

# [MS-OSALER]: Alerts Interoperability Protocol Specification

---

## Intellectual Property Rights Notice for Open Specifications Documentation

- **Technical Documentation.** Microsoft publishes Open Specifications documentation for protocols, file formats, languages, standards as well as overviews of the interaction among each of these technologies.
- **Copyrights.** This documentation is covered by Microsoft copyrights. Regardless of any other terms that are contained in the terms of use for the Microsoft website that hosts this documentation, you may make copies of it in order to develop implementations of the technologies described in the Open Specifications and may distribute portions of it in your implementations using these technologies or your documentation as necessary to properly document the implementation. You may also distribute in your implementation, with or without modification, any schema, IDL's, or code samples that are included in the documentation. This permission also applies to any documents that are referenced in the Open Specifications.
- **No Trade Secrets.** Microsoft does not claim any trade secret rights in this documentation.
- **Patents.** Microsoft has patents that may cover your implementations of the technologies described in the Open Specifications. Neither this notice nor Microsoft's delivery of the documentation grants any licenses under those or any other Microsoft patents. However, a given Open Specification may be covered by Microsoft [Open Specification Promise](#) or the [Community Promise](#). If you would prefer a written license, or if the technologies described in the Open Specifications are not covered by the Open Specifications Promise or Community Promise, as applicable, patent licenses are available by contacting [iplg@microsoft.com](mailto:iplg@microsoft.com).
- **Trademarks.** The names of companies and products contained in this documentation may be covered by trademarks or similar intellectual property rights. This notice does not grant any licenses under those rights.
- **Fictitious Names.** The example companies, organizations, products, domain names, e-mail addresses, logos, people, places, and events depicted in this documentation are fictitious. No association with any real company, organization, product, domain name, email address, logo, person, place, or event is intended or should be inferred.

**Reservation of Rights.** All other rights are reserved, and this notice does not grant any rights other than specifically described above, whether by implication, estoppel, or otherwise.

**Tools.** The Open Specifications do not require the use of Microsoft programming tools or programming environments in order for you to develop an implementation. If you have access to Microsoft programming tools and environments you are free to take advantage of them. Certain Open Specifications are intended for use in conjunction with publicly available standard specifications and network programming art, and assumes that the reader either is familiar with the aforementioned material or has immediate access to it.

**Preliminary Documentation.** This Open Specification provides documentation for past and current releases and/or for the pre-release (beta) version of this technology. This Open Specification is final

documentation for past or current releases as specifically noted in the document, as applicable; it is preliminary documentation for the pre-release (beta) versions. Microsoft will release final documentation in connection with the commercial release of the updated or new version of this technology. As the documentation may change between this preliminary version and the final version of this technology, there are risks in relying on preliminary documentation. To the extent that you incur additional development obligations or any other costs as a result of relying on this preliminary documentation, you do so at your own risk.

## Revision Summary

Date	Revision History	Revision Class	Comments
04/04/2008	0.1		Initial Availability
06/27/2008	1.0	Major	Revised and edited the technical content
12/12/2008	1.01	Editorial	Revised and edited the technical content
03/18/2009	1.02	Editorial	Revised and edited the technical content
07/13/2009	1.03	Major	Changes made for template compliance
08/28/2009	1.04	Editorial	Revised and edited the technical content
11/06/2009	1.05	Editorial	Revised and edited the technical content
02/19/2010	2.0	Editorial	Revised and edited the technical content
03/31/2010	2.01	Editorial	Revised and edited the technical content
04/30/2010	2.02	Editorial	Revised and edited the technical content
06/07/2010	2.03	Editorial	Revised and edited the technical content
06/29/2010	2.04	Editorial	Changed language and formatting in the technical content.
07/23/2010	2.05	Minor	Clarified the meaning of the technical content.
09/27/2010	2.05	No change	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	2.05	No change	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	2.05	No change	No changes to the meaning, language, or formatting of the technical content.
03/18/2011	2.05	No change	No changes to the meaning, language, or formatting of the technical content.
06/10/2011	2.05	No change	No changes to the meaning, language, or formatting of the technical content.
01/20/2012	2.6	Minor	Clarified the meaning of the technical content.

Date	Revision History	Revision Class	Comments
04/11/2012	2.6	No change	No changes to the meaning, language, or formatting of the technical content.

