Module 22 Behavior Extension

IBM InfoSphere Master Data Management Fundamentals

© 2013 IBM Corporation

Unit objectives

After completing this unit, you should be able to:

- Create a new Behavior Extension using the MDM Workbench
- Understand the Metadata involved in MDM Behavior Extensions
- Understand have Behavior Extensions fit into the MDM Architecture

© 2013 IBM Corporation

The Physical Behavior Extensions

Mechanism for extending the behavior of a Service

- e.g. Notification sent to client if client's account drops below min amount

Behavior extension can modify an existing service using:

- Proprietary business rules (for example, ILOG) or
- Custom written Java code

Can be implemented at various points of the Physical module

- Pre/Post Transaction controller
- Pre/Post Action components

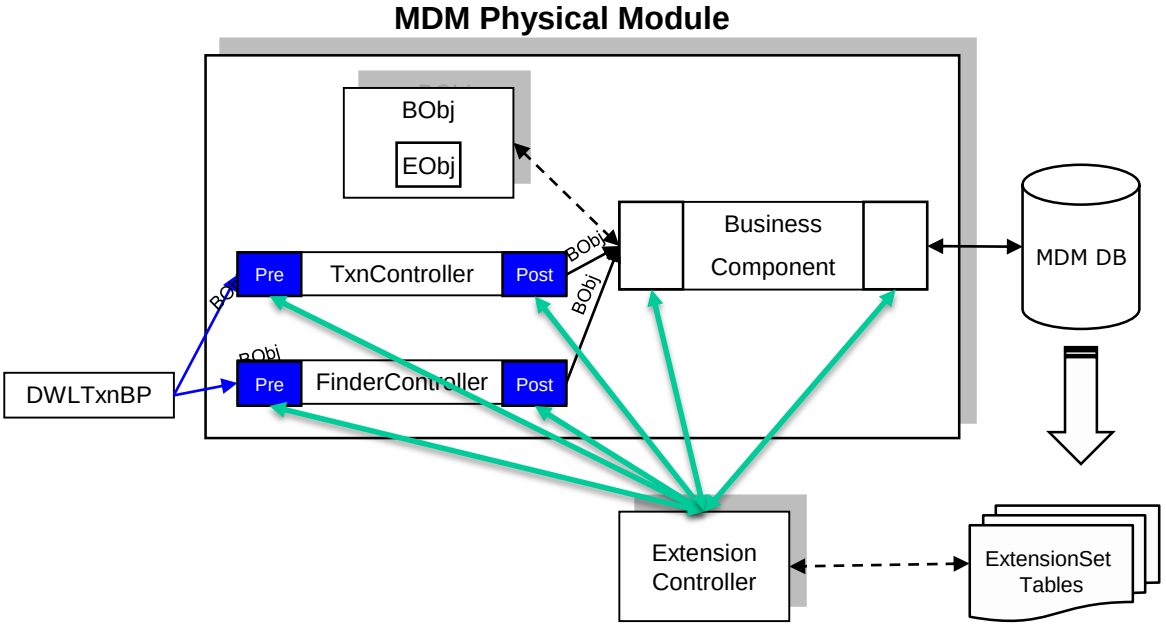
© 2013 IBM Corporation

Behavior Extension Framework

- Extension Controller
  - Runs during extension points
  - Decides if extension should be run
- Behavior Extension Class or Rule
- Extension Metadata
  - ExtensionSet, ExtSetCondval, CDConditionValTP
  - Identifies when Behavior Extension should be run

