

Workshare Compare Server 9.5.1

Developer Guide

Table of Contents

Chapter 1: Introduction	3
Introducing Workshare Compare Server	4
Communicating with Workshare Compare Server	4
Chapter 2: Documentation and Examples	5
Overview	6
Get Request Example	6
What this example does	6
Example CURL command	6
Explanation of the CURL command	6
Explanation of request parameters	7
Post Request Example	9
What this command does	9
Example CURL command	9
Explanation of the CURL command	10
Explanation of request parameters	10
Appendix A. Change Summary Information	12
RedlineML	13
RedlineML Schema	13
Character set values	20
ChangeT values	21

Chapter 1: Introduction

This chapter describes the functionality provided by Workshare Compare Server and how to communicate with Workshare Compare Server.

Note: *In this document, the terms Workshare Compare Server and Workshare Compare service are interchangeable. Workshare Compare Server is the name of the product but where you find references to Workshare Compare service, it is in order to be technically accurate.*

Introducing Workshare Compare Server

Workshare Compare Server is a web service that provides extremely fast and robust document comparison and returns a range of outputs including a comprehensive comparison document (RTF, DOC, DOCX or PDF) traditionally known as a 'Redline', an XML change summary and a Workshare Professional compatible WDF document. Using the Workshare Compare service, you can write applications that will:

- Provide extremely fast and robust document comparison, including change identification and extraction.
- Verify and highlight all changes and differences between drafts and versions, no matter how complex the document.
- Validate that all changes closely adhere to policies and procedures, and an approved boilerplate.

The sample applications demonstrate the use of the Workshare Compare web service to produce and display comparison outputs, given two input documents.

Communicating with Workshare Compare Server

You can communicate with Compare Server over HTTP REST calls. To fully benefit from this guide, you should have an understanding of RESTful APIs.

Chapter 2: Documentation and Examples

This chapter provides the link to the documentation for the HTTP API endpoints and provides examples of GET and POST requests.

Overview

The HTTP API endpoints are documented here: [http://\[your server name\]/swagger](http://[your server name]/swagger)

There are two requests available:

- **GET**: Perform a comparison by fetching two documents (specified by their URLs)
- **POST**: Perform a comparison by POSTing the contents of exactly two documents as a multipart/form-data

An example of each is below.

Get Request Example

What this example does

In this example, the request compares two documents from source URLs and applies the specified rendering options to the resulting redline. The redline is then saved to the current directory as an RTF called **GetTest.rtf**.

Example CURL command

```
curl -X GET --header 'Accept: application/rtf' 'http://[your server name]/api/Compare?originalSourceUrl=http%3A%2F%2Finstall.workshare.com%2Fcompare%2FSampleOriginal.doc&modifiedSourceUrl=http%3A%2F%2Finstall.workshare.com%2Fcompare%2FSampleModified.doc&outputFormat=Rtf&renderingOptions=DETECT%20LIST%20NUMBERING%20CHANGES%3D1%3BCOMPARE%20HEADERS%2FFOOTERS%3D0' -o GetTest.rtf
```

Explanation of the CURL command

Snippet	Function
<code>curl</code>	Beginning the request.
<code>-X GET</code>	Type of request.
<code>--header 'Accept: application/rtf'</code>	This part of the request is not required but if used, should match the output format application. These can be found on the Swagger page.

Snippet	Function
<code>-o GetTest.rtf</code>	Tells the request to save the response as the named file in the current directory. The format should request the specified output format. For combined requests in output, this should be .zip.

Explanation of request parameters

In the example CURL command above, the request URL is as follows:

Root

`http://[your server name]/WorkshareCompareApi/api/compare`

The root will vary according to what you selected during the installation.

Rest of query:

`originalSourceUrl=http%3A%2F%2Finstall.workshare.com%2Fcompare%2FSampleOriginal.doc&modifiedSourceUrl=http%3A%2F%2Finstall.workshare.com%2Fcompare%2FSampleModified.doc&outputFormat=Rtf&renderingOptions=DETECT%20LIST%20NUMBERING%20CHANGES%3D1%3BCOMPARE%20HEADERS%2FFOOTERS%3D0'`

An explanation of each element of the request URL is in the table below.