Preliminary

# Table of Contents

<b>1 Introduction</b> .....	<b>6</b>
1.1 Glossary .....	6
1.2 References.....	6
1.2.1 Normative References.....	6
1.2.2 Informative References .....	7
1.3 Protocol Overview (Synopsis) .....	7
1.4 Relationship to Other Protocols.....	8
1.5 Prerequisites/Preconditions .....	8
1.6 Applicability Statement.....	8
1.7 Versioning and Capability Negotiation.....	8
1.8 Vendor-Extensible Fields.....	8
1.9 Standards Assignments .....	8
<b>2 Messages</b> .....	<b>9</b>
2.1 Transport.....	9
2.2 Message Syntax .....	9
2.2.1 Message-ID .....	9
2.2.2 X-AlertId .....	10
2.2.3 X-AlertTitle .....	10
2.2.4 X-AlertServerType.....	10
2.2.5 X-AlertWebUrl.....	10
2.2.6 X-AlertWebSoap.....	11
2.2.7 X-Sharing-Config-Url .....	11
2.2.8 X-Sharing-Remote-Uid .....	11
2.2.9 X-Sharing-WssBaseUrl .....	12
2.2.10 X-Sharing-ItemId .....	12
2.2.11 X-Sharing-Title.....	12
<b>3 Protocol Details</b> .....	<b>13</b>
3.1 Alerts Interoperability Server Details.....	13
3.1.1 Abstract Data Model .....	13
3.1.2 Timers .....	13
3.1.3 Initialization .....	13
3.1.4 Higher-Layer Triggered Events.....	14
3.1.5 Message Processing Events and Sequencing Rules.....	14
3.1.6 Timer Events .....	14
3.1.7 Other Local Events .....	14
3.2 Alerts Interoperability Client Details .....	14
3.2.1 Abstract Data Model .....	14
3.2.2 Timers .....	15
3.2.3 Initialization .....	15
3.2.4 Higher-Layer Triggered Events.....	15
3.2.5 Message Processing Events and Sequencing Rules.....	15
3.2.5.1 X-AlertId .....	15
3.2.5.2 X-AlertTitle .....	15
3.2.5.3 X-AlertServerType .....	15
3.2.5.4 X-AlertWebUrl .....	15
3.2.5.5 X-AlertWebSoap .....	15
3.2.5.6 X-Sharing-Config-Url.....	15
3.2.5.7 X-Sharing-Remote-Uid .....	16

3.2.5.8 X-Sharing-WssBaseUrl .....	16
3.2.5.9 X-Sharing-ItemId .....	16
3.2.5.10 X-Sharing-Title .....	16
3.2.6 Timer Events .....	16
3.2.7 Other Local Events .....	16
<b>4 Protocol Examples.....</b>	<b>17</b>
<b>5 Security.....</b>	<b>19</b>
5.1 Security Considerations for Implementers.....	19
5.2 Index of Security Parameters .....	19
<b>6 Appendix A: Product Behavior .....</b>	<b>20</b>
<b>7 Change Tracking.....</b>	<b>22</b>
<b>8 Index .....</b>	<b>23</b>

# 1 Introduction

This document specifies the Alerts Interoperability Protocol, which is used to identify and interpret Internet messages that can be sent to protocol clients when a document, Web page or other type of resource is changed on a protocol server. This protocol also specifies the syntax and semantics of user-defined fields in message headers of those messages.

Sections 1.8, 2, and 3 of this specification are normative and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in RFC 2119. Sections 1.5 and 1.9 are also normative but cannot contain those terms. All other sections and examples in this specification are informative.

## 1.1 Glossary

The following terms are defined in [\[MS-GLOS\]](#):

**ASCII**  
**Augmented Backus-Naur Form (ABNF)**

The following terms are defined in [\[MS-OFCGLOS\]](#):

**alert**  
**alert subscription**  
**Simple Mail Transfer Protocol (SMTP)**  
**Uniform Resource Locator (URL)**  
**workflow task**

The following terms are specific to this document:

**alert GUID:** A fixed GUID value in an Internet message header that identifies an Internet message as an alert.

**alert metadata:** The values that are contained in the X-headers of an alert (1).

**MAY, SHOULD, MUST, SHOULD NOT, MUST NOT:** These terms (in all caps) are used as described in [\[RFC2119\]](#). All statements of optional behavior use either MAY, SHOULD, or SHOULD NOT.

## 1.2 References

References to Microsoft Open Specifications documentation do not include a publishing year because links are to the latest version of the documents, which are updated frequently. References to other documents include a publishing year when one is available.

### 1.2.1 Normative References

We conduct frequent surveys of the normative references to assure their continued availability. If you have any issue with finding a normative reference, please contact [dochelp@microsoft.com](mailto:dochelp@microsoft.com). We will assist you in finding the relevant information. Please check the archive site, <http://msdn2.microsoft.com/en-us/library/E4BD6494-06AD-4aed-9823-445E921C9624>, as an additional source.

[MS-ALERTSS] Microsoft Corporation, "[Alerts Service Protocol Specification](#)".

[MS-OUTSPS] Microsoft Corporation, "[Lists Client Sync Protocol Specification](#)".

[MS-STSSYN] Microsoft Corporation, "[StsSync Data Structure](#)".

[RFC2047] Moore, K., "MIME (Multipurpose Internet Mail Extensions) Part Three: Message Header Extensions for Non-ASCII Text", RFC 2047, November 1996, <http://ietf.org/rfc/rfc2047.txt>

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997, <http://www.rfc-editor.org/rfc/rfc2119.txt>

[RFC2821] Klensin, J., "Simple Mail Transfer Protocol", STD 10, RFC 2821, April 2001, <http://www.ietf.org/rfc/rfc2821.txt>

[RFC2822] Resnick, P., Ed., "Internet Message Format", STD 11, RFC 2822, April 2001, <http://www.ietf.org/rfc/rfc2822.txt>

### 1.2.2 Informative References

[MS-GLOS] Microsoft Corporation, "[Windows Protocols Master Glossary](#)".

[MS-OFCGLOS] Microsoft Corporation, "[Microsoft Office Master Glossary](#)".

