# FlowPanel Control Reference

Dynamically arrange multiple elements.

**Reference** **Controls** **Viewer** FlowPanel

---

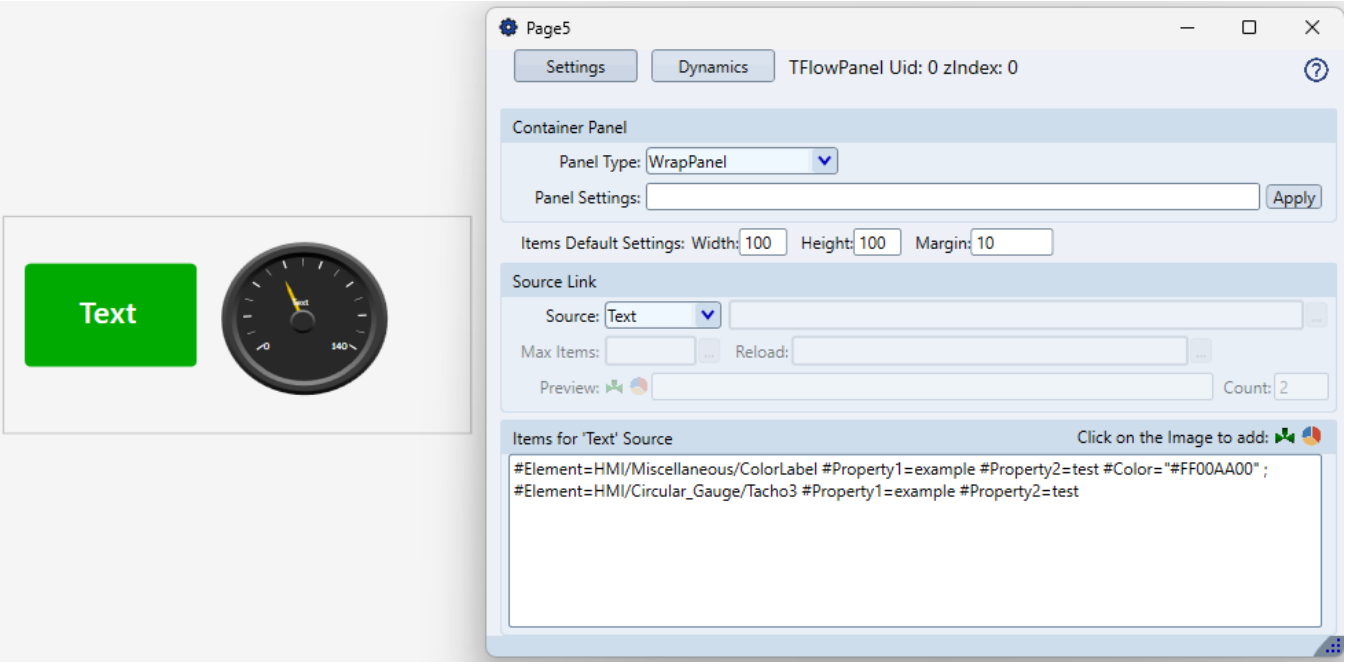
Allows the easy configuration of multiple symbols and components in a repetitive and responsive way. It offers a user-friendly interface that automatically adjusts the layout of items based on panel characteristics and the datasource. This topic explains how to configure the FlowPanel, its layout options (panel type), data source types, and how the control dynamically adapts to the runtime environment.


Requirements
- Panel Types
- Data Sources

Configuration

Runtime Execution
- Responsive Behavior
- Dynamic DataSource Updates


---




---

## Requirements

This component is Portable. It runs both on Windows and on Web Pages hosted in any platform.

### Panel Types

The FlowPanel supports several panel types that control the layout of symbols or components. Each panel type has specific behavior, described below:

#### 1. WrapPanel

- Layout: Aligns items horizontally, breaking to the next line when the width limit is reached.
- Ideal for: Horizontal listings or galleries where the number of items can vary.
- Responsiveness: Line breaks adjust the items as the window resizes.

#### 2. Canvas

- Layout: Allows positioning of items at absolute coordinates (x, y).
- Ideal for: Scenarios where precise control of the position is needed, like maps or complex layouts.
- Responsiveness: Less responsive since positions are fixed and may not adapt well to resizing.

#### 3. DockPanel

- Layout: Items are anchored to the edges (top, bottom, left, right) or fill the remaining space in the center.
- Ideal for: Layouts focused on distributing items around a central element or along the screen edges.
- Responsiveness: Items resize according to the available space.

### 4. Grid

- Layout: Organizes items in rows and columns, with flexible sizing for each cell.
- Ideal for: Tabular or grid-based layouts where items need to be aligned in a structured manner.
- Responsiveness: The layout adjusts well to resizing, maintaining the proportion between rows and columns.

### 5. StackPanelHorizontal

- Layout: Aligns items horizontally, without line breaks.
- Ideal for: Simple horizontal listings, like menus or toolbars.
- Responsiveness: Horizontal alignment does not automatically adjust to screen width, which may result in scrolling.

### 6. StackPanelVertical

- Layout: Aligns items vertically, one below the other.
- Ideal for: Simple vertical listings, like task lists or side menus.
- Responsiveness: Items remain vertically stacked, and the height of the FlowPanel may adjust to fit all items.

### 7. UniformGrid

- Layout: Distributes items evenly in a grid where all cells have the same size.
- Ideal for: Displays of items with fixed size and balanced layout, like image galleries.
- Responsiveness: The size of the cells and the number of items per row/column automatically adjust as the screen is resized.

