

CINX API Integration

Harrison Publishing House, Inc.

Version 1.6

Contents

[Background 6](#_Toc13052924)

[CINX Application/Subscription Model Introduction 7](#_Toc13052925)

[CINX Data Source Introduction 8](#_Toc13052926)

[CINX Application and Data Source Introduction 9](#_Toc13052927)

[ReST API 11](#_Toc13052928)

[API – PING Server 13](#_Toc13052929)

[API – GET User Subscriptions/Authorizations (Orgs and Applications) 14](#_Toc13052930)

[Working with the GET User Subscriptions/Authorizations API Response 15](#_Toc13052931)

[API – GET Item Information using the “mcaa” Response View – Item GUID 20](#_Toc13052932)

[API – GET Item Information using the “mcaa” Response View – Org Part Number 21](#_Toc13052933)

[API – GET Item Information using the “mcaa” Response View – HPH Code 22](#_Toc13052934)

[“mcaa” View Response Contents 24](#_Toc13052935)

[API – GET Item Information using the “full” Response View – Item GUID 26](#_Toc13052936)

[API – GET Item Information using the “full” Response View – Org Part Number 27](#_Toc13052937)

[API – GET Item Information using the “full” Response View – HPH Code 28](#_Toc13052938)

[API – GET Item Information using the “custom” Response View – Item GUID 29](#_Toc13052939)

[API – GET Item Information using the “custom” Response View – HPH Code 30](#_Toc13052940)

[CINX Private Catalog Item Data Options – Custom View 31](#_Toc13052941)

[API – GET Item List Price History – HPH Code 32](#_Toc13052942)

[API – Add a New Catalog Item – HPH Code 33](#_Toc13052943)

[API – Create a non-HPH Private Catalog Item 34](#_Toc13052944)

[API – Modify a CINX Private Catalog Item – using a CINX GUID 36](#_Toc13052945)

[CINX Private Catalog Item JSON Structures 37](#_Toc13052946)

[API – DELETE a Catalog Item (based on an CINX Item GUID) 39](#_Toc13052947)

[API – GET Private Catalog Lines – Root Level 40](#_Toc13052948)

[API – GET a Private Catalog Line’s Children (sub-lines) 41](#_Toc13052949)

[CINX Catalog Update Introduction 42](#_Toc13052950)

[API – GET Catalog Updates List 45](#_Toc13052951)

[API – GET Catalog Update File - Download 47](#_Toc13052952)

[API – GET Catalog Update Bulletin File ONLY - Download 48](#_Toc13052953)

[API – Mark Catalog Update File As Applied 49](#_Toc13052954)

[API – GET Catalog Item Count 50](#_Toc13052955)

[API – GET Catalog Item Ids 51](#_Toc13052956)

[Manufacturer Price Sheet Introduction 52](#_Toc13052957)

[API – GET Manufacturer’s Price Sheet Masters 53](#_Toc13052958)

[API – GET Manufacturer’s Price Sheet Master Details 54](#_Toc13052959)

[API – GET Project List 55](#_Toc13052960)

[API – GET Project Details 56](#_Toc13052961)

[API – PUT New Project 57](#_Toc13052962)

[API – Modify a CINX Project 58](#_Toc13052963)

[CINX Project JSON Structures 59](#_Toc13052964)

[API – Delete a CINX Project 61](#_Toc13052965)

[CINX Project Job Cost Introduction 62](#_Toc13052966)

[API – GET a Project’s Phases and Cost Codes List 64](#_Toc13052967)

[API – Create a Project’s Phase and Cost Code 65](#_Toc13052968)

[API – Modify a Project’s Phase and Cost Code 67](#_Toc13052969)

[CINX Project Phase/Cost Code JSON Structures 68](#_Toc13052970)

[API – POST Project Items – csv Format for Autodesk Fabrication 70](#_Toc13052971)

[CINX BOM and BOM-ITEM Introduction 71](#_Toc13052972)

[API – GET BOM List – General Response 72](#_Toc13052973)

[API – GET Quotes for a Project 73](#_Toc13052974)

[API – GET Quotes for a Request for Quotation 75](#_Toc13052975)

[API – GET Requisitions for a Project 77](#_Toc13052976)

[API – GET Request for Quotations for a Project 79](#_Toc13052977)

[API – GET Purchase Orders for a Project 81](#_Toc13052978)

[API – GET Purchase Order Change Orders for a Project 83](#_Toc13052979)

[API – GET Invoices for a Project 85](#_Toc13052980)

[API – GET Vendor Deliveries for a Project 87](#_Toc13052981)

[API – GET Vendor Returns for a Project 89](#_Toc13052982)

[API – GET Child Transactions for a Specific Purchase Order 91](#_Toc13052983)

[API – GET Back-Ordered Items 92](#_Toc13052984)

[API – Mark a PO as Applied in an ERP System 94](#_Toc13052985)

[API – Create a New Requisition 95](#_Toc13052986)

[CINX Requisition JSON Structures 96](#_Toc13052987)

[API – GET BOM ITEMs 98](#_Toc13052988)

[API – Create a BOM ITEM – HPH Code 99](#_Toc13052989)

[API – GET BOM ITEM Details 100](#_Toc13052990)

[API – Modify a BOM ITEM 101](#_Toc13052991)

[CINX BOM-Item JSON Structures 102](#_Toc13052992)

[API – Delete a BOM ITEM 105](#_Toc13052993)

[CINX Job Cost Introduction 106](#_Toc13052994)

[API – GET a List of the Job Cost Phases 107](#_Toc13052995)

[API – Create a Job Cost Phase 108](#_Toc13052996)

[API – Modify a Job Cost Phase 110](#_Toc13052997)

[API – GET a List of the Job Cost Cost Codes 112](#_Toc13052998)

[API – Create a Job Cost Cost Code 113](#_Toc13052999)

[API – Modify a Job Cost Cost Code 115](#_Toc13053000)

[API – GET a List of the Job Cost Categories 117](#_Toc13053001)

[API – Create a Job Cost Category 118](#_Toc13053002)

[API – Modify a Job Cost Category 120](#_Toc13053003)

[API – GET a List of the Job Cost Tax Groups 122](#_Toc13053004)

[API – Create a Job Cost Tax Group 123](#_Toc13053005)

[API – Modify a Job Cost Tax Group 125](#_Toc13053006)

[CINX Vendor Introduction 127](#_Toc13053007)

[API – GET Vendor List 128](#_Toc13053008)

[API – GET Vendor Details – using CINX Commerce GUID 129](#_Toc13053009)

[API – PUT New Vendor 130](#_Toc13053010)

[API – Modify a Vendor 131](#_Toc13053011)

[CINX Vendor JSON Structures 132](#_Toc13053012)

[API Call – POST External file with User Notifications 133](#_Toc13053013)

[API – GET Received User Alerts 134](#_Toc13053014)

[API – Mark User Alerts Shown or Closed 136](#_Toc13053015)

[API – GET Organization Details 137](#_Toc13053016)

[API – Modify Organization Details 138](#_Toc13053017)

[CINX Organization JSON Structures 139](#_Toc13053018)

[API – GET an Organization’s User List 141](#_Toc13053019)

[API – GET an Organization’s Addresses 142](#_Toc13053020)

[API – GET an Organization’s Customer Type List 143](#_Toc13053021)

[API – GET an Organization’s Vendor Type List 144](#_Toc13053022)

[API – GET an Organization’s Project Type List 145](#_Toc13053023)

[API – GET an Organization’s Invoice Dispute Reasons List 146](#_Toc13053024)

[API – GET an Organization’s Return Reasons List 147](#_Toc13053025)

[API – GET an Organization’s Return Actions List 148](#_Toc13053026)

[Appendix A - CINX Concepts 149](#_Toc13053027)

[Appendix B - CINX URLs 151](#_Toc13053028)

### Background - DONE

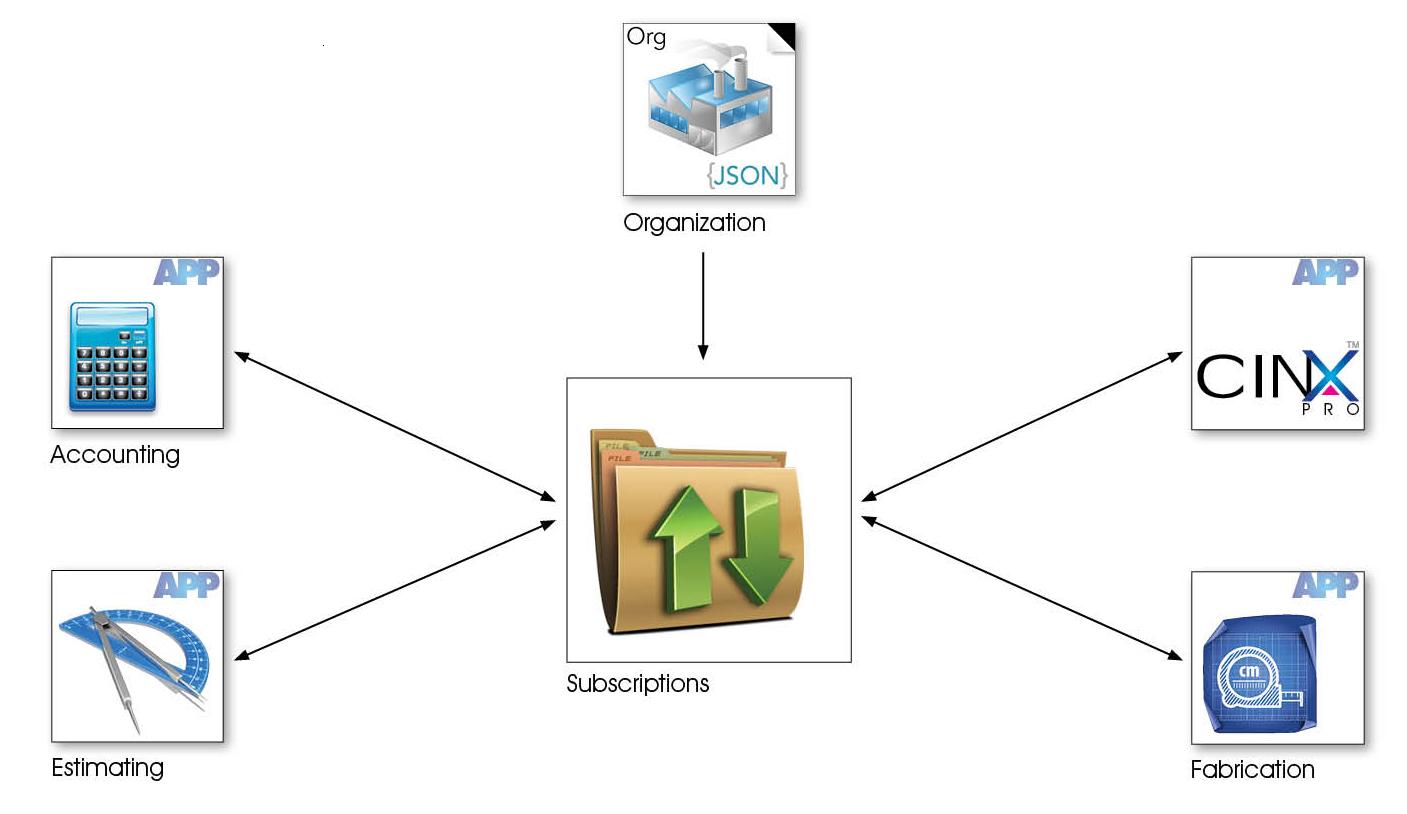
Harrison Publishing House (HPH) is a content and software provider to the construction industry in North America. We aggregate building product information, extend the data with our own content, and make it available to trade participants.

HPH has historically supported mechanical contractor estimating systems with price updates so that estimators could use current information when preparing bids.

With our new product CINX (Construction Information Network & Exchange) we are expanding our software offering for both contractors and wholesale distributors (vendors) from whom contractors purchase materials. For contractors CINX will provide a Master Data Management platform and transaction integration with their vendors. For wholesale distributors CINX will provide a standardized customer integration portal and limited CRM functionality.

### CINX Application/Subscription Model Introduction - DONE

A CINX customer (organization) has the opportunity to subscribe to different applications available on the platform. An application can be a grouping of pages and functionality on the CINX web site or external applications that consume content from CINX. Each application on the CINX platform has a unique identifier and a range of fields that define its attributes. When an organization subscribes to an application a new unique identifier is created to link the application to the subscribing organization. This application id will be used in API calls to identify and grant access to requested data operations.



### CINX Data Source Introduction - DONE

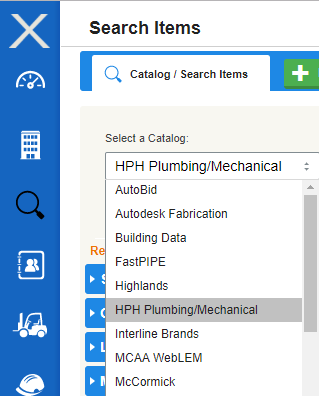
CINX supports different “catalogs” of content. Companies subscribing to CINX can have access to multiple catalogs or databases of content. Here are some common catalogs in CINX:

* HPH Plumbing/Mechanical catalog
* MCAA WebLEM
* PHCC Labor Units

Another catalog that each CINX company can use is a “Private” catalog. This catalog of content is specific to the subscribing company and can contain material items. These items can be copied from another catalog on CINX or manually entered/uploaded by the company.

Estimating systems, CAD/BIM software systems, and other software partners that utilize CINX content will also have a separate CINX Catalogs. This allows each system to have a specific set of items that CINX maintains.

A good way to visualize this concept is to open the Catalog drop-down in the CINX Search app. Here is a screen shot that shows a company that has access to many catalogs. Note: each company’s private catalog will be listed using the company’s name as registered in CINX.



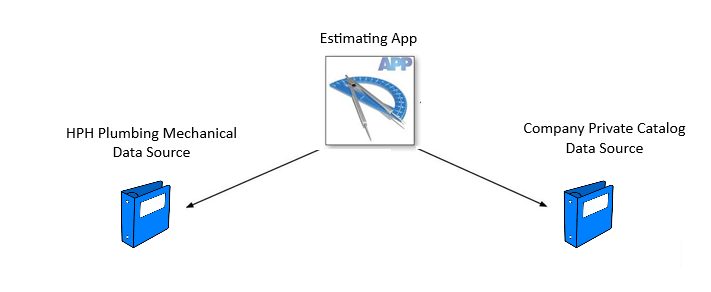
When working with the CINX API in conjunction with a company’s subscription a catalog is often named Data Source.

### CINX Application and Data Source Introduction - DONE

For software partner (external) applications that consume data from CINX there is an important attribute in the CINX subscription that defines which CINX catalog is used to acquire information for data outputs. This attribute is called a **data source**.

Each application subscription in CINX has this attribute.

Importantly, a company can subscribe to an application multiple times provided that the data source is different. For example, a company using an industry estimating system may have CINX produce updates for different locations using different CINX Catalogs (data sources). One location may want standard list prices from the standard estimating database and another location may want update files to contain net buy prices based on their private catalog of content.



**Please note that when working with the CINX Subscription API response each combination of an application and a data source will be listed in separate blocks**.

Below is a sample showing an App with its data source

App Name: Viewpoint Estimation

Data Source Name: Viewpoint Estimation



App Name: Viewpoint Estimation

Data Source Name: My Private Catalog



### ReST API - DONE

The CINX platform offers programmatic access to read and write data using ReST web services and standard HTTP verbs.

##### Authentication & Headers

HTTP Basic Authentication over HTTPS is used to control access to CINX platform resources.

With each API call, you’ll need to set request headers, including a CINX User’s User Name and Password.

##### Account Authorization

CINX accounts are subscription based and have expiration dates. Accounts with expired subscriptions will not have access to CINX data.

##### Production Endpoint

The production endpoint for the CINX API is:

**https://api.cinx.com/**

##### API Response Formatting – Standard/JSON

All CINX API responses use a standardized format. By default a JSON formatted response is provided. Please see below if you prefer XML.

The **“response”** **object** will contain meta data about the server’s response to the request.

The **“rows” array** will contain the query results.

{

"start\_time": 1411574224.8083,

"response": {

"status\_code": 200,

"message": "OK",

"record\_count": 1,

"server\_response\_time": 0,

"total\_elapsed\_time": 0.02492094039917,

"total\_count": 0

},

"rows": [

{

]

}

##### API Response Formatting – XML

To receive an XML response, please add the following parameter to the url:

**format=xml**

<cinx\_api>

<start\_time>1414436492.2267</start\_time>

<response>

<status\_code>200</status\_code>

<message>OK</message>

<record\_count>2</record\_count>

<server\_response\_time>0</server\_response\_time>

<total\_elapsed\_time>0.013047933578491</total\_elapsed\_time>

<total\_count>0</total\_count>

</response>

<rows />

<row>

</row>

</rows>

</cinx\_api>

##### Result Rows - Paging

For many HTTP GET requests you can use URL parameters to define the number of rows returned in the response. Valid input parameters are:

**start={number}**

**limit={number}**

### API – PING Server

This request can be used to test the availability of the CINX API.

URL Pattern:

**{api path}/ping**

URL Samples:

**http://api.cinx.com/ping**

**http://api.cinx.com/ping?format=xml**

Sample JSON Response



### API – GET User Subscriptions/Authorizations (Orgs and Applications)

This request will be used to get the applications/subscriptions that the user is authorized to access. The response will contain information required to make additional API calls.

URL Pattern:

**{api path}/sub/user/subscriptions**

URL Sample:

[**https://api.cinx.com/sub/user/subscriptions**](https://api.cinx.com/sub/user/subscriptions)



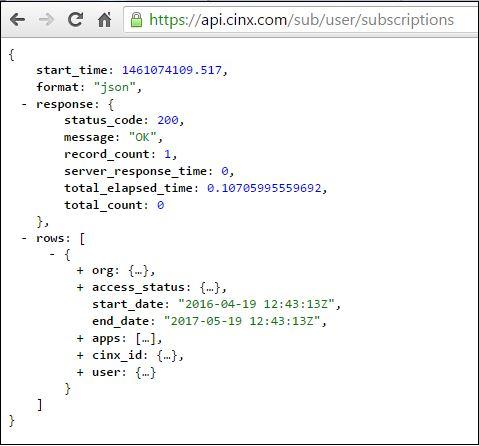
### Working with the GET User Subscriptions/Authorizations API Response

The Subscriptions/Authorizations API response will provide the necessary information to make subsequent API requests.

Please keep the following CINX Platform attributes in mind as you begin to work with the Subscriptions response:

* A User may be associated with more than one Org (multiple Orgs in the CINX subscription result)
* An Org can subscribe to multiple catalogs of content
* An Org might have multiple subscriptions to a single App (the data for the App coming from different sources) (multiple instances of the same App within a single Org in the CINX subscription result)
* A User may use a CINX Addin for multiple partner applications on the same computer
* Objects in the CINX Platform are assigned Unique Ids that will be required API URL elements

The JSON response uses the following structure:



Each **Row** array of the response will correspond to an Org to which the user is a member and will contain the following components:

**Org**: contains two objects labeled **Names** and **CINX Id**. The **CINX Id** will contain the Unique Id to identify the Org on the CINX Platform.

**Access Status**: defines the org’s subscription status which controls access to the API.

**Start Date**: lists the date the CINX subscription was established.

**End Date**: provides the scheduled expiration date for the subscription.

**Apps**: an array of applications that the org currently subscribes to and can access using the API. The structure of an app object is governed by its **Type**. Current CINX App types include **Data-Source** (catalogs of content), **Integration** (interactions with other software systems), and **Website** (CINX web site features).

**CINX Id**: contains the Unique Id for the org’s subscription.

**User**: provides basic information about the user and his/her CINX Unique Id.

The following sections contain important information about using the API and the CINX Unique Ids that will be required for making some API calls.

**Is the subscription active?**

You will first need to verify that the subscription is active. This is done by checking the **access\_status** object of the response.

If the subscription is active the response will have the following information:

"access\_status": {

"code": 200,

"message": "OK",

"reason": null

},

If the subscription has expired the response will have the following information:

"access\_status": {

"code": 401,

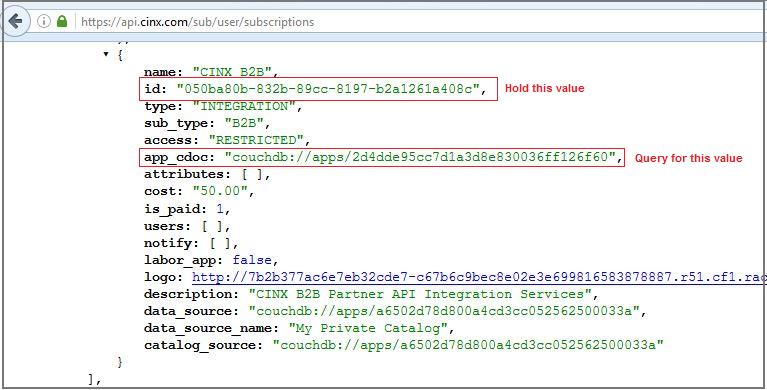
"message": Unauthorized",

"reason": "Subscription expired on [2014-08-03]"

},

**Does the subscription response contain a B2B (data integration) application?**

* A B2B CINX application allows programmatic access to content on the CINX Platform. To determine if an Org has a B2B app in their subscription, you will need to query the apps array for an **app\_cdoc** that has this value: **"couchdb://apps/2d4dde95cc7d1a3d8e830036ff126f60"** (which is a CINX Platform constant). Once this is located please hold the value of the **id:** field. The string shown in this field is required to make additional API calls to retrieve item and project information.



In the above screenshot, the necessary **id** value is:  **050ba80b-832b-89cc-8197-b2a1261a408c**

**Does the subscription response contain multiple B2B (data integration) applications?**

It is possible for an org to have multiple B2B applications. This is allowed to let CINX link to different data sources. A data source is a catalog of content. One org may have access to multiple content catalogs.

It is recommended that B2B application with an org’s Private Catalog data source be used when working with the public API for a specific org.

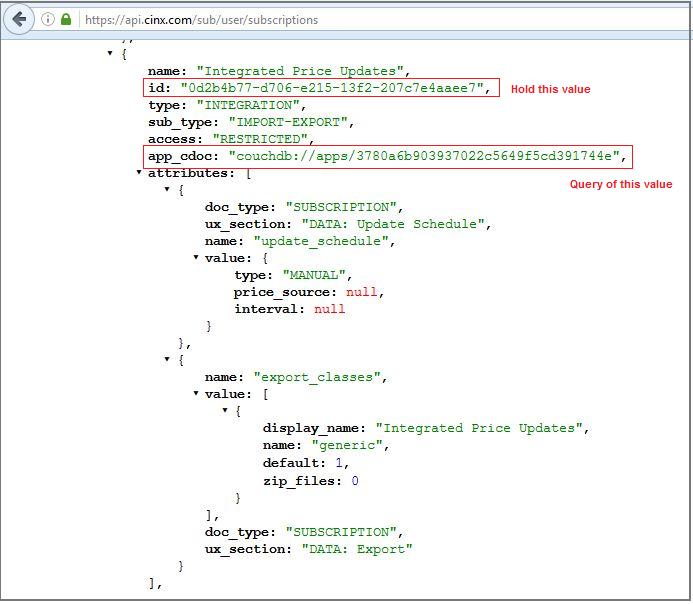
Similar to the B2B app cdoc, the My Private Catalog data source will have a static/constant that can be used when querying the API’s Subscription response. Inside a B2B app there will be a **data\_source** field. The value in this field for the My Private Catalog will be **couchdb://apps/a6502d78d800a4cd3cc052562500033a**. This value is a CINX Platform constant.



