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.
## Revision Summary

<table>
<thead>
<tr>
<th>Date</th>
<th>Revision History</th>
<th>Revision Class</th>
<th>Comments</th>
</tr>
</thead>
<tbody>
<tr>
<td>11/06/2009</td>
<td>0.1</td>
<td>Major</td>
<td>Initial Availability</td>
</tr>
<tr>
<td>02/19/2010</td>
<td>1.0</td>
<td>Major</td>
<td>Updated and revised the technical content</td>
</tr>
<tr>
<td>03/31/2010</td>
<td>1.01</td>
<td>Editorial</td>
<td>Revised and edited the technical content</td>
</tr>
<tr>
<td>04/30/2010</td>
<td>1.02</td>
<td>Editorial</td>
<td>Revised and edited the technical content</td>
</tr>
<tr>
<td>06/07/2010</td>
<td>1.03</td>
<td>Editorial</td>
<td>Revised and edited the technical content</td>
</tr>
<tr>
<td>06/29/2010</td>
<td>1.04</td>
<td>Editorial</td>
<td>Changed language and formatting in the technical content.</td>
</tr>
<tr>
<td>07/23/2010</td>
<td>1.04</td>
<td>No change</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>09/27/2010</td>
<td>1.04</td>
<td>No change</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>11/15/2010</td>
<td>1.04</td>
<td>No change</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>12/17/2010</td>
<td>1.04</td>
<td>No change</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>03/18/2011</td>
<td>1.04</td>
<td>No change</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>06/10/2011</td>
<td>1.04</td>
<td>No change</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>01/20/2012</td>
<td>1.5</td>
<td>Minor</td>
<td>Clarified the meaning of the technical content.</td>
</tr>
<tr>
<td>04/11/2012</td>
<td>1.5</td>
<td>No change</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
<tr>
<td>07/16/2012</td>
<td>1.5</td>
<td>No change</td>
<td>No changes to the meaning, language, or formatting of the technical content.</td>
</tr>
</tbody>
</table>
## 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 ..................................................................................... 5
   1.3 Protocol Overview (Synopsis) .................................................................................... 7
   1.4 Relationship to Other Protocols ............................................................................... 7
   1.5 Prerequisites/Preconditions ....................................................................................... 7
   1.6 Applicability Statement ............................................................................................. 8
   1.7 Versioning and Capability Negotiation ...................................................................... 8
   1.8 Vendor-Extensible Fields ......................................................................................... 8
   1.9 Standards Assignments ............................................................................................. 8

2 Messages ............................................................................................................................. 9
   2.1 Transport ..................................................................................................................... 9
   2.2 Common Message Syntax .......................................................................................... 9
      2.2.1 Namespaces .......................................................................................................... 9
      2.2.2 Messages ............................................................................................................. 10
      2.2.3 Elements ............................................................................................................. 10
      2.2.4 Complex Types ................................................................................................... 10
         2.2.4.1 ArrayOfLogMessageWCF ........................................................................... 10
         2.2.4.2 LogMessageWCF ...................................................................................... 10
      2.2.5 Simple Types ....................................................................................................... 11
      2.2.6 Attributes ........................................................................................................... 11
      2.2.7 Groups ................................................................................................................ 11
      2.2.8 Attribute Groups ............................................................................................... 11

3 Protocol Details .................................................................................................................. 12
   3.1 Protocol Server Details ............................................................................................. 12
      3.1.1 Abstract Data Model .......................................................................................... 12
      3.1.2 Timers ................................................................................................................ 12
      3.1.3 Initialization ........................................................................................................ 12
      3.1.4 Message Processing Events and Sequencing Rules ............................................ 12
         3.1.4.1 Log .............................................................................................................. 12
            3.1.4.1.1 Messages ............................................................................................... 13
            3.1.4.1.1.1 ILogServer_Log_InputMessage ....................................................... 13
            3.1.4.1.1.2 ILogServer_Log_OutputMessage ............................................... 13
            3.1.4.1.2 Elements .............................................................................................. 13
            3.1.4.1.2.1 Log ................................................................................................. 13
            3.1.4.1.2.2 LogResponse ................................................................................... 13
            3.1.4.1.3 Complex Types .................................................................................... 14
            3.1.4.1.4 Simple Types ....................................................................................... 14
            3.1.4.1.5 Attributes ............................................................................................. 14
            3.1.4.1.6 Groups ................................................................................................. 14
            3.1.4.1.7 Attribute Groups .................................................................................. 14
      3.1.5 Timer Events ......................................................................................................... 14
      3.1.6 Other Local Events ............................................................................................. 14

