PTC LIVEWORX

arena®

USING ARENA'S REST API

SPEAKER(s):

Charles Vernor

Senior Solution Architect – Arena PLM/QMS

May 16, 2023

GETTING STARTED

INTEGRATION OPTIONS

1 Marketplace Solutions

- Provided by PTC Arena or a qualified partner

- Available for common use cases and systems based on market demand

- Support & maintenance provided as Arena and other system change

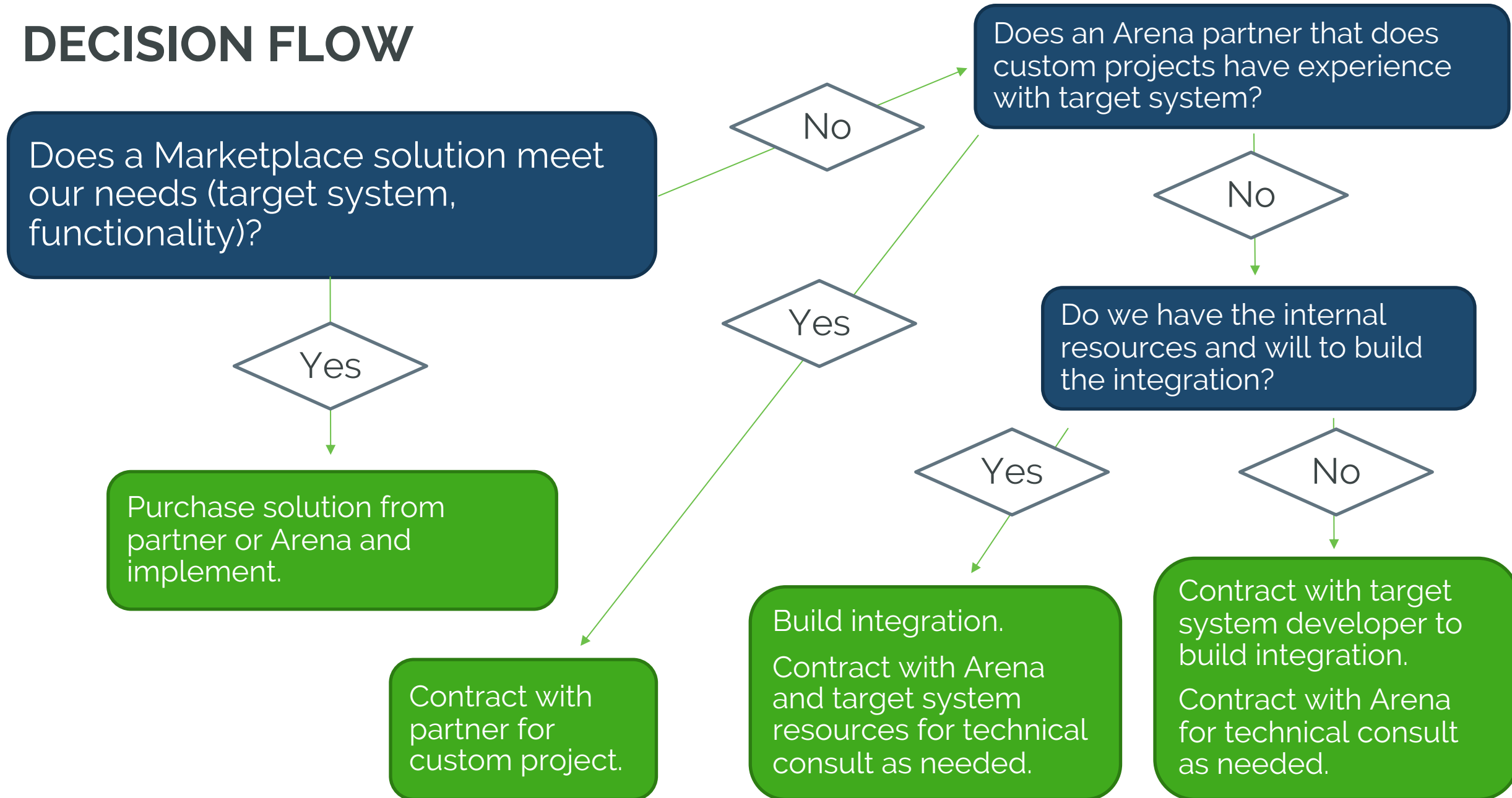- Purchase as product subscription with standard T&Cs

