Enterprise and system architecting with International Standards:
An INCOSE International Symposium 2008 tutorial

Richard A. Martin
Tinwisle Corporation
Convener
ISO TC 184/SC 5/WG 1

Johan Bendz
Swedish Defense Material Admin.
Convener
ISO/IEC JTC 1/SC 7/WG 42

Abraham Meilich
Lockheed Martin
INCOSE Liaison
ISO TC 184/SC 5/WG 1

Stuart Arnold
QinetiQ
INCOSE Liaison
ISO/IEC JTC 1/SC7/WG 42
Why are we here?

Architecture is a people thing we often associate with design intelligence. We believe all built systems have it. Understanding architecture helps us understand better the world we build.

Standards are a people thing as well. Standards can be an architectural pattern. We believe standards help build better systems for use in the real world.
What will we accomplish?

Promotion of architecture as an aid in communication among people to enable better understanding of systems

Identify the prominence of people in architecture and the systems they build for enterprises

Use of standards from ISO/IEC JTC1/SC7/WG42 for architecture practice in general and from ISO TC184/WC5/WG1 for architecting enterprise systems in particular
Tutorial Approach

• Frame the ISO TC184/SC5/WG1 architectural standards in an enterprise context
• Frame the ISO/IEC JTC1/SC7/WG42 architecture description standard in a system context
• Convey the standards with significant detail, often with content from related standards
• Discuss the interpretation of the standards to craft architectural artefacts
Architectural Discord

Discussions about architecture of any kind are metaphor-rich and consequently subject to much misinterpretation.

Since architecture is pervasive (in good, bad, and ugly forms), context is critical.

The enterprise context ranges from two people in a garage through tens of thousands cooperating in a global supply chain to millions governed by nation states.
Architectural Harmony

Common concepts - we are all doing very similar things even though we are not using the same words or methods. Those concepts are independent of methodology and range across many domains of practice.

We recognize the value of “good” architecture and want to encourage its use.
Architectural Value

Architecture is the means by which enterprise mission and objectives are represented.

Architecture representations are critical communication vehicles for managers and engineers, especially in large, complex, and dispersed enterprises and programs.

Architectural standards facilitate completeness and consistency of representation.
Many Architectures

Many diverse perspectives result in many views and viewpoints about architecture - at least as many as lecturers on the topic - after all it is a human thing

Characterized with aspects for:
- Form of realization
- Functions to enable or perform
- Objective (or subjective) experience in use

Derived from principles and patterns

our focus
Dawn of Architecture

• **Marcus Vitruvius Pollio**, c. 90 - 20 B.C.E.
  - Architect for Augustus Caesar
• **de Architectura** - 30 B.C.E. in 10 volumes
• First complete text on architecture and the “standard” used for 1500 years
• Three aspects [public] architecture must possess:
  - **Strength** (firmitatis) [firmness]
  - **Utility** (utilitatis) [commodity]
  - **Beauty** (venustatis) [delight]
Architecture definitions

ISO 42010 (IEEE 1471) -

3.5 architecture: The fundamental organization of a system embodied in its components, their relationships to each other, and to the environment, and the principles guiding its design and evolution.

INCOSE IEWG Knowledge Claims -
Architecture concerns the arrangement of function and feature that maximizes the objective of the system structure.
My working favorite

ISO 15704:2000

3.2 architecture

a description (model) of the basic arrangement and connectivity of parts of a system (either a physical or a conceptual object or entity)

Observation - ISO/IEC 42010 is about architecture descriptions and ISO 15704 is a reference architecture description
Only two types

There are two, and only two, types of architectures that deal with enterprise integration

I. **system architectures** that deal with the design of a system, e.g. the computer control system part of an overall enterprise integration system;

II. **enterprise-reference architectures** that deal with the organisation of the development and implementation of a project such as an enterprise integration or other enterprise development programme.
Models and architecture

Architectural intent is embodied in enterprise models (EM).

Architectural realization is embodied in instance *manifestations* of those models.
Abstraction quiz

What is the next element of the sequence?

7853981633974483096156608...
How do we proceed?

• Tutorial part 1
  - Reference points
    • Liaisons and harmonization
    • Principles for EA and modeling
And continuing

• Tutorial part 2
  - ISO/IEC 42010:2007 Recommended Practice for Architectural Description of Software-Intensive Systems
    • Focus on requirements for describing architecture
    • Emphasis on accommodating stakeholder concerns
    • Revision extends description with concepts for correspondence among descriptive elements and containment in architecture frameworks
After the break

• Tutorial part 3
  - ISO 15704:2000 Industrial automation systems - Requirements for enterprise-reference architectures and methodologies
    • Under revision to incorporate material from ISO 14258
    • Includes annexes: Generalized Enterprise Reference Architecture and Methodologies (GERAM), Economic view and Decision view
Standard interoperation

- Tutorial part 4
    - Duality of enterprise and system
    - Stakeholder as consumer vs. producer
    - ISO 15704 and architecture description
    - ISO/IEC 42010 as enterprise artifact
  - Interoperation through standardization
    - Integration and unification architecture
    - Federation architecture
Harmonizing standards

The great thing about standards is that there are so many from which to choose.

Internationally, the interoperation of standards is a BIG concern.

Culture, language, and use domain are all barriers to interoperation.

Breaking down standard silos requires cooperation and accommodation.

- TC184/SC5/WG1 & JTC1/SC7/WG42
ISO's EA Groups

ISO/TC184/SC5/WG1: Developing enterprise architecture standards based on manufacturing industry for international architecture applications.


Related EA activity

ISO/TC184/SC4/WG3: Developing Data Standards including ISO 10303, 100s of standards including AP233

+ many other indirectly related efforts
# E-A standards

*Proposed new title*

ISO/IEC 42010 Systems and software engineering – Architecture description

| ISO 14258 Rules and Guidelines for Modelling |
| ISO 15704 Requirements for Enterprise Reference Architectures |
| (Needs for Frameworks, Methodologies, Languages, Tools, Models, Modules) |

<table>
<thead>
<tr>
<th>Frameworks</th>
<th>Languages</th>
<th>Modules</th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>CEN/ISO 19439</strong> Framework for Modelling</td>
<td><strong>CEN/ISO 19440</strong> Constructs for Modelling</td>
<td>ISO 16100 Mfg. Software Capability Profiling</td>
</tr>
<tr>
<td><strong>ISO 10746</strong> Ref. Model - ODP</td>
<td><strong>ISO/IEC 15414</strong> ODP Enterprise Language</td>
<td></td>
</tr>
<tr>
<td><strong>OMG MDA</strong> Model Driven Architecture</td>
<td><strong>ebXML</strong> Electronic Business using eXtensible Mark-up Language</td>
<td></td>
</tr>
</tbody>
</table>

SC5 & **WG1**
SC7 & **WG42**

K. KOSANKE and M. ZELM (CIMOSA Association)
D. CHEN (LAPS, University Bordeaux 1)
Architecture description

ISO/IEC 42010:2007 Recommended practice for architectural description of software-intensive systems