[RFC5234] Crocker, D., Ed., and Overell, P., "Augmented BNF for Syntax Specifications: ABNF", STD 68, RFC 5234, January 2008, <http://www.rfc-editor.org/rfc/rfc5234.txt>

[RFC822] Crocker, D.H., "Standard for ARPA Internet Text Messages", STD 11, RFC 822, August 1982, <http://www.ietf.org/rfc/rfc0822.txt>

### 1.3 Protocol Overview (Synopsis)

This protocol specifies how a protocol server can use X-headers of an Internet message to indicate to a protocol client that the message is an **alert (1)**. The protocol assumes the message conforms fully to [RFC2822](#). The protocol extends the **Message-ID** header (section [2.2.1](#)) and introduces ten X-headers to provide the following information about the alert (1):

- The **alert GUID** identifying that the message is an alert (1).
- The unique identifier for the **alert subscription**.
- The title of the alert (1).
- The protocol server software that sent the alert (1).
- The **URL** of the protocol server that sent the alert (1).
- The URL of the Web service associated with the originating protocol server to manage alerts (1).
- The URL to initiate synchronizing the protocol client with the container of the resource that is referred by the alert (1).
- The identifier and URL of the container of the resource referred by the alert (1).
- The unique identifier and title for the resource referred by the alert (1).

A protocol client receiving an alert (1) can choose the information it needs to provide a richer experience for its users.

## 1.4 Relationship to Other Protocols

Alerts (1) are Internet messages as described in [\[RFC2822\]](#). The **alert metadata** is contained in X-headers as described in [\[RFC822\]](#) section 4.7.5.

Alerts (1) on a protocol server can be managed by the protocol client using the Web services as described in the Alerts Service Protocol Specification ([\[MS-ALERTSS\]](#)).

## 1.5 Prerequisites/Preconditions

There are no fixed preconditions for a protocol server to send alerts (1). Any preconditions are specific to the implementation of that protocol server.

## 1.6 Applicability Statement

The purpose of this protocol is to allow the protocol client to distinguish alerts (1) from other Internet messages, use the metadata to provide a richer user experience, or to build an alert management user interface.

## 1.7 Versioning and Capability Negotiation

None.

## 1.8 Vendor-Extensible Fields

This protocol defines the **X-AlertServerType** header (section [2.2.4](#)) where a protocol server MAY<1> identify itself to the protocol client. Based on the type of the server identified, the protocol client MAY<2> use its knowledge about any services that this type of server offers and provide them to the end user accordingly.

## 1.9 Standards Assignments

None.

## 2 Messages

The following sections specify how alerts (1) are transported and the alert syntax.

### 2.1 Transport

Alerts (1) are Internet messages, fully compliant with [\[RFC2822\]](#). They have a specific value in the **Message-ID** header (section [2.2.1](#)), and contain a variety of metadata in X-headers, as allowed by [\[RFC2822\]](#). These headers and values are specified in [Message Syntax](#).

Internet messages, and thus alerts (1), can be transported in many ways. The exact transport method is not relevant to this protocol. The default transport method is **Simple Mail Transfer Protocol (SMTP)** specified in [\[RFC2821\]](#).

### 2.2 Message Syntax

Alerts (1) conform to the form and behavior of Internet messages as specified in [\[RFC2822\]](#). The following sections specify extensions and additions to headers of alerts (1).

#### 2.2.1 Message-ID

This protocol extends **message-id** that is defined in [\[RFC2822\]](#). In this protocol, the **Message-ID** header indicates that the Internet message is an alert (1) by beginning with a left angle bracket (<) and the alert GUID. The alert GUID is fixed for all alerts (1) and has the value "3BD50098E401463AA228377848493927".

The syntax of this header is defined as follows by using the **Augmented Backus-Naur Form (ABNF)**, as defined in [\[RFC5234\]](#), syntax, as specified in [\[RFC2822\]](#):

```
alert-message-id = "Message-ID:" alert-msg-id CRLF
alert-msg-id     = [CFWS] "<" alert-guid "-" id-left "@"
                  id-right ">" [CFWS]
alert-guid       = "3BD50098E401463AA228377848493927"
id-left         = dot-atom-text / no-fold-quote / obs-id-left
id-right        = dot-atom-text / no-fold-literal /
                  obs-id-right
```

To show that the **Message-ID** header in this protocol is an extension of **Message-ID** in [\[RFC2822\]](#), [\[RFC2822\]](#) section 3.6.4 defines a **message-id** as follows:

```
message-id      = "Message-ID:" msg-id CRLF
msg-id          = [CFWS] "<" id-left "@" id-right ">" [CFWS]
id-left        = dot-atom-text / no-fold-quote / obs-id-left
id-right       = dot-atom-text / no-fold-literal /
                  obs-id-right
```

Based on the preceding definitions of **alert-message-id** and **message-id**, if **alert-guid** is considered as a portion of **id-left**, an alert (1), represented by **alert-message-id**, conforms to the definition of **message-id**.

The **Message-ID** header MUST be present. If **alert-message-id** as defined earlier is present in **Message-ID**, the protocol client considers the Internet message as an ALERT and processes the additional alert metadata in the headers as defined in **X-AlertId** (section [2.2.2](#)) through **X-Sharing-Title** (section [2.2.11](#)).