2 Bespoke API Project by an Arena Partner

- Provided by a qualified partner

- Utilizes established integration methods

- Can include support & maintenance

- Partner will quote based on LOE for project scope

3 Develop Your Own Integrations or Contract with a Development Resource (not an Arena Partner)

- Utilize established Arena integration methods – developers need to understand the other system @ integration level, appropriate coding capabilities, and experience with data flows and integrations

- Refer to Arena API and integration documentation, materials, and guides – customer responsible for ensuring developers understand Arena to level needed

- Contract with Arena for additional professional services support for Arena system configuration, integration trigger creation, technical consultation

- Customer owns integration support & maintenance and all testing

DECISION FLOW

Does a Marketplace solution meet our needs (target system, functionality)?

No → **Does an Arena partner that does custom projects have experience with target system?**

No → **Do we have the internal resources and will to build the integration?**

Yes → **Purchase solution from partner or Arena and implement.**

Yes → **Contract with partner for custom project.**

Yes → **Build integration.**

Contract with Arena and target system resources for technical consult as needed.

No → **Contract with target system developer to build integration.**

Contract with Arena for technical consult as needed.

ACCOUNT MANAGER DISCUSSIONS

- Ensure REST API Feature enabled in workspace

- Determine/purchase adequate number of API calls per day

- Purchase 1 or more API user/integration license(s)

- Determine/purchase other software or architecture based on your use case and double-check their requirements/supported formats

SET UP YOUR DEVELOPMENT ENVIRONMENT (POSTMAN)

- Download Postman - the industry standard for making REST API calls during development/testing

- Get Arena Sample API library for Postman – very quick to understand the format and examples for each endpoint via Arena Help > Downloads

- There are other samples in the file using different code languages
  - (ex. Java, Node, Python)

- Gather credentials for API user login (email, password) and Arena workspace ID

LOGIN TO ARENA REST API – GET SESSION ID

- Before you can make any API request, you must create a new API session and **get a Session ID**

- **Only 1 active workspace session is allowed per API user at any time**
  - re-login will invalidate any prior user session(s) on the API server

- Store the retrieved session ID in a code variable and add to the request header for any subsequent API calls

LOGIN TO ARENA REST API – REQUEST/RESPONSE FORMAT

POST Log In **{url}/login**

Sample Request Body

```
{
    "email":"user@customer.com",
    "password":"xxxx",          .
    "workspaceId":123456789
}
```

Note: workspaceId is *optional*, but highly suggested instead of defaulting to active workspace (more reliable)

Sample Response Body

`/login`

The response body will also give you the **request limit** set for the workspace

```
{
    "arena_session_id":"LABS-MXQKenjxWmyUIlJTKwQDfimkZksjpkO0|",
    "workspaceId":123456789,
    "workspaceName":"EveryRoad GPS",
    "workspaceRequestLimit":1000
}
```

COMMON API USE CASES

ARENA'S API INTEGRATION ENGINE OVERVIEW

- Many integrations to Arena require the ability to quickly and easily:
  - obtain bulk data from Arena:
  - load bulk data into Arena
  - be notified when something occurs in Arena, for example when an item is released and becomes effective

- To address these common requirements, we provide the Arena Integration Engine, which is composed of:
  - **Export Engine** to export data from Arena
  - **Import Engine** to import data into Arena
  - **Event Engine** to notify integrations when something occurs in Arena

