[MS-SSISPARAMS-Diff]:

Integration Services Project Parameter File Format

Revision Summary

Date	Revision History	Revision Class	Comments
~~07/07~~7/7/2011	0.1	New	Released new document.
11/~~03~~3/2011	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
~~01~~1/19/2012	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
~~02~~2/23/2012	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
~~03~~3/27/2012	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
~~05~~5/24/2012	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
~~06~~6/29/2012	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
~~07~~7/16/2012	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
10/~~08~~8/2012	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
10/23/2012	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
~~03~~3/26/2013	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
~~06~~6/11/2013	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
~~08/08~~8/8/2013	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
12/~~05~~5/2013	0.1	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
~~02~~2/11/2014	1.0	Major	~~Significantly changed~~Updated and revised the technical content.
~~05~~5/20/2014	1.0	~~No change~~None	No changes to the meaning, language, or formatting of the technical content.
5/10/2016	2.0	Major	Significantly changed the technical content.

Table of Contents

1 Introduction 4
1.1 Glossary..4
References 4
1.2 4
1.2.1 Normative References 5
1.2.2 Informative References 5
1.3 Overview..5
1.4 Relationship to Protocols and Other Structures..5
1.5 Applicability Statement 5
1.6 Versioning and Localization 5
1.7 Vendor-Extensible Fields 5

2 Structures 6
2.1 Project Parameter File..6
2.1.1 XML Namespace 6
2.1.2 ParametersType 6
2.1.3 ParameterType..6
2.1.4 ParameterPropertiesType...7
2.1.5 ParameterPropertyType 7

3 Structure Examples 9

4 Security 10
4.1 Security Considerations for Implementers 10
4.2 Index of Security Fields... 10

5 Appendix A: XML Schema Definition 11

6 Appendix B: Product Behavior 12

7 Change Tracking...13

8 Index...15

1 ~~1~~ Introduction

The Integration Services project parameter file format enables the capture of metadata about the parameters that may be supplied to an Integration Services project when it is configured or executed. The **project parameter file** is a file type that is used to store the metadata for a **project connection manager**.

Sections 1.7 and 2 of this specification are normative~~ and can contain the terms MAY, SHOULD, MUST, MUST NOT, and SHOULD NOT as defined in RFC 2119.~~. All other sections and examples in this specification are informative.

1.1 ~~1.1~~ Glossary

~~The~~This document uses the following terms~~ are defined in [MS-GLOS]:~~

~~universally unique identifier (UUID)~~

~~The following terms are specific to this document~~:

connection manager: A component that is referenced by an SSIS package. A connection manager stores the information necessary to establish connections to external resources and establishes and provides these connections, on demand, to other components within the SSIS package.

project: A collection of ~~**SSIS**~~Integration Services (IS) packages that are developed and deployed as a unit.

project parameter: A named entity that is defined at the scope of the project, with an associated data type that allows a value to be passed in to the execution of packages within that project.

project parameter file: The file that contains the definitions of the parameters that are defined at the scope of the project.

SQL Server Integration Services (SSIS) package: A module of a project. The module contains control flow and data flow, as specified in [MS-DTSX] section 1.3.

universally unique identifier (UUID): A 128-bit value. UUIDs can be used for multiple purposes, from tagging objects with an extremely short lifetime, to reliably identifying very persistent objects in cross-process communication such as client and server interfaces, manager entry-point vectors, and RPC objects. UUIDs are highly likely to be unique. UUIDs are also known as globally unique identifiers (GUIDs) and these terms are used interchangeably in the Microsoft protocol technical documents (TDs). Interchanging the usage of these terms does not imply or require a specific algorithm or mechanism to generate the UUID. Specifically, the use of this term does not imply or require that the algorithms described in [RFC4122] or [C706] must be used for generating the UUID.

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

~~1.2~~ References

1.2 ~~References~~

Links to a document in the Microsoft Open Specifications ~~documentation do not include a publishing year because links are to the latest~~library point to the correct section in the most recently published version of the ~~documents, which are updated frequently. References to other~~referenced document. However, because individual documents ~~include a publishing year when one is available.~~in the library are not updated at the same time, the section numbers in the documents may not match. You can confirm the correct section numbering by checking the Errata.

~~1.1.1~~1.2.1 ~~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.

[MS-DTSX] Microsoft Corporation, "Data Transformation Services Package XML File Format".

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

~~1.1.2~~1.2.2 ~~1.2.2~~ Informative References

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

~~1.3 Overview~~

None.

1.3 Overview

The file format for the Microsoft SQL Server Integration Services (SSIS) project parameter file is a file type that is used to store the metadata for the **project parameters** of an SSIS project.

The project parameter file is an XML document.