Name	Example value	Comments
Server name	<code>http://[your server name]</code>	
Virtual Directory Path	<code>WorkshareCompareApi</code>	If you install the Workshare Compare API in the root of a new web site then this component will not be needed in the URL
Endpoint	<code>/api/Compare</code>	

Name	Example value	Comments
<code>originalSourceUrl</code>	<code>http://install.workshare.com/compare/SampleOriginal.doc</code>	The URL from which to fetch the original document. <code>http://</code> and <code>https://</code> are supported. Note 1: The files need to be available for HTTP/S requests. Note 2: This parameter must be URL Encoded before it's included in your request URL. You can convert between decoded and encoded URLs at this site: http://meyerweb.com/eric/tools/dencoder/
<code>modifiedSourceUrl</code>	<code>http://install.workshare.com/compare/SampleModified.doc</code>	The URL from which to fetch the original document. <code>http://</code> and <code>https://</code> are supported. Note: Follow the same guidance as indicated in the notes for originalSourceUrl.
<code>outputFormat</code>	One of: • Rtf (default) • Wdf • Doc • DocX • Pdf • RedlineMI • RedlineMIAndRtf • RedlineMIAndDoc • RedlineMIAndDocX • RedlineMIAndPdf	The format of the returned comparison. When more than one format is specified (ie RedlineMIAndXXX) the result is returned as a ZIP file containing both formats. The default output format is RTF if this option is not specified. WDF is the comparison format used by the Workshare Compare desktop application (comparisons created in WDF format can only be opened in the Workshare Compare desktop application). RedlineMI is an XML-based format that describes the changes between the two documents in an easy-to-use manner (see Appendix A: Change Summary Information).

Name	Example value	Comments
renderingOptions	DETECT LIST NUMBERING CHANGES=1;COMPARE HEADERS/FOOTERS=0'	Optional rendering options to customize the comparison process and the format of the comparison document. Rendering options are best generated by creating a rendering set using the Workshare Compare desktop application. The contents of the saved .set file can be passed as the value for this parameter. Semicolons can be used in place of \r or \n as line separators in this parameter. Note: For more information about rendering sets, see the <i>Compare Server Rendering Set Guide</i>

Post Request Example

What this command does

In this example, the request compares two documents from the local machine and applies the specified rendering options to the resulting redline. The redline is then saved to the current directory as a DOC called **PostTest.docx**.

Example CURL command

```
curl -X POST --header 'Content-Type: multipart/form-data' --header
'Accept: application/rtf' -F outputFormat=Docx -F
"file1=@documents/Comp2.docx" -F "file2=@documents/Comp1.docx" -F
'renderingOptions=Display Workshare Compare Footers=1; Inserted
Text Color=8388736 ' 'http://[your server name]/api/Compare' -o
PostTest.docx
```

Explanation of the CURL command

Snippet	Function
<code>curl</code>	Beginning the request.
<code>-X POST</code>	Type of request.
<code>--header 'Content-Type: multipart/form-data'</code>	Format of the request. Note: Must always be included for POST requests.
<code>--header 'Accept: application/rtf'</code>	This part of the request is not required but if used, should match the output format application. These can be found on the Swagger page.
<code>-o PostTest.docx</code>	Tells the request to save the response as the named file in the current directory. The format should request the specified output format. For combined requests in output, this should be .zip.

Explanation of request parameters

In the example CURL command above, the request URL is as follows:

```
outputFormat=Docx -F "file1=@documents/Comp2.docx" -F
"file2=@documents/Comp1.docx" -F 'renderingOptions=Display
Workshare Compare Footers=1; Inserted Text Color=8388736 '
'http://[your server name]/api/Compare
```

An explanation of each element of the request URL is in the table below.

Name	Example value	Comments
Server name	http://[your server name]	
Endpoint	/api/Compare	
originalDocument	file1=@documents/Comp2.docx	The location of the original document on the local system.
modifiedDocument	file2=@documents/Comp1.docx	The location of the modified document on the local system.