- Arena also has the **Data Extract** service to export Arena data to other analytics tools
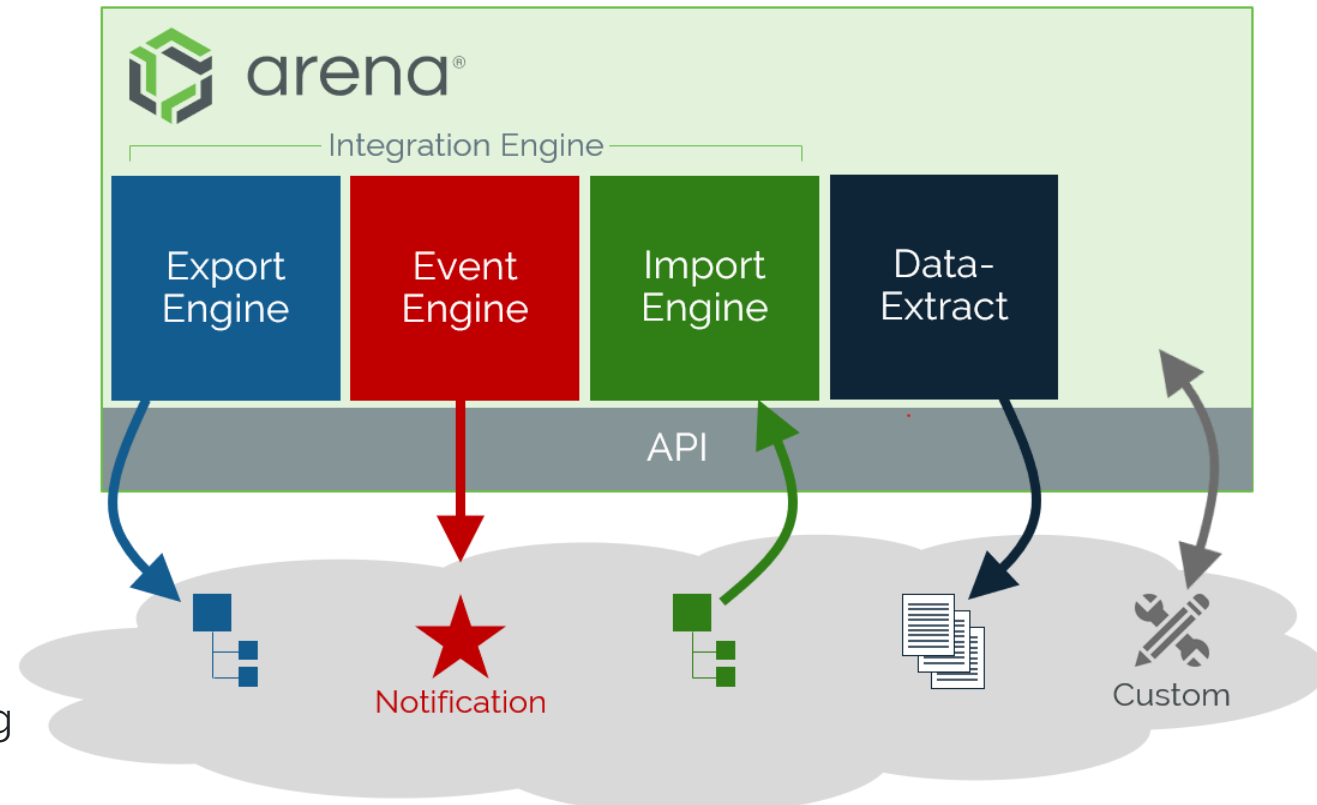
SCENARIO 1: DOWNSTREAM INTEGRATIONS SENDING DATA FROM ARENA

SCENARIO 1: DOWNSTREAM INTEGRATIONS (EVENTS / EXPORT)

- Most common downstream integrations are to ERP and MES systems

- The **preferred** method to consider for this is using the **Events Engine** for outbound integrations

- Based on execution of an Event Trigger, which is a particular action occurring on a record:
  - Supported Arena worlds include **Items, Changes, Requests,** and **Quality Processes**
  - Supported event triggers include **Create, Edit,** and **Workflow** (status change)

- The record is populated into an event queue, which can be queried periodically, processed, and then reconciled in the integration queue

SETTING UP API OUTBOUND EVENT INTEGRATIONS

PROVISIONING THE INTEGRATION

- Provisioning an Events Engine integration is simple and straight-forward. It requires one **API Integration User license** which includes a dedicated **Integration User Account**

- A Solution Architect or a member of the Arena Support team must enable the Event Integration for you first before you can proceed

