# Design Insights
# Enhanced Cue Functionality

White Paper                                          May 2014

# Contents

**Design Insights - Enhanced Cue Functionality**

White Paper

Microsoft Dynamics NAV

## Introduction

In Microsoft Dynamics NAV 2015, we have enhanced the functionality of Cues to give application developers, as well as end users, more control over the appearance of Cues. The following figure illustrates some of the new Cue features, which include colored indicators (styles), integer and decimal data types, data formatting (such as percentages), and icons (images), and more.
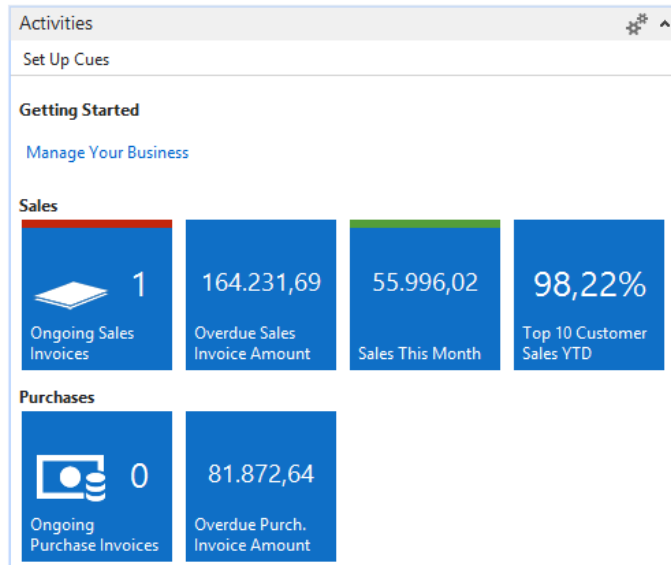


This white paper provides a high-level overview of how the enhanced Cue functionality is designed and some best practices on how to implement Cue controls.
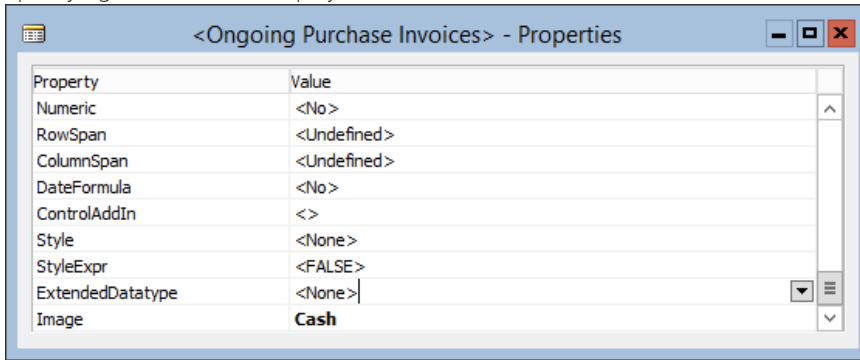
## Basics of Implementing Cue Controls

When creating a page with Cue controls in the Microsoft Dynamics NAV Development Environment, the application developer creates a Group control that has the subtype CueGroup, and then adds Field controls for the Cues under CueGroup. This is illustrated in the following figure.
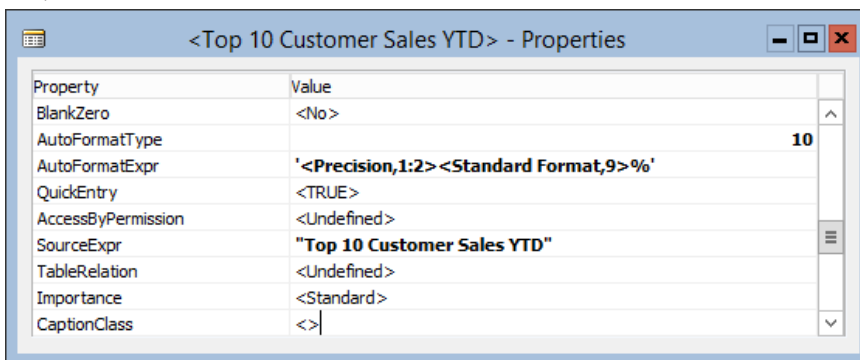
The application developer can control various aspects about how the Cue control appears, including the following:

- Specifying which icon to display on the Cue or whether to show an icon at all.



