Dashboards Reference

Build dashboards from components.

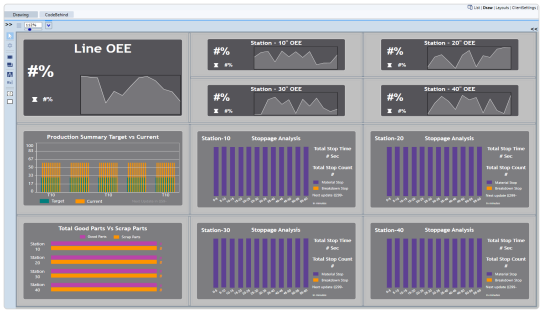
Dashboards Drawing (Reference) provide responsive grid-based layouts that automatically adapt to screen sizes, with unique support for both structured dashboards and responsive canvas designs.

Dashboard capabilities:

- **Grid Dashboards** - Auto-arranging blocks
- **Responsive Canvas** - Positioned elements with responsive anchoring
- Mobile-optimized stacking
- Drag-drop design
- Container nesting
- Cross-platform support

Two Approaches to Responsive Design
Creating Dashboard Displays
Dashboard Editor Interface
Dashboard Elements
Dashboard Properties
Responsive Behavior
Advanced Features
Responsive Canvas Alternative
Dashboard vs Canvas Comparison
Best Practices Checklist
Troubleshooting

Unique Feature: Unlike typical dashboard tools, this platform supports responsiveness in **BOTH** grid-based dashboards AND **traditional canvas displays (responsive canvas).**



Two Approaches to Responsive Design

1. Grid Dashboards (This Page)

- Block-based layout
- Auto-arranging cells
- Mobile stack switching
- Container-based design

2. Responsive Canvas

- Vector drawing with anchoring
- Proportional scaling
- Position locking
- Mixed fixed/responsive elements

See Responsive Design for canvas responsiveness

Creating Dashboard Displays

1. Navigate to **Displays Draw**
2. Click **Create New Document**
3. Select **BasicDashboard** template
4. Design in grid mode

Dashboard template automatically enables:

- Grid-based editor
- Block arrangements
- Responsive behavior
- Mobile optimization

Dashboard Editor Interface

Drawing Mode Changes

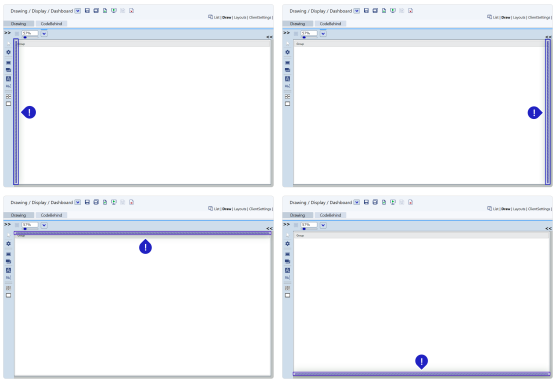
When Dashboard selected:

- Vector tools disabled
- Grid blocks enabled
- Drag-drop positioning
- Container management

Available Operations

- **Add Elements** - Drag from Components Panel
- **Move Blocks** - Drag to reposition
- **Create Rows/Columns** - Auto-grid expansion
- **Delete Elements** - Select and Delete key
- **Configure Properties** - Right panel

When mouse is over the borders, it will highlight that when dropping a component, a new row or column is created



Dashboard Elements

Components

From Components Panel:

- Charts (Trend, Bar, Pie, XY)
- Gauges (Circular, Linear)
- Data displays (Grid, Table)
- Custom symbols
- Input controls

Containers

Special layout elements:

- **Group** - Grid within grid
- **Vertical Stack** - Column layout
- **Horizontal Stack** - Row layout
- **Border Panel** - Framed content
- **Expander** - Collapsible section

Dashboard Properties

Block Configuration

Property	Function	Location
Title	Block header text	Dashboard Item Title
Margin	External spacing	Dashboard Item Margin

Padding	Internal spacing	Dashboard Item Padding
Fill	Background color	Appearance Fill

Styling Options

- **Theme** - Apply color schemes
- **Borders** - Block outlines
- **Shadows** - Depth effects
- **Corners** - Radius settings

Responsive Behavior

Desktop Mode

- Grid maintains columns/rows
- Blocks resize proportionally
- Minimum sizes respected
- Aspect ratios preserved

Mobile Mode

- Switches to stack panel
- Single column layout
- Touch-optimized spacing
- Scroll-enabled

Breakpoints

Automatic switching based on:

- Screen width
- Device type
- Orientation
- User settings

Advanced Features

Dynamic Titles

Support for runtime values:

```
Title: {Client.DateTime}
Title: {Tag.Temperature} °C
Title: {Server.Status}
```

Runtime Interactions

- **Maximize** - Full-screen block
- **Resize** - Drag corner handle
- **Collapse** - Minimize block
- **Reorder** - Drag to rearrange

CodeBehind Customization

csharp

```csharp
public void OnDashboardCustomizeItem(TCell item)
{
    item.TitleElement.TextAlignment = TextAlignment.Center;
    item.TitleElement.FontStyle = FontStyles.Italic;
    item.TitleElement.FontSize = 20;
    item.TitleElement.Foreground = Brushes.Black;
}
```

Responsive Canvas Alternative

For mixed vector/responsive designs:

1. Create standard Canvas display
2. Set **OnResize** to **Responsive**
3. Check layout anchors per element
4. Combine fixed and responsive objects

Benefits:

- Precise positioning
- Vector graphics
- Selective responsiveness
- Complex layouts

Dashboard vs Canvas Comparison

Aspect	Grid Dashboard	Responsive Canvas
Layout	Auto-arranging blocks	Manual positioning
Design	Drag-drop blocks	Vector drawing tools
Resize	Reflow content	Proportional scaling
Mobile	Stack switching	Scaled view
Use Case	Data displays	Process graphics
Complexity	Simple	Advanced

Best Practices Checklist

- **Choose Right Approach** - Grid for data, Canvas for graphics
- **Plan Breakpoints** - Test mobile early
- **Use Containers** - Group related items
- **Set Minimum Sizes** - Prevent crushing
- **Test Scaling** - Verify all sizes
- **Optimize Mobile** - Simplify for small screens
- **Consider Hybrid** - Mix approaches when needed

Troubleshooting

Blocks not arranging:

- Check grid settings
- Verify container nesting
- Review minimum sizes
- Test breakpoints

Mobile view issues:

- Preview in mobile mode
- Check stack settings
- Verify touch targets
- Test orientation

Performance problems:

- Limit chart complexity
- Optimize update rates
- Reduce container depth
- Simplify mobile view

In this section...
