# Json Splitter Job Details :

The usecase is to split a big and complex nested json file domain wise into multiple json file. We can use this codebase for any kind of json file with dynamic nodes.

The Codebase enhancement is in progress to make to more generic by adding more functionalities.

**Sample input file can be refer from folder “input\_file”.**

|  |  |
| --- | --- |
| **High Level Description** | **Split Level** |
| * No nested objects in the model * All child objects will have a json model * No data loss – It should support for new attribute addition in the source * For each model, add Metadata like object\_class (domain level), object\_class\_instance (leaf level), itemId, relationTree (unique at leaf level) etc. | Leaf |
| * Nested objects will present for the provided json nodes (json path) in config file for which we don’t want to split for each child node. * There will be json objects according to json nodes provided in config file. * No data loss – It should support for new attribute addition in the source * For each model, add Metadata like object\_class (domain level), object\_class\_instance (leaf level), itemId, relationTree (unique at leaf level) etc. | domain |

**Description :** This Application is developed using Scala. **Play JSON** (a powerful Scala **JSON** library) has been used to split the json.

This Application make sure that there will not be any data loss and based upon given DOMAIN\_LEVEL\_KEYS in config file, it supports split at :

1. Only at Leaf Level
2. Only at Domain Level
3. Both Leaf and Domain Level Split

This Application removes all the Json Nodes having Blank Values.

This code base is generic and configurable which is able to split any kind of json without any change in codebase. This codebase supports leaf level and domain level both level of splits. the nodes we want to split at domain level, needs to be provide in config file and remaining all json nodes will be split at leaf level by default.

For example, below is the comma (',') separated Domain key (Json Path) has been provided into config file, so all these nodes will be spliited as a single json for each domain, remaining all nodes of the input json file will be splitted at leaf level.

**DOMAIN\_LEVEL\_KEYS =**"HW\_Info/Cables,HW\_Info/Disk\*\*/Partition”

We have provided **\*\*** (Example : HW\_Info/**Disk\*\***/Partition) with the nodes to be identify that this Node is an Array which contains Array inside Array. Here Disk is an Array of Multiple Disk where each Disk can have Multiple Partitions.

**Let's see this example with a small input json file to get clarity and Better Understanding :**

Sample Json file with few Nodes : {

"TEST\_Log": {

"Test": {

"Data\_Type": "Test",

"Test\_Info": "Splitting Nested Json at leaf or Domain Level or in both as per requirement",

"Test\_Version": "1"

},

"HW\_Info": {

"HardwareInfoID": "A30B",

"Test\_Revision": "1.2.0",

"Data\_Date": "2017-05-04T16:45:58+05:30",

"File\_Creation\_Datetime": "2019-08-12T16:48:40+06:30",

"Monitor": {

"Monitor\_Type": "Indian Monitor",

"HardwareInfoID": "A30B",

"Model\_Name": "Test.1150",

"Index": 0

},

"Disk": [

{

"Read\_Time": 0,

"Partition": [

{

"HardwareInfoID": "A30B",

"DiskId": "Test-177A",

"Index": 0,

"Size\_MB": 499,

"Name": "Not Used"

},

{

"HardwareInfoID": "A30B",

"DiskId": "Test-177A",

"Index": 1,

"Size\_MB": 100,

"Name": "Not Used"

},

{

"HardwareInfoID": "A30B",

"DiskId": "Test-177A",

"Index": 2,

"Size\_MB": 243582,

"Name": "C:"

}

],

"Idle\_Time": 97,

"HardwareInfoID": "A30B",

"DiskId": "Test-177A",

"Index": 0,

"Bytes\_Write\_MB": 649,

"Size\_MB": 244191,

"Make\_Model": "Test Model 256GB",

"Write\_Time": 0,

"Name": "DISK0",

"Bytes\_Read\_MB": 62766

},

{

"Read\_Time": 0,

"Partition": {

"HardwareInfoID": "A30B",

"DiskId": "Test-177A",

"Index": 0,

"Size\_MB": 14943,

"Name": "D:"

