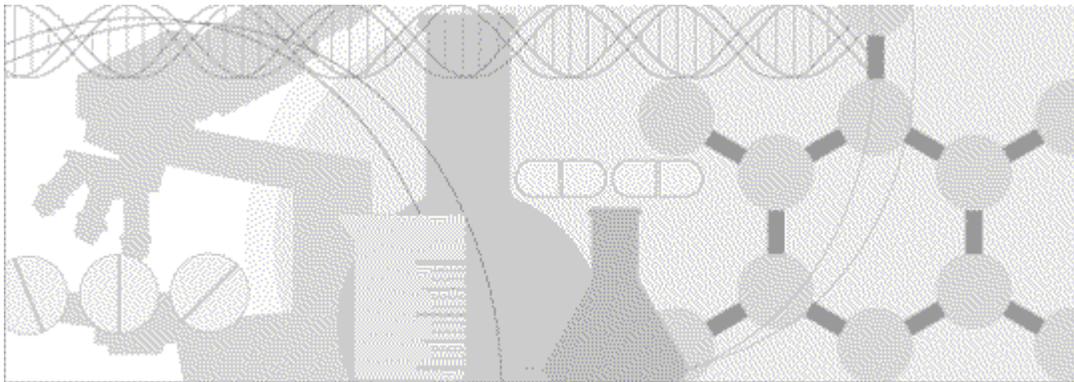


# InForm Clinical Data API Guide

Oracle<sup>®</sup> Health Sciences InForm 6.0



ORACLE<sup>®</sup>

Copyright © 2013, Oracle and/or its affiliates. All rights reserved.

The Programs (which include both the software and documentation) contain proprietary information; they are provided under a license agreement containing restrictions on use and disclosure and are also protected by copyright, patent, and other intellectual and industrial property laws. Reverse engineering, disassembly, or decompilation of the Programs, except to the extent required to obtain interoperability with other independently created software or as specified by law, is prohibited.

The information contained in this document is subject to change without notice. If you find any problems in the documentation, please report them to us in writing. This document is not warranted to be error-free. Except as may be expressly permitted in your license agreement for these Programs, no part of these Programs may be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose.

If the Programs are delivered to the United States Government or anyone licensing or using the Programs on behalf of the United States Government, the following notice is applicable:

U.S. GOVERNMENT RIGHTS Programs, software, databases, and related documentation and technical data delivered to U.S. Government customers are "commercial computer software" or "commercial technical data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, use, duplication, disclosure, modification, and adaptation of the Programs, including documentation and technical data, shall be subject to the licensing restrictions set forth in the applicable Oracle license agreement, and, to the extent applicable, the additional rights set forth in FAR 52.227-19, Commercial Computer Software -- Restricted Rights (June 1987). Oracle USA, Inc., 500 Oracle Parkway, Redwood City, CA 94065.

The Programs are not intended for use in any nuclear, aviation, mass transit, medical, or other inherently dangerous applications. It shall be the licensee's responsibility to take all appropriate fail-safe, backup, redundancy and other measures to ensure the safe use of such applications if the Programs are used for such purposes, and we disclaim liability for any damages caused by such use of the Programs.

The Programs may provide links to Web sites and access to content, products, and services from third parties. Oracle is not responsible for the availability of, or any content provided on, third-party Web sites. You bear all risks associated with the use of such content. If you choose to purchase any products or services from a third party, the relationship is directly between you and the third party. Oracle is not responsible for: (a) the quality of third-party products or services; or (b) fulfilling any of the terms of the agreement with the third party, including delivery of products or services and warranty obligations related to purchased products or services. Oracle is not responsible for any loss or damage of any sort that you may incur from dealing with any third party.

Oracle is a registered trademark of Oracle Corporation and/or its affiliates. Other names may be trademarks of their respective owners.

This documentation may include references to materials, offerings, or products that were previously offered by Phase Forward Inc. Certain materials, offerings, services, or products may no longer be offered or provided. Oracle and its affiliates cannot be held responsible for any such references should they appear in the text provided.

# Contents

<b>About this guide</b>	<b>v</b>
Overview of this guide.....	vi
Audience .....	vi
Related information.....	vii
Documentation.....	vii
If you need assistance.....	x
<b>Chapter 1 Clinical Data API</b>	<b>1</b>
Overview of the Clinical Data API .....	2
Clinical Data API.....	2
Message format.....	3
ODM Submit web service methods.....	4
ODM Submit elements.....	9
About ODM Submit schema elements .....	9
Non-supported elements and attributes.....	9
Supported elements and attributes.....	9
<b>Chapter 2 Running the ODM Submit utility from the command line</b>	<b>31</b>
ODM Submit utility command line parameters.....	32



# About this guide

## **In this preface**

Overview of this guide.....	vi
Related information.....	vii
If you need assistance.....	x

## Overview of this guide

The *Clinical Data API Guide* provides information about submitting data to the InForm application in InForm ODM format using the following:

- Clinical Data API—Web service interface that allows a remote InForm user to import clinical data using an API.
- ODM Submit utility—Command line tool that allows a local user to execute an import file in ODM format with InForm-specific extensions using the command line.

## Audience

This guide is for users who submit clinical data. Remote users; that is, users who do not have access to the physical machine on which the InForm application is installed, can submit clinical data using the Clinical Data API. Local users; that is, users who have access to the physical machine on which the InForm application is installed, can submit clinical data using the ODM Submit utility.

## Related information

### Documentation

All documentation is available from the Oracle Software Delivery Cloud (<https://edelivery.oracle.com>) and the Download Center (<https://extranet.phaseforward.com>).