• Identifying and arranging products to document an architecture
• Distinguishes an architecture description from the mechanisms for description generation
Generalizing standards

ISO 15704:2000 - Requirements for enterprise-reference architectures and methodologies

- Merging of previous work - PERA, IEM, GRAI GIM, CIMOSA, and GERAM
- Presents principles for enterprise architecture
- Extends ISO 14258 concepts of model, view, life cycle, recursion, and iteration with life history and genericity
Unified model framework

ISO 19439 - Enterprise integration: Framework for enterprise modelling

• Based upon CEN ENV 40003:1990
• Objective is to further enable model based execution using enactable models
• Aligned with ISO 15704 (a GERA model)
• Articulates 3 dimensions of enterprise modeling as a framework: Phase, View, and Genericity
Modeling constructs

ISO 19440  Enterprise integration - Constructs for enterprise modelling

• Based upon CEN ENV 12204:1996
• Aligned with ISO 15704 (an EML artifact)
• Articulates modeling constructs for manufacturing automation
• Elaborates the CIMOSA Baseline example of 19439 with constructs
ISO 19440 (cont.)

- Constructs for enterprise modeling
  - common semantics enable model unification
  - usable across phases of model development
  - support process-oriented approach

- Arrangement and specialization using templates into structures for a specific purpose

- No mapping between functional operations and capabilities

- No explicit versioning mechanism
Standards reflect practice

WG1 standards reflect industrial modeling practice of the 1990’s
- Purdue Enterprise Reference Architecture
- Computer Integrated Manufacturing Open Systems Architecture
- Graphe a Resultats et Activite Inter-relies

WG42 standard reflects software intensive systems practice of the 1990’s

Do they harmonize with current practice?
- DoDAF/MoDAF, Zachman, TOGAF, Dual-Vee, etc., and other gallery members
### Zachman Framework

**Enterprise Architecture - A Framework**

<table>
<thead>
<tr>
<th>DATA</th>
<th>FUNCTION</th>
<th>NETWORK</th>
<th>PEOPLE</th>
<th>TIME</th>
<th>MOTIVATION</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scope (Contextual)</td>
<td>List of Things Important to the Business</td>
<td>List of Processes the Business Performs</td>
<td>List of Locations in which the Business Operates</td>
<td>List of Organizations Important to the Business</td>
<td>List of Business Goals/Strategy</td>
</tr>
<tr>
<td>Planner</td>
<td>Entity = Class of Business Thing</td>
<td>Function = Class of Business Process</td>
<td>Node = Major Business Location</td>
<td>People = Major Organizations</td>
<td>Time = Major Business Event</td>
</tr>
<tr>
<td>Enterprise Model (Conceptual)</td>
<td>E.g. Semantic Model</td>
<td>E.g. Business Process Model</td>
<td>E.g. Logistics Network</td>
<td>E.g. Work Flow Model</td>
<td>E.g. Master Schedule</td>
</tr>
<tr>
<td>System Model (Logical)</td>
<td>Node = Business Location</td>
<td>Link = Business Linkage</td>
<td>People = Organization Unit</td>
<td>Work = Work Product</td>
<td>Cycle = Business Cycle</td>
</tr>
<tr>
<td>Designer</td>
<td>Ent = Data Entity</td>
<td>Proc. = Application Function</td>
<td>People = Role</td>
<td>Work = Deliverable</td>
<td>End = Business Objective</td>
</tr>
<tr>
<td>Technology Model (Physical)</td>
<td>Rehn = Data Relationship</td>
<td>Link = Business Characteristics</td>
<td>People = Role</td>
<td>Action = Action Assertion</td>
<td>Means = Business Strategy</td>
</tr>
<tr>
<td>Sub-contractor</td>
<td>Ent = Field</td>
<td>Proc. = Computer Function</td>
<td>Node = Hardware/System</td>
<td>People = User</td>
<td>Time = Execute</td>
</tr>
<tr>
<td>Functioning Enterprise</td>
<td>Rehn = Address</td>
<td>Link = Data Format</td>
<td>Link = Line Specifications</td>
<td>Work = Screen Format</td>
<td>Cycle = Component Cycle</td>
</tr>
</tbody>
</table>

**Role by Interrogative Grid of Cells containing models of the enterprise. A proto-typical Framework!**

Zachman Institute for Framework Advancement - (810) 231-0531

Copyright - John A. Zachman, Zachman International

© Copyright 2008 by R. Martin, J. Bendz, S. Arnold, A. Meilich

INCOSE IS08 Tutorial June 2008
FRAT

Systems thinking - each of the FRAT views can be described with:

- Upper Level FRAT Data Provides Scope For Next Level
- Lower Level Must Roll-Up and Map to Upper Level
- Each Level of FRAT Establishes a BASELINE
- Simulation Models Provide Dynamic Views of FRAT

Function
Requirements
Answers
Test

Detail elaboration adds both depth and breadth to the system description

Aligning functional applications along axis to identify dimensions of the global manufacturing enterprise

Source: ISO/TC 184/SC5 N913, E. delaHostria, Chairman, and ARC Advisory Group (used with permission)

© Copyright 2008 by R Martin, J Bendz, S Arnold, A Meilich
Architectural Perspectives

Perspective Filters:
- Development & Verification
- Manufacturing & Production
- Storage & Transportation
- Installation & Deployment
- Simulation & Training
- Operational
- Maintenance & Support
- Disposal
- Project
- Safety
- Functional
- Physical
- Information/ Data Flow


Permission is hereby granted to use this copyrighted material for any lawful purpose.
Purdue Enterprise Reference Architecture

Source: T. J. Williams, A Handbook on Master Planning and Implementation for Enterprise Integration Programs, Institute for Interdisciplinary Engineering Studies, Purdue Univ.

© Copyright 2008 by R Martin, J Bendz, S Arnold, A Meilich  INCOSE IS08 Tutorial June 2008
CSC_PLM

Verband Deutscher Maschinen- und Anlagenbau - German Engineering Federation

© Copyright 2008 by R Martin, J Bendz, S Arnold, A Meilich

INCOSE IS08 Tutorial June 2008 35
DoDADF

Source: Architecture Working Group, C4ISR Architecture Framework
Version 2.0, 1997
DoDAF redux

DoDAF v2.0 Architecture Conceptual Data Model
Stair-like Architecture

Yuliu Chen and Li Qing
National CIMS Engineering and Research Centre
Department of Automation, Tsinghua University
Mooz and Forsberg

Dual Vee Model

Architecture Vee
for architecture management

Entity Vee
for entity management

Depicts architecture baseline evolution. Vertical dimension is architecture decomposition. Horizontal dimension is system realization. Third and normal dimension is quantity of entities and their interfaces.

The vertical dimension of the Architecture Vee is decomposition into levels of architecture entities which is based on the Product Breakdown Structure. Only three of the INCOSE seven levels of decomposition are illustrated.

