Conversations View

Last Modified on 08/17/2021 9:01 am EDT

V9.7

Overview

The Conversation View is a solution-level component that enables end users to see all case-related conversations in one place within a case.

You can use this component in a form within any solution-based workflow, to display different types of case-related conversations available in Cora SeQuence.

The Conversation View is deployed with the solution package, and you can configure different conversation types to be enabled or disabled in the relevant Solution. The list of conversation types is loaded from all the system-available solutions.

Conversation types are enabled by default for Solutions. When conversation types are disabled, there is no option to add them through UI, built-in command, or API. Any existing disabled conversation items are displayed with the default template.
The Delete action is allowed only through API.

Starting with V9.6, you can make the case conversations as read-only. For read-only conversations, the users can only read an email or comment and download any existing attachments in the conversation. They can't send new emails or comments from the conversations.

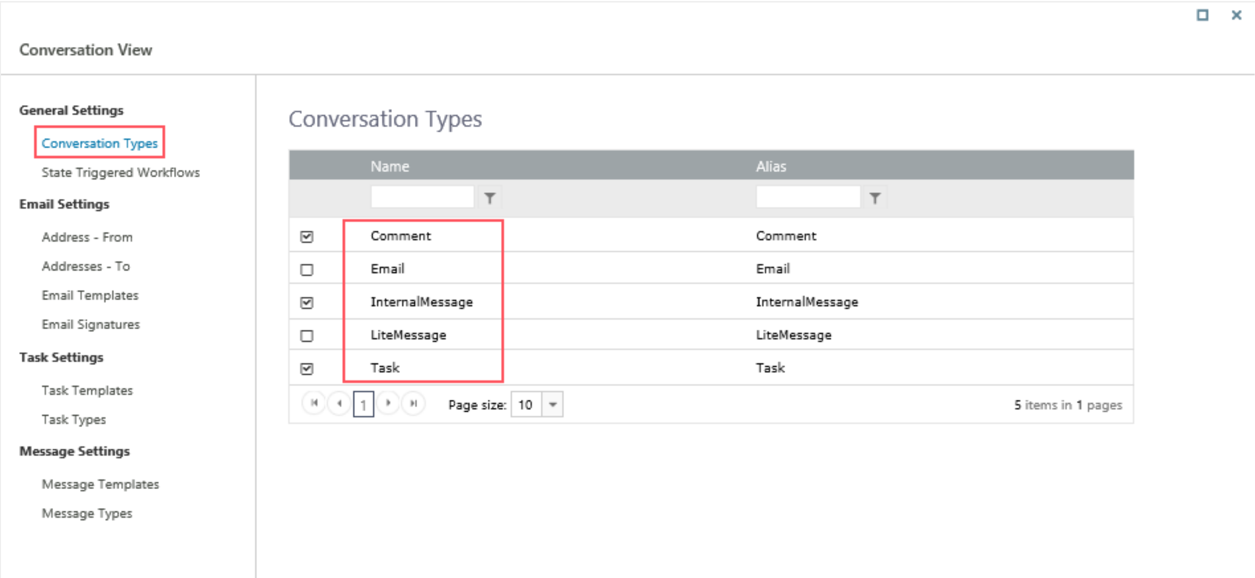
Starting with V9.7, the users can filter conversation items in the Conversations View based on different parameters.

> **NOTE**
> Post upgrade, for any existing Solutions, all the conversation types are disabled. To enable them, open and close the Conversation View window in the Solution once.

Conversation types

All conversation-related items are saved in a product database table. The Comment, Email, InternalMessage, LiteMessage, Task, and General conversation types are the out-of-the-box types available in Cora SeQuence.

For configuring **Email** conversation type, see this article.
For configuring **Comment** conversation type, see this article.
For configuring **InternalMessage** and **LiteMessage** conversation types, see this article.
For configuring **Task** conversation type, see this article.



You can add any custom conversation type to your Solutions per need.
For adding **Custom Conversation** type, see this article.

You can add a **General conversation** item to the Conversations View. The general items are added using a system default type called Conversation Item. This type uses the default template and is the basis of all conversation items.

Configure the Conversations View

To enable conversation view functionality in a workflow instance, you need to configure this functionality in the Solution, of which the workflow is part of.

1. In the Administration site, open the Solution from the list.
2. In the Solution Components section, click **Conversations View**.
3. In the Conversation View window:
    a. Select conversation types you want to enable for the Solution.
    b. Configure state triggered workflows for event handling.
       For details, see the Configure conversation item events for the event handling section in this article below.

