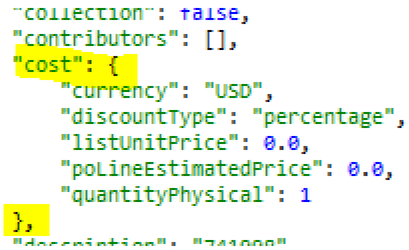
Querying array elements from a table's data blob (useful for non-MetaDB data)

This link is to a very short write up by Angela Zoss on how to query elements out of a table's data blob in the LDP. The formulas here work for simple data elements (elements shown in curly brackets) and array elements (enclosed within square brackets).

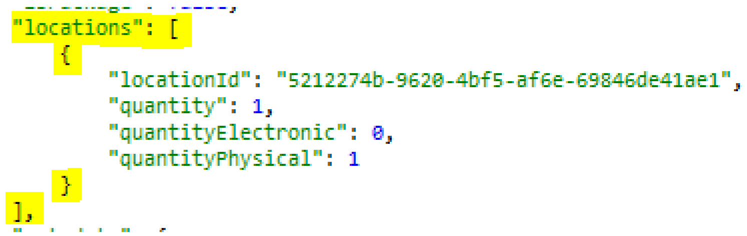
https://wiki.folio.org/pages/viewpage.action?pageId=36578061

Examples from the po_lines table. The element "cost" has values in curly brackets { }. These would be queried out by the simple "JSON_EXTRACT_PATH_TEXT (po_lines.data,'cost','xxxx')" command for each element.



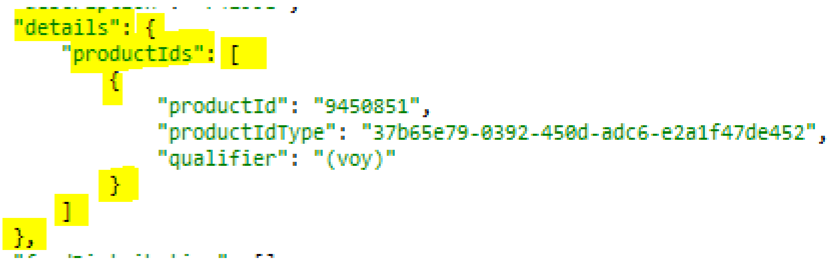
```
"collection": false,
"contributors": [],
"cost": {
    "currency": "USD",
    "discountType": "percentage",
    "listUnitPrice": 0.0,
    "poLineEstimatedPrice": 0.0,
    "quantityPhysical": 1
},
"description": "741998"
```

The "locations" group elements are enclosed in square brackets AND curly brackets [{ }], so would need a more complex command to get each element: JSON_EXTRACT_PATH_TEXT (JSON_ARRAY_ELEMENTS (JSON_EXTRACT_PATH (po_lines.data, 'locations')),'locationId') gets the locationId element. You would repeat this, substituting the name of each element needed at the end of the command (quantity, quantityElectronic, quantityPhysical).



```
"locations": [
    {
        "locationId": "5212274b-9620-4bf5-af6e-69846de41ae1",
        "quantity": 1,
        "quantityElectronic": 0,
        "quantityPhysical": 1
    }
],
```

Some objects in the data blob are even more complex - an example is the "details" stanza of the po_lines data blob. The information is nested inside curly brackets, square brackets and curly brackets { [{ }] } The information at the link doesn't get into how to parse out those elements. You're on your own!



```
"details": {
    "productIds": [
        {
            "productId": "9450851",
            "productIdType": "37b65e79-0392-450d-adc6-e2a1f47de452",
            "qualifier": "(voy)"
        }
    ]
},
```

Extracting from the updated inventory_instances table: as of the Orchid release, changes were made to inventory_instances table. This example shows how to extract values from the data array when the values are "objects". This example shows getting the subjects from the data blob:

```
SELECT

    instances.id AS instance_id,

    instances.hrid AS instance_hrid,

    subjects.data #>> '{value}' AS subject,

    subjects.ordinality AS subject_ordinality

FROM
```

```
    inventory_instances AS instances
CROSS JOIN jsonb_array_elements((instances.data #> '{subjects}')::jsonb)
WITH ORDINALITY AS subjects (data)
```