

Updating IEEE 1471: architecture frameworks and other topics

David Emery
DSCI, Inc.
demery@dsci-usa.com

Rich Hilliard
r.hilliard@computer.org

Abstract

The purpose of this working session is to solicit feedback from the software architecture community for the revision of IEEE Std 1471 (now also ISO/IEC 42010), to identify topics ripe for standardization and to get reactions on current proposals under consideration for the revision.

This paper briefly outlines some of the revision proposals under consideration and provides some detail about one major area: architecture frameworks.

The WICSA Wiki will be used to solicit participation, capture discussions and insights, and organize the working session. Findings and results from the session will then be recorded there.

1. Background

IEEE 1471, *Recommended Practice for Architectural Description of Software-Intensive Systems* [6] has been in active use since 2000. Its original design goals were set forth in [4]. If Google hits are any indication, IEEE 1471 has met several of its original goals:

1. to establish a frame of reference of terms and concepts for architectural description;
2. to codify best practices for the architectural description of software-intensive systems; and
3. to serve as one basis for evolution of thinking in the field.

In March 2006, IEEE 1471 was adopted by ISO as an international standard. ISO and IEEE will jointly revise the standard as ISO/IEC 42010 with the new title, *Systems and Software Engineering—Architectural Description*.

The revision is being carried out by Working Group 42,¹ Architecture, of ISO/IEC JTC1/SC7, *Software and Systems Engineering*. WG42 will produce a series of working drafts

¹42 because Architecture is the answer to Life, the Universe and Everything.

for comment, with a goal of a first ballot in 2009. Working draft WD1 was released in July 2007. Interested parties can participate in the ISO revision through their national member bodies. Individuals may also participate through the parallel IEEE Architecture Working Group (<http://www.iso-architecture.org/ieee-1471/>).

2. Revision topics

A number of topics are under consideration for the revision of ISO/IEC 42010. We briefly summarize them here.

Terminology Maintenance: The revision is an opportunity to clarify terms, concepts and requirements that were not clearly expressed in the 2000 edition. Based on comments and questions about the standard and review of the literature, the following terms in IEEE 1471 seem to cause confusion for some users: *architectural concern*, *architectural model*, and *architectural viewpoint*. WG42 will work within the ISO context to align its architecture-related terminology with other ISO vocabularies.

Scope of Application: The scope of IEEE 1471 is software-intensive systems. ISO SC7 has a wider charter of *systems in general*. The revision will investigate broadening the scope to match that of ISO SC7's charter.

Alignment with ISO life cycle models: IEEE 1471 was designed to be life-cycle neutral. The revision will retain this principle, but must also ensure its usability within the ISO life cycle process models (ISO 12207 for Software Engineering and ISO 15288 for Systems Engineering).

Conceptual Evolution: One widely referenced part of IEEE 1471 is its conceptual model (sometimes called the "IEEE 1471 metamodel"). The revision will investigate whether there are terms and concepts that have come into use for architectural description since the 2000 edition appeared that are ready for standardization.

Two areas that are under consideration in this regard (and have been the subject of recent workshops) are:

- the role of aspects in architectural descriptions [3]
- architectural decision capture and rationale [1]

Architecture Evaluation: Since the release of IEEE 1471, users have asked, how does this relate to the evaluation of architectures? While there is a large body of work on architecture evaluation (e.g., ATAM, SARA), there is little work on review of the artifacts that convey those architectures. Work on this topic could lead to an annex on review of architectural descriptions in the revised standard, or perhaps new work items for WG42 pertaining to architecture evaluation and review.

3. One more thing... Architecture Frameworks

“Architecture framework” is a very widely used term (249,000 Google hits). Many software and systems architects are expected to work within an architecture framework adopted by their organization or required by their clients (or sometimes both). For example, there are:

1. enterprise architecture frameworks such as Zachman’s information architecture framework, the US DoDAF and UK MoDAF, The Open Group’s Architecture Framework;
2. architectural practices embodied in other standards such as the *Reference Model for Open Distributed Processing* (RM-ODP) [ISO 10746], and the *Generalized Reference Architecture and Methodology* (GERAM) [ISO 15704 and ISO 19439]; and
3. architecting methods not explicitly called frameworks, such as: Kruchten’s 4+1 view model [7], Rozanski and Woods’ software systems architectural approach [8] and the “Siemens approach” [5].

Although the idea of an architecture framework was implicit within the 2000 edition, the notion was not defined in the standard. For the revision, WG42 is considering whether there is an adequate basis to standardize such a definition. The motivation for this is to provide a means to define (existing and future) architecture frameworks in a uniform manner. A uniform basis for architecture frameworks would permit sharing of information and interoperability between frameworks and thereby improve understanding and inter-working across different architecture communities.

