[MS-FSWASDR]:
WebAnalyzer/SPRel Data Receiving 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.
- **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.

Copyright © 2012 Microsoft Corporation.

Revision Summary

Date	Revision History	Revision Class	Comments
11/06/2009	0.1	Major	Initial Availability
02/19/2010	1.0	Minor	Updated the technical content
03/31/2010	1.01	Editorial	Revised and edited the technical content
04/30/2010	1.02	Editorial	Revised and edited the technical content
06/07/2010	1.03	Editorial	Revised and edited the technical content
06/29/2010	1.04	Editorial	Changed language and formatting in the technical content.
07/23/2010	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
09/27/2010	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
11/15/2010	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
12/17/2010	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
03/18/2011	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
06/10/2011	1.04	No change	No changes to the meaning, language, or formatting of the technical content.
01/20/2012	1.5	Minor	Clarified the meaning of the technical content.
04/11/2012	1.5	No change	No changes to the meaning, language, or formatting of the technical content.
07/16/2012	1.5	No change	No changes to the meaning, language, or formatting of the technical content.

Copyright © 2012 Microsoft Corporation.

Table of Contents

1 Introduction 5
1.1 Glossary 5
1.2 References 5
1.2.1 Normative References 5
1.2.2 Informative References 6
1.3 Protocol Overview (Synopsis) 6
1.4 Relationship to Other Protocols 6
1.5 Prerequisites/Preconditions 6
1.6 Applicability Statement 6
1.7 Versioning and Capability Negotiation 7
1.8 Vendor-Extensible Fields 7
1.9 Standards Assignments 7

2 Messages 8
2.1 Transport 8
2.2 Message Syntax 8
2.2.1 Data 8
2.2.1.1 Application 8
2.2.1.2 Collection 8
2.2.1.3 Batch 8
2.2.1.3.1 WebAnalyzer batch 8
2.2.1.3.1.1 links 9
2.2.1.3.1.2 urieq 9
2.2.1.3.1.3 eqrepr 10
2.2.1.3.1.4 delete 10
2.2.1.3.1.5 no_links 11
2.2.1.3.1.6 urimap 11
2.2.1.3.1.7 sitemap 12
2.2.1.3.2 SPRel batch 12
2.2.1.3.2.1 uris 12
2.2.2 Response 13

3 Protocol Details 14
3.1 Client Details 14
3.1.1 Abstract Data Model 14
3.1.2 Timers 14
3.1.3 Initialization 14
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 Server 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
3.2.6 Timer Events 15
3.2.7 Other Local Events 15

Copyright © 2012 Microsoft Corporation.

4 Protocol Examples 16
4.1 Send Data to the WebAnalyzer 16
4.1.1 Parse the Metadata 16
4.1.2 Build the Message 17
4.1.3 Send the Message 18
4.2 Send Data to SPRel 18
4.2.1 Parse the Metadata 18
4.2.2 Build the Message 19
4.2.3 Send the Message 20

5 Security 21
5.1 Security Considerations for Implementers 21
5.2 Index of Security Parameters 21

6 Appendix A: Product Behavior 22

7 Change Tracking 23

8 Index 24

Copyright © 2012 Microsoft Corporation.

1 Introduction

This document specifies the WebAnalyzer/SPRel Data Receiving Protocol. This protocol sends data from a protocol client to a protocol server and for the response from the protocol server to the protocol client. This protocol allows the protocol client to send data to a protocol server that runs on another computer.

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
base64
Coordinated Universal Time (UTC)
MD5 hash
Transmission Control Protocol (TCP)

The following terms are defined in [MS-OFCGLOS]:

anchor text
base port
crawl collection
document identifier
equivalence class
host name
hyperlink
search clickthrough
site
Uniform Resource Locator (URL)

The following terms are specific to this document:

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 technical 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. 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.

Copyright © 2012 Microsoft Corporation.

[MS-FSCX] Microsoft Corporation, "Configuration (XML-RPC) Protocol Specification".

[MS-FSWCU] Microsoft Corporation, "WebAnalyzer/Crawler Utility Structure Specification".

[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

1.2.2 Informative References

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

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

1.3 Protocol Overview (Synopsis)

This protocol enables a protocol client to send data to a protocol server. It transfers **URLs**, **hyperlinks** and **anchor text** and other document metadata to a protocol server that analyzes either hyperlinks or **search clickthrough** logs.

The protocol sends multiple data packets over the same connection. All information is sent using **TCP** in one batch. The protocol does therefore not support streaming data, because batch size depends on the amount of available memory.

Communications consist of the following:

1. The protocol client sends a message that contains the data to the protocol server.
2. The protocol server parses the message and sends a message to the protocol client that indicates whether the data was successfully received.

1.4 Relationship to Other Protocols

This protocol uses TCP as its transport protocol to send messages between the protocol client and the protocol server, as shown in the following diagram.

