Impact Assessment of IEEE 1471 on The Open Group Architecture Framework

Rich Hilliard*
Director of Architecture
Integrated Systems and Internet Solutions, Inc.
Concord, Massachusetts USA
rh@isis2000.com
+1 978 318 0000

March 30, 2000

Abstract

At the request of John Spencer, this note was prepared to assess the expected impact of adopting IEEE 1471, *Recommended Practice on Architectural Description* on The Open Group's Architecture Framework (TOGAF).

1 Introduction

This note is an impact assessment of the adoption of IEEE 1471, *Recommended Practice for Architectural Description* on The Open Group's Architectural Framework. It was prepared at the request of John Spencer, in anticipation of the April meeting of The Open Group in London.

The remainder of this section outlines the salient characteristics of the TOGAF and of IEEE 1471. Subsequent sections address expected impact from several points of view: conceptual compatibility, terminological compatibility, relevant artifacts, and conformance issues.

TOGAF. TOGAF is The Open Group Architectural Framework [7]. It consists of two principal ingredients:

- "The TOGAF Architecture Development Method (ADM), which explains how to derive an organization-specific IT architecture that addresses business requirements. The ADM provides:
  - A reliable, proven way of developing the architecture

*The opinions expressed herein are solely those of the author and not of his employer, the IEEE, the IEEE Architecture Working Group, or The Open Group.

  - Architecture views which enable the architect to ensure that a complex set of requirements are adequately addressed
  - A worked example and linkages to practical case studies
  - Tools for architecture development" [FAQ][1]

- "The TOGAF Foundation Architecture—an architecture of generic services and functions that provides a foundation on which specific architectures and architectural building blocks can be built. This Foundation Architecture includes:

  The TOGAF Standards Information Base (SIB), a database of open industry standards that can be used to define the particular services and other components of an organization-specific architecture. " [FAQ]

IEEE 1471. IEEE 1471 is the Draft *Recommended Practice for Architectural Description* [4].[2] IEEE 1471 represents an emerging consensus for the description of the architectures of software-intensive systems. It was developed by the IEEE's Architecture Working Group, chartered and sponsored by the Software Engineering Standards Committee of the IEEE Computer Society. The draft *Recommended Practice* was produced between 1995 and 1998 by a group of approximately thirty participants, and over 150 international reviewers. A *Guide* to the *Recommended Practice* is currently in development.

IEEE Goals for 1471. Given the widespread interest in the architecture of software-intensive systems, IEEE recognized the need for providing direction in this area, for both industry and academic application. IEEE set the following goals for the standard:

1. **To take a "wide scope" interpretation of *architecture* applicable to software-intensive systems.** This includes computer-based systems ranging from software applications, information systems, embedded systems, systems-of-systems, product lines and product families—wherever software plays a substantial role in the development, operation, or evolution of a system.

2. **To establish a conceptual framework and vocabulary for systems architectural description.** Despite widespread interest in architecture in both the systems and software engineering communities, there is no common frame of reference, no agreed-upon definitions for terms such as "architecture," "architectural description," and "view."

3. **To identify and promulgate sound architectural practices.** There are already a wide range of software and systems architecture practices. It is a goal of IEEE 1471 to provide a basis on which current practices may be defined, applied, and interoperate.

[1] References in brackets like this are to portions of the TOGAF.

[2] At the time of this writing, IEEE 1471 has passed ballot by the IEEE; it is expected to be approved for use by mid–2000. Up-to-date information about IEEE 1471 can be obtained from the IEEE Architecture Working Group web site (http://www.pithecanthropus.com/~awg).

4. **To allow for the evolution of those practices as relevant technologies mature.** The IEEE recognized that software systems architectural practices are rapidly evolving, both in industrial use and in the research arena, with respect to technologies such as architecture description languages, architectural methods, analysis techniques, and architecting processes. It is hoped these practices can be communicated, documented and shared via the framework of IEEE 1471. For this reason, the framework is intended to be general enough to encompass current techniques and flexible enough to evolve.

Using IEEE 1471. IEEE 1471 is a *recommended practice*—which is one type of IEEE standard.[3] The important ingredients of IEEE 1471 are:

1. a normative set of definitions for terms including *architectural description*, *architectural view*, *architectural viewpoint*;

2. a conceptual framework which establishes these terms in the context of the many uses of architectural descriptions for system construction, analysis and system evolution; and,

3. a set of requirements on an architectural description of a system.

