# Aras Innovator 12

## Configurable InBasket

***Document #:* *12.0.02019050601***

***Last Modified:*** *5/12/2020*

# Copyright Information



Aras Corporation
100 Brickstone Square
Suite 100
Andover, MA 01810

**Phone:** 978-806-9400
**Fax:** 978-794-9826

**E-mail:** Support@aras.com

**Website:** https://www.aras.com/

# Table of Contents



 

# Send Us Your Comments

Aras Corporation welcomes your comments and suggestions on the quality and usefulness of this document. Your input is an important part of the information used for future revisions.

- Did you find any errors?
- Is the information clearly presented?
- Do you need more information? If so, where and what level of detail?
- Are the examples correct? Do you need more examples?
- What features did you like most?

If you find any errors or have any other suggestions for improvement, indicate the document title, and the chapter, section, and page number (if available).

You can send comments to us in the following ways:

**Email:**
Support@aras.com
Subject: Aras Innovator Documentation

Or,

**Postal service:**
Aras Corporation
100 Brickstone Square
Suite 100
Andover, MA 01810
Attention: Aras Innovator Documentation

Or,

**FAX:**
978-794-9826
Attn: Aras Innovator Documentation

If you would like a reply, provide your name, email address, address, and telephone number.

If you have usage issues with the software, visit https://www.aras.com/support/

# Document Conventions

The following table highlights the document conventions used in the document:

Table 1: Document Conventions

| Convention | Description |
|---|---|
| **Bold** | Indicates the names of menu items, dialog boxes, dialog box elements, and commands.<br>Example: Click **OK**. |
| `Code` | Code examples appear in `courier` font. It may represent text you type or data you read. |
| `Yellow highlight` | Code highlighted in yellow draws attention to the code that is being indicated in the content. |
| `Yellow highlight with red text` | Red text highlighted in yellow indicates the code parameter that needs to be changed or replaced. |
| *Italics* | Reference to other documents. |
| **Note:** | Notes contain additional useful information. |
| **Warning** | Warnings contain important information. Pay special attention to information highlighted this way. |
| Successive menu choices | Successive menu choices appear with a greater than sign (-->) between the items that you will select consecutively.<br>Example: Navigate to **File** --> **Save** --> **OK**. |

# 1 Overview

My InBasket, the destination for Workflow assignments, is now configurable, customizable, and searchable. The new features are:

- Search and filter InBasket: You can search and filter InBasket Items. All the search modes: Simple, Advanced, AML and Hide Search Criteria are now available for InBasket items.
- Saved Search: You can save searches that can be used to retrieve searches that have been created previously.
- Configuration (administrators only): You can add additional source items to the InBasket query, beyond the currently supported items (Workflow Activities Items).
- Customization (administrators only): You can customize the InBasket to add columns to the InBasket display, hide existing columns, re-label columns, and change column widths and order.

**Warning** Make sure that you do not use any of the InBasket tasks as related source id or data source or data property in the relationship grid.

# 2 Filtering Tasks/Assignments

## 2.1 Searching and Filtering Tasks/Assignments

You can now use the search toolbar and the various search modes to search and filter assignments in the InBasket. Using the search toolbar, you can specify different search criteria and filter assignments in the search grid. You can also use these search modes: Hide Search Criteria, Simple Search, Advanced Search, and AML Search to create different search criteria.

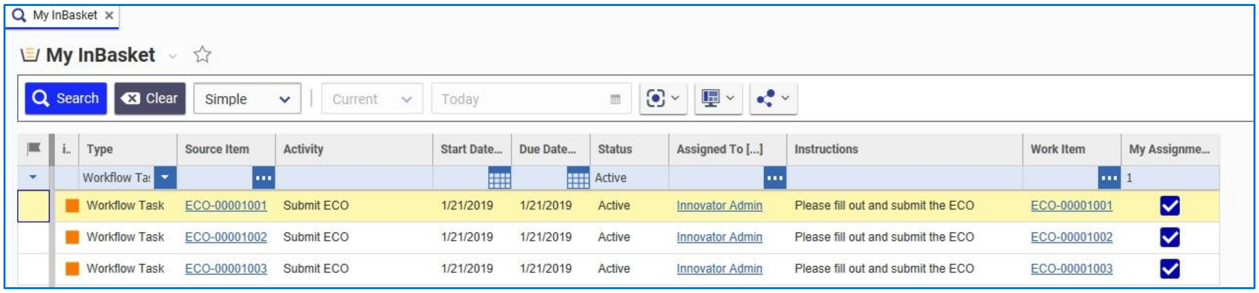


Figure 1.

To perform a simple search, type search terms into the blue search bar. You can use * or % as wildcards.

For example, the term ‘a*b’ in the Activity column searches for any Item beginning with ‘a’ and ending with ‘b’. You can use * on its own to select non-blank values. For columns with checkboxes, use 0 or 1, where 0 means not-checked.

## 2.2 Saved Search

The Saved Search enables you to retrieve searches that have been created previously.

Whenever you create a search, the Search Mode and search terms are saved automatically, when a user returns to an item the last search is automatically restored. After you have made a Search, you may save the Search, giving it a name, so that it can be retrieved and re-used, with or without modification, in the future. Each user can execute Saved Searches they have made in addition to searches shared by other users.

## 2.3 End User Configurability

You can configure the InBasket grid similar to other search grids. The configuration includes rearranging, resizing, and hiding or showing columns in the grid. The changes you make are stored as preferences.

# 3 Claiming and Unclaiming a Task

You can claim ( ) or unclaim a task if you have the required access rights to the task. When you claim a task, it prevents other users from claiming the task. After you unclaim the task, you or another user may claim the task to submit additional changes to the task.

Table 2: Different icons for search criteria and their descriptions.

| Icon | Description |
|---|---|
|  | Claimed by you. Other users are prohibited from claiming the task you have claimed. |
|  | Claimed by others. You cannot claim or complete a task claimed by another user. |
|  | Claimed by Anyone. This icon appears in the grid. |

## Claiming a Task

When you claim ( ) a task, other users are prevented from claiming that task. Similarly, you cannot claim a task claimed by another user.

There are multiple ways to claim a task. From the grid, select the task to be claimed, and then opt for one of the following choices:

- Click on the down arrow in the first grid column and select from the context menu.
- Use the right mouse button (right-click) to open the context menu and click **Claim Task**.

## Unclaiming a Task

You can only unclaim a task that is claimed by you. You cannot unclaim a task that is claimed by another user. From the grid, right-click on the task to be unclaimed, and select **Unclaim Task** from the context menu.

 

# 4 Completing a Task

## 4.1 Completing Workflow Activity/Task

When a Workflow Activity becomes active, it is immediately sent to all of the assignees, specified in the **Assignments** tab of the Activity. The identities who receive these activities can then view all of their assigned activities in their InBasket.

### To update Workflow Activity

1. Log on to Aras Innovator.
2. Navigate to **My Innovator** --> **InBasket**.
3. Select the Activity to be updated from the main grid, right-click and select **Complete Task** from the context menu that pops up.

   The **Workflow Activity Completion** window appears.

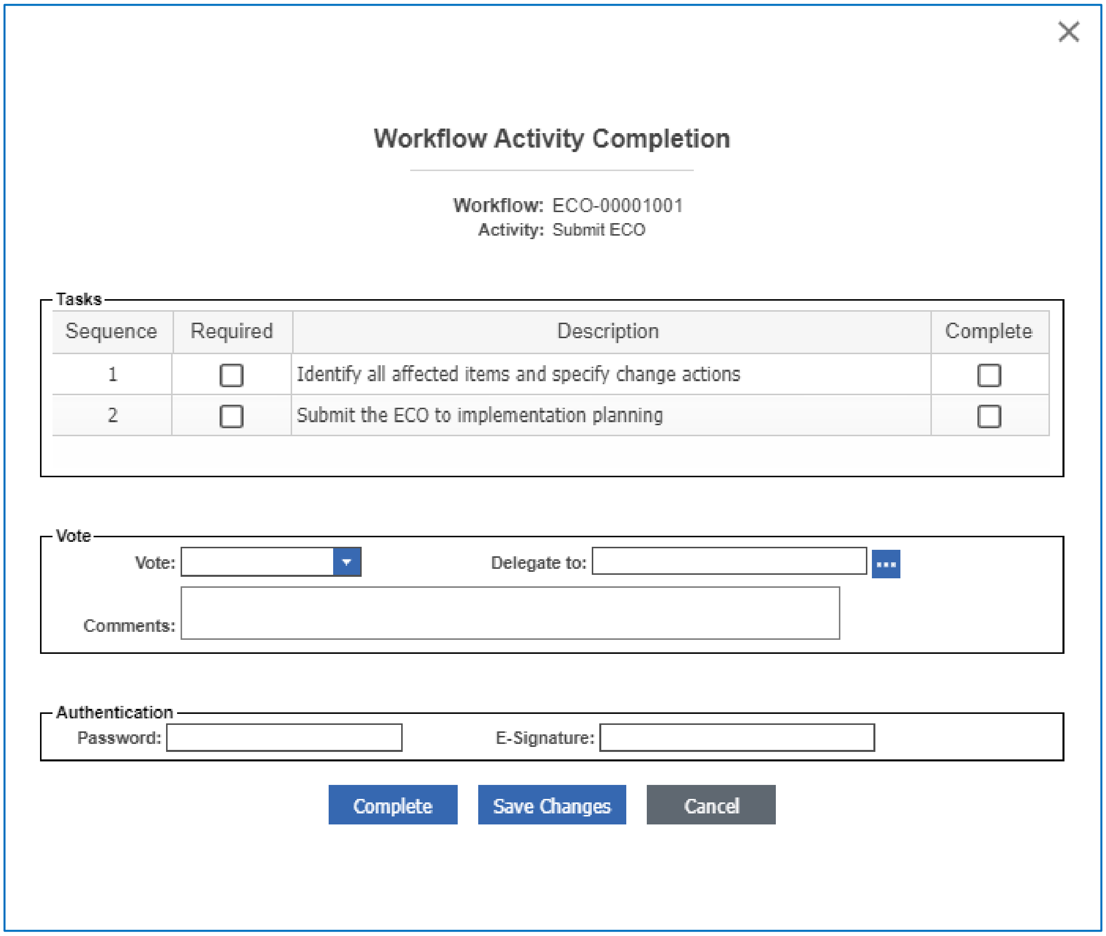


Figure 2.