~~1.2~~1.4 ~~1.4~~ Relationship to Protocols and Other Structures

The project parameter file format can be used as a payload in protocols that support the transport of binary data.

~~1.3~~1.5 ~~1.5~~ Applicability Statement

The project parameter file format is applicable for use in an SSIS project.

~~1.4~~1.6 ~~1.6~~ Versioning and Localization

This document describes version 1.0 of the project parameter file format. There are no localization-dependent structures in the project parameter file format.

~~1.5~~1.7 ~~1.7~~ Vendor-Extensible Fields

Extensions to the file format that is specified in this document are not allowed. Tools that process this format do not have to preserve unrecognized structures when loading or persisting this file format.

2 ~~2~~ Structures

2.1 ~~2.1~~ Project Parameter File

A project parameter file is an XML file. A project parameter file MUST be named Project.params.

2.1.1 ~~2.1.1~~ XML Namespace

The project parameter file contains an XML structure. The namespace URI for the project parameter XML structure is www.microsoft.com/SqlServer/SSIS.

2.1.2 ~~2.1.2~~ ParametersType

The **ParametersType** complex type is the container type for a collection of elements of type ParameterType.

The following is the XSD of the **ParametersType** complex type.

```
<xs:complexType name="ParametersType">
  <xs:sequence>
    <xs:element name="Parameter" maxOccurs="unbounded" type="SSIS:ParameterType" />
  </xs:sequence>
</xs:complexType>
```

The following table provides additional information about the elements, types, and constraints for the **ParametersType** complex type.

Element	Type definition	Description
Parameter	ParameterType	Specifies a **Parameter** element, which contains the options, parameters, and settings for a parameter.

2.1.3 ~~2.1.3~~ ParameterType

The **ParameterType** complex type is used to specify a parameter.

The following is the XSD of the **ParameterType** complex type.

```
  <xs:complexType name="ParameterType">
    <xs:sequence>
      <xs:element name="Properties" minOccurs="1" maxOccurs="1"
type="SSIS:ParmameterPropertiesType">
      </xs:element>
    </xs:sequence>
    <xs:attribute ref="SSIS:Name" use="required" />
  </xs:complexType>
```

The **Name** attribute specifies the name of the parameter.

2.1.4 ~~2.1.4~~ ParameterPropertiesType

The **ParameterPropertiesType** complex type is the container type for a collection of elements of type ParameterPropertyType.

The following is the XSD of the **ParameterPropertiesType** complex type.

```
      <xs:complexType name="ParameterPropertiesType">
        <xs:sequence>
          <xs:element name="Property" minOccurs="8" maxOccurs="8" 
    type="SSIS:ParameterPropertyType" />
        </xs:sequence>
      </xs:complexType>
```

2.1.5 ~~2.1.5~~ ParameterPropertyType

The **ParameterPropertyType** complex type represents a property of a parameter.

The following is the XSD of the **ParameterPropertyType** complex type.

```
      <xs:complexType name="ParameterPropertyType">
        <xs:simpleContent>
          <xs:extension base="xs:string">
            <xs:attribute ref="SSIS:Name" use="required" />
            <xs:attribute ref="SSIS:Sensitive" use="optional" />
          </xs:extension>
        </xs:simpleContent>
      </xs:complexType>
```

The value of the **Name** attribute MUST be one of the values from the following table, with the associated implication for that property. Moreover, a property element MUST exist for the parameter for each of the property names in this table, and there cannot be more than one property with the same property name for a given parameter.

Property name	Implication
ID	The element contains a **universally unique identifier (UUID)** for the parameter, in the following form: {hhhhhhhh-hhhh-hhhh-hhhh-hhhhhhhhhhhh}
CreationName	The element contains the creation name of the parameter. This value is ignored.
Description	The element value text that is the description of the parameter.
IncludeInDebugDump	The element contains an integer value that specifies whether the parameter is included in a binary dump. The integer value MUST be 0 or 1. An integer value of 0 specifies that the parameter value is not included in a dump; an integer value of 1 specifies that the parameter value is included in a dump.
Required	The element contains an integer value that specifies whether the parameter is a required parameter. The integer value MUST be 0 or 1. An integer value of 0 specifies that the parameter is not a required parameter; an integer value of 1 specifies that the parameter is a required parameter.
Sensitive	The element contains an integer value that specifies whether the parameter is a sensitive parameter. The integer value MUST be 0 or 1. An integer value of 0 specifies that the parameter is not a sensitive parameter; an integer value of 1 specifies that the parameter is a sensitive parameter.

