

# OnBase<sup>®</sup>

*a Hyland Software solution*

## **Unity Forms**

# COPYRIGHT

Information in this document is subject to change without notice. The OnBase® Information Management System software described in this document is furnished only under a separate license agreement and may be used or copied only according to the terms of such agreement. It is against the law to copy the software except as specifically allowed in the license agreement, or without the expressed written consent of Hyland Software, Inc. If Hyland Software, Inc. and you have entered into a nondisclosure agreement, then this document or accompanying materials provided by Hyland Software, Inc. contains certain information which is confidential information of Hyland Software, Inc. and which may be used or copied only according to the terms of such nondisclosure agreement. All data, names, and formats used in this document's examples are fictitious unless noted otherwise. Complying with all applicable copyright laws is the responsibility of the user. Without limiting the rights under copyright, no part of this document may be reproduced, stored in or introduced into a retrieval system, or transmitted in any form or by any means (electronic, mechanical, photocopying, recording, or otherwise), or for any purpose, without the express written permission of Hyland Software, Inc.

©2011 Hyland Software, Inc. All rights reserved.

Depending on the modules licensed, The OnBase® Information Management System software may contain portions of: Imaging technology, Copyright © Snowbound Software Corporation; CD-R technology, Copyright © Sonic Solutions; CD-R technology, Copyright © Rimage Corporation; OCR technology, Copyright © Nuance Corporation; Mail interface technology, Copyright © Intuitive Data Solutions; Electronic signature technology, Copyright © Silanis Technology, Inc.; Full text search technology, Office core assembly, ASP.NET extensions, application blocks, smart client architecture, Object Builder, and WPF controls, Copyright © Microsoft Corporation; Full Text Indexing technology, Copyright © Verity, Inc.; SYBASE Adaptive Server Anywhere Desktop Runtime, Copyright © SYBASE, Inc., portions Copyright © Rational Systems, Inc.; ISIS technology, Copyright © EMC Corporation; JLex technology, Copyright © 1996-2003 by Elliot Joel Berk and C. Scott Ananian; A2iA CheckReader, Copyright © A2iA; Terminal emulation technology, Copyright © Attachmate; User interface controls, Copyright © Infragistics; Terminal emulation technology, Copyright © NetManage; CAD document technology, Copyright © Open Text Corporation; ISIS scanning interface, Copyright © Pegasus Imaging Corporation; CD/DVD burner technology, Copyright © Prassi Software Incorporated; Code obfuscation technology, Copyright © PreEmptive Solutions; Icon library, Copyright © Professional Icons; OSA dlls, Copyright © Sharp Electronics Corp.; JAVA components, Copyright © Sun Microsystems; Signature pad technology, Copyright © Topaz Systems Incorporated; and User interface tools, Copyright © Xceed Software, Incorporated.

Portions of the OnBase® software modules may be covered by one or more of the following U.S. Patents: 7,644,091 and 7,765,271. Portions contained within OnBase® are licensed by U.S. Patent Nos. 6,094,505; 5,768,416; 5,625,465 and 5,258,855.

Hyland Software® and OnBase® are registered trademarks of Hyland Software, Inc. Application Enabler™ is an unregistered trademark of Hyland Software, Inc. EMC Centera® is a registered trademark of EMC Corporation. All other trademarks, service marks, trade names and products of other companies are the property of their respective owners.

Attribute	Detail
Document Name	Unity Forms
Department/Group	Documentation
Revision Number	11.0.0
Part Number	UFRM-11.0.0- -OB

## Exposure

<b>OVERVIEW</b> .....	<b>1</b>
Form Data Validation .....	1
Editing Submitted Forms .....	1
Designer .....	2
<i>Testing Driving</i> .....	<i>2</i>
<i>Publishing</i> .....	<i>2</i>
<b>APPLICATIONS</b> .....	<b>3</b>
<b>LICENSING</b> .....	<b>3</b>

## Unity Client Usage

<b>CREATING A FORM</b> .....	<b>5</b>
Validating Data in Forms .....	5
Repeating Sections .....	5
Using Masking .....	7
Using Forms with Multiple Pages .....	7
Using AutoFill Keyword Sets on Forms .....	7
Required Fields.....	8
Using Date Fields.....	8
Using Drop-Down Select List .....	8
<b>RETRIEVING AND VIEWING A FORM</b> .....	<b>8</b>
Editing Submitted Forms .....	9
Saving Changes When Navigating Away From a Form .....	9
Re-indexing Unity Forms .....	10
<b>AUDIT LOG</b> .....	<b>10</b>

## Web Client Usage

<b>CREATING UNITY FORMS IN THE WEB CLIENT</b> .....	<b>11</b>
Validating Data in Forms .....	12
Repeating Sections .....	12
Using Masking .....	14
Using Forms with Multiple Pages .....	14
Using AutoFill Keyword Sets on Forms .....	14
Required Fields.....	15
Using Date Fields.....	15
Using Drop-Down Select List .....	15

## Unity Forms Configuration

<b>CONFIGURATION OVERVIEW</b> .....	<b>17</b>
<b>CONFIGURING DOCUMENT TYPES WITH UNITY FORMS</b> .....	<b>18</b>
Configuring Document Types for Similar Forms.....	18
Configuring Icons for Unity Forms .....	19
<b>USER RIGHTS</b> .....	<b>19</b>
User Rights Required for Publishing Templates.....	19
Granting User Rights for Unity Form Templates .....	19
Configuring User Groups the Right to Configure Form Templates .....	20
<b>ACCESSING THE DESIGNER</b> .....	<b>20</b>
<b>LAYOUT OF FORMS DESIGNER</b> .....	<b>21</b>
<b>CREATING A UNITY FORM TEMPLATE</b> .....	<b>23</b>
Creating Form Controls.....	25
<b><i>Available Property Controls</i></b> .....	<b>29</b>
<b><i>Properties</i></b> .....	<b>31</b>
Check Box .....	33
Multiline Text Box .....	33
Panel .....	33
Paragraph .....	33
Repeating Section .....	34
Table or Repeating Section (Multi-Instance Keyword Type Groups) .....	34
Section .....	34
Select List .....	34
Submit Button .....	35
Table .....	35
Text Box .....	35
Keyword Groups .....	36
Keywords .....	36
Identifying Unity Form Name and Associated Document Type.....	36
Tabbed Form Layout .....	37
Creating Multiple Page Forms .....	39
Adding Keyword Type Groups to a Form.....	40
Configuring Non-Keyword Text Fields .....	42
AutoFill Keyword Sets in Unity Forms.....	43
<b><i>Configuring AutoFill Keyword Sets on a Form</i></b> .....	<b>43</b>
Configuring Panels.....	45
Configuring Field Validation.....	45
<b><i>Currency Fields</i></b> .....	<b>46</b>
<b><i>Date Fields</i></b> .....	<b>48</b>
<b><i>Numeric Fields</i></b> .....	<b>49</b>

*Alphanumeric Fields* ..... 50

Configuring Values for Check Box Fields ..... 51

Setting Default Values for Check Box Fields ..... 51

Creating Data Set Fields..... 52

Configuring Self Configured Data Sets ..... 52

*Field Requirements for Self Configured Data Sets* ..... 53

Moving Form Controls ..... 53

Deleting Form Controls..... 54

**CONFIGURING FORM SECURITY** ..... 55

**TEST DRIVING FORMS** ..... 56

**SAVING A FORM TEMPLATE LOCALLY** ..... 58

**OPENING A LOCALLY SAVED FORM TEMPLATE**..... 58

**PUBLISHING TEMPLATES**..... 59

Republishing Form Templates..... 59

Sharing Forms ..... 60

*Shared Form Options* ..... 60

    Server Information ..... 60

    Authentication ..... 61

    Target URL ..... 61

    Embed ..... 61

Adding Keyword Type Groups to a Unity Form ..... 61

Headers and Footers ..... 61

Form Visibility..... 62

Disabling Specific E-Forms ..... 62

Designing Forms for Multiple Languages..... 62

**LIMITATIONS**..... 63

**KEYBOARD SHORTCUTS** ..... 64

**SYSTEM INTERACTION**..... 64

    Application Enabler..... 64

## Installation

**REQUIREMENTS**..... 65

    Unity Forms Supported Operating Systems..... 65

    Unity Hardware and Browser Requirements..... 65

    .NET Framework..... 66

*.NET Framework on Client Machines* ..... 66

    Databases Supported..... 67

*Microsoft SQL Server* ..... 67

*Oracle* ..... 67

*Sybase SQL Anywhere* ..... **68**  
Database Client / Server Version Compatibility ..... 68  
Hyland Software - MS Service Pack Statement..... 69  
About Virtual Environments ..... 70  
64-Bit Support Statement ..... 70  
Windows User Account Control Statement ..... 70  
*Modifying Configuration Files* ..... **71**  
**INSTALLATION** ..... **71**  
**CONTACTING SUPPORT**..... **71**

## OVERVIEW

The Unity Forms Designer is a tool that allows you to easily create Unity Forms. This designer provides a point-and-click interface that makes creating Unity Forms simple, yet provides a robust feature set for form configuration. The Unity Forms Designer was designed to provide users with limited technical experience the ability to create forms that are useful and aesthetically appealing, while providing users with a great wealth of technical knowledge the ability to create forms in a fraction of the time they would have spent previously. The designer provides the advanced features necessary for their business processes with no custom scripting. It also provides the ability to create standard form fields as well as form fields that are linked directly to OnBase Keyword Types. This tight integration with OnBase ensures easy configuration for building expansive and easily maintained OnBase solutions.

### Form Data Validation

Form validation allows users to ensure that data saved or submitted in a form is valid. Integrity of data is greatly improved by natively providing data validation and by signaling to users that values need to be edited to comply with expected data type values and to ensure that useful data has been entered into the form.

### Editing Submitted Forms

Submitted forms can be retrieved for viewing and edited to update information on a form. Either the form will be updated or a new revision of the form will be created, depending on whether the Document Type is configured as revisable. Also, entering information in a repeating section is supported, allowing easy data entry for Multiple Instance Keyword Type Groups values.

## Designer

1 The Unity Forms Designer allows the easy creation of useful and user configured error free form templates that will not require custom scripting, making forms easy to maintain. This designer includes drag and drop control creation for form fields, including Keyword Type value fields. In addition, templates can be saved locally to allow incremental saving during form configuration without making a form active for use in OnBase. Form templates saved locally can be accessed to complete the configuration process and save the form template into OnBase for use. Also, repeating sections with Multi-Instance Keyword Type Group values can be easily configured.

In the Designer, multiple page forms can be created. This allows you to group pertinent information onto a page and control who can access the page, adding another layer of security to a form. In addition, the Designer provides layout flexibility. Using panels, columns can be created to format data, adding an additional layer of layout control within pages.

The Designer has also added the ability to store data in non-keyword fields. Using XML, user can store data on a form without having to map each value to a Keyword Type.

Field validation can be defined in the Designer. This allows the form designer to configure what type of value a field should contain and then can be validated in the Client. Field validation can be configured for the following data types: currency, dates, numeric values, and alphanumeric values.

### **Testing Driving**

Test driving a form allows you to render the form in a browser and “use” the form in the same way you would in the Client within the Designer during configuration. You can test drive a form as different user groups to assess configured form security as well. This allows you to edit a form without publishing it and then having to republish if changes need to be made.

### **Publishing**

Once forms are published from the Designer, the forms are available for use in the Client. There is no need to import template files into a Document Type as the E-Forms module requires. In this way, Unity Forms provides one-click publishing.



## APPLICATIONS

Many customers use OnBase Unity Forms in conjunction with OnBase Workflow to implement paperless business processes, whereby standardized internal documents, such as expense reports and vacation requests, are created, stored, and routed entirely within OnBase. The addition of OnBase Web Server to this configuration enables the submission of online forms, such as order forms and membership applications, making OnBase a vehicle for facilitating ecommerce.

Unity Forms can be used for request processes that may require several levels of verification. For example: Before approval, a Human Resources department's vacation request process may require that one or more managers approve the request.

Example Applications:

- Requesting purchase orders
- Ordering office supplies
- Reporting software bugs and requesting software enhancements
- Submitting online questionnaires
- Creating shipping requests

## LICENSING

A Unity Forms license is required. Please contact your first line of support for further information regarding this license.

In order to create Unity Forms, an E-Forms license is required.

1

# UNITY CLIENT USAGE

---

2

## CREATING A FORM

To create a Unity Form in the Unity client:

1. Click the **Forms** button on the **Home** ribbon.
2. Click on the form you want to create in the **Forms** pane.
3. Enter the correct values in the form.
4. On the **Unity Form** tab, click **Submit**. Alternately, you can press **Enter** to submit a form.

After submitting a form, it is available for retrieval.

## Validating Data in Forms

Data is validated in forms to ensure that the data entered in the form is in the expected format.

Upon leaving a field after entering data by pressing the **Tab** key or by clicking in a new field, the field value is validated. When a field's value is not valid, the field box has red borders. If you click in an invalid field, a box is displayed listing information about why the value is not valid.

Upon clicking **Submit** on a form, the fields are validated. If any field contains an invalid value, a message displays stating **One or more fields are invalid.** and the form is not submitted.