IEEE 1471 applies to *architectural descriptions* (ADs)—any collection of products that purports to describe the architecture of a software-intensive system. An AD is said to *conform* to IEEE 1471 if it meets the requirements of IEEE 1471.

Requirements in IEEE 1471 are signalled with *shalls*, following usual standards practice. In this way, ADs may be readily checked for conformance to the recommended practice. The requirements of IEEE 1471 are designed to be independent of any individual architectural technique, and therefore should be applicable within a variety of architectural methods and architecture frameworks.

IEEE 1471 neither describes nor requires any kind of conformance of systems, projects, organizations, processes, methods, or tools – which are the province of individual methods, frameworks (such as TOGAF) and practicing organizations.

2 Impact Assessment—Overview

The remainder of the document is organized by topics, where I see a potential impact of IEEE 1471, on TOGAF. These are organized as follows:

- Scope/Intent
- Conceptual Compatibility
- Terminological Compatibility
- Conformance

[3] There are three types of IEEE standard: (i) standards, (ii) recommended practices and (iii) guides.

3 Impact Assessment—Compatibility of Scope

Scopes: By *scope* I mean the intended field of application of the two efforts.

IEEE 1471 addresses the architectural description of software-intensive systems. "TOGAF is designed to support ... the development of a technology (IT) architecture." [FAQ] Technology architectures in the sense of TOGAF certainly fall within the scope of IEEE 1471's intended domain of application—since IT systems are a kind of software-intensive system.

Conversely, in TOGAF terms, IEEE 1471 may be used to produce architectural descriptions of architectures across the Enterprise Continuum to document the Foundation Architecture, Common System Architectures, Industry Architectures, and Organization Architectures. [Part III] This is because for each kind of architecture its concerns (called "characteristics," in Part III), may be identified, and institutionalised, as a set of reusable viewpoints.

Therefore the intended usage, and applicability of IEEE 1471 and TOGAF are compatible.

Assumptions: TOGAF defines an architectual development method (ADM) which "describes the process of moving from the TOGAF Foundation Architecture to an organization-specific architecture (or set of architectures), leveraging the elements of the TOGAF Foundation Architecture and other relevant architectural components and building blocks along the way." [Part II] Part of the method is an "Architecture Development Cycle" for accomplishing the method.

IEEE 1471 is designed to be life cycle neutral, process neutral and method-neutral. It does not assume a particular life cycle, particular architectural methods, or techniques for architectural development. IEEE 1471 is also "notation-independent"—it does not specify any particular notations to be used in an architectural description, leaving this to individual architectural methods or practices.

Therefore, IEEE P1471 does not make any assumptions about process, method, technique or notation which would be incompatible with those specified by the TOGAF Architecture Development Method (ADM), or its life cycle—The Architecture Development Cycle.

4 Impact Assessment—Conceptual Compatibility

This section discusses the key concepts of IEEE 1471 to assess their compatibility with TOGAF underlying concepts.

The IEEE 1471 Conceptual Framework Figure 1, adapted from IEEE 1471, depicts the major conceptual entities referred to by the standard. The central abstraction, and primary focus, of the standard is Architectural Description. In IEEE 1471, an Architectural Description is a collection of products to document the architecture of a system. IEEE 1471 does not specify the format or media for an architectural description. What IEEE 1471 *does* specify is certain minimal required content of an AD reflecting current practices and industry consensus.