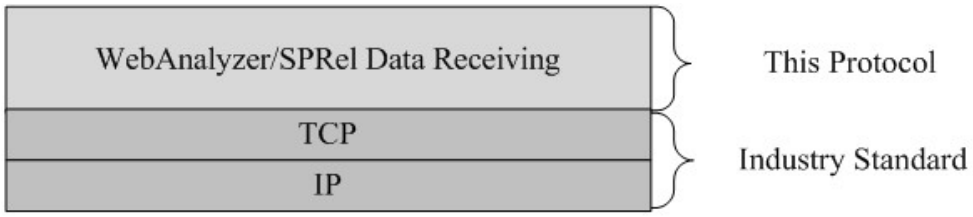

Figure 1: The protocol in relation to other protocols

1.5 Prerequisites/Preconditions

A TCP/IP connection from the protocol client to the protocol server MUST exist before the protocol can be used. The protocol server port is **base port** plus 291. This protocol requires that the protocol client acquire the **host name** and port for the protocol server that receives the WebAnalyzer/SPRel data.

1.6 Applicability Statement

None.

Copyright © 2012 Microsoft Corporation.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

Copyright © 2012 Microsoft Corporation.

2 Messages

2.1 Transport

Messages are transmitted over TCP, and the protocol server MUST listen on base port plus 291. There are two types of messages: the protocol client sends requests to the protocol server, and the protocol server sends responses to the protocol client.

2.2 Message Syntax

2.2.1 Data

The **data** message represents the message that the protocol client sends to the protocol server. It is a dictionary that is serialized as specified in [MS-FSWCU] section 2.2.2.

The dictionary contains three entries: application, collection, and batch entries. Each entry in the dictionary is referred to as a key-value pair. The serialized string ends with the following string, which the protocol server uses to determine whether it received a complete message:

```
%x2E%x2E%x2E%x00%x00%x00%x00
```

2.2.1.1 Application

The **application** entry determines which application to which to send the data message. The key of the key-value pair is an **ASCII** string, and it contains one of the values specified in the following table.

Key	Description
webanalyzer	Specifies that the protocol client sends the data message to the WebAnalyzer subsystem.
sharepoint	Specifies that the protocol client sends the data message to the SPRel subsystem.

2.2.1.2 Collection

The **collection** entry contains the name of the **crawl collection** to which the data belongs. The key of the key-value pair is an ASCII string that contains the value "collection**"**.

2.2.1.3 Batch

The **batch** entry contains the data or payload to send to the receiver. The key of the key-value pair is a string that contains the value "batch". The value of the key-value pair is specified in sections 2.2.1.3.1 and 2.2.1.3.2 respectively.

2.2.1.3.1 WebAnalyzer batch

If the **application** entry contains the value "webanalyzer" then the batch contains at least one entry specified in sections 2.2.1.3.1.1 through 2.2.1.3.1.7. The batch can contain more than one entry, or it can contain all entries.

Copyright © 2012 Microsoft Corporation.

2.2.1.3.1.1 links

The **links** key-value pair contains one entry for each hyperlink in each document. The key of the key-value pair is an ASCII string that contains the value "links". The value of the key-value pair contains at least one line if it is present in the message. There is more than one line per document when the message contains more than one hyperlink.

The format for each line is specified as follows.

```
<CIDHASH> <TO> <INTRA> <TIMESTAMP> <ANCHORTEXT>%x0A
```

The values for each element are specified in the following table.

Name	Description
CIDHASH	Represents the **document identifier (3)** of the document and it contains the **base64** encoded **MD5 hash** value for the URL. The length is 21 characters.
TO	Represents the URL to which the hyperlink points. It is the base64 encoded MD5 hash value for the URL. The length is 21 characters.
INTRA	Specifies the location of the URL relative to the document identifier. Contains the value 1 if the hyperlink points to a URL that is located on the same site as the document identifier; otherwise it contains the value 0.
TIMESTAMP	Represents a timestamp that specifies the time that the hyperlink was extracted from the document. All hyperlinks extracted from the document during a run have the same timestamp. Each timestamp is an integer that specifies the time in seconds that elapsed after 00:00:00 1970-01-01 **UTC**. The value is encoded as an ASCII string so that each digit is represented by a byte in the range 48 through 57.
ANCHORTEXT	Contains the anchor text for the hyperlink.

2.2.1.3.1.2 urieq

The **urieq** entry contains mappings between document identifier (3) and URLs in the **equivalence class**. The key of the key-value pair is an ASCII string that contains the value "urieq". The value of the key-value pair contains at least one line if it is present in the message. If a document identifier is associated with more than one URL there is one line for each entry in the equivalence class.

The format for each line is specified as follows.

```
<CIDHASH> <MEMBER> <TIMESTAMP>%x0A
```

The values for each element are specified in the following table.

