IBM Case Manager 5.2
Embedded Business Rules

Introduction

- Course Overview
  - Provide an overview of new Embedded Business Rules functionality
  - Describe how existing ICM 5.1.1 customers are impacted
- Target Audience:
  - IBM development, support, and services organizations that work with case management solutions
  - Personnel working with IBM Case Manager customers such as Sales System Engineers, Specialists and SMEs, Consulting Specialists, Field Technical Consultants and Technical Support
- Suggested Prerequisites:
  - Familiarity with IBM Case Manager
  - Rules familiarity
- Version Release Date: September, 2013

Course Objectives

After this course you will be able to:

- Describe the salient features of new Embedded Rules functionality
- Configure ICM to enable Embedded Rules features
- Author Text-based & Table-based business rules from Case Builder, use the rules in Tasks and verify rule execution results from Case Client
- Describe the steps involved in migrating Embedded Rules to full IODM (IBM Operational Decision Manager) product.
- Explain how ICM 5.1.1 customers may be impacted by this new feature
- Troubleshoot rules related issues in your ICM environment.
- Learn about Known Issues, Limitations, Best Practices in rules functionality.

Course Roadmap

- ➔ Introduction
- – Configuring Embedded Rules in ICM environment
- – Rules Authoring in Case Builder
- – Executing Rules as part of a workflow
- – Moving Rules from embedded environment to IODM
- – Impact to ICM 5.1.1 customers
- – Troubleshooting
- – Known Issues, Limitations, Best Practices
- – Course Summary

Introduction

- IBM Case Manager bundles IBM Business Rules Embedded in place of ILOG JRules
  - IBM Business Rules Embedded is an SDK provided by IODM that allows for tight integration into a product
  - Simplified installation
    - No separate installation of ILOG Rules Studio and Rule Execution Server
    - Rules are installed as part of ICM installation
  - Simplified authoring
    - Authoring of Rules from inside Case Builder
    - Simplified Rule Steps provided in Step Builder
    - No need to transfer rules to Rules Execution Server
    - No need to manually set rule WSDL in Process Designer

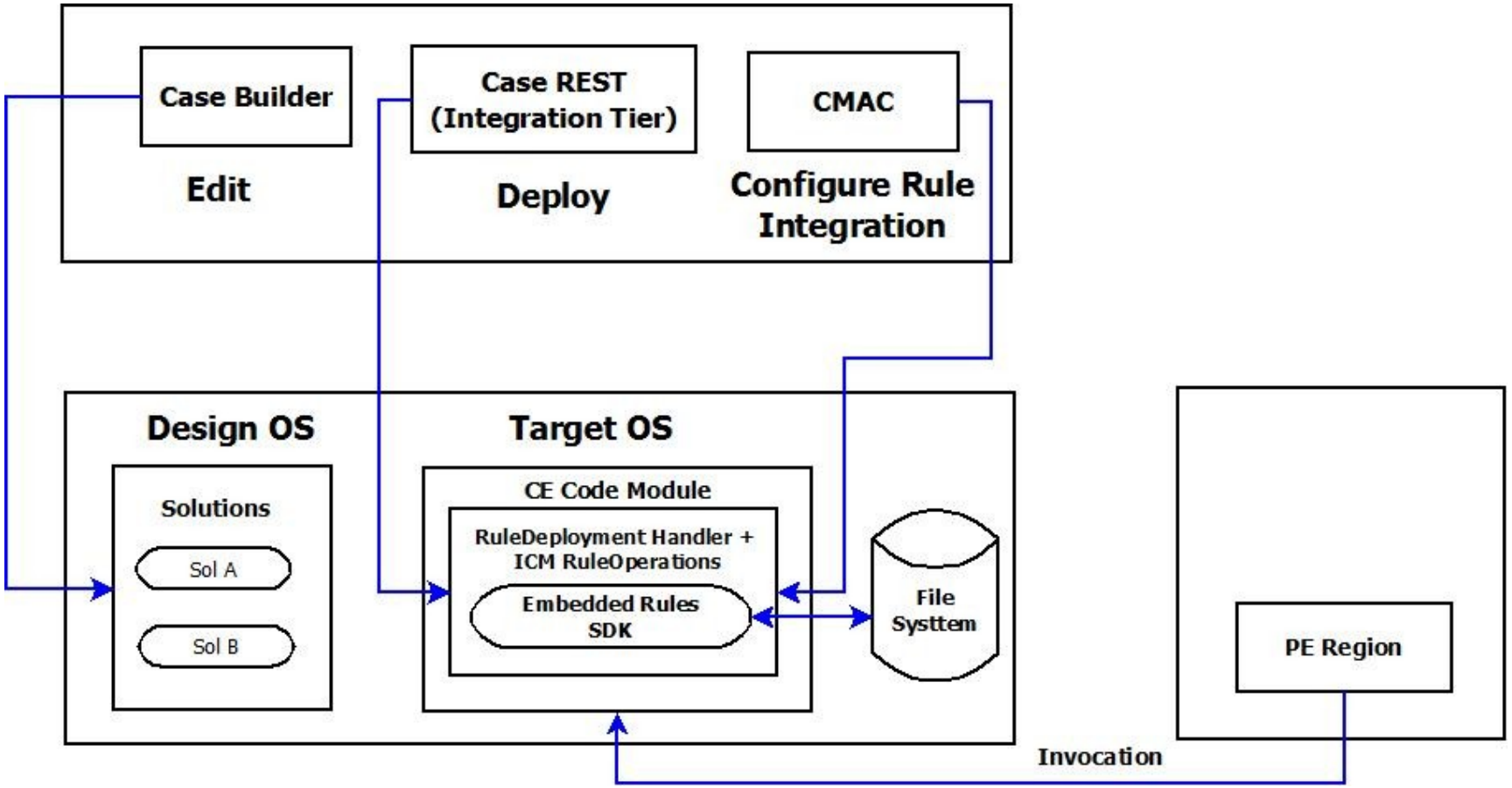
Introduction – Feature Overview

- Better User Experience
- Friendlier for the Business Analyst
- Overall simplification for ICM solution
- Case properties get read/updated automatically
- Limitations in first version of ICM with Embedded Rules
  - Must redeploy entire solution to change a Business Rule
  - Business Rules only available as Task Step
  - Limited change management and governance
  - Languages supported by the IODM SDK are limited compared to what is supported in Case Builder. If the user's locale is not supported by IODM SDK, user will have to write the rule in English.

