Object Management Group

140 Kendrick Street
Building A Suite 300
Needham, MA 02494
USA

Telephone: +1-781-444-0404
Facsimile: +1-781-444-0320

Request For Proposal

Web-Enabled DDS

OMG Document: mars/2009-09-19

Letters of Intent due: December 15, 2009
Submissions due: June 15, 2010

Objective of this RFP

The *Data Distribution Service for Real-Time Systems* (DDS) is the Object Management Group (OMG) standard for data-centric publish subscribe. The objective of this RFP is to facilitate the use of DDS from web-client applications.

The OMG DDS standard has experienced a record-breaking adoption within the Aerospace and Defense domains, and is swiftly expanding into new domains such as Transportation, Financial Services, and SCADA. To sustain and further propel its adoption, it is essential to expand the kinds of applications and technologies that can easily access DDS data and benefit from DDS's performance and quality of service.

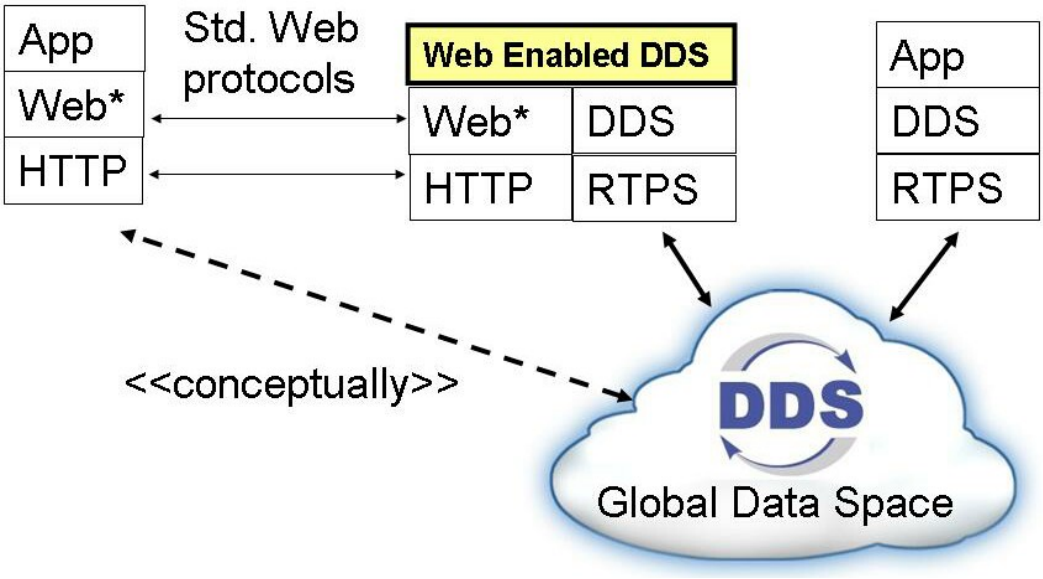
An important class of applications that would greatly benefit from a simple, standards-based access to DDS are web-based applications. Both client-side web applications, such as those programmed using technologies like JavaScript or AJAX, as well as server-side web applications programmed using scripting languages like PHP, Perl, Ruby, and Python.

The OMG DDS standard defines local interfaces that can be used by an application to access a virtual “Global Data Space” where applications can publish and subscribe data. These interfaces are realized in concrete programming languages such as C/C++, Java, .NET, or ADA from their IDL description. An application using the DDS API language bindings will need to load a programming-language specific library that implements these local interfaces and uses the DDS Interoperability Wire Protocol to communicate with the “Global Data Space” implemented collaboratively with other DDS applications on the network.

The need to load a programming-language specific implementation library limits the potential use of DDS to applications written in one of the supported programming languages. For example if there are no DDS implementations that support scripting languages like Perl, PHP, Ruby, or Python, then a program written in those languages cannot directly use DDS.

Furthermore, there are deployment scenarios where it is impossible or impractical to load a library in the client computer. All the client can do is access a web page and perhaps load into the browser code that it will execute within the environment of the browser (e.g. JavaScript or Flash). These scenarios are currently precluded from being able to consume or produce data in the DDS Global Data Space.

