

# ProjectHOL (PoC) - Documentation

## Transaction Processors (TP)

There is a total of 4 TPs initialized in the Docker environment of this project:

- excise 1.0 [not active within the scope of this PoC]
- manufacturing 1.0
- transfer 1.0
- sales 1.0

Each TP has a unique business function as follows:

### excise 1.0

This TP is initialized however not actively used. The sole purpose of this TP is to showcase how this TP would act according to the architecture as mentioned in the design decision document.

Ideally this TP would provide access rights to the network for each of the stakeholders involved in the supply chain. i.e. Manufacturer, Stockist, Warehouse, Point of Sale (POS).

### manufacturing 1.0

This TP provides the business logic to create an asset within the chain i.e. the starting point of the asset before it flows down through the supply chain network. The *'createAsset'* function in processor/ManufacturerHandler.js executes this feature.

### transfer 1.0

This TP provides the business logic to record the transfer of custodianship of the created asset between any two entities within the supply chain i.e. from manufacturer → stockist or stockist → warehouse or warehouse → POS.

The *'transferAsset'* function in processor/TransferHandler.js executes this. The function updates the state value based on from where the transfer is being initiated.

### sales 1.0

This TP provides the business logic to update the state with sale detail of the asset i.e. the timestamp of the sale made.

The *'saleAsset'* function in processor/SalesHandler.js executes this.

## Clients

This project assumes that each individual stakeholder has a unique view and access to the network with capabilities limited to their functionality. Hence this project has 5 different client views, each of which can be accessed via its individual URL as follows:

## Manufacturer

<http://localhost:4200> or <http://localhost:4200/manufacturer>

This view would allow the user (manufacturer) to create a new asset and make a consecutive transfer of that asset to the next stakeholder i.e. the stockist. This view invokes the **manufacturing 1.0** TP mentioned above.

## Stockist

<http://localhost:4200/stockist>

This view would allow the user (stockist) to push an asset to the next stakeholder i.e. the warehouse. This view invokes the **transfer 1.0** TP mentioned above.

## Warehouse

<http://localhost:4200/warehouse>

This view would allow the user (warehouse) to push an asset to the next stakeholder i.e. the point of sale. This view also invokes the **transfer 1.0** TP.

## Sale

<http://localhost:4200/sale>

This view would allow the user (POS) to push an asset to the next stakeholder i.e. the customer. This view **sales 1.0** TP.

## Customer

<http://localhost:4200/customer>

This view would allow the end user to view the detail as stored in the state. As for this particular functionality when running this view do ensure that you are using Firefox/Chrome and you have the inspection window opened (Ctrl+Shift+i). Since the values being returned

are written to console and not onto the webpage. The values are in the order that its being taken. Refer snap below:

