MediaBeacon 20.2

Copyright Notice

01 - 2021

ESKO

Contents

1. Copyright Notice.......... 4

2. Welcome.......... 6

3. WebUI Configuration.......... 7

3.1. Workspace Configuration.......... 7

3.1.1. Admin Mode.......... 8

3.1.2. Admin Only Menu Commands.......... 9

3.2. Manage Workspace Dialog.......... 10

3.2.1. Dialog Sections.......... 11

3.3. Using Workspaces.......... 19

Workspaces Overview.......... 19

Workspace Options Contextual Menu.......... 19

3.3.1. Adding Workspaces.......... 21

3.3.2. Sharing Workspaces and Duplicating Workspaces.......... 22

3.3.3. Deleting Workspaces.......... 27

3.3.4. Hide Workspace.......... 27

3.3.5. Arrangement of Workspaces.......... 27

3.3.6. Workspace Names.......... 27

3.4. Component Configuration.......... 28

Component List.......... 28

3.4.1. Using Components on Workspaces.......... 35

3.5. Administrator Menu Commands.......... 42

3.5.1. Dictionaries Dialog.......... 42

3.5.2. History Dialog.......... 49

3.5.3. R3Search Dialog.......... 49

3.5.4. System Status Dialog.......... 49

3.5.5. Workflow Manager Dialog.......... 49

3.6. Global Default Icon.......... 51

3.6.1. Global Icon Key.......... 51

3.7. Single Sign On.......... 53

4. Admin Core.......... 54

4.1. Admin Core Interface Overview.......... 55

Workspace Structure.......... 55

4.2. ACL Configuration.......... 56

Editing ACL Settings.......... 58

4.3. Changing Type Validator and UI Element Settings with Forms.......... 58

4.4. Field Type and UI Settings.......... 59

4.4.1. Type Validators.......... 59

4.4.2. UI Elements....64
4.4.3. List of Valid Combinations.... 64
4.5. List of Preconfigured Forms....65
4.6. Permission Restrictions List....67
4.7. Fields Workspace....70
4.7.1. XMP and Other Metadata Standards....70
4.7.2. Field Configuration Asset Metadata.... 71
4.7.3. Predefined XMP Fields....73
4.8. Groups Workspace....73
Groups Overview....73
4.8.1. Managing Groups with an IdP.... 75
4.9. Loading Dock Types Workspace.... 75
4.10. Forms Workspace....76
4.10.1. Form Configuration Asset Metadata.... 77
4.10.2. Create Custom Forms....80
4.11. Users Workspace.... 83
4.11.1. User Account....83
4.12. External Portals....87
4.12.1. Using an External Portal....90
4.12.2. Creating an External Portal....92
4.12.3. Customization Options....93
4.12.4. Managing External Portal....94
4.12.5. Reusing Portal....95
4.12.6. External Portal History....95

5. Contact Sheet Cover Page....97
Creating a Contact Sheet Cover Page.... 98
Configuring a Contact Sheet Cover Page....102
MediaBeacon 2018.X Configuration.... 104
5.1. Appendix.... 105
Text Field Properties in Acrobat Pro....105

6. MediaBeacon Port Information for Remote Components.... 110

1. Copyright Notice

Correspondence regarding this publication should be forwarded to:

Esko Software BV

Kortrijksesteenweg 1095

B – 9051 Gent

info.eur@esko.com

2. Welcome

Welcome to MediaBeacon's manual for administrative users of our product.

We are currently working on creating full documentation on three main sections of the Administrator Manual:

- **Admin Core**, configuration of enterprise-level options in a MediaBeacon instance.
- **Server GUI Application**, configuration of IT-level options in the MediaBeacon server software (in progress)
- **Web UI Configuration**, configuration of workspaces, components, and other parts of the MediaBeacon Web UI (In progress)

3. WebUI Configuration

This section covers administrator-level configuration options for workspaces and components.

3.1. Workspace Configuration

MediaBeacon's workspaces and components can be configured by Global Administrators in three ways, Admin Mode, Admin-Only menu commands, and Option & Configuration dialogs. The first two are discussed below, while dialogs are covered in context throughout Workspace Configuration subsections.

3.1.1. Admin Mode

Figure 1: Normal View

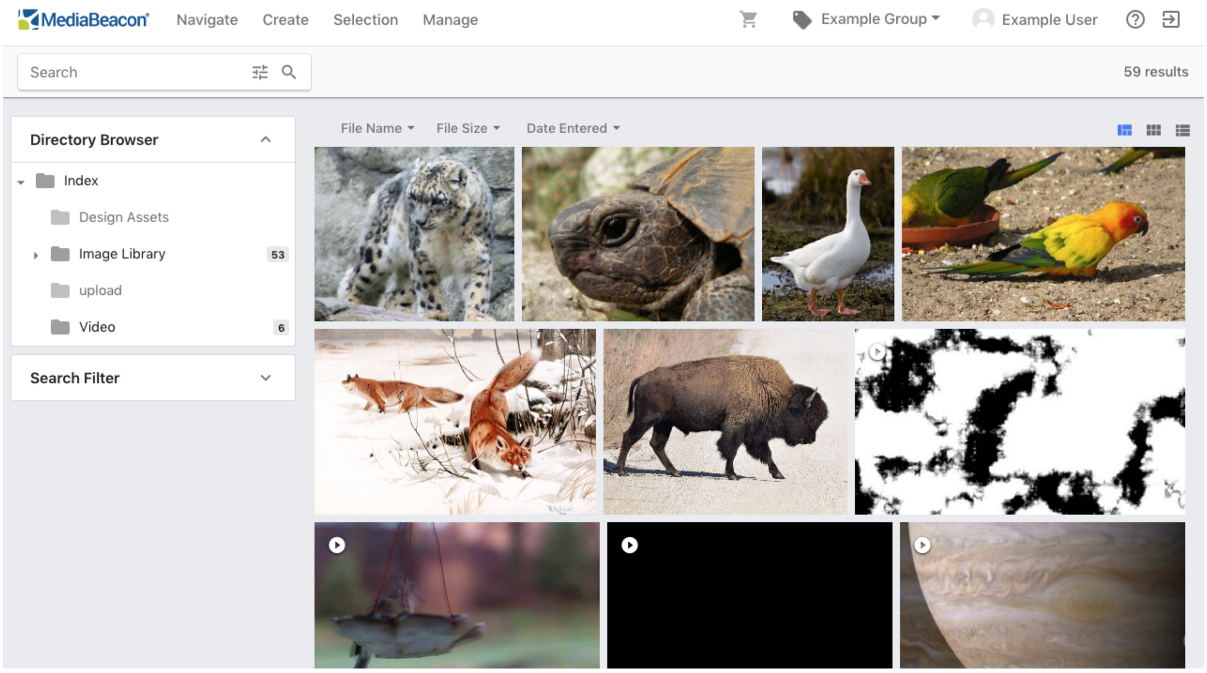
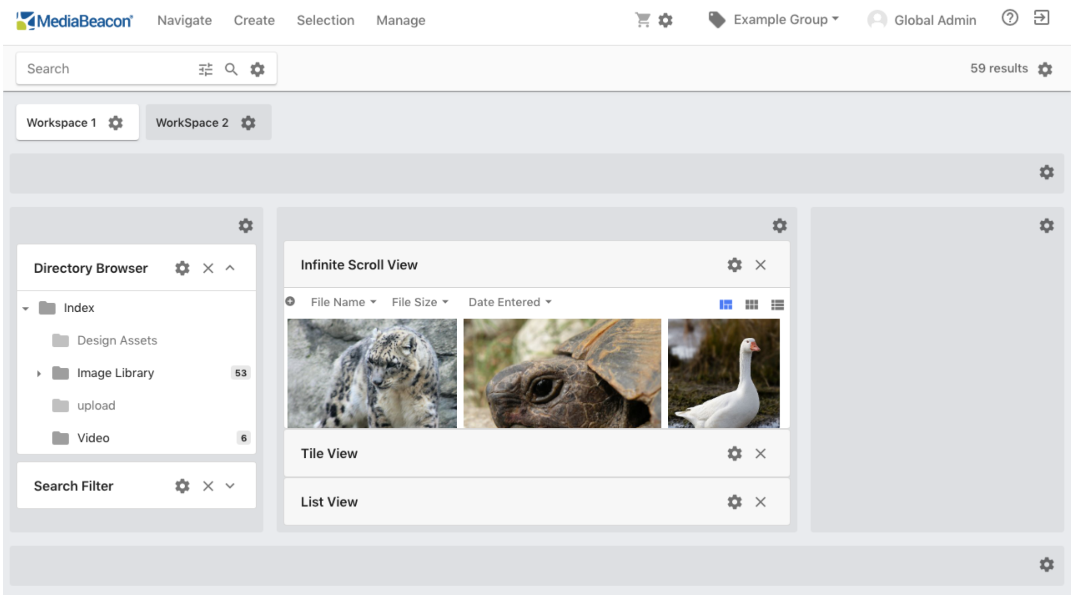


Figure 2: Admin Mode

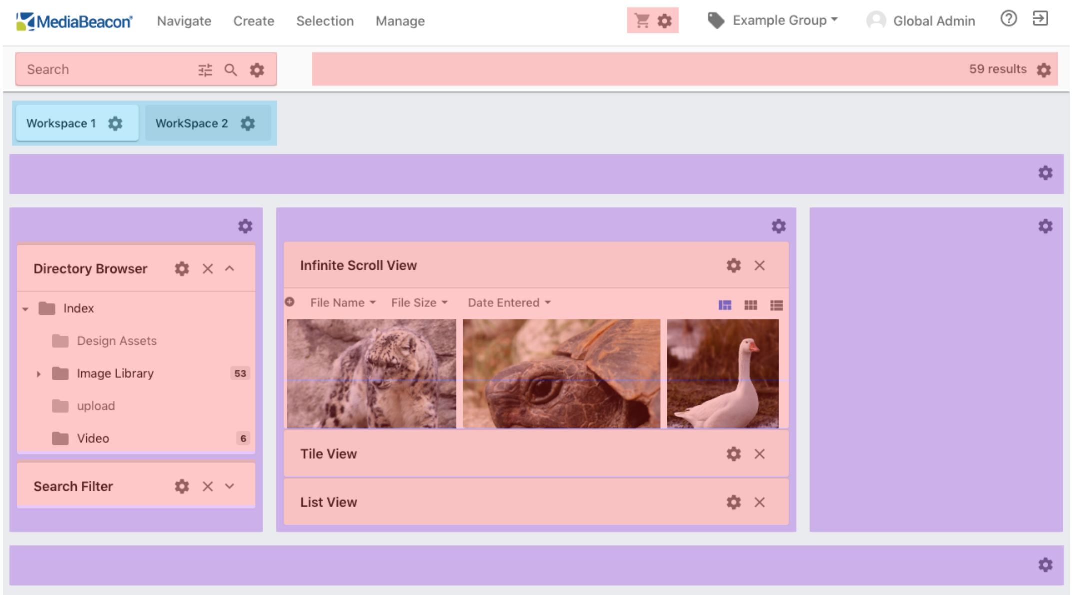


A key concept in MediaBeacon configuration is "admin mode". In this state, the UI is displayed with additional elements allowing access to a variety of different configuration interfaces.

Admin mode is enabled by either a contextual menu command, or a keyboard shortcut:

- Selecting the "Admin Mode" command in the workspace contextual menu.
- macOS: Command-Shift-A
- Windows: Control-Shift-A

While in Admin mode, the UI displays several configuration elements:



- Workspace Tabs (blue)
- Layout Zones (purple)
- Components (pink)
- View component filter header configuration
- Form Stamp component configuration
- Gear Icons, indicating
  - Workspace Tab Menus
  - Layout Zone Configuration Panels
  - Component Configuration Panels

3.1.2. Admin Only Menu Commands

These commands are always visible to Global Administrator-level users.

1. Menu Bar > Create > Workspace: Opens the "Add Workspace" dialog.
2. Menu Bar > Manage:
   a. Workspace: Opens the "Manage Workspace" dialog.
   b. History: Opens the "History" dialog.
      a. Use of this command is granted to non-admin users when the "View Global Asset Log" permission restriction is not set in user's current ACL.
   c. Dictionaries: Opens the "Dictionary" dialog.

        a. Use of this command is granted to non-admin users when the "Administer Dictionaries" permission restriction is not set in user's current ACL.

    d. External Plugins: Opens the "External Plugins" dialog.

        a. Use of this command is granted to all users if the installation's License permits use of the Connectors.

    e. Workflow: Opens the "Workflow Manager" dialog.

    f. R3Search: Opens the "R3Search" dialog.

    g. System Status: Opens the "System Status" dialog.

    h. System: Redirects the user to the "Admin Core" interface.

        a. **Note: This interface can be opened as a separate tab.**

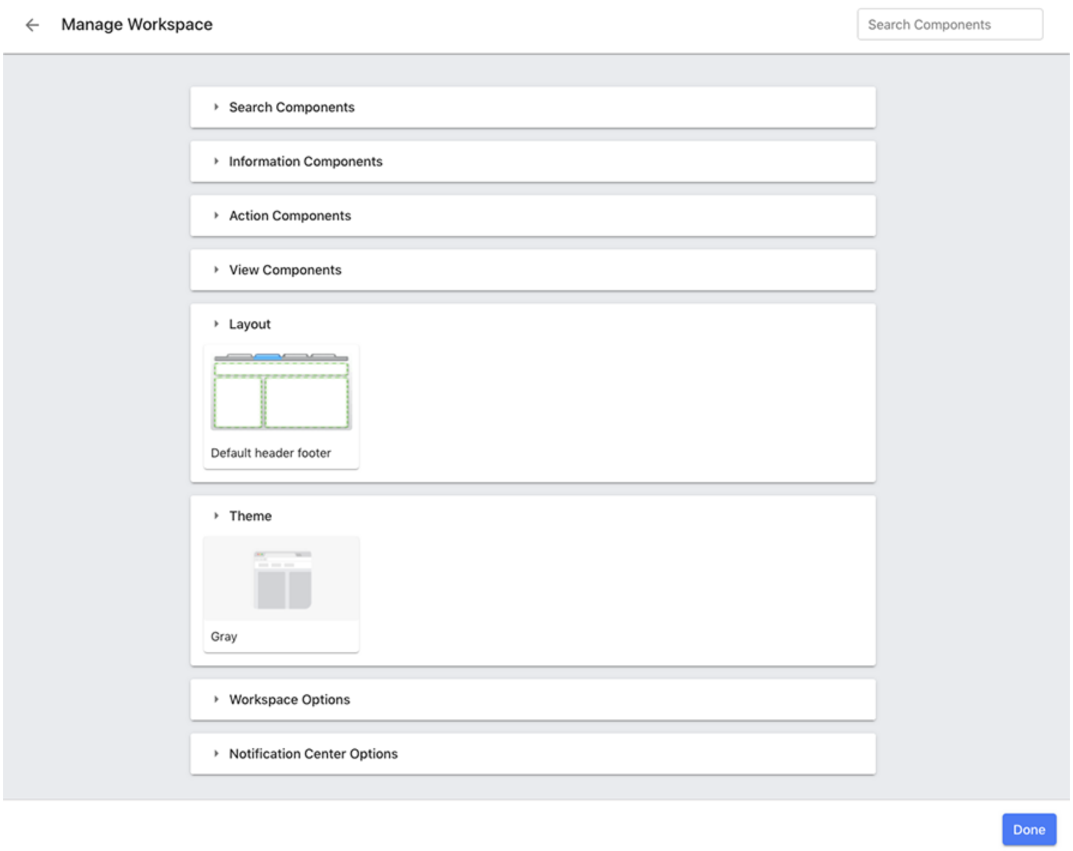
3. Contextual Menu and Asset Menu, when used on a .mbwidget or .xmpper asset.

    a. "Load Component"

    b. "Load Workspace"

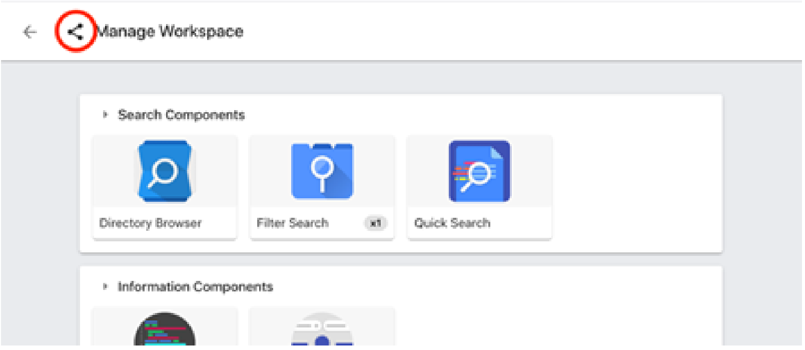
3.2. Manage Workspace Dialog

Options for workspace configuration in the Manage Workspace dialog.



The Manage Workspace dialog controls workspace behavior in most respects.

A visual indicator has been added to the Manage Workspace dialog to indicate when a workspace is shared with one or more other ACLs.



3.2.1. Dialog Sections

General

- "**Back" button** (left arrow in the dialog's upper left): Cancels changes to workspace configuration.
- **"Done" button**: Applies changes to configuration.
  - **Note:** Adding workspaces in the component sections are added or removed immediately, regardless of clicking "Done".
- **"Search Components" field** (in the upper right of the dialog): Text entered here will filter the components displayed in the components sections, see below.

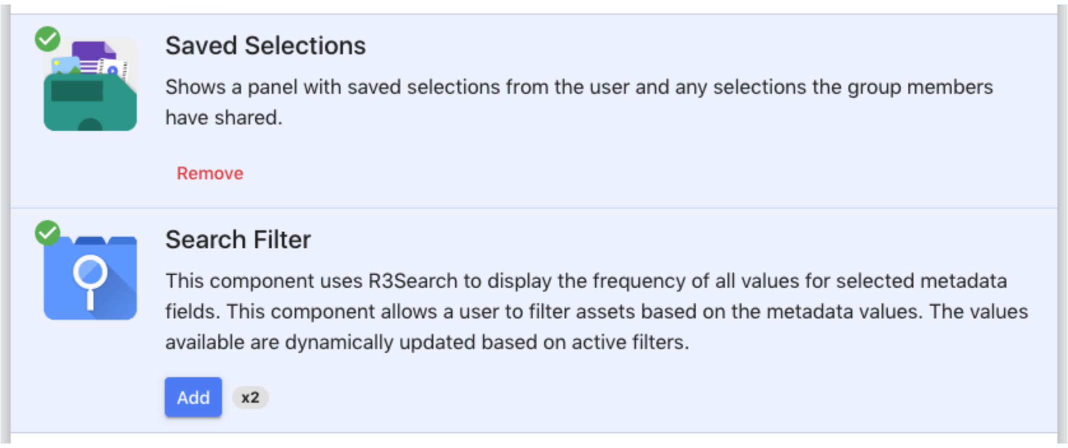
Component Sections

Four spindown sections are listed here, each a different class of component:

- Search Components
- Information Components
- Action Components
- View Components

Components can be added to the workspace in these sections, and the "Search Components" field will filter components based upon their name.

Note that some components can have multiple instances present on a workspace, and others cannot. When a component is added to the workspace via the "Add" button, this is indicated as shown below.



Saved Selections

Shows a panel with saved selections from the user and any selections the group members have shared.

Search Filter

This component uses R3Search to display the frequency of all values for selected metadata fields. This component allows a user to filter assets based on the metadata values. The values available are dynamically updated based on active filters.

- Single-instance components: The "Add" button becomes a "Remove" button,
- Multi-instance components will show a "number of instances" indicator (xNumber) to the left of the "Add" button.
  - Note that to remove multi-instance components, this must be done outside of the Manage Workspace dialog, by clicking the "X" button on the component's title bar while in admin mode.

Layout & Theme

The Layout section allows the user to choose between different arrangements of layout zones for the workspace. The "Default header footer" is the most flexible of these and is the most commonly used.

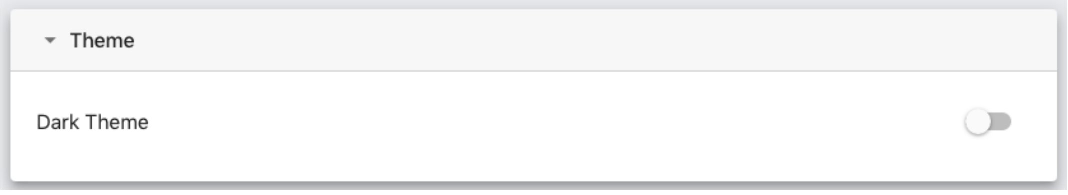
- **Recommended Layout:** Default Header footer

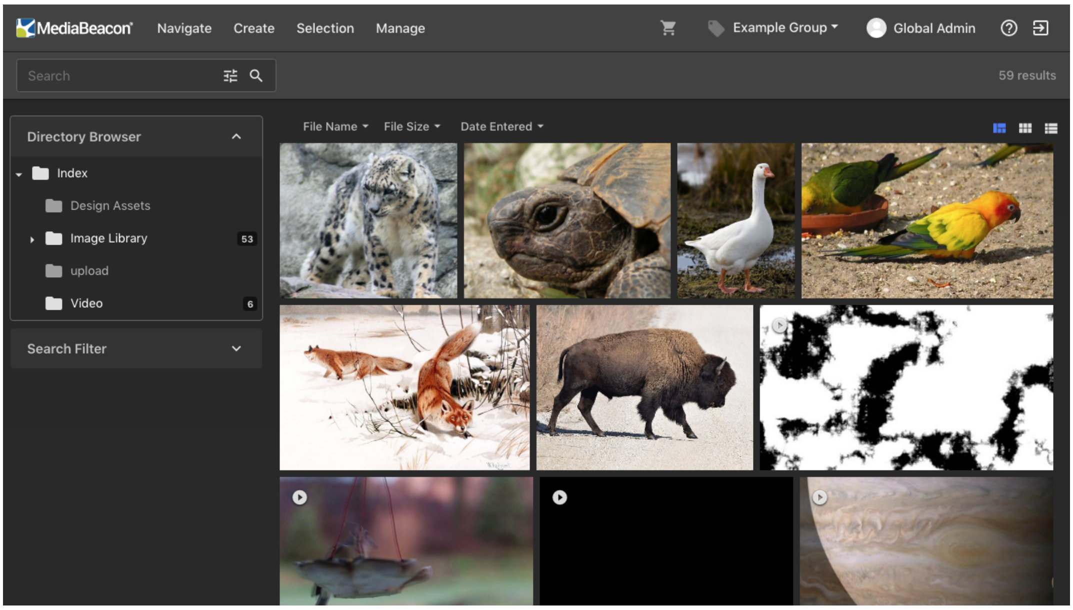
- **Other Layouts:** Default, Default header widget, Single column, Single column view, Three by two, Two column, Two column header footer, Vertical columns

Note that when a layout is changed, the components may be rearranged.

Themes

This area has been substantially simplified in MediaBeacon 2018.0 and greater, now showing a single toggle control to use the Dark Theme.

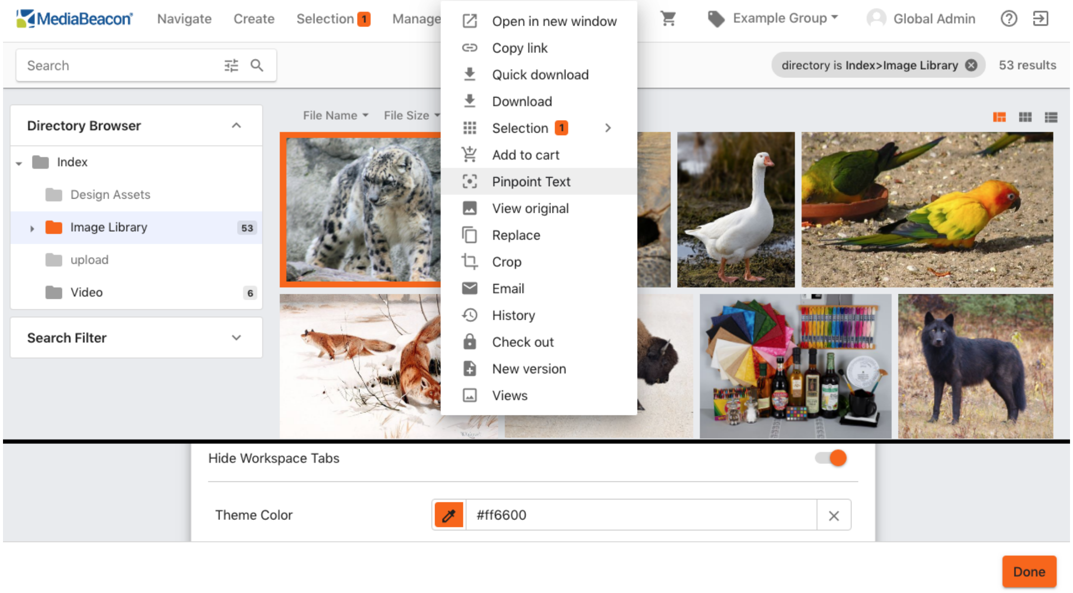




Workspace Options

Styling Options

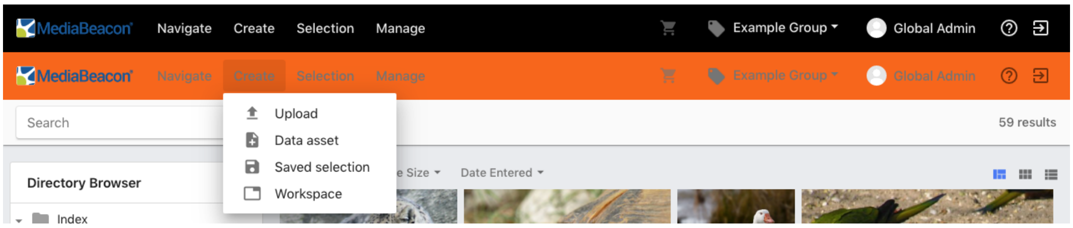
- **Hide Menu Bar**:Hides the menu bar on the workspace. This option is disabled by default.
- **Hide Workspace Tabs**:Hides the workspace tabs. This option is enabled by default.
- **Theme Color**:This changes the "primary" color of the workspace: button color, toggle color, folder and asset selection color, etc.



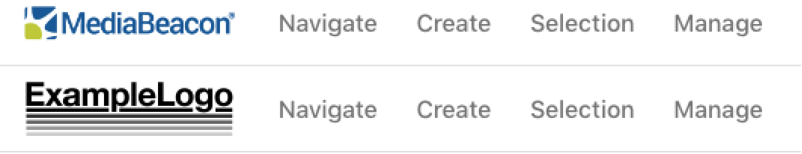
Hide Workspace Tabs ●

Theme Color #ff6600

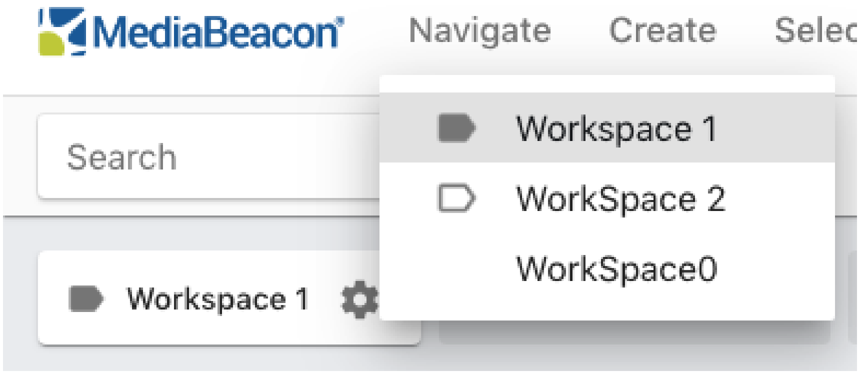
- **Menu Bar Background Color**:This color selector allows the admin to choose a different background color for the menu bar. Note that the text will automatically change to a contrasting grey scale color, and that the pulldown menus themselves remain as the theme color dictates.



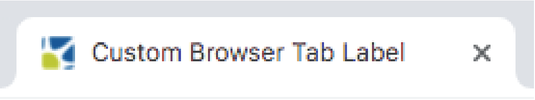
- **Home Button Icon**:An icon may be chosen here, using the standard icon picker. The home button displays at 140x38px, and the icon will scale to fit within that area.



- **Workspace Icon**:An icon chosen here will be displayed next to the name of the workspace in the Navigate Menu and the workspace tab.



- **Custom CSS Classes**: Strings entered here become CSS classes on the body tag of the current workspace.
- **Browser's Window Title**: A string entered here becomes the browser tab label.



URL Options

Note: These options are not commonly used in most installations of MediaBeacon.

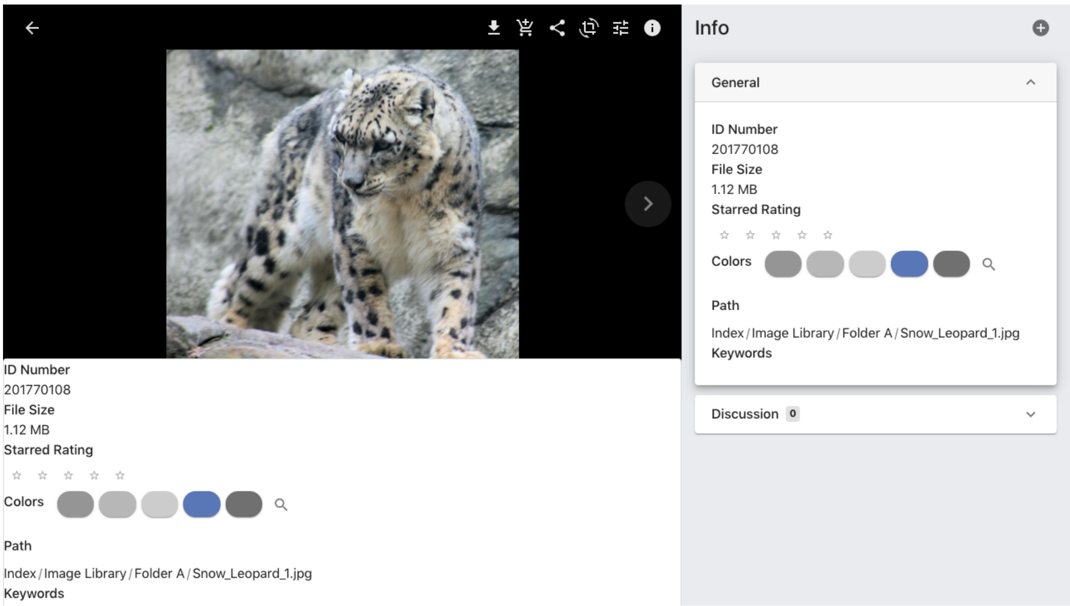
- **Custom URL for Workspace**: A string entered here acts as a redirect to the given workspace, for example: "*https://externaltraining2.mediabeacon.com/p/example*" might resolve to "*https://externaltraining2.mediabeacon.com/gwtmain/GwtMain.html#!MP;;p=464EBF9D-94E5-419B-*

A741-7EC76D9A2EE9;;par=0=1=50". The custom URL does not function as a "vanity URL", the workspace's URL remains the same. Note that users who follow such a link cannot be shown workspaces to which their ACL's do not have access.

- **Custom Login URL**: A URL entered here will cause MediaBeacon to redirect the user to that URL with the user accesses MediaBeacon in a non-logged in state. This can be after logging out, but also connecting to MediaBeacon while not logged in. This behavior is dependant upon what workspace the user last used, which is stored in a browser cookie.
- **Custom Link URL for Emails**: A partial URL entered here can redefine the URL sent via email mechanisms in the system. This can be useful if your MediaBeacon instance has a different external DNS assignment then internal to your network.

Default Behavior Options

- **Always show floating general tab**: This option causes the "General" tab to persistently display below the image preview in the Asset View.



ID Number
201770108
File Size
1.12 MB
Path
Index / Image Library / Folder A / Snow_Leopard_1.jpg
Keywords

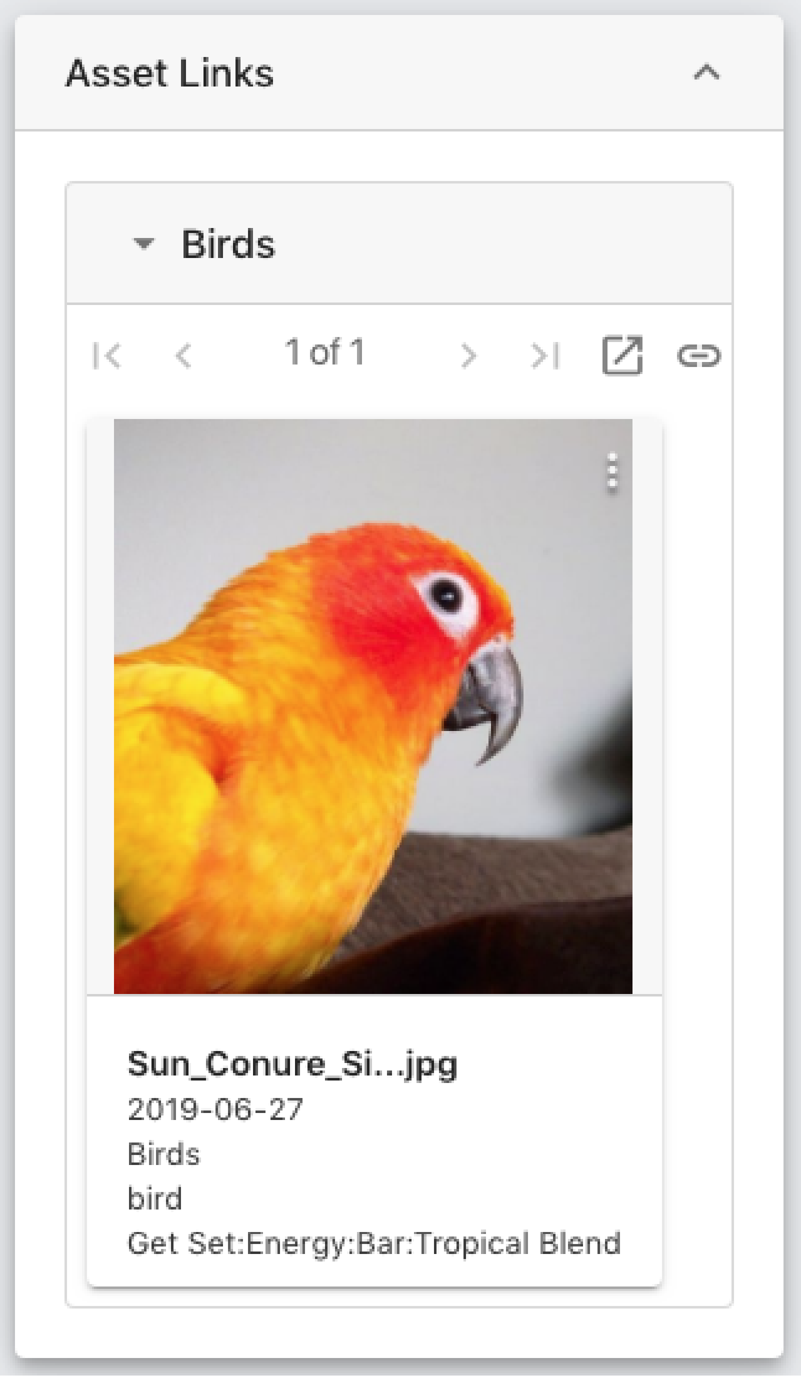
ID Number
201770108
File Size
1.12 MB
Path
Index / Image Library / Folder A / Snow_Leopard_1.jpg
Keywords

- **Zip Quick Downloads**: Enabled by default, this option sets the default behavior for Quick Downloads. When disabled, quick downloaded assets will be automatically decompressed, no ZIP file being left behind. **Note:** this behavior only pertains to the Quick Download function, not the Download function.
- **Set Default Download File Name**: When multiple assets are downloaded at once, they are combined into one ZIP file with the name "MediaBeaconFiles.zip". A string entered here changes this name for downloads from the workspace being configured.
- **Default Asset View Form**: A single Form may be designated as the default Form which will be expanded when the user opens the asset view. This Form can not be removed from the asset view, and will always be displayed when browsing to a new asset from within the asset view (using the next and previous buttons). Note: Although a global administrator may choose any Form for this field, the ACL's "Visible Forms" entries have precedence. The upshot being that if the Default Asset View Form is not in the ACL's Visible Forms list, it will not be displayed.
- **Default Asset View Tabs**: Multiple Forms may be designated as "default tabs". These Forms tabs will always be listed in the metadata panel of the asset view, and cannot be removed. Note: Although a global administrator may choose any Forms for this field, the ACL's "Visible Forms" entries have precedence. The upshot being that if a Default Asset View Tab is not in the ACL's Visible Forms list, it will not be displayed.