- Query the **Integration Guid** using the API call shown below

- ❖ **BEST PRACTICE:** *Develop the integration in a Sandbox environment then, once testing is complete, re-deploy in Production*

- ❖ **IMPORTANT:** *GUID values are workspace-specific. Once sandbox testing is complete, provision a new Events Engine integration in your Production workspace and update your integration layer to refer to its new GUID*

GET Integration Guid

API Call: GET /outboundevents

Response: "guid": "Q8SBG49GXZA8RATCVG91"

API OUTBOUND EVENT INTEGRATION FILTERS

APPLYING FILTERS

- With the integration in place, create trigger(s) for data filtering. Triggers support **Items**, **Changes** and **Quality** resources. For an ERP integration, choose **Items**

- Configure the trigger to fire on the **Workflow** action meaning, _when an item is released_

- Apply **Category** and **Lifecycle Phase** filtering to control which items are added to the queue. To filter at the item level, add _conditional filtering_ using a custom item attribute for example, **ERP Transfer** = **Yes**

- **OR** logic filtering is achieved by creating additional triggers

- An integration requires at least one trigger to be operational

- ❖ _**NOTE:** To create a trigger, you must be a **Workspace Administrator**_

Trigger Information

Name	Items - Workflow
Description	
Resource	Items
Action	Workflow
Creator	Doug Johnston (Arena)
Created on	11/21/2022 10:22 AM:37
Last Modified By	Doug Johnston (Arena)
Last Modified On	12/15/2022 01:34 PM:27

Filter

Category	Part, Assembly, Cable Assembly, Finished Good, Kit, Packaging Assembly, PCBA, Printed Circuit Board Assembly, Product Assembly, Subassembly, Electrical, Battery/Charger, Cable, Capacitor, Connector, Crystal, Detector, Diode, Fuse, IC, Inductor, LCD, Misc. Electrical, Oscillator, Power Control, Printed Circuit Board, Resistor, Switch, Transducer, Transformer, Transistor, Mechanical, Adhesive, Consumable, Dowel Pin, Enclosure, Fabricated Metal, Fabricated Other, Fabricated Plastic, Fixture, Gasket, Label, Misc. Mechanical, Nut, O-Ring, Optic, Packaging, Retaining Ring, Screw/Bolt, Server Rack, Shield, Spacer, Spring, Standoff, Tape, Tool, Tubing, Washer, Software, Firmware, Object Code, Source Code
From Lifecycle Phase	Unreleased, Abandoned, In Design, Deprecated, In Production, Obsolete
To Lifecycle Phase	Abandoned, In Design, Deprecated, In Production, Obsolete
Conditions	

#	ATTRIBUTES	OPERATOR	VALUE	LOGIC
1	ERP Transfer	equals	Yes	

API OUTBOUND EVENT INTEGRATIONS

FILTERED ITEMS ARE AUTOMATICALLY ADDED TO THE QUEUE

- Here, <u>a single item</u> was released using *Revision Control*. Seconds later, *trigger filtering* automatically added one **Outbound Event** (the release of that item) to the queue (**To Be Reconciled**)

- If a *Change Order* had been used to release <u>multiple items</u>, we'd still see one **Outbound Event** (representing the release of the Change Order) but this time, we'd see all items on that Change that passed trigger filtering

- Use the API call shown below to return events currently in the queue

- ❖ *NOTE The <u>maximum</u> number of events you can read at a time is **400**. This value is known as the **limit**. Since the queue may contain more than 400 events, start with **offset=0** then, for each iteration, set **offset=offset+400***

- Enumerate all events, capturing each event's **Event Guid** and **Date/Time Created**

- ❖ *IMPORTANT: You must read all events in the queue to determine processing order. Events need to be processed in* creationDateTime *order*