 

4. Complete the Activity:

    a. Select the **Complete** check box for the tasks that you have completed.

| | |
|---|---|
| **Note:** | If the **Required** check box is selected for a task, it implies that it is mandatory for you to complete the task. If you try to vote and complete the Workflow Activity without completing the Required Task, an error appears. |

    b. Submit your vote by selecting the appropriate choice from the Vote drop-down. The voting options are dynamically built based on the Workflow Process definition.

    c. If required, enter your Password and/or E-Signature for authentication. Depending on the configuration by the administrator, you may have to provide either a Password or an E-Signature or both for authentication purposes.

    The Workflow Activity Completion provides a location to collect Authentication such as passwords and e-signatures and a location to vote an Exit Path. The result of this vote is then used by the Workflow Process to determine the next set of actions to perform.

5. Click one of the following:

    - **Complete**: Checks the activity (to make sure that all required tasks have been checked, variables entered, etc.), marks the activity as complete, and continues the Workflow.

    - **Save Changes**: Saves all information entered on the form, but does not process the activity.

      This is a useful option for activities having a long list of tasks, where an assignee wishes to keep track of what has been completed. Also, if the activity is assigned to a group, then different members of the group may work on different tasks. Once a task is completed, marked as complete, and saved, all members of that group see this information on their Worksheets, thereby reducing duplication of effort.

    - **Cancel**: Closes the form without saving any of the changes made to it since it was opened or last saved.

    Before the Activity moves to the next state, the Activity needs to be unlocked.

# 5 Configuring InBasket

## 5.1 Viewing Others' Assignments/Tasks

Users can now easily view tasks that are not assigned to them directly. Users can search for tasks in other users' **InBasket**.

By default, only members of the *Administrators* group have the permissions necessary to see others' tasks.

### Viewing Others' Assignments

1. Log on to Aras Innovator with administrative privileges.
2. Select **My Innovator** --> **My InBasket** in the TOC. Click the magnifying glass icon  to access the Search grid.
3. Clicking the My Inbasket icon in the TOC causes the following menu to appear:


Figure 3.

By default, the search grid displays tasks assigned to you with 'Active' status.

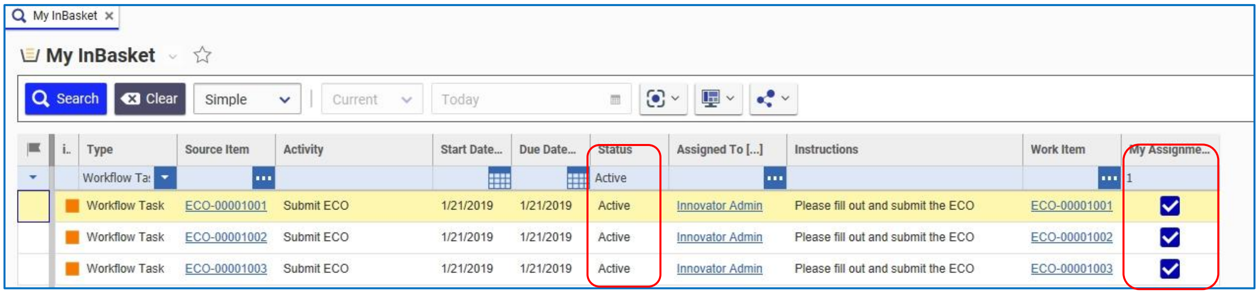


Figure 4.

4. Click  in the grid to filter the search criteria. A menu similar to the following appears:

 

aras

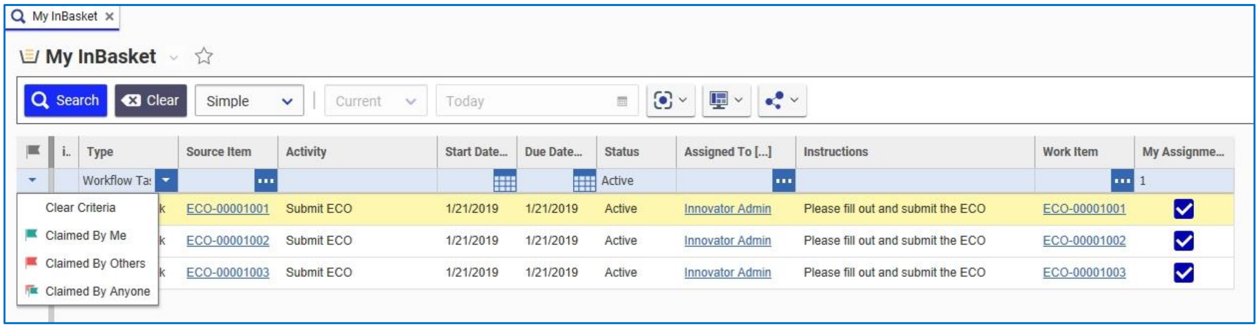


Figure 5.

- **Clear Criteria** erases any search criteria that you entered.
- **Claimed by Me** displays a list of the tasks that you have claimed.
- **Claimed by Others** displays a list of tasks that have been claimed by other users.
- **Claimed by Anyone** displays tasks claimed by you along with tasks claimed by other people.

5. Click 

to view the search results. The InBasket search grid now displays all the tasks.

# 5.2 Creating InBasket Task ItemType

By default, the configurable InBasket supports Workflow Tasks. However, you can now create and add new Task/Activity ItemTypes to the InBasket Task.

Let us consider an example of creating a Meeting Task ItemType and adding it to the InBasket Task Item. Creating InBasket Task ItemType involves four main procedures. The procedures are described in the following sections.

## 5.2.1 Creating an ItemType

1. Log on to Aras Innovator with administrative privileges.
2. Select **Administration > ItemTypes**. The following menu appears:

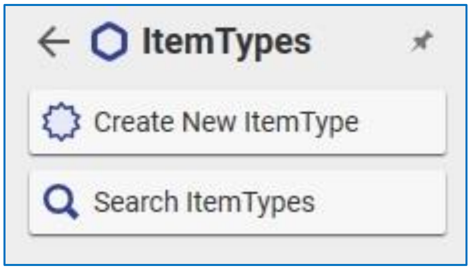


Figure 6.

3. Select **Create New ItemType**. The ItemType window appears.

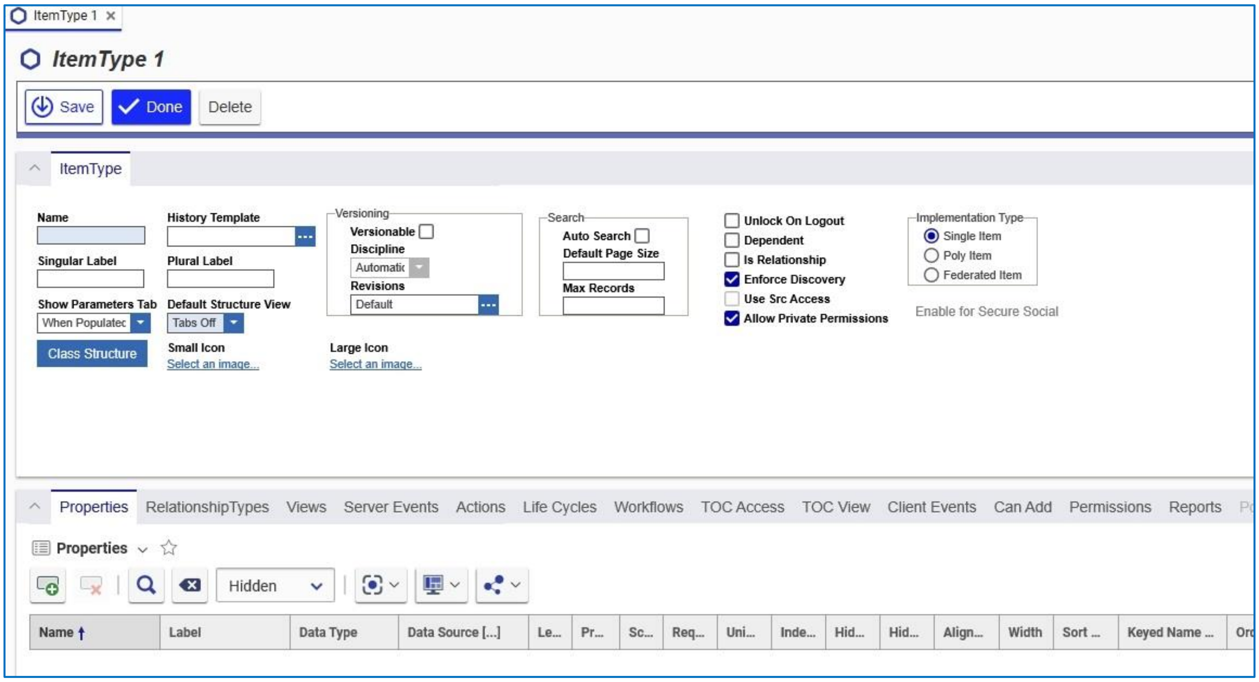


Figure 7.