## Repeating Sections

Repeating sections allow you to easily configure Multi-Instance Keyword Type Groups on a form. Repeating sections also allow you to configure non-keyword fields that you would like to have multiple instances of on a form. Repeating sections allow you to define what fields can be repeated without requiring you to define the number of instances allowed during form design. You can create fields on the fly at run time. This is accomplished by simply creating a repeating section or a table and placing controls that you want to repeat within it.

In the following example, a set of fields has been configured in a table. Upon entering data in the first set of fields, you can click **Add** to create another set of fields for data entry.

2

The screenshot shows a form titled "QUOTE REQUEST" with a dark blue header. Below the header are two input fields: "EMPLOYEE NAME" and "COMPANY NAME". Below these is a table with a light blue header row containing the text "QUOTE REQUEST" and an "Add" button. The table has three columns: "QUANTITY", "VENDOR", and "PHONE NUMBER". At the bottom of the form is a "Submit" button.

This form allows effortless field creation at the time the user needs additional fields. In the next screen, we can see that another row was added to enter an additional item.

This screenshot shows the same "QUOTE REQUEST" form, but now the table has two rows. Each row has input fields for "QUANTITY", "VENDOR", and "PHONE NUMBER". The "Add" button is now enclosed in a dashed border, and there are "Remove" buttons at the end of each row. The "Submit" button remains at the bottom.

To create a set of values for a Multi-Instance Keyword Type Group:

1. In the form, click **Add**.
2. Fields for the keywords within a Multiple Instance Keyword Type Group are displayed for input.
3. Enter the appropriate values for the keywords.
4. Repeat steps 1 to 3 for each instance of a Multi-Instance Keyword Type Group you want to add to the form.
5. Save or submit the form.

**Note:** You can remove an instance by clicking the corresponding **Remove** button.

**2**

## Using Masking

When a field has a required mask, upon entering the field, the mask is displayed.

## Using Forms with Multiple Pages

When a form has multiple pages, tabs are available at the top of the form. To access a different page, click on the corresponding tab.

## Using AutoFill Keyword Sets on Forms

When a form is configured with an AutoFill Keyword Set, a primary value can be entered on a form and the secondary values will be populated. Enter a primary keyword value and press **Tab** on the keyboard to expand the AutoFill Keyword Set. Alternately, you can press **Enter** to expand an AutoFill Keyword Set after entering a primary keyword value. Upon pressing **Enter** a second time after expanding the AutoFill Keyword Set, the form will submit.

Keyword Types belonging to an AutoFill Keyword Set that are assigned to the Document Type the form belongs to will be expanded regardless of whether the Keyword Types are on the form itself. The expanded Keyword Type values will be displayed in the keyword panel.

When using Single Instance Keyword Type Groups and standard Keyword Types together, values will be expanded in both the Keyword Type Group and across standard Keyword Types. If the primary keyword is in a Single Instance Keyword Type Group or standard Keyword Type, values in Multiple Instance Keyword Type Groups are not expanded.

When a primary Keyword Type is in a Multiple Instance Keyword Type Group, only the Keyword Types in the AutoFill Keyword Type Set that are a part of the Multiple Instance Keyword Type Group are expanded.

## Required Fields

When a field is required, a red asterisk is displayed beside the field. When a red asterisk is displayed, a value must be entered in the field before you can successfully submit the form.

## Using Date Fields

When a field is associated with date formatted data, clicking inside of the field will display the following date selection control:

June						
2011						
Su	Mo	Tu	We	Th	Fr	Sa
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

The arrow button at the top allows you to navigate to a different month. The drop-down select list allows you to select the relevant year. Click on the date to select it.

## Using Drop-Down Select List

Some fields in a form are configured to use Data Sets. You can use the drop-down select list next to these fields to select a value. In addition, you can type values into a field and all values in the Data Set that match the currently entered characters will display for selection.

## RETRIEVING AND VIEWING A FORM

To retrieve and view a form:

1. In the **Home** ribbon, click **Retrieval**.
2. Select the Document Type in which your form resides.
3. Enter search criteria.

4. Click **Find**.
5. Double-click on the form you want to open from the **Search Results** list.

## Editing Submitted Forms

To edit a submitted form:

1. Retrieve a form from the Unity Client.
2. Open the form.
3. Edit the form as needed.
4. Click **Submit**. Upon submitting the form, a new revision of a form is created if the Document Type is revisable. If the Document Type is not revisable, the form will be updated.

**2**

---

**Caution:** As a user, if you open multiple instances of a Unity Form, including within the Workflow interface, all instances of the Unity Form will be editable. When multiple instances of a Unity Form are open at the same time, there is a potential to overwrite data from within one of the instances with data from within another instance of the Unity Form. When there are multiple instance of a Unity Form open, the data in the instance of the Unity Form that is saved last is kept and overwrites any data that was previously saved in another open instance. It is recommended that a user opens only one instance of a Unity Form at a time.

---

## Saving Changes When Navigating Away From a Form

Upon leaving the form by selecting another form or by accessing another context of the Unity Client, a dialog box displays asking **You have not submitted the form. Would you like to submit the form before continuing?** You can click **Yes** to submit form changes or click **No** to cancel changes.

---

**Note:** If the form contained invalid values, the values are not saved upon navigating away from the form.

---

## Re-indexing Unity Forms

When re-indexing a Unity Form, in order to maintain the keyword values on a form, the Document Type the form is re-indexed to must have those Keyword Types assigned. If the new Document Type does not have the Keyword Types assigned, the keyword values associated with the Keyword Types not assigned to the Document Type will not be assigned to a Keyword Type. The values that were once mapped to the Keyword Types that are not assigned to the new Document Type will be saved and stored as non-keyword text data.

## AUDIT LOG

2

The document history of a Unity Form includes the history of values in form fields. To view the history of form fields within a form:

1. From an open Unity Form, click **History**, or from a document hit list, right-click on the Unity Form and select **History**. The **Document History** tab is displayed.
2. Click the **Form** tab. The **Form** tab contains information about changes to form field values.

The following columns are displayed in the **Form** tab:

Column	Description
<b>Log Date</b>	The date that the change was made to the field.
<b>Log Time</b>	The time that the change was made to the field.
<b>User name</b>	The user name of the user that made the change to the field.
<b>Field Name</b>	The name of the field that was changed.
<b>Instance ID</b>	When changes are made to a Multi-Instance Keyword Type Group, this column identifies the instance that was affected.
<b>Action</b>	The action that occurred on the field identified in the <b>Field Name</b> column. <b>Added</b> , <b>Deleted</b> , or <b>Modified</b> will be displayed.
<b>Old Value</b>	The previous value that was associated with the field.
<b>New Value</b>	The new value that was changed in the associated field.



# WEB CLIENT USAGE

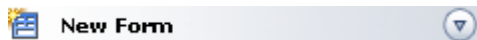
---

## 3

## CREATING UNITY FORMS IN THE WEB CLIENT

To create a new form:

1. Select **New Form** from the **Document Retrieval** drop-down.
2. This displays the **New Form** panel, which lists the forms that are available to you.



The list of forms displayed are both E-Forms and Unity Forms in the system.

3. Select the Unity Form that you want to complete and submit. The form displays.
4. Complete the form.
5. When you have finished filling out the form, submit the form.

---

**Caution:** In some cases, if a user modifies a Unity Form and navigates away from it before saving, the user is not prompted to save the changes, and the changes are not saved. Always click a submit button to ensure that changes are saved.

---

A message stating **The form is being saved.** is displayed. When the form is successfully transmitted, the system asks **Would you like to complete another form?**

6. To complete another form of the same type, click **OK**. Click **Cancel** if you are finished completing forms of this type.

---

**Note:** Date formats follow the workstation's system locale.

---

---

**Note:** When you create a new form in a revisable Document Type, a new form is created, regardless of whether or not keyword values of the new form match the keyword values of an existing form.

---

---

**Tip:** You can cut (Ctrl + x), copy (Ctrl + c), and paste (Ctrl + v) text to and from form fields.

---

## Validating Data in Forms

Data is validated in forms to ensure that the data entered in the form is in the expected format.

Upon leaving a field after entering data by pressing the **Tab** key or by clicking in a new field, the field value is validated. When a field's value is not valid, the field box has red borders. If you click in an invalid field, a box is displayed listing information about why the value is not valid.

Upon clicking **Submit** on a form, the fields are validated. If any field contains an invalid value, a message displays stating **One or more fields are invalid.** and the form is not submitted.

## Repeating Sections

Repeating sections allow you to easily configure Multi-Instance Keyword Type Groups on a form. Repeating sections also allow you to configure non-keyword fields that you would like to have multiple instances of on a form. Repeating sections allow you to define what fields can be repeated without requiring you to define the number of instances allowed during form design. You can create fields on the fly at run time. This is accomplished by simply creating a repeating section or a table and placing controls that you want to repeat within it.

In the following example, a set of fields has been configured in a table. Upon entering data in the first set of fields, you can click **Add** to create another set of fields for data entry.

The screenshot shows a form titled "QUOTE REQUEST" with a dark blue header. Below the header are two input fields: "EMPLOYEE NAME" and "COMPANY NAME". Below these is a table with a light blue header row containing the text "QUOTE REQUEST" and an "Add" button. The table has three columns: "QUANTITY", "VENDOR", and "PHONE NUMBER". At the bottom of the form is a "Submit" button.

3

This form allows effortless field creation at the time the user needs additional fields. In the next screen, we can see that another row was added to enter an additional item.

The screenshot shows the same "QUOTE REQUEST" form, but now the table has two rows. Each row has input fields for "QUANTITY", "VENDOR", and "PHONE NUMBER". The "Add" button is now outlined with a dashed border, and a "Remove" button has appeared at the end of each row. The "Submit" button remains at the bottom.

To create a set of values for a Multi-Instance Keyword Type Group:

1. In the form, click **Add**.
2. Fields for the keywords within a Multiple Instance Keyword Type Group are displayed for input.
3. Enter the appropriate values for the keywords.
4. Repeat steps 1 to 3 for each instance of a Multi-Instance Keyword Type Group you want to add to the form.
5. Save or submit the form.

---

**Note:** You can remove an instance by clicking the corresponding **Remove** button.

---

**3**

## Using Masking

When a field has a required mask, upon entering the field, the mask is displayed.

## Using Forms with Multiple Pages

When a form has multiple pages, tabs are available at the top of the form. To access a different page, click on the corresponding tab.

## Using AutoFill Keyword Sets on Forms

When a form is configured with an AutoFill Keyword Set, a primary value can be entered on a form and the secondary values will be populated. Enter a primary keyword value and press **Tab** on the keyboard to expand the AutoFill Keyword Set. Alternately, you can press **Enter** to expand an AutoFill Keyword Set after entering a primary keyword value. Upon pressing **Enter** a second time after expanding the AutoFill Keyword Set, the form will submit.

Keyword Types belonging to an AutoFill Keyword Set that are assigned to the Document Type the form belongs to will be expanded regardless of whether the Keyword Types are on the form itself. The expanded Keyword Type values will be displayed in the keyword panel.

When using Single Instance Keyword Type Groups and standard Keyword Types together, values will be expanded in both the Keyword Type Group and across standard Keyword Types. If the primary keyword is in a Single Instance Keyword Type Group or standard Keyword Type, values in Multiple Instance Keyword Type Groups are not expanded.

When a primary Keyword Type is in a Multiple Instance Keyword Type Group, only the Keyword Types in the AutoFill Keyword Type Set that are a part of the Multiple Instance Keyword Type Group are expanded.

## Required Fields

When a field is required, a red asterisk is displayed beside the field. When a red asterisk is displayed, a value must be entered in the field before you can successfully submit the form.

## Using Date Fields

When a field is associated with date formatted data, clicking inside of the field will display the following date selection control:

June						
2011						
Su	Mo	Tu	We	Th	Fr	Sa
			1	2	3	4
5	6	7	8	9	10	11
12	13	14	15	16	17	18
19	20	21	22	23	24	25
26	27	28	29	30		

The arrow button at the top allows you to navigate to a different month. The drop-down select list allows you to select the relevant year. Click on the date to select it.

## Using Drop-Down Select List

Some fields in a form are configured to use Data Sets. You can use the drop-down select list next to these fields to select a value. In addition, you can type values into a field and all values in the Data Set that match the currently entered characters will display for selection.

**3**

# UNITY FORMS CONFIGURATION

---

## 4

## CONFIGURATION OVERVIEW

The following steps are necessary to configure a Unity Form template:

1. Create a Document Type to associate with the form template. See *Configuring Document Types with Unity Forms* on page 18 for more information.
2. Configure User Rights for Unity Forms. See *User Rights* on page 19 for more information.
3. Access the Designer. See *Accessing the Designer* on page 20 for more information.
4. Create the form template. See *Creating a Unity Form Template* on page 23 for more information.
5. Configure form security. See *Configuring Form Security* on page 55 for more information.
6. Publish the form template. See *Publishing Templates* on page 59 for more information.

---

**Note:** Unity Forms can be created, retrieved, viewed, and modified in the Unity Client and Web Client only. Unity Form templates can only be designed in the Unity Client Form Designer.

