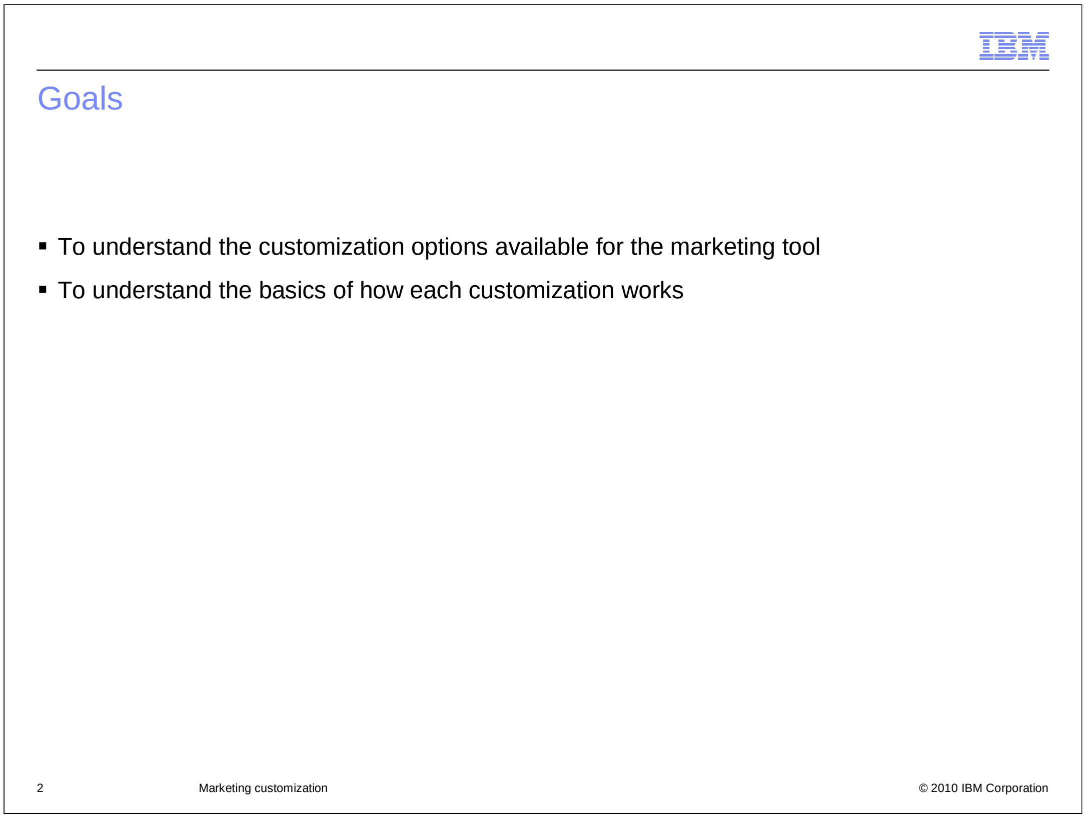
WebSphere Commerce V7.0

Marketing customization

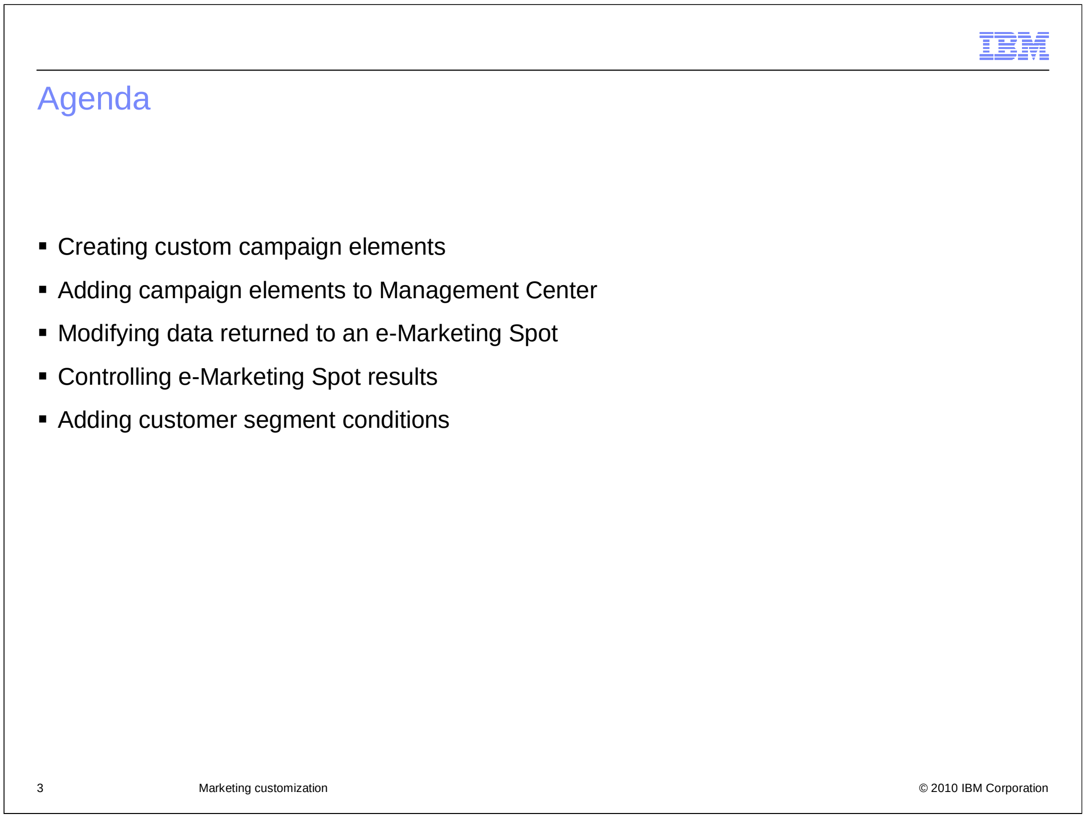
This presentation introduces marketing customization in WebSphere® Commerce version 7.



Goals

- To understand the customization options available for the marketing tool
- To understand the basics of how each customization works

By the end of this presentation, you should understand the various customization options available for the marketing tool. You should also have a basic idea how each type of customization is achieved.

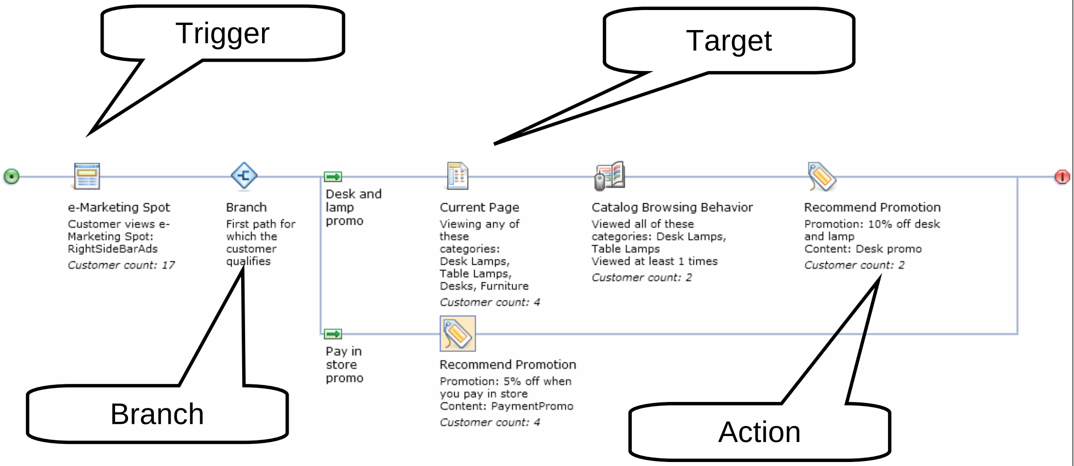


Agenda

- Creating custom campaign elements
- Adding campaign elements to Management Center
- Modifying data returned to an e-Marketing Spot
- Controlling e-Marketing Spot results
- Adding customer segment conditions

Many different customization scenarios will be covered beginning with creating custom campaign elements such as triggers, targets and actions. Next, you will see how custom campaign elements can be added to the Management Center. The presentation will also examine how to modify the data returned to an e-Marketing Spot and how to control the results displayed in an e-Marketing Spot. The presentation will conclude with a look at adding customer segment conditions.

Campaign elements



e-Marketing Spot
Customer views e-Marketing Spot: RightSideBarAds
Customer count: 17

Branch
First path for which the customer qualifies

Desk and lamp promo

Current Page
Viewing any of these categories: Desk Lamps, Table Lamps, Desks, Furniture
Customer count: 4

Catalog Browsing Behavior
Viewed all of these categories: Desk Lamps, Table Lamps
Viewed at least 1 times
Customer count: 2

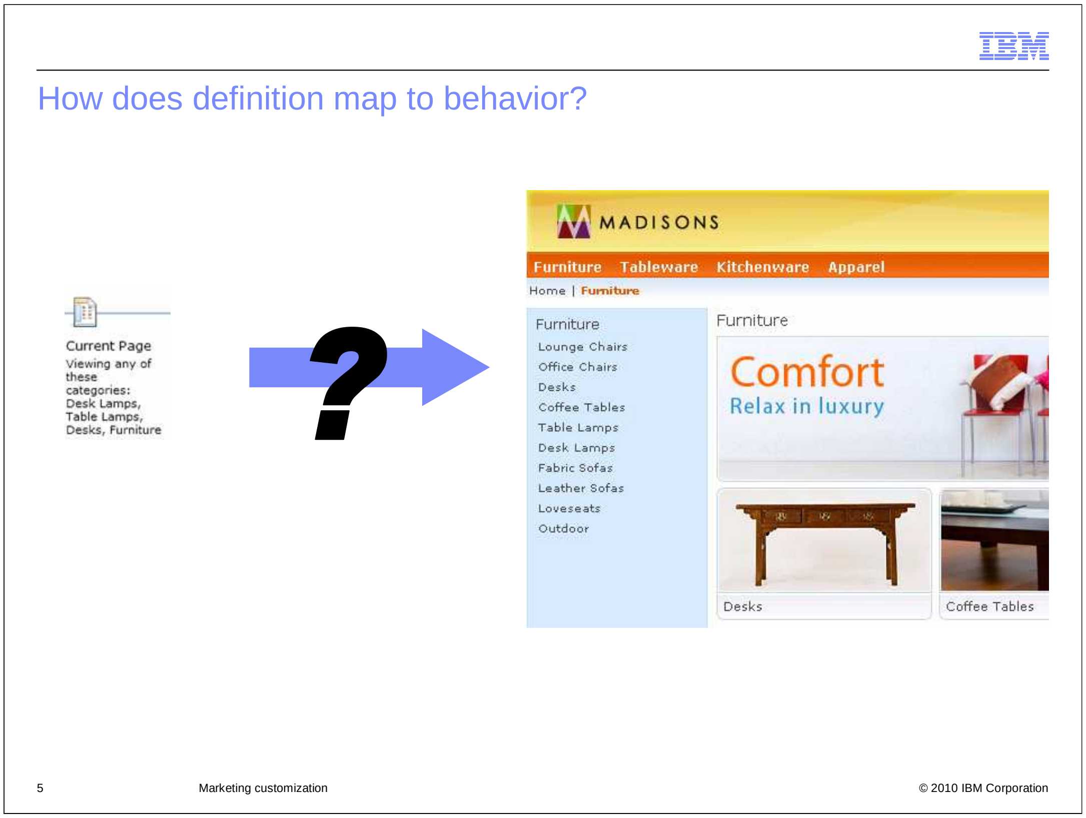
Recommend Promotion
Promotion: 10% off desk and lamp
Content: Desk promo
Customer count: 2