Property name	Implication
	If the value of this property element is 1, the value of the sibling property that has the name **Value** MUST represent an encrypted value.
Value	The property contains the value of the parameter.
DataType	Specifies that the property contains the data type of the parameter. The value of the attribute is an integer that represents the data type of the property. The value of the property that has the name **Value** of this parameter MUST conform to this data type. See the following table for the mapping of integers to data types.

The following table provides the integer value for each data type.

Data type	Integer value
Boolean	3
Byte	6
DateTime	16
Decimal	15
Double	14
Int16	7
Int32	9
Int64	11
SByte	5
Single	13
String	18
UInt32	10
UInt64	12

3 ~~3~~Structure Examples

The following is an example of a typical project parameter file, including all the mandatory elements and examples of property and parameter declarations.

```
<?xml version="1.0"?>
<SSIS:Parameters xmlns:SSIS="www.microsoft.com/SqlServer/SSIS">
  <SSIS:Parameter
    SSIS:Name="projparam1">
    <SSIS:Properties>
      <SSIS:Property
        SSIS:Name="ID">{f12e6b1b-4b15-4f3d-a02c-8ba9175af385}</SSIS:Property>
      <SSIS:Property
        SSIS:Name="CreationName"></SSIS:Property>
      <SSIS:Property
        SSIS:Name="Description">asdfadsf</SSIS:Property>
      <SSIS:Property
        SSIS:Name="IncludeInDebugDump">0</SSIS:Property>
      <SSIS:Property
        SSIS:Name="Required">0</SSIS:Property>
      <SSIS:Property
        SSIS:Name="Sensitive">1</SSIS:Property>
      <SSIS:Property
        SSIS:Name="Value"

SSIS:Sensitive="1">AQAAANCMnd8BFdERjHoAwE/Cl+sBAAAACpzx4TP7X067dKtrwnEH9AAAAAACAAAAAAADZgAAwA
AAABAAAADvs3E7hbK9uRLfNB6d77W4AAAAAASAAACgAAAAEAAAAMXImEjaQIIoURKGcCvkF9N4AAAAgLqX+CxeW3P4IzV
q6O6clh+JfS0C62ninTMfu0I/uLWgnYvdGdLp/h/5LfqLiNcPB4M9lFf4V92Odzgejp0JeaXTkWQtYp/dw/2SbAs70Bxr
fdoLcN90O0/Btr5OBr/1/VEhcjKKcd+bhoyPIUdNel+qoE4HaloVFAAAAPv3AM7RcowuTr3NzlNJ9KUY4Pj6</SSIS:Pr
operty>
      <SSIS:Property
        SSIS:Name="DataType">9</SSIS:Property>
    </SSIS:Properties>
  </SSIS:Parameter>
  <SSIS:Parameter
    SSIS:Name="projparam2">
    <SSIS:Properties>
      <SSIS:Property
        SSIS:Name="ID">{498bf8a2-4533-4517-ae79-b65f92b84303}</SSIS:Property>
      <SSIS:Property
        SSIS:Name="CreationName"></SSIS:Property>
      <SSIS:Property
        SSIS:Name="Description">asdfasdfasdf</SSIS:Property>
      <SSIS:Property
        SSIS:Name="IncludeInDebugDump">0</SSIS:Property>
      <SSIS:Property
        SSIS:Name="Required">1</SSIS:Property>
      <SSIS:Property
        SSIS:Name="Sensitive">0</SSIS:Property>
      <SSIS:Property
        SSIS:Name="Value">0</SSIS:Property>
      <SSIS:Property
        SSIS:Name="DataType">9</SSIS:Property>
    </SSIS:Properties>
  </SSIS:Parameter>
</SSIS:Parameters>
```

4 ~~4~~ Security

4.1 ~~4.1~~ Security Considerations for Implementers

The project parameter file can contain sensitive information, such as user names and passwords that are used to access data sources.

When sensitive values are present in a project or its **SQL Server Integration Services (SSIS) packages**, the user should use the appropriate protection level for serialization, as described in this document and in [MS-DTSX] section 4.1.

4.2 ~~4.2~~ Index of Security Fields

None.