---

## CONFIGURING DOCUMENT TYPES WITH UNITY FORMS

There are some specific things that need to be configured related to Document Types before configuring a form template.

- A specific default file format is not required for Document Types that are associated with Unity Forms. The Unity Forms engine does not rely on any specific default file format to be configured at the Document Type level. The Unity Forms file type is reserved for and used by the OnBase system to store new Unity Forms created in the Client.
- All Keyword Types and Keyword Type Groups that will be needed on a form should be configured on the Document Type that will be used in conjunction with the form. AutoFill Keyword Sets used in Unity Forms also need to be configured and assigned to the appropriate Keyword Types in the Configuration module.

---

**Note:** If the Keyword Types assigned to a Document Type associated with Unity Forms are changed, the values on existing Unity Forms associated with Keyword Types that are no longer assigned to the Document Type will be saved as non-keyword data.

---

- At the time of form template configuration in the Unity Forms Designer, a form template is associated with a Document Type.

### Configuring Document Types for Similar Forms

In some cases, multiple forms that have similar purposes, but differ slightly, need to be routed through the same business process. Here are a few examples of when this may occur:

- A form can be slightly different depending on which state invoices were generated in. State laws and taxation can require different data and calculations from state to state to be incorporated on a form. In this scenario, a form template would be created for each state involved in the process tailored to the data requirements for that state.
- A form needs to be available on an external web site, as well as internally in a company. The external form needs to have more instructions and has a fewer number of fields on it than the internal version. The forms follow the same exact process.
- A form needs to be in multiple languages to support users internationally. The form is the same in every way except the language that is displayed on the form.

If all of the forms follow the same process that is defined in a Workflow Life Cycle, it would not make sense to create a Workflow Life Cycle for each form when these similar forms follow the exact same process.



In this scenario, it would be most efficient to associate all of the similar form templates with the same Document Type. Multiple Unity form templates can be configured and associated with a single Document Type. Multiple templates should be associated with a single Document Type only when all of the templates are used for the same purpose.

When using multiple templates within a single Document Type, it is recommended to use a Keyword Type to clearly identify what template a document is using and to configure an autoname string that will clearly identify it.

## Configuring Icons for Unity Forms

When configuring a Document Type that will be used for a Unity Form template, you can configure an icon that will be displayed next to the Unity Form name during form creation in the Unity Client. To define the icon used for the Unity Form in the Unity Client:

1. In the Configuration module, select **Document | Document Types**.
2. Select the Document Type.
3. Click **Icon**.
4. Select the icon you want to associate with the Unity Form from the **Icon** drop-down select menu.
5. Click **Save**.

## USER RIGHTS

### User Rights Required for Publishing Templates

In order to publish Unity form templates, a user must have rights to SYS HTML Forms Document Type. When a template is published, it is stored in the SYS HTML Forms Document Type.

### Granting User Rights for Unity Form Templates

In order to grant user groups the rights to use specific sections or pages within a form template, there are rights a user must have.

In order for a user to grant rights to other users that belong to their user group(s), a user must belong to a user group that has the **Usergroup Security** in the **Configuration Rights** dialog box, granted at the user group level. If a user belongs to a user group that has this right granted, the user can configure user groups rights on form templates for all the user groups to which he or she belongs.

In order for a user to grants rights to all user groups, the user must have the **User Group Administrator** right granted in the **User Settings** dialog box.

## Configuring User Groups the Right to Configure Form Templates

In order for user groups to have the ability to configure Unity Form templates, they must be granted the **Unity Forms** Configuration right. If a user group does not have this right granted, users in the user group will not have access to the **Unity Forms | Designer** menu. To grant user groups the right to configure Unity Form templates:

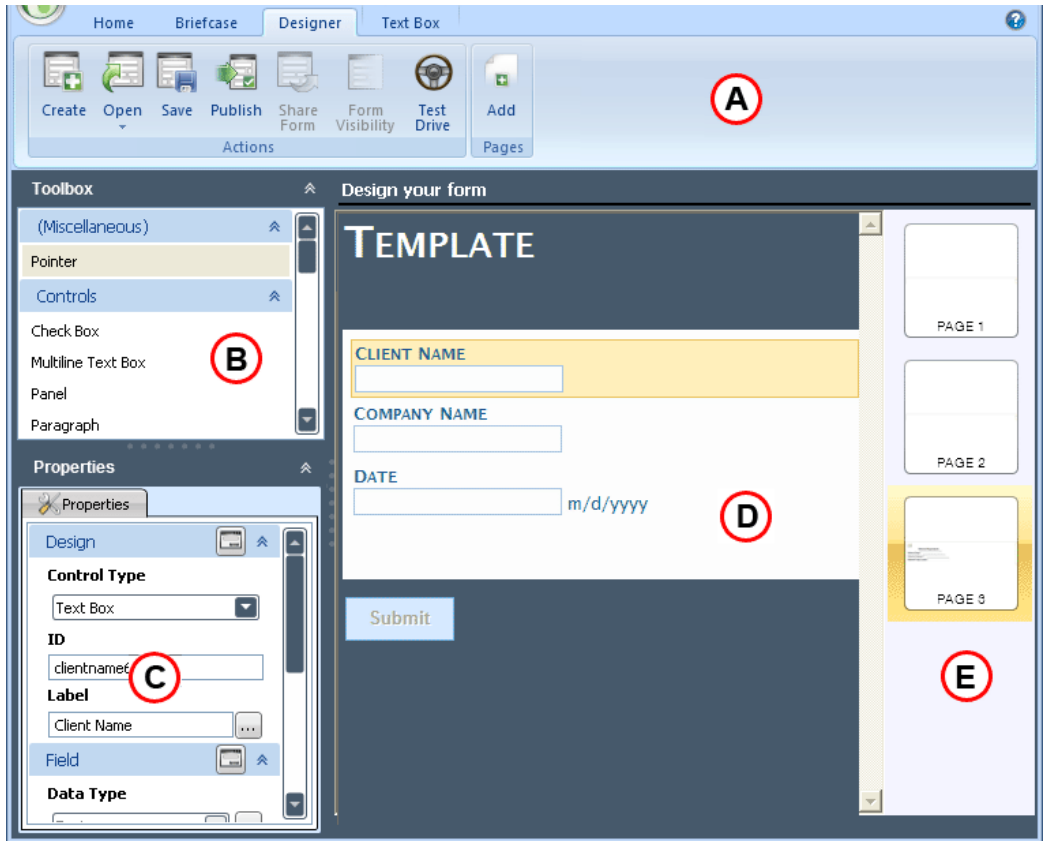
1. In the Configuration module, select **Users | User Groups/Rights**.
2. Select the user group you want to grant rights to.
3. Click **Configuration Rights**.
4. Select the **Products** tab.
5. Select **Unity Forms**.
6. Click **Save**.

## ACCESSING THE DESIGNER

In the Unity Client, select **Unity Forms | Designer**. The **Designer** tab is displayed in the Unity Client.

# LAYOUT OF FORMS DESIGNER

The following is the layout of the Forms Designer.



**A**

The **Designer** ribbon contains the buttons that allow you to accomplish the following actions:

- create a new form template
- open an existing form template
- save a template
- publish a template
- test drive a template before publishing
- add pages to the template.

In addition to the **Designer** ribbon, a ribbon is displayed for the selected control in the form. In the example, a text box is selected. In this example, this is accessed by clicking on the **Text Box** tab.

---

**B**

The **Toolbox** window contains all of the form controls that can be placed on a form. All controls can be added to the form by clicking on the control in the **Toolbox** and then clicking within the design area.

---

**C**

The **Properties** window contains the **Properties** tab.

The **Properties** tab contains all of the properties that can define a visual display of a control. Each control has different property options. The **Properties** tab also allows you to customize fields for validation and masking. In addition, field requirements and AutoFill Keyword Sets field configuration can be configured in this tab.

---

**D**

The **Design your form** area is the area of the screen where you design the layout of the form. The design area displays the page that is selected in the page pane.

---

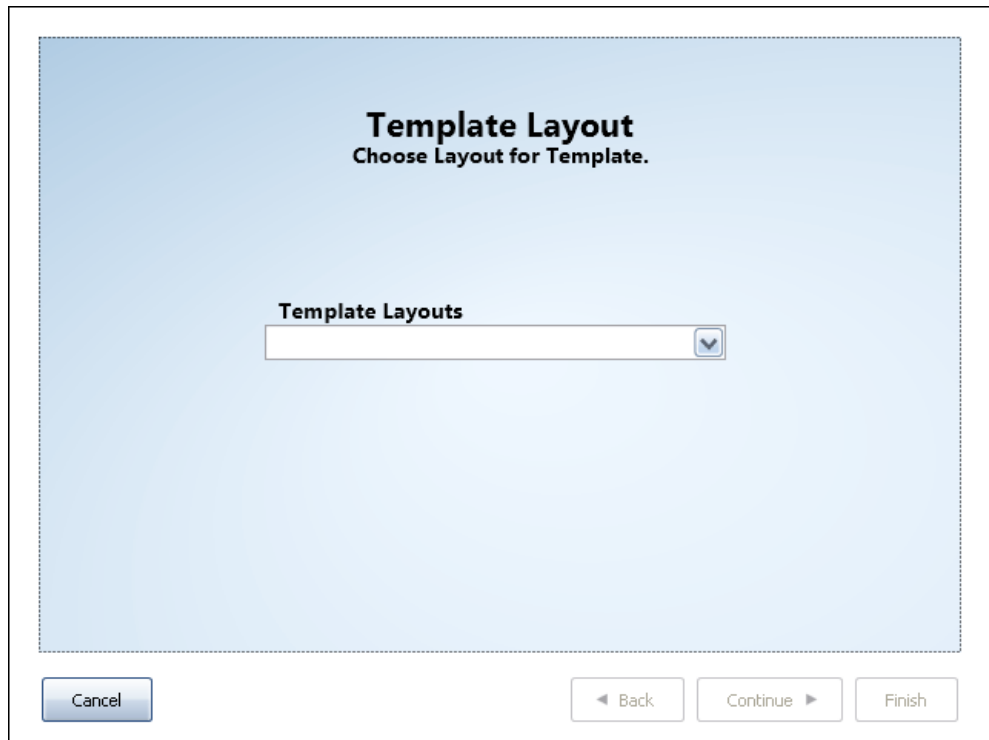
**E**

The page pane contains all of the pages the form contains. In the Client, each page is displayed as a tab.

## CREATING A UNITY FORM TEMPLATE

In the **Designer** tab:

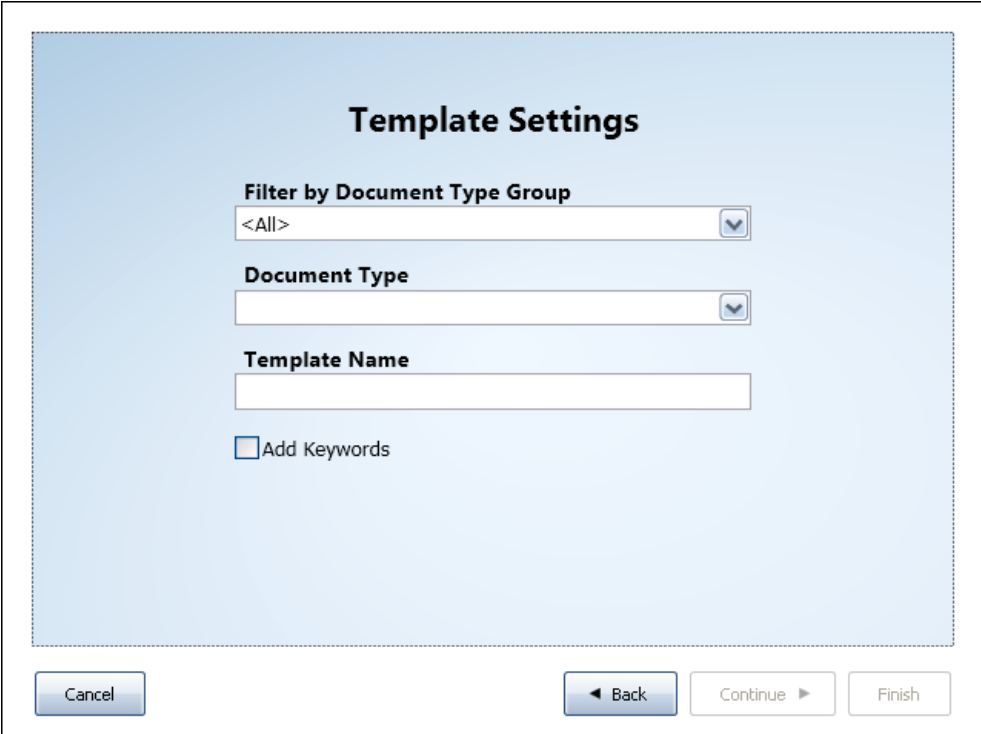
1. Click **Create**. A new form wizard is launched.



The screenshot shows a dialog box titled "Template Layout" with the subtitle "Choose Layout for Template." The dialog has a light blue background. In the center, there is a label "Template Layouts" above a white drop-down menu with a blue arrow icon on the right. At the bottom of the dialog, there are three buttons: "Cancel" on the left, "Back" in the middle (with a left-pointing arrow), and "Continue" on the right (with a right-pointing arrow). Below the "Continue" button is a "Finish" button.