Name	Description
CIDHASH	Represents the document identifier (3) of the document and it contains the base64 encoded MD5 hash value for the URL. The length is 21 characters.
MEMBER	Represents a URL that is the base64 encoded MD5 hash value. The length is 21 characters.
TIMESTAMP	Represents a timestamp that specifies the time that the equivalence class was extracted from the document. All entries extracted from the same document during the same run

Copyright © 2012 Microsoft Corporation.

Name	Description
	MUST have the same timestamp. Each timestamp is an integer that specifies the time in seconds that elapsed after 00:00:00 1970-01-01 UTC. The value is encoded as an ASCII string so that each digit is represented by a byte in the range 48 through 57.

2.2.1.3.1.3 eqrepr

The **eqrepr** key-value pair contains the entire equivalence class for a document identifier (3) as a single entry. The key of the key-value pair is an ASCII string that contains the value "eqrepr". The value of the key-value pair is a string that contains at least one line if it is present in the message.

The format for each line is specified in the following format:

```
<CIDHASH> <DOCUMENT ID> <TIMESTAMP> <EQREPR>%x0A
```

The values for each element are specified in the following table:

Name	Description
CIDHASH	Represents the document identifier (3) of the document and it contains the base64 encoded MD5 hash value for the URL. The length is 21 characters.
DOCUMENT ID	Represents the document identifier(3) as a URL.
TIMESTAMP	Represents a timestamp that specifies the time that the equivalence class was extracted from the document. Each timestamp is an integer that specifies the time in seconds that elapsed after 00:00:00 1970-01-01 UTC. The value is encoded as an ASCII string so that each digit is represented by a byte in the range 48 through 57.
EQREPR	An ASCII-encoded string that contains all URLs in the equivalence class. The URLs are delimited with a single null byte.

2.2.1.3.1.4 delete

The **delete** key-value pair contains all delete operations that have passed through the system. The key of the key-value pair is an ASCII string that contains the value "delete". The value of the key-value pair is a string that contains at least one line if it is present in the message. There MUST NOT be more than one line per document.

The format for each line is specified as follows.

```
<CIDHASH> <TIMESTAMP>%x0A
```

The values for each element are specified in the following table:

Name	Description
CIDHASH	Represents the document identifier (3) of the document and it contains the base64 encoded MD5 hash value for the URL. The length is 21 characters.
TIMESTAMP	Represents a timestamp that specifies the time that the **delete** operation was received. Each timestamp is an integer that specifies the time in seconds that elapsed after 00:00:00 1970-01-01 UTC. The value is encoded as an ASCII string so that each digit is

Copyright © 2012 Microsoft Corporation.

Name	Description
	represented by a byte in the range 48 through 57.

2.2.1.3.1.5 no_links

The **no_links** key-value pair contains all document identifier (3) that are not present in the **CIDHASH** column in the **links** field. The key of the key-value pair is an ASCII string that contains the value "no_links". The value of the key-value pair is a string with at least one line if it is present in the message. There MUST NOT be more than one line per document.

The format for each line is specified as follows.

```
<DOCUMENT ID> <CIDHASH> <TIMESTAMP>%x0A
```

The values for each element are specified in the following table:

Name	Description
DOCUMENT ID	Represents the document identifier (3) as a URL.
CIDHASH	Represents the document identifier (3). It is the base64 encoded MD5 hash value for the URL. The length is 21 characters.
TIMESTAMP	Represents a timestamp that specifies the time that the document identifier (3) was extracted from the document. Each timestamp is an integer that specifies the time in seconds that elapsed after 00:00:00 1970-01-01 UTC. The value is encoded as an ASCII string so that each digit is represented by a byte in the range 48 through 57.

2.2.1.3.1.6 urimap

The **urimap** key-value pair contains mappings between all **CIDHASH** and **DOCUMENT ID** in **links** and **urieq**. The key of the key-value pair is an ASCII string that contains the value "urimap". The value of the key-value pair is a string that contains at least one line if it is present in the message. There can be more than one line per document.

The format for each line is specified as follows.

```
<DOCUMENT ID> <CIDHASH>%x0A
```

The values for each element are specified in the following table:

Name	Description
DOCUMENT ID	Represents the document identifier (3) as a URL.
CIDHASH	Represents the document identifier (3). It is the base64 encoded MD5 hash value for the URL. The length is 21 characters.

Copyright © 2012 Microsoft Corporation.

2.2.1.3.1.7 sitemap

The **sitemap** key-value pair maps between **sites (2)** and document identifiers (3). The key of the key-value pair is an ASCII string that contains the value "sitemap". The value of the key-value pair is a string with at least one line if it is present in the message. There MUST NOT be more than one line per document.

