TrakSYS™ Training

Day 2

Training Overview

Training Agenda

<table>
<tr><th>Day 1</th><th>Day 2</th><th>Day 3</th><th>Day 4</th><th>Day 5</th></tr>
<tr><td>TrakSYS Overview
Setup and Installation
Configuration Basics
Navigation Introduction
Functionality and Data</td><td>Content Pages
Values Dictionary
Visual Pages
Content Parts and Features
Users and Permissions</td><td>Performance Management
Content Page Functionality
Batching and Storage Systems
Template Systems
Task Configuration</td><td>API Introduction
Logic Service
Data Management Service
TrakSYS Extensibility
Sites, Translations, and Audit</td><td>Production Scheduling
Alerts and Notifications
Inventory Management
Statistical Process Control
Support and Resources</td></tr>
<tr><td colspan="3">Introduction Training</td><td colspan="2"></td></tr>
<tr><td colspan="2"></td><td colspan="3">Advanced Training</td></tr>
<tr><td colspan="5">Comprehensive Training</td></tr>
</table>

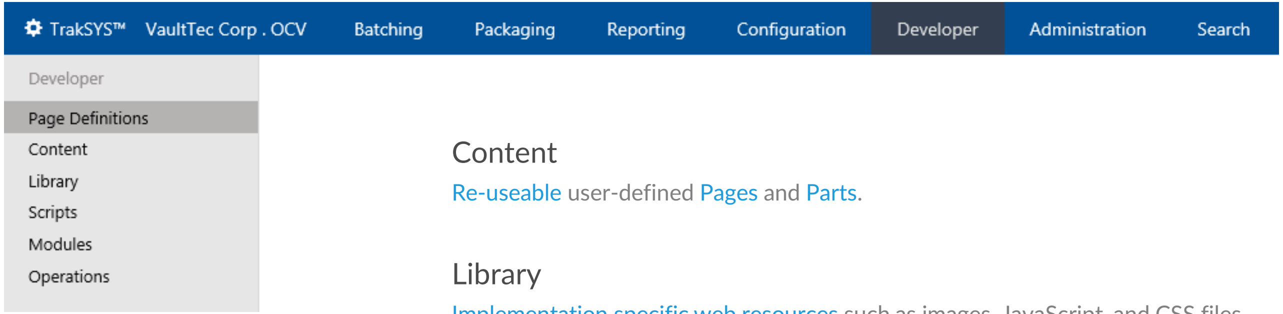
Page Construction

Training Objectives

Introduce the basic concepts for creating and configuring application Pages within TrakSYS.

Describe the different types of Pages that can be used to create a TrakSYS user interface experience.

Developer Section



Content

Re-useable user-defined Pages and Parts.

Library

Implementation specific web resources such as images, JavaScript, and CSS files.

Scripts

Definitions for all .NET script classes used in Logic Service and the Web.

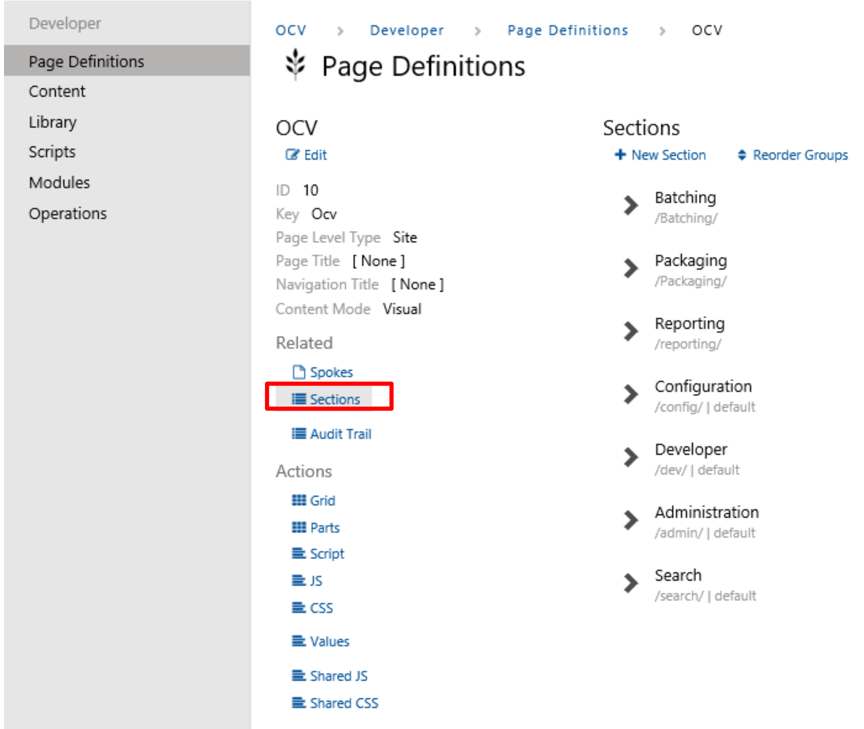
Modules

Modules that can be executed by the Data Management Service.

Operations

Advanced entities that allow the extension and customization of the TrakSYS menus.

Page Definitions Hub



Developer

Page Definitions

Content

Library

Scripts

Modules

Operations

OCV › Developer › Page Definitions › OCV

Page Definitions

OCV

Edit

ID 10

Key Ocv

Page Level Type Site

Page Title [None]

Navigation Title [None]

Content Mode Visual

Related

Spokes

Sections

Audit Trail

Actions

Grid

Parts

Script

JS

CSS

Values

Shared JS

Shared CSS

Sections

+ New Section ⬍ Reorder Groups

› Batching /Batching/

› Packaging /Packaging/

› Reporting /reporting/

› Configuration /config/ | default

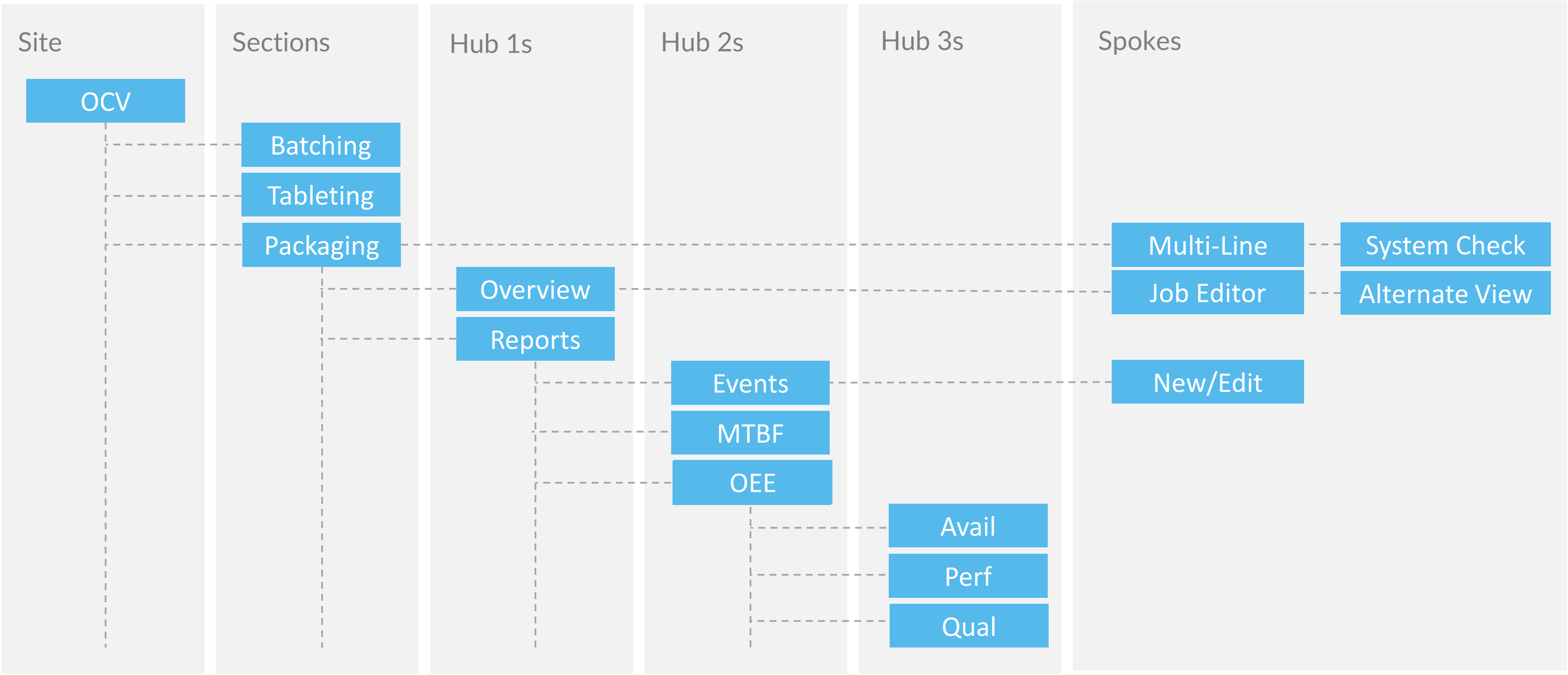
› Developer /dev/ | default

› Administration /admin/ | default

› Search /search/ | default

- Used to create, configure and organize the Page hierarchy that makes up a TrakSYS Solution
- Manage Content Pages, Visual and Open type Page Definitions
- Contains both user-defined and built-in Sections / Hubs
- Note that Sites, Sections and Hubs (1 and 2) also contain Spokes

Page Hierarchy Review



Site

Sections

Hub 1s

Hub 2s

Hub 3s

Spokes

OCV

Batching

Tableting

Packaging

Multi-Line

System Check

Overview

Job Editor

Alternate View

Reports

Events

New/Edit

MTBF

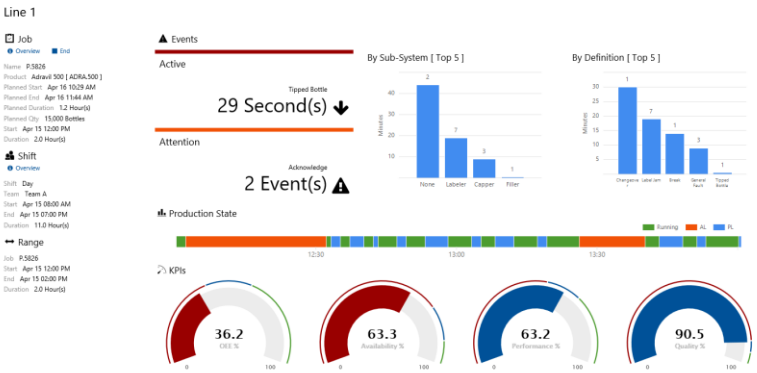
OEE

Avail

Perf

Qual

Page Definition Types | Content Page



Line 1

Job

Overview End

Name P.5826
Product Adravil 500 [ADRA.500]
Planned Start Apr 16 10:29 AM
Planned End Apr 16 11:44 AM
Planned Duration 1.2 Hour(s)
Planned Qty 15,000 Bottles
Start Apr 15 12:00 PM
Duration 2.0 Hour(s)

Shift

Overview

Shift Day
Team Team A
Start Apr 15 08:00 AM
End Apr 15 07:00 PM
Duration 11.0 Hour(s)

Range

Job P.5826
Start Apr 15 12:00 PM
End Apr 15 02:00 PM
Duration 2.0 Hour(s)

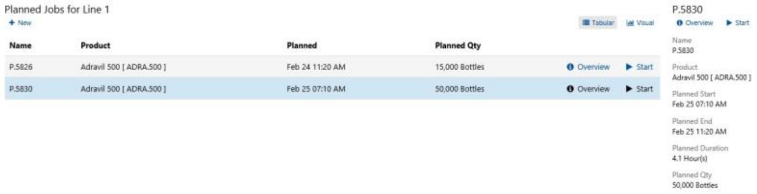
Events

Active

Tipped Bottle
29 Second(s)

Attention

Acknowledge
2 Event(s)



Planned Jobs for Line 1

+ New

Name	Product	Planned	Planned Qty		
P.5826	Adravil 500 [ADRA.500]	Feb 24 11:20 AM	15,000 Bottles	Overview	Start
P.5830	Adravil 500 [ADRA.500]	Feb 25 07:10 AM	50,000 Bottles	Overview	Start

P.5830

Overview Start

Name
P.5830

Product
Adravil 500 [ADRA.500]

Planned Start
Feb 25 07:10 AM

Planned End
Feb 25 11:20 AM

Planned Duration
4.1 Hour(s)

Planned Qty
50,000 Bottles

Content Page (out-of-the-box)

Content built into the TrakSYS product. These Pages cannot be altered (other than what is exposed via Parameters and configuration settings).

Visual

Pages that are constructed by assembling Content Parts using a grid based visual editor. The behavior of the Visual Page is primarily manipulated by setting properties through user friendly form based interfaces. For more advanced needs, the server side behavior of the page can be controlled using ASP.NET script.

Open

Pages built by hand-coding the user interface portion (HTML, CSS and JavaScript), and the server side code (ASP.NET). While these types of pages are more complex to create, there are no limitations or restrictions as to the functionality and layout that can be created.

Page Definition Types | Visual

Content Page (out-of-the-box)

Content built into the TrakSYS product. These Pages cannot be altered (other than what is exposed via Parameters and configuration settings).

Visual

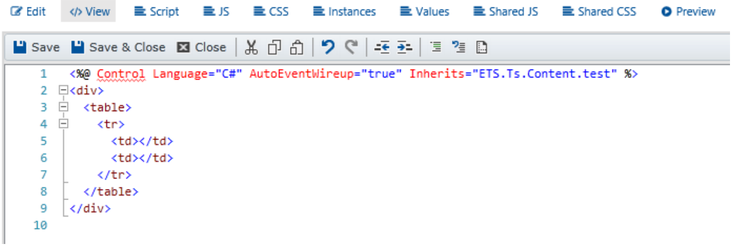
Pages that are constructed by assembling Content Parts using a grid based visual editor. The behavior of the Visual Page is primarily manipulated by setting properties through user friendly form based interfaces. For more advanced needs, the server side behavior of the page can be controlled using ASP.NET script.

Open

Pages built by hand-coding the user interface portion (HTML, CSS and JavaScript), and the server side code (ASP.NET). While these types of pages are more complex to create, there are no limitations or restrictions as to the functionality and layout that can be created.

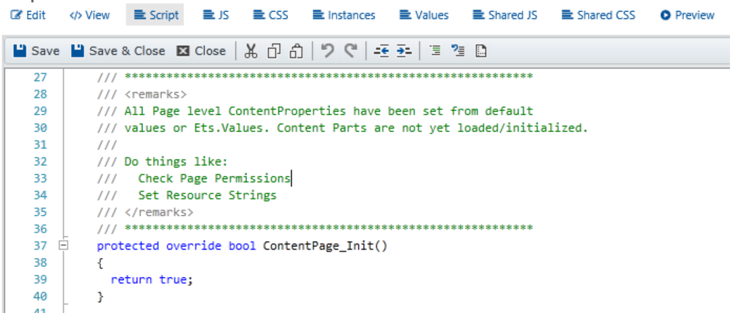
Page Definition Types | Open

View



```
<%@ Control Language="C#" AutoEventWireup="true" Inherits="ETS.Ts.Content.test" %>
<div>
  <table>
    <tr>
      <td></td>
      <td></td>
    </tr>
  </table>
</div>
```

Script



```
    /// **********************************************************
    /// <remarks>
    /// All Page level ContentProperties have been set from default
    /// values or Ets.Values. Content Parts are not yet loaded/initialized.
    ///
    /// Do things like:
    ///   Check Page Permissions
    ///   Set Resource Strings
    /// </remarks>
    /// **********************************************************
    protected override bool ContentPage_Init()
    {
      return true;
    }
```

Content Page (out-of-the-box)

Content built into the TrakSYS product. These Pages cannot be altered (other than what is exposed via Parameters and configuration settings).

Visual

Pages that are constructed by assembling Content Parts using a grid based visual editor. The behavior of the Visual Page is primarily manipulated by setting properties through user friendly form based interfaces. For more advanced needs, the server side behavior of the page can be controlled using ASP.NET script.

Open

Pages built by hand-coding the user interface portion (HTML, CSS and JavaScript), and the server side code (ASP.NET). While these types of pages are more complex to create, there are no limitations or restrictions as to the functionality and layout that can be created.

Content Page Definitions

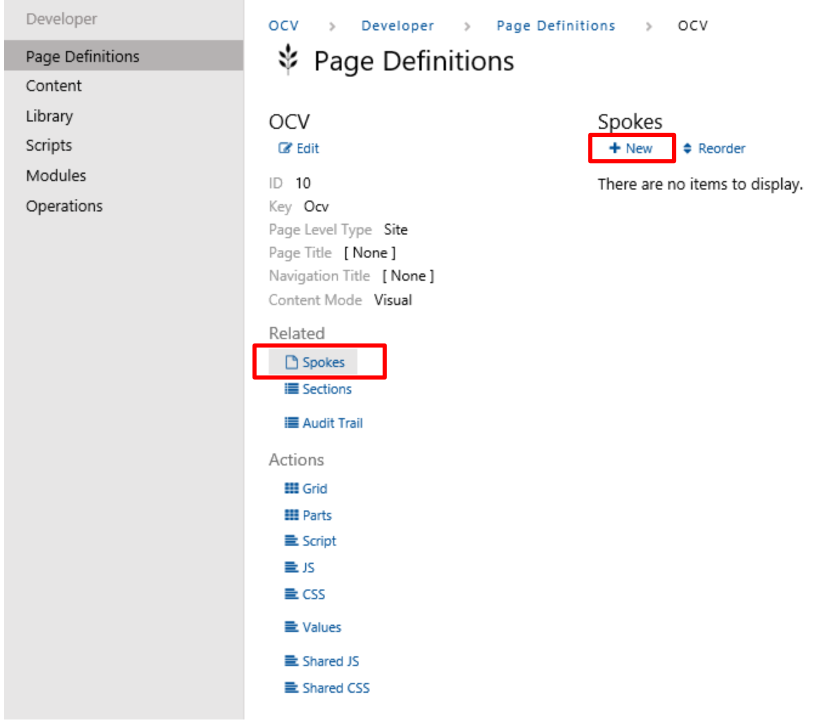
Training Objectives

Understand how a standard Content Page is added to the Page hierarchy.

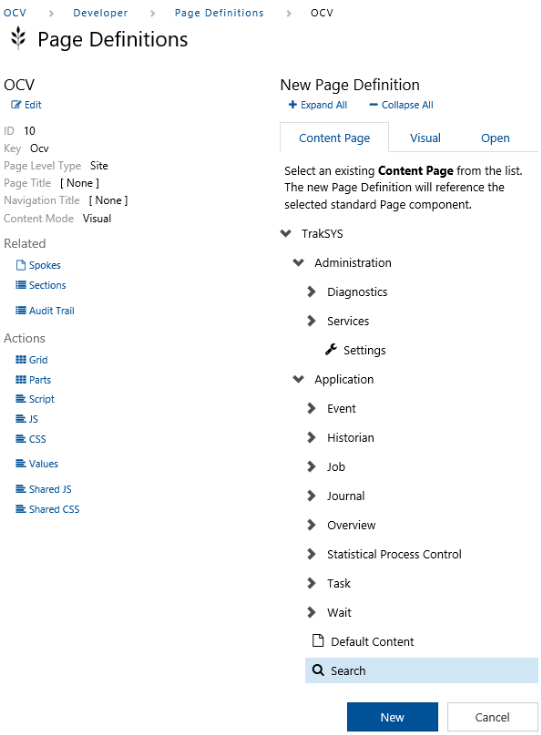
Explore Content Page properties pages and how they can be used to easily configure and manipulate the behavior of standard Pages.

