A Comparison of Frameworks for Enterprise Architecture Modeling

Richard Martin
Tinwisle Corporation
Bloomington, Indiana

Edward Robertson
Computer Science Department
Indiana University

© Copyright 2003 by R. Martin and E. Robertson

A Comparison of Frameworks for Enterprise Architecture Modeling

- Framework Principles
  Structure, Connections, Views, Constraints
- Usage Observations
  Prototypes, Time, Purpose
- Archetypes
  Zachman, ISO 15704, ISO/CEN 19439, ISO/IEC 15288
- Complements
  Prototypes, Purpose, Artifacts, Change

© Copyright 2003 by R. Martin and E. Robertson

What is a Framework?

A containment structure

- context for model artifacts
- interconnections between models
- access to model components
- model fidelity and consistency

NOT a programming framework.

© Copyright 2003 by R. Martin and E. Robertson

Structure

A space of one or more dimensions

meta-model:

Arrangement

- Ordinant (label) - Ordered, Unordered
- Decomposing (path)

Scale

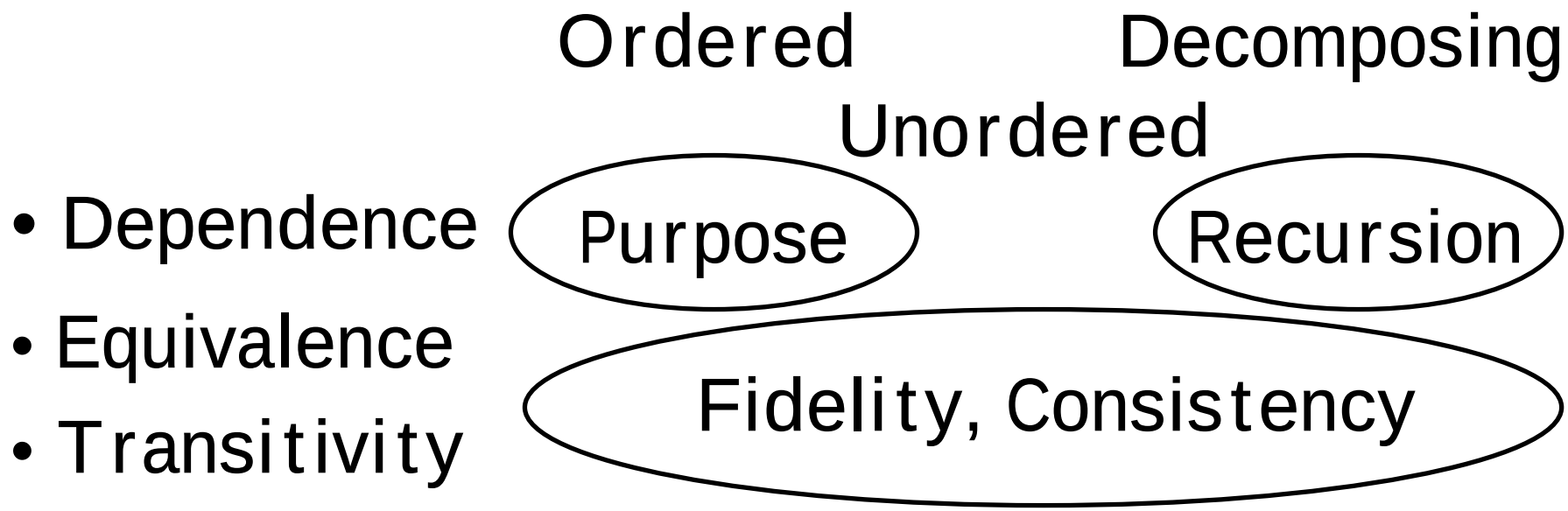
- Scope (general to specific)
- Abstract (abstract to concrete)
- Detail (coarse to fine)

© Copyright 2003 by R. Martin and E. Robertson

Connections

Structural linkage along and among dimensions

meta-model:



© Copyright 2003 by R. Martin and E. Robertson

Views

Different ways of looking at artifacts

meta-model:

- Filter along a dimension
- From one dimension to another
- Rearrange a framework – derive a view
- Use selection and projection

Formal meta-model harder than mechanism

© Copyright 2003 by R. Martin and E. Robertson

Constraints

Evaluate conformance to a standard

meta-model:

- Structure – a place for everything of interest
- Connection – within and between dimensions, typically binary
- View – something must be placed to be seen, often used to define constraints
- Distinguish model from instance constraints
- Formal mechanisms within one model

© Copyright 2003 by R. Martin and E. Robertson

Artifact Prototypes

- Frameworks are conceived with prototype artifacts in mind
- Framework artifacts are models we build both formally and informally
- Frameworks partition artifacts along conceptual categories (dimensions) with coordinates and paths
- Prototypes range over all enterprise aspects – automated, mechanical, human
- Framework expression is the realized model instances derived from prototype artifacts

© Copyright 2003 by R. Martin and E. Robertson

Entities in Time

The characterization of a framework with respect to time informs us about the nature of change in the framework's context.

- **Continuant** - identity continues to be recognizable over some extended interval of time
- **Occurrent** – identity is not stable during any interval of time.

(see SOWA)

© Copyright 2003 by R. Martin and E. Robertson

Continuants / Occurrents

- **Continuants** are wholly present (i.e., all their parts are present) at any time they are present.
- **Occurrents** just extend in time by accumulating different temporal parts, so that, at any time they are present,they are only partially present.
- **Continuants** are entities that are in time. Lacking temporal parts all their parts flow with them.
- **Occurrents** are entities that happen in time. Their temporal parts are fixed in time.

(see Masolo, Borgo, Guarino, et. al.)

© Copyright 2003 by R. Martin and E. Robertson

Enterprise Description

- Enterprise as product is continuant
- Enterprise as process is occurrent

- Purpose emerges from an ordered dependency
- Dependency is not necessarily chronology

- Purpose can be found in both continuant and occurrent enterprise descriptions
- Frameworks address continuant and occurrent purposes in enterprise description – but a single framework cannot do both!

© Copyright 2003 by R. Martin and E. Robertson

Zachman Framework for Enterprise Architecture

Principles
Observations
Archetypes
Complements

ENTERPRISE ARCHITECTURE - A FRAMEWORK ™

	DATA *What*	FUNCTION *How*	NETWORK *Where*	PEOPLE *Who*	TIME *When*	MOTIVATION *Why*	
SCOPE (CONTEXTUAL) *Planner*	List of Things Important to the Business ENTITY = Class of Business Thing	List of Processes the Business Performs Function = Class of Business Process	List of Locations in which the Business Operates Node = Major Business Location	List of Organizations Important to the Business People = Major Organizations	List of Events Significant to the Business Time = Major Business Event	List of Business Goals/Strat Ends/Means=Major Bus. Goal/ Critical Success Factor	SCOPE (CONTEXTUAL) *Planner*
ENTERPRISE MODEL (CONCEPTUAL) *Owner*	e.g. Semantic Model Ent = Business Entity Reln = Business Relationship	e.g. Business Process Model Proc. = Business Process I/O = Business Resources	e.g. Logistics Network Node = Business Location Link = Business Linkage	e.g. Work Flow Model People = Organization Unit Work = Work Product	e.g. Master Schedule Time = Business Event Cycle = Business Cycle	e.g. Business Plan End = Business Objective Means = Business Strategy	ENTERPRISE MODEL (CONCEPTUAL) *Owner*
SYSTEM MODEL (LOGICAL) *Designer*	e.g. Logical Data Model Ent = Data Entity Reln = Data Relationship	e.g. "Application Architecture" Proc .= Application Function I/O = User Views	e.g. "Distributed System Architecture" Node = I/S Function (Processor, Storage, etc) Link = Line Characteristics	e.g. Human Interface Architecture People = Role Work = Deliverable	e.g. Processing Structure Time = System Event Cycle = Processing Cycle	e.g., Business Rule Model End = Structural Assertion Means =Action Assertion	SYSTEM MODEL (LOGICAL) *Designer*
TECHNOLOGY MODEL (PHYSICAL) *Builder*	e.g. Physical Data Model Ent = Segment/Table/etc. Reln = Pointer/Key/etc.	e.g. "System Design" Proc.= Computer Function I/O = Screen/Device Formats	e.g. "System Architecture" Node = Hardware/System Software Link = Line Specifications	e.g. Presentation Architecture People = User Work = Screen Format	e.g. Control Structure Time = Execute Cycle = Component Cycle	e.g. Rule Design End = Condition Means = Action	TECHNOLOGY CONSTRAINED MODEL (PHYSICAL) *Builder*
DETAILED REPRESEN-TATIONS (OUT-OF-CONTEXT) *Sub-Contractor*	e.g. Data Definition Ent = Field Reln = Address	e.g. "Program" Proc.= Language Stmt I/O = Control Block	e.g. "Network Architecture" Node = Addresses Link = Protocols	e.g. Security Architecture People = Identity Work = Job	e.g. Timing Definition Time = Interrupt Cycle = Machine Cycle	e.g. Rule Specification End = Sub-condition Means = Step	DETAILED REPRESEN-TATIONS (OUT-OF CONTEXT) *Sub-Contractor*
FUNCTIONING ENTERPRISE	e.g. DATA	e.g. FUNCTION	e.g. NETWORK	e.g. ORGANIZATION	e.g. SCHEDULE	e.g. STRATEGY	FUNCTIONING ENTERPRISE