- **Required Forms for Upload**: One or more Forms may be chosen for this entry, becoming the upload Form used for all uploads in this workspace. This setting will supersede the ACL upload Form setting, and is overridden by upload Form settings in components. Component upload Form settings only pertain to uploads initiated by clicking or asset-dragging directly on that given component. The upshot is that the Create > Menu command's upload Form behavior can be controlled only with the ACL or workspace "Upload Form" setting. Of special note: Forms chosen here are **NOT** restricted by the ACL's "Upload Form" setting; therefore a workspace can use an upload Form that is not otherwise visible to the user in asset view.

- **Fields On Asset View General Tab**: Fields chosen here are displayed at the bottom of the "General" tab in the asset view. At least one must be specified. Any field may be chosen, and is not limited to fields that the ACL has in any "Visible Form". Note: Adding a field here will make that field editable by the user, unless the user's Group "Edit Level" setting is lower that the given field.

- **Fields below Asset Link**: Fields chosen here will be visible in the asset preview pane of the "Asset Links" Form tab. Note: Adding a field here will make that field editable by the user, unless the user's Group "Edit Level" setting is lower that the given field.



Sun_Conure_Si...jpg
2019-06-27
Birds
bird
Get Set:Energy:Bar:Tropical Blend

Advanced JavaScript Options

The following areas allow JavaScript code (using MediaBeaconJavaScript API) to replace the function of certain actions in the interface.

Note that if these sections are turned off via the right-hand toggle button, the contents will be deleted.

- **Enable javascript on direct download**: Code entered here replaces the function of all download actions on a workspace, except those in the Notification Center.

- **Enable javascript in the asset view**: Code entered here replaces the function of opening the asset view.
- **Enable javascript on home button click**: Code entered here replaces the function of the home button. This is sometimes used to change this interface element to navigate to a specific workspace, rather than clearing search criteria.
- **Notification Center Options: Enable javascript on direct download**: Replaces the function of download actions in the Notification Center. Note that this includes conversion downloads that use the "Download on conversion complete" option in their Loading Dock conversion preset.

General

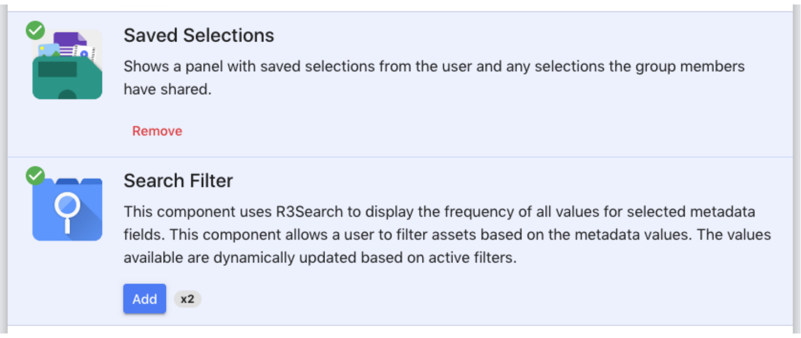
- **Back" button** (left arrow in the dialog's upper left): Cancels changes to workspace configuration.
- **"Done" button**: Applies changes to configuration.
  - Note: Adding workspaces in the component sections are added or removed immediately, regardless of clicking "Done".
- **"Search Components" field** (in the upper right of the dialog): Text entered here will filter the components displayed in the components sections, see below.

Component Sections

Four spindown sections are listed here, each a different class of component:

- Search Components
- Information Components
- Action Components
- View Components

Components can be added to the workspace in these sections, and the "Search Components" field will filter components based upon their name. Note that some components can have multiple instances present on a workspace, and others cannot. When a component is added to the workspace via the "Add" button, this is indicated as shown below.



Saved Selections

Shows a panel with saved selections from the user and any selections the group members have shared.

Search Filter

This component uses R3Search to display the frequency of all values for selected metadata fields. This component allows a user to filter assets based on the metadata values. The values available are dynamically updated based on active filters.

- Single-instance components: The "Add" button becomes a "Remove" button,
- Multi-instance components will show a "number of instances" indicator (xNumber) to the left of the "Add" button.
  - Note that to remove multi-instance components, this must be done outside of the Manage Workspace dialog, by clicking the "X" button on the component's title bar while in admin mode.

Layout & Theme

The Layout section allows the user to choose between different arrangements of layout zones for the workspace. The "Default header footer" is the most flexible of these and is the most commonly used.

- **Recommended Layout:** Default Header footer
- **Other Layouts:** Default, Default header widget, Single column, Single column view, Three by two, Two column, Two column header footer, Vertical columns

Note that when a layout is changed, the components may be rearranged.

3.3. Using Workspaces

How to add, arrange and share workspaces.

This article deals with workspaces as arranged within ACLs. For "Manage Workspace" functionality, see *Manage Workspace Dialog* on page 10.

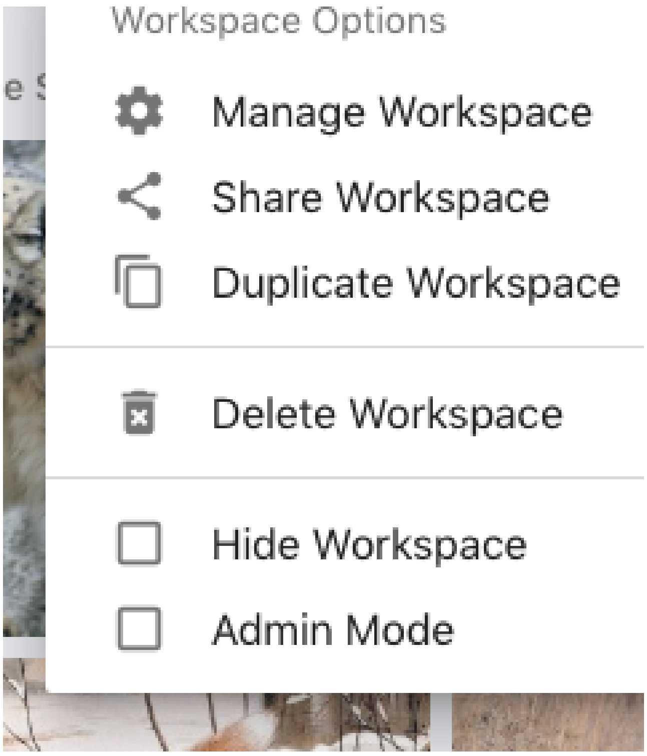
Workspaces Overview

Each ACL can contain a number of workspaces, each of which has a UUID that is displayed as part of the browser's URL. This UUID is always specifically tied to a given workspace. Identically configured workspaces will still be unique.

Workspaces can be shared between ACLs. A shared workspace contains the same components and settings wherever it appears. A change made to a workspace in one ACL will change it in all ACLs with which is it shared.

Workspace Options Contextual Menu

Administrator-Level users may use several commands in the Workspace contextual menu. It may be displayed in either of two ways:



Contextual Menu

1. Right-click any non-component, non-menu, or non-menu area
2. **Note:** this method does not require the user be in admin mode.

Selected Workspace Tab Gear Menu

1. Enter admin mode
2. Select the desired workspace tab
3. Left-click the gear icon in the highlighted workspace tab.

Non-Selected Workspace Tab Gear Menu

1. Enter admin mode
2. Left-click the gear icon in a non-highlighted workspace tab.

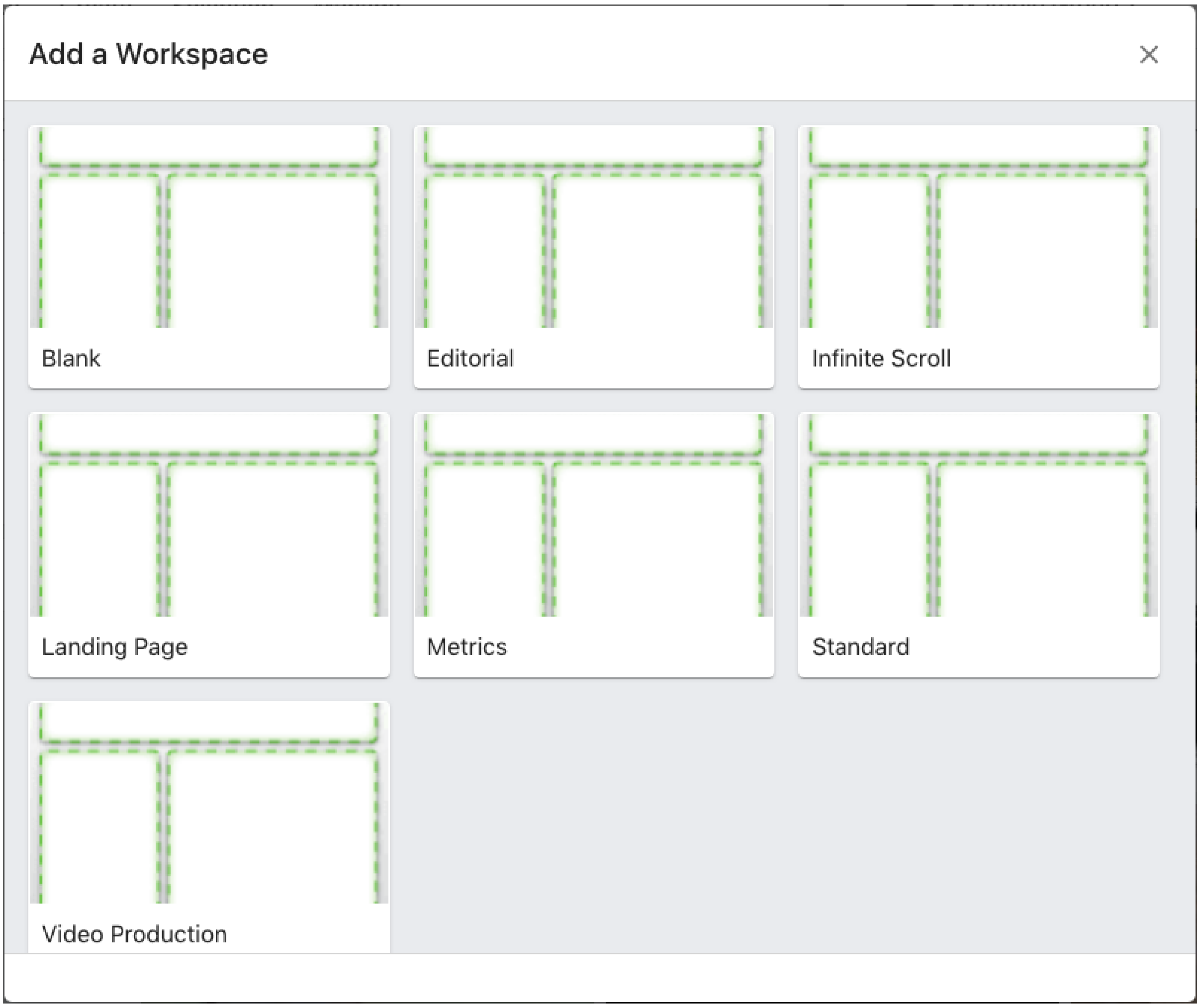
It contains the commands:

- Manage Workspace (absent in Non-Selected Workspace Tab Gear Menu)
- Share Workspace
- Duplicate Workspace (absent in Non-Selected Workspace Tab Gear Menu)
- Delete Workspace
- Hide Workspace

- Admin Mode

3.3.1. Adding Workspaces

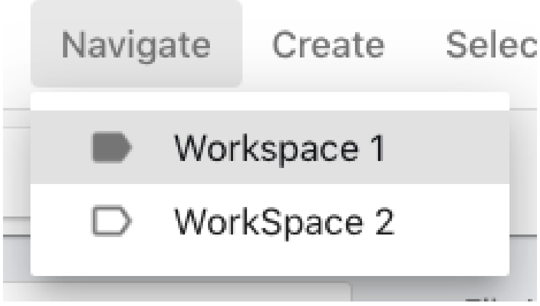
Workspaces may be added by using the Menu Bar > Create > Workspace command, which opens the "Add A Workspace" dialog.



The workspaces available here, except for the "Blank" option, are templates. The templates contain a set of components that are automatically shared between each added instance of that template.

Note: Using a template (other than "Blank") in the "Add a Workspace" dialog is analogous to duplicating a workspace using the "Duplicate as an asset, retaining component links" option. For this reason it is recommended to create new workspaces with only the "Blank" option, then adding and/or sharing components as necessary is recommended.

Each workspace added to an ACL is displayed in the Navigation Menu.



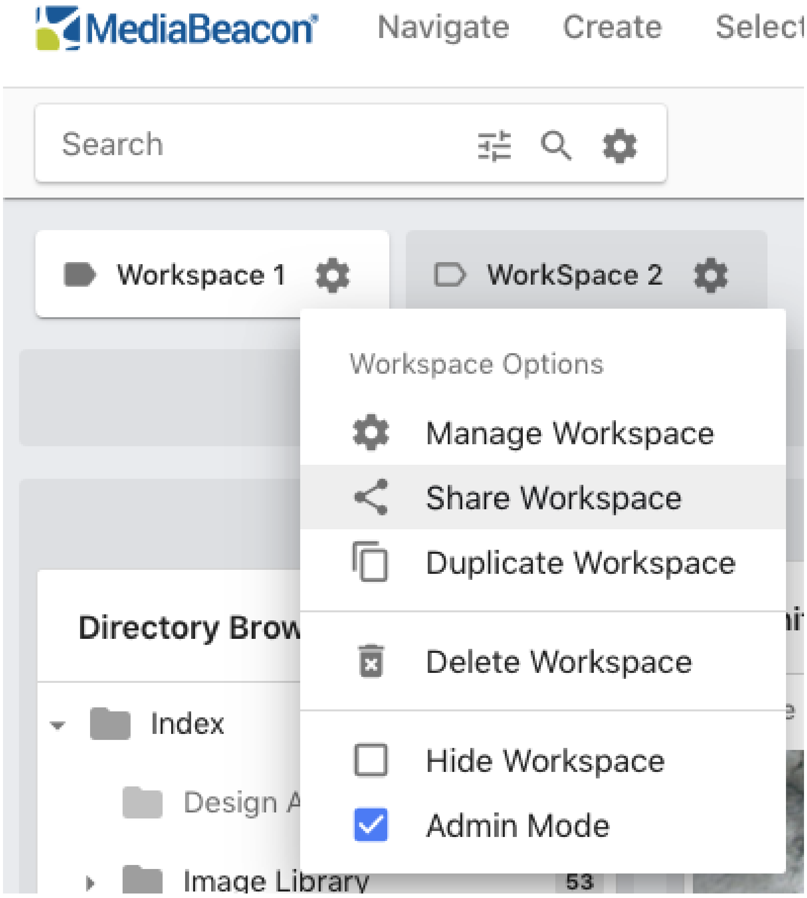
3.3.2. Sharing Workspaces and Duplicating Workspaces

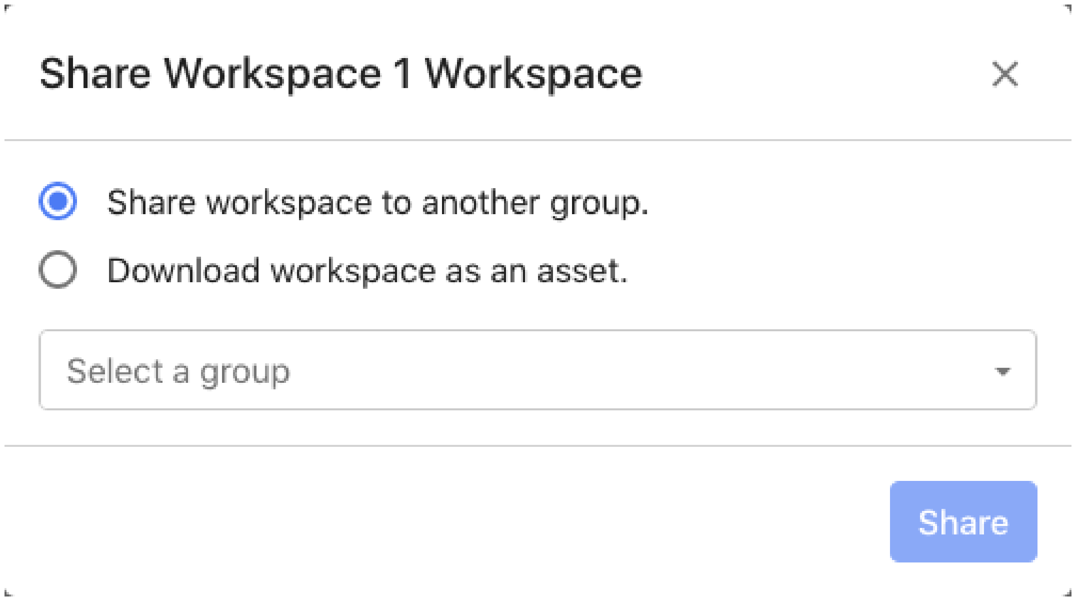
Workspaces may be shared and duplicated.

- Sharing a workspace does not make additional workspaces, but rather gives another ACL access to the shared workspace.
- Duplicated workspaces are new workspaces that have the same settings as the original workspace at the time of the duplication.

Sharing

Workspaces may be shared in the "Share Workspace" dialog. To open this dialog, choose "Share Workspace" in the Workspace Options Menu.




Share Workspace 1 Workspace

● Share workspace to another group.

○ Download workspace as an asset.

Select a group

1. Choose: "Share workspace to another group" Selecting this option allows the admin to choose a group.

2. A list of ACLs is shown for the chosen group.
3. If the group has only one ACL, as is common, it will automatically be selected.
4. Once an ACL is selected, the admin may click "Share", and the workspace will be accessible to the specified ACL.
5. INFO CALLOUT: When a workspace is shared with an ACL with currently logged in users. The shared workspace will not be accessible to them until they logout and log in again.

The "Download workspace as an asset" option is discussed below.

Duplicating

Duplicating a workspace always assigns a new UUID to the duplicate, and may be done in 3 ways:

- Duplicate in the current ACL, retaining component links
  - Usage: Often used for creating duplicate workspaces that might have a handful of different components between them.
- Duplicate as an asset, retaining component links
  - Usage: Often used for making templates that will inherit the current settings of extant components when used.
- Duplicate as an asset, **not** retaining component links.
  - Usage: Often used for making independent, non-linked workspace templates.

The "component linking" function is discussed in detail in the Component Linking and Component ID section. To summarize this feature: Component linking allows components with identical "Component IDs" to share the same configuration.

Duplicate a workspace in the current ACL, retaining component links

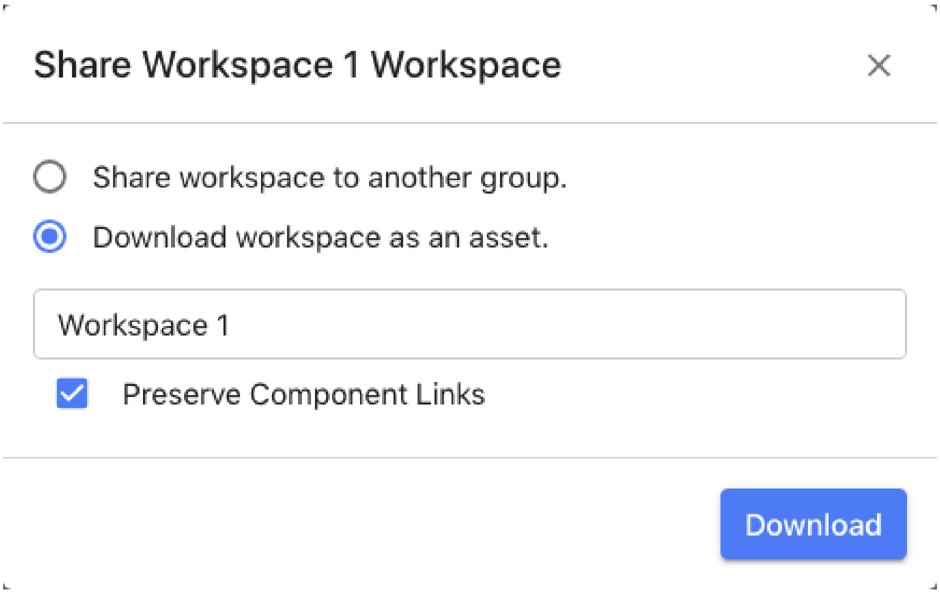
1. Select the "Duplicate Workspace" command from the workspace contextual menu.
2. Workspace is duplicated with the same label, new UUID and linked components

Duplicate a workspace as an asset, retaining component links

This option will save the current workspaces as an asset (.xmpper file). It is first downloaded by the admin's browser, and must be re-uploaded into MediaBeacon to be used.

To create an .xmpper asset.

1. Open the "Share Workspace" dialog.
2. Choose the "Download workspace as an asset" option.
3. **Enable** the "Preserve Component Links" checkbox.



Share Workspace 1 Workspace

○ Share workspace to another group.

● Download workspace as an asset.

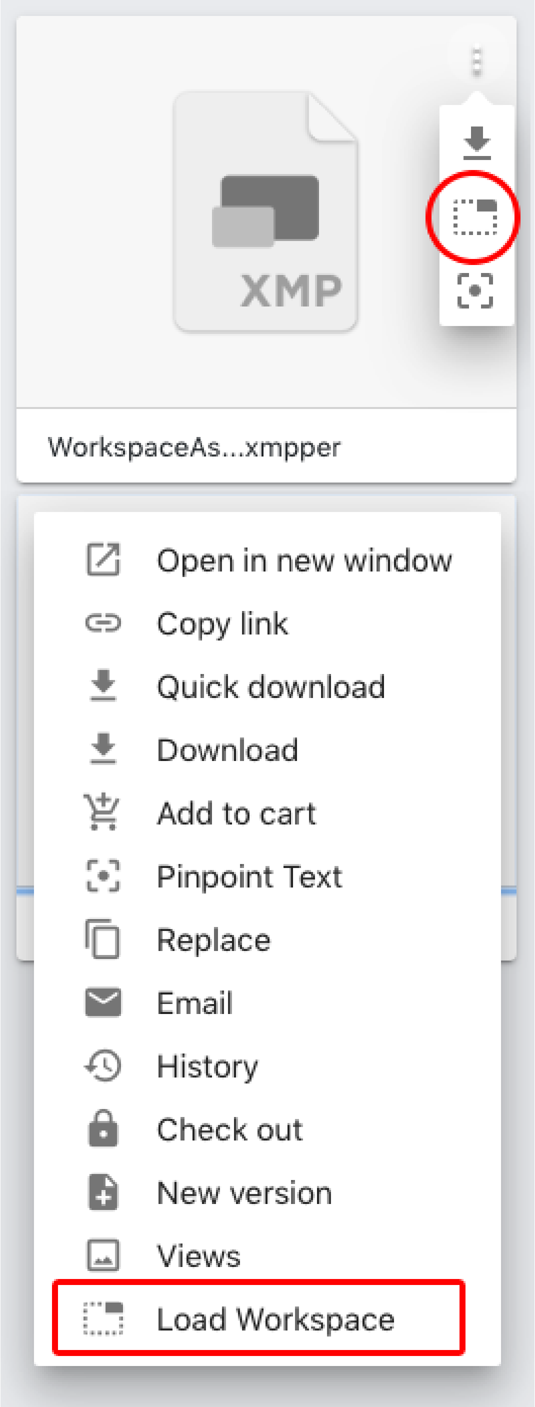
Workspace 1

☑ Preserve Component Links

4. A name field will be shown, allowing the admin to change the name of the downloaded asset
   a. **Note:** Spaces in the string will be replaced with "+" characters upon download.

To use a .xmpper asset.

1. Upload the .xmpper asset into MediaBeacon
2. Select the "Load Workspace" command in the.xmpper asset's contextual menu or asset menu



3. In the "Import Workspace" dialog, confirm.
4. The workspace will be added to the current ACL. **Note:** The name of workspace will be taken from the .xmpper's file name.

Duplicate a workspace as an asset, not retaining component links

To create a .xmpper with non linked components, use the instructions above but disable the "Preserve Component Links" checkbox.

3.3.3. Deleting Workspaces

This command deletes the current workspace from the current ACL. If the workspace is not shared with another ACL, it and any unshared components are permanently deleted.

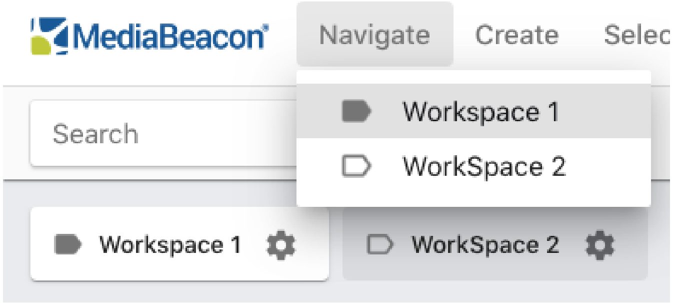
3.3.4. Hide Workspace

Choosing this setting has two effects, hiding the workspace in

1. Hiding the workspace in the Navigation menu.
2. Hiding the workspace from any users not in admin mode.
   a. Admin-level will still have access to the workspace in admin mode.

3.3.5. Arrangement of Workspaces

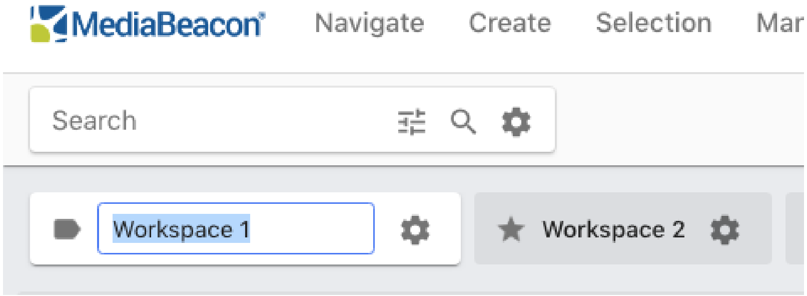
An ACL's workspaces are shown in the Navigate menu, and as workspace tabs while in Admin Mode. These tabs may be dragged and dropped to rearrange their order in the Navigate menu; tabs arranged left-to-right translate to top-to-bottom.



The upper-leftmost tab position is special, being the workspace that the given ACL will view first by default.

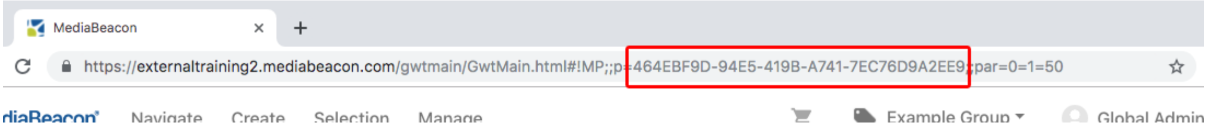
3.3.6. Workspace Names

Workspace names may be changed while in admin mode by double-clicking the name in the workspace tab.



This will change the name of the tab in the Navigate menu and other admin interface elements.

It is important to remember that workspace names are only labels that are specific to only the current ACL. Each workspace has a UUID that is MediaBeacon's identifier for that workspace. If a workspace is shared with another ACL, and its name is changed, the change will **not** be reflected in the other ACL. The UUID number of the workspace, however does not change.



https://externaltraining2.mediabeacon.com/gwtmain/GwtMain.html#!MP;;p=464EBF9D-94E5-419B-A741-7EC76D9A2EE9;par=0=1=50

Note: Capitalization changes to name strings will not be saved. For example, if the workspace "WorkSpace0" is changed to "Workspace0", that change will not be saved. If another character is added, such as a space ("Workspace 0"), the name will be saved.

3.4. Component Configuration

Options for Configuring Components.

MediaBeacon allows administrators to add a wide array of components to workspaces. Each component has its own end user and administrator articles, detailing configuration and usage. Below is a list of existing components.

Component configuration changes are now tracked in Metrics.

Component List

This section lists each component available in MediaBeacon with a short description. For more information on a component, follow the link to read the expanded article.

Component Type	Component Name	Description	User Article	Administrator Article
Search Components	Color Search	Uses the R3Search back end to search	Color Search (User)	Color Search (Administrator)

Component Type	Component Name	Description	User Article	Administrator Article
		assets based on the colors inside them.		
Search Components	Date Search	This component uses R3Search to display information and filter by any date-based field. It will display a frequency bar graph of the number of assets per date range. The date ranges can be narrowed and assets in the view component will be filtered accordingly.	Date Search (User)	Date Search (Administrator)
Search Components	Directory Browser	The directory browser displays the file system as a tree, allowing users to view assets by their position in the file system. Granted appropriate permissions, users can add, move, rename, copy, or delete folders.	Directory Browser (User)	Directory Browser (Administrator)
Search Components	Facet Graph	Visually drill down on asset metadata.	Facet Graph (User)	Facet Graph (Administrator)
Search Components	Featured Content	This component displays content from an arbitrary source with paging.	Featured Content (User)	Featured Content (Administrator)
Search Components	Keyword Search	This component allows for searching based on keywords.	Keyword Search (User)	Keyword Search (Administrator)
Search Components	Metrics View	Component which displays MediaBeaconusage metrics. Adding this component to a workspace will	Metrics View (User)	Metrics View (Administrator)

Component Type	Component Name	Description	User Article	Administrator Article
		cause only metrics data to be displayed on the workspace. No searching of data beyond the metrics core is possible until this is removed.		
Search Components	Places Browser	This component holds Places that can be arranged and organized in a folder structure tree. A Place is a shortcut to a specific ACL, workspace, directory, saved selection, saved search, or hierarchy tree link.	Places Browser (User)	Places Browser (Administrator)
Search Components	Quick Search	This component searches assets by filename (case-sensitive) and the MediaBeacon ID number by default. The MediaBeacon administrator can enable additional fields or types of metadata to be available for searching.	Quick Search (User)	Quick Search (Administrator)
Search Components	Saved Searches	Shows a panel with saved searches from the user and any searches the group members have shared.	Saved Searches (User)	Saved Searches (Administrator)
Search Components	Saved Selections	Shows a panel with saved selections from the user and any selections the group members have shared.	Saved Selections (User)	Saved Selections (Administrator)

Component Type	Component Name	Description	User Article	Administrator Article
Search Components	Search Filter	This component uses R3Search to display the frequency of all values for selected metadata fields. This component allows a user to filter assets based on the metadata values. The values available are dynamically updated based on active filters.	Search Filter (User)	Search Filter (Administrator)
Search Components	Search Refinement	Provides information on which metadata fields contain your most recent search terms, and allows you to further focus your search by field and term.	Search Refinement (User)	Search Refinement (Administrator)
Search Components	Tag Cloud	Create a tag cloud from values in any solr indexed field.	Tag Cloud (User)	Tag Cloud (Administrator)
Search Components	Taxonomy Browser	This component allows browsing by hierarchical metadata fields. The taxonomy fields available are setup in the user or group administration pages.	Taxonomy Browser (User)	Taxonomy Browser (Administrator)
Information Components	HTML	Component displays HTML markup as well as MediaBeacon wiki markup.	HTML (User)	HTML (Administrator)
Information Components	Loading Dock Status	Component for displaying the contents of the	Loading Dock Status (User)	Loading Dock Status (Administrator)

Component Type	Component Name	Description	User Article	Administrator Article
		loading dock. Also has global loading dock actions.		
Information Components	Logged In Users	Component to display currently logged in users.	Logged In Users (User)	Logged In Users (Administrator)
Information Components	Login	This component displays a Login form.	Login (User)	Login (Administrator)
Information Components	M4sterPlan Component	Component displays active M4sterPlans and allows creation of new.	M4sterPlan Component (User)	M4sterPlan Component (Administrator)
Information Components	Form	This component displays a Form and tracks either the current selection or the single selection, displaying information accordingly.	Form (User)	Form (Administrator)
Information Components	Report	A component to give reports on certain points of data such as number of users or total asset size.	Report (User)	Report (Administrator)
Information Components	RSS	This component allows the user to define the URL to an RSS feed to display.	RSS (User)	RSS (Administrator)
Information Components	Search Status	This component allows a user to view and remove any filters applied to the current view.	Search Status (User)	Search Status (Administrator)
Information Components	Upload HTML	Displays HTML markup as well as MediaBeacon wiki markup. Users can drag and	Upload HTML (User)	Upload HTML (Administrator)

Component Type	Component Name	Description	User Article	Administrator Article
		drop files onto the component to upload		
Information Components	Video Playback	This component leverages an HTML5 video tag to play videos back to back.	Video Playback (User)	Video Playback (Administrator)
Action Components	ACL Selection	Shows the available ACLs in a panel which the user can select.	ACL Selection (User)	ACL Selection (Administrator)
Action Components	Cart	This component allows a user to drag assets from a view component, such as Tile View, and drop them into a cart. A cart can be saved as a new selection or be set as the current selection.	Cart (User)	Cart (Administrator)
Action Components	Conversions	This component allows for assets to be sent to the loading dock for processing. It can also be added to the action palette at the bottom of the page.	Conversions (User)	Conversions (Administrator)
Action Components	Form Stamper	This component allows a user to create an Form stamp that can be used to easily update assets' metadata.	Form Stamper (User)	Form Stamper (Administrator)
Action Components	Navigation	Component to display workspaces in some form. Used to navigate to different pages	Navigation (User)	Navigation (Administrator)

Component Type	Component Name	Description	User Article	Administrator Article
		within MediaBeacon.		
Action Components	Trash	A component to drag items onto to delete them.	Trash (User)	Trash (Administrator)
View Components	Infinite Scroll View	View component which displays a page of assets with scroll. Once bottom of scroll is reached, a new page is added to the bottom of the container, so the user can keep scrolling to new pages	Infinite Scroll View (User)	Infinite Scroll View (Administrator)
View Components	List View	Spreadsheet style text view with Editable XMP Metadata.	List View (User)	List View (Administrator)
View Components	Tile View	Displays a tile view of assets.	Tile View (User)	Tile View (Administrator)
View Components	Video Editor View	Component for creating/editing videos from video files in the repository	Video Editor View (User)	Video Editor View (Administrator)

Components of Greatest Use

The following is a list of the most-used components in MediaBeacon.

1. Quick Search
2. Search Status
3. Search Filter
4. Directory Browser
5. Infinite Scroll View
6. Tile View
7. List View
8. Cart
9. Conversions
10. Upload HTML
11. Places Browser
12. Form Stamper

13. Featured Content

14. Form

3.4.1. Using Components on Workspaces

Adding, removing, arranging and sharing components on workspace layout zones.

Adding Components to a Workspace

The components resident to any given workspace can be added by either:

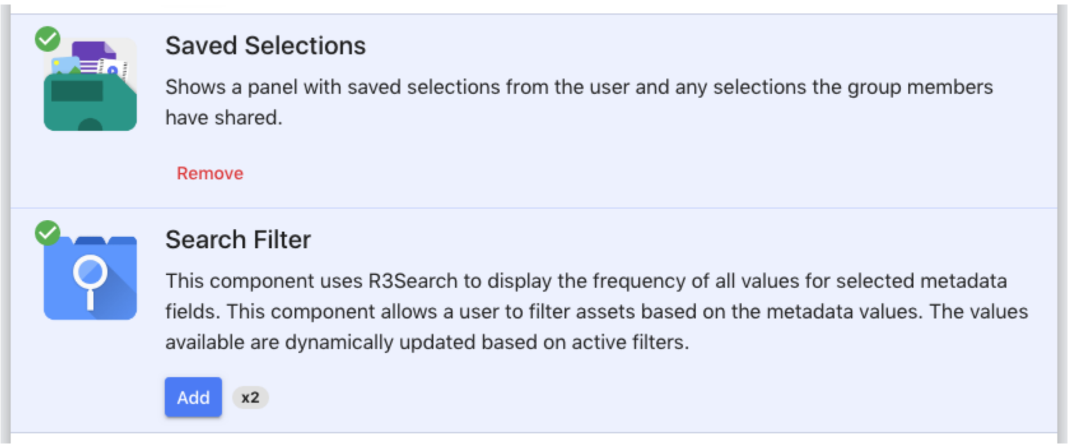
- The Manage Workspace dialog, see below.
- Loading a component asset (.mbwidget).
- Sharing a workspaces between ACLs. See the *Using Workspaces* on page 19 article for more information.

When new components are added, they typically appear in the left-hand layout zone, and shared ones appear in the bottom layout zone. components can be moved from these initial locations.

See *Manage Workspace Dialog* on page 10 for information on the Manage Workspace dialog.

Adding Components via the Manage Workspace Dialog

In the component sections of the Manage Workspace dialog, components may be added to the current workspace by clicking the "Add" button under each component section.



Saved Selections

Shows a panel with saved selections from the user and any selections the group members have shared.

Search Filter

This component uses R3Search to display the frequency of all values for selected metadata fields. This component allows a user to filter assets based on the metadata values. The values available are dynamically updated based on active filters.

- **Single-instance components**: Workspaces may contain only a single instance of this type of component.
  - These components will display a "Remove" button under a component's entrywhen on the current workspace. Clicking this button will remove the component form the current workspace.
- **Multi-instance components**: Workspaces may contain any number of this type of component.
  - These components will display a count indicator to the right of the "Add" button.

- **Note**: Removing multi-instance components cannot be done in the Manage Workspace dialog.

Loading a Component Asset

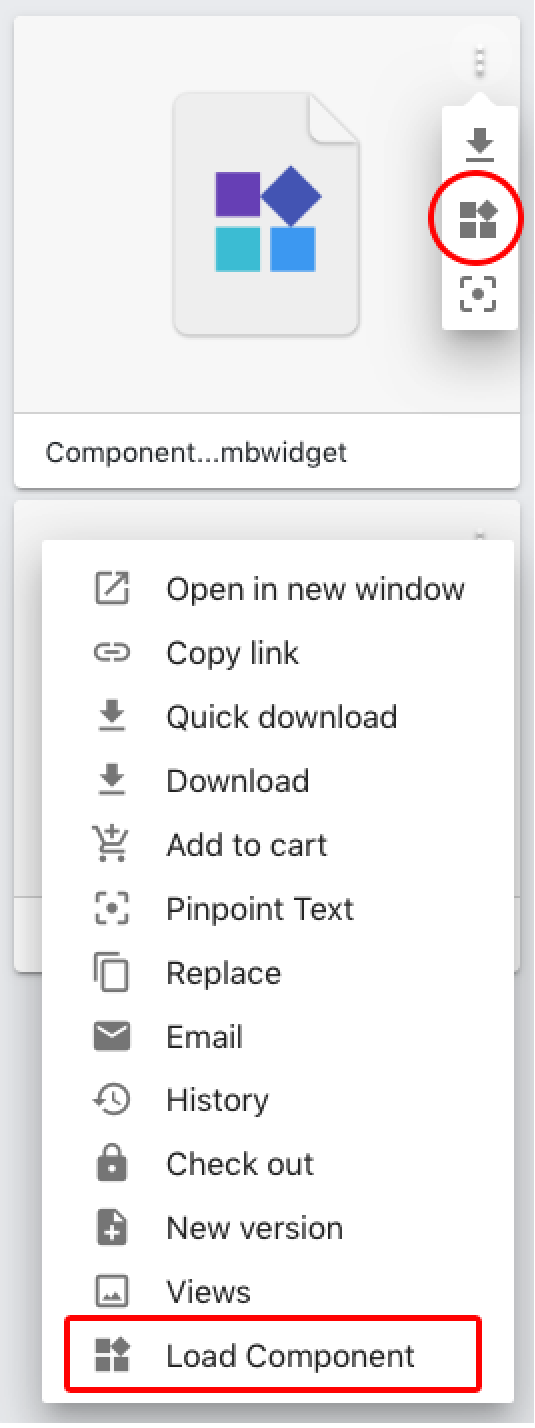
A component's current state may be saved to a component asset that allows the component to be copied onto other workspaces. Components added via this method are are not "linked" or "shared", but separate instances of a component.

Create

1. Enter admin mode.
2. Open the configuration dialog for a given component.
3. Expand the "Share component with other workspaces" section.
4. Click the "Export component to asset" button.
5. The asset (.mbwidget)will be downloaded to the administrator's desktop.
6. Upload the .mbwidget asset into MediaBeacon.

Load

1. Right-click the asset and choose "Load Component".



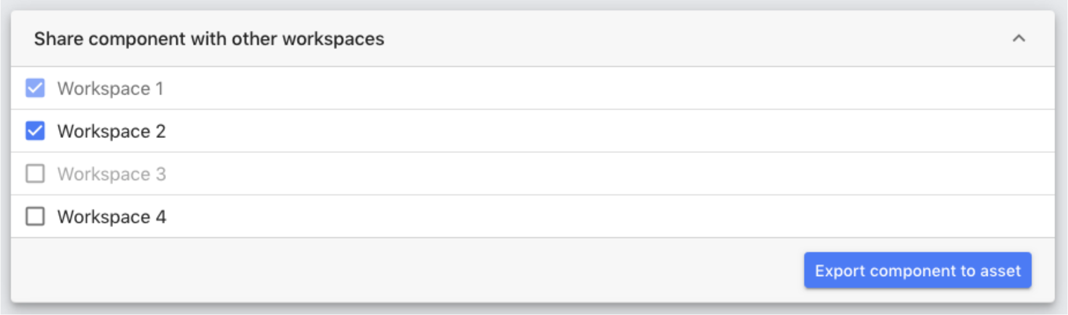
2. Component is added to the current workspace.

Note: If the component being loaded is a single-instance component, and the current workspace contains one of that class of component, it will not be added.

Sharing Components Between Workspaces

Components can be shared across different workspaces, similar to the way that workspaces are shared across ACLs. In this state, known as "component linking" the settings of the shared component is the same on every workspace on which it appears.

A component is shared in the "Share component with other workspaces" section of its configuration dialog (accessible via the gear icon displayed on each component in admin mode.)



Share component with other workspaces

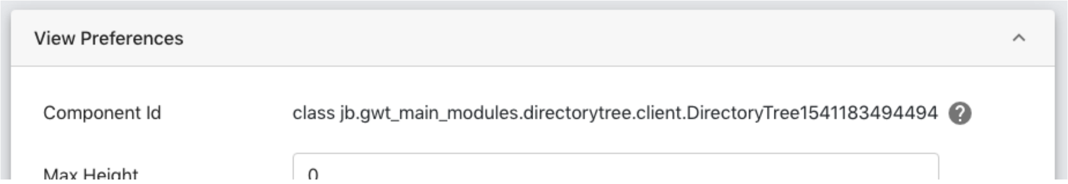
☑ Workspace 1
☑ Workspace 2
□ Workspace 3
□ Workspace 4

- Checked box, Gray text: The component is resident to this workspace.
- Checked box, Black text: The component is currently shared with this workspace.
- No check, Gray text: This component may not be shared with this workspace. This is due to another instance of a single-instance component being resident on that workspace.
- No check, Black text: The component is not currently shared with this workspace.

Component Linking and Component ID

When the "Preserve Component Links" option is used to duplicate workspaces, all components on the new workspace are linked to the ones on the old workspace.

The mechanism that controls this the "Component ID", which can be seen in the "View Preferences" section of a component's configuration dialog.



View Preferences

Component Id: class jb.gwt_main_modules.directorytree.client.DirectoryTree1541183494494

Max Height: 0

The Component Linking function is discussed in detail in the [Managing Components] section, but to summarize this feature: Component linking allows components with identical "Component IDs" to share the same configuration.

Custom Component ID

The Component ID can be altered to a set string, maintaining this ID if the workspace is exported and shared. A locked padlock icon indicates the ID is stable.

This feature cannot be used to "sync" a component with a pre-existing custom ID. If the administrator enters an already used ID, a dialog is displayed that says "Cannot set the component id to that value. That value is already used with another component."

Removing Components from a Workspace

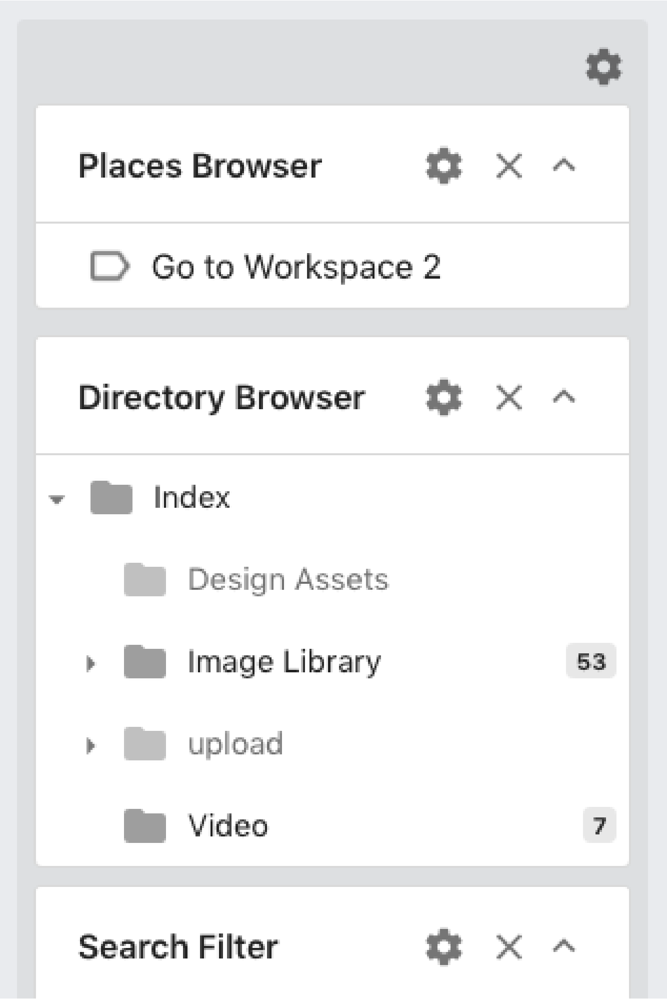
When components are removed, they will be permanently deleted unless they are shared with another workspace, or have been saved as a .mbwidget asset.

Components may be removed from workspaces in a variety of ways

- When viewing a workspace in admin mode, clicking the "X" button in a component's title bar.
  - A toast notification will appear to inform the admin that the component has been removed, which also presents an "Undo" button.
  - This gives the admin 10 seconds to revert that change, after which the component is permanently removed.
- Unchecking a component in the "Share component with other workspaces" section.
  - This removes components on other workspaces.

Layout Zones

The layout area of each workspace allows components to be placed into discrete regions, called layout zones. In admin mode, these are indicated by a darker background and a gear icon in the upper left corner.



Most components can be placed in any layout zone, and will expand or contract to fit the width of their containing zones. Some exceptions to free layout zone placement are:

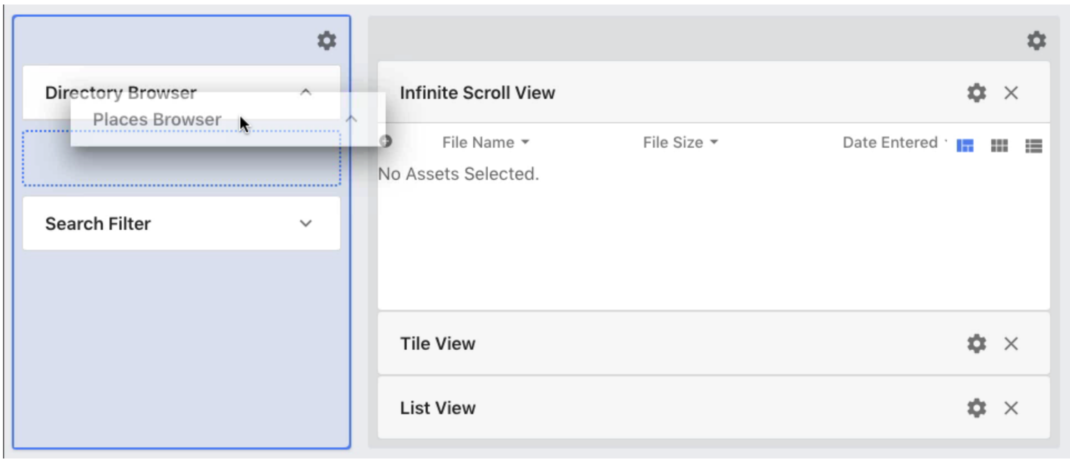
- Some components have a "Menu Bar Mode", that fix their positions to preset areas on the page:
  - Quick Search
  - Search Status
  - Cart
- View components may only reside in a special "view component zone". However, other types of components may be placed above, below, or between them in this zone.
  - This is usually the center middle zone in any given layout, but other layouts may have a different location for the view component zone.

Arranging Components

Components can be moved, while the user is in Admin mode, by click-dragging the title bar of the component

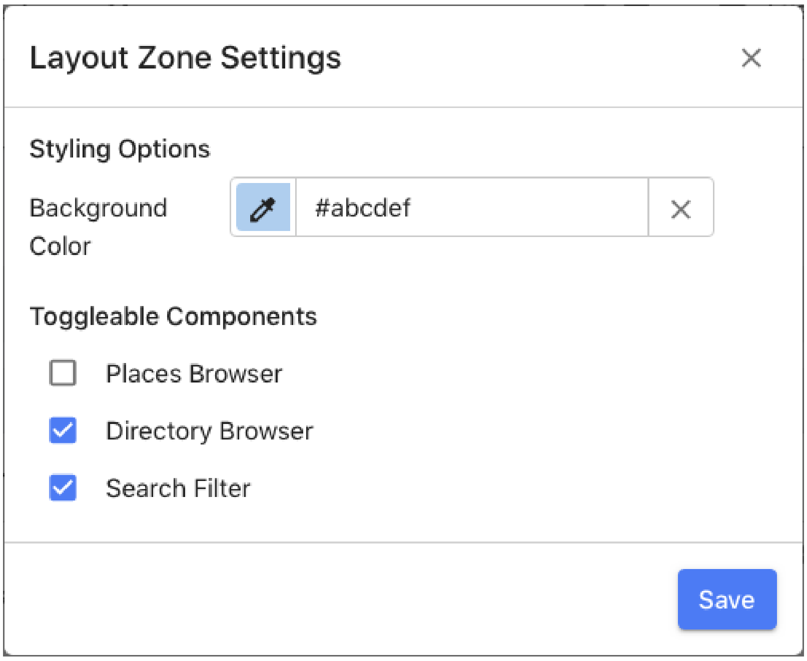
.

- The body of the components in the current layout zone will snap closed.
- The title bar of the dragged component will become transparent during the drag.
- While in this state, the component may be moved to another layout zone or rearranged with respect to other components in the current layout zone.
- The new position indicated by a blue outline. When the user releases the mouse button, the component will be placed in the blue outline and the body of components will again be shown.



Layout Zone Settings

Layout zones themselves have settings that affect the look and visibility of components. The Layout Zone Settings dialog is displayed when the gear icon in the upper right of a layout zone is clicked.



Layout Zone Settings

Styling Options

Background Color: #abcdef

Toggleable Components

☐ Places Browser

☑ Directory Browser

☑ Search Filter

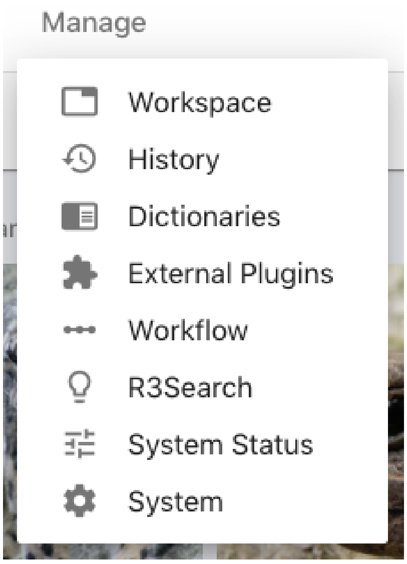
- Background Color
  - The controls the background color of the layout zone, which is normally transparent
- Toggleable Components
  - This option lists the current components contained in the layout zone with a checkbox for each. When checked, only one "toggleable" component can be viewed at a given time in a workspace.
  - Clicking a closed title bar will collapse the title bars of other toggleable components and open the clicked component.
  - Clicking the title bar of an open, toggleable component will collapse the clicked component, other toggleable components remaining unchanged.
  - Note the toggleable option is not available for View components, as they have an auto-toggling feature shared between other view components.

In 2019.1, the "toggleable" setting will override a non-View component's "Hide Component Title" setting, hiding it in the Configuration dialog.

The "toggleable" setting is per layout zone, and setting a component to toggleable in one zone does not make is toggleable by default when moved.

3.5. Administrator Menu Commands

Administrator-only options in the Manage menu.



3.5.1. Dictionaries Dialog

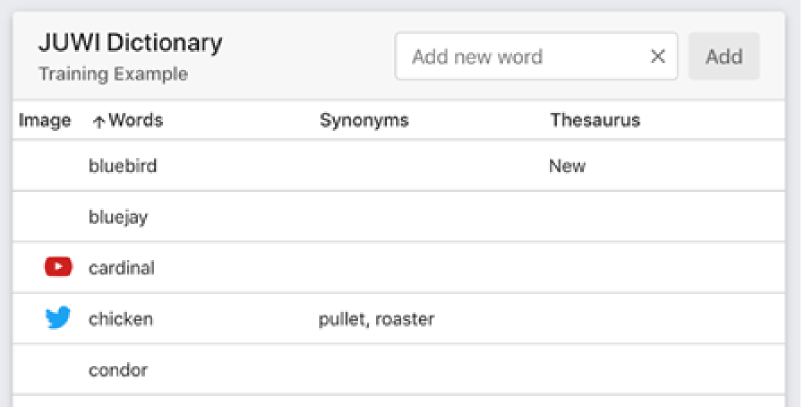
This section is covers different Dictionary fields and functions.

Dynamic Dictionary Fields

This feature allows a single dictionary to control#a set of interdependent fields; these fields in turn will be able to filter and automatically#autofill other fields controlled by that dictionary.

A Dynamic Dictionary is a new type of dictionary that is required to enable this functionality. In contrast to a normal MediaBeacon dictionary, it has an arbitrary number of columns, each#of which represent#(and will be linked to)#a different field.

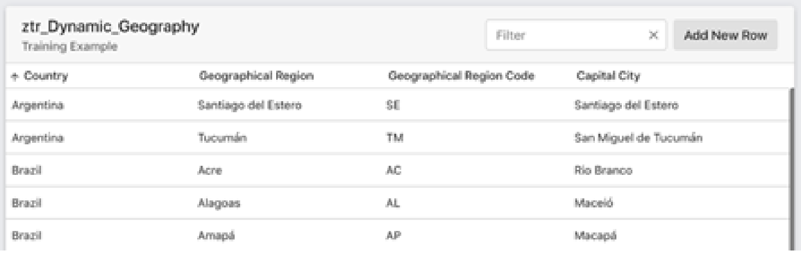
Figure 3: Regular MediaBeacon dictionary



JUWI Dictionary
Training Example

Image	↑Words	Synonyms	Thesaurus
	bluebird		New
	bluejay		
▶	cardinal		
	chicken	pullet, roaster	
	condor		

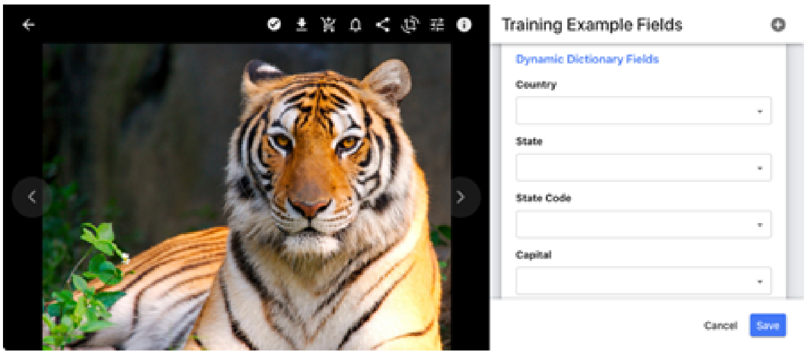
Figure 4: Dynamic Dictionary



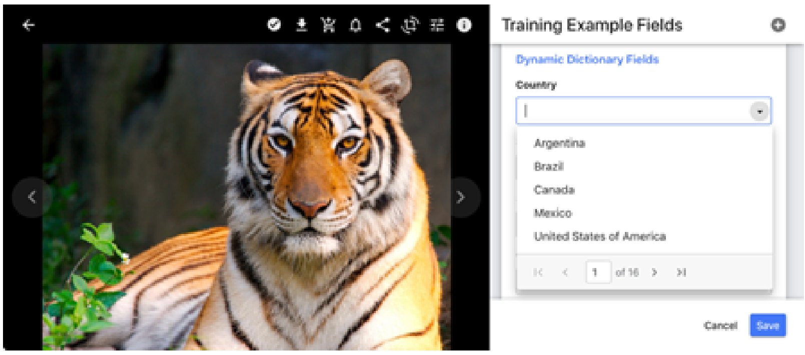
ztr_Dynamic_Geography
Training Example

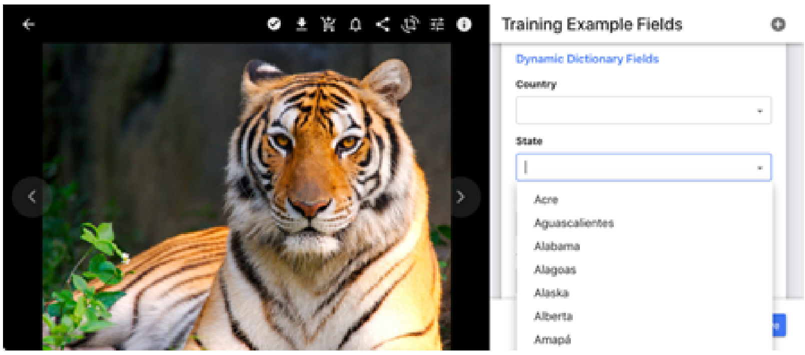
↑ Country	Geographical Region	Geographical Region Code	Capital City
Argentina	Santiago del Estero	SE	Santiago del Estero
Argentina	Tucumán	TM	San Miguel de Tucumán
Brazil	Acre	AC	Rio Branco
Brazil	Alagoas	AL	Maceió
Brazil	Amapá	AP	Macapá

When Dynamic Dictionary fields are used in the same Form, they automatically become reactive to the state of#other fields that share that dictionary. In this example, the "ztr_Dynamic_Geography" dictionary's columns are mapped to four Form Fields.

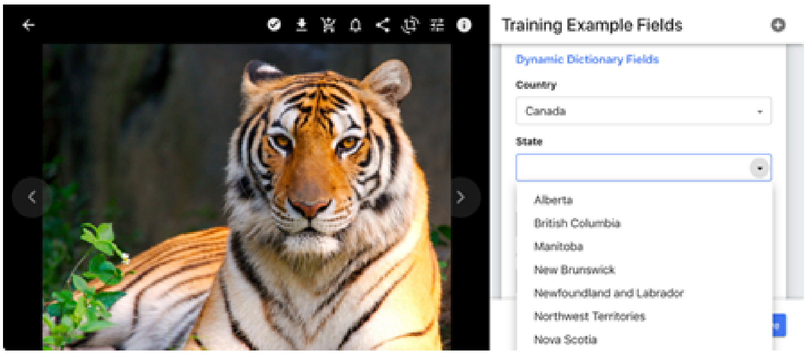


When#a user uses the dictionary pulldown for a field, only unique values are shown.





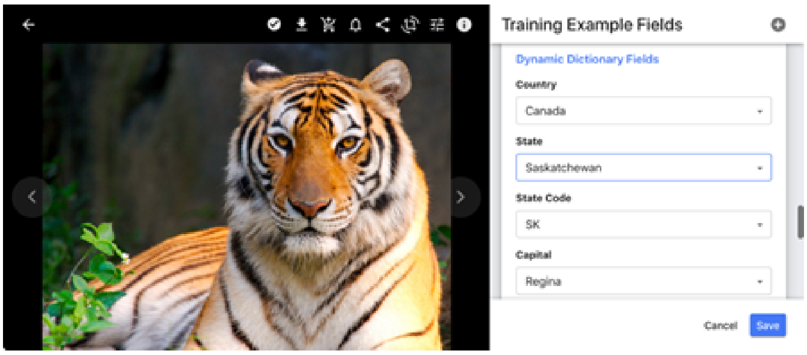
Fields are dynamic in two ways.#First, for example, when a user chooses "Canada" the values in "State" become limited to only those provinces in Canada.


Training Example Fields

Dynamic Dictionary Fields

Country: Canada

The second dynamic behavior occurs when other field values within the set of dynamic fields are chosen enough to#narrow#values down#enough#so that there is only one#possible choice, those values will be automatically filled. For example, if "Saskatchewan" is chosen,#the values for State Code and Capital#are filled in, as there are no other logical choices.



Training Example Fields

Dynamic Dictionary Fields

Country: Canada

State: Saskatchewan

State Code: SK

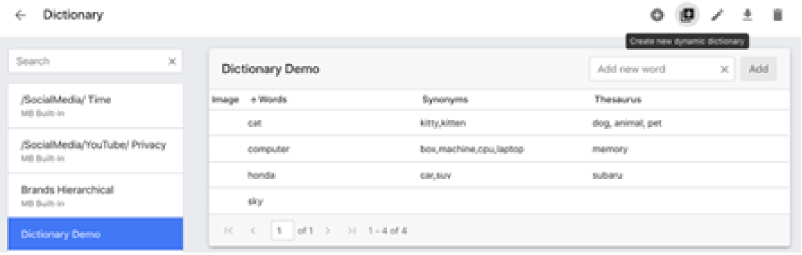
Capital: Regina

It is worth noting that specific behavior of the fields is dependent upon the disposition of the attached Dynamic Dictionary.

Both behaviors are not explicitly “top-down” in terms of hierarchy#(as#hierarchical#dictionaries are), being instead based upon term frequency in the dictionary. See the Dictionary Configuration section for more#information.

Dictionary Configuration

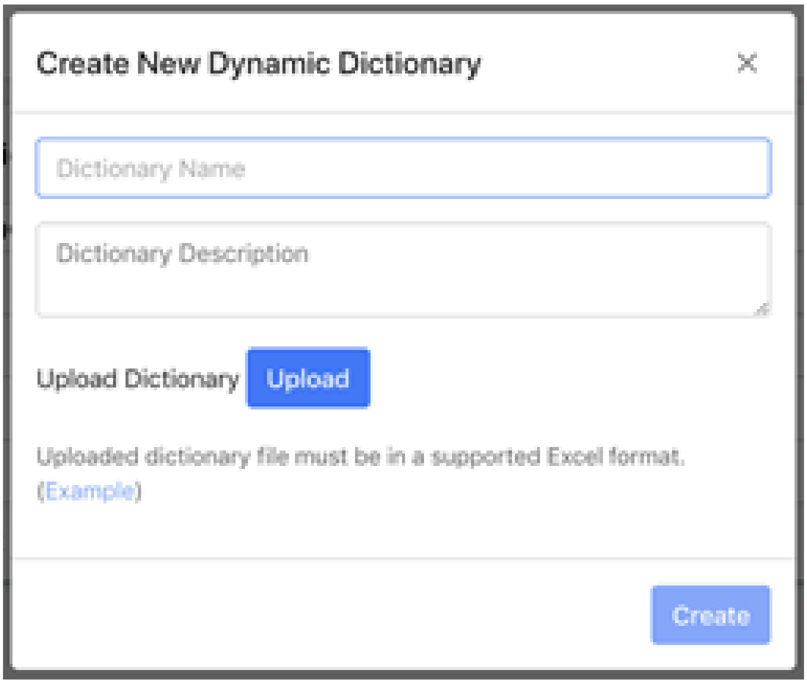
1. To create a Dynamic Dictionary, click the “Create new dynamic dictionary” button



Dictionary Demo

Image	+ Words	Synonyms	Thesaurus
	cat	kitty,kitten	dog, animal, pet
	computer	box,machine,cpu,laptop	memory
	honda	car,suv	subaru
	sky		

2. This button opens the “Create New Dynamic Dictionary” dialog:



Uploaded dictionary file must be in a supported Excel format. (Example)

Here the user can enter the Name and Description for the new dictionary, as well as#upload a dictionary from an external file.

IMPORTANT NOTE: The format for Dynamic Dictionaries is NOT in#a native#Excel format, but in CSV. It is very important that no XML or XLS file is uploaded.#When the above example is downloaded, it is a CSV file.

A#correctly-configured#dynamic dictionary input file looks like this:

example_dynamic_dict.csv

Animal,Place,Says

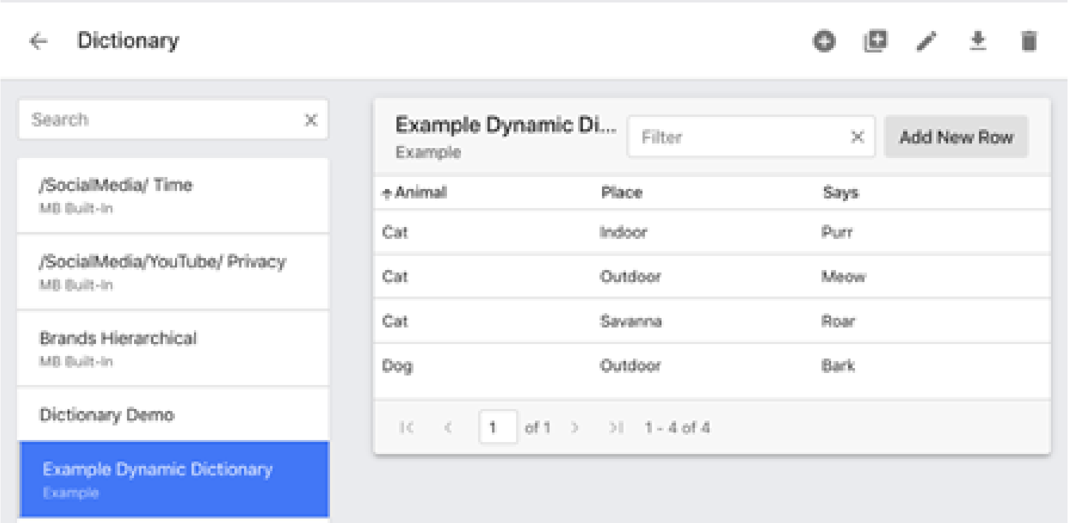
Cat,Outdoor,Meow

Cat,Indoor,Purr

Cat,Savanna,Roar

Dog,Outdoor,Bark

When this file is used to create a#Dynamic Dictionary, the following is created:



Dictionary

Search

/SocialMedia/ Time
MB Built-In

/SocialMedia/YouTube/ Privacy
MB Built-In

Brands Hierarchical
MB Built-In

Dictionary Demo

Example Dynamic Dictionary
Example

Example Dynamic Di...
Example

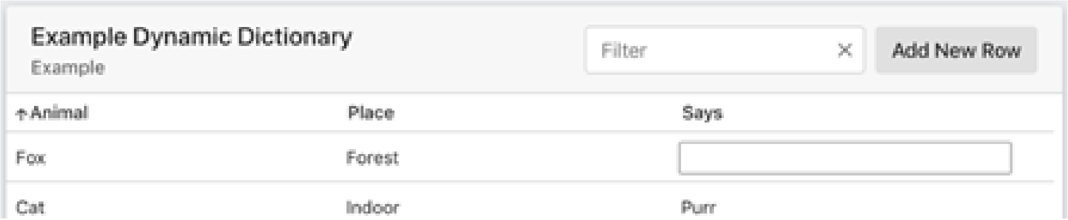
Filter

Add New Row

Animal	Place	Says
Cat	Indoor	Purr
Cat	Outdoor	Meow
Cat	Savanna	Roar
Dog	Outdoor	Bark

1 of 1 1 - 4 of 4

Individual rows may be added#by clicking the “Add New Row”#button, and#filling out a value for each column.