Comparison between Embedded Rules, JRules 7.1 & IODM

Capabilities	Embeddable Business Rules	Host Bundled JRules 7.1 + Rule Studio	IBM ODM	
			Decision Server	Decision Center
Rule authoring in natural language based on a business vocabulary using rules and decision tables	✓ Business users	✓ Technical users	✓ Technical users	✓ Business users
Rule authoring for business users embeddable in host application web UI allowing authoring in host solution context.	✓ Host application rule repository			✓ Decision Center repository
Business Vocabulary Generation and customization	Vocabulary generated from schema	Host defined vocabulary Customization	✓ Full vocabulary Customization	
Extended rule authoring: packages, ruleflows, decision trees, templates		✓ Technical users	✓ Technical users	✓ Business users
Support for Business event rules in natural language			✓ Technical users	✓ Business users
Rule execution as part of solution	✓ Embeddable Runtime	✓ Rule Execution server	✓ Scalable Server Decision services	
Management and Deployment of rulesets and decision services to execution runtime	✓ API for Ruleset deployment	✓ Management Console	✓ Management Console	✓ Governed Deployment
Centralized business rule repository				✓
Business rules lifecycle governance,				✓
Multiple release management for reliable change mgt				✓
Enterprise-wide collaboration on business rules, including IT/Business Alignment				✓

Embedded Rules Integration - Architecture



Embedded Rules Integration - Architecture

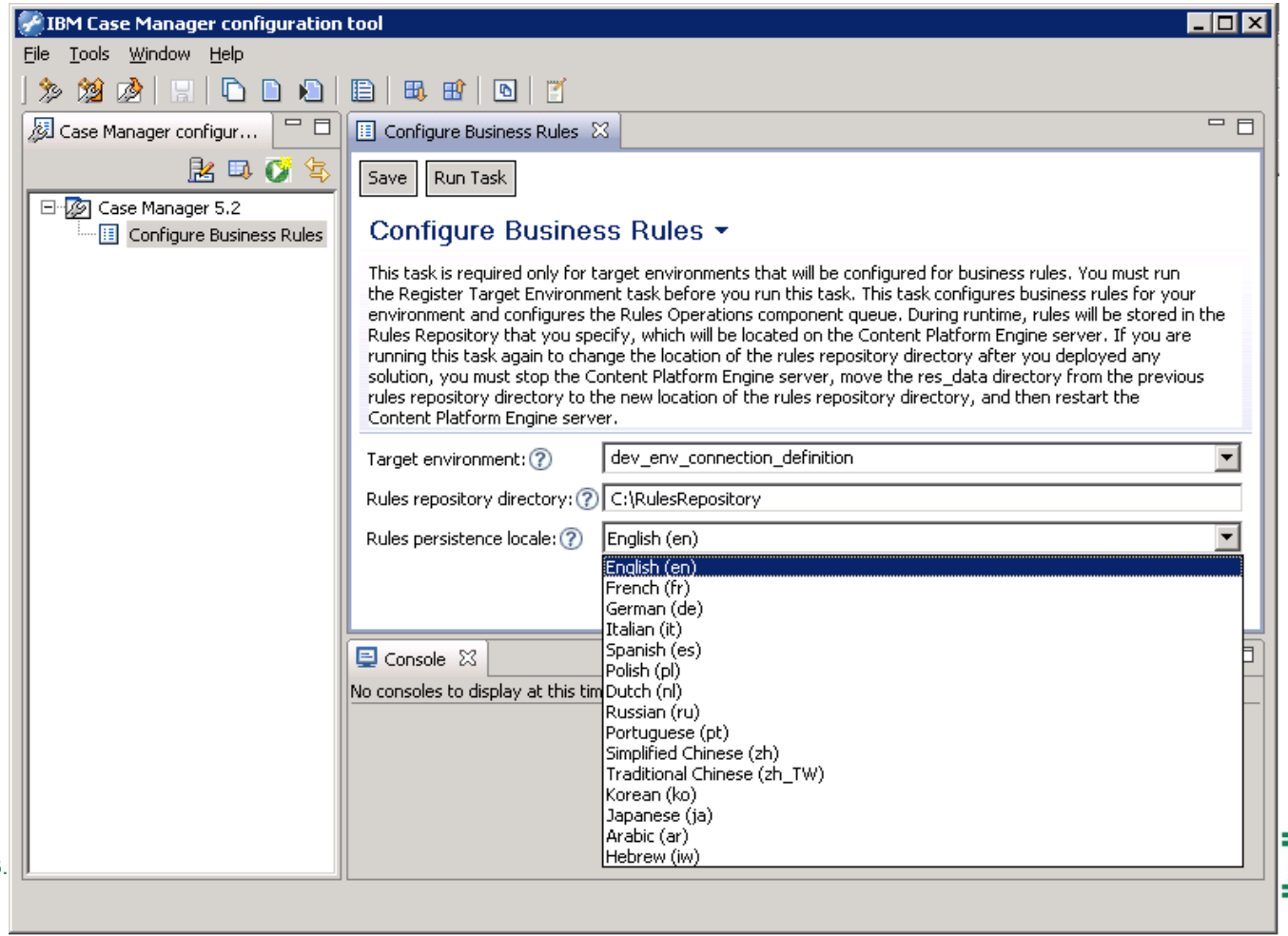
- Dojo based Rule Editor and Decision Table embedded in Case Builder
- Rule Definition artifacts are persisted in CPE (Content Platform Engine) Design Object Store and compiled ruleSet.jar is stored in file system (Rules Repository)
- Embedded SDK is hosted as code module in CPE Target Object store
- Rule runtime runs as a Java component in CPE

Course Roadmap

- Introduction
- ➔ Configuring Embedded Rules in ICM environment
- Rules Authoring in Case Builder
- Executing Rules as part of a workflow
- Moving Rules from embedded environment to IODM
- Impact to ICM 5.1.1 customers
- Troubleshooting
- Known Issues, Limitations, Best Practices
- Course Summary

Configuring Embedded Rules in ICM environment

- Rules feature is disabled by default
- Use CMAC to configure business rules. This task needs to be run for each project area.