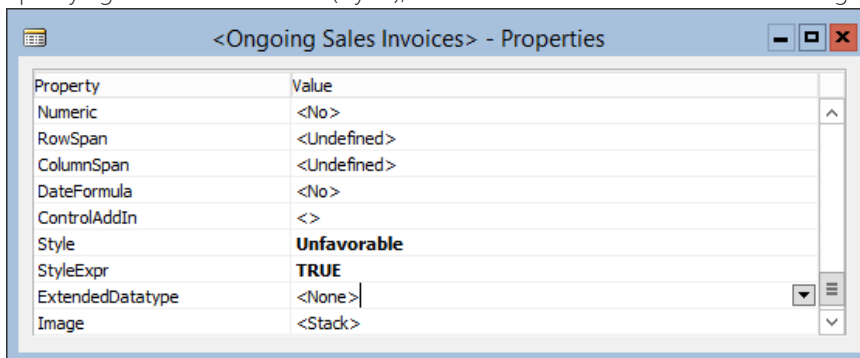| Property | Value |
|---|---|
| Numeric | <No> |
| RowSpan | <Undefined> |
| ColumnSpan | <Undefined> |
| DateFormula | <No> |
| ControlAddIn | <> |
| Style | <None> |
| StyleExpr | <FALSE> |
| ExtendedDatatype | <None> |
| Image | **Cash** |

*<Ongoing Purchase Invoices> - Properties*

This is only supported on Cue controls that are bound to integer fields.

- Specifying how to format the number string, including adding literal strings before or after such as "%"or "$" or "M".



| Property | Value |
|---|---|
| BlankZero | <No> |
| AutoFormatType | 10 |
| AutoFormatExpr | **'<Precision,1:2><Standard Format,9>%'** |
| QuickEntry | <TRUE> |
| AccessByPermission | <Undefined> |
| SourceExpr | **"Top 10 Customer Sales YTD"** |
| TableRelation | <Undefined> |
| Importance | <Standard> |
| CaptionClass | <> |

*<Top 10 Customer Sales YTD> - Properties*

- Specifying the indicator colors (styles), such as Unfavorable=red or Favorable=green.



| Property | Value |
|---|---|
| Numeric | <No> |
| RowSpan | <Undefined> |
| ColumnSpan | <Undefined> |
| DateFormula | <No> |
| ControlAddIn | <> |
| Style | **Unfavorable** |
| StyleExpr | **TRUE** |
| ExtendedDatatype | <None> |
| Image | <Stack> |

*<Ongoing Sales Invoices> - Properties*

As best practice, you should not customize the indicators this way. The reason for this is explained in the next section.

# Best Practices for Implementing Cue Controls: Enabling End User Customization of Indicators

The properties outlined in the previous section enable an application developer to control the appearance of Cue controls. However, it doesn't open up for end user customization.

With Microsoft Dynamics NAV 2015, it is possible for end users to customize the colored indicators on Cues.

To enable end user customization of indicators, the following best practices must be followed:

1. The table that is specified in the SourceTable property of the Cue page must have a name that ends with "Cue".

   For example, the SourceTable property of the "Mini Activities" page (1310) is set to "Mini Activities Cue" (1313).

2. The Field control of the Cue must be bound to a table field.

   For example, the "Ongoing Sales Invoices" Cue control on the "Mini Activities" page has the SourceExpr property set to "Ongoing Sales Invoices" which is a field in the "Mini Activities Cue" table.

3. A Field control in the CueGroup must have empty Style and StyleExpr properties.

   If the application developer specifies values for the Style and StyleExpr properties, then these properties will always be respected, and end user customization will be disabled for the control.

## How End User Customization Works Behind the Scenes

The previous sections give guidance about how to use the enhanced Cue functionality. This section provides a high-level overview about how the end user customization is designed to give you a deeper understanding, which might be helpful if you choose to customize the feature.

There are two sides to end user customization: 1) How an end user customizes the indicators, and 2) How the system picks up and applies the indicator customizations.

**How the end user customizes indicators**

An end user can customize the indicator of each Cue control by using either page 9701 Cue Setup Administration or page 9702 Cue Setup End User. Page 9701 is designed for administrators who can set up company defaults for indicators as well as indicators for specific users. Page 9702 is designed for end users who can launch the page from an action on the page that contains the Cue controls.