Adding a Content Page

- Locate the parent Page in the Hierarchy and select the appropriate child Page Definition List
- Select the New menu option at the top of the Page Definition List in Slice 2
- Once a Page Definition is added to the Hierarchy, it is available in the TrakSYS User Interface



Content Page Catalog



OCV › Developer › Page Definitions › OCV

Page Definitions

OCV
Edit

ID 10
Key Ocv
Page Level Type Site
Page Title [None]
Navigation Title [None]
Content Mode Visual

Related
- Spokes
- Sections
- Audit Trail

Actions
- Grid
- Parts
- Script
- JS
- CSS
- Values
- Shared JS
- Shared CSS

New Page Definition
+ Expand All − Collapse All

Content Page | Visual | Open

Select an existing **Content Page** from the list. The new Page Definition will reference the selected standard Page component.

- TrakSYS
  - Administration
    - Diagnostics
    - Services
    - Settings
  - Application
    - Event
    - Historian
    - Job
    - Journal
    - Overview
    - Statistical Process Control
    - Task
    - Wait
  - Default Content
  - Search

New | Cancel

Search

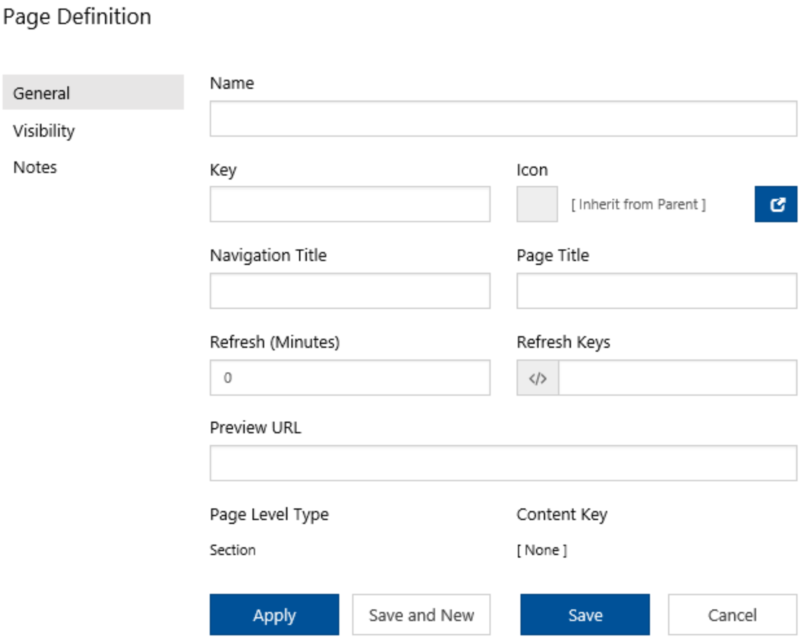
Name Search
Key ETS.Search

Description
Allows text based searching through the various TrakSYS database entities.

- Select the first tab in the New Page to choose a pre-built Content Page from the Catalog
- Content Pages are grouped by Application Function
- Both Standard TrakSYS Content Pages, and user-defined Content Pages are listed in the Catalog
- Selecting a Content Page reveals a description and details in the Slice 3.

Page Definition | General Properties

- **Name**
  Primary name of the Page
- **Key**
  Unique identifier for the page within its parent navigational element. Use to build the URL address for accessing the Page. The Key will also be used for the .NET class name for the Page, so it is recommended to use PascalCase conventions.
- **Navigation Title**
  Optional name to be displayed in the navigation links.
- **Page Title**
  Optional name to be displayed at the top of the Page.
- **Refresh Minutes**
  Timer interval for auto-refreshing the Page contents.
- **Refresh Keys**
  Identifier(s) for enabling push page refreshes.



Page Definition

Icon: [Inherit from Parent]

Refresh (Minutes): 0

Page Level Type: Section

Content Key: [None]

Page URLs by Key

Base Pages URL

http://servername/ts/pages/

http://servername/ts/pages/site

http://servername/ts/pages/ocv

http://servername/ts/pages/site/section

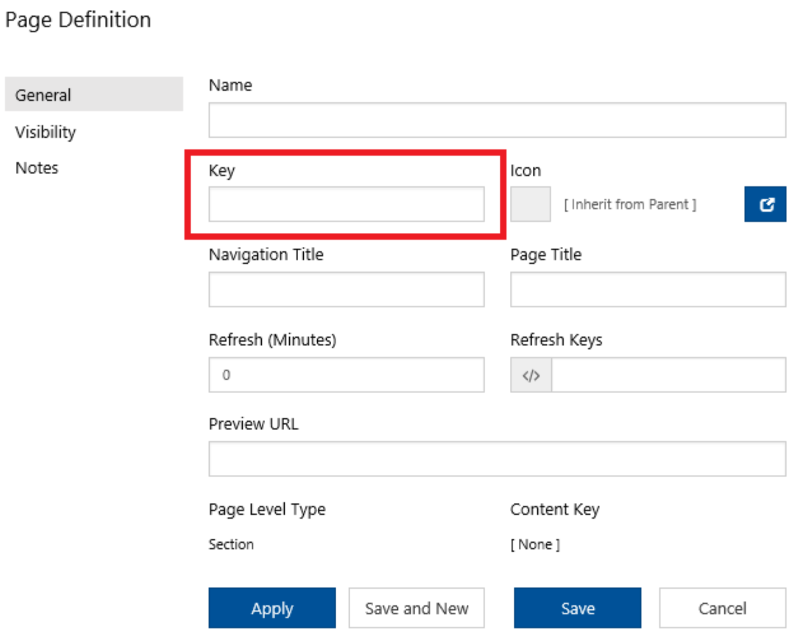
http://servername/ts/pages/ocv/packagingss

http://servername/ts/pages/site/section/hub1

http://servername/ts/pages/ocv/packaging/line1

http://servername/ts/pages/site/section/spoke

http://servername/ts/pages/ocv/packaging/overview



Page Definition

General

Visibility

Notes

Name

Key

Icon [Inherit from Parent]

Navigation Title

Page Title

Refresh (Minutes): 0

Refresh Keys

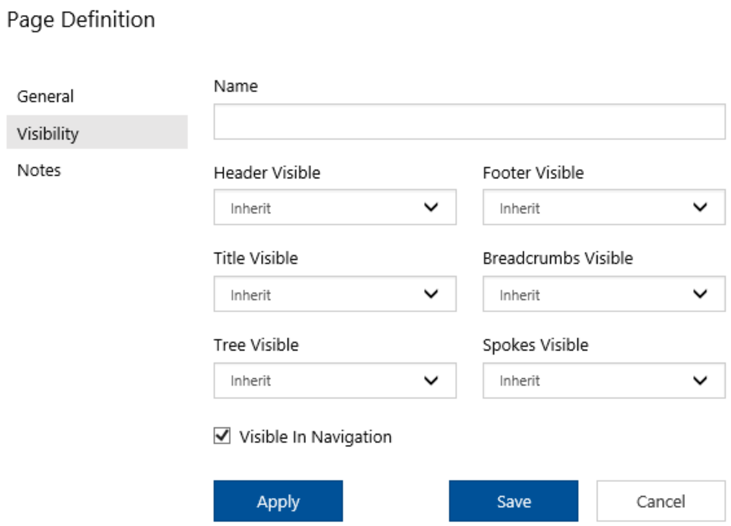
Preview URL

Page Level Type: Section

Content Key: [None]

Page Definition | Visibility Properties

- Settings to control the Visibility of Navigational elements displayed on the Page
- Options include…
  - **True**
    Always show the element.
  - **False**
    Never show the element.
  - **Inherit**
    Show the element based on the parent’s setting.
- Visible in Navigation shows/hides the Page from being displayed in the Navigation



Page Definition

Name

Header Visible: Inherit

Footer Visible: Inherit

Title Visible: Inherit

Breadcrumbs Visible: Inherit

Tree Visible: Inherit

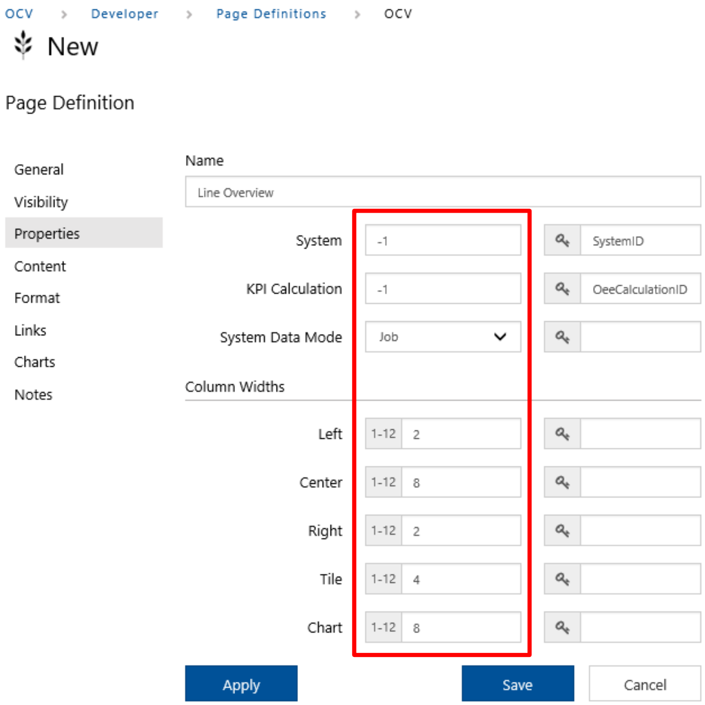
Spokes Visible: Inherit

☑ Visible In Navigation

Content Page Properties

- Specify property values at Design Time
- Fixed values configured by the Page Developer
- Default values if Dynamic values are not Present

Dynamic values will be covered in a later section.



OCV › Developer › Page Definitions › OCV

New

Page Definition

- General
- Visibility
- Properties
- Content
- Format
- Links
- Charts
- Notes

Name: Line Overview

System	-1	SystemID
KPI Calculation	-1	OeeCalculationID
System Data Mode	Job	

Column Widths

Left	1-12 2	
Center	1-12 8	
Right	1-12 2	
Tile	1-12 4	
Chart	1-12 8	

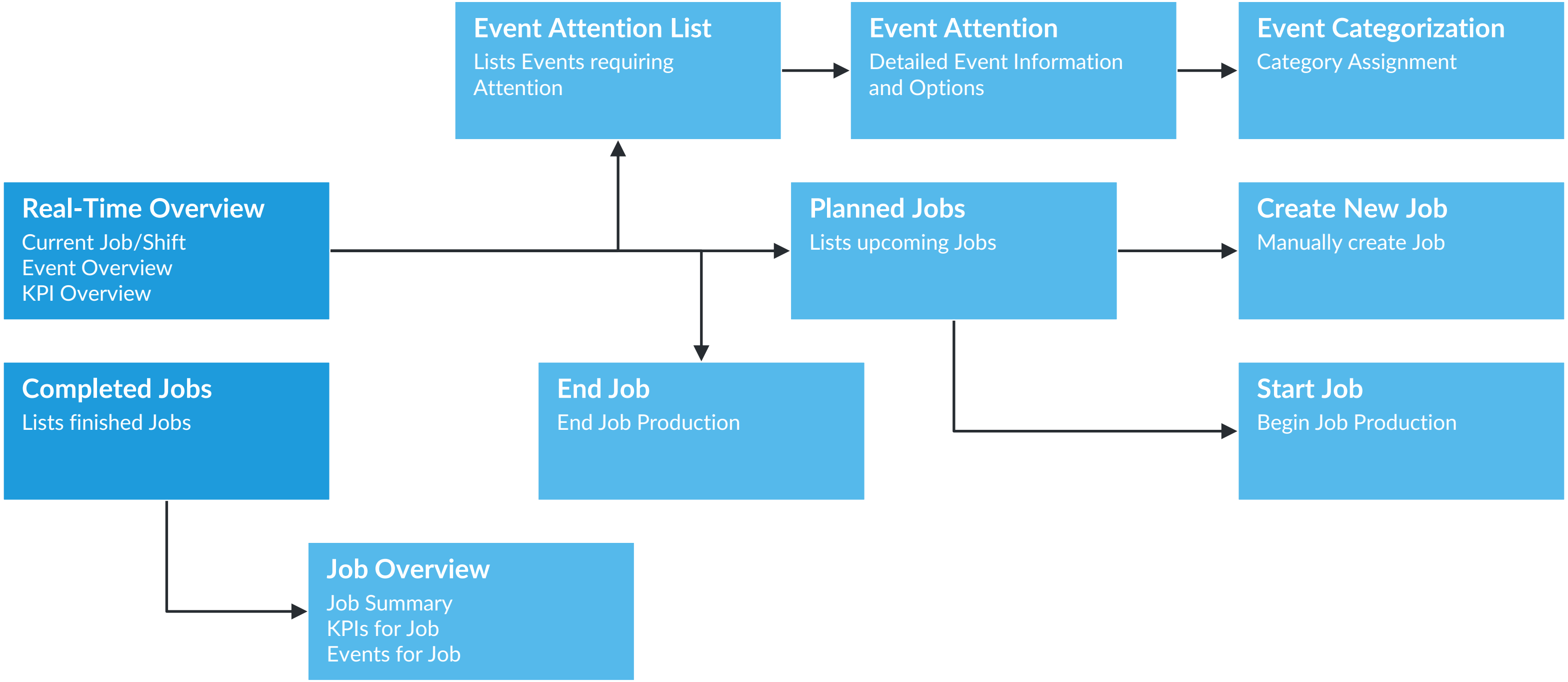
Content Page Definitions
System Discrete Real-Time Overview

Training Objectives

Explore the page properties of the System Discrete Real Time Overview Content page.

Understand common properties and capabilities of Content Pages.

Performance Related Pages



Event Attention List
Lists Events requiring Attention

Event Attention
Detailed Event Information and Options

Event Categorization
Category Assignment

Real-Time Overview
Current Job/Shift
Event Overview
KPI Overview

Planned Jobs
Lists upcoming Jobs

Create New Job
Manually create Job

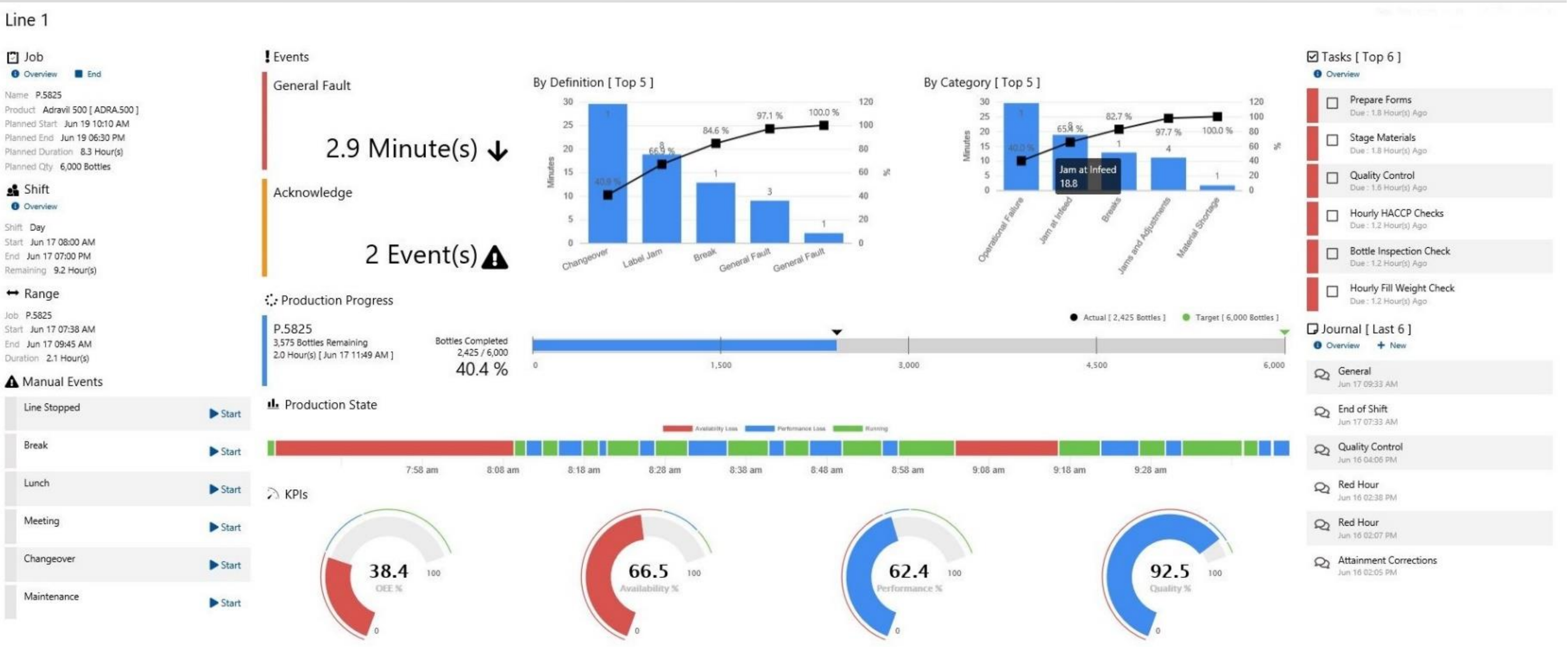
Completed Jobs
Lists finished Jobs

End Job
End Job Production

Start Job
Begin Job Production

Job Overview
Job Summary
KPIs for Job
Events for Job

System Real-Time Overview



Line 1

Job

Overview End

Name P.5825
Product Adravil 500 [ADRA.500]
Planned Start Jun 19 10:10 AM
Planned End Jun 19 06:30 PM
Planned Duration 8.3 Hour(s)
Planned Qty 6,000 Bottles

Shift

Overview

Shift Day
Start Jun 17 08:00 AM
End Jun 17 07:00 PM
Remaining 9.2 Hour(s)

Range

Job P.5825
Start Jun 17 07:38 AM
End Jun 17 09:45 AM
Duration 2.1 Hour(s)

Manual Events

Line Stopped	▶ Start
Break	▶ Start
Lunch	▶ Start
Meeting	▶ Start
Changeover	▶ Start
Maintenance	▶ Start

Events

General Fault

2.9 Minute(s) ↓

Acknowledge

2 Event(s) ⚠

By Definition [Top 5]

By Category [Top 5]

Production Progress

P.5825
3,575 Bottles Remaining
2.0 Hour(s) [Jun 17 11:49 AM]

Bottles Completed
2,425 / 6,000
40.4 %

Production State

KPIs

38.4 OEE %

66.5 Availability %

62.4 Performance %

92.5 Quality %

Tasks [Top 6]

