Updating IEEE 1471

David Emery & Rich Hilliard*

WICSA 2008 Working Session 4
http://wwwp.dnsalias.org/wiki/WICSA2008_WS4_ArchitectureDocumentationFrameworks
Background

- IEEE Std 1471–2000, *Recommended Practice for Architectural Description of Software-intensive Systems*
- Became ANSI standard, 2001
- ISO adopted IEEE 1471 on a fast-track ballot, March 2006
  - published as international standard, July 2007
ISO/IEC 42010:2007

Systems and software engineering —
Recommended practice for architectural
description of software-intensive systems

Ingénierie des logiciels et des systèmes — Pratique recommandée pour la description architecturale des systèmes exigeant beaucoup de logiciels
Revision by ISO/IEC JTC1/SC7 WG 42

- ISO & IEEE will jointly revise the standard as...

- Revision basis:
  - 184 comments from fast-track ballot
Revision: must do

- Align with ISO life cycle process models:
  - ISO 15288 (systems)
  - ISO 12207 (software)
- Change scope from “software-intensive systems” to include “general systems”
Revision: play nice with ISO

- Harmonize with other ISO “architecture-related” standards
  - RM–Open Distributed Processing (ISO 10746*)
  - Enterprise Architecture (“GERAM” ISO 15704*)
Revision: Timeline

- Moscow SC7 Plenary
  - WD1 (July 2007)
- Montréal SC7 Interim (Oct 2007)
  - WD2 (March 2008)
- Berlin SC7 Plenary (May 2008)
  - joint with TC 184 (GERAM)
  - CD1
- China SC7 Interim (Oct 2008)
  - CD2
- India SC7 Plenary (May 2009)
  - FDIS 42010
Core Conceptual Model
Advances in Architectural Description (since 2000)

- Refine architectural rationale, support decision capture
- Relations on views: inter-view consistency, other uses
- Architectural Descriptions for multiple systems of interest
- Aspects in architectural description
Architectural Rationale & Decision Capture

Based on work from SHARK 2007
Revision: Fixes and Clarifications

- Clarify architectural models as major parts of views
- Clean up terminology and the “metamodel”
  - tiers: conceptual, core; extensions
- documents v. repositories?
- “architectural” v. “architecture description”?
Revision: Annexes

- More & better examples!
- Standard viewpoints?
  - scenarios (= use cases, change cases & “stakeholder cases”)
  - component & connector
  - behavioral
- Evaluation of architecture descriptions
One more thing...

Architecture frameworks

- Most Architects must work within an architecture framework

- Some existing frameworks
  - architecture methods: Kruchten’s 4+1; Hofmeister, Nord & Soni; Rozanski & Woods; ...
  - Zachman, TOGAF, DoDAF, MoDAF, ...
  - RM–ODP, GERAM, ...
Architecture frameworks

- architecture framework:
  - a predefined set of concerns, stakeholders, viewpoints, and viewpoint correspondence rules; established to capture common practice for architecture descriptions within specific domains or user communities

- New conformance points ("shall") for the Standard
Architecture frameworks

- **Stakeholder**: identifies as generic 1..*
  - is important to 1..*
  - holds 1..*
- **Concern**: identifies 1..*
  - frames 1..*
- **Architecture Framework**: defines 1..*
- **Architectural Viewpoint**: defines 1..*
  - relates 2..*
- **Architectural Description**: organized by 1..*
- **Architectural View**: governed 1
  - contributes to 1..*
  - consists of 1..*
- **Architectural Model**: sanctions 1..*
- **Viewpoint Correspondence Rule**: defines 0..*
Architecture frameworks & Conformance

- Conformance of a framework to Standard
  - identifies stakeholders, concerns, viewpoints, rules
  - metamodel reflects Standard metamodel
- Conformance of an AD to a framework
  - AD’s data includes that specified by framework definition
For more information...

- Visit web site, join users email group
- To participate in revision:
  - become an IEEE reviewer of revision drafts, or
  - join your ISO national member body

http://www.iso-architecture.org/ieee-1471/
Reviewing Architectural Descriptions

WICS 2008 Workshop

WG 42 Interests

- Is Review of Architectural Descriptions ripe for standardization?
- Can we consider this in on-going revision of ISO 42010 (né IEEE 1471)?
- Can we express it in a “process–neutral” manner?
- Is current conceptual model adequate to capture evaluation?
WG 42 Work Program

- 42000 series on architecture
- possible future work
  - standard viewpoints
  - architecture evaluation/assessment
  - processes for architecting
- ontologies
- 42000 branded items
ISO/IEC 42000 Certification

- Guarantees high quality architecture practices
- Suggests risk-reduction for both suppliers and acquirers
- “Improves World trade”
Relations between Views

Rich Hilliard
r.hilliard@computer.org

http://wwwp.dnsalias.org/wiki/Wicsa7:BOF:Relations_between_Views
IEEE 1471:2000 requires analysis and recording of any inconsistencies between views.

Can we do better in ISO 42010 revision?
Current proposal (WD1)

- Introduces new mechanism, view correspondences (VC)
  - records a relation between two architectural views
  - used to capture: a consistency relation, a traceability relation, a constraint or obligation of one view upon another
Current proposal: VC example

Consider two views of a system, $S$, a software component view, $SC(S)$, with software elements, $e_1, \ldots, e_6$, and a hardware view, $HW(S)$, with hardware platforms, $p_1, \ldots, p_4$.

A view correspondence expressing which software elements execute on which platforms might be:

ExecutesOn = { $(e_1, p_1), (e_1, p_4), (e_2, p_2), (e_2, p_3), (e_3, p_3), (e_4, p_4), e_6, p_2$ }
Current proposal: VCs & VCRs

- A viewpoint correspondence rule (VCR) expresses a contract between two architectural viewpoints, realized by a VC.
- VCR either holds in its VC, or is violated by the VC.
- Example: Every software element, $e_i$, as defined by SC(S), must execute on one or more platforms, $p_j$, as defined by HW(S).
Beginnings of a model

**Viewpoint**
- name
- etc.

2 links

**Viewpoint Correspondence Rule**
- expression

0..1

**View**
- name
- etc.

2 applies to

**View Correspondence**
- name
- optional: VCR
- items ⊂ V1×V2

1 realizes
Issues to consider

- Have we got the right (all) use cases?
  - Can we make a taxonomy of VCs and use cases?
- VCs are binary mathematical relations
  - functions too restrictive
- What is the language for expression of VCRs?
- Terminology (e.g., some folks don’t like “correspondence”)