Name	Example value	Comments
<code>outputFormat</code>	One of: • Rtf (default) • Wdf • Doc • DocX • Pdf • RedlineMI • RedlineMIAndRtf • RedlineMIAndDoc • RedlineMIAndDocX • RedlineMIAndPdf	The format of the returned comparison. When more than one format is specified (ie RedlineMIAndXXX), the result is returned as a .zip file containing both formats. The default output format is RTF if this option is not specified. WDF is the comparison format used by the Workshare Compare desktop application (comparisons created in WDF format can only be opened in the Workshare Compare desktop applicaion). RedlineMI is an XML based format that describes the changes between the two documents in an easy-to-use manner (see Appendix A: Change Summary Information).
<code>renderingOptions</code>	Display Workshare Compare Footers=1; Inserted Text Color=8388736	The optional rendering options to customize the comparison process and format the comparison document. Rendering options are best generated by creating a rendering set using the Workshare Compare desktop application. The contents of the saved .set file can be passed as the value for this parameter. Semicolons can be used in place of \r or \n as line separators in this parameter. Note: For more information about rendering sets, see the Compare Server Rendering Set Guide.

Appendix A. Change Summary Information

This appendix describes the schema for the Change Summary produced with a comparison and the character set values.

RedlineML

RedlineML is now the preferred format for extracting change summary information. It contains the entire content of the Redline document in an easy-to-work-with XML format.

