

[MS-FSDAP]: Forms Services Design and Activation Web Service 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's Open Specification Promise (available here: <http://www.microsoft.com/interop/osp>) or the Community Promise (available here: <http://www.microsoft.com/interop/cp/default.mspx>). 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 iplq@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

Date	Revision History	Revision Class	Comments
04/04/2008	0.01	Major	Initial Availability
06/27/2008	1.0	Minor	Revised and edited technical content
12/12/2008	1.01	Editorial	Revised and edited technical content
07/13/2009	1.02	Major	Revised and edited the technical content
08/28/2009	1.03	Major	Updated and revised the technical content
11/06/2009	1.04	Editorial	Revised and edited the technical content
02/19/2010	2.0	Major	Updated and revised 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	Major	Significantly changed 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.

Table of Contents

1 Introduction	6
1.1 Glossary	6
1.2 References.....	7
1.2.1 Normative References.....	7
1.2.2 Informative References	8
1.3 Protocol Overview (Synopsis)	8
1.4 Relationship to Other Protocols.....	9
1.5 Prerequisites/Preconditions	9
1.6 Applicability Statement.....	10
1.7 Versioning and Capability Negotiation.....	10
1.8 Vendor-Extensible Fields.....	10
1.9 Standards Assignments	10
2 Messages	11
2.1 Transport.....	11
2.2 Common Message Syntax	11
2.2.1 Namespaces	11
2.2.2 Messages	11
2.2.3 Elements.....	11
2.2.4 Complex Types	11
2.2.4.1 Categories	12
2.2.4.2 CategoryType.....	12
2.2.4.3 ComplexType Messages.....	13
2.2.4.4 DesignCheckerInformation.....	13
2.2.4.5 Message	13
2.2.4.6 SourceLocation.....	16
2.2.5 Simple Types	17
2.2.5.1 MessageType	17
2.2.5.2 Features	18
2.2.5.3 Category	19
2.2.6 Attributes.....	20
2.2.7 Groups.....	20
2.2.8 Attribute Groups	20
3 Protocol Details	21
3.1 Server Details	21
3.1.1 Abstract Data Model	21
3.1.2 Timers	21
3.1.3 Initialization	21
3.1.4 Message Processing Events and Sequencing Rules.....	21
3.1.4.1 BrowserEnableUserFormTemplate.....	22
3.1.4.1.1 Messages	23
3.1.4.1.1.1 BrowserEnableUserFormTemplateSoapIn.....	23
3.1.4.1.1.2 BrowserEnableUserFormTemplateSoapOut.....	23
3.1.4.1.2 Elements.....	23
3.1.4.1.2.1 BrowserEnableUserFormTemplate.....	23
3.1.4.1.2.2 BrowserEnableUserFormTemplateResponse	23
3.1.4.1.3 Complex Types	24
3.1.4.1.3.1 MessagesResponse	24
3.1.4.2 DesignCheckFormTemplate.....	24

3.1.4.2.1	Messages	25
3.1.4.2.1.1	DesignCheckFormTemplateSoapIn	25
3.1.4.2.1.2	DesignCheckFormTemplateSoapOut	25
3.1.4.2.2	Elements	25
3.1.4.2.2.1	DesignCheckFormTemplate	25
3.1.4.2.2.2	DesignCheckFormTemplateResponse	26
3.1.4.2.3	Complex Types	26
3.1.4.2.3.1	DesignCheckerInformation	26
3.1.4.3	GetListFormLocation	26
3.1.4.3.1	Messages	27
3.1.4.3.1.1	GetListFormLocationSoapIn	27
3.1.4.3.1.2	GetListFormLocationSoapOut	27
3.1.4.3.2	Elements	28
3.1.4.3.2.1	GetListFormLocation	28
3.1.4.3.2.2	GetListFormLocationResponse	28
3.1.4.4	GetUserCodeDeploymentDependencies	29
3.1.4.4.1	Messages	29
3.1.4.4.1.1	GetUserCodeDeploymentDependenciesSoapIn	29
3.1.4.4.1.2	GetUserCodeDeploymentDependenciesSoapOut	30
3.1.4.4.2	Elements	30
3.1.4.4.2.1	GetUserCodeDeploymentDependencies	30
3.1.4.4.2.2	GetUserCodeDeploymentDependenciesResponse	30
3.1.4.4.3	Simple Types	30
3.1.4.4.3.1	UserSolutionActivationStatus	30
3.1.4.5	SetFormsForListItem	31
3.1.4.5.1	Messages	32
3.1.4.5.1.1	SetFormsForListItemSoapIn	32
3.1.4.5.1.2	SetFormsForListItemSoapOut	32
3.1.4.5.2	Elements	32
3.1.4.5.2.1	SetFormsForListItem	32
3.1.4.5.2.2	SetFormsForListItemResponse	33
3.1.4.5.3	Complex Types	34
3.1.4.5.3.1	DesignCheckerInformation	34
3.1.4.6	SetSchemaChangesForList	34
3.1.4.6.1	Messages	35
3.1.4.6.1.1	SetSchemaChangesForListSoapIn	35
3.1.4.6.1.2	SetSchemaChangesForListSoapOut	35
3.1.4.6.2	Elements	35
3.1.4.6.2.1	SetSchemaChangesForList	35
3.1.4.6.2.2	SetSchemaChangesForListResponse	36
3.1.4.6.3	Complex Types	37
3.1.4.6.3.1	NewFieldsNameMapping	37
3.1.4.6.3.2	NewFieldNameMapping	37
3.1.5	Timer Events	38
3.1.6	Other Local Events	38
4	Protocol Examples	39
4.1	DesignCheckFormTemplate Operation Examples	39
4.1.1	DesignCheckFormTemplate Request and Response with No Message Elements (No Issues Found)	39
4.1.2	DesignCheckFormTemplate Response with Message Elements (One or More Issues Found)	40
4.2	BrowserEnableUserFormTemplate Operation Examples	41

4.2.1	BrowserEnableUserFormTemplate Request/Response Indicating the Successful Browser Enabling of a Form Template (.xsn) File	41
4.3	SetFormsForListItem Operation Examples	42
4.3.1	SetFormsForListItem Request/Response Indicating Successful Operations on a List	42
4.3.2	SetFormsForListItem Response with Message Elements (One or More Issues Found)	43
4.4	GetListFormLocation Operation Examples	44
4.4.1	GetListFormLocation Request/Response	44
4.5	SetSchemaChangesForList Operation Examples	45
4.5.1	SetSchemaChangesForList Request/Response Indicating Successful Operations on a List	45
4.6	GetUserCodeDeploymentDependencies Operation Examples	46
4.6.1	GetUserCodeDeploymentDependencies Request/Response Indicating that the Form Template (.xsn) File with Code Can Be Browser-Enabled as a Sandboxed Solution	47
5	Security	48
5.1	Security Considerations for Implementers	48
5.2	Index of Security Parameters	48
6	Appendix A: Full WSDL	49
7	Appendix B: Product Messages	57
7.1	Messages for Office InfoPath 2007 Forms	57
7.1.1	Message Elements of Type "Error"	57
7.1.2	Message Elements of Type "Warning"	75
7.1.3	Message Elements of Type "Information"	80
7.2	Messages for InfoPath 2010 Forms	81
7.2.1	Message Elements of Type "Error"	81
7.2.2	Message Elements of Type "Warning"	101
7.2.3	Message Elements of Type "Information"	107
8	Appendix C: Product Behavior	109
9	Change Tracking	111
10	Index	112

1 Introduction

This document specifies the Forms Services Design and Activation Web Service Protocol. This protocol enables a protocol client to request that a protocol server do the following:

- Transform a client-provided electronic form into a format that can be served from the protocol server and rendered in a web browser.
- Verify whether the protocol server can successfully execute such a transformation on a client-provided electronic form.
- Set and update the location of the electronic form on the protocol server and the relationship of that electronic form to the data structure of the protocol server.

1.1 Glossary

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

base64
Hypertext Transfer Protocol (HTTP)
Hypertext Transfer Protocol over Secure Sockets Layer (HTTPS)
language code identifier (LCID)
Unicode
Universal Naming Convention (UNC)

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

assembly
bind
browser compatibility
browser-enable
browser-optimization issue
business logic
CAML
conditional formatting
content type
content type identifier
control
data adapter
data validation
design check
digital signature
document library
field
field internal name
form definition (.xsf) file
form template (.xsn) file
form view
Internationalized Resource Identifier (IRI)
list
list identifier
list schema
localization
rule
sandboxed solution

site
site content type
SOAP (Simple Object Access Protocol)
SOAP action
SOAP body
SOAP fault
SOAP message
Status-Code
submit
URL (Uniform Resource Locator)
user code
Web service
WSDL operation
XML fragment
XML namespace
XML Schema
XPath expression

The following terms are specific to this document:

source files: A collection of files that are used to implement an InfoPath form. File types can include HTML, XML, XSD, XSLT, and script.

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

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.

[CSS-LEVEL2] Bos, B., Celik, T., Hickson, I., and Lie, H., "Cascading Style Sheets Level 2 Revision 1 (CSS2.1) Specification: W3C Candidate Recommendation", July 2007, <http://www.w3.org/TR/2007/CR-CSS21-20070719/>

[HTML] World Wide Web Consortium, "HTML 4.01 Specification", December 1999, <http://www.w3.org/TR/html4/>

[MS-IPFF] Microsoft Corporation, "[InfoPath Form Template Format](#)", June 2008.

[MS-IPFF2] Microsoft Corporation, "[InfoPath Form Template Format Version 2](#)", July 2009.

[MS-LCID] Microsoft Corporation, "[Windows Language Code Identifier \(LCID\) Reference](#)", July 2007.

[MS-LISTSWS] Microsoft Corporation, "[Lists Web Service Protocol Specification](#)", June 2008.

[MS-WSSFO2] Microsoft Corporation, "[Windows SharePoint Services \(WSS\): File Operations Database Communications Version 2 Protocol Specification](#)", July 2009.

[MS-WSSTS] Microsoft Corporation, "[Windows SharePoint Services Technical Specification](#)", June 2008.

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

[RFC2396] Berners-Lee, T., Fielding, R., and Masinter, L., "Uniform Resource Identifiers (URI): Generic Syntax", RFC 2396, August 1998, <http://www.ietf.org/rfc/rfc2396.txt>

[RFC2616] Fielding, R., Gettys, J., Mogul, J., et al., "Hypertext Transfer Protocol -- HTTP/1.1", RFC 2616, June 1999, <http://www.ietf.org/rfc/rfc2616.txt>

[SOAP1.1] Box, D., Ehnebuske, D., Kakivaya, G., et al., "Simple Object Access Protocol (SOAP) 1.1", May 2000, <http://www.w3.org/TR/2000/NOTE-SOAP-20000508/>

[SOAP1.2/1] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 1: Messaging Framework", W3C Recommendation, June 2003, <http://www.w3.org/TR/2003/REC-soap12-part1-20030624>

[SOAP1.2/2] Gudgin, M., Hadley, M., Mendelsohn, N., Moreau, J., and Nielsen, H.F., "SOAP Version 1.2 Part 2: Adjuncts", W3C Recommendation, June 2003, <http://www.w3.org/TR/2003/REC-soap12-part2-20030624>

[W3C-XSLT] Clark, J., Ed., "XSL Transformations (XSLT) Version 1.0", W3C Recommendation, November 1999, <http://www.w3.org/TR/xslt>

[WSDL] Christensen, E., Curbera, F., Meredith, G., and Weerawarana, S., "Web Services Description Language (WSDL) 1.1", W3C Note, March 2001, <http://www.w3.org/TR/2001/NOTE-wsdl-20010315>

[XMLNS] World Wide Web Consortium, "Namespaces in XML 1.0 (Third Edition)", W3C Recommendation 8 December 2009, <http://www.w3.org/TR/REC-xml-names/>

[XMLSCHEMA1] Thompson, H.S., Ed., Beech, D., Ed., Maloney, M., Ed., and Mendelsohn, N., Ed., "XML Schema Part 1: Structures", W3C Recommendation, May 2001, <http://www.w3.org/TR/2001/REC-xmlschema-1-20010502/>

[XMLSCHEMA2] Biron, P.V., Ed. and Malhotra, A., Ed., "XML Schema Part 2: Datatypes", W3C Recommendation, May 2001, <http://www.w3.org/TR/2001/REC-xmlschema-2-20010502/>

[XPath] Clark, J. and DeRose, S., "XML Path Language (XPath), Version 1.0", W3C Recommendation, November 1999, <http://www.w3.org/TR/xpath>

1.2.2 Informative References

[MS-FSFD] Microsoft Corporation, "[Forms Services Feature Detection Protocol Specification](#)", June 2008.

[MS-GLOS] Microsoft Corporation, "[Windows Protocols Master Glossary](#)", March 2007.

[MS-OFCGLOS] Microsoft Corporation, "[Microsoft Office Master Glossary](#)", June 2008.

[RFC2818] Rescorla, E., "HTTP Over TLS", RFC 2818, May 2000, <http://www.ietf.org/rfc/rfc2818.txt>

1.3 Protocol Overview (Synopsis)

Users of a protocol server often interact with sets of data – adding and modifying data in any given set. A **form template (.xsn) file** can be used as an interface to a dataset if the form template

(.xsn) file can be rendered by the protocol server and if the correct relationships are established between the form template (.xsn) file and the appropriate data structure on the protocol server.

This protocol defines several methods by which a protocol client can request a protocol server to **browser-enable** or **design check** a specified form template (.xsn) file. There are several different scenarios in which different combinations of the web methods defined in this protocol can be used but these methods can also be used independently. Scenarios involving the methods defined in this protocol include:

- Verifying that a form template (.xsn) file can be successfully browser-enabled by a protocol server without a file transfer, as a preemptive check, and sending back to the protocol client a list of any issues blocking browser-enabling.
- Transferring a form template (.xsn) file to a protocol server and browser-enabling it.
- Mapping a form template (.xsn) file to a **content type** on a protocol server and maintaining both of these items if changes are made to either one.
- Verifying whether a form template (.xsn) file that contains user-authored code can be uploaded to a protocol server, be browser-enabled, and successfully execute that user code.

1.4 Relationship to Other Protocols

This protocol uses the **SOAP** message protocol for formatting request and response messages, as described in [\[SOAP1.1\]](#), [\[SOAP1.2/1\]](#) and [\[SOAP1.2/2\]](#). It transmits those messages by using **HTTP**, as described in [\[RFC2616\]](#), or **HTTPS**, as described in [\[RFC2818\]](#).

A protocol client can use Form Server Feature Detection protocol ([\[MS-FSFDPI\]](#)) to determine whether a protocol server implements the Forms Service Design and Activation protocol and also whether a protocol server implements Version 1 or Version 2 of the Forms Service Design and Activation Protocol (see [\[MS-FSFDPI\]](#) section 1.3).

The following diagram shows the underlying messaging and transport stack used by the protocol:

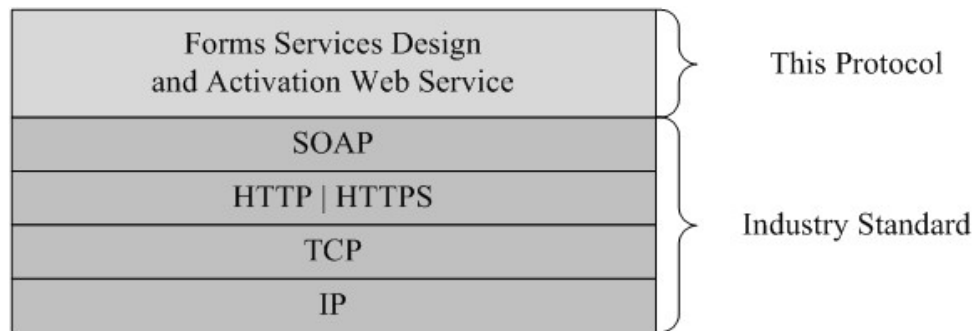


Figure 1: This protocol in relation to other protocols

1.5 Prerequisites/Preconditions

This protocol operates against a site that is identified by a URL that is known by protocol clients. The protocol server endpoint is formed by appending "_vti_bin/FormsServices.asmx" to the URL of the site; for example, http://www.contoso.com/Repository/_vti_bin/FormsServices.asmx. This protocol assumes that authentication has been performed by the underlying protocols.

1.6 Applicability Statement

None.

1.7 Versioning and Capability Negotiation

This document covers versioning issues in the following areas:

File Format: This protocol refers to different file format specifications, [\[MS-IPFF\]](#) and [\[MS-IPFF2\]](#), both of which define the structure of a valid form template (.xsn) file. In cases where both specifications are cited as references, the **SolutionFormatVersion** attribute of the **xDocumentClass** element, as described in [\[MS-IPFF2\]](#) section 2.2.1.2.1, specifies whether to use the InfoPath Form Template Format Structure, as described in [\[MS-IPFF\]](#), or the InfoPath Form Template Format Version 2 Structure, as described in [\[MS-IPFF2\]](#).

Supported Transports: This protocol uses multiple transports with SOAP as specified in section [2.1](#), Transport.

Localization: This protocol includes text strings in various messages. **Localization** considerations for such strings are specified in section [2.2.4.5](#).

1.8 Vendor-Extensible Fields

None.

1.9 Standards Assignments

None.

2 Messages

2.1 Transport

Protocol messages MUST be formatted as specified either in [\[SOAP1.1\]](#) section 4 (SOAP Envelope), or in [\[SOAP1.2/1\]](#) section 5 (SOAP Message Construct). Protocol server faults MUST be returned either using HTTP Status Codes as specified in [\[RFC2616\]](#) section 10 (Status Code Definitions), or using **SOAP faults** as specified either in [\[SOAP1.1\]](#) section 4.4 (**SOAP Fault**), or in [\[SOAP1.2/1\]](#) section 5.4 (**SOAP Fault**).

2.2 Common Message Syntax

This section contains common definitions used by this protocol. The syntax of the definitions uses **XML Schema (1)** as defined in [\[XMLSCHEMA1\]](#) and [\[XMLSCHEMA2\]](#) and Web Services Description Language as defined in [\[WSDL\]](#).

2.2.1 Namespaces

This specification defines and references various **XML namespaces** using the mechanisms specified in [\[XMLNS\]](#). Although this specification associates a specific XML namespace prefix for each XML namespace that is used, the choice of any particular XML namespace prefix is implementation-specific and not significant for interoperability.

Prefix	Namespace URI	Reference
wSDL	http://schemas.xmlsoap.org/wSDL/	[WSDL]
s	http://www.w3.org/2001/XMLSchema	[XMLSCHEMA1] [XMLSCHEMA2]
soap12	http://schemas.xmlsoap.org/wSDL/soap12/	[SOAP1.2/1] [SOAP1.2/2]
tns	http://schemas.microsoft.com/office/infopath/2007/formsServices	
soap	http://schemas.xmlsoap.org/wSDL/soap/	[SOAP1.1]

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 set of common XML Schema complex types defined by this specification. XML Schema complex type definitions that are specific to a particular **WSDL operation** are described with the WSDL operation.

Complex Type	Description
Categories	Contains a set of CategoryType elements.

Complex Type	Description
CategoryType	Contains suggestions for how to display the information contained within Message elements.
Messages	Contains a set of Message elements.
Message	Contains information (such as strings for display in a protocol client) about an issue found during the analysis of a form template during the design check or browser-enable process.
SourceLocation	Details the location within the form template source files of an issue represented by an associated Message element.

2.2.4.1 Categories

This element is a container for **Category** elements returned by the protocol server to the protocol client after design checking the form template (.xsn) file.

```
<s:complexType name="Categories">
  <s:sequence>
    <s:element name="Category" type="tns:CategoryType" nillable="true" minOccurs="1"
maxOccurs="unbounded"/>
  </s:sequence>
</s:complexType>
```

Category: Specified in section [2.2.4.2](#), **CategoryType**.

2.2.4.2 CategoryType

This element provides suggestions for the protocol client on how to display the information contained within **Message** elements returned by the protocol server after the design check operation has finished. [<1>](#)

```
<s:complexType name="CategoryType">
  <s:sequence>
    <s:element name="Id" type="tns:Category" minOccurs="1" maxOccurs="1"/>
    <s:element name="Label" type="s:string" minOccurs="1" maxOccurs="1"/>
    <s:element name="HideWarningsByDefault" type="s:boolean" minOccurs="1" maxOccurs="1"/>
  </s:sequence>
</s:complexType>
```

Id: Specified in section [2.2.5.3](#), **Category**.

Label: Specifies a **Unicode** string describing the **Category** element, as specified in section [2.2.5.3](#), **Category**, appropriate for display in the protocol client. For localization considerations see section [2.2.4.5](#), **Message**.

HideWarningsByDefault: Specifies a Boolean value indicating whether **Message** elements with an associated **MessageType** of "Warning" are to be displayed on the protocol client application by default.

2.2.4.3 ComplexType Messages

This is the container for **Message** elements as specified in section [2.2.4.5, Message](#), returned by the protocol server after attempting to browser-enable or design check the form template.

```
<s:complexType name="Messages">
  <s:sequence>
    <s:element name="Message" type="tns:Message" nillable="true" minOccurs="0"
maxOccurs="unbounded"/>
  </s:sequence>
</s:complexType>
```

Message: Specified in section [2.2.4.5, Message](#).

2.2.4.4 DesignCheckerInformation

This element specifies the **DesignCheckerInformation** complex element. This element is used by the protocol server to return data to the protocol client.

```
<s:complexType name="DesignCheckerInformation">
  <s:sequence>
    <s:element name="ApplicationId" type="s:string" minOccurs="1" maxOccurs="1"/>
    <s:element name="Lcid" type="s:int" minOccurs="1" maxOccurs="1"/>
    <s:element name="Categories" type="tns:Categories" minOccurs="1" maxOccurs="1"/>
    <s:element name="Messages" type="tns:Messages" minOccurs="1" maxOccurs="1"/>
  </s:sequence>
</s:complexType>
```

ApplicationId: Specifies the Unicode string that identifies the protocol client that initiated the request. This Unicode string **MUST** be identical to the string sent by the protocol client in the **ApplicationId** element.

Lcid: Specifies a **language code identifier (LCID)** (as specified in [\[MS-LCID\]](#)) of the preferred language for any messages returned by the protocol server (specified in section [2.2.4.2, CategoryType](#)).

Categories: Specified in section [2.2.4.1, Categories](#).

Messages: Specified in section [2.2.4.3, Messages](#).

2.2.4.5 Message

This element indicates a specific issue the protocol server found within a form template during the browser-enable or design check processes.

```
<s:complexType name="Message">
  <s:sequence>
    <s:element name="ShortMessage" type="s:string" minOccurs="1" maxOccurs="1"/>
    <s:element name="DetailedMessage" type="s:string" minOccurs="0" maxOccurs="1"/>
    <s:element name="SourceLocation" type="tns:SourceLocation" minOccurs="0"
maxOccurs="1"/>
  </s:sequence>
  <s:attribute name="Id" type="s:int" use="required"/>
  <s:attribute name="Type" type="tns:MessageType" use="required"/>
  <s:attribute name="Feature" type="tns:Feature" use="required"/>
```

```
<s:attribute name="Category" type="tns:Category" use="required"/>
</s:complexType>
```

ShortMessage: A Unicode string suitable for display in the protocol client describing the issue. This element MUST have a value populated by the protocol server.

DetailedMessage: A Unicode string suitable for display in the protocol client describing the issue.

Any localized Unicode strings returned by the protocol server SHOULD be in the language requested by the protocol client in the associated **Web service** call. If the protocol server is unable to provide strings in the requested language, then the protocol server MAY [<2>](#) use any available language.

SourceLocation: Specified in section [2.2.4.6](#), SourceLocation. **Id:** Specifies an integer that identifies the type of **Message** element.

Type: Specified in section [2.2.5.1](#), MessageType.

Feature: Specified in section [2.2.5.2](#), Features.

Category: Specified in section [2.2.5.3](#), Category.

The following table specifies a set of reserved **Message** elements. For the check specified in the Description column, a protocol server MUST construct a Message element matching the details in the table row, in conjunction with the table specified in section 2.2.4.6, SourceLocation. A protocol server MUST NOT redefine the Id attribute for any **Message** element in the following table [<3>](#).

For the reserved elements in the following table, **ShortMessage** and **DetailedMessage** are overwritten by the protocol client.

The **SolutionFormatVersion** attribute of the **xDocumentClass** element, as described in [\[MS-IPFF2\]](#) section 2.2.1.2.1, specifies whether the InfoPath Form Template Format Structure, as described in [\[MS-IPFF\]](#), or the InfoPath Form Template Format Version 2 Structure, as described in [\[MS-IPFF2\]](#), MUST be used when both versions of those structures are cited.

Id	Category	Feature	Type	Description
44	Browser-Compatibility	Controls	Error	This message MUST be generated when the form definition (.xsf) file contains an invalid signatureLocation element, as specified in [MS-IPFF] section 2.2.125 and [MS-IPFF2] section 2.2.1.2.106.
65	Browser-Compatibility	Controls	Error	This message MUST be generated when the form definition (.xsf) file contains an invalid xmlToEdit element, as specified in [MS-IPFF] section 2.2.124 and [MS-IPFF2] section 2.2.1.2.105.
70	Browser-Compatibility	Controls	Error	This message MUST be generated when the form view file contains an invalid construct, as specified in [MS-IPFF] section 2.4.1.23 and [MS-IPFF2] section 2.4.1.24.
101	Browser-Compatibility	Controls	Warning	This message MUST be generated when the form definition (.xsf) file contains an invalid externalViews element, as specified in [MS-IPFF] section 2.2.103 and [MS-IPFF2] section 2.2.1.2.84.

