A Developer Diary

{about:"code learn and share"}

June 9, 2015 By Abhisek Jana — 11 Comments (Edit)

Compare 2 XML in JAVA – Part 1

There are many ways to compare 2 XML in Java, however we will see here one of the process that I follow. I have split this in two seperate parts. The Part1 is about how to use the APIs in order to compare the XML and Part2 would explain how to utilize that and create a SWT/Eclipse RCP Application to visually represent the differences.

I am using XMLUnit to compare two XML. I found that it's one of the best usable API that can be pluged into the code easily. Lets see how this API works.

Download XMLUnit:

You can go to http://www.xmlunit.org/ and download the latest JAR for Java. I am using the 1.6 version since at this the version 2.0 is still in development. I got it from maven repository, here is the
link http://mvnrepository.com/artifact/xmlunit/xmlunit

Configuration:

You can configure the initial details using the static methods. XMLUnit provies few functions for setting the Ignore XML Attribute, Comments, whitespace etc. For our use we will set all the following options to true.

```
XMLUnit.setIgnoreAttributeOrder(true);
XMLUnit.setIgnoreComments(true);
XMLUnit.setIgnoreDiffBetweenTextAndCDATA(true);
XMLUnit.setIgnoreWhitespace(true);
XMLUnit.setNormalizeWhitespace(true);
```

Code:

Here is the remaining code base :

```
String strGoldVersion=FileUtils.readFileToString(new
File("GoldCopy.xml"));
String strTestVersion=FileUtils.readFileToString(new
File("TestCopy.xml"));

Diff myDiff=null;

DifferenceListener myDifferenceListener = new
IgnoreTextAndAttributeValuesDifferenceListener();

try {
        myDiff = new Diff(strGoldCopy,strTestCopy);

myDiff.overrideDifferenceListener(myDifferenceListener);
} catch (SAXException | IOException e) {
        e.printStackTrace();
}

DetailedDiff myDiffd = new DetailedDiff(myDiff);
List allDifferences = myDiffd.getAllDifferences();

for(Difference diff:allDifferences){
        System.out.print(diff.getDescription()+" ");
```

```
        System.out.print(diff.getId()+" ");

System.out.print(diff.getControlNodeDetail().getXpathLocation()+"
");

System.out.print(diff.getTestNodeDetail().getXpathLocation()+"
");
        System.out.println(" ");
}
```

Test:

You can read through the code, its easy to understand. Now, to test this I have created 2 XML files.

```
Reminder
Tove
Jane
Dinner!
```

The following TestCopy.xml would be compared against the GoldCopy.xml.

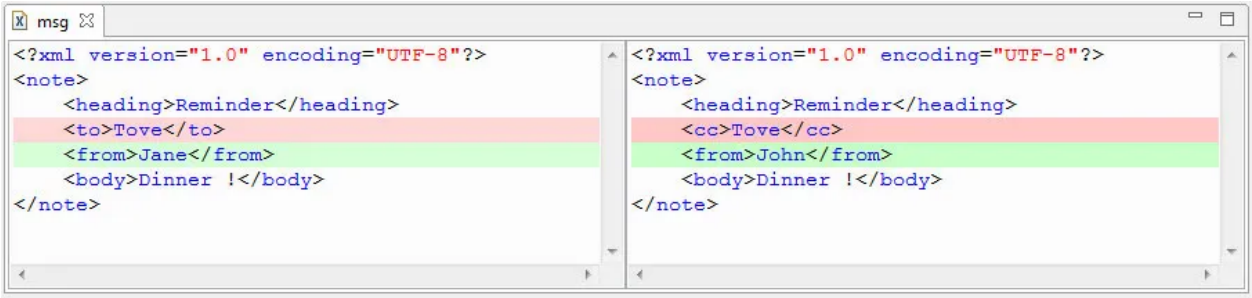
```
Reminder
Tove
John
Dinner!
```

I ran the program and here is the result.

```
element tag name 10 /note[1]/to[1] /note[1]/cc[1]
text value 14 /note[1]/from[1]/text()[1]
```

```
/note[1]/from[1]/text()[1]
```

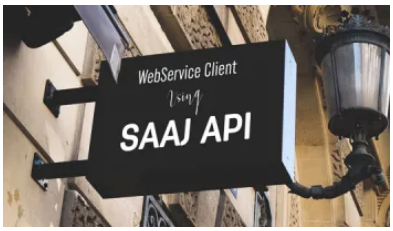
You can see in the result that the element tag & text difference was printed in the command prompt. Here is the screenprint of the application we will be creating using this.The error(element difference) will be highlited in RED and TEXT difference in GREEN.



Related

How to convert XML to JSON in Java

In "XML"



How to write Java WebService Client using SAAJ API?

In "Java"

An Introduction to Spring Boot

In "Spring Boot"

Filed Under: XML | Tagged With: Compare, Java, XML, XMLUnit

Subscribe to stay in loop

* indicates required

Email Address *

Comments

Vinay S Reddy says
September 12, 2016 at 9:20 am

(Edit)

Hi, I need your help in the code. I am getting “List cannot be resolved to a type” when I copied the code from https://www.adeveloperdiary.com/java/xml/compare-2-xml-in-java-part-1/.

I tried importing different import and still had issues. Please advise.

Reply

SFSZ says
December 19, 2016 at 5:51 pm

(Edit)

List allDifferences = myDiffd.getAllDifferences();

System.out.println(“Difference size: ” + allDifferences.size());
for (int i = 0; i < allDifferences.size(); i++) {
Difference diff = (Difference) allDifferences.get(i);
System.out.print(diff.getDescription() + " ");
System.out.print(diff.getId() + " ");

```
System.out.print(diff.getControlNodeDetail().getXpathLocation() + " ");
System.out.print(diff.getTestNodeDetail().getXpathLocation() + " ");
System.out.println(" ");
}
```

Reply

Anil says
December 30, 2016 at 1:05 pm

(Edit)

You can use List allDifferences = myDiffd.getAllDifferences(); instead also

Reply

Anil says
December 30, 2016 at 1:07 pm

(Edit)

Use List(Difference) allDifferences = myDiffd.getAllDifferences();

Replace '(' with less than and ')' with greater than symbols

Reply

Preethi says
February 14, 2017 at 8:25 am

(Edit)

Hi,

Can you please share me the complete source code to compare XML files.

Thanks and Regards,
Preethi

Reply

Rakesh says
May 2, 2017 at 7:08 am

(Edit)

Hi

Can you please share me the complete source code to compare XML files.

Thanks and Regards,
Rakesh

Reply

Krishna says
September 27, 2017 at 12:16 pm

(Edit)

Can you pls. give me the link for Part2

Reply

nanda says
April 1, 2020 at 8:44 am

(Edit)

hi may i know on what bases your coloring tags in ui?

Reply

Abhisek Jana says
April 5, 2020 at 12:15 am

(Edit)

Mm Dark Blue

Reply

Pavan says
September 16, 2020 at 3:23 am

(Edit)

Hi,

Please share part-2 link

Thanks

Reply

Hiren says

November 10, 2021 at 6:48 pm

(Edit)

How would I do this if I had just one XML file and wanted to compare two nodes within that?

...

...

Reply

Leave a Reply

Logged in as Abhisek Jana. Edit your profile. Log out? Required fields are marked *

Comment *

Post Comment

This site uses Akismet to reduce spam. Learn how your comment data is processed.