},

"Idle\_Time": 98,

"HardwareInfoID": "A30B",

"DiskId": "Test-177A",

"Index": 1,

"Bytes\_Write\_MB": 0,

"Size\_MB": 14943,

"Make\_Model": "Test USB Device",

"Write\_Time": 0,

"Name": "DISK1",

"Bytes\_Read\_MB": 0

}

],

"Processor": [

{

"Index": 0,

"Min\_ActiveFreq": 0.75,

"Max\_ActiveFreq": 0.85,

"Avg\_ActiveFreq": 0.80

},

{

"Index": 1,

"Min\_ActiveFreq": 0.62,

"Max\_ActiveFreq": 0.91,

"Avg\_ActiveFreq": 0.72

},

{

"Index": 2,

"Min\_ActiveFreq": 0.77,

"Max\_ActiveFreq": 1.23,

"Avg\_ActiveFreq": 0.99

},

{

"Index": 3,

"Min\_ActiveFreq": 0.57,

"Max\_ActiveFreq": 1.23,

"Avg\_ActiveFreq": 0.95

}

]

}

}

}

In this above json, we can see that json has root node as TEST\_Log and then its one child node as HW\_Info, Monitor, Disk (which is an array of multiple Disk and each Disk may have its child node as Array of Partition Nodes) and Processor (Which is an Array of Multiple Processors).

**Note :** While we split the json file, we add these as headers into each splitted json which makes it unique and easy to identify as per need :

parent\_id, (Common number for each splitted json from a single file to identify that all these json objects belongs to a particular file.)

item\_id, (Unique number for each splitted json to perform aggregation or any other required function.)   
scan\_time, (File Create Time from File Data or extract timestamp from file name if File Create Time node is not present in file.)  
file\_path, (The file path (we can use it for real time use case))  
object\_class, (For which node json is splitted.)  
object\_class\_instance, (Number of instances in case of Array of json.)  
relation\_tree, (Unique relation tree)  
split\_level (Split level is leaf or domain.)

# 1. ****How this Json will be splitted at leaf level : (each Child node will be splitted as a separate json) :****

**Domain Key in Config file will be :**

**DOMAIN\_LEVEL\_KEYS = "domain/defaultKey"**

**Output : Splitted Json files at leaf level :**

[{"item\_id":-453360329,"Processor":{"Avg\_ActiveFreq":"0.99","Max\_ActiveFreq":"1.23","Index":"2","Min\_ActiveFreq":"0.77"},"parent\_id":449967470,"object\_class\_instance":"Processor-3","object\_class":"Processor","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Processor~>Processor-3"},

{"item\_id":508169023,"parent\_id":449967470,"object\_class\_instance":"HW\_Info","object\_class":"HW\_Info","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info","HW\_Info":{"File\_Creation\_Datetime":"2019-08-12T16:48:40+06:30","Test\_Revision":"1.2.0","HardwareInfoID":"A30B","Data\_Date":"2017-05-04T16:45:58+05:30"}},

{"item\_id":-1761754140,"parent\_id":449967470,"Disk":{"Idle\_Time":"98","HardwareInfoID":"A30B","Name":"DISK1","Bytes\_Write\_MB":"0","Make\_Model":"Test USB Device","Bytes\_Read\_MB":"0","Size\_MB":"14943","DiskId":"Test-177A","Write\_Time":"0","Read\_Time":"0","Index":"1"},"object\_class\_instance":"Disk-2","object\_class":"Disk","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Disk~>Disk-2"},

{"item\_id":-453359368,"Processor":{"Index":"3","Max\_ActiveFreq":"1.23","Min\_ActiveFreq":"0.57","Avg\_ActiveFreq":"0.95"},"parent\_id":449967470,"object\_class\_instance":"Processor-4","object\_class":"Processor","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Processor~>Processor-4"},