Example Dynamic Dictionary
Example

Filter

Add New Row

Animal	Place	Says
Fox	Forest	
Cat	Indoor	Purr

- Columns can only be added when uploading#a#dictionary input file.

Dynamic Dictionary Best Practices

There are three#major#information theory#paradigms for creating a#Dynamic Dictionary.

- Chained#Terms: Each#value on#row of values will be unique,#occurring#on no other row or column.

Card#Type,#WidthMM,#HeightMM,#WidthIN,#HeightIN

Poker,#63.5, 88.9,#2.5, 3.5

Micro,#31.75,#44.45,#1.25,#1.75

- Semi-Chained: Some values may be repeated, but these imply no#hierarchy in information on each row.

Animal,Place,Says

Cat,Outdoor,Meow

Cat,Indoor,Purr

Cat,Savanna,Roar

Dog,Outdoor,Bark

- **Implied Hierarchy:**#Arbitrary terms may be used to build a#taxonomic#(nested)#hierarchy.#This is done by#using each column to define#“clades”#of values#that describe#ever increasingly specific#description.

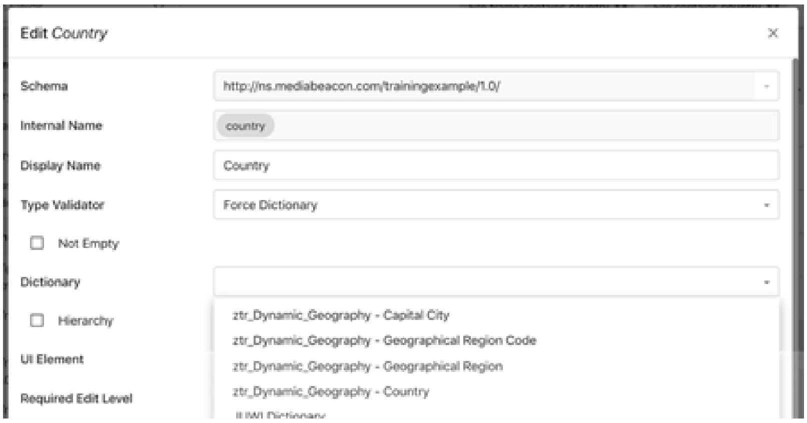
Common Name, Domain, Kingdom, Phylum, Class, Order, Suborder, Family, Subfamily, Genus, Species

Domestic Cat,#Eukaryota, Animalia, Chordata, Mammalia, Carnivora,#Feliformia, Felidae,#Felinae, Felis, F.#catus

Dynamic Dictionary Field Configuration

Once a Dynamic Dictionary has#been set up, fields will need to be configured to take advantage of this#dictionary.

When a Dynamic Dictionary is created, the Edit Field#dialog will show#list that#dictionary#not as a single#entry, but#split into as many sub-dictionaries#as there are columns in the Dynamic Dictionary.



Edit Country

Schema: http://ns.mediabeacon.com/trainingexample/1.0/

Internal Name: country

Display Name: Country

Type Validator: Force Dictionary

□ Not Empty

Dictionary:

□ Hierarchy

UI Element

Required Edit Level

ztr_Dynamic_Geography - Capital City
ztr_Dynamic_Geography - Geographical Region Code
ztr_Dynamic_Geography - Geographical Region
ztr_Dynamic_Geography - Country

To make use of these sub-dictionaries, create and configure#a field to correspond to each of them. Note that these#dictionaries are not hierarchical,#and that checkbox must remain**unchecked**.

The recommended configuration for Dynamic Dictionary fields is Type Validator:#“Force Dictionary”#& UI Element:#“String”.#Multi-value fields are not supported, so any of the Type Validator’s#“Comma Separated” formats are incorrect configuration.

Form Configuration

As always, in order to use fields, they must be included on a Form. This is no different than adding any other#fields to Forms, and they can be arranged in any order, as there is no#explicit#hierarchy#insofar as fields themselves are concerned.

It is worth noting that, as always,#with MediaBeacon#Forms,#the#property#settings of#a#field can be#abstracted from#its#“base” settings in the Edit Field dialog. So, one option for configuring Dynamic Dictionary fields is to change the Type Validator#and dictionary#assignment in the Form itself to allow fields to use the Dynamic Dictionary functionality without changing base field configuration. If used in this fashion, be sure to only use fields configured as Type Validator: String and UI Element: String

3.5.2. History Dialog

This section is under construction.

3.5.3. R3Search Dialog

Management of the Solr Admin Interface.

The Menu Bar > Manage > R3Search command opens the Solr Admin UI. MediaBeacon automatically configures this interface, so there are no changes that need to be performed in this interface to change MediaBeacon function.

For reference, the documentation for this interface can be found at: *https://lucene.apache.org/solr/guide/6_6/using-the-solr-administration-user-interface.html*

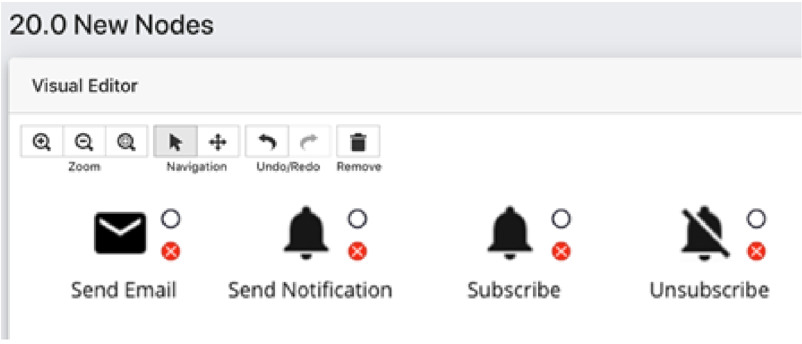
3.5.4. System Status Dialog

A dashboard for various system performance interfaces.

This section is under construction.

3.5.5. Workflow Manager Dialog

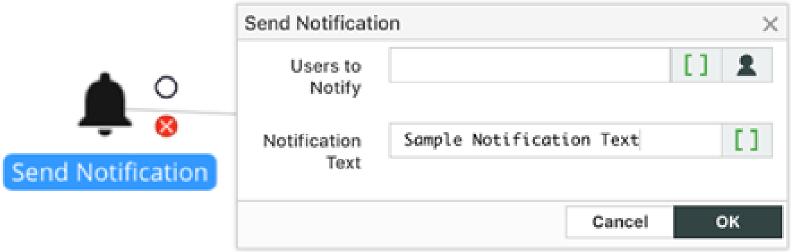
Visual Workflow Nodes



Send Email

In previous versions of MediaBeacon#this node#was named “Send Notification”. In 20.0, its name has changed to reflect that it creates emails, distinct from#the “bell” style notifications of 20.0

Send Notification

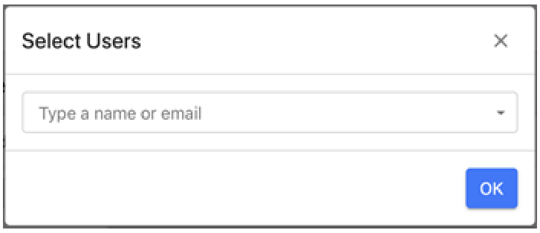


Users to Notify

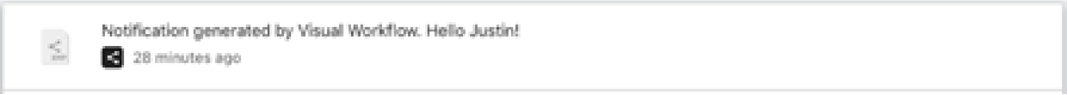
Notification Text: Sample Notification Text

This node allows a visual workflow to send a notification to one or more users as part of the workflow.

- **Users to Notify**: Allow#one or more#recipients to be specified. Note if the#Smartname#"[User.Username]" is used, the user who initiated the workflow will receive the notification. The#User icon button#opens#the Select Users#dialog to choose current active users ofMediaBeacon



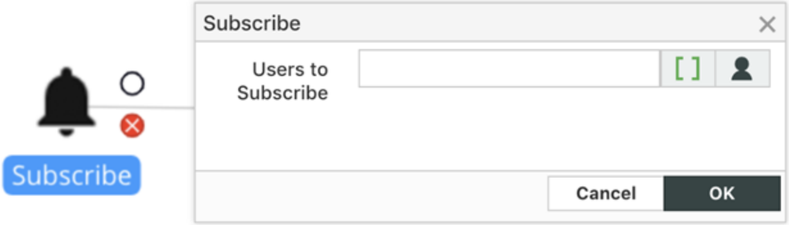
- Notification#Text: Defines the text to send in the notification.

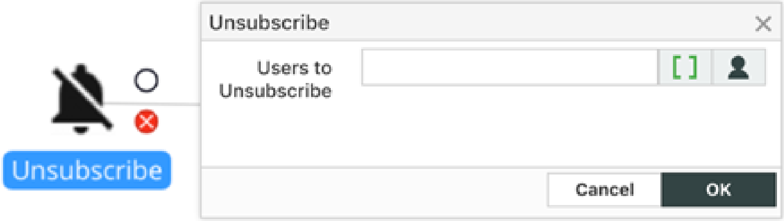


Notification generated by Visual Workflow. Hello Justin!

28 minutes ago

Subscribe and Unsubscribe





These nodes allow an asset that is running through the workflow to be subscribed or unsubscribed#to/ from the listed user or users.

3.6. Global Default Icon

The default icons shown in various interfaces may be changed with a set of system properties:

3.6.1. Global Icon Key

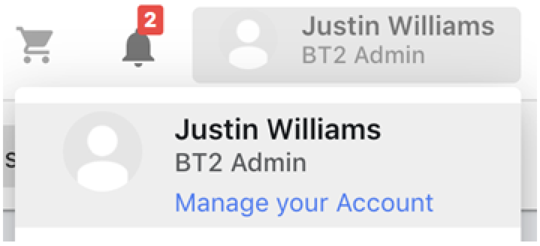
- "Home" button on workspaces. This still may be overridden by workspace "Home" icon assignments.
- Login Component Logo.
- External Links Logo

These may be replaced with a logo resident in the Icon Picker, and implemented with a system property: "mb.global.icon.key=UUID".

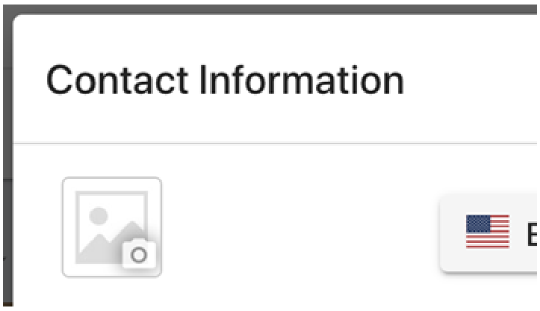
Example: mb.global.icon.key=05155BD9-3A93-4AB3-8025-13872B0079D3

To implement:

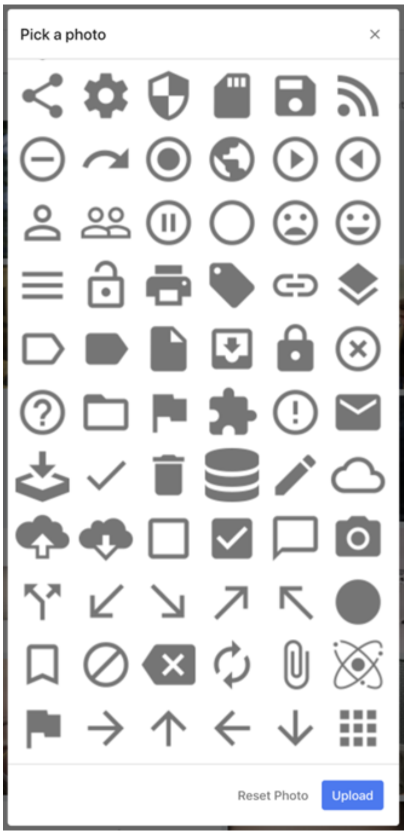
1. Open the User Menu and select "Manage your Account"



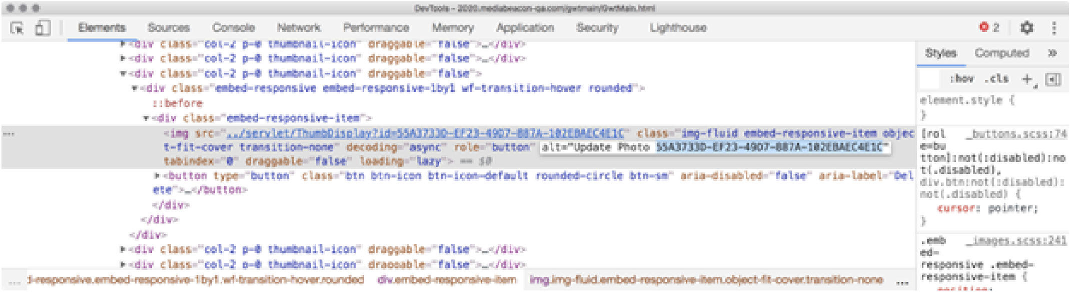
2. Click the picture icon in the left. This will open the Icon Picker.



3. In the Icon picker, add a new icon by clicking "Upload?". A PNG image of a minimum of 252 px X 45 px is recommended.



4. Once the icons is upload and visible in the dialog, open your browser's developer tools menu (usually Shift-Right Click) and choose "Inspect".



```
        ▸<div class="col-2 p-0 thumbnail-icon" draggable="false">…</div>
        ▸<div class="col-2 p-0 thumbnail-icon" draggable="false">…</div>
        ▾<div class="col-2 p-0 thumbnail-icon" draggable="false">
          ▾<div class="embed-responsive embed-responsive-1by1 wf-transition-hover rounded">
              ::before
            ▾<div class="embed-responsive-item">
...             <img src="../servlet/ThumbDisplay?id=55A3733D-EF23-4907-887A-102EBAEC4E1C" class="img-fluid embed-responsive-item objec
                t-fit-cover transition-none" decoding="async" role="button" alt="Update Photo 55A3733D-EF23-4907-887A-102EBAEC4E1C"
                tabindex="0" draggable="false" loading="lazy"> == $0
              ▸<button type="button" class="btn btn-icon btn-icon-default rounded-circle btn-sm" aria-disabled="false" aria-label="Del
              ete">…</button>
              </div>
            </div>
          </div>
        ▸<div class="col-2 p-0 thumbnail-icon" draggable="false">…</div>
        ▸<div class="col-2 p-0 thumbnail-icon" draggable="false">…</div>
 ... l-responsive.embed-responsive-1by1.wf-transition-hover.rounded   div.embed-responsive-item   img.img-fluid.embed-responsive-item.object-fit-cover.transition-none   ...
```

```
element.style {
}
[rol     _buttons.scss:74
e=bu
tton]:not(:disabled):no
t(.disabled),
div.btn:not(:disabled):
not(.disabled) {
    cursor: pointer;
}
.emb     _images.scss:241
ed-
responsive .embed-
responsive-item {
```

5. Copy the UUID as listed above.
6. In the mediabeacon.properties and catalina.properties files, add the System Property as listed above, substituting your UUID.

3.7. Single Sign On

If Single Sign On is configured in the Control Panel Gui App (Server-Side), the user will not be able to reset their password in the Web UI.

4. Admin Core

This section covers front-end Administrative functions in MediaBeacon.

What is the Admin Core?

MediaBeacon can be configured on three levels.

1. The Web User Interface, or (Web UI) may be configured by a user with the correct permissions, usually a Global Administrator-level user..
2. The application's installation itself is configured via the MediaBeacon Server GUI Application Manual.
3. The Admin Core.

The Admin Core interface is itself a secondary, internal MediaBeacon which uses the metadata of specialized data-only assets to define system parameters of Users, Groups, Fields, Forms, and Loading Dock Types. Each of these areas of system configuration has its own Workspace.

Nomenclature

For clarity's sake, this section will use the following terminology to differentiate between some similarly named concepts:

Because the Admin Core itself is an instance of MediaBeacon, there can be some confusion of nomenclature. In this section, we will use the following terms to differentiate similarly named points of configuration

- **Web UI**: MediaBeacon for non-admin users. Admins may configure Workspaces and Components here.
- **Admin Core**: Allows Admins to configure the parameters of Users, Groups / ACLs, Fields, Forms, and Loading Dock Types.
- **Assets**
  - **DAM Asset**: An asset that is managed by the WebUI.
  - **Configuration Asset**: An asset managed by the Admin Core that defines parameters of WebUI behavior
- **Fields**
  - **XMP fields**: refers to the fields that are used to store data about assets uploaded by users in the WebUI.
  - **Property fields**: The fields that define the parameters of configuration assets. Also refers to fields presented in workspace or component configuration.
  - **Form fields**: XMP fields displayed in a Form.
  - **Configuration value**: The values entered into property fields. Often the names of various other configuration assets will be used as values in property fields.

Admin Core Database

MediaBeacon requires two databases to function properly, one for the WebUI part of MediaBeacon, and one for the Admin core. The WebUI database can have any name, but the Admin Core database must be have the string "_admin" postpended to the WebUI name. For example "mediabeacon" and "mediabeacon_admin".

See the Database Users section in the MediaBeacon Server GUI Application section for more info.

4.1. Admin Core Interface Overview

Using the Admin Core

The Admin Core is only accessible to Global Administrator-level users. Once such a user is logged in, they can access the admin core via the Menubar > Manage > System command.

Workspace Structure

The interface of the Admin Core is much the same as the WebUI, but tailored to a subset of functions: creating assets, editing metadata, Quick Search, search filtering and the List view. The following are interface elements shared across all workspaces of the Admin Core.

Menubar Functions

Many of the menus here have reduced and contextual functionality compared to the normal WebUI.

[image: ac-menubar]

- **"Left Arrow" Icon**: Returns to the user to the WebUI.
- **MediaBeacon Logo**: Clears all search criteria
- **Navigate**: Changes to any of the 5 preset workspaces in the Admin Core.
- **Create**: This menu houses commands to create configuration assets of the type shown the current workspace. When used, this will display the appropriate Create dialog to make the current workspace's type of configuration asset.
- **Selection**:
  - **Duplicate**: Duplicates a configuration asset, only works with user account assets.
  - **Delete**: Standard functionality.
- **Bulk Edit**: Allows metadata changes to be made to multiple configuration assets at once.
  - [image: ac-bulkedit]
  - **"Required" Fields**: When using Bulk Edit in the Admin Core, some fields may say "Required" but this is ignored by the Bulk Edit function.
  - **Limitations**:
    - **Fields**: "Schema" and "Internal Name" should never be edited with Bulk Edit.
    - **Make Stamp**: This function does not make a stamp in the Admin Core.
  - **Data Blocks**: Several configuration assets make use of the Data Block metadata structure, which has limited ability to be used with Bulk Edit. Specifically, Entire data blocks (and

constituent field values) can be added to multiple assets, but constituant fields can't be subsequently edited with Bulk Edit. Only whole data blocks can be added or removed. Examples of these structures:

        - **Groups**: Each ACL is a data block.
        - **Forms**: Each Field is a data block.
        - **Loading Dock Types**: Platform Properties, Image Formats Properties, Rotate Properties, Scale Properties, Resolution Properties, Intent Properties, Source Colorspace Properties, Conversions Properties, Video Formats Properties, and Script Options Properties are all data blocks.
    - **Select All:** Standard functionality.
- **Manage**: Usually only shows the System Status command, usually only needed to display the logs for the Admin Core itself.
    - In the User workspace, also shows the Invitations command.
- **Username**: Name of the current user.
- **Help** Icon: Redirects the user to MediaBeacon University, MediaBeacon's online documentation.
- **Quick Search / Search Status components**: Functions as normal, but advanced search is disabled.

Search, Filter and View Assets

On each workspace, the view of configuration assets is restricted to just those pertaining to that workspace. Users can further search for configuration assets using Quick Search, and/or the search filter.

The List view on the right, displays results of search and filter functions. The total number of results is displayed on the right side of the search status.

Filters at the top of the List view may be used to further filter or sort results.

See the Quick Search, Search Status, Search Filter, List View and Filter Headers sections for more information on these functions.

Create and Edit Dialogs

The Admin Core does not use the Asset View as the WebUI does, instead each workspace will display a dialog tailored to the creation and editing of that type of configuration asset.

These dialogs are unlike regular Forms in that they have some dynamic fields, that is, some options may reveal other configuration fields.

Also, saving behavior is different. All changes must be saved using the lower right-hand button (the label changes, Create/Update Asset Type). If the user clicks outside of the dialog itself, the gray surrounding area, the dialog will close without saving changes.

4.2. ACL Configuration

Configuring Access list in MediaBeacon.

- MediaBeacon

  ACL Metadata is contained in a group configuration asset's metadata, but for the sake of clarity it is described here.

  ```
  ACL Metadata is in the Data Block metadata structure.
  ```

  - **Name**: The label of the of the current ACL. Note that general users do not see the names of groups in the ACL menu, only the string designated in this field.
  - **Default ACL checkbox**: Enabling this option automatically copies this ACL to the parent group's Default ACL field.
  - **Description**: A note area to describe the ACL. This string is only displayed in this field and in the ACL Selection component.
  - **Root Path**: This field designates a "root folder" to filter assets available to ACL users. When set, the user of the ACL will only have access to assets that are contained in that folder and subfolders. Can be used in conjunction with the "Search Filter" setting, below.
    - This is noted in the WebUI as "Index/", in the Directory Browser, Move/Copy dialog.
    - "Index" does not always refer to an ACL Root Path. See [The Index Folder] Article for more information.
  - **Hide**: When enabled, this setting will prevent this ACL from being listed in the ACL menu or the ACL List in the Places Browser. This may be desirable in the case of an IdP pooling all users into a common group.
  - **Icon**: Defines the icon displayed for this ACL in the ACL menu and the ACL Selection component.
  - **Workspaces**: This field lists workspaces this ACL can access. The user cannot add workspaces here, but they can be removed.

  Note: If a workspace is removed and it is not shared by any other ACL, the workspace is permanently deleted.

  Search Filter: This field designates a saved search to filter assets available to ACL users. Similar to the Root Path, this will restrict the user's access. However, instead of the restriction being the contents of a folder, the assets returned by the search are available to users.

  - Can be used in conjunction with a Root Folder, as described above.
  - The Search Filter dropdown lists all non-private saved searches to be used as an ACLs search filter. Best practice is to create "Admin Searches" for use in ACL configuration. See the [???] section for more information.
- **Upload Form**: This field allows a specific Form to be designated as the default upload Form for all workspaces in that ACL. Note that the Upload Form designated for a workspace or component will override this setting.
- **Javascript on Download**: Client-side Javascript entered here will be executed prior to allowing the download of a file. This option is usually only required for add-on products.
- **Permission Restrictions**: Entries in this field suppress the specific functionalities for users of this ACL. This field is multivalue, and can clear all or add all restrictions with the buttons below it. Please see the [Permissions Restrictions List] article for full detail on specific restrictions.

Note: If a permission restriction is listed in this field, the functionalities associated with it are **disabled** for users of the ACL.

- **Permission Workflows**: This field designates an advanced workflow that can automatically and dynamically set permission restrictions for the ACL.
    - This allows a server side script to be called to check if the permission should be enabled or not. For example, you could have a conditional permission based on metadata or call an external rights management system.
- **Conditional Download**: This field allows a saved search to be used to restrict download of assets based upon metadata.
    - Assets that are the returned by the designated search will be downloadable, all others will still be accessible but cannot be downloaded.
        - The "View Original", "Crop", and "Check Out" commands are also unavailable in this case.
    - Best practice is to create "Admin Searches" for use in ACL configuration.
- **Visible Forms**: This field defines what Forms are available in the Bulk Edit or Asset View interfaces.
    - Changed from previous versions of MediaBeacon, newly-created Forms **are not** automatically added to all ACLs Forms are now "opt-in" within the ACL configuration.
- **GWT JS**, **GWT CSS**, and **GWT Body**: Code entered here will be added to all served pages in MediaBeacon.

Editing ACL Settings

As ACLs are metadata contained in a group, ACLs are edited within a group's Create/Edit dialog. Each ACL section no longer has an independent "Save" button, **all changes to all of a group's metadata** is saved when the "Update Group" button is pressed.

4.3. Changing Type Validator and UI Element Settings with Forms

Changing Type Validator and UI Element Settings with Forms.

Forms allow abstraction of an XMP field's parameters, to alter the function of that field as displayed in that Form. Below are some well-known combinations.

- **Upload Form Auto Metadata:** Using this combination will enforce automatic metadata entry without the user's interaction or knowledge on an upload Form. Using this method, a specific metadata value will be entered whenever an upload is made with that Form.
    - XMP Field: Any Type/UI combo
    - Form field settings: Use UI element "Hidden", and enter a value in the "Default" parameter field. The entry must be valid for the type of field. For example, a valid dictionary value if the field is a Force Dictionary field.
- **Subset Dictionaries per Business Unit**: These techniques can be useful if a different sets of users need to enter exclusive sets of values (dictionaries) in a single XMP field. An example of this might

be a "Flavor" field. The first of the two business groups that use the field makes baby food and the second makes pet food. Thus, the dictionaries used by each group need to be different. To avoid having to create business-specific fields, we can use one of the techniques below.

  - **Master Dictionary & Subsets**: In this scenario, the XMP field in question will be a Force Dictionary field, with a "master" dictionary attached. When this field is used on a business-group specific Form, it will be paired with a business-specific "subset dictionary". The master dictionary must contain at least all terms contained in all subset dictionaries, to avoid write errors.
  - **Subsets Only**: In this scenario, a non-dictionary XMP field would be used (Type Validator: String). The Forms themselves ensure no business group user will enter an inappropriate term within their individual context. The major advantage here is that this eliminates the need to maintain a set of master/subset dictionaries. Metadata administrators may find the Search Filter's drag-and-drop tagging feature useful for metadata editing in this case.
  - **Auto Updating Suggest Dictionary & Subsets**: This scenario uses several configuration points in conjunction.
    - Create a "Master" Dictionary, and connect it to a Suggest Dictionary XMP field.
    - Enable the "Synchronize Dictionary to Field" option on that master dictionary.
    - Create subset dictionaries as normal, however, there is no need to add all terms into the master dictionary.
    - Add the XMP field to Forms, but change Type Validator to Force Dictionary, and use a subset dictionaries as needed.
    - This method will cause the master dictionary to auto-populate with terms as they become tagged into assets.
- **View-Only Fields**: While simple to set up, this technique makes it easy to control who can enter metadata and when on specific fields.
  - The UI element "View Only" isn't recommended for use with XMP field configuration, only Form field configuration. This would create an uneditable field in the former case.
  - Do not use "View Only" in conjunction with the "Not Empty" option. This will cause the Form to prevent any edits to all displayed fields.
  - On an upload Form, a View Only field can be used with the "Default" value to show users values that are being auto-added.

4.4. Field Type and UI Settings

A guide to XMP field Type Validator and UI Element settings.

4.4.1. Type Validators

An XMP field's type validator setting controls what type of data is able be entered, preventing invalid date entry.

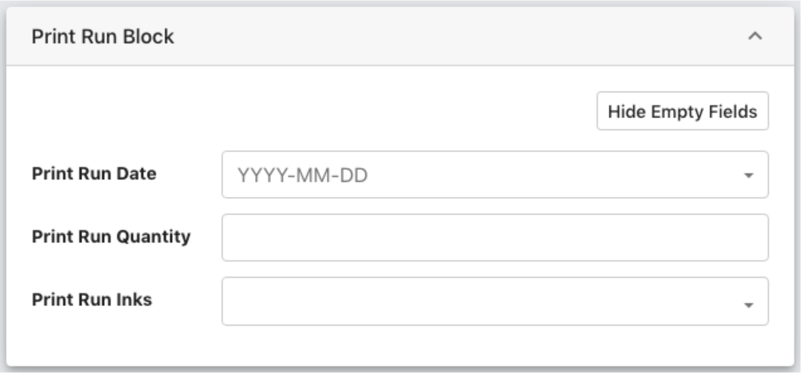
- **None**: Effectively the same as String, used by legacy MediaBeacon configurations to denote a field that had just been created.
- **String**: Field will allow UTF-8 characters.

- **Date**: Allows only ISO 8601 date format data (YYYY-MM-DD, 1984-10-20).
- **Force Dictionary**: Allows a dictionary to be chosen to restrict entries to the values in the dictionary.
- **Suggest Dictionary**: Allows a dictionary to be chosen to suggest values to be entered in the XMP field.
- **Hierarchy**: This option enables hierarchical functionality for dictionary fields. This option is set manually and the dictionary that is configured with the dictionary field must be formatted as a hierarchical dictionary.
- **EMail**: Validates the string entered has a single "@" character with at least 1 letter on preceding and following it.
- **Phone**: Allows digit characters and "-" or "." characters.
- **Integer**: Allows a positive or negative integer value. The value must be between -2147483647 and 2147483647. Only digits and the minus sign ("-") may be entered. This is a 32-bit signed integer.
- **Positive Integer**: Allows a positive or negative integer value. The value must be between 0 and 2147483647. Only digits may be entered.
- **Real Number**: Allows entry of a decimal number.
- **Positive Real Number**: Allows entry of a positive decimal number.
- **Large Number**: Allows a positive integer from 0 to 9223372036854775807 (2^63). This is a 64-bit signed integer.
- **Asset Reference**: Validates that an entered string matches a currently extant asset's Asset ID / Record ID. Makes a field CSV even if set to String in UI element. A field with this type validator must have the "Index SQL" option enabled to operate correctly.
- **Data Block**: See the Data Block section, below.

Data Blocks

Data Block fields (aka Repeating Data Block fields) are a data structure that allow a set of fields to be added in a repeating fashion. This is similar in function to multivalue fields, but include several field values for each repeating instance.

The data block uses another Form as a "template" of fields to repeat, using the settings inherent to that Form. This is assigned in the "Data Block" property field in the Edit Field dialog, similar to setting a dictionary for a Force or Suggest Dictionary field.



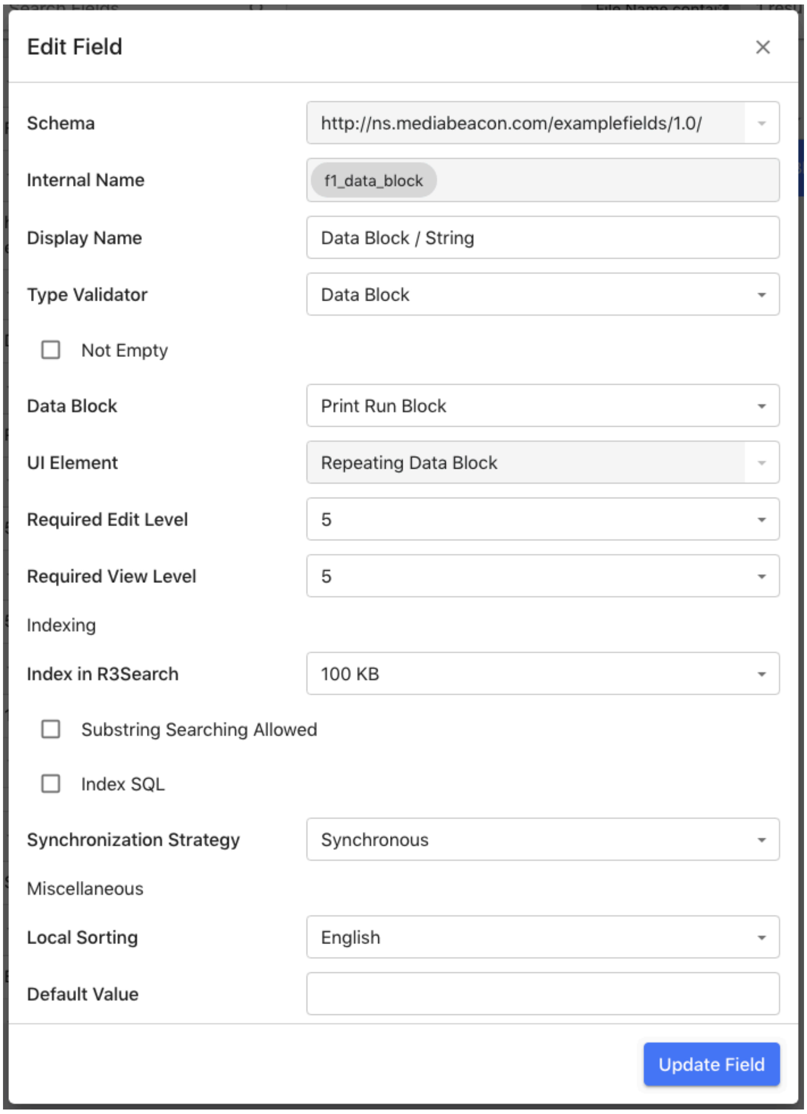
Print Run Block

Print Run Date: YYYY-MM-DD

Print Run Quantity:

Print Run Inks:

This Form is the element that will be repeated for the Data Block field.