Add the Conversation View to a form

You can add conversations view to a form in two ways:

- Add a code snippet to the form source
- Add Conversation View control from the Toolbox

Add snippet to the form source

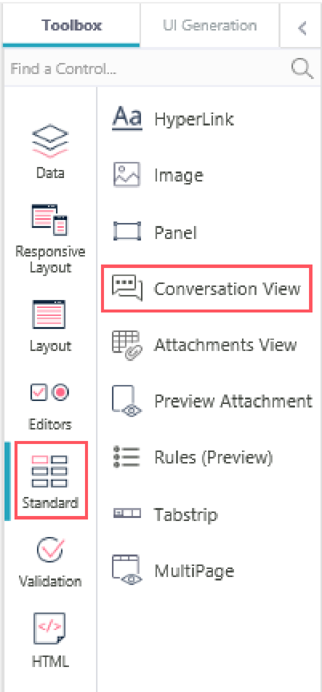
1. In a workflow, add or edit a form or task activity.
2. Add the following script to the source in the relevant location.

```

```

Add the Conversation View from the Toolbox

1. In a workflow, edit the form activity.
2. Select from the Toolbox, **Standard** > **Conversation View**.



In the Conversation View properties, select **Read-only** under Behavior to make the conversations read-only.

3. Save the form.

NOTE

- Only the workflows that are in the context of a Solution support Conversation View.
- Conversations used in sub workflows can also work in the master workflow and will display the same conversation items.

Conversation View UI

Consider the following design parameters for the conversation view in Solutions.

- Supported resolution in the form is:
    - Height: Minimal height for the view in the form (empty state) is 100px
    - Width: 33% - 100% of the screen; minimum 600px
- The view is responsive.
- The most recent item is listed on top, from newest to oldest, or last update date descending.

Language support

You can localize the Conversation View in an implementation.
For details, see this article.

The Conversation View component is translated according to the form's *Globalization* settings:

- if *Localizable* = False, use the *Language* activity definition
- if *Localizable* = True, use the user's language

Conversations item built-in command

Various built-in commands within the Case Operations category lets you create and manage different conversation items.

Create Conversation Item built-in command

The Create Conversation Item built-in command lets you insert conversation items into the Conversations View in the master workflow context.

1. Add a Built-In Command activity.
2. Select **Case Operations** > **Create Conversation Item**.
3. Click **Next**.
4. Add the command parameters.
    - workflowInstanceId: can be the instance id of the master or the sub workflow.
    - conversationItemState: can be any state depending on the conversation item selected. For details, see this article.
    - ConversationItem: select from the list the conversation item you want to add. Unselect 'IsNull' to display the command parameters for the selected conversation type. The command parameters for each conversation type are explained in their respective articles listed in the Related Articles section below.
    - createdByUserId: the current user executing the built-in command.
5. Click **Finish**.

Update Conversation Item built-in command

The Update Conversation Item built-in command lets you update conversation items added in the Conversations View in the master workflow context.

1. Add a Built-In Command activity.
2. Select **Case Operations** > **Update Conversation Item**.
3. Click **Next**.
4. Set the following command parameters.
    - workflowInstanceId: can be the instance id of the master or the sub workflow.
    - conversationItemId: The guid of the conversation item to be updated.
    - ConversationItemPropertyAssignment: Add a list of conversation item properties you want to update.
        - Key: name of the property to be updated. For example "Subject".
        - Value: the updated value. For example "New Request".
5. Click **Finish**.

Copy Case Conversation Items Between Cases built-in command

The Copy Case Conversation Items Between Cases built-in command lets you copy case conversations between cases.

For details, see this article.

IMPORTANT
Make sure you have *Execute* permissions in the master workflow.

Conversation expressions

In some scenarios, you may need to display or use the conversation item details. For example, there may be a need to create a new Cora SeQuence task based on a conversation task. The conversation task triggers a workflow through the configured State Triggered Workflows. You can then execute the expression to fetch the conversation item details.

Execute the expression in the following format:

```
mwf.Extensions["ConversationsItemsExtension"].Items[CONVERSATION_ITEM_UNIQUE_ID].AsType("TYPE_NAME").PROPERTY
```

Examples

mwf.Extensions["ConversationsItemsExtension"].Items[{Builtin command listener1}.In["conversationItemParameters"].ConversationItemUniqueId].AsType("Email").Body

mwf.Extensions["ConversationsItemsExtension"].Items[{Send to Lite Solution Listener}.In["conversationItemParameters"].ConversationItemUniqueId].AsType("InternalMessage").From

Conversation View client API

The Conversations View component exposes the following four events to define additional actions and configurations.

Parameter	Description
addConversationViewItemActionClicked	is called when a user clicks on an action related to a specific item. For example, click on Reply.
addConversationViewItemToolbarActionClicked	is called when a user clicks on the Conversation View toolbar buttons. For example, click on New email.
addConversationViewParametersBinding	is called when Conversation View binds the Conversation Item parameters. This event helps the implementation set the Email Composer properties.
addConversationViewActionCompleted	is called when a user completes an action performed in the Conversation View. For example, when closing the Email Composer window, either by clicking the X button, or by clicking Save or Send.
addConversationViewEditorLoading	is called for finding the custom control user added to the editor and putting it in the data.