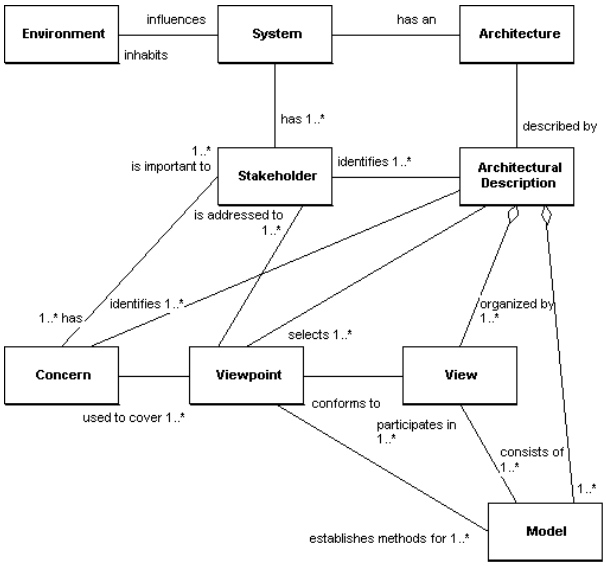

Figure 1: The IEEE 1471 Conceptual Model

A key tenet of that consensus is the notion of multiple views. In 1471, an Architectural Description is organized into one or more architectural Views. Most architectural methods and frameworks advocate the use of one or more views of the system as a part of the architectural description. However, the exact views used vary from technique to technique. Rather than require a particular set of views, IEEE 1471 leaves this selection to users of the standard.

One of IEEE 1471's contributions is to make explicit the notion of an architectural Viewpoint to embody the rules governing a view. It is anticipated that this will allow the definition and reuse of viewpoints, so that varying approaches to architecture may better be able to exchange results, and that in general the growth of the discipline will be facilitated by codifying certain useful patterns of description.

The IEEE 1471 conceptual framework appears to be "upward compatible" with the conceptual framework assumed by TOGAF. By *upward compatible* I mean that it makes finer distinctions about certain concepts of interest than TOGAF. If adopting IEEE 1471 for use, TOGAF could make these distinctions with minor impact. These distinctions are detailed in the remainder of this section.

Architecture v. Architectural Description: The IEEE 1471 conceptual model distinguishes an *architecture* from an *architectural description*. The architecture of a system is *conceptual*—perhaps not even written down—whereas an architectural description is a tangible, concrete engineering artifact which records an architecture. IEEE AWG and users of IEEE 1471 have found it very useful to make this distinction in practice. IEEE 1471 specifies requirements on architectural descriptions.

Statements like the following blur the distinction between an architecture and archi-

tectural description—an "architecture is a set of elements (sometimes called building blocks) depicted in an architectural model, and a specification of how these elements are connected to meet the overall requirements of an information system." [Introduction to the Architecture Development Method (ADM)]

I recommend that TOGAF usage of the term "architecture" be clarified in this regard; and that the term "architectural description" be used systematically when referring to artifacts.

Architecture v. View: TOGAF defines "four types of architecture that are commonly accepted as subsets of an overall Enterprise Architecture" [FAQ]:

1. business architecture;
2. technology (IT) architecture;
3. data/information architecture; and
4. application (systems) architecture;

Elsewhere in the document, the architect is directed to "Consider different architectural views" [ADM Phase C, Step 2]. TOGAF recommends a number of views [IV Architecture Views]:

- the Function View; various *implementation views*:
- the Management View;
- the Security View;
- the Builder's View;
- the Data Management View;
- the User View; and the following *physical views*:
- the Computing View; and
- the Communications View

Within the IEEE 1471 framework of use, both of these sets above would be considered views. So for a given system of interest[4], a *business view*, a *technology (IT) view*, a *data/information view* and an *application (systems) view* might be prepared; together these views would comprise the architectural description. There is experience in using IEEE 1471 leading exactly to views like this [1, 5, 6] Similiar views are produced in other approaches as well (e.g., RM-ODP).

Although "architecture" is widely used informally for "view", I recommend that TOGAF when trying to provide guidance to others be clear that these are *views* of a system.

[4]That system of interest may be an organization-specific system, or a more general system within the Enterprise Continuum of TOGAF.

View v. Viewpoint: IEEE 1471 distinguishes views of a system and the viewpoints from which those views are constructed. IEEE 1471 makes explicit the notion of an architectural Viewpoint to embody the rules governing a view. It is anticipated that this will allow the definition and reuse of viewpoints, so that varying approaches to architecture may better be able to exchange results, and that in general the growth of the discipline will be facilitated by codifying certain useful patterns of description. To the extent that it is helpful to capture the recurring (reusable) resources for describing a particular kind of view, it is useful to *declare its viewpoint*. (See the standard [4] for the requirements on declaring a viewpoint. For discussion, see [2].)