Document	Description
<i>Release Notes</i>	The <i>Release Notes</i> document describes enhancements introduced and problems fixed in the current release, upgrade considerations, release history, and other late-breaking information.
<i>Known Issues</i>	<p>The <i>Known Issues</i> document provides detailed information about the known issues in this release, along with workarounds, if available.</p> <p><b>Note:</b> The most current list of known issues is available on the Extranet. To sign in to the Extranet, go to <a href="https://extranet.phaseforward.com">https://extranet.phaseforward.com</a>.</p>
<i>Upgrade and Migration Guide</i>	The <i>Upgrade and Migration Guide</i> provides instructions for upgrading and migrating the InForm software and InForm Portal software to the current InForm release, and for upgrading the Cognos 8 Business Intelligence software for use with the Reporting and Analysis module. The guide also describes any changes and additions made to the database schema, MedML, and resource files.
<i>Secure Configuration Guide</i>	The <i>Secure Configuration Guide</i> provides an overview of the security features provided with the Oracle® Health Sciences application, including details about the general principles of application security, and how to install, configure, and use the application securely.
<i>Installation Guide</i>	The <i>Installation Guide</i> describes how to install the software and configure the environment for the InForm application and Cognos 8 Business Intelligence software.
<i>Study and Reporting Setup Guide</i>	The <i>Study and Reporting Setup Guide</i> describes how to perform the tasks that are required to set up an InForm study and configure the Reporting and Analysis module for the study.
<i>User Guide</i>	<p>The <i>User Guide</i> provides an overview of the InForm application including details on multilingual studies, how to navigate through the user interface, and how to use the application to accomplish typical tasks you perform while running a clinical study.</p> <p>This document is also available from the Documentation CD and the InForm user interface.</p>

Document	Description
<i>Reporting and Analysis Guide</i>	<p>The <i>Reporting and Analysis Guide</i> provides an overview of the Reporting and Analysis module. It includes a brief overview of the Reporting and Analysis interface, illustrates how to access the Ad Hoc Reporting feature, and describes the study management and clinical data packages available for Reporting and Analysis. It also provides detailed descriptions of each standard report that is included with your installation.</p> <p>This document is also available from the Documentation CD.</p>
<i>Reporting Database Schema Guide</i>	<p>The <i>Reporting Database Schema Guide</i> describes the Reporting and Analysis database schema, and provides information on creating Reporting Database Extracts (RDEs).</p>
<i>Portal Administration Guide</i>	<p>The <i>Portal Administration Guide</i> provides step-by-step instructions for setting up the InForm Portal software, and configuring and managing the InForm Portal application.</p> <p>This document is also available from the Documentation CD.</p>
<i>InForm Utilities Guide</i>	<p>The <i>InForm Utilities Guide</i> provides information about and step-by-step instructions for using the following utilities:</p> <ul style="list-style-type: none"> <li>• PFConsole utility</li> <li>• MedML Installer utility</li> <li>• InForm Data Import utility</li> <li>• InForm Data Export utility</li> <li>• InForm Performance Monitor utility</li> <li>• InForm Report Folder Maintenance utility</li> </ul> <p>This guide also provides reference information for the MedML elements and scripting objects that are used to import and export data to and from the InForm application, as well as sample data import XML.</p> <p>This document is also available from the Documentation CD.</p>
MedML Installer utility online Help	<p>The MedML Installer utility online Help provides information about, and step-by-step instructions for using, the MedML Installer utility, which is used to load XML that defines study components into the InForm database.</p> <p>This guide also provides reference information for the MedML elements and scripting objects that are used to import and export data to and from the InForm application, as well as sample data import XML.</p> <p>This document is also available from the user interface.</p>

---

Document	Description
InForm Data Export utility online Help	<p>The InForm Data Export utility online Help provides information about and step-by-step instructions for using the InForm Data Export utility, which is used to export data from the InForm application to the following output formats:</p> <ul style="list-style-type: none"><li>• Customer-defined database (CDD).</li><li>• Name value pairs.</li></ul> <p>This document is also available from the user interface.</p>
InForm Data Import utility online Help	<p>The InForm Data Import utility online Help provides information about and step-by-step instructions for using the InForm Data Import utility, which is used to import data into the InForm application.</p> <p>This document is also available from the user interface.</p>

---

## If you need assistance

Oracle customers have access to support through My Oracle Support. For information, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=info>, or if you are hearing impaired, visit <http://www.oracle.com/pls/topic/lookup?ctx=acc&id=trs>.

# CHAPTER 1

# Clinical Data API

## In this chapter

Overview of the Clinical Data API.....	2
ODM Submit elements.....	9

# Overview of the Clinical Data API

The Clinical Data API is an application program interface (API) that allows remote users; that is, users who do not have access to the physical machine on which the InForm application is installed, to submit clinical data to the InForm application.

The format for this API is based on the Operational Data Model (ODM), which is a representation of clinical data created by the Clinical Data Interchange Standards Consortium (CDISC). The XML format that the InForm application accepts is called InForm ODM because it has InForm-specific extensions to the base ODM XML schema.

This document describes the InForm ODM format that can be submitted to the InForm application. The format is based on the ODM that is exported from the InForm application through the InForm Adapter application.

## Clinical Data API

The Clinical Data API is a web service interface that allows remote users to submit data to the InForm application. The web service interface has three methods, which support:

- Submission of clinical data.  
For more information, see *ODM Submit—Submission messages* (on page 4).
- Summary reporting for each ODM Submit document that was processed.  
For more information, see *ODM Submit—Document status messages* (on page 6).
- Detailed reporting of historical data.  
For more information, see *ODM Submit—Report messages* (on page 8).

All web service methods are synchronous for the remote client. Remote clients send request messages, and the web service returns a response message for each request that indicates success or identifies errors that occurred.

## Web service URL

The web service URL must resolve to a study name. Additional study name parameters supplied in the message ensure that the routing is correct.

All requests require secure HTTP (HTTPS).

A web service URL includes the following:

*<target web server address>/<study name>/<web service support directory>/<web service request handler>*

where:

- **Web Service Support Directory**— IIS web service support directory for InForm Core web services.
- **Web service request handler**—ODMSubmit.asmx.

## WSDL access

Clients may obtain the web service description language (WSDL) from the ODMSubmit.asmx file.

Deployments may also prevent access to the WSDL by changing the httpsGetEnabled parameter in the web.config file.

## Security and authentication

The InForm application uses a gateway that allows user access to the web service interfaces. This InForm application user does not require any specific study rights.

The user credentials for the gateway are supplied as parameters to the web service client proxy class. When authentication succeeds, the web method is processed.