Edit Field

Schema	http://ns.mediabeacon.com/examplefields/1.0/
Internal Name	f1_data_block
Display Name	Data Block / String
Type Validator	Data Block

□ Not Empty

Data Block	Print Run Block
UI Element	Repeating Data Block
Required Edit Level	5
Required View Level	5

Indexing

Index in R3Search	100 KB

□ Substring Searching Allowed

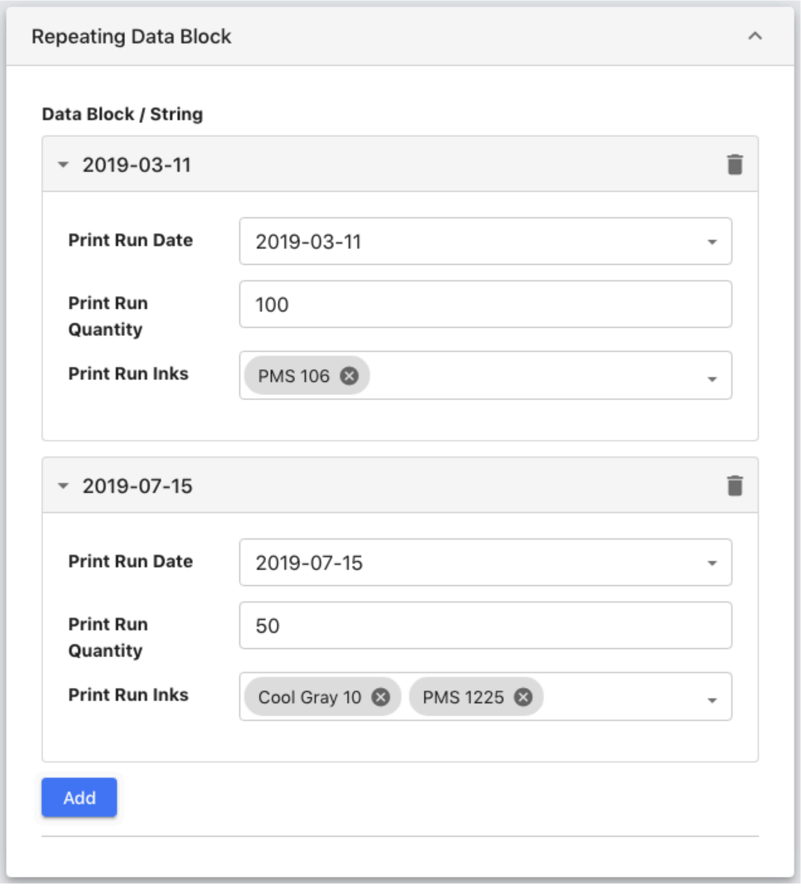
□ Index SQL

Synchronization Strategy	Synchronous

Miscellaneous

Local Sorting	English
Default Value	

The Field configuration for the data block field.



Repeating Data Block

Data Block / String

2019-03-11

Print Run Date: 2019-03-11

Print Run Quantity: 100

Print Run Inks: PMS 106

2019-07-15

Print Run Date: 2019-07-15

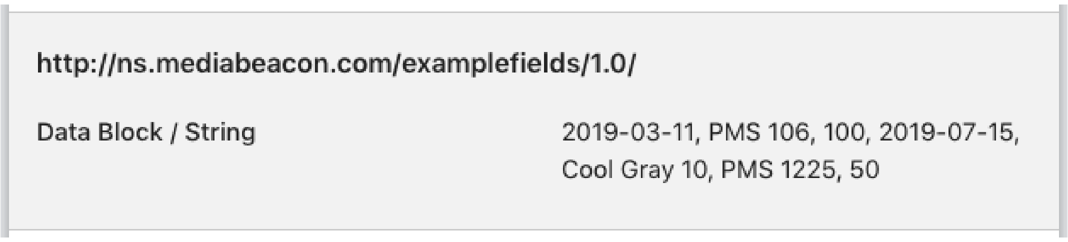
Print Run Quantity: 50

Print Run Inks: Cool Gray 10, PMS 1225

A Form containing a Repeating Data Block.

Data Block Data Structure & Search Filtering

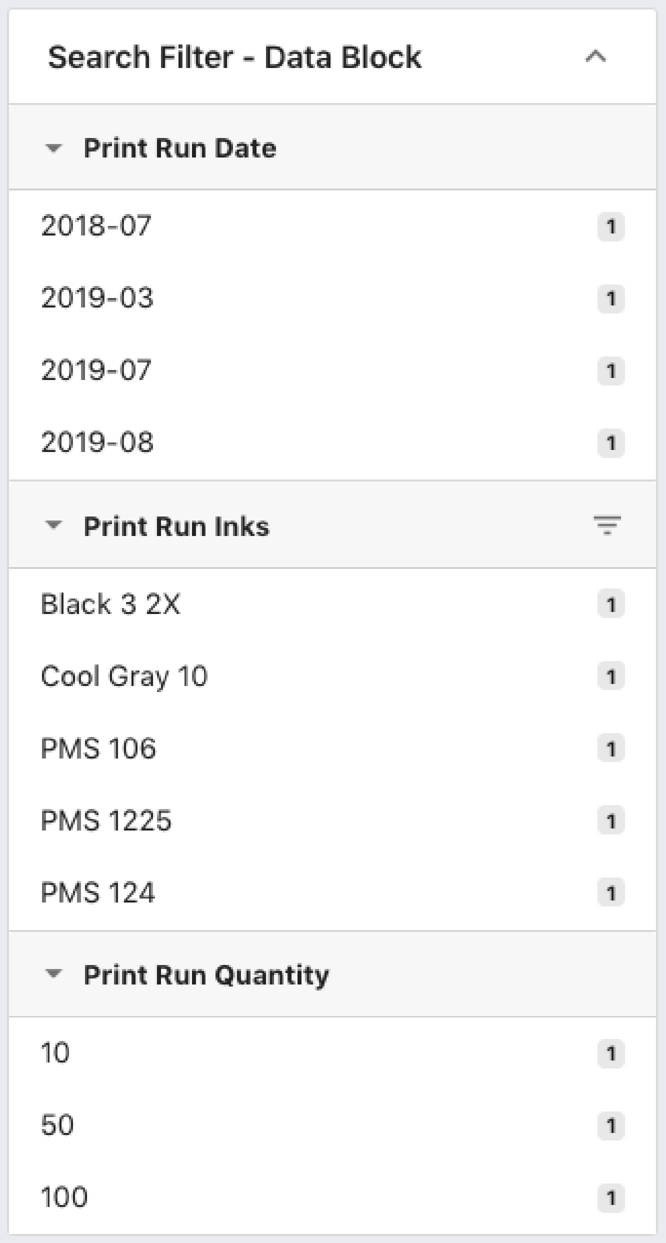
The values written by a Data Block are not copied to the fields that compose the Form used for the "Data Block" property field setting, instead being written as a strided value.



http://ns.mediabeacon.com/examplefields/1.0/

Data Block / String: 2019-03-11, PMS 106, 100, 2019-07-15, Cool Gray 10, PMS 1225, 50

This data is displayed by the Search Filter as though the values were written to the fields in that "template" Form.



Search Filter - Data Block

Print Run Date

2018-07	1
2019-03	1
2019-07	1
2019-08	1

Print Run Inks

Black 3 2X	1
Cool Gray 10	1
PMS 106	1
PMS 1225	1
PMS 124	1

Print Run Quantity

10	1
50	1
100	1

Data Block Configuration

- Using the "Data Block" Type Validator, the UI Element is automatically set to "Repeating Data Block" and the Synchronization Strategy is automatically set to "Synchronous".

Using Data Blocks

- Data block fields are not displayed directly in Search Filter components. Instead, the Search Filter considers the data to be written to the "source" fields.

- When a data block field values are written, all other field values entered using that Form are written Synchronously.

4.4.2. UI Elements

A field's UI element setting controls how it's displayed in a Form. However, some UI element settings directly control data entered or underlying data structure.

- **String**: No special UI, basic data. Displays a single line of text, prevents line-breaking characters from being entered via keyboard.
- **Textarea**: Displays an expandable text-entry XMP field element. Entered line-breaking characters are displayed as breaking strings.
- **Cascade**: Displays hierarchical dictionary XMP fields as a cascade element.
- **Comma Separated Cascade**: Displays multivalue hierarchical dictionary XMP fields as a cascade element.
- **Date**: Displays a date picker on the right side of the field element.
- **Checkbox**: Displays the field element as a checkbox. When the element is enabled (checked), the value of the field will be the string "true". When the element is disabled (unchecked) the value is null.
- **Comma Separated Values**: Allows the field to contain multivalue string data.
- **Comma Separated Dates**: Allows the field to contain multivalue date data.
- **Comma Separated Textarea**: Allows the field to contain multivalue string data that can contain line-break characters.
- **Repeating Data Block**: This UI element is automatically selected when the Type Validator "Data Block" is selected.
- **Simple HTML**: Allows entry of html text. This element uses a rich text editor.
- **Advanced HTML**: Allows entry of html text. This element uses a raw text editor, and a preview of rendered HTML.
- **URL**: This acts as both a validator and as a UI element setting. The string entered must start with "http". Saved values are presented as a clickable hyperlink.
- **Starred Rating**: Allows a starred rating to be entered.
- **Code**: Displays the XMP field value in a code editor.
- **View Only**: Prevents data entry into an XMP field when displayed on a Form. Best practice is to use this setting only in Form configuration.
- **Hidden**: Prevents display of the field when included in a Form . Best practice is to use this setting only in Form configuration, and is often paired with a value entered in the "Default" property field.

4.4.3. List of Valid Combinations

Type Validator / UI Element

- String / String
- String / CSV
- String / Text Area
- String / CST
- String / Checkbox
- Date / Date
- Date / CSD
- Force Dictionary / String
- Force Dictionary / CSV
- Force Dictionary Hierarchical / String

- Force Dictionary Hierarchical / CSV
- Force Dictionary Hierarchical / Cascade
- Force Dictionary Hierarchical / CSC
- Suggest Dictionary / String
- Suggest Dictionary / CSV
- String / Advanced HTML
- String / Simple HTML
- String / URL
- Email / String
- Phone / Email
- Integer / String
- Positive Integer / String
- Real Number / String
- Positive Real Number / String
- Large Number / String
- Asset Reference / String
- Data Block / Repeating Data Block

4.5. List of Preconfigured Forms

Forms predefined in MediaBeacon.

Industry Standard

- Adobe PDF
- Adobe PDF Properties
- Adobe Photoshop
- Adobe Photoshop Properties
- Adobe Stock Photos
- Camera Data 1
- Camera Data 2
- Dublin Core
- Dublin Core Properties
- EXIF Properties*
- EXIF Properties Additional
- IPTC Contact
- IPTC Content
- IPTC Properties
- IPTC Status
- QuickTime Metadata*
- TIFF Properties
- Rights Management
- Rights Management Properties
- Stock Photo Properties
- Stock Photo Resolutions
- XMP Basic
- XMP Core Properties*

- XMP Media Management Properties

MediaBeacon Defaults

- Approval
- Archive
- Brand Hierarchy
- Campaign Information
- Download Approval
- Facebook Post
- Marketing Fields
- MediaBeacon Default
- MediaBeacon Link
- MediaBeacon MetaData
- MobileApprovalDisplayFields
- Permissions
- Post to Social Media
- R3volution Workflow
- Regions
- Restriction on Download
- Social Media
- Social Media Schedule
- Task Initiation
- Twitter Post
- YouTube Upload

Special Forms

These Forms do not display data as standard

- Advanced XMP: Displays all XMP metadata in an asset, and allows the user to delete an schema of metadata at once. Only Global Administrators have access to this. And they always do regardless of ACL setting.
- Asset Links: Displays MediaBeacon Links.
- Automatic Versions
- Brightcove Cue Points
- Document Report
- Managed Versions
- Used in Assets
- Video Clips
- Video Summary
- Video Transcodes

4.6. Permission Restrictions List

Note: A comprehensive guide to Permission Restriction functionality.

The Permission Restriction field is a multivalue field in each ACL that lists a set of "tags". Each of these define set of functionality that suppressed by inclusion on this field.

The "Add All" button will add all permission restrictions to the list, and the "Clear" button removes all permission restrictions.

Note: When a permission restriction is added to an ACL it PREVENTS a user from using a particular functionality.

- **Add Annotations**: Prevents the user from adding or deleting items in the Discussions Form tab in the Asset View.
  - If the View Annotations permission restriction is enabled, the Discussions tab is not visible, preventing the user from adding discussion items, making this permission restriction redundant in that case.
- **Add Placeholders:** Prevents the user from creating data-only assets. If the File Action permission restriction is enabled, the user will not be able to create placeholder assets, making making this permission restriction redundant in that case.
- **Administer Bundles**: Prevents the user from creating and editing bundle status. Users will still be able to edit metadata in bundles (even bundle fields) if given access to those fields. This is considered an Admin privilege, and the Configure Widgets permission restriction should be set for most users.
- **Administer Dictionaries**: Prevents the user from creating, editing, or deleting dictionaries. This is considered an Admin privilege, and the Configure Widgets permission restriction should be set for most users. Note that when a user can administer dictionaries, they can do so for all dictionaries in the system.
- **Allow Quick Download**: Prevents the user from using the Quick Download feature .
  - Note: If the Download Hi-Res permission is enabled, Allow Quick Download is redundant.
- **Archive / Restore**: This permission restriction is deprecated, and has no function.
- **Change Password**: Prevents users that have local accounts from changing their passwords, contact info, and email settings.
- **Check Out**: Prevents the user from using the Check Out functionality.
- **Check Out Admin**: Prevents Global Administrators from using the "Check In" and "Cancel Check Out" commands and downloading checked out assets.
- **Configure Widgets**: Prevents the user from accessing component's configuration panel. This is considered an Admin privilege, and the Configure Widgets permission restriction should be set for most users.
- **Contact Sheet**: Prevents the user from creating contact sheets.

- **Content Pools Filter**: This permission restriction is deprecated, and has no function.
- **Custom Download**: Prevents the display of the "Custom" tab in the Conversions dialog. If the Download Hi-Res permission restriction is enabled, the user will not be able to use Custom Download, making this permission restriction redundant in that case.
- **Delete Access**: Prevents the user from using the Delete command.
  - If the File Action permission restriction is enabled, the user will not be able to use the Delete Command, making this permission restriction redundant in that case.
- **Desktop Drag and Drop**: This permission restriction is deprecated, and has no function.
- **Download Hi-Res**: Prevents the user from download non-converted original) files. In addition it also:
  - Prevents use of the Quick Download function.
  - Prevents use of the Custom Download tab.
  - Prevents use of the View Original command.
  - Prevents attaching assets to Email.
  - Makes redundant the Allow Quick Download and Custom Download permission restrictions
- **Edit Metadata**: Prevents the user from editing any metadata. This permission restriction ignores field and group Edit Level settings. Note the following caveats:
  - Non-directly editable metadata will still be changed if the user performs other actions, such as uploading a new file. MediaBeacon will add any metadata that is included in the file, as well as metadata such as file size and Modification Time.
  - Does not recapitulate the Change Password permission restriction.
- **Enable Contextual Menus**: Prevents the user from accessing contextual menus. Note that a Global Administrator user will always be able to access the Admin Mode via a contextual menu.
  - When this permission restriction is used, the following commands will be unavailable. Commands that may be accessed elsewhere in the interface are marked with an asterisk (*).
    - Open in New Window
    - Copy Link
    - Quick Download (also available in asset upper-right pulldown buttons)
    - Download*
    - Selection (The main selection menu
    - Add to Cart (Drag, Menubar > Selection > Add to Cart
    - Pinpoint Text*
    - View Original*
    - Replace
    - Crop
    - E-mail*
    - History (also available in the Menubar > Manage menu, if user has access)
    - Check Out, Check In, Cancel Check Out
    - New Version
    - Views
- **File Action**: Prevents the use of the Move, Copy or Delete commands. Also prevents renaming assets, creating data-only assets, and hides the Create Menu .

- This permission restriction makes redundant the Add Placeholder, Delete Access, Rename Asset permission restrictions.
- **File Upload**: Prevents the user from uploading new assets toMediaBeacon. This **does not** restrict the Import, Replace, Check Out, or Versioning functions.
- **Folder Action**: Prevents the user from using various commands on folders. The following commands in the Directory Browser and Places Tree Directory Place contextual menus are disabled.
  - New
  - Rename
  - Delete
  - Move/Copy
  - Duplicate Folder
- **Generate External Links**: Prevents the user from using the External Links command in either the Create or folder contextual menus. Note: This function may or may not be considered an Admin function, and as it allows access to assets without login or authentication, care should be given in determining which users can use External Links.
- **Import Assets**: Prevents users from using the Import command on assets or folders. This is considered an Admin privilege, and the Configure Widgets permission restriction should be set for most users.
- **Link Assets**: Prevents the use of the Link Assets functionality. Note that this is a wholly different function from External Links.
- **Manage Workspaces**: Prevents the user from changing workspace settings, as well as adding / removing components, rearranging workspace tab order, setting workspace tab names, and changing the the layout their workspaces. This is considered an Admin privilege, and the Configure Widgets permission restriction should be set for most users.
- **Maximize/Minimize Widgets**: This permission restriction is deprecated, and has no function. This was removed in MediaBeacon 2019.1
- **Rename Assets**: Prevents the user from renaming assets. This permission is redundant when the File Action permission restriction is enabled.
- **Replace Assets**: Prevents the use of the Replace functionality.
- **RSS**: Allows the Global Administrators to have access to RSS functionalities, listed below. In MediaBeacon 2018.x and 2019.x, this functionality is not enabled, so for those versions this permission restriction is effectively deprecated.
  - Manage > "Feeds"
  - Directory Browser "Create RSS Feed" and "Attach RSS Feed"
- **Send E-mail**: Prevents the use of email-related functions:
- **Thumbnail Deletion**: Prevents General Users from deleting icons in the standard icon picker interface. Added in 2019.1.
- **Trigger Workflow**: Prevents a user's actions from triggering custom workflows. Visual Workflows (VWFE) and Advanced Workflows require this privilege to be unrestricted to operate in a given ACL. The Adobe® Connector is prevented from being usable in an ACL when this permission restriction is used.

- **Upload Anywhere**: Prevents uploading to any directory other than the standard directory.
- **Version**: Prevents the use of the Versions functionality. This permission restriction has no bearing on the Check Out functionality, which also creates versions in MediaBeacon.
- **Video Original**: Prevents access to the original video file as opposed to renditions with lower frame rates, more compact formats, etc.
- **View Annotations**: Prevents display of the Discussions Form tab. When this permission restriction is enabled, the Add Annotations permission restriction becomes redundant. Global Administrator users may always use this function, regardless of setting.
- **View Global Asset Log**: Prevents users from accessing the History command.
- **ACL Permission Restriction on External Portal**: This feature is enabled by default for all users of an#ACL but#may hidden by using the "External Brand Portal" permission restriction
- **ACL Permission Restriction on Metadata Template**: This feature is enabled by default for all users#of an#ACL but#may#hidden#by using the#"Create Metadata Template"#permission restriction.
- "External Brand Portal": When added to the "Permission Restrictions" field of an ACL, the ability to create and manage External Portals will be unavailable to all users of that ACL.
-

4.7. Fields Workspace

Defining XMP fields in MediaBeacon

The Fields workspace defines each XMP metadata field in MediaBeacon using a specific field configuration asset and metadata in that asset defines the parameters of the field.

This interface for configuring fields has changed greatly from previous versions of MediaBeacon, as well as some nomenclature used. Notably the concept of **schemas** has changed:

- **No Schema Tabs**: In past versions of MediaBeacon, a schema (aka **URI**: Universal Resource Identifier,) was created as a framework for subsequently creating fields. Current versions of MediaBeacon allow independent creation of fields, although a URI must still be defined.
- **No Auto generated Forms**: Each schema automatically generated a Form, but these were not fully editable. Current versions of MediaBeacon do not autogenerate Forms for fields, preventing a proliferation of unused Forms.
- **No Namemapping Tab**: Fields no longer need to be "namemapped" as each field has a "display name" field.

4.7.1. XMP and Other Metadata Standards

While MediaBeacon offers flexibility for field and Form definition, it must be stressed that XMP nomenclature standards must still be rigorously followed.

Fields also have a large array of possible settings, some of which have implications for downstream data re-use, data design and governance, user experience, or other substantive things.

For these reasons, sections below will list Background and Best Practices for important points.

4.7.2. Field Configuration Asset Metadata

- **Schema**: he identifier that all fields in a schema share.
  - Background
    - In the XMP standard, this is referred to a "namespace URI"
  - Best Practices:
    - Currently is less enforced in MediaBeacon than previous versions, it is generally desirable to have a small number of schema for fields created for private use by an organization.
  - **Example**:

    ```
    http://ns.companyname.com/en/packaging/administrative/1.0/
    ```

    - **Protocol:** Many URI's begin with "http://", but there is no commitment that they be resolvable web resources. The "ns.", therefore denotes this as a namespace, not web page.
    - **Organization Name and top level domain**: "companyname.com/"
    - **Descriptors**: "*/en/packaging/descriptive*" - This can be any number of descriptive identifiers that characterise the use of the schema as a whole, in this example "/en" identifies that the data is in English, "/packaging" identifies the fields are packaging related, and "/administrative" might indicate metadata that is not intended for general users.
    - **Version "/1.0"**: a number that will identify the version of the schema, "1.0" is standard.
    - **Ending slash**: "/" this is mandatory on all schema names.
    - **Allowed characters**: a-z, A-Z, 0-9 (as long as a number is not the first character in a string), hyphen "-", and underscore "_". Spaces are not allowed.
- **Internal Name** : This is the canonical name of the field, which is the data structure written into XMP and the field's name as it is stored internally in the database.
  - Background:
    - In the XMP standard, referred to as the "local name".
    - Technically, fields are properly referenced to using the XML Expanded Name, composed of the namespace URI and the internal name).
  - Best Practices
    - Allowed characters: a-z, A-Z, 0-9 (as long as a number is not the first character in a string), hyphen "-", and underscore "_".
    - Spaces and slashes "/ \"are not allowed.
- **Display Name**: Display name is the default string used to identify a field in the interface, or the name as it is displayed to end users. It can contain any UTF-8 character.
  - Background: This is what used to be referred to as a Namemap or Field Label.
- **Type** (aka Type Validator): This defines the type of data allowed in a field. See the Field Types section for more information on the settings available

- **UI Element**: This setting defines a field's presentation in the interface. See the Field Types section for more information on the settings available.
- **Required Edit Level**: This setting controls a field's edit-ability compared to a user's group's Edit setting. The group setting must be equal to or higher than a field's setting for a user of that group to edit it. Ranges from 0-9.
- **Required View Level**: This setting controls a field's visibility compared to a user's group View setting. The group setting must be equal to or higher than a field's setting for a user of that group to view it. Ranges from 0-9.
- **Index SQL**: When enabled, this option will index this field (all values in all assets) as a SQL table The main benefit of this function is that it would allow the field to be used as a Highlight or as an Import/export Key field.
  - Background: By default, MediaBeacon allows up to 29 "String," 10 "Date," and 10 "Integer"-type fields. This can be increased but requires a modification to SQL database settings. A number of predefined fields in MediaBeacon use indexing, so the numbers above do not indicate the number available for use.
  - Best Practice: There are very few reasons to SQL index a field in current versions of MediaBeacon.
    - This setting allows a field to be used in View component's Highlight config.
    - This setting allows a field to be used as a Key Field in the Import/Export Wizard.
- **Index in Research**: This option allows a field to be searchable in MediaBeacon. The setting here may be 100kb or 1Mb of metadata per file that is indexed.
  - Best Practice: Which is better? What are the implications of each setting? When would a admin know to see to one or the other?
- **Synchronization Strategy**. Will affect how a field's data is written to the SQL database and into Asset XMP.
  - Best Practice: Always keep set to Asynchronous. The only exceptions are: custom fields
- **Display Order**: The number here will determine the display order of the field. This field has no function other than when defined in a Form, and is not inherited when a fields is placed in a Form.
  - Best Practice: Safe to leave blank.
- **Locale Sorting**: Changes rules for the alphanumeric sorting of a field's data in the Web UI, according to the sorting rules for the specified language.
- **Default Value**: A value entered here will automatically be pre-filled to the field whenever it is blank in a Form.
  - Best Practice: This is best left blank in Field configuration, to be used primarily in Form configuration.
- **Substring Searching Allowed**: When enabled, this option allows a field's value to be searched on a substring level
  - Background:
    - Without this option, MediaBeacon's standard behavior is to break space delimited data into "whole words". So, normally, a search for "great" will return an asset that had a Keyword of "super great".
    - Enabling the substring option is only needed when it is desirable to return an asset containing the value "aubergine" when a search for the strings "gine" or "berg".
  - Best Practice:

- While of moderate impact on most fields, enabling this option on a field that contains paragraphs of text over hundreds of thousands of files may create unnecessary load on the system.

4.7.3. Predefined XMP Fields

Note: It is highly recommended to leave as is the settings of predefined fields in MediaBeacon. These may have an internal function that require specific settings to operate.

4.8. Groups Workspace

Configuring Groups and ACLs in MediaBeacon.

The Groups workspace in the Admin Core allows admins to create and edit groups and ACLs.

Groups Overview

Groups in MediaBeacon are organizational objects that users can be assigned to. Each group can impart specific privileges to users, such as access to certain functionalities, or access to specific assets. Conceptually,MediaBeacon groups are roughly analogous to the groups that an identity provider (IdP) defines.

ACL Overview

MediaBeaconadds an additional layer of configuration for flexibility: the Access Control List or ACL. ACLs control a key set of users' capabilities in the system. ACLs are not inherited from Identity Management systems, being a MediaBeacon-exclusive configuration.

ACL configuration will be covered in the Configuring ACLs article, but organizationally, they are subordinate to groups.

Group Asset Metadata

- **Name**: The label used for the current group.
  - The Group name is usually not visible to users, except when sharing a Saved Search or Saved Selection.
- **Hidden?**: Enabling this option hides the group in the Group workspace's List View. When groups are hidden, the Search Filter will display the Hidden facet, and clicking True will show the hidden groups. This setting is often used when an IdP imports a large number of unused groups into the system.
- **LDAP DN**: Although labeled "LDAP", this field inherits the Domain Name that is supplied when importing a group from an IdP.
- **Description**: A text description of the group, only visible in the Admin Core.

- **Login Message**: This field is used to add text to the Login component shown on the login workspace. It has a number of limitations:
    - As this is a group-based property, it will only be displayed after a user logs out from a specific group.
    - A record of what group the user was in when they logged out is stored in a browser cookie.
    - If a user has never logged into MediaBeacon, or has removed browser cookie, this message will not be displayed.
- **Users**: Displays the current members of the group. Users may also be added to a group by adding their usernames here.
    - This is bidirectional with a user's group metadata. When a user is added to or removed from this field, this group will be added or removed from the user's Group Membership field. If the user has only one group (that one group being present in both Primary Group and Group Membership), the "Default" group will be automatically added.
- **Loading Dock Type**: Assign the loading dock type for the group. All ACLs within this group will share this setting. See the [Loading Dock Type] section for more information.
- **Loading Dock Quota**: Set the maximum number of megabytes that a group user's Loading Dock can hold. All ACLs within this group will share this setting. See the [Loading Dock Type] section for more information.
- **Default Language**: Sets the default language the WebUI will use upon initial login. All ACLs within this group will share this setting. See the [Languages] section for more information.
- **Edit Level and View Level**: These settings control what metadata fields can be edited or viewed by users in a given group. The levels range from 0 (lower) to 9 (higher). All ACLs within this group will share this setting.
    - These levels carry no inherently different functions, instead being relative to the corresponding settings defined for each field. If a group's Edit level is 5, all users (using that group at the time) will be able to edit fields at levels 0 through 5. The same goes for the View level, a field is visible to a user if the ratings correspond.
    - Edit Levels are overridden by the Edit Metadata permission restriction. When set, the user cannot write metadata.
    - It is important to remember that the view and edit levels do not *provide* access as much as they *limit* it. These settings are of limited use if the user is not first granted the ability to see a given field via the Visible Forms setting.
    - Below are some examples of caveats with these levels. The examples assume the field edit/view levels are set higher than the user's current group's levels.
        - Fields on a Forms will be non-writable or hidden, respectively.
        - List View columns (defined by fields) will not be hidden by View level. Edit level will prevent writing to these fields.
        - Fields used in the Search Filter will be hidden.
        - A field used in the Taxonomy component is always visible and editable.
- **Priority**: A ranking system of 1-20 (20 being the highest) to determine a user's "login destination", that is, the Group, ACL and workspace the user is automatically directed to upon login.
    - The system also ranks groups differently based upon the Default ACL setting. If this field is blank, that group will always be considered lower priority than any group that has a defined

Default ACL. For example, a group with Priority 1 and a Default ACL defined will be the login destination even if that user has access to a group with Priority 20 and no Default ACL.

- **Default ACL**: The ACL to which the user is directed upon login by default if there is more than 1 ACL. This is contingent upon the user's group's Priority setting.
- **ACLs** section: This area lists the ACLs that have been defined for the current group. Click the "Add" button to add an ACL section. See the ACL Metadata article for information on these settings.
- **PDF Title Template:** This section allows the user to set a cover page for PDF files generated by the Contact Sheet functionality.
  - Note that all ACLs in a group will share the came cover page.
  - See the **How to Create a Custom Contact Sheet Cover Page** article for more information.

4.8.1. Managing Groups with an IdP

When using an IdP with MediaBeacon, group management will have some differences.

Upon user authenticates against the IdP:

- A user account configuration asset is created if it does not already exist. The user account configuration asset will then sync metadata the IdP supplies.
- Group configuration assets are created for all groups of which the user is a member, if they do not already exist. Those group's configuration assets will then sync metadata the IdP supplies.
- The metadata synced to groups is limited: "Name" is always present, "LDAP DN" maybe supplied, and any users who have previously authenticated who are part of the IdP group will be listed in the "Users" field. The IdP supplies no other metadata fields or ACLs, so these will need to be added in MediaBeacon.
  - Pre-Existing MediaBeacon groups can be synced to an IdP after they have been created. All that is required is to match the names of theMediaBeacon groups to those of the corresponding IdP groups **prior to user authentication**.
- User's membership in groups must be managed in the IdP system.

User / Group Assignment with IdP

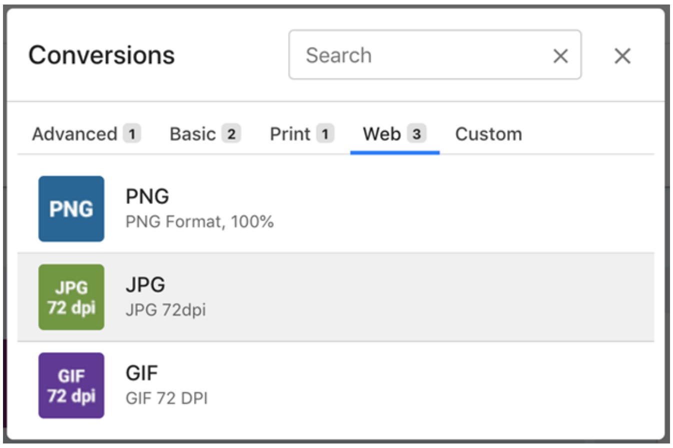
In default configuration, MediaBeacon does not have the ability to change the assignment of users to groups, as this is wholly managed via the IdP itself.

Using an optional configuration, MediaBeacon can be allowed to edit user membership of IdP groups.

If an ACL is not configured for a group, the user will be shown a dialog: "Warning, No ACLs available. Please contact your administrator."

4.9. Loading Dock Types Workspace

Ability to categorize and display preset loading dock conversions based on intended use.#This is set in the Loading Dock#Type, in each conversion preset.



Conversions

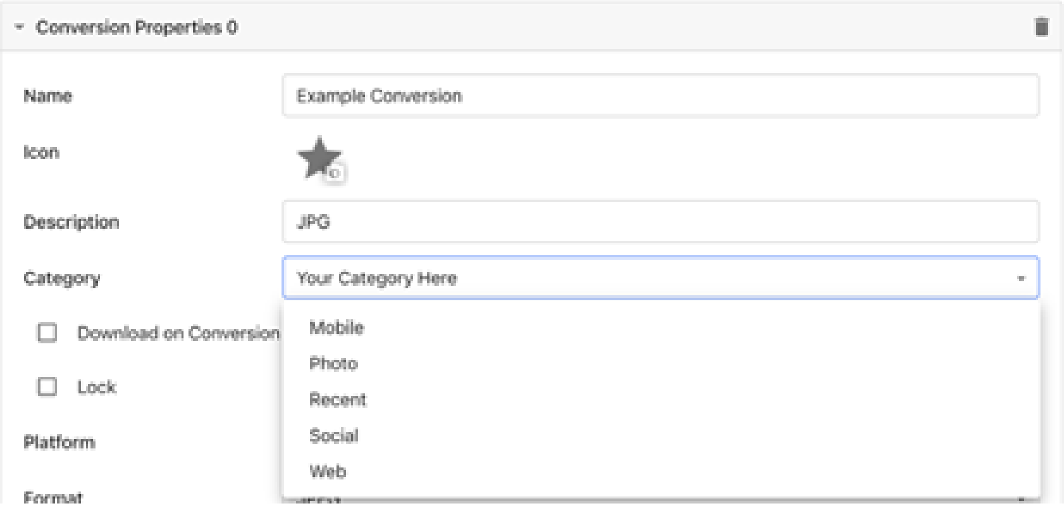
Search

Advanced 1 Basic 2 Print 1 Web 3 Custom

PNG
PNG Format, 100%

JPG
JPG 72dpi

GIF
GIF 72 DPI



▾ Conversion Properties 0

Name: Example Conversion

Icon

Description: JPG

Category: Your Category Here

□ Download on Conversion

□ Lock

Platform

Format

- In MediaBeacon 20.2, the “Loading Dock Quota” has been removed from the Group Configuration. This changes how Conversions (AKA Loading Dock files/assets) are retained in the system.
- Conversions will be removed from the Loading Dock automatically after 7 days, or when their notification is deleted

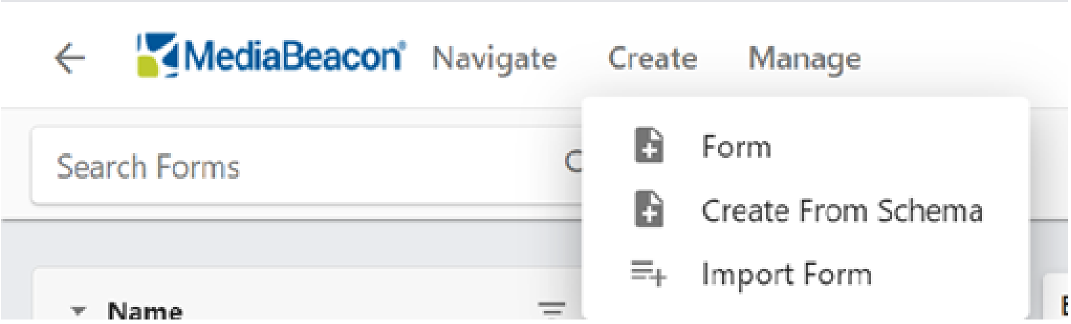
4.10. Forms Workspace

How to Configure Forms in MediaBeacon?