I recommend that TOGAF declare the viewpoints (IAW P1471) that it wants to promulgate for reuse. TOGAF can make a valuable contribution to the community by serving as a "repository" for viewpoints in its Resource Base. For purposes of tailoring, or extending TOGAF, users should be permitted to develop new viewpoints, as long as they are declared and used in conformance with the requirements of IEEE 1471.

Stakeholders and Concerns: IEEE 1471 uses the concepts Stakeholder of a system, and the stakeholders' concerns for the system in a number of ways.

Frequently it is difficult to discover/identify business objectives or environmental constraints in the abstract. By personifying them in terms of specific stakeholders of the system, it is much easier to articulate these objectives, constraints, and particular concerns. There is another reason for considering stakeholders in this context; frequently individual stakeholder will have different, perhaps conflicting needs (or requirements) for the system. These must be discovered as early as possible, and frequently managed by the Architect.

In the ADM process, the following activities may be impacted/improved by introduction of the stakeholders abstraction:

- "Confirm that the business goals and objectives are met"
- "Determine criteria for specification selection'
- "Conduct a gap analysis"
- "Opportunities and Solutions ... identifies the parameters of change"

In addition, the IEEE 1471 requirements imply several checkpoints between an AD and its stakeholders, as follows:

- Each viewpoint must be selected to address one or more (stakeholder) concerns.
- Each resulting view must cover one or more concerns.
- Each concern must be covered by at least one view.

Therefore, the appropriateness of the architectural description (i.e., Is this useful to the system stakeholders, or just a bunch of diagrams?) is insured by the proper selection of viewpoints. Then coverage of the full set of documented stakeholders' concerns can

be checked against the resulting set of architectural views. The viewpoints guide this check by focusing one in where to look for the treatment of each identified concern.

This powerful mechanism—a concrete approach to traceability—can be readily applied within the TOGAF.

5 Impact Assessment—Terminology

Most of the issues of terminology between IEEE 1471 and TOGAF, simply mirror the conceptual issues already described above.

It appears that it would be little impact on TOGAF to adopt the terminology and definitions used by IEEE 1471, and apply them within the TOGAF.

6 Impact Assessment—Conformance

As noted above, the notion of conformance to IEEE 1471 is fairly narrowly defined. Conformance to IEEE 1471 applies to architectural descriptions only—not to methods, processes, architects, tool, notations, etc. So TOGAF, as an architectural framework, *cannot itself be conformant* with IEEE 1471.

However, TOGAF defines a number of artifacts (work products) to be produced. Some of these artifacts constitute architectural descriptions in the sense of IEEE 1471.

Therefore, to be compatible with IEEE 1471, TOGAF work products which are Architectural Descriptions, *should be conformant with IEEE 1471*.

The following artifacts defined by TOGAF could be managed as architectural descriptions, in the sense of IEEE 1471:

- business architecture
- technical architecture

The Foundation Architecture is an architecture. The TOGAF Foundation Architecture is "an architecture of generic services and functions that provides a foundation on which more specific architectures and architectural components can be built." [Part III]

Insofar as the Foundation Architecture *is* an architecture and is intended for use by practitioners of ADM to create individual organization-specific architectures, it should be documented via an architectural description, in conformance with IEEE 1471.

Is the baseline description an Architecture? If so, and TOGAF seeks to document it; that resulting architectural description should be conformant with IEEE 1471.

7 Impact Assessment—Impacts on Architectural Tools and Resources

Although the following is stated as a need for ADML— "Views - a model must have the ability to 'morph' other views, including logical, physical, and organization views."— ADML, based on Acme, is insufficient to capture a multi-viewpointed AD in the sense of IEEE 1471. This is discussed in [3].

There is an on-going effort to create a markup language based on IEEE 1471— MLAD, MARKUP LANGUAGE FOR ARCHITECTURAL DESCRIPTION. Perhaps the ADML and MLAD efforts should coordinate. An excerpt from the MLAD schema is shown in an appendix.