Configuring Embedded Rules in ICM environment

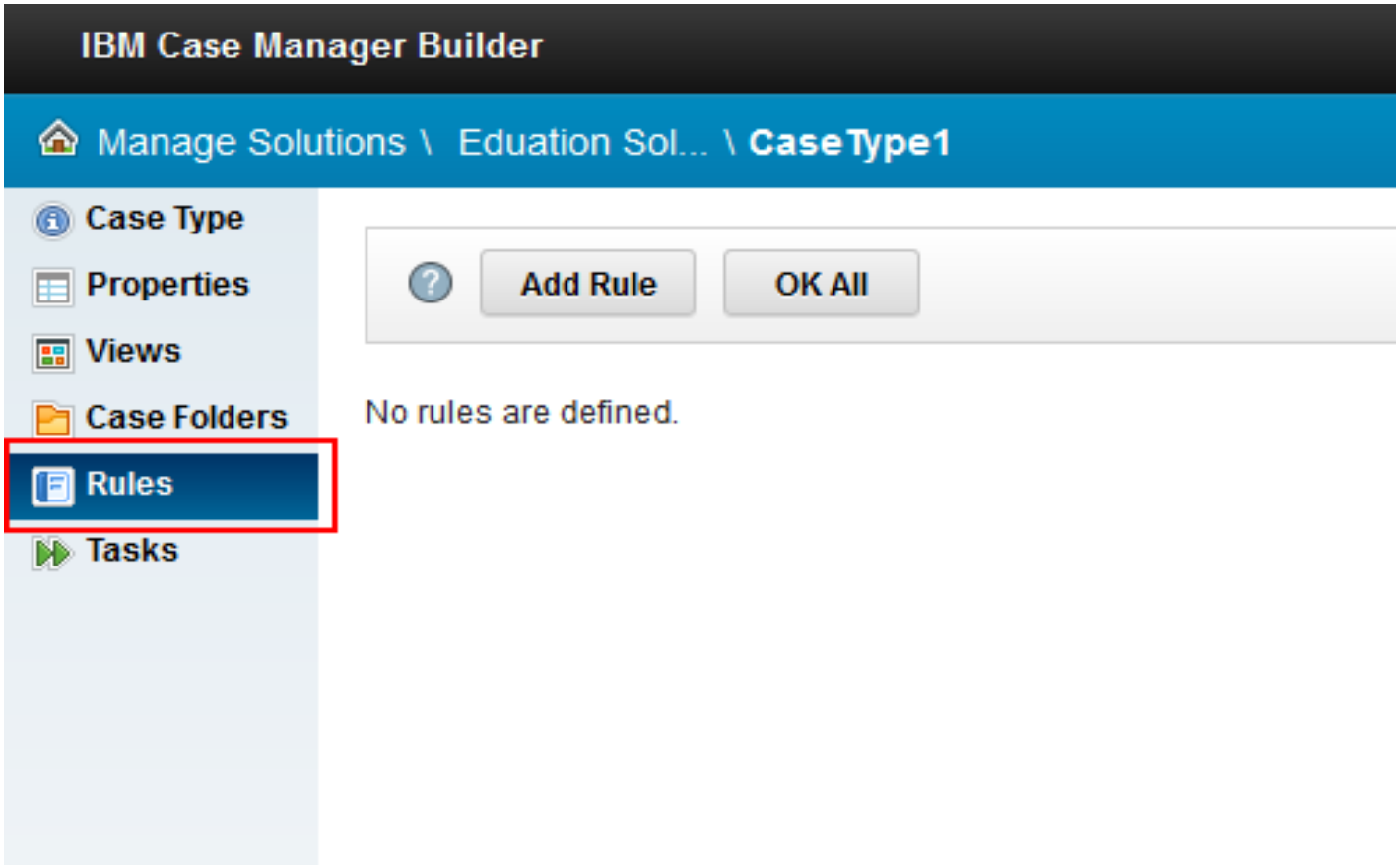
- CMAC configures ICM_RuleOperations queue and Rule Deployment code module in CPE and does additional configurations (create custom events and register event handlers)
- Rule repository path can be a local file system directory. If CPE is clustered, specify a shared directory which is accessible to all nodes.
- You can select only one locale in which to write the business rules.
  - If need to create solutions with business rules in other locales, rerun this task to change the rule persistence locale.
  - After a solution is created, its rule persistence locale cannot be changed.
  - If exporting rules to Decision Center component of IBM Operational Decision Manager, persistence locale should match the persistence locale of Decision Center.

Course Roadmap

- Introduction
- Configuring Embedded Rules in ICM environment
- ➔ Rules Authoring in Case Builder
- Executing Rules as part of a workflow
- Moving Rules from embedded environment to IODM
- Impact to ICM 5.1.1 customers
- Troubleshooting
- Known Issues, Limitations, Best Practices
- Course Summary

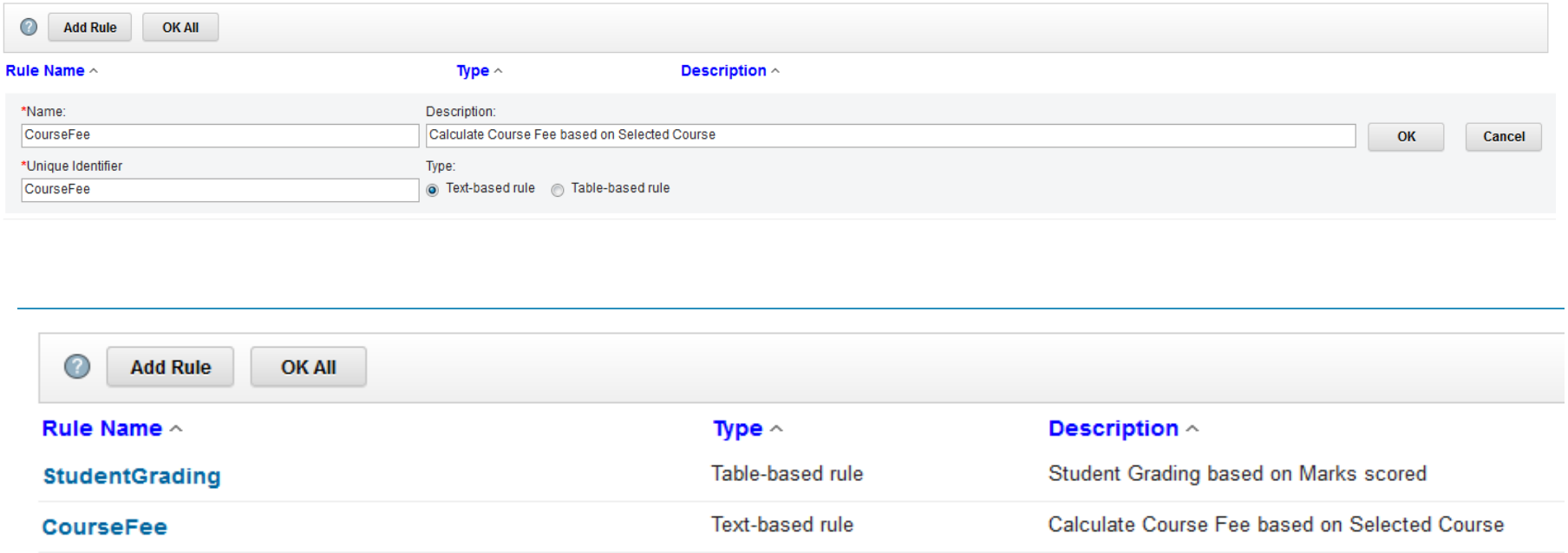
Authoring rules in Case Builder : CaseType