The vertical dimension of the Entity Vee is extent of elaboration detail at the decomposition level of interest such as Subsystem or Lowest Configuration Item. The elaboration includes the baselines of concept, architecture, design-to, build-to, code-to, as-built, etc.
Federal Enterprise Architecture Framework
Key to the Puzzle

Master key of the Enterprise

Soumen Chatterjee, SDA Asia 2006
TOGAF

Architecture Development Method cycle with expansion

The Open Group Architecture Framework v8.1.1

http://www.opengroup.org/architecture/togaf8-doc/arch/
MDA

OMG Model Driven Architecture

http://www.omg.org/mda/
Synergetic efforts

Liaison
- a channel for communication between groups

Observation - a critical pattern within the architecture of international standards
Liaisons

Liaisons help to expand the reach for input and validation

Internal (within ISO, like SC4 & JTC1)

External (beyond ISO national bodies)

- IEC, CEN, etc.
- INCOSE, OMG, ISA, OASIS
- Experts from industry and academia

A place for anyone who wants to participate
INCOSE liaison

INCOSE Standards Tech Committee (STC)
- Coordinate INCOSE WG involvement, tasks, etc.
- STC Liaisons: JTC1/SC7 & TC184/SC5

Other INCOSE WGs involved:
- Architecture - Charles Dickerson
- Integration & Interoperability - John Nallon
- Model Based SE - Phil Spiby

INCOSE Connect webpage established
INCOSE INSIGHT articles (Jan & Apr 07)
INCOSE Liaison future

• Evaluating 15704/GERAM & 42010 through liaison participation
• Submitted initial comments for 15704 revision
• Submitted initial comments for 42010
• Participating in WG1 and WG42 meetings
• Terminology & definition issues
• Special EA workshop @ Jan '08 IW
• EA Panel at IS '08
Mindful of Principles

General Principles

1. Models are formal artifacts developed and used by people.

2. A complexity tradeoff exists between modeling medium and model instances in that medium.

3. Naming serves as the bridge between the formal and the human.

4. Separate model and instance decompositions - do not confuse meta-levels.

5. Dependency is not chronology

6. Don’t hide architecture in methodology.
More Principles

Framework Principles

7. Frameworks organize artifacts to facilitate understanding.

8. To improve quality, distinguish structure from connectivity.

9. Separate policy from mechanism.

10. Both grid (ordinant) and tree (decomposition) structures appear in models.

11. Scale dimensions include:
    - abstractness (abstract to concrete),
    - refinement (coarse to fine) and
    - scope (general to special)
More Principles - 2

12. Within a framework, use of components are driven along one ordered dimension.
13. Along this ordered dimension, all prior context is relevant.
14. Refinement is recursive using iteration.
15. Connections can be of arbitrary arity.
16. Views are important in standards and methodologies.
17. Views are used both to “see” contents and to “create” contents.
18. Separate model and instance constraints.
Scope & Meta-confusion

- **Entity Relationship Model space**
  - Data space / instance
  - World space / thing
  - Concept space / meta-model

- **Relationship**
  - abstract
  - concrete

- **Department**
  - works for

- **Employee**
  - IsA (IsA subtype)

- **Party**

- **Data**
  - Did  Pid
  - Pid  PName

- **World**
  - Did  DName
  - 35  Accounting
  - 95  Manufacturing

- **Context**
  - Reference
  - 42010
  - 15704
  - Harmony

- **Joe**
  - Manufacturing Dept.
  - Joe

- **Table**
  - Did | DName
    - 35  Accounting
    - 95  Manufacturing

- **Table**
  - Did | Pid
    - 35 | 95
    - 95 | 47

- **Table**
  - Pid | PName
    - 47 | Joe
Distinguish structure from connectivity

NY Div.

- Sales
  - Mrktng.
    - Inside
      - Outside
  - Cntrcts
  - Buyer
    - Cell A
      - Cell B
        - Cell C
    - Line 01
      - Shpng
        - Pkng
          - Logistic
      - Rcving
    - Procur.
    - Manuf.
    - Wrhse.
Two structural aspects

Ordinant

Decomposition
Purposeful dimensions

Zachman: **Role**

{Context, Owner, Designer, Builder, Out-of-context}

ISO 19439: **Model Phase**

{Domain, Concepts, Requirements, Design, Implementation, Operation, Decommission}

ISO 15288: **Process Group**

{Agreement, Enterprise, Project, Technical}

C4ISR/DoDADF: **Guidance**

{Focus, Scope, Characterize, Determine, Build, Use}
Recursive refinement
cf. ISO 15288
Views

Views are for communication and analysis.

A static collection of views is insufficient.

Views exist at all meta-levels.

View of structure is meta with respect to view of data.

View update often crosses meta-levels.

Standards sometimes specify a view using a viewpoint.
Observation points
Terminology wars

No one speaks the way you do
- Different training
- Different disciplines
- Different customs
- Different translations

A large global upper ontology is a myth
Local taxonomies and meaningful phrases are achievable standards

Precision is more important than recall
ISO 42010:2007
(formerly IEEE 1471)

- Developed by the IEEE’s Architecture Working Group under the sponsorship of the Software Engineering Standards Committee of IEEE
- Effort began in 1995 with large working group and reviewer group
- Focus on best practices and a vocabulary for architecture concepts
Whence cometh 42010

- Fast track approval by ISO/IEC in 2007 as 25961 and relabeled 42010 at time of publication
- Under revision to harmonize with JTC1 standards and other ISO standards
ISO 42010 Scope

• Expression of the system and its evolution

• Communication among the system’s stakeholders

• Planning, managing, and executing the activities of system development

• Planning, managing, and effective utilization of a system’s elements and resources throughout its life cycle*

* Added to scope for revision draft
More 42010 Scope

- Evaluation and comparison of system architectures in a consistent manner
- Expression of the persistent characteristics and governing principles of a system to guide acceptable change
- Verification of a system's implementation for compliance with an architectural description
- Recording contributions to the body of knowledge of systems and software architecture.
Big ideas from 1471

1. Architecture exist to satisfy known concerns from stakeholders
   - Ensures architecture and its description are relevant
   - Stakeholder concerns, often non-functional, drive the architecture

2. Architecture Descriptions are inherently multi-view
   - No single view addresses all concerns
   - A view should cover the entire system
3. Viewpoints ('what to describe') are separate from Views ('this description')

- Represents current practice with 'viewpoint sets'
- Ensures consistency and repeatability, particularly when evaluating alternative architectures
- Supports development of architecture tools, techniques and methods
Limits of 42010

• All about (don’t extend expectations)
  - a single architecture description of
  - a single architecture of
  - a single system

• No specification of notation, format or media

• No required content of an architecture description reflecting current practice and consensus

• Conformance to the standard is with respect to a point in time
Conceptual space

- environment
- system
- architecture
- stakeholder
- mission

- influences
- situated in
- fulfills 1..*
- has 1..*
- has 1..*
Current AD model

ISO 42010 Figure 1 (partial)
A more robust model

Conceptual model of architectural description in ISO 42010 (IEEE 1471) [Martin CMap version]

This needs to be updated for emerging model from Berlin
42010 requirements

• **Stakeholders** *(relevant to architecture)*
  - Architectural description must explicitly identify the system’s stakeholders
  - Two key stakeholder roles are acquirer and architect