## 2.2.2 X-AlertId

The **X-AlertId** header contains a string uniquely identifying the alert subscription on the protocol server. The syntax of this header is defined as follows by using the ABNF, as defined in [\[RFC5234\]](#), syntax, as specified in [\[RFC2822\]](#):

```
alert-id = "X-AlertId:" unstructured CRLF
```

**X-AlertId** MUST be present in the header of the alert (1). If the protocol client uses the protocol specified by [\[MS-ALERTSS\]](#) to manage alert subscriptions on a protocol server, the protocol client MUST use the value of **X-AlertId** formatted as a GUID to identify an alert subscription in a protocol operation. Aside from passing the string back as a parameter to the protocol server in such contexts, the protocol client does not use this string for other purposes.

## 2.2.3 X-AlertTitle

The **X-AlertTitle** header contains the title of the alert (1). The syntax of this header is defined as follows by using the ABNF, as defined in [\[RFC5234\]](#), syntax, as specified in [\[RFC2822\]](#):

```
alert-title = "X-AlertTitle:" encoded-alert-word CRLF
encoded-alert-word = "=?utf-8?B?" encoded-text "=?"
```

**X-AlertTitle** MUST be present in the header of the alert (1). The protocol client can choose to use the value of **X-AlertTitle** to display user interface specific to this alert (1). See [\[RFC2047\]](#) for details of the **encoded-text** field. **Encoded-alert-word** conforms to **encoded-word** as defined in [\[RFC2047\]](#), with charset always being set to "utf-8" and encoding always being set to "B" (indicating a Base64 encoding).

## 2.2.4 X-AlertServerType

The **X-AlertServerType** header contains an **ASCII** string identifying the protocol server software which generated the alert (1). The syntax of this header is defined as follows by using the ABNF, as defined in [\[RFC5234\]](#), syntax, as specified in [\[RFC2822\]](#):

```
alert-server-type = "X-AlertServerType:" unstructured CRLF
```

**X-AlertServerType** is an optional header in an alert (1) [<3>](#). When **X-AlertServerType** is present, in conjunction with the **X-AlertWebUrl** header, a protocol client [MAY<4>](#) take advantage of any other services it is aware of that is provided by the protocol server identified in this header.

## 2.2.5 X-AlertWebUrl

The **X-AlertWebUrl** header contains a Web URL to the protocol server which generated the alert (1). The syntax of this header is defined as follows by using the ABNF, as defined in [\[RFC5234\]](#), syntax, as specified in [\[RFC2822\]](#):

```
alert-web-url = "X-AlertWebUrl:" encoded-alert-word CRLF
```

The **encoded-alert-word** is defined in section [2.2.3](#).

**X-AlertWebUrl** is an optional header in an alert (1).

### 2.2.6 X-AlertWebSoap

The **X-AlertWebSoap** header contains a URL to the Web service which implements the [\[MS-ALERTSS\]](#) protocol and which is associated with the protocol server that generated this alert (1). The syntax of this header is defined as follows by using the ABNF, as defined in [\[RFC5234\]](#), syntax, as specified in [\[RFC2822\]](#):

```
alert-web-soap = "X-AlertWebSoap:" encoded-alert-word CRLF
```

The `encoded-alert-word` is defined in section [2.2.3](#).

**X-AlertWebSoap** is an optional header in an alert (1). When **X-AlertWebSoap** is present, the protocol client can use this URL to edit or delete alert subscriptions using the protocol defined in [\[MS-ALERTSS\]](#).

### 2.2.7 X-Sharing-Config-Url

The **X-Sharing-Config-Url** header contains a URL which, when activated, initiates the synchronization of the container of the resource referred by the alert (1). The syntax of this header is defined as follows by using the ABNF, as defined in [\[RFC5234\]](#), syntax, as specified in [\[RFC2822\]](#):

```
sharing-config-url = "X-Sharing-Config-Url:" unstructured CRLF
```

**X-Sharing-Config-Url** is an optional header in an alert (1)[<5>](#). If synchronization of the container of the resource with the protocol client is applicable[<6>](#), **X-Sharing-Config-Url** MUST be present. When **X-Sharing-Config-Url** is present, the protocol client MAY[<7>](#) present user interface to allow the user to call this URL.

### 2.2.8 X-Sharing-Remote-Uid

The **X-Sharing-Remote-Uid** header contains a unique identifier that identifies the container of the resource referred by the alert (1). The syntax of this header is defined as follows by using the ABNF, as defined in [\[RFC5234\]](#), syntax, as specified in [\[RFC2822\]](#):

```
sharing-remote-uid = "X-Sharing-Remote-Uid:"  
                    encoded-alert-word CRLF
```

The `encoded-alert-word` is defined in section [2.2.3](#)[.<8>](#)

**X-Sharing-Remote-Uid** is an optional header in an alert (1). If synchronization of the container of the resource with the protocol client is applicable,[<9>](#) **X-Sharing-Remote-Uid** MUST be present. The value of this header does not carry any special meaning to the protocol client; the protocol client does not process the value of this header for any purpose.

## 2.2.9 X-Sharing-WssBaseUrl

The **X-Sharing-WssBaseUrl** header contains the Web URL to the container of the resource referred by the alert (1). The syntax of this header is defined as follows by using the ABNF, as defined in [\[RFC5234\]](#), syntax, as specified in [\[RFC2822\]](#):

```
sharing-wssbaseurl = "X-Sharing-WssBaseUrl:"  
                    encoded-alert-word CRLF
```

The `encoded-alert-word` is defined in section [2.2.3](#).