- Rules can be added from the “Rules” tab in a CaseType



Authoring rules in Case Builder : Add Rule

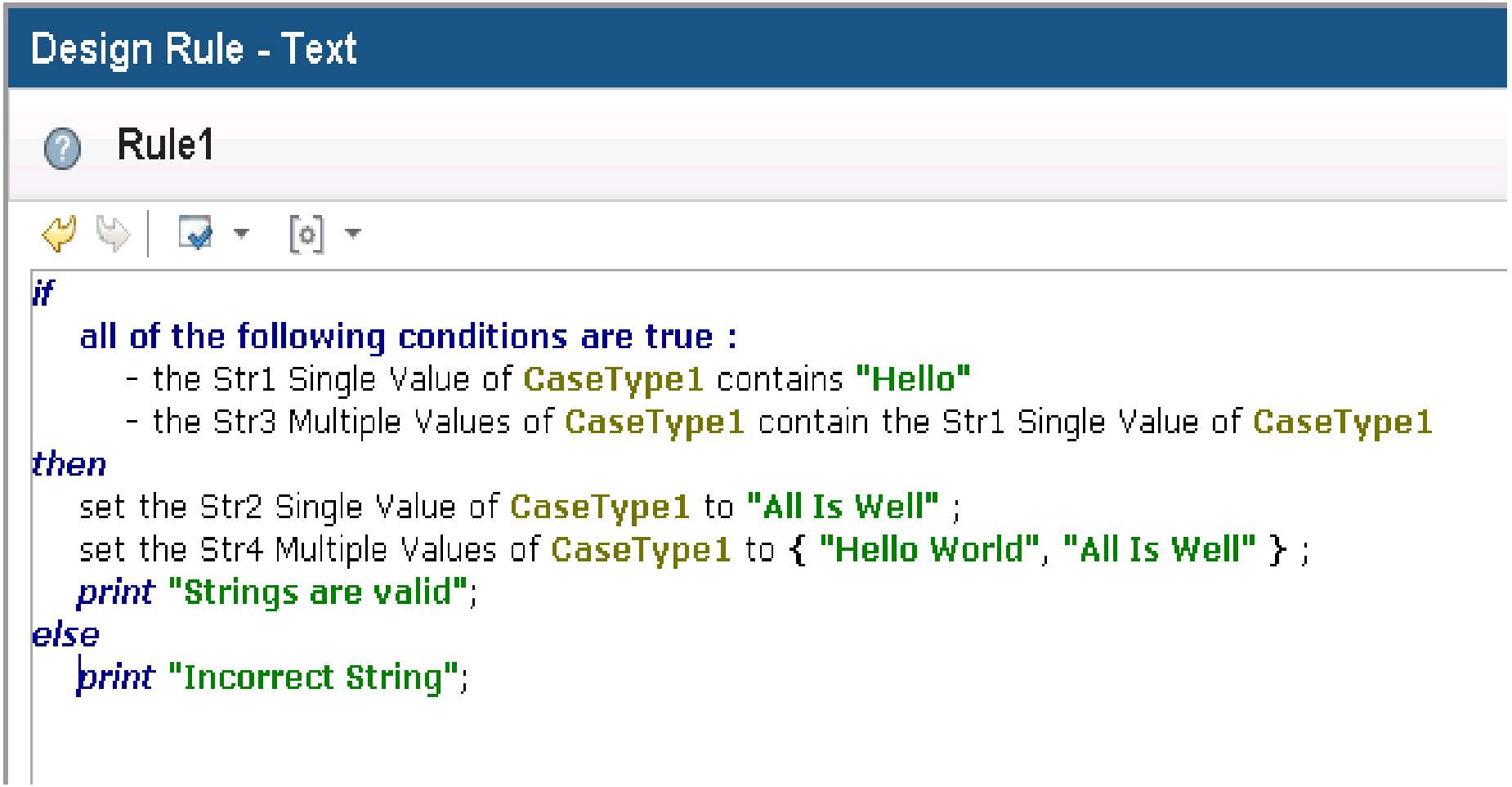
- Rule Name – 64 characters, also used to associate with a rule step
- Rule Unique Identifier – 64 characters (unique within CaseType)
- Rule Description – Meaningful description of the rule



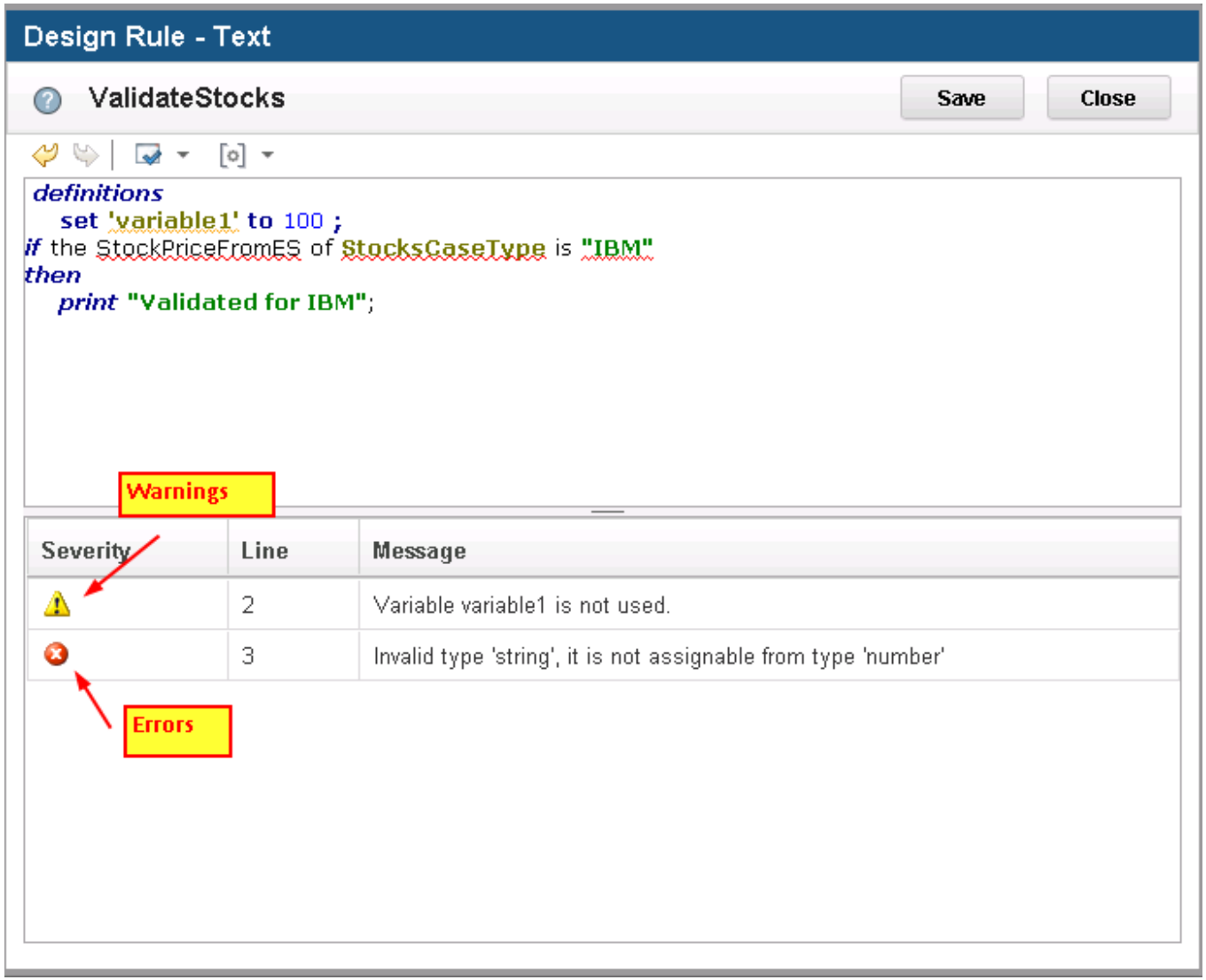
Authoring rules in Case Builder : Text-Based Rules