It is not recommended to use the **data\_source\_name** field as the name text may change in the future.

**Does the subscription response contain Integration (Import-Export) application?**

A CINX Import-Export application provides API access to formats required for integration with external applications. This will be used in the API calls to pull price updates for a Your company catalog.



The necessary **id** value in this screenshot is:  **0d2b4b77-d706-e215-13f2-207c7e4aaee7**

**What is the user’s CINX Unique Id?**

The CINX User Unique Id (GUID) will be required by some API requests. These requests will generally be ones in which new information is created or existing content is updated. The user will then be identified as the “last modified by”.

### API – GET Item Information using the “mcaa” Response View – Item GUID

This request will be used to get a catalog item’s information using the API’s “**mcaa**” view. You will need to include the item’s Private Catalog GUID as a parameter in the URL.

URL Pattern:

{api path}/sub/{B2B Subscription Id}/catalog/item/{Private Catalog Item GUID}?view=mcaa

URL Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/catalog/item/28468a6d-5858-1e50-af40-73d62f787a9a?view=mcaa>



### API – GET Item Information using the “mcaa” Response View – Org Part Number

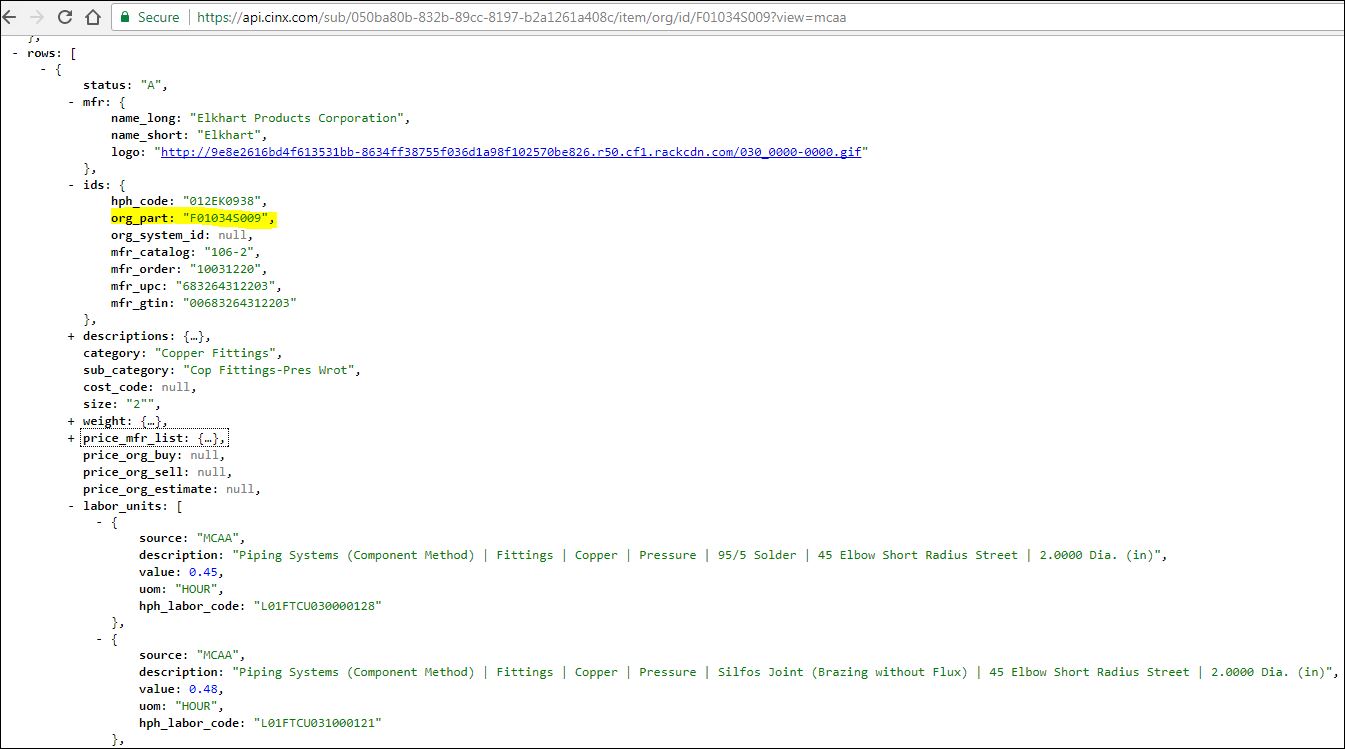
This request will be used to get a catalog item’s information using the API’s “**mcaa**” view. You will need to include the item’s Org Part Number as a parameter in the URL.

URL Pattern:

{api path}/sub/{B2B Subscription Id}/item/org/id/{Org Part Number}?view=mcaa

URL Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/item/org/id/F01034S009?view=mcaa>



### API – GET Item Information using the “mcaa” Response View – HPH Code

This request will be used to get a catalog item’s information using the API’s “**mcaa**” view. You will need to include the item’s HPH Code as a parameter in the URL.

Single Item

URL Pattern:

{api path}/sub/{B2B Subscription Id}/item/hphcode/{HPH Item Code}?view=mcaa

URL Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/item/hphcode/006wb0212?view=mcaa>



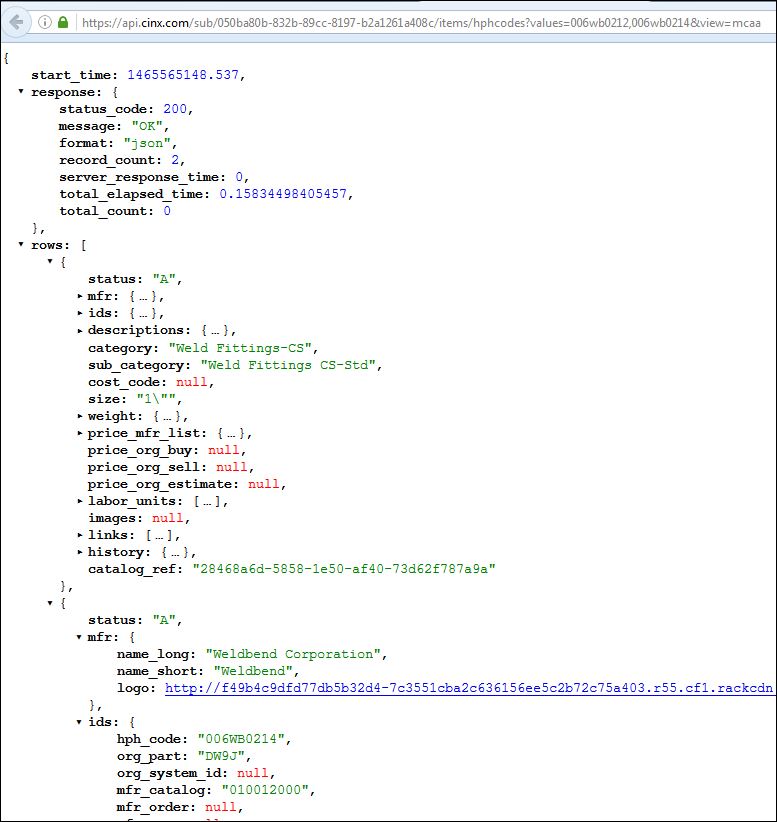
Multiple Items

URL Pattern:

{api path}/sub/{B2B Id GUID}/items/hphcodes?values={HPH Item Code(s);comma-separated if multiple}&view=mcaa

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/items/hphcodes?values=006wb0212,006wb0214&view=mcaa>



### “mcaa” View Response Contents

|  |  |
| --- | --- |
| **CINX API - Item Response Contents** | |
| **"mcaa" View** | |
| JSON Format | |
| **Response** | **Description** |
| status | Mfr product availability status. A=Available with a list price, P=Available calling for pricing, D=Discontinued/Obsolete |
| mfr -> name\_long | Manufacturer full name including reference to legal entity type |
| mfr -> name\_short | Manufacturer abbreviated name |
| mfr -> name\_logo | Link to the manufacturer's logo on the CINX CDN. |
| ids -> hph\_code | HPH's unique identifier for a product |
| ids -> org\_part | Customer defined part number |
| ids -> org\_system\_id | Customer defined auxiliary part identifier. |
| ids -> mfr\_catalog | Manufacturer catalog/part number |
| ids -> mfr\_order | Manufacturer order number |
| ids -> mfr\_upc | Manufacturer UPC (12 digit - when defined by the mfr) |
| ids -> mfr\_gtin | Manufacturer GTIN (14 digit - when defined by the mfr) |
| descriptions -> hph\_long | HPH description excluding the item's size |
| descriptions -> hph\_short | HPH "invoice" description that includes the item's size |
| descriptions -> org | Customer defined description |
| descriptions -> mfr | Manufacturer defined description |
| category | HPH top-level category value |
| sub\_category | HPH secondary-level categorization value |
| cost\_code | Customer defined default cost code |
| size | HPH-defined product size |
| weight -> value | Product's weight |
| weight -> uom | Weight unit-of-measure |
| price\_mfr\_list -> value | Manufacturer's list price |
| price\_mfr\_list -> uom | List price unit-of-measure |
| price\_mfr\_list -> effective | Manufacturer price sheet effective date |
| price\_mfr\_list -> price\_sheet\_number | Manufacturer price sheet number |
| price\_mfr\_list -> price\_sheet\_name | Manufacturer price sheet name |
| price\_org\_buy | Customer defined buy/purchase price |
| price\_org\_sell | Customer defined sell/change-order price |
| price\_org\_estimate | Customer defined estimate price |
| labor\_units **<array structure>** |  |
| labor\_units -> source | MCAA or PHCC |
| labor\_units -> description | Labor unit description |
| labor\_units -> value | Labor unit value |
| labor\_units -> uom | Labor unit unit-of-measure |
| labor\_units -> hph\_labor\_code | HPH defined unique identifier for the labor unit |
| images **<array structure>** |  |
| images -> url | Product picture url on the CINX CDN |
| images -> mime\_type | Mime-type for the image |
| links **<array structure>** |  |
| links -> url | Manufacturer documentation url on the CINX CDN |
| links -> type | Documentation type (SDS, Spec, etc.) |
| links -> mime\_type | Mime-type for the file |
| links -> date | Date the url was created |
| history -> created | Date the item was added to the customer's CINX catalog |
| history -> creator | User who added the item |
| history -> modified | Date of the last modification to the catalog item |
| history -> modifier | User who last modified the item |
| catalog\_ref | Private catalog GUID |

### API – GET Item Information using the “full” Response View – Item GUID

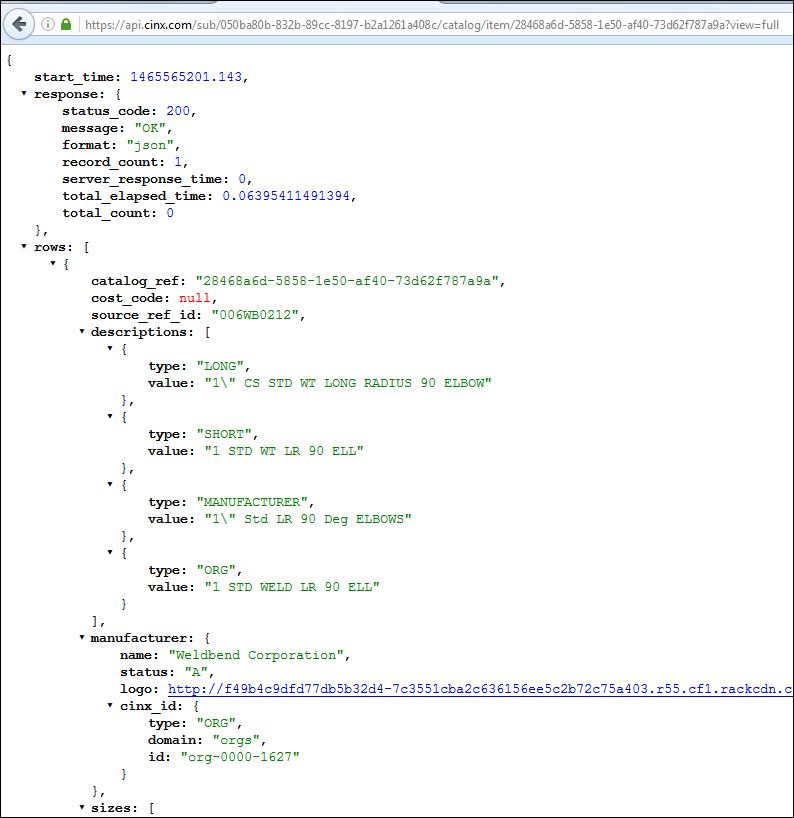
This request will be used to get an item’s information using the API’s **“full”** view and its’ CINX Private Catalog Item GUID.

URL Pattern:

{api path}/sub/{B2B Subscription Id}/catalog/item/{Private Catalog Item GUID}?view=full

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/catalog/item/28468a6d-5858-1e50-af40-73d62f787a9a?view=full>



### API – GET Item Information using the “full” Response View – Org Part Number

This request will be used to get a catalog item’s information using the API’s “**full**” view. You will need to include the item’s Org Part Number as a parameter in the URL.

URL Pattern:

{api path}/sub/{B2B Subscription Id}/catalog/item/org/id/{Org Part Number}?view=full

URL Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/item/org/id/F01034S009?view=full>



### API – GET Item Information using the “full” Response View – HPH Code

This request will be used to get an item’s information using the API’s **“full”** view and its HPH Code.

URL Pattern:

{api path}/sub/{B2B Id GUID}/items/hphcodes?values={HPH Item Code(s);comma-separated if multiple}&view=full

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/item/hphcode/006wb0212?view=full>

****

### API – GET Item Information using the “custom” Response View – Item GUID

This request will can be used to submit a Private Catalog Item GUID with the response fields defined by the requesting URL.

URL Pattern:

{api path}/sub/{B2B Id GUID}/catalog/item/{Private Catalog Item GUID}?view=custom&fields={comma separated list of the fields}

Please see the ***CINX Private Catalog Item Data Options – Custom View*** section of this document for a listing of the available response fields.

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/catalog/item/28468a6d-5858-1e50-af40-73d62f787a9a?view=custom&fields=image,desc_long,hphcode,mfr_name,price_current,upc,weight,price_buy,price_sell,price_estimate,status,desc_org,desc_short,desc_mfr,partno_mfr,partno_org,system_id,specsheet,hier,xref,price_list_effective>



### API – GET Item Information using the “custom” Response View – HPH Code

This request will can be used to submit a list of HPH Codes with the response fields defined by the requesting URL.

URL Pattern:

{api path}/sub/{B2B Id GUID}/items/hphcodes?values={HPH Item Code(s);comma-separated if multiple}&view=custom&fields={comma separated list of the fields}

Please see the ***CINX Private Catalog Item Data Options – Custom View*** section of this document for a listing of the available response fields.

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/items/hphcodes?values=006WB0212&view=custom&fields=image,desc_long,hphcode,mfr_name,price_current,upc,weight,price_buy,price_sell,price_estimate,status,desc_org,desc_short,desc_mfr,partno_mfr,partno_org,system_id,specsheet,hier,xref,price_list_effective>



### CINX Private Catalog Item Data Options – Custom View

The list shown below is the available data options for constructing a Custom View URL.



|  |
| --- |
| **NOTES:** |
| API Request should have the field names comma-delimited without any spaces. |
| The response fields will be in the ordered that they were requested. |
| Fields without values in the contractor's catalog will be returned as **null**. |
| Requests for a field not yet supported will be returned with the requested field name and a value of “**Unsupported Field"**. |

### API – GET Item List Price History – HPH Code

This request can be used to obtain the Manufacturer List Price history of an item.

URL Pattern:

{api path}/sub/{B2B Subscription Id}/catalog/item/hphcode/{HPH Item Code}/price/history

URL Sample:

<https://api.cinx.com/sub/14b8fb3e-f62e-cd1c-b71f-fd5b1936d1b2/item/hphcode/012NI0018/price/history>



### API – Add a New Catalog Item – HPH Code

This request will be used to submit a HPH Item Code for an item and have it added to an Org’s private catalog.

URL Pattern:

{api path}/sub/{B2B Subscription Id}/catalog/add/item?hphcode={HPH Item Code}

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/catalog/add/item?hphcode=005GR0006>

A successful request will have a response similar to below:



The GUID shown in the **id** field is the item’s CINX Private Catalog GUID.

### API – Create a non-HPH Private Catalog Item

This request will be used to create a Private Catalog Item that is not linked to an HPH Item. Please note that this URL will require a valid CINX user GUID. This will be used to indicate who created the item.

URL Pattern:

{api path}/sub/{B2B Id GUID}/catalog/add/non-catalog/item? values={JSON Structure}

The **values** section of this URL must contain properly formatted JSON. If the structure is an array, all elements of the array must be submitted. Please see the ***CINX Private Catalog Item JSON Structures*** section of this document for a listing of the available JSON structures.

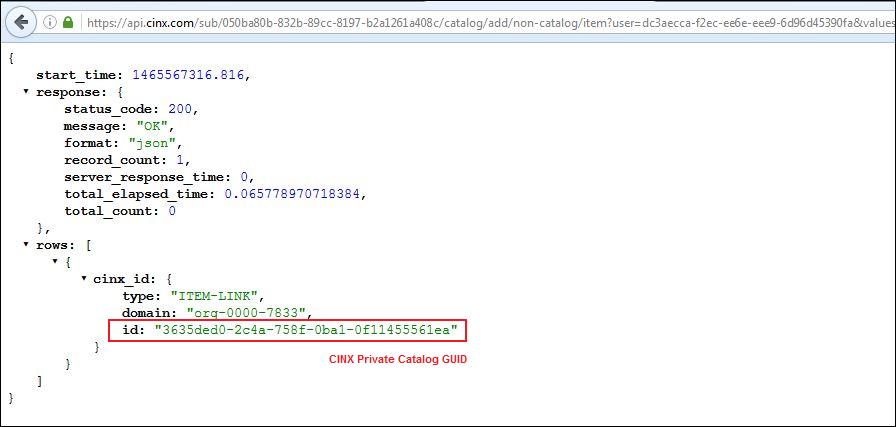
Basic Sample:

https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/catalog/add/non-catalog/item? values={"mfr\_name":"Rubbermaid","descriptions":[{"type":"LONG","value":"30GAL Black Trash Can"},{"type":"SHORT","value":"Trash Can"}],"pkg\_ids":[{"type":"MFR CATALOG","value":"101"}]}

Expanded Sample:

https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/catalog/add/non-catalog/item?user=dc3aecca-f2ec-ee6e-eee9-6d96d45390fa&values={%22mfr\_name%22:%22Rubbermaid%22,%22descriptions%22:[{%22type%22:%22LONG%22,%22value%22:%2230GAL%20Black%20Trash%20Can%22},{%22type%22:%22SHORT%22,%22value%22:%22Trash%20Can%22}],%22pkg\_ids%22:[{%22type%22:%22MFR%20CATALOG%22,%22value%22:%22101%22},{%22type%22:%22UPC%22,%22value%22:%22464645656410%22}],%22size%22:%20{%22type%22:%22PHYSICAL%22,%22value%22:%2230%22,%22uom%22:%22GALLON%22},%22prices%22:[{%22type%22:%22BUY%22,%22cdoc%22:%22%22,%22value%22:%2210.256%22,%22price\_uom%22:%22E%22,%22date\_updated%22:%22%22,%22user\_cdoc%22:%22dc3aecca-f2ec-ee6e-eee9-6d96d45390fa%22,%22source\_method%22:%22USER%22},{%22type%22:%22ESTIMATED%22,%22cdoc%22:%22%22,%22value%22:%2211.256%22,%22price\_uom%22:%22E%22,%22date\_updated%22:%22%22,%22user\_cdoc%22:%22dc3aecca-f2ec-ee6e-eee9-6d96d45390fa%22,%22source\_method%22:%22USER%22}],%22weight%22:%222.300%22,%22weight\_uom%22:%22LBS%22,%22depth%22:%2223.00%22,%22height%22:%2238.00%22,%22width%22:%2223.00%22,%22cube%22:%22123%22,%22dimension\_uom%22:%22INCHES%22,%22attributes%22:[{%22name%22:%22Color%22,%22value%22:%22Black%22},{%22name%22:%22Body%20Style%22,%22value%22:%22Tapered%22}],%22cost\_code%22:%22123456%22,%22pkg\_status%22:%22A%22}

A successful response will be similar to this:



The GUID shown in the **id** field is the item’s CINX Private Catalog GUID.

### API – Modify a CINX Private Catalog Item – using a CINX GUID

This request will be used to modify a Private Catalog Item using its CINX Item GUID.

URL Pattern:

{api path}/sub/{B2B Id GUID}/catalog/item/{CINX ITEM GUID}/modify? values={JSON Structure}

The **values** section of this URL must contain properly formatted JSON. If the structure is an array, all elements of the array must be submitted. Please see the ***CINX Private Catalog Item JSON Structures*** section of this document for a listing of the available JSON structures.

Sample:

[https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/catalog/item/3635ded0-2c4a-758f-0ba1-0f11455561ea/modify? values={"size":{"type":"PHYSICAL","value":"40","uom":"GALLON"}}](https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/catalog/item/3635ded0-2c4a-758f-0ba1-0f11455561ea/modify?%20values=%7b%22size%22:%7b%22type%22:%22PHYSICAL%22,%22value%22:%2240%22,%22uom%22:%22GALLON%22%7d%7d)



### CINX Private Catalog Item JSON Structures

The JSON structures shown below can be used in API calls to **Create** or **Modify** CINX Private Catalog Items.

If the structure is an array, all elements of the array must be submitted.