Overview

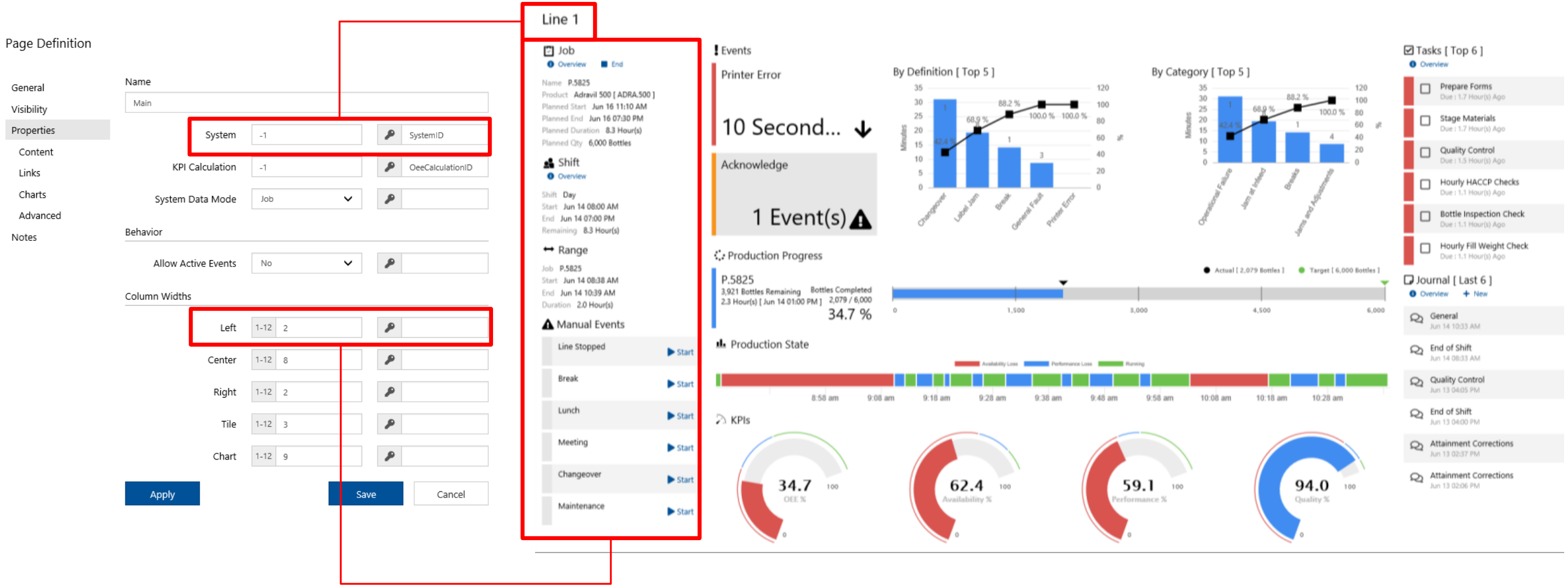
- □ Prepare Forms
  Due : 1.8 Hour(s) Ago
- □ Stage Materials
  Due : 1.8 Hour(s) Ago
- □ Quality Control
  Due : 1.6 Hour(s) Ago
- □ Hourly HACCP Checks
  Due : 1.2 Hour(s) Ago
- □ Bottle Inspection Check
  Due : 1.2 Hour(s) Ago
- □ Hourly Fill Weight Check
  Due : 1.2 Hour(s) Ago

Journal [Last 6]

Overview + New

- General
  Jun 17 09:33 AM
- End of Shift
  Jun 17 07:33 AM
- Quality Control
  Jun 16 04:06 PM
- Red Hour
  Jun 16 02:38 PM
- Red Hour
  Jun 16 02:07 PM
- Attainment Corrections
  Jun 16 02:05 PM

System Real-Time Overview Properties



Page Definition

- General
- Visibility
- Properties
  - Content
  - Links
  - Charts
  - Advanced
- Notes

Name: Main

System	-1	SystemID
KPI Calculation	-1	OeeCalculationID
System Data Mode	Job	

Behavior

Allow Active Events	No	

Column Widths

Left	1-12	2	
Center	1-12	8	
Right	1-12	2	
Tile	1-12	3	
Chart	1-12	9	

Apply | Save | Cancel

Line 1

Job
Overview | End

Name P.5825
Product Adravil 500 [ADRA.500]
Planned Start Jun 16 11:10 AM
Planned End Jun 16 07:30 PM
Planned Duration 8.3 Hour(s)
Planned Qty 6,000 Bottles

Shift
Overview

Shift Day
Start Jun 14 08:00 AM
End Jun 14 07:00 PM
Remaining 8.3 Hour(s)

Range

Job P.5825
Start Jun 14 08:38 AM
End Jun 14 10:39 AM
Duration 2.0 Hour(s)

Manual Events

Line Stopped	Start
Break	Start
Lunch	Start
Meeting	Start
Changeover	Start
Maintenance	Start

Events

Printer Error
10 Second...

Acknowledge
1 Event(s)

Production Progress

P.5825
3,921 Bottles Remaining
2.3 Hour(s) [Jun 14 01:00 PM]
Bottles Completed
2,079 / 6,000
34.7 %

Production State

KPIs

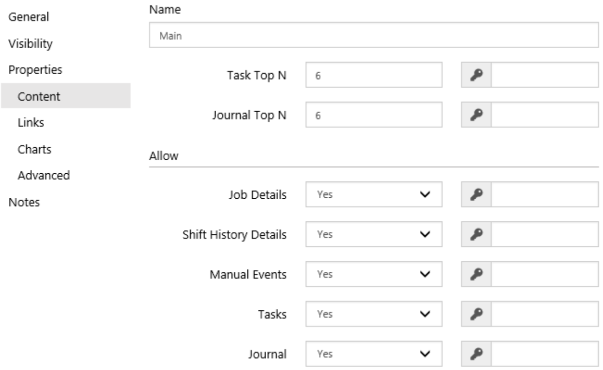
Tasks [Top 6]
Overview

- □ Prepare Forms Due : 1.7 Hour(s) Ago
- □ Stage Materials Due : 1.7 Hour(s) Ago
- □ Quality Control Due : 1.5 Hour(s) Ago
- □ Hourly HACCP Checks Due : 1.1 Hour(s) Ago
- □ Bottle Inspection Check Due : 1.1 Hour(s) Ago
- □ Hourly Fill Weight Check Due : 1.1 Hour(s) Ago

Journal [Last 6]
Overview | + New

- General Jun 14 10:33 AM
- End of Shift Jun 14 08:33 AM
- Quality Control Jun 13 04:05 PM
- End of Shift Jun 13 04:00 PM
- Attainment Corrections Jun 13 02:37 PM
- Attainment Corrections Jun 13 02:06 PM

System Real-Time Overview Properties

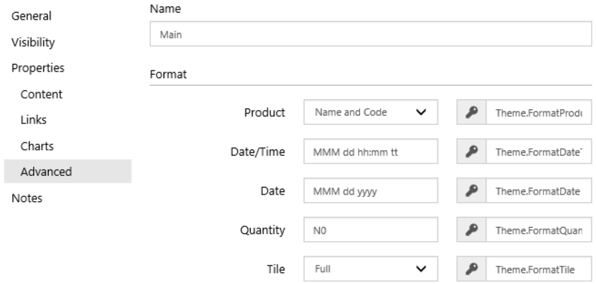


Name: Main

Task Top N	6	
Journal Top N	6	

Allow

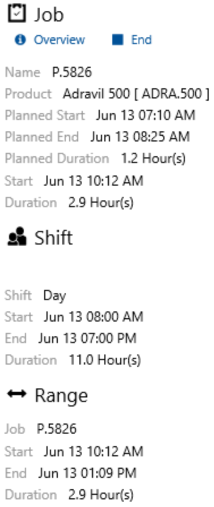
Job Details	Yes	
Shift History Details	Yes	
Manual Events	Yes	
Tasks	Yes	
Journal	Yes	



Name: Main

Format

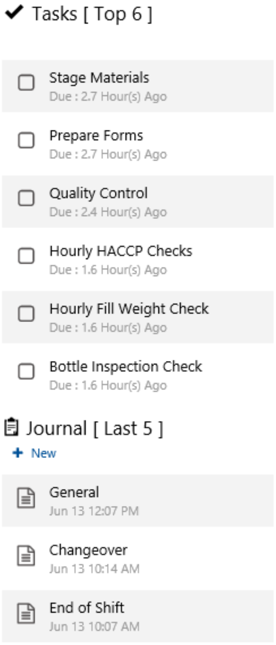
Product	Name and Code	Theme.FormatProd
Date/Time	MMM dd hh:mm tt	Theme.FormatDate
Date	MMM dd yyyy	Theme.FormatDate
Quantity	N0	Theme.FormatQuan
Tile	Full	Theme.FormatTile



Name P.5826
Product Adravil 500 [ADRA.500]
Planned Start Jun 13 07:10 AM
Planned End Jun 13 08:25 AM
Planned Duration 1.2 Hour(s)
Start Jun 13 10:12 AM
Duration 2.9 Hour(s)

Shift Day
Start Jun 13 08:00 AM
End Jun 13 07:00 PM
Duration 11.0 Hour(s)

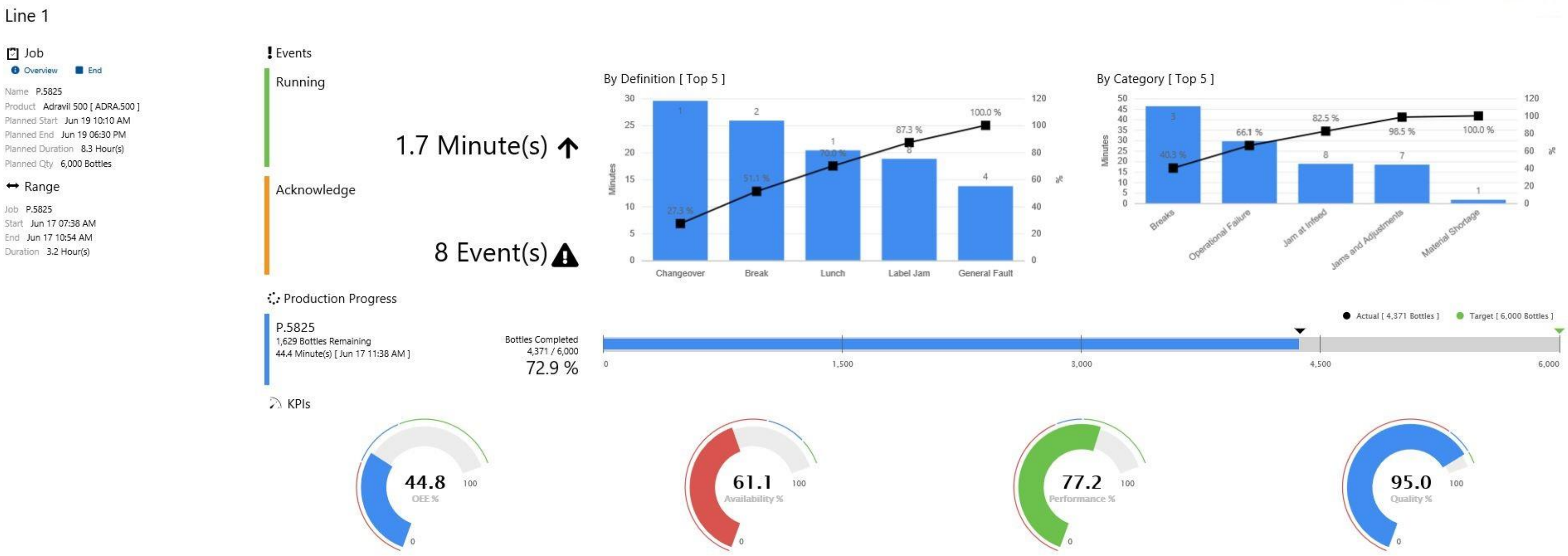
Job P.5826
Start Jun 13 10:12 AM
End Jun 13 01:09 PM
Duration 2.9 Hour(s)



- □ Stage Materials
  Due : 2.7 Hour(s) Ago
- □ Prepare Forms
  Due : 2.7 Hour(s) Ago
- □ Quality Control
  Due : 2.4 Hour(s) Ago
- □ Hourly HACCP Checks
  Due : 1.6 Hour(s) Ago
- □ Hourly Fill Weight Check
  Due : 1.6 Hour(s) Ago
- □ Bottle Inspection Check
  Due : 1.6 Hour(s) Ago

- General
  Jun 13 12:07 PM
- Changeover
  Jun 13 10:14 AM
- End of Shift
  Jun 13 10:07 AM

Adjusted System Real-Time Overview



Line 1

Job

Overview End

Name P.5825
Product Adravil 500 [ADRA.500]
Planned Start Jun 19 10:10 AM
Planned End Jun 19 06:30 PM
Planned Duration 8.3 Hour(s)
Planned Qty 6,000 Bottles

Range

Job P.5825
Start Jun 17 07:38 AM
End Jun 17 10:54 AM
Duration 3.2 Hour(s)

Events

Running

1.7 Minute(s)

Acknowledge

8 Event(s)

Production Progress

P.5825
1,629 Bottles Remaining
44.4 Minute(s) [Jun 17 11:38 AM]

Bottles Completed
4,371 / 6,000
72.9 %

KPIs

Replacement Pages

Remove & Replace

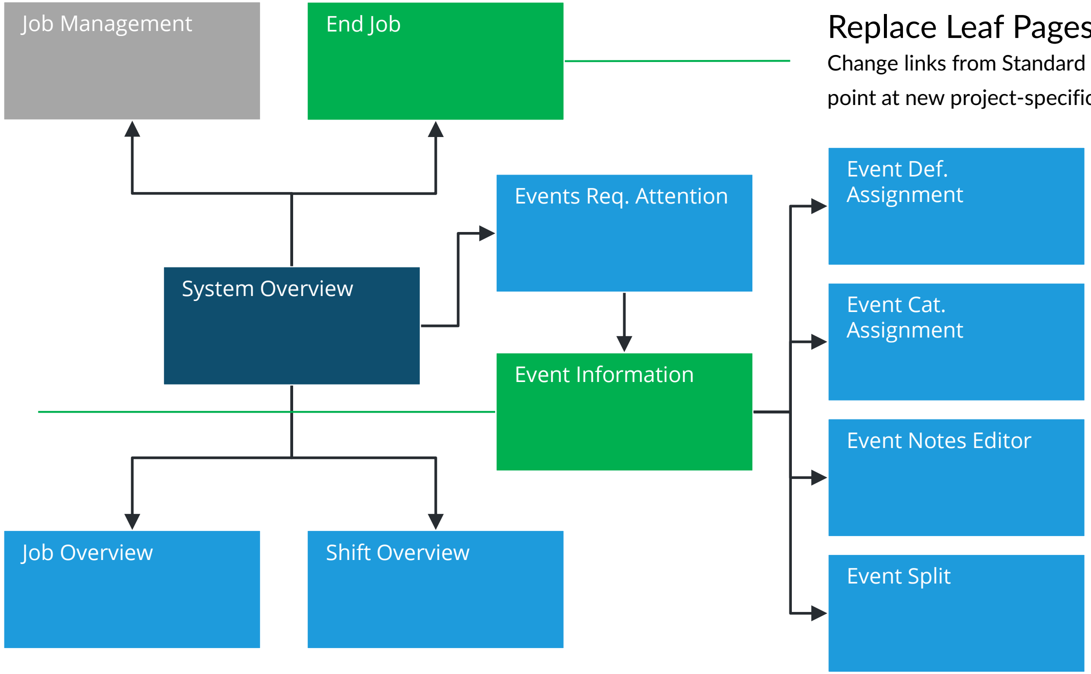
Simply swap out Standard Content Pages for project-specific Page Definitions

Replace Leaf Pages

Change links from Standard Content Pages to point at new project-specific Page Definitions

Replace Hub Pages

Project-specific Hubs cab link out to Standard Content Pages



Demonstration

- Navigate the Developer Page Definition Hub
- Create a new Content Page Definition
- Manipulate Content Page Properties

Lab 5

Values Dictionary

Training Objectives

Define the Values Dictionary and how it improves passing variables into and around Page execution.

Demonstrate how to add to and retrieve items from the Values Dictionary.

Understand how to view and troubleshoot the Values Dictionary using Page Trace.

Decoupling

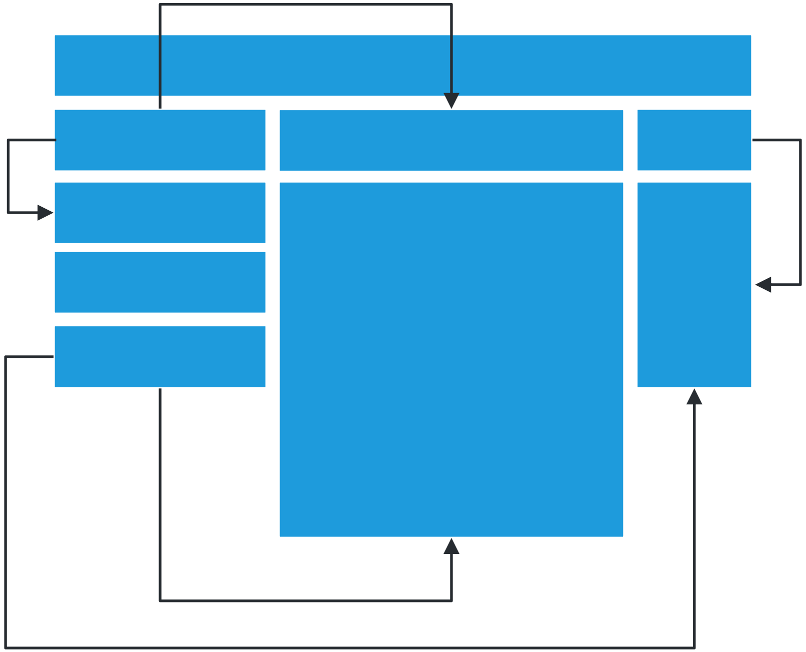
Coupling describes the degree of dependency between one entity to another entity. Often classes or objects.

Loose coupling is good because we don't want the components of our system to heavily depend on each other. We want to keep our system modular, where we can safely change one part without affecting the other.

Strong coupling usually occurs when entity A knows too much about entity B. If entity A makes too many assumptions about how entity B operates or how it is built, than there is a high risk that a change in entity B will affect entity A. This is because one of it's assumptions about entity B are now incorrect.

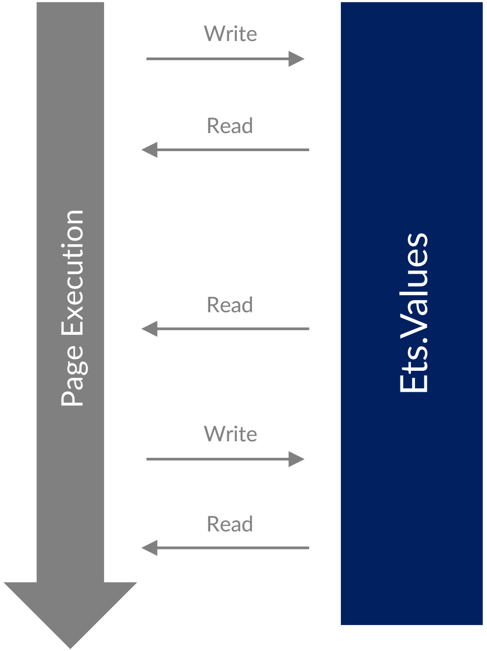
Decoupling

- Sharing/Passing data during Page Execution
  - Values Passed-In
  - Values Created or Manipulated by Parts and Script
  - Values Passed-On to Parts and Script later in the Execution
- Reduce Direct Contact and Coupling between Parts
- Think Answering Machines, Email, Texting vs. Face to Face Meetings and Scheduled Interactions



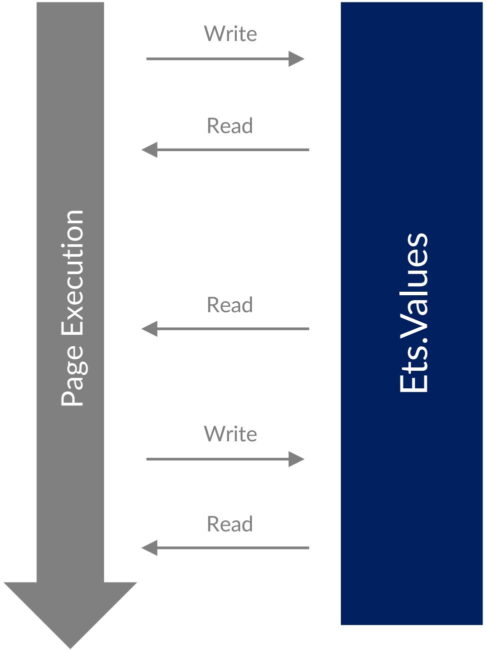
Values Dictionary

- Alternatively referred to as Values, “ETS Values”, or the Values Collection
- A repository for holding variables during Page Execution.
- Is cleared and rebuilt on every Page GET or POST (refresh)
- Is technically a .NET Dictionary<string, object>
    - String is the Key (or the way to look things up)
    - Object is the actual Value (it can be of any data type)
- Populated automatically by the TrakSYS Infrastructure



Values Dictionary

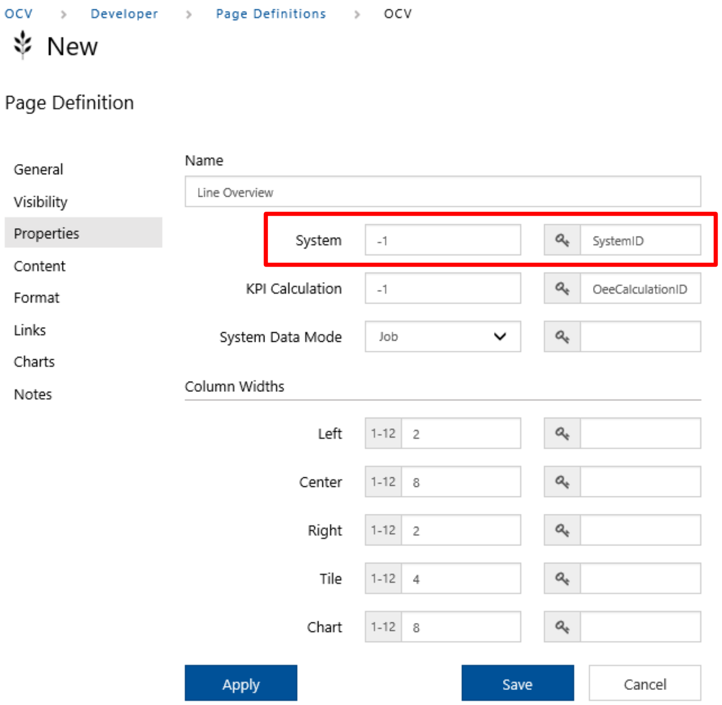
- Contains values from common ASP.NET dictionaries (loaded in this order)…
  - Site Values (TrakSYS)
  - Section Values (TrakSYS)
  - Hub1 Values (TrakSYS)
  - Hub2 Values (TrakSYS)
  - Spokes Values (TrakSYS)
  - Session (ASP.NET)
  - QueryString (ASP.NET)
  - Form (ASP.NET)
  - Page and Parts direct Loading
- As it is loaded, existing values are overwritten if a new one with the same Key already exists.