2. Select a form template from the **Template Layouts** drop-down. You can select Blank, Standard, or Multi-Page. Blank provides a completely blank form. Standard provides a default header and footer on the form. Multi-Page provides a form with multiple pages with default headers and footers on each page of the form.

3. Click **Continue**.



**Template Settings**

**Filter by Document Type Group**  
<All>

**Document Type**  
[Empty dropdown]

**Template Name**  
[Empty text box]

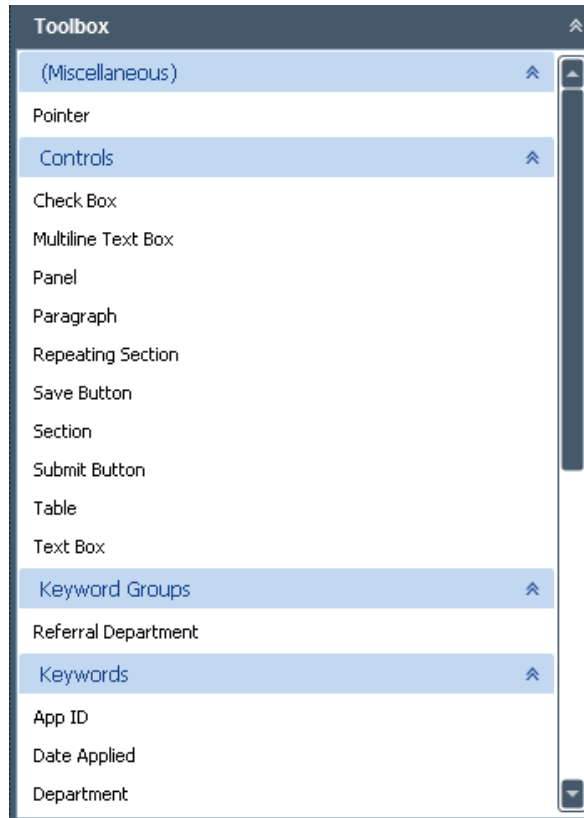
Add Keywords

Cancel      Back      Continue      Finish

4. Select a Document Type from the **Filter by Document Type Group** to narrow the Document Types available for selection.
5. Select the Document Type in which you would like to store the forms that originate from this template from the **Document Type** drop-down list.
6. Enter a **Template Name**.
7. If you want to initially add all of the Keyword Types assigned to the selected Document Type as fields on the form, select the **Add Keywords** check box.
8. Click **Finish**.

## Creating Form Controls

Field options are available in the **Toolbox** window. With two simple clicks, form fields can be added to a form. Select a control from the **Toolbox** and click in the design area to place the field in the location desired.



Keywords and Keyword Type Groups fields are automatically mapped to the appropriate OnBase Keyword. No additional configuration or coding is necessary.

Once an HTML template is created, you can add controls to a form.

To add a control to a form:

1. Select the control you want to add in the **Toolbox** window.
2. Place your cursor where you want to drop the control. The control will be dropped into the form in the area that is highlighted in blue. The cursor will display as a cross when the cursor is in an area where a control can be placed.



3. Click in the form editor in the position you want the control placed.

The following controls are available for selection in the **Toolbox** window:

Option	Description
<b>Pointer</b>	This option allows you to select existing form fields for editing and deleting.
<b>Check Box</b>	<p>Inserts a check box field that can be edited by users when forms are created.</p> <p>Check boxes can be placed within panels, repeating sections, sections, tables, and the body of the form.</p> <hr/> <p><b>Note:</b> Check boxes cannot be mapped to date or currency formatted fields.</p> <hr/>
<b>Multiline Text Box</b>	<p>Inserts a text field that allows multiple lines of text that can be edited by users when forms are created and contains a scroll bar.</p> <p>Multiline text boxes can be placed within panels, repeating sections, sections, tables, and the body of the form.</p>
<b>Panel</b>	<p>Inserts a container element that can be used to group other elements.</p> <p>Any control type can be placed within a panel.</p> <p>Panels can be placed within panels, repeating sections, sections, and the body of the form.</p>



Option	Description
<p><b>Paragraph</b></p>	<p>Inserts a paragraph into the form. The paragraph is defined in the designer. This option is used to present read-only information to users when a form is being created.</p> <p>Paragraphs can be placed within panels, repeating sections, sections, and the body of the form.</p>
<p><b>Repeating Section</b></p>	<p>Inserts a repeating section. When configuring Multiple Instance Keyword Type Groups and using a repeating section, other controls cannot be placed in a Multiple Instance Keyword Type Group's repeating section. You must create a standalone repeating section for other controls.</p> <p>Repeating sections can also contain check boxes, multiline text boxes, panels, paragraphs, sections, and text boxes.</p> <p>Section controls can be placed within a repeating section control.</p> <p>Repeating sections can be placed within panels, sections, and the body of the form.</p> <hr/> <p><b>Note:</b> Only Keyword Types that belong to a Multi-Instance Keyword Type Group can be added to a repeating section.</p> <hr/>
<p><b>Section</b></p>	<p>Inserts a section. A section can have other controls dropped into it and it can be configured to be collapsible. A section divider line is also inserted.</p> <p>Any control type can be placed within a section.</p> <p>Sections can be placed within panels, repeating sections, other sections, and the body of the form.</p>
<p><b>Select List</b></p>	<p>Inserts a field for which a self-configurable date set can be configured. A drop-down select list is available in the Client.</p> <p>Select lists can be placed with panels, repeating sections, sections, tables, and the body of the form.</p>
<p><b>Submit Button</b></p>	<p>Inserts a <b>Submit</b> button.</p> <p>Submit buttons can be placed within panels, sections, or the body of the form.</p>

Option	Description
<b>Table</b>	<p>Inserts a table that allows you to render data in a table format. Text boxes, multiline text boxes, and check boxes can be added to tables.</p> <p>Tables can contain check boxes, multiline text boxes, and text boxes.</p> <p>Tables can be placed within panels, sections, and the body of the form.</p>
<b>Text Box</b>	<p>Inserts a single line text field that can be edited by users when forms are created.</p> <p>Text boxes can be placed with panels, repeating sections, sections, tables, and the body of the form.</p>
<b>Keyword Groups</b>	<p>When the Document Type selected to save the template in has Multi-Instance Keyword Type Groups configured, a list of the Keyword Type Groups are available for insertion in the template. You can place a Multi-Instance Keyword Type Group in a repeating section or a table.</p> <p>When a Multi- Instance Keyword Type Group is placed in a template and <b>Repeater</b> is selected, a repeating section is created and automatically associated with the group. Only the Multi-Instance Keyword Type Group can be placed in this repeating section. No other controls can be added.</p> <p>When a Multi- Instance Keyword Type Group is placed in a template and <b>Table</b> is selected, a table is created and automatically associated with the group. Only the Multi-Instance Keyword Type Group can be placed in this table. No other controls can be added.</p>

Option	Description
<p><b>Keywords</b></p>	<p>When the Document Type selected to save the template in has Keyword Types or Single Instance Keyword Type Groups configured, a list of Keyword Types related to the Document Type is available for insertion into the template.</p> <hr/> <p><b>Note:</b> Only one instance of a Keyword Type can be placed within a form. If a form is a multiple page form, only one instance of a Keyword Type can exist among all of the pages.</p> <hr/> <p><b>Note:</b> Encrypted Keyword Types, Date &amp; Time, and Specific Currency Keyword Types are not available for selection in the Unity Forms Designer.</p> <hr/> <p><b>Note:</b> If you are using a Keyword Type configured to be <b>Invisible</b> and you place the Keyword Type on a Unity Form, you must use Unity Forms security to hide the field. Otherwise, the Keyword Type will display on the form.</p> <hr/>
<p><b>Properties</b></p>	<p>Inserts a system property on the form. See Available Property Controls on page 29 for more information.</p>

**Available Property Controls**

Property	Description
<p><b>Created By</b></p>	<p>The user who brought the document into the system.</p>
<p><b>Current User's Display Name</b></p>	<p>Reflects the display name of the currently logged in user, based on the Global Client Setting in the Configuration module. If the Global Client Setting is set to show the user's real name, and no real name has been configured, this property will display the OnBase user name.</p>
<p><b>Current User's ID</b></p>	<p>Reflects the User ID of the currently logged in user.</p>
<p><b>Current User's Locale</b></p>	<p>Displays the user locale of the currently logged in user's session on the form. It will be displayed in the format "language-region". For example, the United States English locale would be displayed EN-US.</p>
<p><b>Current User's Name</b></p>	<p>Reflects the OnBase user name of the currently logged in user.</p>

Property	Description
<b>Current User's Real Name</b>	Reflects the real name of the currently logged in user, which is specified in the Configuration module. If no real name has been specified, this property will be left blank.
<b>Date Stored</b>	The date when the document was imported into the system. If an invoice from December 28, 1996, was brought into the system on March 11, 1997, December 28, 1996, is the document date and March 11, 1997, is the date stored. Documents cannot be searched based on the date stored.
<b>Document Date</b>	The document date. The document date is assigned to a document at the time of import.
<b>Document ID</b>	The document handle, which is the unique number that identifies a document in the database.
<b>Form Revision</b>	When a Unity Form template is edited and republished, the form revision is incremented by 1. This property will display the form template revision that is used for the displayed form.
<b>Time Stored</b>	The time when the document was imported into the system.

You can edit the properties of an added control. To edit the properties of a control:

1. Select the **Pointer** option in the **Toolbox** window.
2. Click on the control in the editor that you want to edit.
3. The **Properties** window will display the properties options available for the type of control that was selected.
4. Make the appropriate property changes.

## Properties

Each control type has a set of properties that can be used to define a control. The following properties are available for all control types:

Property	Description
<b>ID</b>	<p>The ID name for the field. This ID value should be unique.</p> <p>The following are rules for entering ID values:</p> <ul style="list-style-type: none"> <li>• You cannot paste into the ID field.</li> <li>• You cannot type more than 75 characters into ID field.</li> <li>• You cannot type spaces into ID field.</li> <li>• You cannot type characters not allowed in the current language you have selected for naming an HTML field.</li> <li>• Valid characters for IDs: [a-z][A-Z][0-9].</li> <li>• ID names must start with [a-z] or [A-Z].</li> <li>• Special characters, such as underscores, cannot be entered as part of an ID name.</li> </ul>

4

Each field control has the following option:

Property	Description
<b>Control Type</b>	This defines the type of control associated with the field. You can change a fields control type by selecting an option from this drop-down select list.

Fields that data is entered into have the following properties available. Some properties are dependent on the Data Type selected for the field:

Property	Description
<b>Data Type</b>	<p>This property defines the data type associated with the field. The options are: <b>Text</b>, <b>Numeric (Up to 9 Digits)</b>, <b>Numeric (Up to 20 Digits)</b>, <b>Floating Point</b>, <b>Date</b>, or <b>Currency</b>.</p> <hr/> <p><b>Note:</b> This property is not editable for fields configured as System Properties.</p> <hr/>

Property	Description
<p><b>Date Range</b></p>	<p><b>Note:</b>Available when configuring a Date field.</p> <p>This property displays the range of dates the field can accept as a valid date. To configure this range, click the ellipse button next to the field. See Date Fields on page 48 for more information.</p>
<p><b>Default Value</b></p>	<p>The value specified in this field is the value associated with the field by default.</p> <p><b>Note:</b>When configuring a check box, enter the value of <b>Check Value</b> property in the <b>Default Value</b> property field if you want the check box to be selected by default. Entering any other value will leave the check box unselected by default.</p>
<p><b>Field</b></p>	<p>This property allows you to set a field to be a data field not associated with Keyword Type or System Property by selecting <b>XML</b>. You can associate a field with a Keyword Type or System Property by selecting it from the drop-down select list.</p>
<p><b>Mask</b></p>	<p><b>Note:</b>Available when configuring a Text field.</p> <p>The property displays the masking configured for a field. To configure a mask for an alphanumeric date field, click the ellipsis button next to the property. See Alphanumeric Fields on page 50 for more information.</p>
<p><b>Max Length</b></p>	<p><b>Note:</b>Available when configuring a Text field.</p> <p>This property defines the maximum number of characters allowed in the field. See Alphanumeric Fields on page 50 for more information.</p>
<p><b>Numeric Range</b></p>	<p><b>Note:</b>Available when configuring a Numeric or Floating Point field.</p> <p>This property displays the range of numeric values in which the field can accept as a valid value. To configure this range, click the ellipse button next to the field. See Currency Fields on page 46 for more information.</p>

Property	Description
<b>Required</b>	When this property is selected, the field is required. When this property is not selected, the field is not required.  <b>Note:</b> This property is not available for fields configured as a System Property field.

**CHECK BOX**

Property	Description
<b>CheckedValue</b>	This property allows you to define the value that is associated with the field when it is checked.
<b>Label</b>	This property allows you to define text that will be used to define the check box field. Enter the label text in the property's field.



**MULTILINE TEXT BOX**

Property	Description
<b>Label</b>	This property allows you to define text that will be used to define the text box field. Enter the label text in the property's field.