4. Enter Meeting in the Name field and click 

. In the **Properties** tab, click the **New Property** icon  and add the following properties for the **Meeting** ItemType.

| Property Name | Label | Data Type | Data Source | Length |
|---|---|---|---|---|
| assigned_to | Assigned To | item | Identity | |
| container | Source Item | item | Document | |
| container_type_id | Container Type | item | ItemType | |
| due_date | Due Date | date | | |
| instructions | Instructions | string | | 1024 |
| item | Work Item | item | Part | |
| item_type_id | Item Type | item | ItemType | |
| my_assignment | My Assignment | boolean | | |
| name | Activity | string | | 128 |
| start_date | Start Date | date | | |
| status | Status | String | | 64 |
| icon | Icon | Image | | |
| language_code_filter | Language Code Filter | String | | 3 |

5. In the **TOC Access** tab, add users (Identities), who should be able to see the ItemType in the TOC Pane so that they can select the Item.
6. In the **Can Add** tab, add users (Identities), who can add a new Item of this type to the database.
7. In the **Permissions** tab, add users (Identities) to indicate the access rights to the ItemType.
8. Click the **Save** and **Done** buttons to save and unlock.

### 5.2.2 Configure InBasket Task ItemType (Meeting Task ItemType)

1. Log on to Aras Innovator with administrative privileges.
2. Navigate to **Administration** --> **ItemType**.
3. From the grid, search for **InBasket Task** Item, double-click on it and then click 

to open the Item for editing.
4. In the **Poly Sources** tab, click  to add a new related Item. The ItemType Search dialog box appears:

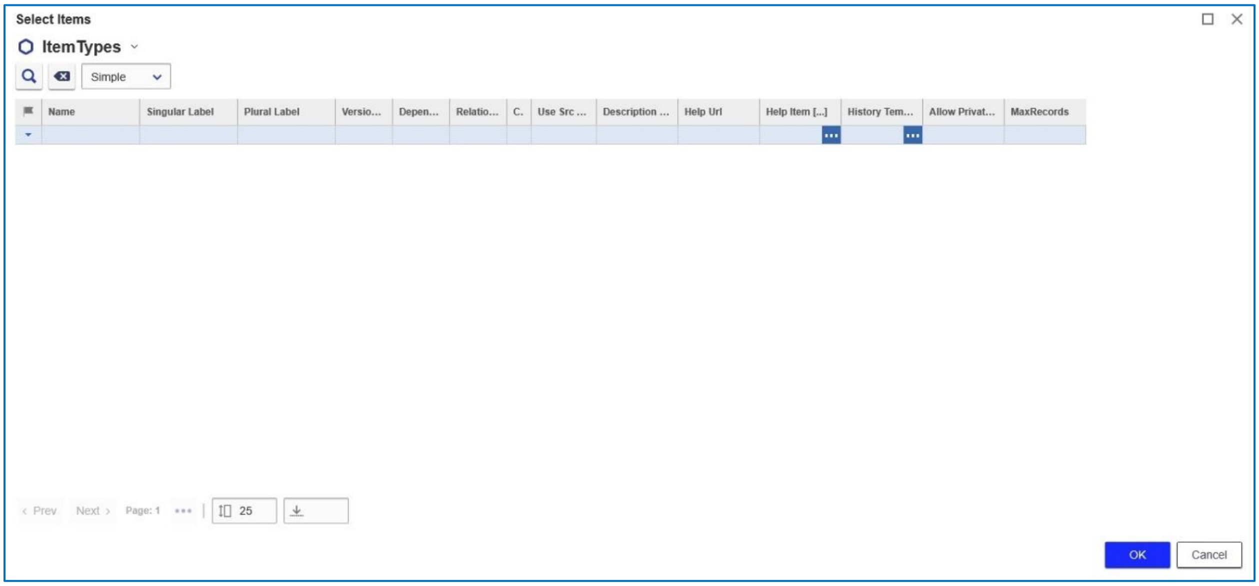


Figure 8.

5. Enter **Meeting** in the Name field to search for the Meeting ItemType. In our example, we create a Meeting Task Item as a related Item with the following properties on the **Properties** tab:

| Property Name | Label | Data Type | Data Source | Length |
|---|---|---|---|---|
| assigned_to | Assigned To | Item | Identity | |
| container | Source Item | Item | NULL (or any value) | |
| container_type_id | Source Item | Item | ItemType | |
| due_date | Due Date | Date | | |
| instructions | Instructions | String | | 1024 |

| Property Name | Label | Data Type | Data Source | Length |
|---|---|---|---|---|
| item | Work Item | Item | NULL (or any value) | |
| item_type_id | Item Type | Item | ItemType | |
| my_assignment | My Assignment | Boolean | | |
| name | Activity | String | | 128 |
| start_date | Start Date | Date | | |
| status | Status | String | | 64 |
| icon | Icon | Image | | |
| language_code_filter | Language Code Filter | String | | 3 |

6. Add the following forms to the **Views** tab:

| Form Name | Type | Description |
|---|---|---|
| Meeting Task | Default | Form to be displayed to the user for actions such as View, Edit, and so on related to the Meeting ItemType. |
| Meeting Task Complete | Complete | An empty form/or a placeholder used when the Meeting Task is Complete. |

7. In the **TOC Access** tab, add users (Identities), who should be able to see the ItemType in the TOC Pane so that they can select the Item.
8. In the **Can Add** tab, add users (Identities), who can add a new Item of this type to the database.
9. In the **Permissions** tab, add users (Identities) to indicate the access rights to the ItemType.
10. Click **Class Structure**. Add a node for Meeting Task.
11. Click Save and Done to save and unlock the ItemType.

### 5.2.3 SQL Table and View Execution

1. Log on to Aras Innovator with administrative privileges.
2. Navigate to **Administration --> SQLs**.
3. Create a SQL View to drop the Meeting table:
   a. Click **File --> New** or use the right mouse button (right-click) and select **New SQL** from the context menu.

b. The SQL window appears. Configure the following properties:

**Name** = Meeting_Task_Step1_Drop
**Type** = Table
**Execution Flag** = Manual
**SQL**:

```
IF  EXISTS (SELECT * FROM sys.tables WHERE object_id =
OBJECT_ID(N'[innovator].[Meeting_Task]'))
DROP TABLE [Meeting_Task]
IF  EXISTS (SELECT * FROM sys.views WHERE object_id =
OBJECT_ID(N'[innovator].[Meeting_Task]'))
DROP VIEW [Meeting_Task]
```

c. Click **Save** and click **Unlock** to save and unlock the SQL Item.

d. Navigate to **Actions --> SQL Execute** in the SQL window.

e. Verify and ensure that table [innovator].[Meeting_Task] is dropped from your database.

4. Create a temporary view for Meeting Task ItemType

a. Navigate to **Administration --> SQLs**.

b. Click **File --> New** or use the right mouse button (right-click) and select **New SQL** from the context menu.

c. The SQL window appears. Configure the following properties:

**Name** = Meeting_Task_Step02_Create_Tmp_View
**Type** = View
**Execution Flag** = Manual

**SQL**:

```
CREATE VIEW [Meeting_Task_Step02_Create_Tmp_View]
AS
  SELECT STUFF(MEETING_TASK.id, 13, 1, '0') AS ID,
         MEETING_TASK.ASSIGNED_TO    AS ASSIGNED_TO,
         MEETING_TASK.ITEM           AS ITEM,
         MEETING_TASK.Item_TYPE_ID   AS ITEM_TYPE_ID,
         MEETING_TASK.START_DATE     AS START_DATE,
         MEETING_TASK.Due_DATE       AS DUE_DATE,
         MEETING_TASK.INSTRUCTIONS   AS INSTRUCTIONS,
         '1'                    AS MY_ASSIGNMENT,
         MEETING_TASK.NAME,
         MEETING_TASK.CONTAINER      AS CONTAINER,
         MEETING_TASK.CONTAINER_TYPE_ID AS CONTAINER_TYPE_ID,
         MEETING_TASK.STATUS            AS STATUS,
MEETING_TASK.Agenda AS AGENDA,
```