The remaining security is provided through the XML Submit Clinical Data Interface authentication method.

## Creating the Clinical Data API user

You must create a user in the InForm application with rights to use the Clinical Data API. This user must:

- Be assigned the ODM Submit right. The user should not be assigned any other right.
- Not be associated with any sites.

For instructions for creating a user, see the *User Guide*.

## Message format

Each request is enclosed in a SOAP envelope that begins with a SOAP header, followed by a Body statement that contains the web method, and its input parameters and values.

### SOAP header

The SOAP header contains information about the service, the message, and sender credentials. It must include all the fields in the following table.

ODM Submit— SOAP Header		
Field	Description	Standard
wsa:Action	SOAP action for the operation. Include this field as part of the summary for each SOAP operation.	WS-Addressing
wsa:MessageID	Unique ID for the message.	WS-Addressing

### ODM Submit— SOAP Header

wsa:ReplyTo	Specify the following value:	WS-Addressing
	<pre>&lt;wsa:ReplyTo xmlns:wsa="http://www.w3.org/2005/08/addressing"&gt; &lt;wsa:Address&gt;http://www.w3.org/2005/08/addressing/anonymous&lt;/wsa:Address&gt; &lt;/wsa:ReplyTo&gt;</pre>	
wsa:To	Full URL of the service.	WS-Addressing
wsu:Timestamp	<p>Time the message was created and when it expires. Oracle recommends a five minute time window.</p> <p>The software allows for five minutes of clock drift between servers. You can modify this value in the web.config file by setting the MaxClockSkew value.</p>	SOAP Message Security 1.0
wsse:Username	User name of the user in the InForm study.	SOAP Message Security 1.0
wsse:Password	Password of the user in the InForm study.	SOAP Message Security 1.0

**Note:** The requirement for requests to contain the **wsu:Timestamp**, **wsse:Username**, and **wsse:Password** elements are defined in the WSDL by ws-policy. If your web service toolkit does not support ws-policy, you must manually add these elements to the web service request.

## ODM Submit web service methods

### ODM Submit—Submission messages

The WSODMXMLSubmit method accepts incoming messages to be processed by the Clinical Data API. It has two formats:

- Request format—For submitting data.  
For more information, see *SOAP body for Request message—Submissions* (on page 4).
- Response format—For returning the submit status of the operation.  
For more information, see *Response XML format—Submissions* (on page 5).

## SOAP body for Request message—Submissions

Request message parameters	
Name	Description
TrialName	Name of the study to which the data will be submitted.
UserName	Username of the InForm user who is submitting the clinical data.
OrgID	Reserved for future use. Leave blank.
UserPassword	Password of the InForm user who is submitting the clinical data.
RulesEnabled	Specifies whether rules should be executed when processing the ODM Submit XML document. <ul style="list-style-type: none"> <li>• True—Execute rules.</li> <li>• False—Do not execute rules.</li> </ul>
StopOnError	Specifies action to take if any individual submit operation fails during processing. <ul style="list-style-type: none"> <li>• True—Stop processing.</li> <li>• False—Continue processing the document.</li> </ul>
SubmitToLockedOrFrozenForms	Allows data to be submitted to locked or frozen forms.
ODMSubmitDocument	ODM Submit XML document.

## Response XML format—Submissions

Response message parameters	
Name	Description
WSODMXMLSubmitResult	<p>Top-level element that indicates whether the request was processed.</p> <ul style="list-style-type: none"> <li>• If PROCESSED=TRUE, this contains the actual results.</li> <li>• If PROCESSED=FALSE, the message could not be processed.</li> </ul>
Error	If the WSODMXmlSubmitResult attribute PROCESSED=FALSE, this element contains information about why the web message could not be processed.
SubmitResults	If WSODMXmlSubmitResult attribute PROCESSED=TRUE, this parameter exists.
SubmitStatus	If WSODMXmlSubmitResult attribute PROCESSED=TRUE, this parameter exists.
MessageID	<p>An HRESULT error code, which can occur at the web service or the Clinical Data API interface.</p> <p><b>Note:</b> Occurs only when the Status attribute for the SubmitStatus element is FAILED.</p>
Message	<p>English text of the MessageID.</p> <p><b>Note:</b> Occurs only when the Status attribute for the SubmitStatus element is FAILED.</p>
ActivityGUID	<p>Corresponds to the InForm ActivityGUID when an error is written to the event log.</p> <p><b>Note:</b> Occurs only when the Status attribute for the SubmitStatus element is FAILED.</p>

## ODM Submit—Document status messages

The WSODMXMLSubmitStatus method accepts an XML Submit Document Identifier (FileOID in ODM terminology) and returns a summary status of the given FileOID. This method handles scenarios in which the web service connection timed out or the submit operation took too long.

### SOAP body for Request messages—Document status

Request message parameters	
Name	Description
TrialName	Name of the study to which the data is submitted.
UserName	Username of the InForm user who is submitting the clinical data.
OrgID	Reserved for future use. Leave blank.
UserPassword	The password of the InForm user who is submitting the clinical data.
ODMSubmitDocumentID	The ODM Submit document unique identifier.

### Response XML format—Document status

Response message parameters	
Name	Description
WSODMXMLSubmitResult	<p>Top-level element that indicates whether the request was processed.</p> <ul style="list-style-type: none"> <li>• If PROCESSED=TRUE, this contains the actual results.</li> <li>• If PROCESSED=FALSE, the message could not be processed.</li> </ul>
Error	If the WSODMXmlSubmitResult attribute PROCESSED=FALSE, this element contains information about why the web message could not be processed.
SubmitStatus	If WSODMXmlSubmitResult attribute PROCESSED=TRUE, this parameter exists.

## ODM Submit—Report messages

WSODMXMLSubmitSummary method accepts an XML Submit Document Identifier (FileOID in ODM terminology), and returns a summary status, and if available, a complete list of individual submit statuses.

This method is primarily used for remote clients to diagnose submit operations.

**Note:** The individual submit commands may not be available because the background purging job may have removed the detailed submit records from the logging facility.