Writing to the Values Dictionary

- Passing values on the QueryString
  - http://localhost/ts/pages/ocv?SystemID=3
- Form controls during a POST (clicking save on a form)
- ASP.NET Script
  - this.Ets.Values[“SystemID”] = 3; // adds or overwrites
  - this.Ets.Values.ContainsKey(“SystemID”); // check if exists
- Added by developers using the Page Definition’s Values Editor (advanced – will be covered later)

Reading from the Values Dictionary



Name: Line Overview

System	-1	SystemID
KPI Calculation	-1	OeeCalculationID
System Data Mode	Job	

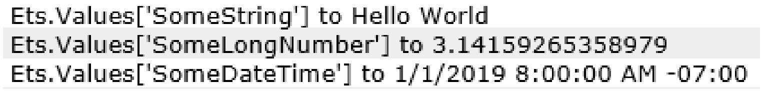
Column Widths

Left	1-12	2	
Center	1-12	8	
Right	1-12	2	
Tile	1-12	4	
Chart	1-12	8	

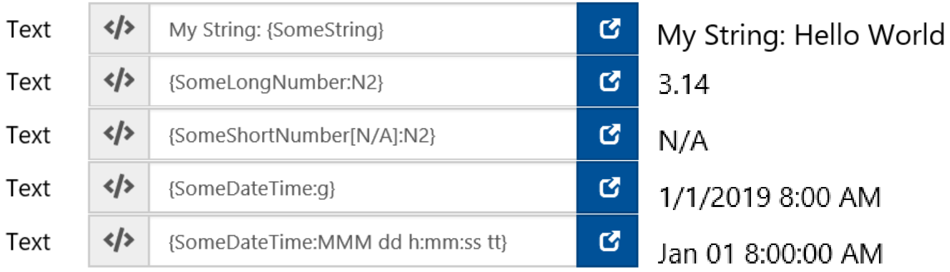
- Mapped via configuration to Page and Part Properties
- Expressions {SystemID}
- ASP.NET Script
  - int value = this.Ets.Values.GetAsInt(“SystemID", -1);
  - string value = this.Ets.Values.GetAsString(“Name", "");

Expressions

- Allow the retrieval of Values Dictionary items for integration into string display Properties
- Basic syntax is Curly Braces around the desired Values Dictionary Key {SystemID}
- Optional formatting suffix supports .NET Format Codes {Key:FormatCode}
- Default values can be provided using Square Brackets {SystemID[-1]}
- Property fields supporting Expressions are marked with the Script Icon



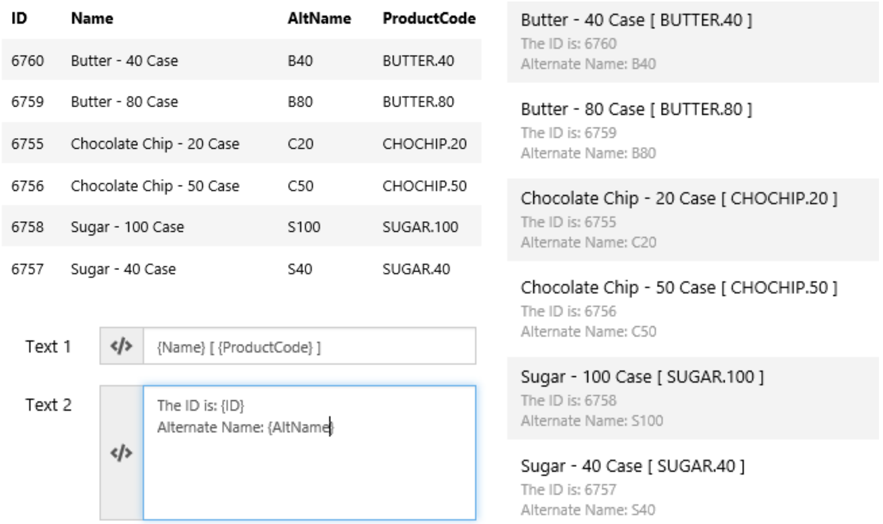
```
Ets.Values['SomeString'] to Hello World
Ets.Values['SomeLongNumber'] to 3.14159265358979
Ets.Values['SomeDateTime'] to 1/1/2019 8:00:00 AM -07:00
```



Text	My String: {SomeString}	My String: Hello World
Text	{SomeLongNumber:N2}	3.14
Text	{SomeShortNumber[N/A]:N2}	N/A
Text	{SomeDateTime:g}	1/1/2019 8:00 AM
Text	{SomeDateTime:MMM dd h:mm:ss tt}	Jan 01 8:00:00 AM

Row-Based Expressions

- Row based Parts include:
  - Most Charts
  - List Views
  - Table View
  - Etc.
- Properties for Row based Parts also allow Expressions to retrieve data for the current Row by Column Name.
- If Column Name is not found, the Expression will use the Values Dictionary next.



ID	Name	AltName	ProductCode
6760	Butter - 40 Case	B40	BUTTER.40
6759	Butter - 80 Case	B80	BUTTER.80
6755	Chocolate Chip - 20 Case	C20	CHOCHIP.20
6756	Chocolate Chip - 50 Case	C50	CHOCHIP.50
6758	Sugar - 100 Case	S100	SUGAR.100
6757	Sugar - 40 Case	S40	SUGAR.40

Text 1: `{Name} [ {ProductCode} ]`

Text 2:

```
The ID is: {ID}
Alternate Name: {AltName}
```

Butter - 40 Case [BUTTER.40]
The ID is: 6760
Alternate Name: B40

Butter - 80 Case [BUTTER.80]
The ID is: 6759
Alternate Name: B80

Chocolate Chip - 20 Case [CHOCHIP.20]
The ID is: 6755
Alternate Name: C20

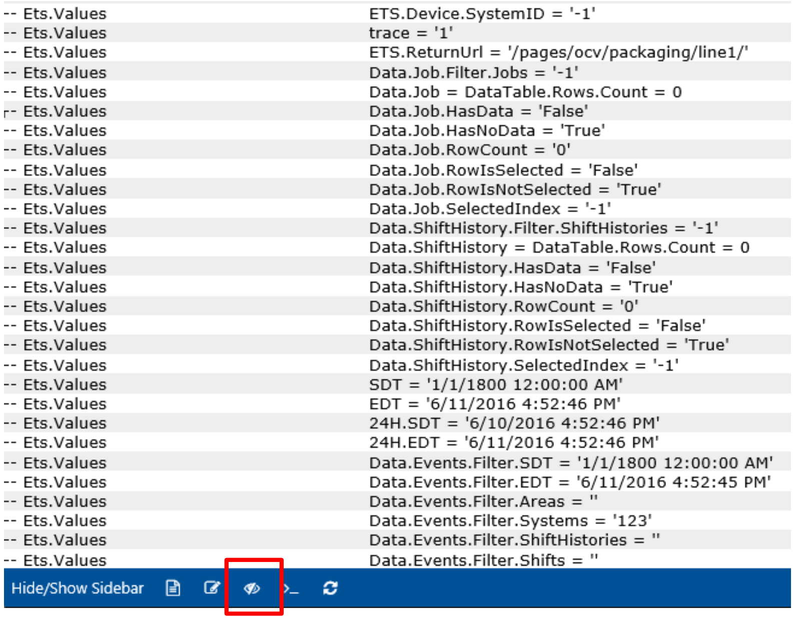
Chocolate Chip - 50 Case [CHOCHIP.50]
The ID is: 6756
Alternate Name: C50

Sugar - 100 Case [SUGAR.100]
The ID is: 6758
Alternate Name: S100

Sugar - 40 Case [SUGAR.40]
The ID is: 6757
Alternate Name: S40

Tracing

- Show/Hide the ASP.NET Trace results using the Eyeball Icon (lower left corner)
- Trace contents are displayed at the bottom of the Page
- The Trace contents contain…
    - Built-in ASP.NET Page Lifecycle Information
    - TrakSYS Page Lifecycle Information
    - Values Dictionary Contents
    - User Defined / Script Inserted Entries



```
-- Ets.Values                ETS.Device.SystemID = '-1'
-- Ets.Values                trace = '1'
-- Ets.Values                ETS.ReturnUrl = '/pages/ocv/packaging/line1/'
-- Ets.Values                Data.Job.Filter.Jobs = '-1'
-- Ets.Values                Data.Job = DataTable.Rows.Count = 0
-- Ets.Values                Data.Job.HasData = 'False'
-- Ets.Values                Data.Job.HasNoData = 'True'
-- Ets.Values                Data.Job.RowCount = '0'
-- Ets.Values                Data.Job.RowIsSelected = 'False'
-- Ets.Values                Data.Job.RowIsNotSelected = 'True'
-- Ets.Values                Data.Job.SelectedIndex = '-1'
-- Ets.Values                Data.ShiftHistory.Filter.ShiftHistories = '-1'
-- Ets.Values                Data.ShiftHistory = DataTable.Rows.Count = 0
-- Ets.Values                Data.ShiftHistory.HasData = 'False'
-- Ets.Values                Data.ShiftHistory.HasNoData = 'True'
-- Ets.Values                Data.ShiftHistory.RowCount = '0'
-- Ets.Values                Data.ShiftHistory.RowIsSelected = 'False'
-- Ets.Values                Data.ShiftHistory.RowIsNotSelected = 'True'
-- Ets.Values                Data.ShiftHistory.SelectedIndex = '-1'
-- Ets.Values                SDT = '1/1/1800 12:00:00 AM'
-- Ets.Values                EDT = '6/11/2016 4:52:46 PM'
-- Ets.Values                24H.SDT = '6/10/2016 4:52:46 PM'
-- Ets.Values                24H.EDT = '6/11/2016 4:52:46 PM'
-- Ets.Values                Data.Events.Filter.SDT = '1/1/1800 12:00:00 AM'
-- Ets.Values                Data.Events.Filter.EDT = '6/11/2016 4:52:45 PM'
-- Ets.Values                Data.Events.Filter.Areas = ''
-- Ets.Values                Data.Events.Filter.Systems = '123'
-- Ets.Values                Data.Events.Filter.ShiftHistories = ''
-- Ets.Values                Data.Events.Filter.Shifts = ''
```

Visual Page Definitions

Training Objectives

Introduce some basic responsive design concepts that used in TrakSYS Page development.

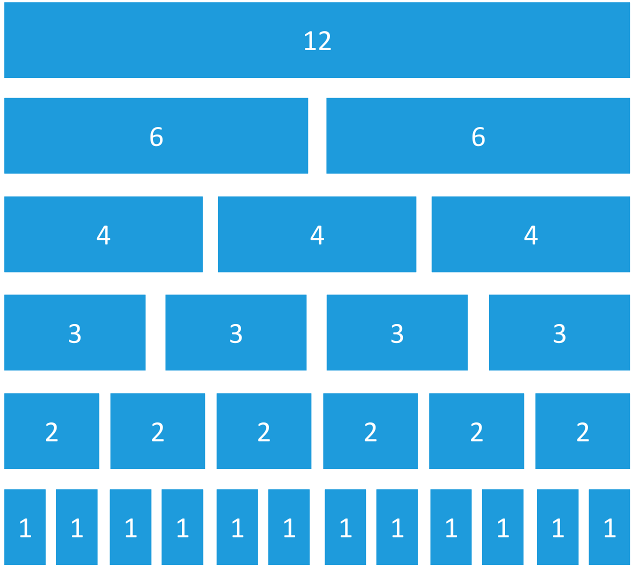
Explore the process of creating a simple Visual Page Definition layout.

Explain the Grid, Parts, Script, JavaScript and CSS editors.

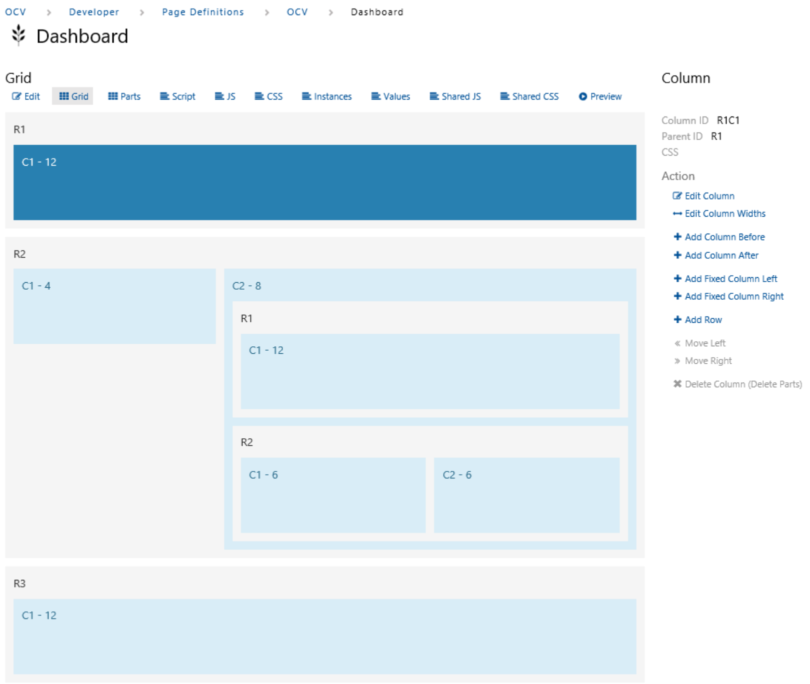
Add and configure a simple Content Part within a Visual Page layout.

Responsive Design

- Responsive Web Design (RWD) is an approach to web design aimed at crafting sites to provide an optimal viewing and interaction experience with a minimum of resizing, panning, and scrolling across a wide range of devices.
- A site using RWD adapts the layout to the viewing environment by using fluid, proportion-based grids.
- TrakSYS utilizes the Bootstrap CSS framework in for implementing screens with Responsive Design.
- It is recommended that TrakSYS developers become familiar with the Bootstrap Grid system (http://getbootstrap.com/css/#grid), which is designed to appropriately scale up to 12 columns depending on screen/viewport size.



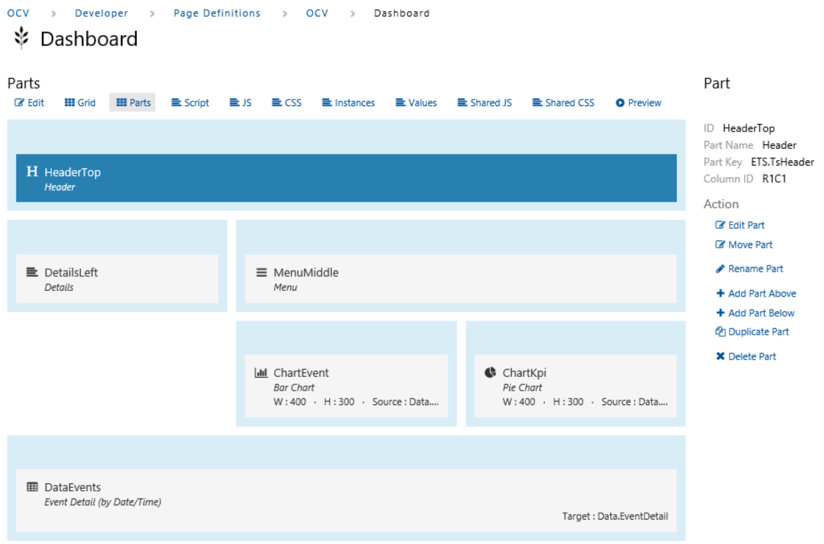
Grid Editor



Column ID R1C1
Parent ID R1
CSS

- Used to configure and layout the Visual Page's Rows and Columns
- A Row may contain one or more Columns
- A Column may contain Rows
- Content Parts are added to Columns (not Rows)
- Columns may be Fixed (pixel) or Dynamic (bootstrap 1-12) Width

Parts Editor



Dashboard

Parts

HeaderTop
Header

DetailsLeft
Details

MenuMiddle
Menu

ChartEvent
Bar Chart
W : 400 · H : 300 · Source : Data....

ChartKpi
Pie Chart
W : 400 · H : 300 · Source : Data....