**PANEL**

Property	Description
<b>Columns</b>	This property allows you to specify the number of columns in the panel. You can select <b>1 Column</b> , <b>2 Columns</b> , or <b>3 Columns</b> .

**PARAGRAPH**

Property	Description
<b>Content</b>	This property allows you to define a paragraph of text that will be displayed in the form as read-only.
<b>Label</b>	This property allows you to define text that will be used to define the paragraph field. Enter the label text in the property's field.

**REPEATING SECTION**

Property	Description
<b>Label</b>	This property allows you to define text that will be used to define the repeating section. Enter the label text in the property's field.

**TABLE OR REPEATING SECTION (MULTI-INSTANCE KEYWORD TYPE GROUPS)**

Property	Description
<b>Label</b>	This property allows you to define text that will be used to define the repeating section. Enter the label text in the property's field.
<b>Keywords</b>	Clicking the ellipsis button next to this property allows you to access the Keyword Type configuration screen for the selecting repeating section/table. This allows you to add or remove Keyword Types from the repeating section/table on the form.

**SECTION**

Property	Description
<b>DefaultVisualState</b>	Define the default view of the selected section. Options are <b>Expanded</b> or <b>Collapsed</b> .
<b>IsCollapsible</b>	Defines whether or not the selected section can be expanded or collapsed. Options are <b>true</b> or <b>false</b> .
<b>Label</b>	This property allows you to define the text that will be used as the title for the selected section. Enter the label text in the property's field.

For information on the Security properties, see *Configuring Form Security* on page 55.

**SELECT LIST**

Property	Description
<b>Label</b>	This property allows you to define text that will be used to define the select list field. Enter the label text in the property's field.



Property	Description
<b>Data Set</b>	Displays the Data Set number that the field is associated with. You can configure a Self Configured Data Set to associate with the field by clicking on the ellipsis button.
<b>Value must exist</b>	When this property is selected, the value must exist in the Data Set associated with the Keyword Type in order to save data in the field. If the <b>Value must exist</b> option is unchecked, the value does not have to exist in the Data Set in order to be saved. This will override the <b>Keyword Must Exist</b> setting in the Configuration module.  <hr/> <b>Note:</b> Data set value comparison is case insensitive. <hr/>

**SUBMIT BUTTON**

Property	Description
<b>Label</b>	This property allows you to define text that will be used to define the submit button. Enter the label text in the property's field.

**TABLE**

Property	Description
<b>Label</b>	This property allows you to define text that will be used to define the Table. Enter the label text in the property's field.

**TEXT BOX**

Property	Description
<b>Label</b>	This property allows you to define text that will be used to define the text box field. Enter the label text in the property's field.
<b>Maximum Length</b>	This property allows you to define the maximum length of the value entered in the selected text box field.

## KEYWORD GROUPS

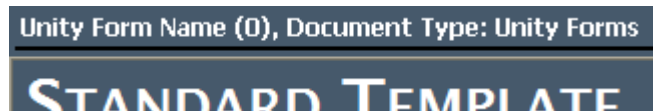
Property	Description
<b>ID</b>	This value is system generated and uniquely identifies the Multiple Instance Keyword Type. This value is not editable.
<b>Title</b>	Allows you to define a label for the system generated Repeating Section associated with the Multiple Instance Keyword Type Group.

## KEYWORDS

Property	Description
<b>ID</b>	This value is system generated and uniquely identifies the field. This value is not editable.
<b>Label</b>	This property allows you to define text that will be used to define the Keyword Type field. Enter the label text in the property's field.

## Identifying Unity Form Name and Associated Document Type

When working in the Unity Forms Designer, you can identify what the name of the form is, the document id number of the form template, and the Document Type the form template is associated with. This information is displayed above the design space. The following is an example:



First, the name of the Unity Form is displayed. Next, the document id number of the form template is displayed in parentheses. If the form template has not been published, the document id is 0. Last, the Document Type that the form template belongs to is displayed.

## Tabbed Form Layout

To create forms with a tabbed layout, multiple pages are configured. The number of pages corresponds to the number of tabs in a form. The pages are displayed in the page pane in the designer.



In the Client, the pages are displayed as tabs. The above Designer configuration looks like the following in the Client:

4

The screenshot shows a tabbed interface with three tabs: 'Referral Info' (selected), 'Application Info', and 'Additional Info'. The 'Referral Info' tab contains the following elements:

- A header bar with the text 'REFERRAL DEPARTMENT' and an 'Add' button on the right.
- A text input field labeled 'REFERRAL DEPT'.
- A text input field labeled 'REFERRAL MANAGER' with a 'Remove' button to its right.
- A text input field labeled 'REFERRAL TEAM LEADER'.

You can click on another tab to access different information. Clicking **Application Info** displays the following fields:

The screenshot shows the 'Application Info' tab selected. The interface includes the following elements:

- Three tabs: 'Referral Info', 'Application Info' (selected), and 'Additional Info'.
- A header bar with three columns: 'FIRST NAME', 'LAST NAME', and 'JOB TITLE', and an 'Add' button on the right.
- Three corresponding text input fields for 'FIRST NAME', 'LAST NAME', and 'JOB TITLE'.
- A 'Remove' button on the right side.
- A 'Submit' button at the bottom left of the form area.

Each tab can have separate security. This allows pertinent information to be grouped together on a page and security can control what users see a page, adding another layer of security to a form.

## Creating Multiple Page Forms

To add another page to an HTML form, with a template open in the designer, click **Add New Page** on the **Designer** tab. A new page is displayed. Once the new page is created, you can add form controls to the page.

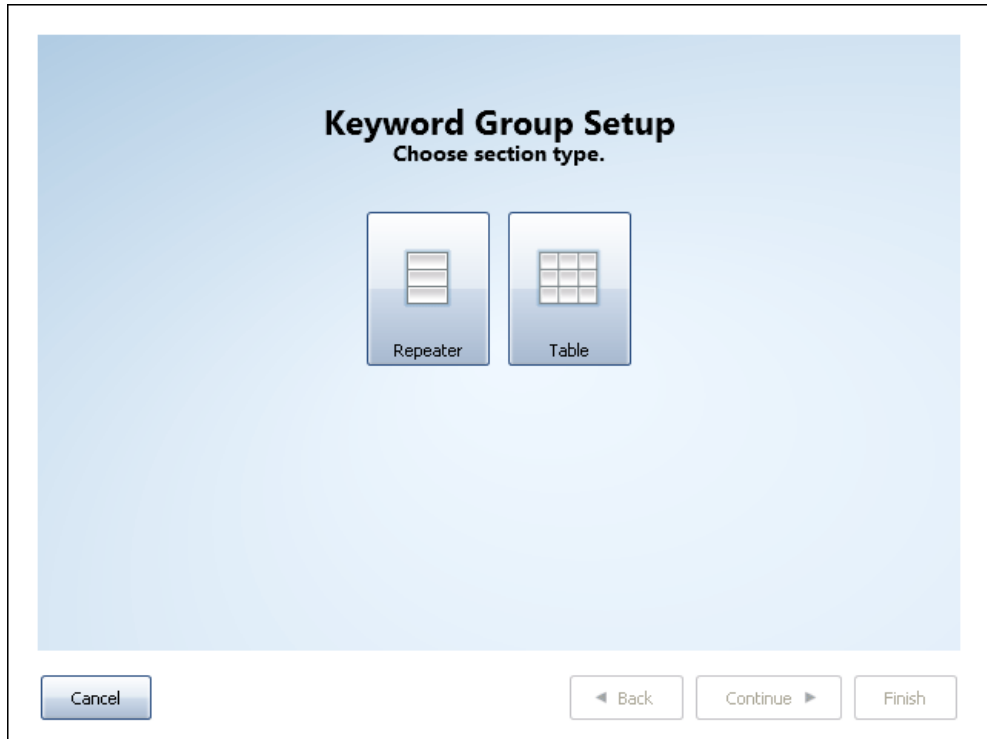
You can navigate from one page to another by clicking on the page thumbnails in the right pane. You can also delete a page by hovering over the page and clicking the red X button that corresponds to the page.

You can also reorder pages of a form. To reorder pages, click on a page's thumbnail and drag it to the correct position in the sequence.

## Adding Keyword Type Groups to a Form

To add a Multiple Instance Keyword Type Group to a form:

1. Select a Keyword Type Group from the **Keyword Groups** toolbox.
2. Click in the location on the form where you want to place the control. The **Keyword Group Setup** screen is displayed.



3. Click **Repeater** if you want to place the Keyword Type Group in a repeater section, or click **Table** to place the Keyword Type Group in a table control.

- Click **Continue**. The **Keyword Selection** screen is displayed.

**Keyword Selection**  
Choose the Keywords to Add to the Form.

Available Keyword Types

Name  
Address  
Phone Number

Selected Keyword Types

Cancel Back Continue Finish

4

- Select a Keyword Type you would like to include on the form and click **Add Selected Items**. Alternately, you can double-click on a Keyword Type to add it. Repeat this step for each Keyword Type you would like to include.

**Note:** You can select multiple Keyword Types at one time by pressing **Ctrl** on the keyboard while selecting Keyword Types. You can select a block of Keyword Types at one time by pressing **Shift** on the keyboard, selecting the first Keyword Type in the list that you want to add, and then selecting the last Keyword Type in the list that you want to add.

If you want to add all of the Keyword Types to the form, click the **Add All Items** button.

**Note:** You can use the **Remove Selected Items** button to remove a Keyword Type from the **Selected Keyword Types** box or click **Remove All Items** to remove all of the Keyword Types.

- Click **Finish**.

**Note:** You can edit the Keyword Types placed in the repeating section/table by clicking the ellipsis button next to the **Keywords** property.

## Configuring Non-Keyword Text Fields

When a text box is placed on a form, by default it is configured as a non-keyword field that will store the data entered as XML in the post information on the disk group. This allows data to be stored in a form without requiring fields to be mapped to Keyword Types. To verify that a field is configured as a non-keyword field:

1. Select the field on the form.
2. In the **Properties** pane, the **Field** property should display as **XML**.
3. To select the data type associated with the field, click the ellipsis button next to the Data Type property field.

### Field Type

Choose Data Type for Field.

1234


Number

ABC

Text

\$£¥

Currency



Date

**Max Length**

**Mask**  Static Mode

Mask Mode is selected. Press Alt+S to switch to Static Mode.  
 [A = Letter] [X = Alphanumeric] [0 = Numbers] [9 = Numbers (including '+', '-', '.', '\*')]

Required

Cancel
Finish



4. Select the data type for the field. You can select **Number**, **Text**, **Currency**, or **Date**.
5. Click **Finish**.

To add Keyword Type fields to a form template, place the appropriate **Keyword** control on the form.

## AutoFill Keyword Sets in Unity Forms

AutoFill Keyword Set configuration is simple. All that is needed is to select an AutoFill Keyword Type Set for a primary Keyword Type from a drop-down select list. All Keyword Types belonging to a AutoFill Keyword Set are automatically filled with values when the primary value is entered and **Tab** is pressed.

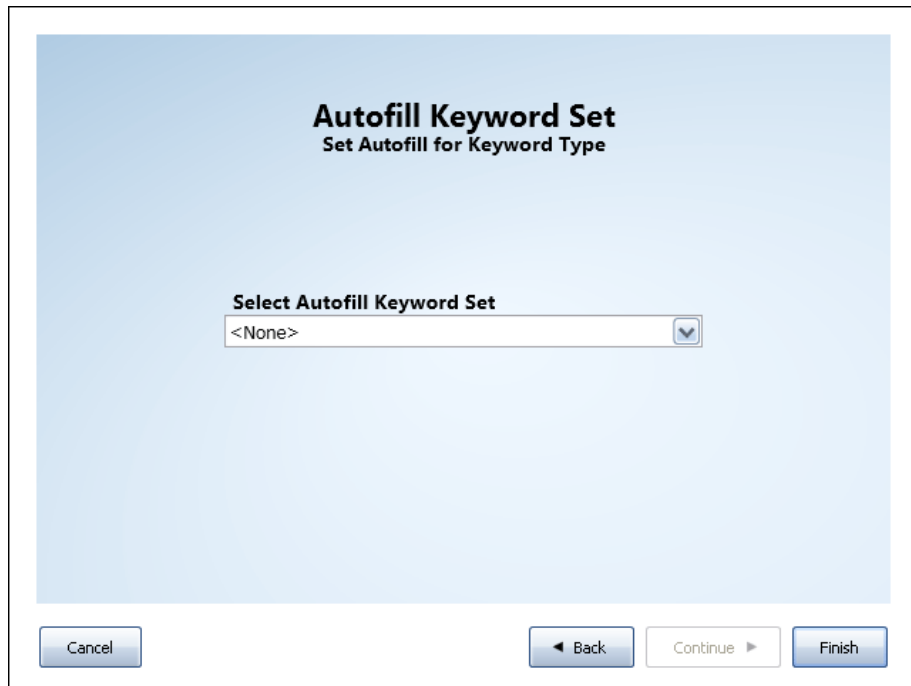
Any AutoFill Keyword Set that is configured in OnBase can be used. This includes AutoFill Keyword Sets that are linked to external data sources. For more information about AutoFill Keyword Sets, please see the AutoFill Keyword Sets Module Reference Guide or help files.