Id	Category	Feature	Type	Description
104	Browser-Compatibility	Controls	Error	This message MUST be generated when the form definition (.xsf) file contains an unsupported roles element, as specified in [MS-IPFF] section 2.2.81 and [MS-IPFF2] section 2.2.1.2.62.
105	Browser-Compatibility	Controls	Error	This message MUST be generated when the form definition (.xsf) file contains an unsupported save element as specified in [MS-IPFF] section 2.2.80 and [MS-IPFF2] section 2.2.1.2.61.
106	Browser-Compatibility	Controls	Error	This message MUST be generated when the form definition (.xsf) file contains an unsupported scripts element as specified in [MS-IPFF] section 2.2.33 and [MS-IPFF2] section 2.2.1.2.14.
107	Browser-Compatibility	Controls	Warning	This message MUST be generated when the form definition (.xsf) file contains an unsupported taskpane element as specified in [MS-IPFF] section 2.2.121 and [MS-IPFF2] section 2.2.1.2.102.
108	Browser-Compatibility	Controls	Error	This message MUST be generated when the form definition (.xsf) file contains an unsupported importParameters element that contains a useScriptHandler attribute set to yes as specified in [MS-IPFF] section 2.2.67 and [MS-IPFF2] section 2.2.1.2.58.
146	Browser-Compatibility	Controls	Warning	This message MUST be generated when the form definition (.xsf) file contains an invalid errorMessage element as specified in [MS-IPFF] section 2.2.64 and [MS-IPFF2] section 2.2.1.2.45, type attribute.
154	Browser-Compatibility	Controls	Error	This message MUST be generated by the protocol server when the form definition (.xsf) file contains an invalid adoAdapter element as specified in [MS-IPFF] section 2.2.38 and [MS-IPFF2] section 2.2.1.2.19.
188	Browser-Compatibility	Controls	Error	This message MUST be generated when either the form template (.xsn) file contains an unsupported dialogBoxExpressionAction element as specified in [MS-IPFF] section 2.2.137 and [MS-IPFF2] section 2.2.1.2.118, or when the form template (.xsn) file contains an unsupported dialogBoxMessageAction element as specified in [MS-IPFF] section 2.2.136 and [MS-IPFF2] section 2.2.1.2.117.
190	Browser-Compatibility	Controls	Warning	This message MUST be generated when the form definition (.xsf) file contains an invalid closeDocumentAction element as specified in [MS-IPFF] sections 2.2.133 and 2.2.142 and [MS-IPFF2] sections 2.2.1.2.114 and 2.2.1.2.124 .
282	Browser-Compatibility	Controls	Error	This message MUST be generated when the form definition (.xsf) file contains a dataObject element, as specified in [MS-IPFF2] section 2.2.1.2.18, that references a bdcAdapter element, as specified in [MS-IPFF2] section 2.2.1.2.128, along with additional references in the dataObject elements.

Id	Category	Feature	Type	Description
283	Browser-Compatibility	Controls	Error	This message MUST be generated when the form definition (.xsf) file contains a Signature Line Control as specified in [MS-IPFF2] section 2.2.3.2.12.
284	Browser-Compatibility	DataAdapters	Error	This message MUST be generated when a form definition (.xsf) file contains a dataObject element, [MS-IPFF2] section 2.2.1.2.17, that references a UNC path or local path.
285	Browser-Compatibility	Controls	Warning	This message MUST be generated when the form definition (.xsf) file contains a signSignatureLineAction element, as specified in [MS-IPFF2] section 2.2.1.2.134.
294	Browser-Compatibility	DataAdapters	Error	This message MUST be generated when a form definition (.xsf) file contains a dataObject element, as specified in [MS-IPFF2] section 2.2.1.2.17, that references a UNC path or local path.

2.2.4.6 SourceLocation

This element indicates the location of an issue within the form template source files that is responsible for generating the corresponding **Message** element as specified in section [2.2.4.5](#), **Message**.

```
<s:complexType name="SourceLocation">
  <s:attribute name="ControlId" type="s:string"/>
  <s:attribute name="FileName" type="s:string"/>
  <s:attribute name="LineNumber" type="s:int"/>
  <s:attribute name="LinePosition" type="s:int"/>
</s:complexType>
```

ControlId: Specifies the Unicode string that identifies the **control** causing the associated **Message** element. This value MUST be as specified in [\[MS-IPFF\]](#) and [\[MS-IPFF2\]](#) section 2.4.

FileName: Specifies the Unicode string file name of the file within the form template source files containing the issue that caused the associated **Message** element.

LineNumber: Specifies the integer that represents the line within the file where the issue responsible for generating the corresponding **Message** element exists.

LinePosition: Specifies the integer column that represents the column within the file where the issue responsible for generating the corresponding **Message** element begins.

The following table specifies when the **ControlId**, **FileName**, **LineNumber** and **LinePosition** attributes MUST be populated when **Message** elements matching those specified in section 2.2.4.5, **Message**, are constructed by the protocol server. In this table, True indicates that the attribute MUST be set and False indicates that the attribute MUST NOT be set.

Id	ControlId	FileName	LineNumber	LinePosition
44	False	True	True	False

Id	ControlId	FileName	LineNumber	LinePosition
65	False	True	True	True
70	False	True	True	True
101	False	True	False	False
104	False	True	False	False
105	False	True	False	False
106	False	True	False	False
107	False	True	False	False
108	False	True	False	False
146	False	True	False	False
154	False	True	False	False
188	False	True	False	False
190	False	True	False	False
282	False	True	False	False
283	True	True	True	False
284	True	True	True	False
285	False	True	False	False
294	False	False	False	False

2.2.5 Simple Types

The following table summarizes the set of common XML Schema simple type definitions defined by this specification. XML Schema simple type definitions that are specific to a particular WSDL operation are described with the WSDL operation.

Simple Type	Description
MessageType	Specifies the possible error levels of a Message element.
Feature	Specifies the possible form template features associated with a Message element.
Category	Specifies two types for Message elements, categorizing them into browser-optimization issues or browser compatibility messages.

2.2.5.1 MessageType

This element indicates whether the protocol server encountered an error, warning, or is simply relaying informative text while browser-enabling or design checking the form template (.xsn) file.

```
<s:simpleType name="MessageType">
  <s:restriction base="s:string">
```

```

    <s:enumeration value="Error"/>
    <s:enumeration value="Information"/>
    <s:enumeration value="Warning"/>
  </s:restriction>
</s:simpleType>

```

The following table specifies **MessageType** values and their proper use:

Value	Meaning
Error	The protocol server MUST use this value for an issue that prevents a form template from being successfully browser-enabled.
Warning	The protocol server MUST use this value to specify an issue that could degrade end user experience. The issue MUST NOT prevent a form template from being successfully browser-enabled.
Information	The protocol server MUST use this value to specify informative text for the protocol client. The issue MUST NOT prevent a form template from being successfully browser-enabled.

2.2.5.2 Features

This element indicates the feature in the form template that is responsible for generating the corresponding **Message** element as specified in section 2.2.4.5, Message, when the protocol server is attempting to browser-enable or design check the form template (.xsn) file.

```

<s:simpleType name="Feature">
  <s:restriction base="s:string">
    <s:enumeration value="GenericXsf"/>
    <s:enumeration value="XsfSchema"/>
    <s:enumeration value="GenericXsl"/>
    <s:enumeration value="GenericXPath"/>
    <s:enumeration value="TemplateXml"/>
    <s:enumeration value="Layout"/>
    <s:enumeration value="Controls"/>
    <s:enumeration value="BusinessLogic"/>
    <s:enumeration value="Calculations"/>
    <s:enumeration value="Validation"/>
    <s:enumeration value="DigitalSignatures"/>
    <s:enumeration value="DataAdapters"/>
    <s:enumeration value="Submit"/>
    <s:enumeration value="Views"/>
    <s:enumeration value="Rules"/>
    <s:enumeration value="ConditionalFormatting"/>
    <s:enumeration value="VersionUpgrade"/>
  </s:restriction>
</s:simpleType>

```

The following table specifies **Feature** values and their proper use [<4>](#):

The **SolutionFormatVersion** attribute of the **xDocumentClass** element, as described in [\[MS-IPFF2\]](#) section 2.2.1.2.1, specifies whether the InfoPath Form Template Format Structure, as described in [\[MS-IPFF\]](#), or the InfoPath Form Template Format Version 2 Structure, as described in [\[MS-IPFF2\]](#), MUST be used when both versions of those structures are cited.

Value	Meaning
XsfSchema	Issues related to the form definition (.xsf) file as specified in [MS-IPFF] and [MS-IPFF2] section 2.2, XSF specification.
TemplateXml	Issues in the template.xml as specified in [MS-IPFF] and [MS-IPFF2] section 2.7, template.xml.
Layout	Issues relating to form view layout as specified in [MS-IPFF] and [MS-IPFF2] section 2.4, XSL specification.
Controls	Issues related to controls as specified in [MS-IPFF] section 2.3, XSD specification, and [MS-IPFF] and [MS-IPFF2] section 2.4, XSL specification.
BusinessLogic	Issues related to business logic (1) declarations as specified in [MS-IPFF] and [MS-IPFF2] section 2.2, XSF specification.
Calculations	Issues related to calculation declarations as specified in [MS-IPFF] and [MS-IPFF2] section 2.2, XSF specification.
Validation	Issues related to data validation declarations.
DigitalSignatures	Issues related to digital signatures (2) as specified in [MS-IPFF] section 2.2.126 and [MS-IPFF2] section 2.2.1.2.107, signed data block .
DataAdapters	Issues related to data adapter as specified in [MS-IPFF] and [MS-IPFF2] section 2.2, XSF specification.
Submit	Issues related to submit .
Views	Issues related to form views as specified in [MS-IPFF] and [MS-IPFF2] section 2.2, XSF specification and section 2.4, XSL specification.
Rules	Issues related to rules (1) as specified in [MS-IPFF] and [MS-IPFF2] section 2.2, XSF specification.
ConditionalFormatting	Issues related to conditional formatting as specified in [MS-IPFF] and [MS-IPFF2] section 2.4, XSL specification.
VersionUpgrade	Issues related to version upgrade, as specified in [MS-IPFF] and [MS-IPFF2] section 2.2.128.
GenericXsl	Issues related to form views without a more specific Feature value available, as specified in [MS-IPFF] and [MS-IPFF2] section 2.4, XSL specification.
GenericXPath	Issues related to a XPath expression without a more specific Feature value available.
GenericXsf	Issues related to the form definition (.xsf) file without a more specific Feature value available, as specified in [MS-IPFF] and [MS-IPFF2] section 2.2, XSF specification.

2.2.5.3 Category

This element indicates whether the corresponding **Message** element as specified in section [2.2.4.5](#), **Message**, is a potential performance issue or whether it represents an issue preventing the form template from being browser-enabled.

```
<s:simpleType name="Category">
  <s:restriction base="s:string">
```

```
<s:enumeration value="BrowserOptimization"/>
<s:enumeration value="BrowserCompatibility"/>
</s:restriction>
</s:simpleType>
```

The following table specifies **Category** values and their proper use:

Value	Meaning
BrowserOptimization	This enumeration value MUST be used when a browser-optimization issue is found during the analysis of the form template (.xsn) file.
BrowserCompatibility	This enumeration value MUST be used for all issues that are not browser-optimization issues

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 SHOULD interpret HTTP **Status-Codes** returned by the protocol server as specified in [\[RFC2616\]](#) section 10, **Status Code Definitions**. This protocol allows protocol servers to perform implementation-specific authorization checks and notify protocol clients of authorization faults either using HTTP Status-Codes or using SOAP faults.

3.1 Server Details

3.1.1 Abstract Data Model

This section describes a conceptual model of possible data organization that a server 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.

For **list (2)**-related web methods, the protocol server maintains a mapping between a form template (.xsn) file and a list content type (as specified in [\[MS-WSSTS\]](#) section 2.1.2.8).

For browser enabling web methods, the protocol server maintains a mapping between a form template (.xsn) file and a **document library** or **site content type**.

For design checking and user code checking web methods, the state of the protocol server does affect the responses it sends in response to the respective queries from the protocol client.

3.1.2 Timers

None.

3.1.3 Initialization

None.

3.1.4 Message Processing Events and Sequencing Rules

A protocol server MUST support form template (.xsn) files conformant to [\[MS-IPFF\]](#) and MUST support the **BrowserEnableUserFormTemplate** (section [3.1.4.1](#)) and **DesignCheckFormTemplate** (section [3.1.4.2](#)) WSDL operations. A protocol server SHOULD support form template (.xsn) files conformant to [\[MS-IPFF2\]](#).^{<5>} If a protocol server supports form template (.xsn) files conformant to [\[MS-IPFF2\]](#) that protocol server MUST support all WSDL operations specified in the following table. In response to any unsupported WSDL operation requests, the protocol server MAY^{<6>} return a SOAP fault.

Any protocol server that supports any of the WSDL operations **GetListFormLocation** (section [3.1.4.3](#)), **SetFormsForListItem** (section [3.1.4.5](#)) or **SetSchemaChangesForList** (section [3.1.4.6](#)) MUST support all three of these WSDL operations.

This specification includes the following WSDL operations:

WSDL Operation	Description
BrowserEnableUserFormTemplate	The BrowserEnableUserFormTemplate WSDL operation is used to browser-enable a form template (.xsn) file.
DesignCheckFormTemplate	The DesignCheckFormTemplate WSDL operation is used to design check a form template (.xsn) file.
GetListFormLocation	The GetListFormLocation WSDL operation is used to determine if there is a form template (.xsn) file mapped to a content type, and where that file is located.
GetUserCodeDeploymentDependencies	The GetUserCodeDeploymentDependencies WSDL operation is used to determine if a form template (.xsn) file with code can be browser-enabled on a site (2) as a sandboxed solution .
SetFormsForListItem	The SetFormsForListItem WSDL operation is used to design check a form template (.xsn) file and subsequently map it to a content type.
SetSchemaChangesForList	The SetSchemaChangesForList WSDL operation is used to change the list schema of the specified list (2) by performing adds, deletes and updates to the fields (2) of the list (2).

3.1.4.1 BrowserEnableUserFormTemplate

The **BrowserEnableUserFormTemplate** WSDL operation is used to browser-enable a form template (.xsn) file.

```
<wsdl:operation name="BrowserEnableUserFormTemplate">
  <wsdl:input message="tns:BrowserEnableUserFormTemplateSoapIn"/>
  <wsdl:output message="tns:BrowserEnableUserFormTemplateSoapOut"/>
</wsdl:operation>
```

As shown in the following figure, the protocol client sends a **BrowserEnableUserFormTemplateSoapIn** request **SOAP message**, and the protocol server responds with a **BrowserEnableUserFormTemplateSoapOut** response SOAP message.

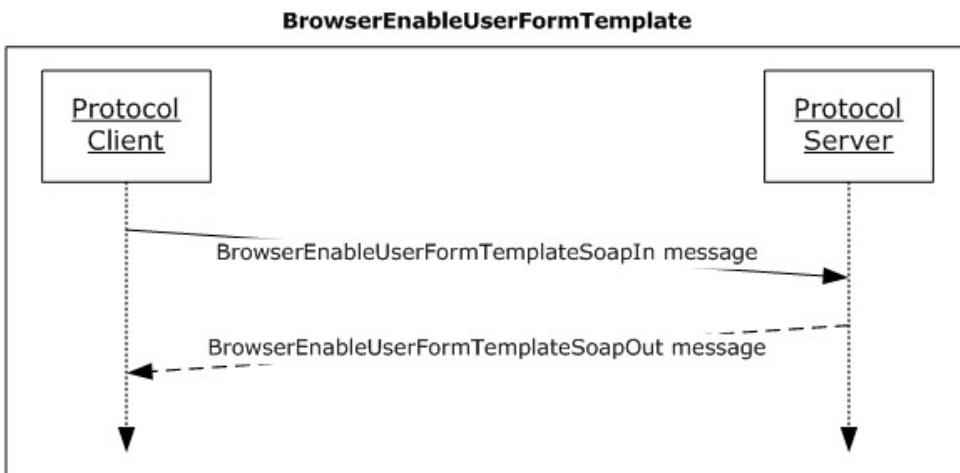


Figure 2: Exchange of SOAP messages for the `BrowserEnableUserFormTemplate` operation between protocol client and protocol server

3.1.4.1.1 Messages

3.1.4.1.1.1 `BrowserEnableUserFormTemplateSoapIn`

The requested WSDL message for the `BrowserEnableUserFormTemplate` WSDL operation.

The **SOAP action** value is:

```
http://schemas.microsoft.com/office/infopath/2007/formsServices/BrowserEnableUserFormTemplate
```

The **SOAP body** contains the `BrowserEnableUserFormTemplate` element.

3.1.4.1.1.2 `BrowserEnableUserFormTemplateSoapOut`

The response WSDL message for the `BrowserEnableUserFormTemplate` method.

The SOAP action value is:

```
http://schemas.microsoft.com/office/infopath/2007/formsServices/BrowserEnableUserFormTemplate
```

The SOAP body contains the `BrowserEnableUserFormTemplateResponse` element.

3.1.4.1.2 Elements

3.1.4.1.2.1 `BrowserEnableUserFormTemplate`

The input data for the `BrowserEnableUserFormTemplate` WSDL operation.

```
<s:element name="BrowserEnableUserFormTemplate">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="formTemplateLocation" type="s:string"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

formTemplateLocation: This is the **URL** of a form template (.xsn) file on the protocol server. The protocol server MUST be able to decode a URL as specified in [\[RFC2396\]](#).

3.1.4.1.2.2 `BrowserEnableUserFormTemplateResponse`

The result data for the `BrowserEnableUserFormTemplate` WSDL operation.

```
<s:element name="BrowserEnableUserFormTemplateResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="BrowserEnableUserFormTemplateResult"
type="MessagesResponse"/>
    </s:sequence>
  </s:complexType>
```

```
</s:element>
```

BrowserEnableUserFormTemplateResult: This element MUST be returned by the protocol server after attempting to browser-enable the form template. The form template (.xsn) file MUST be valid according to [\[MS-IPFF\]](#) or [\[MS-IPFF2\]](#) to be browser-enabled. The **SolutionFormatVersion** attribute of the **xDocumentClass** element, as described in [\[MS-IPFF2\]](#) section 2.2.1.2.1, specifies whether the InfoPath Form Template Format Structure, as described in [\[MS-IPFF\]](#), or the InfoPath Form Template Format Version 2 Structure, as described in [\[MS-IPFF2\]](#), MUST be used when both versions of those structures are cited.

If the form template (.xsn) file was successfully browser-enabled, the protocol server MUST send zero **Message** elements in the **Messages** element of the associated **MessagesResponse** element. [<7>](#)

If the attempt to browser-enable the form template (.xsn) file was unsuccessful, the protocol server MUST send one or more **Message** elements with the **MessageType** attribute set to "Error" in the **Messages** element of the associated **MessagesResponse** element.

The protocol server MUST NOT return Message elements with a **MessageType** of "Warning" or "Information" in a response to the **BrowserEnableUserFormTemplate** operation.

3.1.4.1.3 Complex Types

The following XML Schema complex type definitions are specific to this operation.

3.1.4.1.3.1 MessagesResponse

This is the container for the **Messages** element, as specified in section [2.2.4.3](#), returned by the protocol server after attempting to browser-enable.

```
<s:complexType name="MessagesResponse">
  <s:sequence>
    <s:element name="Messages" type="tns:Messages" minOccurs="1" maxOccurs="1"/>
  </s:sequence>
</s:complexType>
```

Messages: Specified in section 2.2.4.3, **Messages**.

3.1.4.2 DesignCheckFormTemplate

The **DesignCheckFormTemplate** WSDL operation is used to design check a form template (.xsn) file.

```
<wsdl:operation name="DesignCheckFormTemplate">
  <wsdl:input message="tns:DesignCheckFormTemplateSoapIn"/>
  <wsdl:output message="tns:DesignCheckFormTemplateSoapOut"/>
</wsdl:operation>
```

As shown in the following figure, the protocol client sends a **DesignCheckFormTemplateSoapIn** request SOAP message and the protocol server responds with a **DesignCheckFormTemplateSoapOut** response SOAP message.

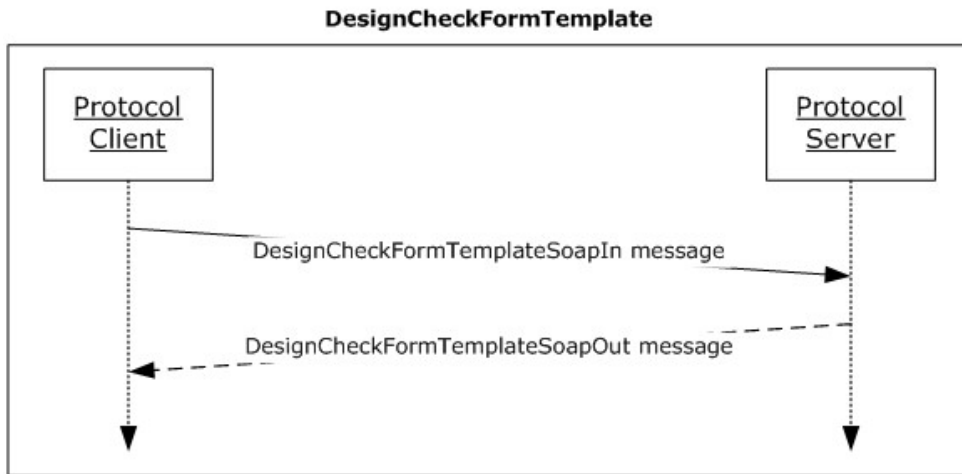


Figure 3: Exchange of SOAP messages for the DesignCheckFormTemplate operation between protocol client and protocol server

3.1.4.2.1 Messages

3.1.4.2.1.1 DesignCheckFormTemplateSoapIn

The requested WSDL message for the **DesignCheckFormTemplate** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/infopath/2007/formsServices/DesignCheckFormTemplate
```

The SOAP body contains the **DesignCheckFormTemplate** element.

3.1.4.2.1.2 DesignCheckFormTemplateSoapOut

The response WSDL message for the **DesignCheckFormTemplate** method.

The SOAP action value is:

```
http://schemas.microsoft.com/office/infopath/2007/formsServices/DesignCheckFormTemplate
```

The SOAP body contains the **DesignCheckFormTemplateResponse** element.

3.1.4.2.2 Elements

3.1.4.2.2.1 DesignCheckFormTemplate

The input data for the **DesignCheckFormTemplate** WSDL operation.

```
<s:element name="DesignCheckFormTemplate">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="lcid" type="s:int"/>
    
```

```

        <s:element minOccurs="1" maxOccurs="1" name="base64FormTemplate" type="s:string"/>
        <s:element minOccurs="1" maxOccurs="1" name="applicationId" type="s:string"/>
    </s:sequence>
</s:complexType>
</s:element>

```

lcid: Specifies a language code identifier (LCID), as specified in [\[MS-LCID\]](#).

base64FormTemplate: The Base64 encoded form template (.xsn) file to design check. If the Base64 encoding is not valid, then the protocol server SHOULD return a **Message** element with a **MessageType** of "Error" to the protocol client, but it MAY [<8>](#) return a SOAP fault. When the Base64 value is decoded it MUST be a form template (.xsn) file as specified in [\[MS-IPFF\]](#) or [\[MS-IPFF2\]](#). The **SolutionFormatVersion** attribute of the **xDocumentClass** element, as described in [\[MS-IPFF2\]](#) section 2.2.1.2.1, specifies whether the InfoPath Form Template Format Structure, as described in [\[MS-IPFF\]](#), or the InfoPath Form Template Format Version 2 Structure, as described in [\[MS-IPFF2\]](#), MUST be used when both versions of those structures are cited.

applicationId: Specifies the Unicode string that identifies the protocol client that initiated the request to **DesignCheckFormTemplate**. [<9>](#)

3.1.4.2.2 DesignCheckFormTemplateResponse

The result data for the **DesignCheckFormTemplate** WSDL operation.

```

<s:element name="DesignCheckFormTemplateResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="DesignCheckFormTemplateResult"
type="DesignCheckerInformation"/>
    </s:sequence>
  </s:complexType>
</s:element>

```

DesignCheckFormTemplateResult: This element is returned by the protocol server after design checking the form template (.xsn) file.

DesignCheckFormTemplateResult: Specified in section [2.2.4.4](#), **DesignCheckerInformation**.

Any localized strings returned by the protocol server SHOULD be in the language requested by the protocol client. If the protocol server is unable to provide strings in the requested language, then the protocol server MAY [<10>](#) use any available language.

3.1.4.2.3 Complex Types

The following XML Schema complex type definitions are specific to this operation.

3.1.4.2.3.1 DesignCheckerInformation

This element is specified in section [2.2.4.4](#), **DesignCheckerInformation**.

3.1.4.3 GetListFormLocation

The GetListFormLocation WSDL operation is used to determine if there is a form template (.xsn) file mapped to a content type, and where that file is located.

```

<wsdl:operation name="GetListFormLocation">
  <wsdl:input message="tns:GetListFormLocationSoapIn"/>
  <wsdl:output message="tns:GetListFormLocationSoapOut"/>
</wsdl:operation>

```

As shown in the following figure, the protocol client sends a **GetListFormLocationSoapIn** request SOAP message and the protocol server responds with a **GetListFormLocationSoapOut** response SOAP message.

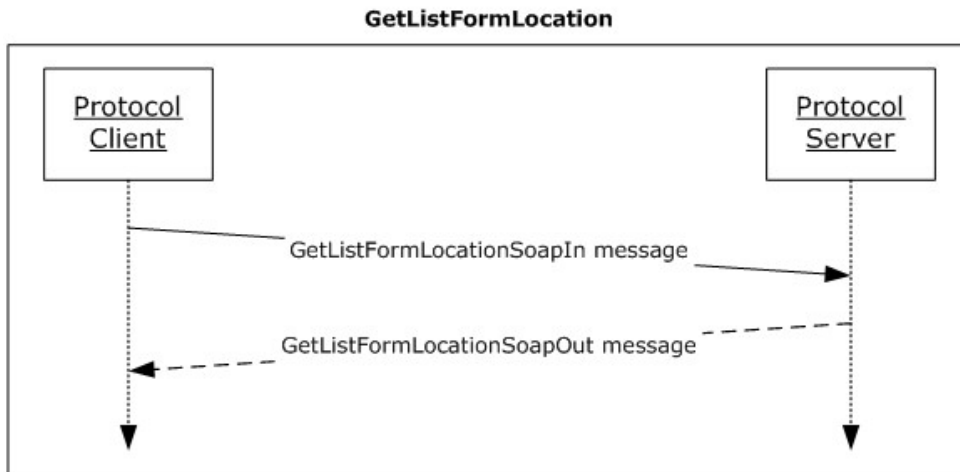


Figure 4: Exchange of SOAP messages for the GetListFormLocation operation between protocol client and protocol server

3.1.4.3.1 Messages

3.1.4.3.1.1 GetListFormLocationSoapIn

The requested WSDL message for the **GetListFormLocation** WSDL operation.

The SOAP action value is:

```

http://schemas.microsoft.com/office/infopath/2007/formsServices/GetListFormLocation

```

The SOAP body contains the **GetListFormLocation** element.