4 Protocol Examples .............................................................................................................. 15
   4.1 Log Message .............................................................................................................. 15
1 Introduction

The Logging Protocol enables a protocol client to send log messages to a protocol server.

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]:

Hypertext Transfer Protocol (HTTP)

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

collection
content
event
Simple Object Access Protocol (SOAP)
SOAP action
SOAP body
SOAP fault
Uniform Resource Locator (URL)
Web Services Description Language (WSDL)
Web site
WSDL message
WSDL operation
XML namespace

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.


1.2.2 Informative References


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


1.3 Protocol Overview (Synopsis)

This protocol enables a protocol client to send log messages to a protocol server. Each log message contains, among other information, a timestamp, a string message, the origin of the message, and the message severity.

A typical scenario for using this protocol is to have a single protocol server that aggregates log messages from multiple protocol clients in a distributed system. The protocol server typically sorts and persists incoming log messages according to timestamp and severity. By having a single protocol server, a system administrator can use the persisted messages to monitor all protocol clients from a central location. This is shown in the following diagram.

![Diagram of communication flow in the logging protocol]

Figure 1: Communication flow in the logging protocol

1.4 Relationship to Other Protocols

This protocol uses the SOAP messaging protocol for formatting requests and responses as described in [SOAP1.2/1] and [SOAP1.2/2]. It also uses SOAP message security with security contexts, as described in [WSSE 1.0] and [WSSC1.3]. The protocol transmits these messages by using the HTTP protocol, as described in [RFC2616].

The following diagram shows the underlying messaging and transport stack that this protocol uses.

![Diagram of this protocol in relation to other protocols]

Figure 2: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a Web site that is identified by a URL that is known by protocol clients. The protocol server endpoint is formed by appending ”/LogServer/service” to the URL of the site, for example http://www.contoso.com:3333/LogServer/service.
This protocol assumes that the underlying protocols have performed authentication.

1.6 Applicability Statement

This protocol is designed to monitor a distributed system where multiple protocol clients submit text log messages to a single protocol server. This protocol is not applicable for submission of any content other than text.

1.7 Versioning and Capability Negotiation

None.

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.
2 Messages

2.1 Transport

Protocol servers MUST support SOAP over HTTP. Protocol servers MUST additionally support SOAP message security and security context establishment, as specified in [WSSE 1.0] and [WSSC1.3]. Authentication MUST be performed through client side certificates.

Protocol messages are formatted as specified in [SOAP1.2/1], Section 5. Protocol server faults are returned using either HTTP Status Codes as specified in [RFC2616], or using SOAP faults as specified in [SOAP1.2/1], section 5.4.

2.2 Common Message Syntax

This section contains common definitions that are used by this protocol. The syntax of the definitions uses XML Schema as defined in [XMLSCHEMA1] and [XMLSCHEMA2], and WSDL as defined in [WSDL].

2.2.1 Namespaces

This protocol specifies and references XML namespaces using the mechanisms specified in [XMLNS]. Although this document associates an XML namespace prefix for each XML namespace that is used, the choice of any particular XML namespace prefix is implementation-specific. The following table describes these namespaces.