### **Configuring AutoFill Keyword Sets on a Form**

To configure an AutoFill Keyword Set on a form:

1. In the Designer, place the Keyword Types that are a part of the AutoFill Keyword Set on the form.
2. Select the primary Keyword Type field.

- In the **Properties** pane, select the ellipsis button next to the **Autofill** property.



**Autofill Keyword Set**  
Set Autofill for Keyword Type

Select Autofill Keyword Set

<None>

Cancel Back Continue Finish

- On the **AutoFill Keyword Set** screen, select the appropriate AutoFill Keyword Set from the **Select AutoFill Keyword Set** drop-down select list.
- With the **Create Autofill record on create** option selected, when a new form of this form template type is created and the primary value entered does not exist in the associated AutoFill Keyword Set, a new AutoFill Keyword Set record will be created for the primary value.
- Click **Finish**.

## Configuring Panels

You can configure a panel to have multiple columns in a form. To configure a panel for multiple columns:

1. In the **Designer**, select a panel within the form.
2. In the **Properties** pane, in the **Columns** property, select **1 Column** to make the panel have one column, select **2 Columns** to make the panel have two columns, or select **3 Columns** to make the panel have three columns. Other controls can be placed inside of a column cell.

## Configuring Field Validation

You can configure validation of a field by selecting the field and clicking on the ellipsis button next to the **Data Type** property in the **Properties** pane. Each data type has different validation options.


All data types can be set to be required by selecting the **Required** check box.

## Currency Fields

**4**

### Validation

Choose Validation Options for Keyword.



Currency

Currency Format

Required

Currency fields pull the currency format directly from the configuration of the Keyword Type in the Configuration module. A currency format must be assigned to a field. When a certain format is not specified at the Keyword Type level, a **Currency Format** must be selected from the drop-down select list to associate a currency format to the field. The currency formats available in the drop-down select list are the currency formats configured in the Configuration module. Alternately, the **Currency Format** property can be configured in the **Properties** pane to define the appropriate currency format.

---

**Caution:** When using the **Create Autofill record on create** option, if a currency Keyword Type is used, the currency format assigned to the form field must exactly match the currency format configured at the Keyword Type level in order for the currency value to be a valid value for future use in Unity Forms. This means that if a Keyword Type is configured to use **Workstation Regional Settings** and a field is set to use a certain currency format, the current value stored will not be a valid value.

---

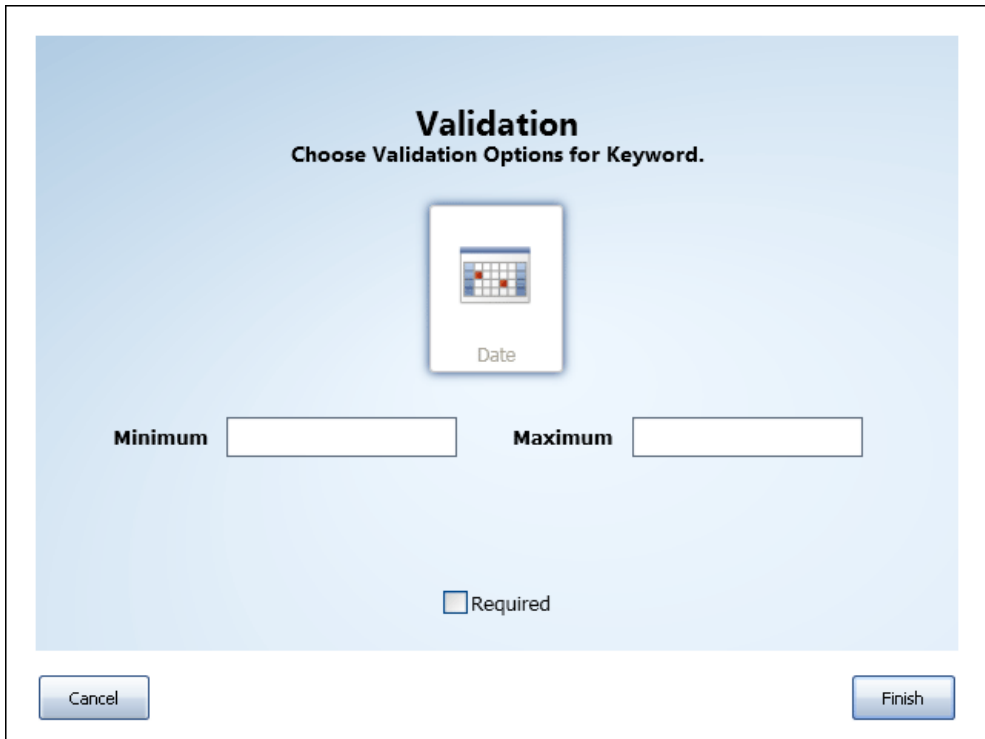
4

---

**Caution:** Specific currency formatted Keyword Types are not supported in Unity Forms.

---

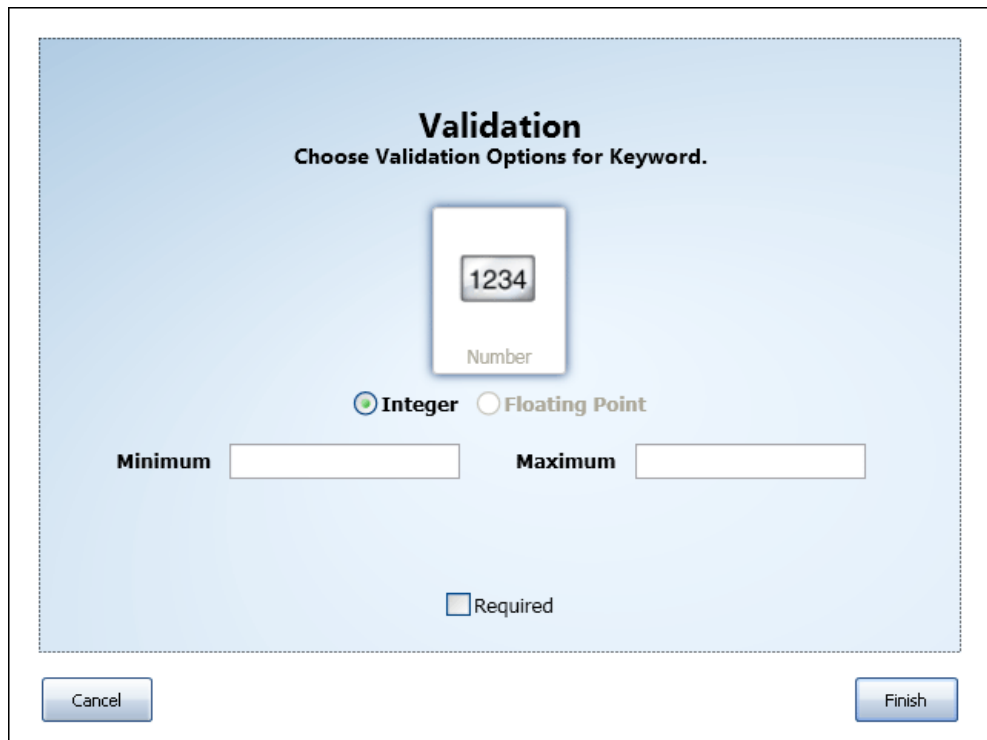
## Date Fields



The screenshot shows a dialog box titled "Validation" with the subtitle "Choose Validation Options for Keyword." The dialog has a light blue background. In the center, there is a white box containing a calendar icon and the word "Date". Below this, there are two input fields: "Minimum" on the left and "Maximum" on the right. At the bottom center, there is a checkbox labeled "Required". At the bottom left, there is a "Cancel" button, and at the bottom right, there is a "Finish" button.

Date fields can be configured to only accept a date from within a configured date range. In the **Minimum** field, enter the earliest day that can be entered in the field in the Client. In the **Maximum** field, enter the latest date that can be entered in the field in the Client. If a date does not fall between the **Minimum** and **Maximum** date values, the date cannot be entered into the field in the Client. Blank fields are still valid.

## Numeric Fields



The image shows a 'Validation' dialog box with a light blue background. At the top, it says 'Validation' and 'Choose Validation Options for Keyword.' Below this is a preview of a text box containing the number '1234' with the label 'Number' underneath. There are two radio buttons: 'Integer' (which is selected) and 'Floating Point'. Below these are two empty text boxes labeled 'Minimum' and 'Maximum'. At the bottom, there is a checkbox labeled 'Required'. At the very bottom of the dialog are two buttons: 'Cancel' on the left and 'Finish' on the right.

4

For a numeric field, the **Integer** or **Floating Point** radio button indicates the type of numeric field that is selected.

In the **Minimum** field, you can enter the minimum value that can be entered in the field in the Client.

In the **Maximum** field, you can enter the maximum value that can be entered in the field in the Client.

---

**Note:** Decimals can only be entered when using **Floating Point**. Floating point **Minimum** and **Maximum** values can have up to 6 decimal places.

---

## Alphanumeric Fields

4

When configuring alphanumeric Keyword Type fields, you can specify the maximum number of characters for a field in the **Max Length** field.

Masking can be configured in the **Mask** field. Masking can be toggled using the **Static Mode/ Mask Mode** buttons. In addition, the entry mode can be changed by pressing **Alt + s** on the keyboard. In order to successfully configure masking for a field, you must toggle back and forth between Mask Mode and Static Mode to enter all of the necessary character placeholders in the proper format.

In Mask Mode, characters to represent the type of data that is allowed to be entered in the Client. Character placeholders that are entered in Mask Mode are black during configuration. The following key character are used for masking in Mask Mode:

Key Character	Usage
A	The expected character is alphabetic.



Key Character	Usage
<b>X</b>	The expected character is alphanumeric.
<b>0</b>	The expected character is numeric.
<b>9</b>	The expected character is numeric with a possibility of a special symbol (+, -, ., *)

In Static Mode, enter specific characters that will be used as a mask in the Client. The characters entered will be the exact characters displayed in the Client. Characters entered in Static Mode will display in blue during configuration.

**Note:** Masking formats should match the mask format configured for the Keyword Type in the Configuration module.

4

## Configuring Values for Check Box Fields

A value can be associated with a check box. When the check box is checked, the value configured for the field is associated with the check box on the form. If the check box is not checked, no value is associated with the field on the form.

Once a check box has been placed on a form, you can configure the value associated with it. To configure a value for a check box:

1. Select the check box on the form.
2. In the **Properties** box, enter the value you want to associate with the check box when the check box is checked in the **Checked Value** field.
3. Click **Ok**.

## Setting Default Values for Check Box Fields

You can configure the default value that a check box field has upon creating a form. To configure a default value for a check box:

1. Select the check box on the form.
2. In the **Properties** box, enter the value of **Check Value** property in the **Default Value** property field if you want the check box to be selected by default. Entering any other value will leave the check box unselected by default.

## Creating Data Set Fields

When a field is created on a form that is linked to a Keyword Type configured to use a Data Set, the Data Set is automatically associated with the field in the Designer and the Data Set is available in the Client to use for value selection. The following is an example of a Keyword Type field associated with a data set in the Designer:



4 Additionally, if a Keyword Type with a Data Set is configured with the **Keyword Must Exist** option in the Configuration module, by default the **Value must exist property** is selected, and the value must exist in the Data Set associated with the Keyword Type in order to save data in the field. If the **Value must exist** option is unchecked, the value does not have to exist in the Data Set in order to be saved. This will override the **Keyword Must Exist** setting in the Configuration module.

## Configuring Self Configured Data Sets

When there is a select list field or a Keyword Type field configured with a Data Set, a self configured Data Set can be configured for a field. A self configured Data Set is a list of Data Set values that can be created and associated with a field on an ad hoc basis. Self configured Data Sets can be used by multiple fields in multiple forms.

To configure a self configured Data Set for a field:

1. Select a field in the Designer that has a drop-down control.
2. In the **Properties** pane, click the ellipsis button next to the **Data Set** property field.
3. Select either select an existing Data Set or a new Data Set.

If you select **Existing**, select the Data Set from the box that you want to associate with the field.

If you select **New**, enter a name for the Data Set in the field.

4. If you selected **Existing**, click **Finish**. If you selected **New**, click **Continue**.

5. Enter a value in the **New Dataset value** field. Click the **Add** button. Repeat this step for each value you want to enter.

You can arrange the values by selecting a value and using the **Up** and **Down** buttons to move the value to the desired position in the list of values. You can also sort values in the list by clicking **Sort Ascending** or **Sort Descending** buttons. You can also delete a value from the list by selecting the value and clicking **Remove**.

6. Click **Finish**.

If you have a Keyword Type field associated with a Keyword Type that has a Data Set configured for it in the Configuration module, any self configured Data Set that is configured in the Unity Form Designer for a field will override the Keyword Type's Data Set for that specific field.

You can specify that the value has to exist in the configured Data Set in order to submit the form by selecting the **Value must exist** property.

### **Field Requirements for Self Configured Data Sets**

In order to configure a self configured Data Set for a field, one of the following requirements must be met:

- The field was initially created using the **Select List** control.
- The field is associated with a Keyword Type that has a Data Set configured for it in the Configuration module.
- The field is associated with a Keyword Type that has a Data Set configured for it in the Configuration module within a Multi-Instance Keyword Type Group.