3.1.4.3.1.2 GetListFormLocationSoapOut

The response WSDL message for the **GetListFormLocation** method.

The SOAP action value is:

```

http://schemas.microsoft.com/office/infopath/2007/formsServices/GetListFormLocation

```

The SOAP body contains the **GetListFormLocationResponse** element.

3.1.4.3.2 Elements

3.1.4.3.2.1 GetListFormLocation

The input data for the **GetListFormLocation** WSDL operation.

```
<s:element name="GetListFormLocation">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="lcid" type="s:int"/>
      <s:element minOccurs="1" maxOccurs="1" name="listGuid" type="s:string"/>
      <s:element minOccurs="1" maxOccurs="1" name="contentTypeId" type="s:string"/>
      <s:element minOccurs="1" maxOccurs="1" name="checkDesignPermissions" type="s:boolean"/>
      <s:element minOccurs="1" maxOccurs="1" name="checkCustomFormEnabled" type="s:boolean"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

lcid: Specifies a language code identifier (LCID) (as specified in [\[MS-LCID\]](#)) identifying the preferred message language of the protocol client. See section [2.2.4.5, Message](#), for localization considerations.

listGuid: Specifies a **list identifier** (as specified in [\[MS-WSSTS\]](#) section 2.1.2.7).

contentTypeId: Specifies a **content type identifier** (as specified in [\[MS-WSSTS\]](#) section 2.1.2.8.1). The content type identified MUST be associated with the list (2) specified by **listGuid**.

checkDesignPermissions: If this parameter is "true" the protocol server MUST check for the same level of permissions as required for the **SetFormsForListItem** operation. Else if the parameter value is "false", the protocol server MUST check for a WSS Rights Mask with the OpenItems flag set as specified in [\[MS-WSSFO2\]](#) section 2.2.3.14. If in either case the protocol client does not have sufficient rights the protocol server MUST return a Status-Code "401" (Unauthorized).

checkCustomFormEnabled: If this parameter is "true", the protocol server MUST only return a non-empty **GetListFormLocationResult** as specified in section [3.1.4.3.2.2](#) when the **listGuid** and **contentTypeId** parameters identify a browser-enabled form template (.xsn) file.

3.1.4.3.2.2 GetListFormLocationResponse

The result data for the **GetListFormLocation** WSDL operation.

```
<s:element name="GetListFormLocationResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="GetListFormLocationResult"
type="s:string"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

GetListFormLocationResult: If the **contentTypeId** parameter does not identify an existing content type, then the protocol server MUST return a SOAP fault. If there is no form template (.xsn) file mapped to the specified content type, then the protocol server MUST return an empty **GetListFormLocationResult** element. If a form template (.xsn) file is mapped to the content type but the form template is not browser-enabled, and the **checkCustomFormEnabled** parameter is

"true", then the protocol server MUST return an empty `GetListFormLocationResult`. In all other cases, the `GetListFormLocationResult` element MUST be an absolute **Internationalized Resource Identifier (IRI)** pointing to the location of the form template (.xsn) file.

3.1.4.4 GetUserCodeDeploymentDependencies

The **GetUserCodeDeploymentDependencies** WSDL operation is used to determine if a form template (.xsn) file with code can be browser-enabled on a site (2) as a sandboxed solution.

```
<wsdl:operation name="GetUserCodeDeploymentDependencies">
  <wsdl:input message="tns:GetUserCodeDeploymentDependenciesSoapIn"/>
  <wsdl:output message="tns:GetUserCodeDeploymentDependenciesSoapOut"/>
</wsdl:operation>
```

As shown in the following figure, the protocol client sends a **GetUserCodeDeploymentDependenciesSoapIn** request SOAP message, and the protocol server responds with a **GetUserCodeDeploymentDependenciesSoapOut** response SOAP message.

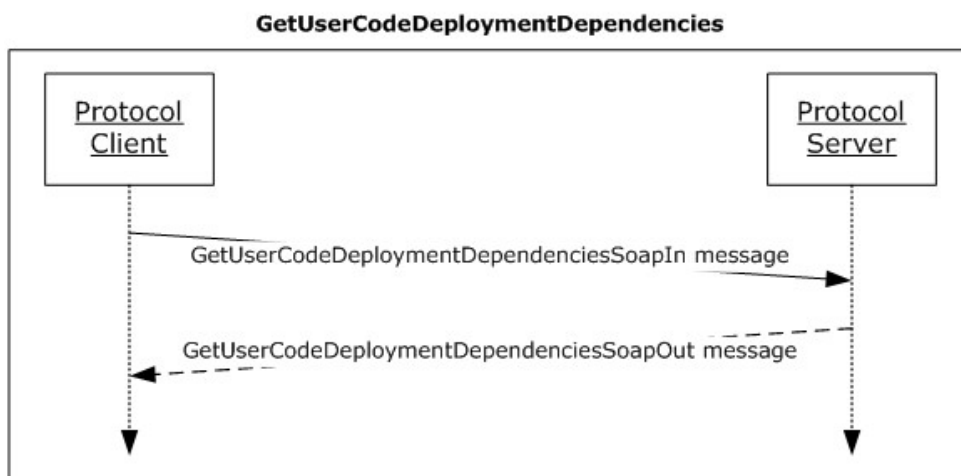


Figure 5: Exchange of SOAP messages for the GetUserCodeDeploymentDependencies operation between the protocol client and protocol server

3.1.4.4.1 Messages

3.1.4.4.1.1 GetUserCodeDeploymentDependenciesSoapIn

The requested WSDL message for the **GetUserCodeDeploymentDependencies** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/infopath/2007/formsServices/GetUserCodeDeploymentDependencies
```

The SOAP body contains the **GetUserCodeDeploymentDependencies** element.

3.1.4.4.1.2 GetUserCodeDeploymentDependenciesSoapOut

The response WSDL message for the **GetUserCodeDeploymentDependencies** method.

The SOAP action value is:

```
http://schemas.microsoft.com/office/infopath/2007/formsServices/GetUserCodeDeploymentDependencies
```

The SOAP body contains the **GetUserCodeDeploymentDependenciesResponse** element.

3.1.4.4.2 Elements

3.1.4.4.2.1 GetUserCodeDeploymentDependencies

The input data for the **GetUserCodeDeploymentDependencies** WSDL operation.

```
<s:element name="GetUserCodeDeploymentDependencies">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="siteCollectionLocation" type="s:string"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

siteCollectionLocation: This element is the URL of the site (2) to verify that sandboxed solution activation is permitted. The protocol server MUST be able to decode the URL as specified in [\[RFC2396\]](#).

3.1.4.4.2.2 GetUserCodeDeploymentDependenciesResponse

The result data for the **GetUserCodeDeploymentDependencies** WSDL operation.

```
<s:element name="GetUserCodeDeploymentDependenciesResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="GetUserCodeDeploymentDependenciesResult"
type="tns:UserSolutionActivationStatus"/>
    </s:sequence>
  </s:complexType>
</s:element>
```

GetUserCodeDeploymentDependenciesResult: This element is specified as a child of the **GetUserCodeDeploymentDependenciesResponse** container. The protocol server MUST return one of the values specified in [3.1.4.4.3.1](#).

3.1.4.4.3 Simple Types

3.1.4.4.3.1 UserSolutionActivationStatus

This element specifies the allowed values for the **GetUserCodeDeploymentDependenciesResult** element.

```

<s:simpleType name="UserSolutionActivationStatus">
  <s:restriction base="s:string">
    <s:enumeration value="ActivationAllowed"/>
    <s:enumeration value="ActivationDenied"/>
    <s:enumeration value="UserCodeServiceNotAvailable"/>
    <s:enumeration value="ServiceQuotaExceeded"/>
    <s:enumeration value="InsufficientUserPermissions"/>
  </s:restriction>
</s:simpleType>

```

The following table specifies the allowable values for UserSolutionActivationStatus:

Value	Meaning
ActivationAllowed	The protocol server MUST return this value to specify that the user code support is available on the server and the web service user is allowed to browser-enable a form template (.xsn) file with code.
ActivationDenied	The protocol server MUST return this value to specify that the web service user is not allowed to browser-enable a form template (.xsn) file with code on the site (2) specified in 3.1.4.4.2.1. The protocol server MUST NOT return this enumeration value for conditions that are covered by UserCodeServiceNotAvailable , ServiceQuotaExceeded and InsufficientUserPermissions values specified in this table.
UserCodeServiceNotAvailable	The protocol server MUST return this value to specify that the user code support is not available on the server.
ServiceQuotaExceeded	The protocol server MUST return this value to specify that the user code support is not available on the site (2) specified in 3.1.4.4.2.1 because the quota allotted to sandboxed solutions has been exceeded.
InsufficientUserPermissions	The protocol server MUST return this value to specify that the web service user is not allowed to browser-enable a form template (.xsn) file with code because the web service user is not a site collection administrator on the site (2) specified in 3.1.4.4.2.1.

3.1.4.5 SetFormsForListItem

The **SetFormsForListItem** WSDL operation is used to design check a form template (.xsn) file and subsequently map it to a content type.

```

<wsdl:operation name="SetFormsForListItem">
  <wsdl:input message="tns:SetFormsForListItemSoapIn"/>
  <wsdl:output message="tns:SetFormsForListItemSoapOut"/>
</wsdl:operation>

```

As shown in the following figure, the protocol client sends a **SetFormForListItemSoapIn** request SOAP message and the protocol server responds with a **SetFormForListItemSoapOut** response SOAP message.

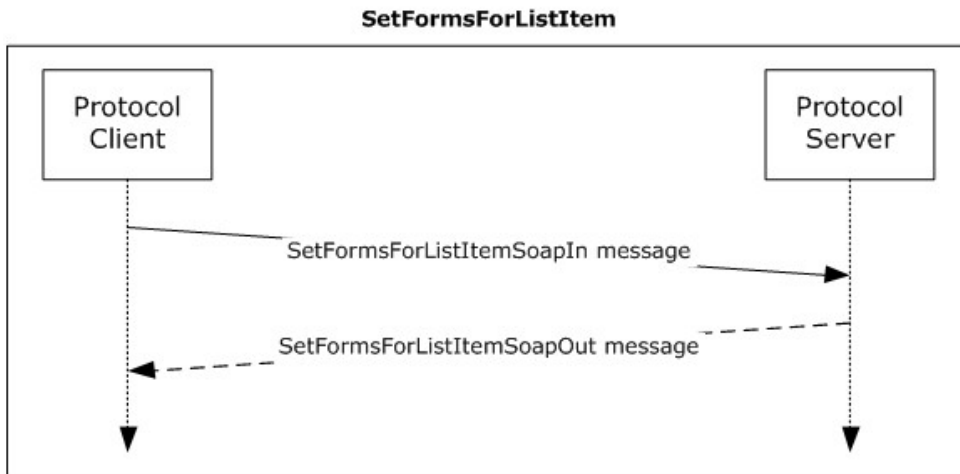


Figure 6: Exchange of SOAP messages for the SetFormsForListItem operation between protocol client and protocol server

3.1.4.5.1 Messages

3.1.4.5.1.1 SetFormsForListItemSoapIn

The requested WSDL message for the **SetFormsForListItem** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/infopath/2007/formsServices/SetFormsForListItem
```

The SOAP body contains the **SetFormsForListItem** element.

3.1.4.5.1.2 SetFormsForListItemSoapOut

The response WSDL message for the **SetFormsForListItem** method.

The SOAP action value is:

```
http://schemas.microsoft.com/office/infopath/2007/formsServices/SetFormsForListItem
```

The SOAP body contains the **SetFormsForListItemResponse** element.

3.1.4.5.2 Elements

3.1.4.5.2.1 SetFormsForListItem

The input data for the **SetFormsForListItem** WSDL operation.

```
<s:element name="SetFormsForListItem">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="lcid" type="s:int"/>
    </s:sequence>
  </s:complexType>
</s:element>
```



```

    <s:element minOccurs="1" maxOccurs="1" name="base64FormTemplate" type="s:string"/>
    <s:element minOccurs="1" maxOccurs="1" name="applicationId" type="s:string"/>
    <s:element minOccurs="1" maxOccurs="1" name="listGuid" type="s:string"/>
    <s:element minOccurs="1" maxOccurs="1" name="contentTypeId" type="s:string"/>
  </s:sequence>
</s:complexType>
</s:element>

```

lcid: Specifies a language code identifier (LCID) (as specified in [\[MS-LCID\]](#)).

base64FormTemplate: The **Base64** encoded form template (.xsn) file to map. If the Base64 encoding is not valid, then the protocol server SHOULD return a **Message** element with a **MessageType** of Error to the protocol client, but it MAY [<11>](#) return a SOAP fault. When the Base64 value is decoded it MUST be a form template (.xsn) file as specified in [\[MS-IPFF\]](#) or [\[MS-IPFF2\]](#). The **SolutionFormatVersion** attribute of the **xDocumentClass** element, as described in [\[MS-IPFF2\]](#) section 2.2.1.2.1, specifies whether the InfoPath Form Template Format Structure, as described in [\[MS-IPFF\]](#), or the InfoPath Form Template Format Version 2 Structure, as described in [\[MS-IPFF2\]](#), MUST be used when both versions of those structures are cited.

applicationId: Specifies the Unicode string that identifies the protocol client that initiated the request to **SetFormsForListItem**. This element SHOULD be ignored by the protocol server. [<12>](#)

listGuid: Specifies a list identifier (as specified in [\[MS-WSSTS\]](#) section 2.1.2.7).

contentTypeId: Specifies a content type identifier (as specified in [\[MS-WSSTS\]](#) section 2.1.2.8.1). The content type identified MUST be associated with the list (2) specified by **listGuid**.

3.1.4.5.2.2 SetFormsForListItemResponse

The result data for the **SetFormsForListItem** WSDL operation.

```

<s:element name="SetFormsForListItemResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="SetFormsForListItemResult"
type="DesignCheckerInformation"/>
    </s:sequence>
  </s:complexType>
</s:element>

```

SetFormsForListItemResult: This element is returned by the protocol server after attempting to map the form template (.xsn) file to the content type as specified by the protocol client request in the **SetFormsForListItem** request element. The form template (.xsn) file MUST be valid according to [\[MS-IPFF2\]](#) to be browser-enabled and mapped to a content type.

If the form template (.xsn) file was successfully mapped to the content type and also browser-enabled, the protocol server MUST send zero **Message** elements in the **Messages** element of the associated **MessagesResponse** element. [<13>](#)

If the attempt to browser-enable the form template (.xsn) file was unsuccessful, the protocol server MUST send one or more **Message** elements with the **MessageType** attribute set to "Error" in the **Messages** element of the associated **MessagesResponse** element. [<14>](#) For localization considerations regarding preferred message string language, see section [2.2.4.5, Message](#).

If the attempt to browser-enable the form template (.xsn) file was successful, but the attempt to map the form template (.xsn) file to the content type was unsuccessful, the protocol server MUST return a SOAP fault.

The protocol server MUST NOT return Message elements with a **MessageType** of "Warning" or "Information" in a response to the **SetFormsForListItem** operation.

3.1.4.5.3 Complex Types

The following XML Schema complex type definitions are specific to this operation.

3.1.4.5.3.1 DesignCheckerInformation

This element is specified in section [2.2.4.4](#), **DesignCheckerInformation**.

3.1.4.6 SetSchemaChangesForList

The **SetSchemaChangesForList** WSDL operation is used to change the list schema of the specified list (2) by performing adds, deletes and updates to the fields (2) of the list (2).

```
<wsdl:operation name="SetSchemaChangesForList">
  <wsdl:input message="tns:SetSchemaChangesForListSoapIn"/>
  <wsdl:output message="tns:SetSchemaChangesForListSoapOut"/>
</wsdl:operation>
```

As show in the following figure, the protocol client sends a **SetSchemaChangesforListSoapIn** request **SOAP message** and the protocol server responds with a **SetSchemaChangesForListSoapOut** response **SOAP message**.

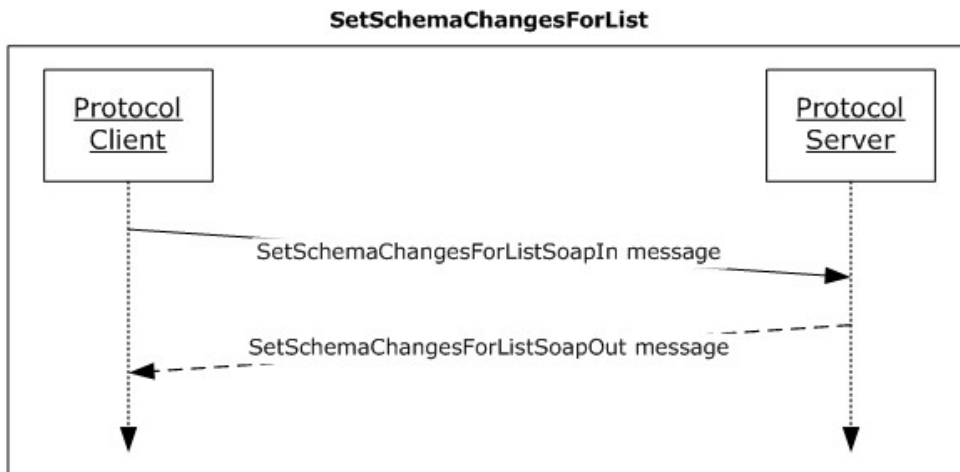


Figure 7: Exchange of SOAP messages for the SetSchemaChangesForList operation between the protocol client and protocol server

3.1.4.6.1 Messages

3.1.4.6.1.1 SetSchemaChangesForListSoapIn

The requested WSDL message for the **SetSchemaChangesForList** WSDL operation.

The SOAP action value is:

```
http://schemas.microsoft.com/office/infopath/2007/formsServices/SetSchemaChangesForList
```

The SOAP body contains the **SetSchemaChangesForList** element.

3.1.4.6.1.2 SetSchemaChangesForListSoapOut

The response WSDL message for the **SetSchemaChangesForList** method.

The SOAP action value is:

```
http://schemas.microsoft.com/office/infopath/2007/formsServices/SetSchemaChangesForList
```

The SOAP body contains the **SetSchemaChangesForListResponse** element.

3.1.4.6.2 Elements

3.1.4.6.2.1 SetSchemaChangesForList

The input data for the **SetSchemaChangesForList** WSDL operation.

```
<s:element name="SetSchemaChangesForList">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="lcid" type="s:int"/>
      <s:element minOccurs="1" maxOccurs="1" name="listGuid" type="s:string"/>
      <s:element minOccurs="1" maxOccurs="1" name="contentTypeid" type="s:string"/>
      <s:element minOccurs="1" maxOccurs="1" name="newFields">
        <s:complexType>
          <s:sequence>
            <s:element ref="s:schema"/>
            <s:any/>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element minOccurs="1" maxOccurs="1" name="updateFields">
        <s:complexType>
          <s:sequence>
            <s:element ref="s:schema"/>
            <s:any/>
          </s:sequence>
        </s:complexType>
      </s:element>
      <s:element minOccurs="1" maxOccurs="1" name="deleteFields">
        <s:complexType>
          <s:sequence>
            <s:element ref="s:schema"/>
            <s:any/>
          </s:sequence>
        </s:complexType>
      </s:element>
    </s:sequence>
  </s:complexType>
</s:element>
```

```

        </s:sequence>
      </s:complexType>
    </s:element>
  </s:sequence>
</s:complexType>
</s:element>

```

lcid: Specifies a language code identifier (LCID) (as specified in [\[MS-LCID\]](#)) identifying the preferred language of the protocol server for all returned strings – see section [2.2.4.5](#), **Message**, for details.

listGuid: Specifies a list identifier (as specified in [\[MS-WSSTS\]](#) section 2.1.2.7).

contentTypeid: Specifies a content type identifier (as specified in [\[MS-WSSTS\]](#) section 2.1.2.8.1).

newFields: Specifies an **XML fragment** which defines the **CAML** representation for the fields (2) to add in the list (2) (as specified in [\[MS-LISTSWS\]](#) section 2.2.4.15).

updateFields: Specifies an XML fragment which defines the CAML representation for the fields (2) to update in the list (2) (as specified in [\[MS-LISTSWS\]](#) section 2.2.4.15).

deleteFields: Specifies an XML fragment which defines the CAML representation for the fields (2) to delete in the list (2) (as specified in [\[MS-LISTSWS\]](#) section 2.2.4.15).

3.1.4.6.2.2 SetSchemaChangesForListResponse

The result data for the **SetSchemaChangesForList** WSDL operation.

```

<s:element name="SetSchemaChangesForListResponse">
  <s:complexType>
    <s:sequence>
      <s:element minOccurs="1" maxOccurs="1" name="SetSchemaChangesForListResult"
        type="NewFieldsNameMapping"/>
    </s:sequence>
  </s:complexType>
</s:element>

```

SetSchemaChangesForListResult: This element specifies an XML fragment returned by the protocol server after performing the requested field (2) operations and represents the CAML of the new fields (2) added to the list (2) by the protocol server.

The protocol server **MUST** return a **NewFieldNameMapping** XML fragment for every field for which the protocol server assigns a different **field internal name** than the field internal name specified in **SetSchemaChangesForList**'s **newFields** element.

The **NewFieldsNameMapping** XML fragment **MUST NOT** contain any **NewFieldNameMapping** XML fragments if the protocol server does not assign a new field internal name to any field (2) added to the list (2).

The protocol server **MUST NOT** return **NewFieldNameMapping** XML fragments where the **Name** value does not match a field internal name specified in the **newFields** element of the **SetSchemaChangesForList** web method.

The protocol server **SHOULD** return a SOAP fault if any field (2) operation fails, but the protocol server **MAY** [<15>](#) proceed with the remaining field (2) operations requested by the protocol client.

SetFormsForListItemResult: This element is returned by the protocol server after attempting to map the form template (.xsn) file to the content type as specified by the protocol client request in the **SetFormsForListItem** request element. The form template (.xsn) file MUST be valid according to [\[MS-IPFF2\]](#) to be browser-enabled and mapped to a content type.

If the form template (.xsn) file was successfully mapped to the content type and also browser-enabled, the protocol server MUST send zero **Message** elements in the **Messages** element of the associated **MessagesResponse** element. [<16>](#)

If the attempt to browser-enable the form template (.xsn) file was unsuccessful, the protocol server MUST send one or more **Message** elements with the **MessageType** attribute set to "Error" in the **Messages** element of the associated **MessagesResponse** element. [<17>](#) For localization considerations regarding preferred message string language, see section [2.2.4.5, Message](#).

If the attempt to browser-enable the form template (.xsn) file was successful, but the attempt to map the form template (.xsn) file to the content type was unsuccessful, the protocol server MUST return a SOAP fault.

The protocol server MUST NOT return Message elements with a **MessageType** of "Warning" or "Information" in a response to the **SetFormsForListItem** operation.

3.1.4.6.3 Complex Types

The following XML Schema complex type definitions are specific to this operation.

3.1.4.6.3.1 NewFieldsNameMapping

This element contains **NewFieldNameMapping** elements as specified in section [3.1.4.6.2.2, NewFieldNameMapping](#).

```
<s:complexType name="NewFieldsNameMapping">
  <s:sequence>
    <s:element minOccurs="0" maxOccurs="unbounded" name="NewFieldsMapping"
      type="tns:NewFieldNameMapping"/>
  </s:sequence>
</s:complexType>
```

NewFieldsMapping: Specified in section [3.1.4.6.2.2, NewFieldNameMapping](#).

3.1.4.6.3.2 NewFieldNameMapping

This element specifies the mapping between a field internal name specified in the **SetSchemaChangesForList**'s **newFields** element and the field internal name assigned by the protocol server.

```
<s:complexType name="NewFieldNameMapping">
  <s:sequence>
    <s:element minOccurs="1" maxOccurs="1" name="Name" type="s:string"/>
    <s:element minOccurs="1" maxOccurs="1" name="NewName" type="s:string"/>
  </s:sequence>
</s:complexType>
```

Name: Specifies the field internal name as set in the **newFields** XML fragment of the **SetSchemachangesForList** web method.

NewName: Specifies the field internal name as set by the protocol server when adding the new field (2) to the list (2).

3.1.5 Timer Events

None.

3.1.6 Other Local Events

None.

4 Protocol Examples

4.1 DesignCheckFormTemplate Operation Examples

This example shows the use of the **DesignCheckFormTemplate** operation.

4.1.1 DesignCheckFormTemplate Request and Response with No Message Elements (No Issues Found)

This protocol example details a request to the **DesignCheckFormTemplate** operation and the subsequent response with zero **Message** elements, indicating no browser compatibility or browser-optimization issues were found.

The following is the request to the **DesignCheckFormTemplate** operation sent by the protocol client:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
  <SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
    xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance" xmlns:SOAP-
    ENV="http://schemas.xmlsoap.org/soap/envelope/"> <SOAP-ENV:Body> <DesignCheckFormTemplate
    xmlns='http://schemas.microsoft.com/office/infopath/2007/formsServices'>
      <lcid>1033</lcid>
      <base64FormTemplate>...</base64FormTemplate>
      <applicationId>InfoPath 14</applicationId>
    </DesignCheckFormTemplate>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

The **lcid** element value of "1033" indicates the protocol client is requesting a response from the protocol server with strings suitable for display localized in English (United States).

The **base64FormTemplate** element value of ellipses ("...") indicate a Base64 encoded form template (.xsn) file.

The **applicationId** element value of "InfoPath 14" identifies the protocol client.

The following is the response from the **DesignCheckFormTemplate** operation sent by the protocol server. Within the **Messages** element, it shows zero **Message** elements, indicating the form template (.xsn) file is compatible with the protocol server.