<table>
<thead>
<tr>
<th>Prefix</th>
<th>Namespace URI</th>
<th>Reference</th>
</tr>
</thead>
<tbody>
<tr>
<td>wsd1</td>
<td><a href="http://schemas.xmlsoap.org/wsdl/">http://schemas.xmlsoap.org/wsdl/</a></td>
<td>[WSDL]</td>
</tr>
<tr>
<td>wsa10</td>
<td><a href="http://www.w3.org/2005/08/addressing">http://www.w3.org/2005/08/addressing</a></td>
<td>[WSA1.0 SOAP Binding] [WSA1.0 Core]</td>
</tr>
<tr>
<td>soap12</td>
<td><a href="http://schemas.xmlsoap.org/wsdl/soap12/">http://schemas.xmlsoap.org/wsdl/soap12/</a></td>
<td>[SOAP1.2/1] [SOAP1.2/2]</td>
</tr>
<tr>
<td>wsu</td>
<td><a href="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd">http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd</a></td>
<td>[WSSE 1.0]</td>
</tr>
<tr>
<td>wsaw</td>
<td><a href="http://www.w3.org/2006/05/addressing/wsdl">http://www.w3.org/2006/05/addressing/wsdl</a></td>
<td>[WSA1.0]</td>
</tr>
<tr>
<td>sp</td>
<td><a href="http://schemas.xmlsoap.org/ws/2005/07/securitpolicy">http://schemas.xmlsoap.org/ws/2005/07/securitpolicy</a></td>
<td>[WSSP1.2]</td>
</tr>
<tr>
<td>xsd</td>
<td><a href="http://www.w3.org/2001/XMLSchema">http://www.w3.org/2001/XMLSchema</a></td>
<td>[XMLSCHEMA1] [XMLSCHEMA2]</td>
</tr>
</tbody>
</table>
2.2.2 Messages

None

2.2.3 Elements

This specification does not define any common XML Schema element definitions.

2.2.4 Complex Types

The following table summarizes the complex types that are specified in this document.

<table>
<thead>
<tr>
<th>Complex type</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>ArrayOfLogMessageWCF</td>
<td>An array of LogMessageWCF elements.</td>
</tr>
<tr>
<td>LogMessageWCF</td>
<td>Contains information about a log message.</td>
</tr>
</tbody>
</table>

2.2.4.1 ArrayOfLogMessageWCF

The ArrayOfLogMessageWCF complex type contains an array of LogMessageWCF elements, as follows.

```xml
<xsd:complexType name="ArrayOfLogMessageWCF">
  <xsd:sequence>
    <xsd:element minOccurs="0" maxOccurs="unbounded" name="LogMessageWCF" nillable="true" type="tns:LogMessageWCF"/>
  </xsd:sequence>
</xsd:complexType>
```

LogMessageWCF: Zero or more LogMessageWCF elements, as specified in section 2.2.4.2.

2.2.4.2 LogMessageWCF

The LogMessageWCF complex type provides information about the message to log, as follows.

```xml
<xsd:complexType name="LogMessageWCF">
  <xsd:sequence>
    <xsd:element minOccurs="0" name="Collection" nillable="true" type="xsd:string"/>
    <xsd:element minOccurs="0" name="Host" nillable="true" type="xsd:string"/>
    <xsd:element minOccurs="0" name="Level" type="xsd:int"/>
    <xsd:element minOccurs="0" name="Message" nillable="true" type="xsd:string"/>
    <xsd:element minOccurs="0" name="MessageId" type="xsd:int"/>
    <xsd:element minOccurs="0" name="Module" nillable="true" type="xsd:string"/>
    <xsd:element minOccurs="0" name="TimeStamp" type="xsd:dateTime"/>
  </xsd:sequence>
</xsd:complexType>
```

Collection: The name of a content collection. If a log message is not associated with any content collection, this element MUST contain the value "systemmsg".

Host: A server identifier and optional port number where the message originates.
**Level:** An integer whose values MUST be 1 through 6. Each number represents the relative importance of the message, as described in the following table.

<table>
<thead>
<tr>
<th>Value</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>CRITICAL</td>
</tr>
<tr>
<td>2</td>
<td>ERROR</td>
</tr>
<tr>
<td>3</td>
<td>WARNING</td>
</tr>
<tr>
<td>4</td>
<td>INFO</td>
</tr>
<tr>
<td>5</td>
<td>VERBOSE</td>
</tr>
<tr>
<td>6</td>
<td>DEBUG</td>
</tr>
</tbody>
</table>

**Message:** The message to log.

**MessageId:** Identifies log messages of a certain type. The protocol server specifies unique message identifiers that protocol clients can use. The message identifier MUST be a message identifier that was specified by the protocol server, or it MUST contain a value of −1, which specifies that this message is of no specific type. If the message identifier does not conform to these restrictions, then protocol server behavior is undefined.