Zachman Institute for Framework Advancement - (810) 231-0531 Copyright - John A. Zachman, Zachman International

(used with permission)

© Copyright 2003 by R. Martin and E. Robertson

Zachman Framework for Enterprise Architecture

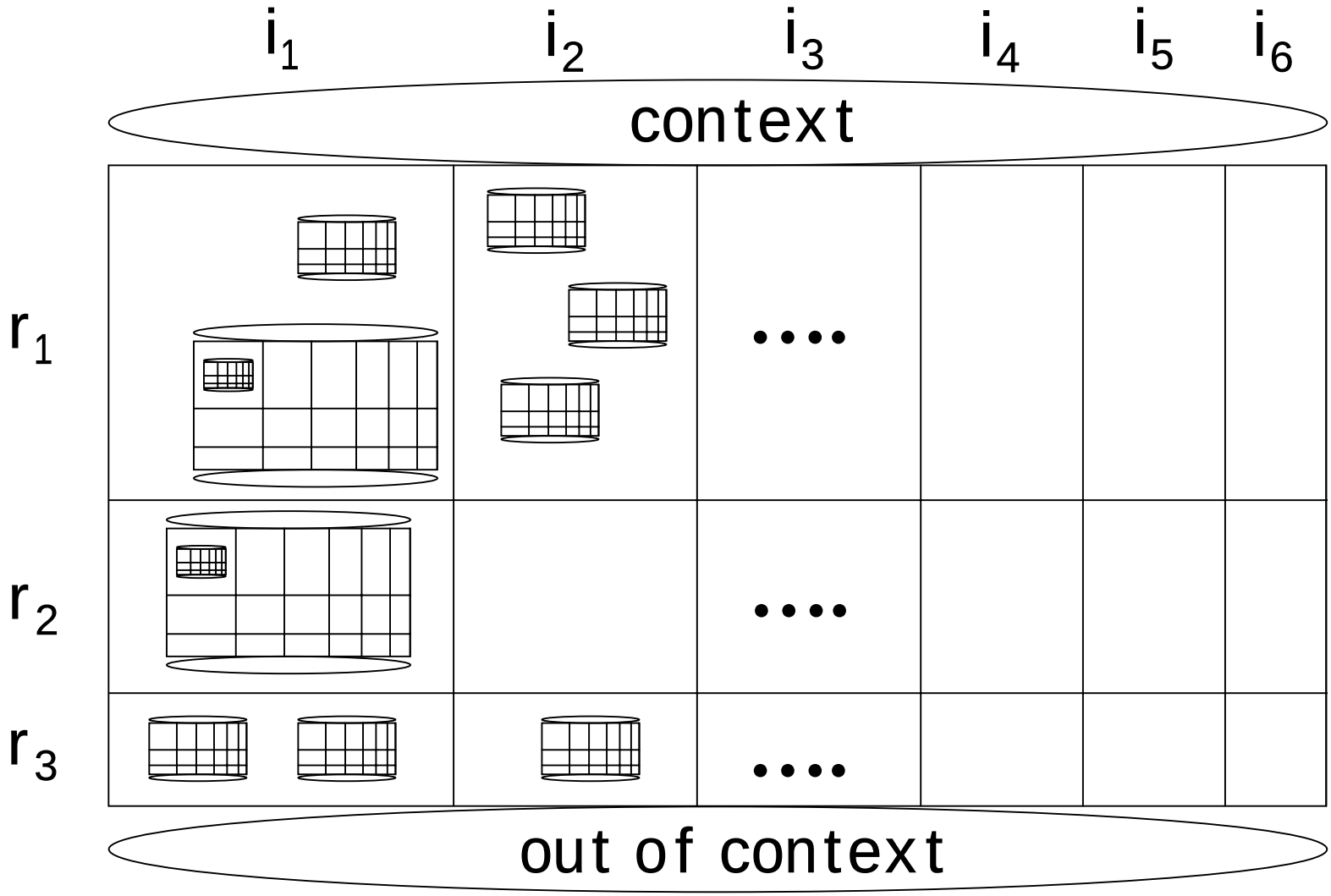
(Information System version)

Principles
Observations
Archetypes
Complements

R \ I	What	How	Where	Who	When	Why
	Entity - Relation	I/O - Process	Node - Link	People - Work	Time-Cycle	Ends - Means
Context	Important things	Processes performed	Operating locations	People and groups	Events and cycles	Goals and strategies
Owner	Semantic model	B-process model	Logistics network	Work flow model	Master schedule	Business plan
Designer	Logical data model	Application model	Distributed system	Human interface	Processing structure	Business rule model
Builder	Physical data model	System design	System arch.	Presenta-tion arch.	Control structure	Rule design
Out of context	Data definition	Program code	Network arch.	Security arch	Timing definition	Rule speci-fication