• **Concerns** *(relevant to architecture)*
  - Interest of stakeholder in system development, operation or other aspect
  - Include system consideration such as performance, reliability, security, distribution, and evolvability
  - Drive viewpoint selection
42010 viewpoints

- Viewpoints are first-class, i.e., they are 'declared' before use
- May originate in an AD or elsewhere
- Establishes the conventions by which a view is created, depicted and analyzed
- Determines languages and associated modeling methods
- AD includes rationale for a viewpoint
- Each stakeholder & concern is addressed by at least one viewpoint

People may expect a different notion of "viewpoint"
42010 models

• Viewpoint specifies model elements
• Architectural models are developed using the methods established by associated viewpoint(s)
• Models have many forms and manifestations - physical, logical, etc.
• Architectural models are the constituents of architectural views
• May participate in more than one view

Importance elevated in revision version
42010 views

• The stakeholder perspective of architecture (like Zachman)
• Each architectural view spans the whole system of interest with respect to one or more concerns
• May consist of one or more kinds of architectural model
• The AD is composed of architectural views and supporting information, particularly rationale
42010 correspondence

- Each view expresses exactly one viewpoint
- The set of views in an architecture description corresponds to a complete allocation of concerns
- AD should contain analysis of consistency across all of its architectural views

Revision version adds notion of correspondence rules between views and between viewpoints
Digging Deeper

Context Reference
42010
15704 Harmony

Stakeholder → has interest in → System/Enterprise
is characterized by
is property of
is a primary member of

Concern(s)* → (n:m) is subset of
is evaluator of
(1:n) is selector of

Architecture → (1:n) is described by
is a primary member of

An Architecture Description Document

Architecture Description Identification
(1:1) is identifier for

Architecture Concern(s)
(1:1) is expression of whole system/enterprise with respect to
(1:1) is frame of reference for

An Architecture View
(1:n) is arrangement of

An Architecture Viewpoint
(1:1) is specification for
(n:1) are constituents of
(1:n) are composed in accordance with

Architecture Model(s)

Architecture Modeling Constructs and Methods

An Architectural View

Architecture Description
(1:n) is reason for selection of

An Architecture Description Rationale

An Architecture Description
(1:n) is reason for choice made during creation of

© Copyright 2008 by R Martin, J Bendz, S Arnold, A Meilich
INCOSE IS08 Tutorial June 2008
75
42010 frameworks

- Architecture framework identifies architecture-related concerns, stakeholders holding those concerns, and one or more architectural viewpoints that frame those concerns
- May define viewpoint correspondence rules to relate its viewpoints
- Any architecture framework meta-model shall reflect the 42010 Core Model
- AD conforms to AF if and only if AD views correspond to AF viewpoints
ISO 42010 Conformance

Conformance requires that:
- For each architecture description, all views, viewpoints, concerns and stakeholders are properly identified
- Each concern is addressed by at least one viewpoint
- Each view corresponds to exactly one viewpoint
- Models consistent with the viewpoint compose the view associated with that viewpoint
- Inconsistencies between views are explicit
- Rationale for architectural choices is provided
Extent of architecture

• No agreement on what is an architectural concern and what is not - it is situation and purpose dependent

• Certainly more than simple allocation of form and function - recall _venustatis_ - and thus the focus on stakeholder concerns as _fit for intended use_

• Architecture is the trade-off space for requirements while engineering is the trade-off space for implementation

(Don’t confuse titles with tasks - it’s all design)
ISO 14258:1998

Adopted after 14 year effort
- Over 300 documents reviewed
- Focus on manufacturing systems architecture
- Lead by National Institute of Standards and Technology - USA

A consensus predicate to on-going reference model effort

10 basic definitions (some still contentious)
Systems theory aspects

structural - elements have multiple interdependencies leading to emergent qualities

behavioral - identification of variable and functional relationships

hierarchical - systems within systems and levels of abstraction embodied in emergent qualities
Levels of abstraction

Lower levels reveal detail and the means to achieve purpose - more concrete
Higher levels reveal the role of system within environment - more abstract
Each level has structure and behavior

Observation - enterprise architecture (EA) is manifest in enterprise models (EM) as a pattern. Constraints on the EM are the EA.
Need for life-cycle

**EM shall** address what happens to the factors of production (such as people, capital, material, information, energy, and tools) during the phases of the enterprise or product life-cycle.

- Products, processes, projects, and enterprises are systems.
- Systems have a life cycle that can be partitioned into phases such as plan/build, use/operate, and recycle/dispose.
Broad model scope

**EM shall define relevant aspects of the enterprise necessary to**

- conceive, design, procure for, and construct an enterprise consisting of any set of related chosen processes
- manage and operate an enterprise so that it can meet its objectives
- support an enterprise to modify, redesign, dismantle and rebuild it
Must be accessible

As architectural representations of enterprises, models shall exhibit syntax and semantics so that contents of the model are understandable to human users.

- The syntax of a model refers to the permissible kinds of relations.
- The semantics of a model encompass the meanings of the elements and relations with respect to enterprise-model concepts.
## ISO 14258:1998 Figure 1

<table>
<thead>
<tr>
<th>Phase</th>
<th>“What” Activities</th>
<th>“How” Activities</th>
<th>“Do” Activities</th>
</tr>
</thead>
</table>
| Plan and Build Phase                       | • Develop goals  
• Define strategy  
• Define product needs                                                                 | • Develop Requirements  
• Define concept  
• Design product  
• Plan to produce product  
• Plan to support product                                                              | • Procure parts  
• Produce product  
• Test product  
• Ship product                                                                         |
| Use and Operate Phase                      | • Define support needs  
• Define Use                                                                              | • Define Use Requirements  
• Define Support Requirements                                                             | • Use the product  
• Support product                                                                          |
| Dispose and Recycle Phase                  | • Define recycle/dispose needs                                                   | • Define recycle/dispose requirements                                              | • Recycle product  
• Dispose product                                                                       |
## Another way to view it

### ISO 14258:1998 Figure 1 Transposed

<table>
<thead>
<tr>
<th>Phase</th>
<th>Plan and Build Phase (e.g., before sell/buy title transfer)</th>
<th>Use and Operate Phase (e.g., after sell/but title transfer)</th>
<th>Dispose and Recycle Phase (e.g., after product is no longer useful)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Specify</td>
<td>“What” Activities</td>
<td>“How” Activities</td>
<td>“Do” Activities</td>
</tr>
<tr>
<td>Issue-solving activities</td>
<td>w</td>
<td>How</td>
<td>w</td>
</tr>
<tr>
<td>Specify</td>
<td>“What” Activities</td>
<td>“How” Activities</td>
<td>“Do” Activities</td>
</tr>
<tr>
<td>Design</td>
<td>“How” Activities</td>
<td>“Do” Activities</td>
<td>“Dispose and Recycle Phase”</td>
</tr>
<tr>
<td>Design</td>
<td>“How” Activities</td>
<td>“Do” Activities</td>
<td>“Dispose and Recycle Phase”</td>
</tr>
<tr>
<td>Build Operate</td>
<td>“Do” Activities</td>
<td>“Use the product”</td>
<td>“Recycle product”</td>
</tr>
<tr>
<td>Build Operate</td>
<td>“Do” Activities</td>
<td>“Use the product”</td>
<td>“Recycle product”</td>
</tr>
</tbody>
</table>

- **Specify “What” Activities**
  - Develop goals
  - Define strategy
  - Define product needs

- **Design “How” Activities**
  - Develop Requirements
  - Define concept
  - Design product
  - Plan to produce product
  - Plan to support product

- **Build Operate “Do” Activities**
  - Procure parts
  - Produce product
  - Test product
  - Ship product

- **Operate Phase**
  - Use the product
  - Support product

- **Dispose and Recycle Phase**
  - Recycle product
  - Dispose product
Life-cycles of systems

Different life-cycle phases may have different models.