**X-Sharing-WssBaseUrl** is an optional header in an alert (1). If synchronization of the container of the resource with the protocol client is applicable, [<10>](#) **X-Sharing-WssBaseUrl** MUST be present.

## 2.2.10 X-Sharing-ItemId

The **X-Sharing-ItemId** header contains a unique identifier for the resource referred by the alert (1). The syntax of this header is defined as follows by using the ABNF, as defined in [\[RFC5234\]](#), syntax, as specified in [\[RFC2822\]](#):

```
sharing-itemid = "X-Sharing-ItemId:" encoded-alert-word CRLF
```

The `encoded-alert-word` is defined in section [2.2.3](#).

If the alert is under **workflow task**, **X-Sharing-ItemId** SHOULD be present in the header of the alert (1).

## 2.2.11 X-Sharing-Title

The **X-Sharing-Title** header contains the title of the resource that this alert (1) refers to. The syntax of this header is defined as follows by using the ABNF, as defined in [\[RFC5234\]](#), syntax, as specified in [\[RFC2822\]](#):

```
sharing-title = "X-Sharing-Title:" encoded-alert-word CRLF
```

The `encoded-alert-word` is defined in section [2.2.3](#).

If the alert is under workflow task, **X-Sharing-Title** SHOULD be present in the header of the alert (1).

## 3 Protocol Details

In this section, the behaviors of the protocol server and protocol client for this protocol are specified. This section also includes details on abstract data models, initialization, higher-layer triggered events, and message processing rules.

### 3.1 Alerts Interoperability Server Details

#### 3.1.1 Abstract Data Model

This section describes a conceptual model of possible data organization that an implementation maintains to participate in this protocol. The described organization is provided to facilitate the explanation of how the protocol behaves. This document does not mandate that an implementation adhere to this model as long as their external behavior is consistent with that described in this document.

The headers of an alert (1) MUST contain the following information:

- An identifier indicating that the message is an alert (1).
- A unique identifier for the alert subscription.
- The title of the alert (1).

The headers of an alert (1) SHOULD contain the following information:

- The protocol server software that sent the alert (1).
- The URL of the protocol server that sent the alert (1).
- The URL of the Web service that the protocol server used to manage the alert subscription.

The headers of an alert (1) SHOULD contain the following information, if relevant to the container of the resource that is the subject of the alert (1):

- The URL to initiate synchronizing the protocol client with the container of the resource referred by the alert (1).
- The identifier and URL of the container of the resource referred by the alert (1).
- The identifier and title of the resource referred by the alert (1).

#### 3.1.2 Timers

None.

#### 3.1.3 Initialization

Conceptually, the protocol initialization occurs when a user, while browsing a protocol server, chooses to sign up for some type of alert (1). The exact method of signing up for an alert (1) is not specified; and in fact can vary depending on the alert type and condition. The protocol server MUST keep an active list of alert subscriptions, per user, and be aware of each user's e-mail address.

### 3.1.4 Higher-Layer Triggered Events

The condition(s) that trigger an alert (1) on a protocol server vary according to the design of the protocol server. When creating an alert subscription, the protocol server SHOULD indicate the condition that will trigger the alert (1).

Conditions that a protocol server can provide as mechanisms to alert the user include the following:

- When a given resource on the protocol server is modified.
- When additional content is added to the protocol server.
- When a given resource is removed.

When a condition occurs on the protocol server and triggers an alert (1), the protocol server MUST check the active list of alert subscriptions for all users, and MUST then send an alert (1) to each user who requested an alert (1) for that condition.

### 3.1.5 Message Processing Events and Sequencing Rules

To put together an alert (1), a protocol server MUST specify the alert GUID in **Message-ID** (section [2.2.1](#)). If using the SMTP transport, the protocol server MUST also specify **X-AlertId** (section [2.2.2](#)) and **X-AlertTitle** (section [2.2.3](#)), and SHOULD specify the other X-headers in the header if those headers are applicable. See section [2.2.7](#) to section [2.2.11](#) for details about certain X-headers being applicable in only certain scenarios.

The protocol server SHOULD NOT, however, assume that the protocol client is aware of these headers. The protocol server MUST therefore, also include context about the alert (1) in the subject and body of the Internet message to make the alert (1) useful to the user. Specifically, the alert SHOULD have the following information:

- The subject SHOULD include the title of the alert (1).
- The body SHOULD [<11>](#) include the condition that triggered the alert (1).
- The body SHOULD [<12>](#) include the context in which the alert was triggered. For example, if the alert (1) was triggered by a page on the protocol server being modified, the protocol server will include who modified the page in the alert (1), and provide links to the protocol server so that if the protocol client is not aware of the alert protocol, headers, and alert metadata described in this document, the user can still manually manage the alert settings.

### 3.1.6 Timer Events

None.

### 3.1.7 Other Local Events

None.

## 3.2 Alerts Interoperability Client Details

### 3.2.1 Abstract Data Model

None.

### 3.2.2 Timers

None.

### 3.2.3 Initialization

None.

### 3.2.4 Higher-Layer Triggered Events

None.

### 3.2.5 Message Processing Events and Sequencing Rules

When a protocol client downloads an e-mail, it MUST check the contents of the **Message-ID** header (section [2.2.1](#)). If it contains the alert GUID, the protocol client MUST consider that e-mail an alert (1) and SHOULD check for the additional alert metadata contained in the various X-headers, and take whatever action it deems appropriate.