**Module:** The application that created this message.

**TimeStamp:** A *DateTime* that represents the time of the event.

### 2.2.5 Simple Types

This specification does not define any common XML Schema simple type definitions.

### 2.2.6 Attributes

This specification does not define any common XML Schema attribute definitions.

### 2.2.7 Groups

This specification does not define any common XML Schema group definitions.

### 2.2.8 Attribute Groups

This specification does not define any common XML Schema attribute group definitions.
3 Protocol Details

The client side of this protocol is simply a pass-through. That is, no additional timers or other state is required on the client side of this protocol. Calls made by the higher-layer protocol or application are passed directly to the transport, and the results returned by the transport are passed directly back to the higher-layer protocol or application.

Except where specified, protocol clients MUST interpret the HTTP status codes, as specified in [RFC2616] Status Code Definitions section 10.

3.1 Protocol 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 implementations adhere to this model as long as their external behavior is consistent with that described in this document.

The protocol client sends a log message to the protocol server to persist a message, and the protocol server sends a return message to the protocol client. The protocol client determines whether the protocol server correctly processed the log message by verifying the return message, as specified in section 3.1.4.1.2.2.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

The following table summarizes the list of WSDL operations that are defined in this specification.

<table>
<thead>
<tr>
<th>Operation</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>Log</td>
<td>Logs one or more messages.</td>
</tr>
</tbody>
</table>

3.1.4.1 Log

This operation adds one or more messages to the log, as follows.

```xml
<wsdl:operation name="Log">
  <wsdl:input
  <wsdl:output
</wsdl:operation>
```
The client sends an `ILogServer_Log_InputMessage` request message, and the server responds with an `ILogServer_Log_OutputMessage` response message.

### 3.1.4.1.1 Messages

The following WSDL message definitions are specific to this operation.

#### 3.1.4.1.1.1 `ILogServer_Log_InputMessage`

This is the request to initiate the operation that logs messages.

The SOAP action value is as follows:


The SOAP body contains a `Log` element, as specified in section 3.1.4.1.2.1.

#### 3.1.4.1.1.2 `ILogServer_Log_OutputMessage`

This message represents the response associated with the Log WSDL operation.

The SOAP action value is as follows:


The SOAP body contains a `LogResponse` element, as specified in section 3.1.4.1.2.2.

### 3.1.4.1.2 Elements

The following XML Schema element definitions are specific to this operation.

#### 3.1.4.1.2.1 Log

This structure is contained in an `ILogServer_Log_InputMessage` message and contains the log messages that the protocol client is requesting to add, as follows.

```
<xsd:element name="Log">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" name="messages" nillable="true" type="tns:ArrayOfLogMessageWCF"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```

**Messages:** MUST be of type `ArrayOfLogMessageWCF`.

#### 3.1.4.1.2.2 LogResponse

```
<xsd:element name="LogResponse">
  <xsd:complexType>
    <xsd:sequence>
      <xsd:element minOccurs="0" name="messages" nillable="true" type="tns:ArrayOfLogMessageWCF"/>
    </xsd:sequence>
  </xsd:complexType>
</xsd:element>
```
LogResult: The string "ok" MUST be returned upon successful completion. The Log operation allows no other messages.

3.1.4.1.3 Complex Types
None.

3.1.4.1.4 Simple Types
None.

3.1.4.1.5 Attributes
None.

3.1.4.1.6 Groups
None.

3.1.4.1.7 Attribute Groups
None.

3.1.5 Timer Events
None.

3.1.6 Other Local Events
None.
4 Protocol Examples

As specified in section 1.4, this protocol uses SOAP message security. For clarity, all SOAP requests and responses are shown in clear text, without SOAP message security.

4.1 Log Message

The protocol client sends a **ILogServer_Log_InputMessage** request to log a message, as follows.