© Copyright 2003 by R. Martin and E. Robertson

Zachman Recursion



© Copyright 2003 by R. Martin and E. Robertson

Zachman Properties

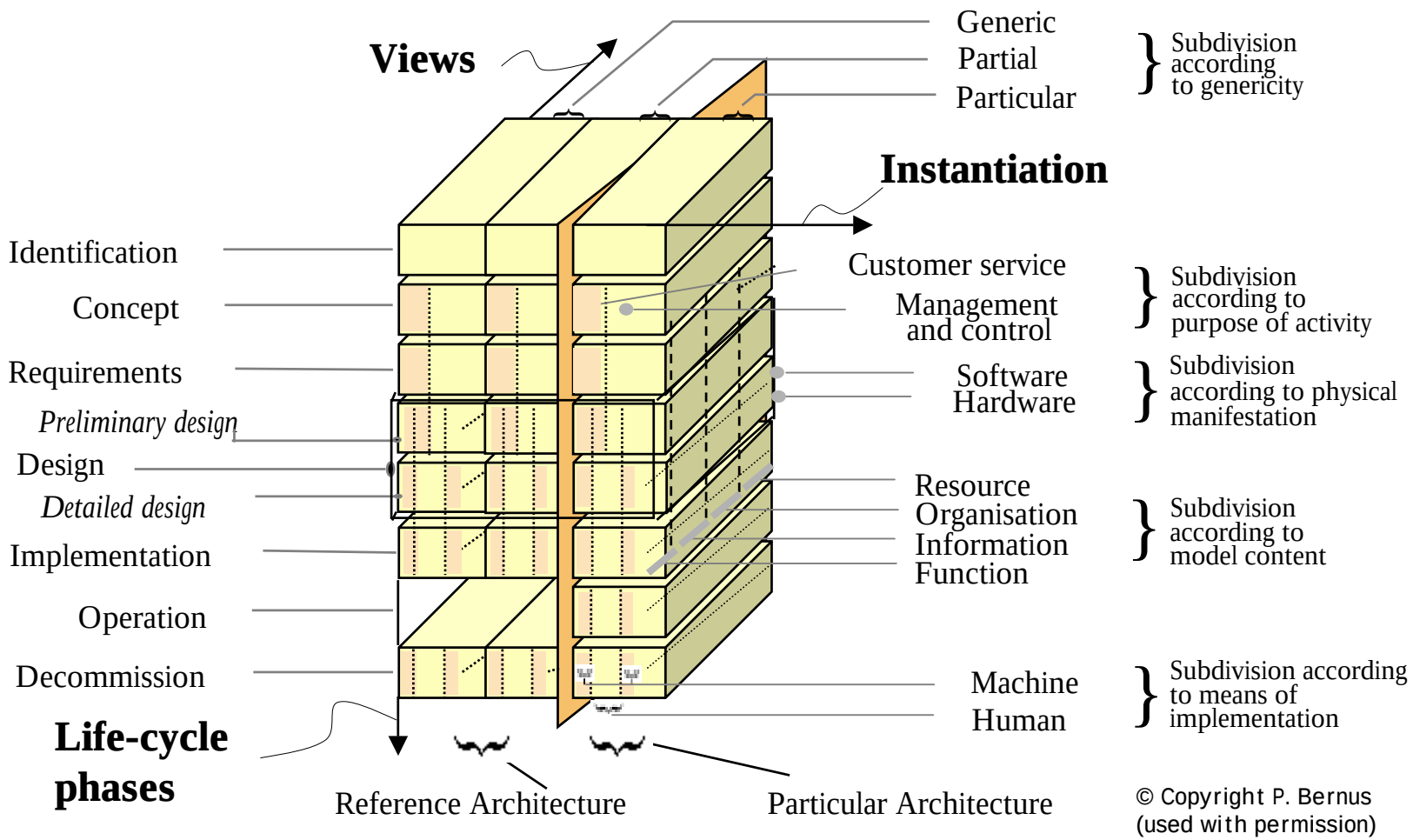
- **Role** dimension is ordinant, ordered, and purposive
- **Purposive** dimension is timeless
- **Interrogative** dimension is ordinant and unordered
- **Primitive model** contents facilitate complex model composition
- **Recursive** decomposition (frameworks nested in frameworks)

© Copyright 2003 by R. Martin and E. Robertson

ISO 15704: Annex A - GERAM

Principles
Observations
Archetypes
Complements

Generalised **E**nterprise **R**eference
Architecture and **M**ethodology

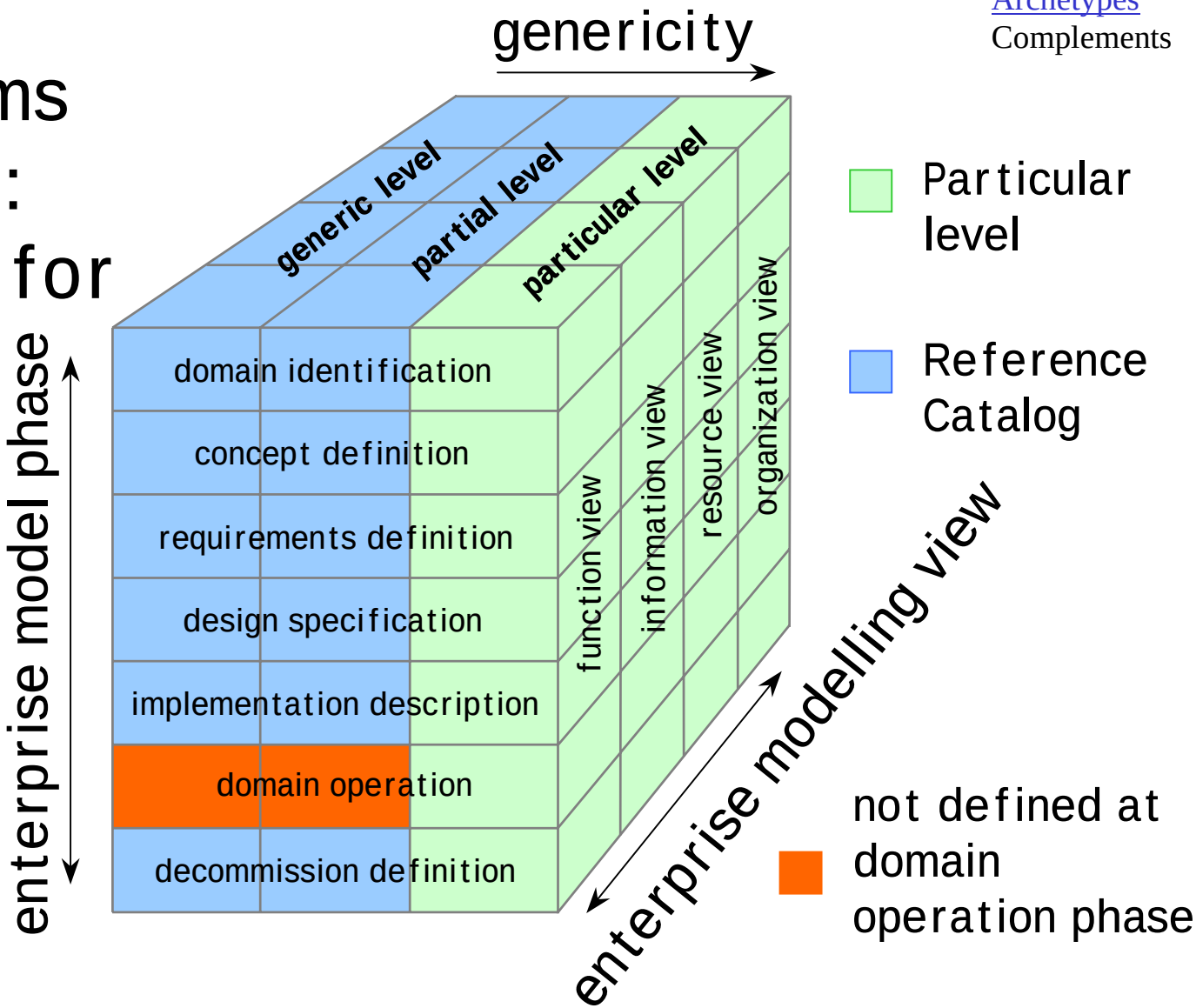


© Copyright P. Bernus
(used with permission)

© Copyright 2003 by R. Martin and E. Robertson

ISO/CEN FDIS 19439

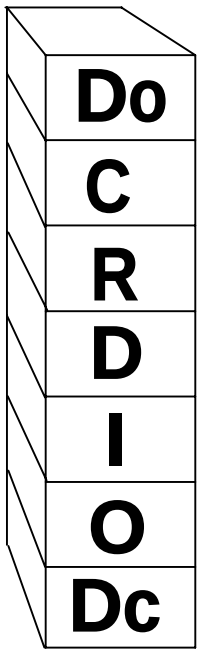
CIM Systems Integration: Framework for Enterprise Modelling



© Copyright 2003 by R. Martin and E. Robertson

19439 – Model Dimension

Model – the purposive ordinant dimension ordered by coordinates corresponding to the phases of the enterprise model life-cycle.



Enterprise **model phase**:

- **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.

© Copyright 2003 by R. Martin and E. Robertson

19439 - View Dimension

View – an unordered ordinant dimension with pre-defined or user selected coordinates that partition facts in the integrated model relevant to particular interests and context.

Enterprise **modelling view**:

Function - the system behavior, mutual dependencies, and influence of elements during function execution

Information - the material and information used and produced in the course of operations

Resource - capabilities of people and technological components

Organization - authority and decision-making responsibility during operations

© Copyright 2003 by R. Martin and E. Robertson

19439 – Genericity Dimension

Genericity – an ordered ordinant dimension that reflects 19439 as a "standard" framework.

