**Version Slice Mapping:**

*1.1 - Function Signature:*

*x.y.z.versionSlice[1], x.versionSlice[2], x.y.versionSlice[0] etc.*

1. **Inputs:**

*( Items, Vn )*

*Items* *– Type (String)*

Items up to which JSON must be sliced.

*Version number – Type (Integer)*

Nth version of the last item in Items

1. **Output:**

*Temporal JSON Object:*

This JSON object contains slice of Nth version of item.

A slice is a sub-part of parent JSON which contains item information along temporal data (Timestamps when the item was alive).

* 1. *– Function Mapping:*

“ x.y.versionSlice[2] ”

data = temporalJSON

**Algorithm Steps:**

1. For each ***x*** version in data:

list\_x.append( data [***x]*** ) //List of all versions of *x*

1. For each ***y*** in list\_x:

list\_y.append( list\_x [***y***] ) //List of all versions of y

1. For each *version* in list\_y: //Keep only unique versions

Unique\_versions.append(**Coalesce**(*version*))

1. Return Unique\_versions [2] //Returns object

**Time Snapshot Mapping:**

* 1. *Function Signature:*

x.y.timeSnapshot[2015-2017]

x.y.z.timeSnapshot[2018-2018]

x.timeSnapshot[2016]

**a. Inputs - (Items, T)**

***Items* -** Type (Strings)

Items for which time Snapshots need to be made. The last item in this list would be the one whose time snapshot will be generated.

***T*** – timestamp, Type (String)

**b. Output – JSON array**

An array of valid JSON objects containing data information of item without any temporal information.

These objects will be the snapshots of the item for timestamp T.

*1.2 Function Mapping:*

*data* = temporalJSON

Ex: *Items = [Specimen.Colloquial]*

**Algorithm Steps:**

1. While data[Items[i].timestamp] == T and i! = length(***Items***):

**new\_data** = recursively get the ith item from data[***Items***]

1. *Get timestamps from* ***new\_data****:*

timestamps = getTimestamps(new\_data)

1. *Preprocess the timestamps array and create skeleton for snapshots:*

Skeleton = preProcess(timestamps)

1. *Populate Skeleton:*

For each obj in Skeleton:

if Skeleton[obj][“timestamp”] **OVERLAPS** with data[item][“timestamp”]:

Skeleton[data].update(data[item][data])

1. *Return Snapshots array*