 

```
            it.open_icon AS ICON,
            MEETING_TASK.CLASSIFICATION,
            MEETING_TASK.KEYED_NAME,
            MEETING_TASK.CREATED_ON,
            MEETING_TASK.CREATED_BY_ID,
            MEETING_TASK.OWNED_BY_ID,
            MEETING_TASK.MANAGED_BY_ID,
            MEETING_TASK.MODIFIED_ON,
            MEETING_TASK.MODIFIED_BY_ID,
            MEETING_TASK.CURRENT_STATE,
            MEETING_TASK.STATE,
            MEETING_TASK.LOCKED_BY_ID,
            MEETING_TASK.IS_CURRENT,
            MEETING_TASK.MAJOR_REV,
            MEETING_TASK.MINOR_REV,
            MEETING_TASK.IS_RELEASED,
            MEETING_TASK.NOT_LOCKABLE,
            MEETING_TASK.CSS,
            MEETING_TASK.GENERATION,
            MEETING_TASK.NEW_VERSION,
            MEETING_TASK.CONFIG_ID,
            MEETING_TASK.PERMISSION_ID,
           MEETING_TASK.TEAM_ID,
            lang.code
   AS language_code_filter
   FROM   innovator.Meeting AS MEETING_TASK
   INNER JOIN innovator.ITEMTYPE AS it on it.name = N'Meeting Task'
            LEFT OUTER JOIN innovator.LANGUAGE AS lang
                    ON lang.ID is NOT NULL
```

d. Click **Save** and **Done** to save and unlock the SQL Item.

e. Navigate to **Actions --> SQL Execute** in the SQL window.

f. Make sure that temporary View with name **Meeting_Task_Step02_Create_Tmp_View** is created in your database.

5. Rename view for Meeting Item Type.

   a. Navigate to **Administration --> SQLs**.

   b. Click **File --> New** or use the right mouse button (right-click) and select **New SQL** from the context menu.

c. The SQL window appears. Configure the following properties:

Name = Meeting_Task_Step03_Rename_View
Type = Table
Execution Flag = Manual
SQL:

```
sp_rename 'Meeting_Task_Step02_Create_Tmp_View', 'Meeting_Task'
```

d. Click **Save** and **Done** to save and unlock the SQL Item.

e. Navigate to **Actions --> SQL Execute** in the SQL window.

f. Make sure that temporary view `Meeting_Task_Step02_Create_Temp_View` is renamed to Meeting.

### 5.2.4 Viewing New InBasket Task ItemType

1. Log on to Aras Innovator with administrative privileges.
2. Click **Meetings** in the TOC. In our example, we created a Meeting ItemType. The following menu appears:

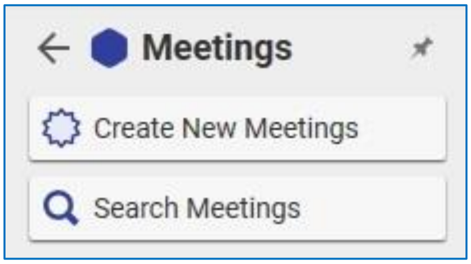


Figure 9.

3. Click **Create New Meeting** to create an instance of Meeting ItemType.
4. Provide the required information for the Meeting ItemType:
   - **Container**: Document
   - **Name**: Meeting Task
   - **Start Date**: Today
   - **Due Date**: Today+1
   - **Status**: Active
   - **Assigned to**: Administrators
   - **Instructions**: Activity Instructions 1
   - **Item**: Doc_1 (create if necessary)
5. Click Save, Done and Close.
6. Navigate to **My Innovator** --> **InBasket**.
7. Execute search.

The Meeting Task appears in the InBasket.

| Type | Source Item | Activity | Start Date [... | Due Date [.... | Status | Assigned To [...] | Instructions | Work Item | My Assignment |
|---|---|---|---|---|---|---|---|---|---|
| Workflow Task | PR-100001 | Verify PR | | | Pending | Owner | Please verify the Problem Report | PR-100001 | ☐ |
| Workflow Task | PR-100001 | Approve PR | | | Pending | Change Specialist I | Please approve the Problem Report | PR-100001 | ☐ |
| Workflow Task | ECO-00001001 | Change Review | | | Pending | Change Control Board | Review changes made by the ECO | ECO-00001001 | ☐ |
| Workflow Task | DCO-00001001 | Final Review | | | Pending | Change Control Board | Review document changes | DCO-00001001 | ☐ |
| Workflow Task | ECO-00001002 | Change Review | | | Pending | Change Control Board | Review changes made by the ECO | ECO-00001002 | ☐ |
| Workflow Task | DCO-00001002 | Final Review | | | Pending | Change Control Board | Review document changes | DCO-00001002 | ☐ |
| Workflow Task | ECO-00001001 | Submit ECO | 10/15/2014 | 10/15/2014 | Active | Innovator Admin | Please fill out and submit the ECO | ECO-00001001 | ☑ |
| Workflow Task | ECO-00001002 | Submit ECO | 10/15/2014 | 10/15/2014 | Active | Innovator Admin | Please fill out and submit the ECO | ECO-00001002 | ☑ |
| Workflow Task | DCO-00001001 | Planning | 10/15/2014 | 10/15/2014 | Active | Innovator Admin | Perform DCO Planning | DCO-00001001 | ☑ |
| Workflow Task | PR-100001 | Review PR | 10/15/2014 | 10/15/2014 | Active | Change Specialist I | Please Review the Problem Report | PR-100001 | ☐ |
| Workflow Task | DCO-00001002 | Planning | 10/15/2014 | 10/15/2014 | Active | Change Control Board | Perform DCO Planning | DCO-00001002 | ☐ |
| Meeting Task | Doc 1 | Activity Test 1 | 10/15/2014 | 10/17/2014 | Pending | CRB | Activity Instructions 2 | Doc 1 | ☐ |
| Meeting Task | Doc 1 | Activity 1 | 10/16/2014 | 10/22/2014 | Active | Administrators | Activity Instructions 1 | Doc 1 | ☑ |

Figure 10.

## 5.3 Adding a New Column to the InBasket

The InBasket Task is a Poly Source Item. To add a new column or a new property to any of the Poly Source Items, you need to add the column or the property to all the other Poly Source Items as well as to the InBasket Task Item.

Adding a new column to the InBasket involves five main procedures, which are described in the following sections.

Let us consider the example of adding a column named Agenda to the newly created Meeting ItemType.

**Note:** If you add a column or a property to any of the Poly Source Items, ensure that it is added to other Poly Source Items and the InBasket Task Item. Else, an error is displayed.

### 5.3.1 Adding a new column to the New InBasket ItemType

1. Log on to Aras Innovator with administrative privileges.
2. Navigate to **Administration** --> **ItemTypes**.
3. Search for the newly created InBasket ItemType in the search grid. Select the Item and open for editing.

   In our example, we have created the **Meeting** ItemType. Let us open the **Meeting** ItemType for editing. The Item window appears.

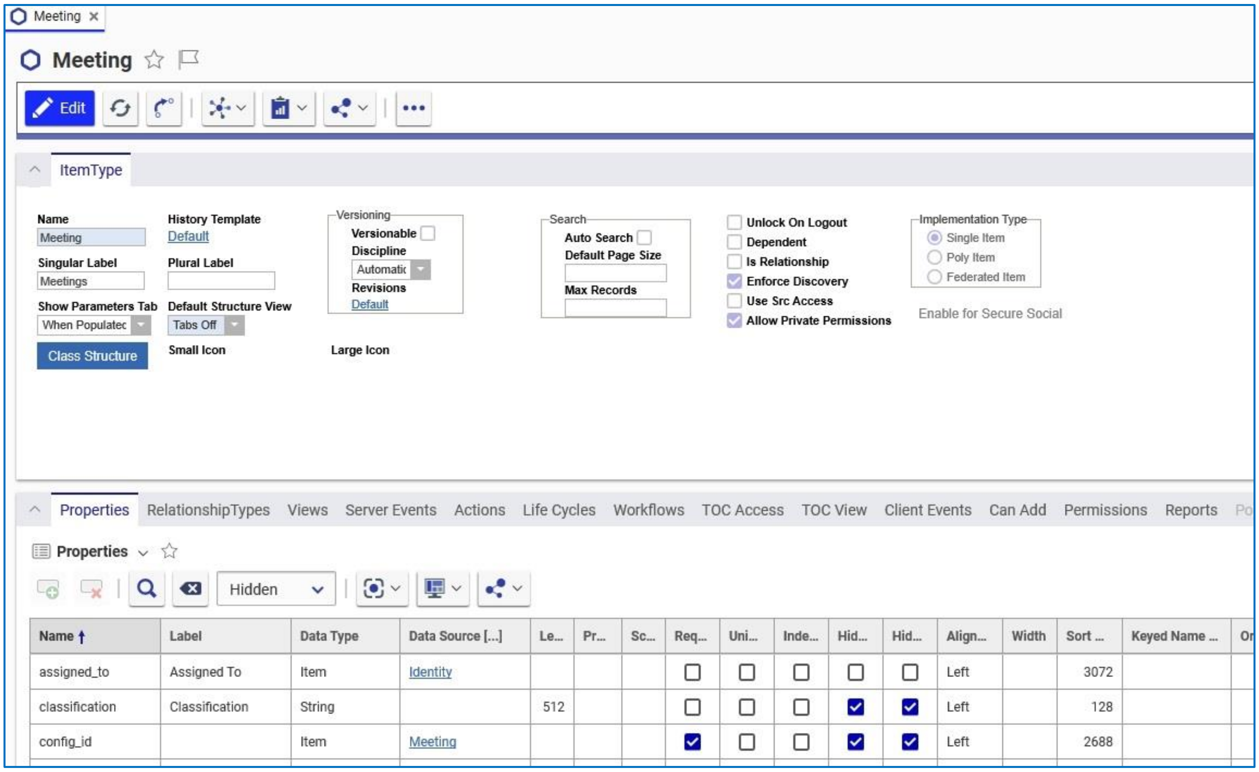


Figure 11.