Enterprise **genericity level**:

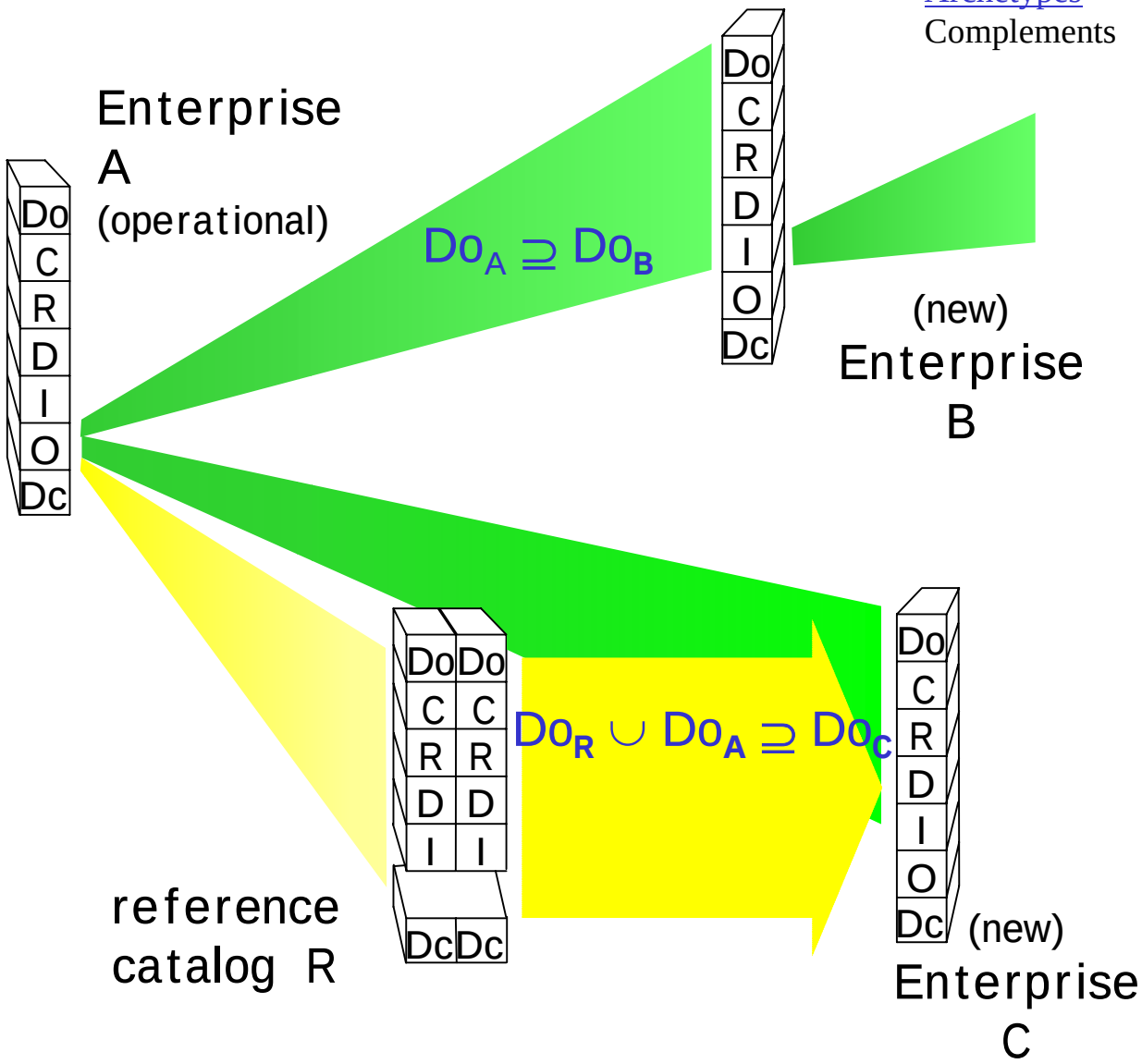
- **Generic** - reusable modeling language constructs
- **Partial** - prototype models of industry segment or industrial activity
- **Particular** - models of a particular enterprise domain

Reference catalog

© Copyright 2003 by R. Martin and E. Robertson

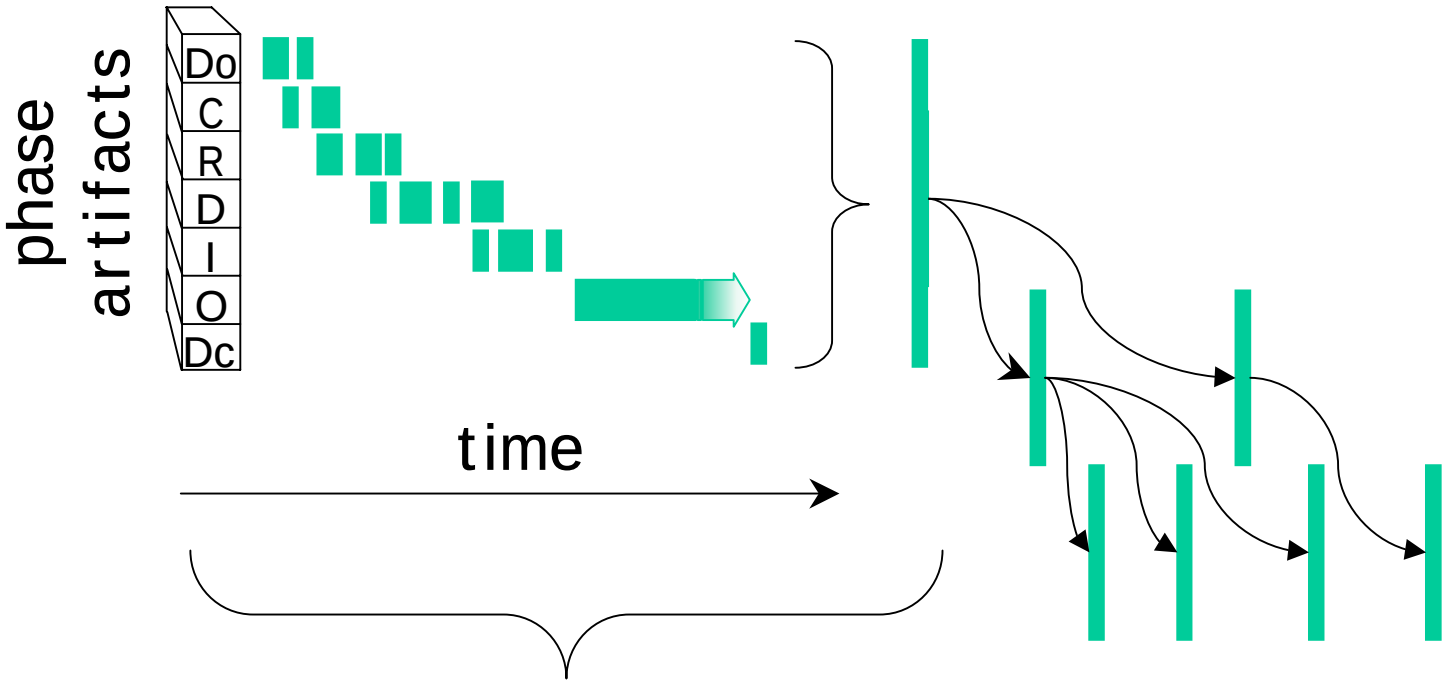
19439 - Recursion

Enterprise operations can model new enterprises either from its own particular models or using reference constructs and partial models.



© Copyright 2003 by R. Martin and E. Robertson

19439 – Life History



a complete life-cycle

a life history pictogram of related life-cycles

(point-in-time solution set)

Adapted from P. Bernus, Griffith University, Australia

© Copyright 2003 by R. Martin and E. Robertson

ISO/IEC 15288 Systems engineering – System life cycle processes

- Common process framework covering life cycle of man-made systems…spans conception of ideas through to retirement
- For acquiring and supplying systems
- Assess and improve life cycle processes
- Comprehensive set from which an organization can construct system life cycle models
- Can be applied at any level of system structure and throughout life cycle

© Copyright 2003 by R. Martin and E. Robertson

15288 - Structure

A degenerative case where framework structure is trivial but has many constraints that govern instances, e.g.,

Modularity – maximal cohesiveness of the functions of a process and minimal coupling among processes.

Ownership – a process is associated with a responsibility.