DataEvents
Event Detail (by Date/Time)
Target : Data.EventDetail

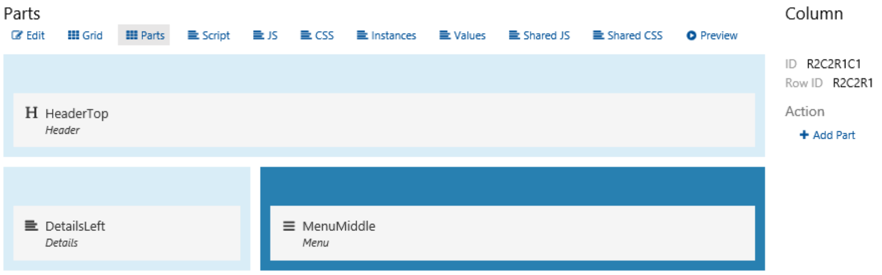
Part

ID HeaderTop
Part Name Header
Part Key ETS.TsHeader
Column ID R1C1

Action

- Simplified view of the configured Page Grid for adding and manipulating Content Parts
- Multiple Parts can be added to a Column (Stacked)
- Parts can be Duplicated and Moved
- Select Content Part Properties are displayed for Convenience (Height, Width, Data Source/Target)

Adding a Content Part



Parts

H HeaderTop
Header

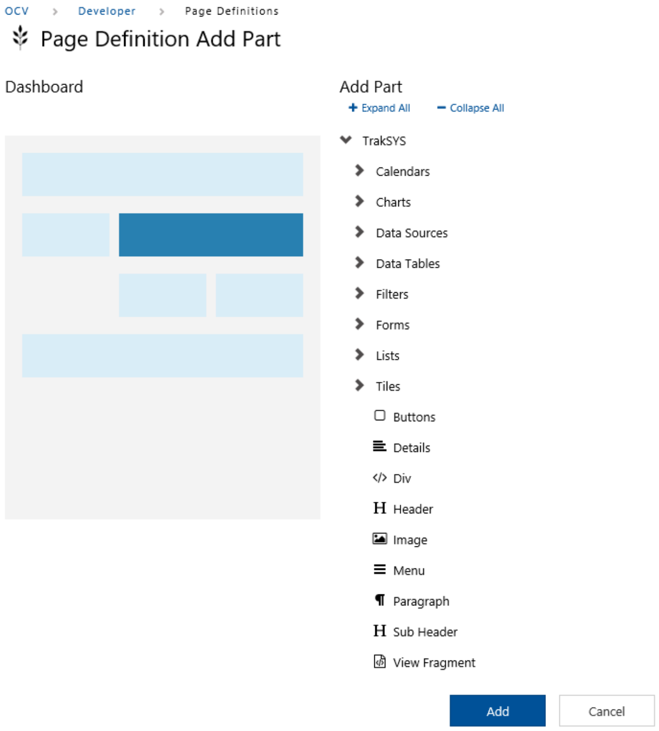
DetailsLeft
Details

MenuMiddle
Menu

Column

ID R2C2R1C1
Row ID R2C2R1

Action



Page Definition Add Part

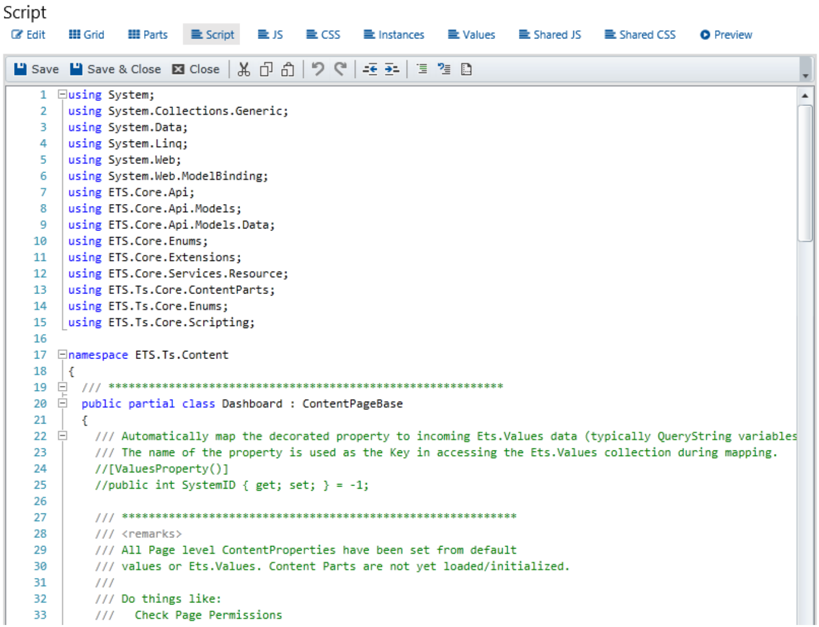
Dashboard

Add Part

- Select the Column where the Part will be Added
- Select the Add Part menu option in the right Menu
- Parts are added to the bottom of the Column

The recommended naming convention for Parts is Pascal Case, with the first word representing the type of Part (see examples above).

Script Editor

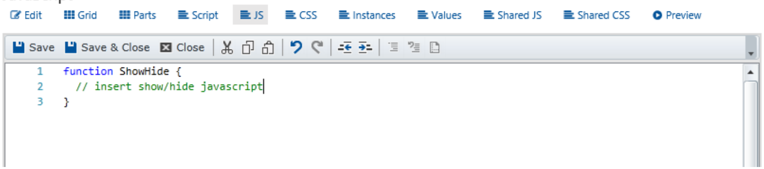


```
using System;
using System.Collections.Generic;
using System.Data;
using System.Linq;
using System.Web;
using System.Web.ModelBinding;
using ETS.Core.Api;
using ETS.Core.Api.Models;
using ETS.Core.Api.Models.Data;
using ETS.Core.Enums;
using ETS.Core.Extensions;
using ETS.Core.Services.Resource;
using ETS.Ts.Core.ContentParts;
using ETS.Ts.Core.Enums;
using ETS.Ts.Core.Scripting;

namespace ETS.Ts.Content
{
  /// ***********************************************************
  public partial class Dashboard : ContentPageBase
  {
    /// Automatically map the decorated property to incoming Ets.Values data (typically QueryString variables
    /// The name of the property is used as the Key in accessing the Ets.Values collection during mapping.
    //[ValuesProperty()]
    //public int SystemID { get; set; } = -1;

    /// ***********************************************************
    /// <remarks>
    /// All Page level ContentProperties have been set from default
    /// values or Ets.Values. Content Parts are not yet loaded/initialized.
    ///
    /// Do things like:
    ///   Check Page Permissions
```

- Allows the creation of server-side ASP.NET code that executes as the Page is Rendered
- Common uses include…
  - Setting Content Part Properties
  - Retrieving data Programmatically and setting it to Parts
  - Consuming the output of Content Parts, modifying it and setting it back to other Parts
  - API Access
  - Form POST Processing (saving and updating data and configuration)

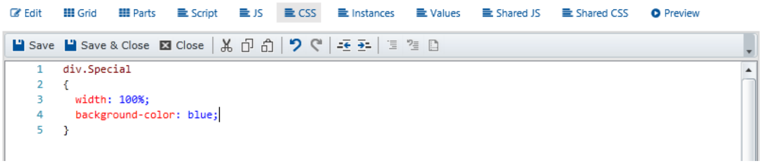
JavaScript and CSS Editors

JavaScript



```
function ShowHide {
  // insert show/hide javascript
}
```

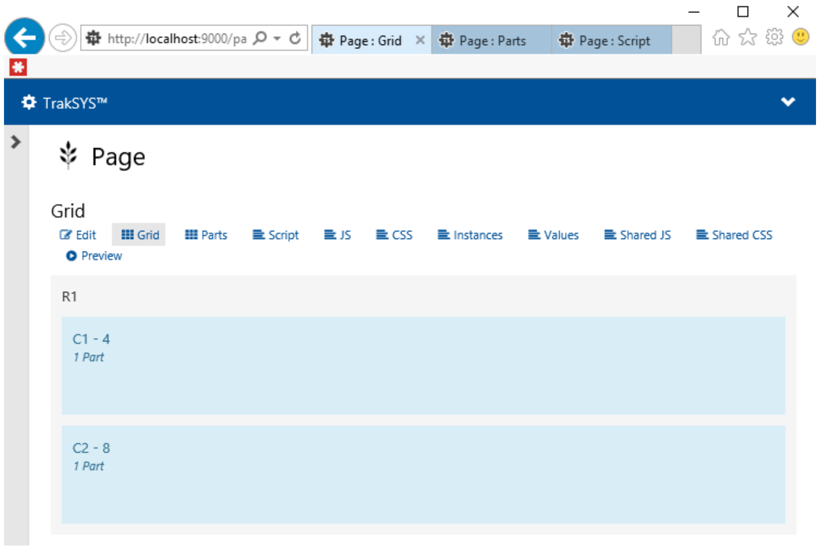
CSS



```
div.Special
{
  width: 100%;
  background-color: blue;
}
```

- Enables advanced Page Functionality and Layout Extensibility
- JavaScript
  - Client-Side Scripting
  - Hide/Show
  - Confirmation Dialogs
  - Web Service Interactions
- CSS (Cascading Style Sheets)
  - Alter Standard Component Look/Feel

Tabbed / Windowed Development



- Edit multiple Pages at once using different Tabs/Windows
- Edit multiple aspects of the same Page at once using different Tabs/Windows
- Recommend multi-monitor setup for Power Developers

The Preview action allows a new window or tab to be refreshed with the latest Page content while leaving the development editor open.

Demonstration

- Add to the Values Dictionary
  - QueryString
  - Script
- Examine the Values Dictionary using Trace
- Retrieve from the Values Dictionary
  - Property Mapping
  - Expressions
  - Script
- Create a Visual Page
- Grid Layout
- Part Management
- Show Advanced Editors
  - Script
  - JavaScript
  - CSS
- Multi-Tab/Window Development
  - Preview

Lab 6

Content Parts

Training Objectives

Present and explain the concept of Content Parts in more detail.

Introduce the different types of standard Content Parts available to create Pages and user interfaces within TrakSYS.

Explore how Content Parts consume and provide parameters/data during the Page rendering lifecycle.

Content Part Types and Initialization Order

Each part on a page has an Initialization Order integer setting. In general parts fall into one of three categories.

Filter Parts

Order = 2. Retrieve information from a User and pass on to other Parts to use.

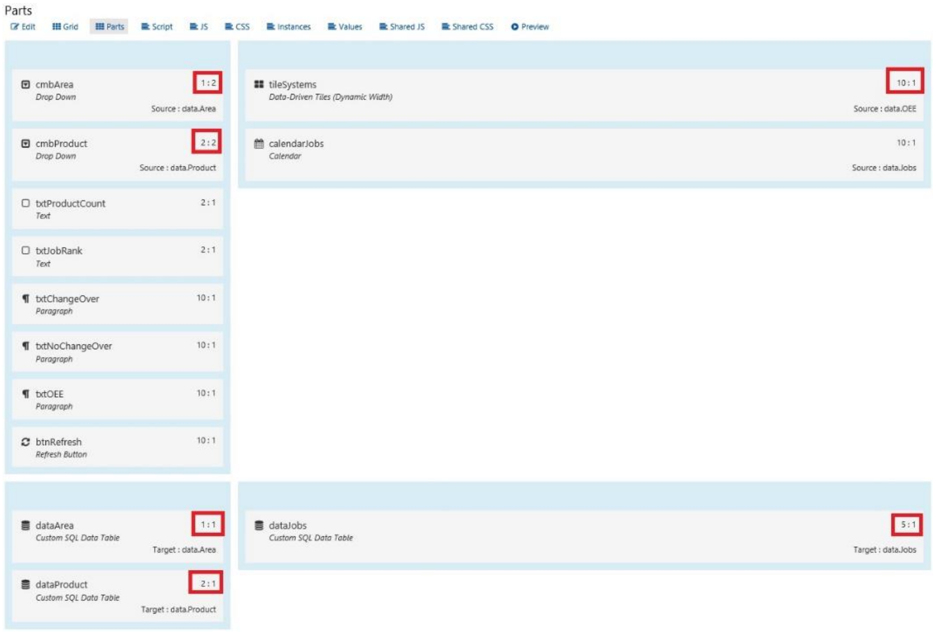
Data Provider Parts

Order = 5. Consume filter information and other values, retrieve data from the TrakSYS database or other source. Pass data on to other Parts to use.

Data Consumer Parts

Order = 10. Retrieve data from providers and format for display to the user interface.

The Initialization Order of a part can be modified for special situations.



cmbArea *Drop Down* 1 : 2 Source : data.Area

cmbProduct *Drop Down* 2 : 2 Source : data.Product

txtProductCount *Text* 2 : 1

txtJobRank *Text* 2 : 1

txtChangeOver *Paragraph* 10 : 1

txtNoChangeOver *Paragraph* 10 : 1

txtOEE *Paragraph* 10 : 1

btnRefresh *Refresh Button* 10 : 1

tileSystems *Data-Driven Tiles (Dynamic Width)* 10 : 1 Source : data.OEE

calendarJobs *Calendar* 10 : 1 Source : data.Jobs

dataArea *Custom SQL Data Table* 1 : 1 Target : data.Area

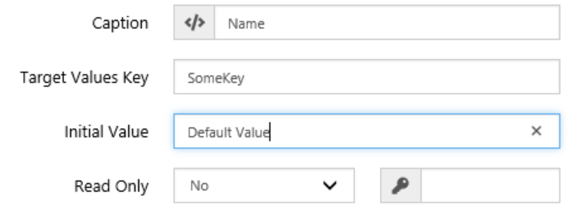
dataProduct *Custom SQL Data Table* 2 : 1 Target : data.Product

dataJobs *Custom SQL Data Table* 5 : 1 Target : data.Jobs

Common Filter Properties

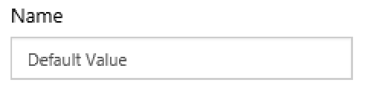
- Have a Caption to describe the Filter
- Target Values Key determines how the part Reads and Writes to the Values Dictionary
- Additional Formatting properties determine the appearance

Properties Example



Caption	Name
Target Values Key	SomeKey
Initial Value	Default Value
Read Only	No

Rendering



Name

Default Value

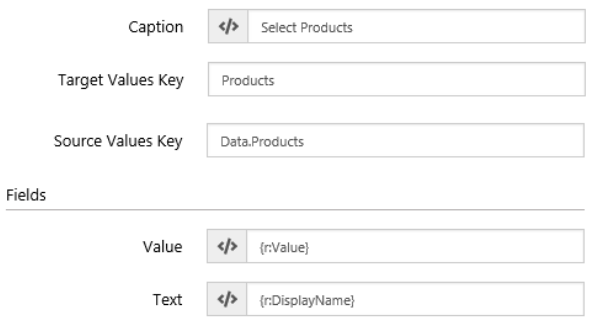
Values Dictionary

SomeKey = ‘Default Value’

Data-Driven Filters

- Have similar properties to Basic Filters
- Requires a Data table for their Source Values Key
- Use Row-Based Expression Syntax for their display properties

Properties Example

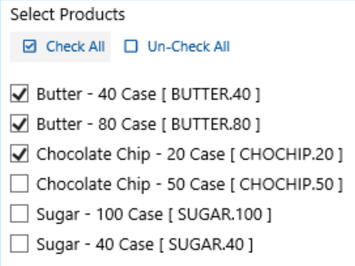


Caption	</> Select Products
Target Values Key	Products
Source Values Key	Data.Products

Fields

Value	</> {r:Value}
Text	</> {r:DisplayName}

Rendering



Select Products

☑ Butter - 40 Case [BUTTER.40]
☑ Butter - 80 Case [BUTTER.80]
☑ Chocolate Chip - 20 Case [CHOCHIP.20]
☐ Chocolate Chip - 50 Case [CHOCHIP.50]
☐ Sugar - 100 Case [SUGAR.100]
☐ Sugar - 40 Case [SUGAR.40]

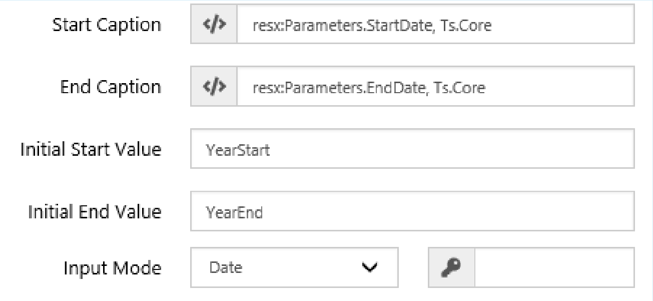
Values Dictionary

Products = ‘6760,6759,6755’

Date/Time Filter

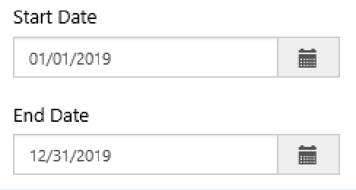
- Can be configured for Date or Date/Time Mode
- Emits selected range to the Values Dictionary
- Produces multiple Values Dictionary Keys

Properties Example



Start Caption	resx:Parameters.StartDate, Ts.Core
End Caption	resx:Parameters.EndDate, Ts.Core
Initial Start Value	YearStart
Initial End Value	YearEnd
Input Mode	Date

Rendering



Start Date	01/01/2019
End Date	12/31/2019

Values Dictionary

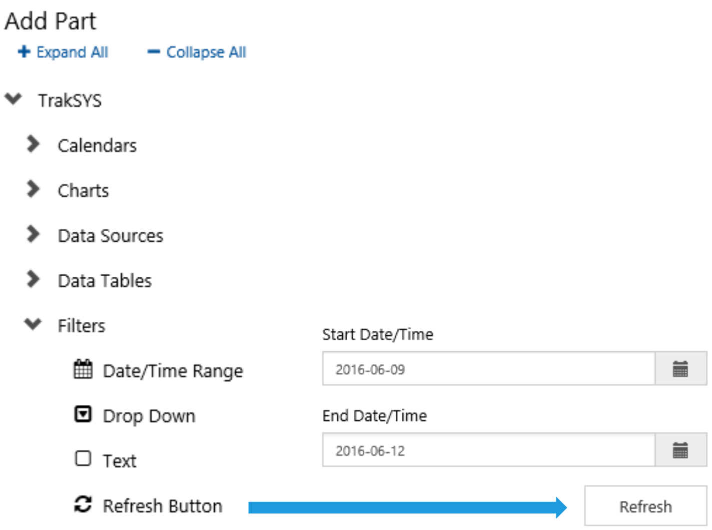
SDT = 2019-01-01 00:00:00	SD = 2019-01-01
EDT = 2020-01-01 00:00:00	ED = 2019-12-31

Refresh Button Filter

Necessary when using Filter Parts

After the User modifies Filter Values, this special Refresh button collects Filter values and re-navigates (GET) to the current Page passing the Filter values on the QueryString.

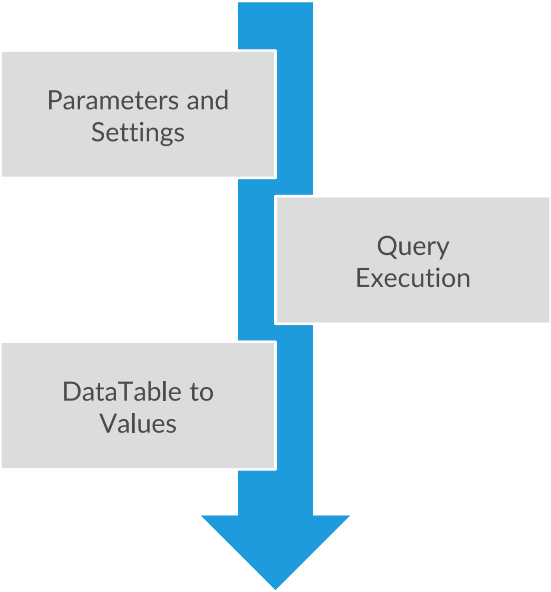
Some parts include built-in refresh buttons and other parts also have auto-postback options that feature the same functionality. While they may be useful for individual inputs, a Refresh Button is still generally recommended.