Pay in store promo

Recommend Promotion
Promotion: 5% off when you pay in store
Content: PaymentPromo
Customer count: 4

4 Marketing customization © 2010 IBM Corporation

This slide is a review of WebSphere Commerce marketing terminology. A trigger is used to wait for a shopper to do something, or to wait for a certain period of time to elapse. When a trigger occurs, the activity flow can continue from where the trigger was defined in the flow. A target is used to qualify shoppers for subsequent actions or continue execution of the activity flow. Targets are typically based on a shopper’s behavior and segmentation. An action is a step to perform as part of the activity flow. Actions are the “to dos” based on the previous sequence of triggers and targets. Most actions are for marketing purposes, such as display content in an e-Marketing Spot, or sending the shopper an SMS message. A branch is a decision node in an activity where shoppers might see or qualify for different content based on some outcome or result. The multi-channel precision marketing presentation introduced the many triggers, targets and actions available to create marketing activities. In V7, you also have the ability to create new triggers, targets and actions.

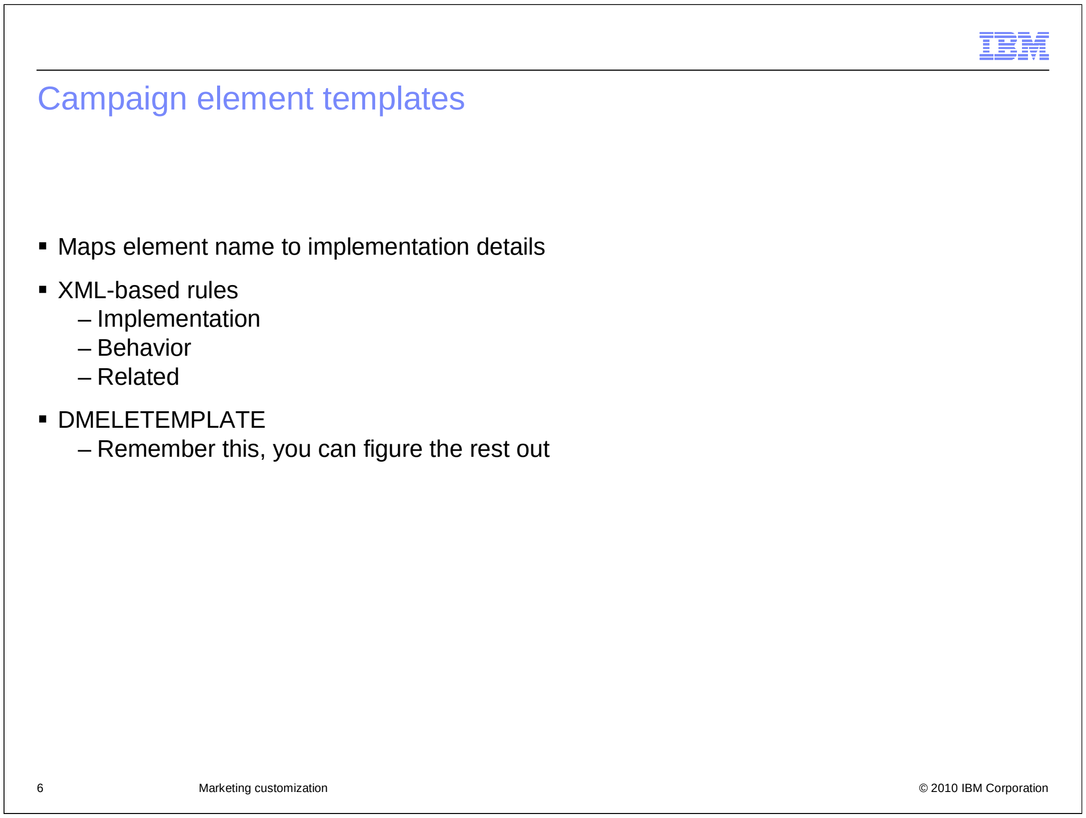


How does definition map to behavior?

Current Page
Viewing any of these categories: Desk Lamps, Table Lamps, Desks, Furniture

5 Marketing customization

If you have worked with Management Center in previous versions, you probably have a general idea how you might approach adding a new trigger, target or action to the activity builder. This presentation will discuss those steps later on. For now, assume you have added a new target element. How do you relate that element from an icon in a visual flow to new capability in your store? If you think of an existing target, how does adding a 'Current Page' target actually cause the store front to recognize when a shopper is viewing that page? Customizing marketing is about understanding what happens in between.



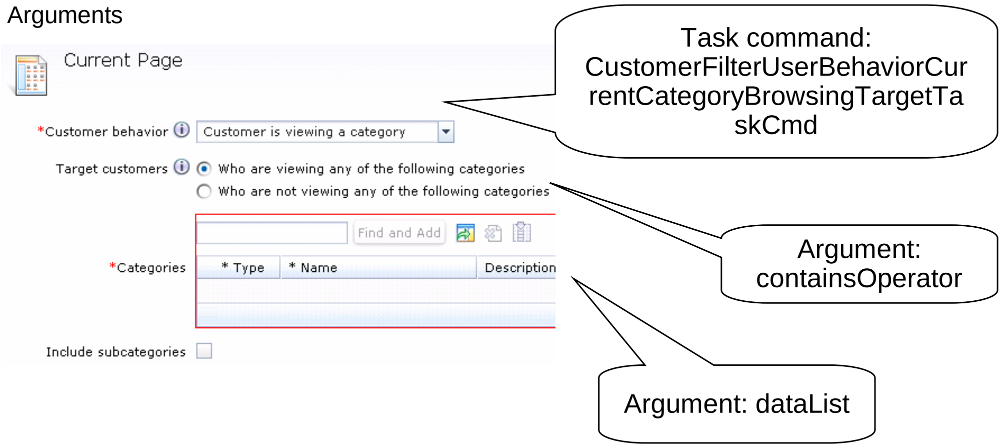
Campaign element templates

- Maps element name to implementation details
- XML-based rules
  - Implementation
  - Behavior
  - Related
- DMELETEMPLATE
  - Remember this, you can figure the rest out

Campaign element templates are the key to mapping a campaign element definition in Management Center to an implementation in the store front. An element template is a series of XML-based rules. The database table DMELETEMPLATE stores the templates for all campaign elements. If you have experience working with WebSphere Commerce, you can probably complete your customization just by examining the existing entries in this table. The next few slides will examine the implementation, behavior and related rules in more detail.

Implementation rule – targets and actions

- Task command
  - MarketingCampaignElementTaskCmd
- Arguments



Current Page

*Customer behavior: Customer is viewing a category

Target customers: ● Who are viewing any of the following categories ○ Who are not viewing any of the following categories

*Categories: Find and Add

* Type	* Name	Description

Include subcategories □

Task command: CustomerFilterUserBehaviorCurrentCategoryBrowsingTargetTaskCmd

Argument: containsOperator

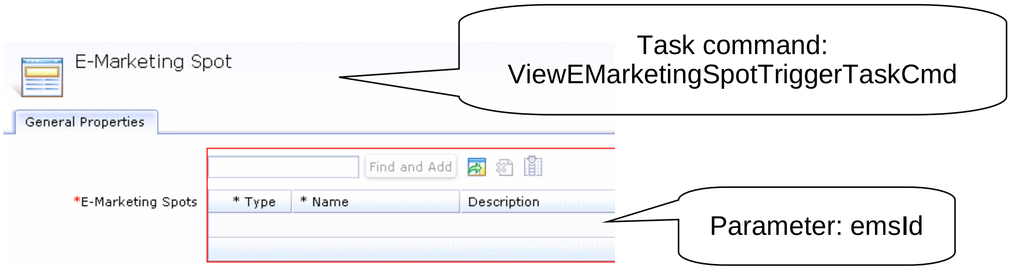
Argument: dataList

7 Marketing customization © 2010 IBM Corporation

The implementation rule for a target or action contains two key pieces of information. The first is the task command to call to validate the target or perform the action. This task command must be a subclass of MarketingCampaignElementTaskCmd. The second is the list of arguments to pass to the command. These arguments are typically specified by the business user when configuring the target or action. However you can also specify fixed value arguments. In the case of the ‘Current Page’ target, the task command is determined by the type of behavior selected in the properties pane. The drop-down menu has five options for behavior to match and there are five corresponding element templates in DMELETEMPLATE, each with a separate task command. The selected behavior, ‘Customer is viewing a category’, has two arguments specified by the business user. The values of these two arguments are stored in the table DMELEMENTNVP.