## Data Sources

The FlowPanel supports four types of data sources: **Text**, **StringTag**, **Array**, and **Object**. The choice of data source determines how items are generated and displayed in the panel.

### 1. Text

- Format: `#Element=<control or symbol name> #<property name1>=<property value1> #<property name2>=<property value2> etc`
    - `where, Element and Property Name are case sensitive.`
- Use: Manual configuration of symbols or components/controls through static text. Allows linking properties to dynamic values from SCADA tags.
- Example:
```
#element=LampSymbol #Color=red #Size=medium
#element=PumpControl #State=on #FlowRate=Tag.Tag1
```

### 2. StringTag

- Format: Same as **Text**, but linked to dynamic system tags.
- Use: Allows linking symbols or components/controls properties to dynamic values from SCADA tags.
- Example:
StringTag = `"#element=LampSymbol #Color=red #Size=medium #element=PumpControl #State=on #FlowRate=Tag.Tag1"`

### 3. Array

- Format: String array, where each entry follows the same format as **Text** or **StringTag**.
- Use: Configures multiple items at once, offering flexibility for scenarios where several items need to be dynamically generated.
- Example:
```
StringTag1[0] = #Element=ValveSymbol #Status="Open"
StringTag1[1] = #Element=ConveyorBelt #Speed=100 #State=Tag.State
["#element=ValveSymbol #status=open", "#element=ConveyorBelt #speed=fast"]
```

### 4. Object

- Format:
    - DataTable, where each row represents an item (element), and each column defines the properties of the item.
    - UserType, where each member represents a properties of the item.  The tag of this template type can be an array.
- Use: Advanced configuration where data comes from tables, such as query results from a database.
- Example DataTable:

| element | color | state |
|---|---|---|
| LampSymbol | red | on |
| MotorSymbol | green | off |

- Example Template: Tag array of 2 position with datatype of a template

| member | index 0 | index 1 |
|---|---|---|
| LampSymbol | red | on |
| MotorSymbol | green | off |

---

## Configuration

To configure a FlowPanel:

1. Go to **Displays / Draw**.
2. On the Components Panel, select **Viewer**, then **FlowPanel**.
3. Click or drag-and-drop it on the Drawing area to use it.
4. Double-click the component to open the configuration window.

| **FlowPanel Settings** | |
|---|---|
| **Field** | **Description** |
| Panel Type | Sets the Panel mode for the component. Options are:<br><br>**WrapPanel:** Positions child elements in sequential position from left to right, breaking content to the next line at the edge of the containing box.<br><br>**Canvas:** arrange the items using the standard Canvas container.<br><br>**DockPanel:** arrange the items using the standard Dock container.<br><br>**Grid:** Defines a flexible grid area consisting of columns and rows.<br><br>**StackPanelHorizontal:** Arranges child elements into a single line oriented horizontally.<br><br>**StackPanelVertical:** Arranges child elements into a single line oriented vertically.<br><br>**UniformGrid:** Used to arrange children in a grid with all equal cell sizes. |
| Panel Settings | Allows configure panel setting properties following the same format of data source. Example: #Background=Green, where Background is a property of the panel. |
| Items Source | Determines the source of the items displayed in the FlowPanel. Can be bound to a list or collection, and users can specify the type of data, such as text or objects. Options are:<br><br>**Text**: Allows the FlowPanel to display a list of simple text items. Useful for representing straightforward string values such as machine states or operational modes.<br><br>**StringTag:** Enables the FlowPanel to display a list of items generated from a pre-existing Tag of the Text type. This feature is particularly useful when multiple FlowPanels across different displays need to present the same information consistently.<br><br>**Array**: Allows the FlowPanel to display items from an array. Suitable for predefined sets of options, such as lists of sensor IDs or operational steps, enabling efficient management of uniform data.<br><br>**Object**: Allows the FlowPanel to display items from a collection of objects. Ideal for complex data representation where each item has multiple attributes, such as equipment lists with properties like name, ID, and status.<br><br>The object can be Databases Queries, or any realtime Tag, AssetPath, or property from the system namespaces |
| Max Items | Specifies the maximum number of items that the FlowPanel can display or hold. Limits the number of items to prevent overload and improve performance. |
| Reload | Specifies a command or function that reloads the items in the FlowPanel. Ensures that the FlowPanel displays the most current data, useful for dynamic or real-time data sources. |
| Quantity | Defines how many items will be sourced in the design preview. (not applied to runtime) |

| | |
|---|---|
| Width | Defines the width of each panel, ensuring that each panel is properly sized according to the user's preferences and fits seamlessly within the FlowPanel. |
| Height | Defines the height of each panel, ensuring that each panel is properly sized according to the user's preferences and fits seamlessly within the FlowPanel. |
| Margin | Specifies the margin for each panel, ensuring sufficient spacing to avoid a cramped layout. |

## Runtime Execution

### Responsive Behavior

The FlowPanel is a responsive control, meaning it adjusts the arrangement of items based on the screen size and the size and type of panel used. When the window or browser is resized, the FlowPanel automatically reorganizes items to optimize the available space. (if the display is configured as Responsive)

- In panels like **WrapPanel** and **UniformGrid**, items may shift between rows/columns as the screen size changes.
- In the **Canvas** panel, as positions are fixed, the layout does not dynamically adapt to resizing.
- **DockPanel** and **Grid** panels offer great flexibility, dynamically distributing items to better utilize the available space.

### Dynamic DataSource Updates

The FlowPanel's data source can be updated dynamically. To do this, simply trigger a reload on the control, and it will refresh with the new data source. This feature is useful in systems where data or conditions change frequently.

**Solution Example**

Check out FlowPanel Dynamic Symbols Example to see this feature in action.

**In this section...**