- Natural language based business rule authoring
- Based on Business Action Language (BAL)
- Intuitive editing / error checking
- Text-based Rules sections
  - Local variables (Definitions) - optional
  - Conditions (if) - optional
  - Actions (then /else)
- All Case Properties available for rule authoring
- Case System Properties available for rule authoring
- Custom Parameters to reference external data sources

Authoring rules in Case Builder : Text-based Rule - An Example

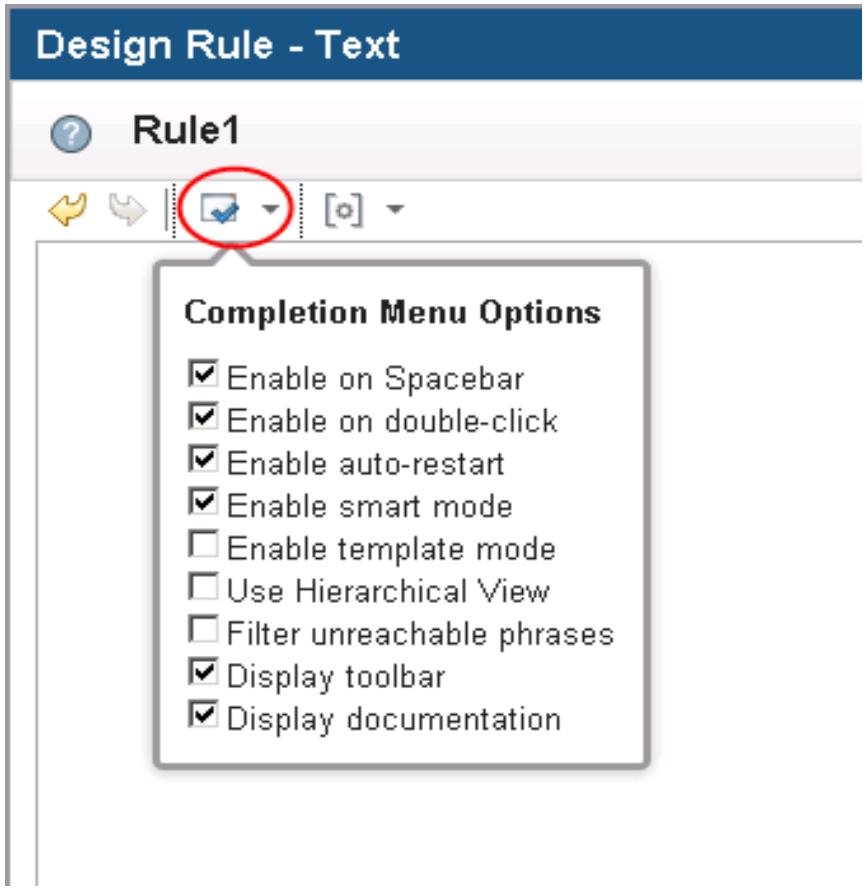


Authoring rules in Case Builder : Text-based Rule – Errors and Warnings

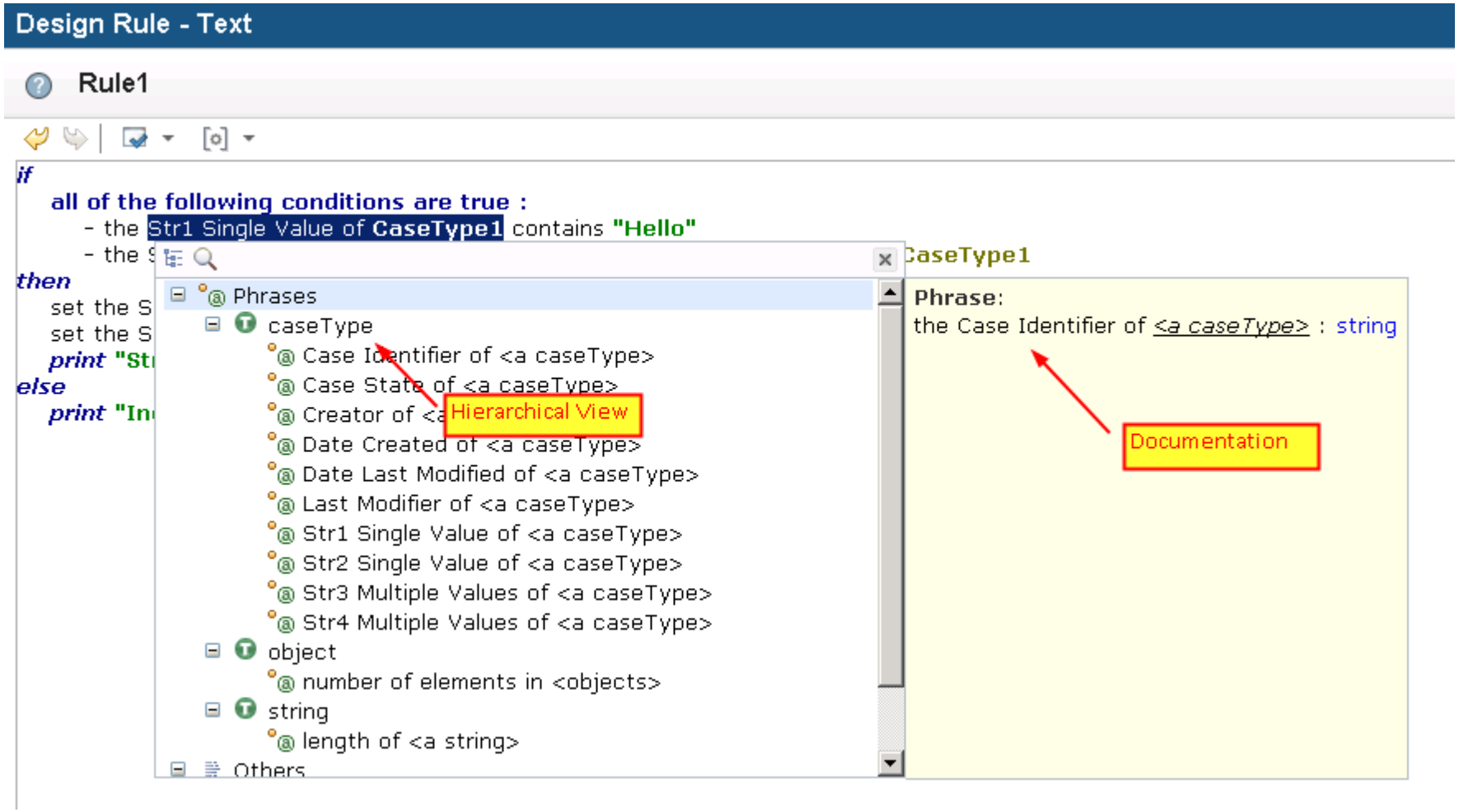


Authoring rules in Case Builder : Completion Menu options

- Assistance by providing all legal options while rule authoring

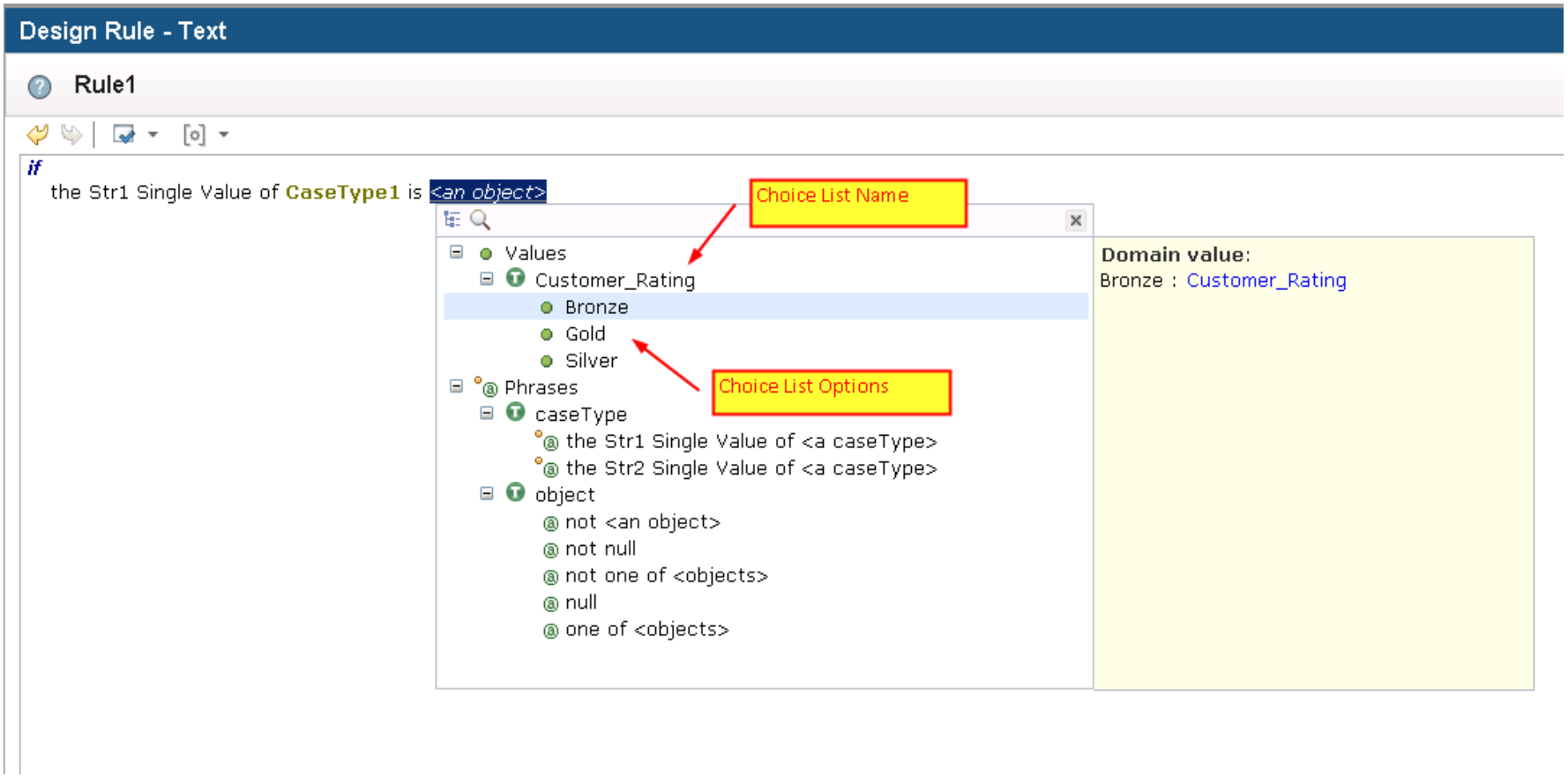


Authoring rules in Case Builder : Completion Menu options - Example