Implementation rule - triggers

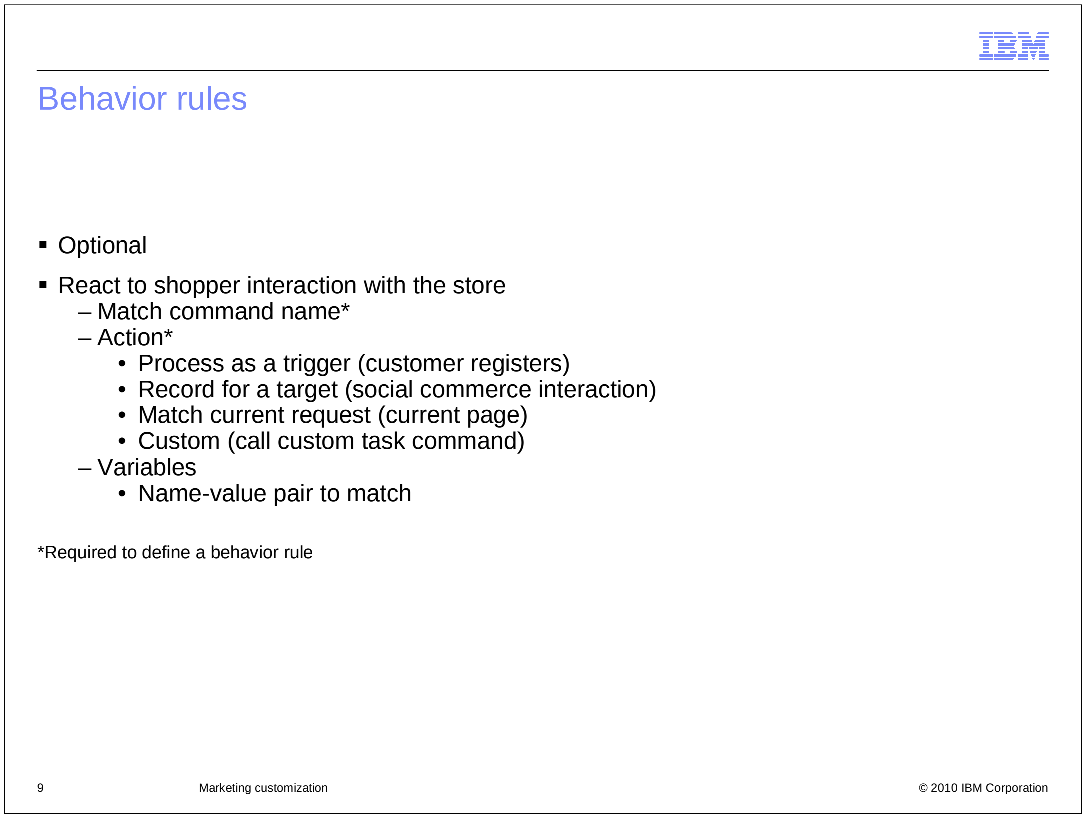
- Parameters
- Task command to validate parameters
  - MarketingCampaignElementTaskCmd



Task command:
ViewEMarketingSpotTriggerTaskCmd

Parameter: emsId

The trigger implementation definition has a slightly different XML format than targets and actions. The only information required is a unique name for the trigger. A trigger might optionally have parameters specified in Management Center when configuring the marketing activity. If parameter values are set, a task command is defined to validate the parameter values. In the example shown here, the "View e-Marketing Spot" trigger has one parameter: the ID of the e-Spot. There is a corresponding task command assigned to validate the parameter.

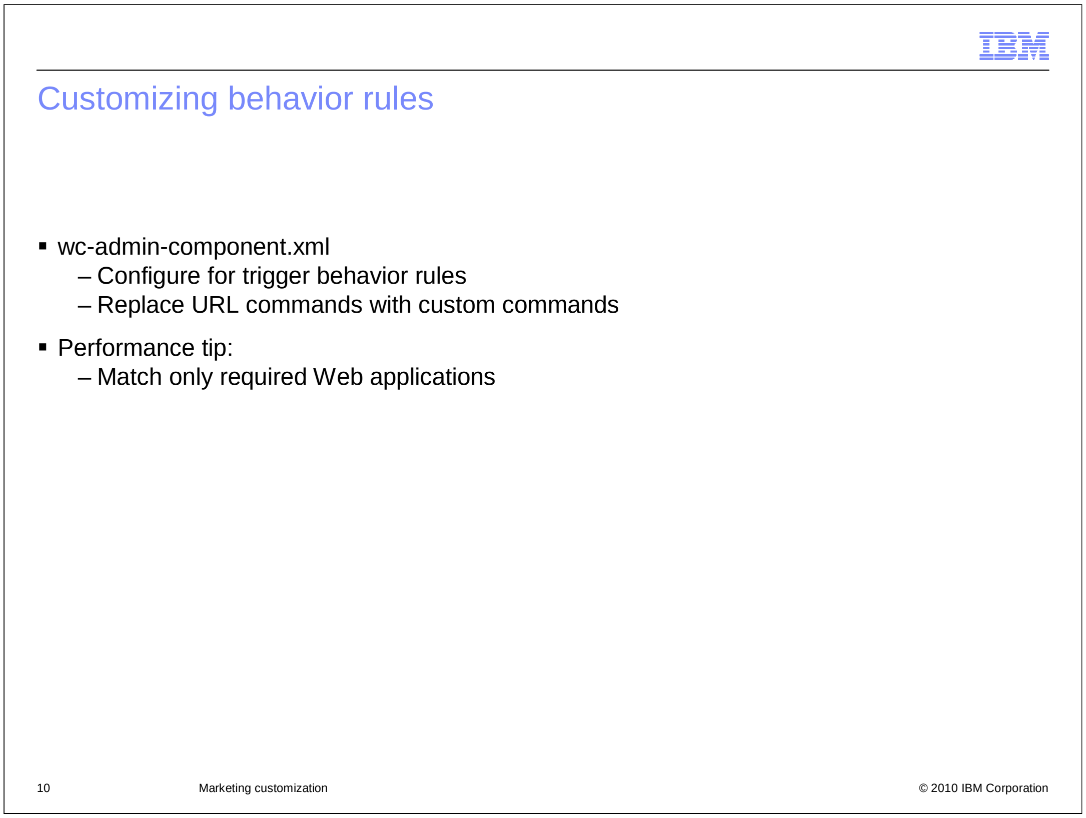


Behavior rules

- Optional
- React to shopper interaction with the store
  - Match command name*
  - Action*
    - Process as a trigger (customer registers)
    - Record for a target (social commerce interaction)
    - Match current request (current page)
    - Custom (call custom task command)
  - Variables
    - Name-value pair to match

*Required to define a behavior rule

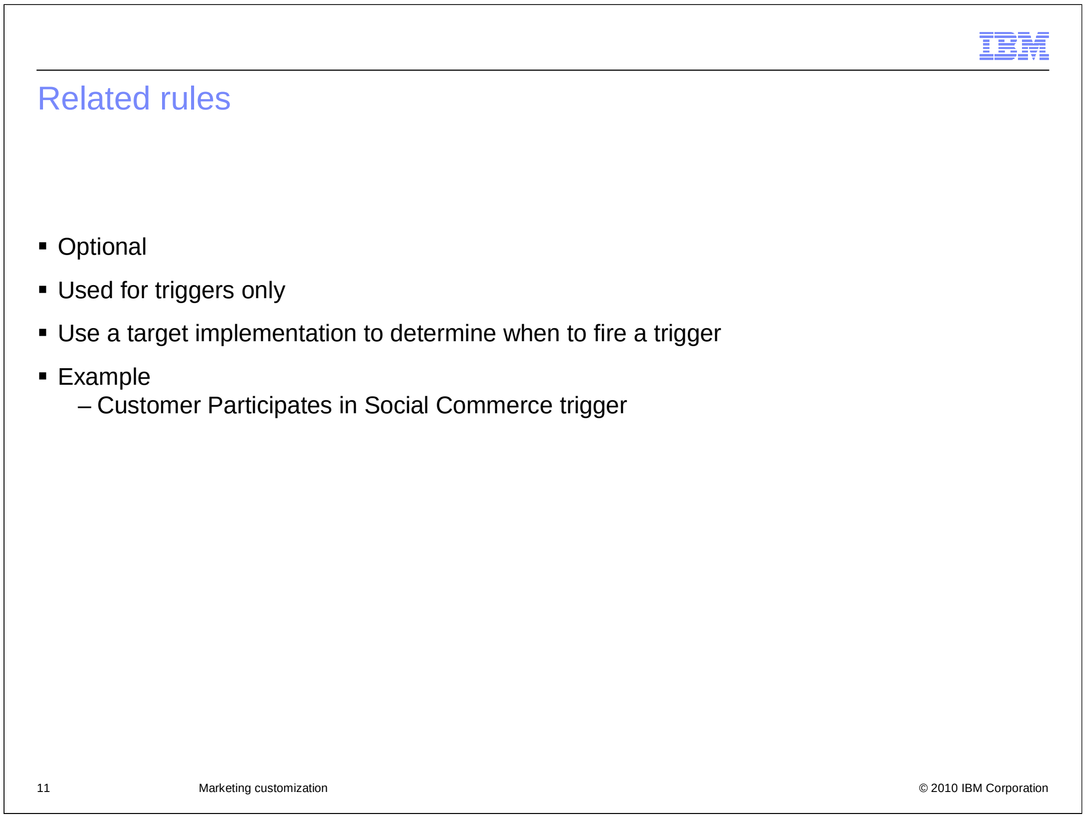
Behavior rules are optional and can be used by triggers, targets and actions. The purpose of a behavior rule is to allow the marketing engine to react to shopper interactions with the store. Like implementation rules, behavior rules are defined in the DMELETEMPLATE table. Behavior rules have two required attributes and many optional ones. The first required attribute is ‘command’, which is the name of the command to match. The second is ‘action’, which specifies what to do when the command is matched. Since behavior rules can be defined for any campaign element type, various actions are possible. The first is to send the command as an event to the marketing engine so that it can be processed as a trigger. An example of this is the registration command which fires the ‘Customer Registers’ Dialog activity trigger. The second type of action is to record the occurrence of the command for an associated target. For example each time the shopper creates social commerce content it is recorded until they reach the number of times required to match a target. The third type of action does not actually take any action at all. In this case, the behavior rule is used as a template to match the current request against. This is used by the current behavior targets such as browsing products or categories. Finally, the fourth type of action is custom. In this case, the command is not handled by the marketing runtime but by a specified custom task command. Optionally, variables can be specified for matching name value pairs.



Customizing behavior rules

- wc-admin-component.xml
  - Configure for trigger behavior rules
  - Replace URL commands with custom commands
- Performance tip:
  - Match only required Web applications

The configuration file wc-admin-component.xml contains several options for behavior rules. Trigger rules, specified using the send action, must have an entry mapping the command name in the behavior rule to the fully qualified controller command. You can also specify Web applications to ignore when checking for the command. For example you can filter out user registration commands coming from the organization administration tool. For URL command names that are being matched and recorded, there is no entry required in this file. Adding an entry to this file allows you to replace a URL command with your custom version. For example, you can replace CategoryDisplay with MyCompanyCategoryDisplay and not need to modify the behavior rule in the database. The wc-admin-component.xml file gives examples of both types of customization.

