



Tessitura Custom
Screens
v15.0+

Copyright 1999-2020 by Impresario L.L.C.
(as an unpublished work subject to limited distribution and restricted disclosure only).
All Rights Reserved

NOTICES

The material contained in this manual is protected by the copyright laws of the United States of America and by international treaty as an unpublished work. Unauthorized use, reproduction or distribution of the material contained in this manual is punishable by civil and criminal penalty.

This material contains proprietary, confidential, and trade secret information of Impresario L.L.C. and is provided solely for use by licensees of the **Tessitura Software**[®] program and their authorized employees. Except as expressly authorized by Impresario L.L.C. pursuant to the terms of a written license agreement, no part may be used, disclosed, distributed, reproduced, transcribed, stored in a retrieval system, adapted or transmitted in any form by any means, electronic, mechanical, magnetic, optical, chemical, manual or otherwise.

Tessitura Software is a registered trademark of Impresario L.L.C.

Brand or product names mentioned in these materials are the trademarks or registered trademarks of their respective companies.

THE MATERIAL CONTAINED IN THIS MANUAL IS FURNISHED "AS IS" WITHOUT WARRANTY OF ANY KIND AND ALL WARRANTIES EXPRESS OR IMPLIED, ARE HEREBY EXCLUDED (INCLUDING, WITHOUT LIMITATION, ANY CONDITIONS OR WARRANTIES RELATING TO MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE). IMPRESARIO L.L.C. SHALL NOT BE LIABLE FOR ANY TECHNICAL OR EDITORIAL INACCURACIES OR OMISSIONS MADE HEREIN

The material contained in this manual is subject to change without notice. Impresario L.L.C. assumes no obligation to keep customers informed of any inaccuracies, updates, additions, deletions or other changes or modifications to any of the material contained in this manual.

IMPRESARIO L.L.C. SHALL NOT BE LIABLE FOR ANY SPECIAL, INCIDENTAL, INDIRECT OR CONSEQUENTIAL, PUNITIVE OR EXEMPLARY DAMAGES OR FOR LOST PROFIT, REVENUE, USE OR DATA IN CONNECTION WITH THE MATERIAL CONTAINED IN THIS MANUAL, EVEN IF ADVISED OF THE POSSIBILITY OF SUCH LOSS.

Licensees are solely responsible for their use of this manual and the software to which it relates in full compliance with all applicable laws, rules and regulations and Impresario L.L.C. assumes no responsibility for informing licensees of any applicable laws, rules or regulations or any changes thereto that might occur from time to time.

Table of Contents

INTRODUCTION	1
INFOMAKER DATAWINDOW GUIDELINES	2
BROWSER-BASED CUSTOM SCREEN GUIDELINES	7
TR_CUSTOM_TAB	8
CUSTOM SCREEN SECURITY	10
SAMPLE CUSTOM SCREENS	11

Introduction

Custom screens can be added to the following areas of the Tessitura application:

- Constituent Record
- Ticket History (Constituent window)
- Subscription History (Constituent window)
- Production Elements (Title, Production, Production Season, Performance detail windows)
- Package Setup
- Package Product Guide
- Performance Product Guide
- Campaigns
- Elevated Events
- Appeals
- Tools menu (browser-based screens only; added v15.1.14)

Custom screens can be built as InfoMaker datawindows or browser-based pages accessible through a URL. This document outlines methods and techniques for incorporating custom screens into the application.

Note: V15.1 of Tessitura supports InfoMaker 2017 R3, and we have found that version of InfoMaker to be compatible with v15.0, v14.1 and v14.0 as well. InfoMaker libraries and datawindows created in prior versions of InfoMaker must be migrated to this version to function correctly in Tessitura.

InfoMaker Datawindow Guidelines

The following guidelines apply to custom screens designed as InfoMaker datawindows:

- You can use either a SQL statement in the datawindow or a stored procedure to return data for the screen. Data can come from multiple tables, but if the datawindow is to be editable, all the editable columns must be in a single table.
- The data source (either SQL statement or stored procedure) must have one and only one retrieval argument, representing the id of the record for which you are retrieving data. For example, the constituent ID (customer_no) for constituent record custom screens, the inventory number for production element custom screens, or the package number for package custom screens.

Note: Inventory number is a generic way to refer to production element IDs, derived from T_INVENTORY, the table in which all production elements are stored. The inventory number will match the title_no, prod_no, prod_season_no or perf_no depending upon the type of production element.