4. Click 

to unlock the ItemType.

5. In the **Properties** tab, click  to create a new property. A blank row is added in the relationship grid.

6. Provide the required information for the property. In our example, we add the details for Agenda column.
   - **Name**: agenda
   - **Label**: Agenda
   - **Data Type**: Text

7. Click 

and 

.

### 5.3.2 Adding the column to other Poly Source Items

1. Log on to Aras Innovator with administrative privileges.
2. Navigate to **Administration** --> **ItemTypes**.
3. From the grid, search for **InBasket Task** Item and open the Item for editing.

4. Click the **Poly Sources** tab.

The Poly Source Items appear.

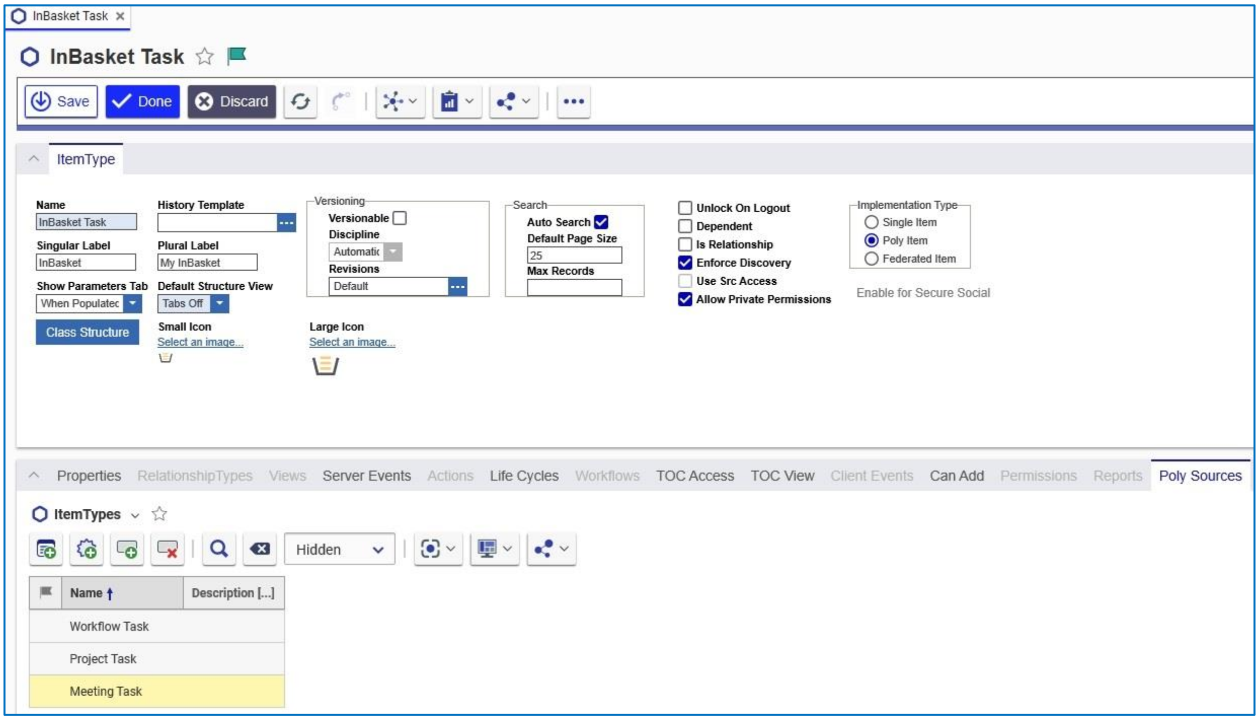


Figure 12.

You need to edit and add the property to each of the Poly Source Items. Start with the **Workflow Task** Item.

5. Right-click on the **Workflow Task**, click **Open**. The Workflow Task Item window appears.

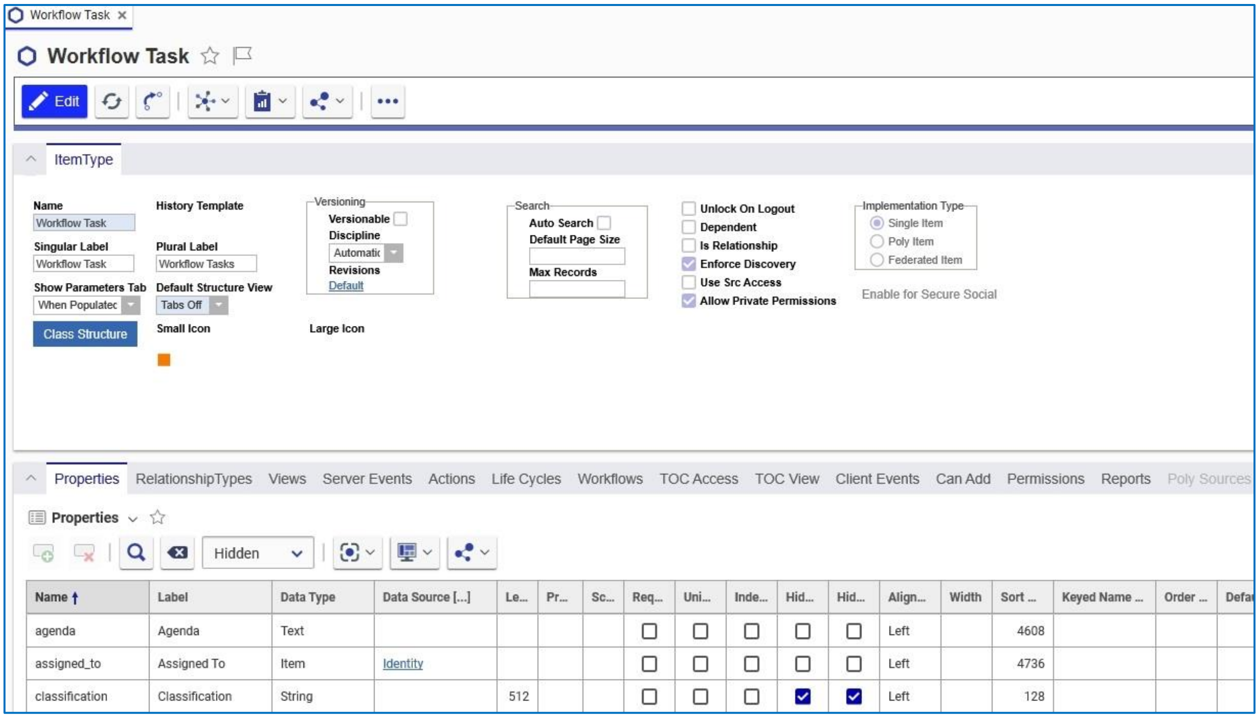


Figure 13.

6. Click  to unlock the Workflow Task Item for editing.
7. In the **Properties** tab, click  to create a new property.

   A blank row is added in the relationship grid.
8. Provide the required information for the property. Here we add the details for Agenda column.
   - **Name**: agenda
   - **Label**: Agenda
   - **Data Type**: Text
9. Click Save, Unlock and Close on the Workflow Task Item window.
10. Repeat step 5 to step 9 for Meeting Task.
11. Close the **InBasket Task** Item window.

**Note:** If adding a property of type Item, the behavior of this property must be set to *Fixed* on **InBasket Task** and all of its **Poly Sources**.

 

### 5.3.3 Rebuilding SQL Views

#### *5.3.3.1 Rebuilding SQL View for New InBasket ItemType*

1. Log on to Aras Innovator with administrative privileges.
2. Navigate to **Administration** --> **SQLs**.
3. Search for the Drop SQL View that we created for the new ItemType.

   In our example, we created `Meeting_Task_Step1_Drop SQL Vie`w. For more information, refer to section 5.2.3 SQL Table and View Execution.
4. Select `Meeting_Task_Step1_Drop` in the search grid, right-click and select **SQL Execute** from the context menu.
5. Search for the Create Temp SQL View that we created for the new ItemType.

   In our example, we created `Meeting_Task_Step02_Create_Temp_view`. For more information, refer to section 5.2.3 SQL Table and View Execution.
6. Select `Meeting_Task_Step02_Create_Temp_view` in the search grid, right-click and select **Edit** from the context menu.