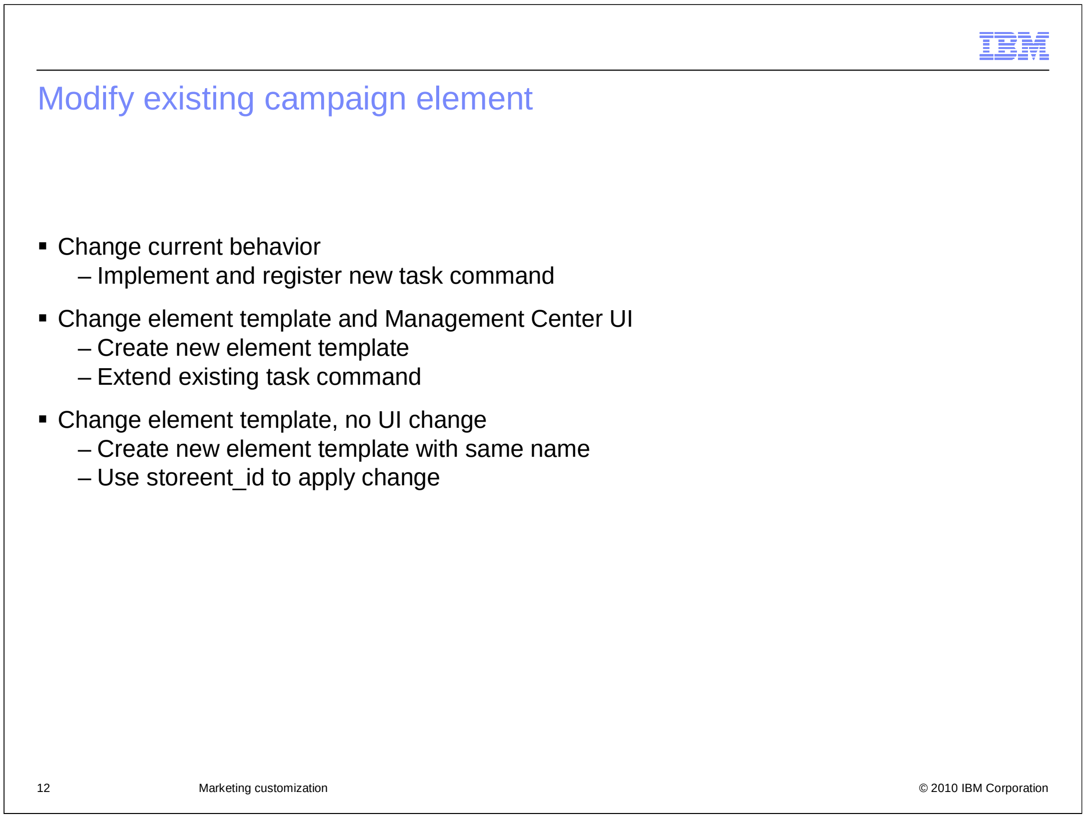


Related rules

- Optional
- Used for triggers only
- Use a target implementation to determine when to fire a trigger
- Example
  - Customer Participates in Social Commerce trigger

11 Marketing customization © 2010 IBM Corporation

Currently the related rule option is only used for the ‘Customer Participates in Social Commerce’ trigger but it offers many options for custom triggers. When the behavior rule for a trigger matches, the related target implementation is evaluated to determine if the trigger event should be fired. Any target based on recorded behavior can be turned into a Dialog activity trigger.



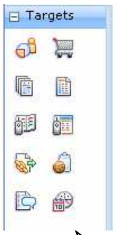
Modify existing campaign element

- Change current behavior
  - Implement and register new task command
- Change element template and Management Center UI
  - Create new element template
  - Extend existing task command
- Change element template, no UI change
  - Create new element template with same name
  - Use storeent_id to apply change

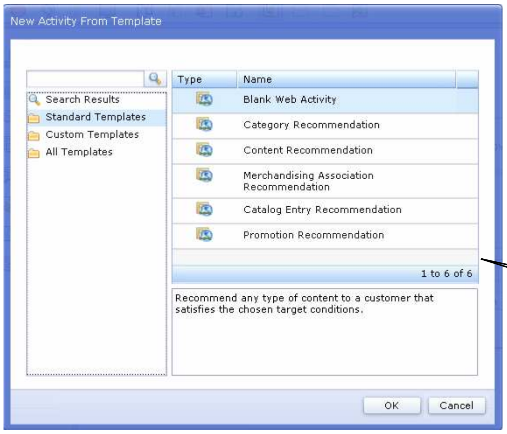
In addition to creating new campaign elements, you can also modify existing ones. To change the behavior of an existing campaign element, create a task command implementation that implements the campaign element interface. Register the new implementation in the CMDREG database table. To extend or change the behavior of an existing campaign element, you first consider whether or not you require changes to the Management Center UI. If you require Management Center UI changes, you follow a process similar to creating a new campaign element. First, create a new campaign element template definition. Then extend the existing campaign element task command and finally, modify the Management Center UI to use the new campaign element. If your behavior change does not require changes to the Management Center UI, first create a new campaign element template definition. The new template definition should have the same name as the existing campaign element template, but should have a different STOREENT_ID. If the new campaign element template applies to all stores, then the STOREENT_ID should be 0. If it applies to a specific store, then the STOREENT_ID should be the applicable STOREENT_ID of the store.

Management Center

- Customize palette
- Customize template picker



New target here


New Activity From Template

- Search Results
- Standard Templates
- Custom Templates
- All Templates

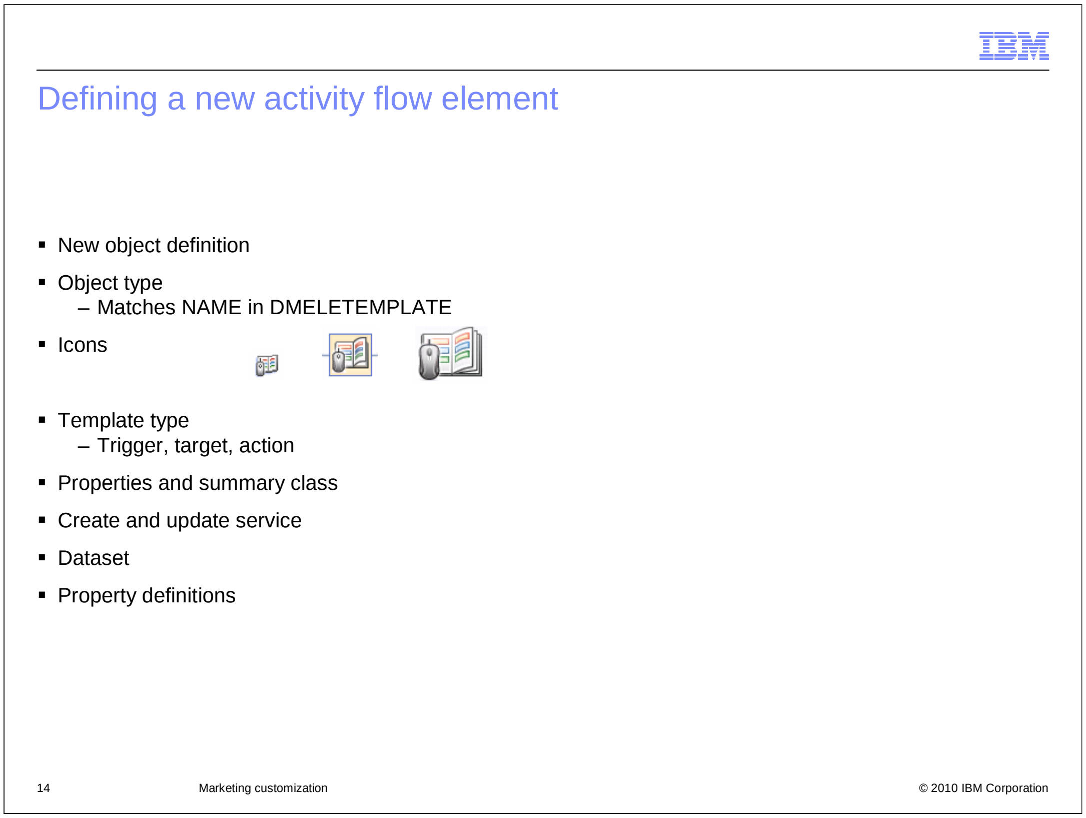
Type	Name
	Blank Web Activity
	Category Recommendation
	Content Recommendation
	Merchandising Association Recommendation
	Catalog Entry Recommendation
	Promotion Recommendation

1 to 6 of 6

Recommend any type of content to a customer that satisfies the chosen target conditions.

New template here

This section looks at two types of customization to the Management Center Marketing tool. The first is adding a new element to the activity builder palette and the second is customizing the template picker.

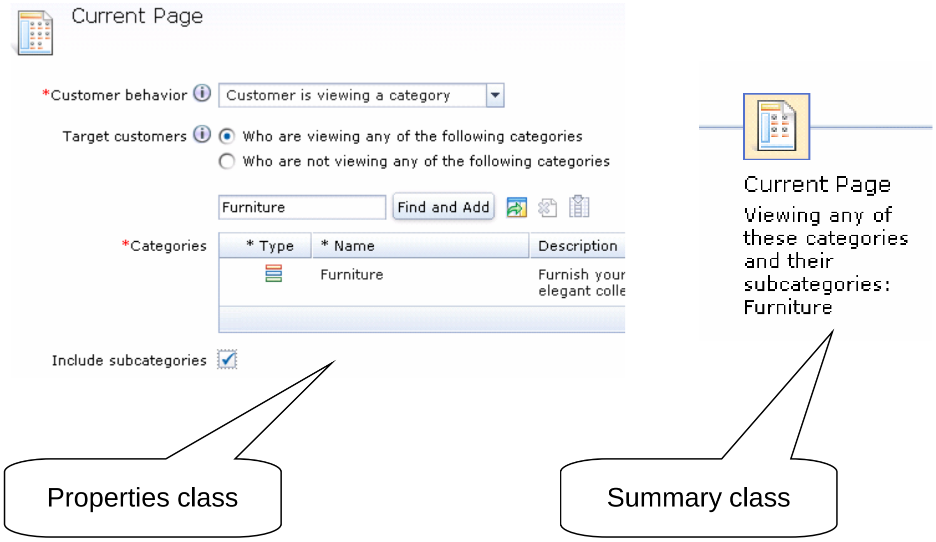


Defining a new activity flow element

- New object definition
- Object type
  - Matches NAME in DMELETEMPLATE
- Icons
- Template type
  - Trigger, target, action
- Properties and summary class
- Create and update service
- Dataset
- Property definitions

Each activity element requires an object definition. For a custom element, the new class extends from mktFlowElementObjectDefinition. The class definition also has several attributes. The objectType attribute is the unique name of the flow element. The icon attributes for flow, palette and header define the name of the icon resources to use. Triggers, targets and actions are all considered flow elements so the particular type must be specified as an attribute. The properties class and summary class attributes are optional. They should be specified if you need to provide input to configure the flow element. In addition to these attributes, the object definition should contain a "create and update" service. Optionally, the object definition might also specify a dataset and one or more property definitions to be used by the properties view class.