To support these use cases, this RFP solicits proposals for exposing DDS via a variety of web-friendly protocols such that client applications do not need to load any libraries in order to produce and consume data in the DDS Global Data Space. Furthermore, web applications, including applications that execute inside a web browser, will be able to use DDS to Publish and Subscribe information and thereby benefit from the Performance, Scalability, and Quality of Service (QoS) available in the DDS Implementations. The intended result is illustrated in the figure below.



For further details see Chapter 6 of this document.

1.0 Introduction

1.1 Goals of OMG

The Object Management Group (OMG) is the world's largest software consortium with an international membership of vendors, developers, and end users. Established in 1989, its mission is to help computer users solve enterprise integration problems by supplying open, vendor-neutral portability, interoperability and reusability specifications based on Model Driven Architecture (MDA). MDA defines an approach to IT system specification that separates the specification of system functionality from the specification of the implementation of that functionality on a specific technology platform, and provides a set of guidelines for structuring specifications expressed as models. OMG has established numerous widely used standards such as OMG IDL[IDL], CORBA[CORBA], Realtime CORBA [CORBA], GIOP/IIOP[CORBA], UML[UML], MOF[MOF], XMI[XMI] and CWM[CWM] to name a few significant ones.

1.2 Organization of this document

The remainder of this document is organized as follows:

Chapter 2 - *Architectural Context* - background information on OMG's Model Driven Architecture.

Chapter 3 - *Adoption Process* - background information on the OMG specification adoption process.

Chapter 4 - *Instructions for Submitters* - explanation of how to make a submission to this RFP.

Chapter 5 - *General Requirements on Proposals* - requirements and evaluation criteria that apply to all proposals submitted to OMG.

Chapter 6 - *Specific Requirements on Proposals* - problem statement, scope of proposals sought, requirements and optional features, issues to be discussed, evaluation criteria, and timetable that apply specifically to this RFP.

< Note to RFP Editors: Additional RFP-specific chapters may also be included following Chapter 6. If additional chapters are included, please insert brief description of each such chapter here. Insert the additional chapters immediately following Chapter 6, and preceding Appendix A. >

Appendix A – *References and Glossary Specific to this RFP*

< Note to RFP Editors: Please insert any references that are specific to this RFP in section A.1 as per the instructions that appear in that section.

Note to RFP Editors: Please insert any glossary items that are specific to this RFP in section A.2 as per the instructions that appear in that section. >

Appendix B – General References and Glossary

< Note to RFP Editors: Additional RFP-specific appendices may also be included following Appendix B. If additional appendices are included, please insert brief description of each such appendix here. Insert the additional appendices immediately following Appendix B. >

1.3 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 [RFC2119].

1.4 Contact Information

Questions related to the OMG's technology adoption process may be directed to *omg-process@omg.org*. General questions about this RFP may be sent to *responses@omg.org*.

OMG documents (and information about the OMG in general) can be obtained from the OMG's web site *(http://www.omg.org/)*. OMG documents may also be obtained by contacting OMG at *documents@omg.org*. Templates for RFPs (like this document) and other standard OMG documents can be found at the OMG Template Downloads Page at *http://www.omg.org/technology/template_download.htm*

2.0 Architectural Context

<RFP writers shall not change this section>

3.0 Adoption Process

<RFP writers shall not change this section>

4.0 Instructions for Submitters

<RFP writers shall not change this section>

5.0 General Requirements on Proposals

<RFP writers shall not change this section>

6.0 Specific Requirements on Proposals

6.1 Problem Statement

The OMG DDS specification defines an API that applications can use to publish and subscribe data in a "Global Data Space". The API defined by the DDS specification must be implemented by means of "local interfaces" on each of the supported programming languages. This approach limits the use of DDS to applications that (a) use one of the programming-languages supported by the DDS implementations, and (b) can link into the application the vendor-provided DDS-implementation libraries. In practice these limitations mean that there are no standard APIs to access DDS from many commonly-used scripting languages, such as PHP, Perl, Ruby, or Python. Similarly there are currently no standard APIs or mechanisms to access the DDS Global Data Space from application running inside a web browser (e.g. JavaScript applications).