## **Moving Form Controls**

To move a control within a form in the Designer:

1. Hover over the control that you want to move until it is highlighted in yellow.
2. Click the Move icon on the right-hand side of the control and, with it clicked, drag the control to the desired location in the form.



The location where the control will be moved is denoted by a yellow area surrounded by a dashed line border. Release the mouse click when the control is in the correct location. The control will be moved to the new location.

To move a control within a table, you must perform the move action at the header of a column. The Move icon displays next to the header. By clicking the Move icon and dragging the header to a new location, the whole column will move to the new location.

When working with tables and repeating sections, controls within cannot be moved outside of the repeating section or table they are in. They can only be moved within the repeating section or table.

## Deleting Form Controls

To delete a control from a form:

1. Hover over the control that you want to delete until it is highlighted in yellow.
2. Click the Delete icon on the right-hand side of the control.



3. A message asking **Are you sure you want to delete the selected item?** is displayed. Click **OK** to complete the deletion. Click **Cancel** to abort the deletion.

## CONFIGURING FORM SECURITY

Security can be assigned at the page or section level. To set security:

1. Select the form container, page container, or control in the Designer.
2. In the **Properties** pane, click the ellipsis button next to **User Groups**. The security window is displayed.



3. Select the user group you want to assign rights from the **Available User Groups** box. You can select multiple user groups at a time if you want to assign the same rights to multiple groups.
4. Click **Add Selected Items**. Alternately, you can click **Add All Items** to add all user groups at once.

---

**Note:** You can remove user groups by selecting the user groups in the **Selected User Groups** box and clicking **Remove Selected Items**. You can remove all user groups by clicking **Remove All Items**.

---

5. When finished configuring security, click **Finish**.
6. To enable security select the **Is Secured** check box in the **Properties** pane.

---

**Tip:** You can double-click on user groups to move them to and from the **Available User Groups** and **Selected User Groups** boxes.

---

---

**Note:** Configured form template security for sections and pages within the form is respected in Workflow, regardless of permissions and overrides granted at the Workflow level.

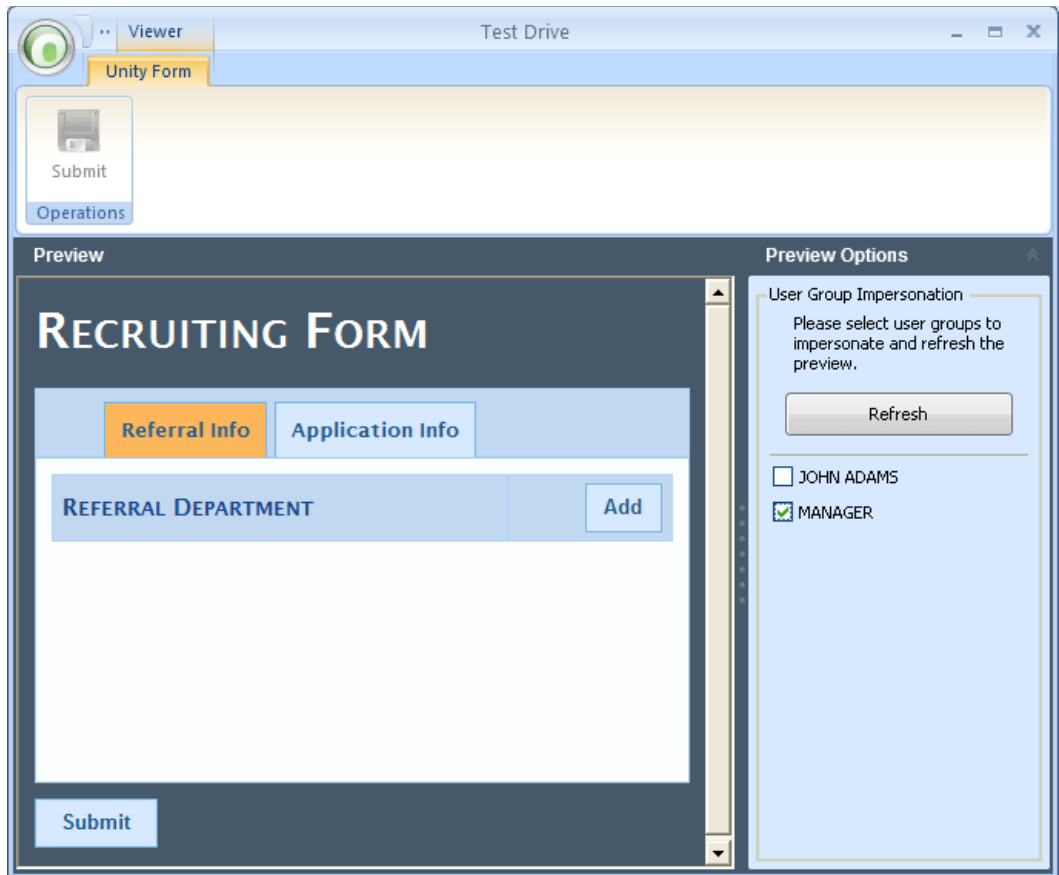
---

## 4

### TEST DRIVING FORMS

The Designer allows you the ability to “test drive” forms before publishing them for public use. This allows you to use the form the same way a user would. You can test out the functionality of fields and tabs. In addition, you can test the security applied to pages, sections, and fields. You can run test drives as specific user groups to assure that the form is configured in a way that makes sense to the user and provides the appropriate data security.

The following is an example of a test drive session run as the MANAGER user group:



4

You can test drive forms to view how they look and function on the Client side. To test drive a form:

1. With a form open in the Designer, on the **Designer** ribbon, click **Test Drive**.
2. A viewer opens with the form displayed as it will appear in the Client. You can select the user group you would like to impersonate to preview the form in the right pane of the viewer. Select the user group check box you would like to impersonate and click **Refresh**.

You can preview different user groups by selecting a different user group check box and clicking **Refresh**.

3. When you are finished test driving the form, exit the test drive window.

## SAVING A FORM TEMPLATE LOCALLY

You can save a form template locally when you are not ready to publish the form template in the OnBase system, but you want to save your progress. To save a form template locally:

1. When you are done working on the form template, click **Save** in the **Designer** ribbon. The **Save As** dialog box is displayed.
2. Browse to the directory in which you would like to save the form template.
3. Enter a **File name** for the template.
4. Click **Save**.

---

**Note:** If you have not saved or published a form template, a message stating **You have not saved changes made to this form. If you click 'Yes' your changes may be lost. Are you sure you want to navigate away from your form?** is displayed when you navigate away from the form template in the Designer. Click **Yes, Navigate Away** to discard changes. Click **Cancel** to return to the form template and save or publish the form template.

---

## OPENING A LOCALLY SAVED FORM TEMPLATE

You can open form templates that have been saved locally. To open a previously saved form template:

1. In the **Designer** ribbon, select **Open | Open from file**. The **Open** dialog box is displayed.
2. Browse to the form template that you want to open and select it.
3. Click **Open**.

Alternatively, you can select **Open | My Recently Save Forms | <form>** to access a recently saved form.



## PUBLISHING TEMPLATES

In order to make a form available in the Client for data entry, the form template must be published. Multiple form templates can be associated with a single Document Type. To publish a form template:

1. Once the form is configured appropriately, in the **Designer** ribbon, click **Publish**. A message stating **You are about to Publish the current electronic form. Publishing an electronic form saves the template into the repository and makes it available to users filling out forms. Any new forms of this type created will use the published template. Are you sure that you would like to Publish the form?**
2. Click **Publish** to continue with the publishing of the form. Click **Cancel** to abort publishing.
3. Upon clicking **Publish**, a status bar is displayed. When the status bar disappears, the form is published. In order to access a newly published form, the Client must be restarted.

---

**Note:** If you want to use a Unity Form on a web site, see Sharing Forms on page 60 for more information.

---

---

**Note:** If you have not saved or published a form template, a message stating **You have not saved changes made to this form. Would you like to discard all changes?** is displayed when you navigate away from the form template in the Designer. Click **OK** to discard changes. Click **Cancel** to return to the form template and save or publish the form template.

---

### Republishing Form Templates

You can also edit a previously published form and republish it. To access a previously published form:

1. In the **Designer** tab, select **Open | Open**. The **Open Published Template** dialog box is displayed.
2. Select the template you would like to access.
3. Click **Finish**. Alternately, you can double-click on the template name. Upon opening the form, you can edit it. In order to access a newly republished form, the Client must be restarted.

If you republish a form, a new revision of the form template is created. All new forms will use the latest revision of the form. Any forms that were created using an older revision of the format template will be displayed with the revision of the form template with which they were created.

## Sharing Forms

You can use Unity Forms in external web sites to allow form creation and submission external to the OnBase system. Published form templates can be shared to an external site. To share a form template:

1. Open a published form template.
2. Click **Share Form**. The **Share Form** dialog box is displayed.
3. Select the appropriate options for your form. See Shared Form Options on page 60 for more information.
4. Click **Copy**.
5. Paste the generated text in the appropriate place in relation to the external web site. You can use a generated URL to open the form directly or you can use the code generated to embed the form into a web site page, depending on the options you selected.

---

**Note:** Forms can be shared using any browser that the Web Client supports. In addition to these browsers, shared forms are supported in Safari using a Mac computer or an iPad.

---

### Shared Form Options

The following options are available when configuring a shared form.

#### SERVER INFORMATION

Option	Description
<b>Use SSL</b>	Select this option if you want the URL that is created to include SSL.
<b>Server</b>	Enter the host name and virtual directory associated with the OnBase Web Server that is associated with the form template.

Option	Description
<b>Datasource</b>	<p>The datasource entered in this field will be used by the Web Server to access the OnBase system during form creation. The datasource entered here will override the datasource configured as the Web Server's default datasource.</p> <hr/> <p><b>Note:</b> This field is not mandatory. If a value is not specified, the datasource configured in the Web Server's web.config file is used.</p> <hr/>

## AUTHENTICATION

If you want to specify a user account that will always be used to automatically log in to the OnBase system in order to submit forms from an external site, select **Used fixed credentials**. Selecting this option enables the **User name** and **Password** fields. Enter the **User name** and **Password** for the account you want to be used for external form submission.

## TARGET URL

In the **Target URL** field, enter the full URL of the site to which the user will be redirected after submitting the form.

## EMBED

If you want to embed the form into a Web site page, select **Embed**. You can then select the size of the embedded form. You can specify a custom size or select one of the following pre-defined sizes: 320x240, 640x480, 800x600, or 1024x768.

## Adding Keyword Type Groups to a Unity Form

When adding Multi-Instance Keyword Type Groups to a form during form design, only one instance of Multi-Instance Keyword Type Group can be placed in the same repeater section or table in the Designer. Additional instances of a Multi-Instance Keyword Type Group are added in the Client when data is entered into a form.

When adding Keyword Types that belong to a Single Instance Keyword Type Group to a form in the Designer, only one instance of these type of Keyword Types can be added to a form.

## Headers and Footers

When using a Standard or Multi-Page template layout, header and footer controls are inserted by default. Headers and footers can only contain Paragraph and Submit Button controls.

## Form Visibility

Once a form template has been published, you can set form visibility to further constrain which user groups have rights to create a specific form. In order for a user group to have rights to a form for creation in the Client, the user group must have rights to the Document Type to which the form template belongs and rights to the form template itself. By default, all published forms are available to all user groups that have rights to the Document Type.

To set form visibility:

1. When the appropriate published form template is open, click **Form Visibility**.
2. Uncheck the **Display for creation for all user groups** check box.
3. Select the user group(s) you would like to grant access to the form.
4. Click **Add**.
5. Click **Save**.

## Disabling Specific E-Forms

In some cases you may need to disable an E-Form so users cannot create new E-Forms of a specific type. One of these situations is when you convert an existing E-Form to a Unity Form. Once a Unity Form is configured to replace an E-Form, you would want to disable the E-Form equivalent. To disable an E-Form:

1. In the Configuration module, select **Document | Document Types**. The **Document Types** dialog box displays.
2. Select the Document Type that is associated with the E-Form you want to disable.
3. Click **E-Form**.
4. Select the **Don't allow form creation** option. Selecting this option disables all revisions for an E-Form; you do not have to apply this option for each revision.
5. Click **Apply**.
6. Click **Close**.

## Designing Forms for Multiple Languages

When designing form templates in the Designer, the language used during configuration is the language that will be displayed in the Client during form creation. For example, if you configure a form in French in the Designer, when you open the form on any machine, regardless of locale settings, the form will be displayed in French.

