# **SmartLens for Retail API Documentation**

## **Introduction**

SmartLens for Retail provides a RESTful web services interface to query position and state data for items and assets, as well as region configuration data, tag associations, and import services.

**Authorization** All REST queries require authorization. We currently support **BASIC authorization**. Please contact your SmartLens system administrator for the authorization credentials for your site.

## **API Endpoints**

### **1. Query Items by ID**

Retrieve the location of a single item using the EPC value that identifies the RFID tag.

#### **Request**

**GET** https://<hostname>/zebra/smartlens/items('<item\_id>')

#### **Example Request**

GET https://<hostname>/zebra/smartlens/items('30340789004D0D8000000074')

#### **Response**

{

"id":"30340789004D0D8000000074",

"x":900,

"y":900,

"z":900,

"confidence":60,

"floor":0,

"type":"ITEM",

"timestamp": "2020-08-01T03:11:09Z",

"region":"4d03cf0b-f856-49f6-87fb-f8a40a10271d",

"site":"936c1954-dce8-4783-90d3-cb9a7db19698",

"productId":"00123456789029",

"idFormat":"EPC",

"associatedId":"",

"associatedIdFormat":"",

"state":"MISSING",

"reason": "NONE"

}

### **2. Query Items by Product ID**

Retrieve items matching a full or partial Product ID.

#### **Request**

**GET** https://<hostname>/zebra/smartlens/items?$filter=productId eq '<product\_id>'

#### **Example Request**

GET https://<hostname>/zebra/smartlens/items?$filter=productId eq '0012345'

#### **Response**

{

"value":[

{

"id":"30340789004D0D8000000074",

"x":900,

"y":900,

"z":900,

"confidence":60,

"floor":0,

"type":"ITEM",

"timestamp": "2020-07-31T02:46:18Z",

"region":"4d03cf0b-f856-49f6-87fb-f8a40a10271d",

"site":"936c1954-dce8-4783-90d3-cb9a7db19698",

"productId":"00123456789029",

"idFormat":"SGTIN96",

"associatedId":"",

"associatedIdFormat":"",

"state":"VALID",

"reason": "NONE"

}

]

}

### **3. Query Items by Multiple Product IDs**

Retrieve items matching multiple Product IDs.

#### **Request**

**GET** https://<hostname>/zebra/smartlens/items?$filter=productId eq '<product\_id1>,<product\_id2>'

#### **Example Request**

GET https://<hostname>/zebra/smartlens/items?$filter=productId eq '00123456789029,00123456789031'

#### **Response**

{

"value":[

{

"id":"30340789004D0D8000000074",

"productId":"00123456789029",

"state":"VALID"

},

{

"id":"30340789004D0DC00001869F",

"productId":"00123456789031",

"state":"VALID"

}

]

}

### **4. Query Items by Region**

Retrieve all items within a given region.

#### **Request**

**GET** https://<hostname>/zebra/smartlens/items?$filter=region eq '<region\_id>'

#### **Example Request**

GET https://<hostname>/zebra/smartlens/items?$filter=region eq '4d03cf0b-f856-49f6-87fb-f8a40a10271d'

### **5. Query Items by State**

Retrieve items filtered by state (MISSING, VALID, DEPARTED, GHOST).

#### **Request**

**GET** https://<hostname>/zebra/smartlens/items?$filter=state eq '<state>'

#### **Example Request**

GET https://<hostname>/zebra/smartlens/items?$filter=state eq 'MISSING,DEPARTED'

### **6. Query All Items**

Retrieve all items (may return a large dataset).

#### **Request**

**GET** https://<hostname>/zebra/smartlens/items

#### **Example Request**

GET https://<hostname>/zebra/smartlens/items

## **Regions API**

### **7. Query All Regions**

Retrieve all regions configured in the SmartLens Application Server.