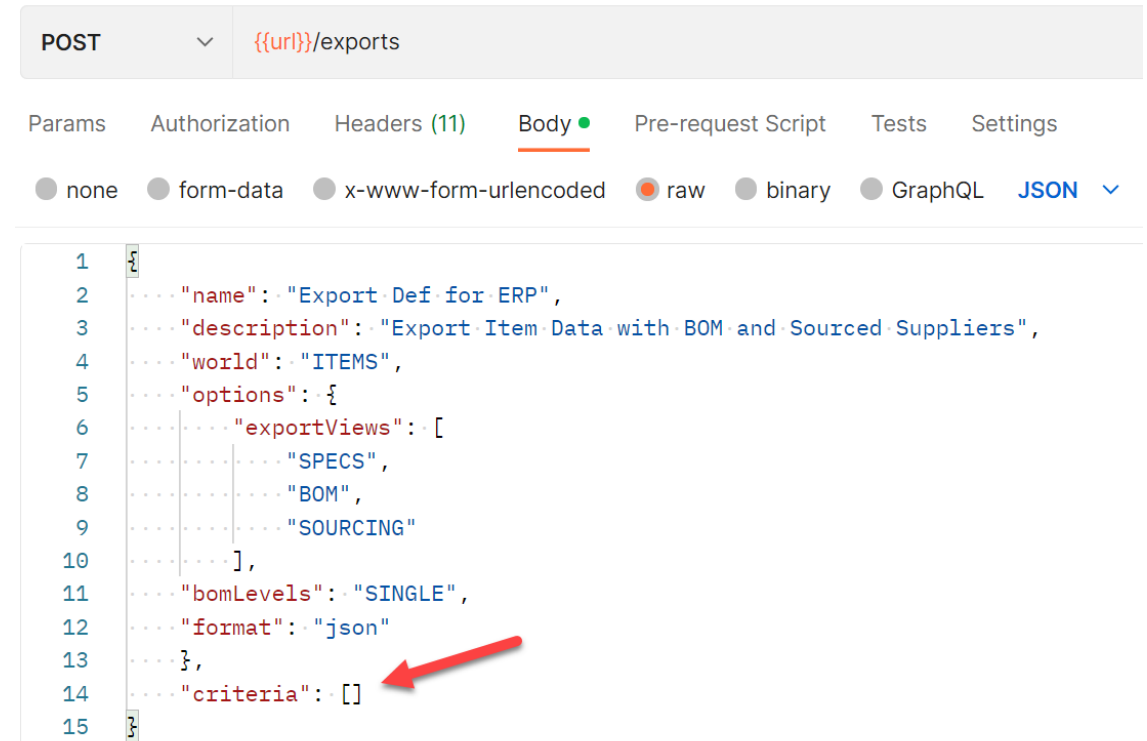| **GET Integration Event Guids**	
API Call:	GET /outboundevents/<Integration Guid>/events?limit=400&offset=<varOffset>
Response:	"guid": "GYI16UZ6NP00J2L4NYM1" <u>and</u> "creationDateTime": "2022-12-12T09:36:50Z"

API OUTBOUND EVENT INTEGRATIONS – EXPORT DEFINITIONS

CREATE AN EXPORT DEFINITION

- **Export Definitions** <u>dramatically reduce</u> the number of required API calls

- Create 1:n export definitions for different purposes. They are stored permanently in your workspace and can be referenced at any time.

- This export definition is designed to retrieve *Item Specification, Bill-of-Material* and *Source Relationships* for items

- The selected output format is **JSON** (**CSV** is also supported)

- Executing this call returns an **Export Definition Guid**

- ❖ *BEST PRACTICE: leave **criteria** blank **[]** to dynamically export data for <u>any item</u>. If criteria is specified, it can only be used to export data for that "fixed" criteria*

- ❖ *IMPORTANT: GUID values are workspace-specific. Once sandbox testing is complete, you'll need to re-create the Export Definition in your Production workspace and update your integration layer to refer to its new GUID*

- ❖ *NOTE: leverage default values for <u>required</u> ERP-specific attributes to avoid having to create custom item attributes in Arena*

POST	⌄	{{url}}/exports

Params Authorization Headers (11) **Body** ● Pre-request Script Tests Settings

⚪ none ⚪ form-data ⚪ x-www-form-urlencoded 🔴 raw ⚪ binary ⚪ GraphQL **JSON** ⌄

```
1   {
2       "name": "Export Def for ERP",
3       "description": "Export Item Data with BOM and Sourced Suppliers",
4       "world": "ITEMS",
5       "options": {
6           "exportViews": [
7               "SPECS",
8               "BOM",
9               "SOURCING"
10          ],
11      "bomLevels": "SINGLE",
12      "format": "json"
13      },
14      "criteria": []
15  }
```