```
<?xml version="1.0" encoding="utf-8"?> <soap:Envelope
xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema"> <soap:Body> <DesignCheckFormTemplateResponse
xmlns="http://schemas.microsoft.com/office/infopath/2007/formsServices">
  <DesignCheckFormTemplateResult>
    <ApplicationId>InfoPath 14</ApplicationId>
    <Lcid>1033</Lcid>
    <Categories>
      <Category>
        <Id>BrowserOptimization</Id>
        <Label>Browser Optimization</Label>
        <HideWarningsByDefault>true</HideWarningsByDefault>
      </Category>
      <Category>
        <Id>BrowserCompatibility</Id>
```

```

        <Label>Browser Compatibility (Verified on server) </Label>
        <HideWarningsByDefault>>false</HideWarningsByDefault>
    </Category>
</Categories>
    <Messages />
</DesignCheckFormTemplateResult>
</DesignCheckFormTemplateResponse>
</soap:Body>
</soap:Envelope>

```

The **ApplicationId** element value of "InfoPath 14" demonstrates that the protocol server retains the value from the applicationId element in the request message and includes that value in the response.

The **Lcid** element value of "1033" indicates the protocol server is returning string suitable for display to an end user in English (United States).

4.1.2 DesignCheckFormTemplate Response with Message Elements (One or More Issues Found)

This protocol example details a response from the **DesignCheckFormTemplate** with **Message** elements, indicating browser compatibility or browser-optimization issues were found. See section 4.1.1, DesignCheckFormTemplate Request and Response with zero Message elements (no issues found), for a typical request message.

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <DesignCheckFormTemplateResponse
xmlns="http://schemas.microsoft.com/office/infopath/2007/formsServices">
            <DesignCheckFormTemplateResult>
                <ApplicationId>InfoPath 14</ApplicationId>
                <Lcid>1033</Lcid>
                <Categories>
                    <Category>
                        <Id>BrowserOptimization</Id>
                        <Label>Browser Optimization</Label>
                        <HideWarningsByDefault>>true</HideWarningsByDefault>
                    </Category>
                    <Category>
                        <Id>BrowserCompatibility</Id>
                        <Label>Browser Compatibility (Verified on server)</Label>
                        <HideWarningsByDefault>>false</HideWarningsByDefault>
                    </Category>
                </Categories>
                <Messages>
                    <Message Id="107" Type="Warning" Feature="Controls"
Category="BrowserCompatibility">
                        <ShortMessage>Custom task panes are not supported</ShortMessage>
                        <DetailedMessage>Custom task panes are not supported by Infopath Forms Services.
The custom task Pane will not be displayed when users fill out a form in a Web
browser.</DetailedMessage>
                        <SourceLocation FileName="manifest.xsf" />
                    </Message>
                    <Message Id="44" Type="Error" Feature="Controls" Category="BrowserCompatibility">

```



```

    <ShortMessage>Digital signatures on the entire form are not
supported</ShortMessage>
    <DetailedMessage>Digital signatures on the entire form are not supported by
Infopath Forms Services. Digital signature settings must be changed before the form can be
published. On the Tools menu, click Form Options, and then disable the digital signatures or
enable the digital signatures on specific data in the form.</DetailedMessage>
    <SourceLocation FileName="manifest.xsf" />
  </Message>
</Messages>
</DesignCheckFormTemplateResult>
</DesignCheckFormTemplateResponse>
</soap:Body>
</soap:Envelope>

```

The **ApplicationId** element value of "InfoPath 14" demonstrates the protocol server's pass-thru of the value from the request's applicationId element.

The **Lcid** element value of "1033" indicates the protocol server is returning string suitable for display to an end user in English (United States).

The two **Message** element's demonstrate checks as specified in section [2.2.4.5](#), Message. One has the **Message Id** value "107", detailing a warning that does not prevent browser enabling a form template (.xsn) file. The other has the **Message Id** value being "44", detailing an error that would prevent browser enabling a form template (.xsn) file.

4.2 BrowserEnableUserFormTemplate Operation Examples

This example shows the use of the **BrowserEnableUserFormTemplate** operation.

4.2.1 BrowserEnableUserFormTemplate Request/Response Indicating the Successful Browser Enabling of a Form Template (.xsn) File

This protocol example details a request to the **BrowserEnableUserFormTemplate** operation and the subsequent response with zero **Message** elements, indicating the form template (.xsn) file was successfully browser enabled.

The following is the request to the **BrowserEnableUserFormTemplate** operation sent by the protocol client:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <BrowserEnableUserFormTemplate
xmlns='http://schemas.microsoft.com/office/infopath/2007/formsServices'>
      <formTemplateLocation>http://contoso/DocLib/forms/template.xsn</formTemplateLocation>
    </BrowserEnableUserFormTemplate>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

The **formTemplateLocation** element value of "http://contoso/DocLib/forms/template.xsn" represents the URL on the protocol server of a form template (.xsn) file that the protocol client is requesting be browser-enabled.

The following is the response from the **BrowserEnableUserFormTemplate** operation sent by the protocol server. Within the **Messages** element, it shows zero **Message** elements, indicating the form template (.xsn) file is compatible with the protocol server.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <BrowserEnableUserFormTemplateResponse
xmlns="http://schemas.microsoft.com/office/infopath/2007/formsServices">
      <BrowserEnableUserFormTemplateResult>
        <Messages />
      </BrowserEnableUserFormTemplateResult>
    </BrowserEnableUserFormTemplateResponse>
  </soap:Body>
</soap:Envelope>
```

4.3 SetFormsForListItem Operation Examples

This example shows the use of the **SetFormsForListItem** operation.

4.3.1 SetFormsForListItem Request/Response Indicating Successful Operations on a List

This protocol example details a request to the **SetFormsForListItem** operation and the subsequent response with zero **Message** elements, indicating no browser compatibility or browser-optimization issues were found, and there were no issues with mapping the form template (.xsn) file to the content type.

The following is the request sent by the protocol client:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SetFormsForListItem
xmlns='http://schemas.microsoft.com/office/infopath/2007/formsServices'>
      <lcid>1033</lcid>
      <base64FormTemplate>...</base64FormTemplate>
      <applicationId>InfoPath 14</applicationId>
      <listGuid>{6747CDB8-ADB5-4A5F-9323-AFCC3D329358}</listGuid>
      <contentTypeId>0x010500C8B78A06972AA34682872D813D9D0ED3</contentTypeId>
    </SetFormsForListItem>
  </SOAP-ENV:Body>
</SOAP-ENV:Envelope>
```

The **lcid** element value of "1033" indicates the protocol client is requesting a response from the protocol server with strings suitable for display localized in English (United States).

The **base64FormTemplate** element value of ellipses ("...") indicates a Base64 encoded form template (.xsn) file.

The **applicationID** element value of "InfoPath 14" identifies the protocol client.

The **listGuid** element value is a valid list identifier on the protocol server.

The **contentTypeId** element value is a valid content type identifier on the protocol server.

The following is the response from the **SetFormsForListItem** operation sent by the protocol server. Within the Messages element, it shows zero Message elements, indicating the form template (.xsn) file is compatible with the protocol server.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <SetFormsForListItemResponse
xmlns="http://schemas.microsoft.com/office/infopath/2007/formsServices">
      <SetFormsForListItemResult>
        <ApplicationId>InfoPath 14</ApplicationId>
        <Lcid>1033</Lcid>
        <Categories>
          <Category>
            <Id>BrowserOptimization</Id>
            <Label>Browser Optimization</Label>
            <HideWarningsByDefault>true</HideWarningsByDefault>
          </Category>
          <Category>
            <Id>BrowserCompatibility</Id>
            <Label>Browser Compatibility (Verified on server)</Label>
            <HideWarningsByDefault>>false</HideWarningsByDefault>
          </Category>
        </Categories>
        <Messages />
      </SetFormsForListItemResult>
    </SetFormsForListItemResponse>
  </soap:Body>
</soap:Envelope>
```

The **ApplicationId** element value of "InfoPath 14" demonstrates the protocol server's pass-through of the value from the request's **applicationId** element.

The **Lcid** element value of "1033" indicates the protocol server is returning strings suitable for display to an end user in English (United States).

4.3.2 SetFormsForListItem Response with Message Elements (One or More Issues Found)

This protocol example details a response from the **SetFormsForListItem** operation with Message elements, indicating there was an issue with mapping the form template (.xsn) file to the content type. See section 4.3.1, SetFormsForListItem Request/Response indicating successful operations on a list, for a typical request message.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <SetFormsForListItemResponse
xmlns="http://schemas.microsoft.com/office/infopath/2007/formsServices">
```

```

<SetFormsForListItemResult>
  <ApplicationId>InfoPath 14</ApplicationId>
  <Lcid>1033</Lcid>
  <Categories>
    <Category>
      <Id>BrowserOptimization</Id>
      <Label>Browser Optimization</Label>
      <HideWarningsByDefault>true</HideWarningsByDefault>
    </Category>
    <Category>
      <Id>BrowserCompatibility</Id>
      <Label>Browser Compatibility (Verified on server)</Label>
      <HideWarningsByDefault>>false</HideWarningsByDefault>
    </Category>
  </Categories>
  <Messages>
    <Message Id="278" Type="Error" Feature="Controls" Category="BrowserCompatibility">
      <ShortMessage>Invalid location for the xsn being published. </ShortMessage>
      <DetailedMessage>Invalid location for the xsn being published. </DetailedMessage>
    </Message>
  </Messages>
</SetFormsForListItemResult>
</SetFormsForListItemResponse>
</soap:Body>
</soap:Envelope>

```

The **Message** element demonstrates checks as specified in section [2.2.4.5](#), Message. It has the Message **Id** value "278", detailing an error that would prevent mapping the form template (.xsn) file to the content type.

4.4 GetListFormLocation Operation Examples

This example shows the use of the **GetListFormLocation** operation.

4.4.1 GetListFormLocation Request/Response

This protocol example details a request to the **GetListFormLocation** operation and the subsequent response pointing to the location of a form template (.xsn) file on the protocol server which is mapped to the content type requested by the protocol client.

The following is the request to the **GetListFormLocation** operation sent by the protocol client:

```

<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
  xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
  ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <GetListFormLocation
      xmlns='http://schemas.microsoft.com/office/infopath/2007/formsServices'>
      <lcid>1033</lcid>
      <listGuid>{6747CDB8-ADB5-4A5F-9323-AFCC3D329358}</listGuid>
      <contentTypeId>0x010500C8B78A06972AA34682872D813D9D0ED3</contentTypeId>
      <checkDesignPermissions>true</checkDesignPermissions>
    </GetListFormLocation>
  </SOAP-ENV:Body>

```

```
</SOAP-ENV:Envelope>
```

The **checkDesignPermissions** element value indicates that the protocol server should check for ManageLists WSS rights.

The following is the response from the **GetListFormLocation** operation sent by the protocol server.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetListFormLocationResponse
xmlns="http://schemas.microsoft.com/office/infopath/2007/formsServices">
      <GetListFormLocationResult>http://contoso/Lists/Links/Link/template.xsn</GetListFormLocationResult>
    </GetListFormLocationResponse>
  </soap:Body>
</soap:Envelope>
```

The **GetListFormLocationResult** element value of "http://contoso/Lists/Links/Link/template.xsn " represents the URL of a form template (.xsn) file on the protocol server which is mapped to the content type specified by the client protocol in the request.

4.5 SetSchemaChangesForList Operation Examples

This example shows the use of the **SetSchemaChangesForList** operation.

4.5.1 SetSchemaChangesForList Request/Response Indicating Successful Operations on a List

This protocol example details a request to the **SetSchemaChangesForList** operation and the subsequent response indicating the new field internal name mappings returned by the protocol server.

The following is the request to the **SetSchemaChangesForList** operation sent by the protocol client:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <SetSchemaChangesForList
xmlns='http://schemas.microsoft.com/office/infopath/2007/formsServices'>
      <lcid>1033</lcid>
      <listGuid>{3bbd50d4-524e-405a-b05f-22c15baac9a5}</listGuid>
      <contentTypeId>0x010041CAB86258A82E4BAFC7F3A9A8B00AA0</contentTypeId>
      <newFields>
        <Fields>
          <Field Type="Text" Name="title" DisplayName="title" MaxLength="255" Required="FALSE"
><Default></Default></Field>
          <Field Type="Note" Name="field6" DisplayName="field6" AppendOnly="FALSE"
RichText="TRUE" RichTextMode="FullHtml" Required="FALSE" ></Field>
```

```

        <Field Type="Choice" Name="field7" DisplayName="field7" Required="FALSE"
FillInChoice="FALSE" ><Default></Default><CHOICES><CHOICE>Enter Choice
#1</CHOICE><CHOICE>Enter Choice #2</CHOICE><CHOICE>Enter Choice #3</CHOICE></CHOICES></Field>
    </Fields>
    </newFields>
    <updateFields>
<Fields>
    <Field Type="Text" Name="Title" DisplayName="Title" MaxLength="255" Required="TRUE"
><Default></Default></Field><Field Type="Text" Name="field4" DisplayName="field4"
MaxLength="255" Required="FALSE" ><Default></Default></Field>
    </Fields>
    </updateFields>
    <deleteFields>
<Fields>
    <Field Name="field1" ></Field><Field Name="field2" ></Field>
    </Fields>
    </deleteFields>
</SetSchemaChangesForList>
</SOAP-ENV:Body>
</SOAP-ENV:Envelope>

```

The **newFields**, **updateField** and **deleteFields** element values represent the field operations that the protocol server will attempt to perform on the list.

The following is the response from the **SetSchemaChangesForList** operation sent by the protocol server.

```

<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
xmlns:xsd="http://www.w3.org/2001/XMLSchema">
    <soap:Body>
        <SetSchemaChangesForListResponse
xmlns="http://schemas.microsoft.com/office/infopath/2007/formsServices">
            <SetSchemaChangesForListResult>
                <Fields>
                    <newFields>
                        <Field Name="title" NewName="title0"></Field>
                    </newFields>
                </Fields>
            </SetSchemaChangesForListResult>
        </SetSchemaChangesForListResponse>
    </soap:Body>
</soap:Envelope>

```

The **NewName** element value of "title0" indicates that the protocol server is returning the new field internal name assigned for the requested field **Name** element value of "title".

4.6 GetUserCodeDeploymentDependencies Operation Examples

This example shows the use of the **GetUserCodeDeploymentDependencies** operation.

4.6.1 GetUserCodeDeploymentDependencies Request/Response Indicating that the Form Template (.xsn) File with Code Can Be Browser-Enabled as a Sandboxed Solution

This protocol example details a request to the **GetUserCodeDeploymentDependencies** operation and the subsequent response indicating the form template (.xsn) file can be browser-enabled as a sandboxed solution.

The following is the request to the **GetUserCodeDeploymentDependencies** operation sent by the protocol client:

```
<?xml version="1.0" encoding="UTF-8" standalone="no"?>
<SOAP-ENV:Envelope xmlns:SOAPSDK1="http://www.w3.org/2001/XMLSchema"
  xmlns:SOAPSDK2="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:SOAPSDK3="http://schemas.xmlsoap.org/soap/encoding/" xmlns:SOAP-
  ENV="http://schemas.xmlsoap.org/soap/envelope/">
  <SOAP-ENV:Body>
    <GetUserCodeDeploymentDependencies
      xmlns='http://schemas.microsoft.com/office/infopath/2007/formsServices'>
    <siteCollectionLocation>http://contoso/site</siteCollectionLocation>
      </GetUserCodeDeploymentDependencies>
    </SOAP-ENV:Body>
  </SOAP-ENV:Envelope>
```

The **siteCollectionLocation** element value of `http://contoso/site` represents the URL of the site collection the protocol client requests to verify that sandboxed solution activation is permitted.

The following is the response from the **GetUserCodeDeploymentDependencies** operation sent by the protocol server. The **GetUserCodeDeploymentDependenciesResponse** element has the value **ActivationAllowed** that indicates that the form template (.xsn) file with code can be browser-enabled as a sandboxed solution.

```
<?xml version="1.0" encoding="utf-8"?>
<soap:Envelope xmlns:soap="http://schemas.xmlsoap.org/soap/envelope/"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xsd="http://www.w3.org/2001/XMLSchema">
  <soap:Body>
    <GetUserCodeDeploymentDependenciesResponse
      xmlns="http://schemas.microsoft.com/office/infopath/2007/formsServices">
    <GetUserCodeDeploymentDependenciesResult>ActivationAllowed</GetUserCodeDeploymentDependencies
    Result>
      </GetUserCodeDeploymentDependenciesResponse>
    </soap:Body>
  </soap:Envelope>
```

5 Security

5.1 Security Considerations for Implementers

In addition to the security considerations applicable to the underlying protocols, a protocol server implementation of **DesignCheckFormTemplate** operation could involve expensive processing operations that need to be appropriately protected against denial of service attacks. The implementer could use an authorization policy to ensure that the Web service protocol cannot be used to mount this type of attack.

A protocol server implementation should consider if it needs to execute compiled or interpreted code on the protocol server during the Forms Services Design and Activation Web Service Protocol. This is dangerous because it can allow any user to use this protocol to execute un-trusted code in the server process.

5.2 Index of Security Parameters

None.

6 Appendix A: Full WSDL

For ease of implementation, the full WSDL is provided below:

```
<?xml version="1.0" encoding="utf-8" ?>
<wsdl:definitions xmlns:soap="http://schemas.xmlsoap.org/wsdl/soap/"
xmlns:tns="http://schemas.microsoft.com/office/infopath/2007/formsServices"
xmlns:s="http://www.w3.org/2001/XMLSchema"
xmlns:soap12="http://schemas.xmlsoap.org/wsdl/soap12/"
targetNamespace="http://schemas.microsoft.com/office/infopath/2007/formsServices"
xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/">
  <wsdl:documentation xmlns:wSDL="http://schemas.xmlsoap.org/wsdl/">Forms
Service</wsdl:documentation>
<wsdl:types>
<s:schema elementFormDefault="qualified"
targetNamespace="http://schemas.microsoft.com/office/infopath/2007/formsServices">
<s:element name="BrowserEnableUserFormTemplate">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="formTemplateLocation" type="s:string" />
  </s:sequence>
</s:complexType>
</s:element>
<s:element name="BrowserEnableUserFormTemplateResponse">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="BrowserEnableUserFormTemplateResult"
type="tns:MessagesResponse" />
  </s:sequence>
</s:complexType>
</s:element>
<s:complexType name="MessagesResponse">
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="Messages" type="tns:Messages" />
  </s:sequence>
</s:complexType>
<s:complexType name="Messages">
<s:sequence>
  <s:element minOccurs="0" maxOccurs="unbounded" name="Message" nillable="true"
type="tns:Message" />
  </s:sequence>
</s:complexType>
<s:complexType name="Message">
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="ShortMessage" type="s:string" />
  <s:element minOccurs="0" maxOccurs="1" name="DetailedMessage" type="s:string" />
  <s:element minOccurs="0" maxOccurs="1" name="SourceLocation" type="tns:SourceLocation" />
  </s:sequence>
  <s:attribute name="Id" type="s:int" use="required" />
  <s:attribute name="Type" type="tns:MessageType" use="required" />
  <s:attribute name="Feature" type="tns:Feature" use="required" />
  <s:attribute name="Category" type="tns:Category" use="required" />
</s:complexType>
<s:complexType name="SourceLocation">
  <s:attribute name="ControlId" type="s:string" />
  <s:attribute name="FileName" type="s:string" />
  <s:attribute name="LineNumber" type="s:int" />
  <s:attribute name="LinePosition" type="s:int" />
</s:complexType>
```

```

<s:simpleType name="MessageType">
<s:restriction base="s:string">
  <s:enumeration value="Error" />
  <s:enumeration value="Information" />
  <s:enumeration value="Warning" />
</s:restriction>
</s:simpleType>
<s:simpleType name="Feature">
<s:restriction base="s:string">
  <s:enumeration value="GenericXsf" />
  <s:enumeration value="XsfSchema" />
  <s:enumeration value="GenericXsl" />
  <s:enumeration value="GenericXPath" />
  <s:enumeration value="TemplateXml" />
  <s:enumeration value="Layout" />
  <s:enumeration value="Controls" />
  <s:enumeration value="BusinessLogic" />
  <s:enumeration value="Calculations" />
  <s:enumeration value="Validation" />
  <s:enumeration value="DigitalSignatures" />
  <s:enumeration value="DataAdapters" />
  <s:enumeration value="Submit" />
  <s:enumeration value="Views" />
  <s:enumeration value="Rules" />
  <s:enumeration value="ConditionalFormatting" />
  <s:enumeration value="VersionUpgrade" />
</s:restriction>
</s:simpleType>
<s:simpleType name="UserSolutionActivationStatus">
<s:restriction base="s:string">
  <s:enumeration value="ActivationAllowed" />
  <s:enumeration value="ActivationDenied" />
  <s:enumeration value="UserCodeServiceNotAvailable" />
  <s:enumeration value="ServiceQuotaExceeded" />
  <s:enumeration value="InsufficientUserPermissions" />
</s:restriction>
</s:simpleType>
<s:simpleType name="Category">
<s:restriction base="s:string">
  <s:enumeration value="BrowserOptimization" />
  <s:enumeration value="BrowserCompatibility" />
</s:restriction>
</s:simpleType>
<s:element name="DesignCheckFormTemplate">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="loid" type="s:int" />
  <s:element minOccurs="1" maxOccurs="1" name="base64FormTemplate" type="s:string" />
  <s:element minOccurs="1" maxOccurs="1" name="applicationId" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
<s:element name="DesignCheckFormTemplateResponse">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="DesignCheckFormTemplateResult"
type="tns:DesignCheckerInformation" />
</s:sequence>
</s:complexType>

```

```

    </s:element>
<s:element name="SetFormsForListItem">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="lcid" type="s:int" />
  <s:element minOccurs="1" maxOccurs="1" name="base64FormTemplate" type="s:string" />
  <s:element minOccurs="0" maxOccurs="1" name="applicationId" type="s:string" />
  <s:element minOccurs="1" maxOccurs="1" name="listGuid" type="s:string" />
  <s:element minOccurs="1" maxOccurs="1" name="contentTypeId" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
<s:element name="SetFormsForListItemResponse">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="SetFormsForListItemResult"
type="tns:DesignCheckerInformation" />
</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetListFormLocation">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="lcid" type="s:int" />
  <s:element minOccurs="1" maxOccurs="1" name="listGuid" type="s:string" />
  <s:element minOccurs="1" maxOccurs="1" name="contentTypeId" type="s:string" />
  <s:element minOccurs="1" maxOccurs="1" name="checkDesignPermissions" type="s:boolean" />
</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetListFormLocationResponse">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="GetListFormLocationResult" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
<s:element name="SetSchemaChangesForList">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="lcid" type="s:int" />
  <s:element minOccurs="1" maxOccurs="1" name="listGuid" type="s:string" />
  <s:element minOccurs="1" maxOccurs="1" name="contentTypeId" type="s:string" />
  <s:element minOccurs="1" maxOccurs="1" name="newFields">
<s:complexType mixed="true">
<s:sequence>
  <s:any />
</s:sequence>
</s:complexType>
</s:element>
  <s:element minOccurs="0" maxOccurs="1" name="updateFields">
<s:complexType mixed="true">
<s:sequence>
  <s:any />
</s:sequence>
</s:complexType>
</s:element>
  <s:element minOccurs="0" maxOccurs="1" name="deleteFields">
<s:complexType mixed="true">

```

```

<s:sequence>
  <s:any />
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="SetSchemaChangesForListResponse">
<s:complexType>
<s:sequence>
<s:element minOccurs="1" maxOccurs="1" name="SetSchemaChangesForListResult">
<s:complexType mixed="true">
<s:sequence>
  <s:any />
</s:sequence>
</s:complexType>
</s:element>
</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetUserCodeDeploymentDependencies">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="siteCollectionLocation" type="s:string" />
</s:sequence>
</s:complexType>
</s:element>
<s:element name="GetUserCodeDeploymentDependenciesResponse">
<s:complexType>
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="GetUserCodeDeploymentDependenciesResult"
type="tns:UserSolutionActivationStatus" />
</s:sequence>
</s:complexType>
</s:element>
<s:complexType name="DesignCheckerInformation">
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="ApplicationId" type="s:string" />
  <s:element minOccurs="1" maxOccurs="1" name="Lcid" type="s:int" />
  <s:element minOccurs="1" maxOccurs="1" name="Categories" type="tns:Categories" />
  <s:element minOccurs="1" maxOccurs="1" name="Messages" type="tns:Messages" />
</s:sequence>
</s:complexType>
<s:complexType name="Categories">
<s:sequence>
  <s:element minOccurs="1" maxOccurs="unbounded" name="Category" nillable="true"
type="tns:CategoryType" />
</s:sequence>
</s:complexType>
<s:complexType name="CategoryType">
<s:sequence>
  <s:element minOccurs="1" maxOccurs="1" name="Id" type="tns:Category" />
  <s:element minOccurs="1" maxOccurs="1" name="Label" type="s:string" />
  <s:element minOccurs="1" maxOccurs="1" name="HideWarningsByDefault" type="s:boolean" />
</s:sequence>
</s:complexType>
</s:schema>
</wsdl:types>

```

```

<wsdl:message name="BrowserEnableUserFormTemplateSoapIn">
  <wsdl:part name="parameters" element="tns:BrowserEnableUserFormTemplate" />
</wsdl:message>
<wsdl:message name="BrowserEnableUserFormTemplateSoapOut">
  <wsdl:part name="parameters" element="tns:BrowserEnableUserFormTemplateResponse" />
</wsdl:message>
<wsdl:message name="DesignCheckFormTemplateSoapIn">
  <wsdl:part name="parameters" element="tns:DesignCheckFormTemplate" />
</wsdl:message>
<wsdl:message name="DesignCheckFormTemplateSoapOut">
  <wsdl:part name="parameters" element="tns:DesignCheckFormTemplateResponse" />
</wsdl:message>
<wsdl:message name="SetFormsForListItemSoapIn">
  <wsdl:part name="parameters" element="tns:SetFormsForListItem" />
</wsdl:message>
<wsdl:message name="SetFormsForListItemSoapOut">
  <wsdl:part name="parameters" element="tns:SetFormsForListItemResponse" />
</wsdl:message>
<wsdl:message name="GetUserCodeDeploymentDependenciesSoapIn">
  <wsdl:part name="parameters" element="tns:GetUserCodeDeploymentDependencies" />
</wsdl:message>
<wsdl:message name="GetUserCodeDeploymentDependenciesSoapOut">
  <wsdl:part name="parameters" element="tns:GetUserCodeDeploymentDependenciesResponse" />
</wsdl:message>
<wsdl:message name="GetListFormLocationSoapIn">
  <wsdl:part name="parameters" element="tns:GetListFormLocation" />
</wsdl:message>
<wsdl:message name="GetListFormLocationSoapOut">
  <wsdl:part name="parameters" element="tns:GetListFormLocationResponse" />
</wsdl:message>
<wsdl:message name="SetSchemaChangesForListSoapIn">
  <wsdl:part name="parameters" element="tns:SetSchemaChangesForList" />
</wsdl:message>
<wsdl:message name="SetSchemaChangesForListSoapOut">
  <wsdl:part name="parameters" element="tns:SetSchemaChangesForListResponse" />
</wsdl:message>
<wsdl:portType name="FormsServicesWebServiceSoap">
<wsdl:operation name="BrowserEnableUserFormTemplate">
  <wsdl:input message="tns:BrowserEnableUserFormTemplateSoapIn" />
  <wsdl:output message="tns:BrowserEnableUserFormTemplateSoapOut" />
</wsdl:operation>
<wsdl:operation name="DesignCheckFormTemplate">
  <wsdl:input message="tns:DesignCheckFormTemplateSoapIn" />
  <wsdl:output message="tns:DesignCheckFormTemplateSoapOut" />
</wsdl:operation>
<wsdl:operation name="SetFormsForListItem">
  <wsdl:input message="tns:SetFormsForListItemSoapIn" />
  <wsdl:output message="tns:SetFormsForListItemSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetListFormLocation">
  <wsdl:input message="tns:GetListFormLocationSoapIn" />
  <wsdl:output message="tns:GetListFormLocationSoapOut" />
</wsdl:operation>
<wsdl:operation name="SetSchemaChangesForList">
  <wsdl:input message="tns:SetSchemaChangesForListSoapIn" />
  <wsdl:output message="tns:SetSchemaChangesForListSoapOut" />
</wsdl:operation>
<wsdl:operation name="GetUserCodeDeploymentDependencies">
  <wsdl:input message="tns:GetUserCodeDeploymentDependenciesSoapIn" />