Working draft WD1 makes an initial proposal for architecture frameworks. The highlights of that proposal are presented in the remainder of this section. The proposed definition is:

An *architecture framework* establishes a common practice for creating, organizing, interpreting and analyzing architectural descriptions used within a particular domain of application or stakeholder community.

Adding a definition of architecture framework creates opportunities for new conformance points within ISO 42010. In IEEE 1471, an architectural description (AD) could conform to the standard. Now it is possible to define:

- conformance of an architecture framework to the standard;
- conformance of an AD to an architecture framework;

Figure 1 shows how this concept would fit into the “metamodel”.

3.1. Conformance of a framework

An architecture framework definition shall include:

- the identification of one or more architectural concerns;
- the identification of one or more generic stakeholders holding those architectural concerns;
- the definition of one or more architectural viewpoints which frame those architectural concerns; and
- the definition of zero or more viewpoint correspondence rules over those viewpoints (see below).

In a predefined viewpoint, a *generic stakeholder* is identified to establish concerns that the viewpoint frames. Generic stakeholders will be instantiated as specific stakeholders within an actual architectural description using the framework. The definition of each viewpoint in a framework must follow the existing ISO 42010 requirements on viewpoints.

Metamodel Conformance: One might go further, and exploit the IEEE 1471 metamodel as a basis for conceptual conformance, as follows:

When the definition of an architecture framework includes a metamodel, that metamodel shall reflect the Core Model (familiar class diagram of the metamodel from IEEE 1471).

A metamodel M_1 *reflects* a metamodel M_0 if and only if: all classes in M_0 occur in M_1 , and all the class associations in M_0 occur in M_1 , with the same roles and multiplicities.

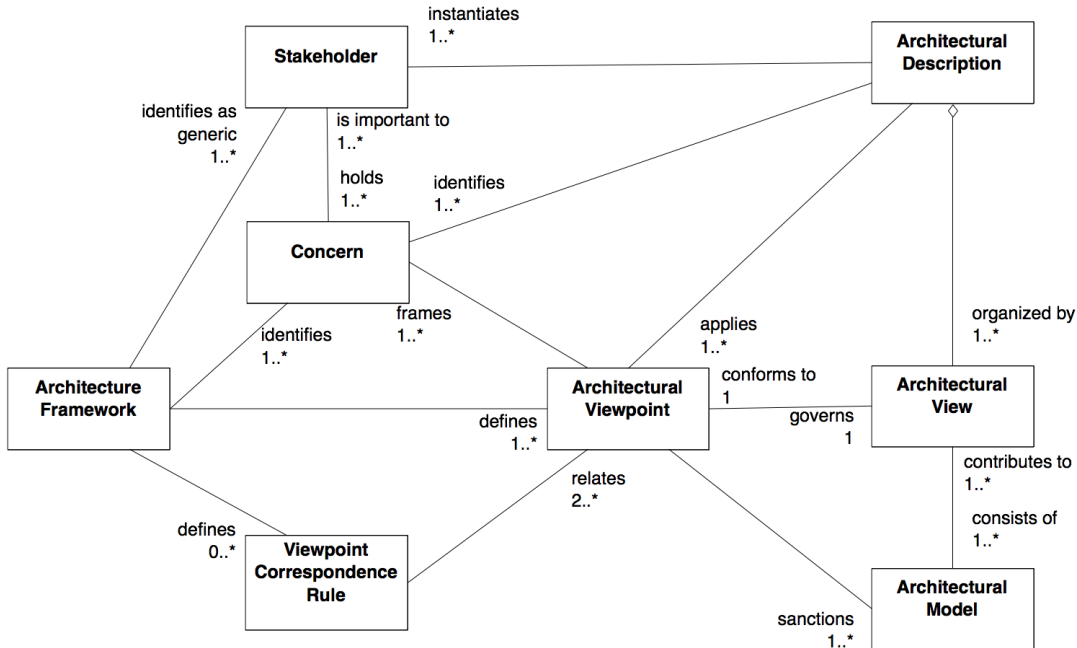


Figure 1. Adding Architecture Framework to the metamodel

3.2. Conformance to a framework

An architectural description, *AD*, conforms to an architecture framework, *AF*, if and only if:

- *AD* conforms to the requirements in ISO 42010 for any architectural description;
- the architectural concerns identified in *AD* include those defined by *AF*;
- each of the generic stakeholders of *AF* has been instantiated among the identified stakeholders in *AD*;
- there is a conforming architectural view in *AD* for each predefined viewpoint in *AF*; and
- there is a view correspondence in *AD* for each viewpoint correspondence rule defined by the framework *AF*.

3.3. Relations between views

A consequence of using multiple views for architectural description is the need to express and maintain consistency among those views. In the 2000 edition, the only requirement on consistency is that an AD must record any known inconsistencies among its views. At that time, there was no well-established practice to be codified for expressing or enforcing consistency. WD1 proposes a mechanism of *view*

correspondence to record a relation between two architectural views. A view correspondence may be an instance of a *viewpoint correspondence rule*.