- Each custom datawindow must have a column called **Id_Key (long)**, which represents the primary key of the table the datawindow is set to update (if the datawindow is set to make updates), and a column for the primary key of the record you are retrieving. The primary key column must be named based on the record type as specified in the following table:

Custom Screen Area	Primary Key Column
Appeals	appeal_no (long)
Campaign	campaign_no (long)
Campaign Events	campaign_no (long)
Constituent	customer_no (long)
Package Information	pkg_no (long)
Package Setup	pkg_no (long)
PE Performance	inv_no (long)
PE Production	inv_no (long)

Custom Screen Area	Primary Key Column
PE Production Season	inv_no (long)
PE Title	inv_no (long)
Performance Information	inv_no (long)
Ticket History	customer_no (long)

If the datawindow is not going to be editable, these columns must still exist in the datawindow definition, but the data in them is not relevant. For instance, if you want to display data from a table where the customer_no is in a column called cust_id and the, and the table has no primary key of its own or doesn't need to be updated you could use the SQL statement

```
Select customer_no = cust_id, id_key = 0, From yourtable Where cust_id = :a_retrieve.
```

Alternately, create a view of your table in which customer_no and id_key are returned.

- The Id_Key and primary key columns can never be edited. These columns do not have be visible on the data object, they simply must be in the dataset.
- If the datawindow is editable, the Id_Key column must represent the primary key in the editable table.
- If you have other columns that should not be edited, place the word NOEDIT in the tag property of the column object.
- If you have other columns that for some reason should be editable but not updated in the database, place the word NOUPDATE in the tag property of the column object.
- Default values for editable columns may also be placed in the tag property. For example if you have a checkbox column for which the default value is 'N', place 'N' (including single quotes) in the tag property. A date column can default to the current date by setting the tag to 'today' (single quotes included).
- If a custom screen is designated as editable then all columns other than Customer_No, Id_Key and those with NOEDIT in the tag property will be editable. Make sure that any columns not from the Update Table are marked NOEDIT.
- The tab order of the editable columns will be set to the order of the columns in the data set. If you want to arrange the tab order, you must also change the retrieval order.

- If you want a calendar chicklet next to a date column, place a text object with **no text**, a **3D Raised Border**, and **color ButtonFace** in the detail band next to the data column. Call the text object **CAL_colname**. For instance if the date column is called vol_date, then the button should be called CAL_vol_date. If you want the calendar button to only be visible in the row currently being edited, then place this expression in the Visible Property Expression for the text object:

```
If(CurrentRow = GetRow(), 1, 0)
```

- If you want a constituent search chicklet next to a constituent id column, then place a text object with **no text**, a **3D Raised Border**, and **color ButtonFace** in the detail band next to the data column. Call the text **object IDNO_colname**. For instance if the data column is called teacher_no, then the button should be called IDNO_teacher_no. If you want the chicklet to only be visible in the row currently being edited, then place this expression in the Visible Property Expression for the text object:

```
If(CurrentRow() = GetRow(), 1, 0)
```

- If you want the columns to be sorted when you click on column headers like other columns in Tessitura, then place text objects either in the Header band or in the Detail band for each sortable column and name them **CB_col#**. For instance the name of the Text Object in the Header band for the 3rd column should be CB_3.
- You can use your own colors or let Tessitura set the colors to match other windows with a setting in [TR_CUSTOM_TAB](#).
- You can use dropdown datawindows on your datawindow fields. If you want a dropdown to act like a Tessitura dropdown (inactive rows grayed out and sorted to the bottom), make sure that the dropdown datawindow has a column called "inactive", which contains the value from the system table. If you want a dropdown to act like a Tessitura dropdown but still allow inactive rows to be selected, name the dropdown datawindow column "inactive_select".
- It is possible to have dropdowns in custom screens dependent on the value in another column in the screen. For an example of this functionality, see the **samplecustomscreens.pbl** to see the datawindow, **ld_program** and the dropdown datawindow, **ld_donation_level**. Note the following (referencing the samples):
 1. The column **tr_donation_level_desc** is a text column that retrieves the description value from the database. The visible property of the column is such that it is only visible on rows that are not the current row. It has a tag property of NOUPDATE, because it is not saved to the database.

2. The `tr_donation_level_desc` column is positioned at exactly the same position as the dependent dropdown column, `donation_level`. The dropdown column `donation_level` has a visible property that makes it visible only on the current row.
 3. The tag value for the dependent dropdown column `donation_level` is `{tr_donation_level_description, program_no}`. This array contains the following two values: the name of the column that is positioned over the dependent dropdown column; and the name of the column whose value is the retrieval argument for this dropdown.
 4. The data column name must refer to a numeric column and the display column name must refer to a string column.
 5. The dropdown datawindow `id_donation_level` has a numeric retrieval argument.
 6. The `donation_level` column has the `AutoRetrieve` checkbox unchecked on the Edit tab in InfoMaker. This prevents the dropdown from populating automatically.
- It is possible to configure columns (or computed fields) that hold a `customer_no`, `order_no`, or contribution `ref_no` in custom screens to open a corresponding record when double-clicked. To enable this feature, add one of the following to the columns tag property: `ID_CONST(n)`, `ID_ORDER(n)`, or `ID_CONTR(n)` where `n` represents the name of a column in the datawindow. If more than one value must be placed in a tag (such as `ID_ORDER(n)` and `NOEDIT`), separate the values with a semicolon. If a column or computed field in a custom screen has one of these tag values, the cursor will change to a magnifying glass when it is over the column (unless the column is also an editable column). If the column or computed field contains a numeric value, double-clicking on the column will open the constituent window, the order details window or the contributions details window, depending on the value of the tag.