8 Conclusion

The conclusion of my brief impact assessment of the adoption of IEEE 1471 by TOGAF may be summarized as follows:

- The scope of applicability and conceptual bases of IEEE 1471 and TOGAF are sufficiently similar that it is possible to consider adoption without major rework.
- Since IEEE 1471 is process- and method-neutral, there is no significant impact to ADM processes in its adoption.
- The IEEE 1471 conceptual framework is upward compatible with the conceptual assumptions of TOGAF. Some finer-grained distinctions made in the IEEE 1471 conceptual framework may be of use in TOGAF.
- Terminological compatibility follows largely from conceptual compatibility. The IEEE 1471 terms and definitions could be largely assimilated by TOGAF without serious impact.
- TOGAF should benefit from the introduction of the notion of viewpoint as a means to state rules on the construction of views. This is potentially applicable across the Architecture Continuum, and has a great potential for reuse.
- IEEE 1471 and its users have found much value in making explicit the notions of stakeholder and concern. TOGAF may want to consider introducing these constructs.
- IEEE 1471 defines conformance with respect to architectural descriptions. Application of TOGAF uses and produces a variety of architectural descriptions from the Foundation Architecture, to individual organization-specific architectures. Having a single set of "content requirements" for these work products would increase understandability and analyzability.

References

[1] David E. Emery, Rich Hilliard, and Timothy B. Rice. Experiences applying a practical architectural method. In Alfred Strohmeier, editor, *Reliable Software Technologies–Ada-Europe '96*, number 1088 in Lecture Notes in Computer Science. Springer, 1996.

[2] Rich Hilliard. Using the UML for architectural description. In Robert France and Bernhard Rumpe, editors, *≪UML≫'99 The Unified Modeling Language, Second International Conference*, volume 1723 of *Lecture Notes in Computer Science*, pages 32–48. Springer, 1999.

[3] Rich Hilliard. Views and viewpoints in software systems architecture. Position paper from the *First Working IFIP Conference on Software Architecture*, San Antonio, 1999.

[4] IEEE Architecture Working Group. *IEEE P1471/D5.2 Draft Recommended Practice for Architectural Description*, December 1999.

[5] Philippe B. Kruchten. The 4+1 view model of architecture. *IEEE Software*, 28(11):42–50, November 1995.

[6] M. A. Ogush, D. Coleman, and D. Beringer. A template for documenting software and firmware architectures. Draft version 1.3, January 2000.

[7] The Open Group. The Open Group Architectural Framework (TOGAF) version 5. http://www.opengroup.org/public/arch/, 1999.

A Fragment of the XML Schema for MLAD

```
<XML:SCHEMA ID="ArchitecturalDescription">
    <ELEMENTTYPE ID="Name">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Obligation">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Freedom">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Orientation">
        <GROUP GROUPTYPE="SEQ">
            <ELT HREF="#Purpose"/>
            <ELT HREF="#Scope"/>
            <ELT HREF="#Context"/>
        </GROUP>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Purpose">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Goal">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="View">
        <GROUP GROUPTYPE="SEQ">
            <ELT HREF="#ViewName"/>
            <ELT HREF="#ViewpointName"/>
            <ELT HREF="#Assumptions"/>
            <ELT HREF="#KeyDecisions"/>
            <ELT HREF="#ViewModel" OCCURS="STAR"/>
            <ELT HREF="#Consequences"/>
        </GROUP>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="NeedsAnalysis">
        <GROUP GROUPTYPE="SEQ">
```

```
            <ELT HREF="#Stakeholder" OCCURS="STAR"/>
            <ELT HREF="#Concern" OCCURS="STAR"/>
            <ELT HREF="#Vision" OCCURS="OPTIONAL"/>
            <ELT HREF="#Goal" OCCURS="STAR"/>
            <ELT HREF="#Need" OCCURS="STAR"/>
        </GROUP>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Scope">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Defines">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Stakeholder">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="ViewpointDeclaration">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Commitment">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="ViewModel">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Context">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Concern">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="ViewpointName">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Consequences">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="ViewName">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="KeyDecisions">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Vision">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="ArchitecturalDescription">
        <GROUP GROUPTYPE="SEQ">
            <ELT HREF="#Name"/>
            <ELT HREF="#Orientation"/>
            <ELT HREF="#NeedsAnalysis" OCCURS="OPTIONAL"/>
            <ELT HREF="#ViewpointDeclaration" OCCURS="STAR"/>
            <ELT HREF="#View" OCCURS="STAR"/>
        </GROUP>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Need">
        <PCDATA/>
    </ELEMENTTYPE>
    <ELEMENTTYPE ID="Assumptions">
        <PCDATA/>
    </ELEMENTTYPE>
</XML:SCHEMA>
```