Start Date/Time: 2016-06-09

End Date/Time: 2016-06-12

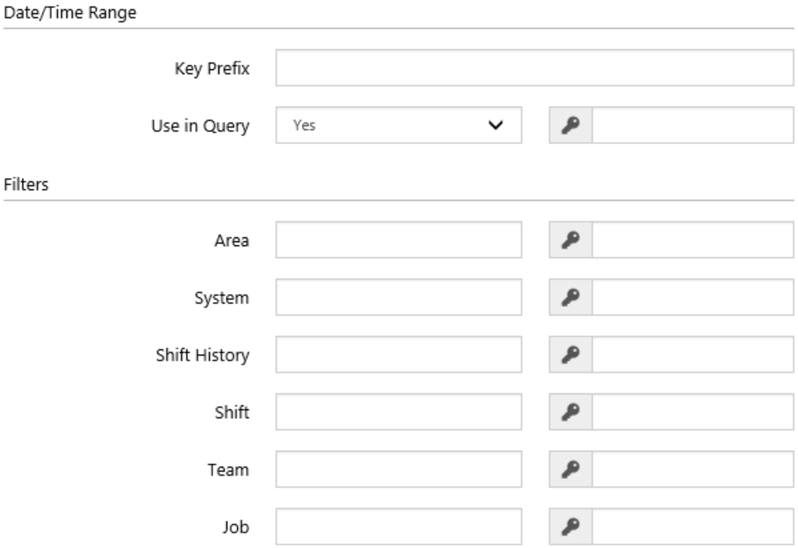
Data Table Parts



- Designed to execute pre-defined queries on the TrakSYS Database
- Return commonly used sets of data as DataTable objects deposited in the Values Dictionary
- Contain properties and settings allowing the Developer to manipulate and configure aspects of the query without writing SQL
  - Filters
  - Selected Columns
  - Sorting
  - Custom Columns

Data Table Range and Filters

- By default the Data Table Part looks for the standard SD, ED, SDT and EDT Keys to determine Date Range filter Values
- Range Key Prefix allows for customizing the expected Values Dictionary Keys
  - Range Key Prefix = Custom.
  - Keys expected = Custom.ED, Custom.SD, Custom.SDT, Custom.EDT
- Filters allow constant values or Values Dictionary Keys to be mapped to Properties



Date/Time Range

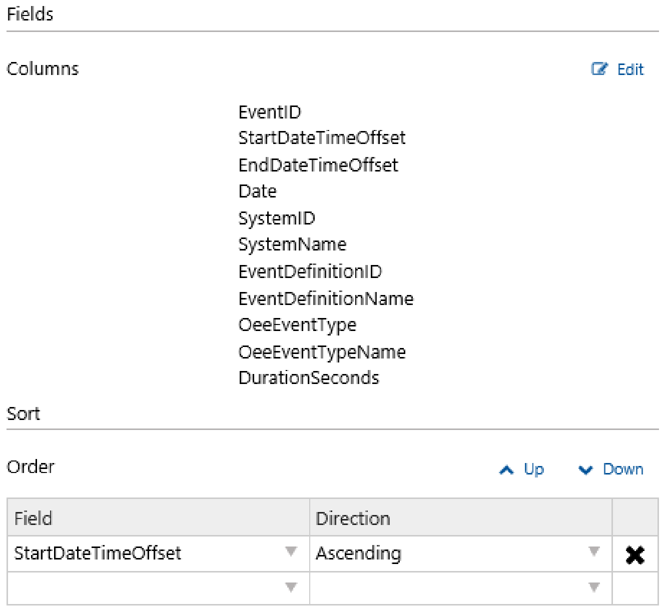
Key Prefix	
Use in Query	Yes

Filters

Area	
System	
Shift History	
Shift	
Team	
Job	

Data Table Columns and Sorting

- The Columns that are returned by the Data Table Part can be selected from a pre-configured list of Options
- Selecting only the needed fields can improve the Query Performance
- The resultant DataTable can be sorted by one ore more configurable Columns (Ascending or Descending)



Fields

Columns

EventID
StartDateTimeOffset
EndDateTimeOffset
Date
SystemID
SystemName
EventDefinitionID
EventDefinitionName
OeeEventType
OeeEventTypeName
DurationSeconds

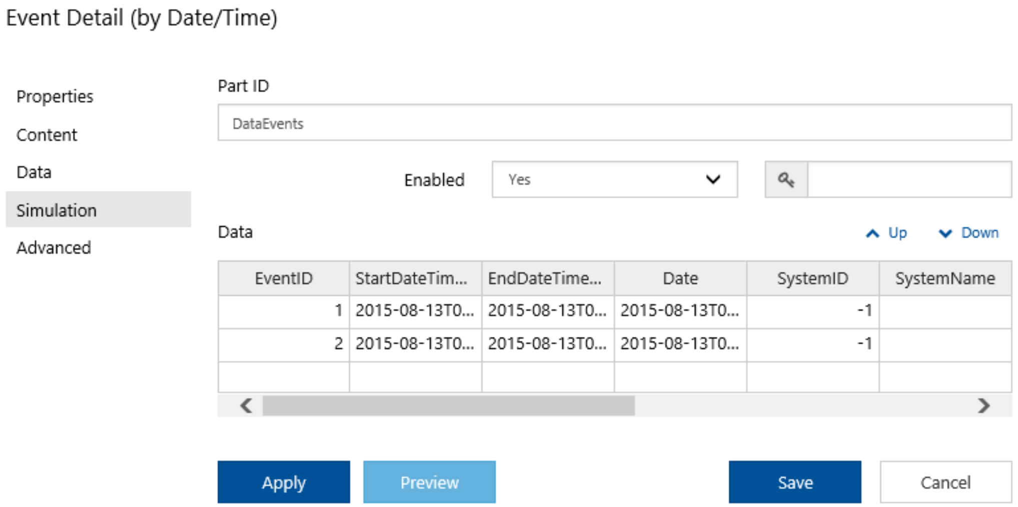
Sort

Order

Field	Direction	
StartDateTimeOffset	Ascending	✖

Data Table Simulation

- Simulation Mode allows a fake set of data to be provided to the Consumers
- Query parameters and settings are ignored in Simulation Mode
- Useful for Demonstrations and user interface Prototyping



Event Detail (by Date/Time)

Part ID: DataEvents

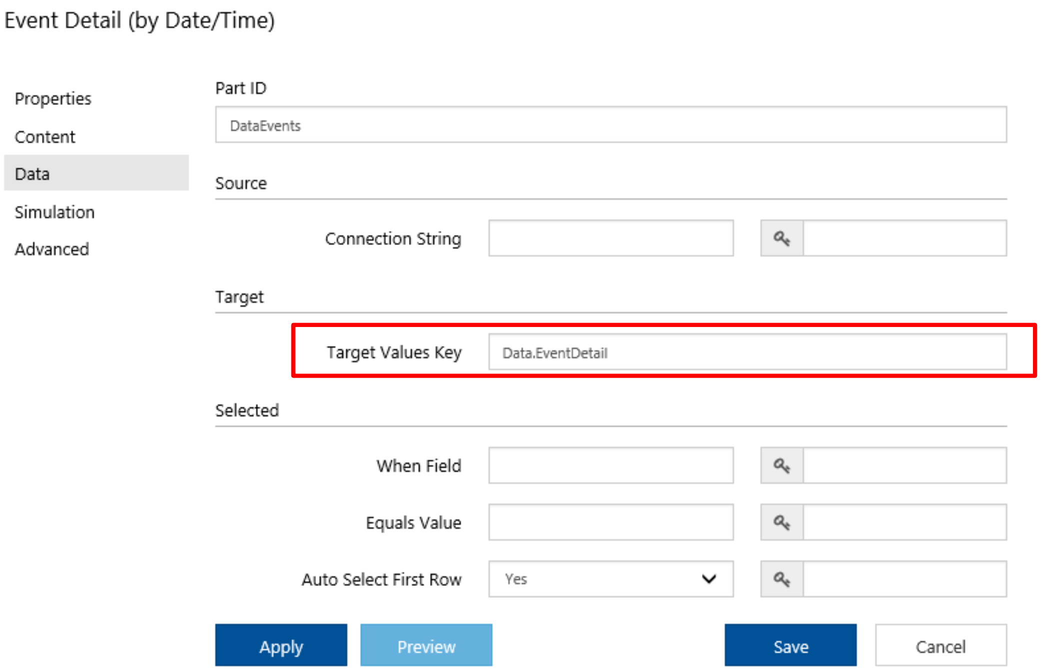
Enabled: Yes

Data

EventID	StartDateTim...	EndDateTime...	Date	SystemID	SystemName
1	2015-08-13T0...	2015-08-13T0...	2015-08-13T0...	-1	
2	2015-08-13T0...	2015-08-13T0...	2015-08-13T0...	-1	

Data Table Target

- The Target Values Key property specifies the Values Key for the resultant DataTable
- Consumer Parts must be configured with a matching Source Values Key
- Data Table Parts must have a lower Initialization Order than Consumer Parts



Event Detail (by Date/Time)

Part ID: DataEvents

Source

Connection String:

Target

Target Values Key: Data.EventDetail

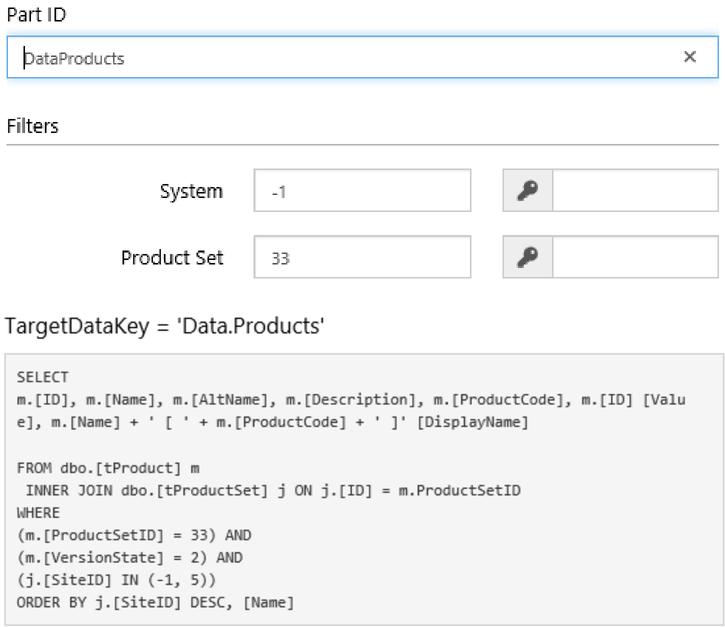
Selected

When Field:

Equals Value:

Auto Select First Row: Yes

Data Table Example



Part ID

DataProducts ×

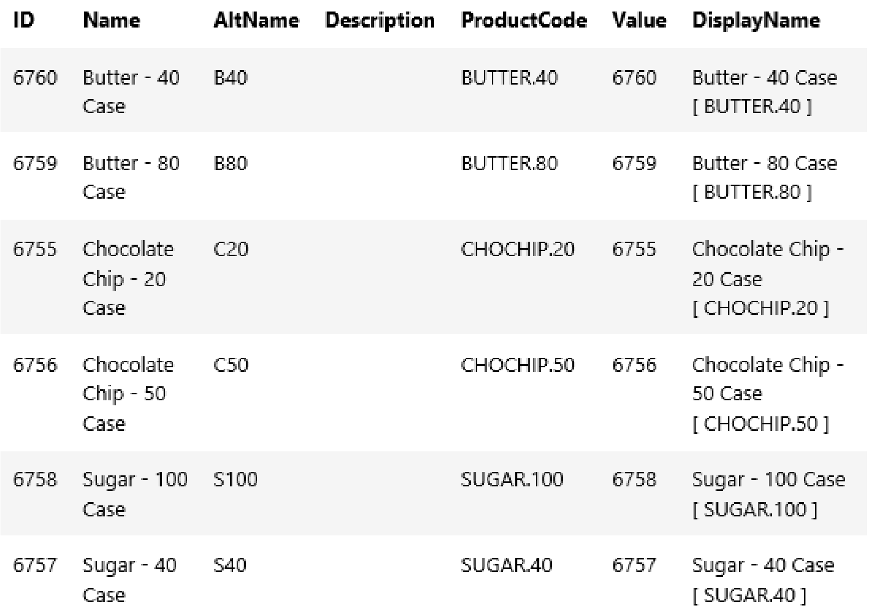
Filters

System -1

Product Set 33

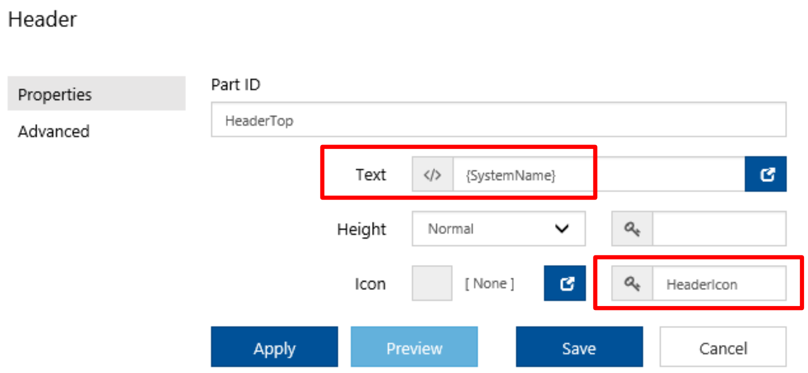
TargetDataKey = 'Data.Products'

```
SELECT
m.[ID], m.[Name], m.[AltName], m.[Description], m.[ProductCode], m.[ID] [Valu
e], m.[Name] + ' [ ' + m.[ProductCode] + ' ]' [DisplayName]

FROM dbo.[tProduct] m
 INNER JOIN dbo.[tProductSet] j ON j.[ID] = m.ProductSetID
WHERE
(m.[ProductSetID] = 33) AND
(m.[VersionState] = 2) AND
(j.[SiteID] IN (-1, 5))
ORDER BY j.[SiteID] DESC, [Name]
```



ID	Name	AltName	Description	ProductCode	Value	DisplayName
6760	Butter - 40 Case	B40		BUTTER.40	6760	Butter - 40 Case [BUTTER.40]
6759	Butter - 80 Case	B80		BUTTER.80	6759	Butter - 80 Case [BUTTER.80]
6755	Chocolate Chip - 20 Case	C20		CHOCHIP.20	6755	Chocolate Chip - 20 Case [CHOCHIP.20]
6756	Chocolate Chip - 50 Case	C50		CHOCHIP.50	6756	Chocolate Chip - 50 Case [CHOCHIP.50]
6758	Sugar - 100 Case	S100		SUGAR.100	6758	Sugar - 100 Case [SUGAR.100]
6757	Sugar - 40 Case	S40		SUGAR.40	6757	Sugar - 40 Case [SUGAR.40]

Simple Consumer Parts



Header

Properties

Advanced

Part ID: HeaderTop

Text: {SystemName}

Height: Normal

Icon: [None] HeaderIcon

- Do NOT have a configured Source Values Key property for receiving a DataTable
- Individual values are mapped to Part properties using the Values Key settings or Content Expressions
- Examples include…
  - Headers
  - Details
  - Single Tiles

Complex Consumer Parts

- Parts that expect one or more DataTable sources as inputs to be Rendered
- The Source Values Key property specifies the Values Key for the expected DataTable
- Provider Parts must be configured with a matching Source Values Key
- Consumer Parts must have a higher Initialization Order than Provider Parts

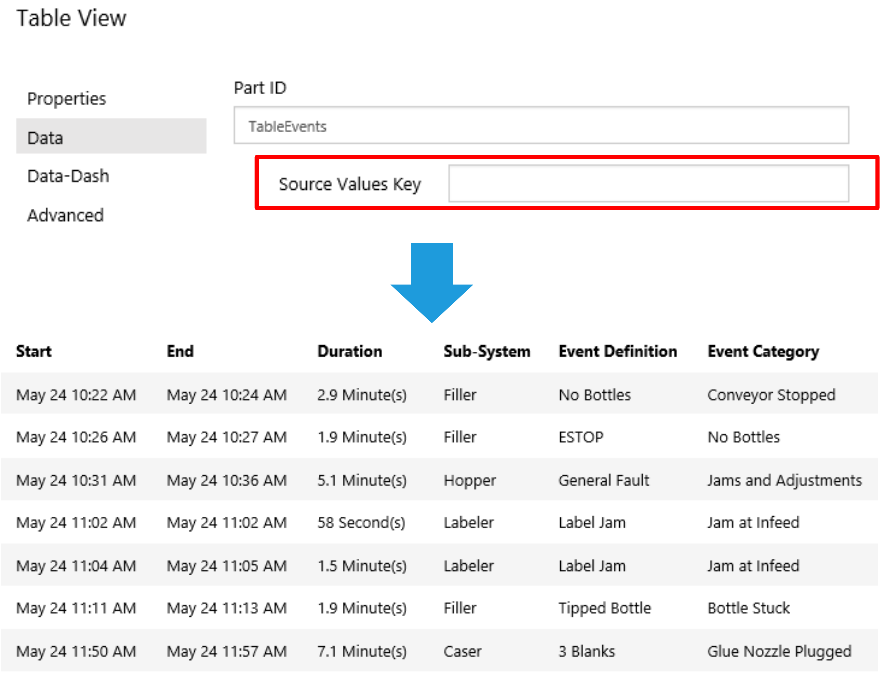


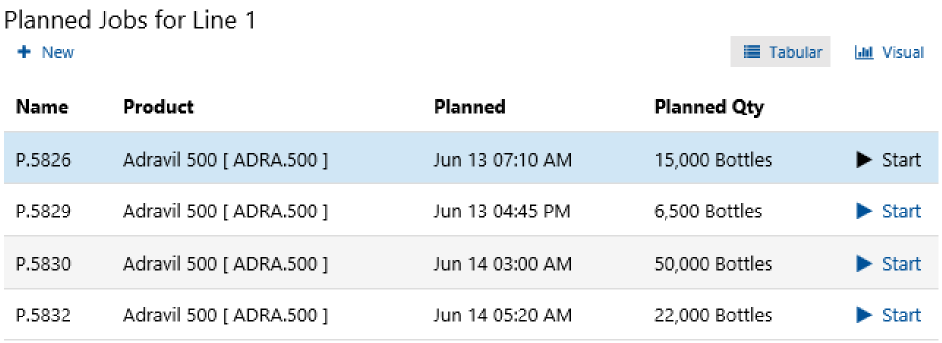
Table View

Part ID: TableEvents

Source Values Key:

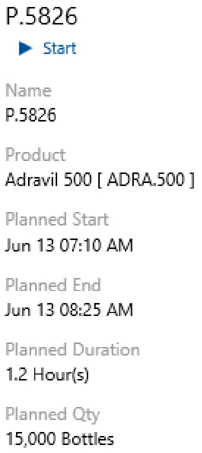
Start	End	Duration	Sub-System	Event Definition	Event Category
May 24 10:22 AM	May 24 10:24 AM	2.9 Minute(s)	Filler	No Bottles	Conveyor Stopped
May 24 10:26 AM	May 24 10:27 AM	1.9 Minute(s)	Filler	ESTOP	No Bottles
May 24 10:31 AM	May 24 10:36 AM	5.1 Minute(s)	Hopper	General Fault	Jams and Adjustments
May 24 11:02 AM	May 24 11:02 AM	58 Second(s)	Labeler	Label Jam	Jam at Infeed
May 24 11:04 AM	May 24 11:05 AM	1.5 Minute(s)	Labeler	Label Jam	Jam at Infeed
May 24 11:11 AM	May 24 11:13 AM	1.9 Minute(s)	Filler	Tipped Bottle	Bottle Stuck
May 24 11:50 AM	May 24 11:57 AM	7.1 Minute(s)	Caser	3 Blanks	Glue Nozzle Plugged

Consumer Part Examples



Planned Jobs for Line 1

Name	Product	Planned	Planned Qty	
P.5826	Adravil 500 [ADRA.500]	Jun 13 07:10 AM	15,000 Bottles	▶ Start
P.5829	Adravil 500 [ADRA.500]	Jun 13 04:45 PM	6,500 Bottles	▶ Start
P.5830	Adravil 500 [ADRA.500]	Jun 14 03:00 AM	50,000 Bottles	▶ Start
P.5832	Adravil 500 [ADRA.500]	Jun 14 05:20 AM	22,000 Bottles	▶ Start



P.5826

Name
P.5826

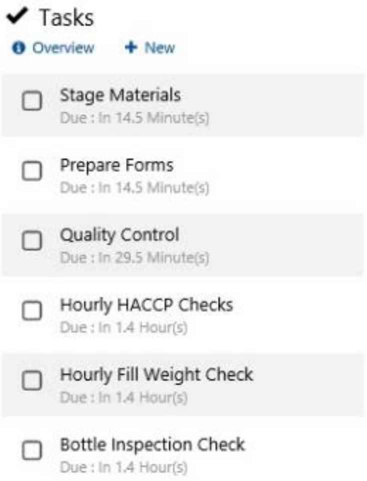
Product
Adravil 500 [ADRA.500]

Planned Start
Jun 13 07:10 AM

Planned End
Jun 13 08:25 AM

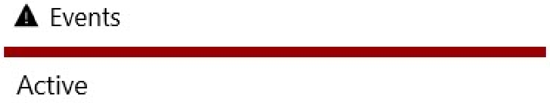
Planned Duration
1.2 Hour(s)