Authoring rules in Case Builder : Choice List in Completion Menu

- Choice list values appears as options in completion menu



Authoring rules in Case Builder : Text-based rules – Use Case

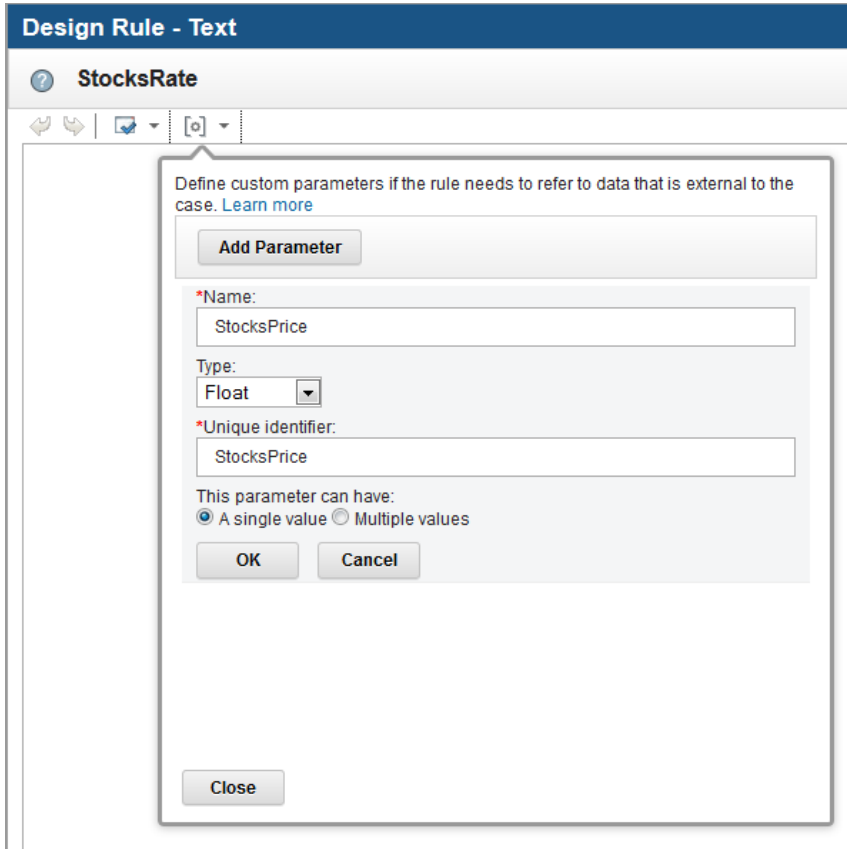
- Get the **StocksPrice** from an external source, declared as a custom parameter in the rule.
- Declare a threshold as a definition, **StockThreshold**, in the rule.
- If the **StocksPrice** is more than the **StockThreshold**, then set the **StockRating** as Excellent, else as Satisfactory.
- Set the boolean case property **Validated**, to true if the **StocksPrice** is more than 0 (verifying that the external source has worked)
- For debugging purposes, we will print **StocksPrice** from external source

Authoring rules in Case Builder : Text-based rules – Use Case Contd.