```xml
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing">
  <s:Header>
    <a:MessageID>urn:uuid:117ae174-0261-4d76-b346-e596672f4ef4</a:MessageID>
    <a:ReplyTo>
      <a:Address>http://www.w3.org/2005/08/addressing/anonymous</a:Address>
    </a:ReplyTo>
    <a:To s:mustUnderstand="1">...</a:To>
  </s:Header>
  <s:Body>
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
        <b:LogMessageWCF>
          <b:Collection>SampleCollection</b:Collection>
          <b:Host>716XQ3J</b:Host>
          <b:Level>3</b:Level>
          <b:Message>Sample log message text</b:Message>
          <b:MessageId>12345</b:MessageId>
          <b:Module>SampleModule</b:Module>
          <b:TimeStamp>2009-01-16T12:30:03.4422457+01:00</b:TimeStamp>
        </b:LogMessageWCF>
      </messages>
    </Log>
  </s:Body>
</s:Envelope>
```

The protocol server processes the log message and responds as follows:

```xml
<s:Envelope xmlns:s="http://www.w3.org/2003/05/soap-envelope"
xmlns:a="http://www.w3.org/2005/08/addressing">
  <s:Header>
    <a:RelatesTo>urn:uuid:117ae174-0261-4d76-b346-e596672f4ef4</a:RelatesTo>
  </s:Header>
  <s:Body>
      <LogResult>ok</LogResult>
    </LogResponse>
  </s:Body>
</s:Envelope>
```
</LogResponse>
</s:Body>
</s:Envelope>
5  Security

5.1  Security Considerations for Implementers

This protocol introduces no additional security considerations beyond those applicable to the underlying protocols.

5.2  Index of Security Parameters

None.
6 Appendix A: Full WSDL

For ease of implementation, the full WSDL is provided as follows:

```xml
<?xml version="1.0" encoding="utf-8"?>
<wsdl:definitions name="LogServer"
    xmlns:wsdl="http://schemas.xmlsoap.org/wsdl/"
    xmlns:wsa10="http://www.w3.org/2005/08/addressing"
    xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/
    xmlns:wsu="http://docs.oasis-open.org/wss/2004/01/oasis-200401-wss-wssecurity-utility-1.0.xsd"
    xmlns:wsaw="http://www.w3.org/2006/05/addressing/wsdl"
    xmlns:sp="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <wsp:Policy wsu:Id="LogServer.ServerILogServer_policy">
        <wsp:ExactlyOne>
            <wsp:All>
                <sp:SymmetricBinding>
                    <wsp:Policy>
                        <sp:ProtectionToken>
                            <wsp:Policy>
                                <sp:SecureConversationToken
                                    sp:IncludeToken="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy/IncludeToken/AlwaysToRecipient">
                                    <wsp:Policy>
                                        <sp:RequireDerivedKeys/>
                                        <sp:BootstrapPolicy>
                                            <wsp:Policy>
                                                <sp:SignedParts>
                                                    <sp:Body/>
                                                    <sp:Header Name="To" Namespace="http://www.w3.org/2005/08/addressing"/>
                                                    <sp:Header Name="From" Namespace="http://www.w3.org/2005/08/addressing"/>
                                                    <sp:Header Name="FaultTo" Namespace="http://www.w3.org/2005/08/addressing"/>
                                                    <sp:Header Name="ReplyTo" Namespace="http://www.w3.org/2005/08/addressing"/>
                                                    <sp:Header Name="MessageId" Namespace="http://www.w3.org/2005/08/addressing"/>
                                                    <sp:Header Name="RelatesTo" Namespace="http://www.w3.org/2005/08/addressing"/>
                                                    <sp:Header Name="Action" Namespace="http://www.w3.org/2005/08/addressing"/>
                                                </sp:SignedParts>
                                                <sp:EncryptedParts>
                                                    <sp:Body/>
                                                    <sp:SymmetricBinding>
                                                        <wsp:Policy>
                                                            <sp:ProtectionToken>
                                                                <wsp:Policy>
                                                                    ...
                                                                </wsp:Policy>
                                                            </sp:ProtectionToken>
                                                        </wsp:Policy>
                                                    </sp:SymmetricBinding>
                                                </sp:EncryptedParts>
                                            </wsp:Policy>
                                        </sp:BootstrapPolicy>
                                    </wsp:Policy>
                                </sp:SecureConversationToken>
                            </wsp:Policy>
                        </sp:ProtectionToken>
                    </wsp:Policy>
                </sp:SymmetricBinding>
            </wsp:All>
        </wsp:ExactlyOne>
    </wsp:Policy>
</wsdl:definitions>
```
<mssp:SslContextToken>
  <sp:IncludeToken="http://schemas.xmlsoap.org/ws/2005/07/securitypolicy/IncludeToken/AlwaysToRecipient">
    <wsp:Policy>
      <sp:RequireDerivedKeys/>
      <mssp:RequireClientCertificate/>
    </wsp:Policy>
  </mssp:SslContextToken>
  <wsp:Policy>
    <sp:AlgorithmSuite>
      <wsp:Policy>
        <sp:Basic256/>
      </wsp:Policy>
    </sp:AlgorithmSuite>
    <sp:IncludeTimestamp/>
    <sp:EncryptSignature/>
    <sp:OnlySignEntireHeadersAndBody/>
  </wsp:Policy>
</sp:ProtectionToken>
</sp:BootstrapPolicy>
</wsp:Policy>
</sp:SecureConversationToken>
</wsp:Policy>
</sp:ProtectionToken>
</sp:AlgorithmSuite>
</wsp:Policy>
</sp:AlgorithmSuite>
</wsp:Policy>
</sp:Strict/>
</wsp:Policy>
</sp:Layout>
</sp:IncludeTimestamp/>
</sp:EncryptSignature/>
</sp:OnlySignEntireHeadersAndBody/>
</wsp:Policy>
</sp:SymmetricBinding>
</sp:Wss11>
</sp:Policy>
</sp:MustSupportIssuedTokens/>
</sp:RequireClientEntropy/>
</sp:RequireServerEntropy/>
</wsp:Policy>
</sp:Trust10>
</sp:Wss11>
</sp:Policy>
</sp:MustSupportRefKeyIdentifier/>
</sp:MustSupportRefIssuerSerial/>
</sp:MustSupportRefThumbprint/>
</sp:MustSupportRefEncryptedKey/>
</wsp:Policy>
</sp:Wss11>