**Manufacturer Name (required)**

“mfr\_name”:”name”

**Item Identifiers** (array)

“pkg\_ids”:[{”type”:”type name”,”value”: “value”}]

Available “type names” are: **MFR CATALOG**, **MFR ORDER**, **UPC**, **GTIN**, **ORG ID**, **ORG SYSTEM ID**

**Descriptions** (array)

“descriptions”:[{”type”:”type name”,”value”: “value”}]

Available “type names” are: **LONG**, **SHORT**, **ORG**

**Size**

“size”:{”type”:”type name”,”value”: “value”,”uom”:”value”}

**Prices** (array)

“prices”:[{”type”:”type name”,”cdoc”:*null*,”value”: “value” ,”price\_uom”: “price\_uom value”,”date\_updated”:””,”user\_cdoc”:”{user guid}”,”source\_method”:”USER”}]

Available “type names” are: **BUY**, **SELL**, **ESTIMATED**

“cdoc”: please submit a ***null***

Available “price uoms” are: **E**, **T**, **C**, **M**

“date\_updated”: datetime format (**yyyy-mm-dd hh:mm:ssZ**)

“source\_method”: please submit “**USER**”

**Price Multiplier**

“multiplier”:{“cdoc”:***null***,”code”:”code value”,”name”: “name”}]

**Attributes** (array)

“attributes”:[{”name”:”name”,”value”: “value”}]

**Physical Properties**

“weight”:”value”

“weight\_uom”: “value”

“depth”:”value”

“height”:”value”

“width”:”value”

“cube”:”value”

“dimension\_uom”:”value”

**Default Cost Code**

“cost\_code\_value”:”value”

**Product Availability Status**

“pkg\_status”:”value”

Available “values” are: **A**,**D**,**P**

### API – DELETE a Catalog Item (based on an CINX Item GUID)

This request will be used to delete an item from an Org’s Private CINX catalog. Please note that this API call requires the item’s CINX Private Catalog Item GUID.

URL Pattern:

{api path}/sub/{B2B Id GUID}/catalog/item/{CINX ITEM GUID}/delete

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/catalog/item/3635ded0-2c4a-758f-0ba1-0f11455561ea/delete>



### API – GET Private Catalog Lines – Root Level

This request will be used to retrieve the Root level Private Catalog lines for an Org’s private catalog.

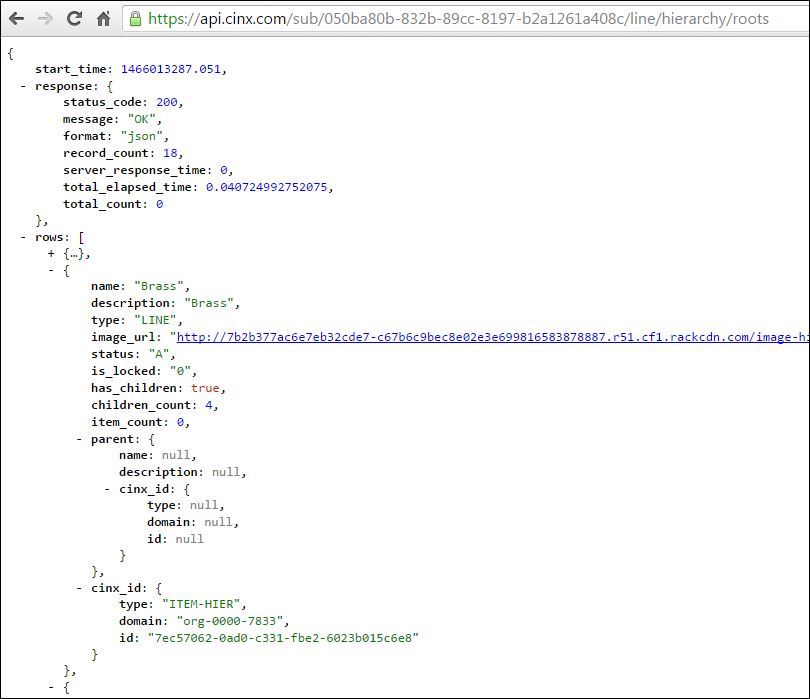
URL Pattern:

{api path}/sub/{B2B Id GUID}/line/hierarchy/roots

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/line/hierarchy/roots>

A successful request will have a response similar to below:



### API – GET a Private Catalog Line’s Children (sub-lines)

This request will be used to retrieve the children (sub-lines) of a parent line within an Org’s Private Catalog.

URL Pattern:

{api path}/sub/{B2B Id GUID}/line/hierarchy/{parent hierarchy GUID}/children

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/line/hierarchy/7ec57062-0ad0-c331-fbe2-6023b015c6e8/children>

A successful request will have a response similar to below:



### CINX Catalog Update Introduction

CINX allows a company to define their own catalog of material content. This Private Catalog can contain items from the Harrison Plumbing/Mechanical database, vendor catalogs that are hosted on CINX, and items created and maintained by the company. In addition, a company can have access to CINX software partner catalogs. For example, databases used by industry estimating systems are available as CINX catalogs.

As part of CINX’s integration services, a company can request catalog updates formatted for industry software or develop a customized format. The company can have unlimited subscriptions to CINX Import-Export Applications – Catalog updates.

Each subscription will define the catalog of items and the output format. This subscription has a CINX GUID that will be required to create API requests. This GUID can be obtained from the CINX Subscriptions/Authorizations response.

**Update Format**

As noted above, CINX can produce update files according to the specifications of leading industry software systems. Custom formats are also supported.

In addition to specifying the catalog and format, a subscription has additional settings that are used to manage the update process. Examples include a schedule for when updates are produced, who is notified when updates are ready, and what prices types (sources) to include in the updates.

**Schedule**

CINX has the ability to define when catalog updates are processed for subscriptions based on a Private CINX catalog. Schedule options include:

* Nightly
* Weekly (select the day)
* Monthly (select the 1st, 15th, of 28th of the month)
* Quarterly

**Notifications**

When a catalog update is posted, CINX checks the subscription settings and notifies the defined users. Notification types include:

* Email
* CINX System Alert

**Price Types/Sources**

CINX has the ability to include four price types in a catalog update file.

* Manufacturer List
* Buy/Purchase (taken from the company’s private catalog)
* Sell/Invoice (taken from the company’s private catalog)
* Estimate (taken from the company’s private catalog)

**Catalog Update File Types**

Each CINX catalog update file has field that defines its **Type**. Catalog update types are:

* **Update** – file contains any items that have had changes since the previous update was published.
* **Full** – file contains ALL items in the catalog.
* **Custom** – file contains user-defined contents.

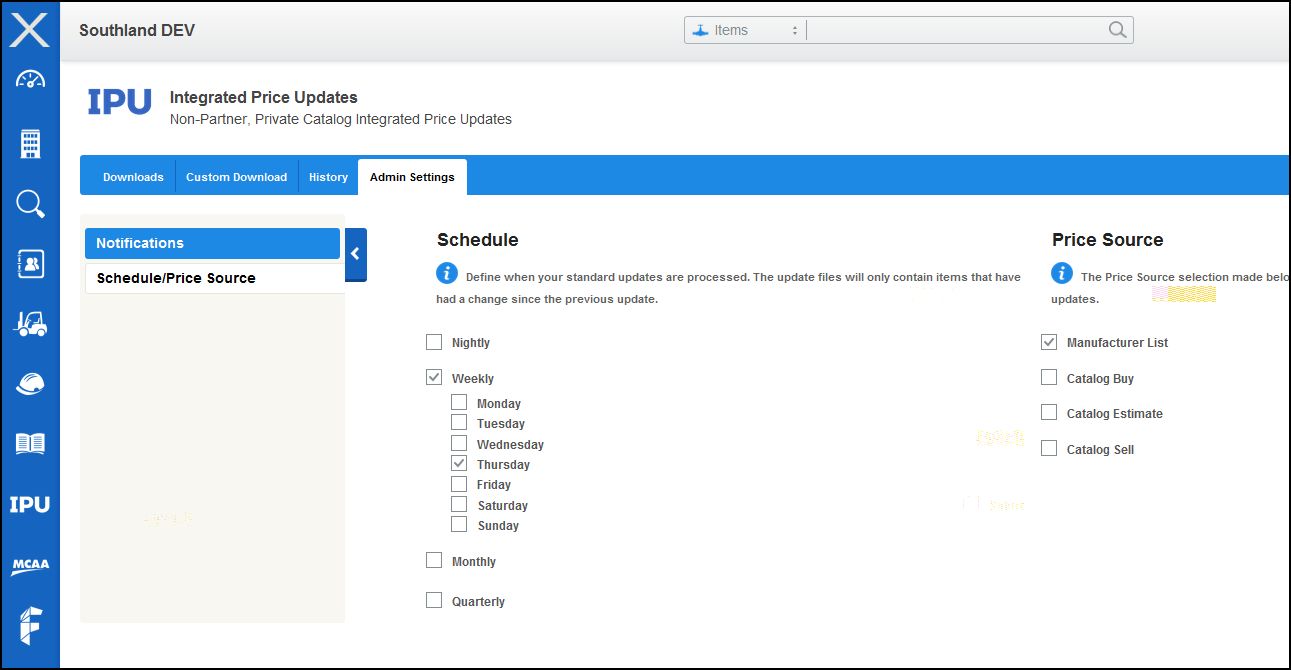
**Catalog Update File Status Filter Types**

Each CINX catalog update file has field that defines its **Status**. When an API call is made to retrieve Catalog Updates a Status filter can be submitted. Catalog update status filters are:

* **New** – files that have not been download/applied.
* **Applied** – files that have been applied to the integrated system. (Requires submission of an API call to mark the file as **Applied**.)
* **All** – returns all update files for the requested subscription.

**CINX UI – Catalog Updates**

The CINX UI provides extensive capabilities to manage the catalog updates. Admin settings can be defined that will allow for the creation of update files that can be accessed via the API. Custom update files can be requested. Updates can be downloaded and a history of activity can be viewed.



(Note: many of the API URLs will contain an IPU parameter. IPU is an acronym for Integrated Product/Price Updates)

### API – GET Catalog Updates List

This request will be used to retrieve catalog updates for a given subscription.

URL Pattern:

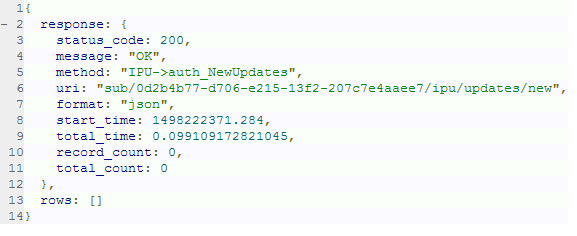
{api path}/sub/{app-id-guid}/ipu/updates/{update status type: **new**, **all**, or **applied**}

Sample:

<https://api.cinx.com/sub/0d2b4b77-d706-e215-13f2-207c7e4aaee7/ipu/updates/all>



If the url uses the **new** update status parameter and there are no new updates, the api response will have an empty **rows []** array.



### API – GET Catalog Update File - Download

This request will be used to download a catalog update file.

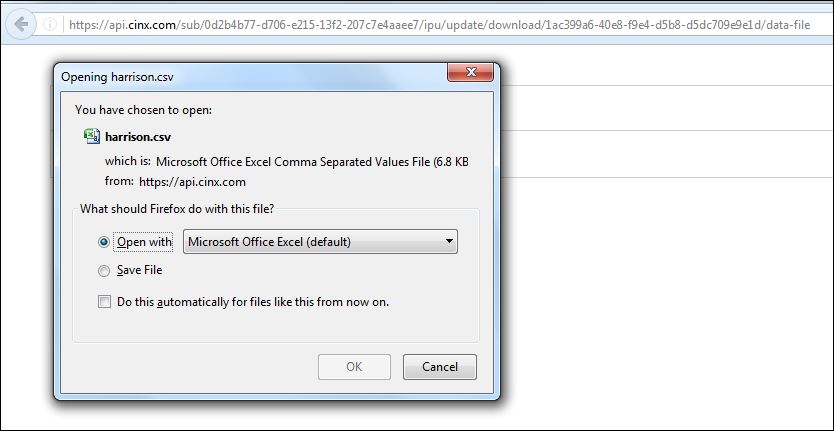
URL Pattern:

{api path}/sub/{app-id-guid}/ipu/update/download/{update id}/data-file

Sample:

<https://api.cinx.com/sub/0d2b4b77-d706-e215-13f2-207c7e4aaee7/ipu/update/download/1ac399a6-40e8-f9e4-d5b8-d5dc709e9e1d/data-file>

This will stream back a data file containing the update



### API – GET Catalog Update Bulletin File ONLY - Download

This will stream back the bulletin file in the requested format.

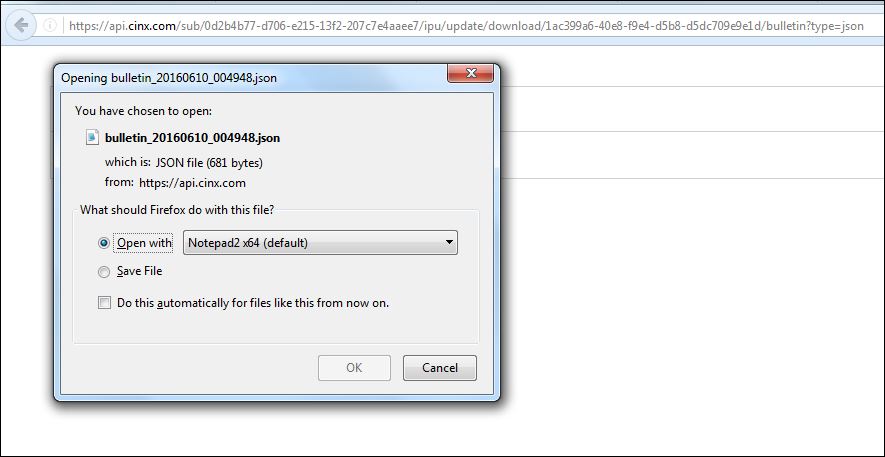
URL Pattern:

{api path}/sub/{app-id-guid}/ipu/update/download/{update id}/bulletin?type={**json** or **xls**}

Samples:

<https://api.cinx.com/sub/0d2b4b77-d706-e215-13f2-207c7e4aaee7/ipu/update/download/1ac399a6-40e8-f9e4-d5b8-d5dc709e9e1d/bulletin?type=json>

<https://api.cinx.com/sub/0d2b4b77-d706-e215-13f2-207c7e4aaee7/ipu/update/download/1ac399a6-40e8-f9e4-d5b8-d5dc709e9e1d/bulletin?type=xls>



### API – Mark Catalog Update File As Applied

This will mark the update as **Applied** and it will no longer show up in the **New** update list.

URL Pattern:

{api path}/sub/{app-id-guid}/ipu/update/apply/{update id}?values={applied\_date:yyyy-mm-dd,download\_date:yyyy-mm-dd}

The update id field is taken from the Get Update response in the following location: row -> cinx\_id -> id

Sample:

[https://api.cinx.com/sub/0d2b4b77-d706-e215-13f2-207c7e4aaee7/ipu/update/apply/1ac399a6-40e8-f9e4-d5b8-d5dc709e9e1d?values={%22applied\_date%22:%222016-06-10%22,%22download\_date%22:%222016-06-10%22}](https://api.cinx.com/sub/0d2b4b77-d706-e215-13f2-207c7e4aaee7/ipu/update/apply/1ac399a6-40e8-f9e4-d5b8-d5dc709e9e1d?values=%7b%22applied_date%22:%222016-06-10%22,%22download_date%22:%222016-06-10%22%7d)



### API – GET Catalog Item Count

This request is used to obtain the count of Catalog Items contained within an Org’s Private CINX catalog.

URL Pattern:

{api path}/sub/{B2B Subscription Id}/catalog/items/ids?count=1

Sample:

<http://api.cinx.com/sub/14b8fb3e-f62e-cd1c-b71f-fd5b1936d1b2/catalog/items/ids?count=1>

The **total\_count** parameter of the response object details the total items in the catalog.



### API – GET Catalog Item Ids

This request is used to obtain the list of Catalog Item Ids contained within an Org’s Private CINX catalog.

Note: If the catalog has a large number of items then the paging elements (**offset** and **limit**) should be used in conjunction with the URL.

URL Pattern:

{api path}/sub/{B2B Subscription Id}/catalog/items/ids

Sample:

<http://api.cinx.com/sub/14b8fb3e-f62e-cd1c-b71f-fd5b1936d1b2/catalog/items/ids?offest=25&limit=100>

The response will contain three Ids for each item. These Ids can be used in subsequent API calls to retrieve other product information. The Ids are:

cinx\_id: This value is the item’s Private Catalog GUID

hphcode: HPH Item Code

org\_id: Org-defined product identifier



### Manufacturer Price Sheet Introduction

A Price Sheet is a document that is issued by a product manufacturer or service provider that define the prices that will be charged to consumers for procuring the listed materials or services. Manufacturers with large product offerings often have multiple price sheets with each containing only a segment of the manufacturer’s overall product listing. For example, Charlotte has different price sheets for their Cast Iron No Hub, Service Weight, and Extra Heavy products.

Due to rising raw material, labor, transportation and many other factors, manufacturers frequently release new instances of price sheets with revised pricing levels.

Every item price in the Harrison Plumbing/Mechanical catalog on CINX is linked to Manufacturer Price Sheet from which it was obtained. The CINX API provides several endpoints for working with Manufacturer Price Sheets.

To better understand the API, please review the following terms/concepts:

**Price Sheet Hierarchy**: In CINX there is a hierarchal structure that is used to manage the manufacturers’ price sheet instances. The three levels of the hierarchy are: Manufacturer, Price Sheet Master, and Price Sheet Detail.

**Manufacturer**: This is the organization that creates and releases the price sheets. Each manufacturer will have a unique CINX identifier that will be used in API requests.

Example Manufacturer: Charlotte

**Price Sheet Master**: This defines a type of price sheet issued by the manufacturer. Each Price Sheet Master will have a unique CINX identifier that will be used in API requests.

Example Price Sheet Master: Charlotte No Hub

**Price Sheet Detail**: This is a specific instance of a price sheet type. Each Price Sheet Detail will have a unique CINX identifier that will be used in API requests.

Example Price Sheet Detail: Charlotte No Hub – January 1, 2018

**Effective Date**: This is the date that the prices on the Price Sheet are scheduled to be implemented. This date is set by the manufacturer. Each Price Sheet Detail will have an Effective Date value.

### API – GET Manufacturer’s Price Sheet Masters

This request will be used to get a list of the manufacturer’s price sheet masters.

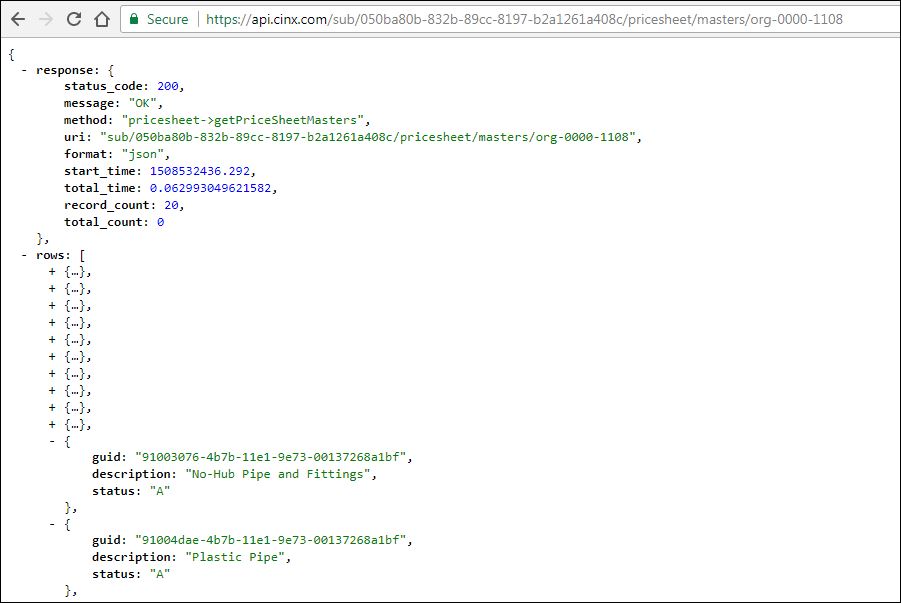
URL Pattern:

{api path}/sub/{B2B Id GUID}/pricesheet/masters/{org id}

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/pricesheet/masters/org-0000-1108>

The results will be sorted alphabetically by **description** field values.



### API – GET Manufacturer’s Price Sheet Master Details

This request will be used to get a list of the price sheet details for a given manufacturer price sheet master.

URL Pattern:

{api path}/sub/{B2B Id GUID}/pricesheet/master/{org id}/{price sheet master id}/details

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/pricesheet/masters/org-0000-1108>

The results will be sorted descending on the **date\_effective** field values.

To make subsequent API calls using a price sheet detail, the **id** field within the **cinx\_doc** object will be required.



### API – GET Project List

This request will be used to get a list of the org’s Projects from CINX.

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms?type=pml

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/boms?type=pml>



### API – GET Project Details

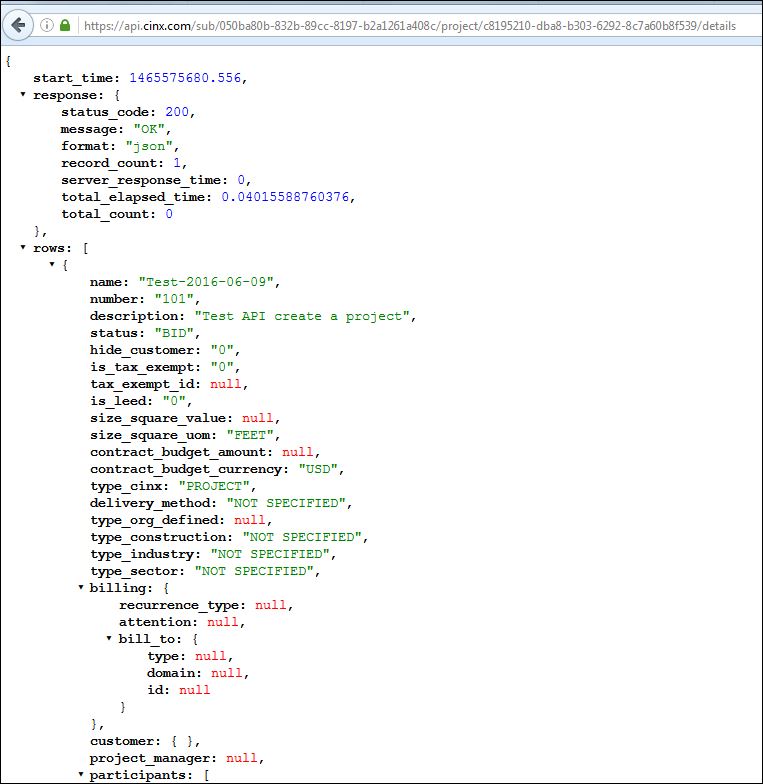
This request will be used to get the details of an org’s project from CINX.

URL Pattern:

{api path}/sub/{B2B Id GUID}/project/{Project GUID}/details

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/project/c8195210-dba8-b303-6292-8c7a60b8f539/details>



### API – PUT New Project

This request will be used to create a new project in the org’s CINX account.

