unqork

Platform Release Notes

Release Highlights

unqork

🎉 Release Highlights 🎉

Introducing UDesigner Dark Mode

UDesigner now offers new color themes to the Unqork platform. After logging into your environment, use the **Experience Dark Mode** tile in the UDesigner homepage's **Discovery** section.

You can select one of two themes when working on the Unqork platform:

- **Light Mode**: Unqork's default user experience. Experience the Unqork platform in light blue and white hues.
- **Dark Mode**: Unqork's newest theme that lets Creators experience the Unqork platform in dark blue and gray hues.

Quick Start

Edit Module

Reopen **CSC Test** and pick up where you left off configurating.

[Edit Module]

Discover

Experience Dark Mode!

Some pages may not fully support Dark Mode. Adjust anytime in Display Settings.

[Enable Dark Mode]

Create a Gen AI Application

You've installed the GenAI Connector already. Click Get Started below to begin buildi...

[Get Started]

Note: Dark Mode is currently in an experimental state and we are actively improving it. All pages might not display in the Dark Mode theme. Creators can switch between Light and Dark Mode without any impact to their work.

Display Settings ✕

ⓘ **Dark mode is currently an experimental feature**
Please note that not all pages may display in the Dark Mode theme. We're actively improving this feature, thank you for your patience!

⦿ Light Mode   ○ Dark Mode

[Cancel] [Save & Close]

unqork

🎉 Release Highlights 🎉

Introducing the Vega Chart Component

The **Vega Chart component** creates a graphical display of data sets in an application. In the Module Builder, **Creators** configure the Vega Chart component using one or more chart types and data sets. In Express View, the Vega Chart lets **end-users** visualize the data sets in a meaningful and intuitive way. Below are some common examples of when to use the Vega Chart component:

- To display a scatter plot to track the number of sales for each sales agent in your department.
- To display a pie chart to show the percentages of end-user **submissions**.
- To display a bar graph that shows product consumption by demographic.

> **Note**: The Vega Chart component is a **Vega (2.0)** runtime component and is not supported in the **Centauri (1.0)** runtime.

unqork

Enhancements

💡 UDesigner Parity

Components

With this release, UDesigner now supports the following Classic Designer components with a new look and feel:

- Advanced DataGrid component
- Browser Storage component
- Data Grid component
- DataMapper component
- Dynamic Grid component
- File Storage component
- Intl Phone Number component
- Map V2 component
- Protected Field component
- Repeater component
- Rich Text Editor component
- Signature component
- Table component

unqork

💡 Enhancements

Bulk Data Operations (BDO)

- **Bulk Data Operation: Delete**: Added new functionality for the BDO Delete feature to perform time-bound bulk data deletions.

Centauri

- **Browser Storage component**: Added three operations to the component's Operations Builder called **Set Persisted Data**, **Read Persisted Data**, and **Delete Persisted Data**. Creators can now use these settings to set, read, and delete data in IndexedDB.
- **Browser Storage component**: Added a new field called **Operation Name** to the component's Operations Builder. It's optional, but supports operation chaining—for example, `$operationContext.OperationName.result` .

Classic Designer

- **Address v1, Address Search, Date Input, and Email component**: Added a **Read Only** setting to these components. If Read Only is enabled, the read-only attribute is applied to these components in Express View. End-users can only view, select, or copy its contents—the fields cannot be edited. When using VoiceOver on the input field, it informs the end-user that this field contains selectable text and text cannot be entered.
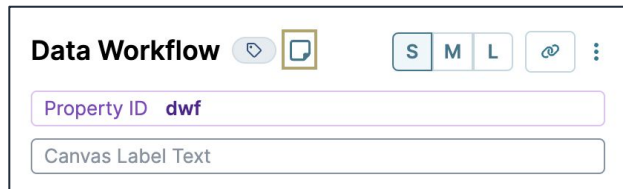
💡 Enhancements (Continued)

Module Builder

- **Resource Button (?)**: Removed the blue Resource (?) button at the bottom of the Unqork platform to promote easy access to the components tray.

UDesigner

- **Component Notes**: UDesigner components now offer a new icon at the top of the Settings drawer that Creators can click to add notes on configuration and functionality.

- **Logic components**: Improved the Unit Testing experience by removing it from a modal in the component's settings and displaying it as its own section for easier access.

unqork

💡 Enhancements (Continued)

UDesigner (Continued)

- **Logic components**: Logic components with **Disable Execution** enabled display as a grey color to represent they are in a disabled state.
- **Module Builder Performance**: Improved Module Builder performance and faster load times.
- **Module Builder Canvas**: Added an enhancement where, even with a component's Settings drawer open, Creators can scroll through the Module Builder canvas to view the other configured components. The component being edited is also highlighted on the canvas to make it easier for Creators to determine the component they are editing.
- **Module Builder Canvas**: Improved the drag-and-drop functionality to make it easier to add components at the bottom of container components on the canvas.
- **Module Outline Search**: Improved scrolling when searching for components in the Module Outline search when including imported elements.
- **Plug-In component**: After selecting an Unqork internal service from the Internal Service drop-down menu, click the Learn More link opens that specific service's API documentation in a new tab.
- **Transforms**: Improved horizontal scrolling and formatting in the **Transforms** field when in read-only view.

unqork

💡 Enhancements (Continued)

Vega

- **Events**: Added Event Scoping to signals. Event Scoping lets Creators specify the event scopes they want applied to the events they are listening for in their component's signals.