Parameter	Description
addConversationViewEditorClosing	is called for taking the data from the control and saving it into parameters object. It is cancellable by using ‘event.preventDefault()’

All the events receive the args object parameters that include the following:

- args.details.actionName: The selected action.
  For example, New.
- args.details.itemType: The Conversation item type on which the action applies.
  For example, Email.
- args.detail.parameters: The Conversation item related parameters.
  For example, windowTitle for Email Composer.
- args.detail.editor: The window object to find user-added controls.
- args.detail.context: The message context of translations resource file.

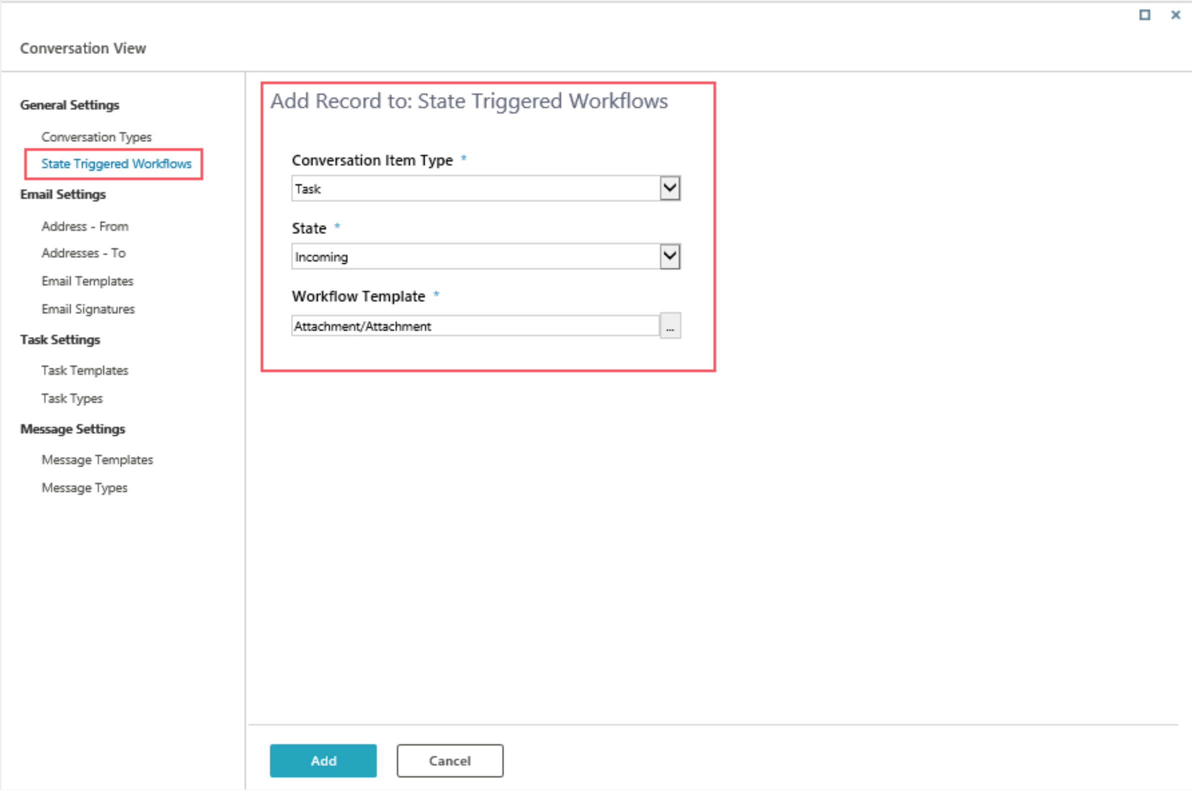
Clients API example

Configure state triggered workflows

You can configure the system to initiate a specific workflow and perform an action when a conversation item changes to a specific status.

You can configure the conversation item events to initiate a workflow in the master workflow context. The initiated workflow should start with a Built-in Command Listener activity to listen to the conversation item event.

1. In the Administration site, open a Solution from the list.
2. In the Solution Components section, click **Conversations View**.
3. In the Conversation View window, click **State Triggered Workflows**.
4. Click **Add New Record**.
5. Select **Conversation Item Type**.
6. Select **State**.
7. Select **Workflow Template**, the workflow to initiate for conversation item.
8. Click **Add**.



NOTE

If the State is not changed the workflow is not initiated. For example, editing a draft and saving it will not trigger the configured workflow.

V9.5

Overview

The Conversation View is a solution-level component that enables end users to see all case-related conversations in one place within a case.

You can use this component in a form within any solution-based workflow, to display different types of case-related conversations available in Cora SeQuence.

The Conversation View is deployed with the solution package, and you can configure different conversation types to be enabled or disabled in the relevant Solution. The list of conversation types is loaded from all the system-available solutions.

Conversation types are enabled by default for Solutions. When conversation types are disabled, there is no option to add them through UI, built-in command, or API. Any existing disabled conversation items are displayed with the default template.
The Delete action is allowed only through API.