## SOAP body for Request messages—Reports

Request message parameters	
Name	Description
TrialName	Name of the study to which the data is submitted.
UserName	Username of the InForm user who is submitting the clinical data.
OrgID	Reserved for future use. Leave blank.
UserPassword	The password of the InForm user who is submitting the clinical data.
ODMSubmitDocumentID	The ODM Submit document unique identifier.

## Response XML format—Reports

Response message parameters	
Name	Description
WSODMXMLSubmitResult	<p>Top-level element that indicates whether the request was processed.</p> <ul style="list-style-type: none"> <li>• If PROCESSED=TRUE, this contains the actual results.</li> <li>• If PROCESSED=FALSE, the message could not be processed.</li> </ul>
Error	If the WSODMXmlSubmitResult attribute PROCESSED=FALSE, this element contains information about why the web message could not be processed.
SubmitStatus	If WSODMXmlSubmitResult attribute PROCESSED=TRUE, this parameter exists.

# ODM Submit elements

## About ODM Submit schema elements

This section describes the ODM elements that are supported in InForm ODM Submit. It includes base ODM elements and InForm extensions. The InForm extensions are part of the XML schema for InForm Adapter ODM Export, unless they are specifically listed as new extensions for InForm ODM Submit.

- In most cases, base ODM elements and attributes do not have a prefix.
- The prefix **pf:** identifies InForm-specific elements and attributes.
- If an InForm-specific element has a base ODM sub-element or attribute, the base ODM element or attribute may be prefixed by **odm:** for clarity.

General XML notes:

- Attributes must be escaped.
- Element contents must either be escaped or enclosed in a CDATA section.
- The first line of the file must be an XML tag that specifies the file encoding. Only UTF-8 encoding is supported, so the first line of the file must be:

```
<?xml version="1.0" encoding="UTF-8"?>
```

## Non-supported elements and attributes

- All mandatory attributes and elements are supported in the ODM Submit XML.
- The description of each element includes a list of attributes and elements that are not supported.
- If any non-supported attributes or elements are present, an error message is generated and the ODM is not submitted.

## Supported elements and attributes

### Annotation (Comment)

For comments and incomplete reasons. The Annotation element can appear as a sub-element of:

- ItemData—For an item comment.
- FormData—For a form comment.

For item comments, the ItemData in which the annotation is contained must be an item path, not a control path. For more information, see *ItemData* (on page 15).

**Note:** Annotation is also a sub-element of the pf:Association element. For more information, see *Annotation (Sub-element of pf:Association)* (on page 10).

**Example**

```
<Annotation SeqNum="1" pf:DBUID="30087"
  pf:GUID="{885FAAB2-3997-482E-9179-97560DAA33B2}">
  <Comment>item comment when item has many sub-controls</Comment>
</Annotation>
```

**Supported sub-elements**

- Comment—Optional.  
Zero or one instance.
- pf:ReasonIncomplete—Optional.  
Zero or one instance.

**Supported attributes**

- SeqNum—Required.  
Value is always 1 because it is used for InForm form comments or item comments, and a form or item can have only one comment.

**Non-supported elements**

- Flag
- pf:AnnotationAuditData
- pf:AnnotationAuditRecord

**Non-supported attributes**

- TransactionType
- pf:DBUID
- pf:GUID

**Annotation (Sub-element of pf:Association)**

Sub-element of pf:Association.

**Example**

```
<Annotation SeqNum="33663" pf:DBUID="33663" pf:GUID="{E34262AA-A279-4BA4-
B8BA-DDABE6CC52BB}">
  <Comment>PF Association</Comment>
  <Flag>
    <FlagValue CodeListOID="AssociationAction">Add</FlagValue>
    <FlagType CodeListOID="FlagType">Association</FlagType>
  </Flag>
</Annotation>
```

## Supported sub-elements

- Flag—Required.

When Annotation is used as a sub-element of pf:Association, the Annotation element must have a Flag sub-element, which in turn must have the sub-element FlagValue with CodeListOID="AssociationAction" (whose values can be Add or Remove).

## pf:Association

Used to add or remove associations between repeating forms. The pf:Association element is a sub-element of the top-level ODM element.

### Example

```
<Association StudyOID="Depression Study"
  MetaDataVersionOID="StudyVersion_ODMTEST03">
  <KeySet StudyOID="Depression Study"
    SubjectKey="29530"
    StudyEventOID="Visit1" StudyEventRepeatKey="342216927026073"
    FormOID="LAE1" FormRepeatKey="343316719654073" />
  <KeySet StudyOID="Depression Study"
    SubjectKey="29530"
    StudyEventOID="Visit1" StudyEventRepeatKey="342216927026073"
    FormOID="L_ConMeds" FormRepeatKey="343316927026073" />
  <Annotation SeqNum="33663" pf:DBUID="33663" pf:GUID="{E34262AA-A279-4BA4-
    B8BA-DDABE6CC52BB}">
    <Comment>PF Association</Comment>
    <Flag>
      <FlagValue CodeListOID="AssociationAction">Add</FlagValue>
      <FlagType CodeListOID="FlagType">Association</FlagType>
    </Flag>
  </Annotation>
</Association>
```

## Supported sub-elements

- KeySet—Required.  
Exactly two occurrences.
- Annotation—Required.  
Exactly one occurrence.

## Supported attributes

- StudyOID—Required.  
Study name. Must be current and must match the study name (STUDYNAME attribute of STUDYVERSION MedML).
- MetaDataVersionOID—Required.  
Study version that is assigned to the site. This does not have to be the current study version.

## Non-supported elements and attributes

- None.

## pf:Candidate

Contains properties of subjects during screening and enrollment.

### Example

```
<pf:Candidate EnrollmentOverride="Date: 11/9/2011 At: 02:03 Monitor: Louise Hill Title: CRA Reason: Subject nearly met criteria" />
```

### Supported sub-elements

- None

### Supported attributes

- EnrollmentOverride—Required.

This text replaces the text that is created by the InForm application.