These models shall have the capability to interoperate where it has been determined that processes need to communicate with each other.

- Feeding modeled information forward and backward in life-cycle activities enables value-added iteration of enterprise processes that improves product quality.
Recursion (structural)

The What, How, and Do activities are recursive and decomposable. Activities can be divided into sub-activities, and these sub-activities will consist of another set of W, H, and D activities.

```
  W1
   ▼
   W21
   ▼
   W231
   ▼
   W231

  H2
   ▼
   H22
   ▼
   H32
   ▼
   H32

  D3
   ▼
   D23
   ▼
   D23
```

Context Reference
42010
15704
Harmony
Enterprise operations can model new enterprises either from its own particular models or using reference constructs and partial models.
Sub-activities interact

Sub-activities may be represented by different types of models. These models shall be able to interoperate where it has been determined that these sub-activities need to communicate with each other.

Observation - recall the distinction between structure and connection where both are relationships.
Iteration (behavioral)

The W, H, and D activities are iterative.

- There is no fixed sequence of activities.
- It is possible to return to previous activities to repeat them with updated input.

![Diagram showing iteration between W, H, and D activities with feedback loops for design process and productivity, support, maintainability.]
Manage change

Each performance of each model-making activity may result in a different model.

Every one of these different models shall be subject to both change and version management.

Enterprise models shall be designed in such a way as to allow their constituent parts to be managed by an automated configuration-management system.
Hierarchy types

**Classification** (Kind-of) hierarchies shall be used within models to classify building blocks for entities to be modeled.

**Composition** (Part-of) hierarchies shall be used to link models of different scope and detailing granularity of decomposition.

Observation – not all structures are so regularly composed
Structure concept types

1. Activities correspond to elements and objects correspond to relations.
   - E.g., a value-adding process where the output objects (considered as relations) of one activity (considered as an element) are the input objects of another action (considered as an element).

2. Activities correspond to relations and objects correspond to elements.
   - E.g., the structure of a process plan where two objects (considered as elements) are linked by an activity (considered as a relation).
Compatible structures

The type of structuring shall be unambiguous to whatever facility is interpreting the models, either human or machine.

The enterprise modeler shall ensure that models obtained by the two structuring approaches are able to interoperate.

Observation - two views of the same underlying conceptual or actual model
Behavioral concepts

An enterprise is a social hybrid system, determined by properties of humans and machines.

Humans (modeled as objects or resources) in the enterprise have a different behavior (e.g. learning and problem solving) from machines (e.g. acting and reacting) and sometimes need a different kind of information.
Representing behavior

The set of models for an enterprise shall have the capability to describe behavior with respect to

- sequentiaity, events, actions, condition
- states, state changes, start states, end states,
- sequencing relationship between actions,
- description of transformation functions.
Time concepts

Time is relative to the observer.

- Static representations are devoid of time
- Dynamic representations express time sensitive properties and dependencies
- Change is immediate (short-term) or continual (long-term)

Describing behavior requires sequentiality.

- Sequential cycles are similar states being traversed at different times.
- Measuring sequence in time enables discrimination between similar cycles progressing at different rates.
Representing time

To trace individual elements, sequence properties shall be modeled to describe short-term changes.

EM shall be able to represent time duration, dynamic performance of processes, and sequential phenomena after specific units of time.

EM used to analyze enterprise performance or to simulate processes shall be able to represent effects of sequential phenomena and the time duration of each sequence step.
Modeling purpose

Models describe essential and relevant parts of an area of concern.

Models do not duplicate reality but are limited approximations of the subset of reality under consideration.

Extent of model detail is relative to its purpose.

Full model description includes purpose, assumptions, and constraints.
Observers ≈ Viewpoints

Observers perceive and analyze with attribution of meaning.

Observer filter is continually modified by experience, personality, politics, society, and situation.

Enterprise modeler is observer whose purpose is to create an enterprise model.

The modeler shall define unambiguously the purpose for the model.

- Model user is observer with task in area of concern addressed by the model.
Views

Views enable observation using assumptions and constraints. (viewpoints)

Views are used to verify completeness, consistency, and integratability of EM.

Two views are of primary importance in representing the structure and behavior of a real world system.

- The information view reveals structure.
- The function view reveals behavior
Model description

A full, integratable description of any model shall include statements and descriptions of its purpose, assumptions, and constraints. This shall be done by including a minimum set of modeler views that ensure adequate completeness and consistency, and provide the potential for integrating multiple models of the same enterprise.
Many models, one solution

There are as many ways to represent in models as there are reasons to model.

Users want to reuse models across applications and not be dependent on specific application and tool configuration.

Users want the many forms to appear as and operate as one solution for their enterprise modeling needs.

Many models must interoperate to be one solution.
Integrated - bottom up

- Diverse models using common template
- Enormous difficulties associated with standardizing large numbers of models
- Standard or reference models must be as rich as the constituent models
  - Stored in standard form filtered or translated by application, e.g., IRDS
  - Standard models agreed by constituent model owners, e.g., STEP
Way to interoperate - 2

Unified  - top down

• Template provides a common meta-level structure across constituent models to enable mapping between models using semantic equivalence.

• Template is the meta-model for all models but, therefore, is not executable.

• Normalized semantics is established by model owners. (OMG UPDM)

• Then interoperation is straightforward.
Way to interoperate - 3

Federated - chaotic reality

- No agent imposes requirements for semantic equivalence across models of the enterprise.
- Template is at the meta-level and is not executable.
- Interoperation requires dynamic accommodation rather than predetermined meta-model.
- A shared ontology is very helpful.
Communication essential

Integrated depends upon standard protocols.

Unified depends upon mediation.

Federated success is highly dependent upon skills and capabilities of humans and machines.

Observation - Interoperation is the exchange of information, material, and energy between entities.
ISO 15704:2000

Adopted in 2000 after 15 year effort to consolidate existing knowledge and practice in the area of industrial automation

Lightweight standard, only 8 pages, and middleweight annex A of 31 pages with heavyweight content

References normative content from ISO 14258
ISO 15704:2000 (cont.)