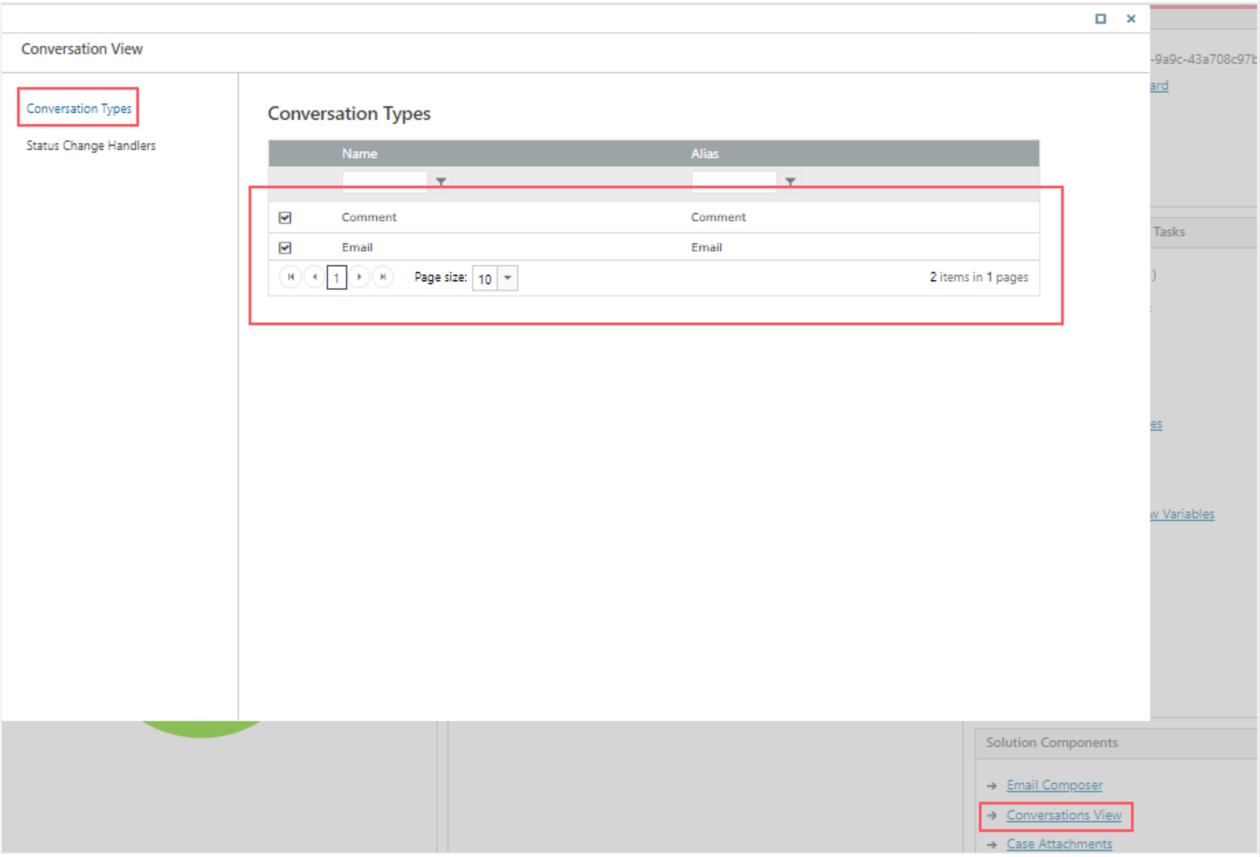
Starting with V9.6, you can make the case conversations as read-only. For read-only conversations, the users can only read an email or comment and download any existing attachments in the conversation. They can't send new emails or comments from the conversations.

> **NOTE**
> Post upgrade, for any existing Solutions, all the conversation types are disabled. To enable them, open and close the Conversation View window in the Solution once.

Conversation types

All conversation related items are saved in a product database table. The Email and Comment conversation types are the out-of-the-box types available in Cora SeQuence.

For configuring **Email Conversation** type, see this article.
For configuring **Comment Conversation** type, see this article.



You can add any custom conversation type to your Solutions per need.
For adding **Custom Conversation** type, see this article.

Configure the Conversation View

To enable conversation view functionality in a workflow instance, you need to configure this functionality in the Solution, of which the workflow is part of.

1. In the Administration site, open the Solution from the list.
2. In the Solution Components section, click **Conversations View**.
3. In the Conversations View window:
    a. Select conversation types you want to enable for the Solution.
    b. Add status change handlers for event handling.
    For details, see the Configure conversation item events for the event handling section in this

article below.