Another important usage scenario is that of disconnected or stateless clients. These are typically implemented as single-command, short-lived processes, for example shell commands or web-server CGI scripts. Under this usage scenario, a user starts a client application to execute a very simple action, such as publish data on a Topic, or receive the latest data on a Topic. The client executes the action, returns the output (typically to the *stdout*) and then exits. This is a common scenario when integrating with web-server applications which use CGI scripts to execute individual actions.

Disconnected or stateless clients are problematic because of the dynamic, one-to-many nature of publish-subscribe applications and the fact that DDS does not require the presence of a broker or centralized server. In order to publish or subscribe to data, a DDS application must join a Domain, discover other participants and other publishers and subscribers, exchange the information, and then remain present long enough for the reliability protocol to ensure that all subscribers receive the information. This makes the implementation of the "short-lived process" challenging. How long should the process wait to discover all subscribers or publishers? The approach is also inefficient. Each time the process starts, it creates new DDS entities that must then be discovered by the rest of the system, only to be destroyed shortly afterwards.

With the increasing adoption of DDS for the integration of large distributed systems, it is desirable to define ***standard ways*** whereby web-based applications can: access DDS; publish and subscribe data into the DDS Global Data Space; and benefit from the performance, scalability, and quality of service offered by DDS implementations. In addition, it is desirable for this approach to support efficient access to the Global Data Space by disconnected or stateless clients. Note that all this is possible today, but the approaches are non-standard.

6.2 Scope of Proposals Sought

This RFP solicits submissions for (1) a Platform Independent Model (PIM) of how web-clients should access a DDS System, and (2) a set of mappings to specific "web" Platforms that realize the PIM in terms of standard web technologies and protocols.

6.2.1 Need for a Platform Independent Model

The PIM is necessary to define the "Web-Enabled DDS Service Object Model" that is exposed to the DDS clients over the web. The reason that this RFP solicits a new Object Model rather than simply reusing the standard "DDS Object Model" is threefold:

- The DDS Object Model is intended for use with a local programming API. For this reason, the standard DDS Object Model contains many objects and methods with strongly-typed parameters, as well as, a direct callback interface by means of listener objects that the application registers with the middleware. This would not be suitable for web clients that normally expect a simplified interface with no callbacks and where all parameters are encoded in text.
- Web client connectivity is inherently intermittent. By the very nature of the HTTP protocol, clients are continually being connected and disconnected from the server. Therefore the Web-Enabled DDS Service Object Model must overcome this by introducing a "session" whose life can span beyond the physical connection.
- Web clients can access Web-Enabled DDS from any location and therefore it is desirable to have an access control model that authenticates each user/principal, controls who can access the DDS Global Data Space, and controls what operations (e.g. which DDS Topics it can read and write) each user can perform.

6.2.2 Need for Platform Specific Mappings

Web clients accessing data and services over the web normally use a mix of standard approaches, technologies, and protocols including: RESTful, Web-Services (WSDL/SOAP), RSS, ATOM, and XMPP. Each of these approaches presents advantages and disadvantages; selecting which to use is often driven by business requirements.

For example, RESTful is the most universally deployed and used for most of the "cloud" services such as those offered by Amazon and Google. It is also simple and friendly to web browsers, which use bookmarks and links to get to the data directly. However, it lacks a formal language with which to define interfaces. Despite being less widely deployed, Web-Services have a language (WSDL) that can be used to formally define interfaces and is supported by the major providers

of Enterprise Service Bus (ESB) infrastructure. However, it is less friendly to web browsers and cannot be easily called from JavaScript. RSS and Atom are popular protocols for retrieving data. RSS is more established but, unlike Atom, the RSS standard only defines how to receive existing data, not how to post new data. XMPP is a simple and popular protocol based on HTTP and XML originally developed for internet chat applications but is now becoming popular as a general protocol for peer-to-peer application communications.