Primary input is annex A produced by an IFIP-IFAC task force

Annex A, known as GERAM, articulates a compliant approach and expands context

Amended in 2005 with 2 new user views

- Economic view that introduces a new dimension of detail
- Decision view that introduces a way to articulate operational decisions in time
Informative introduction

Two primary concerns for enterprise integration base of reference
- Model the whole life history of an enterprise-integration project
- Encompass the people, processes and equipment involved in achieving the enterprise mission

EA is about enterprise project structure.
SA is about system that is part of EA.
Eight key principles

1. Issues of enterprise integration are ubiquitous and apply to any enterprise, regardless of its size and mission or other attributes.
   - Integration goes beyond information and control systems to encompass culture and mission.
   - Manufacturing is a customer service.

2. Enterprise identification and mission definition are essential.
Key principle 3

3. Separate mission-fulfillment functions from mission-control functions

- Fulfillment includes process operation to produce the product or service
- Control includes the use of information to manage processes and maintain operations
- Interconnection between fulfillment and control is operational data to control and operational commands to fulfillment.
Key principles 4 & 5

4. Operations consists of the many transformations of material, energy, and information performed by processes.
   - Processes executed concurrently or sequentially
   - Combination of processes define functionality of the enterprise

5. Evolutionary integration of modules is essential.
Key principle 6

6. Three kinds of processes

- Information and control activities that can be automated by control devices
- Mission activities that can be automated by mission-fulfillment equipment
- Activities carried out by humans, whether for information and control or mission-fulfillment

• Desire a simple way of showing where and how humans fit in the enterprise and how the distribution of functions between humans and machines is accomplished
Function distribution

Adapted from: T. J. Williams, Institute for Interdisciplinary Engineering Studies, Purdue University

© Copyright 2008 by R Martin, J Bendz, S Arnold, A Meilich

INCOSE IS08 Tutorial June 2008

see Annex A, Figure A.11
Key principles 7 & 8

7. Every enterprise and product has a life-cycle.
   - A life-cycle can be partitioned
     • Partitioning is specific to purpose
     • Partitioning does not imply strict sequence
   - One enterprise can be embedded within the life-cycle of another’s operational phase.

8. Modularity should be enforced whenever possible.
Embedded life-cycles

Entity Type 1
Strategic Management Entity
- Initiates, defines, supports

Entity Type 2
Engineering Implementation Entity
- Identification
- Concept
- Requirements
- Preliminary design
- Detailed design
- Implementation
- Operation
- Decommissioning

Entity Type 3
Enterprise Entity
- Identification
- Concept
- Requirements
- Preliminary design
- Detailed design
- Implementation
- Operation
- Decommissioning

Entity Type 4
Product Entity
- Identification
- Concept
- Requirements
- Preliminary design
- Detailed design
- Implementation
- Operation
- Decommissioning

Entity Type 5
Methodology Entity
- Identification
- Concept
- Requirements
- Preliminary design
- Detailed design
- Implementation
- Operation
- Decommissioning

Supports
Task 1
Task 2
Task 3
Task 4
see Annex A
Figure A.5
Stakeholder concerns

The standard enables an enterprise-integration-planning team to:

- Describe the tasks required
- Define the necessary quantity and quality of information
- Specify relationships among humans, processes, and equipment in the integration considered
- Address management needs
More concerns

- Address relevant economic, cultural, and technological factors
- Detail the extent of computer-support required
- Support process-oriented modeling that can model the whole life history of an enterprise
- Checking for completeness with respect to ERAM current and future purpose
ISO 15704 Scope

Enterprise-reference architectures and methodologies (ERAM) covers those constituents deemed necessary to carry out all projects required by the enterprise throughout the whole life of the enterprise, including:

- enterprise creation
- major enterprise restructuring efforts
- incremental changes affecting only parts of the enterprise-life cycle
Requirements of 15704

Enterprise-reference architectures and methodologies (ERAM) shall be capable of assisting and structuring the description, development, operation, and organization of any conceivable enterprise entity, system, organization, product, process, and their supporting technology.

Areas covered by ERAM shall be clearly identified.
General usage

The methodology associated with a reference architecture shall provide the necessary guidelines and management techniques for the initiation and pursuit of a project or program of development and operation of an enterprise or entity.

Such a methodology may or may not be model-based, i.e., the enterprise engineering process may or may not result in a specific enterprise model.
Many potential ways

ERAM need not be based on any one single methodology and its accompanying architecture or framework. Many different methodologies and/or frameworks may be used for it.

The primary consideration **shall** be applicability and capability in relation to these requirements.
Design and operation

ERAM shall identify

- concepts and components
- activities to manage, conceive/define, describe, design, implement, maintain, and decommission any enterprise entity
- activities to use the results of enterprise engineering in the operation itself
  • Such use may include model-based decision support and model-driven operation monitoring and control.
Conceptually broad

Throughout the life-cycle of the enterprise, the ERAM shall address the
- role of humans
- description of processes (function and behavior)
- representation of supporting technologies
Human oriented

ERAM shall exhibit the capability to represent human aspects, such as:
- Organizational and operational roles
- Capabilities, skills, know-how, competencies
- Responsibilities, authorization
- Relations to the organization.
Process oriented

ERAM shall exhibit the capability to represent the enterprise operation. Such representations shall cover both the functionality and behavior of the operation.

The representations shall recognize the life cycle and life-history concepts of enterprise-entity types and shall support process-oriented operations.
Technology oriented

ERAM shall exhibit the capability to represent all technologies employed in the enterprise operation.

Such representation shall provide for the use of integration-technology infrastructures to support

- enterprise engineering and operation of business processes, models of enterprise resources, facility layout models, information-system models, communication-system models and logistics models.
Mission oriented

ERAM **shall** exhibit the capability to represent any process and its constituent activities involved in

- performing the established mission of the enterprise
- accomplishing management and control of the established mission of the enterprise according to the criteria established by enterprise management
Framework for modeling

ERAM that are model-based shall exhibit the capability to model entities within the conceptual space defined by three dimensions.

Model View

Model life-cycle phase

Model Genericity
Genericity

ERAM that are model-based shall provide the capability for representing

- Generic-enterprise elements that are reusable modeling language constructs
- Partial-enterprise models that are prototype models of industry segment or industrial activity
- Particular enterprise models that are models of a particular enterprise domain
Life-cycle & life-history

ERAM shall identify and represent the life-cycle phases pertinent during the life of any enterprise entity.