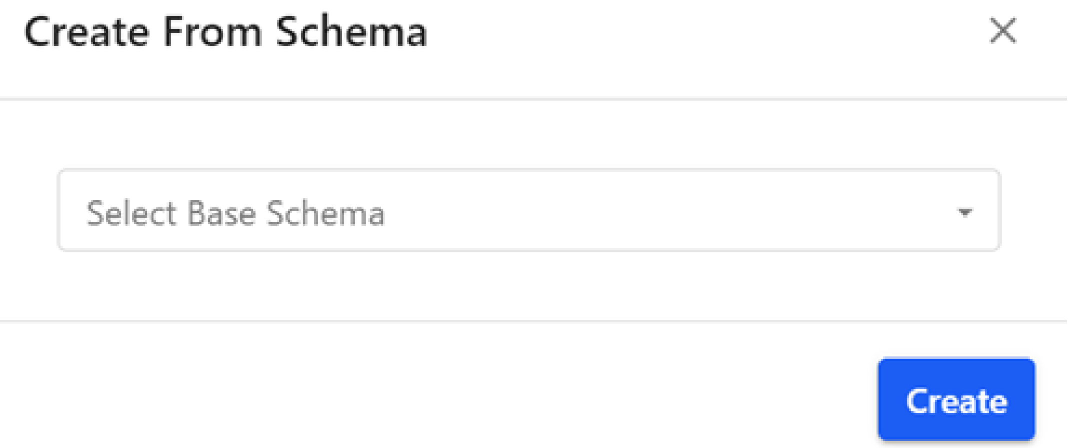
A Form is MediaBeacon's primary way to display metadata. Many predefined Forms are built-in but many will be created by administrators for specific customer use cases. Once XMP fields have been defined in the "Fields" tab, Forms are built using a list of XMP fields, each displayed with separately configurable properties in spindown tabs. The XMP fields that a Form uses may be from multiple schema, and may also display or operate differently than their "base" configuration.

1. Manage > System

While Creating forms using the “Create from Schema” option, in order to eliminate redundancy, choose a base name before the Create from Schema option.



2. Create -> Create from Schema -> Select Base Schema



Note:

Choose unique names to avoid confusion.

- **Create Form:** Displays the Create New Form dialog, creating a new Form when all required property fields have been filled.

- **Create From Schema:** New in MediaBeacon 2019.x, this command allows all current fields that share a schema to be automatically added to a new Form.

4.10.1. Form Configuration Asset Metadata

Name

The name of the Form. Primarily seen in the Asset View and the Bulk Edit dialog, for end users, but is referenced by the Admin Core in several places, primarily, the Visible Forms field in ACL configuration. When a Form's name is changed, it will need to be reassigned to:

- ACL "Upload Form" field.
- ACL "Visible Form" field.
- Workspace Config: "Default Asset View Form" field.
- Workspace Config: "Default Asset View Tabs" field.
- Workspace Config: "Required Forms for Upload" field.
- Upload HTML component configuration dialog: "Required Forms for Upload" field.
- Form component configuration dialog: "Select a Form" field.
- Infinite Scroll View component: "Forms Required for Upload Linking" field.
- Tile View component configuration dialog: "Forms Required for Upload Linking" field.
- List View component configuration dialog: "Forms Required for Upload Linking" field.
- Video Edit component configuration dialog: "Forms Required for Upload Linking" field.
- Form Stamper component configuration dialog: A stamp that uses fields in a renamed Form will need to be deleted and recreated if edits to that field need to be made.

Class

- This CSS class name is applied to the HTML element containing the Form. Note the Form tab itself is not classed via this setting.

Enable Grid View

- When enabled, the Form is displayed in grid view.
  - Requires that only "Comma Separated" fields be used in the Form configuration.
  - Requires that each of the values contain the same number of entries in each multi value field.
  - When an Form is in this view, values cannot be edited.
  - This type of metadata governance is not encouraged as a database rebuild could change the order of the values. Best practice is to use data block functionality.

Fields Section

This area displays "Form fields" as rows of spindown panels. These may be an included XMP field or a "static" row, containing only text. Each of these rows contain a number of property fields to define the Form field. Click the "Add" button to add a Form field section.

[admincore—addfields]

Field Name

- This field allows the user to choose the XMP field used for the current Form field. XMP fields are listed by Display Name, Schema, and Internal Name.

Display Name

- The name of the XMP field as it will be displayed within the current Form.

Static

When enabled, this option changes the field row to display as a static line of text.

- **Display Name**: Text entered here is displayed as line of text instead of a XMP field.
- **Static**: This option toggles off the static mode.
- **HTML**: When enabled, this option displays the value in "Display Name" as rendered HTML.
- **Order**: Operates as described below.

Not Empty

When enabled, this option makes the field "required". When used in a Form, a required field prevents all XMP fields on that Form from being written until the required field is filled.

Field UI Options

See the *Field Type and UI Settings* on page 59article for a list of these settings.

- **Type Validator**: See the *Field Type and UI Settings* on page 59article for a list of these settings.
- **Default Value:** A string entered here will be automatically pre-populated into the Form field when viewed.

Order

This property field accepts integer values that determine the ordering of Form fields on a Form.

- These values do not reorder Forms fields in the Edit Form dialog, only when a Form is displayed in the WebUI.
- Form fields do not inherit the "Order" value from XMP field properties, being null when added. Form fields, therefore, will display in the order they were added unless this value is set.
- Form fields with a null Order value are assumed to have a order value that matches their "initial order", that is, or the position corresponding to the order in which Form fields were added in the Edit Form dialog. For example, the fourth added Form field is automatically ordered as the fourth field in the Form when viewed in the WebUI.
- If a Form field is given an order value that matches a null field's initial position, the null value field will be ordered before the non-null field.
- Fields that have the same Order value (null or specified) will be ordered in the order they were added to the Form.
- Best practice is to number fields with large values (100, 200, 300, etc, so that rearranging Form fields in the future can be done easily.

Edit Form dialog fields	Form View
Field A (Order: null)	Field A
Field B (Order: 4)	Field D
Field C (Order: 4)	Field B
Field D (Order: null)	Field C

Javascript

Forms can execute client-side JavaScript based on different events, which can be used to build panels that update dynamically based on user activity. These events can trigger for an entire panel or individually by field.

- **On Show**: Triggers upon the Form field being called to be displayed.
- **On Focus**: Triggers when the field becomes the focus of user input, usually by tabbing or mousing to the field.
- **On Blur**: Triggers when the field loses input focus.
- **On Change**: Triggers when a field's value changes.
- **On Load**: Triggers when the Form as a whole is loaded.
- **On Submit**: Triggers when the Form as a whole has been submitted to write changed values.
- **On Post Save**: Triggers when the values the Form submitted have been written.

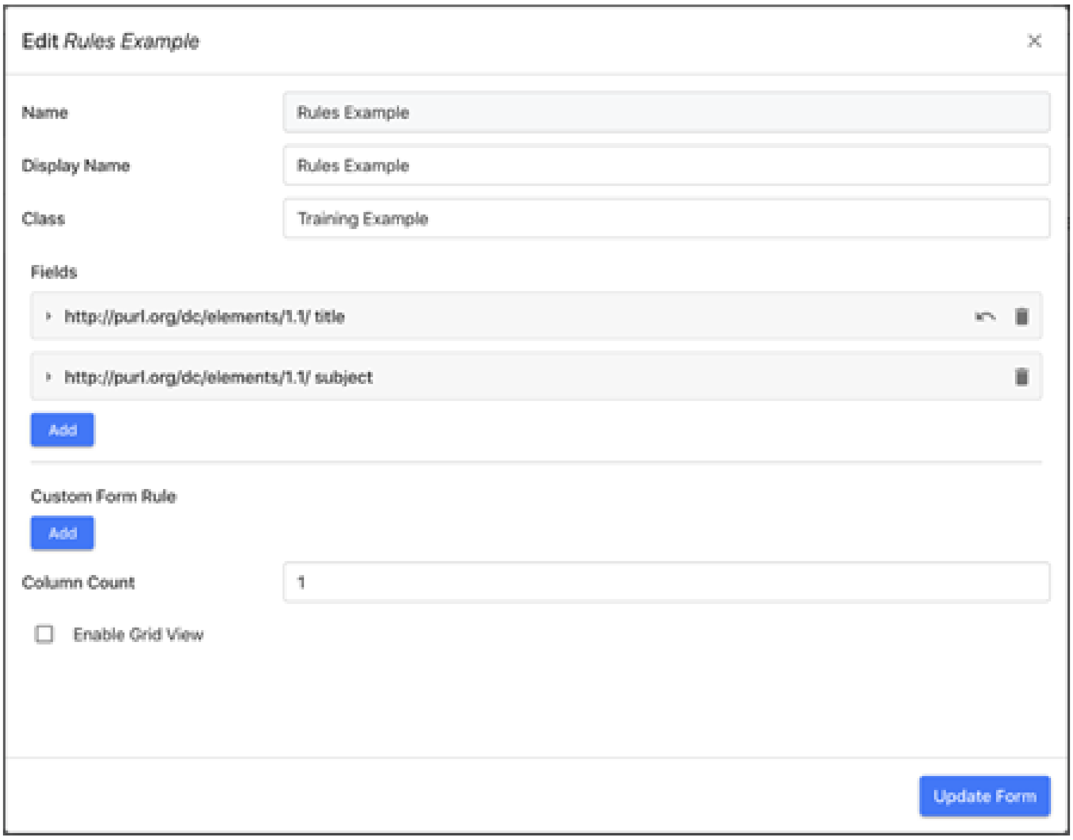
Column Count

The value entered here determines the number of columns used to display Form fields in the WebUI.

4.10.2. Create Custom Forms

This feature is a refinement of MediaBeacon's JavaScript framework in Forms, making many#commonly-requested#dynamic field configurations easy to implement with no JavaScript coding.

The Edit Form dialog now has a new section for Custom Form Rules. It will only be shown when a Form that has existing fields is opened for editing.



Edit *Rules Example*

Name: Rules Example

Display Name: Rules Example

Class: Training Example

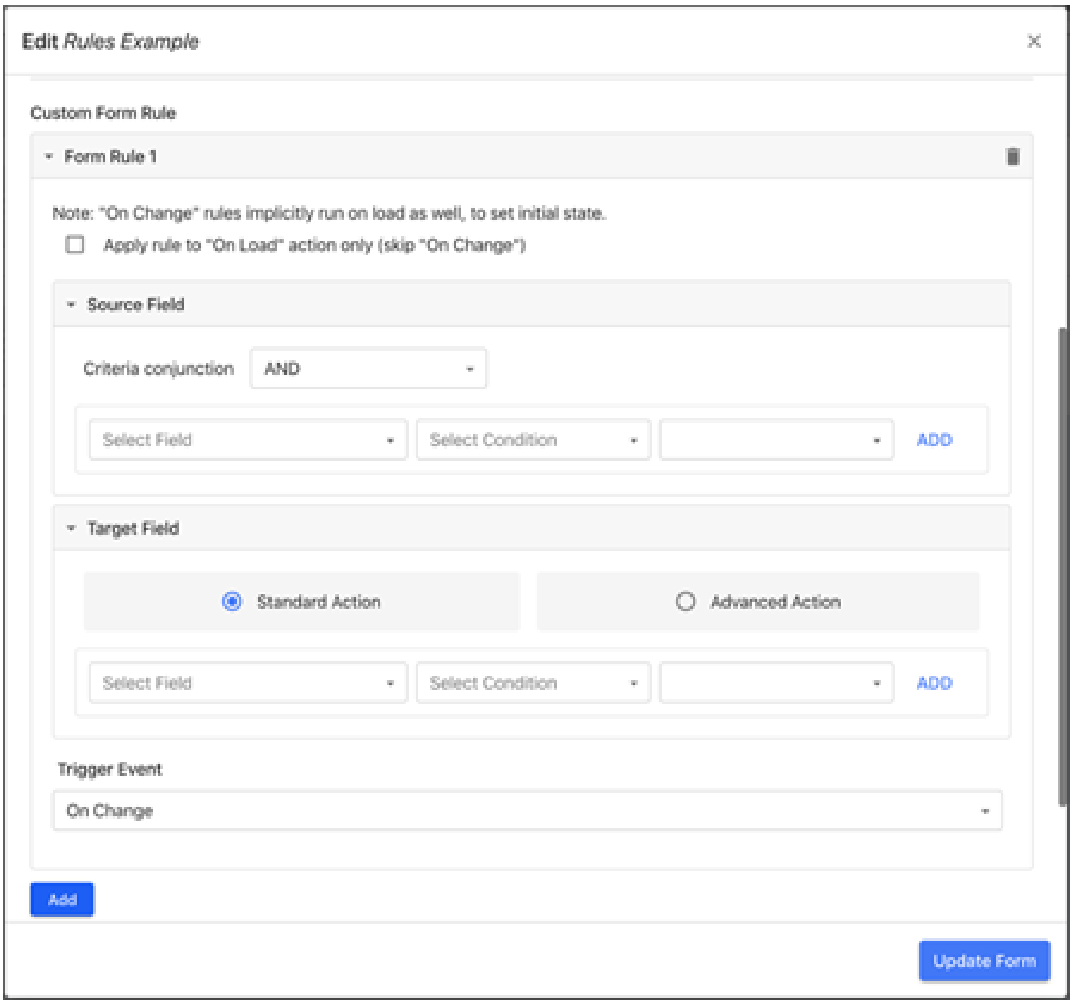
Fields

- http://purl.org/dc/elements/1.1/ title
- http://purl.org/dc/elements/1.1/ subject

Column Count: 1

□ Enable Grid View

- Clicking the “Add” button below “Custom Form Rule” will add a new rule.



Edit *Rules Example* ×

Custom Form Rule

▾ Form Rule 1

Note: "On Change" rules implicitly run on load as well, to set initial state.

□ Apply rule to "On Load" action only (skip "On Change")

▾ Source Field

Criteria conjunction: AND ▾

Select Field ▾ | Select Condition ▾ | ▾ | ADD

▾ Target Field

● Standard Action ○ Advanced Action

Select Field ▾ | Select Condition ▾ | ▾ | ADD

Trigger Event: On Change ▾

Add

Update Form

Feature Specialities of Custom Form Rules

- The behavior of "On Change" rules running on load was added.
- The option to “Apply rule to "On Load" action” was added.
- “Trigger Field” renamed to “Source Field”
- “Action” is relabeled “Target Field”
- Trigger pulldown no longer contains “On Load”, that option becoming a separate checkbox.

Custom Form Rules Interface

Source Field

This subsection defines what field values must be present for a custom rule to be activated. The interface is the same as that used for Advanced Search criteria with the exception that the “Select Field” pulldown will only show the fields that are currently defined in the Form’s “Fields” area.

Target Field

This subsection has two main options, Standard Action and Advanced Action.

- **Standard Action**: This interface allows the user to designate one or more#currently-defined#Form fields (and a condition for each) to be activated then the source field criteria are true. Below is a list of Standard Action Conditions and condition values.
  - **Append Value**(enter a string): This action adds the designated string to the existing value of the target field.
  - **Replace Value**#(enter a string or null value): This action replaces the existing value of the target field with the designated string. A null value will erase values in the target field.
  - **Add Static Text**#(enter a string or a null value): The string here will be displayed as informational text below the target field.
  - **Set Visible**#(choose "true", "false", or "match filter"): If true, the target field will be visible. If false, it is hidden.
  - **Set Invalid**#(choose "true", "false", or "match filter"): If true, the target field will be marked as invalid and prevent saving the Form. If false, it is normally writeable.
  - **Set Not Empty**#(choose "true", "false", or "match filter"): If true, the target field must not be null to allow saving the form. If false, the field may be left empty.#Note that the "Required" label is not shown when this condition is in the true state.
  - **Set Enabled**#(choose "true", "false", or "match filter"): If true, the target field may be written to. If false, it will be non-editable.
  - **"Match Filter"**: This option for all the "Set" conditions above allows the true/false status of the source field to activate/deactivate that given condition. In practice, it means that these conditions will not need to have two rules to define a toggleable condition, such as hiding / showing a field when it contains only a specific value
- **Advanced Action**: This allows client-side JavaScript to be entered as the action to perform upon the source field criteria being true.

Trigger Event

The "Trigger Event" pull-down sets the event that triggers the Rules be executed on a form.

- **On Change**: When a user changes field values while using a Form containing a custom rule. This is the most used trigger type.
- **On Submit**: This trigger type only shows a target field, and as such is most useful for "Append Value" or "Replace Value" actions. These actions only occur when the Form is saved.
- **On Post Save**: This trigger only allows Advanced Actions to be executed, occurring after other metadata writes have concluded.

Other Trigger Event Options

- Implicit On-Load behavior: This feature will automatically execute the "On Change" trigger once when loading the form, in order to use on change events to set an initial state, especially of "Set" conditions.

- Apply rule to "On Load" action only (skip "On Change"): This option, when checked, changes a "On Change" triggered rule to only occur when the Form is opened.

Overridden Set Conditions

"Set" conditions replicate some field#settings, but#are#superceded#by field property settings. If a field has the following:

- UI Element set to “Hidden”, the “Set Visible” condition cannot make the field visible.
- An invalid value is entered in a field for the given Type Validator, the “Set Invalid” condition cannot not make it a valid value.
- “Not Empty” checkbox checked, the “Set Not Empty” condition cannot make the field non-required.
- UI Element set to “View Only”, the “Set Enabled” condition cannot make the field writeable.

The best practice when toggleable field states are required is to avoid configuring the base field configuration (or the fields’ Form properties) to the restrictive property options, and instead use Custom Form Rule logic to enforce those properties.

4.11. Users Workspace

Configuring user accounts and invitations in MediaBeacon.

4.11.1. User Account

User Account Overview

User accounts in MediaBeacon control the user's ability to log into the system, define broadly their level of capability, but rely upon other configurations (Groups, ACLs) to control the fine detail of user permission.

For an explanation of the interactivity between different types of configuration assets, see the Configuration Interactivity article.

User Account Workspace List View

- A user account's active status is indicated by a grey flag in the upper left of its row.
- User accounts may be sorted by: Username, User Type, Group Membership, First Name, Last Name, and Email.
- User accounts may be filtered by: Username, User Type, Primary Group, Group Membership, Company, Department, Position, Country, State/Region, City, and Active.

User Account Asset Metadata

- **Username**: The string that the user enters to authenticate.
- **Password**: The password that the user will enter to authenticate.
  - Passwords in local accounts do not have length or character restrictions by default. However these can be configured, see the [???] article for more information.
- **Active**: When selected, this option makes the account active, that is, able to log in.
- **Change password on next login**: When enabled, this option requires the user, once authenticated, to choose a new password that cannot be the same as the initial password. This is useful if an admin is asked to manually reset a password, making the password given to a user a one-time password. Once the password has been changed, this option will be disabled.
  - ***Note: Both of the Active? And Change password check boxes can be checked concurrently.***

- **User Type**: One of three user types may be chosen here, General User (most users), Group Administrators (seldom used) and Global Administrator (configuring users). See the [User Type] section below for more information on these types.
- **Group Membership**: This multi-value field defines the list of groups to which user has access. It is a dynamic dictionary field, only allowing currently existing groups to be added.
  - The group entered into Primary Group will automatically be entered into Group Membership.
  - This is bidirectional with group's Users field. When a group is added to or removed from this field, this username will be added or removed from that group's Users field.
  - **Loading Dock Type**: Assign the loading dock type for
- **Primary Group**: This field is an artifact of legacy systems, and the group listed here has no special function or bearing on the account. There are a few quirks about how it is used in the Create / Edit dialog:
  - It is a required value, and the value here is automatically added to the list of groups in Group Membership.
  - When a user is created, it is auto-populated with the "users" group. This is usually not desirable. See the *Groups Workspace* on page 73 section for more information.
  - The best practise to use when populating a user:
    - Single group: the group to which the user belongs will be in Primary Group and Group Membership fields.
    - Multiple groups: All groups will be in Group Membership, but one of them (it does not matter which) must be in Primary Membership.
- **User Account Contact Fields**: These fields have limited functionality within MediaBeacon, and do not need to be entered for an account to be valid.
  - **First Name**: If filled, the string here will be displayed in place of the username field when the user is logged in.
  - **Last Name**: If filled, the string here will be displayed in place of the username field when the user is logged in.
  - **Company**
  - **Department**
  - **Position**
  - **Address Line 1**
  - **Address Line 2**
  - **City**
  - **State/Region**
  - **Zip Code**
  - **Country**

  - **Phone Number**
  - **Phone Extension**
  - **Email**: This field must be filled out, otherwise the system cannot send email to them.
- **Content Pools**: This function extends per-user information and metadata search capabilities. See the Content Pools article for more information.

Local vs IdP Users

Local vs IdP Users

User Accounts can be thought of in two major categories. One, local accounts, which are created and managed by the MediaBeacon system itself, and identity provider (IdP) accounts, those controlled by an identity provider that MediaBeacon is configured to use. MediaBeacon integrates with an array of identity providers. See the Enterprise Tab section for more information.

Local Accounts

This article refers to local account functionality in general, and IdP accounts as noted.

IdP Accounts

This section will list those areas where IdP accounts differ from local accounts

- **Username** and **Password**: Controlled by the IdP and are not editable.
- **User Account Contact Fields**: Controlled by the IdP and are not editable.
- Active Setting: An IdP account can be prevented from logging in when its Active option is disabled. The user authenticates as normal via the IdP, but no access is granted to this account.
- **Primary Groups** and **Group Membership**: These fields in an user account are generally controlled by the IdP and are not editable. Configuration options do exist to change this behavior, allowing the IdP to initially populate these fields, with MediaBeacon allowed to make changes

User Type (User Levels)

- **General User**: This type of user has no access to configure the Admin Core, and is granted the ability to configure parts of the WebUI in only very exceptional circumstances. Most users of the system will be this type.
- **Global Administrator**: This type of user has the most power to edit configuration in the system, always having access to the Admin Core, and to Admin Mode. Some group/ACL permission restrictions can limit the abilities of global admin, so it is worthwhile to ensure there is at least one fully unrestricted group set aside for this purpose.
- **Group Administrator**: the group administrator capabilities fall somewhere between the General User and the Global Administrator.
  - Cannot use Admin Mode.
  - Full access to Admin Core.

- For this reason, this user type is not often used. Typically, intermediately permissioned users will be of the General User type, and given access to groups that allow more capability them most other users.

Content Pools

See the Content Pools article for more information.

User Invitations

MediaBeacon's invitation system is a way to quickly on board a number of local users with a minimum of Global Admin effort. An invitation creates an email, link, or activation code that gives a users a self-service method of account creation. One notable difference between invitations and normal user accounts is that invitations are not stored as XMP assets, and as such have limited edit-ability once created.

A note on user count: Users created by an invitation (does not count / counts against) the number if users a given license allows. For more information about license parameters.

Creating an Invitation

Once the **Menubar > Create > Invitation** command, is chosen, the Invitation dialog is shown. It's titled "Activation Code: ".

[image: ac-invitation]

ALERT CALLOUT: This number is both the key for creating accounts, and identifying them for later management so it may be worth copying before filling out the below form.

- **Number of Users**: The maximum number of users that can be created by a given invitation.
- **Email Details**: Sending an email is optional. If so the Activation code or link will need to be copied by hand. It's recommended that global admins send the invitation email to themselves for forwarding even if they are not going to designate recipients initially.
  - **To**: Allows one or more (comma separated) email addresses to receive the invitation email. Unlike other email forms in MediaBeacon, this one does not prefill user accounts / email addresses as it is assumed the recipients are not already in the system.
  - **Subject**: Standard email functionality.
  - **Body**: Standard email functionality. Will also contain the link to complete an account activation when sent.
  - **Default Group**: The invitation system allows the user to choose one group the users will belong to. If more than one group is needed, this may be added to users via normal editing or bulk edit once they have created their account. This field cannot be left blank.
  - **Required Fields**: In addition to a username, the User Account Contact fields may be set to required, for the invited user to supply. It's recommended to include "Email" with every invitation.
- **Create Invites and Send Email** button: The To and Subject lines will need to be filled to send an email properly.

- **Create Invites**: Immediately creates the invitation, displaying the "Email Users" dialog, to remind the admin to send the activation link to users.

ALERT CALLOUT: This is the only time that the link can be copied at this point. Double check it has been completely copied. Clicking "OK" or "X" clears the dialog.

Managing Invitations

Once created, an invitation's status can be reviewed by clicking the **Menubar > Manage > Invitations** command, opening the Invitation User Summary dialog

- **Invitation Name**: Unique Identifier for each invitation.
- **Activation Code**: The code as seen in the "Activation Code" dialog. This is the easiest cross reference to each invitation.
- **Invited Users**: Number of users initially invited.
- **Remaining Users**: Remaining users for the given.
- **"X" (Delete)** button: This will delete the invitation.

Using an Invitation

Via Link

When the user follows an activation link, the user is redirected to the MediaBeacon site:

1. The Create Account dialog is displayed.
2. The user enters required fields and clicks "Create".
3. If the account is successfully created, the "Success" dialog is displayed, the user then clicks "Close" to return to the MediaBeacon login screen.

Via Activation Code

The activation code from an invitation may also be used to create accounts to be used, but the "Create Account" link option must be enabled in the Log In component on the Log in workspace. For more information on how to enable this function.

To use, the user clicks the "Create Account" link, displaying the "Enter Activation Code" dialog.

1. The user enters the activation code, and clicks "Create". If the activation code is valid, the steps of account creation above in "Via Link" are followed.

INFO CALLOUT: The activation may also be postponed to the following string to make an activation link: "*http://externaltraining2.mediabeacon...tMain.html#!CU*"

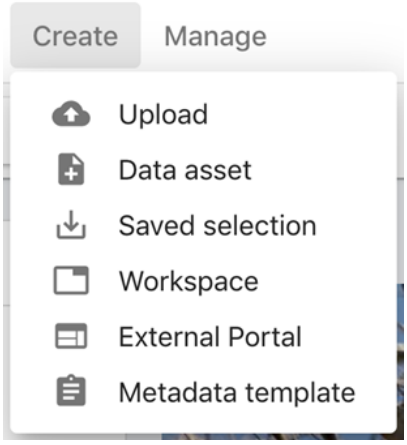
4.12. External Portals

This feature allows Global Administrators (or properly privileged General Users) to create#a#styled portal that allows#visual#access to assets for#non-users of MediaBeacon. The#portal contains a selection of assets that may, searched and#downloaded (if the portal settings allow).

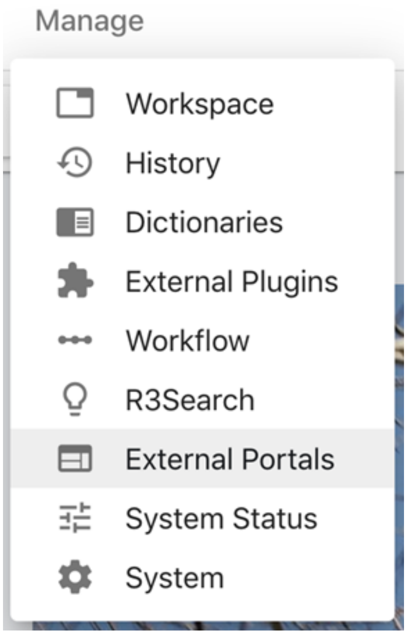
This feature’s UI is comprised of

Note that the External Portals function requires additional setup of an AWS S3 bucket to operate.

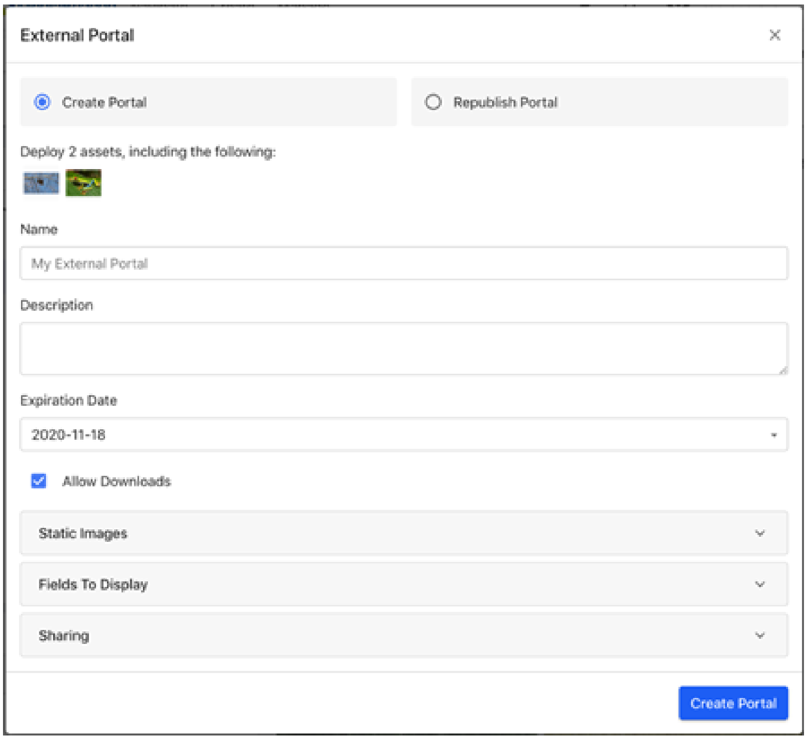
- the#Menubar#> Create > External#Portals#command



- the#Menubar#> Manage > External Portals command



- the Create External Portals dialog



External Portal

● Create Portal ○ Republish Portal

Deploy 2 assets, including the following:

Name

My External Portal

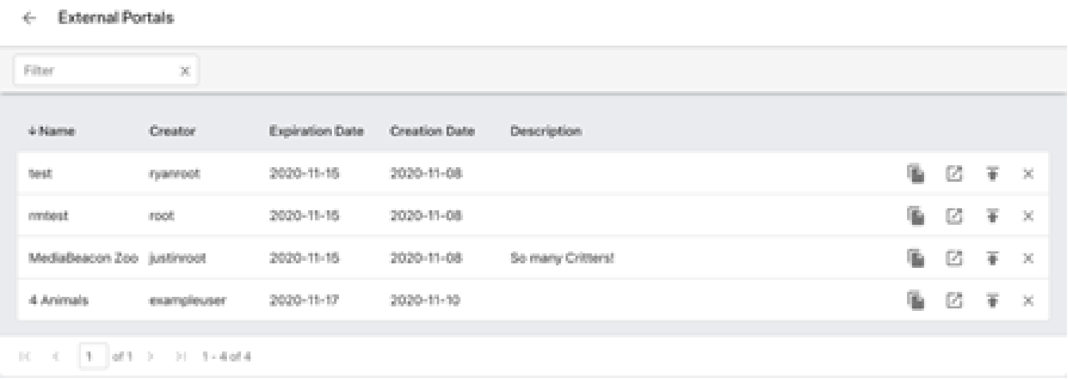
Description

Expiration Date

2020-11-18

☑ Allow Downloads

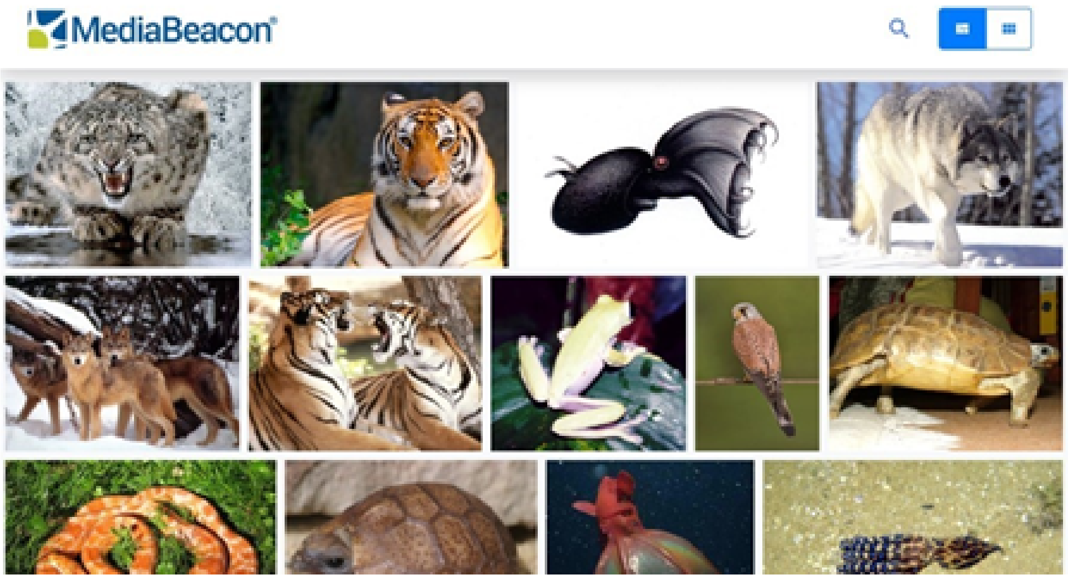
- the Manage External Portals dialog



External Portals

↓Name	Creator	Expiration Date	Creation Date	Description
test	ryanroot	2020-11-15	2020-11-08	
rmtest	root	2020-11-15	2020-11-08	
MediaBeacon Zoo	justinroot	2020-11-15	2020-11-08	So many Critters!
4 Animals	exampleuser	2020-11-17	2020-11-10	

- the#Portal UI itself



Note:

- **ACL Permission Restriction**: This feature is enabled by default for all users of an#ACL but#may hidden by using the "External Brand Portal" permission restriction

External Portal Default Graphics

The default Logo, Banner and Favicon of new portals may be set using the following system properties. The Images that are used are not resident in MediaBeacon as assets, but at a given location on the host server. Note: This location is arbitrary and is specified the system property.