#### **Request**

**GET** https://<hostname>/zebra/smartlens/regions

#### **Example Request**

GET https://<hostname>/zebra/smartlens/regions

#### **Response**

{

"@odata.context":"$metadata#regions",

"value":[

{

"id":"6a0027c7-86fd-48e2-8642-a7a1bbb822c3",

"name":"World",

"type":"EXCLUSION",

"floorNumber":0,

"site":"936c1954-dce8-4783-90d3-cb9a7db19698",

"parent":""

},

{

"id":"d11c1bf5-b45b-4ab9-9d65-f4f591b2152a",

"name":"Inside",

"type":"INCLUSION",

"floorNumber":0,

"site":"936c1954-dce8-4783-90d3-cb9a7db19698",

"parent":"6a0027c7-86fd-48e2-8642-a7a1bbb822c3"

}

]

}

## **Conclusion**

This API documentation provides an overview of the RESTful endpoints available in SmartLens for Retail. Ensure that authentication details are provided with each request. Contact your system administrator for further support.

## **RESTful Web Services Interface**

### **1. Items**

#### **Query Items by ID**

Query for the location of a single item using the ID (EPC value) that identifies the RFID tag of the item.

**Request:**

GET https://<hostname>/zebra/smartlens/items('30340789004D0D8000000074')

**Response:**

{

"id": "30340789004D0D8000000074",

"x": 900,

"y": 900,

"z": 900,

"confidence": 60,

"floor": 0,

"type": "ITEM",

"timestamp": "2020-08-01T03:11:09Z",

"region": "4d03cf0b-f856-49f6-87fb-f8a40a10271d",

"site": "936c1954-dce8-4783-90d3-cb9a7db19698",

"productId": "00123456789029",

"idFormat": "EPC",

"associatedId": "",

"associatedIdFormat": "",

"state": "MISSING",

"reason": "NONE"

}

#### **Query Items by ProductId**

Query for the location of items by matching a full or partial ProductId value.

**Request:**

GET https://<hostname>/zebra/smartlens/items?$filter=productId eq '0012345'

**Response:**

{

"value": [

{

"id": "30340789004D0D8000000074",

"x": 900,

"y": 900,

"z": 900,

"confidence": 60,

"floor": 0,

"type": "ITEM",

"timestamp": "2020-07-31T02:46:18Z",

"region": "4d03cf0b-f856-49f6-87fb-f8a40a10271d",

"site": "936c1954-dce8-4783-90d3-cb9a7db19698",

"productId": "00123456789029",

"idFormat": "SGTIN96",

"associatedId": "",

"associatedIdFormat": "",

"state": "VALID",

"reason": "NONE"

}

]

}

#### **Query Items by Multiple ProductIds**

Query for the location of items by matching multiple ProductId values.

**Request:**

GET https://<hostname>/zebra/smartlens/items?$filter=productId in ('00123456789029','00123456789031')

**Response:**

{

"value": [

{

"id": "30340789004D0D8000000074",

"x": 900,

"y": 900,

"z": 900,

"confidence": 60,

"floor": 0,

"type": "ITEM",

"region": "4d03cf0b-f856-49f6-87fb-f8a40a10271d",

"site": "936c1954-dce8-4783-90d3-cb9a7db19698",

"productId": "00123456789029",

"idFormat": "EPC",

"associatedId": "",

"associatedIdFormat": "",

"reason": "NONE",

"state": "VALID"

}

]

}

### **2. Regions**

#### **Query All Regions**

Query for all regions configured within the SmartLens Application Server.