```

```

    <wsdl:output message="tns:GetUserCodeDeploymentDependenciesSoapOut" />
  </wsdl:operation>
</wsdl:portType>
<wsdl:binding name="FormsServicesWebServiceSoap" type="tns:FormsServicesWebServiceSoap">
  <soap:binding transport="http://schemas.xmlsoap.org/soap/http" />
<wsdl:operation name="BrowserEnableUserFormTemplate">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/BrowserEnableUser
FormTemplate" style="document" />
<wsdl:input>
  <soap:body use="literal" />
</wsdl:input>
<wsdl:output>
  <soap:body use="literal" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="DesignCheckFormTemplate">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/DesignCheckFormTe
mplate" style="document" />
<wsdl:input>
  <soap:body use="literal" />
</wsdl:input>
<wsdl:output>
  <soap:body use="literal" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetFormsForListItem">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/SetFormsForListIt
em" style="document" />
<wsdl:input>
  <soap:body use="literal" />
</wsdl:input>
<wsdl:output>
  <soap:body use="literal" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetListFormLocation">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/GetListFormLocati
on" style="document" />
<wsdl:input>
  <soap:body use="literal" />
</wsdl:input>
<wsdl:output>
  <soap:body use="literal" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="SetSchemaChangesForList">
  <soap:operation
soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/SetSchemaChangesF
orList" style="document" />
<wsdl:input>
  <soap:body use="literal" />
</wsdl:input>
<wsdl:output>
  <soap:body use="literal" />
</wsdl:output>
</wsdl:operation>

```

```

<wsdl:operation name="GetUserCodeDeploymentDependencies">
  <soap:operation
    soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/GetUserCodeDeploymentDependencies" style="document" />
  <wsdl:input>
    <soap:body use="literal" />
  </wsdl:input>
  <wsdl:output>
    <soap:body use="literal" />
  </wsdl:output>
</wsdl:operation>
</wsdl:binding>
<wsdl:binding name="FormsServicesWebServiceSoap12" type="tns:FormsServicesWebServiceSoap">
  <soap12:binding transport="http://schemas.xmlsoap.org/soap/http" />
  <wsdl:operation name="BrowserEnableUserFormTemplate">
    <soap12:operation
      soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/BrowserEnableUserFormTemplate" style="document" />
    <wsdl:input>
      <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="DesignCheckFormTemplate">
    <soap12:operation
      soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/DesignCheckFormTemplate" style="document" />
    <wsdl:input>
      <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="SetFormsForListItem">
    <soap12:operation
      soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/SetFormsForListItem" style="document" />
    <wsdl:input>
      <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="GetListFormLocation">
    <soap12:operation
      soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/GetListFormLocation" style="document" />
    <wsdl:input>
      <soap12:body use="literal" />
    </wsdl:input>
    <wsdl:output>
      <soap12:body use="literal" />
    </wsdl:output>
  </wsdl:operation>
  <wsdl:operation name="SetSchemaChangesForList">