### Non-supported elements

- AuditRecord

### Non-supported attributes

- pf:DateOfBirth
- pf:EnrollmentFailure
- pf:EnrollmentNumber
- pf:PatientNumber
- pf:PatientRevision
- pf:ScreeningDate
- pf:ScreeningFailure
- pf:ScreeningNumber
- pf:SubjectInitials

## ClinicalData

High-level element contained by the ODM element. It can contain data for multiple subjects, each in its own SubjectData sub-element.

### Example

```
<ClinicalData StudyOID="Depression Study" MetaDataVersionOID="Study Version 1"
  pf:TransactionGuid="{69B04136-0872-464A-8870-A0BDA864943C}">
  <SubjectData ... details omitted
</ClinicalData>
```

**Supported sub-elements**

- SubjectData—Optional.  
Zero or more instances. Each instance can be for a different subject.

**Supported attributes**

- StudyOID—Required.  
Study name. Must be current and must match the study name (STUDYNAME attribute of STUDYVERSION MedML).
- MetadataVersionOID—Required.  
Study version that is assigned to the site. This does not have to be the current study version.

**Comment**

Sub-element of Annotation, used for item and form comments.

**Example**

```
<Comment>item comment when item has many sub-controls</Comment>
```

**Supported sub-elements**

- None.

**Supported attributes**

- None.

**Non-supported elements**

- None.

**Non-supported attributes**

- SponsorOrSite

## FormData

Contained by StudyEventData and has data for one form instance for one subject.

### Supported sub-elements

- Annotation—Optional.  
Zero or one instance.
- ItemGroupData—Optional.  
Zero or more instances.
- pf:FormStatus—Optional.  
Zero or one instance.

### Supported attributes

- FormOID—Required.  
Value is the form RefName.
- FormRepeatKey—Optional.
  - For non-repeating forms—Do not use.
  - For repeating forms—Required. A non-null string that identifies the repeating form instance uniquely within the subject visit instance.
- pf:FormRepeatKeyOrdinal—Optional.
  - For non-repeating forms—Do not use.
  - For repeating forms—Optional.  
If used, it is the 1-based ordinal of the repeating form instance. All instances of the repeating form within a specific subject visit instance are counted when computing an ordinal, including all prior repeating forms whether or not they specified the pf:FormRepeatKeyOrdinal (optional attribute), and including deleted repeating form instances.

### Non-supported elements

- ArchiveLayoutRef
- AuditRecord
- Signature
- pf:FormAuditData
- pf:PFFormSignatures

### Non-supported attributes

- TransactionType

## pf:FormStatus

An InForm-specific sub-element of FormData, used to specify form status for a single form instance.

### Supported sub-elements

- None.

### Supported attributes

- Frozen—Optional.
- Locked—Optional.
- Deleted—Optional.
  - Yes—Delete a repeating form.
  - No—Undelete a repeating form.
- SVReady—Optional.  
Form state that is set for a CRF in the user interface by the action Mark SV Ready.
- The following attributes are set if the review state for a form is changed:
  - ReviewStageRefName—Optional.  
RefName of the REVIEWSTATE MedML element.
  - ReviewState—Optional.  
STATE attribute of the REVIEWSTATE MedML element. Value can be 1, 2, 3, 4, or 5.
  - ReviewStage—Optional.  
STAGE attribute of the REVIEWSTAGE MedML element. Value can be 0, 1, or 2.

**Note: The ReviewStageRefName and ReviewState are redundant. You can compute one from the other.**

### Non supported elements

- AuditRecord

### Non supported attributes

- pf:Activated
- pf:ReviewStateRevision
- pf:Signed
- pf:SVComplete
- pf:SVPartial
- pf:SVSelected

## ItemData

Sub-element of ItemGroupData that can be used to specify an item comment or the value of one InForm control.

### Example

```
<ItemData ItemOID="frmDOV.DOV.DOV.DOV"  
  Value="2011-10-19T09:17">  
</ItemData>
```

### Supported sub-elements

- MeasurementUnitRef—Optional.  
Zero or one instance.
- Annotation—Optional.  
Zero or one instance.

### Supported attributes

- ItemOID—Required.  
The RefName path from the form to the item or control.
  - For non-repeating items—FORM.SECTION.ITEM.[CONTROL]
  - For itemset items—FORM.SECTION.ITEMSET.ITEM.[CONTROL]
  - The ItemData element is contained in ItemGroupData, which also has a RefName path; the two paths must not conflict.
- Value—Optional.
- pf:FormattedDateValue—Optional.  
pf:FormattedDateValue differentiates between date components that have no value and date components that are set to Unknown. If pf:FormattedDateValue is present, it must not conflict with the date in the Value attribute.
  - The parts of the date that are supplied in the Value attribute must exactly match pf:FormattedDateValue.
  - The parts of the date that are unspecified in the Value attribute must be unspecified in pf:FormattedDateValue. However, in the pf:FormattedDateValue attribute, unspecified date parts will be identified as either **No value supplied** or **Date part unknown**.

**Non-supported elements**

- AuditRecord
- Signature

**Non-supported attributes**

- IsNull
- pf:ItemAuditData
- pf:NormalizedValue
- pf:ReasonIncomplete
- pf:TransactionType

**ItemGroupData**

Element contained by FormData that has data for one section of an InForm form instance.

**Example**

```
<ItemGroupData ItemGroupOID="frmDOV.DOV">
  <ItemData ItemOID="frmDOV.DOV.DOV.DOV"
    Value="2011-10-19T09:17">
  </ItemData>
</ItemGroupData>
```

**Supported sub-elements**

- ItemData—Optional.  
Zero or more instances, in any order.
- pf:ItemStatus—Optional.  
Zero or one instance.

**Supported attributes**

- ItemGroupOID—Required.  
REFNAME path:
  - For a non-repeating item—FORM.SECTION
  - For an itemset—FORM.SECTION.ITEMSET

ItemGroupData always appears within the FormData element, which also specifies the form RefName. The form RefName in the FormData element must match the form RefName in the ItemGroupOID attribute.

- ItemGroupRepeatKey—Optional.
  - For a non-repeating item—Not required.
  - For an item in an itemset—Required.  
A non-null string that uniquely identifies an itemset row within a form instance.
- pf:ItemGroupRepeatKeyOrdinal—Optional.  
If present, this is the 1-based ordinal value of the itemset row in the form instance. All rows in

the itemset are included when figuring the ordinal value of a row, including all prior rows, whether or not they had the pf:ItemGroupRepeatKeyOrdinal attribute, and any deleted Add Entry itemset rows.