For example, if you want to click on a column called `ref_no` that contains a contribution reference number and open the contributions detail window, put the following in the column's tag: `ID_CONTR(ref_no)`. If you want to click on a column called `const_name` and open the constituent record, put the following in the column's tag: `ID_CONST(customer_no)` (assuming there is also a column in the datawindow called `customer_no`).

Note: Both the contributions and order details windows will open in view-only mode. Neither window will open if the current user does not have View rights on the appropriate security object (Const Contributions Tab for contribution details and Const Orders Tab for order details).

- Customer service issues (CSIs) can also be opened from custom screens. To enable this feature, place ID_CSI(xxx) in an object tag, where xxx is a valid CSI ID (activity_no). The window will only open if the user has View rights for the Const Cust Service RB security object. The issue can be edited if user has Edit rights for the Const Cust Service RB security object.
- Constituent record custom screens can use the Group by Constituent checkbox available on standard constituent screens for when data from primary affiliates is displayed. To use this checkbox, you must create two versions of the custom screen, one with the data grouped by constituent and a second with the data ungrouped. This functionality is not available for browser-based screens.
- The phrase 'AffiliateDropdown' can be put in the Tag property of a customer_no field. This will cause the customer_no field to display the name of the constituent instead of the number, providing the constituent is either the current constituent or an affiliate of the current constituent. In addition, if the field is in the detail band of the datawindow, for new rows, it will be a dropdown where an affiliate can be selected. The intent is to combine this ability with grouping/ungrouping and a select statement including V_CUSTOMER_WITH_PRIMARY_AFFILIATES.
- The word 'RESIZE' can be put in the Tag property of the rightmost header text element and any horizontal lines that are part of the design. If this is present, those elements will automatically extend across the remainder of the screen if the width of the display is greater than the width of the datawindow object.
- If you would like to add the red X icon as a delete button beside rows on the screen like on standard Tessitura screens, add a graphic element to your datawindow called p_delete. Tessitura will automatically assign the same red X icon as in the rest of the application to your graphic object and size it appropriately. If a graphic object with this name is included, Tessitura will also hide the delete button on that particular custom tab.
- To match the bolding of the current constituent's name when data is also listed for affiliated constituents, first add a computed column object to the datawindow called currentconstituentid. The expression doesn't matter as it will get replaced by the system (the sample screens just use 0 as the expression). Uncheck the Visible property of the computed column so that it doesn't show. Then, for the Bold property of the customer_no column, add the following expression: If(customer_no = currentconstituentid, 700, 400).

Save the datawindow in one of your Report Libraries (referenced in the **Local_Library_List** entry in T_DEFAULTS).

Browser-Based Custom Screen Guidelines

Browser-based custom screens are not editable except by custom code managed outside the Tessitura application.

A limited number of tokens whose actual values will be substituted at runtime can be passed in the query string for a browser-based screen:

Token	Translated Value
<<key>>	The current constituent ID.
<<ug>>	The user group of the current Tessitura user.
<<user>>	The current Tessitura user.
<<company>>	The company name from the Company Name entry in T_DEFAULTS.
<<login>>	The current SQL login.
<<url>>	The SSRS URL entry value from T_DEFAULTS.

TR_CUSTOM_TAB

The **TR_CUSTOM_TAB** table is used to configure custom screens for display in the Tessitura application.

Note: Tools menu custom screens were added in v15.1.14. They must be browser-based.

This table is supplied with data.

Id	Description	Rb No	D Object	Update Table	Standard Look	Area
17	Ticket History	1	ld_sample_performance_history_ungrouped		<input checked="" type="checkbox"/>	Ticket History
18	Subscription History	2	ld_sample_subscription_history_ungrouped		<input checked="" type="checkbox"/>	Ticket History
1	Volunteers	1	d_custom_sample.psr	lt_custom_example	<input checked="" type="checkbox"/>	Constituent

Entries in this table have the following columns:

Description – The name of the custom screen, which will appear alongside the radio button for the screen.

Rb No – The number of the radio button to which the custom screen is assigned. The Ticket History screen must use **1** and the Package History screen **2**. Constituent records can have an unlimited number of custom screens. For Production Elements, the Product Reference Guide, campaigns, elevated events, appeals, and the Tools menu the options are **1-6**. For packages the options are **1-5**.

