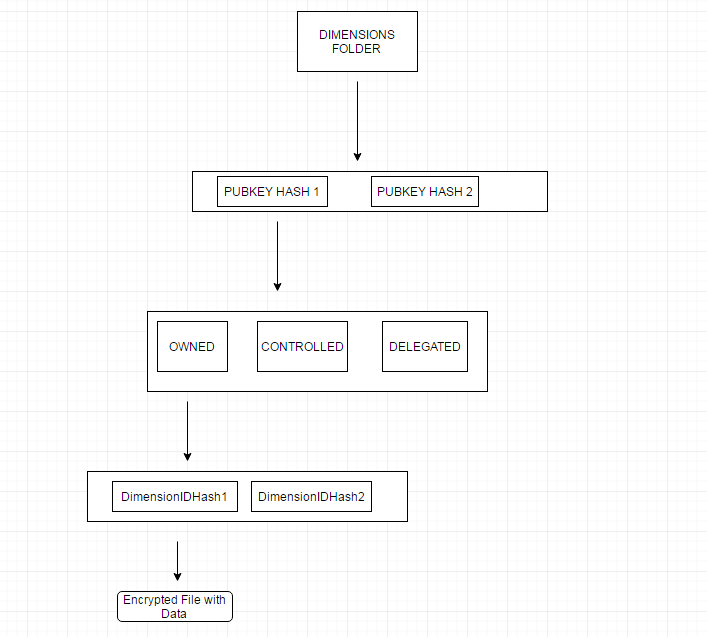
**DIGITAL TWIN—IDENTITY DIMENSIONS DOCUMENTATION**

Below is a diagram. It describes the folder structure of identity dimensions in the digital twin.



The folder structure is located at:

/home/demoadmin/DigitalTwin/dimensions

Regarding the implementation, there are three relevant files:

1. IdentityDimensionCtrl.js (analogous to AssetCtrl.js)
2. DimensionCtrlConfig.json (analogous toAssetCtrlConfig.json)
3. TwinListener.js (where dimension endpoints are called)

**Keeping Multiple Twins in Sync with Information**

Suppose there are five owners and four controllers for a Core Identity. Denote one owner as *Owner A*. Suppose *Owner A* wants to add an entry. Upon adding an Entry, this Entry should appear in all of the other owners and controller’s wallets.

Therefore, here is the process for ADD ENTRY:

1. Have the Owner make the post request to the digital twin
2. Have the digital twin forward the request to the identity dimension JavaScript
3. Wait for the response from the JavaScript and validate that the add was successful
4. Get all owners and controllers for the relevant Core Identity and call setDimension by adding the value

The process for DELETE ENTRY is analogous to the above. The process is also similar for ADD DIMENSION and DELETE DIMENSION.

Now, what about changes to a Core Identity. Let’s suppose *User A* has just been added to a Core Identity as an owner. When they log into their wallet, they should see all dimensions. Here is the relevant process when ADDING AN OWNER TO A CORE IDENTITY:

1. Have the call to the new COID for addOwner
2. Have the MyCOID javascript contain reference to the Identity Dimension
3. Create the folder structure for dimensions to the added owner.

The process is similar for REMOVE CONTROLLER.

Here are a list of operations that will require syncing of information:

1. Add/remove Owner/Controller for a COID
2. Add/remove/Update

There are five functions in IdentityDimensionCtrl.js. These functions are called in TwinListener.js in response to endpoints:

|  |  |
| --- | --- |
| **Function** | **INPUT (JSON)** |
| getOwnedDimensions |  |
| getControlledDimensions |  |
| getDelegatedDimensions |  |
| getDimension |  |
| setDimension |  |
| deleteDimension | pubKey  flag (0 = owned, 1 = controlled, 2 = delegated)  fileName |