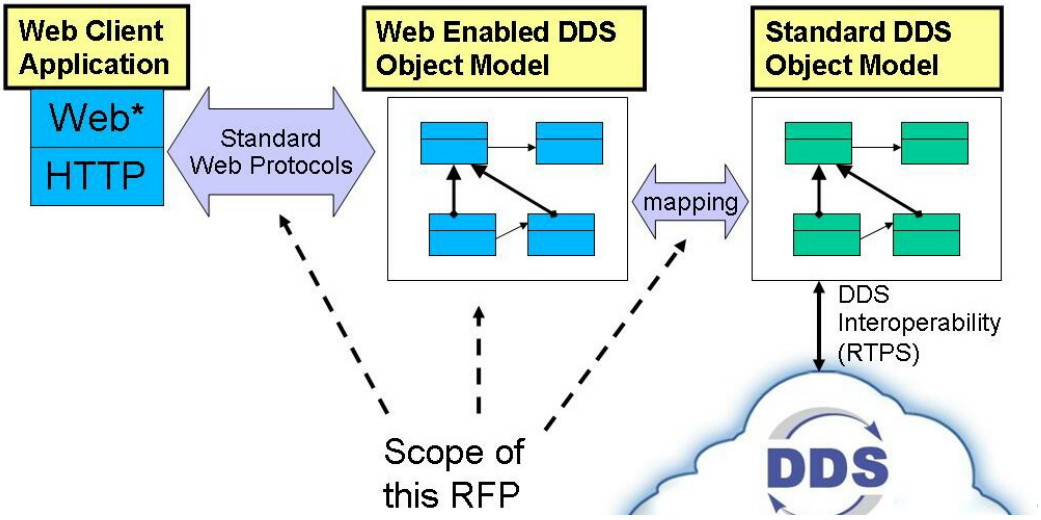
Given the existence of multiple popular web technologies and protocols (referred in this RFP as "web-platforms"), each with their own strengths and limitations, it is imperative that the requested Web-Enabled DDS specification not be tied to or associated with a single web platform. Therefore, this RFP asks submitters to map the PIM into one or more of these web platforms.. The intent of this approach is for all platform mappings to be equivalent and interoperable.

6.2.3 Goals of this specification

This specification must define an object model for Web-Enabled DDS using UML. The specification must define how this object model relates to the DDS Object Model as defined in the OMG Data-Distribution Service Specification (we shall refer to this object model as the "Standard DDS Object Model"). In other words it must define the effects that each operation has, if any, on the related DDS Entities.

The Web-Enabled DDS Object Model must be both an extension and a simplification of the Standard DDS Object Model. It must extend the Standard DDS Object Model in order to include an *access control model* and a *session model* to support disconnected clients. It must also be a simplification of the Standard DDS Object Model in order to reduce the number of objects and operations and make it more suitable for web clients.

This Specification must also provide a collection of mappings describing how the Web-Enabled DDS Object Model can be accessed and manipulated using common Web technologies, such as, RESTful, Web-Services (WSDL/SOAP), ,



RSS, ATOM, and XMPP.

6.3 Relationship to other OMG Specifications and activities

6.3.1 Relationship to OMG specifications

Main relevant specifications:

- The DDS specification (formal/2005-12-04)
- DDS for Light-Weight CCM specification Beta 1 (ptc/09-02-02)

6.3.2 Relationship to other OMG Documents and work in progress

Main relevant specifications:

- The Extensible and Dynamic Topic Types for DDS Specification. RFP (mars/08-06-22).