Planned Qty
15,000 Bottles



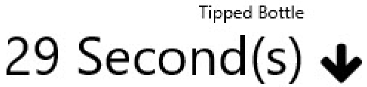
✓ Tasks

- □ Stage Materials
  Due : In 14.5 Minute(s)
- □ Prepare Forms
  Due : In 14.5 Minute(s)
- □ Quality Control
  Due : In 29.5 Minute(s)
- □ Hourly HACCP Checks
  Due : In 1.4 Hour(s)
- □ Hourly Fill Weight Check
  Due : In 1.4 Hour(s)
- □ Bottle Inspection Check
  Due : In 1.4 Hour(s)



⚠ Events

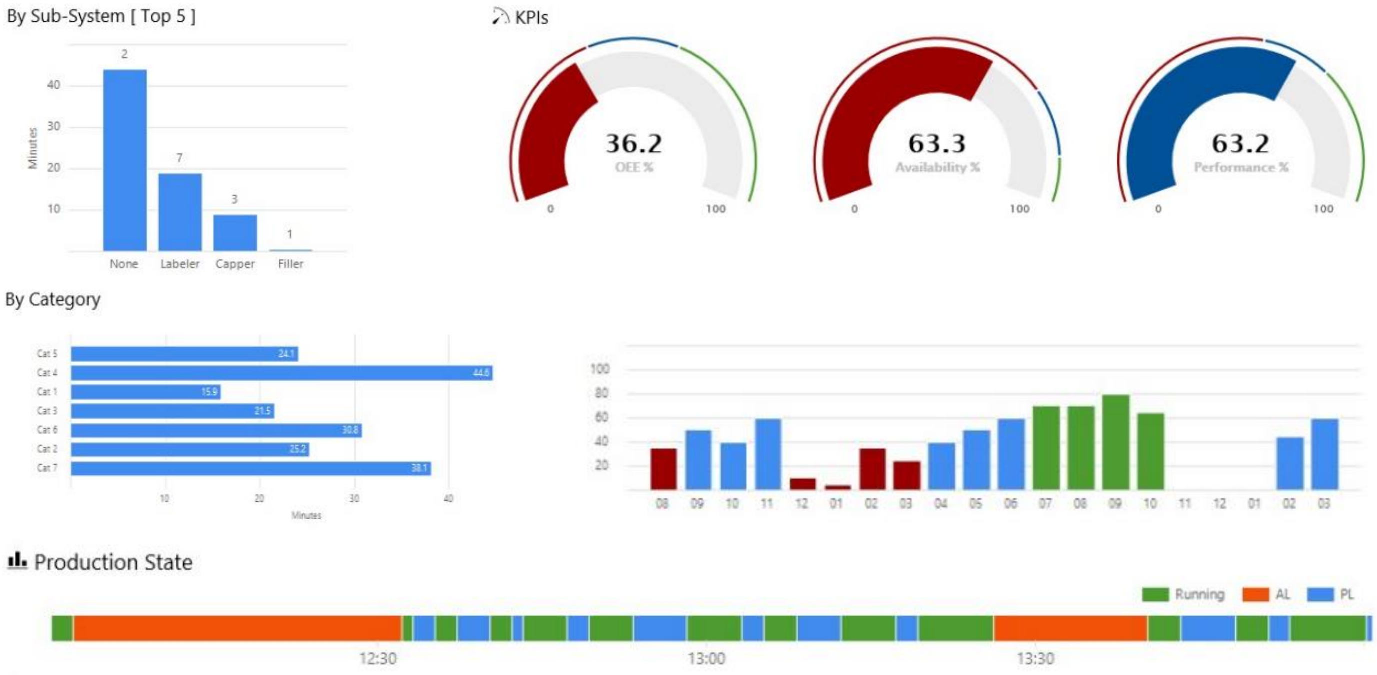
Active



Tipped Bottle

29 Second(s) ↓

Charts and Gauges



By Sub-System [Top 5]

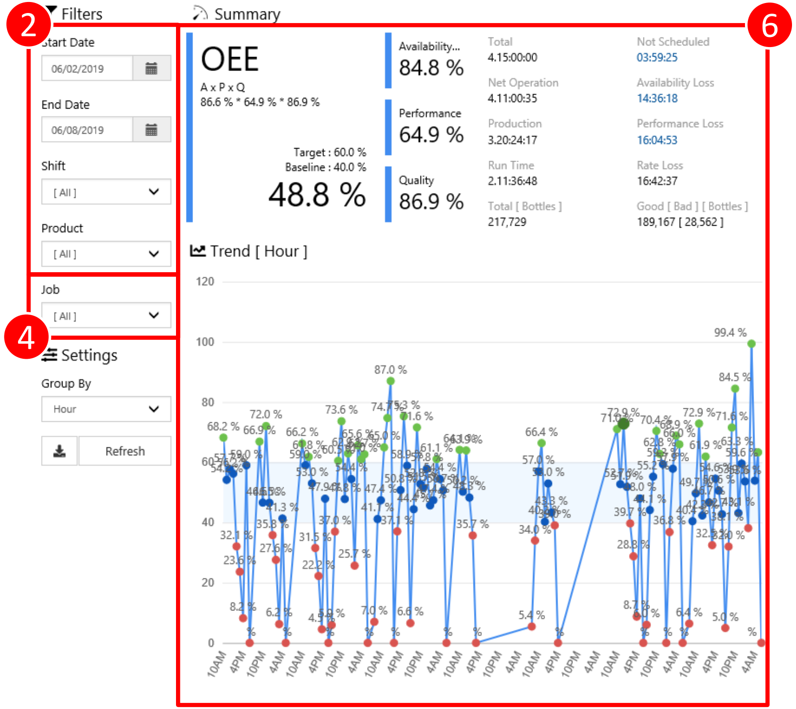
KPIs

By Category

Production State

Initialization Order

Visual Example



Filters

Start Date: 06/02/2019

End Date: 06/08/2019

Shift: [All]

Product: [All]

Job: [All]

Settings

Group By: Hour

Refresh

Summary

OEE

A x P x Q

86.6 % * 64.9 % * 86.9 %

Target : 60.0 %

Baseline : 40.0 %

48.8 %

Availability...: 84.8 %

Performance: 64.9 %

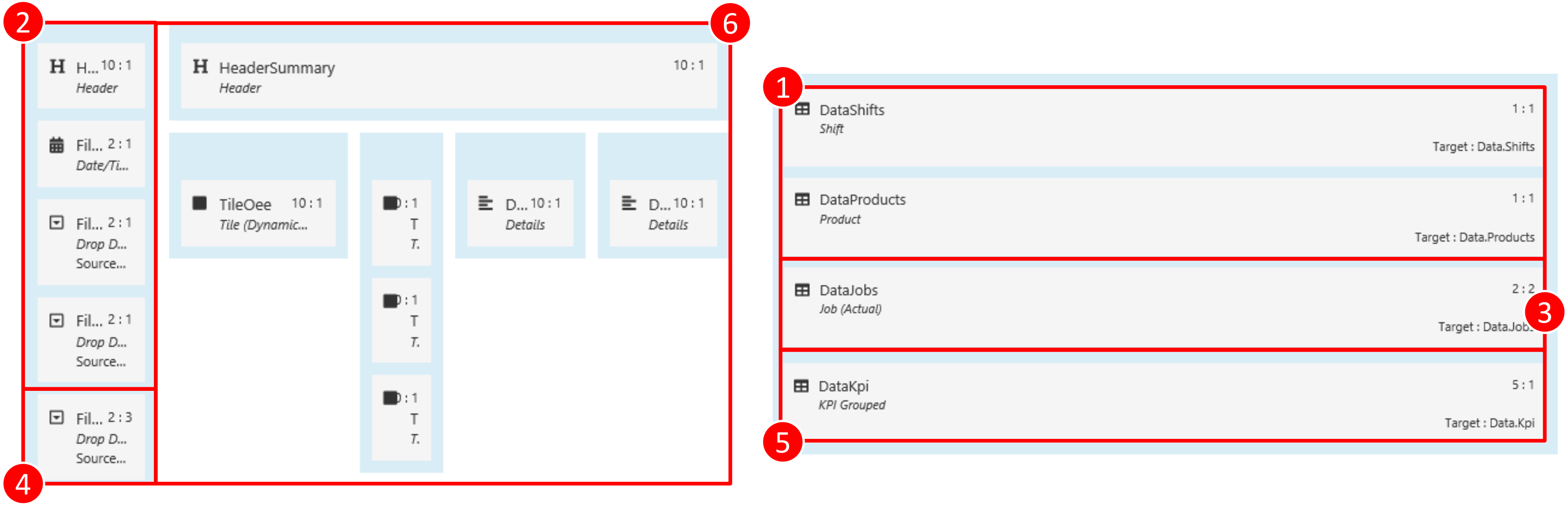
Quality: 86.9 %

Total 4.15:00:00	Not Scheduled 03:59:25
Net Operation 4.11:00:35	Availability Loss 14:36:18
Production 3.20:24:17	Performance Loss 16:04:53
Run Time 2.11:36:48	Rate Loss 16:42:37
Total [Bottles] 217,729	Good [Bad] [Bottles] 189,167 [28,562]

- Parts must initialize in the correct order for data to be passed properly
- In this example, the groups of parts have the following dependencies:
  - [4] is filtered based upon [2]
  - [6] is filtered based upon [2] and [4]
- The next slide shows the relevant parts and their Initiation Orders, including the Data Providers

Initialization Order

Parts Example



Demonstration

- Add Simple Consumer Parts to a Content Page
  - Header
  - Details
  - Tile
- Map properties of Simple Consumer Parts to incoming Values Keys
- Show Initialization Order Property
- Add Filter Parts to a Content Page
  - Date/Time
  - Refresh
- Add a Data Table Part to a Content Page
  - Configure Target Key Value
- Add a Table View Part to a Content Page
  - Configure Source Key Value

Lab 7

Content Part Features

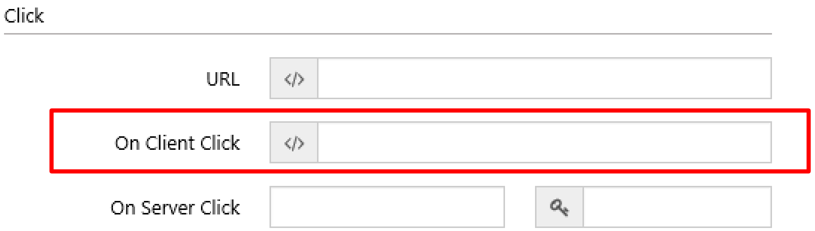
Training Objectives

Explain the Part infrastructure for receiving user clicks and navigating to new Pages or calling server side script.

Explore examples of common Content Part features allowing users to interact with and navigate around Pages

Show how Data Table and Consumer Parts can retrieve and render data sets with the concept of a selected row.

Click and Navigation Client Click



Click

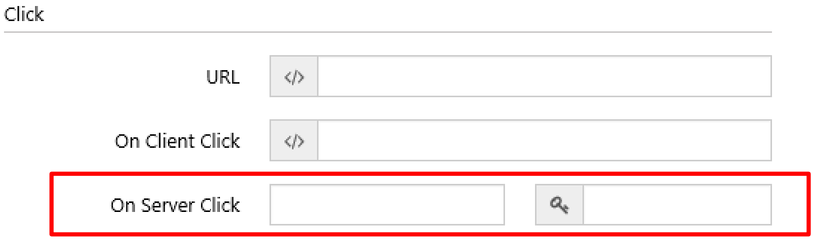
URL

On Client Click

On Server Click

- Allows JavaScript to be executed when the Part element is Clicked
- Single or Multiple Lines of JavaScript may be Executed
- May call Functions defined in the JavaScript (JS) editor for the Page
- Expressions are Allowed

Click and Navigation Server Click



Click

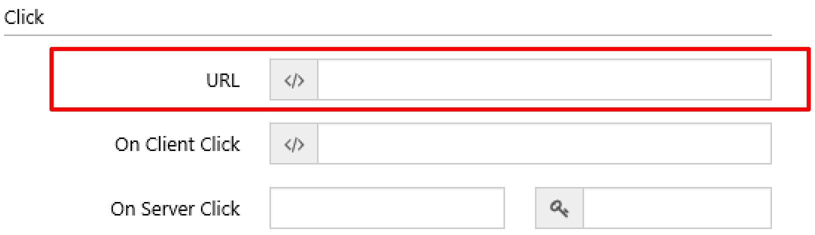
URL

On Client Click

On Server Click

- Forces a POST back and calls the specified ASP.NET server side method in the Page’s script Class
    - Example Row_Click
- For individual items (Buttons, Menus, Images, etc…) the referenced method must implement the following signature…
    - (object sender, EventArgs e)
- For rows (Tables, Lists, Charts, etc…) the referenced method must implement the following signature…
    - (object sender, RowItemEventArgs e)

Click and Navigation URL



- Renders a simple HTML anchor link (<a href=“”>) containing the URL string Specified
- Expressions are Allowed
- URL click text allows for a special syntactical element called self:
  - self: is a placeholder for the current Page’s URL
  - Any Name=Value pairs following self: are merged into the QueryString

Current Page URL

http://sn/ts/pages/ocv/test?SystemID=3&SelectedID=1582

URL

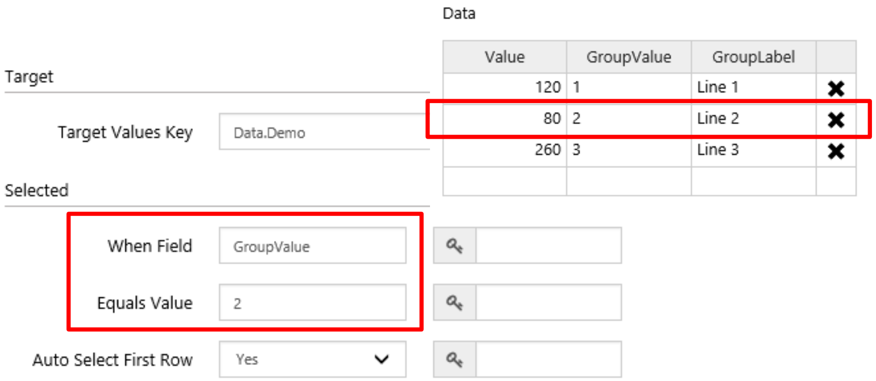
self:SystemID=4&Mode=2

Link Rendered

<a href=“http://sn/ts/pages/ocv/test?SystemID=4 &SelectedID=1582 &Mode=2”></a>

Data Table Selected Value

- Properties and settings that allow the identification of a specific row to be Selected
- When Field and Equals Value determine the Selected Row
- The columns of the Selected Row are emitted to the Values Dictionary
- Values are prefixed with the Target Values Key + .Selected.



Target

Target Values Key: Data.Demo

Selected

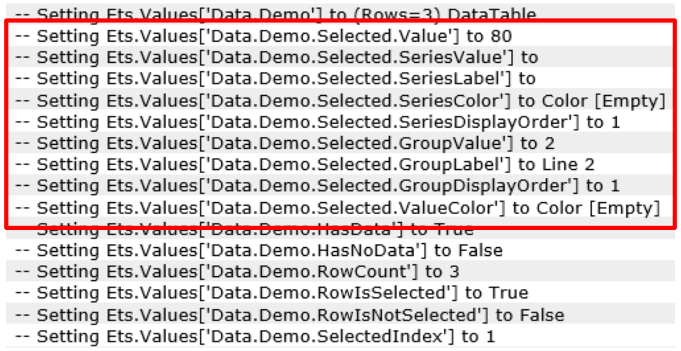
When Field: GroupValue

Equals Value: 2

Auto Select First Row: Yes

Data

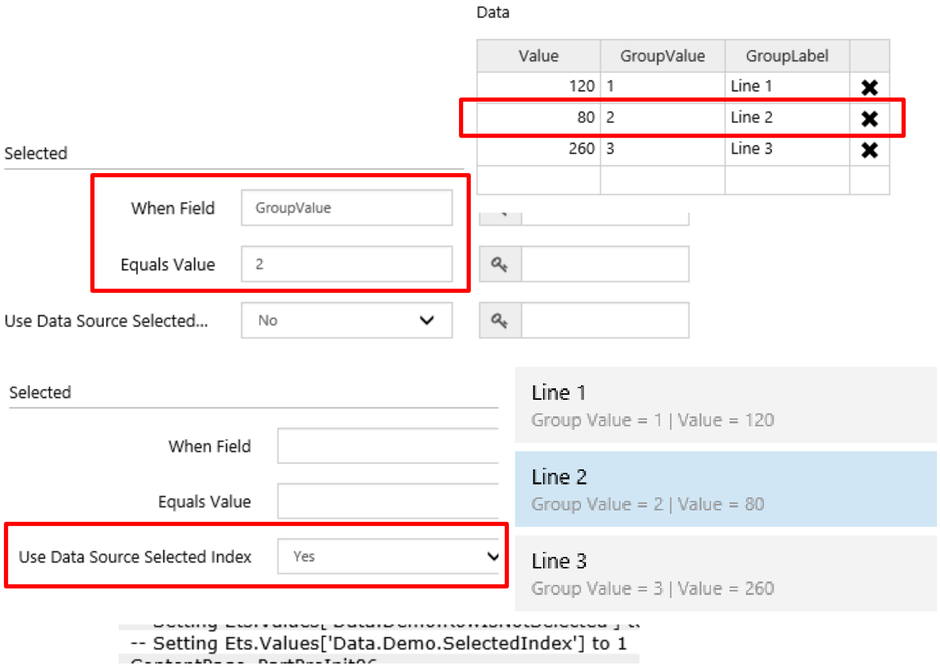
Value	GroupValue	GroupLabel	
120	1	Line 1	✖
80	2	Line 2	✖
260	3	Line 3	✖



```
-- Setting Ets.Values['Data.Demo'] to (Rows=3) DataTable
-- Setting Ets.Values['Data.Demo.Selected.Value'] to 80
-- Setting Ets.Values['Data.Demo.Selected.SeriesValue'] to
-- Setting Ets.Values['Data.Demo.Selected.SeriesLabel'] to
-- Setting Ets.Values['Data.Demo.Selected.SeriesColor'] to Color [Empty]
-- Setting Ets.Values['Data.Demo.Selected.SeriesDisplayOrder'] to 1
-- Setting Ets.Values['Data.Demo.Selected.GroupValue'] to 2
-- Setting Ets.Values['Data.Demo.Selected.GroupLabel'] to Line 2
-- Setting Ets.Values['Data.Demo.Selected.GroupDisplayOrder'] to 1
-- Setting Ets.Values['Data.Demo.Selected.ValueColor'] to Color [Empty]
-- Setting Ets.Values['Data.Demo.HasData'] to True
-- Setting Ets.Values['Data.Demo.HasNoData'] to False
-- Setting Ets.Values['Data.Demo.RowCount'] to 3
-- Setting Ets.Values['Data.Demo.RowIsSelected'] to True
-- Setting Ets.Values['Data.Demo.RowIsNotSelected'] to False
-- Setting Ets.Values['Data.Demo.SelectedIndex'] to 1
```

Table/List/Chart Selected Value

- Properties and settings that allow specific row to be rendered as Selected
- When Field and Equals Value determine the Selected Row
- Or use the Selected Index Values Key from the Data Source
- The Selected Row is highlighted in the Table / List / Chart



Data

Value	GroupValue	GroupLabel	
120	1	Line 1	✖
80	2	Line 2	✖
260	3	Line 3	✖

Selected

When Field	GroupValue
Equals Value	2
Use Data Source Selected...	No

Selected

When Field	
Equals Value	
Use Data Source Selected Index	Yes

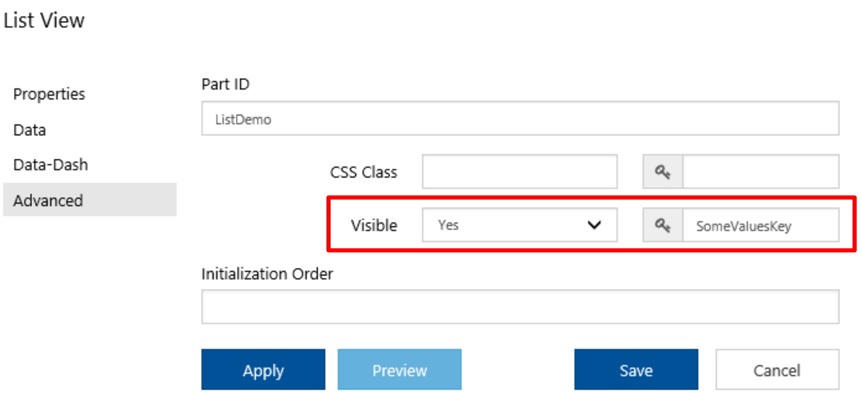
Line 1
Group Value = 1 | Value = 120

Line 2
Group Value = 2 | Value = 80

Line 3
Group Value = 3 | Value = 260

```
-- Setting Ets.Values['Data.Demo.SelectedIndex'] to 1
```

Part Visible



List View

Part ID: ListDemo

CSS Class:

Visible: Yes — SomeValuesKey

Initialization Order:

- Show or Hide any Part based on a Values Dictionary variable (Boolean)
- When a Part is NOT visible is does NOT render (it does not take up space on the Page)

Authentication and Page Permissions

Training Objectives

Explain how Users and Roles are used to identify individuals and groups within the TrakSYS application.