The format for each line is specified as follows.

```
<CIDHASH> <SITEURL>%x0A
```

The values for each element are specified in the following table.

Name	Description
CIDHASH	Represents the document identifier (3). It is the base64 encoded MD5 hash value for the URL. The length is 21 characters.
SITEURL	Represents a site (2).

2.2.1.3.2 SPRel batch

If the **application** entry value contains "sharepoint**"** then the batch contains only one entry as specified in the following section.

2.2.1.3.2.1 uris

The **uris** key-value pair contains xxxx. The key of the key-value pair is an ASCII string that contains the value "uris". The value of the key-value pair is an ASCII string that contains at least one line if it is present in the message. There can be more than one entry for each document, depending on the **OPERATION** field and the content of the equivalence class.

The format for each line is specified as follows.

```
<DOCUMENT ID> <TIMESTAMP> <OPERATION> <DUPLICATE>%x0A
```

The values for each element are specified in the following table:

Name	Description
DOCUMENT ID	Represents the document identifier (3) as a URL.
TIMESTAMP	Represents a timestamp that specifies the time that the document was processed. Each timestamp is an integer that specifies the time in seconds that elapsed after 00:00:00 1970-01-01 UTC. The value is encoded as an ASCII string so that each digit is represented by a byte in the range 48 through 57.
OPERATION	Represents the operation to perform on the document. Values are specified in the following list. The value that the protocol server specifies depends on whether it adds the document to or deletes the document from the index.

Copyright © 2012 Microsoft Corporation.

Name	Description
	ADD DEL URLSCHANGE
DUPLICATE	Specifies that both the **OPERATION** column and the content of the equivalence class are relevant to xxx. The following table specifies the rules that for the value of the **DUPLICATE** column.

This following table specifies the rules that for the value of the **DUPLICATE** column.

OPERATION	equivalence class	DUPLICATE
DEL	Empty	The value "%xc7%x82"
DEL	Not empty	The value "%xc7%x82"
ADD	Empty	The value "%xc7%x82"
ADD	Not empty	A URL in the equivalence class. There is one line in the **uris** field for each entry in the equivalence class.
URLSCHANGE	Empty	The value "%xc7%x82"
URLSCHANGE	Not empty	A URL in the equivalence class. There is one line in the **uris** field for each entry in the equivalence class.

2.2.2 Response

The response message specifies whether the protocol server successfully received the data message. It is a string whose value MUST be either "ack%x0A" or "failed%x0A".

Copyright © 2012 Microsoft Corporation.

3 Protocol Details

3.1 Client Details

This section specifies details that apply to protocol client behavior.

3.1.1 Abstract Data Model

None.

3.1.2 Timers

None.

3.1.3 Initialization

To locate the host on which the protocol server is running, the protocol client queries the **GetActiveModulesList** method in the Configuration Component, as specified in [MS-FSCX],with a value of "WaLinkStorerReceiver" as module type. See [MS-FSCX] section 2.2.5 for more details.

3.1.4 Higher-Layer Triggered Events

None.

3.1.5 Message Processing Events and Sequencing Rules

The protocol client creates a data message as specified in section 2.2.1, and sends it to the protocol server. If a connection is not available, as specified in section 3.1.3, the protocol client opens a new connection to the protocol server and sends the message.

Then, the protocol client waits for the protocol server to send a response message as specified in section 2.2.2.

If the transmission fails or if the protocol server sends a response that specifies that the message was not successfully received, the protocol client connects to another protocol server, or keeps the message for later submission.

3.1.6 Timer Events

None.

3.1.7 Other Local Events

None.

3.2 Server Details

This section specifies details that apply to protocol server behavior.

3.2.1 Abstract Data Model

None.

Copyright © 2012 Microsoft Corporation.

3.2.2 Timers

None.

3.2.3 Initialization

The protocol server registers with the Configuration Component as specified in [MS-FSCX], with a value of "WaLinkStorerReceiver" as module type.

3.2.4 Higher-Layer Triggered Events

None.

3.2.5 Message Processing Events and Sequencing Rules

When receiving a message, the protocol server reads the message until it encounters the message terminator. When it locates the terminator, the protocol server unmarshals the message to determine the application and collection in which to store the information.

The information is then stored on disk, sorted by application and collection. When the protocol server finishes, it sends a response message that specifies that the transmission was successful.

If the protocol server cannot unmarshal the data or if it cannot write the information to disk, it sends a response message as specified in section 2.2.2 that specifies that the transmission failed.

3.2.6 Timer Events

None.

3.2.7 Other Local Events

None.

Copyright © 2012 Microsoft Corporation.

4 Protocol Examples

4.1 Send Data to the WebAnalyzer

4.1.1 Parse the Metadata

In this example, the protocol client sends a message with metadata from two documents to the WebAnalyzer protocol server. The documents are described in the following table as *doc1* and *doc2*. The documents are processed at exactly 2010-01-01T00:00:00 UTC, and they are contained in the collection named "sp**"**.