Properties – the purposes, outcomes and activities for a process

© Copyright 2003 by R. Martin and E. Robertson

15288 – Dimensions

Process Group – a hierarchic arrangement where enterprise processes manage project processes composed of technology processes all mediated by agreement processes

Life cycle – minimal normative requirement

"A life cycle model that is comprised of stages shall be established…The purpose and outcomes shall be defined for each stage of the life cycle."

© Copyright 2003 by R. Martin and E. Robertson

15288 - Process Groups

- Agreement – define activities that establish agreement between internal/external entities
- Enterprise – manage capability to acquire and supply through project initiation, support and control
- Project – establish and evoke project plans, assess achievement, control execution
- Technical – define the activities that enable functions to optimize benefits and reduce risks of technical decisions and actions

© Copyright 2003 by R. Martin and E. Robertson

15288 - Process Hierarchy

Enterprise Environment Mgmt

System Life Cycle Processes Mgmt

Investment Mgmt

Resource Mgmt

Quality Mgmt

< 11p, 21o, 34a >

Project Planning

Project Assessment

Project Control

Decision-making

Risk Mgmt

Configuration Mgmt

Information Mgmt

< 16p, 35o, 61a >

Stakeholder Requirements Definition

Validation

Operation

Transition

Maintenance

Requirements Analysis

Verification

Architectural Design

Integration

Disposal

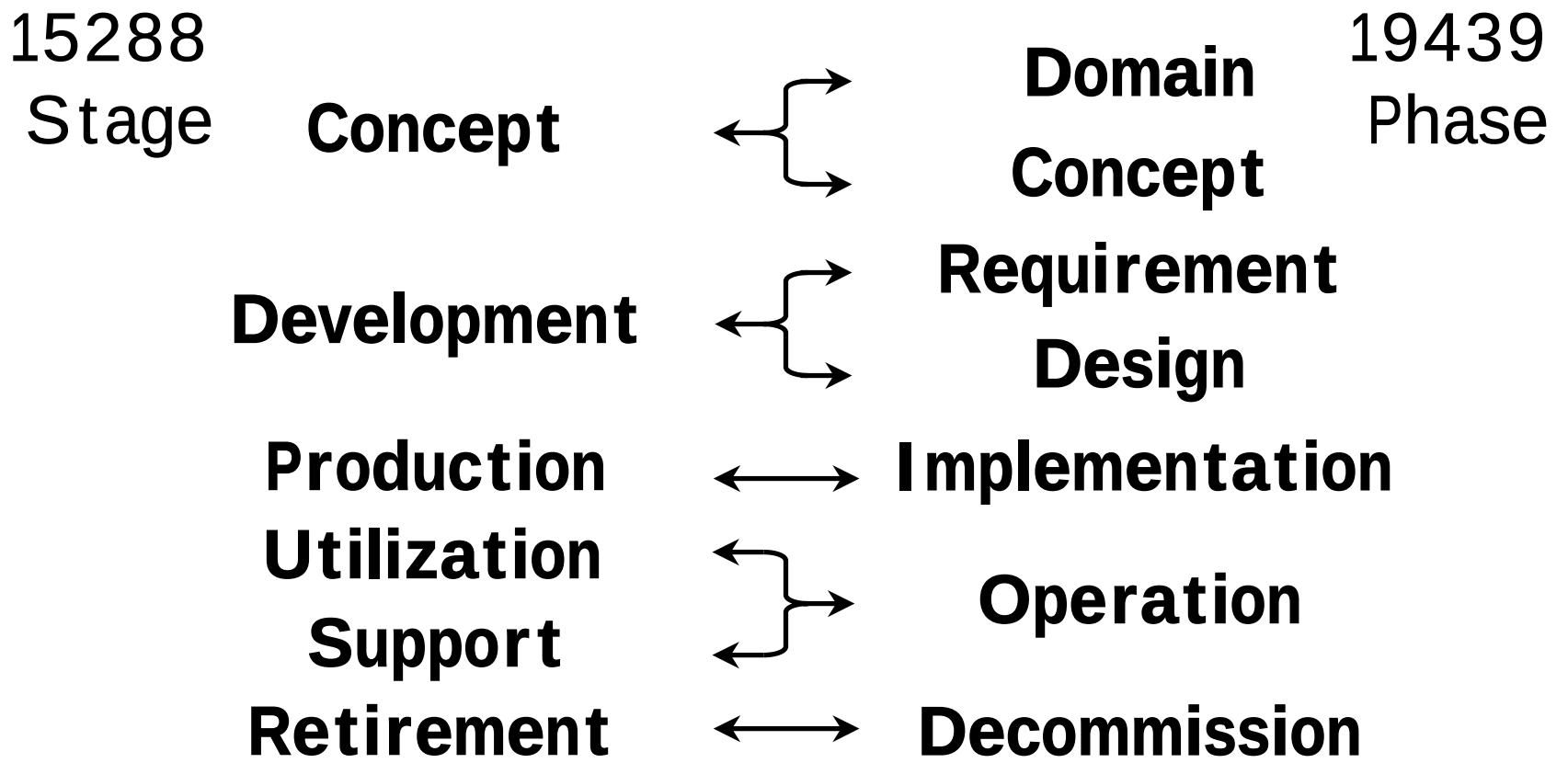
Implementation

< 34p, 53o, 96a >

© Copyright 2003 by R. Martin and E. Robertson

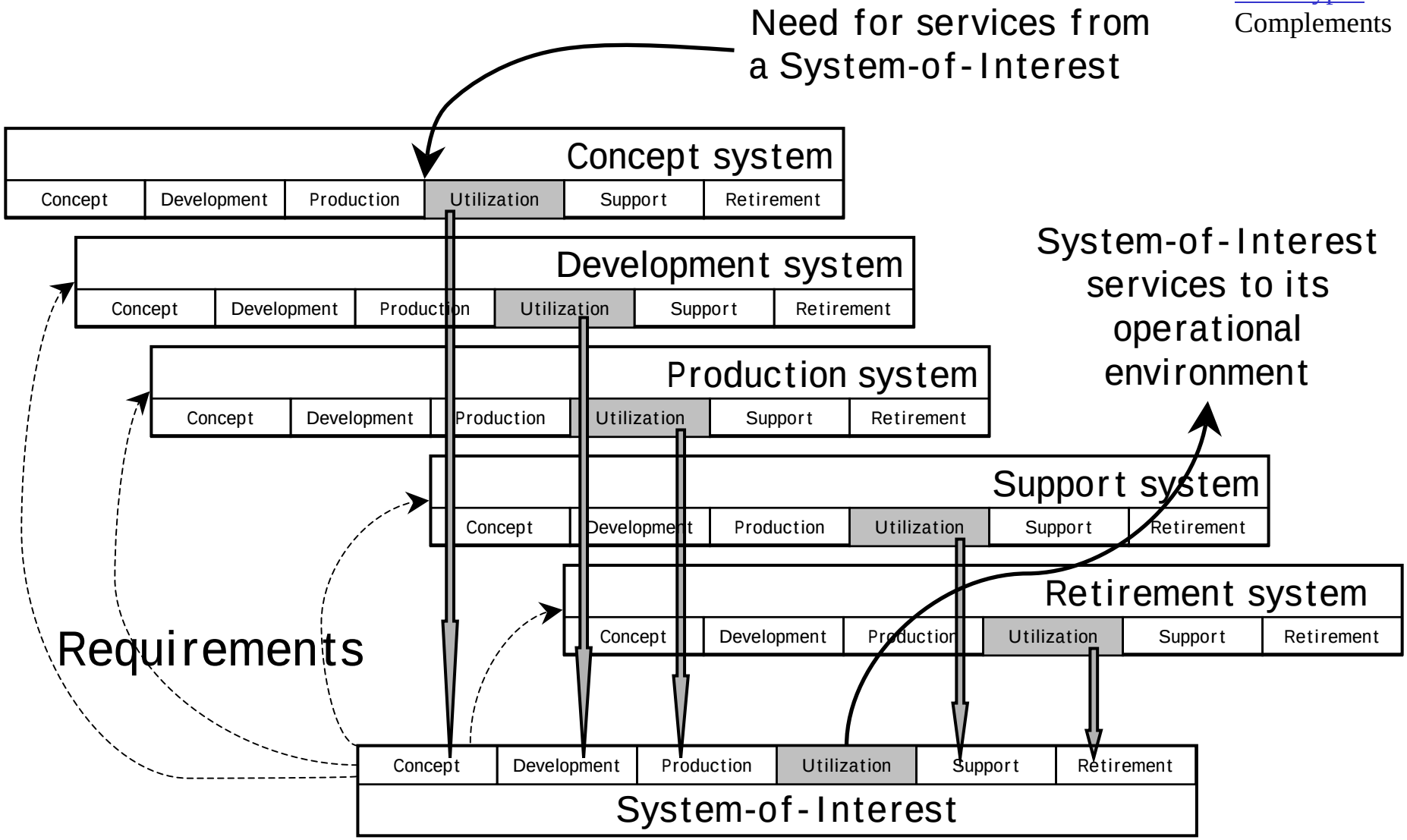
15288 - Life Cycle

Informative guidance for life cycle stages



© Copyright 2003 by R. Martin and E. Robertson