URL Pattern:

{api path}/sub/{B2B Id GUID}/project/create?values={JSON Structure}

The **values** section of this URL must contain properly formatted JSON. If the structure is an array, all elements of the array must be submitted. Please see the ***CINX Project JSON Structures*** section of this document for a listing of the available JSON structures.

Sample:

https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/project/create? values={"name":"Test-2016-06-09","description":"Test API create a project","number":"101"}



### API – Modify a CINX Project

This request will be used to modify a project in the org’s CINX account.

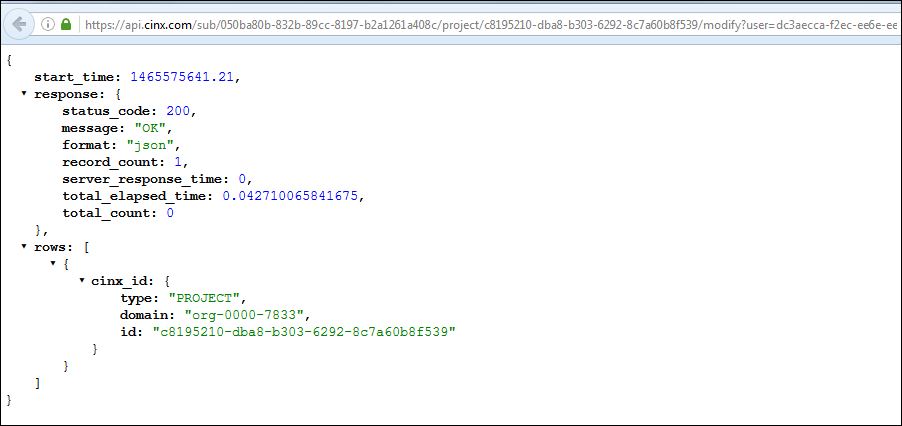
URL Pattern:

{api path}/sub/{B2B Id GUID}/project/{Project GUID}/modify?values={JSON Structure}

The **values** section of this URL must contain properly formatted JSON. If the structure is an array, all elements of the array must be submitted. Please see the ***CINX Project JSON Structures*** section of this document for a listing of the available JSON structures.

Sample:

[https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/project/c8195210-dba8-b303-6292-8c7a60b8f539/modify?values={"status":"Bid"}](https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/project/c8195210-dba8-b303-6292-8c7a60b8f539/modify?values=%7b%22status%22:%22Bid%22%7d)



### CINX Project JSON Structures

The JSON structures shown below can be used in API calls to **Create** or **Modify** a Project.

If the structure is an array, all elements of the array must be submitted.

**Project Name (required)**

“name”:”value”

**Project Number**

“number”:”value”

**Project Description**

“description”:”value”

**Project Status**

“status”:”value”

Available options are: **CONCEPTUAL**, **DESIGN**, **BID**, **CONSTRUCTION**, **CLOSEOUT**, **COMPLETED**, **NOT SPECIFIED**

Default value: **NOT SPECIFIED**

**Is Tax Exempt**

“is\_tax\_exempt”:” {Boolean value}”

**Project Type - Construction**

“type\_construction”:”value”

Available options are: **NEW**, **REMODEL**, **REPAIR**, **NOT SPECIFIED**

Default value: **NOT SPECIFIED**

**Project Type - Industry**

“type\_industry”:”value”

Available options are: **RESIDENTIAL**, **LIGHT COMMERCIAL**, **COMMERCIAL**, **INDUSTRIAL**, **UTILITY**, **HOSPITAL & HEALTHCARE**, **BIO/PHARM**, **PULP & PAPER**, **BREWERY**, **HEAVY/CIVIL**, **NOT SPECIFIED**

Default value: **NOT SPECIFIED**

**Project Type – Org Defined** (each org can define a set list of project types to be used to classify their projects)

“type\_org\_defined”:”value”

**Project Delivery Method**

“delivery\_method”:”value”

Available options are: **DESIGN-BUILD**, **DESIGN-BID-BUILD**, **INTEGRATED PROJECT DELIVERY**, **CM-AGENCY**, **CM-AT RISK**, **CM-GC**, **GUARANTEED MAXIMUM PRICE**, **DESIGN-BID-BUILD-OPERATE-MAINTAIN**, **BUILD-OPERATE-TRANSFER**, **NOT SPECIFIED**

Default value: **NOT SPECIFIED**

**Contract Budget Amount**

“contract\_budget\_amount”:”value”

**Contract Budget Amount Currency**

“contract\_budget\_amount”:”value”

Default value: **USD**

**Project Size - Square**

“size\_square\_value”:”value”

**Project Size – Square Unit-of-Measure**

“size\_square\_uom”:”value”

Available options are: **FEET**, **METERS**

Default value: **FEET**

**Is a LEED Project**

“is\_leed”:” {Boolean value}”

**Customer Project Number**

“customer\_project\_number”:”value”

**Dates** (array)

“dates”:[{”type”:”type name”,”value”: “value”}]

Available “date types” are: **SCHEDULED START**, **ACTUAL START**, **SCHEDULED COMPLETE**, **ACTUAL COMPLETE**, **BID DUE**, **CONTRACT AWARD, {user defined}**

Additional content types are available; please contact HPH if your system requires additional project content.

### API – Delete a CINX Project

This request will be used to delete a project in the org’s CINX account.

URL Pattern:

{api path}/sub/{B2B Id GUID}/project/{Project GUID}/delete

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/project/c8195210-dba8-b303-6292-8c7a60b8f539/delete>



### CINX Project Job Cost Introduction

To allow for the proper reporting of Project items and integration of transactions with contractor accounting systems, CINX has the ability for a company to define their job costing attributes or each Project.

CINX supports the following Job Cost information at the project level:

**PHASES\***

A Phase is construction management concept that allows a company to break-down a project into a smaller group of tasks commonly used for scheduling, sequencing, and reporting purposes. A Phase is also referred to as a Sub-Job, Area, or Extra by other software systems. Examples Phases include, Site Work, Floor, and System.

**MATERIAL COST CODES\***

A Cost Code is an accounting and construction management concept that allows a company to track materials for cost reporting purposes. Other systems refer to Cost Codes as Item Groups. Example Cost Codes are Sanitary Sewer, Domestic Water, and Gas.

**CATEGORIES\***

A Category is an accounting concept that allows a company to classify project costs for expensing purposes. Categories are also known as Cost Classes. For example, Material, Labor, Sub-Contractor, and Rental.

**ESTIMATED VALUES**

CINX supports the entry of estimated values for a project’s phase/cost code for the following data types: Material Costs, Labor Units, and Labor Costs.

**BUDGET VALUES**

CINX supports the entry of budget values for a project’s phase/cost code for the following data types: Material Costs, Labor Units, and Labor Costs.

**ACTUAL VALUES**

CINX supports the entry of actual values for a project’s phase/cost code for the following data types: Material Costs, Labor Units, and Labor Costs.

**DATES**

CINX supports the entry of the following key date types related to the project’s phase/cost code: Scheduled Start, Actual Start, Scheduled Completion, and Actual Completion.

**CONSTRUCTION STATUS**

The following statuses can be used to track a project’s phase/cost code: Not Installed, In-Process, Installed, Tested, Commissioned, and Operational.

**PERCENT COMPLETE**

Each CINX project phase/cost code has a field to track the percent complete.

\*Phases, Cost Codes, and Categories must be established in the company’s CINX account before they can be attached to a project. Please see these documentation sections for information about creating job cost information in CINX:

***API – Create a Job Cost Phase***

***API – Create a Job Cost Cost Code***

***API – Create a Job Cost Category***

### API – GET a Project’s Phases and Cost Codes List

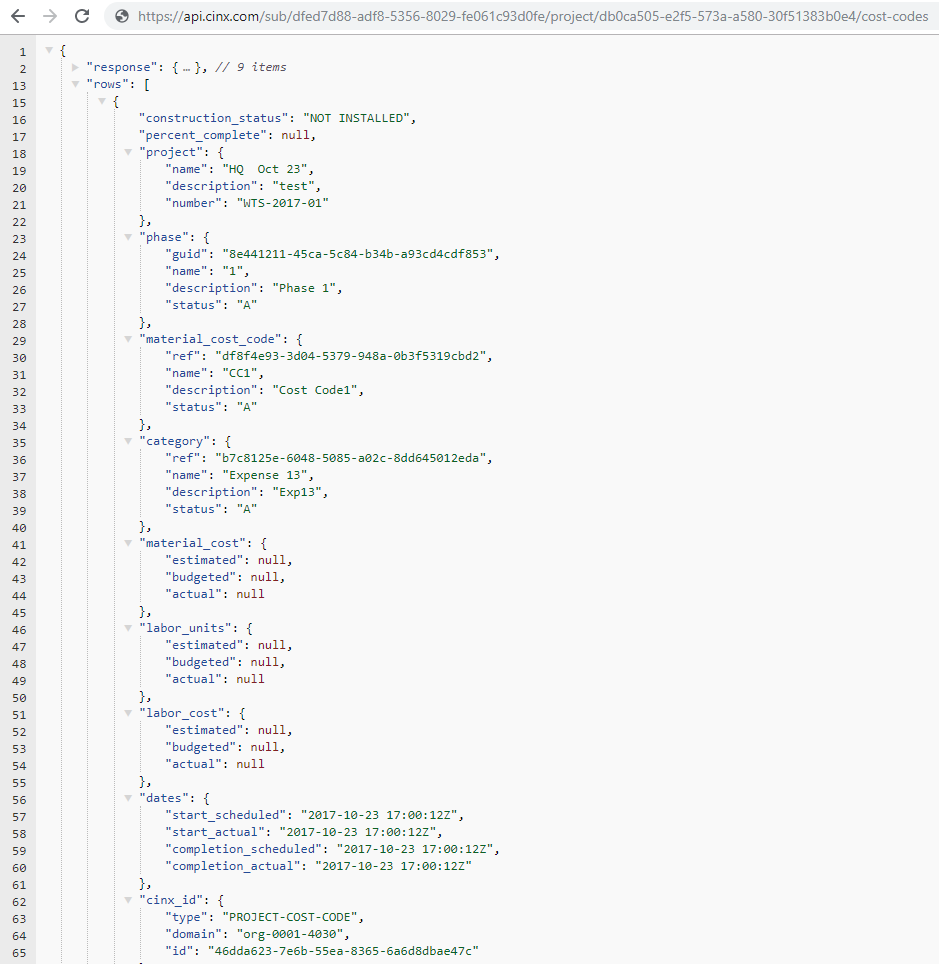
This request will be used to get a list of a project’s job cost information – phases and cost codes. The response will contain full details for each phase/cost code.

URL Pattern:

{api path}/sub/{B2B Id GUID}/project/{Project GUID}/cost-codes

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/project/db0ca505-e2f5-573a-a580-30f51383b0e4/cost-codes>



### API – Create a Project’s Phase and Cost Code

This request will be used to create a project’s phase and cost code.

URL Pattern:

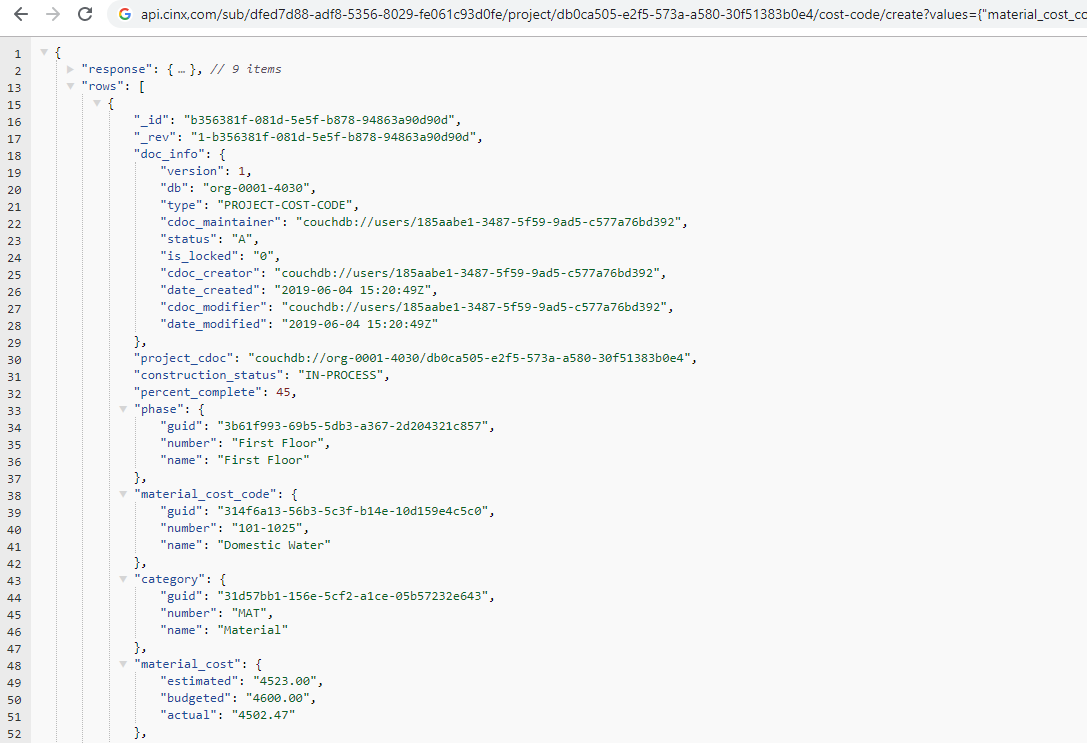
{api path}/sub/{B2B Id GUID}/project/{Project GUID}/cost-code/create?values={JSON Structure}

The **values** section of this URL must contain properly formatted JSON. Please see the ***CINX Project Phase/Cost Code JSON Structures*** section of this document for a listing of the available JSON structures.

*Please note that Phases, Cost Codes, and Categories must be established in the company’s CINX account before they can be attached to a specific project.*

Sample:

https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/project/db0ca505-e2f5-573a-a580-30f51383b0e4/cost-code/create?values={"material\_cost\_code":"101-1025","phase":"First Floor","category":"MAT","construction\_status":"IN-PROCESS","percent\_complete":"45","material\_cost":{ "estimated":"4523.00","budgeted":"4600.00","actual":"4502.47"},"labor\_cost":{ "estimated":"1001.12","budgeted":"1050.00","actual":"1093.00"},"labor\_units":{"estimated":"100","budgeted":"150","actual":"93"},"dates":{ "start\_scheduled":"2019-06-01","start\_actual":"2019-06-10","completion\_scheduled":"2020-01-14","completion\_actual":"2019-02-01"}}



### API – Modify a Project’s Phase and Cost Code

This request will be used to modify a project’s phase and cost code.

URL Pattern:

{api path}/sub/{B2B Id GUID}/project-cost-code/{Project Cost Code GUID}/modify?values={JSON Structure}

The **values** section of this URL must contain properly formatted JSON. Please see the ***CINX Project Phase/Cost Code JSON Structures*** section of this document for a listing of the available JSON structures.

Note: Only modified values need to be submitted.

Sample:

https:/ /api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/project-cost-code/b356381f-081d-5e5f-b878-94863a90d90d/modify?values={"percent\_complete":"100","construction\_status":"INSTALLED"}



### CINX Project Phase/Cost Code JSON Structures

The JSON structures shown below can be used in API calls to **Create** or **Modify** a Project.

**Material Cost Code (required)**

“material\_cost\_code”:”value”

Potential values are: **The name/number of the cost code or the CINX GUID for the cost code.**

**Phase (name or CINX GUID)**

“phase”:”value”

Potential values are: **The name/number of the phase or the CINX GUID for the phase.**

**Category (name or CINX GUID)**

“category”:”value”

Potential values are: **The name/number of the category or the CINX GUID for the category.**

**Construction Status**

“construction\_status”:”value”

Available options are: **NOT INSTALLED, IN-PROCESS, INSTALLED, TESTED, COMMISSIONED, OPERATIONAL**

Default value: **NOT INSTALLED**

**Percent Complete**

“percent\_complete”:”value”

**Material Cost** (object)

“material\_cost”:

“estimated”:”value”,

“budgeted”:”value”,

“actual”:”value”

Sample: "material\_cost":{ "estimated":”4523.00”,"budgeted":”4600.00”,"actual":”4502.47” }

**Labor Cost** (object)

“labor\_cost”:

“estimated”:”value”,

“budgeted”:”value”,

“actual”:”value”

Sample: "labor\_cost":{ "estimated":”1001.12”,"budgeted":”1050.00”,"actual":”1093.00” }

**Labor Units** (object)

“labor\_units”:

“estimated”:”value”,

“budgeted”:”value”,

“actual”:”value”

Sample: "labor\_units":{ "estimated":”100”,"budgeted":”150”,"actual":”93” }

**Dates** (object)

“dates”:

“start\_scheduled”:”value”,

“start\_actual”:”value”,

“completion\_scheduled”:”value”,

“completion\_actual”:”value”

Date Format: **YYYY-MM-DD**

Sample: "dates":{ "start\_scheduled”:”2019-06-01”,”start\_actual”:”2019-06-10”,”completion\_scheduled”:”2020-01-14”,”completion\_actual”:”2019-02-01” }

### API – POST Project Items – csv Format for Autodesk Fabrication

This URL will be used to POST a csv file containing Autodesk Fabrication items for a specified project.

URL Pattern:

{api path}/sub/{B2B Id GUID}/partner/exec/cinx/import-excel-bom-items?bom\_cdoc=couchdb://{CINX Org Id}/{CINX Project GUID}&user=CINX-SYSTEM&partner=autodesk-fab

Sample:

<https://api.cinx.com/sub/e8478738-8c4b-a152-609d-f50e34be0090/partner/exec/cinx/import-excel-bom-items?bom_cdoc=couchdb://org-0000-6138/3dde31ee-694e-0e43-8af2-8382bb6dcc1f&user=CINX-SYSTEM&partner=autodesk-fab>

Parameter Information

**/partner/exec/cinx/import-excel-bom-items**  <literal>

**?bom\_cdoc= couchdb://** <literal>

*CINX Org ID* = CINX Id of Org that the current user is associated with /default Org Id (ie, **org-0000-8761** for the Your company Development Account)

*Project GUID* = CINX Project GUID (ie, 3dde31ee-694e-0e43-8af2-8382bb6dcc1f)

**&user=CINX-SYSTEM** <literal>

**&partner=autodesk-fab** <literal>

### CINX BOM and BOM-ITEM Introduction

**BOM**

Like many construction software systems, CINX employs the BOM concept. BOM is an acronym for Bill of Materials. A BOM is a collection of materials/services that are grouped together for a particular job or transactional purpose. A BOM will generally contain some descriptive information about its use and then have a listing of materials or services that are linked to it.

CINX supports the following BOM types:

BOM-ASSEMBLY (Assembly)

BOM-REQ (Requisition)

BOM-RFQ (Request for Quotation)

BOM-QUOTE (Quotation)

BOM-PO (Purchase Order)

BOM-PO-MEMO (Purchase Order Change Order)

BOM-DELIVERY

BOM-RETURN

BOM-INVOICE

PROJECT

**BOM-ITEM**

Once a Catalog item is added to a BOM/Project it becomes a BOM-ITEM. Items imported from another system (estimating, CAD) will also become BOM-ITEMs.

BOM-ITEM is a common CINX item data structure that is used of all BOM types. It inherits from the Catalog Item structure and also provides additional fields for defining its use within a project (quantities, breakdown structure, spec requirements) and transactions (availability and delivery information).

The BOM-ITEM will be assigned a CINX GUID and also contain references (GUIDs) to the “BOM/Project” it was attached to and also its original catalog item.

### API – GET BOM List – General Response

This request will be used to get a list of the org’s BOMs from CINX. The response will contain general information that is common to all BOMs in CINX. Note: Other API calls are available to provide additional information specific to each type of BOM

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms

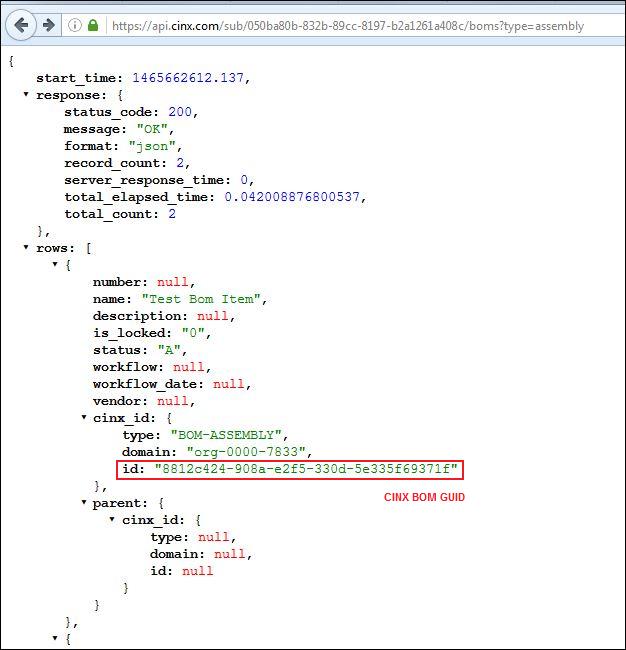
URL Pattern using a BOM-Type filter:

{api path}/sub/{B2B Id GUID}/boms?type=pml

Available BOM-Type filters: **type=assembly**, **type=po**, **type=quote**, **type=rfq**, **type=req**, **type=po**, **type=po-memo**, **type=delivery**, **type=return**, **type=invoice**

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/boms?type=assembly>



### API – GET Quotes for a Project

This request will be used to get a list of the org’s Quotes for a given project.

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms/project/{Project GUID}?type=quote

Supported Filters - The following filters are also supported for this call:

Vendor Reference – will limit results to a single vendor

URL parameter: **&vendor={vendor CINX Id}**

Quote Workflow Status – will limit results to a single workflow status

URL parameter: **&workflow={workflow status value}**

URL workflow status values: **IN-PROCESS, SUBMITTED, RECEIVED, ERP PENDING, ERP ACCEPTED, ERP FAILED**

Delivery Location Type – will limit results to a specific delivery type location

URL Parameter: **&delivery={option from below}**

Available options are: **JOB SITE, FABRICATION SHOP, OFFICE, WAREHOUSE, FABRICATOR**

Ship Via – will limit results to a specific ship via (transport type) value

URL Parameter: **&ship\_via={option from below}**

Available options are: **SUPPLIER TRUCK, MOTOR COMMON CARRIER, CUSTOMER PICKUP, TRACKING GROUND, GROUND, AIR EXPRESS, AIR, PRIVATE PARCEL SERVICE**

Deliver By Date – will limit results to a specific date

URL Parameter: **&deliver\_date={date}**

Date Format: **YYYY-MM-DD**

Submitter – will limit results to a specified CINX User

URL parameter: **&submitter={submitter CINX User Id}**

Transaction Number– will limit results to a specified transaction number

URL parameter: **&number={transaction number}**

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/boms/project/f19fc7e7-5f79-aa03-5429-21b6367dd8e3?type=quote>



### API – GET Quotes for a Request for Quotation

This request will be used to get a list of the org’s Quotes for a given Request for Quotation.