Key	Value
doc1	
document identifier	http://www.fourthcoffee.com/
Redirect	http://www.fourthcoffee.com/index.html and http://www.fourthcoffee.com/index.aspx
hyperlinks	http://www.fourthcoffee.com/about.html with the anchor text "about us" http://www.contoso.com/ with the anchor text "another example"
doc2	
document identifier	http://www.fourthcoffee.com/example1.html
Redirect	None
hyperlinks	None

Doc1 uses the following batch attributes : **links**, **urieq**, **eqrepr**, and **urimap**.

The **links** attribute contains one entry for each hyperlink and each entry ends with a newline character (%x0A):

```
7fD/x4S22g9fsZM8Kbi3+ F73ZPmkvq5reFgsSEa4dA 1 1244208852 about us%x0A
7fD/x4S22g9fsZM8Kbi3+ zcHqC/594P3ec4oHm2hoS 0 1244208852 another example%x0A
```

The **urieq** attribute contains one entry for each redirect URL and each entry ends with a newline character (%x0A):

```
7fD/x4S22g9fsZM8Kbi3+ TMj7tEFgrTGlWETqbY5L5 1244208852%x0A
7fD/x4S22g9fsZM8Kbi3+ 5c9HshqGRoVVma8oX4DPz 1244208852%x0A
```

The **eqrepr** contains one entry that ends with a newline character (%x0A):

```
7fD/x4S22g9fsZM8Kbi3+ http://fourthcoffe.com/ 1244208852
http://fourthcoffe.com/index.aspx%x00http://fourthcoffe.com/index.html%x0A
```


Copyright © 2012 Microsoft Corporation.

The **urimap** contains one entry for each URL described in the previous table, and each entry ends with a newline character (%x0A):

```
http://fourthcoffe.com/ 7fD/x4S22g9fsZM8Kbi3+%x0A
http://fourthcoffe.com/about.html F73ZPmkvq5reFgsSEa4dA%x0A
http://contoso.com/ zcHqC/594P3ec4oHm2hoS%x0A
http://fourthcoffe.com/index.aspx TMj7tEFgrTGlWETqbY5L5%x0A
http://fourthcoffe.com/index.html 5c9HshqGRoVVma8oX4DPz%x0A
```

Doc2 uses the following batch attributes: **no_links** and **urimap**.

The **no_links** attribute contains one entry for each hyperlink:

```
http://fourthcoffe.com/example1.html wGLEKGuSUWctKO5xzzJ7J 1244208852%x0A
```

The **urimap** contains one entry for each URL in the previous table:

```
http://fourthcoffe.com/example1.html wGLEKGuSUWctKO5xzzJ7J%x0A
```

4.1.2 Build the Message

The protocol client sends metadata from both documents as a single message. The message resembles the following example.

```
{'application': 'webanalyzer', 'batch': {'eqrepr': '7fD/x4S22g9fsZM8Kbi3+
http://www.fourthcoffee.com/ 1244209302
http://www.fourthcoffee.com/index.aspx%x00http://www.fourthcoffee.com/index.html%x0A',
'urimap': 'http://www.fourthcoffee.com/
7fD/x4S22g9fsZM8Kbi3+%x0Ahttp://www.fourthcoffee.com/about.html
F73ZPmkvq5reFgsSEa4dA%x0Ahttp://www.contoso.com/
zcHqC/594P3ec4oHm2hoS%x0Ahttp://www.fourthcoffee.com/index.aspx
TMj7tEFgrTGlWETqbY5L5%x0Ahttp://www.fourthcoffee.com/index.html
5c9HshqGRoVVma8oX4DPz%x0Ahttp://www.fourthcoffee.com/example1.html
wGLEKGuSUWctKO5xzzJ7J%x0A', 'no_links': 'http://www.fourthcoffee.com/example1.html
wGLEKGuSUWctKO5xzzJ7J 1244209302%x0A', 'links': '7fD/x4S22g9fsZM8Kbi3+ F73ZPmkvq5reFgsSEa4dA
1 1244209302 about us%x0A7fD/x4S22g9fsZM8Kbi3+ zcHqC/594P3ec4oHm2hoS 0 1244209302 another
example%x0A', 'urieq': '7fD/x4S22g9fsZM8Kbi3+ TMj7tEFgrTGlWETqbY5L5
1244209302%x0A7fD/x4S22g9fsZM8Kbi3+ 5c9HshqGRoVVma8oX4DPz 1244209302%x0A'}, 'collection':
'sp'}
```