Element properties and summary



Current Page

*Customer behavior ⓘ Customer is viewing a category ▾

Target customers ⓘ ● Who are viewing any of the following categories
○ Who are not viewing any of the following categories

Furniture [Find and Add]

* Type	* Name	Description
☰	Furniture	Furnish your elegant colle

*Categories

Include subcategories ☑

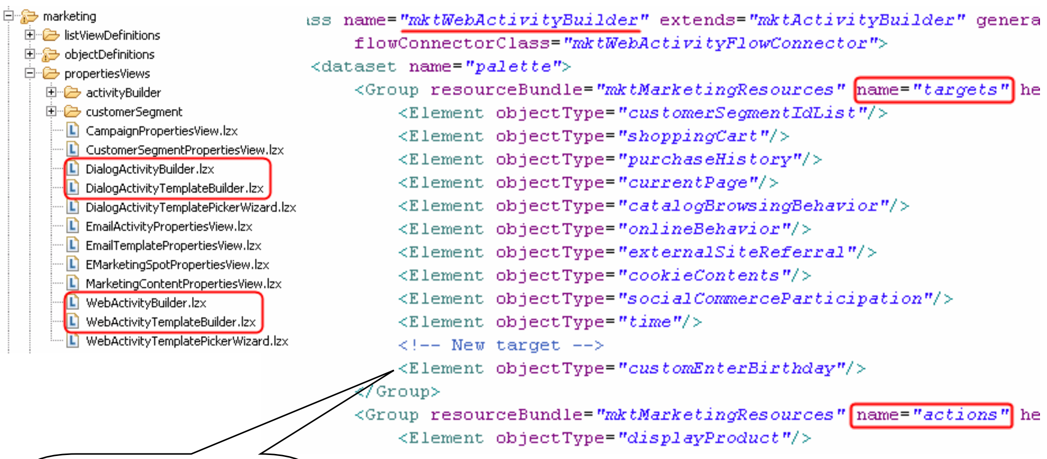
Current Page
Viewing any of these categories and their subcategories: Furniture

Properties class

Summary class

The properties class is responsible for providing input fields for all the arguments defined in the element implementation template. The summary class creates a text string based on the currently selected values. The text string is displayed underneath the element in the activity flow.

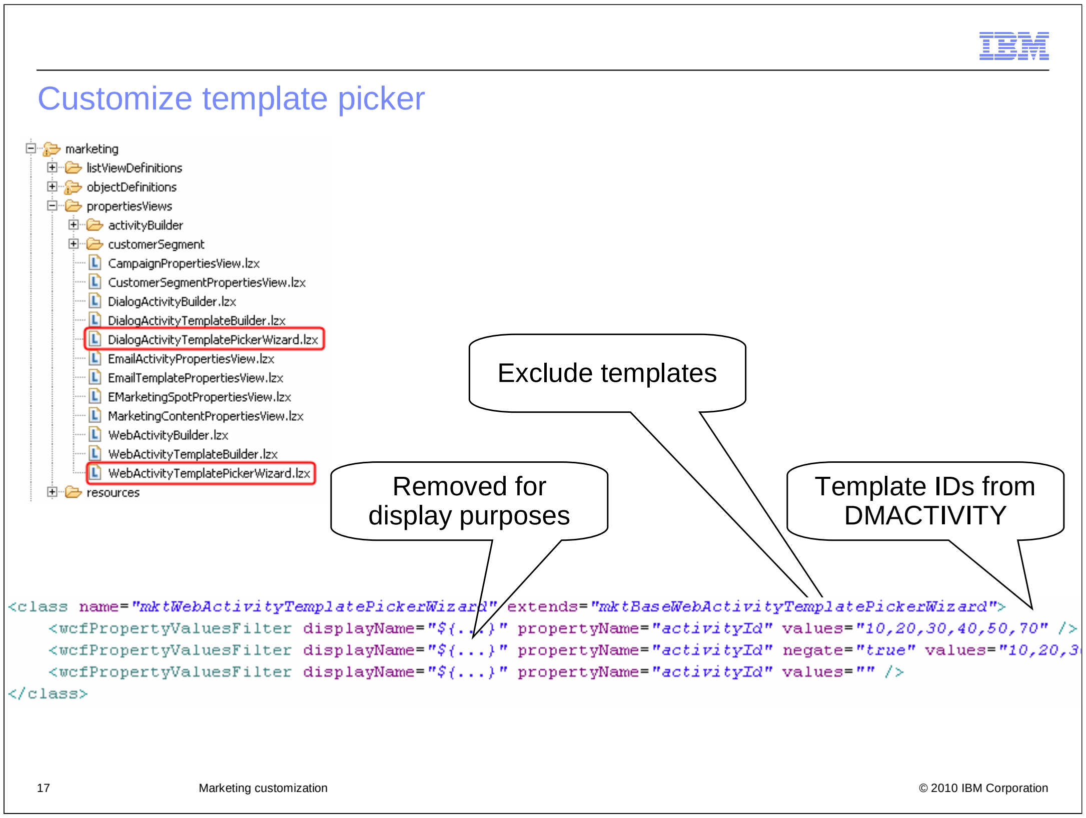
Customize palette


marketing
listViewDefinitions
objectDefinitions
propertiesViews
activityBuilder
customerSegment
CampaignPropertiesView.lzx
CustomerSegmentPropertiesView.lzx
DialogActivityBuilder.lzx
DialogActivityTemplateBuilder.lzx
DialogActivityTemplatePickerWizard.lzx
EmailActivityPropertiesView.lzx
EmailTemplatePropertiesView.lzx
EMarketingSpotPropertiesView.lzx
MarketingContentPropertiesView.lzx
WebActivityBuilder.lzx
WebActivityTemplateBuilder.lzx
WebActivityTemplatePickerWizard.lzx

```
ıss name="mktWebActivityBuilder" extends="mktActivityBuilder" genera
    flowConnectorClass="mktWebActivityFlowConnector">
<dataset name="palette">
    <Group resourceBundle="mktMarketingResources" name="targets" he
        <Element objectType="customerSegmentIdList"/>
        <Element objectType="shoppingCart"/>
        <Element objectType="purchaseHistory"/>
        <Element objectType="currentPage"/>
        <Element objectType="catalogBrowsingBehavior"/>
        <Element objectType="onlineBehavior"/>
        <Element objectType="externalSiteReferral"/>
        <Element objectType="cookieContents"/>
        <Element objectType="socialCommerceParticipation"/>
        <Element objectType="time"/>
        <!-- New target -->
        <Element objectType="customEnterBirthday"/>
    </Group>
    <Group resourceBundle="mktMarketingResources" name="actions" he
        <Element objectType="displayProduct"/>
```

Custom target

Once the object definition for a custom flow element is complete, customizing the activity builder palette is simple. There are four separate activity builder files and you need to update each one that you want your new element to be visible in. Adding a new flow element to the palette is done by creating a new element instance with the name of the custom type in the corresponding group for targets, actions or triggers.



Customize template picker

- marketing
  - listViewDefinitions
  - objectDefinitions
  - propertiesViews
    - activityBuilder
    - customerSegment
    - CampaignPropertiesView.lzx
    - CustomerSegmentPropertiesView.lzx
    - DialogActivityBuilder.lzx
    - DialogActivityTemplateBuilder.lzx
    - DialogActivityTemplatePickerWizard.lzx
    - EmailActivityPropertiesView.lzx
    - EmailTemplatePropertiesView.lzx
    - EMarketingSpotPropertiesView.lzx
    - MarketingContentPropertiesView.lzx
    - WebActivityBuilder.lzx
    - WebActivityTemplateBuilder.lzx
    - WebActivityTemplatePickerWizard.lzx
  - resources

Exclude templates

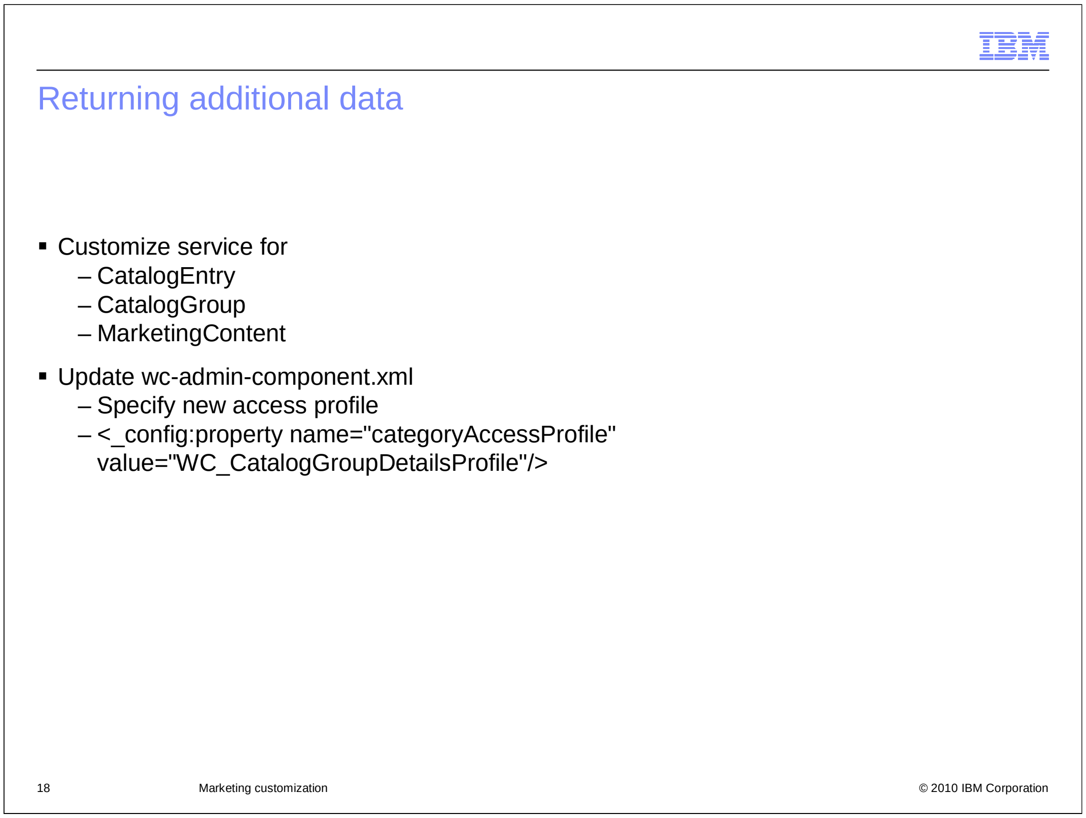
Removed for display purposes

Template IDs from DMACTIVITY

```
<class name="mktWebActivityTemplatePickerWizard" extends="mktBaseWebActivityTemplatePickerWizard">
    <wcfPropertyValuesFilter displayName="${...}" propertyName="activityId" values="10,20,30,40,50,70" />
    <wcfPropertyValuesFilter displayName="${...}" propertyName="activityId" negate="true" values="10,20,3
    <wcfPropertyValuesFilter displayName="${...}" propertyName="activityId" values="" />
</class>
```

