

Article

[Chris Stewart](#)

· Feb 18, 2021



2m read

## Basic JSON Compare

Hi Dev Community

I thought i would share a little method I knocked together to traverse and compare 2 JSON objects for basic equivilance. I'm currently working on some data migration, and wanted a basic sanity check to validate that the JSON output is basically equivliant between the old and new, excluding a few things like timestamps.

It's a basic little recursive method, that will bubble up any differences over a nested structure. It's very low tech, as that's all I need it to do, but I thought it might be useful for others?

It can accept a source and target JSON object or array, and optionally a RefNo if you are wanting to batch run this, and track the instances of differences in a ^zKeyTrap global. Additionally. there's an exclusion list for any keys which would would always expect to be differnt, such as ModificationTimestamps

```
ClassMethod CompareJSON(source As %String, target As %String, RefNo As
%String = 1) As %Boolean
{
set Identical = 1
Set propsIteratorsource = source.%GetIterator()
While (propsIteratorsource.%GetNext(.key,.value)) {
//Check for differences, and optionally exclude some keys
if (value '= target.%Get(key)&&("LastModifiedTime,LocalTime"[key)) {
if (source.%GetTypeOf(key)="object" || (source.%GetTypeOf(key)="array")){
s tSC= ..CompareJSON(value,target.%Get(key),RefNo)
} else {
w !,"Source ",key,?30,value
w !,"Target ",key,?30,target.%Get(key)
//Store a reference if running in Batch mode, for later review
s ^zKeyTrap(key)=RefNo
set tSC = 0 // A mismatch means key-value is not identical
}
// Multiply so that any zeros bubble up
set Identical=(Identical*tSC)
}
}
q Identical
}
```

Please feel free to use and adapt this if it would be helpful in your use of JSON

[#JSON](#) [#ObjectScript](#) [#InterSystems](#) [IRIS](#)

Source URL: <https://community.intersystems.com/post/basic-json-compare>