API OUTBOUND EVENT INTEGRATION – GET ITEMS

GET ITEMS FOR CURRENT EVENT

- Use the API call shown below to return items for the current event

  ❖ *NOTE The <u>maximum</u> number of items you can read at a time is **400**. This value is known as the **limit**. Since an event may contain more than 400 items, start with **offset=0** then, for each iteration, set **offset=offset+400***

- Enumerate all **Item Guid** (item.guid) and associated **Item Event Guid** (guid) values

- We'll need to reference <u>all</u> **Item Guid** values in our next API call and later, <u>each individual</u> **Item Event Guid** to clear each item from the queue after being processed

Integration
Events Engine Integration

To Be Reconciled   Reconciled

1 updates to reconcile in Events Engine Integration

☐	#	NAME	NAME/TITLE
☐		**Outbound Event: GYI16UZ6NP00J2L4NYM1**	Trigger Item Workflow
☐	1	451-00003	RES .10 OHM 1/8W 5% 0805

GET Integration Event Item Guids

API Call: GET /outboundevents/<Integration Guid>/events/<Event Guid>/items?limit=400&offset=<varOffset>

Response: "item.guid": " XFZINBGN46PXGUMNTN5O"   <u>and</u>   "guid": " R9TCH5AHY0BATCV1EB2X"

API OUTBOUND EVENT INTEGRATIONS – EXPORT RUNS

INITIATE AN EXPORT RUN BASED ON ITEMS FROM THE CURRENT EVENT

- This API call refers to the **Export Definition Guid** and the **Item Guid** (item.guid) values enumerated in the prior call

- When an event contains more than one item, the syntax is "value": ["<Item Guid 1>", "<Item Guid 2>"…]. In doing so, a single call can be used for all items in the current event

- ❖ *NOTE: Export Runs are limited to 20,000 items or 5,000 items when BOM and FILES views are included*

- Execution of this call will return two GUIDs required to retrieve the json **content** associated with the run. In this example, the following two values were returned in the response:

  Run GUID: "guid": "O6Q9E27EVXDSBULL2G5E"

  File GUID: "files.guid": "WEYHMAFM35O4N6VR94WH"

POST ▾ {{url}}/exports/DVFY3RW3KM2I1KYB7G8G/runs

Params   Authorization   Headers (12)   Body ●   Pre-request Sc

○ none   ○ form-data   ○ x-www-form-urlencoded   ● raw   ○ bin

```
1  {
2      "criteria": [[
3          {
4              "attribute": "guid",
5              "operator": "IS_IN",
6              "value": ["XFZINBGN46PXGUMNTN5O"]
7          }
8      ]]
9  }
```

API OUTBOUND EVENT INTEGRATIONS – GET CONTENT

GET CONTENT FROM EXPORT DEFINITION RUN

- This call returns a single json response with the views specified in the definition <u>for all items</u> specified in the prior call

GET Export Definition Run Content	
API Call:	GET /exports/<Export Definition Guid>/runs/<Run Guid>/files/<File Guid>/content

API OUTBOUND EVENT INTEGRATIONS – GET CONTENT

ADDITIONAL PROCESSING OF RETURNED CONTENT

- By design, the returned json content includes one **Item** element for **each item in the event** <u>plus</u> additional **item** element(s) for **each unique child referenced in their BOMs**

- It's also designed to return <u>all children</u> in each **BOM** element even if the child would otherwise have been excluded by trigger filtering. So, we need a way to process only <u>event item elements</u> and know which children to add to their BOMs

- The logic is quite simple:

  - Within the json response, only process **item** elements specified in the current event. For each of those **item** elements, process their **spec** and **sourcing** sections. *NOTE: Skip all other **item** elements since they are not part of the current event*!