{"item\_id":1329186251,"parent\_id":449967470,"object\_class\_instance":"Partition-3","object\_class":"Partition","Partition":{"HardwareInfoID":"A30B","Name":"C:","Size\_MB":"243582","DiskId":"Test-177A","Index":"2"},"file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Disk~>Disk-1~>Partition~>Partition-3"},

{"item\_id":-56133403,"parent\_id":449967470,"object\_class\_instance":"Partition","object\_class":"Partition","Partition":{"HardwareInfoID":"A30B","Name":"D:","Size\_MB":"14943","DiskId":"Test-177A","Index":"0"},"file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Disk~>Disk-2~>Partition"},

{"item\_id":1329185290,"parent\_id":449967470,"object\_class\_instance":"Partition-2","object\_class":"Partition","Partition":{"HardwareInfoID":"A30B","Name":"Not Used","Size\_MB":"100","DiskId":"Test-177A","Index":"1"},"file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Disk~>Disk-1~>Partition~>Partition-2"},

{"item\_id":-453362251,"Processor":{"Max\_ActiveFreq":"0.85","Index":"0","Min\_ActiveFreq":"0.75","Avg\_ActiveFreq":"0.8"},"parent\_id":449967470,"object\_class\_instance":"Processor-1","object\_class":"Processor","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Processor~>Processor-1"},

{"item\_id":215123559,"parent\_id":449967470,"object\_class\_instance":"Test","object\_class":"Test","file\_path":"input\_file\_path","Test":{"Data\_Type":"Test","Test\_Version":"1","Test\_Info":"Splitting Nested Json at leaf or Domain Level or in both as per requirement"},"scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>Test"},

{"item\_id":-517174416,"parent\_id":449967470,"Monitor":{"Monitor\_Type":"Indian Monitor","Model\_Name":"Test.1150","Index":"0","HardwareInfoID":"A30B"},"object\_class\_instance":"Monitor","object\_class":"Monitor","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Monitor"},

{"item\_id":1329184329,"parent\_id":449967470,"object\_class\_instance":"Partition-1","object\_class":"Partition","Partition":{"HardwareInfoID":"A30B","Name":"Not Used","Size\_MB":"499","DiskId":"Test-177A","Index":"0"},"file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Disk~>Disk-1~>Partition~>Partition-1"},

{"item\_id":-1761755101,"parent\_id":449967470,"Disk":{"Idle\_Time":"97","HardwareInfoID":"A30B","Name":"DISK0","Bytes\_Write\_MB":"649","Make\_Model":"Test Model 256GB","Bytes\_Read\_MB":"62766","Size\_MB":"244191","DiskId":"Test-177A","Write\_Time":"0","Read\_Time":"0","Index":"0"},"object\_class\_instance":"Disk-1","object\_class":"Disk","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Disk~>Disk-1"},

{"item\_id":-453361290,"Processor":{"Min\_ActiveFreq":"0.62","Avg\_ActiveFreq":"0.72","Max\_ActiveFreq":"0.91","Index":"1"},"parent\_id":449967470,"object\_class\_instance":"Processor-2","object\_class":"Processor","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Processor~>Processor-2"}]

# 2. ****How this Json will be splitted at domain level : (each provided node in in below DOMAIN\_LEVEL\_KEYS will be splitted at domain level and remaining nodes will be at leaf level) :****

**Domain Key in Config file will be :**

**DOMAIN\_LEVEL\_KEYS = "domain/defaultKey,HW\_Info/Cables,HW\_Info/Disk\*\*/Partition"**

**Output : Splitted Json files at both leaf and domain level :**

[

{"item\_id":508169023,"parent\_id":449967470,"object\_class\_instance":"HW\_Info","object\_class":"HW\_Info","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info","HW\_Info":{"File\_Creation\_Datetime":"2019-08-12T16:48:40+06:30","Test\_Revision":"1.2.0","HardwareInfoID":"A30B","Data\_Date":"2017-05-04T16:45:58+05:30"}},