6.4 Related non-OMG Activities, Documents and Standards

< Note to RFP Editors: List documents, URLs, standards, etc. that are relevant to the problem and the proposals being sought. Also describe any known overlaps with specification activities or specifications, competing or complementary, from other standards bodies. >

6.5 Mandatory Requirements

- Proposals shall provide a PIM that defines a Web-Enabled DDS Object Model suitable for implementation as a stand-alone service or gateway.
    - The PIM shall allow clients to control which DDS Domains to join and which DDS Topics to publish or subscribe.
    - The PIM shall allow clients to specify the data-type associated with the published DDS Topics
    - The PIM shall allow clients to either specify or discover the data-type of the subscribed DDS Topics
    - The PIM shall allow clients to read data on subscribed Topics
    - The PIM shall allow clients to write data on published Topics

  - The PIM shall allow clients to control the QoS of the DDS Entities used to subscribe to a DDS Topic. Optionally, the control could be limited to just the selection of a QoS profile as defined by the DDS for Light-Weight CCM specification.

  - The PIM shall allow clients to control the QoS of the DDS Entities used to publish a DDS Topic. Optionally, the control could be limited to just the selection of a QoS profile as defined by the DDS for Light-Weight CCM specification.

  - The PIM shall allow clients to specify Content-Based Subscriptions to Topics

  - The PIM shall expose a means for a client to discover which Topics are available on the Domain

  - The PIM shall include a session model allowing a client to survive transport disconnection.

  - The PIM shall include an access control model that limits which operations can be performed by a client based on the Domain and Topic it is trying to access.

- Proposals shall define the RESTful platform to which they map and provide a mapping that defines how to access a service implementing the Web-Enabled DDS Object Model from a RESTful client.

- Proposals shall provide a platform mapping that maps the Web-Enabled DDS Object into WSDL such that the service can be accessed from a WS-I Basic Profile 1.1 compliant client.

- The platform mapping that a client uses shall be transparent to other clients. Clients shall be able interoperate with other clients regardless of which platform they use.

- Web clients shall interoperate with regular DDS clients independently of the platform used by the web client and the language binding used by the regular DDS client.

- Web clients shall be able to publish and subscribe to data of all DDS types, just like a regular DDS application can.

- Multiple instantiations of Web-Enabled DDS Object Model shall be allowed to co-exist on the same network and communicate with each other.

6.6 Optional Requirements

- Proposals may provide a platform mapping that defines how an RSS client can read data from a service implementing the Web-Enabled DDS Object Model

- Proposals may provide a platform mapping that defines how an XMPP client can access a service implementing the Web-Enabled DDS Object Model

- Proposals may provide a platform mapping that defines how an Atom client can access a service implementing the Web-Enabled DDS Object Model

- Proposal may provide a platform mapping that defines how other web-based clients can access a service implementing the Web-Enabled DDS Object Model.

6.7 Issues to be discussed

- Proposals should discuss the style choices available for each web platform selected for mappings and the reasons behind the selected choices.

- Proposals should discuss the selection of security and access control models and how it relates to commonly used web-service standards

- Proposals should discuss the performance impact of using Web Enabled DDS and any specific performance issues that a particular platform mapping may introduce

- Proposals should discuss the scalability impact of using Web Enabled DDS and any specific scalability issues that a particular platform mapping may introduce

6.8 Evaluation Criteria

- Submissions that leverage standards such as XSD and XML for schema definition and data representation will be favored over other less standard approaches

- Submissions that leverage security standards such as PKI, SAML, XACML, HTTPS, etc. will be favored over less standard approaches

- Submissions that result in higher performance or more scalable systems will be favored.

6.9 Other information unique to this RFP

None

6.10 RFP Timetable

The timetable for this RFP is given below. Note that the TF or its parent TC may, in certain circumstances, extend deadlines while the RFP is running, or may elect to have more than one Revised Submission step. The latest timetable can always be found at the OMG *Work In Progress* page at http://www.omg.org/schedules under the item identified by the name of this RFP. Note that "<month>" and "<approximate month>" is the name of the month spelled out; e.g., January.