(Dependency links phases)

ERAM shall be capable of representing the life history of any enterprise entity, i.e., the representation in time of activities carried out on any enterprise entity (traceability)

(Chronology links history)
ISO 19439 Model phase

The purposive ordinant dimension ordered by coordinates corresponding to the life-cycle phases

- Domain identification
- Concept definition
- Requirements definition
- Design specification
- Implementation description
- domain Operation
- Decommission definition

Identify
Elaborate
Use
Dispose

Emphasize model development process for process oriented modeling.
Early phases

Domain identification
- Business objectives, functions, capabilities

Concept definition
- Enablers of objectives & operations
- Means for achievement of functions & capabilities

Requirements definition
- Functional, behavioral, informational, capability for service, manufacturing, management and control

Design specification
- Processes with all components necessary to satisfy requirements
Post-design phases

Implementation description
- All information needed for all tasks of operational system

Domain operation
- Operational usage of model released from implementation

Decommission definition
- Tasks and resources for retraining, redesign, recycling, preservation, transfer, disbanding, disassembly, disposal
Many possible coordinates

15288 Stage 19439 Phase DoDAAF Guidance

Concept Development Production Utilization Support Retirement

Domain Concept Requirement Design Implementation Operation Decommission

Focus Scope Characterize Determine Build Use
Life history

Enterprise Engineering Projects

Redesign/continuous improvement project

Enterprise Operation

Decommissioning Project

see Annex A, Figure A.3
Life history example

Model “view”

ERAM that are model-based shall provide concepts for representing views of a model to allow it to be described as an integrated model but presented to users in different subsets.

- Views contain subsets of facts present in the integrated model
- Concentrate on relevant questions respective stakeholders may wish to consider.
- The concept of view is applicable to models of all entity types across their life cycle.
Four views required

Different views may be made available highlighting certain aspects of the model and hiding others.

ERAM that are model based shall include four model-content views:

- Function
- Information
- Resource
- Organization
Function view of 19439

Function

- Enables representation and modification of the processes of the enterprise, their functionalities, behaviors, inputs and outputs
- Emphasis on system behavior, mutual dependencies, and influence of elements during function execution
- Includes decisional, transformational and support activities
- Identifies all entities (material, information, resources and control) required for function execution
Other views of 19439

Information

• The material and information related objects used and produced in the course of operations

Resource

• Capabilities of people and technological component assets

Organization

• Authority and responsibility during operations
• Expresses decision support structure
Many views possible

See Annex A, Figure A.9

© Copyright 2008 by R Martin, J Bendz, S Arnold, A Meilich

INCOSE IS08 Tutorial June 2008
Graphic 19439 dimensions

CIM Systems Integration: Framework for Enterprise Modelling

domain identification
concept definition
requirements definition
design specification
implementation description
domain operation
decommission definition

enterprise model phase

enterprise modelling view

genericity

context
reference
Harmony

Particular level
Reference Catalog

not defined at domain operation phase

function view
information view
resource view
organization view

© Copyright 2008 by R Martin, J Bendz, S Arnold, A Meilich
INCOSE IS08 Tutorial June 2008
Architecture components

Enterprise-engineering methodologies (EEM) for every type of life-cycle activity for any enterprise-entity type.

Enterprise modeling languages (EML) or modeling constructs that allow the enterprise operation to be described.

- Constructs shall allow users to represent the different elements and thereby improve both efficiency and understanding.
GERA entity types

Annex A Figure A.6
Languages for people

The form of modeling constructs shall be adapted to the needs of people creating and using enterprise models.

EML shall be expressive enough to model human roles, operational processes and their functional contents and support EML semantics can be described in terms of ontological theories but the definition of the formal semantics shall be supported by natural language explanations of the concepts.
## Constructs of 19440

<table>
<thead>
<tr>
<th>Domain</th>
<th>Enterprise Object</th>
</tr>
</thead>
<tbody>
<tr>
<td>Business Process</td>
<td>Object View</td>
</tr>
<tr>
<td>Enterprise Activity</td>
<td>Product</td>
</tr>
<tr>
<td>Event</td>
<td>Order</td>
</tr>
<tr>
<td>Resource</td>
<td>Operational Role</td>
</tr>
<tr>
<td>Functional Entity</td>
<td>Organizational Unit</td>
</tr>
<tr>
<td>Capability</td>
<td>Organizational Role</td>
</tr>
<tr>
<td>Decision Centre</td>
<td>Person Profile</td>
</tr>
</tbody>
</table>

© Copyright 2008 by R Martin, J Bendz, S Arnold, A Meilich

INCOSE IS08 Tutorial June 2008
Construct template

Common format

- Header
  - Type label
  - Identifier unique to model
  - Name
  - Authority for design of construct

- Body
  - Descriptives in textual form
  - Relationships specified by reference
Construct descriptives

Predefined for each construct
User-defined by extension
May be qualified (e.g. mandatory or optional)
Possibly XML schemas or EXPRESS notation

Attributes
  - Name (meaningful in domain)
  - Data type (simple or complex)

Complementary Concepts
Complementary concepts

Not fully developed as constructs but have particular significance and semantics for the purpose of enterprise modeling

• Behavior rule
• Constraint
• Declarative rule
• Functional operation
• Integrity rule
• Objective
• Performance indicator
Construct relationships

Model the dynamics between run-time instances

Types of relationship
- Operational authority and responsibility
- Membership in specialization
- Part of an aggregation
- Consist of an aggregation
- Other associations

Possibly reflexive
Constructs and roles

Human organizational role captures assigned responsibilities and required capabilities (skills)

Human operational role captures the operational capabilities of person assigned to a task

Machine operational role captures the operating capabilities of machine assigned to a task

Machine product role captured by attributes that describe input and output of activities to change state
**Construct across phases**

Construct label: **EO** (Enterprise Object)

Identifier: `<model-unique string>`

Name: name of the Enterprise Object instance

Design Authority: `[[<identifier> "/" <name>] [NIL | :" <identifier> "/" <name>]]` of Organizational Role and Organizational Unit respectively, having authority to design or maintain this particular instance

**Body**

### A1 Descriptives relevant for all enterprise model phases

- **Description**: short textual description
- **Nature of Object**: PHYSICAL | INFORMATION
- **Properties**: `[^<property_name> = <property_value>]+` - elements representing properties and their values for the entity represented by the Enterprise Object instance
- **Constraints**: `[^<constraint>]`* imposed on selected named attributes of the Enterprise Object instance

### A2 Descriptives relevant for different enterprise model phases

**A2.1 applicable at concept definition and later phases**

Not applicable

**A2.2 applicable at requirements definition and later phases**

- **Integrity Rules**: `[^<integrity rule>]`* applicable to attributes of the Enterprise Object instance in the requirements definition phase
Life-cycle expression

Expressive effort

Domain function

Enterprise activity

Business process

Functional Entity

Do  C  R  D  I  O  Dc

Context Reference
42010
15704
Harmony

© Copyright 2008 by R Martin, J Bendz, S Arnold, A Meilich

INCOSE IS08 Tutorial June 2008
Process behavior

Shall be described in its behavioural rule set attribute by a set of *behavioural rules*, which control the sequence of constituent Business Processes and Enterprise Activities

- well-structured is completely defined
- semi-structured is known at run-time
- ill-structured is non-deterministic
Behavior rules

Shall enable

- the capturing of all the conditions that control the sequencing and the dynamic behaviour of Business Processes
- their presentation in both human and machine understandable form

Apply only to Business Processes

- Internal behaviour of an Enterprise Activity, the sequencing of its set of functional operations, is considered to be an implementation issue.
Formal syntax in BNF

behavioural rule set = behavioural rule {behavioural rule} ;

behavioural rule = WHEN condition DO action “;”

condition = condition term
  | condition term {AND condition term}
  | condition term {OR condition term} ;

condition term = single condition | ( condition ) ;

single condition = event occurrence | action completion ;

event occurrence = event | start | exception ;

event = ev ; (*the named Event needs to be included in the containing Business Process's list of Event Inputs *)
Object view as pattern

D. N. Shorter, ICEIMT’04

Note the update semantics
Module representation

ERAM shall be able to represent the concept of enterprise modules, building blocks or systems (products, or families of products) for use as common resources in enterprise engineering and enterprise integration.