The following system properties set the default images for new portals.

mb.external_portal.logo.path=C:\\Users\\Administrator\\ExternalPortalDefaults\\logo.png

mb.external_portal.banner.path=C:\\Users\\Administrator\\ExternalPortalDefaults\\banner.png

mb.external_portal.favicon.path=C:\\Users\\Administrator\\ExternalPortalDefaults\\favicon.ico

The following system properties prevent users from changing the images for new portals.

mb.external_portal.logo.locked=true

mb.external_portal.banner.locked=true

mb.external_portal.favicon.locked=true

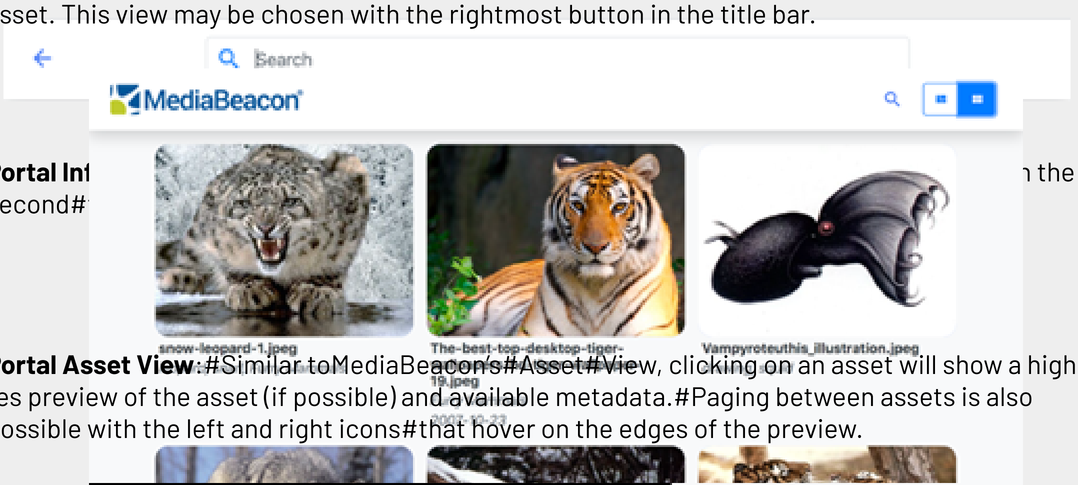
Note: These properties and files must be used on the webhead server as well (in the catalina.properties file).

4.12.1. Using an External Portal

- **Portal#Quick#Search**:#Clicking the#Magnifying#Glass icon will switch the title bar to a search field. This function is analogous to#Quick#Search in MediaBeacon. It will#search available metadata fields and file names for the entered string.#Search may be cancelled by clicking the "Back" arrow button.

Note: External portal quick search is case sensitive.

- **Portal Tile View**:#This view allows the user to see the filename and included metadata for each asset. This view may be chosen with the rightmost button in the title bar.



snow-leopard-1.jpeg

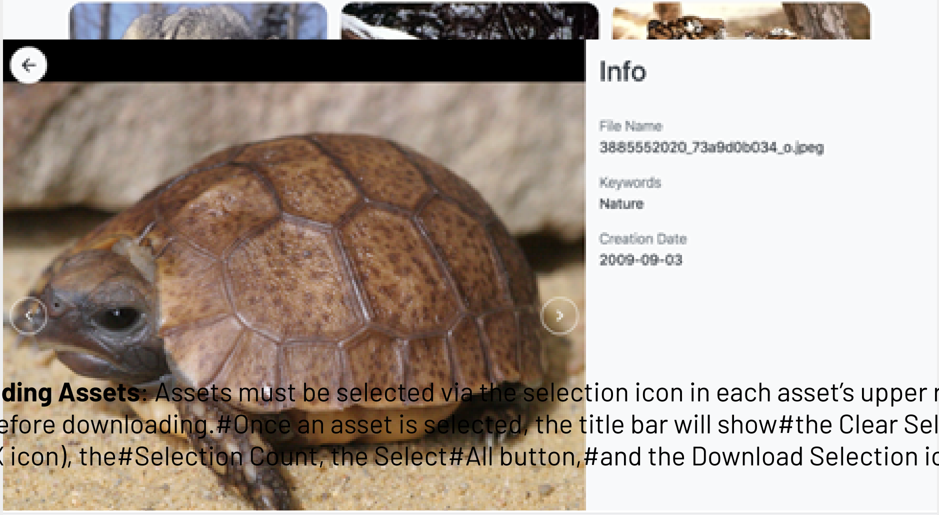
The-best-top-desktop-tiger-19.jpeg

2007-10-23

Vampyroteuthis_illustration.jpeg

- **Portal Inf** the second#

- **Portal Asset View**:#Similar toMediaBeacon's#Asset#View, clicking on an asset will show a higher res preview of the asset (if possible) and available metadata.#Paging between assets is also possible with the left and right icons#that hover on the edges of the preview.

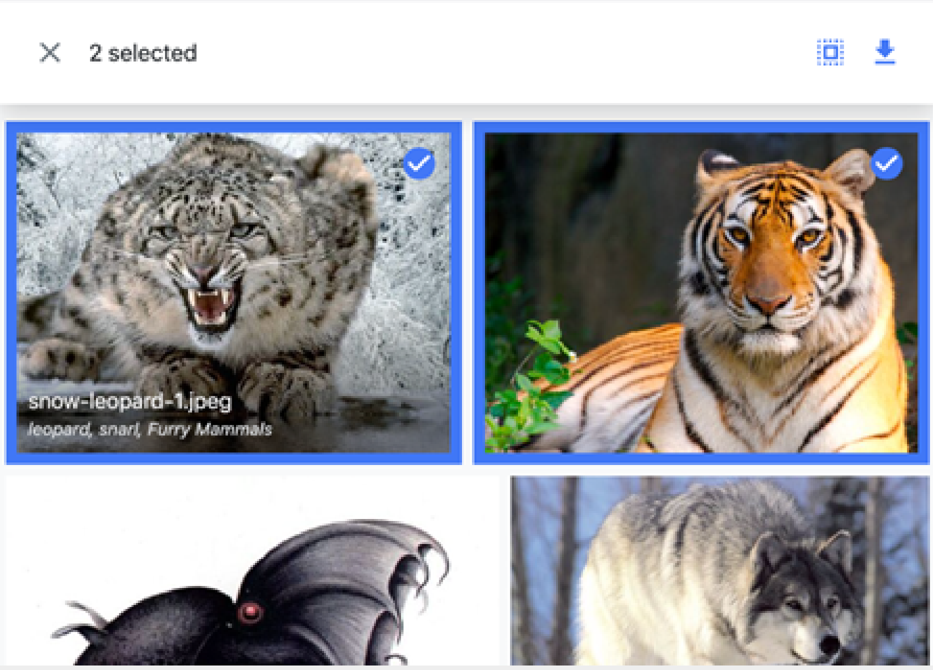

Info

File Name
3885552020_73a9d0b034_o.jpeg

Keywords
Nature

Creation Date
2009-09-03

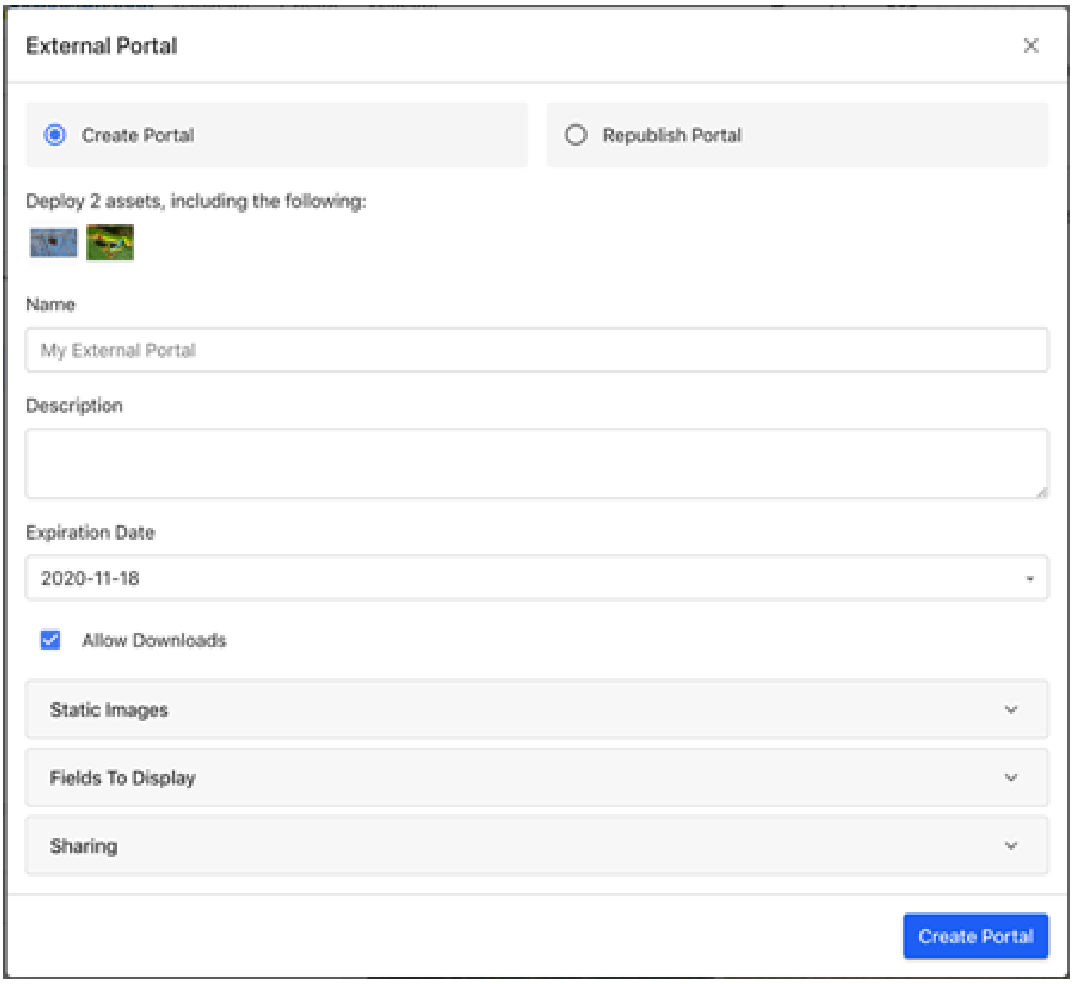
- **Downloading Assets**: Assets must be selected via the selection icon in each asset's upper right corner before downloading.#Once an asset is selected, the title bar will show#the Clear Selection button (X icon), the#Selection Count, the Select#All button,#and the Download Selection icon.


2 selected

snow-leopard-1.jpeg
leopard, snarl, Furry Mammals

4.12.2. Creating an External Portal

To create an External#Portal,#select one or more assets in a View component, and choose#Menubar#> Create > External Portals, which displays the#Create#External#Portal#Dialog.



External Portal

● Create Portal ○ Republish Portal

Deploy 2 assets, including the following:

Name: My External Portal

Description:

Expiration Date: 2020-11-18

☑ Allow Downloads

- **Create Portal**:#This radio button#is#displayed#(and#auto selected)#when an asset#selection is extant when this dialog is opened.#It allows new portals to be created.
- **Republish Portal**: Expired portals may be republished when this button is chosen. The user may modify all parameters of the portal except the included assets (Asset List).
- **Asset List**: Displays the#assets#that will be used in the portal, derived from the selection of assets at the time the dialog was opened.
- **Name**:#The name for the portal. This will be displayed as its tab name in the browser.
- **Description**: A description of the portal, not shown to the user.
- **Expiration Date**: Automatically set to a week from the current date, this is the date the portal will automatically be unpublished and unavailable.#Once a portal has expired it will be automatically deleted.
- **Allow Downloads**: Allows users of the portal to download the assets in the portal. The original files are downloaded

When an External Portal is created, creator gets notification.

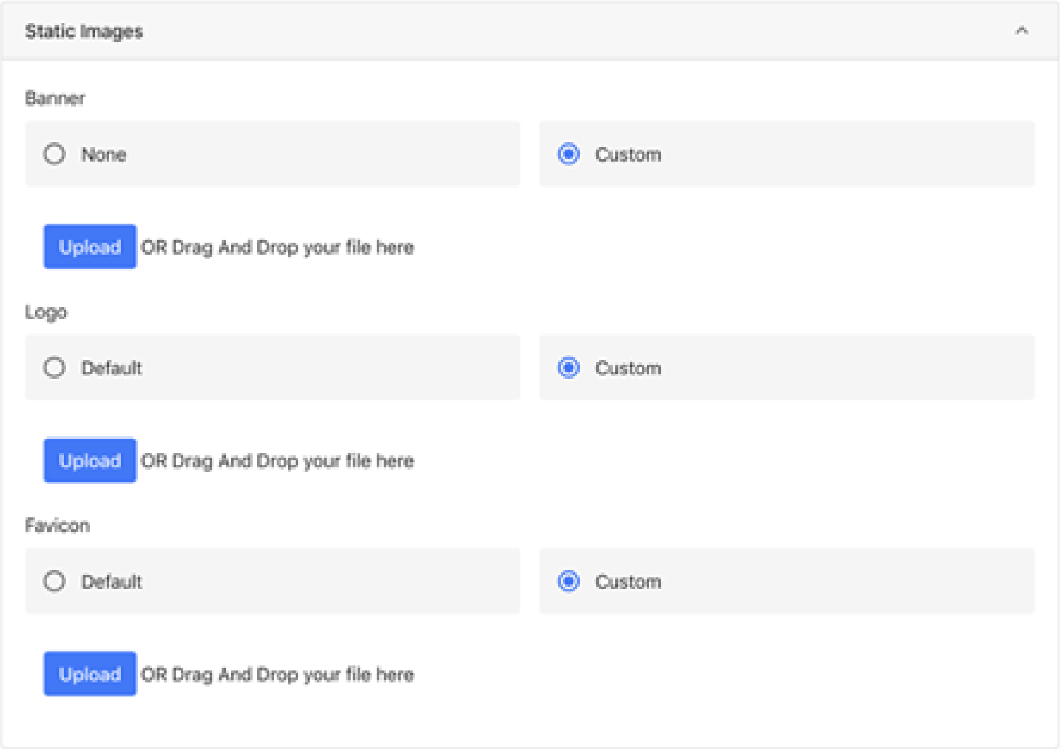
A given General User will be able to see those view portal entries in the Manage Portals dialog (Manage > External Portals) that they created, or created by users in ACLs that the given user is also a member. Global Administrator users are still able to see all External Portal entries in this interface.

Asset Retention Policy

In#External Portals, whenever a user downloads an asset, the cache of downloaded files is stored. The default value set is 50GB (previously it was 24 hours). User can download again using this cache instead of pulling files from the core file system. If the size of downloads causes the total cache size to be greater than the set value, the older files will be automatically removed until the cache size is back under the set value.

4.12.3. Customization Options

- **Static Images**



Static Images

Banner

○ None ● Custom

Upload OR Drag And Drop your file here

Logo

○ Default ● Custom

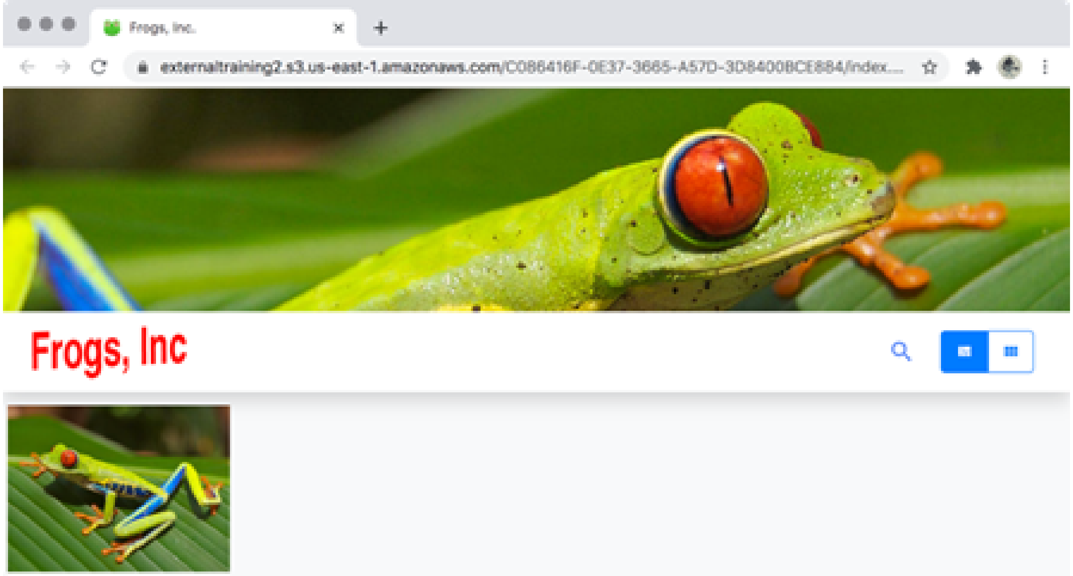
Upload OR Drag And Drop your file here

Favicon

○ Default ● Custom

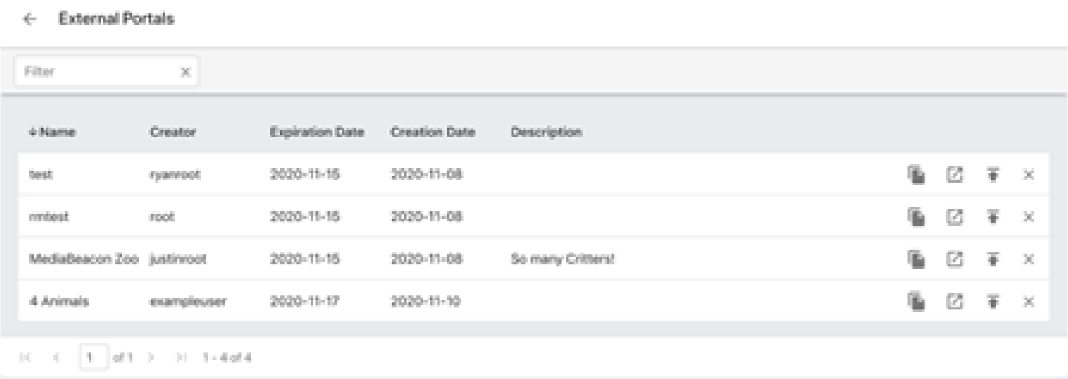
Upload OR Drag And Drop your file here

- Banner: None / Custom: This image will appear above the Portal's Action bar.
- Logo: Default (MediaBeacon) / Custom: Replaces theMediaBeacon Logo with one of the user's choice.
- Favicon: Default / Custom


Frogs, Inc.
externaltraining2.s3.us-east-1.amazonaws.com/C086416F-0E37-3665-A57D-3D8400BCE884/index...
Frogs, Inc

- Fields to Display: One or more fields may be chosen to display in the portal's Tile and Asset views. Note that the list of fields available here are those available#to the ACL in which the Portal is created.
- Sharing: Allows an email to be sent, showing a To, Subject, and Message body dialog. The URL for the portal#is auto filled in the Message Body field. Sending an email is not required to create the portal.

4.12.4. Managing External Portal



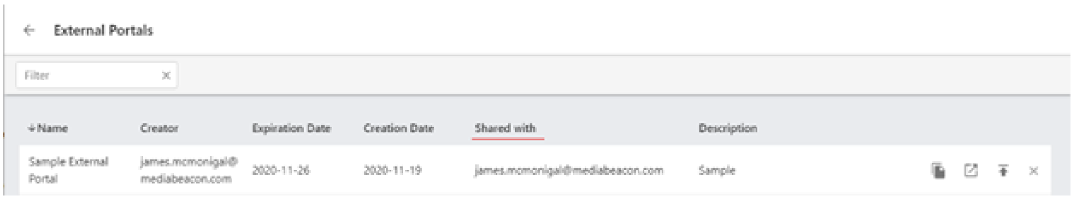
↓Name	Creator	Expiration Date	Creation Date	Description
test	ryanroot	2020-11-15	2020-11-08	
rmtest	root	2020-11-15	2020-11-08	
MediaBeacon Zoo	justinroot	2020-11-15	2020-11-08	So many Critters!
4 Animals	exampleuser	2020-11-17	2020-11-10	

The Manage Portals dialog is opened when a user chooses#Menubar#> Manage > External Portals. In this dialog, each row represents a live, accessible portal

- **Name**:#Portal's name
- **Creator**: The user who created the portal
- **Expiration Date**: The date the portal will automatically become unavailable
- **Creation Date**: The date the portal became available, during this current publication
- **Description**:#Description of the portal

- **Copy URL button**: Copies the#portal's URL#to the user's clipboard
- **Open URL Button**: Opens the portal in a new browser window/tab
- **Republish Portal#button**:#Allows the portal's details to be updated, including expiration date#and other details. See the Reusing Portals.
- Unlike previous versions, external portal can be sorted according to its name, creator, creation date and expiring date

External portals now track who they were shared with based on the email addresses on portal creation.



↓Name	Creator	Expiration Date	Creation Date	Shared with	Description
Sample External Portal	james.mcmonigal@ mediabeacon.com	2020-11-26	2020-11-19	james.mcmonigal@mediabeacon.com	Sample

4.12.5. Reusing Portal

A portal's publishing status#may be extended by "republishing" the portal before the expiration date. This may be done:

- In the Manage Portals dialog, with the#Republish Portal button,

OR

- Opening the Create Portal dialog without an asset selection being made,

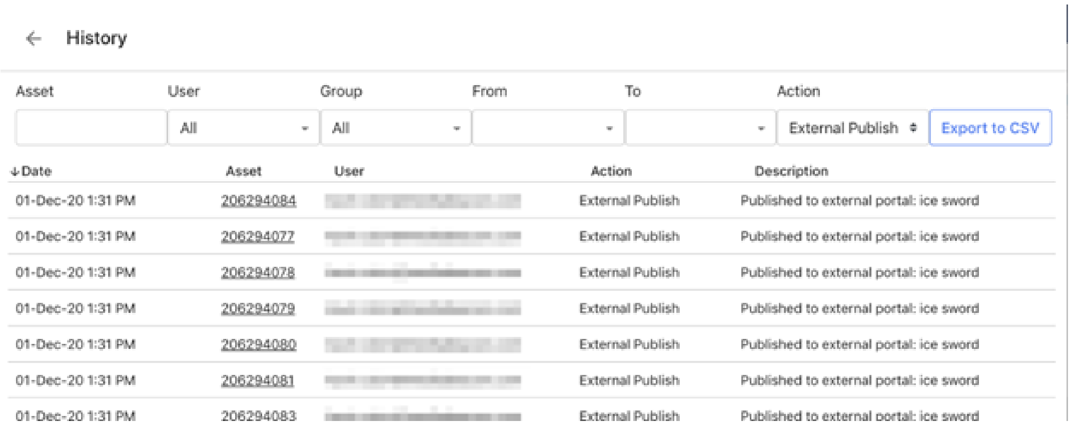
OR

- Choosing "Republish Portal" in the Create Portal dialog.

4.12.6. External Portal History

History of the actions taken in External Portals is now recorded in MediaBeacon History (Manage > History). The actions recorded are:

The "External Portal" History item has been added. This records when an asset is published via an External Portal, or when assets are in a republished External Portal.



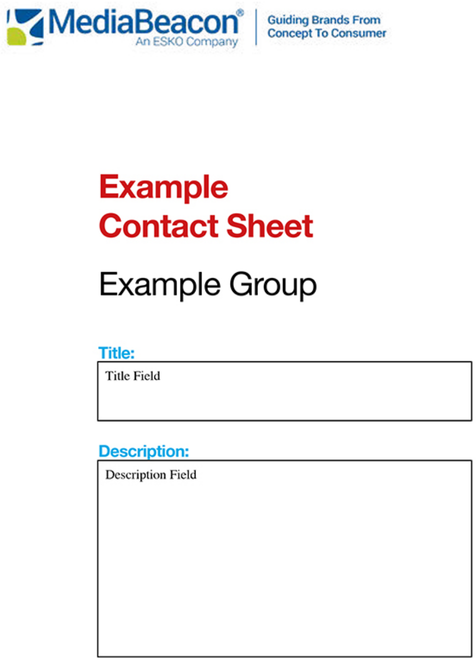
↓Date	Asset	User	Action	Description
01-Dec-20 1:31 PM	206294084	[redacted]	External Publish	Published to external portal: ice sword
01-Dec-20 1:31 PM	206294077	[redacted]	External Publish	Published to external portal: ice sword
01-Dec-20 1:31 PM	206294078	[redacted]	External Publish	Published to external portal: ice sword
01-Dec-20 1:31 PM	206294079	[redacted]	External Publish	Published to external portal: ice sword
01-Dec-20 1:31 PM	206294080	[redacted]	External Publish	Published to external portal: ice sword
01-Dec-20 1:31 PM	206294081	[redacted]	External Publish	Published to external portal: ice sword
01-Dec-20 1:31 PM	206294083	[redacted]	External Publish	Published to external portal: ice sword

5. Contact Sheet Cover Page

How to Create a Custom Contact Sheet Cover Page

Administrators can define a custom contact sheet cover page on a per-group basis. In the example below (the default cover page in MediaBeacon 2018.X and 2019.X), only the text strings "Title Field" and "Description Field" are dynamic. These are designated per contact sheet created in the Create Contact Sheet dialog.

- The file used as a cover page is a single-page PDF.
- If inheriting the Title and Description strings is desired, the PDF file must include specifically-named form field elements.
- The only way MediaBeacon effects the deign of this page is to supply the values (strings) of the Title and Description field.


MediaBeacon
An ESKO Company
Guiding Brands From Concept To Consumer

Example Contact Sheet

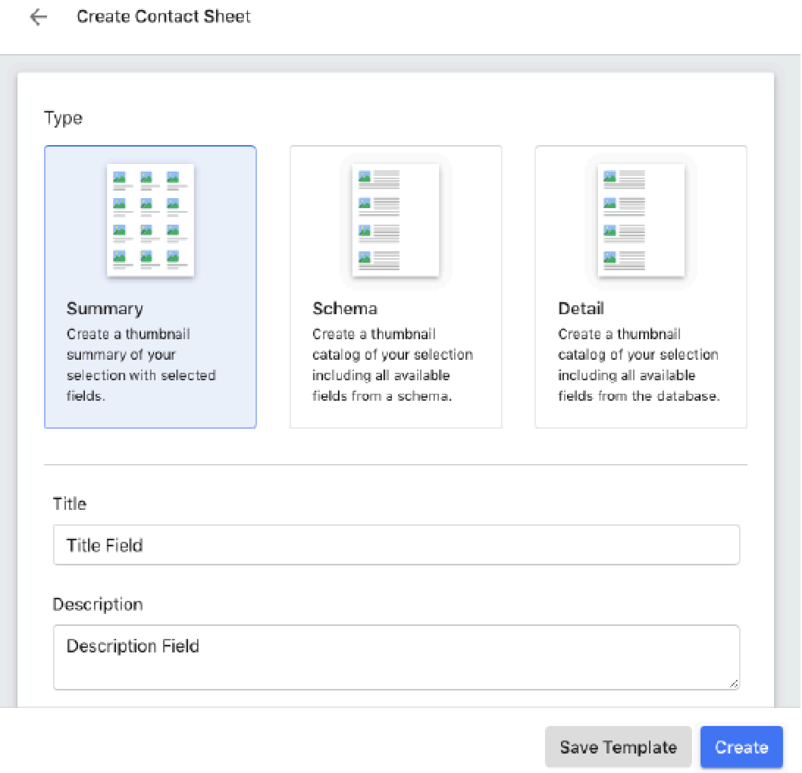
Example Group

Title:

Title Field

Description:

Description Field



Create Contact Sheet

Type

Summary
Create a thumbnail summary of your selection with selected fields.

Schema
Create a thumbnail catalog of your selection including all available fields from a schema.

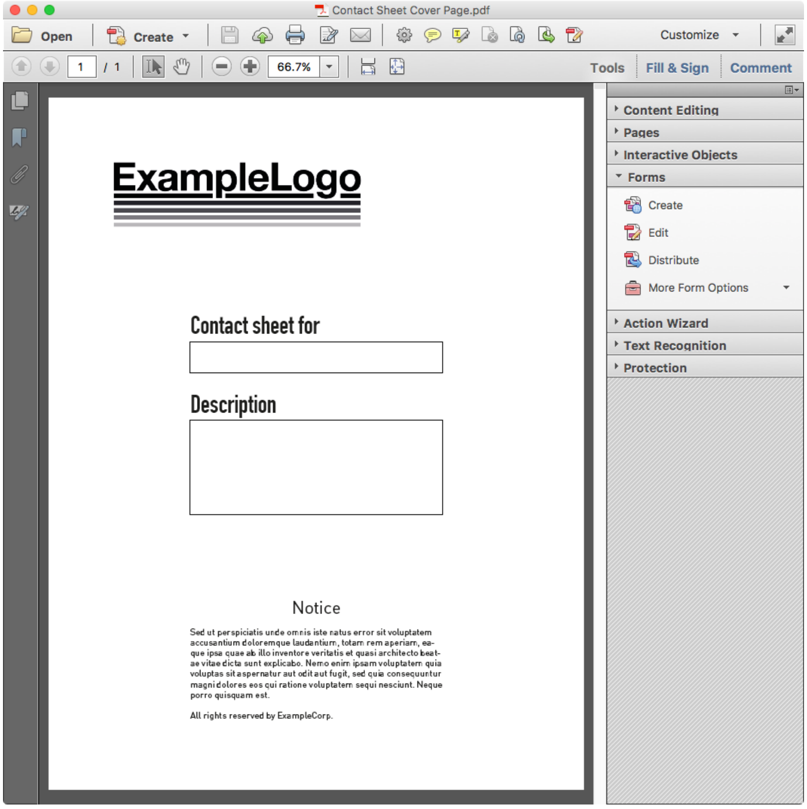
Detail
Create a thumbnail catalog of your selection including all available fields from the database.

Title: Title Field

Description: Description Field

Creating a Contact Sheet Cover Page

1. Create a design.
   a. A tool like Adobe® InDesign works well for this, as it can natively create PDF files, but any PDF is a valid source. Please note that the fields that will be filled in are not designated at this time, this is done later with Acrobat Pro's form field functionality.
   b. It is recommended to create a cover page in standard Letter size (8.5" x 11").
   c. Save a PDF of your design.
2. Open that PDF in Adobe® Acrobat Pro.
   a. Any version of Acrobat able to create and edit form fields can be used. This guide shows Acrobat XI Pro, 11.0.23.
3. Go to View > Tools > Forms.
4. The window will go into Form Editing Mode.



Contact sheet for

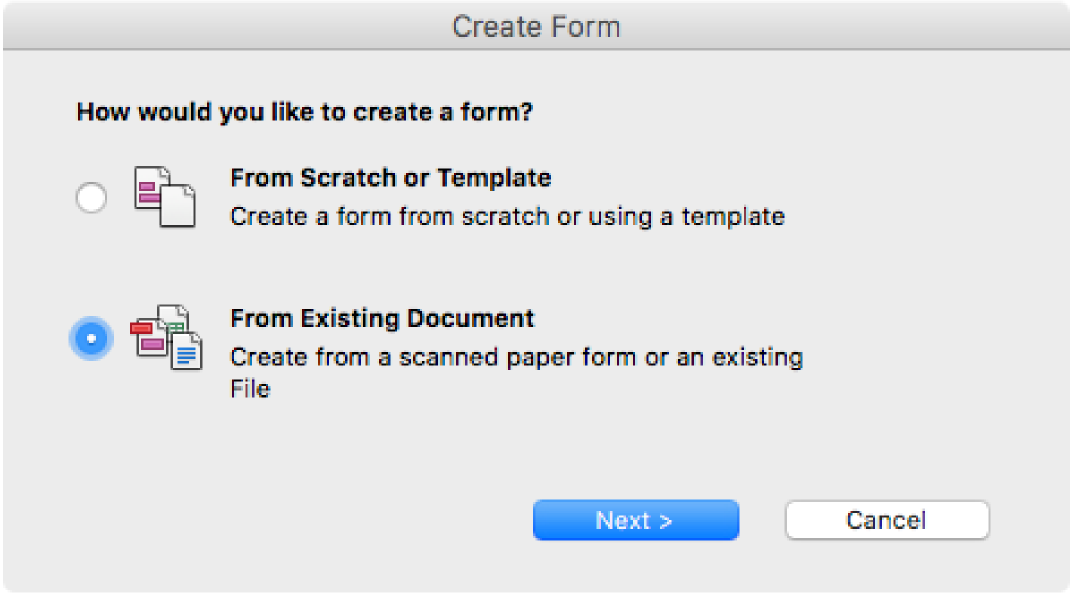
Description

Notice

Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est.

All rights reserved by ExampleCorp.

5. Choose “Create", and the "Create Form" dialog will be displayed. Choose "From Existing Document", and press "Next >".

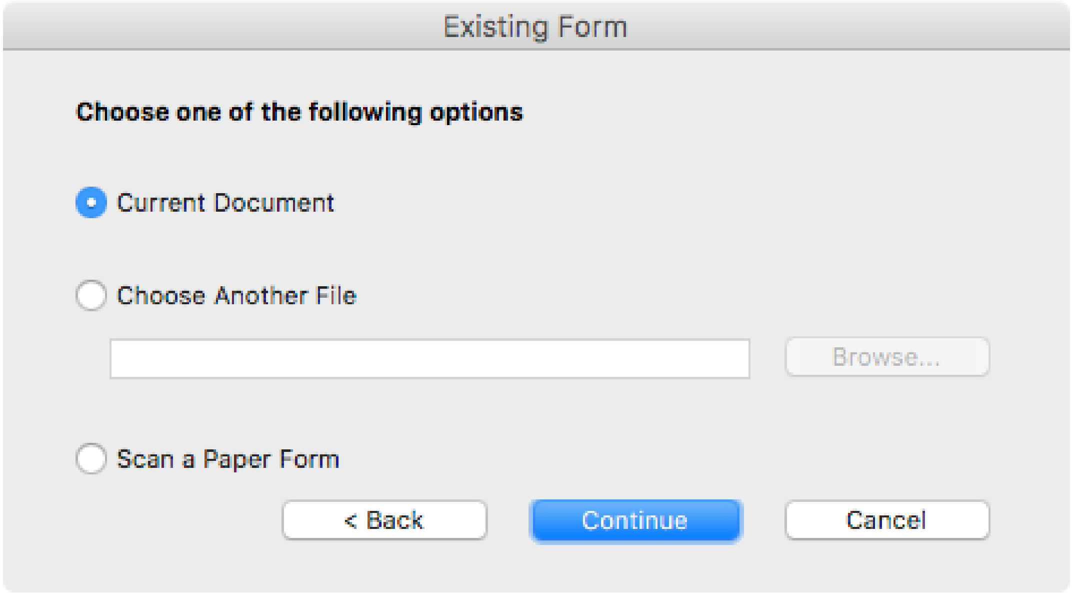

Create Form

How would you like to create a form?