Event or Activity	Actual Date
Preparation of RFP by TF	
RFP placed on OMG document server	*August 20th, 2009* *"Four week rule"*
Approval of RFP by Architecture Board *Review by TC*	*September 2009*
TC votes to issue RFP	*September 2009*
LOI to submit to RFP due	*February 15, 2010*
Initial Submissions due and placed on OMG document server ("Four week rule")	*March 1st, 2010*
Voter registration closes	*March 7th, 2010*
Initial Submission presentations	*March 2010*
Preliminary evaluation by TF	*March 2010*
Revised Submissions due and placed on OMG document server ("Four week rule")	*May 31st,2010* *"Four week rule"*
Revised Submission presentations	*June,2010*
Final evaluation and selection by TF *Recommendation to AB and TC*	*June 2010*
Approval by Architecture Board *Review by TC*	*June 2010*
TC votes to recommend specification	*June 2010*
BoD votes to adopt specification	*September 2010*

< Note to RFP Editors: Insert additional chapter if needed here and update the list and brief description of chapters in Chapter 1. >

Appendix A References and Glossary Specific to this RFP

A.1 References Specific to this RFP

The following documents are referenced in this document:

[ATOM] Atom Syndication Format. IETF RFC 4287. http://www.ietf.org/rfc/rfc4287.txt; Atom Publishing Protocol. IETF RFC 5023. http://www.ietf.org/rfc/rfc5023.txt

[HTTP] Hypertext Transfer Protocol. version 1.1. IETF RFC 2616. http://www.ietf.org/rfc/rfc2616.txt

[PKI] The Public Key Infrastructure (PKI) refers to a collection of technology that supports the creation, management, storing, distribution, and revocation of digital certificates. There are many standards such as X.509, PGP, etc. under this umbrella.

[REST] Representational State Transfer (REST). Roy Thomas Fielding's PhD dissertation "Architectural Styles and the Design of Network-based Software Architectures". http://www.ics.uci.edu/~fielding/pubs/dissertation/top.htm

[RSS] RSS 2.0 Specification. http://www.rssboard.org/rss-specification.

[SAML] Security Assertion Markup Language (SAML) is an XML-based standard for exchanging authentication and authorization data between security domains, that is, between an identity provider (a producer of assertions) and a service provider (a consumer of assertions). SAML is a product of the OASIS Security Services Technical Committee. For the purposes of this RFP SAML refers to the OASIS SAML V2.0 specification http://saml.xml.org/saml-specifications#samlv20

[WSDL] The Web Services Description Language (WSDL) 1.1. http://www.w3.org/TR/wsdl

[WSI] WS-I Basic Profile 1.1 specification. http://www.ws-i.org/profiles/basicprofile-1.1.html

[XACML] XACML stands for eXtensible Access Control Markup Language. It is a declarative access control policy language implemented in

XML and a processing model, describing how to interpret the policies. For the purposes of this RFP XACML refers to the OASIS XACML 2.0 Specification Set http://docs.oasis-open.org/xacml/2.0/XACML-2.0-OS-NORMATIVE.zip

[XML] Extensible Markup Language (XML) 1.1 (Second Edition). W3C recommendation, August 2006.

[XMPP] Extensible Messaging and Presence Protocol (XMPP): IETF RFC 3920. http://www.ietf.org/rfc/rfc3920.txt. XMPP extensions for basic instant messaging and presence: IETF RFC 3921. http://www.ietf.org/rfc/rfc3921.txt. See also http://xmpp.or2.0 Specification. http://www.rssboard.org/rss-specification.

A.2 Glossary Specific to this RFP

< Note to RFP Editors: Insert any glossary items specific to this RFP that are used in Section 6 and any additional sections in the same format as in Section B.2 and in alphabetical order in this section. >

ATOM platform – For the purposes of this RFP the name ATOM refers to to a pair of related standards: (1) the Atom Syndication Format which is an XML language used for web feeds, published by the IETF as RFC 4287 and (2) the Atom Publishing Protocol which is a simple HTTP-based protocol for creating and updating web resources published by the IETF as RFC 5023.

RESTful platform - As described in a dissertation by Roy Fielding, REST is an "architectural style" that basically exploits the existing technology and protocols of the Web, including HTTP (Hypertext Transfer Protocol) and XML. REST is simpler to use than the well-known SOAP (Simple Object Access Protocol) approach, which requires writing or using a provided server program (to serve data) and a client program (to request data). SOAP, however, offers potentially more capability. For example, a syndicator that wanted to include up-to-date stock prices to subscribing Web sites might need to use SOAP, which allows a greater amount of program interaction between client and server.