  - After the item's **spec** and **sourcing** sections have been processed <u>for all items in the event</u>, we're ready to process their **BOM** sections. Since Arena ensures that children are released <u>before or at the same time</u> as their parents, and since we processed the item **spec** sections earlier, any referenced BOM children not already in ERP should <u>**not**</u> be added to the item's BOM. To recap:

    - Check to see if the child is in ERP. If it is, add it to the item's BOM. If the child does not exist in ERP, do not attempt to add it to the BOM

API OUTBOUND EVENT INTEGRATIONS - RECONCILE

RECONCILING THE ITEM

- As each item in the event is successfully processed (spec, sourcing and BOM), use the call shown below to clear it from the queue

❖ *Note an alternative form of the call shown below allows you to reconcile at the <u>event</u> level as opposed to the <u>item event</u> level*

- Once cleared, shown at right, the item event has been moved from the **To Be Reconciled** to the **Reconciled** view

❖ *Note the reconciled view retains the last three months of reconciled events*

Reconcile Event:	
API Call:	PUT /outboundevents/**<Integration Guid>**/events/**<Event Guid>**/items/**<Item Event Guid>**

Integration
Events Engine Integration

To Be Reconciled   Reconciled

0 updates to reconcile in Events Engine Integration

☐	#	NAME

There are no updates that need to be reconciled.

Integration
Events Engine Integration

To Be Reconciled   **Reconciled**

7 updates recently reconciled in Events Engine Integration

☐	#	NAME	NAME/TITLE
☐		**Reconciled 10/17/2022 08:23 AM:20** By Doug Johnston (Arena)	
☐	1	451-00003	RES .10 OHM 1/8W 5% 0805

PUT	∨	{{url}}/outboundevents/Q8SBG49GXZA8RATCVG91/events/GYI16UZ6NP00J2L4NYM1/items/R9TCH5AHY0BATCV1EB2X

Params   Authorization   Headers (11)   **Body** ●   Pre-request Script   Tests   Settings

○ none   ○ form-data   ○ x-www-form-urlencoded   ● raw   ○ binary   ○ GraphQL   **JSON** ∨

```
1  {
2      "reconciled": true
3  }
```

AUTOMATED / MANAGED OUTBOUND INTEGRATIONS

- Arena also has options for **automated** and **managed outbound integrations (Export Engine).**

- In **automated** integrations, an integration logs into Arena at regular intervals and reconciles its product record to match the Arena product record
  - Can set up for item release or change release

- In **managed** integrations, an Arena user manually reconciles the product record in the target system with the product record in Arena by entering updates from Arena into the target system
  - Can be set up for item release only

AUTOMATED / MANAGED OUTBOUND INTEGRATIONS: DASHBOARD TAB

- When outbound export events occur, the Integrations tab on the users main Dashboard will be populated

- One or more integration admins can be assigned to monitor the export queue

- Any exports still in queue will email the integration admin after a defined period

SCENARIO 2: REPORTING AND ANALYTICS

SCENARIO 2: REPORTING / ANALYTICS (DATA EXTRACT)

- Arena Data Extract lets you connect your product data to business intelligence, analytics, and other reporting tools for greater insight into PLM data trends and bottlenecks in your process

- The Data Extract service creates a ZIP file that can be downloaded and consumed offline or in another system

- Arena Admin must set up a Data Extract definition within Workspace Settings for the Arena World to be included and the scheduled run time
  - **Note:** the data extract service currently runs <u>at most, 1x per day</u> on a schedule. More frequent data gathering will require another method to manually get the desired data

- Can retrieve the latest data extract results via the REST API

DATA EXTRACT PRE-REQUISITE: EXPORT DEFINITION SETUP

- Arena Admin can set up 1 Export Definition with a workspace. The setup can be found in Workspace Settings under the DataExtract tab

- All export definitions are set to Full data extract only. You can set up a dedicated data extract email and password, which can be used to connect to REST API

- Under the filtering section, select the Arena world/objects to export

- Under the schedule section, set the start date and time, and the recurrence period

GET LATEST DATA EXTRACT EXPORT ZIP FILE VIA API

■ **Call the following endpoints (in order) and pass the GUIDs to the next query**

1. **GET /extracts - Get the extract definition GUID**

2. **GET /extracts/<GUID #1>/runs/latestCompleted**
   - Get the result latest export GUID (ie. since the previous day)

3. **GET /extracts/<GUID #1>/runs/<GUID #2>/files**
   - Get the result file metadata GUID

4. **GET /extracts/<GUID #1>/runs/<GUID #2>/files/<GUID #3>/content**
   - Get ZIP file contents from Data Extract and read the CSV data into your target application:
      - download raw file binary content
      - decode ZIP file, unzip the contents
      - filter out non-CSV files (only keep the actual Arena data)
      - parse out special characters that break CSV import (line feeds, double quotes, slashes, etc.)