**Request:**

GET https://<hostname>/zebra/smartlens/regions

**Response:**

{

"@odata.context": "$metadata#regions",

"value": [

{

"id": "6a0027c7-86fd-48e2-8642-a7a1bbb822c3",

"name": "World",

"type": "EXCLUSION",

"floorNumber": 0,

"site": "936c1954-dce8-4783-90d3-cb9a7db19698",

"parent": ""

},

{

"id": "d11c1bf5-b45b-4ab9-9d65-f4f591b2152a",

"name": "Inside",

"type": "INCLUSION",

"floorNumber": 0,

"site": "936c1954-dce8-4783-90d3-cb9a7db19698",

"parent": "6a0027c7-86fd-48e2-8642-a7a1bbb822c3"

}

]

}

This documentation provides the necessary details to interact with the SmartLens API, including query parameters, request formats, and expected responses. Please ensure you use appropriate authentication before making API calls.

**TagAssociations Documentation**

## **Introduction**

The **TagAssociations API** in **SmartLens for Retail** provides an interface to associate or disassociate an **EPC** (Electronic Product Code) to a **ProductId**. It also enables modifying the "inventory state" of a tag in the **BAR (Business Analytics Repository)**. This API is commonly used with reusable tags and to manage sellable item statuses for **BOPIS (Buy Online, Pick Up In Store)**, claims, displays, and damaged tag use cases.

## **Association Status**

The table below describes how different **SmartLens App Server actions** impact the **BAR Inventory** availability:

| **Association Status** | **SmartLens App Server Action** | **BAR Available Inventory** |
| --- | --- | --- |
| Associated | Update ProductID | Yes |
| Disassociated | Remove ProductID | No |
| Ignore | N/A | No |
| Unignore | N/A | Yes |
| Discounted | N/A | No |
| Sold | N/A | No |
| Remove Association | N/A | No |
| Deleted | N/A | No |

## **Impact on BAR REST API**

Using **Item Disposition** affects the following endpoints in the **BAR REST API**:

* **Inventory-Controller**
* **Inventory-Event-Data-Controller**
* **Insight-Controller**
* **Insight-State-Controller**

Items marked as **Not Available Inventory** in BAR will **NOT** be included in inventory results or used in insight creation/state determination via these endpoints.

### **Exclusions**

Item Disposition **does not impact** the following:

* **SLAP API (all endpoints)**: The **"State"** value in SLAP is independent of inventory availability. An item may be in a **VALID** state but still be **unavailable** in inventory.
* **BAR Event-Data-Controller**: The event history remains unaffected, and the item continues to be tracked as it moves through the store and departs.

## **API Endpoints**

### **Query TagAssociations**

Retrieves all current **TagAssociations**.

**Endpoint:** GET https://<hostname>/zebra/smartlens/system/tagassociations

**Example Response:**

{

"id": "1131",

"productId": "00223003082122",

"upos": "",

"site": "9774ed35-1af8-446b-a339-7ca8ef80cd20",

"status": "ASSOCIATED",

"timestamp": "2020-08-03T20:55:52.500Z",

"state": "INVALID"

}

### **Insert or Update TagAssociation**

Inserts or updates a **TagAssociation**.

**Endpoint:** POST https://<hostname>/zebra/smartlens/system/tagassociations/insert

**Request Body:**

[

{

"id": "30340789004D0D8000000074",

"productId": "00123456789029",

"status": "IGNORE",

"timestamp": "2020-08-05T15:22:44+02:00",

"site": "Xe4b59c34-9167-3dea-8ad9-4cb19adcbccf",

"upos": ""

}

]

**Example Response:**

[

{

"id": "30340789004D0D8000000074",

"productId": "00123456789029",

"upos": "",

"site": "Xe4b59c34-9167-3dea-8ad9-4cb19adcbccf",

"status": "IGNORE",

"timestamp": "2020-08-05T15:22:44+02:00",

"state": "SUCCESS"

}

]

## **Conclusion**

The **TagAssociations API** provides robust functionality for managing item tracking and inventory disposition. It supports key operations such as:

* Associating/disassociating **EPC tags** with **ProductId**
* Updating the **inventory state** in BAR
* Managing **sellable statuses** for BOPIS, claims, and damaged tags

For further assistance, please contact your **SmartLens system administrator**.