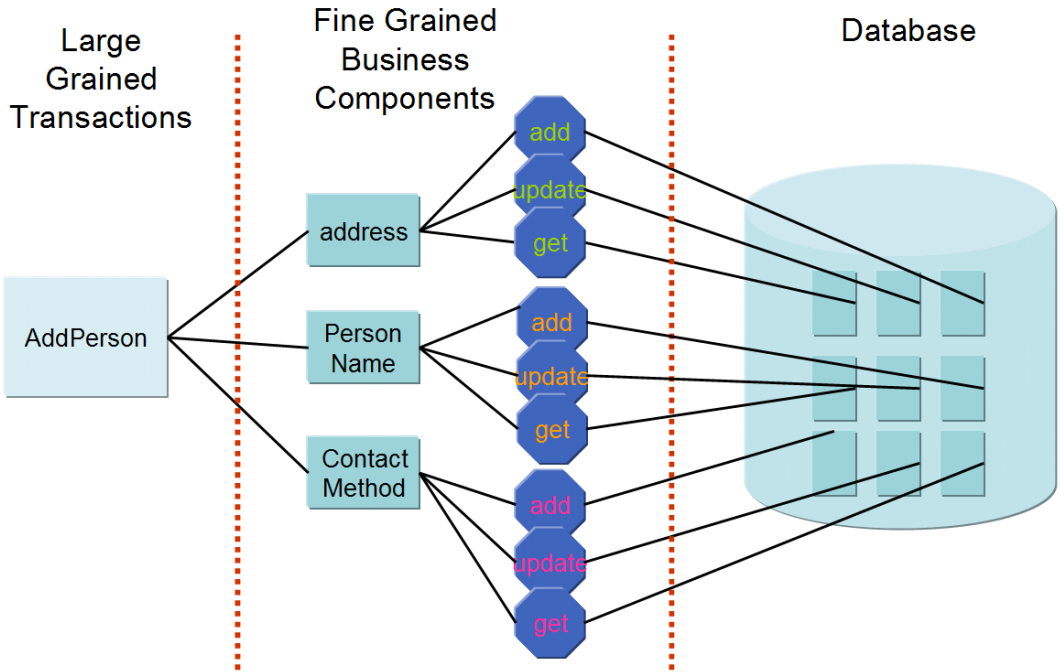
© 2013 IBM Corporation

How it works: Behavior extensions



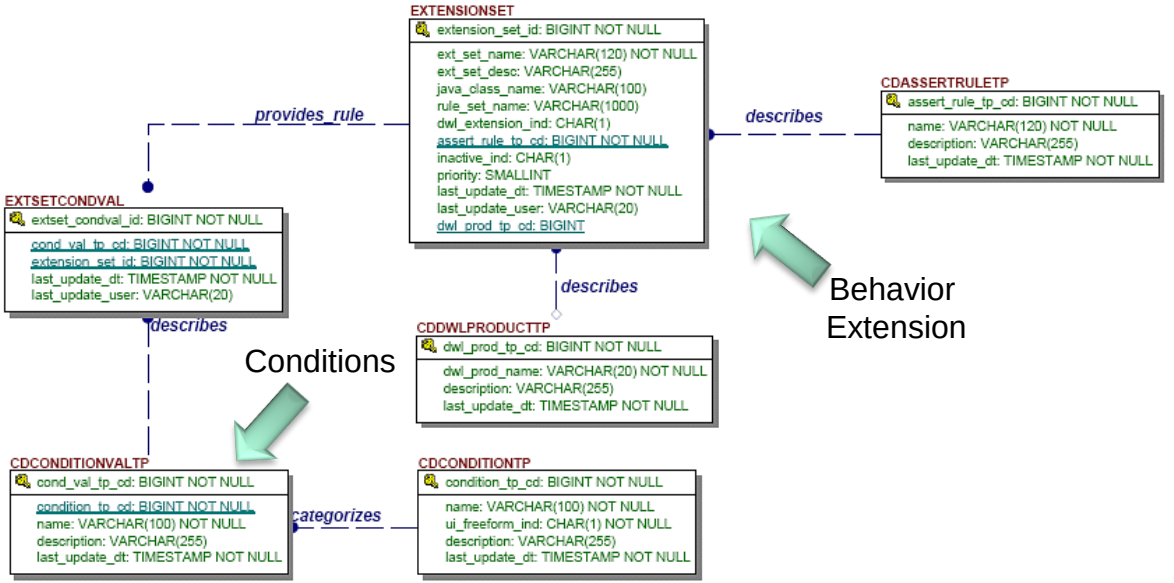
© 2013 IBM Corporation

Controller versus Component



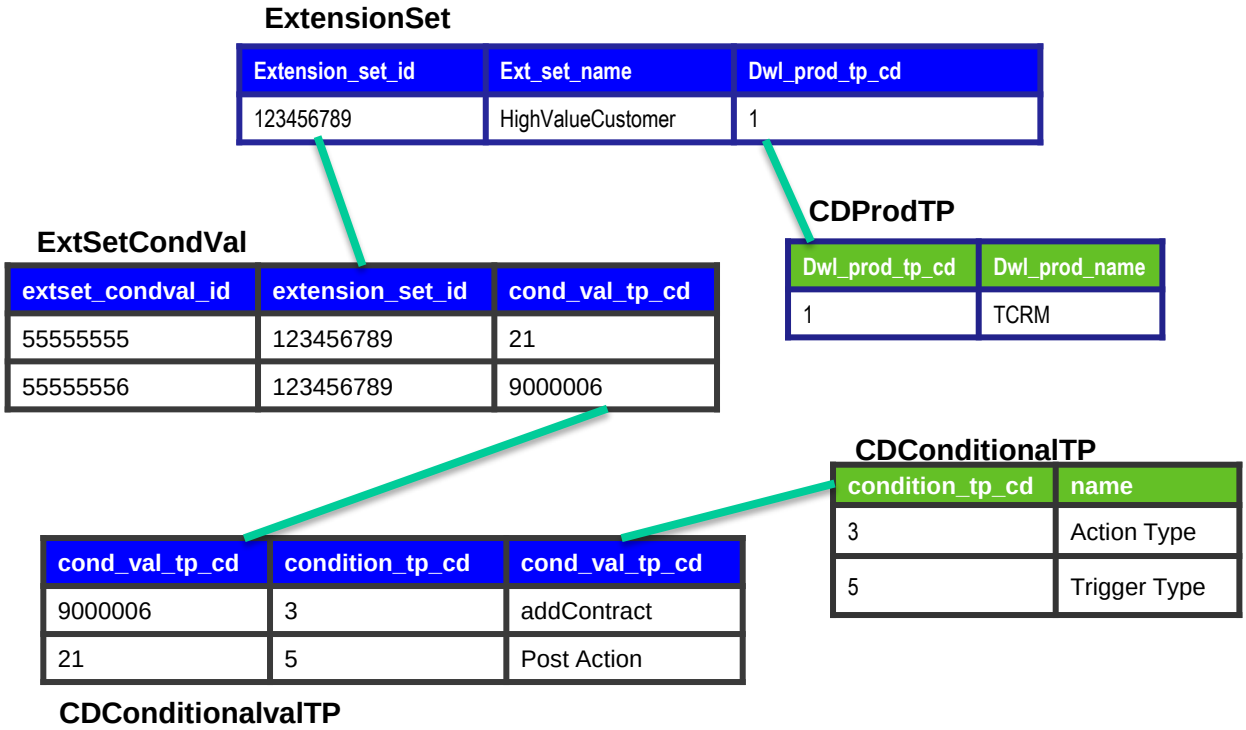
© 2013 IBM Corporation

Extension framework data model (1 of 2)



© 2013 IBM Corporation

Extension framework data model (2 of 2)



© 2013 IBM Corporation

Behavior extension Java class

public class ExtensionBehavior extends ClientJavaExtensionSet {

- public void execute(ExtensionParameters params)
  - The *params* object contains the following objects:
    - **transactionType** (for example, addContract)
    - **TransactionCategoryType** (for example, Persistent)
    - **actionType** (for example, addPersonName)
    - **actionCategoryType** (for example, Persistent)
    - **triggerCategoryType** (for example, Post Transaction)
    - **DWLControl**
    - **WorkingObjectHierarchy** (for example, TCRMPersonNameBObj)
    - **transactionObjectHierarchy** (for example, TCRMPersonBObj)
    - **additionalDataMap** (for example, parameters)
    - **extensionSetStatus**

© 2013 IBM Corporation

Exercise introduction

- In this exercise, you will:
  - Create a new Behavior Extension for the addContractComponent and updateContractComponent
  - Test the new Behavior Extension

  * Make sure to check the Alert table to insure the Extension was invoked.

Unit summary

Having completed this unit, you should be able to:

- Create a new Behavior Extension using the InfoSphere MDM Workbench
- Understand the Metadata involved in InfoSphere MDM Behavior Extensions
- Understand have Behavior Extensions fit into the InfoSphere MDM Architecture

© 2013 IBM Corporation