15288 - Recursion



© Copyright 2003 by R. Martin and E. Robertson

Archetype Dimension Summary

Zachman –
Role {Context, Owner, Designer, Builder, Out-of-context}
Interrogative {What, How, Where, Who, When, Why}

ISO\CEN FDIS 19439 –
Model {Domain, Concepts, Requirements, Design,
Implementation, Operation, Decommission}
View {Function, Information, Resource, Organization}
Genericity {Generic, Partial, Particular}

ISO 15288 –
Process Group {Agreement, Enterprise, Project, Technical}

© Copyright 2003 by R. Martin and E. Robertson

Prototype Models

Zachman - **interrogative models** {entity-relationship, input-process-output, node-link, people–work, time–cycle, ends–means}

Zachman - **cell models** {Semantic Model, System Design, Control Structure, Business Plan, etc.}

19439 - **constructs** {domain, business process, enterprise activity, event, enterprise object, resource, capability, decision centre, etc.}

19439 - **partial models** {industry sector, company size, national variation, etc.}

15288 – **process definitions** { 25 processes consisting of 63 purposes, 123 outcomes, and 208 activities (in 33 pages of text)

© Copyright 2003 by R. Martin and E. Robertson

Purposive Dimension

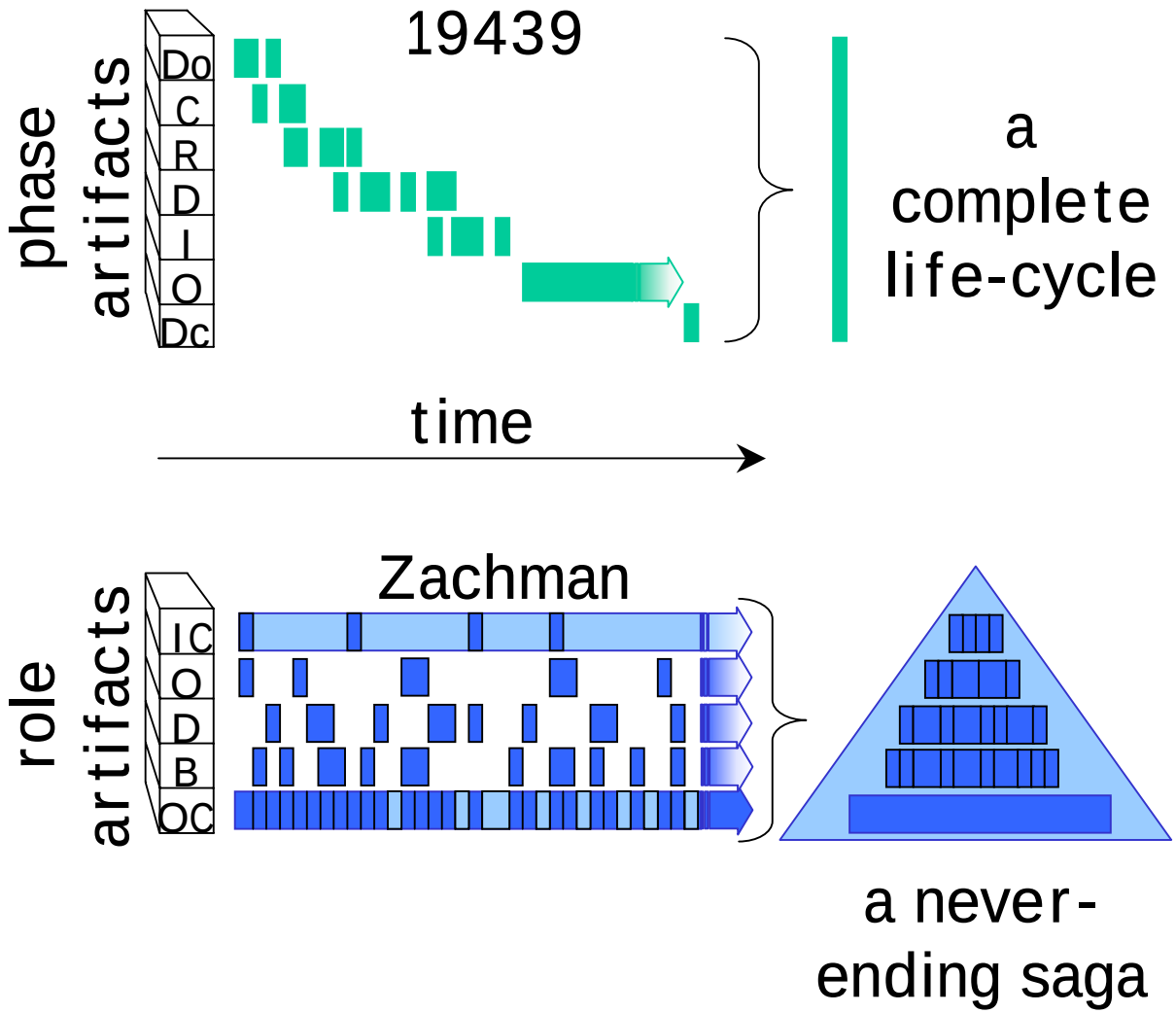
Zachman has a continuant purposive dimension (Role) and therefore serves well in an **analytic resource and reference mode**. It is always all there – either explicitly or implicitly.

19439 has an occurrent purposive dimension (Model Phase) and therefore serves well in a **realization and operational mode**. It provides the point-in-time solutions we use.

15288 has a decompositional purposive dimension (Process Group) with descriptive process artifacts **suitable for use in Zachman or 19439**.