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms/source/{RFQ GUID}?type=quote

Supported Filters - The following filters are also supported for this call:

Vendor Reference – will limit results to a single vendor

URL parameter: **&vendor={vendor Id}**

Quote Workflow Status – will limit results to a single workflow status

URL parameter: **&workflow={workflow status value}**

Available options are: **IN-PROCESS, SUBMITTED, RECEIVED, ERP PENDING, ERP ACCEPTED, ERP FAILED**

Delivery Location Type – will limit results to a specific delivery type location

URL Parameter: **&delivery={option from below}**

Available options are: **JOB SITE, FABRICATION SHOP, OFFICE, WAREHOUSE, FABRICATOR**

Ship Via – will limit results to a specific ship via (transport type) value

URL Parameter: **&ship\_via={option from below}**

Available options are: **SUPPLIER TRUCK, MOTOR COMMON CARRIER, CUSTOMER PICKUP, TRACKING GROUND, GROUND, AIR EXPRESS, AIR, PRIVATE PARCEL SERVICE**

Deliver By Date – will limit results to a specific date

URL Parameter: **&deliver\_date={date}**

Date Format: **YYYY-MM-DD**

Submitter – will limit results to a specified CINX User

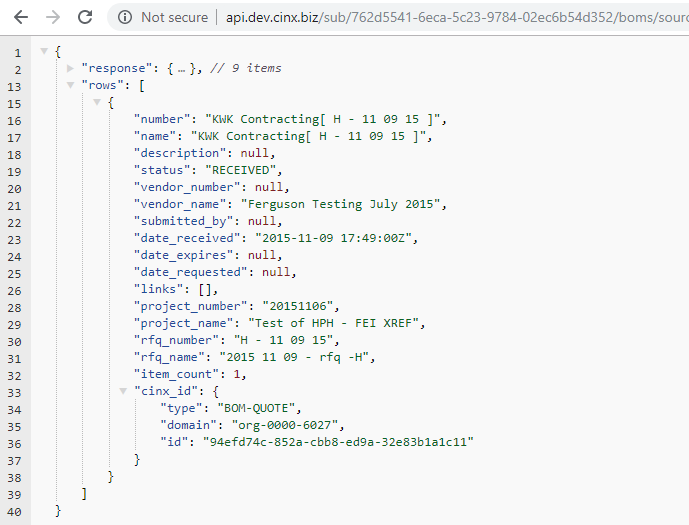
URL parameter: **&submitter={submitter CINX User Id}**

Transaction Number– will limit results to a specified transaction number

URL parameter: **&number={transaction number}**

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/boms/source/86469e52-8a53-667c-1a9e-cf621880535d?type=quote>



### API – GET Requisitions for a Project

This request will be used to get a list of the org’s Requisitions for a given project.

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms/project/{Project GUID}?type=req

Supported Filters - The following filters are also supported for this call:

Delivery Location Type – will limit results to a specific delivery type location

URL Parameter: **&delivery={option from below}**

Available options are: **JOB SITE, FABRICATION SHOP, OFFICE, WAREHOUSE, FABRICATOR**

Procurement Status – will limit results to a specific procurement status

URL Parameter: **&procurement={option from below}**

Available options are: **OPEN, SUBMITTED, IN-REVIEW, APPROVED, APPROVED W/MODS, REJECTED, PENDING ORDER, COMPLETE, CLOSED, CANCELLED, RESUBMITTED**

Vendor Reference – will limit results to a single vendor

URL parameter: **&vendor={CINX vendor Id}**

Ship Via – will limit results to a specific ship via (transport type) value

URL Parameter: **&ship\_via={option from below}**

Available options are: **SUPPLIER TRUCK, MOTOR COMMON CARRIER, CUSTOMER PICKUP, TRACKING GROUND, GROUND, AIR EXPRESS, AIR, PRIVATE PARCEL SERVICE**

Deliver By Date – will limit results to a specific date

URL Parameter: **&deliver\_date={date}**

Date Format: **YYYY-MM-DD**

Submitter – will limit results to a specified CINX User

URL parameter: **&submitter={submitter CINX User Id}**

Current Owner– will limit results to a specified CINX User to whom the transaction is assigned

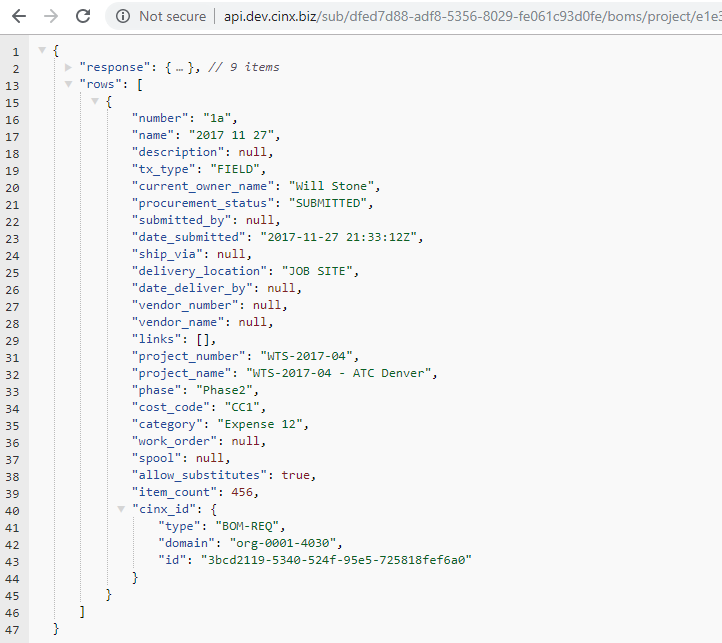
URL parameter: **&owner={CINX User Id}**

Transaction Number– will limit results to a specified transaction number

URL parameter: **&number={transaction number}**

Sample:

https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/project/e1e3cb54-b00a-55ae-a079-072732542bf4?type=req&delivery=JOB SITE



### API – GET Request for Quotations for a Project

This request will be used to get a list of the org’s Request for Quotations for a given project.

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms/project/{Project GUID}?type=rfq

Supported Filters - The following filters are also supported for this call:

Delivery Location Type – will limit results to a specific delivery type location

URL Parameter: **&delivery={option from below}**

Available options are: **JOB SITE, FABRICATION SHOP, OFFICE, WAREHOUSE, FABRICATOR**

Workflow Status – will limit results to a single workflow status

URL parameter: **&workflow={workflow status value}**

Available options are: **IN-PROCESS, SUBMITTED, RECEIVED, ACCEPTED, DECLINED, QUOTED**

Vendor Reference – will limit results to a single vendor

URL parameter: **&vendor={CINX vendor Id}**

Ship Via – will limit results to a specific ship via (transport type) value

URL Parameter: **&ship\_via={option from below}**

Available options are: **SUPPLIER TRUCK, MOTOR COMMON CARRIER, CUSTOMER PICKUP, TRACKING GROUND, GROUND, AIR EXPRESS, AIR, PRIVATE PARCEL SERVICE**

Deliver By Date – will limit results to a specific date

URL Parameter: **&deliver\_date={date}**

Date Format: **YYYY-MM-DD**

Submitter – will limit results to a specified CINX User

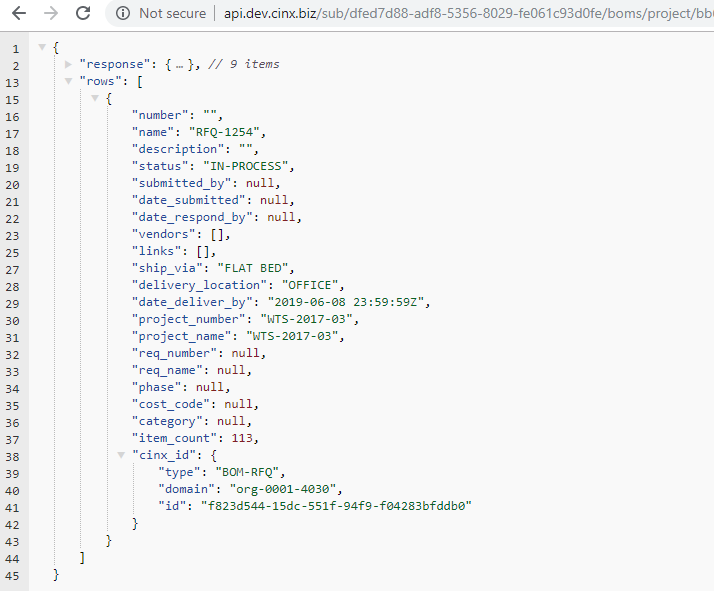
URL parameter: **&submitter={submitter CINX User Id}**

Transaction Number– will limit results to a specified transaction number

URL parameter: **&number={transaction number}**

Sample:

[https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/project/bb613aff-4256-5359-b7cd-391505d19e01?type=rfq&vendor=org-0001-4221](http://api.dev.cinx.biz/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/project/bb613aff-4256-5359-b7cd-391505d19e01?type=rfq&vendor=org-0001-4221)



### API – GET Purchase Orders for a Project

This request will be used to get a list of the org’s Purchase Orders for a given project.

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms/project/{Project GUID}?type=po

Supported Filters - The following filters are also supported for this call:

Delivery Location Type – will limit results to a specific delivery type location

URL Parameter: **&delivery={option from below}**

Available options are: **JOB SITE, FABRICATION SHOP, OFFICE, WAREHOUSE, FABRICATOR**

Procurement Status – will limit results to a specific procurement status

URL Parameter: **&procurement={option from below}**

Available options are: **NOT ORDERED, ORDER PENDING, ORDERED, BACK-ORDERED, DELIVERED, RETURNED, CANCELLED, CLOSED, ON-HOLD, INVOICE-APPROVED**

Workflow Status – will limit results to a single workflow status

URL parameter: **&workflow={workflow status value}**

Available options are: **IN-PROCESS, SUBMITTED, RECEIVED, ACCEPTED, DECLINED, QUOTED**

ERP Status – will limit results to a specific ERP integration status

URL Parameter: **&erp\_status={option from below}**

Available options are: **NOT SUBMITTED, PENDING, SUBMITTED, APPLIED**

Vendor Reference – will limit results to a single vendor

URL parameter: **&vendor={CINX vendor Id}**

Ship Via – will limit results to a specific ship via (transport type) value

URL Parameter: **&ship\_via={option from below}**

Available options are: **SUPPLIER TRUCK, MOTOR COMMON CARRIER, CUSTOMER PICKUP, TRACKING GROUND, GROUND, AIR EXPRESS, AIR, PRIVATE PARCEL SERVICE**

Deliver By Date – will limit results to a specific date

URL Parameter: **&deliver\_date={date}**

Date Format: **YYYY-MM-DD**

Submitter – will limit results to a specified CINX User

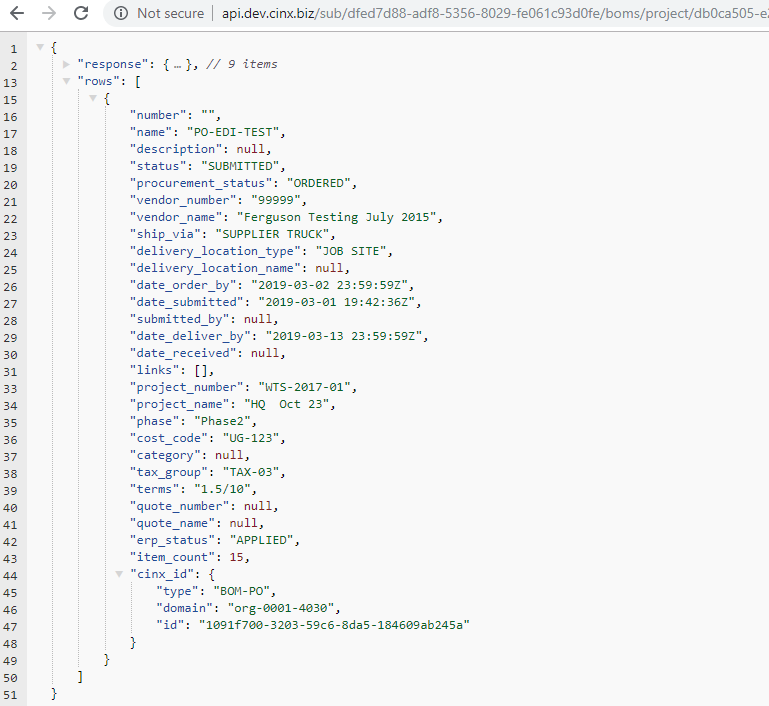
URL parameter: **&submitter={submitter CINX User Id}**

Transaction Number– will limit results to a specified transaction number

URL parameter: **&number={transaction number}**

Sample:

[https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/project/db0ca505-e2f5-573a-a580-30f51383b0e4?type=po&delivery=JOB%20SITE](http://api.dev.cinx.biz/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/project/db0ca505-e2f5-573a-a580-30f51383b0e4?type=po&delivery=JOB%20SITE)



### API – GET Purchase Order Change Orders for a Project

This request will be used to get a list of the org’s Purchase Order Change Orders for a given project.

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms/project/{Project GUID}?type=poco

Supported Filters - The following filters are also supported for this call:

Delivery Location Type – will limit results to a specific delivery type location

URL Parameter: **&delivery={option from below}**

Available options are: **JOB SITE, FABRICATION SHOP, OFFICE, WAREHOUSE, FABRICATOR**

Procurement Status – will limit results to a specific procurement status

URL Parameter: **&procurement={option from below}**

Available options are: **NOT ORDERED, ORDER PENDING, ORDERED, BACK-ORDERED, DELIVERED, RETURNED, CANCELLED, CLOSED, ON-HOLD, INVOICE-APPROVED**

Workflow Status – will limit results to a single workflow status

URL parameter: **&workflow={workflow status value}**

Available options are: **IN-PROCESS, SUBMITTED, RECEIVED, ACCEPTED, DECLINED, QUOTED**

ERP Status – will limit results to a specific ERP integration status

URL Parameter: **&erp\_status={option from below}**

Available options are: **NOT SUBMITTED, PENDING, SUBMITTED, APPLIED**

Vendor Reference – will limit results to a single vendor

URL parameter: **&vendor={CINX vendor Id}**

Ship Via – will limit results to a specific ship via (transport type) value

URL Parameter: **&ship\_via={option from below}**

Available options are: **SUPPLIER TRUCK, MOTOR COMMON CARRIER, CUSTOMER PICKUP, TRACKING GROUND, GROUND, AIR EXPRESS, AIR, PRIVATE PARCEL SERVICE**

Deliver By Date – will limit results to a specific date

URL Parameter: **&deliver\_date={date}**

Date Format: **YYYY-MM-DD**

Submitter – will limit results to a specified CINX User

URL parameter: **&submitter={submitter CINX User Id}**

Transaction Number– will limit results to a specified transaction number

URL parameter: **&number={transaction number}**

Sample:

[https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/project/bb613aff-4256-5359-b7cd-391505d19e01?type=poco&delivery=FABRICATION%20SHOP](http://api.dev.cinx.biz/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/project/bb613aff-4256-5359-b7cd-391505d19e01?type=poco&delivery=FABRICATION%20SHOP)

### API – GET Invoices for a Project

This request will be used to get a list of the org’s Vendor Invoices for a given project.

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms/project/{Project GUID}?type=invoice

Supported Filters - The following filters are also supported for this call:

Status – will limit results to a single status

URL parameter: **&status={status value}**

Available options are: **PENDING E-INVOICE, ENTERED, REVIEWED, ACCEPTED, IN-DISPUTE, ON-HOLD, REJECTED, SUBMITTED**

ERP Status – will limit results to a specific ERP integration status

URL Parameter: **&erp\_status={option from below}**

Available options are: **NOT SUBMITTED, PENDING, SUBMITTED, APPLIED**

Vendor Reference – will limit results to a single vendor

URL parameter: **&vendor={CINX vendor Id}**

Transaction Number– will limit results to a specified transaction number

URL parameter: **&number={transaction number}**

Current Owner– will limit results to a specified CINX User to whom the transaction is assigned

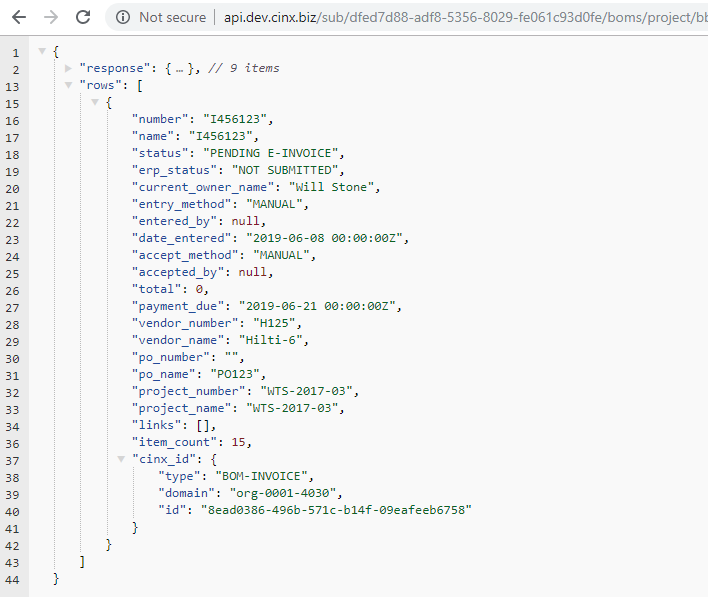
URL parameter: **&owner={CINX User Id}**

Purchase Order Reference – will limit results to a specified CINX Purchase Order

URL parameter: **&po\_guid={CINX Purchase Order Id}**

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/project/bb613aff-4256-5359-b7cd-391505d19e01?type=invoice&number=I456123>



### API – GET Vendor Deliveries for a Project

This request will be used to get a list of the org’s Vendor Deliveries for a given project.

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms/project/{Project GUID}?type=delivery

Supported Filters - The following filters are also supported for this call:

Vendor Reference – will limit results to a single vendor

URL parameter: **&vendor={CINX vendor Id}**

Transaction Number– will limit results to a specified transaction number

URL parameter: **&number={transaction number}**

Delivery Location Type – will limit results to a specific delivery type location

URL Parameter: **&delivery={option from below}**

Available options are: **JOB SITE, FABRICATION SHOP, OFFICE, WAREHOUSE, FABRICATOR**

Ship Via – will limit results to a specific ship via (transport type) value

URL Parameter: **&ship\_via={option from below}**

Available options are: **SUPPLIER TRUCK, MOTOR COMMON CARRIER, CUSTOMER PICKUP, TRACKING GROUND, GROUND, AIR EXPRESS, AIR, PRIVATE PARCEL SERVICE**

Submitter – will limit results to a specified CINX User

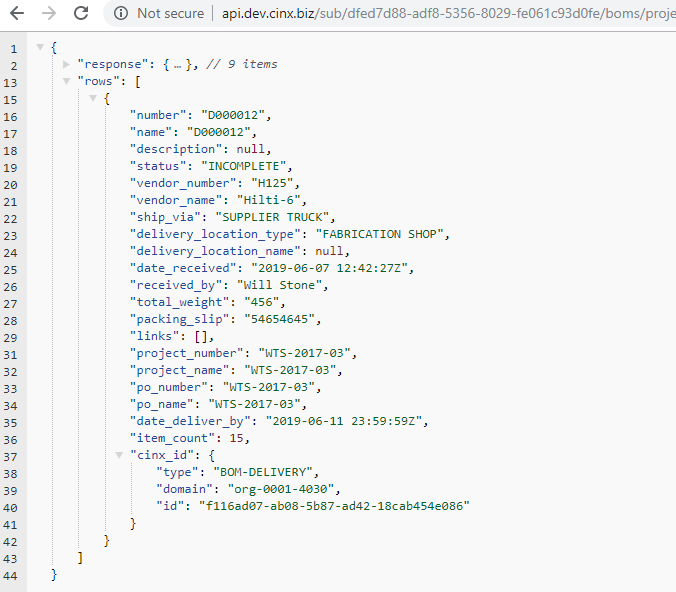
URL parameter: **&submitter={submitter CINX User Id}**

Purchase Order Reference – will limit results to a specified CINX Purchase Order

URL parameter: **&po\_guid={CINX Purchase Order Id}**

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/project/bb613aff-4256-5359-b7cd-391505d19e01?type=delivery&vendor=org-0001-4221>



### API – GET Vendor Returns for a Project

This request will be used to get a list of the org’s Vendor Returns for a given project.

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms/project/{Project GUID}?type=return

Supported Filters - The following filters are also supported for this call:

Vendor Reference – will limit results to a single vendor

URL parameter: **&vendor={CINX vendor Id}**

Transaction Number– will limit results to a specified transaction number

URL parameter: **&number={transaction number}**

Delivery Location Type – will limit results to a specific delivery type location

URL Parameter: **&delivery={option from below}**

Available options are: **JOB SITE, FABRICATION SHOP, OFFICE, WAREHOUSE, FABRICATOR**

Ship Via – will limit results to a specific ship via (transport type) value

URL Parameter: **&ship\_via={option from below}**

Available options are: **SUPPLIER TRUCK, MOTOR COMMON CARRIER, CUSTOMER PICKUP, TRACKING GROUND, GROUND, AIR EXPRESS, AIR, PRIVATE PARCEL SERVICE**

Current Owner– will limit results to a specified CINX User to whom the transaction is assigned

URL parameter: **&owner={CINX User Id}**

Status – will limit results to a single status

URL parameter: **&status={status value}**

Available options are: **CREATED, RETURN REQUESTED, RMA REQUESTED, RMA RECEIVED, RMA REFUSED, IN-PROCESS, SHIPPED, RECEIVED, VENDOR INSPECTION, CLOSED**

Sample:

[https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/project/bb613aff-4256-5359-b7cd-391505d19e01?type=return&vendor=org-0001-4221](http://api.dev.cinx.biz/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/project/bb613aff-4256-5359-b7cd-391505d19e01?type=return&vendor=org-0001-4221)



### API – GET Child Transactions for a Specific Purchase Order

This request will be used to get a list of transactions related to a specific Purchase Order.