   The SQL `Meeting_Task_Step02_Create_Temp_view` window is displayed.
7. Add the following line to the SQL field.

```
MEETING_TASK.Agenda AS AGENDA
```

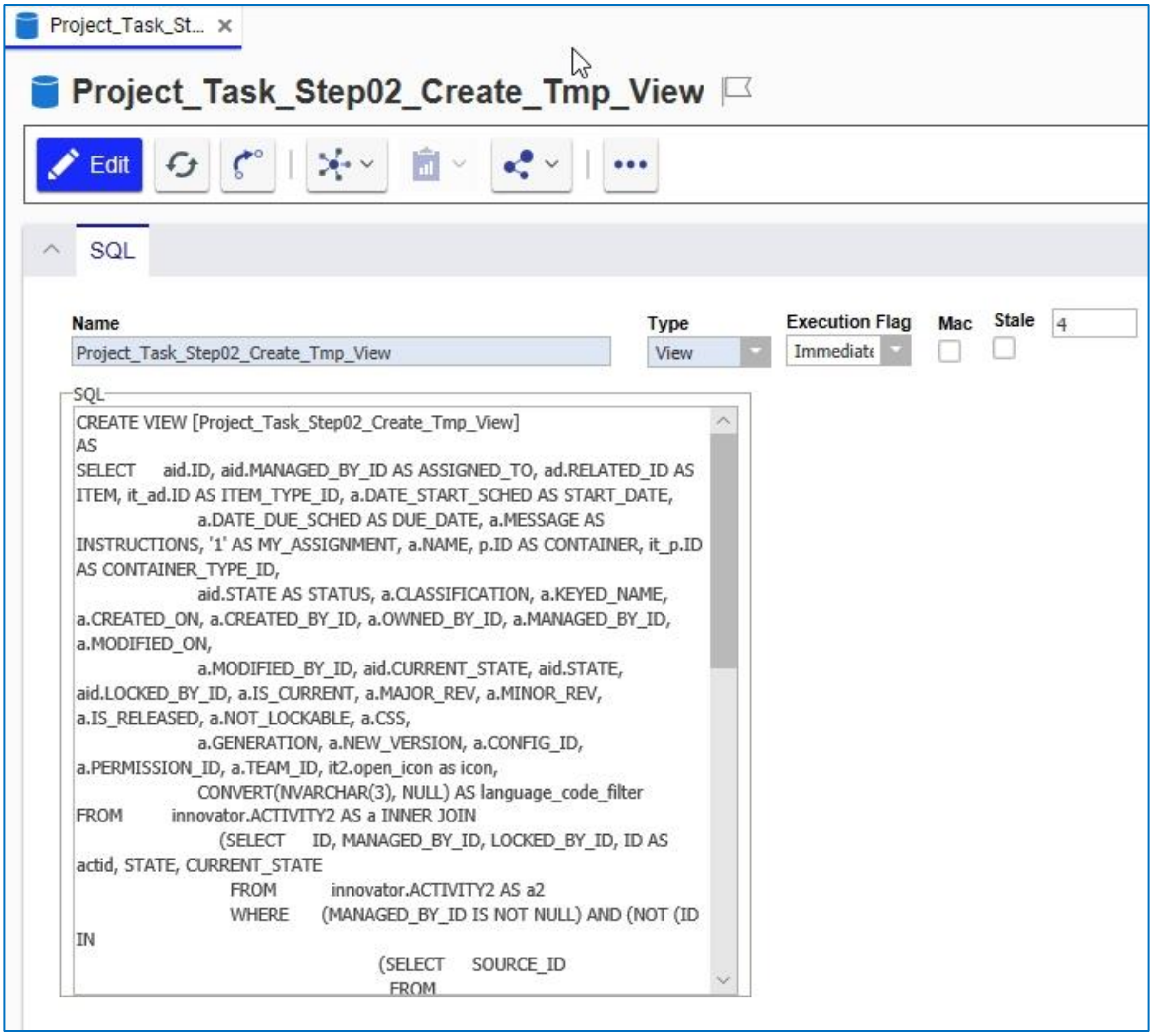


Figure 14.

8. Click  and  to save and unlock the SQL Item.
9. Click **Actions** --> **SQL Execute** in the SQL window and then close the SQL window.
10. Search for the Rename SQL View that we created for the new ItemType.

    In our example, we created `Meeting_Task_Step03_Rename_View`. For more information, refer to section 5.2.3 SQL Table and View Execution.
11. Select `Meeting_Task_Step03_Rename_View` in the search grid, use right mouse button (right-click) and select **SQL Execute** from the context menu.

### *5.3.3.2 Rebuilding SQL View for Other Poly Source Items*

1. Log on to Aras Innovator with administrative privileges.
2. Navigate to **Administration --> SQLs**.
3. Rebuild SQL view for Workflow Item Task.
   a. From the search grid, search `Workflow_Task_Step01_Drop`.
   b. Select `Workflow_Task_Step01_Drop` in the search grid, right-click and select **SQL Execute** from the context menu.
   c. From the search grid, open `Workflow_Task_Step02_Create_Tmp_View`.
   d. Click 

to edit the `Workflow_Task_Step02_Create_Tmp_View`.
   e. In SQL field, add the following line:

```
CONVERT(NVARCHAR(128), NULL) AS AGENDA,
```

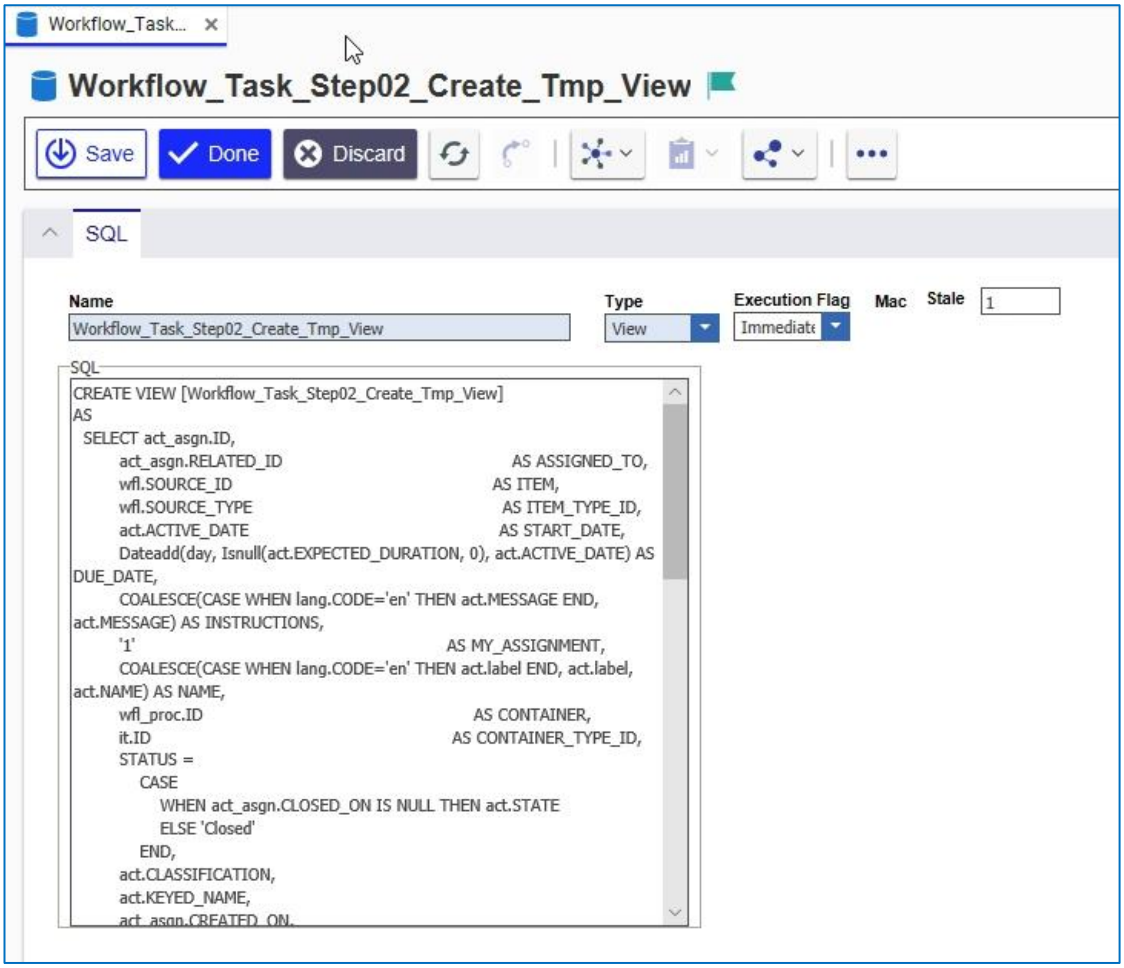


Figure 15.

f. Click 

and 

to save and unlock the SQL Item.

g. Click **SQL Execute** from the **Actions** menu.

h. Close the window.

i. From the search grid, search `Workflow_Task_Step03_Rename_View`.

j. Select `Workflow_Task_Step03_Rename_View` in the search grid, use the right mouse button (right-click) and select **SQL Execute** from the context menu.

### 5.3.4 Adding the column to the InBasket Task Item

1. Log on to Aras Innovator with administrative privileges.
2. Navigate to **Administration --> ItemType**.
3. From the grid, search for Workflow Task Item and click 

.
4. Save, unlock, and close the Workflow Task Item.
5. Repeat steps 3 and 4 for the Meeting Task Item.