○ **From Scratch or Template**
Create a form from scratch or using a template

● **From Existing Document**
Create from a scanned paper form or an existing File

6. The "Existing Form" dialog is displayed. Click "Current Document" and then "Continue".



Existing Form

Choose one of the following options

● Current Document

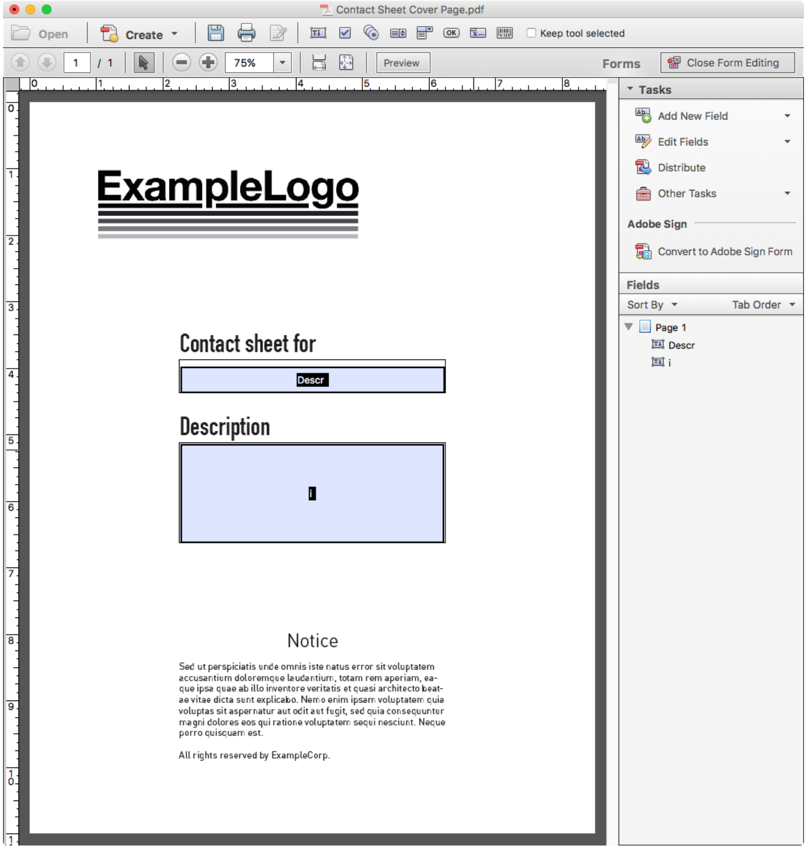
○ Choose Another File

Browse...

○ Scan a Paper Form

< Back | Continue | Cancel

7. If there are any empty frames (as in the example) Acrobat will attempt to create form fields. This is an OCR- like process by Acrobat and not always exact, so some hand-adjustment is required.



ExampleLogo

Contact sheet for

Descr

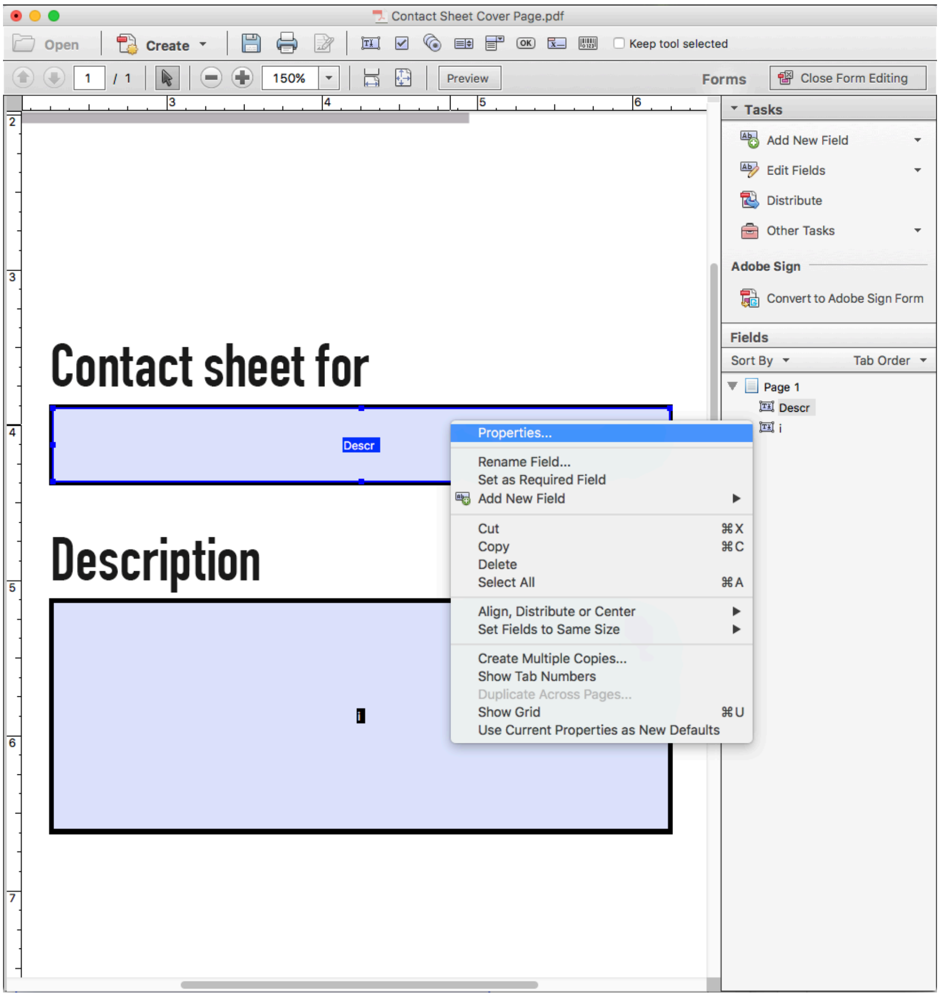
Description

Notice

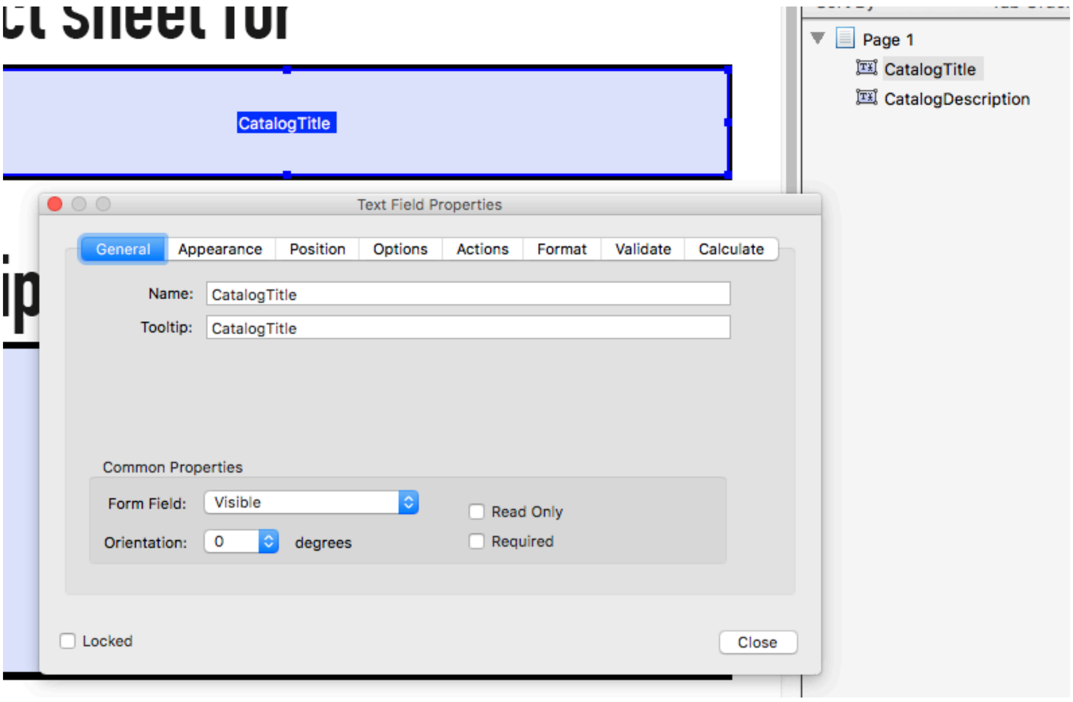
Sed ut perspiciatis unde omnis iste natus error sit voluptatem accusantium doloremque laudantium, totam rem aperiam, eaque ipsa quae ab illo inventore veritatis et quasi architecto beatae vitae dicta sunt explicabo. Nemo enim ipsam voluptatem quia voluptas sit aspernatur aut odit aut fugit, sed quia consequuntur magni dolores eos qui ratione voluptatem sequi nesciunt. Neque porro quisquam est.

All rights reserved by ExampleCorp.

8. Adjust the of the size and placement of these fields to align to the frame edges. Then, right-click a field to edit its properties.



9. In this dialog, designate one field "CatalogTitle" and the other "CatalogDescription". The Name and Tooltip do not need to match, but it's best practice to have them match or to leave the tooltip blank. See the Appendix for more from field properties.



Name: CatalogTitle

Tooltip: CatalogTitle

Form Field: Visible

Orientation: 0 degrees

10. Save the PDF.

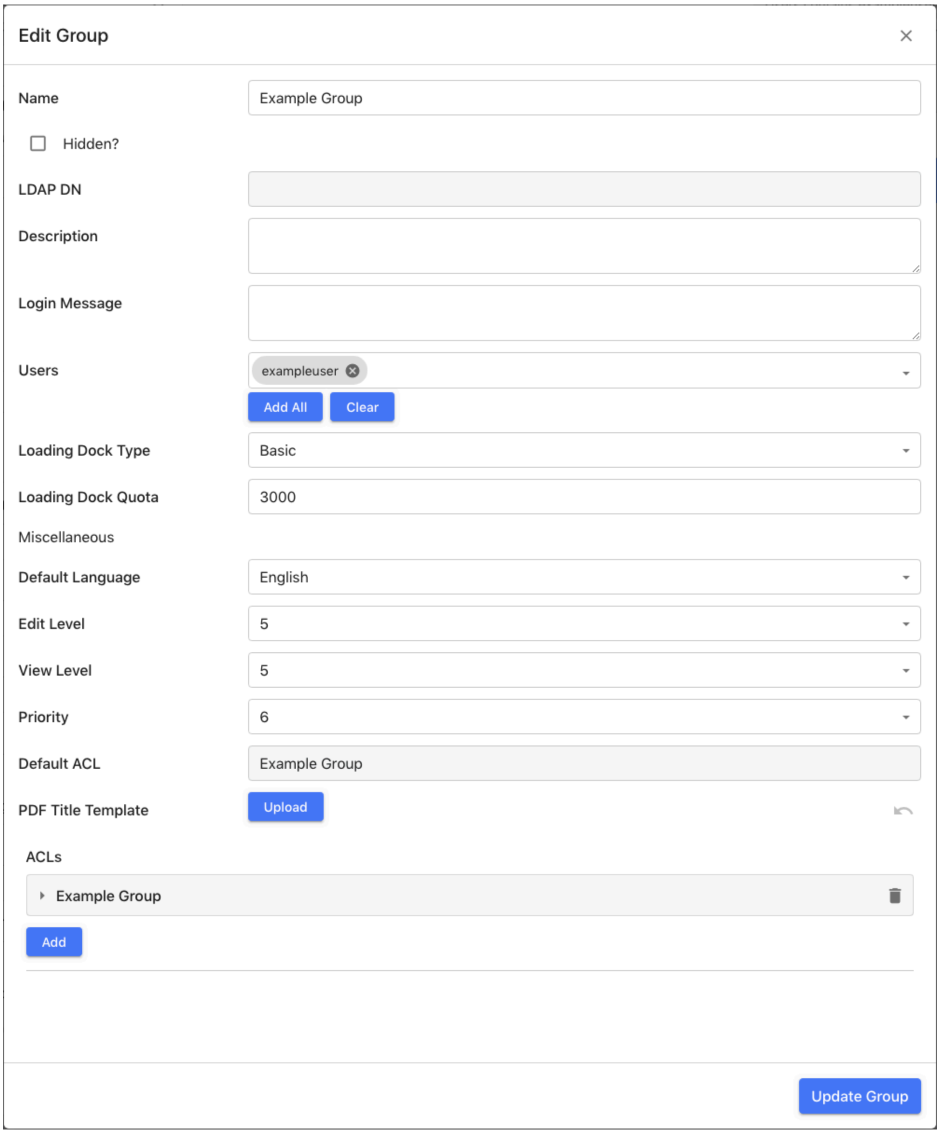
   a. Note: these fields can be further customized in Acrobat by setting font, text size, and paragraph formatting.

Configuring a Contact Sheet Cover Page

Contact sheet cover pages are assigned on the Group level in MediaBeacon. Note that there are configuration differences between 2018.x and 2019.x.

MediaBeacon 2019.X Configuration

1. Log into MediaBeacon as a Global Administrator user, and go to “Manage > System", then to "Navigate > Groups."
2. Double-click the icon of the group you wish to edit. In the Edit Group dialog, next to "PDF Title Template" (an older name for the contact sheet cover page) click the "Upload" button and choose the PDF from above, and click the next "Upload" button.



Edit Group ×

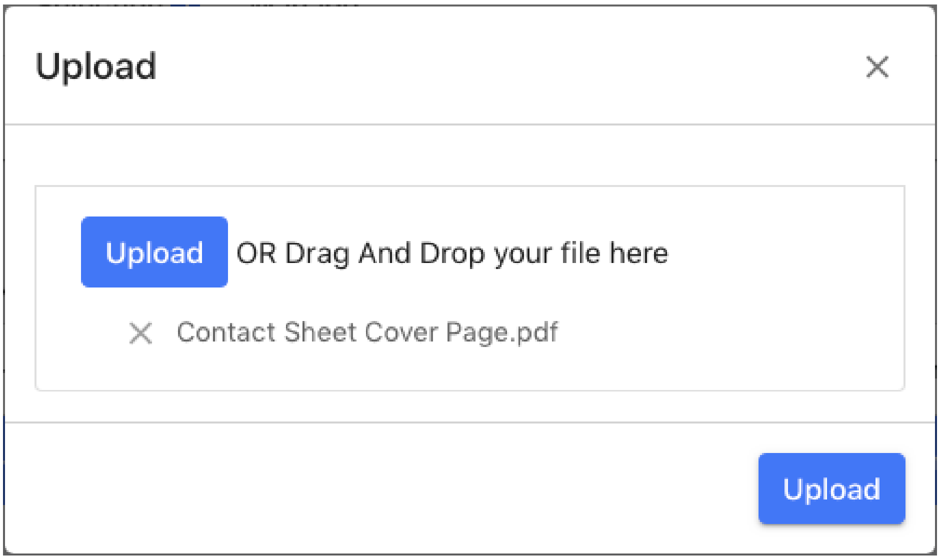
Name	Example Group
□ Hidden?	
LDAP DN	
Description	
Login Message	
Users	exampleuser ⊗ Add All Clear
Loading Dock Type	Basic
Loading Dock Quota	3000
Miscellaneous	
Default Language	English
Edit Level	5
View Level	5
Priority	6
Default ACL	Example Group
PDF Title Template	Upload

ACLs

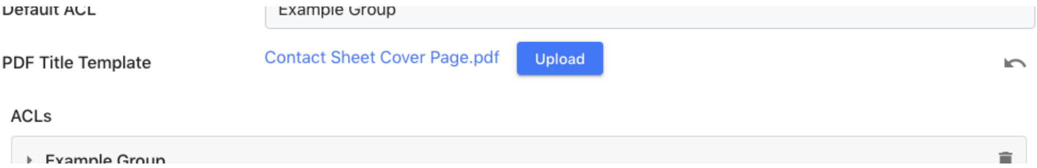
▸ Example Group

Add

Update Group



3. If necessary, the custom PDF may be cleared with the revert button to the right of the Upload button.



4. Click "Update Group", and leave the Admin Core.
5. Once all users who are members of the group have logged out and back in again, the new coper page will be used for Contact Sheets.

MediaBeacon 2018.X Configuration

This version of MediaBeacon does not have a UI-based approach. In 2018.x, a custom contact sheet cover page can be defined as a group specific cover page or as a fallback.

1. Set system property. This defines the "template location" folder, within which the system will look for cover pages. The folder specified is the recommended location, and is relative to <Shared Files Directory>/m-userinfo/

```
mb.pdf.template.location="/m-templates/pdf/"
```

2. After this property is set, MediaBeacon must be restarted.
3. Name Cover Page Files.
   a. The names of PDFs residing in the "template location" directory affect their function. No restart of MediaBeacon is required when new cover pages PDF files are added or changed in this directory, as MediaBeacon checks this directory each time a catalog is made.
   b. Default (Fallback) Cover Page: This will be used when a group has no specific pdf specified.
      a.
      ```
      pdfCoverpage.pdf
      ```

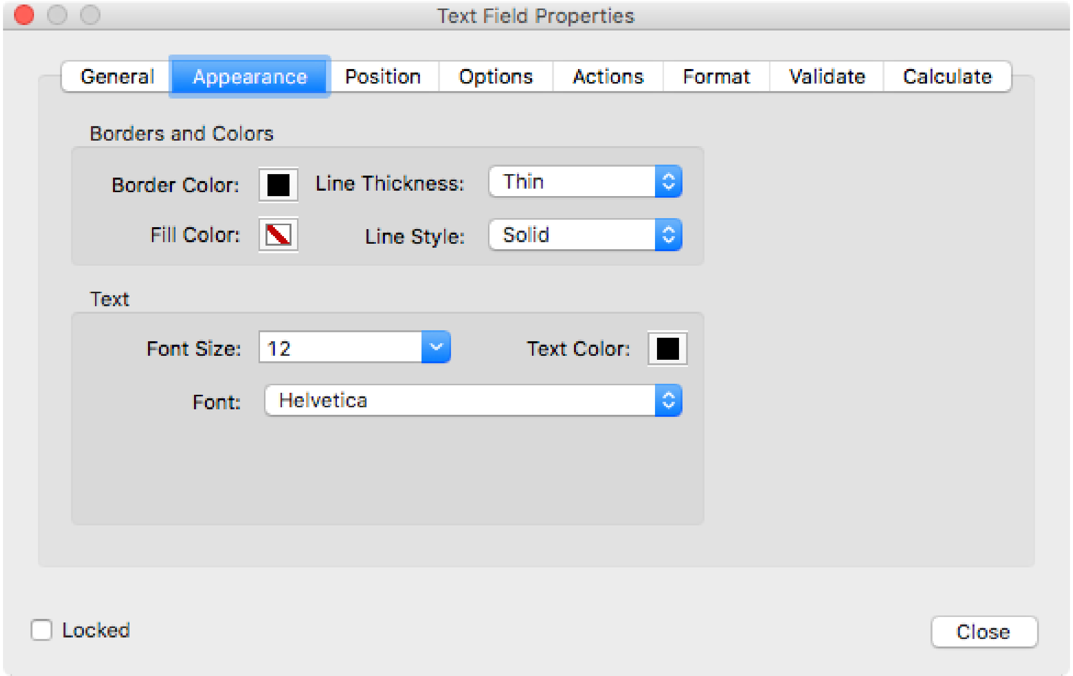
c. Group-specific Cover Page: This will be used for a given group, replacing the string "groupname", without the "<>".

   a.
   ```
   <groupname>.template.pdf
   ```

5.1. Appendix

Text Field Properties in Acrobat Pro

Appearance Tab



Text Field Properties

Borders and Colors

Border Color: ■ Line Thickness: Thin

Fill Color: ▨ Line Style: Solid

Text

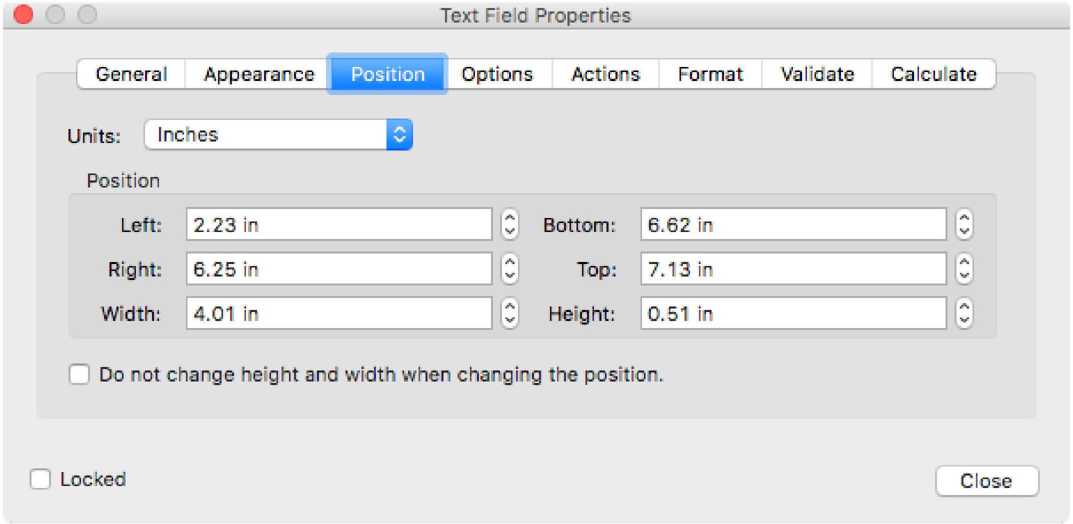
Font Size: 12 Text Color: ■

Font: Helvetica

□ Locked

- Borders: If your design did not start with boxes as graphic elements, you can add a border to text frames directly.
- The "Font Size" property may be set to "Auto". If this is set, the text this box inherits will scale in size to fit the box. If this is not desirable, set the Font Size to a static value.
- Font may be set here to provide a consistent look in your forms. It is not recommended to use custom fonts that are not likely to be on the user's system.

Position Tab



Text Field Properties

General | Appearance | Position | Options | Actions | Format | Validate | Calculate

Units: Inches

Position

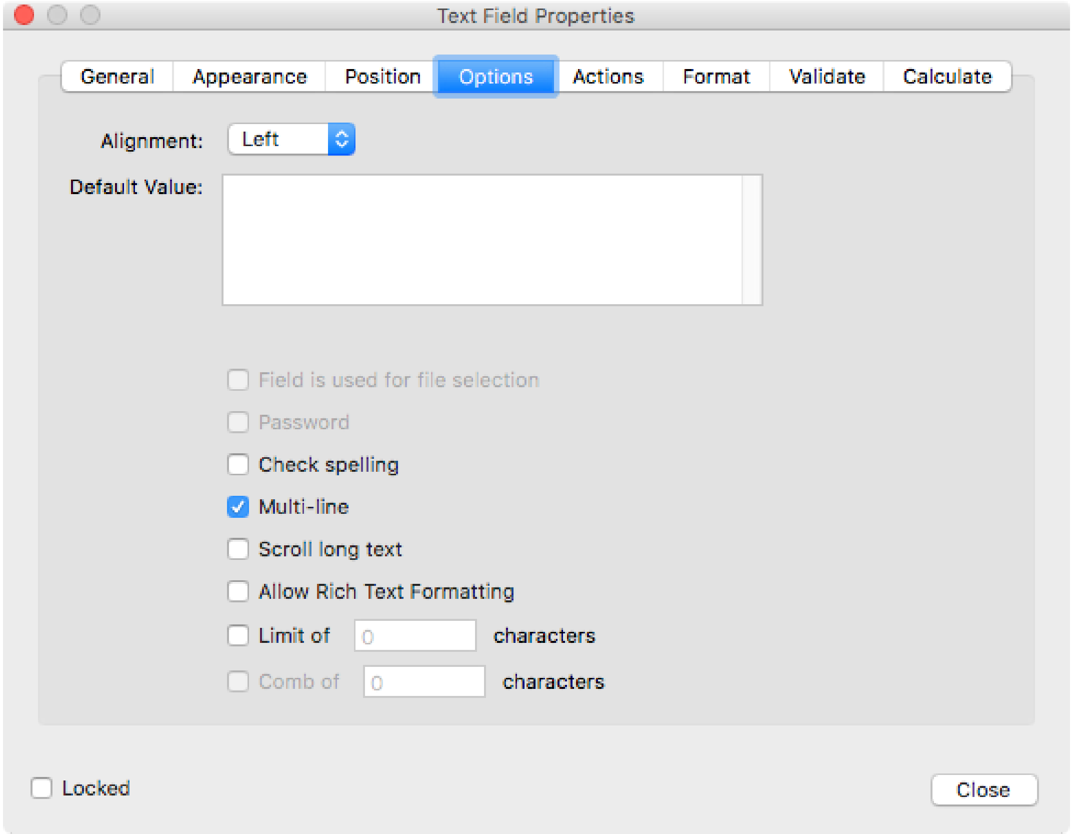
Left: 2.23 in
Right: 6.25 in
Width: 4.01 in
Bottom: 6.62 in
Top: 7.13 in
Height: 0.51 in

□ Do not change height and width when changing the position.

□ Locked

- If necessary, very fine adjustments to size and position of fields can be made here.

Options Tab



Text Field Properties

General | Appearance | Position | Options | Actions | Format | Validate | Calculate

Alignment: Left

Default Value:

□ Field is used for file selection
□ Password
□ Check spelling
☑ Multi-line
□ Scroll long text
□ Allow Rich Text Formatting
□ Limit of 0 characters
□ Comb of 0 characters

□ Locked

- Alignment can be set here.
- Default Value: leave this blank.
- Check Spelling is not required.

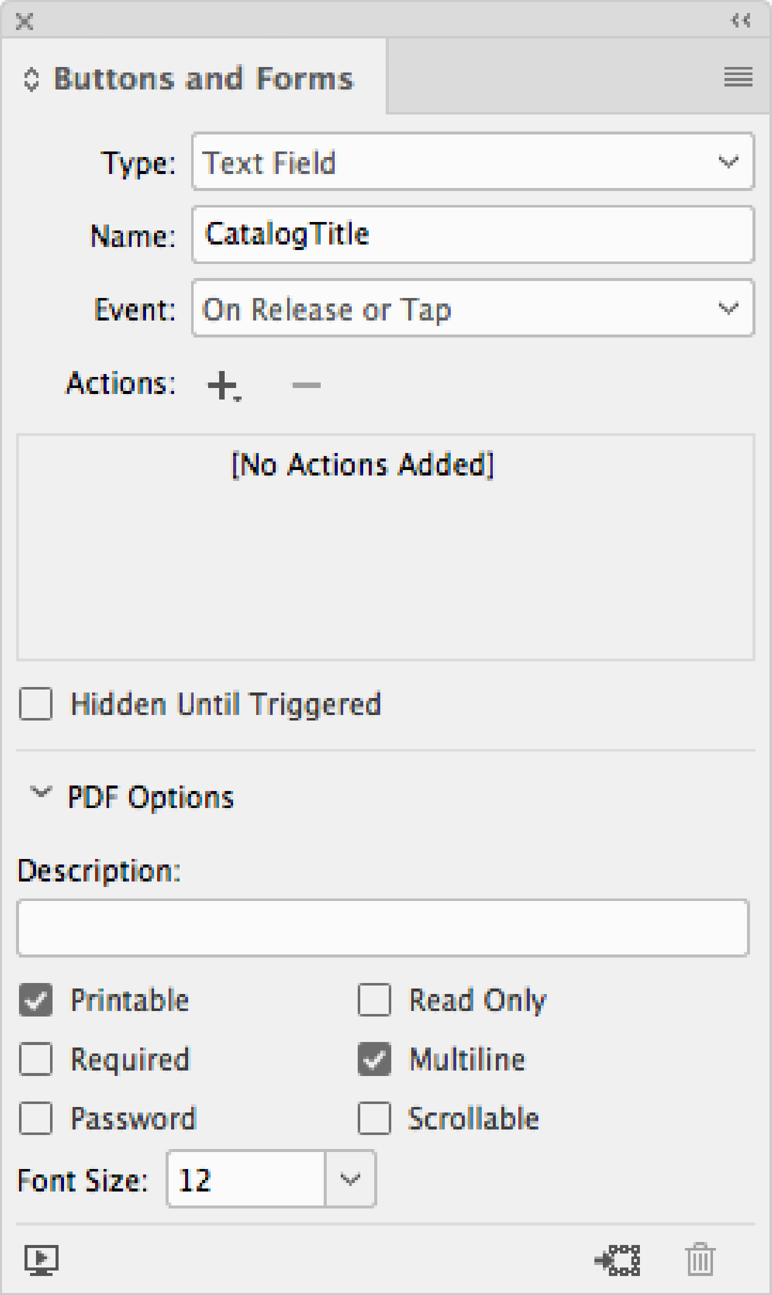
- Multi-line will align text to the top of the field, allowing text to wrap. If this is off, the text will appear vertically centered in the form field and will be truncated if the string size exceeds the box size.
- Scroll long text: Only useful if the PDF on a computer.

Creating Form Field PDFs Directly From InDesign

Adobe® InDesign can be used to create a PDF with form fields, however, pertinent field properties will still need to be edited in Acrobat Pro (Font-Size, Font, Check Spelling, Multi-line, and Scroll Long Text). This technique is most useful to create form fields in exact positions.

In InDesign:

1. Add Window > Interactive > Buttons and Forms to your toolbar.
2. Choose a frame to make into a form field.
3. In the Buttons and Forms panel, choose Type: Text Field.
4. Enter the correct name of the Field.
5. Note: The multi-line and scrollable options can be adjusted here.



Buttons and Forms

Type: Text Field

Name: CatalogTitle

Event: On Release or Tap

Actions:

[No Actions Added]

□ Hidden Until Triggered

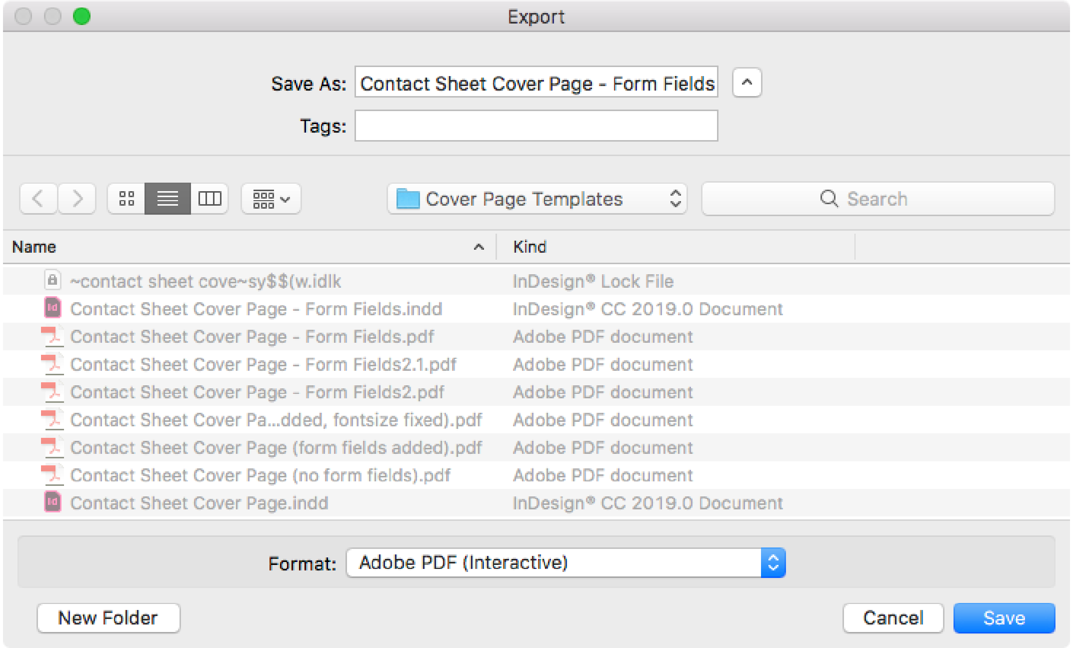
PDF Options

Description:

☑ Printable
□ Read Only
□ Required
☑ Multiline
□ Password
□ Scrollable

Font Size: 12

6. Save the INDD file.



Export

Save As: Contact Sheet Cover Page - Form Fields

Tags:

Cover Page Templates

Name	Kind
~contact sheet cove~sy$$(w.idlk	InDesign® Lock File
Contact Sheet Cover Page - Form Fields.indd	InDesign® CC 2019.0 Document
Contact Sheet Cover Page - Form Fields.pdf	Adobe PDF document
Contact Sheet Cover Page - Form Fields2.1.pdf	Adobe PDF document
Contact Sheet Cover Page - Form Fields2.pdf	Adobe PDF document
Contact Sheet Cover Pa...dded, fontsize fixed).pdf	Adobe PDF document
Contact Sheet Cover Page (form fields added).pdf	Adobe PDF document
Contact Sheet Cover Page (no form fields).pdf	Adobe PDF document
Contact Sheet Cover Page.indd	InDesign® CC 2019.0 Document

Format: Adobe PDF (Interactive)

7. Choose File > Export, and choose format "Interactive PDF".
8. From here, edit the properties of the fields as necessary in Acrobat Pro.

6. MediaBeacon Port Information for Remote Components

MediaBeacon has a number of remote components that can be used in conjunction with the core MediaBeacon application. Below are the default ports various components use to connect.

Tomcat Webhead

- Connects to MediaBeacon R3volution core: 44444, 44445, and 44446.
- Connects to R3Search node(s): 30000.
- Connects to Zookeeper(s): 31000
- Connects to Zookeeper(s): 31000.
- Connects to external authentication server, if in use (Active Directory, LDAP, etc.)

Remote Preview Node

- Connects to MediaBeacon R3volution core: 44444, 44445, and 44446.
- Connects to the database.
- Connects to the file system.

Remote Import Node

- Connects to MediaBeacon R3volution core: 44444, 44445, and 44446.
- Connects to the database.
- Connects to the file system.

Remote R3search

- Connects to MediaBeacon core: 44444, 44445, and 44446.
- Connects to R3Search node(s): 30000.
- Connects to Zookeeper(s): 31000.
- Connects to the database.

Remote Zookeeper

- Connects to MediaBeacon core: 44444, 44445, and 44446.
- Connects to R3Search node(s): 30000.
- Connects to Zookeeper(s): 31000., 31001, 31002.
- Connects to the database.