5 ~~5~~ Appendix A: XML Schema Definition

The following is the complete XML schema definition for the project parameter file format.

```
<?xml version="1.0" encoding="Windows-1252"?>
<xs:schema
    xmlns:SSIS="www.microsoft.com/SqlServer/SSIS"
    attributeFormDefault="unqualified"
    elementFormDefault="qualified"
    targetNamespace="www.microsoft.com/SqlServer/SSIS"
    xmlns:xs="http://www.w3.org/2001/XMLSchema">
  <xs:element name="Parameters" type="SSIS:ParametersType">
  </xs:element>
  <xs:complexType name="ParametersType">
    <xs:sequence>
      <xs:element name="Parameter" maxOccurs="unbounded" type="SSIS:ParameterType" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ParameterType">
    <xs:sequence>
      <xs:element name="Properties" minOccurs="1" maxOccurs="1"
type="SSIS:ParameterPropertiesType">
      </xs:element>
    </xs:sequence>
    <xs:attribute ref="SSIS:Name" use="required" />
  </xs:complexType>
  <xs:complexType name="ParameterPropertiesType">
    <xs:sequence>
      <xs:element name="Property" minOccurs="8" maxOccurs="8"
type="SSIS:ParameterPropertyType" />
    </xs:sequence>
  </xs:complexType>
  <xs:complexType name="ParameterPropertyType">
    <xs:simpleContent>
      <xs:extension base="xs:string">
        <xs:attribute ref="SSIS:Name" use="required" />
        <xs:attribute ref="SSIS:Sensitive" use="optional" />
      </xs:extension>
    </xs:simpleContent>
  </xs:complexType>
  <xs:attribute name="Name" type="xs:string" />
  <xs:attribute name="Sensitive" type="xs:unsignedByte" />
</xs:schema>
```

6 ~~6~~ Appendix B: Product Behavior

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

- ~~▪~~ Microsoft SQL Server 2012
- ~~▪~~ Microsoft SQL Server 2014
- Microsoft SQL Server 2016

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.

7 ~~7~~ Change Tracking

~~No table of~~ This section identifies changes ~~is available. The~~that were made to this document ~~is either new or has had no changes~~ since ~~its~~the last release. Changes are classified as New, Major, Minor, Editorial, or No change.

~~8~~ The revision class **New** means that a new document is being released.

The revision class **Major** means that the technical content in the document was significantly revised. Major changes affect protocol interoperability or implementation. Examples of major changes are:

- A document revision that incorporates changes to interoperability requirements or functionality.
- The removal of a document from the documentation set.

The revision class **Minor** means that the meaning of the technical content was clarified. Minor changes do not affect protocol interoperability or implementation. Examples of minor changes are updates to clarify ambiguity at the sentence, paragraph, or table level.

The revision class **Editorial** means that the formatting in the technical content was changed. Editorial changes apply to grammatical, formatting, and style issues.

The revision class **No change** means that no new technical changes were introduced. Minor editorial and formatting changes may have been made, but the technical content of the document is identical to the last released version.

Major and minor changes can be described further using the following change types:

- New content added.
- Content updated.
- Content removed.
- New product behavior note added.
- Product behavior note updated.
- Product behavior note removed.
- New protocol syntax added.
- Protocol syntax updated.
- Protocol syntax removed.
- New content added due to protocol revision.
- Content updated due to protocol revision.
- Content removed due to protocol revision.
- New protocol syntax added due to protocol revision.
- Protocol syntax updated due to protocol revision.
- Protocol syntax removed due to protocol revision.
- Obsolete document removed.

Editorial changes are always classified with the change type **Editorially updated**.

Some important terms used in the change type descriptions are defined as follows:

- **Protocol syntax** refers to data elements (such as packets, structures, enumerations, and methods) as well as interfaces.
- **Protocol revision** refers to changes made to a protocol that affect the bits that are sent over the wire.

The changes made to this document are listed in the following table. For more information, please contact dochelp@microsoft.com.

Section	**Tracking number (if applicable) and description**	**Major change (Y or N)**	**Change type**
6 Appendix B: Product Behavior	Added SQL Server 2016 to the list of applicable products.	Y	Content update.

8 Index

A

Applicability 5
~~5~~
C

Change tracking 13
~~13~~
E

Examples 9
Extensions 5
~~5~~
F

Fields - security index 10
Fields - vendor-extensible 5
File format overview 5
~~5~~
G

Glossary 4
~~4~~
I

Implementer - security considerations 10
Index of security fields 10
Informative references 5~~5~~
Introduction 4

L

Localization 5~~5~~

N

Namespace URI 6~~6~~
Normative references 5~~4~~

O

Other protocols and structures
relationship to 5~~5~~
Overview (synopsis) 5

P

ParameterPropertiesType complex type 7~~7~~
ParameterPropertyType complex type 7~~7~~
ParametersType complex type 6~~6~~
ParameterType complex type 6~~6~~
Product behavior 12
Project parameter file overview 6
~~6~~
R

References 4
informative 5~~5~~
normative 5~~4~~
Relationship to other protocols and structures 5~~5~~

Relationship to protocols and other structures 5

S

Security
  field index 10
  implementer considerations 10~~10~~
Security considerations 10

T

Tracking changes 13
~~13~~

V

Vendor-extensible fields 5~~5~~
Versioning 5
~~5~~

X

XML namespace 6

~~6~~