Whichever page is used, the indicator customizations will be saved in the table 9701 Cue Setup as illustrated in the following figure.



| User Name | Table ID | Table Name | Cue ID | Cue Name | Low Range Style | Threshold 1 | Middle Range Style | Threshold 2 | High Range Style |
|---|---|---|---|---|---|---|---|---|---|
| | 1313 | Mini Activities Cue | 4 | Ongoing Sales Invoices | *Unfavorable* | 2,00 | Ambiguous | 10,00 | *Unfavorable* |
| EUROPE\CH... | 1313 | Mini Activities Cue | 6 | Sales This Month | None | 0,00 | None | 55.000,00 | Favorable |

The Cue Setup table contains customizations that are stored according to three values: User Name, Table ID, and Field No. In addition, the customizations contain the styles and thresholds that determine the indicator color. The User Name field can be empty, in which case the customization serves as a company default. The Table ID identifies the SourceTable of the page that contains the Cue controls, and the Cue ID identifies the table field that the Cue control is bound to.

**How the system applies the indicator customizations**

At runtime, when the Microsoft Dynamics NAV Server processes a Cue control, it will call the GetCueStyle trigger in code unit 1, and pass the Table ID and Field No as well as the current value of the field, that is, the value that is shown on the Cue.

The GetCueStyle trigger looks up the TableID/FieldNo in the Cue Setup table and finds rows containing a company default (User Name = empty) and user-specific customization (User Name = current user). If it finds a row, it takes the current value and compares it to the two thresholds in order to determine which style should be applied.

When the GetCueStyle returns e.g. "Favorable", the NAV server populates a hidden page variable with that value, and the NAV clients use this page variable when rendering the cue control.

**Upgrade Considerations**

Microsoft Dynamics NAV 2015 and the demonstration data will work out of the box and there is even some end user customizations already set up, as is the case on the Small Business Role Center.

However, if you do a "technical upgrade" to Microsoft Dynamics NAV 2015, there are a few issues that you should be aware of. A technical upgrade involves mounting a Microsoft Dynamics NAV 2013 R2 or earlier application to a Microsoft Dynamics NAV 2015 Server.

When you perform a technical upgrade, you should address the following issues if you want to implement the end user customization of Cues:

1. When a page that contains Cues is run, the Microsoft Dynamics NAV 2015 Server will try to call the GetCueStyle function (16) on code unit 1. If the function isn't present, then all end user customization will be inactive. To enable the customization, you need to add the GetCueStyle function to codeunit 1 and assign it ID 16, and then add the code that is shown in the following figure.

```
GetCueStyle(TableId : Integer;FieldNo : Integer;CueValue : Decimal) : Text
EXIT(CueSetup.GetCustomizedCueStyle(TableId,FieldNo,CueValue));
```

2. All end users must have indirect permissions to the data in table 9701 Cue Setup. They must have indirect permissions for Read/Insert/Modify/Delete operations. To achieve this, you can add table 9701 Cue Setup to the FOUNDATION permission set as shown.

| | Edit - Permissions - FOUNDATION | | | | | |
|---|---|---|---|---|---|---|
| ▼  HOME  ACTIONS | | | | | | |
| Permissions ▾ | | | | | | |
| Object Type | Object ID | Object Name | Read Permission | Insert Permission | Modify Permission | Delete Permission | Execute Permission | Security Filter |
| Table Data ⌄ | 9701 | Cue Setup | Indirect | Indirect | Indirect | Indirect | | |

**Reflecting on the best practices**

A previous section described certain best practices that need to be followed in order for end user customization to be fully functional. Given the new insight into how the end user customization feature is actually implemented, it should be clear why the best practices exist:

1. Cue controls must be bound to table fields, because end user customizations are stored with a reference to a table field.
2. Source tables must have names that end with "Cue" because the administrator Cue Setup page only shows those tables.
3. If a Cue control has a custom Style property or StyleExpr property setting, then the developer has decided on the logic that controls the indicator of the cue control, and end users will not be able override the settings.

We hope you find this white paper useful.

The Dynamics NAV team.

**Design Insights - Enhanced Cue Functionality**
White Paper

Microsoft Dynamics NAV

Microsoft Dynamics is a line of integrated, adaptable business management solutions that enables you and your people to make business decisions with greater confidence. Microsoft Dynamics works like and with familiar Microsoft software, automating and streamlining financial, customer relationship, and supply chain processes in a way that helps you drive business success.

United States and Canada toll free: (888) 477-7989 Worldwide: (1) (701) 281-6500 www.microsoft.com/dynamics

**Design Insights - Enhanced Cue Functionality**
White Paper

Microsoft Dynamics NAV