17 Marketing customization © 2010 IBM Corporation

There are two template picker files, one for Web activities and one for Dialog activities. Each template group is a wcfPropertyValue filter. Templates are included or excluded by specifying the template ID from the DMACTIVITY table. You can add custom templates to the existing groups or create new groups.

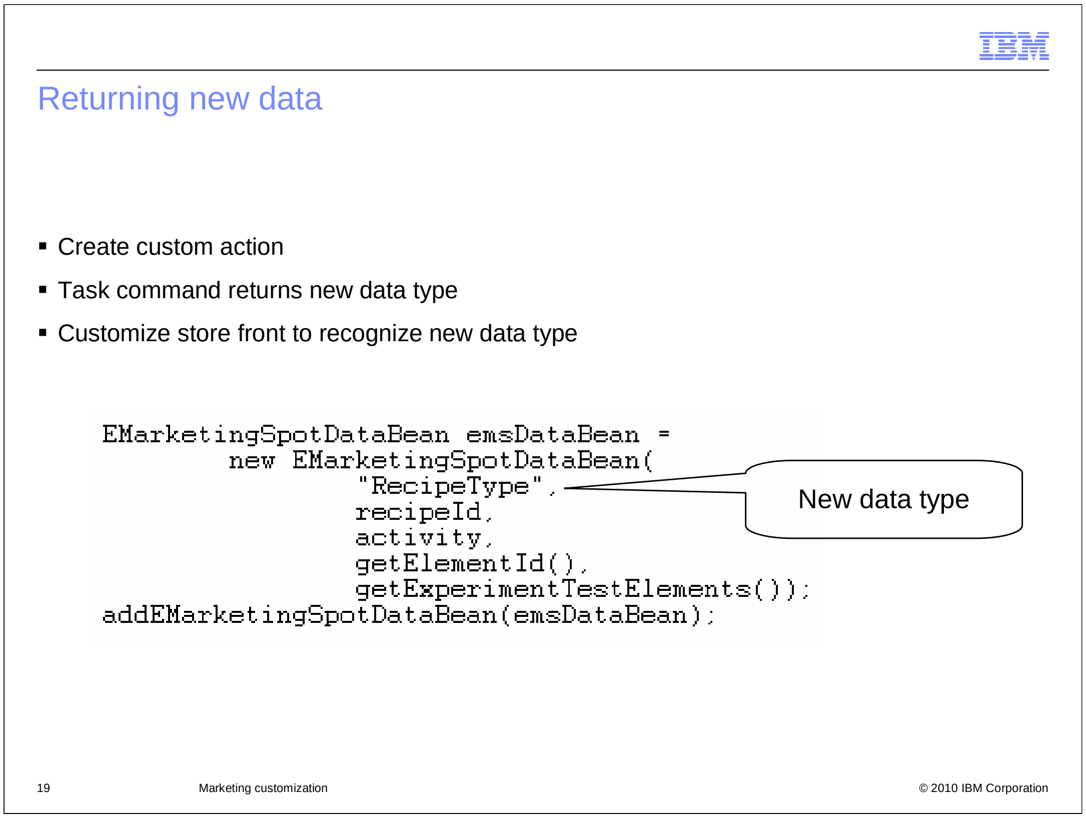


Returning additional data

- Customize service for
  - CatalogEntry
  - CatalogGroup
  - MarketingContent
- Update wc-admin-component.xml
  - Specify new access profile
  - <_config:property name="categoryAccessProfile" value="WC_CatalogGroupDetailsProfile"/>

18 Marketing customization © 2010 IBM Corporation

The marketing runtime uses service calls to load the data associated with the business objects that are returned to the e-Marketing Spot JSP™ snippet. If you require additional data to be returned as part of the CatalogEntry, CatalogGroup or MarketingContent objects, you can customize the appropriate service. When you’ve completed this customization, update the wc-admin-component.xml file with the new access profile name for the custom business object. An example of the default access profile configuration for the CatalogGroup access profile is shown here.



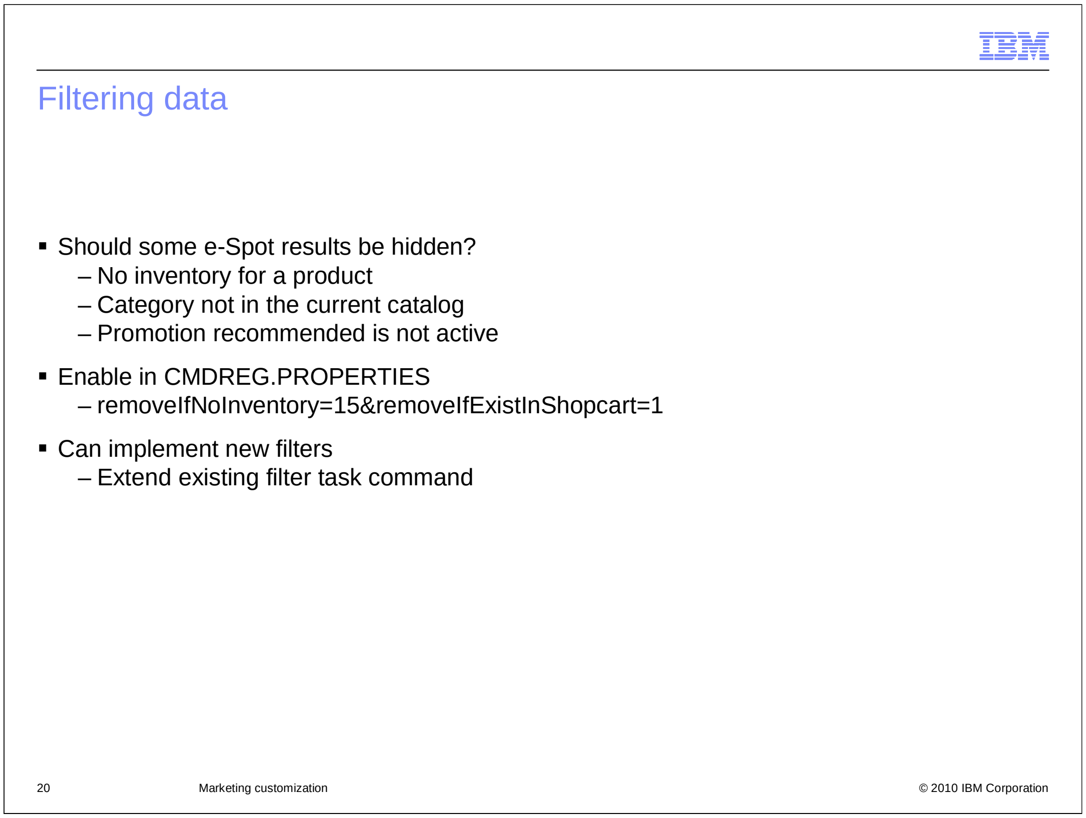
Returning new data

- Create custom action
- Task command returns new data type
- Customize store front to recognize new data type

```
EMarketingSpotDataBean emsDataBean =
        new EMarketingSpotDataBean(
                "RecipeType",
                recipeId,
                activity,
                getElementId(),
                getExperimentTestElements());
addEMarketingSpotDataBean(emsDataBean);
```

New data type

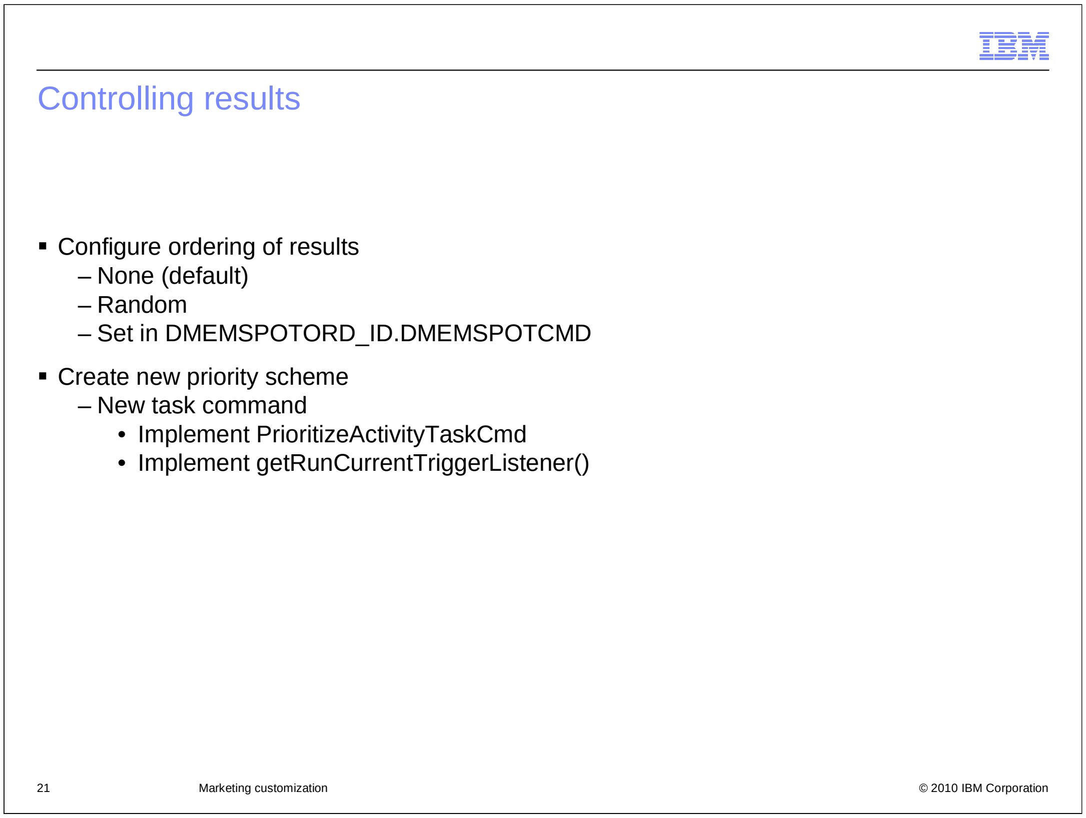
If you want to return completely new data types to an e-Marketing Spot, you need to create a custom action element. The task command that implements the action retrieves the data and sends it the EMarketingSpotDataBean along with a name for the new type. On the store front side, you need to update your e-Marketing Spot JSP snippet to look for the new data type and handle the results accordingly.