URL Pattern:

{api path}/sub/{B2B Id GUID}/boms/source/{PO GUID}?type={transaction type reference}

Supported Transaction Type References:

Purchase Order Change Order: **poco**

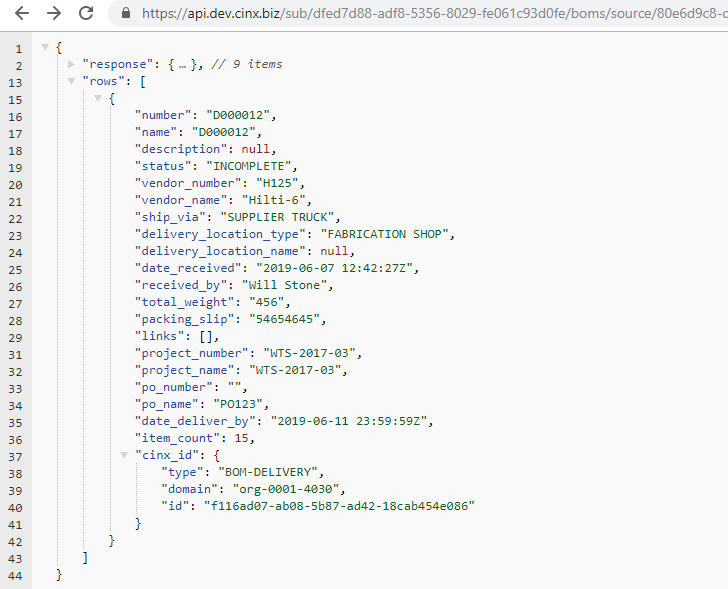
Delivery Reference: **delivery**

Invoice Reference: **invoice**

Return Reference: **return**

Sample:

[https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/source/80e6d9c8-d2e4-5aaa-9bd3-3957430741dd?type=delivery](https://api.dev.cinx.biz/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/boms/source/80e6d9c8-d2e4-5aaa-9bd3-3957430741dd?type=delivery)



### API – GET Back-Ordered Items

This request will be used to get a list of items that have a procurement status of Back-Ordered.

URL Pattern:

{api path}/sub/{B2B Id GUID}/bom/back-ordered/items

{api path}/sub/{B2B Id GUID}/bom/back-ordered/items?{filter type name}={filter value}

Supported Filters - The following filters are also supported for this call:

Delivery Location Type – will limit results to a specific delivery type location

URL Parameter: **delivery={option from below}**

Available options are: **JOB SITE, FABRICATION SHOP, OFFICE, WAREHOUSE, FABRICATOR**

CINX Project GUID– will limit results to a specific project

URL Parameter: **project={CINX Project GUID}**

Vendor Reference – will limit results to a single vendor

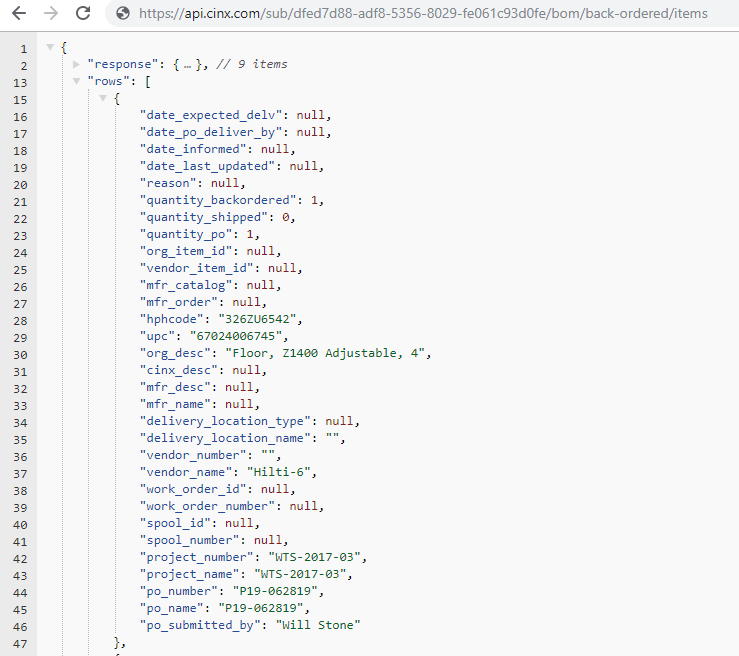
URL parameter: **vendor={CINX vendor Id}**

CINX Purchase Order GUID – will limit results to a specific purchase order

URL Parameter: **po={CINX Purchase Order GUID}**

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/bom/back-ordered/items?po=0435078e-fc66-53a8-b69e-7641e2247437>



### API – Mark a PO as Applied in an ERP System

This request will be used to update the ERP status of a Purchase Order in CINX. This call should be used after a PO from CINX is updated in an ERP/Accounting system.

URL Pattern:

{api path}/sub/{B2B Id GUID}/po/{PO GUID}/applied

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/po/1091f700-3203-59c6-8da5-184609ab245a/applied>



### API – Create a New Requisition

This request will be used to create a new requisition in the org’s CINX account.

URL Pattern:

{api path}/sub/{B2B Id GUID}/req/create?values={JSON Structure}

The **values** section of this URL must contain properly formatted JSON. If the structure is an array, all elements of the array must be submitted. Please see the ***CINX Requisition JSON Structures*** section of this document for a listing of the available JSON structures.

Sample:

https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/req/create?values={"name":"456321","number":"456321","description":"create req"}

### CINX Requisition JSON Structures

The JSON structures shown below can be used in API calls to **Create** or **Modify** a Requisition.

If the structure is an array, all elements of the array must be submitted.

**Name (required)**

“name”:”value”

**Number (required)**

“number”:”value”

**Description**

“description”:”value”

**Procurement Status**

“procurement\_status”:”value”

Available options are: **OPEN, SUBMITTED, IN-REVIEW, APPROVED, APPROVED W/MODS, REJECTED, PENDING ORDER, COMPLETE, CLOSED, CANCELLED, RESUBMITTED**

Default value: **OPEN**

**Project Defaults** (object)

“project\_defaults”:

“construction\_level”:”value”,

“construction\_space”:”value”,

“construction\_sub\_space”:”value”

“construction\_system”:”value”,

“prefabrication\_work\_order\_number”:”value”,

“prefabrication\_spool\_number”:”value”,

“allow\_substitutes”:”{Boolean value}”

Sample: "project\_defaults":{ "construction\_level":”3”,"construction\_space":”lobby”,"construction\_sub\_space":”restrooms”,”construction\_system”:”DWV”,”prefabrication\_work\_order\_number”:”101”,”prefabrication\_spool\_number”:”456”,”allow\_substitutes”:”0” }

**Transaction Information** (object)

“tx”:

“cost\_code\_value”:”value”,

“phase”:”value”,

“category”:”value”

“is\_tax\_exempt”:”Boolean value}”

“tax\_group”:”value”,

“manual\_comment”:”value”,

“allow\_substitutes”:”{Boolean value}”

Sample: "tx":{ "cost\_code\_value":”22-250”,"phase":”Floor 2”,"category":”MAT”,”is\_tax\_exempt”:”0”,”tax\_group”:”NH-13”,”manual\_comment”:”comment text”,”allow\_substitutes”:”1”}

**Delivery** (object)

“delivery”:

“delivery\_attention”:”value”,

“labeling\_instructions”:”value”,

“packaging\_instructions”:”value”,

“delivery\_location\_type”:”value”,

“manual\_instructions”:”value”,

“terms”:”value”

Sample: "delivery":{ "delivery\_attention":”Dan Thomas”,"labeling\_instructions":”101”,"packaging\_instructions":”Pallet”,”delivery\_location\_type”:”fabrication shop”,”manual\_instructions”:”deliver before 9AM”,”terms”:”shipping term text”}

Need By Date

Current Owner

Vendor

Submitter

Project

### API – GET BOM ITEMs

This request will be used to get a list of the BOM-ITEMs for a given BOM.

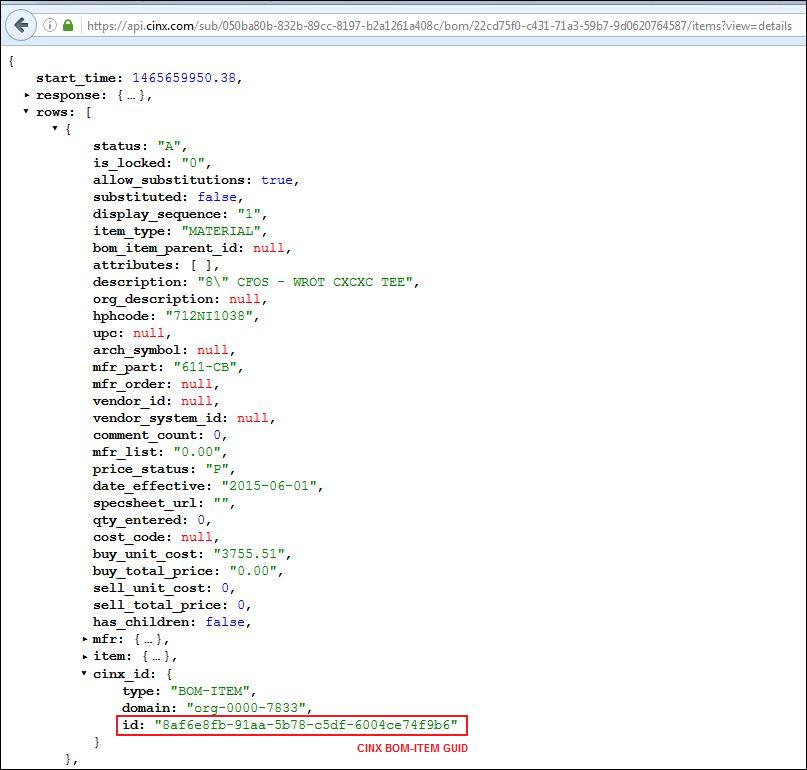
URL Pattern:

{api path}/sub/{B2B Id GUID}/bom/{BOM GUID}/items?view=details

NOTE: If you remove the ?**view=details** the API will return a list of only the IDs for the BOM-ITEMs.

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/bom/22cd75f0-c431-71a3-59b7-9d0620764587/items?view=details>



### API – Create a BOM ITEM – HPH Code

This request will be used to create a new BOM-ITEM for a given BOM using the item’s HPH Code.

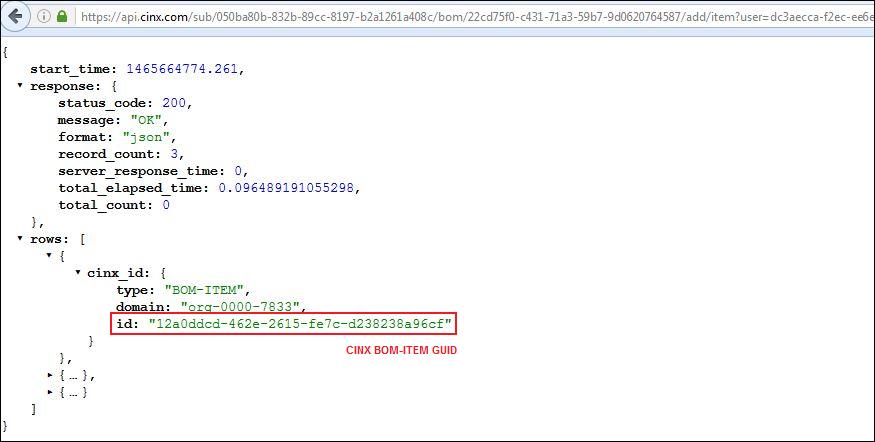
URL Pattern:

{api path}/sub/{B2B Id GUID}/bom/{BOM GUID}/add/item? hphcode={HPH Item Code}

NOTE: The item (HPH Code) must be present in the data source tied to the B2B subscription.

Sample:

[https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/bom/22cd75f0-c431-71a3-59b7-9d0620764587/add/item? hphcode=464PL0011](https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/bom/22cd75f0-c431-71a3-59b7-9d0620764587/add/item?%20hphcode=464PL0011)



### API – GET BOM ITEM Details

This request will be used to get the details of a specific BOM-ITEM.

URL Pattern:

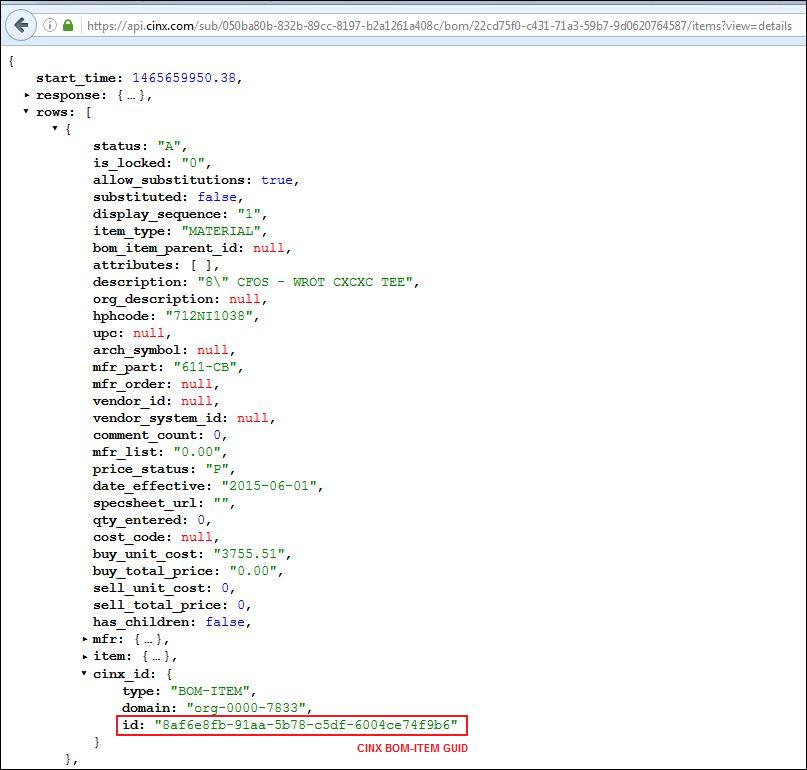
{api path}/sub/{B2B Id GUID}/bom/item/{BOM-ITEM GUID}/xxxxx

NOTE: xxxxx

Sample:

[xxxx](https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/bom/22cd75f0-c431-71a3-59b7-9d0620764587/items?view=details)

NEW PICTURE



### API – Modify a BOM ITEM

This request will be used to modify a BOM-ITEM.

URL Pattern:

{api path}/sub/{B2B Id GUID}/bom/item/{BOM-ITEM GUID}/modify? values={JSON Structure}

The **values** section of this URL must contain properly formatted JSON. If the structure is an array, all elements of the array must be submitted. Please see the ***CINX BOM-Item JSON Structures*** section of this document for a listing of the available JSON structures.

Sample:

[https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/bom/item/579df238-832d-c868-a8b8-91181321840e/modify? values={"cost\_code":"7464"}](https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/bom/item/579df238-832d-c868-a8b8-91181321840e/modify?%20values=%7b%22cost_code%22:%227464%22%7d)



### CINX BOM-Item JSON Structures

The JSON structures shown below can be used in API calls to **Modify** a BOM-Item.

If the structure is an array, all elements of the array must be submitted.

**Description**

“description”:”value”

**Size**

“size”:”value”

**Manufacturer Item Catalog Number**

“mfr\_catalog”:”value”

**Manufacturer Name**

“mfr\_name”:”value”

**Quantity**

“qty\_entered”:”value”

**Order By Date**

“order\_by”:”date value”

**Deliver By Date**

“deliver\_by”:”date value”

**Allow Substitutions**

“allow\_substitutions”:”boolean value”

**Packaging Instructions**

“packaging”:”value”

**Labeling Instructions**

“labeling”:”value”

**Delivery Instructions**

“delivery”:”value”

**Weight**

“weight”:”value”

**Vendor Name (wholesale distributor)**

“vendor\_name”:”value”

**Vendor Availability**

“vendor\_availability\_type”:”value”

Available options are: **STOCKED**, **NON-STOCKED**, **MADE-TO-ORDER**, **NEW – TO BE RELEASED**, **SELL UNTIL DEPLETED**, **OBSOLETE**, {**user defined}**

**Vendor Availability Location**

“vendor\_availability\_location”:”value”

**Vendor Item Id**

“vendor\_id”:”value”

**Vendor Item System Id**

“vendor\_system\_id”:”value”

**Construction Status**

“construction\_status”:”value”

Available options are: **NOT INSTALLED**, **INSTALLED**, **TESTED**, **COMMISSIONED**, **OPERATIONAL**, **{user defined}**

**Project Phase**

“wbs\_phase”:”value”

**Accounting Category**

“wbs\_category”:”value”

**Cost Code**

“cost\_code”:”value”

**Architectural Symbol**

“arch\_symbol”:”value”

**Drawing**

“wbs\_drawing”:”value”

**Level/Floor**

“wbs\_floor”:”value”

**Space/Zone/Section**

“wbs\_space”:”value”

**Sub-Space/Sub-Zone/Room**

“wbs\_sub\_zone”:”value”

**System/Service**

“wbs\_system”:”value”

**Spool**

“wbs\_spool”:”value”

**Work Order**

“wbs\_workorder”:”value”

**Material**

“wbs\_material”:”value”

**Item Type**

“wbs\_item\_type”:”value”

**Buy Price – Base Price**

“buy\_base\_price”:”value”

**Buy Price – Multiplier Value**

“buy\_price\_multiplier”:”value”

**Buy Price – Unit Cost**

“buy\_unit\_cost”:”value”

**Buy Price – Unit-of-Measure**

“buy\_uom”:”value”

**Buy Price – Currency**

“buy\_currency”:”value”

Default value: **USD**

**Sell Price – Base Price**

“sell\_base\_price”:”value”

**Sell Price – Multiplier Value**

“sell\_price\_multiplier”:”value”

**Sell Price – Unit Cost**

“sell\_unit\_cost”:”value”

**Sell Price – Unit-of-Measure**

“sell\_uom”:”value”

**Sell Price – Currency**

“sell\_currency”:”value”

Default value: **USD**

**Labor Unit**

“labor\_unit”:”value”

**Labor Unit Multiplier**

“labor\_unit\_multiplier”:”value”

**Labor Cost**

“labor\_cost”:”value”

**Labor Cost Currency**

“labor\_currency”:”value”

Default value: **USD**

**Attributes** (array)

“attributes”:[{”name”:”name”,”value”: “value”}]

### API – Delete a BOM ITEM

This request will be used to remove a BOM-Item.

URL Pattern:

{api path}/sub/{B2B Id GUID}/bom/item/{BOM-ITEM GUID}/delete

Sample:

[https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/bom/item/12a0ddcd-462e-2615-fe7c-d238238a96cf/delete](https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/bom/item/12a0ddcd-462e-2615-fe7c-d238238a96cf/delete?user=dc3aecca-f2ec-ee6e-eee9-6d96d45390fa)



### CINX Job Cost Introduction - DONE

To allow for the proper reporting of Project items and integration of transactions with contractor accounting systems, CINX has the ability for a company to define their job costing attributes. CINX supports the following Job Cost information types:

**PHASES**

A Phase is construction management concept that allows a company to break-down a project into a smaller group of tasks commonly used for scheduling, sequencing, and reporting purposes. A Phase is also referred to as a Sub-Job, Area, or Extra by other software systems. Examples Phases include, Site Work, Floor, and System.

**COST CODES**

A Cost Code is an accounting and construction management concept that allows a company to track material and labor for cost reporting purposes. Other systems refer to Cost Codes as Item Groups. Example Cost Codes are Sanitary Sewer, Domestic Water, and Gas.

**CATEGORIES**

A Category is an accounting that allows a company to classify project costs for expensing purposes. Categories are also known as Cost Classes. For example, Material, Labor, Sub-Contractor, and Rental.

**TAX GROUPS**

A Tax Group is a collection of tax fields defined in an accounting system that allows for the proper identification and processing of taxes.

### API – GET a List of the Job Cost Phases - DONE

This URL will be used to get the list of Phases in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/get-list/phases

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/get-list/phases>

### API – Create a Job Cost Phase - DONE

This URL will be used to create a new Phase in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/modify/phases? values={“phases”:[{JSON Structure}]}

The JSON **values** section of this URL will begin with **“phases”:** and then will be followed by an array structure comprised of the following components:

**Name**

“name”:”value”

**Description**

“description”:”value”

**Reference Id** (Can be used to store the accounting system’s internal Id for the Phase)

“reference\_id”:”value”

**User Id**

“user\_id”:”value”

**Date Modified**

“date\_modified”:”date/time value”

**Status**

“status”:”value”

Available options are: **A**, **D**

Option Definitions: **A=Active**, **D=Discontinued**

**Material Cost Code** (Can be used to define a default Cost Code for the Phase)

“material\_cost\_code”:”value”

Sample:

[https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/modify/phases? values={%22phases%22:[{%22name%22:%2202-8157%22,%22description%22:%22Site%20Work%22,%22reference\_id%22:%22%22,%22user\_id%22:%22%22,%22date\_modified%22:%22%22,%22status%22:%22A%22,%22material\_cost\_code%22:%225241%22}]}](https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/modify/phases?%20values=%7b%22phases%22:%5b%7b%22name%22:%2202-8157%22,%22description%22:%22Site%20Work%22,%22reference_id%22:%22%22,%22user_id%22:%22%22,%22date_modified%22:%22%22,%22status%22:%22A%22,%22material_cost_code%22:%225241%22%7d%5d%7d)

****

### API – Modify a Job Cost Phase - DONE

This URL will be used to modify an existing Phase in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/modify/phases? values={“phases”:[{JSON Structure}]}

NOTE: This URL pattern is the same as the Create Phase except that the GUID of the Phase to be modified must be included in the JSON array.

The JSON **values** section of this URL will begin with **“phases”:** and then will be followed by an array structure comprised of the following components:

**GUID**

“guid”:”value”

**Name**

“name”:”value”

**Description**

“description”:”value”

**Reference Id** (Can be used to store the accounting system’s internal Id for the Phase)

“reference\_id”:”value”

**User Id**

“user\_id”:”value”

**Date Modified**

“date\_modified”:”date/time value”

**Status**

“status”:”value”

Available options are: **A**, **D**

Option Definitions: **A=Active**, **D=Discontinued**

**Material Cost Code** (Can be used to define a default Cost Code for the Phase)

“material\_cost\_code”:”value”

Sample:

https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/modify/phases?user=dc3aecca-f2ec-ee6e-eee9-6d96d45390fa&values={"phases":[{"guid":"6ebcc266-83b4-85cf-53f9-8db2d52f15cb","name":"02-8158","description":"Site Work","reference\_id":"","user\_id":"","date\_modified":"","status":"A","material\_cost\_code":"5241"}]}



### API – GET a List of the Job Cost Cost Codes - DONE

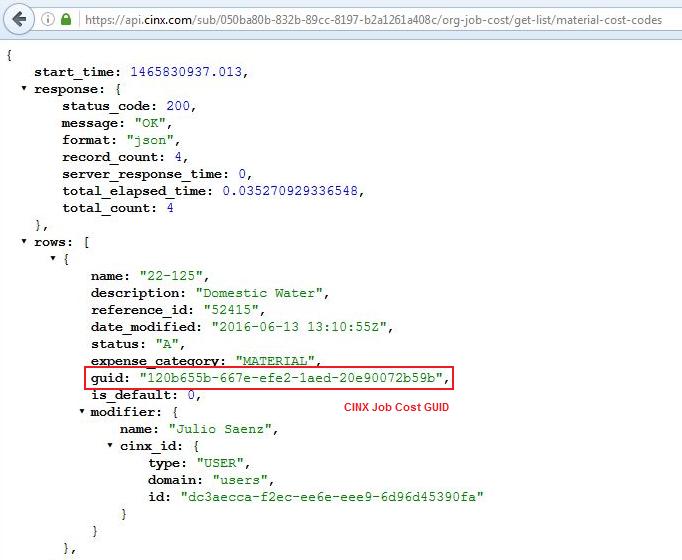
This URL will be used to get the list of Cost Codes in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/get-list/material-cost-codes

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/get-list/material-cost-codes>



### API – Create a Job Cost Cost Code - DONE

This URL will be used to create a new Cost Code in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/modify/material-cost-codes? values={“ material\_cost\_codes”:[{JSON Structure}]}

NOTE: Please use underscores on **material\_cost\_code** when it is within the JSON structure. This is different than the earlier part of the URL where hyphens are used.