RedlineML Schema

```
<?xml version="1.0" encoding="utf-8"?>
<xs:schema targetNamespace="http://workshare.com/2010/RedlineML"
  elementFormDefault="qualified"
  xmlns="http://workshare.com/2010/RedlineML"
  xmlns:xs="http://www.w3.org/2001/XMLSchema"
>

  <xs:element name="document" type="documentT"/>

  <xs:simpleType name="cellStatusT" final="restriction" >
    <xs:restriction base="xs:string">
      <xs:enumeration value="normal" />
      <xs:enumeration value="inserted" />
      <xs:enumeration value="deleted" />
      <xs:enumeration value="moveSource" />
      <xs:enumeration value="moveDestinate"/>
      <xs:enumeration value="dead"/>
      <xs:enumeration value="padding"/>
    </xs:restriction>
  </xs:simpleType>

  <xs:group name="content">
    <xs:sequence>
      <xs:choice minOccurs="0" maxOccurs="unbounded">
        <xs:element ref="paraMarker"/>
        <xs:element ref="change"/>
        <xs:element ref="field"/>
      </xs:choice>
    </xs:sequence>
  </xs:group>
</xs:schema>
```

```
<xs:element ref="bkmk"/>
<xs:element name="table" type="tableT"/>
<xs:element name="shape" type="shapeT"/>
<xs:element name="blob" type="blobT"/>
<xs:element name="pict" type="blobT"/>
<xs:element name="run" type="runT"/>
<xs:element ref="endNote"/>
<xs:element ref="footNote"/>
<xs:element ref="textbox"/>
<xs:element ref="comment"/>
</xs:choice>
</xs:sequence>
</xs:group>

<xs:attributeGroup name="insertedDeletedAttrs">
  <xs:attribute name="isInserted" type="xs:boolean"
use="optional" default="false"/>
  <xs:attribute name="isDeleted" type="xs:boolean"
use="optional" default="false"/>
</xs:attributeGroup>

<xs:group name="section">
  <xs:sequence>
    <xs:element type="sectionMarkerT" name="sectionMarker"/>
    <xs:group ref="content"/>
  </xs:sequence>
</xs:group>

<xs:group name="changeContent">
  <xs:sequence>
    <xs:choice minOccurs="0" maxOccurs="unbounded">
      <xs:element ref="paraMarker"/>
      <xs:element ref="field"/>
      <xs:element ref="bkmk"/>
    </xs:choice>
  </xs:sequence>
</xs:group>
```

```
<xs:element name="shape" type="shapeT"/>
<xs:element name="blob" type="blobT"/>
<xs:element name="pict" type="blobT"/>
<xs:element name="run" type="runT"/>
<xs:element ref="endNote"/>
<xs:element ref="footNote"/>
<xs:element ref="textbox"/>
<xs:element ref="comment"/>
</xs:choice>
</xs:sequence>
</xs:group>

<xs:complexType name="shapeT">
  <xs:group ref="content"/>
  <xs:attributeGroup ref="insertedDeletedAttrs"/>
</xs:complexType>

<xs:complexType name="blobT">
  <xs:attributeGroup ref="insertedDeletedAttrs"/>
</xs:complexType>

<xs:element name="change" type="changeT"/>
<xs:complexType name="changeT" >
  <xs:group ref="changeContent"/>
  <xs:attribute name="number" type="xs:integer" use="required"/>
  <xs:attribute name="type" type="xs:integer" use="required"/>
  <xs:attribute name="crossref" type="xs:integer"
use="optional"/>
</xs:complexType>

<xs:complexType name="documentT">
  <xs:sequence>
    <xs:group ref="content"/>
```

```
<!--sadly the compositor can spit out content before its
first section marker-->
<xs:sequence minOccurs="1" maxOccurs="unbounded">
  <xs:group ref="section"/>
</xs:sequence>
</xs:sequence>
<xs:anyAttribute/>
</xs:complexType>

<xs:complexType name="sectionMarkerT">
  <xs:sequence>
    <xs:sequence minOccurs="0" maxOccurs="unbounded">
      <xs:choice>
        <xs:element name="header" type="headerfooterT"/>
        <xs:element name="footer" type="headerfooterT"/>
      </xs:choice>
    </xs:sequence>
  </xs:sequence>
  <xs:attributeGroup ref="insertedDeletedAttrs"/>
</xs:complexType>

<xs:complexType name="tableT">
  <xs:sequence minOccurs="1" maxOccurs="unbounded">
    <xs:element name="row" type="rowT"/>
  </xs:sequence>
  <xs:attributeGroup ref="insertedDeletedAttrs"/>
</xs:complexType>

<xs:complexType name="rowT">
  <xs:sequence minOccurs="1" maxOccurs="unbounded">
    <xs:element name="cell" type="cellT"/>
  </xs:sequence>
  <xs:attributeGroup ref="insertedDeletedAttrs"/>
</xs:complexType>
```



```
<xs:complexType name="cellT">
  <xs:group ref="content"/>
  <xs:attribute name="cellStatus" type="cellStatusT"
use="optional" default="normal"/>
  <xs:attribute name="isInsertedColumn" type="xs:boolean"
use="optional" default="false"/>
  <xs:attribute name="isDeletedColumn" type="xs:boolean"
use="optional" default="false"/>
  <xs:attribute name="column" type="xs:integer" use="optional"/>
  <xs:attribute name="spanInfoOriginal" type="xs:string"
use="optional"/>
  <xs:attribute name="spanInfoModified" type="xs:string"
use="optional"/>
  <xs:attribute name="spannedInOriginal" type="xs:boolean"
use="optional"/>
  <xs:attribute name="spannedInModified" type="xs:boolean"
use="optional"/>
  <xs:attribute name="isMerged" type="xs:boolean"
use="optional"/>
</xs:complexType>

<xs:element name="textbox" type="textboxT"/>
<xs:complexType name="textboxT">
  <xs:group ref="content"/>
</xs:complexType>

<xs:element name="paraMarker" type="paraMarkerT"/>
<xs:complexType name="paraMarkerT">
  <xs:attribute name="listNumber" type="xs:string"
use="optional"/>
  <xs:attributeGroup ref="insertedDeletedAttrs"/>
  <xs:attribute name="isInsertedListItem" type="xs:boolean"
use="optional" default="false"/>
  <xs:attribute name="listLevel" type="xs:int" use="optional"/>
  <xs:attribute name="listId" type="xs:int" use="optional"/>
</xs:complexType>
```

```
<xs:complexType name="runT" mixed="true">
  <xs:attribute name="font" type="xs:string" use="required"/>
  <xs:attribute name="wasListNum" type="xs:boolean"
default="false" use="optional"/>
  <xs:attribute name="wasField" type="xs:boolean"
default="false" use="optional"/>
  <xs:attribute name="rtl" type="xs:boolean" default="false"
use="optional"/>
</xs:complexType>

<xs:complexType name="commentT">
  <xs:group ref="content"/>
  <xs:attributeGroup ref="insertedDeletedAttrs"/>
</xs:complexType>

<xs:element name="comment" type="commentT"/>

<xs:element name="footNote" type="footEndNoteT" />
<xs:element name="endNote" type="footEndNoteT" />
<xs:complexType name="footEndNoteT">
  <xs:group ref="content"/>
  <xs:attributeGroup ref="insertedDeletedAttrs"/>
</xs:complexType>

<xs:complexType name="headerfooterT">
  <xs:group ref="content"/>
  <xs:attribute name="type" type="hdrftrType" use="required"/>
  <xs:attributeGroup ref="insertedDeletedAttrs"/>
</xs:complexType>

<xs:simpleType name="hdrftrType">
  <xs:restriction base="xs:string">
    <xs:enumeration value="left"/>
```

```
<xs:enumeration value="right"/>
<xs:enumeration value="first"/>
<xs:enumeration value="main"/>
</xs:restriction>
</xs:simpleType>

<xs:element name="field" type="fieldT"/>
<xs:complexType name="fieldT">
  <xs:sequence>
    <xs:element type="fieldCodeT" name="fieldCode"/>
    <xs:element ref="fieldResult" minOccurs="0"/>
  </xs:sequence>
  <xs:attributeGroup ref="insertedDeletedAttrs"/>
</xs:complexType>

<xs:complexType name="fieldCodeT">
  <xs:sequence maxOccurs="unbounded" minOccurs="1">
    <xs:choice>
      <xs:element ref="field"/>
      <xs:element name="run" type="runT"/>
      <xs:element ref="paraMarker"/>
      <xs:element name="blob" type="blobT"/>
    </xs:choice>
  </xs:sequence>
</xs:complexType>

<xs:element name="fieldResult" type="fieldResultT"/>
<xs:complexType name="fieldResultT">
  <xs:group ref="content"/>
</xs:complexType>

<xs:element name="bkmk" type="bkmkT"/>
<xs:complexType name="bkmkT">
  <xs:attribute name="name" type="xs:string" use="required"/>
```

```
<xs:attribute name="start" type="xs:boolean" use="required"/>
</xs:complexType>

</xs:schema>
```