Add the Conversation View to a form

You can add conversations view to a form in two ways:

- Add a code snippet to the form source
- Add Conversation View control from the Toolbox

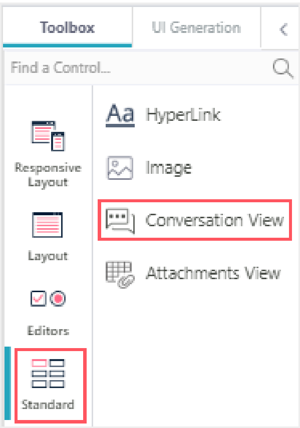
Add snippet to the form source

1. In a workflow, add or edit a form or task activity.
2. Add the following script to the source in the relevant location.

```
```

Add the Conversation View from the Toolbox

1. In a workflow, edit the form activity.
2. Select from the Toolbox, **Standard** > **Conversation View**.



   In the Conversation View properties, select **Read-only** under Behavior to make the conversations read-only.
3. Save the form.

> **NOTE**
>
> - Only the workflows that are in the context of a Solution support Conversation View.
> - Conversations used in sub workflows can also work in the master workflow and will display the same conversation items.

Conversation View UI

Consider the following design parameters for the conversation view in Solutions.

- Supported resolution in the form is:
  - Height: Minimal height for the view in the form (empty state) is 100px
  - Width: 33% - 100% of the screen; minimum 600px

- The view is responsive.
- The most recent item is listed on top, from newest to oldest, or last update date descending.

Language support

You can localize the Conversation View in an implementation.
For details, see this article.

The Conversation View component is translated according to the form's *Globalization* settings:

- if *Localizable* = False, use the *Language* activity definition
- if *Localizable* = True, use the user's language

Conversations item built-in command

The Create Conversations Item built-in command lets you insert conversation items into the Conversation View. You can add a general conversation item or a specific conversation item related to the type of conversation available in the system. The general items are added using a system default type called Conversation Item. This type uses the default template and is the basis of all conversation items.
For details, see this article.

Command parameters

Command parameters for Conversations Item built-in command can be set for different conversation types.

- workflowInstanceId
- conversationItemState
- conversationItem
  - Title
  - Body
  - Attachments
- createdByUserId

IMPORTANT
Make sure you have *Execute* permissions in the master workflow.

Conversation View client API

The Conversations View component exposes the following four events to define additional actions and configurations.

Parameter	Description
addConversationViewItemActionClicked	is called when a user clicks on an action related to a specific item. For example, click on Reply.

Parameter	Description
addConversationViewItemToolbarActionClicked	is called when a user clicks on the Conversation View toolbar buttons. For example, click on New email.
addConversationViewParametersBinding	is called when Conversation View binds the Conversation Item parameters. This event helps the implementation set the Email Composer properties.
addConversationViewActionCompleted	is called when a user completes an action performed in the Conversation View. For example, when closing the Email Composer window, either by clicking the X button, or by clicking Save or Send.
addConversationViewEditorLoading	is called for finding the custom control user added to the editor and putting it in the data.
addConversationViewEditorClosing	is called for taking the data from the control and saving it into parameters object. It is cancellable by using 'event.preventDefault()'

All the events receive the args object parameters that include the following:

- args.details.actionName: The selected action.
  For example, New.
- args.details.itemType: The Conversation item type on which the action applies.
  For example, Email.
- args.detail.parameters: The Conversation item related parameters.
  For example, windowTitle for Email Composer.
- args.detail.editor: The window object to find user-added controls.
- args.detail.context: The message context of translations resource file.

Clients API example

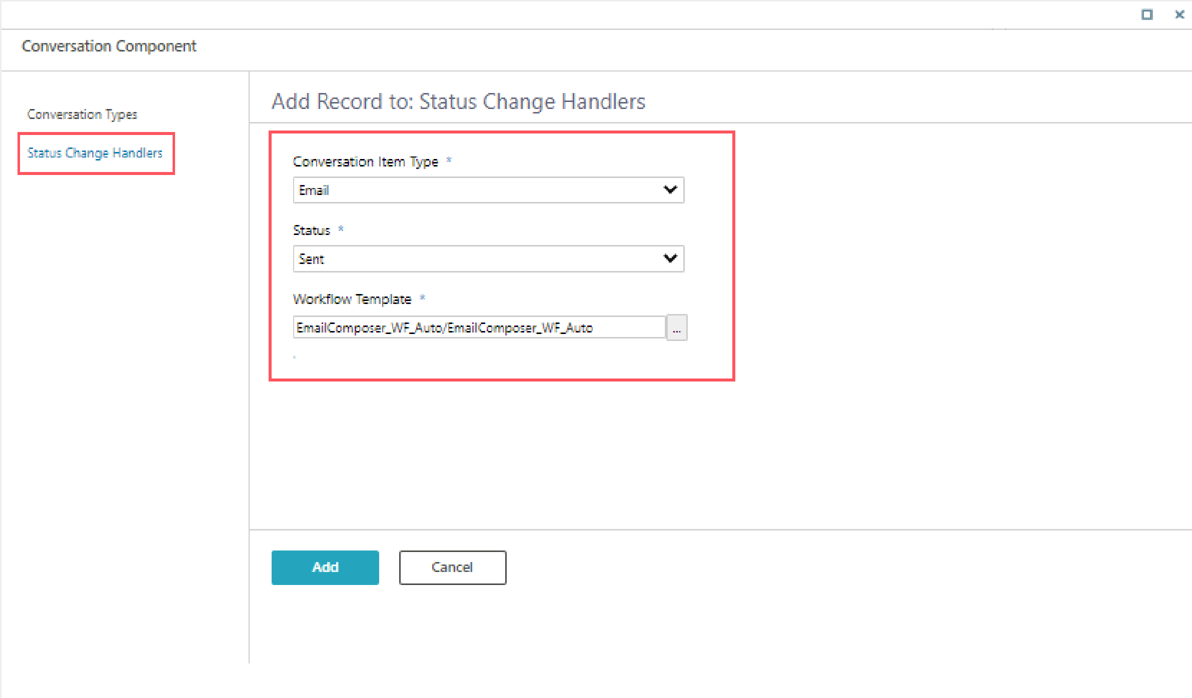
Configure conversation item events for event handling