The JSON **values** section of this URL will begin with **“material\_cost\_codes”:** and then will be followed by an array structure comprised of the following components:

**Name**

“name”:”value”

**Description**

“description”:”value”

**Reference Id** (Can be used to store the accounting system’s internal Id for the Phase)

“reference\_id”:”value”

**User Id**

“user\_id”:”value”

**Date Modified**

“date\_modified”:”date/time value”

**Status**

“status”:”value”

Available options are: **A**, **D**

Option Definitions: **A=Active**, **D=Discontinued**

**Expense Category** (Can be used to define a default Category for the Cost Code)

“expense\_category”:”value”

Sample:

https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/modify/material-cost-codes? values={"material\_cost\_codes":[{"name":"22-125","description":"Domestic Water","reference\_id":"52415","user\_id":"","date\_modified":"","status":"A","expense\_category":"MATERIAL"}]}

****

### API – Modify a Job Cost Cost Code - DONE

This URL will be used to modify an existing Cost Code in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/modify/material-cost-codes? values={“ material\_cost\_codes”:[{JSON Structure}]}

NOTE: This URL pattern is the same as the Create Cost Code except that the GUID of the Cost Code to be modified must be included in the JSON array.

NOTE: Please use underscores on **material\_cost\_code** when it is within the JSON structure. This is different than the earlier part of the URL where hyphens are used.

The JSON **values** section of this URL will begin with **“material\_cost\_codes”:** and then will be followed by an array structure comprised of the following components:

**GUID**

“guid”:”value”

**Name**

“name”:”value”

**Description**

“description”:”value”

**Reference Id** (Can be used to store the accounting system’s internal Id for the Phase)

“reference\_id”:”value”

**User Id**

“user\_id”:”value”

**Date Modified**

“date\_modified”:”date/time value”

**Status**

“status”:”value”

Available options are: **A**, **D**

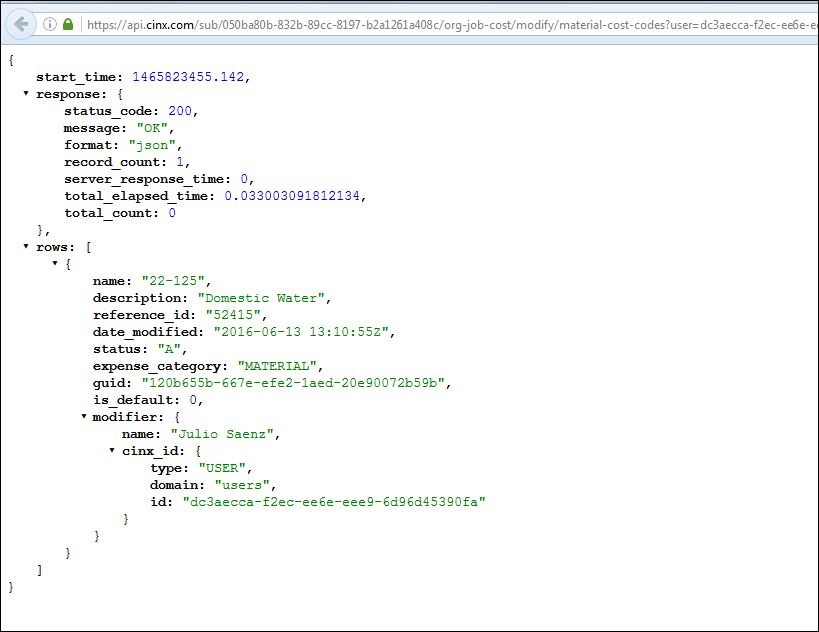
Option Definitions: **A=Active**, **D=Discontinued**

**Expense Category** (Can be used to define a default Category for the Cost Code)

“expense\_category”:”value”

Sample:

https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/modify/material-cost-codes? values={"material\_cost\_codes":[{"guid":"120b655b-667e-efe2-1aed-20e90072b59b","name":"22-125","description":"Domestic Water","reference\_id":"52415","user\_id":"","date\_modified":"","status":"A","expense\_category":"MATERIAL"}]}



### API – GET a List of the Job Cost Categories - DONE

This URL will be used to get the list of Categories in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/get-list/tax-expense-categories

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/get-list/tax-expense-categories>



### API – Create a Job Cost Category - DONE

This URL will be used to create a new Category in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/modify/tax-expense-categories? values={“ tax\_expense\_categories”:[{JSON Structure}]}

NOTE: Please use underscores on **tax\_expense\_categories** when it is within the JSON structure. This is different than the earlier part of the URL where hyphens are used.

The JSON **values** section of this URL will begin with **“tax\_expense\_categories”:** and then will be followed by an array structure comprised of the following components:

**Name**

“name”:”value”

**Description**

“description”:”value”

**Reference Id** (Can be used to store the accounting system’s internal Id for the Phase)

“reference\_id”:”value”

**User Id**

“user\_id”:”value”

**Date Modified**

“date\_modified”:”date/time value”

**Status**

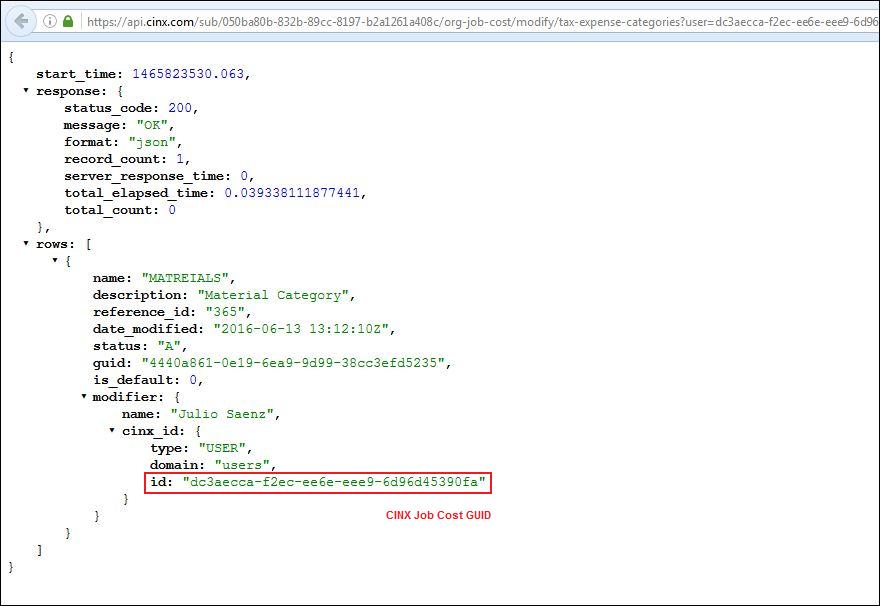
“status”:”value”

Available options are: **A**, **D**

Option Definitions: **A=Active**, **D=Discontinued**

Sample:

https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/modify/tax\_expense\_categories? values={"tax\_expense\_categories":[{"name":"MATREIALS","description":"Material Category","reference\_id":"365","user\_id":"","date\_modified":"","status":"A"}]}

****

### API – Modify a Job Cost Category - DONE

This URL will be used to modify an existing Category in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/modify/tax-expense-categories? values={“ tax\_expense\_categories”:[{JSON Structure}]}

NOTE: This URL pattern is the same as the Create Category except that the GUID of the Category to be modified must be included in the JSON array.

NOTE: Please use underscores on **tax\_expense\_categories** when it is within the JSON structure. This is different than the earlier part of the URL where hyphens are used.

The JSON **values** section of this URL will begin with **“tax\_expense\_categories”:** and then will be followed by an array structure comprised of the following components:

**GUID**

“guid”:”value”

**Name**

“name”:”value”

**Description**

“description”:”value”

**Reference Id** (Can be used to store the accounting system’s internal Id for the Phase)

“reference\_id”:”value”

**User Id**

“user\_id”:”value”

**Date Modified**

“date\_modified”:”date/time value”

**Status**

“status”:”value”

Available options are: **A**, **D**

Option Definitions: **A=Active**, **D=Discontinued**

Sample:

https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/modify/tax-expense-categories? values={"tax\_expense\_categories":[{"guid":"4440a861-0e19-6ea9-9d99-38cc3efd5235","name":"MATREIALS","description":"Material Category","reference\_id":"365","user\_id":"","date\_modified":"","status":"A"}]}



### API – GET a List of the Job Cost Tax Groups - DONE

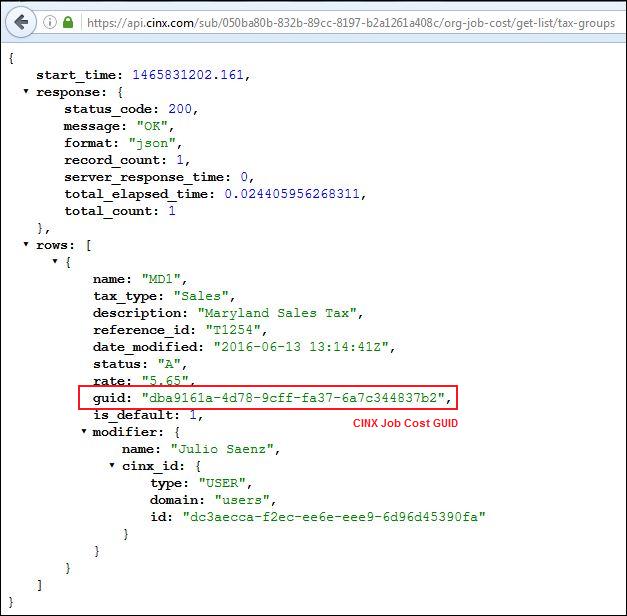
This URL will be used to get the list of Tax Groups in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/get-list/tax-groups

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/get-list/tax-groups>



### API – Create a Job Cost Tax Group - DONE

This URL will be used to create a new Tax Group in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/modify/tax-groups? values={“ tax\_groups”:[{JSON Structure}]}

NOTE: Please use underscores on **tax\_groups** when it is within the JSON structure. This is different than the earlier part of the URL where hyphens are used.

The JSON **values** section of this URL will begin with **“tax\_groups”:** and then will be followed by an array structure comprised of the following components:

**Name**

“name”:”value”

**Tax Type**

“tax\_type”:”value”

Available options are: **SALES**, **USE**, or {user defined}

**Description**

“description”:”value”

**Reference Id** (Can be used to store the accounting system’s internal Id for the Phase)

“reference\_id”:”value”

**User Id**

“user\_id”:”value”

**Date Modified**

“date\_modified”:”date/time value”

**Status**

“status”:”value”

Available options are: **A**, **D**

Option Definitions: **A=Active**, **D=Discontinued**

**Rate**

“rate”:”value”

Sample:

https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/modify/tax-groups? values={"tax\_groups":[{"name":"MD1","tax\_type":"Sales","description":"Maryland Sales Tax","reference\_id":"T1254","user\_id":"","date\_modified":"","status":"A","rate":"5.65"}]}

****

### API – Modify a Job Cost Tax Group - DONE

This URL will be used to modify an existing Tax Group in the company’s CINX Job Costing library.

URL Pattern:

{api path}/sub/{B2B Id GUID}/org-job-cost/modify/tax-groups? values={“ tax\_groups”:[{JSON Structure}]}

NOTE: This URL pattern is the same as the Create Tax Group except that the GUID of the Tax Group to be modified must be included in the JSON array.

NOTE: Please use underscores on **tax\_groups** when it is within the JSON structure. This is different than the earlier part of the URL where hyphens are used.

The JSON **values** section of this URL will begin with **“tax\_groups”:** and then will be followed by an array structure comprised of the following components:

**GUID**

“guid”:”value”

**Name**

“name”:”value”

**Tax Type**

“tax\_type”:”value”

Available options are: **SALES**, **USE**, or {user defined}

**Description**

“description”:”value”

**Reference Id** (Can be used to store the accounting system’s internal Id for the Phase)

“reference\_id”:”value”

**User Id**

“user\_id”:”value”

**Date Modified**

“date\_modified”:”date/time value”

**Status**

“status”:”value”

Available options are: **A**, **D**

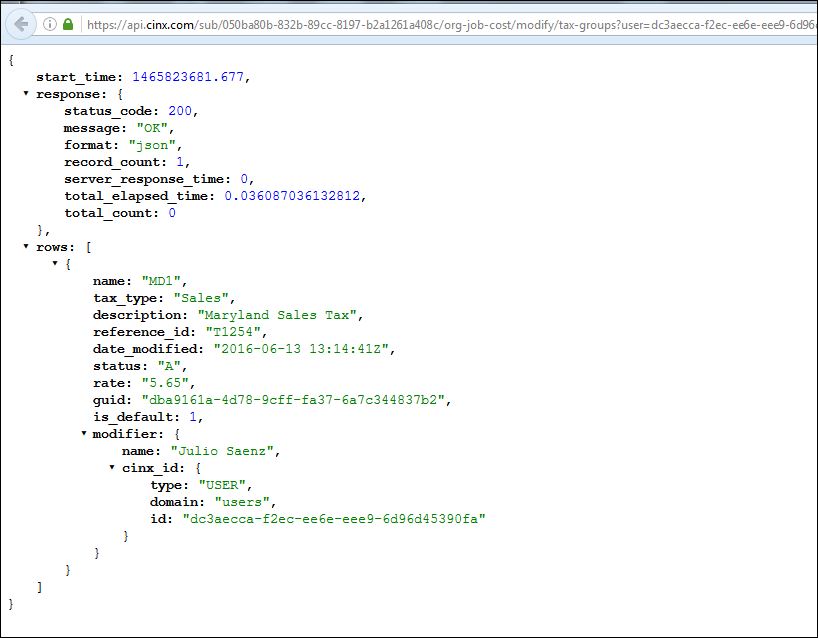
Option Definitions: **A=Active**, **D=Discontinued**

**Rate**

“rate”:”value”

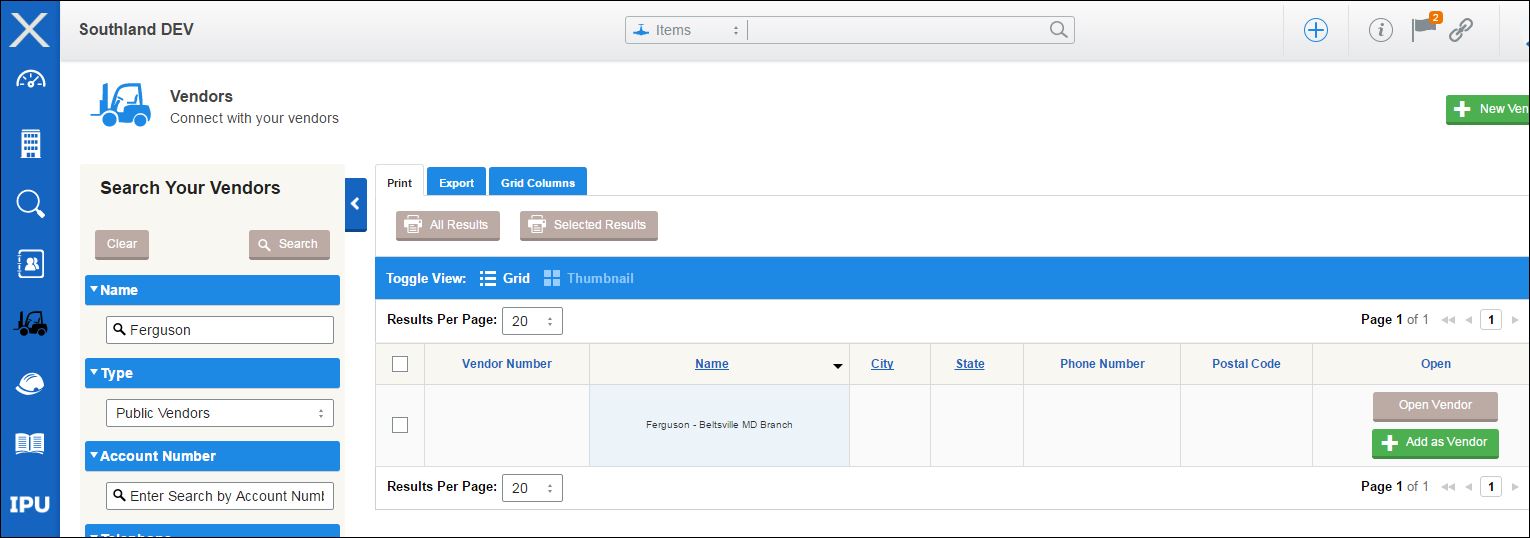
Sample:

https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org-job-cost/modify/tax-groups? values={"tax\_groups":[{"guid":"dba9161a-4d78-9cff-fa37-6a7c344837b2","name":"MD1","tax\_type":"Sales","description":"Maryland Sales Tax","reference\_id":"T1254","user\_id":"","date\_modified":"","status":"A","rate":"5.65"}]}



### CINX Vendor Introduction - DONE

For contractors on the CINX platform, a Vendor is a Wholesale Distributor or other entities that provide products or services. As a web-based platform CINX has the ability for contractors and distributors to connect with one another if they are both already members of the CINX site. Similar to LinkedIn and Facebook, a request is sent by the requesting party to create a relationship whereby CINX activities and transactions can be shared between the two companies.



As shown in the graphic, a simple click of the Add As Vendor button will send a request to the Vendor to accept the contractor’s request.

When a relationship between the two parties is created CINX creates a Commerce GUID that links the two parties. This GUID is the key that allows for the storage of important transaction parameters that are required to deliver and process transactions.

It is therefore also used by the API as a URL parameter when fetching and updating relationship data.

CINX also has the ability for a contractor to create a **Private** vendor. This is a vendor that is not a member of the CINX site. A private vendor is unique to the contractor and will still be linked using a Commerce GUID.

### API – GET Vendor List - DONE

This request will be used to get a list of the company’s Vendors from CINX.

URL Pattern:

{api path}/sub/{B2B Id GUID}/vendors

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/vendors>

****

### API – GET Vendor Details – using CINX Commerce GUID - DONE

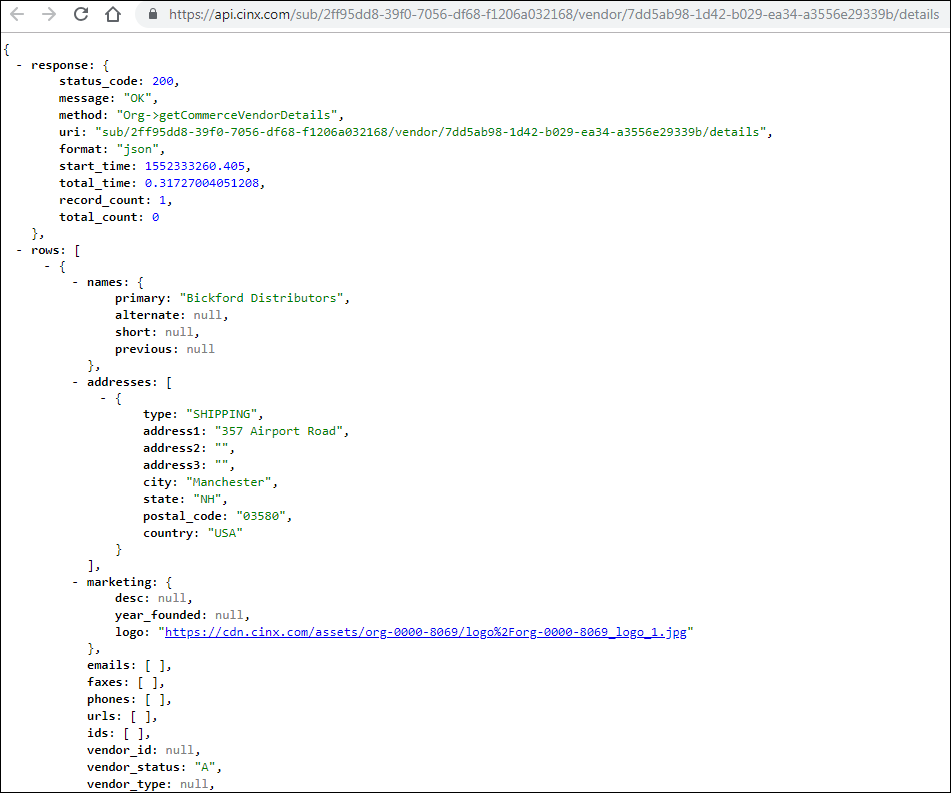
This request will be used to get the details of a vendor.

URL Pattern:

{api path}/sub/{B2B Id GUID}/vendor/{commerce guid}/details

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/vendor/512d5a2f-1d6c-21ca-9b97-9299ac9c802a/details>



### API – PUT New Vendor - DONE

This request will be used to insert a new private vendor for an organization.

URL Pattern:

{api path}/sub/{B2B Id GUID}/private/vendor/create?values={JSON Structure}

The **values** section of this URL must contain properly formatted JSON. If the structure is an array, all elements of the array must be submitted. Please see the ***CINX Vendor JSON Structures*** section of this document for a listing of the available JSON structures.

Sample:

[https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/private/vendor/create?values={%22name\_primary%22:%22Hilti%22,%22vendor\_id%22:%22H123%22,%22vendor\_type%22:%22Hardware%22,%22add\_address%22:{%22type%22:%22Mailing%22,%22address1%22:%22Route%203%22,%22address2%22:%22Building%204%22,%22address3%22:%22Suite%20G%22,%22city%22:%22Woburn%22,%22state%22:%22MA%22,%22postal\_code%22:%2201234%22,%22country%22:%22USA%22,%22latitude%22:%22%22,%22longitude%22:%22%22}}](https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/private/vendor/create?values=%7b%22name_primary%22:%22Hilti%22,%22vendor_id%22:%22H123%22,%22vendor_type%22:%22Hardware%22,%22add_address%22:%7b%22type%22:%22Mailing%22,%22address1%22:%22Route%203%22,%22address2%22:%22Building%204%22,%22address3%22:%22Suite%20G%22,%22city%22:%22Woburn%22,%22state%22:%22MA%22,%22postal_code%22:%2201234%22,%22country%22:%22USA%22,%22latitude%22:%22%22,%22longitude%22:%22%22%7d%7d)



### API – Modify a Vendor - DONE

This request will be used to modify a private vendor for an organization.