{"item\_id":-1761754140,"parent\_id":449967470,"Disk":{"Idle\_Time":"98","HardwareInfoID":"A30B","Name":"DISK1","Bytes\_Write\_MB":"0","Make\_Model":"Test USB Device","Bytes\_Read\_MB":"0","Size\_MB":"14943","DiskId":"Test-177A","Write\_Time":"0","Read\_Time":"0","Index":"1"},"object\_class\_instance":"Disk-2","object\_class":"Disk","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Disk~>Disk-2"},

{"item\_id":215123559,"parent\_id":449967470,"object\_class\_instance":"Test","object\_class":"Test","file\_path":"input\_file\_path","Test":{"Data\_Type":"Test","Test\_Version":"1","Test\_Info":"Splitting Nested Json at leaf or Domain Level or in both as per requirement"},"scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>Test"},

{"item\_id":-517174416,"parent\_id":449967470,"Monitor":{"Monitor\_Type":"Indian Monitor","Model\_Name":"Test.1150","Index":"0","HardwareInfoID":"A30B"},"object\_class\_instance":"Monitor","object\_class":"Monitor","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Monitor"},

{"item\_id":-1761755101,"parent\_id":449967470,"Disk":{"Idle\_Time":"97","HardwareInfoID":"A30B","Name":"DISK0","Bytes\_Write\_MB":"649","Make\_Model":"Test Model 256GB","Bytes\_Read\_MB":"62766","Size\_MB":"244191","DiskId":"Test-177A","Write\_Time":"0","Read\_Time":"0","Index":"0"},"object\_class\_instance":"Disk-1","object\_class":"Disk","file\_path":"input\_file\_path","scan\_time":"2019-08-12 10:18:40.000","split\_level":"leaf","relation\_tree":"TEST\_Log~>HW\_Info~>Disk~>Disk-1"},

{"file\_path":"input\_file\_path","Processor":[{"Index":0,"Min\_ActiveFreq":0.75,"Max\_ActiveFreq":0.85,"Avg\_ActiveFreq":0.8},{"Index":1,"Min\_ActiveFreq":0.62,"Max\_ActiveFreq":0.91,"Avg\_ActiveFreq":0.72},{"Index":2,"Min\_ActiveFreq":0.77,"Max\_ActiveFreq":1.23,"Avg\_ActiveFreq":0.99},{"Index":3,"Min\_ActiveFreq":0.57,"Max\_ActiveFreq":1.23,"Avg\_ActiveFreq":0.95}],"object\_class\_instance":"Processor","item\_id":-163178008,"parent\_id":449967470,"relation\_tree":"TEST\_Log~>HW\_Info~>Processor","split\_level":"domain","scan\_time":"2019-08-12 10:18:40.000","object\_class":"Processor"},

{"Partition":[{"HardwareInfoID":"A30B","DiskId":"Test-177A","Index":0,"Size\_MB":499,"Name":"Not Used"},{"HardwareInfoID":"A30B","DiskId":"Test-177A","Index":1,"Size\_MB":100,"Name":"Not Used"},{"HardwareInfoID":"A30B","DiskId":"Test-177A","Index":2,"Size\_MB":243582,"Name":"C:"}],"parent\_id":449967470,"item\_id":788338468,"scan\_time":"2019-08-12 10:18:40.000","file\_path":"input\_file\_path","object\_class":"Partition","object\_class\_instance":"Partition","relation\_tree":"TEST\_Log~>HW\_Info~>Disk~>Disk-1~>Partition","split\_level":"domain"},

{"Partition":{"HardwareInfoID":"A30B","DiskId":"Test-177A","Index":0,"Size\_MB":14943,"Name":"D:"},"parent\_id":449967470,"item\_id":-56133403,"scan\_time":"2019-08-12 10:18:40.000","file\_path":"input\_file\_path","object\_class":"Partition","object\_class\_instance":"Partition","relation\_tree":"TEST\_Log~>HW\_Info~>Disk~>Disk-2~>Partition","split\_level":"domain"}

]