#### 3.2.5.1 X-AlertId

This header provides the protocol client a way to uniquely identify the alert subscription on the protocol server. Using this value in conjunction with the [\[MS-ALERTSS\]](#) protocol, the protocol client can choose to provide user interface to the user to delete the alert subscription.

#### 3.2.5.2 X-AlertTitle

This header provides the protocol client information to display alert-specific user interface to the user.

#### 3.2.5.3 X-AlertServerType

This header specifies the protocol server software which generated the alert (1). Using this information and **X-AlertWebUrl**, the protocol client MAY [<13>](#) provide mechanisms for the user to access any services that the protocol server provides.

#### 3.2.5.4 X-AlertWebUrl

This header specifies the Web URL of the protocol server that generated the alert (1). Using this information, the protocol client can display alert-specific user interface or access non-alert-related services.

#### 3.2.5.5 X-AlertWebSoap

This header specifies the URL of the web service implementing the [\[MS-ALERTSS\]](#) protocol used by this protocol server. This protocol defines operations for the protocol client to manage alert subscriptions. The protocol client can display user interface to allow users to manage their alert subscriptions.

#### 3.2.5.6 X-Sharing-Config-Url

This header specifies the URL used to initiate synchronization of the container of the resource referred by the alert (1). The protocol client MAY [<14>](#) display user interface and use this URL to allow the user to initiate the synchronization of this resource with the protocol client. [<15>](#) The

protocol client SHOULD ignore this header if the protocol client does not support local synchronization of resources from the protocol server.

#### **3.2.5.7 X-Sharing-Remote-Uid**

This header specifies the remote identifier of the container of the resource referred by the alert (1). The protocol client can use this value as a parameter when accessing non-alert services that the protocol server provides.

#### **3.2.5.8 X-Sharing-WssBaseUrl**

This header specifies the Web URL of the container of the resource referred to by the alert (1). Using this information, the protocol client can provide a user interface for the user to navigate directly to the container by using their Web browser.

#### **3.2.5.9 X-Sharing-ItemId**

This header specifies the unique identifier of the resource referred to by this alert (1). The protocol client can use this value as a parameter when accessing non-alert services that the protocol server provides.

#### **3.2.5.10 X-Sharing-Title**

This header specifies the title of the resource referred to by this alert. The protocol client can choose to display user interface to this effect.

#### **3.2.6 Timer Events**

None.

#### **3.2.7 Other Local Events**

None.

## 4 Protocol Examples

The following is a sample SMTP header from an alert (1). The relevant headers are bold.

Received: from example.microsoft.com (10.0.0.1) by

```
example2.microsoft.com (10.0.0.2) with Microsoft SMTP
Server id 8.1.240.5; Thu, 24 Jan 2008 10:06:05 -0800
Date: Thu, 24 Jan 2008 10:05:45 -0800
To: <john@microsoft.com>
Message-ID: <3BD50098E401463AA228377848493927-{B9F3612B-19E8-4383-AD27-
D5ACD2C2EE6F}@example.microsoft.com>
Title: =?utf-8?B?V2lraSBQYWdlc2==?=
X-AlertId: {913D756A-E669-4969-A930-8CD9845926D8};{D5D996C5-683C-4077-AB40-9F3721E30B9D}
Content-Transfer-Encoding: 8bit
From: =?utf-8?B?T3V0bG9vayBUZWFtIFdpa2lz?= <someserver1@microsoft.com>
X-AlertWebUrl: =?utf-8?B? aHR0cDovL2V4YW1wbGUvc2FtcGxl===?=
X-AlertServerType: STS
Content-Type: text/html; charset="utf-8"
Subject: Wiki Pages - Test.aspx
Reply-To: <mike@example.microsoft.com>
X-AlertWebSoap: =?utf-8?B? aHR0cDovL2V4YW1wbGUvc2VydGVyL192dGlfYmluL2FsZXJ0cy5hc214===?=
MIME-Version: 1.0
X-Mailer: WSS (version 3)
Return-Path: someserver1@example.microsoft.com
X-OriginalArrivalTime: 24 Jan 2008 18:05:45.0895 (UTC) FILETIME=[BDF1B370:01C85EB3]
```

Looking at each header individually:

```
Message-ID: <3BD50098E401463AA228377848493927-{B9F3612B-19E8-4383-AD27-
D5ACD2C2EE6F}@example.microsoft.com>
```

The **Message-ID** header (section [2.2.1](#)) begins with the alert GUID (underlined), identifying this message as an alert (1).

```
X-AlertTitle: =?utf-8?B?V2lraSBQYWdlc2==?=
```

The title for the alert (1) is "Wiki Pages".

```
X-AlertId: {913D756A-E669-4969-A930-8CD9845926D8};{D5D996C5-683C-4077-AB40-9F3721E30B9D}
```

The protocol client does not process this value for any purpose other than passing it as a parameter back to the protocol server as specified in section [2.2.2](#).

```
X-AlertWebUrl: =?utf-8?B? aHR0cDovL2V4YW1wbGUvc2FtcGxl===?=
```

The URL of the protocol server that generated the alert (1) is "http://example/sample".

```
X-AlertServerType: STS
```

The originating protocol server of the alert (1) is a SharePoint server.