© Copyright 2003 by R. Martin and E. Robertson

Different Life History

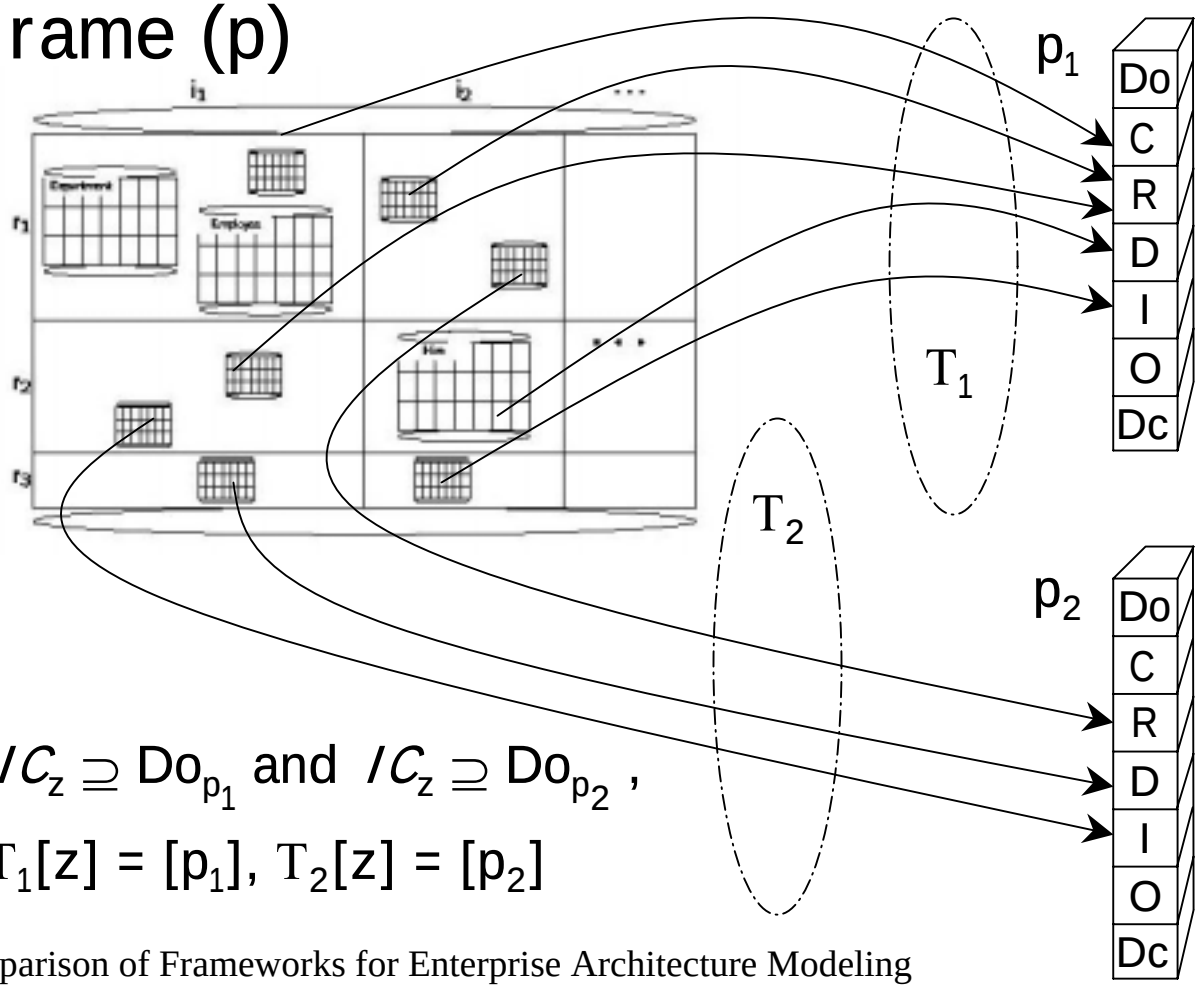


The appearance of artifacts in time imposes a temporal order on the purposive dimension of 19439, whereas the Zachman purposive dimension order is strictly the result of dependency among artifacts.

© Copyright 2003 by R. Martin and E. Robertson

Taking a Snapshot

A Zachman continuant frame (z) can participate in an 19439 occurrent frame (p)



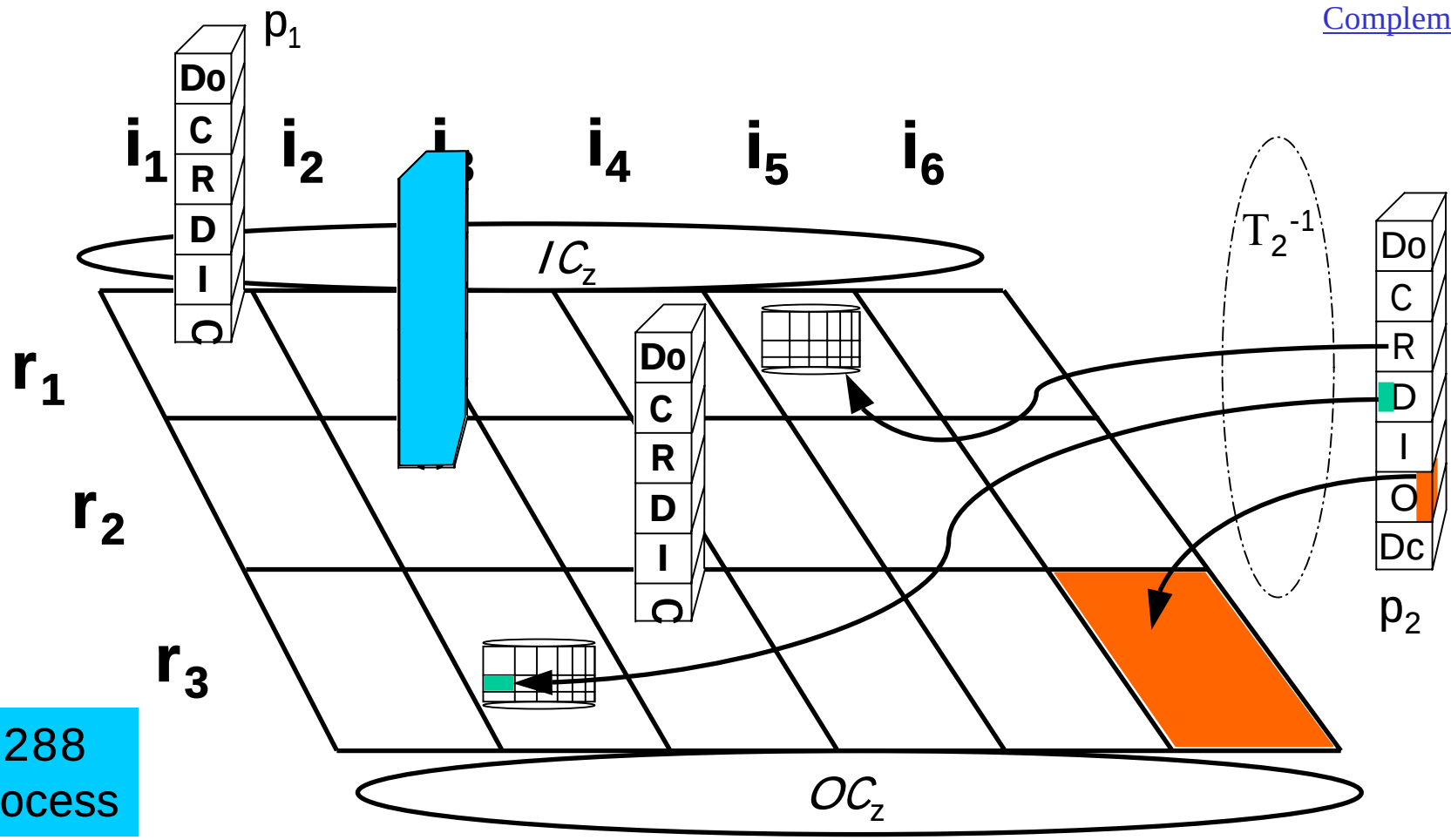
for $IC_z \supseteq Do_{p_1}$ and $IC_z \supseteq Do_{p_2}$,