- One important set of enterprise modules is the integrating infrastructure or the set of integration-technology services required for enterprise engineering and operation in heterogeneous environments.
Operational system

One result of the enterprise-engineering process shall be a design or model for the enterprise-operational system (EOS).

The enterprise operational system shall consist of the hardware and software needed to fulfill the enterprise objectives and goals.

The content of the operating system is derived from enterprise requirements.
Glossary a must

To promote understanding about projects and other co-operative efforts, ERAM shall provide

- consistent glossary and a semantics and syntax for use in enterprise-engineering and integration efforts, or

- reference to other suitable glossaries.
Scope of GERAM

**Generalized Enterprise Reference Architecture**
identifies concepts of enterprise integration

**Generic Enterprise Modeling Concepts**
(Theories and Definitions)
define the meaning of enterprise modeling constructs

**Partial Enterprise Models**
provide reusable reference models and designs of human roles, processes and technologies

**Enterprise models**
provide implementable **MOdules** of human professions, operational processes, technologies

**Enterprise Engineering Methodology**
describe process of enterprise engineering

**Enterprise Engineering Tools**
support enterprise engineering

**Enterprise Modeling Languages**
provide modeling constructs for modeling of human role, processes and technologies

**Enterprise Models**
designs, and models to support analysis and operation

**Enterprise Operational Systems**
support the operation of the particular enterprise

(Adapted from ISO 15704)
GERA modeling framework

O. Noran, Griffith University version of Annex A Fig. 10

Context Reference
42010
15704
Harmony

Subdivide according to:
- genericity
- purpose of activity
- physical manifestation
- model content
- means of implementation

Life Cycle Phases:
- Identification
- Concept
- Requirements
- Prelim. design
- Design
- Detailed design
- Implementation
- Operation
- Decommission

Views:
- Generic
- Partial
- Particular

Phases:
- Design
- Prelim. design
- Detailed design
- Implementation
- Operation
- Decommission

Management and Control
Product or Service
Software
Hardware
Resource
Organisation
Information
Function
Machine
Human

Context Reference
42010
15704
Harmony
Many GERA & standards

15288 technical processes

Reference models
- 15288
- PMBOK
- STEP
- 9003:2000
- IEEE 1058
- EFQM

Infrastructure providers / supporting systems (e.g. logistic support)

Product ("Target System")

Define product

Issue tender

Contribute to project

Create project enterprise

Engineering contractor

15288 Enterprise processes

Engineering subcontractors

Build product

Provide logistic/infrastructure services

Create project enterprise

Context Reference
42010
15704
Harmony

Bernus & Noran, 2004

© Copyright 2008 by R Martin, J Bendz, S Arnold, A Meilich

INCOSE IS08 Tutorial June 2008

IE³ 1058 PM Planning

15288 project (mgmt) processes

Agreement processes

Many GERA & standards
ISO 15704 Compliance

Any assessment of the degree of compliance of a candidate architecture and methodology shall be qualified by the following:

- a preliminary statement as to whether or not they are model based;
- a statement of the degree to which they then conform partially or totally to the appropriate requirements

In the event of partial compliance, areas of non-conformance shall be explicitly identified.
Conformance to 19439

Shall have function and information view
Shall be able to derive resource and organization view
Shall provide distinct model phases
Shall provide for derivation of partial and particular model from generic constructs
Shall propagate model changes to all views
Conformance to 19440

Shall either use the constructs as defined or be able to map to the constructs.

Can claim qualified compliance by using a subset of constructs or mapping to a subset of constructs.

Can be a valid construction of a compliant modeling language.

Shall identify construction and model execution testing levels.
Do 15704 and 42010 fit?

• Check for conformance
  - Are stakeholders and concerns explicit?
  - Is there a viewpoint for concerns?
  - Are there corresponding views?
  - Is there rationale provided?

• What about the constraint that a viewpoint has only one view?
  - Each life-cycle phase has a different AD
  - AD’s can be generic, partial or particular
Another fit for 15704

• Now consider 15704 as an architectural viewpoint for the enterprise system
  - The viewpoint specifies many (sub)views of three kinds: phase, view, and genericity
  - The architectural view, expressed as many (sub)view enterprise models, spans the whole enterprise from concept through demise

• 15704 is a reference enterprise architecture and the standard is its AD
Using 15704 to build AD's

Architecture description (architecture)

Life-cycle phase (system)
Domain Identification

Concept Definition
Requirements Definition
Design Specification
Implementation Desc.
Domain Operation
Must be a meta-model

- Current effort to articulate an information meta-model for a general enterprise reference architecture suitable for ISO 15704
- Believe a subset of that meta-model, when specialized, is the core model for architecture description identified in ISO 42010
- Harmony is established through a meta-model and demonstrated interoperation
A unifying meta-model?

15704 Metamodel V1.3
20 May 2008
Bernus/Noran
WG1 standards status

ISO 14258:1998 - systematic review completed in 2005 with retention until essential content can be moved to revised ISO 15704

ISO 15704:2000 - systematic review completed in 2007 and revision project underway


EN/ISO 19440 - published December, 2007
WG42 standard status

- ISO/IEC 42010:2007 - revision project begun with publication in accordance with ISO/IEEE fast-track agreement
- Working draft 2 now in circulation
Overall Assessment 1

Many semi-independent international architecture standards & activities

Too many conflicting terms, many perspectives, varied applications

US/DoD architecture perspectives (DODAF, CADM, FEAF) - DODAF 2.0 coming in ’08

INCOSE Architecture related WGs biased towards DoD/US Gov’t perspectives/needs

ISO TC184/SC5 EA Standards biased to Int’l & Manufacturing Industry Architecture Perspectives
Overall Assessment 2

Harmonization is critical, but harmonizing these diverse international enterprise architecture points of view together is a major challenge.

Being involved provides an opportunity to contribute toward and learn about the future enterprise architecture for more intelligent enterprise realizations.
Contacts

Richard Martin - SC5/WG1 Convener
  - richardm@tinwisle.com

Johan Bendz - SC7/WG42 Convener
  - johan.bendz@fmv.se ; johan@bendz.se

Abe Meilich - INCOSE liaison to ISO TC184/SC5/WG1
  - abraham.w.meilich@lmco.com

Stuart Arnold - INCOSE liaison to ISO/IEC JTC1/SC7/WG42
  - sarnold@qinetiq.com

We acknowledge the contributions of David Emery, James Martin, and Mark Walker

Graphics assistance from K. P. Martin, Indiana University

© Copyright 2008 by R Martin, J Bendz, S Arnold, A Meilich
INCOSE IS08 Tutorial June 2008