The conversation item events hook up to implementations and make them capable of performing actions on conversation item status change. You can configure the system to initiate a specific workflow and perform an action when an email conversation item changes to a specific status.

You can configure the conversation item events to hook up to implementations at the solution level.

1. In the Administration site, open a Solution from the list.

2. In the Solution Components section, click **Conversation Component**.
3. In the Conversation Component window, click **Status Change Handlers**.
4. Click **Add New Record**.
5. Select **Conversation Item Type** as Email.
6. Select **Status**, Sent or Undelivered.
7. Select **Workflow Template**, the workflow to initiate for conversation item.
8. Click **Add**.



9. In the selected workflow, add a Built-in Command Listener activity to listen to the conversation item event.

V9.4.1

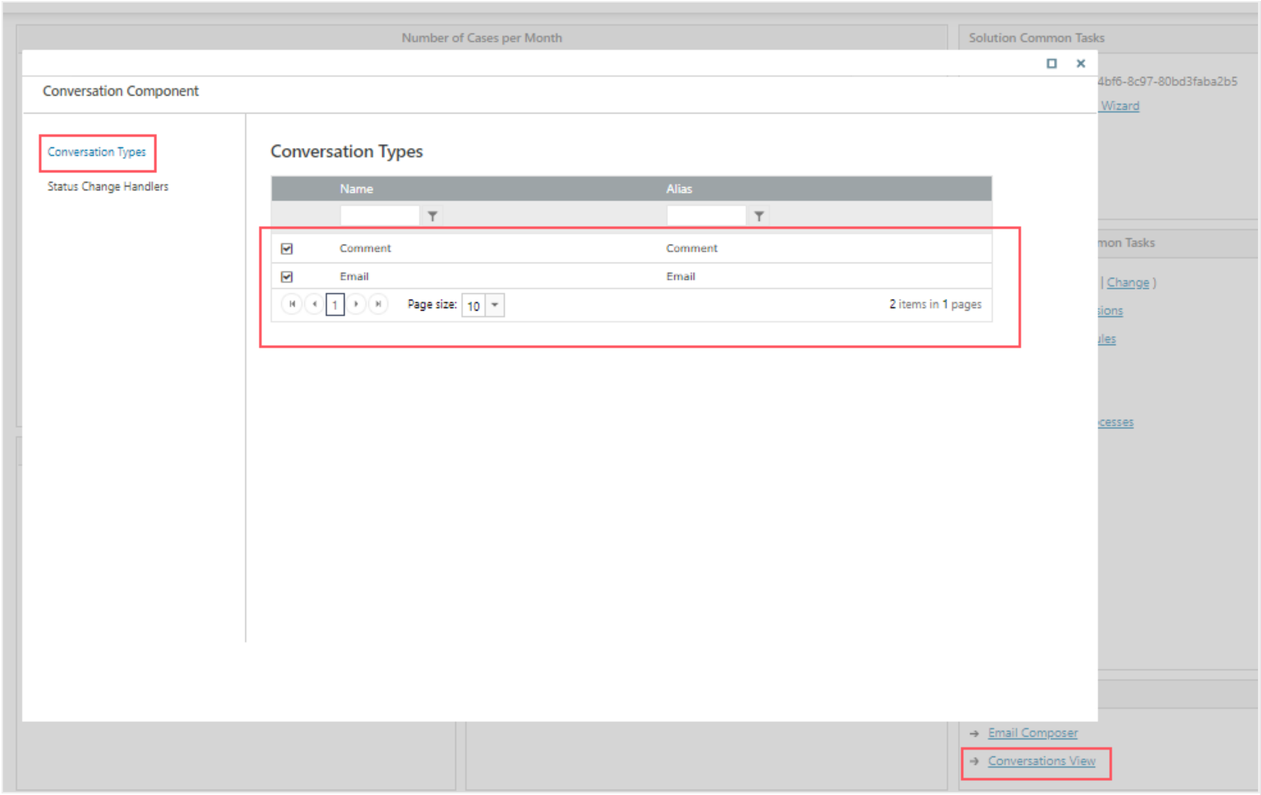
Overview

Conversation View is a solution-level component that enables the users to have all the case related conversations at one place within a case. This reusable component can be combined in a form within any solution-based workflow.