The serialized string that the protocol client sends to the protocol server resembles the following example.

```
%x7b%x73%x0b%x00%x00%x00%x61%x70%x70%x6c%x69%x63%x61%x74%x69%x6f%x6e%x73%x0b%x00%x00%x00%x77%
x65%x62%x61%x6e%x61%x6c%x79%x7a%x65%x72%x73%x05%x00%x00%x00%x62%x61%x74%x63%x68%x7b%x73%x06%x
00%x00%x00%x65%x71%x72%x65%x70%x72%x73%x8c%x00%x00%x00%x37%x66%x44%x2f%x78%x34%x53%x32%x32%x6
7%x39%x66%x73%x5a%x4d%x38%x4b%x62%x69%x33%x2b%x20%x68%x74%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e
%x66%x6f%x75%x72%x74%x68%x63%x6f%x66%x66%x65%x65%x2e%x63%x6f%x6d%x2f%x20%x31%x32%x34%x34%x32%
x30%x39%x33%x30%x32%x20%x68%x74%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e%x66%x6f%x75%x72%x74%x68%x
63%x6f%x66%x66%x65%x65%x2e%x63%x6f%x6d%x2f%x69%x6e%x64%x65%x78%x2e%x61%x73%x70%x78%x00%x68%x7
4%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e%x66%x6f%x75%x72%x74%x68%x63%x6f%x66%x66%x65%x65%x2e%x63
%x6f%x6d%x2f%x69%x6e%x64%x65%x78%x2e%x68%x74%x6d%x6c%x0a%x73%x06%x00%x00%x00%x75%x72%x69%x6d%
x61%x70%x73%x58%x01%x00%x00%x68%x74%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e%x66%x6f%x75%x72%x74%x
68%x63%x6f%x66%x66%x65%x65%x2e%x63%x6f%x6d%x2f%x20%x37%x66%x44%x2f%x78%x34%x53%x32%x32%x67%x3
9%x66%x73%x5a%x4d%x38%x4b%x62%x69%x33%x2b%x0a%x68%x74%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e%x66
```


Copyright © 2012 Microsoft Corporation.

```
%x6f%x75%x72%x74%x68%x63%x6f%x66%x66%x65%x65%x2e%x63%x6f%x6d%x2f%x61%x62%x6f%x75%x74%x2e%x68%
x74%x6d%x6c%x20%x46%x37%x33%x5a%x50%x6d%x6b%x76%x71%x35%x72%x65%x46%x67%x73%x53%x45%x61%x34%x
64%x41%x0a%x68%x74%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e%x63%x6f%x6e%x74%x6f%x73%x6f%x2e%x63%x6
f%x6d%x2f%x20%x7a%x63%x48%x71%x43%x2f%x35%x39%x34%x50%x33%x65%x63%x34%x6f%x48%x6d%x32%x68%x6f
%x53%x0a%x68%x74%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e%x66%x6f%x75%x72%x74%x68%x63%x6f%x66%x66%
x65%x65%x2e%x63%x6f%x6d%x2f%x69%x6e%x64%x65%x78%x2e%x61%x73%x70%x78%x20%x54%x4d%x6a%x37%x74%x
45%x46%x67%x72%x54%x47%x6c%x57%x45%x54%x71%x62%x59%x35%x4c%x35%x0a%x68%x74%x74%x70%x3a%x2f%x2
f%x77%x77%x77%x2e%x66%x6f%x75%x72%x74%x68%x63%x6f%x66%x66%x65%x65%x2e%x63%x6f%x6d%x2f%x69%x6e
%x64%x65%x78%x2e%x68%x74%x6d%x6c%x20%x35%x63%x39%x48%x73%x68%x71%x47%x52%x6f%x56%x56%x6d%x61%
x38%x6f%x58%x34%x44%x50%x7a%x0a%x68%x74%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e%x66%x6f%x75%x72%x
74%x68%x63%x6f%x66%x66%x65%x65%x2e%x63%x6f%x6d%x2f%x65%x78%x61%x6d%x70%x6c%x65%x31%x2e%x68%x7
4%x6d%x6c%x20%x77%x47%x4c%x45%x4b%x47%x75%x53%x55%x57%x63%x74%x4b%x4f%x35%x78%x7a%x7a%x4a%x37
%x4a%x0a%x73%x08%x00%x00%x00%x6e%x6f%x5f%x6c%x69%x6e%x6b%x73%x73%x4b%x00%x00%x00%x68%x74%x74%
x70%x3a%x2f%x2f%x77%x77%x77%x2e%x66%x6f%x75%x72%x74%x68%x63%x6f%x66%x66%x65%x65%x2e%x63%x6f%x
6d%x2f%x65%x78%x61%x6d%x70%x6c%x65%x31%x2e%x68%x74%x6d%x6c%x20%x77%x47%x4c%x45%x4b%x47%x75%x5
3%x55%x57%x63%x74%x4b%x4f%x35%x78%x7a%x7a%x4a%x37%x4a%x20%x31%x32%x34%x34%x32%x30%x39%x33%x30
%x32%x0a%x73%x05%x00%x00%x00%x6c%x69%x6e%x6b%x73%x73%x8b%x00%x00%x00%x37%x66%x44%x2f%x78%x34%
x53%x32%x32%x67%x39%x66%x73%x5a%x4d%x38%x4b%x62%x69%x33%x2b%x20%x46%x37%x33%x5a%x50%x6d%x6b%x
76%x71%x35%x72%x65%x46%x67%x73%x53%x45%x61%x34%x64%x41%x20%x31%x20%x31%x32%x34%x34%x32%x30%x3
9%x33%x30%x32%x20%x61%x62%x6f%x75%x74%x20%x75%x73%x0a%x37%x66%x44%x2f%x78%x34%x53%x32%x32%x67
%x39%x66%x73%x5a%x4d%x38%x4b%x62%x69%x33%x2b%x20%x7a%x63%x48%x71%x43%x2f%x35%x39%x34%x50%x33%
x65%x63%x34%x6f%x48%x6d%x32%x68%x6f%x53%x20%x30%x20%x31%x32%x34%x34%x32%x30%x39%x33%x30%x32%x
20%x61%x6e%x6f%x74%x68%x65%x72%x20%x65%x78%x61%x6d%x70%x6c%x65%x0a%x73%x05%x00%x00%x00%x75%x7
2%x69%x65%x71%x73%x6e%x00%x00%x00%x37%x66%x44%x2f%x78%x34%x53%x32%x32%x67%x39%x66%x73%x5a%x4d
%x38%x4b%x62%x69%x33%x2b%x20%x54%x4d%x6a%x37%x74%x45%x46%x67%x72%x54%x47%x6c%x57%x45%x54%x71%
x62%x59%x35%x4c%x35%x20%x31%x32%x34%x34%x32%x30%x39%x33%x30%x32%x0a%x37%x66%x44%x2f%x78%x34%x
53%x32%x32%x67%x39%x66%x73%x5a%x4d%x38%x4b%x62%x69%x33%x2b%x20%x35%x63%x39%x48%x73%x68%x71%x4
7%x52%x6f%x56%x56%x6d%x61%x38%x6f%x58%x34%x44%x50%x7a%x20%x31%x32%x34%x34%x32%x30%x39%x33%x30
%x32%x0a%x30%x73%x0a%x00%x00%x00%x63%x6f%x6c%x6c%x65%x63%x74%x69%x6f%x6e%x73%x02%x00%x00%x00%
x73%x70%x30
```