RSS platform – RSS stands for Really Simple Syndication. Wikipedia describes RSS as is a family of web feed formats used to publish frequently updated works—such as blog entries, news headlines, audio, and video—in a standardized format. An RSS document (which is called a "feed", "web feed", or "channel") includes full or summarized text, plus metadata such as publishing dates and authorship. Web feeds benefit publishers by letting them syndicate content automatically. They benefit readers who want to subscribe to timely updates from favored websites or to aggregate feeds from many sites into one place. RSS feeds can be read using software called an "RSS reader", "feed

reader", or "aggregator", which can be web-based, desktop-based, or mobile-device-based. A standardized XML file format allows the information to be published once and viewed by many different programs. For the purposes of this RFP the RSS platform refers to the technology described by the RSS 2.0 specification published by the RSS Advisory Board on March 30, 2009.

Web client – Generic term used to refer to an application that is accessing the web-enabled DDS service over standard web protocols, including but not limited to plain HTTP, SOAP over HTTP, RSS over HTTP, etc.

Web-enabled – Generic term used to indicate that a particular technology is accessible by Web Clients by means of standard web protocols, including but not limited to plain HTTP, SOAP over HTTP, RSS over HTTP, etc.

WSDL platform - As described by the W3C, WSDL is an XML format for describing network services as a set of endpoints operating on messages containing either document-oriented or procedure-oriented information. The operations and messages are described abstractly, and then bound to a concrete network protocol and message format to define an endpoint. Related concrete endpoints are combined into abstract endpoints (services). WSDL is extensible to allow description of endpoints and their messages regardless of what message formats or network protocols are used to communicate, however, the only bindings standardized by the W3C describe how to use WSDL in conjunction with SOAP 1.1, HTTP GET/POST, and MIME. For the purposes of this RFP the term “WDSL platform” shall refer to the set of standard specifications defined by the WS-I Basic Profile 1.1 specification.

XMPP platform - The Extensible Messaging and Presence Protocol (XMPP) is an open technology for real-time communication, which powers a wide range of applications including instant messaging, presence, multi-party chat, voice and video calls, collaboration, lightweight middleware, content syndication, and generalized routing of XML data. The base specifications of the Extensible Messaging and Presence Protocol (XMPP) formalize the core protocols developed within the Jabber open-source community in 1999. They are published as IETF RFCs 3920 and 3921.

Appendix B General Reference and Glossary

< Note to RFP Editors: Append additional appendices if needed here and update the list and brief description of appendices in Chapter 1. >

Revision History

Date	Version	Changes
July 29, 2009	v1	Initial version from meeting between Nick Stavros and Gerardo Pardo in San Jose, CA.
August 3, 2009	v2	Draft version from Nick Stavros and Gerardo Pardo sent to the OMG mailing list
August 5, 2009	v3	Incorporated changes from Char Wales document: webenabled_dds_rfp_v2 (wales).doc) and Virginie Watine document: webenabled_dds_rfp_v2 (wales + vw).doc
August 6, 2009	v3	Incorporated additional changes from Char Wales document: webenabled_dds_rfp_v3 (wales).doc)
August 24, 2009	v4	Minor clarifications and edits. Cleaned up language of mandatory versus optional requirements regarding use of 'shall' versus 'may'
September 14, 2009	v5	Addressed remaining AB comments received over email. Completed RFP Timetable in section 6.10. Incorporated the comments made during the mars plenary meeting 9/14/2009
September 16, 2009	v6	Addressed comments from the AB meeting on 9/15/09. Removed the use of the word "PSM" and instead refer to this as a "platform mapping". Added clarifications regarding what is meant by the term "RESTful platform". Moved references to the proper section, and expanded Glossary and References section.

September 17, 2009	**v7**	**Modifications from AB meeting on 9/17/09 affecting two of the mandatory requirements.**