```

```
<soap12:operation
soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/SetSchemaChangesForList" style="document" />
<wsdl:input>
  <soap12:body use="literal" />
</wsdl:input>
<wsdl:output>
  <soap12:body use="literal" />
</wsdl:output>
</wsdl:operation>
<wsdl:operation name="GetUserCodeDeploymentDependencies">
  <soap12:operation
soapAction="http://schemas.microsoft.com/office/infopath/2007/formsServices/GetUserCodeDeploymentDependencies" style="document" />
<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 Messages

The contents of this appendix comprise messages returned to the protocol client by the protocol server.

7.1 Messages for Office InfoPath 2007 Forms

Messages described in this section are generated by InfoPath® Forms Services and Microsoft® SharePoint® Server 2010 when design checking a form template (.xsn) file as specified in [\[MS-IPFF\]](#).

7.1.1 Message Elements of Type "Error"

ID	Category	Feature	Message	Description
1	BrowserCompatibility	Controls	The form template is not browser-compatible. It might be possible to correct the problem by opening the form template in Microsoft® Office InfoPath® 2007, and then republishing it.	The protocol server returns this error message when the form template file cannot be opened. This error will occur when the file is invalid or corrupt.
5	BrowserCompatibility	Controls	Error loading form code assembly: <i>[name of the user code module]</i> .	As specified in [MS-IPFF] section 2.2.147.50
6	BrowserCompatibility	Controls	A rule references a button which is missing from the view.	The protocol server returns this error message when a rule is associated with a button control that is not part of the form view.
12	BrowserCompatibility	Controls	Button <i>[button name]</i> has a value of the xd:action attribute that is not supported in browser-enabled form templates.	As specified in [MS-IPFF] section 2.4.2.1.
13	BrowserCompatibility	Controls	Expected anchor tag is missing.	As specified in [MS-IPFF] section 2.4.1.12.
14	BrowserCompatibility	Controls	List Box control is not supported.	As specified in [MS-IPFF] section 2.4.1.13.
15	BrowserCompatibility	Controls	Choice Section control is not supported.	As specified in [MS-IPFF] section 2.4.1.21.
16	BrowserCompatibility	Controls	Combo Box control is not supported.	As specified in [MS-IPFF] section 2.4.1.21.
18	BrowserCompatibility	Controls	<i>[Control Name]</i>	As specified in [MS-IPFF] section

ID	Category	Feature	Message	Description
	lity		control is not supported.	2.2.108.
21	BrowserCompatibility	Controls	Picture control is not supported.	As specified in [MS-IPFF] section 2.4.1.22.
22	BrowserCompatibility	Controls	Ink Picture control is not supported.	As specified in [MS-IPFF] section 2.4.1.21.
23	BrowserCompatibility	Controls	Horizontal Region control is not supported.	As specified in [MS-IPFF] section 2.4.1.21.
24	BrowserCompatibility	Controls	Master/detail control is not supported.	As specified in [MS-IPFF] section 2.4.2
25	BrowserCompatibility	Controls	Multiple-Selection List Box control is not supported.	As specified in [MS-IPFF] section 2.4.1.21.
26	BrowserCompatibility	Controls	Recursive control not supported.	As specified in [MS-IPFF] section 2.4.1.22.
27	BrowserCompatibility	Controls	Choice Section control is not supported.	As specified in [MS-IPFF] section 2.4.1.22.
28	BrowserCompatibility	Controls	Scrolling Region control is not supported.	As specified in [MS-IPFF] section 2.4.1.21.
30	BrowserCompatibility	Controls	Vertical text is not supported.	As specified in [MS-IPFF] section 2.4.1.21.
31	BrowserCompatibility	Controls	Controls that repeat horizontally are not supported.	As specified in [MS-IPFF] section 2.4.1.21.
40	BrowserCompatibility	Controls	Encountered an encoded identifier <i>[Identifier name]</i> . Encoded identifiers are not supported.	The protocol server returns this error message when parsing the cascading style sheet (CSS) associated with a form view and the CSS contains an encoded identifier.
45	BrowserCompatibility	Controls	The form locale <i>[locale identifier]</i> is not a valid language identifier. Assign a valid locale and try again.	As specified in [MS-IPFF] section 2.2.147.9.
46	BrowserCompatibility	Controls	An XSL file is missing from the form template.	As specified in [MS-IPFF] section 2.2.129.
57	BrowserCompatibility	Controls	Linked images are not supported by InfoPath® Forms Services. To fix this	The protocol server returns this error message when a form view contains an img element with a href attribute with a value of an

ID	Category	Feature	Message	Description
			problem, make the image a part of the form template. The following image was found in view <i>[view name]</i> : <i>[image name]</i> .	absolute URL.
58	BrowserCompatibility	Controls	Unexpected image tag encountered. It is missing a required href attribute.	The protocol server returns this error message when detecting an img element without a href attribute in a form view.
59	BrowserCompatibility	Controls	Only the .png, .gif, and .jpg graphics file formats are supported.	The protocol server returns this error message when the href attribute of an img element in a form view refers a file with an unsupported extension. The Office InfoPath 2007 supports the file extensions.png, .gif, and .jpg.
60	BrowserCompatibility	Controls	The data source <i>[data source name]</i> referenced in the form template is not valid or cannot be found.	As specified in [MS-IPFF] section 2.4.3.9.2.
61	BrowserCompatibility	Controls	Invalid format for location attribute <i>[attribute value]</i> .	As specified in [MS-IPFF] section 2.2.61.
62	BrowserCompatibility	Controls	An eval expression used in the form has invalid arguments.	As specified in [MS-IPFF] section 2.4.3.6.2.
63	BrowserCompatibility	Controls	The XPath <i>[xpath expression]</i> is invalid. <i>[error details]</i>	As specified in [XPATHT] .
64	BrowserCompatibility	Controls	The following file is not a valid Xml Schema: <i>[name of the schema file]</i>	As specified in [XMLSCHEMA1] .
67	BrowserCompatibility	Controls	The following file is either missing or is not part of included in the form template: <i>[filename]</i> . To add a file to the form template in design mode, use the Resource Files dialog box on the Tools menu, and then add	As specified in [MS-IPFF] section 2.2.98.

ID	Category	Feature	Message	Description
			the file.	
68	BrowserCompatibility	Controls	Multiple <code>xsl:template</code> 's found with name corresponding to this <code>xsl:call-template</code> . Using the first <code>xsl:template</code> found.	The protocol server returns this error message when a form view file contains more than one XSL template element with the same value for the name attribute.
69	BrowserCompatibility	Controls	Did not find an <code>xsl:template</code> with the same name as this <code>xsl:call-template</code> . Ignoring the <code>xsl:call-template</code> .	As specified in [W3C-XSLT] .
71	BrowserCompatibility	Controls	Multiple <code>xsl:templates</code> found with mode corresponding to this <code>xsl:apply-templates</code> . Using the first <code>xsl:template</code> found.	The protocol server returns this error message when a form view file contains an apply-templates element that can find one or more xsl:template element to apply with the same mode attribute.
72	BrowserCompatibility	Controls	The <code>xsl:stylesheet</code> element had multiple <code>xsl:template</code> elements with no mode attribute. This is not supported. Using only the first of these <code>xsl:templates</code> .	As specified in [MS-IPFF] section 2.4.1.2.
73	BrowserCompatibility	Controls	The mode of this <code>xsl:apply-templates</code> element does not match the mode attribute of any <code>xsl:template</code> element.	As specified in [MS-IPFF] sections 2.4.1.15 and 2.4.1.18 .
74	BrowserCompatibility	Controls	Could not find an <code>xsl:template</code> without a mode inside <code>xsl:stylesheet</code> .	As specified in [MS-IPFF] section 2.4.1.2 or in [MS-IPFF2] section 2.4.1.2.
75	BrowserCompatibility	Controls	The following construct is not supported: <code>'xsl:apply-imports'</code> .	As specified in [MS-IPFF] section 2.4.1 or in [MS-IPFF2] section 2.4.1.
76	BrowserCompatibility	Controls	The following construct is not supported: <code>'xsl:import'</code> .	As specified in [MS-IPFF] section 2.4.1.
77	BrowserCompatibility	Controls	The following	As specified in [MS-IPFF] section

ID	Category	Feature	Message	Description
	lity		construct is not supported: 'xsl:include'.	2.4.1.
79	BrowserCompatibily	Controls	An unexpected error has occurred while verifying the form template.	As specified in [MS-IPFF] section 2.1.
87	BrowserCompatibily	Calculations	The following expression could not be parsed because of a syntax error or because it uses an undefined namespace prefix or unsupported function: <i>[XPath expression]</i> .	As specified in [MS-IPFF] section 2.2.146, target and expression attributes.
88	BrowserCompatibily	ConditionalFormattin g	Unsupported expression	As specified in [MS-IPFF] section 2.4.1.1.
89	BrowserCompatibily	Controls	Unsupported expression	As specified in [MS-IPFF] section 2.4.1.1.
90	BrowserCompatibily	DataAdapters	Unsupported expression	As specified in [MS-IPFF] section 2.2.44.
91	BrowserCompatibily	DigitalSignatures	Unsupported expression	As specified in [MS-IPFF] section 2.2.126, data and signatureLocation attributes.
92	BrowserCompatibily	GenericXsl	Unsupported expression	As specified in [MS-IPFF] section 2.4.1.
93	BrowserCompatibily	Rules	Unsupported expression	As specified in [MS-IPFF] section 2.2.133.
94	BrowserCompatibily	Controls	Unsupported expression	As specified in [MS-IPFF] section 2.2.63, expression and expressionContext attributes.
95	BrowserCompatibily	Controls	Unsupported expression	As specified in [MS-IPFF] sections 2.2.124 and 2.2.107 .
97	BrowserCompatibily	Controls	Browser-enabled form templates must have at least one browser-compatible view.	As specified in [MS-IPFF] section 2.2.122.
98	BrowserCompatibily	Controls	<i>[Detailed error message the XML Processor]</i>	As specified in [MS-IPFF] section 2.2.
99	BrowserCompatibily	Controls	Unexpected schema validation error: <i>[Schema Error]</i>	As specified in [MS-IPFF] section 2.2.60.

ID	Category	Feature	Message	Description
100	BrowserCompatibility	Controls	[Detailed error message the XML Processor]	As specified in [MS-IPFF] section 2.2.
102	BrowserCompatibility	Controls	HWS data connections are not supported.	As specified in [MS-IPFF] section 2.2.40.
103	BrowserCompatibility	Controls	HWS task pane is not supported.	As specified in [MS-IPFF] section 2.2.87.
111	BrowserCompatibility	Controls	The form template is not browser-compatible, perhaps as a result of modifications made outside of Office InfoPath 2007. It might be possible to correct the problem by republishing the form template from within Office InfoPath 2007.	As specified in [MS-IPFF] section 2.2.147.8, runtimeCompatibility attribute.
112	BrowserCompatibility	Controls	Character expected. Found an empty value.	As specified in [MS-IPFF] section 2.4.1.
113	BrowserCompatibility	Controls	Character expected. Found a string value.	As specified in [MS-IPFF] section 2.4.1.1.
114	BrowserCompatibility	Controls	Double value expected. [value] is not a valid double.	As specified in [MS-IPFF] section 2.4.1.
115	BrowserCompatibility	Controls	Invalid Uri found as value of href attribute of xsl:import.	The protocol server returns this error message when parsing the XSL of a form view file. The href attribute of an xsl:import element uses an invalid URI value.
116	BrowserCompatibility	Controls	Invalid Uri found as value of href attribute of xsl:include.	The protocol server returns this error message when parsing the XSL of a form view file. The href attribute of an xsl:include element uses an invalid URI value.
117	BrowserCompatibility	Controls	Invalid value: [value] for grouping-size attribute of xsl:number. The value must be a positive integer value.	The protocol server returns this error message when parsing the XSL of a form view file. The grouping-size attribute xsl:number element has an invalid value.
11	BrowserCompatibility	Controls	Invalid value:	The protocol server returns this

ID	Category	Feature	Message	Description
8	lity		<i>[value]</i> for letter-value attribute of xsl:number. Valid values are: 'alphabetic' and 'traditional'.	error message when parsing the XSL of a form view file. The letter-value attribute of an xsl:number element has a value different from "alphabetic" or "traditional".
119	BrowserCompatibility	Controls	Invalid value: <i>[value]</i> for level attribute of xsl:number. Valid values are: 'single', 'multiple' and 'any'.	The protocol server returns this error message when parsing the XSL of a form view file. The level attribute of an xsl:number element has a value different from "single", "multiple" or "any".
120	BrowserCompatibility	Controls	Invalid value: <i>[value]</i> for case-order attribute of xsl:sort. Valid values are: 'upper-first' and 'lower-first'.	The protocol server returns this error message when parsing the XSL of a form view file. The case-order attribute of an xsl:sort element has a value different from "upper-first" or "lower-first".
121	BrowserCompatibility	Controls	Invalid value: <i>[value]</i> for order attribute of xsl:sort. Valid values are: 'ascending' and 'descending'.	The protocol server returns this error message when parsing the XSL of a form view file. The order attribute of an xsl:sort element has a value different from "ascending" or "descending".
122	BrowserCompatibility	Controls	xsl:import encountered after other xsl constructs.	The protocol server returns this error message when parsing the XSL of a form view file. The parser found an xsl:import element that is not at the top level of the XSL document.
123	BrowserCompatibility	Controls	Integer value expected. <i>[value]</i> is not a valid integer.	The protocol server returns this error message when parsing the XSL of a form view file. The parser found an invalid value where an integer value was expected.
124	BrowserCompatibility	Controls	The qualified name <i>[name]</i> is not valid.	As specified in [MS-IPFF] section 2.4.1.
125	BrowserCompatibility	Controls	The qualified name <i>[original value]</i> is not a valid. Using qualified name <i>[replace value]</i> .	As specified in [MS-IPFF] section 2.4.1.
126	BrowserCompatibility	Controls	Xsl for view <i>[view name]</i> is not a valid xml file. <i>[Detailed error message the XML Processor]</i>	As specified in [MS-IPFF] section 2.4.

ID	Category	Feature	Message	Description
127	BrowserCompatibility	Controls	Multiple <code>xsl:otherwise</code> elements in an <code>xsl:when</code> are not supported.	As specified in [W3C-XSLT] section 9.2.
128	BrowserCompatibility	Controls	<code>xsl:template</code> found with no <code>match</code> attribute and no <code>name</code> attribute. At least one must be specified.	As specified in [W3C-XSLT] section 5.3, 6
129	BrowserCompatibility	Controls	If an <code>xsl:template</code> element does not have a <code>match</code> attribute, it must not have a <code>mode</code> attribute.	As specified in [W3C-XSLT] section 5.7
130	BrowserCompatibility	Controls	<code>xsl:otherwise</code> element found without a corresponding <code>xsl:when</code> element.	As specified in [W3C-XSLT] section 9.2
131	BrowserCompatibility	Controls	<code>xsl:param</code> encountered after other <code>xsl</code> constructs.	As specified in [W3C-XSLT] section 15
132	BrowserCompatibility	Controls	The prefix <i>[namespace prefix]</i> is not declared in the current scope.	As specified in [W3C-XSLT] section 2.1
133	BrowserCompatibility	Controls	Unexpected attribute <i>{[namespace]}[attribute local-name]</i> encountered.	As specified in [W3C-XSLT] section 2.1
134	BrowserCompatibility	Controls	Unexpected element <i>{[namespace]}[element local-name]</i> encountered.	As specified in [W3C-XSLT] section 2.1
135	BrowserCompatibility	Controls	Unexpected node encountered.	As specified in [W3C-XSLT] section 2.1
136	BrowserCompatibility	Controls	Unexpected type of node <i>[node type]</i> encountered.	As specified in [W3C-XSLT] section 2.1
137	BrowserCompatibility	Controls	Unexpected child element for <code>xsl:for-each</code> . Only <code>xsl:sort</code> elements are allowed.	As specified in [W3C-XSLT] section 8
13	BrowserCompatibility	Controls	Unsupported value:	As specified in [MS-IPFF] section

ID	Category	Feature	Message	Description
8	lity		[value]' found for version attribute of xsl:stylesheet.	2.4.1.2.
139	BrowserCompatibility	Controls	Unexpected text encountered.	As specified in [MS-IPFF] section 2.4.1.22.
140	BrowserCompatibility	Controls	Unexpected value: [value]. Expected 'yes' or 'no'.	As specified in [W3C-XSLT] section B
141	BrowserCompatibility	Controls	Entity References are not supported.	As specified in [W3C-XSLT] section 2.4.
142	BrowserCompatibility	Controls	This element does not permit content if the select attribute is specified.	As specified in [W3C-XSLT] section 2.4.
143	BrowserCompatibility	Controls	Unexpected xsl:when element found after an xsl:otherwise.	As specified in [W3C-XSLT] section 9.2.
144	BrowserCompatibility	Controls	The form definition (.xsf) file has an invalid errorCondition -- errorMessage tag is missing.	As specified in [MS-IPFF] section 2.2.63, errorMessage
145	BrowserCompatibility	Controls	The form definition (.xsf) file has an invalid errorCondition -- errorMessage tag is missing.	As specified in [MS-IPFF] section 2.2.63, errorMessage element.
147	BrowserCompatibility	Controls	Duplicate data adapter name encountered.	As specified in [MS-IPFF] section 2.2.36, name attribute.
148	BrowserCompatibility	Controls	The specified onAfterSubmit action is not supported.	As specified in [MS-IPFF] section 2.2.72, onAfterSubmit attribute.
149	BrowserCompatibility	Controls	Invalid query or submit specification for data adapter.	As specified in [MS-IPFF] section 2.2.37.
150	BrowserCompatibility	Controls	Unsupported Data Adapter.	As specified in [MS-IPFF] section 2.2.59.
151	BrowserCompatibility	Controls	A partFragment tag is missing a match attribute.	As specified in [MS-IPFF] section 2.2.44, match attribute.
152	BrowserCompatibility	Controls	A partFragment tag is missing a replaceWith	As specified in [MS-IPFF] section 2.2.44, replaceWith attribute

ID	Category	Feature	Message	Description
			attribute.	
155	BrowserCompatibility	Controls	The operation failed. The data contained multiple DataSets.	The protocol server returns this message when the data source returns more than one dataset. This could happen when querying data from a [MS-SQL] database that supports multiple datasets.
158	BrowserCompatibility	Controls	One or more data connection library attributes is empty.	As specified in [MS-IPFF] section 2.2.147.30, connectionLinkType attribute.
160	BrowserCompatibility	Controls	Filters on Repeating Sections are not supported.	As specified in [MS-IPFF] section 2.4.1.15.
161	BrowserCompatibility	Controls	Filters on Repeating Tables are not supported.	As specified in [MS-IPFF] section 2.4.1.16.
162	BrowserCompatibility	Controls	The following HTML tag is not supported: 'Text Box'.	As specified in [MS-IPFF] section 2.4.2.11.
171	BrowserCompatibility	Controls	The XSL for the Date Picker is not in the expected form.	As specified in [MS-IPFF] section 2.4.1.8.
172	BrowserCompatibility	Controls	Invalid or malicious HTML was found in view <i>[view name]</i> .	The server returns this message when the body element in a form view is not valid HTML or if an attribute on that element contains a quote (") character.
173	BrowserCompatibility	Controls	Invalid or malicious HTML was found in view <i>[view name]</i> .	The protocol server returns this error message when a form view contains invalid or unsafe HTML elements or attributes, and the only SourceLocation information the protocol server is returning is the source file in which the issue occurs. Determining what HTML to consider unsafe is implementation-specific.
174	BrowserCompatibility	Controls	Invalid or malicious HTML was found in view <i>[view Name]</i> .	The protocol server returns this error message when the server cannot determine whether or not a form view contains invalid or unsafe HTML elements or attributes. Determining what HTML to consider unsafe is implementation-specific.
175	BrowserCompatibility	Controls	Invalid or malicious HTML was found in	The protocol server returns this error message when a form view

ID	Category	Feature	Message	Description
			view [view Name].	contains invalid or unsafe HTML elements or attributes, the SourceLocation information the protocol server is returning contains both the source file and line number in which the issue occurs, and the MessageType value for the corresponding Message is "Error". Determining what HTML to consider unsafe is implementation-specific.
176	BrowserCompatibility	Control	Invalid or malicious HTML was found in view [view Name].	The protocol server returns this error message when the server cannot determine whether or not a form view contains invalid or unsafe HTML elements or attributes. Determining what HTML to consider unsafe is implementation-specific.
177	BrowserCompatibility	TemplateXml	The form template cannot be browser-enabled because the template data is not valid according to its schema. [Error details]	As specified in [MS-IPFF] section 2.7.
178	BrowserCompatibility	Controls	A required parameter is missing for the data adapter [adapter name]: [XPath expression for the missing parameter]	As specified in [MS-IPFF] sections 2.2.39 and 2.2.41 .
179	BrowserCompatibility	Controls	A required parameter is missing for the data adapter [adapter name]: [XPath expression for the missing parameter]	As specified in [MS-IPFF] sections 2.2.39 and 2.2.41 .
180	BrowserCompatibility	Controls	Relative links to Data Connection Libraries located on different SharePoint site collection are not supported.	As specified in [MS-IPFF] section 2.2.147.30, siteCollection attribute.
183	BrowserCompatibility	Controls	Could not execute the following relative query [data adapter name]. Relative queries are not allowed for connections linked to	As specified in [MS-IPFF] section 2.2.147.33.

ID	Category	Feature	Message	Description
			the Data Connection Library.	
184	BrowserCompatibility	Controls	Cannot run the relative query <i>[query]</i> . Relative queries are not allowed for connections linked to the Data Connection Library.	As specified in [MS-IPFF] section 2.2.147.34.
185	BrowserCompatibility	Controls	relativeQuery/@replace cannot be a relative, local, or UNC path.	As specified in [MS-IPFF] section 2.2.147.34.
186	BrowserCompatibility	Controls	Unsupported HTML constructs were found associated with this Repeating Table.	As specified in [MS-IPFF] section 2.4.1.16.
191	BrowserCompatibility	Controls	A text box is bound to an inappropriate datatype.	As specified in [MS-IPFF] section 2.4.1.20.
192	BrowserCompatibility	Controls	Rule was referenced but not defined: <i>[rulename]</i> .	As specified in [MS-IPFF] section 2.2.132, ruleSet attribute.
193	BrowserCompatibility	Controls	There has been a critical error while processing the form.	The protocol server returns this error message protocol server when an unknown error was encountered and the protocol server does not have a default method for handling this unknown error.
196	BrowserCompatibility	DataAdapters	This database is not supported. The database must be a Microsoft® SQL Server®.	The protocol server returns this error message when there is a data adapter that attempts to connection to a database that is not supported by the protocol server .
197	BrowserCompatibility	BusinessLogic	Unsupported object model version.	As specified in [MS-IPFF] section 2.2.147.50, version attribute.
200	BrowserCompatibility	GenericXsf	The definition of view <i>[view name]</i> does not exist.	As specified in [MS-IPFF] sections 2.2.68 , 2.2.114 , and 2.2.129 , transform attribute.
203	BrowserCompatibility	GenericXPath	A call to GetDOM failed. The DataObject does not exist.	As specified in [MS-IPFF] section 2.4.3.9.2.
20	BrowserCompatibility	ExpressionXPathAnal	Invalid expression	As specified in [XPATHT] .

ID	Category	Feature	Message	Description
4	ility	ysis	encountered: <i>[XPath expression]</i> .	
205	BrowserCompatibility	ExpressionXPathAnalysis	Invalid binding XPath <i>[XPath expression]</i> .	As specified in [MS-IPFF] section 2.4.1.1, productions LEAF_XPATH, GROUP_XPATH, RELATIVE_REPEATING_GROUP_XPATH and RELATIVE_LEAF_XPATH.
206	BrowserCompatibility	ExpressionXPathAnalysis	Invalid binding XPath <i>[XPath expression]</i> .	As specified in [MS-IPFF] section 2.4.1.1, productions LEAF_XPATH, GROUP_XPATH, RELATIVE_REPEATING_GROUP_XPATH and RELATIVE_LEAF_XPATH.
207	BrowserCompatibility	ExpressionXPathAnalysis	Invalid binding XPath <i>[XPath expression]</i> .	As specified in [MS-IPFF] section 2.4.1.1 or in [MS-IPFF2] section 2.4.1.1, productions LEAF_XPATH, GROUP_XPATH, RELATIVE_REPEATING_GROUP_XPATH and RELATIVE_LEAF_XPATH.
208	BrowserCompatibility	ExpressionXPathAnalysis	Invalid binding XPath <i>[XPath expression]</i> .	As specified in [MS-IPFF] section 2.4.1.1, productions LEAF_XPATH, GROUP_XPATH, RELATIVE_REPEATING_GROUP_XPATH and RELATIVE_LEAF_XPATH.
209	BrowserCompatibility	ExpressionXPathAnalysis	Invalid binding XPath <i>[XPath expression]</i> .	As specified in [MS-IPFF] section 2.4.1.1, production LEAF_XPATH.
210	BrowserCompatibility	GenericXsf	The form template defines multiple schemas for the same namespace: <i>[namespace]</i>	As specified in [MS-IPFF] section 2.2.61, location attribute.
211	BrowserCompatibility	GenericXsf	Form template is not valid. The following schema file: <i>[filename]</i> was not found in the Form Template.	As specified in [MS-IPFF] section 2.2.98.
212	BrowserCompatibility	GenericXsf	The following DataObject either cannot be created or cannot be initialized: <i>[data objectName]</i> . The data adapter cannot be initialized. The form contains XML that cannot be parsed: <i>[Detailed</i>	As specified in [MS-IPFF] section 2.2.36, schema attribute.

ID	Category	Feature	Message	Description
			<i>error message the XML Processor</i>].	
213	BrowserCompatibility	GenericXsf	The specified XML template file (xsf:initialXmlDocument element) is not a valid XML document. <i>[Detailed error message the XML Processor]</i> .	As specified in [MS-IPFF] section 2.2.92.
214	BrowserCompatibility	GenericXsf	The specified XML template file (xsf:initialXmlDocument element) does not contain required processing instructions.	As specified in [MS-IPFF] section 2.2.94.
215	BrowserCompatibility	GenericXsf	The specified XML template file (xsf:initialXmlDocument element) is not a valid XML document. <i>[Detailed error message the XML Processor]</i>	As specified in [MS-IPFF] section 2.2.94.
216	BrowserCompatibility	GenericXsf	The XML template file (specified in the xsf:initialXmlDocument element of the form template definition file) has a processing instruction with a form template version that does not match the version of the form template.	As specified in [MS-IPFF] section 2.2.20, solutionVersion attribute and [MS-IPFF] section 2.7.
217	BrowserCompatibility	GenericXsf	The specified XML template file (xsf:initialXmlDocument element) contains a urn reference that does not match the solutions name.	As specified in [MS-IPFF] section 2.2.20, name attribute and [MS-IPFF] section 2.7.
218	BrowserCompatibility	GenericXsf	No XML template file is present in the form template.	As specified in [MS-IPFF] section 2.2.94, href attribute.
219	BrowserCompatibility	GenericXsf	The following XML template file is missing or is not part of the form	As specified in [MS-IPFF] section 2.2.94, href attribute.

ID	Category	Feature	Message	Description
			template: [file name]	
220	BrowserCompatibility	GenericXsf	Duplicate property name '[property name]' in file '[file name]'.	As specified in [MS-IPFF] section 2.2.100, name attribute.
221	BrowserCompatibility	GenericXsf	A button element in the form definition file (manifest.xsf) has invalid attributes.	As specified in [MS-IPFF] section 2.2.106 and 2.2.110 .
223	BrowserCompatibility	DigitalSignatures	An error occurred when initializing a set of signable data. A name has not been specified for the set of data.	As specified in [MS-IPFF] section 2.2.126, name attribute.
234	BrowserCompatibility	DigitalSignatures	No group was found at the location specified for storing signatures for the set of signable data: [XPath expression] . Use the Digital Signatures (2) category of the Form Options dialog box to edit the expression specifying the storage location for signatures.	As specified in [MS-IPFF] section 2.2.126, signatureLocation attribute.
235	BrowserCompatibility	DigitalSignatures	An error occurred when initializing the set of signable data: [XPath expression]. No group was found at the specified location.	As specified in [MS-IPFF] section 2.2.126, mode attribute.
236	BrowserCompatibility	DigitalSignatures	An error occurred when initializing the set of signable data: "[XPath expression] ". No group was found at the specified location.	As specified in [MS-IPFF] section 2.2.126, mode attribute.
237	BrowserCompatibility	DigitalSignatures	No fields or groups were found corresponding to the set of signable data: [XPath expression]. Use the Digital Signatures category of the Form Options	As specified in [MS-IPFF] section 2.2.126, data attribute.

ID	Category	Feature	Message	Description
			dialog box to edit the expression specifying the data to be signed.	
238	BrowserCompatibility	GenericXsl	No group was found at the location specified for storing signatures for the set of signable data: "[signed data block name]". Use the Digital Signatures category of the Form Options dialog box to edit the expression specifying the storage location for signatures.	As specified in [MS-IPFF] section 2.2.126, signatureLocation attribute.
239	BrowserCompatibility	GenericXsf	The following expression could not be parsed because of a syntax error or because it uses an undefined namespace prefix or unsupported function: <i>[XPath expression]</i> .	As specified in [MS-IPFF] section 2.4.1.
240	BrowserCompatibility	Controls	Numbered List control is not supported	As specified in [MS-IPFF] section 2.4.1.21, xctName attribute.
241	BrowserCompatibility	Controls	Bulleted List control is not supported	As specified in [MS-IPFF] section 2.4.1.21, xctName attribute.
242	BrowserCompatibility	Controls	Plain List control is not supported	As specified in [MS-IPFF] section 2.4.1.21, xctName attribute.
247	BrowserCompatibility	Controls	This control formats the data to show both date and time. Date and time cannot be displayed together in the same control in server forms. Use two controls to display date and time separately and bind them to the same field.	As specified in [MS-IPFF] section 2.4.2.11.
248	BrowserCompatibility	Controls	This control formats the data using an invalid or unsupported time	As specified in [MS-IPFF] section 2.4.2.11.

ID	Category	Feature	Message	Description
			format <i>[user format string]</i> . Pick a different format.	
249	BrowserCompatibility	Controls	This control formats the data using an invalid or unsupported date format <i>[user format string]</i> . Pick a different format.	As specified in [MS-IPFF] section 2.4.2.11.
250	BrowserCompatibility	GenericXsf	The restricted trust level is not supported.	As specified in [MS-IPFF] section 2.2.20, trustLevel attribute.
252	BrowserCompatibility	GenericXsf	This form template has not been correctly published to be browser-enabled. Open the form template in Office InfoPath 2007 Design mode, and click Publish Form Template in the Design Tasks task pane. Follow the steps in the Publishing Wizard to republish the form template, and then try again.	The protocol server returns this message when it encounters errors while trying to browser-enable a form template.
256	BrowserCompatibility	Controls	Selected rich text formatting options are not supported Selected rich text formatting options are not supported.	As specified in [MS-IPFF] section 2.2.147.43.
260	BrowserCompatibility	Controls	The following expression could not be parsed because of a syntax error or because it uses an undefined namespace prefix or unsupported function: <i>[XPath expression]</i> .	As specified in [MS-IPFF] section 2.4.1.
261	BrowserCompatibility	Controls	Specifying a restricted set of allowable file types for a File Attachment control is not supported in server forms.	As specified in [MS-IPFF] section 2.2.108, allowedFileTypes attribute.

ID	Category	Feature	Message	Description
262	BrowserCompatibility	Controls	Conditionally formatting a File Attachment control is not supported in server forms	As specified in [MS-IPFF] section 2.4.1.11.
267	BrowserCompatibility	GenericXsf	The form template is not browser-compatible, perhaps as a result of modifications made outside of Office InfoPath 2007. It might be possible to correct the problem by republishing the form template from within Office InfoPath 2007.	The protocol server returns this error message when the form template is not a browser-compatible form template and no more specific error message is applicable.
268	BrowserCompatibility	GenericXsf	Forms enabled for use on a mobile device are supported only for administrator-approved form templates.	As specified in [MS-IPFF] section 2.2.147.9, isMobileEnabled attribute
269	BrowserCompatibility	GenericXsf	Unbound Rich Text Box controls are not supported by InfoPath Forms Services.	As specified in [MS-IPFF] section 2.4.1.17.
270	BrowserCompatibility	GenericXsf	Unbound File Attachment controls are not supported by InfoPath Forms Services.	As specified in [MS-IPFF] section 2.4.1.11.
271	BrowserCompatibility	GenericXsf	Invalid or unsupported locale (LCID [<i>LCID</i>]) used in view [<i>view name</i>]. Try using different locale.	As specified in [MS-IPFF] section 2.4.1.4.
272	BrowserCompatibility	GenericXsf	The form cannot be converted because it was designed for a later version of InfoPath Forms Services.	As specified in [MS-IPFF] section 2.2.20, solutionFormatVersion attribute.
273	BrowserCompatibility	GenericXsl	The view contains nested formatting which is not supported on InfoPath Forms	The protocol server returns this error message when an XSL file is not supported because either the depth at which XML elements are nested in the file or

ID	Category	Feature	Message	Description
			Services. Examples of such formatting include heavily nested tables and heavily formatted text.	the complexity of processing the file exceeds what the server supports. The depth and complexity at which this message is reported is an implementation choice left to the protocol implementer.
274	BrowserCompatibility	GenericXPath	An XPath requires complicated processing which is not supported on InfoPath Forms Services.	The protocol server returns this error message when an XPath expression contains more steps and axis than the server supports. The complexity at which this message is reported is an implementation choice left to the protocol implementer

7.1.2 Message Elements of Type "Warning"

ID	Category	Feature	Message	Description
11	BrowserCompatibility	Controls	One or more buttons have the same ID property value as this one. The actions associated with the buttons might not execute correctly.	As specified in [MS-IPFF] section 2.4.2.10.
32	BrowserCompatibility	Controls	Unexpected token encountered. Expected an identifier for the class name.	The protocol server returns this message when the syntax of a CSS class name that follows a '.' flag is invalid.
33	BrowserCompatibility	Controls	Unexpected token encountered. Expected an identifier for an ID property value.	The protocol server returns this message when the syntax of a CSS identifier preceded by a '#' flag is invalid.
34	BrowserCompatibility	Controls	Unexpected token encountered. Expected an identifier for a pseudo style.	The protocol server returns this message when the syntax of a CSS pseudo-style preceded by a ':' flag is invalid.
35	BrowserCompatibility	Controls	Invalid or malicious CSS styles were found in view <i>[view name]</i> .	The protocol server returns this message when the syntax of a CSS associated with a form view contains invalid elements or XML constructs that could permit script injection.
36	BrowserCompatibility	Controls	Unexpected character encountered. It will be ignored	The protocol server returns this message server when parsing the CSS associated with a form view.