X-AlertWebSoap: =?utf-8?B? aHR0cDovL2V4YW1wbGUvc2VydGVyL192dG1fYmluL2FsZXJ0cy5hc214==?="

The URL of the Web service used by the originating protocol server is "http://example/sample/\_vti\_bin/alerts.asmx".

Preliminary

## 5 Security

### 5.1 Security Considerations for Implementers

There are no specific security considerations specific to this protocol. General security considerations pertaining to [RFC2822](#) apply to this protocol.

### 5.2 Index of Security Parameters

None.

## 6 Appendix A: Product Behavior

The information in this specification is applicable to the following Microsoft products or supplemental software. References to product versions include released service packs:

- Microsoft® Office Outlook® 2003
- Microsoft® Office Outlook® 2007
- Microsoft® Outlook® 2010
- Microsoft® Outlook® 15 Technical Preview
- Windows® SharePoint® Services 3.0
- Microsoft® SharePoint® Foundation 2010
- Microsoft® SharePoint® Foundation 15 Technical Preview

Exceptions, if any, are noted below. If a service pack or Quick Fix Engineering (QFE) number appears with the product version, behavior changed in that service pack or QFE. The new behavior also applies to subsequent service packs of the product unless otherwise specified. If a product edition appears with the product version, behavior is different in that product edition.

Unless otherwise specified, any statement of optional behavior in this specification that is prescribed using the terms SHOULD or SHOULD NOT implies product behavior in accordance with the SHOULD or SHOULD NOT prescription. Unless otherwise specified, the term MAY implies that the product does not follow the prescription.

[<1> Section 1.8:](#) Windows SharePoint Services 3.0 and SharePoint Foundation 2010 specify "STS" as the value of the **X-AlertServerType** header.

[<2> Section 1.8:](#) Office Outlook 2007 Service Pack 1 (SP1) displays a specific icon for known server types, and provides access to other services known to exist on Windows SharePoint Services 3.0 SP1.

[<3> Section 2.2.4:](#) Windows SharePoint Services 3.0 and SharePoint Foundation 2010 specify "STS" as the value of the **X-AlertServerType** header.

[<4> Section 2.2.4:](#) Office Outlook 2007 Service Pack 1 (SP1) displays a specific icon for known server types, and provides access to other services known to exist on Windows SharePoint Services 3.0 SP1.

[<5> Section 2.2.7:](#) Windows SharePoint Services 3.0 and SharePoint Foundation 2010 specify a well-formed URL, "stssync://", in the **X-Sharing-Config-Url** header.

[<6> Section 2.2.7:](#) In Windows SharePoint Services 3.0 and SharePoint Foundation 2010, calendars, document libraries, discussion boards, contact lists, and task lists can be synchronized with My Calendar, My Documents, Inbox, address book and contacts, and tasks, respectively, in Office Outlook 2007 SP1 and Outlook 2010.

[<7> Section 2.2.7:](#) If the X-Sharing-Config-Url header is present, it begins with one of the following values: "feed", "feeds", "outlookfeed", "outlookfeeds", "webcal", "webcals", or "stssync", otherwise Office Outlook 2007 SP1 will fail to open or display the user interface to call the URL. If X-Sharing-Config-Url is present and contains a valid value, Office Outlook 2007 SP1 displays user interface to allow the user to call the URL and synchronize Outlook with the resource referred by the alert. The format of an X-Sharing-Config-Url beginning with "stssync" is specified in the StsSync Structure

Specification ([\[MS-STSSYN\]](#)). The SharePoint synchronization protocol for these types of URLs is defined in the Lists Client Sync Protocol Specification ([\[MS-OUTSPS\]](#)).

<8> [Section 2.2.8:](#) SharePoint Foundation 2010 returns the value of the header as a GUID without encoding.

<9> [Section 2.2.8:](#) In Windows SharePoint Services 3.0 and SharePoint Foundation 2010, calendars, document libraries, discussion boards, contact lists, and task lists can be synchronized with My Calendar, My Documents, Inbox, address book and contacts, and tasks, respectively, in Office Outlook 2007 SP1 and Outlook 2010.

<10> [Section 2.2.9:](#) In Windows SharePoint Services 3.0 SP1, calendars, document libraries, discussion boards, contact lists, and task lists can be synchronized with My Calendar, My Documents, Inbox, address book and contacts, and tasks respectively in Outlook 2007 SP1.

<11> [Section 3.1.5:](#) Windows SharePoint Services 3.0, SharePoint Foundation 2010: The condition can be configured. This behavior is updated to support features as described by the MSDN Knowledgebase Article #161512, "Working with Search Alerts in SharePoint 2010".

<12> [Section 3.1.5:](#) Windows SharePoint Services 3.0, SharePoint Foundation 2010: The context can be configured. This behavior is updated to support features as described by the MSDN Knowledgebase Article #161512, "Working with Search Alerts in SharePoint 2010".

<13> [Section 3.2.5.3:](#) Office Outlook 2007 Service Pack 1 (SP1) displays a specific icon for known server types, and provides access to other services known to exist on Windows SharePoint Services 3.0 SP1.