### Non-supported elements

- AuditRecord
- Signature
- TransactionType
- pf:InFormItemData
- pf:Query

### pf:ItemStatus

Sub-element of ItemGroupData.

#### Example

```
<pf:ItemStatus Deleted="No">
</pf:ItemStatus>
```

### Supported sub-elements

- None

### Supported attributes

- Deleted—Required.
  - Yes—Delete an Add Entry itemset row.
  - No—Undelete an Add Entry itemset row.

### pf:KeySet

Sub-element of the pf:Association element. Exactly one occurrence is required.

#### Example

```
<Association StudyOID="Depression Study"
  MetadataVersionOID="StudyVersion_ODMTEST03">
  <KeySet StudyOID="Depression Study"
    SubjectKey="29530"
    StudyEventOID="Visit1" StudyEventRepeatKey="342216927026073"
    FormOID="LAE1" FormRepeatKey="343316719654073" />
  <KeySet StudyOID="Depression Study"
    SubjectKey="29530"
    StudyEventOID="Visit1" StudyEventRepeatKey="342216927026073"
    FormOID="L_ConMeds" FormRepeatKey="343316927026073" />
  <Annotation SeqNum="33663" pf:DBUID="33663" pf:GUID="{E34262AA-A279-4BA4-
    B8BA-DDABE6CC52BB}">
    <Comment>PF Association</Comment>
    <Flag>
    <FlagValue CodeListOID="AssociationAction">Add</FlagValue>
    <FlagType CodeListOID="FlagType">Association</FlagType>
    </Flag>
  </Annotation>
</Association>
```

### Supported attributes

- StudyOID—Required.
- SubjectKey—Required.
- SubjectKeyType—Optional.  
Either GUID (the default) or the SubjectNumber.
- StudyEventOID—Required.  
Visit reference name.
- StudyEventRepeatKey—Optional.  
Must be present for repeating visits.
- FormOID—Required.  
Form reference name.
- FormRepeatKey—Required.

### ODM

The top-level element for any ODM document. The ODM element has several defined sub-elements, but only ClinicalData is supported for InForm ODM format.

This element must match the name for the study, which is specified as the STUDYNAME attribute of the STUDYVERSION MedML element. The InForm ODM format input is rejected if the values do not match.

### Example

```
<pf:ODMSubmit Description="Depression Study" FileType="Transactional"
  pf:HierarchicalOIDs="Yes"
  xmlns="http://www.cdisc.org/ns/odm/v1.3"
  xmlns:pf="http://www.phaseforward.com/InFormAdapter/ODM/Extensions/2.0">
  <ClinicalData ... details omitted
</pf:ODMSubmit>
```

### Supported sub-elements

- ClinicalData—Optional.  
Zero or more instances.

### Supported attributes

- ODMVersion—Required.  
ODM version number. Must be 1.3.1.
- FileType—Required.  
Type of the file. Must be Transactional
- FileOID—Required.  
Uniquely identifies each distinct bit of ODM.
- PriorFileOID—Optional.  
Enables the client to uniquely order bits of ODM input.  
  
The ODM records the FileOID for each bit of ODM processed. If the specified PriorFileOID does not match the FileOID of an already-processed bit of ODM, the submit action fails.
- CreationDateTime—Required.  
Date and time of creation. This is required by ODM, but is not used by the Clinical Data API.
- pf:HierarchicalOIDs—Required.  
Must be Yes.

### Non-supported elements

- AdminData
- ReferenceData
- Study

### Non-supported attributes

- Archival
- AsOfDateTime
- Association
- Description
- Granularity
- Id
- Originator
- SourceSystem
- SourceSystemVersion
- pf:InFormAdapterVersion

- pf:MappingVersion

## pf:ReasonIncomplete

Custom InForm sub-element of the ODM Annotation element, used for marking items and forms incomplete.

The incomplete reason is specified by putting the selection value of the InForm incomplete reason control as the body of the pf:ReasonIncomplete element, not as an attribute.

### Example

```
<pf:ReasonIncomplete
  CodeListOID="ReasonIncomplete">NAElement</pf:ReasonIncomplete>
```

### Supported sub-elements

- None

### Supported attributes

- CodeListOID—Required.  
Must be ReasonIncomplete.

## SiteRef

Identifies a site within the study.

### Example

```
<SiteRef LocationOID="BID" />
```

### Supported sub-elements

- None

### Supported attributes

- LocationOID—Required.  
Must be the site mnemonic.

## StudyEventData

A high-level element, contained by SubjectData, which is used to specify a single visit instance for one subject.

### Example

```
<StudyEventData StudyEventOID="vstUnschVisit1"
  StudyEventRepeatKey="3">
  <FormData FormOID="frmDOVRFI">
    ...Details omitted
  </FormData>
</StudyEventData>
```

**Supported sub-elements**

- FormData—Optional.  
Zero or more instances.

**Supported attributes**

- StudyEventOID—Required.  
RefName of the visit.
- StudyEventRepeatKey—Optional.
  - For non-repeating visits—Do not use.
  - For repeating visits—Required.  
A non-null string that uniquely identifies a given instance of this repeating visit for this subject.
- pf:StudyEventRepeatKeyOrdinal—Optional.
  - For non-repeating visits—Do not use.
  - For repeating visits—Required.  
The 1-based ordinal index of this repeating visit for this subject.

**Non-supported elements**

- Annotation
- AuditRecord
- Signature

**Non-supported elements and attributes**

- TransactionType
- pf:Deleted

**SubjectData**

An element, contained by ClinicalData, which has data for one subject.

**Example**

```
<SubjectData TransactionType="Upsert"
  SubjectKey="{B6E65FC3-0E73-4DA9-898A-8A11D6BCC3B5}" >
  <SiteRef LocationOID="BID" />
  <StudyEventData StudyEventOID="vstCORE2" />
</SubjectData>
```

**Supported sub-elements**

- SiteRef—Optional.  
Zero or one instance. Defines the site for the subject.