42	BrowserCompatibility	Controls	Potentially unsafe HTML was found in view '[view	The protocol server will return this message when a form

ID	Category	Feature	Message	Description
			<i>name</i> ']. It will be modified or removed when the form is shown to the user.	view contains XML determined to be a potential security issue during run time of the browser-enabled form template.
47	BrowserCompatibility	Controls	Unsupported html attribute [<i>attribute name</i>] encountered.	The protocol server returns this message when the form view contains an unsupported XML attribute to describe its layout.
48	BrowserCompatibility	Controls	Unsupported HTML attribute encountered: [<i>attribute name</i>].	The protocol server returns this message when the form view contains an unsupported XML attribute to describe its layout.
50	BrowserCompatibility	Controls	The following HTML tag is not supported: [<i>tag name</i>].	The protocol server returns this message when the form view contains an unsupported XML tag.
51	BrowserCompatibility	Controls	Unsupported value [<i>value</i>] for size attribute on the font element. Defaulting to medium font size (4).	The protocol server returns this message when the form view contains an unsupported value for the font size attribute.
52	BrowserCompatibility	Controls	The form has a control with border and margin that is too restrictive to properly show error visualization.	The protocol server returns this message when the form view contains a control that cannot properly display the error visualizations UI.
53	BrowserCompatibility	Controls	Unexpected type of list found. Supported values are: '1', 'a', 'A', 'i' and 'I'. Using the default list type.	The protocol server returns this message when it encounters invalid XML list or list item bullet styling, as specified in [HTML] section 10.2 and [CSS-LEVEL2] section 12.5.1.
54	BrowserCompatibility	Controls	The form sets the [<i>css-style-name</i>] style to a value (<i>[css-style-value]</i>) where the unit could not be safely approximated to pixels.	The protocol server returns this message when failing to convert the value of a CSS style value to an appropriate number of pixels.
56	BrowserCompatibility	Controls	Potentially unsafe HTML was found in view [<i>view name</i>]. It will be modified or removed when the form is shown to the user.	The protocol server will return this message when a form view contains CSS determined to be a potential security issue during run time of the browser-enabled form template.
78	BrowserCompatibility	Controls	A default height will be	The protocol server returns

ID	Category	Feature	Message	Description
			added to a multi-line Text Box control in view <i>[view name]</i> bound to a node <i>[control name]</i> .	this message when encountering a multiline text box control (see [MS-IPFF] section 2.4.2.11), with a missing height attribute.
84	BrowserCompatibility	Controls	The language pack corresponding to the form locale <i>[locale name]</i> that has not been installed on the server. Text generated by InfoPath® Forms Services, such as menus, messages, and dialog boxes, will use the locale and language settings of the site where the form is activated.	As specified in [MS-IPFF] section 2.2.147.9.
85	BrowserCompatibility	Controls	The language pack corresponding to the form locale <i>[LCID]</i> that has not been installed on the server. Text generated by InfoPath Forms Services, such as menus, messages, and dialog boxes, will use the locale and language settings of the site where the form is activated.	As specified in [MS-IPFF] section 2.2.147.9.
109	BrowserCompatibility	Controls	Printing headers and footers is not supported in server forms. The form printed from the browser will not have a header or footer.	As specified in [MS-IPFF] section 2.2.115.
110	BrowserCompatibility	Controls	Printing headers and footers is not supported	As specified in [MS-IPFF] section 2.2.115.
153	BrowserCompatibility	Controls	The useDataSet option is not supported.	As specified in [MS-IPFF] section 2.2.39.
187	BrowserCompatibility	Controls	Unsupported border styles were found associated with this Repeating Table. They have been converted to solid border. The conversion to solid border is for Internet Explorer® only.	The protocol server returns this error message when a Repeating Table Control uses the border-style attribute and is setting its value to a string different from "solid", "none", or "hidden".
199	BrowserCompatibility	Views	The view attribute xsf:toolbar is not supported.	As specified in [MS-IPFF] section 2.2.118.
222	BrowserCompatibility	GenericXsf	A toolbar button element	As specified in [MS-IPFF]

ID	Category	Feature	Message	Description
			in the form definition file (manifest.xsf) has the following error: <i>[XML of the xsf2:command element]</i>	section 2.2.147.12.
243	BrowserOptimization	Controls	By default this control will send data to the server whenever its value changes. Reason: <i>[details]</i> . This can lead to effects that can be evaluated only on the server. <i>[Optional description and count of any related message]</i> To override this behavior, modify Postback Settings on the Browser Forms tab of the Control Properties dialog box.	The protocol server returns this message when a leaf control has a browser-optimization issue. The detailed message can describe how many preceding related messages are also caused by this browser-optimization issue.
244	BrowserOptimization	Controls	By default this control will send data to the server on insert or delete. Reason: <i>[details]</i> . This can lead to effects that can be evaluated only on the server. <i>[Optional description and count of any related messages]</i> To override this behavior, modify Postback Settings on the Browser Forms tab of the Control Properties dialog box.	The protocol server returns this message when a form section has a browser-optimization issue. The detailed message can describe how many preceding related messages are also caused by this browser-optimization issue.
245	BrowserOptimization	Controls	By default this control will send data to the server on insert or delete. Reason: Other related controls require evaluation on the server. For more information, see the following 'primary cause' message in the Design Checker. To override this behavior, modify Postback Settings on the Browser Forms tab of the Control Properties dialog box.	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. The server sends this message when a form control has a browser-optimization issue that is caused by another control in the same form view. A related message will appear later in the message list that identifies the other control which causes the browser-optimization issue with this control.
257	BrowserOptimization	Controls	Other controls are bound to the same field or group (<i>[XPath expression]</i>). This can lead to effects that can be evaluated only on the server. <i>[Optional</i>	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. This message is returned when the

ID	Category	Feature	Message	Description
			<i>description and count of any related message]</i>	optimization issue is because multiple controls have the same binding.
258	BrowserOptimization	Controls	This section contains a control bound to field or group (<i>[XPath expression]</i>) to which other controls are also bound. This can lead to effects that can be evaluated only on the server. <i>[Optional description and count of any related messages]</i>	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. This message is returned when a form section has an optimization issue because the binding of a control is inside the form section.
259	BrowserOptimization	Controls	The section is bound to the root node (<i>[XPath expression]</i>) of the data source. This can lead to effects that can only be evaluated on the server. <i>[Optional description and count of any related message]</i>	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. The server returns this message when a form section is binding to the root element.
262	BrowserCompatibility	Controls	Conditionally formatting a File Attachment control is not supported in server forms	As specified in [MS-IPFF] section 2.4.1.11.
263	BrowserOptimization	Controls	The binding of this section (<i>[binding]</i>) can lead to effects that can be evaluated only on the server. Reason: <i>[postback reason]</i> .As a result, other section or table controls must also send data to the server on insert or delete. For more information, see the preceding <i>[count]</i> message or messages in the Design Checker.	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. This particular message is returned when a Section Control is determined to be a likely cause of the issue.
264	BrowserOptimization	Controls	By default this button will send data to the server whenever it is clicked. Reason: It is associated with a rule that uses the following expression which can be evaluated only on the server: <i>[postback reason]</i> .To override this behavior, modify Postback Settings on the Browser Forms table of the Button Properties dialog box.	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. This particular message is returned when a XPath expression used by a Button Control is a likely cause of the issue.

ID	Category	Feature	Message	Description
275	BrowserCompatibility	Controls	Merged print views are not supported in browser-enabled forms. Instead a read-only view of the current view will be displayed as the print view.	The protocol server returns this message when the form definition (.xsf) file contains an mergedPrintView element as specified in [MS-IPFF] section 2.2.147.21.
276	BrowserCompatibility	Controls	Digital signatures (2) are not supported by InfoPath Forms Services for list forms. All digital signatures must be removed before the form can be published.	The protocol server returns this message when the form definition (.xsf) file contains an signedDataBlock element as specified in [MS-IPFF] section 2.2.125.

7.1.3 Message Elements of Type "Information"

ID	Category	Feature	Message	Description
37	BrowserCompatibility	Controls	Unexpected character <i>[character]</i> encountered, expected <i>[character]</i> . The unexpected character will be ignored.	The protocol server returns this message when parsing the CSS associated with a form view. The parser expects a '!' character in a comment tag.
38	BrowserCompatibility	Controls	Unexpected character <i>[character]</i> encountered, expected <i>[character]</i> . The unexpected character will be ignored.	The protocol server returns this message when parsing the CSS associated with a form view. The parser expects the first '-' character in a comment tag.
39	BrowserCompatibility	Controls	Unexpected character <i>[character]</i> encountered, expected <i>[character]</i> . The unexpected character will be ignored.	The protocol server returns this message when parsing the CSS associated with a form view. The parser expects the second '-' character in a comment tag.
41	BrowserCompatibility	Controls	Unexpected character encountered. Expected the end of the stylesheet declaration.	The protocol server returns this message when parsing the CSS associated with a form view. The parser could not find the end of the style sheet declaration.
43	BrowserCompatibility	Controls	Unterminated string encountered. This string will be ignored.	The protocol server returns this message when parsing the CSS associated with a form view that contains a string where quotes or double quotes are not properly closed.
96	BrowserCompatibility	Controls	Error in <i>[manifest file name]</i> . Invalid or unsupported value	As specified in [MS-IPFF] section 2.2.100.

ID	Category	Feature	Message	Description
			[LCID value] for the lang property of the [xsl file name] file. Ignoring the language attribute for this view.	

7.2 Messages for InfoPath 2010 Forms

Messages described in this section are generated by Microsoft® SharePoint® Server 2010 when design checking a form template (.xsn) file as specified in [\[MS-IPFF2\]](#).

7.2.1 Message Elements of Type "Error"

ID	Category	Feature	Message	Description
1	BrowserCompatibility	Controls	The form template is not browser-compatible. It might be possible to correct the problem by opening the form template in Microsoft® Office InfoPath® 2007, and then republishing it.	The protocol server returns this error message when the form template file cannot be opened. This error will occur when the file is invalid or corrupt.
5	BrowserCompatibility	Controls	Error loading form code assembly: [name of the user code module].	As specified in [MS-IPFF2] section 2.2.2.2.43.
6	BrowserCompatibility	Controls	A rule references a button which is missing from the view.	The protocol server returns this error message when a rule is associated with a button control that is not part of the form view.
12	BrowserCompatibility	Controls	Button [button name] has a value of the xd:action attribute that is not supported in browser-enabled form templates.	As specified in [MS-IPFF2] section 2.4.2.1.
13	BrowserCompatibility	Controls	Expected anchor tag is missing.	As specified in [MS-IPFF2] section 2.4.1.12.
14	BrowserCompatibility	Controls	List Box control is not supported.	As specified in [MS-IPFF2] section 2.4.1.13.
16	BrowserCompatibility	Controls	Combo Box control is not supported.	As specified in [MS-IPFF2] section 2.4.1.21.
18	BrowserCompatibility	Controls	[Control Name] control is not supported.	As specified in [MS-IPFF2] section 2.2.1.2.89.
21	BrowserCompatibility	Controls	Picture control is not	As specified in [MS-IPFF2]

ID	Category	Feature	Message	Description
	lity		supported.	section 2.4.1.23.
22	BrowserCompatibilty	Controls	Ink Picture control is not supported.	As specified in [MS-IPFF2] section 2.4.1.22.
23	BrowserCompatibilty	Controls	Horizontal Region control is not supported.	As specified in [MS-IPFF2] section 2.4.1.22.
24	BrowserCompatibilty	Controls	Master/detail control is not supported.	As specified in [MS-IPFF2] section 2.4.2.
26	BrowserCompatibilty	Controls	Recursive control not supported.	As specified in [MS-IPFF2] section 2.4.1.23.
28	BrowserCompatibilty	Controls	Scrolling Region control is not supported.	As specified in [MS-IPFF2] section 2.4.1.22.
30	BrowserCompatibilty	Controls	Vertical text is not supported.	As specified in [MS-IPFF2] section 2.4.1.22.
31	BrowserCompatibilty	Controls	Controls that repeat horizontally are not supported.	As specified in [MS-IPFF2] section 2.4.1.22.
40	BrowserCompatibilty	Controls	Encountered an encoded identifier <i>[Identifier name]</i> . Encoded identifiers are not supported.	The protocol server returns this error message when parsing the cascading style sheet (CSS) associated with a form view and the CSS contains an encoded identifier.
43	BrowserCompatibilty	Controls	Unterminated string encountered. This string will be ignored.	The protocol server returns this message when parsing the CSS associated with a form view that contains a string where quotes or double quotes are not properly closed.
45	BrowserCompatibilty	Controls	The form locale <i>[locale identifier]</i> is not a valid language identifier. Assign a valid locale and try again.	As specified in [MS-IPFF2] section 2.2.2.2.2.
46	BrowserCompatibilty	Controls	An XSL file is missing from the form template.	As specified in [MS-IPFF2] section 2.2.1.2.110.
57	BrowserCompatibilty	Controls	Linked images are not supported by InfoPath® Forms Services. To fix this problem, make the image a part of the form template. The following image was	The protocol server returns this error message when a form view contains an img element with a href attribute with a value of an absolute URL.

ID	Category	Feature	Message	Description
			found in view <i>[view name]</i> : <i>[image name]</i> .	
58	BrowserCompatibility	Controls	Unexpected image tag encountered. It is missing a required href attribute.	The protocol server returns this error message when detecting an img element without a href attribute in a form view.
59	BrowserCompatibility	Controls	Only the .png, .gif, and .jpg graphics file formats are supported.	The protocol server returns this error message when the href attribute of an img element in a form view refers a file with an unsupported extension. The Microsoft® InfoPath® 2010 supports the file extensions.png, .gif, and .jpg.
60	BrowserCompatibility	Controls	The data source <i>[data source name]</i> referenced in the form template is not valid or cannot be found.	As specified in [MS-IPFF2] section 2.4.3.9.2.
61	BrowserCompatibility	Controls	Invalid format for location attribute <i>[attribute value]</i> .	As specified in [MS-IPFF2] section 2.2.1.2.42.
62	BrowserCompatibility	Controls	An eval expression used in the form has invalid arguments.	As specified in [MS-IPFF2] section 2.4.3.6.2.
63	BrowserCompatibility	Controls	The XPath <i>[xpath expression]</i> is invalid. <i>[error details]</i>	As specified in [XPATH] .
64	BrowserCompatibility	Controls	The following file is not a valid Xml Schema: <i>[name of the schema file]</i>	As specified in [XMLSCHEMA1] .
67	BrowserCompatibility	Controls	The following file is either missing or is not part of included in the form template: <i>[filename]</i> . To add a file to the form template in design mode, use the Resource Files dialog box on the Tools menu, and then add the file.	As specified in [MS-IPFF2] section 2.2.1.2.79.
68	BrowserCompatibility	Controls	Multiple xsl:template's found	The protocol server returns this error message when a form view

ID	Category	Feature	Message	Description
			with name corresponding to this xsl:call-template. Using the first xsl:template found.	file contains more than one XSL template element with the same value for the name attribute.
69	BrowserCompatibility	Controls	Did not find an xsl:template with the same name as this xsl:call-template. Ignoring the xsl:call-template.	As specified in [W3C-XSLT] .
71	BrowserCompatibility	Controls	Multiple xsl:templates found with mode corresponding to this xsl:apply-templates. Using the first xsl:template found.	The protocol server returns this error message when a form view file contains an apply-templates element that can find one or more xsl:template element to apply with the same mode attribute.
72	BrowserCompatibility	Controls	The xsl:stylesheet element had multiple xsl:template elements with no mode attribute. This is not supported. Using only the first of these xsl:templates.	As specified in [MS-IPFF2] section 2.4.1.2.
73	BrowserCompatibility	Controls	The mode of this xsl:apply-templates element does not match the mode attribute of any xsl:template element.	As specified in [MS-IPFF2] sections 2.4.1.15 and 2.4.1.18 .
74	BrowserCompatibility	Controls	Could not find an xsl:template without a mode inside xsl:stylesheet.	As specified in [MS-IPFF2] section 2.4.1.2.
75	BrowserCompatibility	Controls	The following construct is not supported: 'xsl:apply-imports'.	As specified in [MS-IPFF2] section 2.4.1.
76	BrowserCompatibility	Controls	The following construct is not supported: 'xsl:import'.	As specified in [MS-IPFF2] section 2.4.1.
77	BrowserCompatibility	Controls	The following construct is not supported: 'xsl:include'.	As specified in [MS-IPFF2] section 2.4.1.

ID	Category	Feature	Message	Description
79	BrowserCompatibility	Controls	An unexpected error has occurred while verifying the form template.	As specified in [MS-IPFF2] section 2.1.
87	BrowserCompatibility	Calculations	The following expression could not be parsed because of a syntax error or because it uses an undefined namespace prefix or unsupported function: <i>[XPath expression]</i> .	As specified in [MS-IPFF2] section 2.2.1.2.127, target and expression attributes.
88	BrowserCompatibility	ConditionalFormatting	Unsupported expression	As specified in [MS-IPFF2] section 2.4.1.1.
89	BrowserCompatibility	Controls	Unsupported expression	As specified in [MS-IPFF2] section 2.4.1.1.
90	BrowserCompatibility	DataAdapters	Unsupported expression	As specified in [MS-IPFF2] section 2.2.1.2.25.
91	BrowserCompatibility	DigitalSignatures	Unsupported expression	As specified in [MS-IPFF2] section 2.2.1.2.107, data and signatureLocation attributes.
92	BrowserCompatibility	GenericXsl	Unsupported expression	As specified in [MS-IPFF2] section 2.4.1.
93	BrowserCompatibility	Rules	Unsupported expression	As specified in [MS-IPFF2] section 2.2.1.2.114.
94	BrowserCompatibility	Controls	Unsupported expression	As specified in [MS-IPFF2] section 2.2.1.2.54, expression and expressionContext attributes.
95	BrowserCompatibility	Controls	Unsupported expression	As specified in [MS-IPFF2] sections 2.2.1.2.105 and 2.2.1.2.88 .
97	BrowserCompatibility	Controls	Browser-enabled form templates must have at least one browser-compatible view.	As specified in [MS-IPFF2] section 2.2.1.2.103.
98	BrowserCompatibility	Controls	<i>[Detailed error message the XML Processor]</i>	As specified in [MS-IPFF2] section 2.2.
99	BrowserCompatibility	Controls	Unexpected schema validation error: <i>[Schema Error]</i>	As specified in [MS-IPFF2] section 2.2.1.2.41.
10	BrowserCompatibility	Controls	<i>[Detailed error</i>	As specified in [MS-IPFF2]

ID	Category	Feature	Message	Description
0	lity		<i>message the XML Processor]</i>	section 2.2.
102	BrowserCompatibility	Controls	HWS data connections are not supported.	As specified in [MS-IPFF2] section 2.2.1.2.21.
103	BrowserCompatibility	Controls	HWS task pane is not supported.	As specified in [MS-IPFF2] section 2.2.1.2.68.
111	BrowserCompatibility	Controls	The form template is not browser-compatible, perhaps as a result of modifications made outside of InfoPath 2010. It might be possible to correct the problem by republishing the form template from within Office InfoPath 2007.	As specified in [MS-IPFF2] section 2.2.2.2.1, runtimeCompatibility attribute.
112	BrowserCompatibility	Controls	Character expected. Found an empty value.	As specified in [MS-IPFF2] section 2.4.1.
113	BrowserCompatibility	Controls	Character expected. Found a string value.	As specified in [MS-IPFF2] section 2.4.1.1.
114	BrowserCompatibility	Controls	Double value expected. <i>[value]</i> is not a valid double.	As specified in [MS-IPFF2] section 2.4.1.
115	BrowserCompatibility	Controls	Invalid Uri found as value of href attribute of xsl:import.	The protocol server returns this error message when parsing the XSL of a form view file. The href attribute of an xsl:import element uses an invalid URI value.
116	BrowserCompatibility	Controls	Invalid Uri found as value of href attribute of xsl:include.	The protocol server returns this error message when parsing the XSL of a form view file. The href attribute of an xsl:include element uses an invalid URI value.
117	BrowserCompatibility	Controls	Invalid value: <i>[value]</i> for grouping-size attribute of xsl:number. The value must be a positive integer value.	The protocol server returns this error message when parsing the XSL of a form view file. The grouping-size attribute xsl:number element has an invalid value.
118	BrowserCompatibility	Controls	Invalid value: <i>[value]</i> for letter-value attribute of	The protocol server returns this error message when parsing the XSL of a form view file. The

ID	Category	Feature	Message	Description
			xsl:number. Valid values are: 'alphabetic' and 'traditional'.	letter-value attribute of an xsl:number element has a value different from "alphabetic" or "traditional".
119	BrowserCompatibility	Controls	Invalid value: <i>[value]</i> for level attribute of xsl:number. Valid values are: 'single', 'multiple' and 'any'.	The protocol server returns this error message when parsing the XSL of a form view file. The level attribute of an xsl:number element has a value different from "single", "multiple" or "any".
120	BrowserCompatibility	Controls	Invalid value: <i>[value]</i> for case-order attribute of xsl:sort. Valid values are: 'upper-first' and 'lower-first'.	The protocol server returns this error message when parsing the XSL of a form view file. The case-order attribute of an xsl:sort element has a value different from "upper-first" or "lower-first".
121	BrowserCompatibility	Controls	Invalid value: <i>[value]</i> for order attribute of xsl:sort. Valid values are: 'ascending' and 'descending'.	The protocol server returns this error message when parsing the XSL of a form view file. The order attribute of an xsl:sort element has a value different from "ascending" or "descending".
122	BrowserCompatibility	Controls	xsl:import encountered after other xsl constructs.	The protocol server returns this error message when parsing the XSL of a form view file. The parser found an xsl:import element that is not at the top level of the XSL document.
123	BrowserCompatibility	Controls	Integer value expected. <i>[value]</i> is not a valid integer.	The protocol server returns this error message when parsing the XSL of a form view file. The parser found an invalid value where an integer value was expected.
124	BrowserCompatibility	Controls	The qualified name <i>[name]</i> is not valid.	As specified in [MS-IPFF2] section 2.4.1.
125	BrowserCompatibility	Controls	The qualified name <i>[original value]</i> is not a valid. Using qualified name <i>[replace value]</i> .	As specified in [MS-IPFF2] section 2.4.1.
126	BrowserCompatibility	Controls	Xsl for view <i>[view name]</i> is not a valid xml file. <i>[Detailed error message the XML Processor]</i>	As specified in [MS-IPFF2] section 2.4.
12	BrowserCompatibility	Controls	Multiple	As specified in [W3C-XSLT]

ID	Category	Feature	Message	Description
7	lity		xsl:otherwise elements in an xsl:when are not supported.	section 9.2.
128	BrowserCompatibility	Controls	xsl:template found with no match attribute and no name attribute. At least one must be specified.	As specified in [W3C-XSLT] section 5.3, 6
129	BrowserCompatibility	Controls	If an xsl:template element does not have a match attribute, it must not have a mode attribute.	As specified in [W3C-XSLT] section 5.7
130	BrowserCompatibility	Controls	xsl:otherwise element found without a corresponding xsl:when element.	As specified in [W3C-XSLT] section 9.2
131	BrowserCompatibility	Controls	xsl:param encountered after other xsl constructs.	As specified in [W3C-XSLT] section 15
132	BrowserCompatibility	Controls	The prefix <i>[namespace prefix]</i> is not declared in the current scope.	As specified in [W3C-XSLT] section 2.1
133	BrowserCompatibility	Controls	Unexpected attribute <i>{[namespace]}[attribute local-name]</i> encountered.	As specified in [W3C-XSLT] section 2.1
134	BrowserCompatibility	Controls	Unexpected element <i>{[namespace]}[element local-name]</i> encountered.	As specified in [W3C-XSLT] section 2.1
135	BrowserCompatibility	Controls	Unexpected node encountered.	As specified in [W3C-XSLT] section 2.1
136	BrowserCompatibility	Controls	Unexpected type of node <i>[node type]</i> encountered.	As specified in [W3C-XSLT] section 2.1
137	BrowserCompatibility	Controls	Unexpected child element for xsl:for-each. Only xsl:sort elements are allowed.	As specified in [W3C-XSLT] section 8
138	BrowserCompatibility	Controls	Unsupported value: <i>[value]</i> found for	As specified in [MS-IPFF2] section 2.4.1.2.

ID	Category	Feature	Message	Description
			version attribute of xsl:stylesheet.	
139	BrowserCompatibility	Controls	Unexpected text encountered.	As specified in [MS-IPFF2] section 2.4.1.23.
140	BrowserCompatibility	Controls	Unexpected value: <i>[value]</i> . Expected 'yes' or 'no'.	As specified in [W3C-XSLT] section B
141	BrowserCompatibility	Controls	Entity References are not supported.	As specified in [W3C-XSLT] section 2.4.
142	BrowserCompatibility	Controls	This element does not permit content if the select attribute is specified.	As specified in [W3C-XSLT] section 2.4.
143	BrowserCompatibility	Controls	Unexpected xsl:when element found after an xsl:otherwise.	As specified in [W3C-XSLT] section 9.2.
144	BrowserCompatibility	Controls	The form definition (.xsf) file has an invalid errorCondition -- errorMessage tag is missing.	As specified in [MS-IPFF2] section 2.2.1.2.44, errorMessage
145	BrowserCompatibility	Controls	The form definition (.xsf) file has an invalid errorCondition -- errorMessage tag is missing.	As specified in [MS-IPFF2] section 2.2.1.2.44, errorMessage element.
147	BrowserCompatibility	Controls	Duplicate data adapter name encountered.	As specified in [MS-IPFF2] section 2.2.1.2.17, name attribute.
148	BrowserCompatibility	Controls	The specified onAfterSubmit action is not supported.	As specified in [MS-IPFF2] section 2.2.1.2.53, onAfterSubmit attribute.
149	BrowserCompatibility	Controls	Invalid query or submit specification for data adapter.	As specified in [MS-IPFF2] section 2.2.1.2.18.
150	BrowserCompatibility	Controls	Unsupported Data Adapter.	As specified in [MS-IPFF2] section 2.2.1.2.40.
151	BrowserCompatibility	Controls	A partFragment tag is missing a match attribute.	As specified in [MS-IPFF] section 2.2.44, match attribute.
152	BrowserCompatibility	Controls	A partFragment tag is missing a replaceWith attribute.	As specified in [MS-IPFF2] section 2.2.1.2.23, replaceWith attribute

ID	Category	Feature	Message	Description
155	BrowserCompatibility	Controls	The operation failed. The data contained multiple DataSets.	The protocol server returns this message when the data source returns more than one dataset. This could happen when querying data from a [MS-SQL] database that supports multiple datasets.
158	BrowserCompatibility	Controls	One or more data connection library attributes is empty.	As specified in [MS-IPFF2] section 2.2.1.2.23, connectionLinkType attribute.
162	BrowserCompatibility	Controls	The following HTML tag is not supported: 'Text Box'.	As specified in [MS-IPFF2] section 2.4.2.11.
171	BrowserCompatibility	Controls	The XSL for the Date Picker is not in the expected form.	As specified in [MS-IPFF2] section 2.4.1.8.
172	BrowserCompatibility	Controls	Invalid or malicious HTML was found in view <i>[view name]</i> .	The server returns this message when the body element in a form view is not valid HTML or if an attribute on that element contains a quote (") character.
173	BrowserCompatibility	Controls	Invalid or malicious HTML was found in view <i>[view name]</i> .	The protocol server returns this error message when a form view contains invalid or unsafe HTML elements or attributes, and the only SourceLocation information the protocol server is returning is the source file in which the issue occurs. Determining what HTML to consider unsafe is implementation-specific.
174	BrowserCompatibility	Controls	Invalid or malicious HTML was found in view <i>[view Name]</i> .	The protocol server returns this error message when the server cannot determine whether or not a form view contains invalid or unsafe HTML elements or attributes. Determining what HTML to consider unsafe is implementation-specific.
175	BrowserCompatibility	Controls	Invalid or malicious HTML was found in view <i>[view Name]</i> .	The protocol server returns this error message when a form view contains invalid or unsafe HTML elements or attributes, the SourceLocation information the protocol server is returning contains both the source file and line number in which the issue occurs, and the MessageType value for the corresponding Message is "Error". Determining what HTML to consider unsafe is

ID	Category	Feature	Message	Description
				implementation-specific.
176	BrowserCompatibility	Control	Invalid or malicious HTML was found in view <i>[view Name]</i> .	The protocol server returns this error message when the server cannot determine whether or not a form view contains invalid or unsafe HTML elements or attributes. Determining what HTML to consider unsafe is implementation-specific.
177	BrowserCompatibility	TemplateXml	The form template cannot be browser-enabled because the template data is not valid according to its schema. <i>[Error details]</i>	As specified in [MS-IPFF2] section 2.7.
178	BrowserCompatibility	Controls	A required parameter is missing for the data adapter <i>[adapter name]</i> : <i>[XPath expression for the missing parameter]</i>	As specified in [MS-IPFF2] sections 2.2.1.2.20 and 2.2.1.2.22 .
179	BrowserCompatibility	Controls	A required parameter is missing for the data adapter <i>[adapter name]</i> : <i>[XPath expression for the missing parameter]</i>	As specified in [MS-IPFF2] sections 2.2.1.2.20 and 2.2.1.2.22 .
180	BrowserCompatibility	Controls	Relative links to Data Connection Libraries located on different SharePoint site collection are not supported.	As specified in [MS-IPFF2] section 2.2.2.2.23, siteCollection attribute.
183	BrowserCompatibility	Controls	Could not execute the following relative query <i>[data adapter name]</i> . Relative queries are not allowed for connections linked to the Data Connection Library.	As specified in [MS-IPFF2] section 2.2.2.2.26.
184	BrowserCompatibility	Controls	Cannot run the relative query <i>[query]</i> . Relative queries are not allowed for connections linked to the Data Connection	As specified in [MS-IPFF2] section 2.2.2.2.27.

ID	Category	Feature	Message	Description
			Library.	
185	BrowserCompatibility	Controls	relativeQuery/@replace cannot be a relative, local, or UNC path.	As specified in [MS-IPFF2] section 2.2.2.27.
186	BrowserCompatibility	Controls	Unsupported HTML constructs were found associated with this Repeating Table.	As specified in [MS-IPFF2] section 2.4.1.16.
191	BrowserCompatibility	Controls	A text box is bound to an inappropriate datatype.	As specified in [MS-IPFF2] section 2.4.1.20.
192	BrowserCompatibility	Controls	Rule was referenced but not defined: <i>[rulename]</i> .	As specified in [MS-IPFF2] section 2.2.1.2.113, ruleSet attribute.
193	BrowserCompatibility	Controls	There has been a critical error while processing the form.	The protocol server returns this error message protocol server when an unknown error was encountered and the protocol server does not have a default method for handling this unknown error.
196	BrowserCompatibility	DataAdapters	This database is not supported. The database must be a Microsoft® SQL Server®.	The protocol server returns this error message when there is a data adapter that attempts to connection to a database that is not supported by the protocol server .
197	BrowserCompatibility	BusinessLogic	Unsupported object model version.	As specified in [MS-IPFF2] section 2.2.2.2.43, version attribute.
200	BrowserCompatibility	GenericXsf	The definition of view <i>[view name]</i> does not exist.	As specified in [MS-IPFF2] sections 2.2.1.2.49 , 2.2.1.2.95 , and 2.2.1.2.110 , transform attribute.
203	BrowserCompatibility	GenericXPath	A call to GetDOM failed. The DataObject does not exist.	As specified in [MS-IPFF2] section 2.4.3.9.2.
204	BrowserCompatibility	ExpressionXPathAnalysis	Invalid expression encountered: <i>[XPath expression]</i> .	As specified in [XPath] .
205	BrowserCompatibility	ExpressionXPathAnalysis	Invalid binding XPath <i>[XPath expression]</i> .	As specified in [MS-IPFF2] section 2.4.1.1, productions LEAF_XPATH, GROUP_XPATH, RELATIVE_REPEATING_GROUP_XPATH and

ID	Category	Feature	Message	Description
				RELATIVE_LEAF_XPATH.
206	BrowserCompatibility	ExpressionXPathAnalysis	Invalid binding XPath <i>[XPath expression]</i> .	As specified in [MS-IPFF2] section 2.4.1.1, productions LEAF_XPATH, GROUP_XPATH, RELATIVE_REPEATING_GROUP_XPATH and RELATIVE_LEAF_XPATH.
207	BrowserCompatibility	ExpressionXPathAnalysis	Invalid binding XPath <i>[XPath expression]</i> .	As specified in [MS-IPFF2] section 2.4.1.1, productions LEAF_XPATH, GROUP_XPATH, RELATIVE_REPEATING_GROUP_XPATH and RELATIVE_LEAF_XPATH.
208	BrowserCompatibility	ExpressionXPathAnalysis	Invalid binding XPath <i>[XPath expression]</i> .	As specified in [MS-IPFF2] section 2.4.1.1, productions LEAF_XPATH, GROUP_XPATH, RELATIVE_REPEATING_GROUP_XPATH and RELATIVE_LEAF_XPATH.
209	BrowserCompatibility	ExpressionXPathAnalysis	Invalid binding XPath <i>[XPath expression]</i> .	As specified in [MS-IPFF2] section 2.4.1.1, production LEAF_XPATH.
210	BrowserCompatibility	GenericXsf	The form template defines multiple schemas for the same namespace: <i>[namespace]</i>	As specified in [MS-IPFF2] section 2.2.1.2.42, location attribute.
211	BrowserCompatibility	GenericXsf	Form template is not valid. The following schema file: <i>[filename]</i> was not found in the Form Template.	As specified in [MS-IPFF2] section 2.2.1.2.79.