Explore the Authentication mechanisms and options TrakSYS uses to identify User logins.

Demonstrate how Permissions can be assigned to specific Pages to limit and control access to the application hierarchy.

Users

- Represents a Single or Group of identifiable Individuals
- User Types
  - Windows User

    Reference to a single Windows User (DOMAIN\login)
  - Windows Group

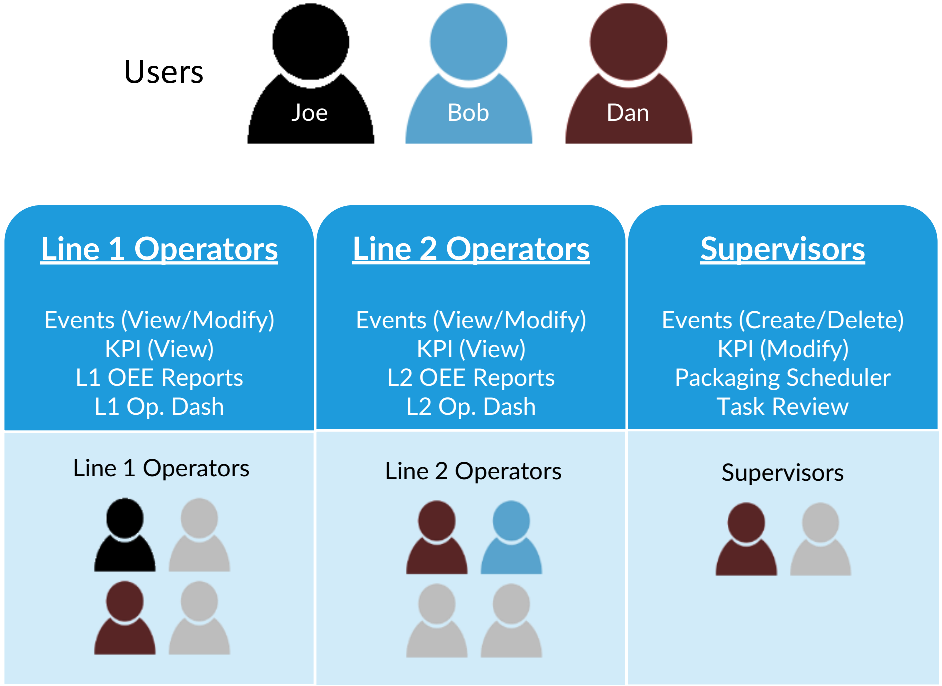
    Reference to a single Window Group (DOMAIN\group)
  - TrakSYS User

    TrakSYS managed Login and Password
  - TrakSYS Group

    One or more TrakSYS or Windows Users

Roles

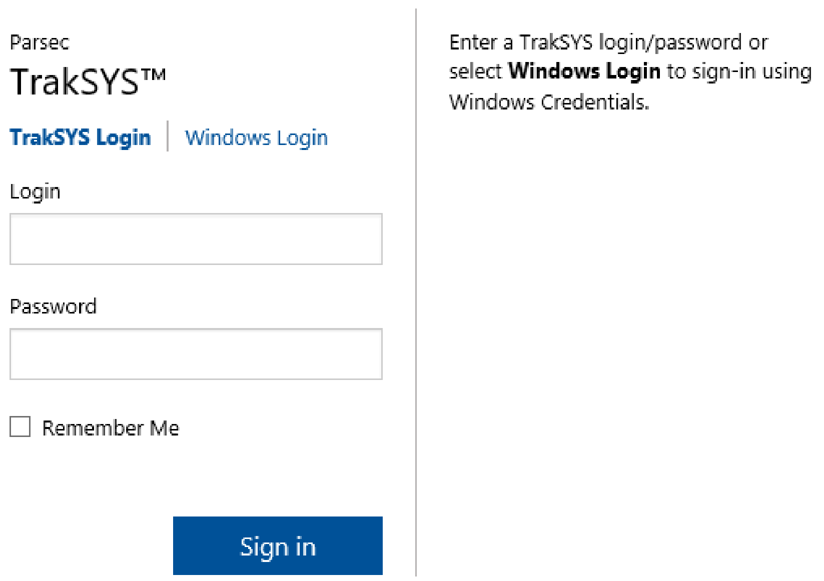
- Represents a named collection of one or more available Permissions defined within TrakSYS™
- Permissions include built-in Product Capabilities, Page Access Rights, and User-Definable Rights
- Users are associated with one or more Roles to establish allowable Page Access and functional Rights



Line 1 Operators	Line 2 Operators	Supervisors
Events (View/Modify) KPI (View) L1 OEE Reports L1 Op. Dash	Events (View/Modify) KPI (View) L2 OEE Reports L2 Op. Dash	Events (Create/Delete) KPI (Modify) Packaging Scheduler Task Review
Line 1 Operators	Line 2 Operators	Supervisors

Authentication

- Allows for integration with Windows Logins OR TrakSYS specific Login/Password Management (Simultaneously)
- TrakSYS attempts Windows Authentication first, then falls back to TrakSYS Login
- Integrated Windows Authentication is based on Browser Support
  - Internet Explorer
  - Edge
  - Chrome



Parsec

TrakSYS™

TrakSYS Login | Windows Login

Login

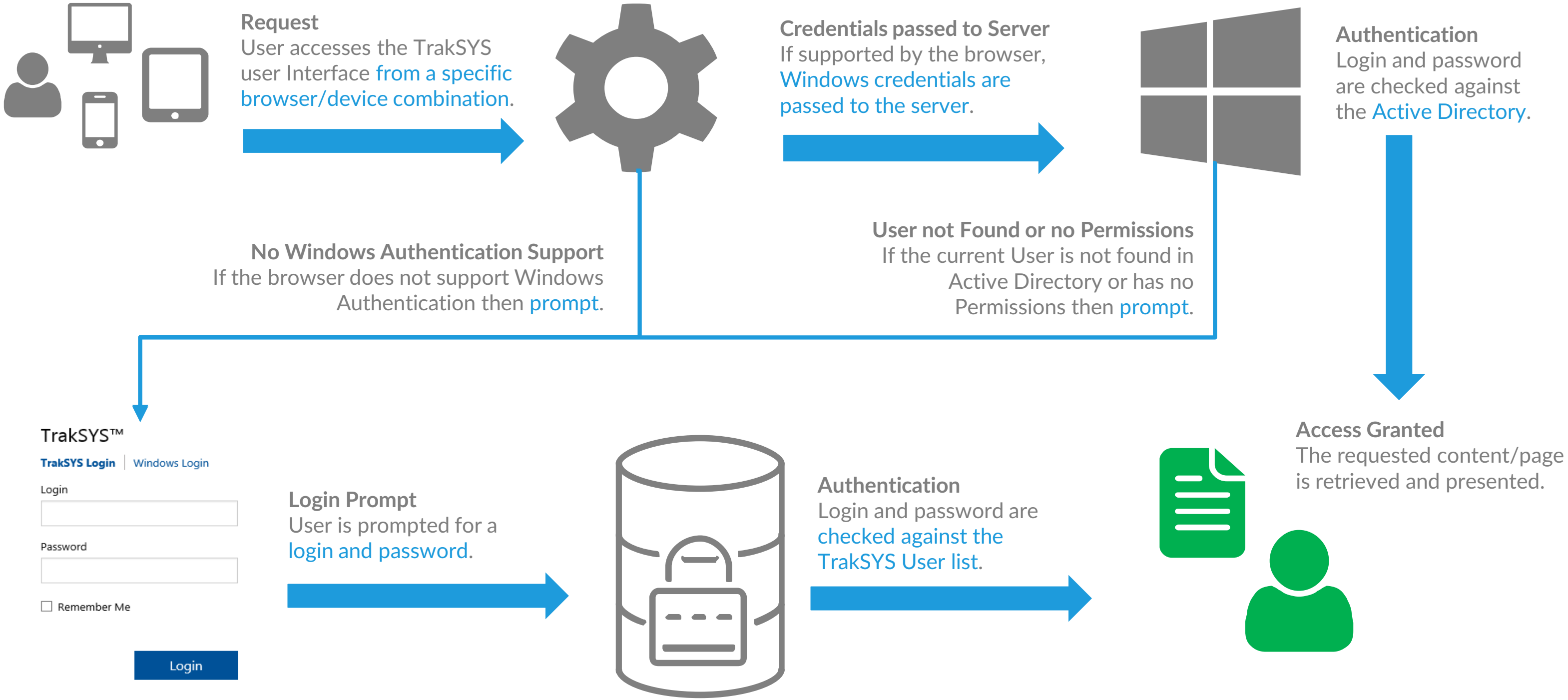
Password

□ Remember Me

Sign in

Enter a TrakSYS login/password or select **Windows Login** to sign-in using Windows Credentials.

Authentication Workflow



Request
User accesses the TrakSYS user Interface from a specific browser/device combination.

Credentials passed to Server
If supported by the browser, Windows credentials are passed to the server.

Authentication
Login and password are checked against the Active Directory.

No Windows Authentication Support
If the browser does not support Windows Authentication then prompt.

User not Found or no Permissions
If the current User is not found in Active Directory or has no Permissions then prompt.

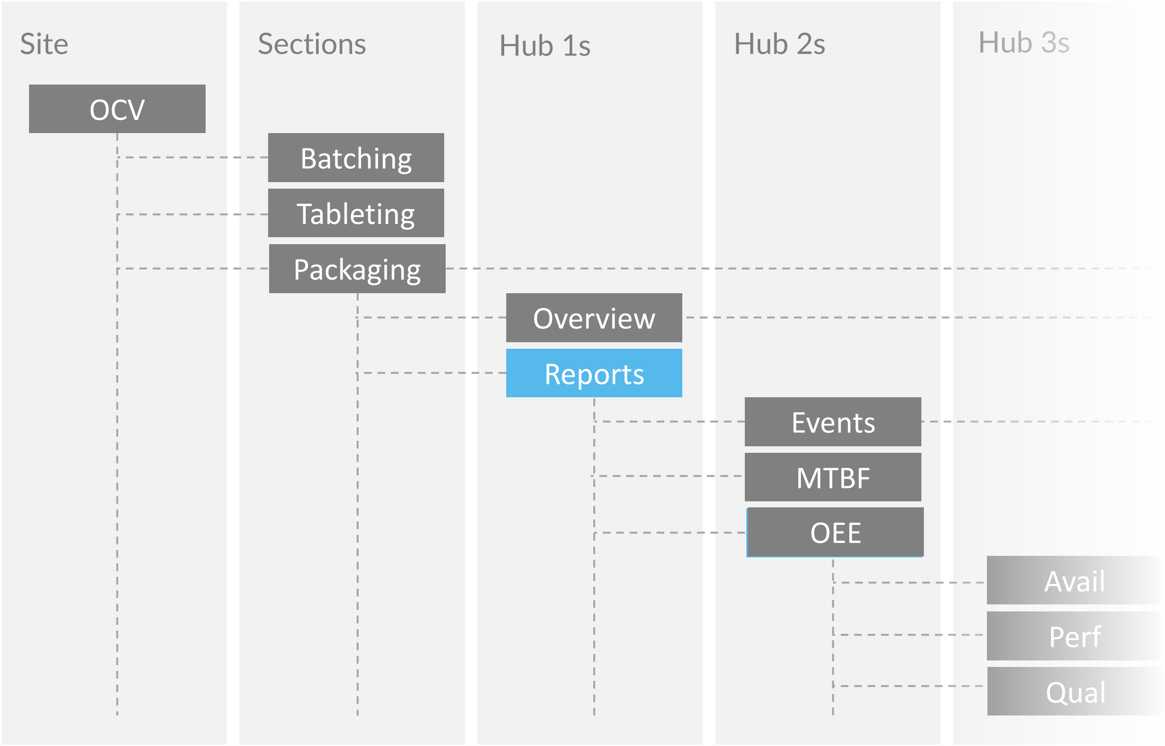
Login Prompt
User is prompted for a login and password.

Authentication
Login and password are checked against the TrakSYS User list.

Access Granted
The requested content/page is retrieved and presented.

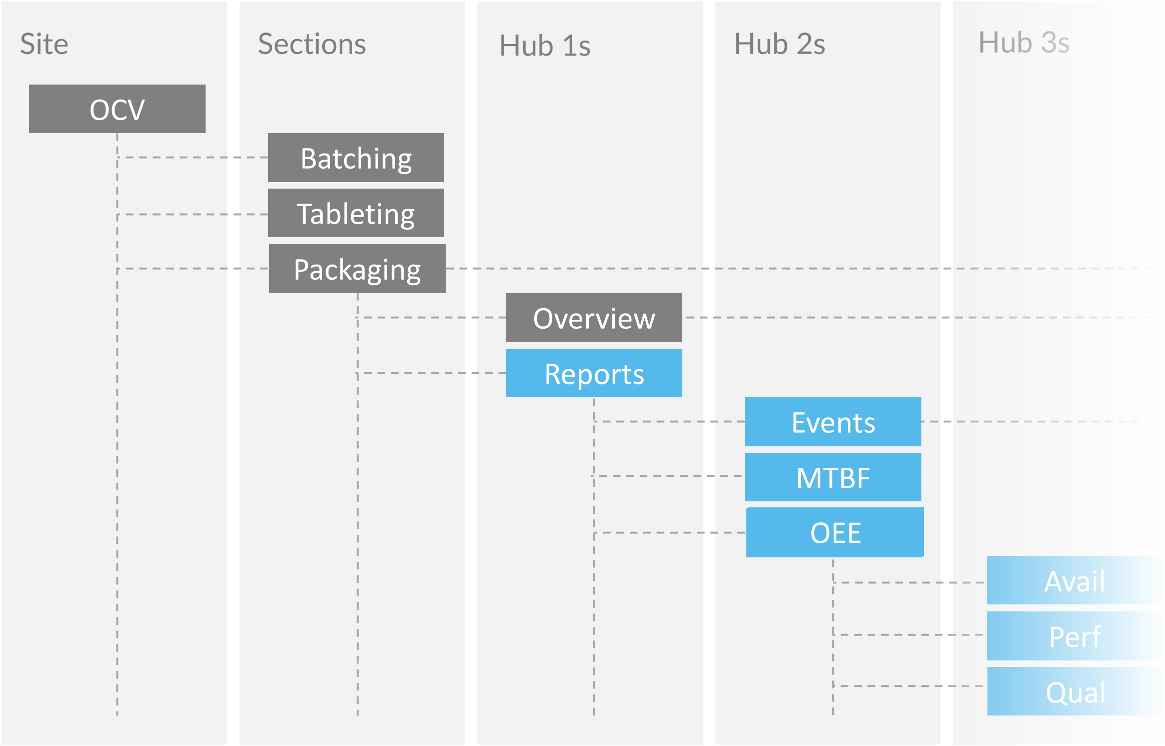
Page Permissions | Yes Single

- Grants or Denies access to Page Definitions within the TrakSYS
  - Includes built-in Sections and Hubs)
- Three Access Modes
  - Yes (single Page)
  - Yes to All (Parent and all Child Pages)
  - No Access



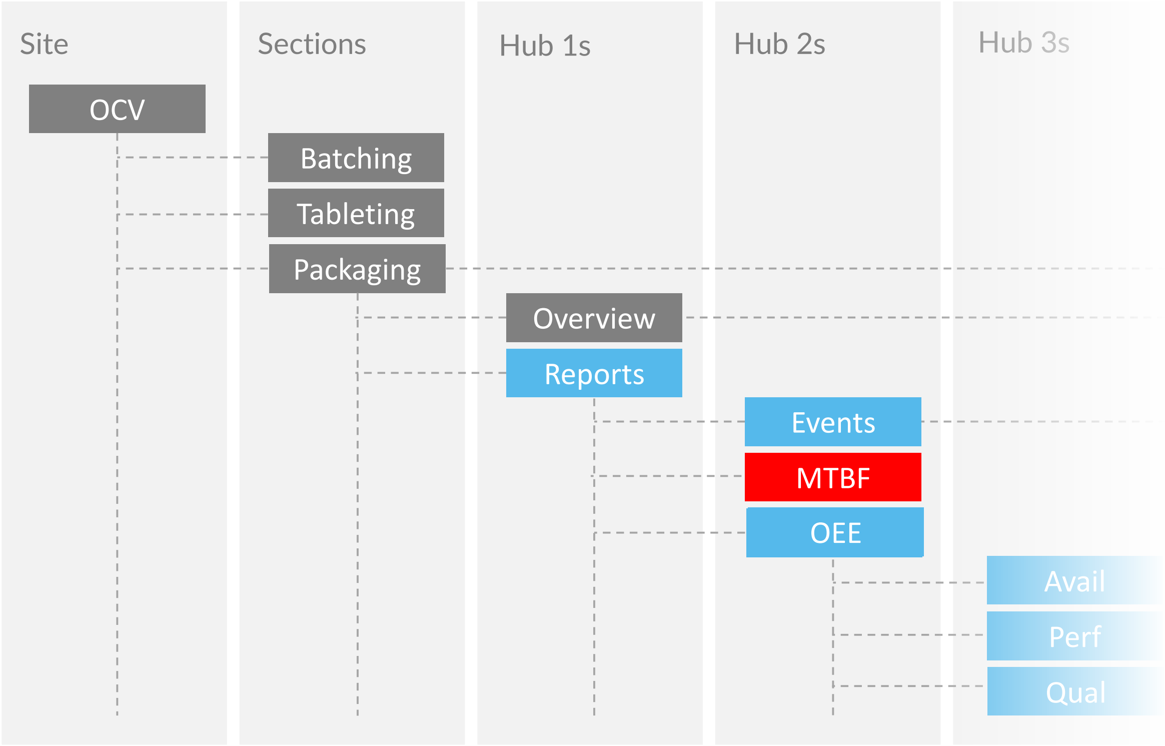
Page Permissions | Yes to All

- Grants or Denies access to Page Definitions within the TrakSYS
  - Includes built-in Sections and Hubs)
- Three Access Modes
  - Yes (single Page)
  - Yes to All (Parent and all Child Pages)
  - No Access



Page Permissions | No Access

- Grants or Denies access to Page Definitions within the TrakSYS
  - Includes built-in Sections and Hubs)
- Three Access Modes
  - Yes (single Page)
  - Yes to All (Parent and all Child Pages)
  - No Access



Demonstration

- Configure a List click using the Client action.
- Configure a List click using the URL action.
- Configure a Data Table to contain a selected row.
- Configure a List to render the selected row.

- Configure a User
- Configure a Group
- Configure a Role
  - Permissions
  - Page Permissions
- Associate a User with a Role

Lab 8