- Solution Artifacts:
- Case Properties:
  - StockRating (String)
  - StockSymbol (String)
  - StockPriceFromES (Float)
  - Validated (Boolean)
- Choice List: RatingOptions
  - Excellent
  - Good
  - Satisfactory
- Role: Agent
- CaseType: StocksCaseType

Authoring rules in Case Builder : Text-based rules – Custom Parameters

- Custom Parameters: Variables local to the rule, holding values from external sources, which can be referenced in the rule



Authoring rules in Case Builder : Text-based rules – Use Case – Rule Definition

Design Rule - Text

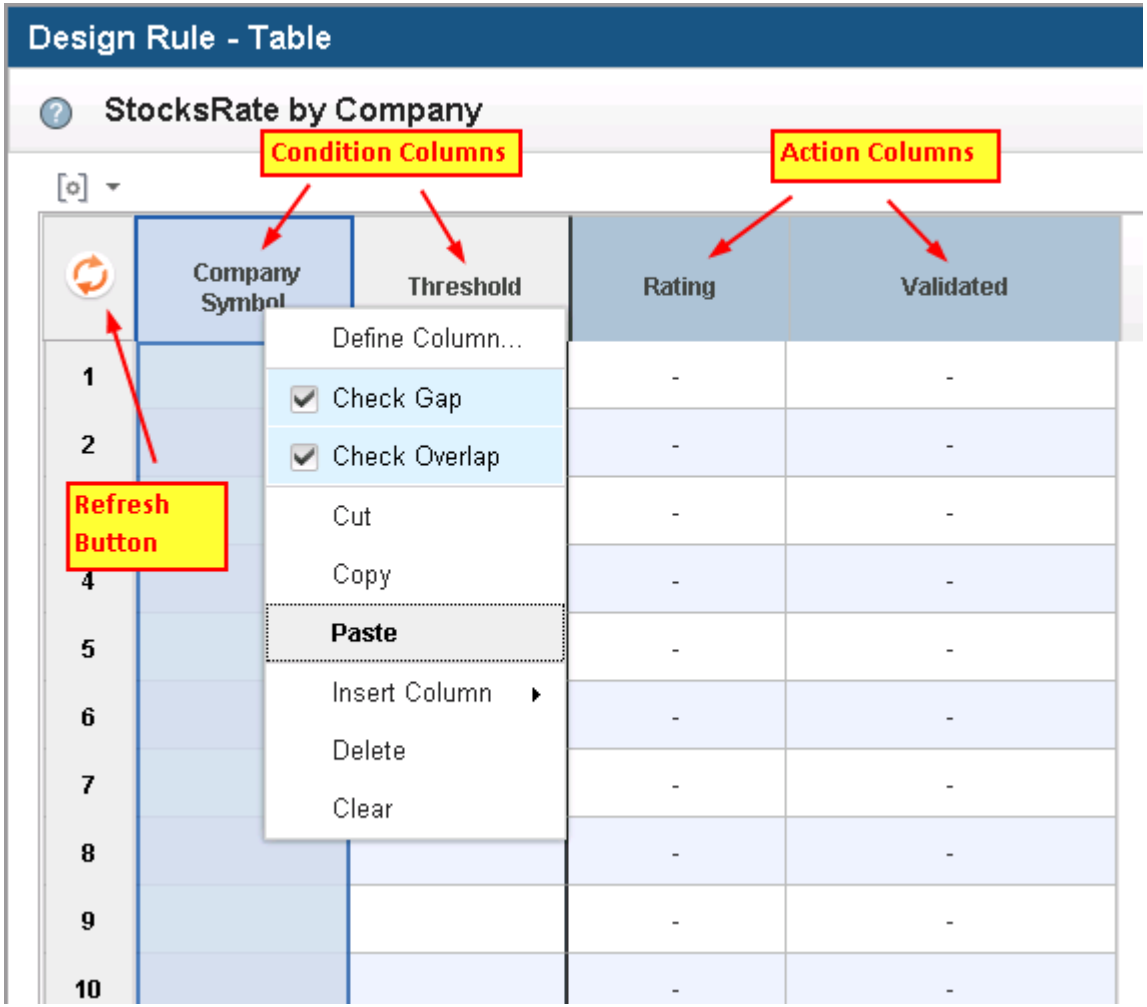
StocksRate

```
definitions
   set StockThreshold to 200 ;

if
    all of the following conditions are true :
      - StocksPrice is more than StockThreshold
      - StocksPrice is more than 0

then
   set the StockRating of StocksCaseType to Excellent ;
   set the StockPriceFromES of StocksCaseType to StocksPrice;
   make it true that StocksCaseType is Validated;
   print "Stock Price: " + the StockPriceFromES of StocksCaseType;

else
   set the StockRating of StocksCaseType to Satisfactory;
   set the StockPriceFromES of StocksCaseType to StocksPrice;
   make it true that StocksCaseType is Validated;
   print "Stock Price: " + the StockPriceFromES of StocksCaseType;
```

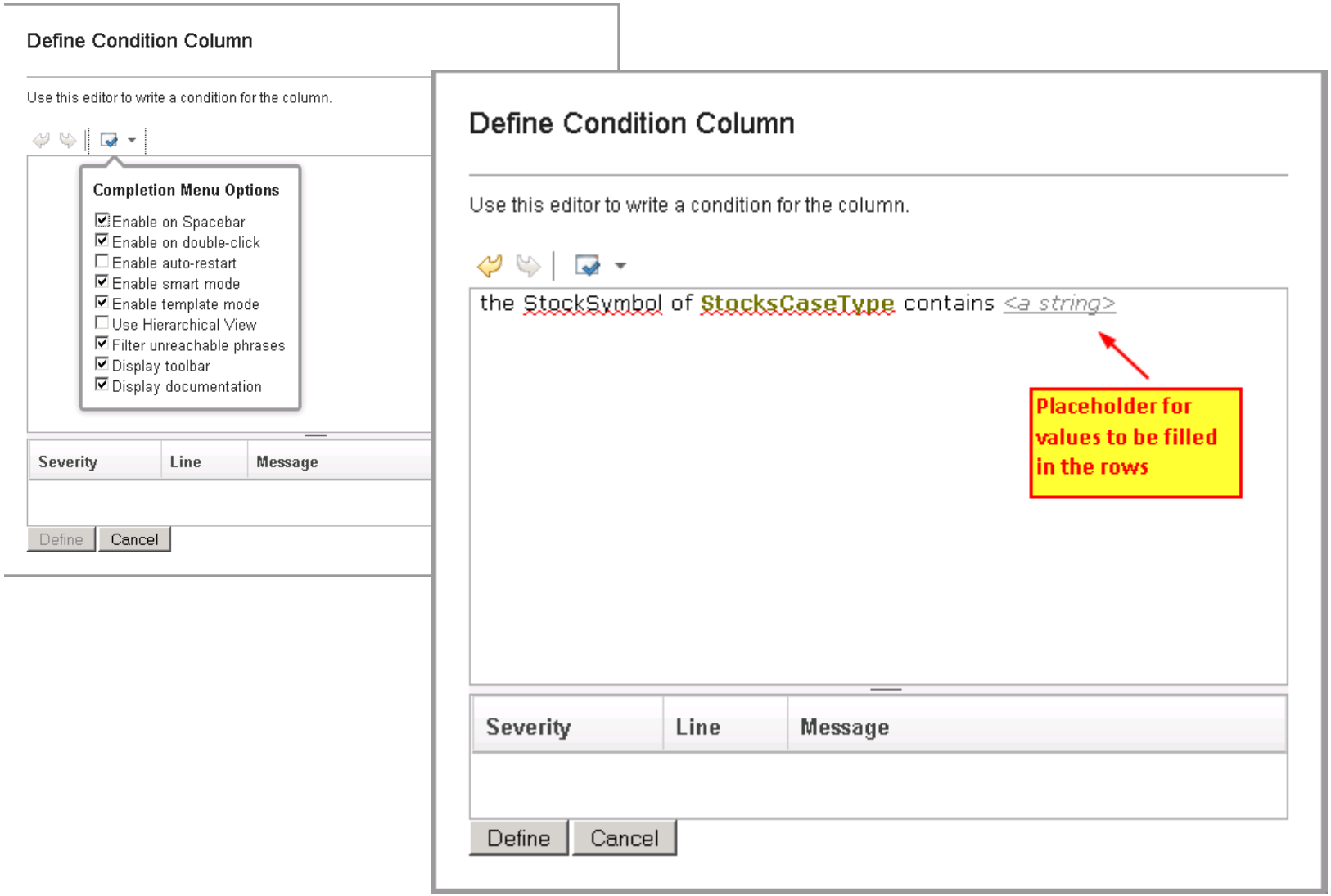
Authoring rules in Case Builder : Table - based Rules