Character set values

ANSI_CHARSET	0
DEFAULT_CHARSET	1
SYMBOL_CHARSET	2
SHIFTJIS_CHARSET	128
HANGEUL_CHARSET	129
HANGUL_CHARSET	129
GB2312_CHARSET	134
CHINESEBIG5_CHARSET	136
OEM_CHARSET	255
JOHAB_CHARSET	130
HEBREW_CHARSET	177
ARABIC_CHARSET	178
GREEK_CHARSET	161
TURKISH_CHARSET	162
VIETNAMESE_CHARSET	163
THAI_CHARSET	222
EASTEUROPE_CHARSET	238
RUSSIAN_CHARSET	204
MAC_CHARSET	77
BALTIC_CHARSET	186

ChangeT values

0	DELETION
1	MOVESOURCE
2	MOVEDESTINATION
3	INSERTION
4	FORMAT_CHANGE
13	MOVEDDELETION
14	STYLECHANGE_TEXT
15	STYLECHANGE_LABEL

 Workshare Ltd.

© 2018. Workshare Ltd. All rights reserved.

Copyright

Workshare Professional and Workshare DeltaView are registered trademarks of Workshare Ltd. Workshare Compare, Workshare Protect, Workshare 3, Workshare DeltaServer, SafetyGain, and the Workshare logo are trademarks of Workshare Ltd. All other trademarks are those of their respective holders.

Trademarked names may appear throughout this guide. Instead of listing these here or inserting numerous trademark symbols, Workshare wishes to state categorically that no infringement of intellectual or other copyright is intended and that trademarks are used only for editorial purposes.

Disclaimer

The authors/publishers of this guide and any associated help material have used their best efforts to ensure accuracy and effectiveness. Due to the continuing nature of software development, it may be necessary to distribute updated help from time to time. The authors would like to assure users of their continued best efforts in supplying the most effective help material possible.

The authors/publishers, however, make no warranty of any kind, expressed or implied, with regard to Workshare programs or help material associated with them, including this guide. The authors/publishers shall not be liable in the event of incidental or consequential damages in connection with, or arising out of, the programs or associated help instructions.

Revisions

Published for Workshare Compare Server 9.5: 14/9/18

Revised for Workshare Compare Server 9.5.1: 8/1/18

Workshare Ltd., 20 Fashion Street, London E1 6PX www.workshare.com