URL Pattern:

{api path}/sub/{B2B Id GUID}/private/vendor/{commerce guid}/modify?values={JSON Structure}

The **values** section of this URL must contain properly formatted JSON. If the structure is an array, all elements of the array must be submitted. Please see the ***CINX Vendor JSON Structures*** section of this document for a listing of the available JSON structures.

Sample:

[http://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/private/vendor/org-0001-4221/modify?values={"name\_primary":"Hilti-5","vendor\_id":"H125"}](http://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/private/vendor/org-0001-4221/modify?values=%7b%22name_primary%22:%22Hilti-5%22,%22vendor_id%22:%22H125%22%7d)



### CINX Vendor JSON Structures - DONE

The JSON structures shown below can be used in API calls to **Create** or **Modify** a Vendor.

If the structure is an array, all elements of the array must be submitted.

**Vendor Number/Id**

“vendor\_id”:”value”

**Vendor Status**

“vendor\_status”:”value”

Available options are: **A**, **I**

Option Definitions: **A=Active**, **I=Inactive**

Default value: **A**

**Vendor Type** (each org can define a set list of vendor types to be used to classify their vendors)

“vendor\_type”:”value”

The options shown below are for creating or modifying **Private** Vendors

**Name - Primary (required)**

“name\_primary”:”value”

**Name - Short**

“name\_short”:”value”

**Name - Alternate**

“name\_alternate”:”value”

**Name - Previous**

“name\_previous”:”value”

**Addresses** (array)

“add\_address”:[{”address\_type”:”type value”,”address1”: “value”,”address2”: “value”,”address3”: “value”,”city”: “value”,”state”: “value”,”postal\_code”: “value”,”country”: “value”,”latitude”: “value” ,”longitude”: “value”}]

Available “address types” are: **SHIPPING**, **MAILING**, **BILLING**

Default Country: **USA**

**Location Id**

“location\_id”:”value”

**Location Type**

“location\_type”:”value”

Available “location types” are: **HEADQUARTERS**, **BRANCH**, **WAREHOUSE**, **FABRICATION FACILITY**, **SHOP**, **SALES OFFICE**, **REGIONAL OFFICE**, **CENTRAL DISTRIBUTION CENTER**, **TRAINING CENTER**, **SHOWROOM**

### API Call – POST External file with User Notifications - DONE

This request will be used to POST an uploaded external file (Excel, PDF, txt) to the CINX CDN. In addition, this POST operation allows for user notifications to be delivered by the API. The notification types are **Email** and **CINX System Alert**.

URL Pattern:

{api path}/sub/{B2B Id GUID}/partner/exec/cinx/lib-doc? type=MATERIAL+REQUISITION &name={requisition id}&parent=couchdb://{CINX Org Id}/{CINX Project GUID}&email\_subject={text}&notify\_email={User Email address}&notify\_alert={CINX User GUID}

Sample:

[https://api.cinx.com/sub/e8478738-8c4b-a152-609d-f50e34be0090/partner/exec/cinx/lib-doc? type=MATERIAL+REQUISITION&name=Req123&parent=couchdb://org-0000-6138/3dde31ee-694e-0e43-8af2-8382bb6dcc1f&email\_subject=NewShopMaterialRequestionReq123&notify\_email=bakigervalla@gmail.com&notify\_email=dmcphaul@hphguide.com&notify\_alert=c61a47a8-3a4a-025a-0db1-2b9e609da089&notify\_alert=54e0afa4-c359-f55b-ad3c-cc63df698900](https://api.cinx.com/sub/e8478738-8c4b-a152-609d-f50e34be0090/partner/exec/cinx/lib-doc?%20type=MATERIAL+REQUISITION&name=Req123&parent=couchdb://org-0000-6138/3dde31ee-694e-0e43-8af2-8382bb6dcc1f&email_subject=NewShopMaterialRequestionReq123&notify_email=bakigervalla@gmail.com&notify_email=dmcphaul@hphguide.com&notify_alert=c61a47a8-3a4a-025a-0db1-2b9e609da089&notify_alert=54e0afa4-c359-f55b-ad3c-cc63df698900)

URL Parameter Information

**/partner/exec/cinx/lib-doc** <literal>

**?type="MATERIAL REQUISITION"**  <literal>

**&name=** {document name} populate with the Requisition Id

**&parent=couchdb://** <literal>

*CINX Org ID* = CINX Id of Org that the current user is associated with /default Org Id (ie, **org-0000-7833** for Your company Dev account)

*Project GUID* = CINX Project GUID

**&email\_subject=** {subject line for email notification} “**New CINX Material Requisition Received –** {*req id*}”

**&notify\_email=**{email address} – Send to User email address - REPEAT as needed using full parameter label and value

**&notify\_alert=**{user id/cdoc} - Send to CINX User GUID - REPEAT as needed using full parameter label and value

### API – GET Received User Alerts - DONE

This request will be used get a User’s CINX system alerts.

URL Pattern:

{api path}/sub/{B2B Id GUID}/user/alerts/received?shown=0

Samples:

**“Shown” Parameter**

This optional URL parameter allows us to filter the alerts given their “User Shown” status.

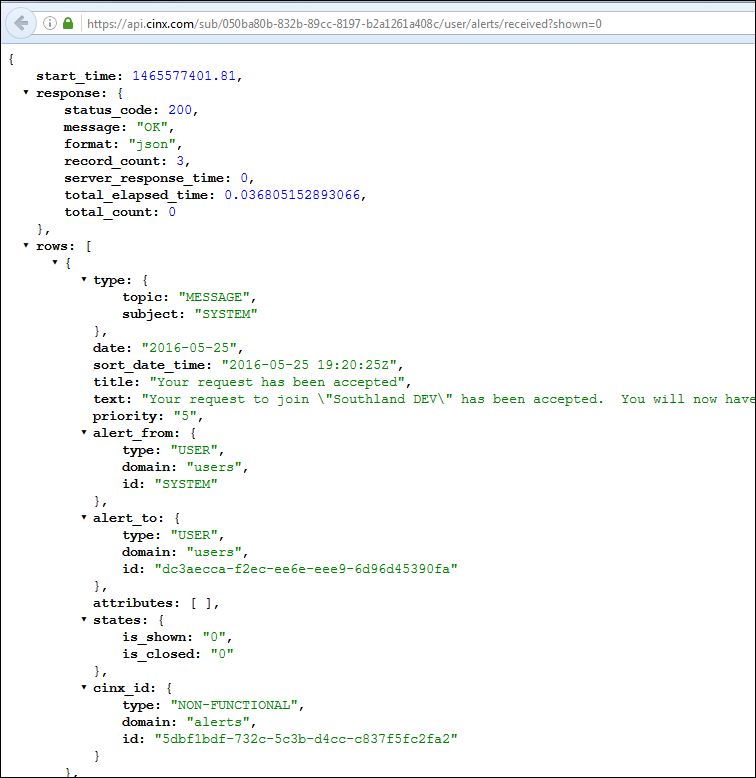
**?shown=0**

**?shown=1**

An alert is marked **shown** when a user has view the alert.

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/user/alerts/received?shown=0>

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/user/alerts/received?shown=1>

****

**“Closed” Parameter**

This optional URL parameter allows us to filter the alerts given their “User Closed” status.

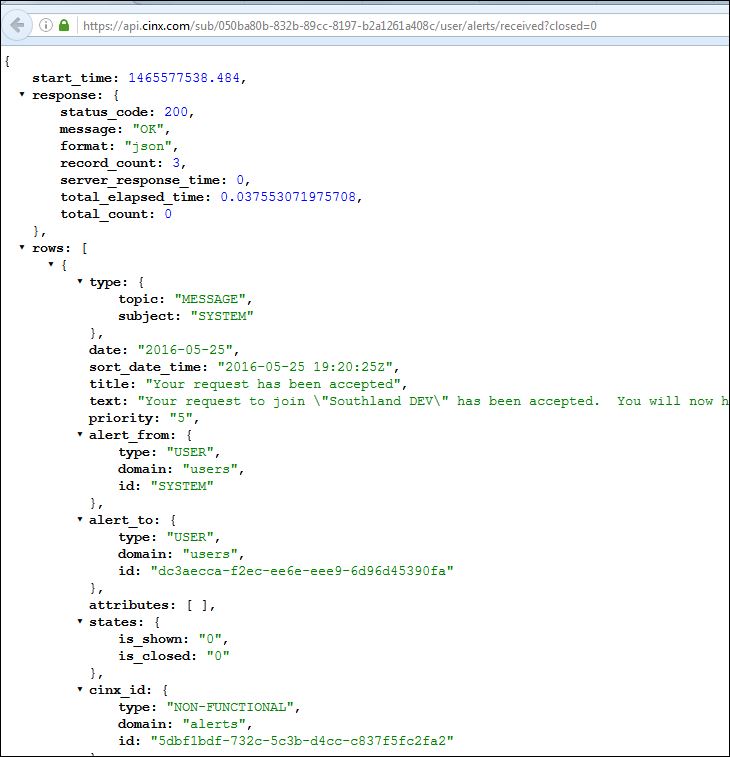
An alert is marked **closed** when a user has “deleted” the alert from a UI.

**?closed=0**

**?closed=1**

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/user/alerts/received?closed=0>

https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/user/alerts/received?closed=1



### API – Mark User Alerts Shown or Closed - DONE

This request will be used update a user’s CINX system alert’s **Shown** or **Closed** status.

URL Pattern:

{api path}/sub/{B2B Id GUID}/user/alert/{alert GUID}/modify?values={“**shown** or **closed** attribute label”:{boolean indicator}

Samples:

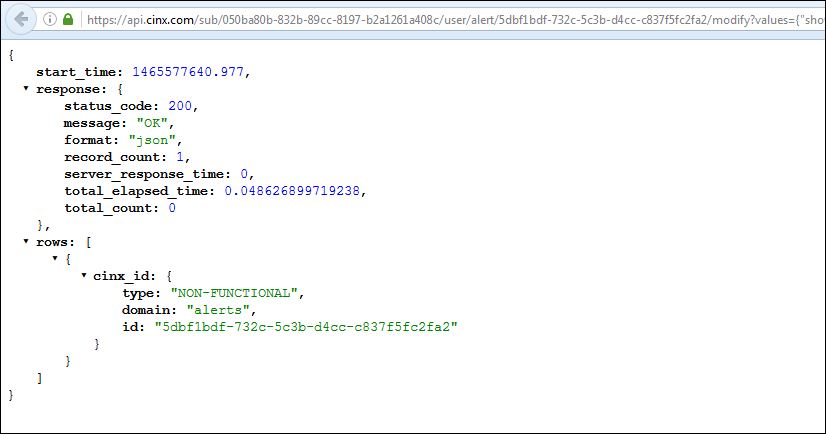
**Mark Shown**

{api path}/sub/{B2B Id GUID}/user /alert/{alert GUID}modify?values={"shown":1 }

[https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/user/alert/5dbf1bdf-732c-5c3b-d4cc-c837f5fc2fa2/modify?values={%22shown%22:1}](https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/user/alert/5dbf1bdf-732c-5c3b-d4cc-c837f5fc2fa2/modify?values=%7b%22shown%22:1%7d)

{api path}/sub/{B2B Id GUID}/user/alert/{alert guid}/modify?values={"closed": 1 }

[https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/user/alert/5dbf1bdf-732c-5c3b-d4cc-c837f5fc2fa2/modify?values={%22closed%22:1}](https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/user/alert/5dbf1bdf-732c-5c3b-d4cc-c837f5fc2fa2/modify?values=%7b%22closed%22:1%7d)



### API – GET Organization Details - DONE

This request will be used to get the company’s CINX organization details.

URL Pattern:

{api path}/sub/{B2B GUID}/org/details

Sample:

<https://api.cinx.com/sub/050ba80b-832b-89cc-8197-b2a1261a408c/org/details>



### API – Modify Organization Details - DONE

This request will be used to modify an organization’s details.

URL Pattern:

{api path}/sub/{B2B Id GUID}/modify/org?values={JSON Structure}

The **values** section of this URL must contain properly formatted JSON. If the structure is an array, all elements of the array must be submitted. Please see the ***CINX Organization JSON Structures*** section of this document for a listing of the available JSON structures.

Sample:

Dev sample using HPH Account:

[http://api.dev.cinx.biz/sub/83d8ba34-ecf1-59ab-1013-0ba2603adb36/modify/org?values={"types\_customers":"Residential,Commercial,Industrial"}](http://api.dev.cinx.biz/sub/83d8ba34-ecf1-59ab-1013-0ba2603adb36/modify/org?values=%7b%22types_customers%22:%22Residential,Commercial,Industrial%22%7d)

### CINX Organization JSON Structures - DONE

The JSON structures shown below can be used in API calls to **Modify** an existing CINX Organization.

If the structure is an array, all elements of the array must be submitted.

**Name - Primary (required)**

“name\_primary”:”value”

**Name - Short**

“name\_short”:”value”

**Name - Alternate**

“name\_alternate”:”value”

**Name - Previous**

“name\_previous”:”value”

**Addresses** (array)

“add\_address”:[{”address\_type”:”type value”,”address1”: “value”,”address2”: “value”,”address3”: “value”,”city”: “value”,”state”: “value”,”postal\_code”: “value”,”country”: “value”,”latitude”: “value” ,”longitude”: “value”}]

Available “address types” are: **SHIPPING**, **MAILING**, **BILLING**

Default Country: **USA**

**Marketing – Org Description**

“marketing\_desc”:”value”

**Marketing – Org Tag Line**

“marketing\_tag”:”value”

**Marketing – Org Year Founded**

“marketing\_founded”:”value”

**Location Id**

“location\_id”:”value”

**Location Type**

“location\_type”:”value”

Available “location types” are: **HEADQUARTERS**, **BRANCH**, **WAREHOUSE**, **FABRICATION FACILITY**, **SHOP**, **SALES OFFICE**, **REGIONAL OFFICE**, **CENTRAL DISTRIBUTION CENTER**, **TRAINING CENTER**, **SHOWROOM**

**Customer Types**

“types\_customers”:”value”

Values parameter can be a comma delimited list of types. For example: **RESIDENTIAL,COMMERCIAL**

**Project Types**

“types\_projects”:”value”

Values parameter can be a comma delimited list of types. For example: **EDUCATION,SERVICE,CHEMICAL,BREWERY,INDUSTRIAL**

**Vendor Types**

“types\_vendors”:”value”

Values parameter can be a comma delimited list of types. For example: **PVF,HANGER,RENTAL,OFFICE SUPPLIES**

Additional content types are available; please contact HPH if your system requires additional project content.

### API – GET an Organization’s User List - DONE

This request will be used to get a company’s list of CINX users. The results of this API will contain the User’s CINX system Id which can be used in other API calls.

URL Pattern:

{api path}/sub/{B2B GUID}/users

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/users>



### API – GET an Organization’s Addresses - DONE

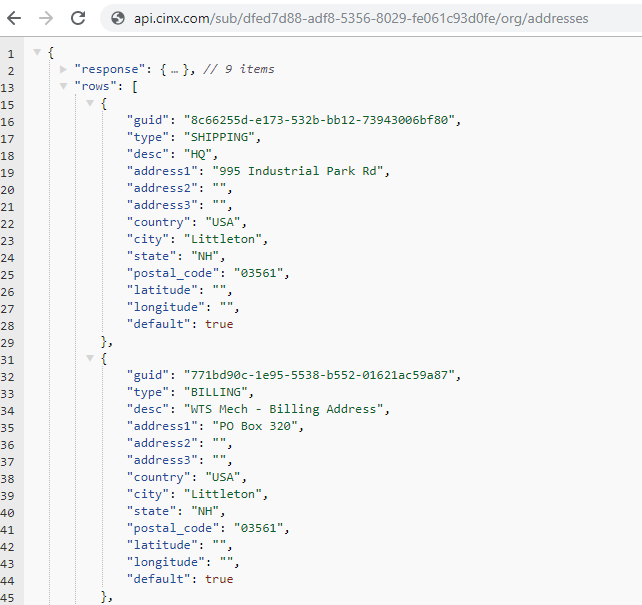
This request will be used to get a company’s list of addresses.

URL Pattern:

{api path}/sub/{B2B GUID}/addresses

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/org/addresses>



### API – GET an Organization’s Customer Type List - DONE

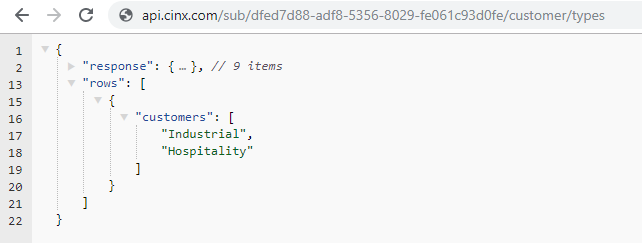
This request will be used to get a company’s list of Customer Type Classifications.

URL Pattern:

{api path}/sub/{B2B GUID}/customer/types

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/customer/types>



### API – GET an Organization’s Vendor Type List - DONE

This request will be used to get a company’s list of Vendor Type Classifications.

URL Pattern:

{api path}/sub/{B2B GUID}/vendor/types

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/vendor/types>



### API – GET an Organization’s Project Type List - DONE

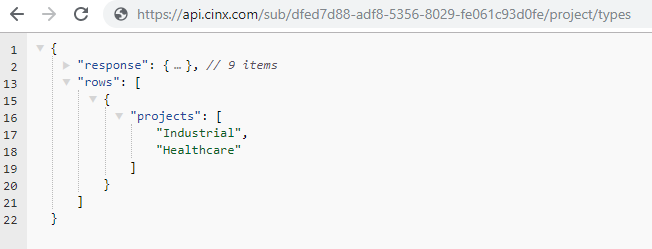
This request will be used to get a company’s list of Project Type Classifications.

URL Pattern:

{api path}/sub/{B2B GUID}/project/types

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/project/types>



### API – GET an Organization’s Invoice Dispute Reasons List - DONE

This request will be used to get a company’s list of Invoice Dispute types.

URL Pattern:

{api path}/sub/{B2B GUID}/dispute/reasons

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/dispute/reasons>



### API – GET an Organization’s Return Reasons List - DONE

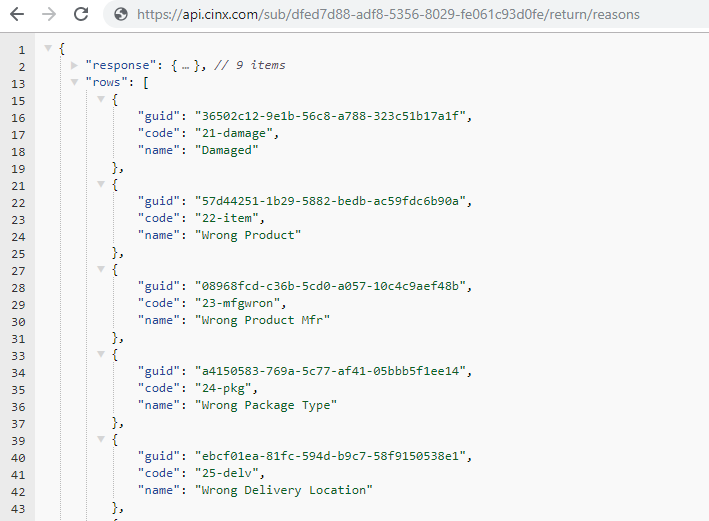
This request will be used to get a company’s list of Return Reasons.

URL Pattern:

{api path}/sub/{B2B GUID}/return/reasons

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/return/reasons>



### API – GET an Organization’s Return Actions) - DONE

This request will be used to get a company’s list of Return Actions.

URL Pattern:

{api path}/sub/{B2B GUID}/return/actions

Sample:

<https://api.cinx.com/sub/dfed7d88-adf8-5356-8029-fe061c93d0fe/return/actions>



### Appendix A - CINX Concepts – DONE (APPENDIX B)

**Application:** is a grouping of pages, functionality, or an external software partner system that interacts

with the CINX platform and is managed via a CINX subscription plan.

**Base Price:** is the initial price for a product or service. It is also commonly referred to as a List Price or

MSRP.

**BOM Item:** a grouping of fields that defines an item once it has been added to a BOM. This structure

inherits from the Item structure and also provides additional fields for defining its use within a project

(quantities, breakdown structure, spec requirements).

**Buy Price:** is the price paid to a vendor for the purchase of a product or service. The Buy Price is often a

discounted price from a previously established base price - which is commonly referred to as a List Price

or a MSRP.

**Catalog:** a collection of items grouped together in a CINX database structure. Catalogs may contain

construction materials, sub-contract services, rental equipment, and other content necessary for the

execution of a construction project or maintenance of a building.

**Catalog Item:** a building material product or construction related service that is stored within a CINX

catalog. Its data structure includes core information about the product including, part identifiers,

descriptions, attributes, dimensional data, and URLs.

**CINX Id:** uniquely identifies an entity within the CINX system. The Id is comprised of three elements

(Entity Type, Domain, and Id) which together will provide the location and Id of the entity.

**Item Labor:** an Item level structure for storing labor unit (installation time) and cost information.

**Item Quantity:** a component of the BOM Item that is used to detail all information about a quantity's

instance including its type, value, uom, and availability information.

**Item Relationship:** a method by which Items can be associated with one another in the CINX system.

The data structure defines the type and identification of an item relationship. Relationship types include

alternates, required, and complimentary items.

**Labor Unit:** a component of an Item's Labor structure which stores installation time elements.

**Manufacturer:** an organization that produces building material products.

**Multiplier:** (also referred to as a discount) is a decimal value that is applied to a price to determine a buy or

sell price.

**Organization:** CINX entity that defines a company or a grouping of individuals (residential customers)

that participates or is referenced within the CINX platform.

**Price Sheet:** a collection of fields that identify an instance of a building product manufacturer's or

service provider's price list stored within the CINX system.

**Project:** a job that is being performed for a customer/client on a building, structure, or other asset (Built

Entity).

**Sell Price:** is the price charged to a customer for a product or service. A Sell Price is normally related to a

Buy Price or an established base price.

**User:** a person who has created a personal CINX account.

### Appendix B - CINX URLs – DONE (APPENDIX A)

Below is a listing of important CINX URLs

**Login** -> <https://secure.cinx.com/login>

**Forgot CINX Password** -> <https://secure.cinx.com/login/reset>

**To set up a CINX account click here** -> <https://secure.cinx.com/join>

**Terms** -> <http://support.cinx.com/terms-of-use-agreement>