Conversation types

All conversation related items are saved in a product database table. The Email and Comment conversation types are the out-of-the-box conversation types available in Cora SeQuence.

For configuring **Email Conversation** type, see this article.
For **Comment Conversation** type, see this article.



You can add any custom conversation type to your solutions per need.
For adding **Custom Conversation** type, see this article.

Add Conversations View to Cora SeQuence forms

1. In a workflow, add or edit a form or task activity.
2. Add the following script to the form's source in the relevant location.

```
<sq:ConversationView runat="server" Id="Conversations">sq:ConversationView>
div>
```

NOTE

- Only the workflows that are in the context of a Solution support Conversations View.
- Conversations View component should be used in the master workflow forms.

Conversations view UI

Consider the following design parameters for Conversations view in Solutions.

- Supported resolution in the form is:
  - Height: Minimal height for the view in the form (empty state) is 100px
  - Width: 33% - 100% of the screen; minimum 600px
- View is responsive.
- Most recent item is listed on top, from newest to oldest, or last update date descending.

Conversation item built-in command

The Create Conversation Item built-in command lets end users insert conversation items into the Conversations View. You can add a general conversation item, or a specific conversation item related to the type of conversation available in the system. The general items are added using a system default type called Conversation Item. This type uses the default template, and is the basis of all conversation items. For details on the Email Conversation type, see this article.

Command parameters

- workflowInstanceId
- conversationItemState
- Title
- Body
- Attachments
- createdByUserId

IMPORTANT
To execute built-in command that writes into the Conversations View, make sure you have *Execute* permissions in the master workflow.
To use Conversations View in a sub-workflow, the WF instance Id parameter should be mapped to the master workflow instance Id.

Conversation solution configuration

You can configure the conversation types to be enabled or disabled in the relevant solution. The list of conversation types is loaded from all of the system-available solutions.

When conversation types are disabled, there is no option to add them through the UI, built-in command, or API. Any existing disabled conversation items are displayed with the default template.
The Delete action is allowed only through API.

These reusable solution components are deployed with the solution package.

NOTE
Conversation Types are enabled by default for new created Solutions. Post upgrade, for any existing Solutions, all the conversation types are disabled.
Post-upgrade, if you want to enable the Conversations View component for an existing Solution, then open and close the Conversations View window in the solution once.

Conversations View client API

The Conversations View component exposes the following four events to define additional actions and configurations.

- addConversationViewItemActionClicked: is called when a user clicks on an action related to a specific item.
  For example, click on Reply.
- addConversationViewItemToolbarActionClicked: is called when a user clicks on the Conversations View toolbar buttons.
  For example, click on New email.

- addConversationViewParametersBinding: is called when Conversations View binds the Conversation Item parameters.
  This event helps implementation set the Email Composer properties.
- addConversationViewActionCompleted: is called when a user completes an action performed in the Conversations View.
  For example, when closing the Email Composer window, either by clicking the X button, or by clicking Save or Send.

All the events receive the args object parameters that include the following:

- args.details.actionName: The selected action.
  For example, New.
- args.details.itemType: The Conversation item type on which the action applies.
  For example, Email.
- args.detail.parameters: The Conversation item related parameters.
  For example, windowTitle for Email Composer.