212	BrowserCompatibility	GenericXsf	The following DataObject either cannot be created or cannot be initialized: <i>[data objectName]</i> . The data adapter cannot be initialized. The form contains XML that cannot be parsed: <i>[Detailed error message the XML Processor]</i> .	As specified in [MS-IPFF2] section 2.2.1.2.17, schema attribute.
213	BrowserCompatibility	GenericXsf	The specified XML template file (xsf:initialXmlDocument element) is not a valid XML document.	As specified in [MS-IPFF2] section 2.2.1.2.75.

ID	Category	Feature	Message	Description
			[Detailed error message the XML Processor].	
214	BrowserCompatibility	GenericXsf	The specified XML template file (xsf:initialXmlDocument element) does not contain required processing instructions.	As specified in [MS-IPFF2] section 2.2.1.2.75.
215	BrowserCompatibility	GenericXsf	The specified XML template file (xsf:initialXmlDocument element) is not a valid XML document. [Detailed error message the XML Processor]	As specified in [MS-IPFF2] section 2.2.1.2.75.
216	BrowserCompatibility	GenericXsf	The XML template file (specified in the xsf:initialXmlDocument element of the form template definition file) has a processing instruction with a form template version that does not match the version of the form template.	As specified in [MS-IPFF2] section 2.2.1.2.1, solutionVersion attribute and [MS-IPFF2] section 2.7.
217	BrowserCompatibility	GenericXsf	The specified XML template file (xsf:initialXmlDocument element) contains a urn reference that does not match the solutions name.	As specified in [MS-IPFF2] section 2.2.1.2.1, name attribute and [MS-IPFF2] section 2.7.
218	BrowserCompatibility	GenericXsf	No XML template file is present in the form template.	As specified in [MS-IPFF2] section 2.2.1.2.75, href attribute.
219	BrowserCompatibility	GenericXsf	The following XML template file is missing or is not part of the form template: [file name]	As specified in [MS-IPFF] section 2.2.94 or in [MS-IPFF2] section 2.2.1.2.75, href attribute.
220	BrowserCompatibility	GenericXsf	Duplicate property name '[property name]' in file '[file name]'.	As specified in [MS-IPFF] section 2.2.100, name attribute.
22	BrowserCompatibility	GenericXsf	A button element in	As specified in [MS-IPFF2]

ID	Category	Feature	Message	Description
1	lity		the form definition file (manifest.xsf) has invalid attributes.	section 2.2.1.2.87 and 2.2.1.2.91 .
223	BrowserCompatibility	DigitalSignatures	An error occurred when initializing a set of signable data. A name has not been specified for the set of data.	As specified in n [MS-IPFF2] section 2.2.1.2.107, name attribute.
234	BrowserCompatibility	DigitalSignatures	No group was found at the location specified for storing signatures for the set of signable data: <i>[XPath expression]</i> . Use the Digital Signatures (2) category of the Form Options dialog box to edit the expression specifying the storage location for signatures.	As specified in [MS-IPFF2] section 2.2.1.2.107, signatureLocation attribute.
235	BrowserCompatibility	DigitalSignatures	An error occurred when initializing the set of signable data: <i>[XPath expression]</i> . No group was found at the specified location.	As specified in [MS-IPFF2] section 2.2.1.2.107, mode attribute.
236	BrowserCompatibility	DigitalSignatures	An error occurred when initializing the set of signable data: " <i>[XPath expression]</i> ". No group was found at the specified location.	As specified in [MS-IPFF2] section 2.2.1.2.107, mode attribute.
237	BrowserCompatibility	DigitalSignatures	No fields or groups were found corresponding to the set of signable data: <i>[XPath expression]</i> . Use the Digital Signatures category of the Form Options dialog box to edit the expression specifying the data to be signed.	As specified in [MS-IPFF2] section 2.2.1.2.107, data attribute.
238	BrowserCompatibility	GenericXsl	No group was found at the location specified for storing	As specified in [MS-IPFF2] section 2.2.1.2.107, signatureLocation attribute.

ID	Category	Feature	Message	Description
			signatures for the set of signable data: "[signed data block name]". Use the Digital Signatures category of the Form Options dialog box to edit the expression specifying the storage location for signatures.	
239	BrowserCompatibility	GenericXsf	The following expression could not be parsed because of a syntax error or because it uses an undefined namespace prefix or unsupported function: <i>[XPath expression]</i> .	As specified in [MS-IPFF2] section 2.4.1.
240	BrowserCompatibility	Controls	Numbered List control is not supported	As specified in [MS-IPFF2] section 2.4.1.21, xctName attribute.
241	BrowserCompatibility	Controls	Bulleted List control is not supported	As specified in [MS-IPFF2] section 2.4.1.21, xctName attribute.
242	BrowserCompatibility	Controls	Plain List control is not supported	As specified in [MS-IPFF2] section 2.4.1.21, xctName attribute.
247	BrowserCompatibility	Controls	This control formats the data to show both date and time. Date and time cannot be displayed together in the same control in server forms. Use two controls to display date and time separately and bind them to the same field.	As specified in [MS-IPFF2] section 2.4.2.11.
248	BrowserCompatibility	Controls	This control formats the data using an invalid or unsupported time format <i>[user format string]</i> . Pick a different format.	As specified in [MS-IPFF2] section 2.4.2.11.
249	BrowserCompatibility	Controls	This control formats the data using an	As specified in [MS-IPFF2] section 2.4.2.11.

ID	Category	Feature	Message	Description
			invalid or unsupported date format [<i>user format string</i>]. Pick a different format.	
250	BrowserCompatibility	GenericXsf	The restricted trust level is not supported.	As specified in [MS-IPFF2] section 2.2.1.2.1, trustLevel attribute.
252	BrowserCompatibility	GenericXsf	This form template has not been correctly published to be browser-enabled. Open the form template in InfoPath 2010 Design mode, and click Publish Form Template in the Design Tasks task pane. Follow the steps in the Publishing Wizard to republish the form template, and then try again.	The protocol server returns this message when it encounters errors while trying to browser-enable a form template.
256	BrowserCompatibility	Controls	Selected rich text formatting options are not supported Selected rich text formatting options are not supported.	As specified in [MS-IPFF2] section 2.2.2.2.36.
260	BrowserCompatibility	Controls	The following expression could not be parsed because of a syntax error or because it uses an undefined namespace prefix or unsupported function: [<i>XPath expression</i>].	As specified in [MS-IPFF2] section 2.4.1.
261	BrowserCompatibility	Controls	Specifying a restricted set of allowable file types for a File Attachment control is not supported in server forms.	As specified in [MS-IPFF2] section 2.2.1.2.89, allowedFileTypes attribute.
262	BrowserCompatibility	Controls	Conditionally formatting a File Attachment control is not supported in server forms	As specified in [MS-IPFF2] section 2.4.1.11.

ID	Category	Feature	Message	Description
267	BrowserCompatibility	GenericXsf	The form template is not browser-compatible, perhaps as a result of modifications made outside of InfoPath 2010. It might be possible to correct the problem by republishing the form template from within Office InfoPath 2007.	The protocol server returns this error message when the form template is not a browser-compatible form template and no more specific error message is applicable.
268	BrowserCompatibility	GenericXsf	Forms enabled for use on a mobile device are supported only for administrator-approved form templates.	As specified in [MS-IPFF2] section 2.2.2.2.2, isMobileEnabled attribute
269	BrowserCompatibility	GenericXsf	Unbound Rich Text Box controls are not supported by InfoPath Forms Services.	As specified in [MS-IPFF2] section 2.4.1.17.
270	BrowserCompatibility	GenericXsf	Unbound File Attachment controls are not supported by InfoPath Forms Services.	As specified in [MS-IPFF2] section 2.4.1.11.
271	BrowserCompatibility	GenericXsf	Invalid or unsupported locale (LCID [<i>LCID</i>]) used in view [<i>view name</i>]. Try using different locale.	As specified in [MS-IPFF2] section 2.4.1.4.
272	BrowserCompatibility	GenericXsf	The form cannot be converted because it was designed for a later version of InfoPath Forms Services.	As specified in [MS-IPFF2] section 2.2.1.2.1, solutionFormatVersion attribute.
273	BrowserCompatibility	GenericXsl	The view contains nested formatting which is not supported on InfoPath Forms Services. Examples of such formatting include heavily nested tables and heavily formatted text.	The protocol server returns this error message when an XSL file is not supported because either the depth at which XML elements are nested in the file or the complexity of processing the file exceeds what the server supports. The depth and complexity at which this message is reported is an implementation choice left to the protocol implementer.

ID	Category	Feature	Message	Description
274	BrowserCompatibility	GenericXPath	An XPath requires complicated processing which is not supported on InfoPath Forms Services.	The protocol server returns this error message when an XPath expression contains more steps and axis than the server supports. The complexity at which this message is reported is an implementation choice left to the protocol implementer
278	BrowserCompatibility	Controls	Invalid location for the xsn being published.	The protocol server returns this message when the form definition (.xsn) file is being published to a location that does not support form publishing.
279	BrowserCompatibility	Controls	Repeating Choice Group control is not supported	The protocol server returns this message when parsing the XSL associated with a form view that contains a Repeating Choice Group Control as specified in [MS-IPFF2] section 2.3.2.1.
280	BrowserCompatibility	Controls	Repeating Choice Section control is not supported	The protocol server returns this message when parsing the XSL associated with a form view that contains a Repeating Choice Section control as specified in [MS-IPFF2] section 2.3.2.1.
281	BrowserCompatibility	Controls	Choice Section controls must be inside a Choice Group control	The protocol server returns this message when the view definition (.xsl) file contains a Choice section as specified in [MS-IPFF2] section 2.3.2.1 that is not defined in a Choice Group control as specified in [MS-IPFF2] section 2.3.2.1.
287	BrowserCompatibility	BusinessLogic	The filter expression is not supported by InfoPath 2010	The protocol server returns this message when the view definition (.xsl) file contains a predicate expression as specified in [MS-IPFF2] section 2.4.1 that cannot be evaluated on the protocol server.
288	BrowserCompatibility	BusinessLogic	The filter expression is on external data and could not be validated at deployment time as a supported expression by InfoPath 2010	The protocol server returns this message when the view definition (.xsl) file contains a predicate expression as specified in [MS-IPFF2] section 2.4.1 on an external data source as specified in [MS-IPFF2] .
289	BrowserCompatibility	BusinessLogic	Unsupported binding expression used.	The protocol sever returns this message when a control as specified in [MS-IPFF2] section 2.4 contains a binding

ID	Category	Feature	Message	Description
				expression that is not supported.
290	BrowserCompatibility	BusinessLogic	Unsupported HTML constructs were found associated with this control.	The protocol server returns this message when the view definition (.xsl) file as specified in [MS-IPFF2] contains HTML that does not match required HTML as specified in [MS-IPFF2].
291	BrowserCompatibility	Controls	An invalid XSL was encountered.	The protocol server returns this message when no specific other specific error is available and parsing the XSL associated with a form definition (.xsn) file contains invalid XSL as specified in [MS-IPFF2] section 1.3.4
295	BrowserCompatibility	DataAdapters	Submitting through e-mail is not supported in sandboxed solutions. To publish this form to SharePoint, use the administrator-approved form template option in the Publishing Wizard or remove the e-mail connection [<i>E-mail adapter name</i>].	The protocol server returns this message when the form template is activated as a sandboxed solution and contains an e-mail adapter as specified in [MS-IPFF2] section 2.2.1.2.32.
296	BrowserCompatibility	GenericXsf	Web Part connection input parameters are not supported in sandboxed solutions. To publish this form to SharePoint, change the parameter to an output parameter or use the administrator-approved form template option in the Publishing Wizard.	The protocol server returns this message when a Web Part connection has input parameters and user code is present in the form template.
297	BrowserCompatibility	Controls	The External Item Picker is not supported in user code solutions.	The protocol server returns this message when the form template is activated as a sandboxed solution and contains an Entity Picker control as specified in [MS-IPFF2] section 2.3.2.4.
299	BrowserCompatibility	Controls	The user solution with code could not be activated.	The protocol server returns this message when a form template which has a managedCode

ID	Category	Feature	Message	Description
				element as specified in [MS-IPFF2] section 2.2.2.2.50 cannot be activated as a sandboxed solution.
300	BrowserCompatibility	Controls	An unexpected error has occurred while activating the user solution with code.	The protocol server returns this message when an unknown error condition occurs during the process of activated a form template as a sandboxed solution.
305	BrowserCompatibility	GenericXsf	Unbound control not supported.	The protocol server returns this message when the view definition (.xsl) file as specified in [MS-IPFF2] contains an unsupported unbound control.

7.2.2 Message Elements of Type "Warning"

ID	Category	Feature	Message	Description
11	BrowserCompatibility	Controls	One or more buttons have the same ID property as this one. The actions associated with the buttons might not execute correctly.	As specified in [MS-IPFF2] section 2.4.2.10.
32	BrowserCompatibility	Controls	Unexpected token encountered. Expected an identifier for the class name.	The protocol server returns this message when the syntax of a CSS class name that follows a '.' flag is invalid.
33	BrowserCompatibility	Controls	Unexpected token encountered. Expected an identifier for an ID property value.	The protocol server returns this message when the syntax of a CSS identifier preceded by a '#' flag is invalid.
34	BrowserCompatibility	Controls	Unexpected token encountered. Expected an identifier for a pseudo style.	The protocol server returns this message when the syntax of a CSS pseudo-style preceded by a ':' flag is invalid.
35	BrowserCompatibility	Controls	Invalid or malicious CSS styles were found in view <i>[view name]</i> .	The protocol server returns this message when the syntax of a CSS associated with a form view contains invalid elements or XML constructs that could permit script injection.
36	BrowserCompatibility	Controls	Unexpected character encountered. It will be ignored	The protocol server returns this message server when parsing the CSS associated with a form view.
42	BrowserCompatibility	Controls	Potentially unsafe HTML	The protocol server will return

ID	Category	Feature	Message	Description
			was found in view '[<i>view name</i>]'. It will be modified or removed when the form is shown to the user.	this message when a form view contains XML determined to be a potential security issue during run time of the browser-enabled form template.
47	BrowserCompatibility	Controls	Unsupported html attribute [<i>attribute name</i>] encountered.	The protocol server returns this message when the form view contains an unsupported XML attribute to describe its layout.
48	BrowserCompatibility	Controls	Unsupported HTML attribute encountered: [<i>attribute name</i>].	The protocol server returns this message when the form view contains an unsupported XML attribute to describe its layout.
50	BrowserCompatibility	Controls	The following HTML tag is not supported: [<i>tag name</i>].	The protocol server returns this message when the form view contains an unsupported XML tag.
51	BrowserCompatibility	Controls	Unsupported value [<i>value</i>] for size attribute on the font element. Defaulting to medium font size (4).	The protocol server returns this message when the form view contains an unsupported value for the font size attribute.
52	BrowserCompatibility	Controls	The form has a control with border and margin that is too restrictive to properly show error visualization.	The protocol server returns this message when the form view contains a control that cannot properly display the error visualizations UI.
53	BrowserCompatibility	Controls	Unexpected type of list found. Supported values are: '1', 'a', 'A', 'i' and 'I'. Using the default list type.	The protocol server returns this message when it encounters invalid XML list or list item bullet styling, as specified in [HTML] section 10.2 and [CSS-LEVEL2] section 12.5.1.
54	BrowserCompatibility	Controls	The form sets the [<i>css-style-name</i>] style to a value (<i>[css-style-value]</i>) where the unit could not be safely approximated to pixels.	The protocol server returns this message when failing to convert the value of a CSS style value to an appropriate number of pixels.
56	BrowserCompatibility	Controls	Potentially unsafe HTML was found in view [<i>view name</i>]. It will be modified or removed when the form is shown to the user.	The protocol server will return this message when a form view contains CSS determined to be a potential security issue during run time of the browser-enabled form template.

ID	Category	Feature	Message	Description
78	BrowserCompatibility	Controls	A default height will be added to a multi-line Text Box control in view <i>[view name]</i> bound to a node <i>[control name]</i> .	The protocol server returns this message when encountering a multiline text box control (see [MS-IPFF2] section 2.4.2.11), with a missing height attribute.
84	BrowserCompatibility	Controls	The language pack corresponding to the form locale <i>[locale name]</i> that has not been installed on the server. Text generated by InfoPath® Forms Services, such as menus, messages, and dialog boxes, will use the locale and language settings of the site where the form is activated.	As specified in [MS-IPFF2] section 2.2.2.2.2.
85	BrowserCompatibility	Controls	The language pack corresponding to the form locale <i>[LCID]</i> that has not been installed on the server. Text generated by InfoPath Forms Services, such as menus, messages, and dialog boxes, will use the locale and language settings of the site where the form is activated.	As specified in [MS-IPFF2] section 2.2.2.2.2.
109	BrowserCompatibility	Controls	Printing headers and footers is not supported in server forms. The form printed from the browser will not have a header or footer.	As specified in [MS-IPFF2] section 2.2.1.2.96.
110	BrowserCompatibility	Controls	Printing headers and footers is not supported	As specified in [MS-IPFF2] section 2.2.1.2.96.
153	BrowserCompatibility	Controls	The useDataSet option is not supported.	As specified in [MS-IPFF2] section 2.2.1.2.20.
187	BrowserCompatibility	Controls	Unsupported border styles were found associated with this Repeating Table. They have been converted to solid border. The conversion to solid border is for Internet Explorer® only.	The protocol server returns this error message when a Repeating Table Control uses the border-style attribute and is setting its value to a string different from "solid", "none", or "hidden".
199	BrowserCompatibility	Views	The view attribute <code>xf:toolbar</code> is not supported.	As specified in [MS-IPFF2] section 2.2.1.2.99.

ID	Category	Feature	Message	Description
222	BrowserCompatibility	GenericXsf	A toolbar button element in the form definition file (manifest.xsf) has the following error: <i>[XML of the xsf2:command element]</i>	As specified in [MS-IPFF2] section 2.2.2.2.5.
243	BrowserOptimization	Controls	By default this control will send data to the server whenever its value changes. Reason: <i>[details]</i> . This can lead to effects that can be evaluated only on the server. <i>[Optional description and count of any related message]</i> To override this behavior, modify Postback Settings on the Browser Forms tab of the Control Properties dialog box.	The protocol server returns this message when a leaf control has a browser-optimization issue. The detailed message can describe how many preceding related messages are also caused by this browser-optimization issue.
244	BrowserOptimization	Controls	By default this control will send data to the server on insert or delete. Reason: <i>[details]</i> . This can lead to effects that can be evaluated only on the server. <i>[Optional description and count of any related messages]</i> To override this behavior, modify Postback Settings on the Browser Forms tab of the Control Properties dialog box.	The protocol server returns this message when a form section has a browser-optimization issue. The detailed message can describe how many preceding related messages are also caused by this browser-optimization issue.
245	BrowserOptimization	Controls	By default this control will send data to the server on insert or delete. Reason: Other related controls require evaluation on the server. For more information, see the following 'primary cause' message in the Design Checker. To override this behavior, modify Postback Settings on the Browser Forms tab of the Control Properties dialog box.	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. The server sends this message when a form control has a browser-optimization issue that is caused by another control in the same form view. A related message will appear later in the message list that identifies the other control which causes the browser-optimization issue with this control.
257	BrowserOptimization	Controls	Other controls are bound to the same field or group (<i>[XPath expression]</i>). This can lead to effects that can be evaluated only on	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. This message is

ID	Category	Feature	Message	Description
			the server. <i>[Optional description and count of any related message]</i>	returned when the optimization issue is because multiple controls have the same binding.
258	BrowserOptimization	Controls	This section contains a control bound to field or group (<i>[XPath expression]</i>) to which other controls are also bound. This can lead to effects that can be evaluated only on the server. <i>[Optional description and count of any related messages]</i>	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. This message is returned when a form section has an optimization issue because the binding of a control is inside the form section.
259	BrowserOptimization	Controls	The section is bound to the root node (<i>[XPath expression]</i>) of the data source. This can lead to effects that can only be evaluated on the server. <i>[Optional description and count of any related message]</i>	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. The server returns this message when a form section is binding to the root element.
262	BrowserCompatibility	Controls	Conditionally formatting a File Attachment control is not supported in server forms	As specified in [MS-IPFF2] section 2.4.1.11.
263	BrowserOptimization	Controls	The binding of this section (<i>[binding]</i>) can lead to effects that can be evaluated only on the server. Reason: <i>[postback reason]</i> .As a result, other section or table controls must also send data to the server on insert or delete. For more information, see the preceding <i>[count]</i> message or messages in the Design Checker.	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. This particular message is returned when a Section Control is determined to be a likely cause of the issue.
264	BrowserOptimization	Controls	By default this button will send data to the server whenever it is clicked. Reason: It is associated with a rule that uses the following expression which can be evaluated only on the server: <i>[postback reason]</i> .To override this behavior, modify Postback Settings on the Browser Forms table of the Button	The protocol server returns this message when it detects this form template has a potential browser-optimization issue. This particular message is returned when a XPath expression used by a Button Control is a likely cause of the issue.

ID	Category	Feature	Message	Description
			Properties dialog box.	
275	BrowserCompatibility	Controls	Merged print views are not supported in browser-enabled forms. Instead a read-only view of the current view will be displayed as the print view.	The protocol server returns this message when the form definition (.xsf) file contains an mergedPrintView element as specified in [MS-IPFF2] section 2.2.1.2.14
276	BrowserCompatibility	Controls	Digital signatures (2) are not supported by InfoPath Forms Services for list forms. All digital signatures must be removed before the form can be published.	The protocol server returns this message when the form definition (.xsf) file contains an signedDataBlock element as specified in [MS-IPFF2] section 2.2.1.2.106.
285	BrowserCompatibility	Controls	The sign signature line rule action is not supported by ip14 forms. The rule will be ignored by Microsoft® InfoPath® 2010 forms.	The protocol server returns this message when the form definition (.xsf) file as specified in [MS-IPFF2] contains a SignatureLine element as specified in [MS-IPFF2] section 2.2.3.2.12.
286	BrowserCompatibility	Controls	The code in this form uses methods or properties that require assembly <i>[Assembly Name]</i>	The protocol server returns this message when the form definition (.xsf) file as specified in [MS-IPFF2] contains a reference to an assembly as specified by the following element managedCode as specified in [MS-IPFF2] section 2.2.2.2.43 that cannot be found when the form is being published.
298	BrowserCompatibility	Controls	Never postback setting is not supported for the Button control in InfoPath 2010. In the Browser forms tab of the Button control properties dialog box, change the postback setting to the recommended option.	The protocol server returns this message when the form definition (.xsf) file contains a button control, as specified in [MS-IPFF2] section 2.4.1.5, that has the postback setting, as specified in [MS-IPFF2] section 2.4.2.29 set to "Never".
303	BrowserCompatibility	Controls	Unsupported text indentation formatting (first line or hanging) was found in view <i>[View Name]</i> . Use clear formatting to remove first line or hanging text indentation.	The protocol server returns this message when the view definition (.xsl) file, as specified in [MS-IPFF2] , contains a text indentation style markup.
304	BrowserCompatibility	Controls	Code events on Person/Group Picker	The protocol server returns this message when the view

ID	Category	Feature	Message	Description
			controls are not supported in sandboxed solutions. To enable code events on Person/Group Picker controls use the administrator-approved form template option in the Publishing Wizard.	definition (.xsl) file, as specified in [MS-IPFF2], contains a Contact Selector as specified in [MS-IPFF2] section 2.3.1.3 that has user code.

7.2.3 Message Elements of Type "Information"

ID	Category	Feature	Message	Description
37	BrowserCompatibility	Controls	Unexpected character <i>[character]</i> encountered, expected <i>[character]</i> . The unexpected character will be ignored.	The protocol server returns this message when parsing the CSS associated with a form view. The parser expects a '!' character in a comment tag.
38	BrowserCompatibility	Controls	Unexpected character <i>[character]</i> encountered, expected <i>[character]</i> . The unexpected character will be ignored.	The protocol server returns this message when parsing the CSS associated with a form view. The parser expects the first '-' character in a comment tag.
39	BrowserCompatibility	Controls	Unexpected character <i>[character]</i> encountered, expected <i>[character]</i> . The unexpected character will be ignored.	The protocol server returns this message when parsing the CSS associated with a form view. The parser expects the second '-' character in a comment tag.
41	BrowserCompatibility	Controls	Unexpected character encountered. Expected the end of the stylesheet declaration.	The protocol server returns this message when parsing the CSS associated with a form view. The parser could not find the end of the style sheet declaration.
96	BrowserCompatibility	Controls	Error in <i>[manifest file name]</i> . Invalid or unsupported value <i>[LCID value]</i> for the lang property of the <i>[xsl file name]</i> file. Ignoring the language attribute for this view.	As specified in [MS-IPFF2] section 2.2.1.2.81.
292	BrowserCompatibility	Controls	The custom control has duplicate attributes or elements. Please check the XSL to ensure there are no duplicate values.	The protocol server returns this message when the view definition (.xsl) file, as specified in [MS-IPFF2] has duplicate attributes for one of the following controls: Entity Picker as specified in [MS-IPFF2] section 2.3.2.4 or Contact Selector as specified

ID	Category	Feature	Message	Description
				in [MS-IPFF2] section 2.3.1.3.
302	BrowserCompatibility	Controls	Submitting to a hosting environment through code is not supported in sandboxed solutions. If your form submits to a hosting environment through code, use the administrator-approved form template option in the Publishing Wizard.	The protocol server returns this message when a solution can submit data to a hosted environment and has user code present.

8 Appendix C: 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 Forms Server 2007
- Microsoft® Office InfoPath® 2007
- Microsoft® InfoPath® 2010
- Microsoft® Office SharePoint® Server 2007
- Microsoft® SharePoint® 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.

[<1> Section 2.2.4.2:](#) Office InfoPath 2007 and InfoPath 2010 ignore the suggestions contained in the **CategoryType** complex type.

[<2> Section 2.2.4.5:](#) InfoPath Forms Services and SharePoint Server 2010 return strings in the protocol server's designated default language.

[<3> Section 2.2.4.5:](#) See section [7](#)

[<4> Section 2.2.5.2:](#) The Office InfoPath 2007 and InfoPath 2010 do not use the value specified for the **Feature** element.

[<5> Section 3.1.4:](#) InfoPath Forms Services supports form template (.xsn) files conformant to [\[MS-IPFF\]](#). SharePoint Server 2010 supports form template (.xsn) files conformant to either [\[MS-IPFF\]](#) or [\[MS-IPFF2\]](#).

[<6> Section 3.1.4:](#) Section 3.1.4: InfoPath Forms Services and SharePoint Server 2010 return SOAP faults in response to any unsupported WSDL operation requests.

[<7> Section 3.1.4.1.2.2:](#) Office InfoPath 2007 and InfoPath 2010 use zero messages as a sign of successfully browser-enabling the form template (.xsn) file.

[<8> Section 3.1.4.2.2.1:](#) InfoPath Forms Services returns a SOAP exception.

[<9> Section 3.1.4.2.2.1:](#) Office InfoPath 2007 uses the value "O12 Designer" and InfoPath 2010 uses the value "O14 Designer". InfoPath Forms Services and SharePoint Server 2010 ignore this element.

[<10> Section 3.1.4.2.2.2:](#) The InfoPath Forms Services and SharePoint Server 2010 returns strings in the protocol server's designated default language.

[<11> Section 3.1.4.5.2.1:](#) The InfoPath Forms Services and SharePoint Server 2010 returns a SOAP exception.

<12> [Section 3.1.4.5.2.1](#): InfoPath 2010 uses the value "O14 Designer". SharePoint Server 2010 ignores this element.

<13> [Section 3.1.4.5.2.2](#): Office InfoPath 2007 and InfoPath 2010 use zero messages as a sign of successfully browser-enabling the form template (.xsn) file.

<14> [Section 3.1.4.5.2.2](#): Office InfoPath 2007 and InfoPath 2010 typically call the **DesignCheckFormTemplate** operation after a response from **BrowserEnableUserFormTemplate** with at least one **Message** element of **MessageType** "Error".

<15> [Section 3.1.4.6.2.2](#): SharePoint Server 2010 returns a SOAP exception and does not continue with additional field operations requested.

<16> [Section 3.1.4.6.2.2](#): Office InfoPath 2007 and InfoPath 2010 use zero messages as a sign of successfully browser-enabling the form template (.xsn) file.

<17> [Section 3.1.4.6.2.2](#): Office InfoPath 2007 and InfoPath 2010 typically call the **DesignCheckFormTemplate** operation after a response from **BrowserEnableUserFormTemplate** with at least one **Message** element of **MessageType** "Error".

9 Change Tracking

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

10 Index

A

Abstract data model
 [server](#) 21
[Applicability](#) 10
[Attribute groups](#) 20
[Attributes](#) 20

B

[BrowserEnableUserFormTemplate example](#) 41

C

[Capability negotiation](#) 10
[Categories complex type](#) 12
[Category simple type](#) 19
[CategoryType complex type](#) 12
[Change tracking](#) 111
Client
 [overview](#) 21
[Complex types](#) 11
 [Categories](#) 12
 [CategoryType](#) 12
 [ComplexType Messages](#) 13
 [DesignCheckerInformation](#) 13
 [Message](#) 13
 [SourceLocation](#) 16
[ComplexType Messages complex type](#) 13

D

Data model - abstract
 [server](#) 21
[DesignCheckerInformation complex type](#) 13
[DesignCheckFormTemplate operation example](#) 39

E

Events
 [local - server](#) 38
 [timer - server](#) 38
Examples
 [BrowserEnableUserFormTemplate operation](#) 41
 [DesignCheckFormTemplate operation](#) 39
 [GetListFormLocation operation](#) 44
 [GetUserCodeDeploymentDependencies operation](#)
 46
 [SetFormsForListItem operation](#) 42
 [SetSchemaChangesForList operation](#) 45

F

[Features simple type](#) 18
[Fields - vendor-extensible](#) 10
[Full WSDL](#) 49

G

[GetListFormLocation operation example](#) 44
[GetUserCodeDeploymentDependencies operation example](#) 46
[Glossary](#) 6
[Groups](#) 20

I

[Implementer - security considerations](#) 48
[Index of security parameters](#) 48
[Informative references](#) 8
Initialization
 [server](#) 21
[Introduction](#) 6

L

Local events
 [server](#) 38

M

[Message complex type](#) 13
Message processing
 [server](#) 21
Messages
 [attribute groups](#) 20
 [attributes](#) 20
 [Categories complex type](#) 12
 [Category simple type](#) 19
 [CategoryType complex type](#) 12
 [complex types](#) 11
 [ComplexType Messages complex type](#) 13
 [DesignCheckerInformation complex type](#) 13
 [elements](#) 11
 [enumerated](#) 11
 [Features simple type](#) 18
 [groups](#) 20
 [Message complex type](#) 13
 [MessageType simple type](#) 17
 [namespaces](#) 11
 [simple types](#) 17
 [SourceLocation complex type](#) 16
 [syntax](#) 11
 [transport](#) 11
[MessageType simple type](#) 17

N

[Namespaces](#) 11
[Normative references](#) 7

O

Operations
 [BrowserEnableUserFormTemplate](#) 22
 [DesignCheckFormTemplate](#) 24
 [GetListFormLocation](#) 26
 [GetUserCodeDeploymentDependencies](#) 29

[SetFormsForListItem](#) 31
[SetSchemaChangesForList](#) 34
[Overview \(synopsis\)](#) 8

P

[Parameters - security index](#) 48
[Preconditions](#) 9
[Prerequisites](#) 9
[Product behavior](#) 109
[Product messages](#) 57

R

References
[informative](#) 8
[normative](#) 7
[Relationship to other protocols](#) 9

S

Security
[implementer considerations](#) 48
[parameter index](#) 48
Sequencing rules
[server](#) 21
Server
[abstract data model](#) 21
[BrowserEnableUserFormTemplate operation](#) 22
[DesignCheckFormTemplate operation](#) 24
[details](#) 21
[GetListFormLocation operation](#) 26
[GetUserCodeDeploymentDependencies operation](#)
29
[initialization](#) 21
[local events](#) 38
[message processing](#) 21
[overview](#) 21
[sequencing rules](#) 21
[SetFormsForListItem operation](#) 31
[SetSchemaChangesForList operation](#) 34
[timer events](#) 38
[timers](#) 21
[SetFormsForListItem operation example](#) 42
[SetSchemaChangesForList operation example](#) 45
[Simple types](#) 17
[Category](#) 19
[Features](#) 18
[MessageType](#) 17
[SourceLocation complex type](#) 16
[Standards assignments](#) 10
Syntax
[messages - overview](#) 11

T

Timer events
[server](#) 38
Timers
[server](#) 21
[Tracking changes](#) 111
[Transport](#) 11

Types
[complex](#) 11
[simple](#) 17

V

[Vendor-extensible fields](#) 10
[Versioning](#) 10

W

[WSDL](#) 49