  Available scopes include the following:

  - **Self**: The configured signal only handles events that originate from the component being edited. For example, when a Button component is configured, it only handles clicks to the button.
  - **Child**: The configured signal only handles events that originate from the component children being edited. For example, when a Panel component is configured, it only handles clicks on the Panel component's children. Creators can combine **Child** with **Self** to handle clicks on the element and on the children.

  Any previously configured signals for events will default to the **Self** scope. The only exception is for DOM events, like on-click. DOM events default to **Self** and **Child** scopes. If you add a new signal for a DOM event, only the **Self** scope exists. However, a **Child** scope can be added.
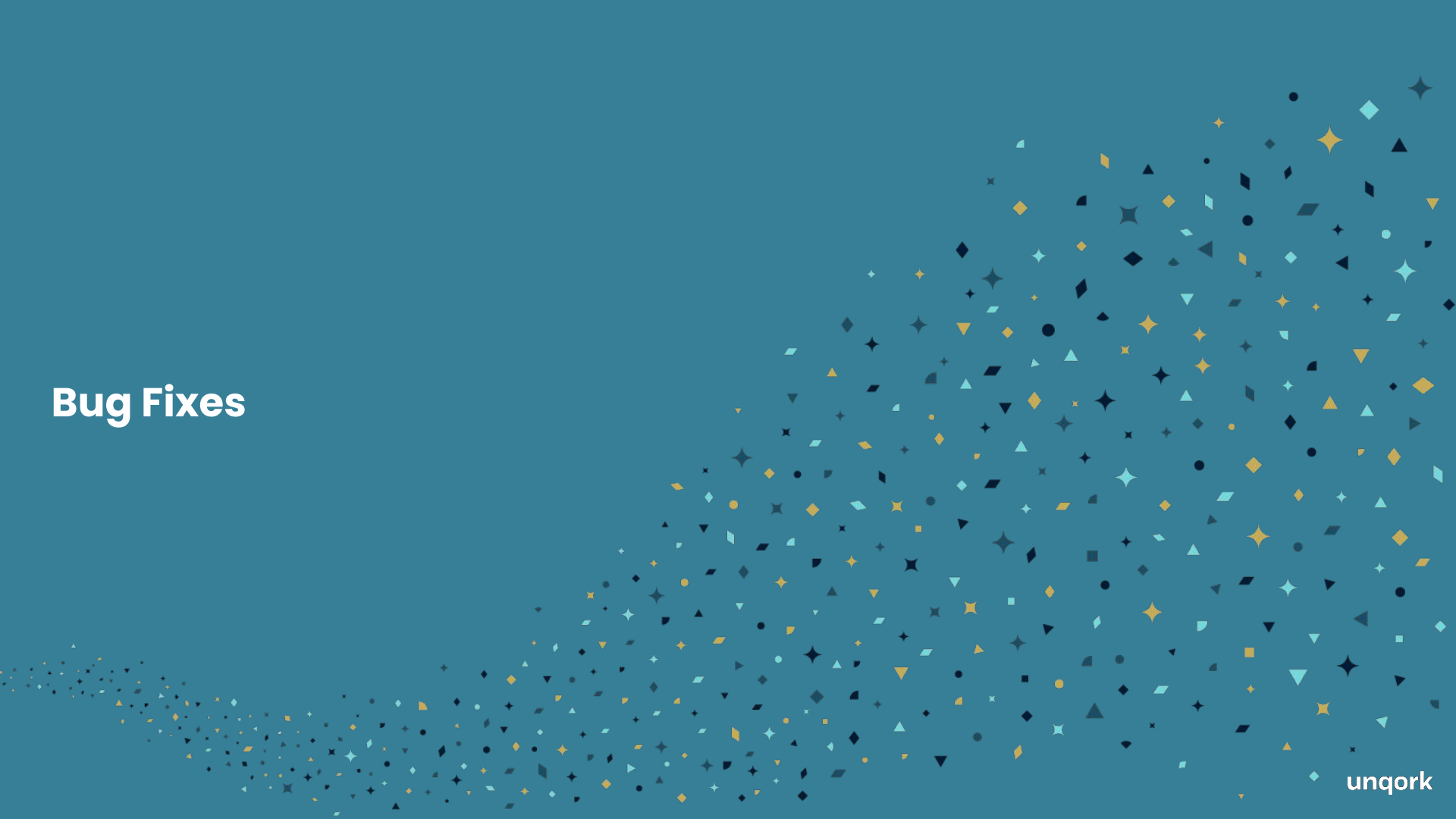
unqork

💡 Enhancements

Vega (Continued)

- **Performance**: Improved Vega runtime codebase to improve page load speeds and performance.
- **Simple Select component**: The component now supports the Operations Builder in it's Settings drawer.
- **Vega Table component**: Added a new operation called **Table Set Column Width**, which lets Creators set the column widths and flex their behavior.