Filtering data

- Should some e-Spot results be hidden?
  - – No inventory for a product
  - – Category not in the current catalog
  - – Promotion recommended is not active
- Enable in CMDREG.PROPERTIES
  - – removeIfNoInventory=15&removeIfExistInShopcart=1
- Can implement new filters
  - – Extend existing filter task command

Data filters can be used to hide results that would otherwise be displayed in an e-Marketing Spot. Several filter options exist for each of the three supported data types. You can enable and disable filters by updating the properties column of the CMDREG table with the required settings. To create a new filter, extend the appropriate filter task command and register your new implementation in CMDREG.



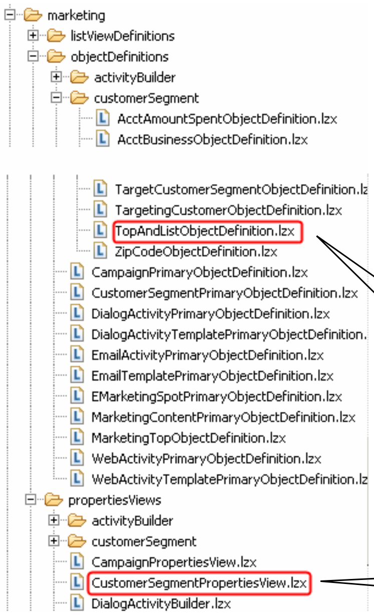
Controlling results

- Configure ordering of results
  - None (default)
  - Random
  - Set in DMEMSPOTORD_ID.DMEMSPOTCMD
- Create new priority scheme
  - New task command
    - Implement PrioritizeActivityTaskCmd
    - Implement getRunCurrentTriggerListener()

21 Marketing customization © 2010 IBM Corporation

After filtering, additional control over displayed results can be achieved by configuring the ordering schema or implementing a new priority scheme. By default, no ordering is done on results returned to an e-Marketing Spot. A random ordering scheme can be applied to any of the three standard data types. Ordering can be enabled for individual e-Marketing Spots or for all e-Marketing Spots in a store. The ordering scheme is set in column DMEMSOPTORD_ID of the DMEMSPOTCMD table. If you want to create a new prioritization scheme, write a new task command that implements the PrioritizeActivityTaskCmd interface. The method getRunCurrentTriggerListener is called for each activity to determine if it should be run. Implement your new prioritization scheme and return true or false for each activity. As usual, you need to register your new implementation in the CMDREG table.

Customer segments



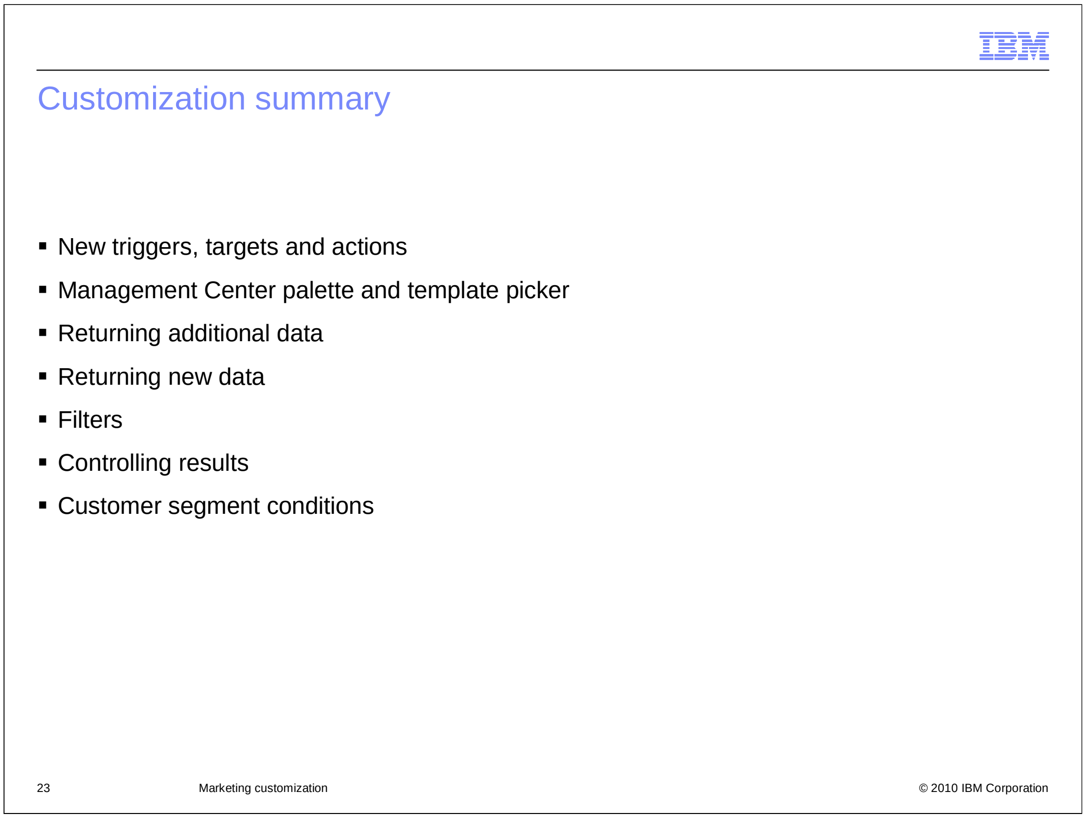
```
marketing
  listViewDefinitions
  objectDefinitions
    activityBuilder
    customerSegment
      AcctAmountSpentObjectDefinition.lzx
      AcctBusinessObjectDefinition.lzx

      TargetCustomerSegmentObjectDefinition.lz
      TargetingCustomerObjectDefinition.lzx
      TopAndListObjectDefinition.lzx
      ZipCodeObjectDefinition.lzx
    CampaignPrimaryObjectDefinition.lzx
    CustomerSegmentPrimaryObjectDefinition.lzx
    DialogActivityPrimaryObjectDefinition.lzx
    DialogActivityTemplatePrimaryObjectDefinition.
    EmailActivityPrimaryObjectDefinition.lzx
    EmailTemplatePrimaryObjectDefinition.lzx
    EMarketingSpotPrimaryObjectDefinition.lzx
    MarketingContentPrimaryObjectDefinition.lzx
    MarketingTopObjectDefinition.lzx
    WebActivityPrimaryObjectDefinition.lzx
    WebActivityTemplatePrimaryObjectDefinition.lz
  propertiesViews
    activityBuilder
    customerSegment
    CampaignPropertiesView.lzx
    CustomerSegmentPropertiesView.lzx
    DialogActivityBuilder.lzx
```

- Remove a condition
- Add a new condition
  - Runtime implementation
  - Object definition
  - Properties view
  - Serialization JSP

Add new object definition

Remove condition, add new properties view

You can remove a condition from the customer segment UI by removing its instance object in the file CustomerSegmentPropertiesView.lzx. To add a custom condition, you need to implement the server-side logic for runtime evaluation of the condition. The classes CheckUserInMemberGroupCmdImpl and SegmentSaveControllerCmdImpl should be extended. To add the condition to Management Center, you need to create a new object definition that extends from wcfChildObjectDefinition. You also need to create a properties view to capture the values for the new condition. The Struts extension file also needs to be updated with a new action mapping for the condition element.

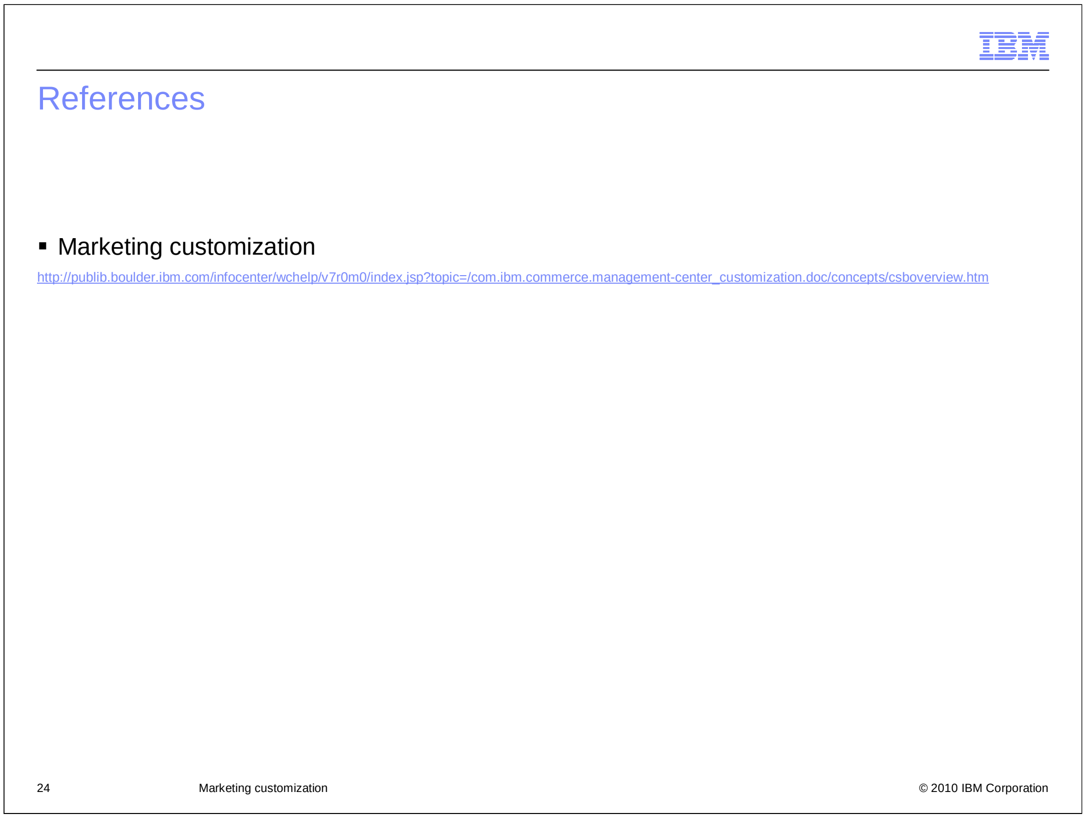


Customization summary

- New triggers, targets and actions
- Management Center palette and template picker
- Returning additional data
- Returning new data
- Filters
- Controlling results
- Customer segment conditions