As an example, consider two viewpoints on a system, a Logical Viewpoint, defining software functionality, and a Deployment Viewpoint, defining the physical configuration of processors, networks and other computing elements in an operational environment on which that software will execute. A useful viewpoint correspondence rule might be:

Every software element identified in the Logical View must be allocated to at least one computational element in the Deployment View.

This rule could be applied to an individual AD or as a general rule on any AD using those two viewpoints. Within an AD, this rule could be realized by a table listing the software elements and their computational hosts, or by a diagram showing each computational element and the software running on those elements.

Consistency is one application for view correspondences; another is view composition. Boucké and Holvoet suggest several kinds of relations for view composition which could be handled with view correspondences [2].

3.4. Managing frameworks

With the addition of the concept of architecture framework, we can consider how architecture frameworks are used in practice. Organizations often mandate the use of a

specific architecture framework (e.g., DoDAF or MoDAF), while its architects may feel more comfortable working within another framework (e.g., 4+1). In such cases, an early decision for the architect is how to use these pre-existing frameworks within an approach that works for the project. A benefit of the proposal described here is that it facilitates merging of frameworks in a principled manner by aligning each framework to the standard model and using stakeholders and concerns to structure the solution.

In practice, many architectural descriptions include some framework-like content. Why not require in the standard that every architecture description must contain (exactly) one framework? There are perhaps some real advantages to doing this. The framework becomes a “container” for stakeholders, concerns, viewpoints and correspondence rules. The conformance points in the standard can make use of this, and significantly, evaluation practices can also make use of this, breaking evaluation into: (1) Evaluate the framework; and then (2) Evaluate the views and other content against the framework.

4. Working Session

This working session is seeking participants from the user community to offer feedback on experiences using IEEE 1471:2000, reactions to the proposals sketched above, and any other insights into architectural description. Prior to the working session, this paper is being posted to the WICSA 2008 Wiki where interested parties can contribute. Following the session, the Wiki will serve to capture all participants’ contributions.

Questions to stimulate participant reactions:

Conceptual (meta) model: *Are there terms that should be added to the standard? Are there definitions to be added?*

Are there fixes, improvements, or items totally missing from the conceptual model that should be considered?

Are there new ways of conceptualizing architectural descriptions different from what is in the conceptual model?

Scope of the standard: *Are there things one does differently when architecting software, systems, enterprises that should be captured in the revision?*

Evolution: *Is there recent work in architecture which “breaks” the conceptual model?*

Are there advances in the state of the art in architecture that need to be addressed? Can they be handled within ISO/IEC 42010?

Are there approaches to defining viewpoints that could be codified at this time?

Has the state of the practice advanced sufficiently to establish any standard viewpoint definitions? Candidates could be a components and connectors viewpoint, behavioral viewpoint, or scenarios (including use cases, change cases, and “stakeholder cases”).

There is a process implied by the standard in the connection from stakeholders to concerns to viewpoints to insure completeness. Are there other, similar methodological insights that should be built into the requirements on an AD?

Does the proposal for architecture frameworks map to users’ actual experiences? If you have defined an architecture framework or related architectural approach, does your approach fit this model?

What is the experience in merging existing architecture frameworks? Where have you found conflicts? Would this model help you to find and resolve the conflicts?

References

- [1] P. Avgeriou, P. Kruchten, P. Lago, P. Grisham, and D. Perry. Architectural knowledge and rationale: issues, trends, challenges. *SIGSOFT Software Engineering Notes*, 32(4):41–46, 2007.
- [2] N. Boucké and T. Holvoet. View composition in multi-agent architectures. *International Journal of Agent-Oriented Software Engineering*, 2007. Special issue on Multi-agent systems and software architecture.
- [3] P. Clements, D. Emery, R. Hilliard, and P. Kruchten. Aspects in architectural description: report on a first workshop at aosd 2007. *SIGSOFT Software Engineering Notes*, 32(4):33–35, 2007.
- [4] W. J. Ellis, R. Hilliard, P. T. Poon, D. Rayford, T. F. Saunders, B. Sherlund, and R. L. Wade. Toward a recommended practice for architectural description. In *Proceedings of 2nd IEEE International Conference on Engineering of Complex Computer Systems, Montreal, Quebec, Canada, October 21–25, 1996*, 1996.
- [5] C. Hofmeister, R. L. Nord, and D. Soni. *Applied Software Architecture*. Addison-Wesley, 2000.
- [6] IEEE. *ANSI/IEEE Std 1471–2000 Recommended Practice for Architectural Description of Software-Intensive Systems*, October 2000.
- [7] P. B. Kruchten. The 4+1 view model of architecture. *IEEE Software*, 28(11):42–50, November 1995.
- [8] N. Rozanski and E. Woods. *Software Systems Architecture: Viewpoint Oriented System Development*. Addison Wesley, 2005.