XML Data Compare

```xml
<person deltaxml:deltaV2="A!=B" customerid="16">
    <name deltaxml:deltaV2="A=B">Sherlock Holmes</name>
    <email deltaxml:deltaV2="A=B">sherlock@sherlock-holmes.com</email>
    <address deltaxml:deltaV2="A=B">
        <line>221b Baker St</line>
        <line>Marylebone</line>
        <line>London</line>
        <zip>NW1 6XE</zip>
    </address>
    <age deltaxml:deltaV2="A!=B">
        <deltaxml:textGroup deltaxml:deltaV2="A!=B">
            <deltaxml:text deltaxml:deltaV2="A">42</deltaxml:text>
            <deltaxml:text deltaxml:deltaV2="B">41.7</deltaxml:text>
        </deltaxml:textGroup>
    </age>
    <notes deltaxml:deltaV2="A=B">One night—it was on the 20th of March, 18:
    <extra deltaxml:deltaV2="A=B">His manner was not effusive. It seldom wa:
</person>
```

Product Benefits

- REST API

- Configurable treatment of element order

- Configurable numeric tolerance

- Configurable output formats

- Granular text changes

- Ignore unimportant changes

Take the guesswork out of what's changed.

XML Data Compare is a powerful REST-based toolkit for managing and processing change in your XML data content. With fully XML-aware compare functionality at your fingertips you'll always know exactly how your XML data has changed and keep on top of diverging data.

If you are frustrated by the way line-based diff tools handle your XML, or if you have found limitations in XML-aware diff tools, then you will find the ability to configure your comparison with XML Data Compare a real time saver. There is no need to write code, the configuration file is a simple XML file and a basic understanding of XPath is all that you need. If you are a more advanced user used to the full capabilities of XPath then you can make XML Data Compare handle complex scenarios easily.

REST API

XML Data Compare is deployed as an on-premise server with all functionality available via its REST API. This brings you the flexibility to access the toolkit however you want to, and total security as deployment is under your control. A command-line invocation, Postman calls, or a tight integration into your own environment all make use of the same simple API.

Changed order

Very often in XML data files the order of the elements is not significant, and you do not want to see changes where the only change is in the order. This is simply configured to allow you to specify exactly which elements should be treated this way. Of course a change in the order of attributes never shows up as a change.

Numeric data

It can be very annoying to see very small changes in numeric values, or a change in the textual representation such as 10.0 and 10, showing up as changes that need to be reviewed. You can configure a tolerance for numeric data to ignore changes that you do not regard as significant. A real time-saver for large numeric data files.

Text handling

Advanced word-by-word comparison of text with the ability to pick out the best matches across two text strings provides results that are easy to understand. If your XML is mainly text rather than data then do look at XML Compare for even more powerful text-handling features.

Output

You may just want to review the changes in which case an HTML output to review in a browser is what you need. But it is often far more useful to be able to put the comparison result into an automated process and handle changes automatically, with no time-consuming manual review. This is where the advanced delta file output really comes into its own, enabling you to take action automatically based on what has changed. This might be automatically acting on some changes or converting changes from XML into SQL or some other format to pass the changes on to the next stage in your process.