Although the SiteRef element is optional in the ODM schema, it is required for InForm ODM input. To transfer a subject to a new site, supply a SubjectData element and have the SiteRef

refer to the new site.

- StudyEventData—Optional.  
Zero or more instances. Has data for one visit instance.
- pf:Candidate—Optional.  
Zero or one instance.
- pf:SubjectStatus—Optional.  
Zero or one instance.

### Supported attributes

- SubjectKey—Required.  
A unique identifier for the subject. For more information, see *SubjectKey and SubjectKeyType attributes* (on page 24).
- SubjectKeyType—Optional.  
Indicates how to interpret the SubjectKey attribute. For more information, see *SubjectKey and SubjectKeyType attributes* (on page 24).

### Non-supported elements

- Annotation
- AuditRecord
- InvestigatorRef
- Signature
- StudyEventStatus

### Non-supported attributes

- TransactionType
- pf:GUID

## pf:SubjectStatus

Used to set several subject-level states.

### Example

```
<pf:SubjectStatus State="Enrolled">
</pf:SubjectStatus>
```

### Supported sub-elements

- None

### Supported attributes

- Frozen—Optional.
- Locked —Optional.
- SVReady—Optional.
- EnrollOverrideReason—Optional.
- State—Optional.

Used to enroll a subject, using State=Enrolled.

Most subject states cannot be set directly, because they are computed through entered CRF data. You cannot use the State attribute to set the subject to the following states:

- ScreenFailed
- Screened
- EnrollmentFailed
- Randomized
- Complete
- DroppedOut

### SubjectKey and SubjectKeyType attributes

Subjects are identified using the standard ODM attribute SubjectKey (within the SubjectData element). Specify the value using either the subject GUID or the InForm Subject Number.

- GUID—The SubjectKey attribute is a unique, unchanging identifier for the subject that is used as the InForm internal GUID for the subject.
  - The GUID can be any string that is unique within the study. It must be a Unicode string occupying no more than 255 bytes in the database UTF-8 character set.
  - Clinical Data API users who are doing the initial screening for a subject and want to identify subjects by GUID must create the GUID manually.
  - Clinical Data API users working with existing subjects must retrieve the subject GUID from the InForm database.

- **InForm Subject Number**—This is the existing Subject Number that appears in the InForm user interface to identify enrolled subjects. The InForm application allows you to change the Subject Number after enrollment (using the Subject Identification form). Therefore, if you use the Subject Number to identify subjects, you must ensure that the ODM Submit document always uses the current Subject Number at the time the ODM Submit occurs.

You can also use the optional attribute `pf:SubjectKeyType` to specify the type of subject identification used. The value can be:

- GUID (the default)
- InForm Subject Number

## Value attribute for ItemData element

The value of any InForm control is specified in Value attribute of the ItemData element.

### Text Controls

For text controls, the ItemData Value attribute has the text, with XML escaping if necessary.

The values entered for an InForm text control must conform to the following MedML attributes specified for the control, which can affect its value:

- MAXLENGTH
- DATATYPE
- MINVALUE
- MAXVALUE
- PRECISION
- CHARSET

### Pulldown Controls

For pulldown controls, the Value attribute has the VALUE of the selected element, not the LABEL.

### Radio Controls

This ODM input sets a radio control to the sub-control with selection value 2:

```
<ItemData ItemOID="frmA.sctB.itmC.radioD" Value="2"/>
```

This ODM input changes that same radio control selection from 2 to 3 (clear one radio button and select a different one).

```
<ItemData ItemOID="frmA.sctB.itmC.radioD" Value="3"/>
```

This sets the radio control to have no selection (clears the radio control):

```
<ItemData ItemOID="frmA.sctB.itmC.radioD" IsNull="Yes"/>
```

### Date/Time Controls

The Value of a date/time control conforms to the ISO 8601 standard.

Here is the simplest example of the ODM Submit for a date control, where all components of the

date/time are specified, so that the attribute Value is all that is needed:

```
<ItemData ItemOID="frmA.sctB.itmC.dtcDateD" Value="2011-10-21T12:44" />
```

This date/time represents October 21, 2011 at 12:44 in an unspecified time zone.

InForm date/time controls do not contain information about the timezone of the date/time.

Importing dates to the InForm application may require using two attributes:

- Value
- pf:FormattedDateValue.

Value is the base ODM attribute and is sufficient for some InForm dates but cannot distinguish between date components that are not entered and those that are entered as Unknown.

For the ODM Value attribute, any missing date component is represented with a single dash character (-). For example, the following ODM represents a date control in which the month and hour are not specified:

```
<ItemData ItemOID="frmA.sctB.itmC.dtcDateD" Value="2004---01T-:02 " />
```

If a date/time control is input and the pf:FormattedDateValue attribute is not present, then any date/time components that are represented with a dash will be treated as not entered in the InForm application.

The following example shows the ODM Input for a date/time control where the month and hour are Unknown, and the day and minute are not entered:

```
<ItemData ItemOID="frmA.sctB.itmC.dtcDateD "
Value="2015----T-:-"
pf:FormattedDateValue="2015-UNK-NULTUNK:NUL:NUL" />
```

The pf:FormattedDateValue element has the value UNK for parts that are Unknown and a value NUL for parts that were not specified.

The value entered for a date/time control must conform to the MedML attributes of the specific version of the control being entered, in particular whether each part of the date can be omitted or specified as Unknown.

## Checkbox Controls

The InForm application and the InForm ODM input format have two subtly different styles of using checkboxes:

- One could have a group control with multiple independent (single-option) checkbox sub-controls.
- One can have a single multiple-option checkbox control.

The ODM Submit format for checkbox controls is the same as the ODM export format.

## Multi-option Checkbox Controls

Consider a multi-select checkbox called MULTICB with three sub-controls:

- Selection value SELECT1 has sub-control TEXTCONTROL TEXT1.
- Selection value SELECT2 has sub-control TEXTCONTROL TEXT2.
- Selection value SELECT3 has sub-control TEXTCONTROL TEXT3.