23 Marketing customization © 2010 IBM Corporation

This presentation covered many different customization scenarios beginning with creating custom campaign elements such as triggers, targets and actions. Next, you saw how custom campaign elements can be added to the Management Center. The presentation also examined how to modify the data returned to an e-Marketing Spot and how to control the results displayed in an e-Marketing Spot. The presentation concluded with a look at adding customer segment conditions. If you are interested in learning more about customization, you can continue on to the customization labs in IBM Education Assistant.

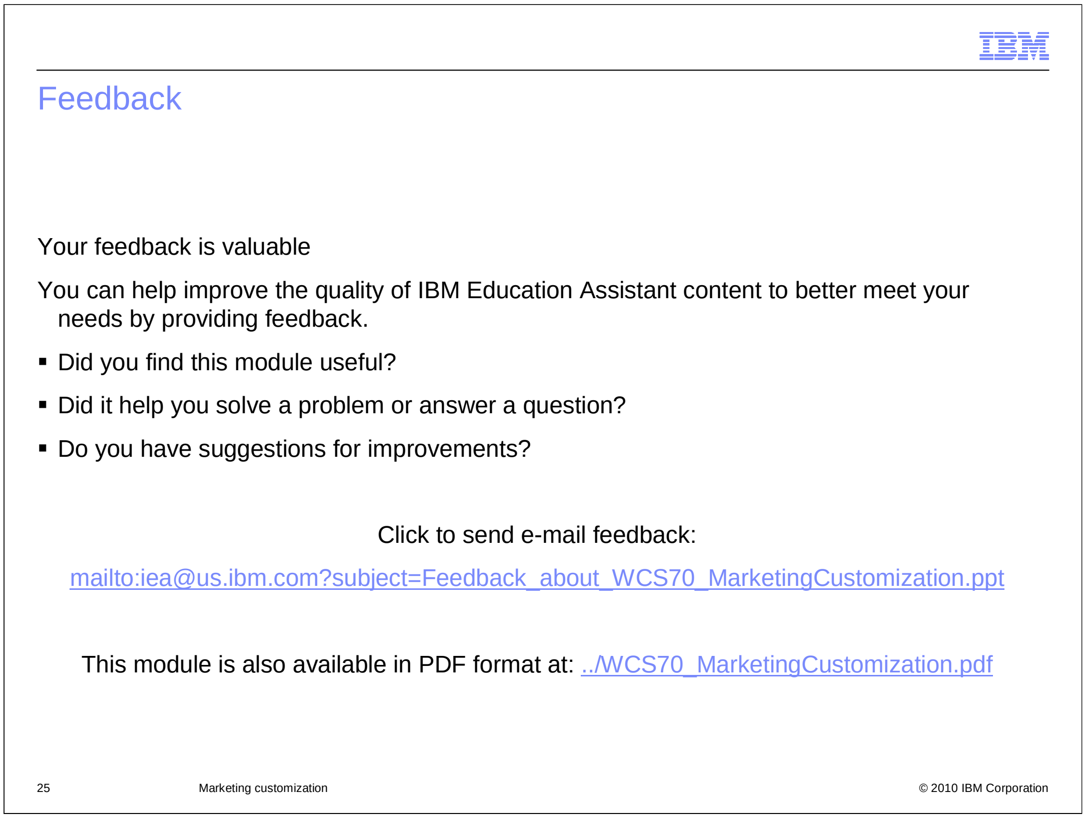


References

- Marketing customization

http://publib.boulder.ibm.com/infocenter/wchelp/v7r0m0/index.jsp?topic=/com.ibm.commerce.management-center_customization.doc/concepts/csboverview.htm

This slide contains a link to the marketing customization topic in the WebSphere Commerce Information Center.



Feedback

Your feedback is valuable

You can help improve the quality of IBM Education Assistant content to better meet your needs by providing feedback.

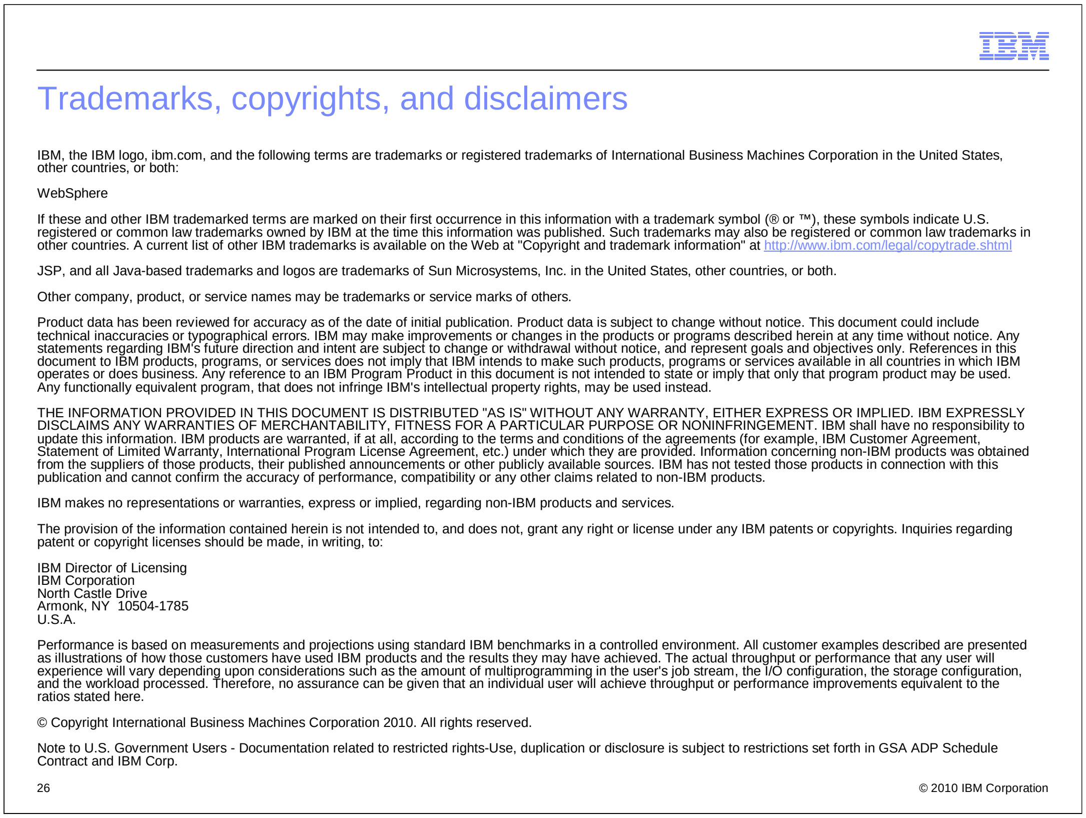
- Did you find this module useful?
- Did it help you solve a problem or answer a question?
- Do you have suggestions for improvements?

Click to send e-mail feedback:

mailto:iea@us.ibm.com?subject=Feedback_about_WCS70_MarketingCustomization.ppt

This module is also available in PDF format at: ../WCS70_MarketingCustomization.pdf

You can help improve the quality of IBM Education Assistant content by providing feedback.



Trademarks, copyrights, and disclaimers

Product data has been reviewed for accuracy as of the date of initial publication. Product data is subject to change without notice. This document could include technical inaccuracies or typographical errors. IBM may make improvements or changes in the products or programs described herein at any time without notice. Any statements regarding IBM's future direction and intent are subject to change or withdrawal without notice, and represent goals and objectives only. References in this document to IBM products, programs, or services does not imply that IBM intends to make such products, programs or services available in all countries in which IBM operates or does business. Any reference to an IBM Program Product in this document is not intended to state or imply that only that program product may be used. Any functionally equivalent program, that does not infringe IBM's intellectual property rights, may be used instead.

THE INFORMATION PROVIDED IN THIS DOCUMENT IS DISTRIBUTED "AS IS" WITHOUT ANY WARRANTY, EITHER EXPRESS OR IMPLIED. IBM EXPRESSLY DISCLAIMS ANY WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE OR NONINFRINGEMENT. IBM shall have no responsibility to update this information. IBM products are warranted, if at all, according to the terms and conditions of the agreements (for example, IBM Customer Agreement, Statement of Limited Warranty, International Program License Agreement, etc.) under which they are provided. Information concerning non-IBM products was obtained from the suppliers of those products, their published announcements or other publicly available sources. IBM has not tested those products in connection with this publication and cannot confirm the accuracy of performance, compatibility or any other claims related to non-IBM products.

IBM makes no representations or warranties, express or implied, regarding non-IBM products and services.

The provision of the information contained herein is not intended to, and does not, grant any right or license under any IBM patents or copyrights. Inquiries regarding patent or copyright licenses should be made, in writing, to:

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Performance is based on measurements and projections using standard IBM benchmarks in a controlled environment. All customer examples described are presented as illustrations of how those customers have used IBM products and the results they may have achieved. The actual throughput or performance that any user will experience will vary depending upon considerations such as the amount of multiprogramming in the user's job stream, the I/O configuration, the storage configuration, and the workload processed. Therefore, no assurance can be given that an individual user will achieve throughput or performance improvements equivalent to the ratios stated here.

Note to U.S. Government Users - Documentation related to restricted rights-Use, duplication or disclosure is subject to restrictions set forth in GSA ADP Schedule Contract and IBM Corp.