</sp:Policy>
</sp:MustSupportIssuedTokens/>
</sp:RequireClientEntropy/>
</sp:RequireServerEntropy/>
</wsp:Policy>
</sp:Trust10>
</sp:Wss11>
</sp:Policy>
</sp:SymmetricBinding>
</sp:Wss11>
</sp:AlgorithmSuite>
</wsp:Policy>
</sp:AlgorithmSuite>
</wsp:Policy>
</sp:IncludeTimestamp/>
</sp:EncryptSignature/>
</sp:OnlySignEntireHeadersAndBody/>
<wsp:Policy>
</wsp:SymmetricBinding>
<sp:Wss11>
</sp:Policy>
</sp:Wss11>
</wsp:Policy>
</sp:Trust10>
</wsp:ExactlyOne>
</wsp:Policy>
</wsdl:types>
<xsd:element name="Log">
<xsd:complexType>
<xsd:sequence>
<xsd:element minOccurs="0" name="messages" nillable="true" type="tns:ArrayOfLogMessageWCF"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:complexType name="ArrayOfLogMessageWCF">
<xsd:sequence>
<xsd:element minOccurs="0" maxOccurs="unbounded" name="LogMessageWCF" nillable="true" type="tns:LogMessageWCF"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:complexType name="ArrayOfLogMessageWCF" nillable="true" type="tns:ArrayOfLogMessageWCF"/>
<xsd:complexType name="LogMessageWCF">
<xsd:sequence>
<xsd:element minOccurs="0" name="Collection" nillable="true" type="xsd:string"/>
<xsd:element minOccurs="0" name="Host" nillable="true" type="xsd:string"/>
<xsd:element minOccurs="0" name="Level" type="xsd:int"/>
<xsd:element minOccurs="0" name="Message" nillable="true" type="xsd:string"/>
<xsd:element minOccurs="0" name="MessageId" type="xsd:int"/>
<xsd:element minOccurs="0" name="Module" nillable="true" type="xsd:string"/>
<xsd:element minOccurs="0" name="TimeStamp" type="xsd:dateTime"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
<xsd:element name="ArrayOfLogMessageWCF" nillable="true" type="tns:ArrayOfLogMessageWCF"/>
<xsd:element name="LogResponse">
<xsd:complexType>
</xsd:complexType>
</xsd:element>
</xsd:schema>

[MS-FSL] — v20120630
Logging Protocol Specification

Copyright © 2012 Microsoft Corporation.

Release: July 16, 2012
<xsd:sequence>
    <xsd:element minOccurs="0" name="LogResult" nillable="true" type="xsd:string"/>
</xsd:sequence>
</xsd:complexType>
</xsd:element>
</xsd:schema>
</wsdl:types>
<wsdl:message name="ILogServer_Log_InputMessage">
    <wsdl:part name="parameters" element="tns:Log"/>
</wsdl:message>
<wsdl:message name="ILogServer_Log_OutputMessage">
    <wsdl:part name="parameters" element="tns:LogResponse"/>
</wsdl:message>
<wsdl:portType name="ILogServer">
    <wsdl:operation name="Log">
    </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="LogServer.Server_ILogServer" type="tns:ILogServer">
    <soap12:binding transport="http://schemas.xmlsoap.org/soap/http"/>
    <wsdl:operation name="Log">
        <wsdl:input>
            <soap12:body use="literal"/>
        </wsdl:input>
        <wsdl:output>
            <soap12:body use="literal"/>
        </wsdl:output>
    </wsdl:operation>
</wsdl:binding>
</wsdl:definitions>
7 Appendix B: 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.
8 Change Tracking

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

A
Abstract data model
   server 12
Applicability 8
ArrayOfLogMessageWCF complex type 10
Attribute groups 11
Attributes 11

C
Capability negotiation 8
Change tracking 23
Client - overview 12
Complex types 10
   ArrayOfLogMessageWCF 10
   LogMessageWCF 10

D
Data model - abstract
   server 12

E
Events
   local - server 14
   timer - server 14
Examples
   Log message example 15
   overview 15

F
Fields - vendor-extensible 8
   Full WSDL 18

G
   Glossary 5
   Groups 11

I
   Implementer - security considerations 17
   Index of security parameters 17
   Informative references 6
Initialization
   server 12
   Introduction 5

L
Local events
   server 14
   Log message example 15
   LogMessageWCF complex type 10

M
Message processing
   server 12
Messages
   ArrayOfLogMessageWCF complex type 10
   attribute groups 11
   attributes 11
   complex types 10
   elements 10
   enumerated 10
   groups 11
   LogMessageWCF complex type 10
   namespaces 9
   simple types 11
   syntax 9
   transport 9

N
   Namespaces 9
   Normative references 5

O
Operations
   Log 12
   Overview (synopsis) 7

P
Parameters - security index 17
Preconditions 7
Prerequisites 7
Product behavior 22

R
References 5
   informative 6
   normative 5
Relationship to other protocols 7

S
Security
   implementer considerations 17
   parameter index 17
Sequencing rules
   server 12
Server
   abstract data model 12
   initialization 12
   local events 14
   Log operation 12
   message processing 12
   overview 12
   sequencing rules 12
   timer events 14
   timers 12
   Simple types 11
Standards assignments 8
Syntax
  messages - overview 9

T
  Timer events
    server 14
  Timers
    server 12
  Tracking changes 23
  Transport 9
Types
  complex 10
  simple 11

V
  Vendor-extensible fields 8
  Versioning 8

W
  WSDL 18