**Note:** InBasket Task won't let you edit and save the Item Type until after you've saved, unlocked, and closed the Poly Items connected to it.

6. From the grid, search for InBasket Task Item and click 

.
7. In the Properties tab, click  to create a new property.

   A blank row is added in the relationship grid.
8. Provide the required information for the property. In our example, we add the details for Agenda column.
   - **Name**: agenda
   - **Label**: Agenda
   - **Data Type**: Text

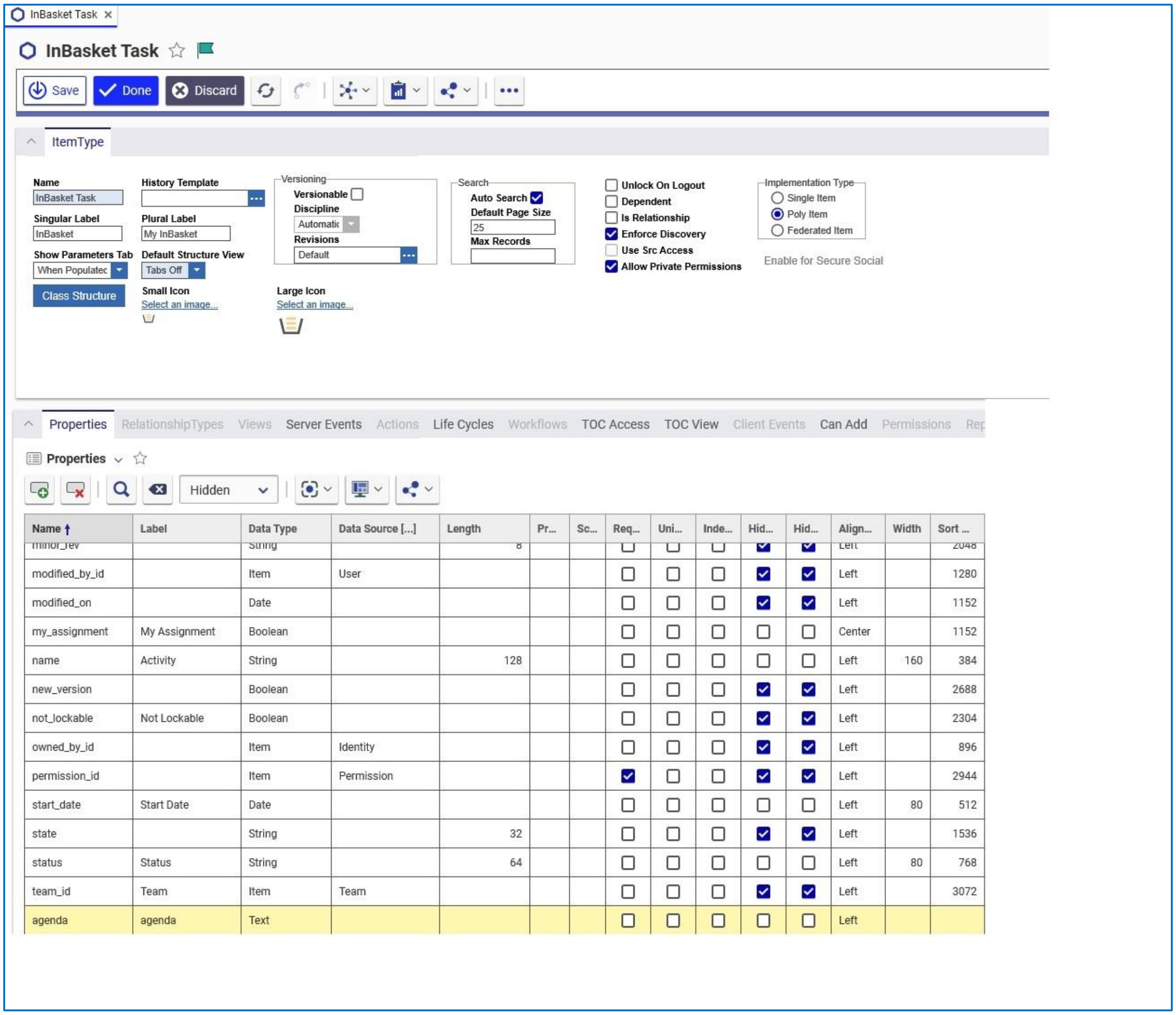


Figure 16.

9. Click Save, Unlock and Close in the **InBasket Task** Item window.

## 5.3.5 Viewing the New Column in Configurable InBasket

1. Log on to Aras Innovator with administrative privileges.
2. Navigate to **My Innovator** --> **My InBasket**.

The InBasket search grid displays the newly added column. In our example, we can now see the newly added column **Agenda […]**.

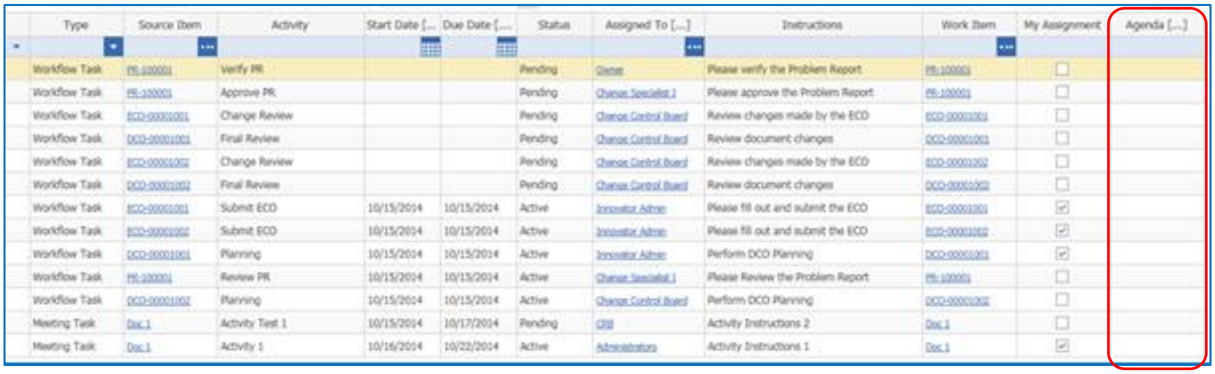


Figure 17.

## 5.4 Creating an Import Package to add a Column

When creating an import package that adds a column to the Configurable InBasket you must ensure that the order in which the property is added to the PolyItems is correct or the import will fail.

In the example below there are steps on how to create a package that will add a column that shows the Part_Number of an InBasket task.

1. From the source database, use the Export tool to export the Workflow_Task and InBasket_Task ItemTypes, and their corresponding SQL queries
    - The Workflow_Task and InBasket_Task ItemTypes, and SQL queries are located in the com.aras.innovator.inbasket package definitions

 