The following ODM Submit XML selects checkboxes 1 and 3, and puts a value in the text sub-control of each. Checkbox 2 is not selected.

```
<ItemData ItemOID="frmA.sctB.itmC.MULTICB.SELECT1" Value="SELECT1"/>
<ItemData ItemOID="frmA.sctB.itmC.MULTICB.SELECT3" Value="SELECT3"/>
<ItemData ItemOID="frmA.sctB.itmC.MULTICB.SELECT1.TEXT1" Value="text value
1"/>
<ItemData ItemOID="frmA.sctB.itmC.MULTICB.SELECT3.TEXT3" Value="text value
3"/>
```

Note that Checkbox 2 is not present, since it had no value.

The following ODM represents a change in checkbox state to de-select selection value 1 and select values 2 and 3:

```
<ItemData ItemOID="frmA.sctB.itmC.MULTICB.SELECT1" IsNull="true"/>
<ItemData ItemOID="frmA.sctB.itmC.MULTICB.SELECT2" Value="SELECT2"/>
<ItemData ItemOID="frmA.sctB.itmC.MULTICB.SELECT2.TEXT2" Value="text value
2"/>
```

The IsNull="true" setting for selection value 1 clears that selection (the text sub-control TEXT1 is implicitly cleared because its parent is cleared); since selection value 3 is not present in the ODM, it does not change, so it remains selected.

The following ODM XML represents a change in the checkbox state to clear all three checkboxes:

```
<ItemData ItemOID="frmA.sctB.itmC.MULTICB.SELECT1" IsNull="true"/>
<ItemData ItemOID="frmA.sctB.itmC.MULTICB.SELECT2" IsNull="true"/>
<ItemData ItemOID="frmA.sctB.itmC.MULTICB.SELECT3" IsNull="true"/>
```

## Single-option Checkbox Controls

For example, suppose there is a checkbox control with just one option with selection value 1. The following represents the checkbox selected:

```
<ItemData ItemOID="frmA.sctB.itmCheckBoxMulti.cbcSingleSelect.1" Value="1"/>
```

The following represents a change in checkbox state to deselected:

```
<ItemData ItemOID="frmA.sctB.itmCheckBoxMulti.cbcSingleSelect "
IsNull="Yes"/>
```

## Group Controls

Group controls have one or more sub-controls. Only the sub-controls have ODM ItemData elements for ODM Submit; the group control itself does not.

This example shows a group control that has three text box sub-controls, where the initial submit fills in values for the first and third sub-controls:

```
<ItemGroupData ItemGroupOID="frmECG.sctECG">
<ItemData ItemOID="frmECG.sctECG.itml.gc1.txtSample1" Value="xyz"/>
<ItemData ItemOID="frmECG.sctECG.itml.gc1.txtSample3" Value="abc"/>
</ItemGroupData>
```

There is no ODM XML for the second sub-control, which was initially not selected.

If the group control is edited to remove the value for the first text control and add a value for the second text control, the following ODM XML should be submitted:

```
<ItemGroupData ItemGroupOID="frmECG.sctECG">
<ItemData ItemOID="frmECG.sctECG.itml.gc1.txtSample1" IsNull="true"/>
<ItemData ItemOID="frmECG.sctECG.itml.gc1.txtSample2" Value="pqr"/>
</ItemGroupData>
```

There is no ODM XML for the third text control, which did not change. Its value remains unchanged.

## Missing values and the use of the IsNull attribute

If data is missing:

- For a data point that was never collected, do not include the ItemData element for that control.
- For a data point that was collected, but then cleared, set the IsNull attribute of the ItemData to Yes.
- The Value and IsNull attributes are mutually exclusive; an error occurs if both are set.
- Use IsNull="Yes" when a control is cleared.
- If you set IsNull="Yes" for a control, the control data is cleared.
- The Value attribute is checked against the type of the ItemData, and if it is incorrect, an error occurs.

## Units and normalized values

ODM export has values with units but also normalized values using the InForm-specific `pf:NormalizedValue` optional attribute.

ODM input does not allow you to submit normalized values. Any control that has units must specify those units during submit to specify a complete control.

Units can be omitted, but it will leave the control as not complete.



## CHAPTER 2

# Running the ODM Submit utility from the command line

### **In this chapter**

ODM Submit utility command line parameters.....	32
---	----

## ODM Submit utility command line parameters

Local users can run the ODM Submit utility from the command line to submit clinical data to the InForm application in InForm ODM format. To run the ODM Submit utility from the command line, use the following parameters.

Parameter	Variable	Description
ODMSubmit.exe		Starts the ODM Submit utility.
-trialname	<i>trialname</i>	The study into which you are importing data. Use the full pathname of the study.
/accountparams:p <i>ath_to_password_file</i>		When specified, includes the path to a text file that contains the user name and passwords required to run the command.  If the accountparams option is not specified, the command prompts for the required user names and passwords.  The format of the parameter file is parameter=value. There is a new line for each parameter, and there are no spaces on a line.
-verbose		The ODM Submit utility will write detailed messages as it processes the input file.
-ValidateXML		Checks to make sure that all required XML tags exist and the specified control paths can be found, without loading data. Optionally, use this parameter to validate an XML file before importing it.
-StopOnError		The ODM Submit utility will stop processing when it encounters an error. When errstop is not specified, the tag containing the error is skipped and the import continues with the next data tag in the file.
-RulesDisabled		Rules will not run during the submit. This parameter is required.
- SubmitToLocked OrFrozenForms		Allows you to submit data to a locked or frozen form. If you do not enable this option, an error occurs when you attempt to submit data to a locked or frozen form.
-File	<i>filename</i>	File name for the ODM Submit document.

---

Parameter	Variable	Description
-FileList	<i>filename</i>	<p>File name for the document that contains a list of ODM Submit file names.</p> <p>For example:</p> <pre>&lt;?xml version="1.0" encoding="utf-8"?&gt; &lt;File&gt; &lt;File&gt;firstFileName.xml&lt;/File&gt; &lt;File&gt;secondFileName xml&lt;/File&gt; &lt;/File&gt;</pre>

---