SCENARIO 3: UPSTREAM INTEGRATIONS LOADING DATA INTO ARENA

SCENARIO 3: LOAD DATA INTO ARENA (IMPORT) NEW FEATURE!

- New feature coming in R78 release (Spring 2023) for pushing Items and BOMs into Arena using the **Import Engine**
  - Useful for CAD / PDM integrations, or getting data from vendors / manufacturers
  - Supports XLSX and CSV file imports (and data validation)

- This can be set up in multiple modes (Create, Modify, Replace) based upon the customer use case:
  - **Bulk Create** new items & BOMs
  - **Bulk Edit** existing items & BOMs
  - **Bulk Overwrite**: edit if data exists, otherwise create
  - **Replace BOM**
  - Internally manages BOM redline add, remove, edit
  - Internally manages item working revisions

Item Master

Bill of Materials
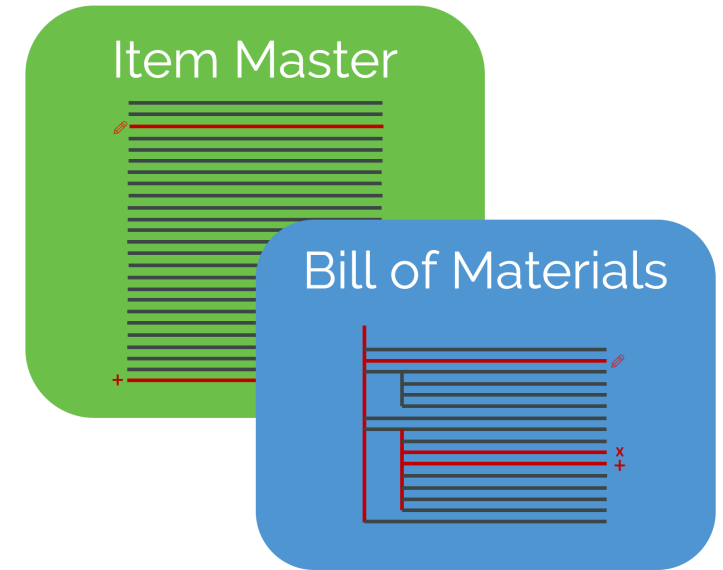
SETTING UP ITEM IMPORT ENGINE

- The import capability is managed with two main resources in a similar manner as the Export Engine:
  - An **Import Definition** must be setup that defines the resource (Item Specs, Item BOM, etc.), mode (Create, Overwrite, etc.), attribute mapping and other options. The intent is to define this for a specific integration, then reuse it to perform the imports.
  - An **Import Run** executes the import. It references the import definition and specifies the content to be imported. You can also control when to commit the imported data

- **Note:** The Import Engine does not affect the existing Arena UI Import capability. The Arena UI Import capability will remain separate and supported. Access roles are also not supported within this feature

- Be sure to check out the announcement and release notes from Arena when this feature is released!

SCENARIO 4: CUSTOM INTEGRATIONS & EVENTS

SCENARIO 4: CUSTOM INTEGRATIONS & EVENTS: (CUSTOM)

- Arena custom integrations and automations (i.e.. creating objects, change status, etc.) can greatly vary in their requirements and logic
  - Needs detailed planning and discussion to determine the best API approach

- Can use API strategies from the prior use cases:
  - events engine to get hooks into certain actions when a prerequisite state is reached
  - Or use specific endpoint(s) to GET data and automate actions when appropriate

- **Note:** Arena's API is REST-based (not Web Socket-based) so API integrations should avoid trying to design real-time automations
  - API requests must be made on a periodic basis to check for data or state updates
  - Keep an eye on the number of API requests consumed per day