W3C Working Draft

Web Services Architecture Requirements

W3C Working Draft 29 April 2002

This version:
http://www.w3.org/TR/2002/WD-wsa-reqs-20020429
Latest version:
http://www.w3.org/TR/wsa-reqs
Editors:
Daniel Austin, W. W. Grainger, Inc. <austin.d@ic.grainger.com>
Abbie Barbir, Nortel Networks, Inc. <abbieb@nortelnetworks.com>
Sharad Garg, The Intel Corporation <sharad.garg@intel.com>

Abstract

The use of Web Services on the World Wide Web is expanding rapidly as the need for application-to-application communication and interoperability grows. These services provide a standard means of communication among different software applications involved in presenting dynamic context-driven information to the user. In order to promote interoperability and extensibility among these applications, as well as to allow them to be combined in order to perform more complex operations, a standard reference architecture is needed. The Web Services Architecture Working Group at W3C is tasked with producing this reference architecture.

This document describes a set of requirements for a standard reference architecture for Web Services developed by the Web Services Architecture Working Group. These requirements are intended to guide the development of the reference architecture and provide a set of measurable constraints on Web Services implementations by which conformance can be determined.

Status of this Document

This section describes the status of this document at the time of its publication. Other documents may supersede this document. The latest status of this document series is maintained at the W3C.

This is the first W3C Working Draft of the Web Services Architecture Requirements document. It is a chartered deliverable of the Web Services Architecture Working Group, which is part of the Web Services Activity. Although the Working Group agreed to request publication of this document, this document does not represent consensus within the Working Group about Web services architecture requirements.

This first version of the requirements document is an early snapshot: it may contain conflicting and incomplete requirements and goals. The next version that the Working Group will publish will be more complete and polished.

Comments on this document should be sent to www-wsa-comments@w3.org (public archive). It is inappropriate to send discussion emails to this address.

Discussion of this document takes place on the public www-ws-arch@w3.org mailing list (public archive) per the email communication rules in the Web Services Architecture Working Group charter.

Patent disclosures relevant to this specification may be found on the Working Group's patent disclosure page.

This is a public W3C Working Draft for review by W3C members and other interested parties. It is a draft document and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use W3C Working Drafts as reference material or to cite them as other than "work in progress". A list of all W3C technical reports can be found at http://www.w3.org/TR/.

Table of Contents

1 Introduction

The use of Web Services on the World Wide Web is expanding rapidly as the need for application-to-application communication and interoperability grows. These services provide a standard means of communication among different software applications involved in presenting dynamic context-driven information to the user. In order to promote interoperability and extensibility among these applications, as well as to allow them to be combined in order to perform more complex operations, a standard reference architecture is needed. The Web Services Architecture Working Group at W3C is tasked with producing this reference architecture.

This document describes a set of requirements for a standard reference architecture for Web Services developed by the Web Services Architecture Working Group. These requirements are intended to guide the development of the reference architecture and provide a set of measurable constraints on Web Services implementations by which conformance can be determined.

1.1 What is a Web service?

The Working Group has jointly come to agreement on the following working definition:

Web service

[Definition: A Web service is a software application identified by a URI, whose interfaces and binding are capable of being defined, described and discovered by XML artifacts and supports direct interactions with other software applications using XML based messages via internet-based protocols]

1.2 Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in RFC 2119.

> **Note:**
>
> A few words on the naming convention used here and throughout this document: all goals, critical success factors and requirements are labeled according to the following convention:
>
> [D-]A(G|F|R|UC)nnn.n.n
>
> [D-] indicates that the item is in a draft state
>
> A indicates that this is an architectural item.
>
> [G|F|R|UC] is one of Goal|Critical Success Factor|Requirement|Use Case.
>
> nnn.n.n indicates the sequence number of the item.

2 Requirements Analysis Method

Many methods of analyzing requirements for software systems are available. While each of them has strengths and weaknesses, the Web Services Architecture Working Group has decided to make use of two methods concurrently, in the hope that together each of these methods will produce a well-defined set of requirements for Web Services Architecture. The two methods chosen are the Critical Success Factor Analysis method, which will be supplemented through the use of gathering Usage Scenarios. Both of these methods are useful but represent different approaches to the problem of gathering requirements.

The Working Groups intends to use these methods together and to cross-reference the results of each approach to ensure consistency of the overall architectural direction. By ensuring that the requirements each serve to meet the goals of the Working Group through the CSF analysis, and also ensuring that the architecture is consistent with the envisioned Usage Scenarios of the Working Groups in the Web Services activity, we can develop a set of architectural requirements that will provide an architectural model that meets the needs of all of those involved.

Note that in the case of Usage Scenarios, the vast majority of these are taken from the work of other W3C Working Groups in the Web Services Activity domain. Few individual Usage Scenarios will be developed by the Web Services Architecture Working Group directly, and those only in response to perceived gaps or omissions in the work of other Working Groups. Usage scenarios will be published separately.

2.1 Understanding Critical Success Factors Analysis

The Critical Success Factors Analysis methodology for determining requirements is a top-down means of determining requirements based on the needs of the organization. For this reason it is well-suited for requirements analysis for large systems with many stakeholders and an audience with multiple and sometimes conflicting interests. The CSF analysis method begins with a mission statement and then begins to divide the mission statement into a set of very high-level goals. These high-level goals are then further divided into Critical Success Factors, which themselves are then further broken down into multiple levels of a hierarchy, becoming more concrete. At the lowest level, each CSF becomes a requirement for the system; a single, well-defined task that must be accomplished in order to be successful. Along the way, problems to be solved and assumptions made are recorded.

Once the CSF hierarchy is established and a set of requirements has been derived, these can then be arranged into a matrix for comparison with the problems identified. In order to be considered complete, each problem must be fully addressed by one or more requirements.

By analyzing the steps necessary to achieve success, and cross-referencing them against problems to be solved, a complete set of requirements can be determined that can then be correlated with specific user scenarios. Each of the requirements should apply to at least one user scenario, and generally more than one.

This methodology allows requirements to be determined that satisfy the needs of the

organization and those of the user. Since architectural frameworks are built and maintained by organizations, this method allows us to create a well-defined and reasonably complete set of requirements.

3 The Analysis Hierarchy

3.1 Mission Statement

3.1.1 Mission

The mission of the Web Services Architecture Working Group is to develop and maintain a standard reference architecture for Web Services.

3.1.2 Users of Web Services Architecture

The W3C Web Services Reference Architecture is intended primarily for the W3C Web Services Architecture Working Group to analyze, prioritize and characterize the technologies that are needed to fully realize an interoperable and extensible realization of the promise of Web Services. It is also intended for the use of other working groups specifying the technologies identified and described in the architecture. A secondary target audience for the architecture are the developers implementing the specified technologies, and the wider IT community that uses these technologies to deploy Web Services.

3.2 Goals

3.2.1 Top-level Goals

The Working Group has determined that at the highest level, its goals can be divided into 6 categories. Each of these is associated with the CSFs and requirements listed in section 3.2.2

Top-level Goals for the Web Services Architecture

- *D-AG001 Interoperability*
  The Web Services Architecture should provide a reference platform for the development of interoperable Web Services across a wide array of environments.
  Critical Success Factors for this goal:
  - D-AC001
  - D-AC004
  - D-AC016

- *D-AG002 Reliability*
  The Web Services Architecture must be reliable and stable over time.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS
Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS
Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS
Sync your system to PACER to automate legal marketing.