Visual Query Builder

- **Visual Query Builder**: Added new functionality to build **lookups** and **unions** when creating queries. Creators can also use the new **Projections** field to determine which fields they want included in the response.

> **Note**: The Visual Query Builder is in Production Early Access. To learn more, please contact your Unqork representative.

unqork

Bug Fixes

🐞 Bug Fixes

Centauri

- **Data Workflow component**: Fixed an issue where, in certain situations, events that were issued to update the component state experienced long delays before the changes were rendered in the UI.
- **Decisions component**: Fixed an issue where various Output Types did not function correctly when targeting the cells and rows of a Dynamic Grid component.

  These outputs include:

  - customClass
  - required
  - hidden
  - value

- **Plug-In component**: Fixed two issues affecting Creators when using an internal service that requires module selection. When selecting a module from the component's Settings menu, the module name now displays correctly instead of the unexpected ******. The **Edit Module** link below the component's Module drop-down menu has been restored. Previously, it did not display after selecting a module.

unqork

🐞 Bug Fixes

Classic Designer

- **Components**: Fixed an issue when a Creator moves existing components on the Module Builder canvas. When moved, the component's ID changes.
- **Decisions component**: Fixed an issue where, in certain situations, the component's Inputs and Outputs tables disappear in Classic Designer and throw a DevTools Console error.

UDesigner

- **Application and Module Creation**: Fixed an issue where the browser would become unresponsive when trying to create an application or module with a lengthy name, followed by a special character.
- **Application Settings**: Fixed an issue when viewing Login, Logout, Header, and Footer modules in the Application Settings modal. Depending on which page the Creator is on in the application, when viewing the Application Settings, Login, Logout, Header, and Footer modules would sometimes disappear from the modal.
- **Application Versioning**: Fixed an issue where validating the uniqueness of a module name was not working for versioned applications.

unqork

🐞 Bug Fixes (Continued)

UDesigner (Continued)

- **Comments**: Fixed an issue when editing a comment in the Module Builder with tagged components. After making edits and saving the module, the tag is removed from the comment.
- **Data Workflow component**: Fixed an issue where hovering over an operator's tooltip displayed the tooltip to the far right of the screen and made it difficult to read.
- **Decisions component**: Fixed an issue in UDesigner when a Decisions component is added to a Field Group component. After saving the module and refreshing, the Decisions component becomes hidden in the Module Builder. The only way to view the component is switching to Classic Designer.
- **Module Path**: Fixed an issue where adding an invalid character in the module path and saving it would cause it to spin indefinitely and not save.
- **Module Settings**: Fixed an issue where canceling out of the Module Settings modal still retained values that were entered in the modal.
- **Snippets**: Fixed an issue where adding a snippet to the canvas did not trigger the Save button to change color.

unqork

🐞 Bug Fixes (Continued)

Vega

- **Button component**: Fixed an issue in Vega where the Button component's **Validate** Action Type fails and returns an error.
- **Dynamic Refresh**: Fixed an issue when an embedded Vega module is unloaded when interacting with a Panel component leveraging Dynamic Refresh. In certain situations, validations would fail.
- **Execute Module API**: Fixed an issue where, in certain situations, executing an API module where a submission ID is stored in a Hidden component returns the submission ID as an unexpected object instead of a string.
- **Focus operation**: Fixed an issue where, in certain situations, the Focus operation was not functioning as expected when executing using the `onFirstRender` event.
- **HTML Element component**: Fixed an issue where, in certain situations, zero values were being displayed as an empty string when rendered by the component.
- **iFrame component**: Fixed an issue where the component was displaying in the incorrect location of the component tray. It has been moved to the correct alphabetical order.
- **Map Keys operator**: Fixed an issue where, in certain situations, the Map Keys operator would unexpectedly remove fields from the output if it did not have a mapping.

unqork

🐞 Bug Fixes (Continued)

Vega (Continued)

- **Merge operator**: Fixed an issue where, in certain situations, the Merge operator unexpectedly dropped nested keys.
- **Remote Executes**: Fixed an issue when non-administrator users attempted to make calls with the **Execute as Super-User** setting enabled. The calls either failed or took 10-15 seconds to return the correct data. Remote execute calls once again honor the **Execute as Super-User** setting for all API modules.
- **Radio Buttons component**: Fixed an issue where the value was being cleared if the options were set by a Data Reference Key.
- **Set Property operation**: Fixed an issue where, in certain situations, the `vegaValue` logic Output Type resulted in an undefined value when sending data from Centauri to a Cross-Runtime-Support-enabled Vega Panel component.

unqork