4.1.3 Send the Message

The protocol client concatenates the following terminator string to the message and sends it to the protocol server.

```
(%x2E%x2E%x2E%x00%x00%x00%x00)
```

The protocol server reads the data until it encounters the terminator, then unmarshals the data. If it parses the message, it sends the following string to the protocol client:

```
%x61%x63%x6b%x0A
```

If the protocol server cannot process the message, for example, it cannot write the message to disk, it returns the following string:

```
%x66%x61%x69%x6c%x65%x64%x0A
```

4.2 Send Data to SPRel

4.2.1 Parse the Metadata

In this example, the protocol client sends a message with metadata from two documents to the SPRel protocol server. The documents are described in the following table as *doc1* and *doc2.* The documents are processed at exactly 2010-01-01T00:00:00 UTC, and they are contained in the collection named "sp".

Copyright © 2012 Microsoft Corporation.

Key	Value
doc1	
document identifier	http://www.fourthcoffee.com/
Redirect	http://www.fourthcoffee.com/index.html and http://www.fourthcoffee.com/index.aspx
Operation	ADD
doc2	
document identifier	http://www.fourthcoffee.com/example1.html
Redirect	None
Operation	DEL

The protocol client creates the following lines based on *doc1*.

```
http://www.fourthcoffee.com/ 1244210475 ADD http://www.fourthcoffee.com/index.aspx%x0A
http://www.fourthcoffee.com/ 1244210475 ADD http://www.fourthcoffee.com/index.html%x0A
```

The protocol client creates the following lines based on *doc2*.

```
http://www.fourthcoffee.com/example1.html 1244210475 DEL %xc7%x82%x0A
```

4.2.2 Build the Message

The two entries are combined into a single message, as described in the following example.

```
{'application': 'sharepoint', 'batch': {'uris': 'http://www.fourthcoffee.com/ 1244210475 ADD
http://www.fourthcoffee.com/index.aspx%x0Ahttp://www.fourthcoffee.com/ 1244210475 ADD
http://www.fourthcoffee.com/index.html%x0Ahttp://www.fourthcoffee.com/example1.html
1244210475 DEL %xc7%x82%x0A'}, 'collection': 'sp'}
```

