, D., "Conceptual Modeling in OneSAF Objective System (OOS)", http://tinyurl.com/natrw6, 2005.

[5] Schreiber, G., Akkermans, H., Anjewierden, A.A., de Hoog, R., Shadbolt, N.R., Van de Velde, W. and Wielinga, B.J., "Knowledge engineering and management: The Common KADS methodology", Cambridge Massachusetts, MIT Press, 2000.

[6] Mojtahed, V., Garcia Lozano, M., Swan, P., Andersson, B. and Kabilan, V., "DCMF – Defence Conceptual Modeling Framework", Technical Report FOI-R–1754-SE, Swedish Defence Research Agency, ISSN 1650-1942. 2005.

[7] Mojtahed, V., Tjörnhammar, E., Zdravkovic, J. and Khan, A., "The Knowledge Use in DCMF", Technical Report FOI-R–2606-SE, Swedish Defence Research Agency, ISSN 1650-1942. 2008.

[8] Topçu, O., "Development, Representation, and Validation of Conceptual Models in Distributed Simulation", DRDC Atlantic TM 2003-142, Defence R&D Canada – Atlantic, 2003.

[9] Chen, P., "The Entity-Relationship Model – Toward a Unified View of Data", ACM Transactions on Database Systems, Vol. 1, No. 1, March, pp. 9-36, 1976.

[10] Karagöz, A., "A Framework for Developing Conceptual Models of the Mission Space for Simulation Systems", Unpublished doctoral dissertation, Middle East Technical University, Ankara, 2008.

[11] http://mindmappingsoftwareblog.com/basic-ordering-ideas/, Retrieved 24 September 2009.

[12] Ahmed, K. and Moore, G. (2005): An Introduction to Topic Maps, Microsoft Corp., http://www.topicmaps.org/, Retrieved 24 September 2009.

[13] Domingue, J., Motta, E. and Garcis, O., "Knowledge Modelling in WebOnto and OCML: A User Guide", v. 2.4, Open University and Knowledge Media Institute, 1999.

[14] Liu, H. and Gluch, D., "Conceptual Modeling with the Object-Process Methodology in Software Architecture", JCSC No. 19 V. 3, 2003.

[15] Halpin, T., "Object-Role Modeling (ORM/NIAM)", found in Handbook on Architectures of Information Systems, Eds. P. Bernus, K. Mertins and G. Schmidt, Springer-Verlag, Berlin, Germany, 1998.

[16] http://www.uml.org/, Retrieved 2 October, 2009.