D Object – The filename of the custom screen Infomaker datawindow or a valid URL if the screen is browser-based. The column limit is 255 characters for both object names and URLs.

Update Table - If the custom screen holds editable fields, then this column must contain the name of the table that will be updated and is used to build the SQL statements within the screen. The custom screens on the **Product Reference Guide** are for display only and cannot be updated. In addition, browser-based screens are not editable except by custom code outside the application.

Note: While the **Ticket History** and **Package History** screens are display only, you must enter the tables where the data for these screens are stored in order for the Ticket and Package History option on the Move Transactions to Household feature to function.

Standard Look – A checkbox that causes the screen to adapt the look and feel of other data objects in the Constituents module. Adapting the look and feel includes changing the background color of the screen and making the columns change color when a row is being edited. Adapting the look and feel does not change the font or size of any column nor does it change any objects that are not columns (like header text). This setting has no effect on browser-based screens.

Area - The area in the application where the screen should appear. The options are:

- **Constituent** - Constituent record Custom tab.
- **Package Information** - Package product reference guide custom tab.
- **Package Setup** - Package custom tab.
- **PE-Performance** - Performance custom tab.
- **PE-Production** - Production custom tab.
- **PE-Production Season** - Production season custom tab.
- **PE-Title** - Title custom tab.
- **Performance Information** - Performance product reference guide.
- **Ticket History** - Constituent record Ticket History tab.
- **Campaigns** - Campaign custom tab.
- **Campaign Events** - Elevated event custom tab.
- **Appeals** - Appeal custom tab.
- **Application** - The Tools menu.

Object ID - This column is used only for constituent and Tools menu custom screens, and it refers to an object_id value from T_APP_OBJECTS. IDs 81-86 correspond to the standard 6 custom screens, with 81 for radio button 1, 82 for radio button 2, and so on up to 86 for radio button 6. When adding additional constituent custom screens (beyond the original six) a row must first be added in T_APP_OBJECTS with an object_id between 1000 and 1099 which represents the security object for the custom screen. That ID should then also be placed in this column.

Display in Browser - A checkbox to indicate that the screen is browser-based and should run in an embedded browser.

Grouped D Object - The filename of the custom screen Infomaker datawindow when data is grouped by constituent. If this column is filled in, the Group by Constituent checkbox will appear on the custom screen and default to checked. If the Group by Constituent checkbox is unchecked, it will switch to the object specified in the D Object column. Browser-based custom screens will ignore this setting.

Use Ie - When checked, a browser-based custom screen will use the legacy IE browser control, instead of the CefSharp Chrome browser.

Custom Screen Security

Security for the custom screens is handled in the Tessitura Security application and is administered just like standard screens.

For the constituent record custom screens, there are security objects for the Constituents Custom Tab (Const Custom Tab) and for each of the Radio Buttons (Const Custom RB 1-6). Security objects for screens beyond the original six must be manually added to T_APP_OBJECTS.

Note: If a custom screen is not editable (i.e. no Update_Table is specified in TR_CUSTOM_TAB) but Add, Edit or Delete rights are granted for the screen, it will appear to the user as if they can make edits, but no changes will actually be saved.

For Tools menu custom screens, security objects must be added manually following the same process as adding objects for constituent record screens beyond the sixth screen.

Security for custom screens outside the constituent record are based on access to the screen or tab in which the custom screen appears. Security is granted to the entire tab (which contains up to six custom screens) and not to individual screens (radio buttons).

Adding Security Objects for Constituent and Tools Menu Custom Screens

Security objects for the custom constituent record screens beyond the original six and Tools menu custom screens must be manually added to T_APP_OBJECTS. Complete these entries as follows:

1. Enter an **Object ID** between 10000 and 10099.
2. Select an **Object Type** of Radio Button.
3. Enter a **Name** for the security object. This is what is displayed in the Security application when managing security rights.
4. Optionally, enter an optional **Description**, which can be displayed in the Security application when right clicking on the object.

Sample Custom Screens

The Tessitura installation and upgrade files include some sample InfoMaker custom screens. They can be found in: **LibrariesAndSamples\TessituraSampleLibrary\Custom Screens**. The folder includes one pbl, **TessituraSampleCustomScreens**, and a SQL script, **CustomSampleTableCreateScript**, which can be run to create the tables referenced by some of the samples.

The sample screens include:

- An example of 2 datawindows designed for use on a constituent custom screen that use the group by constituent functionality (Id_sample_custom_volunteer_grouped, Id_sample_custom_volunteer_ungrouped).
- An example of a screen to setup and display pictures on the Performance Reference Guide (Id_sample_custom_perf_pictures).
- An example of a screen to setup and display pictures on the Package Reference Guide (Id_sample_custom_pkg_pictures).