The serialized string that the protocol client sends to the protocol server resembles the following example:

```
%x7b%x73%x0b%x00%x00%x00%x61%x70%x70%x6c%x69%x63%x61%x74%x69%x6f%x6e%x73%x0a%x00%x00%x00%x73%
x68%x61%x72%x65%x70%x6f%x69%x6e%x74%x73%x05%x00%x00%x00%x62%x61%x74%x63%x68%x7b%x73%x04%x00%x
00%x00%x75%x72%x69%x73%x73%xf1%x00%x00%x00%x68%x74%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e%x66%x6
f%x75%x72%x74%x68%x63%x6f%x66%x66%x65%x65%x2e%x63%x6f%x6d%x2f%x20%x31%x32%x34%x34%x32%x31%x30
%x34%x37%x35%x20%x41%x44%x44%x20%x68%x74%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e%x66%x6f%x75%x72%
x74%x68%x63%x6f%x66%x66%x65%x65%x2e%x63%x6f%x6d%x2f%x69%x6e%x64%x65%x78%x2e%x61%x73%x70%x78%x
25%x78%x30%x41%x68%x74%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e%x66%x6f%x75%x72%x74%x68%x63%x6f%x6
6%x66%x65%x65%x2e%x63%x6f%x6d%x2f%x20%x31%x32%x34%x34%x32%x31%x30%x34%x37%x35%x20%x41%x44%x44
%x20%x68%x74%x74%x70%x3a%x2f%x2f%x77%x77%x77%x2e%x66%x6f%x75%x72%x74%x68%x63%x6f%x66%x66%x65%
x65%x2e%x63%x6f%x6d%x2f%x69%x6e%x64%x65%x78%x2e%x68%x74%x6d%x6c%x25%x78%x30%x41%x68%x74%x74%x
70%x3a%x2f%x2f%x77%x77%x77%x2e%x66%x6f%x75%x72%x74%x68%x63%x6f%x66%x66%x65%x65%x2e%x63%x6f%x6
d%x2f%x65%x78%x61%x6d%x70%x6c%x65%x31%x2e%x68%x74%x6d%x6c%x20%x31%x32%x34%x34%x32%x31%x30%x34
```


Copyright © 2012 Microsoft Corporation.

```
%x37%x35%x20%x44%x45%x4c%x20%x25%x78%x63%x37%x25%x78%x38%x32%x25%x78%x30%x41%x30%x73%x0a%x00%
x00%x00%x63%x6f%x6c%x6c%x65%x63%x74%x69%x6f%x6e%x73%x02%x00%x00%x00%x73%x70%x30
```

4.2.3 Send the Message

For information about sending messages, see section 4.1.3.

Copyright © 2012 Microsoft Corporation.

5 Security

5.1 Security Considerations for Implementers

None.

5.2 Index of Security Parameters

None.

Copyright © 2012 Microsoft Corporation.

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® FAST™ Search Server 2010

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.

Copyright © 2012 Microsoft Corporation.

7 Change Tracking

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

Copyright © 2012 Microsoft Corporation.

8 Index

A

Abstract data model
  client 14
  server 14
Applicability 6
Application entry 8

B

Batch entry 8

C

Capability negotiation 7
Change tracking 23
Client
  abstract data model 14
  higher-layer triggered events 14
  initialization 14
  message processing 14
  other local events 14
  overview 14
  sequencing rules 14
  timer events 14
  timers 14
Collection entry 8

D

Data message 8
Data model - abstract
  client 14
  server 14

E

Entries
  application 8
  batch 8
  collection 8

F

Fields - vendor-extensible 7

G

Glossary 5

H

Higher-layer triggered events
  client 14
  server 15

I

Implementer - security considerations 21
Index of security parameters 21
Informative references 6
Initialization
  client 14
  server 15
Introduction 5

M

Message processing
  client 14
  server 15
Messages
  application entry 8
  batch entry 8
  collection entry 8
  Data 8
  Response (section 2.2.2 13, section 2.2.2 13)
  transport 8

N

Normative references 5

O

Other local events
  client 14
  server 15
Overview (synopsis) 6

P

Parameters - security index 21
Preconditions 6
Prerequisites 6
Product behavior 22

R

References 5
  informative 6
  normative 5
Relationship to other protocols 6
Response message 13

S

Security
  implementer considerations 21
  parameter index 21
Sequencing rules
  client 14
  server 15
Server
  abstract data model 14
  higher-layer triggered events 15
  initialization 15
  message processing 15

Copyright © 2012 Microsoft Corporation.

other local events 15
overview 14
sequencing rules 15
timer events 15
timers 15
Standards assignments 7

T

Timer events
client 14
server 15
Timers
client 14
server 15
Tracking changes 23
Transport 8
Triggered events - higher-layer
client 14
server 15

V

Vendor-extensible fields 7
Versioning 7

Copyright © 2012 Microsoft Corporation.