---

**Note:** The database collation must support the language set on the workstation designing the Unity Forms.

---

## LIMITATIONS

The following are limitations of Unity Forms:

- WorkView objects cannot be created from a Unity Form in Workflow.
- Unity Forms are not supported in the OnBase Client.
- When sending Unity Forms using the Distribution Service, section and page security configured on a Unity Form is not respected. The entire form is sent regardless of the security settings on the form.
- When sending a Unity Form using the Distribution Service or via the right-click **Send to** option, page tabs are not displayed in the form. The contents of the form is displayed laterally.
- Using the briefcase with Unity Forms is not supported.
- Printing Unity Forms from a remote/network print queue is not supported. (These are queues configured as Network Print Queue or Print Cluster Queue in the Configuration module.) Because the HTML Web Client uses remote printing queues only, printing Unity Forms is not supported in the HTML Web Client.
- When using the ActiveX-based Web Client, any text located on the header of a Unity Form will not show when printed.
- When using the Unity Client, if a Unity Form's information exceeds an 8.5" x 11" space, the information outside of this page size is not included on the printed page.
- The following Keyword Type configurations are not supported on Unity Forms: hidden, security masking, encrypted, Date & Time data type, and Specific Currency data type.

## KEYBOARD SHORTCUTS

The following keyboard shortcuts are available in the Designer.

Shortcut	Description
<b>Enter</b>	When a dialog box has an <b>OK</b> button, pressing the <b>Enter</b> key will function the same as the <b>OK</b> button. When in a wizard panel, pressing the <b>Enter</b> will function the same as a <b>Next</b> or <b>Finish</b> button.
<b>Esc</b>	Pressing <b>Esc</b> will function the same as clicking a <b>Cancel</b> button.

## 4

## SYSTEM INTERACTION

### Application Enabler

The **Unity Client - Unity Form Creation** context in Application Enabler can be used to create a new Unity Form from scraped values in a line-of-business application when using the Unity Client.

# INSTALLATION

---

## REQUIREMENTS

The browser used to both configure Unity Form templates in the Designer and create and access Unity Forms in the Client must support Javascript.

### Unity Forms Supported Operating Systems

- Windows XP SP3 or later service pack
- Windows Server 2003 SP2 or later service pack
- Windows Vista SP1 or later service pack
- Windows Server 2008 SP1 or later service pack
- Windows Server 2008 R2 RTM or later service pack
- Windows 7 RTM or later service pack

### Unity Hardware and Browser Requirements

Component	Minimum	Recommended
CPU	1.6 GHz dual-core	2.4 GHz dual-core
Memory (RAM)	1 GB	2 GB
Free Hard Disk Space	200 MB	
Screen Resolution	1024 x 768 (1280 x 800) <b>Note:</b> Using a lower resolution may result in a loss of functionality.	1280 x 1024 (1440 x 900 widescreen)
Graphics Card	128 MB	256 MB with DirectX 9.0c support

Component	Minimum	Recommended
<b>Web Browser</b>	Microsoft Internet Explorer 7.0, 8.0, or 9.0 <hr/> <b>Note:</b> Internet Explorer 9 is not supported on all operating systems supported by OnBase. For information on the operating systems supported by Internet Explorer 9, see <a href="http://support.microsoft.com/kb/2409098">http://support.microsoft.com/kb/2409098</a> . <hr/>	
<b>E-mail Platform</b>	Lotus Notes 7, 8, or 8.5 Microsoft Outlook 2003, 2007, or 2010	

**Note:** The OnBase System Assessment Tool can be used to ensure that your workstation meets the minimum system requirements for the 11.0.0 OnBase Unity Forms. For more information on this tool, see the Core Enterprise Installers documentation.

5

## .NET Framework

The Deployment Wizard requires Microsoft® .NET Framework 4.0 (Full Profile/Extended, which includes the Client Profile) to be installed on the deployment machine.

The installation of deployed applications on client machines may have a different requirement for client machines. See below to determine the specific .NET requirement for client machines.

The required version of the .NET Framework must be installed before launching the installer on either machine. The .NET Framework can be obtained from the Microsoft Download Center at <http://www.microsoft.com/downloads>.

### .NET Framework on Client Machines

Unity Forms requires Microsoft® .NET Framework 4.0 (Client Profile), which can be obtained from the Microsoft Download Center at <http://www.microsoft.com/downloads>.



## Databases Supported

The following tables list the databases supported in OnBase 11.0.0.

### Microsoft SQL Server

Microsoft SQL Server	Additional Information
<p><b>Microsoft SQL Server™ 2000 (SP4 recommended)</b></p> <p><b>Microsoft SQL Server 2005 (SP2 or later recommended)</b></p> <p><b>Microsoft SQL Server 2008 (RTM, SP1, SP2; SP2 recommended)</b></p> <p><b>Microsoft SQL Server 2008 R2 (RTM, SP1; SP1 recommended)</b></p>	<p>Microsoft SQL Server 2000 and 2005 must be running in compatibility mode 7 or greater. Running in compatibility mode 6.5 or lower will result in errors during the upgrade process.</p> <p>SQL Server 2005 drivers must be upgraded to the Feature Pack for Microsoft SQL Server 2005 - December 2008 or a later feature pack.</p> <hr/> <p><b>Note:</b> On January 11, 2011 Microsoft discontinued technical support for Microsoft SQL Server 7.0. As of release 11.0.0, Hyland Software no longer supports SQL Server 7.0.</p> <hr/> <p><b>Note:</b> You must ensure that your SQL Server database client software version matches or exceeds the database server version. For example, if your database server is SQL Server 2005, verify that the database client is SQL Server 2005 (or later). Running a previous client version, such as SQL Server 2000, will result in system instability and memory issues. For instructions on determining your server and client versions, see Database Client / Server Version Compatibility on page 68.</p>

5

### Oracle

**Note:** If you are using an Oracle database, it is strongly recommended that you have a certified Oracle Database Administrator on staff.

Oracle	Additional Information
<p><b>Oracle v 8.0.5.0 or later</b></p>	<p>Oracle version 8.0.5.0 can be used, but it is not recommended due to potential memory leaks. If Oracle 8.0.5.0 is used, a third-party ODBC driver is recommended.</p>
<p><b>Oracle 8i: 8.1.7.7 or later</b></p>	<p>ODBC drivers should be 8.1.7 or later. 8.1.6.x drivers have known issues and are not supported.</p>

Oracle	Additional Information
<p><b>Oracle 9i: Release 1 and Release 2 (9.2)</b></p> <p><b>Oracle 10g: Release 1 and Release 2</b></p>	<p>Oracle driver version 10.2.0.3 is recommended.</p> <p>An issue has been observed with the Oracle ODBC drivers where Unicode characters (e.g., Japanese characters) retrieved from a CLOB data type will be truncated, and could potentially cause errors in OnBase. The data remains intact in the database; however the results when retrieving the data will be truncated. This has specifically been observed in two areas of OnBase that use the CLOB data type to store large amounts of data: VBScripts and License Certificates. This behavior may apply to other areas of the software that use this data type as well.</p> <p>To ensure that Unicode characters retrieved from a CLOB data type are not truncated, the Oracle 10g R2 ODBC drivers (which are backward compatible) should be installed, as well as the latest patchset (version 10.2.0.3) for these drivers.</p>
<p><b>Oracle 11g: Release 1 and Release 2</b></p>	<p>All Oracle 11g drivers can be used.</p>

**Sybase SQL Anywhere**

5

Sybase SQL Anywhere	Additional Information
<p><b>Sybase SQL Anywhere™ 5.5.04, 7.0.0-7.0.3, 8.0.1, 9.0, 10 and 11</b></p>	<p>No 6.0.X releases.</p> <hr/> <p><b>Note:</b>As of release 8.2.0, Sybase Adaptive Server Enterprise (ASE) is no longer supported.</p> <hr/>

**Database Client / Server Version Compatibility**

Due to critical issues that have been reported to Hyland Software, Hyland Software strongly recommends that:

- your database client software version matches or exceeds the database server version and
- you are running the most recent version of the database client.

This will help to reduce compatibility issues and minimize troubleshooting time when issues do occur.

Your database administrator can determine the database server version and identify the most-recent version of the database client software. The ODBC driver number indicates which version of the database client software you are using. For example, if your database server software is Oracle 10 Release 2, verify that the Oracle Client software is Oracle 10 Release 2 (or later). The same is true of SQL databases. For example, if your database server is SQL Server 2005, verify that the database client is SQL Server 2005 (or later).

To check your database client version, perform the following steps from the workstation or server where the ODBC connection is configured:

1. Open your ODBC Data Source Administrator, and click on the **Drivers** tab.
2. Select the driver you are using to connect to your OnBase database.
  - If your database server software is Oracle 10 Release 2, the version number should appear as **10.2.[#.#.#]** (or later), where **10.2** is the version number and **[#.#.#]** represents the service pack.
  - If your database server software is SQL Server 2005, the version number should appear as **2005.[##.####.##]** (or later), where **2005** is the version number and **[##.####.##]** represents the service pack.

The above descriptions are examples of two commonly used database version numbering schemes. Ensure that the supported database you use adheres to the database client/server recommendation.

## Hyland Software - MS Service Pack Statement

Hyland Software is dedicated to ensuring the monthly cumulative updates released by Microsoft® are compatible with OnBase®. On the second Tuesday of each month, the Quality Assurance Department of Hyland Software evaluates the cumulative fixes released and labeled as Critical or Important by Microsoft®. The details of the update provided by Microsoft are reviewed for interaction with OnBase® and installed when appropriate for testing its compatibility with OnBase®. If you have questions regarding a specific Microsoft® cumulative update and its compatibility with OnBase®, please contact our Technical Support Team at 440-788-5600.

## About Virtual Environments

Hyland Software develops, tests, and supports the OnBase suite of products on specific Operating Systems, not specific hardware configurations. When OnBase is operated in a virtual environment (such as Citrix, VMware, Hyper-V, or Windows Remote Desktop) there may be limitations or subtle differences imposed by the environment. The customer and the virtual environment vendor are responsible for any interactions or issues that arise at the Hardware or Operating System layer as a result of their use of a virtual environment.

When it appears that an OnBase performance-related issue is either caused by (or is unique to) the virtual environment, organizations may be asked to validate that the issue occurs in a non-virtual environment. Hyland Software will make this request if there is reason to believe that the virtual environment is a contributing factor to the issue.

Each OnBase site is unique. Hyland Software depends on the customers who deploy OnBase in virtual environments to do so only after careful design and adequate planning (that takes into account the workloads of your organization), and in accordance with recommendations provided by the virtual environment's vendor. As with any implementation, Hyland Software strongly recommends that any customer deploying an OnBase solution in a virtual environment thoroughly test the solution before putting it into production.

For information about using OnBase in a Citrix and Microsoft Windows Remote Desktop environment, please see the **Citrix and Microsoft Windows Remote Desktop Environment Deployment Guide**, available from your solution provider.

**5**

## 64-Bit Support Statement

The OnBase suite of products is tested on 64-bit systems and is capable of being deployed on 64-bit systems using the Windows 32-bit on Windows 64-bit Emulator (WOW64) layer. However, OnBase modules that integrate with third-party applications may not be able to be used with the 64-bit versions of these applications. For these modules, only the 32-bit versions of these third-party applications are currently supported by the OnBase integrations. Consult the module-specific requirements section in each module reference guide for complete requirements details.

Supported database versions that are deployed on a 64-bit database server are also supported. For more information, contact your solution provider.

## Windows User Account Control Statement

Hyland Software is dedicated to ensuring that OnBase is compatible with Windows User Account Control (UAC). UAC is a feature of Windows operating systems that was introduced with Windows Vista. It limits the ability of standard users to make global system changes to a workstation and prevents malicious software from making unauthorized changes to protected areas.

For details on UAC, refer to your Microsoft support information or see [http://technet.microsoft.com/en-us/library/cc709691\(WS.10\).aspx](http://technet.microsoft.com/en-us/library/cc709691(WS.10).aspx).

You may encounter UAC in OnBase when:

- Installing or uninstalling OnBase, an OnBase module, or OnBase ActiveX controls.
- Copying, moving, or saving files to the Program Files directory, Windows directory, or another protected location.
- Modifying system-wide settings, such as the registry.

If Windows UAC is enabled, the above operations will prompt for administrator privileges, even if an administrator is currently logged on.

### **Modifying Configuration Files**

When UAC is enabled, administrators may be unable to modify Web.config or other .config files. To address this issue, the administrator should open a text editor (such as Notepad) by right-clicking it and selecting **Run as administrator**. The administrator can then open the .config file from within the text editor. Because the text editor is running with administrator privileges, the configuration file can be modified and saved using that application.

## **INSTALLATION**

The Unity Forms Designer is installed with the Unity Client. There are no additional installation steps necessary to install Unity Forms. See the Unity Client Module Reference Guide for information on installation. See User Rights on page 19 for information about granting rights to users to allow them to access the Unity Forms Designer.

## **CONTACTING SUPPORT**

When contacting your solution provider, please provide the following information:

- The OnBase module where the issue was encountered.
- The OnBase version and build (Example: 11.0.0.571) and/or the Core Services version and build (Example: 11.0.0.6).
- The type and version of the connected database, such as Microsoft SQL Server 2008 or Oracle 11g, and any Service Packs that have been installed.

- The operating system that the workstation is running on, such as Windows XP or Windows Server 2008, and any Service Packs that have been installed. Check the supported operating systems for this module to ensure that the operating system is supported.
- The name and version of any application related to the issue.
- The version of Internet Explorer, and any Service Packs that have been installed, if applicable.
- A complete description of the problem, including actions leading up to the issue.
- Screenshots of any error messages.

Supplied with the above information, your solution provider can better assist you in correcting the issue.