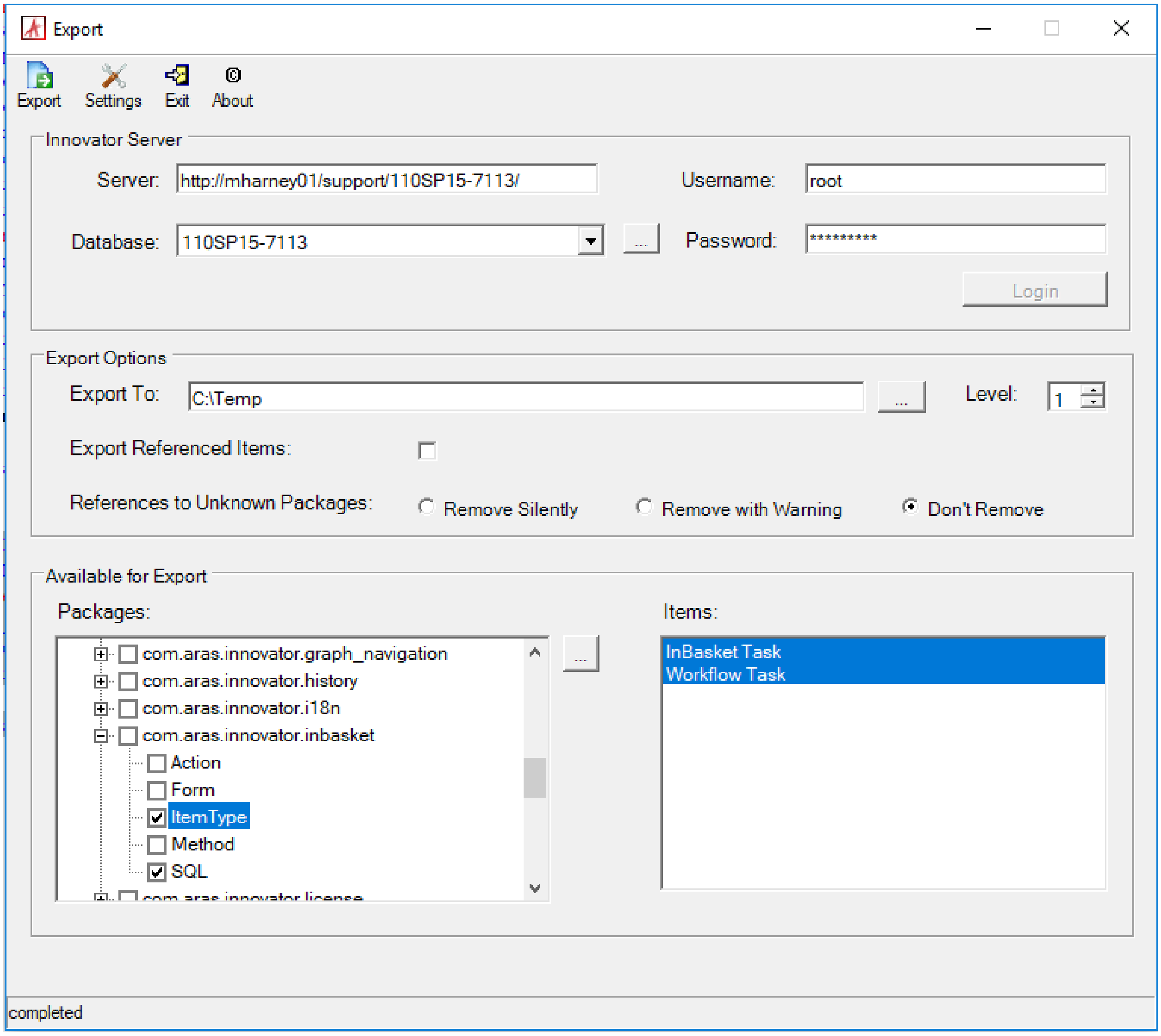


Figure 18.

2. From the InBasket_Task.xml you will need to remove the following lines (typically located near the end of the file):

```
<Item type="ItemType" id="BC7977377FFF40D59FF14205914E9C71" action="edit">
 <Relationships>
  <Item type="Morphae" id="9657383E16C64B86AB06E17CDD20AE23" action="add">
   <related_id keyed_name="Workflow Task" type="ItemType" name="Workflow Task">321BD822949149C597FD596B1212B85C</related_id>
   <sort_order>128</sort_order>
   <source_id keyed_name="InBasket Task" type="ItemType" name="InBasket Task">BC7977377FFF40D59FF14205914E9C71</source_id>
  </Item>
 </Relationships>
</Item>
<Item type="ItemType" id="BC7977377FFF40D59FF14205914E9C71" action="edit">
 <Relationships>
  <Item type="Morphae" id="DE32C8E3C356483A896D36443CB8A288" action="add">
  <source_id keyed_name="InBasket Task" type="ItemType" name="InBasket Task">BC7977377FFF40D59FF14205914E9C71</source_id>
  </Item>
 </Relationships>
</Item>
```

**Note:** It is recommended you copy and paste the above lines for future use in the upcoming steps.

 

3. From the InBasket_Task.xml you will need to remove the new column property (shown below) to avoid errors during the import.

```
<Item type="Property" id="6DCBE94B46044404B74F15A613A02811" action="add">
<column_alignment>left</column_alignment>
<data_type>string</data_type>
<is_hidden>0</is_hidden>
<is_hidden2>0</is_hidden2>
<is_indexed>0</is_indexed>
<is_keyed>0</is_keyed>
<is_multi_valued>0</is_multi_valued>
<is_required>0</is_required>
<label xml:lang="en">test</label>
<range_inclusive>0</range_inclusive>
<readonly>0</readonly>
<sort_order>4608</sort_order>
<source_id keyed_name="InBasket Task" type="ItemType" name="InBasket Task">BC7977377FFF40D59FF14205914E9C71</source_id>
<stored_length>32</stored_length>
<track_history>0</track_history>
<name>test</name>
</Item>
```

4. In the new import package folder, create an … /Imports/Fixes folder
5. Inside the fixes folder you will add the xml files shown below:
   - Workflow_Task.xml
   - InBasket_Task.xml

   a. 1 Workflow_Task.xml will look something like the following:

```
<AML>
<Item type="ItemType" id="321BD822949149C597FD596B1212B85C"
action="edit">
  <Relationships>
    <Item type="Property" id="EB9B7F4B06064C378EF5CEC3E94FB58E"
action="add">
    <column_alignment>left</column_alignment>
    <data_type>string</data_type>
    <is_hidden>0</is_hidden>
    <is_hidden2>0</is_hidden2>
    <is_indexed>0</is_indexed>
    <is_keyed>0</is_keyed>
    <is_multi_valued>0</is_multi_valued>
    <is_required>0</is_required>
    <label xml:lang="en">test</label>
    <range_inclusive>0</range_inclusive>
    <readonly>0</readonly>
    <sort_order>4608</sort_order>
    <source_id keyed_name="Workflow Task" type="ItemType"
name="Workflow Task">321BD822949149C597FD596B1212B85C</source_id>
    <stored_length>32</stored_length>
    <track_history>0</track_history>
    <name>test</name>
   </Item>
  </Relationships>
 </Item>

<Item type="ItemType" id="BC7977377FFF40D59FF14205914E9C71"
action="edit">
  <Relationships>
```

 

```
    <Item type="Morphae" id="9657383E16C64B86AB06E17CDD20AE23"
action="add">
      <related_id keyed_name="Workflow Task" type="ItemType"
name="Workflow Task">321BD822949149C597FD596B1212B85C</related_id>
      <sort_order>128</sort_order>
      <source_id keyed_name="InBasket Task" type="ItemType"
name="InBasket Task">BC7977377FFF40D59FF14205914E9C71</source_id>
    </Item>
  </Relationships>
 </Item>
 </AML>
```

The code highlighted in yellow is taken from the exported Workflow_Task.xml where the information is that of the property that will be used for the new column. The code highlighted in green is taken from the deleted lines of the InBasket_Task.xml.

b. 3 Inbasket_task.xml will only include code to add the new column property as shown below:

```
<AML>
<Item type="ItemType" id="BC7977377FFF40D59FF14205914E9C71"
action="edit">
  <Relationships>
   <Item type="Property" id="6DCBE94B46044404B74F15A613A02811"
action="add">
     <column_alignment>left</column_alignment>
     <data_type>string</data_type>
     <is_hidden>0</is_hidden>
     <is_hidden2>0</is_hidden2>
     <is_indexed>0</is_indexed>
     <is_keyed>0</is_keyed>
     <is_multi_valued>0</is_multi_valued>
     <is_required>0</is_required>
     <label xml:lang="en">test</label>
     <range_inclusive>0</range_inclusive>
     <readonly>0</readonly>
     <sort_order>1024</sort_order>
     <source_id keyed_name="InBasket Task" type="ItemType"
name="InBasket Task">BC7977377FFF40D59FF14205914E9C71</source_id>
     <stored_length>1024</stored_length>
     <track_history>0</track_history>
     <name>test</name>
   </Item>
  </Relationships>
 </Item>
</AML>
```

**Note:** You will need to generate a new ID for the added property. To do this you can go through the following steps:

a. From a browser, go to [http://localhost/innovator/client/scripts/nash.aspx](http://localhost/innovator/client/scripts/nash.aspx)

b. Fill in Login information for the appropriate target database

c. Under Action select generateNewGUID

d. Click on Submit

e. Copy the newly generated ID and add that to the 3 Inbasket_Task.xml file in the highlighted area above

6. To ensure that the fixes folder is imported with the import tool, you will need to add the path as follows:

```
   <package name="com.aras.innovator.solution.InBasketFix"
path="InBasketFix\Import">
    <dependson name="com.aras.innovator.inbasket" />
  </package>
```

## 5.5 Managing Language Preferences

It is possible to set up the Configurable InBasket to handle multiple languages so that it selects the values appropriate to the client environment.

The following sample shows how to adjust the InBasket to handle the German locale:

1. Edit the `Workflow_Task_Step02_Create_Tmp_View` SQL item.
   a. Open the `Workflow_Task_Step02_Create_Tmp_View` SQL item for editing.
   b. Change the query for **Instructions** column:

   *From:*
```
COALESCE(CASE WHEN lang.CODE='en' THEN act.MESSAGE END, act.MESSAGE) AS
INSTRUCTIONS,
```

   *To:*
```
COALESCE(CASE WHEN lang.CODE='en' THEN act.MESSAGE
              WHEN lang.CODE='de' THEN act.MESSAGE_DE END,
         act.MESSAGE) AS INSTRUCTIONS,
```

   c. Change the query for **Activity** column:

   *From:*
```
COALESCE(CASE WHEN lang.CODE='en' THEN act.label END, act.label, act.NAME) AS
NAME,
```

   *To:*
```
COALESCE(CASE WHEN lang.CODE='en' THEN act.LABEL
              WHEN lang.CODE='de' THEN act.LABEL_DE END,
         act.LABEL, act.NAME) AS NAME,
```

   d. Save, unlock, and close the `Workflow_Task_Step02_Create_Tmp_View` SQL item.

2. Rebuild SQL view for Workflow Item Task.
   a. From the search grid, search for `Workflow_Task_Step%` SQL items.
   b. Right-click on `Workflow_Task_Step01_Drop` in the search grid and select **SQL Execute** from the context menu.
   c. Right-click on `Workflow_Task_Step02_Create_Tmp_View` in the search grid and select **SQL Execute** from the context menu.
   d. Right-click on `Workflow_Task_Step03_Rename_View` in the search grid and select **SQL Execute** from the context menu.