$T_1[z] = [p_1]$, $T_2[z] = [p_2]$

15288 processes from "how" column map to p1 and p2 function views

© Copyright 2003 by R. Martin and E. Robertson

Populating with Artifacts



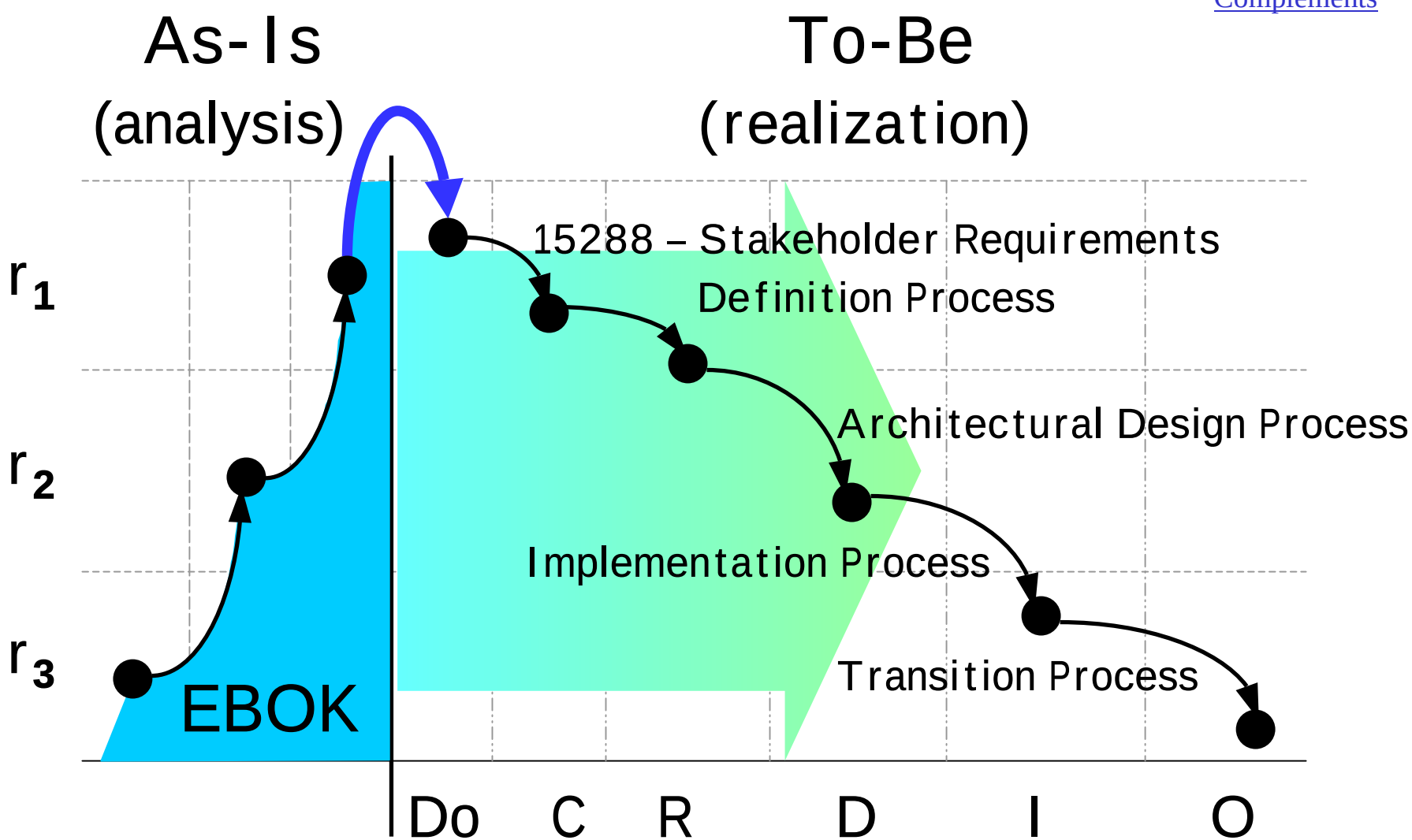
15288 process

for $IC_z \supseteq Do_{p_1}$ and $IC_z \supseteq Do_{p_2}$, $T_1^{-1}[p_1] \subseteq [z]$ and $T_2^{-1}[p_2] \subseteq [z]$

© Copyright 2003 by R. Martin and E. Robertson

Profile of Change

Principles
Observations
Archetypes
Complements



© Copyright 2003 by R. Martin and E. Robertson

Managing Change

To respond to a change in the environment of **z**, say widget W for customer C requires a new process P, we use components of continuant **z** to instantiate the occurrent **p** that realizes the new process operation in one of two ways:

$T_{W,C}[\mathbf{z}] = [\mathbf{p}_{W,C}]$ document the current P
$M : \mathbf{z} \rightarrow \mathbf{z}'$ modify **z** for new process
$T_{W,C}[\mathbf{z}'] = [\mathbf{p}'_{W,C}]$ create new process realization

or

$T_{W,C}[\mathbf{z}] = [\mathbf{p}_{W,C}]$ document the current P
$R_{W,C} : \mathbf{p}_{W,C} \rightarrow \mathbf{p}'_{W,C}$ realize new process P'
$T^{-1}_{W,C}[\mathbf{p}'_{W,C}] \subseteq [\mathbf{z}']$ document new **p** in **z**

© Copyright 2003 by R. Martin and E. Robertson

Comparative Summary

Zachman is the most comprehnsive of the three presented.

Zachman holds primitive models while 19439 extracts those primitives and composes views.

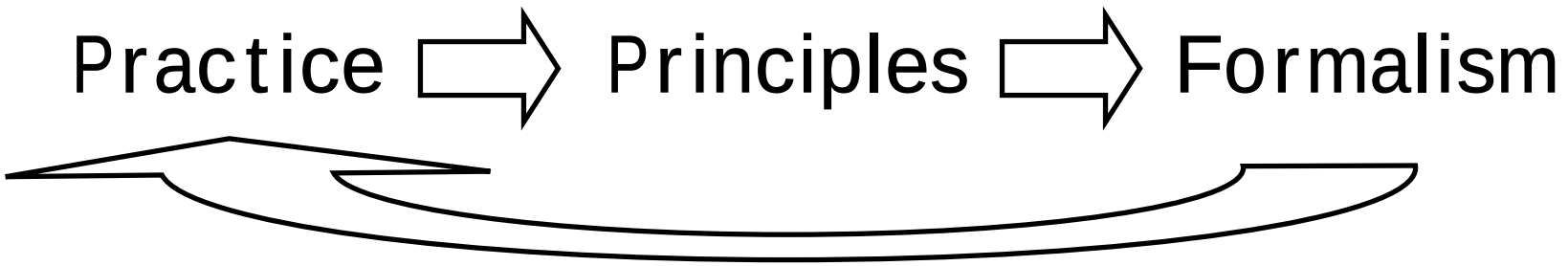
Zachman provides a conceptual partitioning as a major focus whereas the other two focus on support for methodological approaches.

© Copyright 2003 by R. Martin and E. Robertson

Approaching Frameworks

Goal is guidance for constructing and implementing frameworks.

Knowing the model space facilitates model reuse.



© Copyright 2003 by R. Martin and E. Robertson

References

A. T. Bahill and C. Briggs. The Systems Engineering Started in the Middle Process: A Consensus of Systems Engineers and Project Managers. *Systems Engineering*, 4(2), 2001.

C. Masolo, S. Borgo, A. Gangemi, N. Guarino, A. Oltramari, L. Schneider, The WonderWeb Library of Foundational Ontologies Preliminary Report, ISTC-CNR, Italy, 2002.

International Organization for Standardization. ISO 15704:2000 *Industrial automation systems – Requirements for enterprise-reference architectures and methodologies*. Geneva.

International Organization for Standardization and European Committee for Standardization. ISO/CEN FDIS 19439 ISO/CEN parallel enquiry draft prEN ISO 19439 – *Enterprise Integration – Framework for Enterprise Modelling*. Brussels and Geneva, 2003.

International Organization for Standardization and International Electrotechnical Commission. ISO/IEC 15288:2002 *Information Technology – Life Cycle Management – System Life Cycle Processes*. Geneva.

International Organization for Standardization and International Electrotechnical Commission. PDTR 19760 *Systems Engineering – Guide for ISO/IEC 15288 (System Life Cycle Processes).* Geneva, 2002.

J. A. Zachman. A framework for information systems architecture. *IBM Systems Journal*, 26(3), 1987

J. F. Sowa. *Knowledge Representation: Logical, Philosophical, and Conceptual Foundations*, Brooks Cole Publishing., Pacific Grove, CA 2000.
(also see http://www.jfsowa.com/ontology/toplevel.htm, September, 2003)

J. F. Sowa and J. A. Zachman. Extending and formalizing the framework for information systems architecture. *IBM Systems Journal*, 31(3), 1992.

R. Martin and E. Robertson. A Formal Enterprise Architecture Framework to Support Multi-model Analysis. Proceedings of the Fifth CAiSE\IFIP8.1 International Workshop on Evaluation of Modeling Methods in Systems Analysis and Design, Stockholm, 2000.

R. Martin and E. Robertson, Formalization of Multi-level Zachman Frameworks, 1999, http://www.cs.indiana.edu/Research/techreports/TR522.shtml

© Copyright 2003 by R. Martin and E. Robertson