<14> [Section 3.2.5.6:](#) If the X-Sharing-Config-Url header is present, it begins with one of the following values: "feed", "feeds", "outlookfeed", "outlookfeeds", "webcal", "webcals", or "stssync", otherwise Office Outlook 2007 SP1 will fail to open or display the user interface to invoke the URL. If X-Sharing-Config-Url is present and contains a valid value, Office Outlook 2007 SP1 displays user interface to allow the user to invoke the URL and synchronize Outlook with the resource referred by the alert. The format of an X-Sharing-Config-Url beginning with "stssync" is specified in the StsSync Structure Specification ([\[MS-STSSYN\]](#)). The SharePoint synchronization protocol for these types of URLs is defined in the Lists Client Sync Protocol Specification ([\[MS-OUTSPS\]](#)).

<15> [Section 3.2.5.6:](#) In Windows SharePoint Services 3.0 and SharePoint Foundation 2010, calendars, document libraries, discussion boards, contact lists, and task lists can be synchronized with My Calendar, My Documents, Inbox, address book and contacts, and tasks, respectively, in Office Outlook 2007 SP1 and Outlook 2010.

## 7 Change Tracking

No table of changes is available. The document is either new or has had no changes since its last release.

Preliminary

## 8 Index

### A

Abstract data model  
[client](#) 14  
[server](#) 13  
[Applicability](#) 8

### C

[Capability negotiation](#) 8  
[Change tracking](#) 22  
Client  
[abstract data model](#) 14  
[higher-layer triggered events](#) 15  
[initialization](#) 15  
[message processing](#) 15  
[other local events](#) 16  
[overview](#) 13  
[sequencing rules](#) 15  
[timer events](#) 16  
[timers](#) 15

### D

Data model - abstract  
[client](#) 14  
[server](#) 13

### E

Examples  
[overview](#) 17

### F

[Fields - vendor-extensible](#) 8

### G

[Glossary](#) 6

### H

Header  
[X-AlertTitle](#) 15  
Headers  
[X-AlertId](#) 15  
[X-AlertServerType](#) 15  
[X-AlertWebSoap](#) 15  
[X-AlertWebUrl](#) 15  
[X-Sharing-Config-Url](#) 15  
[X-Sharing-ItemId](#) 16  
[X-Sharing-Remote-Uid](#) 16  
[X-Sharing-Title](#) 16  
[X-Sharing-WssBaseUrl](#) 16  
Higher-layer triggered events  
[client](#) 15  
[server](#) 14

### I

[Implementer - security considerations](#) 19  
[Index of security parameters](#) 19  
[Informative references](#) 7  
Initialization  
[client](#) 15  
[server](#) 13  
[Introduction](#) 6

### M

Message processing  
[client](#) 15  
[server](#) 14  
[Message-ID message](#) 9  
Messages  
[Message-ID](#) 9  
[syntax](#) 9  
[transport](#) 9  
[X-AlertId](#) 10  
[X-AlertServerType](#) 10  
[X-AlertTitle](#) 10  
[X-AlertWebSoap](#) 11  
[X-AlertWebUrl](#) 10  
[X-Sharing-Config-Url](#) 11  
[X-Sharing-ItemId](#) 12  
[X-Sharing-Remote-Uid](#) 11  
[X-Sharing-Title](#) 12  
[X-Sharing-WssBaseUrl](#) 12

### N

[Normative references](#) 6

### O

Other local events  
[client](#) 16  
[server](#) 14  
[Overview \(synopsis\)](#) 7

### P

[Parameters - security index](#) 19  
[Preconditions](#) 8  
[Prerequisites](#) 8  
[Product behavior](#) 20

### R

[References](#) 6  
[informative](#) 7  
[normative](#) 6  
[Relationship to other protocols](#) 8

### S

Security  
[implementer considerations](#) 19  
[parameter index](#) 19

Sequencing rules  
[client](#) 15  
[server](#) 14

Server  
[abstract data model](#) 13  
[higher-layer triggered events](#) 14  
[initialization](#) 13  
[message processing](#) 14  
[other local events](#) 14  
[overview](#) 13  
[sequencing rules](#) 14  
[timer events](#) 14  
[timers](#) 13  
[Standards assignments](#) 8

Syntax  
[messages - overview](#) 9

## T

Timer events  
[client](#) 16  
[server](#) 14

Timers  
[client](#) 15  
[server](#) 13  
[Tracking changes](#) 22  
[Transport](#) 9

Triggered events - higher-layer  
[client](#) 15  
[server](#) 14

## V

[Vendor-extensible fields](#) 8  
[Versioning](#) 8

## X

[X-AlertId header](#) 15  
[X-AlertId message](#) 10  
[X-AlertServerType header](#) 15  
[X-AlertServerType message](#) 10  
[X-AlertTitle header](#) 15  
[X-AlertTitle message](#) 10  
[X-AlertWebSoap header](#) 15  
[X-AlertWebSoap message](#) 11  
[X-AlertWebUrl header](#) 15  
[X-AlertWebUrl message](#) 10  
[X-Sharing-Config-Url header](#) 15  
[X-Sharing-Config-Url message](#) 11  
[X-Sharing-ItemId header](#) 16  
[X-Sharing-ItemId message](#) 12  
[X-Sharing-Remote-Uid header](#) 16  
[X-Sharing-Remote-Uid message](#) 11  
[X-Sharing-Title header](#) 16  
[X-Sharing-Title message](#) 12  
[X-Sharing-WssBaseUrl header](#) 16  
[X-Sharing-WssBaseUrl message](#) 12