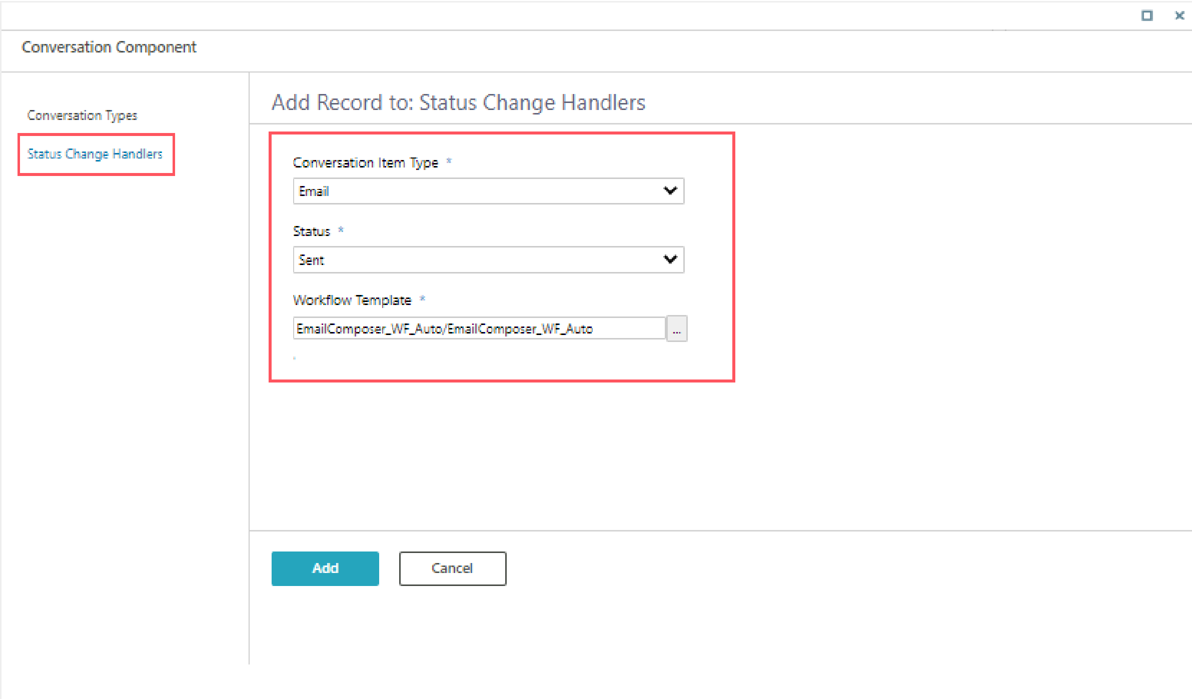
Clients API example

Configure conversation item events for event handling

The Conversation item events hook up to implementations, and make them capable of performing actions on conversation item status change. You can configure the system to initiate a specific workflow and perform action when an email conversation item changes to a specific status.

You can configure the conversation item events to hook up to implementations at the solution level.

1. In the Administration site, open a Solution from the list.
2. In the Solution Components section, click **Conversation Component**.
3. In the Conversation Component window, click **Status Change Handlers**.
4. Click **Add New Record**.
5. Select **Conversation Item Type** as Email.
6. Select **Status**, Sent or Undelivered.
7. Select **Workflow Template**, the workflow to initiate for conversation item.
8. Click **Add**.



9. In the selected workflow, add a Built-in Command Listener activity to listen to the conversation item event.

V9.4

Overview

Conversations View is a solution-level component that enables the end users to have all the case related conversations at one place within a case. This reusable component can be combined in a form within any solution-based workflow.

Conversation types

All conversation related items are saved in a product database table. The Email conversation type is the out-of-the-box conversation type available in Cora SeQuence. You can add any custom conversation type to your solutions per need.

For configuring email conversation type, see this article.

For adding custom conversation type, see this article.

Add Conversations View to Cora SeQuence forms

1. Add or edit a form or task activity in a workflow.
2. Add the following script to the form's source in the relevant location.

```
<sq:ConversationView runat="server" Id="Conversations">sq:ConversationView>
div>
```

NOTE
Conversations View is supported only in workflows that are in the context of a Solution.

Conversations view UI

Consider the following design parameters for Conversations view in Solutions.

- Supported resolution in the form is:
  - Height: Minimal height for the view in the form (empty state) is 100px
  - Width: 33% - 100% of the screen; minimum 600px
- View is responsive.
- Most recent item is listed on top, from newest to oldest, or last update date descending.

Built-in command

The Create Conversation Item built-in command lets end users insert conversation items into the Conversations View. You can add a General conversation item, or a specific conversation item related to the type of conversation available in the system. The general items are added using a system default type called Conversation Item. This type uses the default template, and is the basis of all conversation items.

Command parameters

- WF instance Id
- conversationItemState
- Title
- Body
- Attachments

IMPORTANT
To execute built-in-command that write into the Conversations View, make sure you have *Execute* permissions in the master workflow.

Conversation solution configuration

You can configure the conversation types to be enabled or disabled in the relevant solution. The list of conversation types is loaded from all of the system-available solutions.

When conversation types are disabled, there is no option to add them through the UI, built-in-command, or API. Any existing disabled conversation items are displayed with the default template.
The Delete action is allowed only through API.

These reusable solution components are deployed with the solution package.

Conversations View client API

The Conversations View component exposes four events to define additional actions and configurations.

- addConversationViewItemActionClicked: is called when a user clicks on an action related to a specific item.

For example, click on Reply.

- addConversationViewItemToolbarActionClicked: is called when a user clicks on the conversation view toolbar buttons.
  For example, click on New email.
- addConversationViewParametersBinding: is called when Conversation View binds the Conversation Item parameters.
  This event helps implementation set the Email Composer properties.
- addConversationViewActionCompleted: is called when a user closes the Email Composer window, either by clicking the X button, or by clicking Save or Send.

All the events receive the args object parameters that include:

- args.details.actionName: The selected action. For example, New.
- args.details.itemType: The Conversation item type on which the action applies. For example, Email.
- args.detail.parameters: The Conversation item related parameters. For example, windowTitle for Email Composer.

Clients API example