- Table based business rule authoring
- Used to express sets of similar conditions and actions in rules
- Intuitive editing / error checking
- Each row in a decision table rule can be considered as a text-based rule
- Multiple condition columns and multiple action columns
- Gap checks and overlap checks done by default
- Supports custom parameters

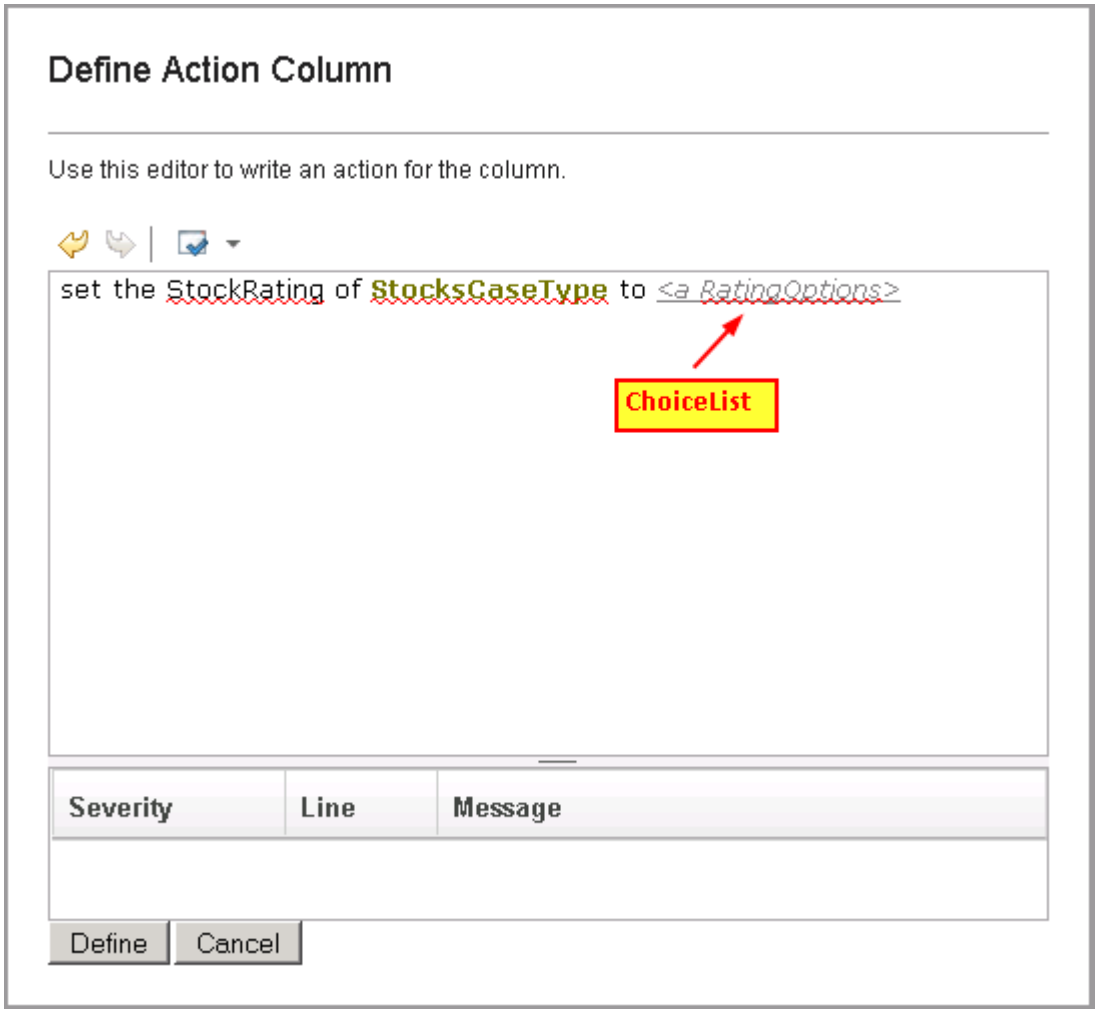
Authoring rules in Case Builder – Table-based rules Editor



Authoring rules in Case Builder : Defining Condition Column

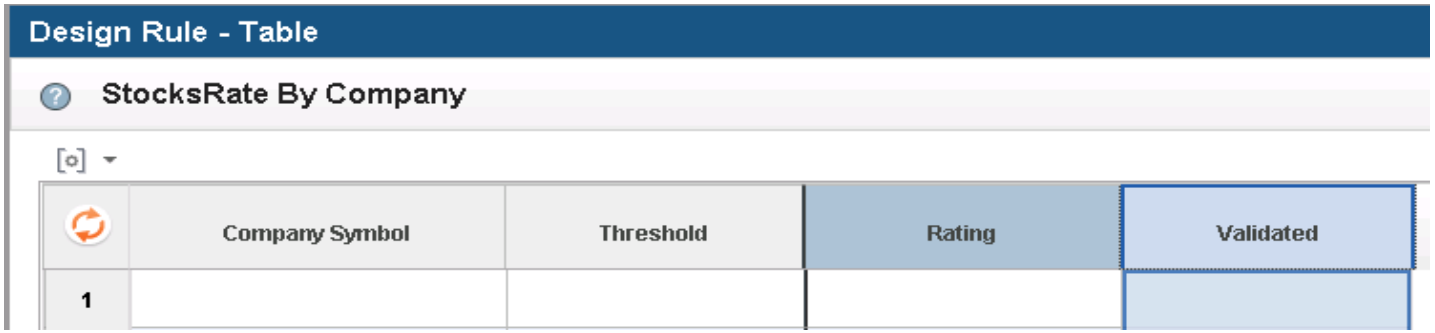


Authoring rules in Case Builder – Defining Action Column

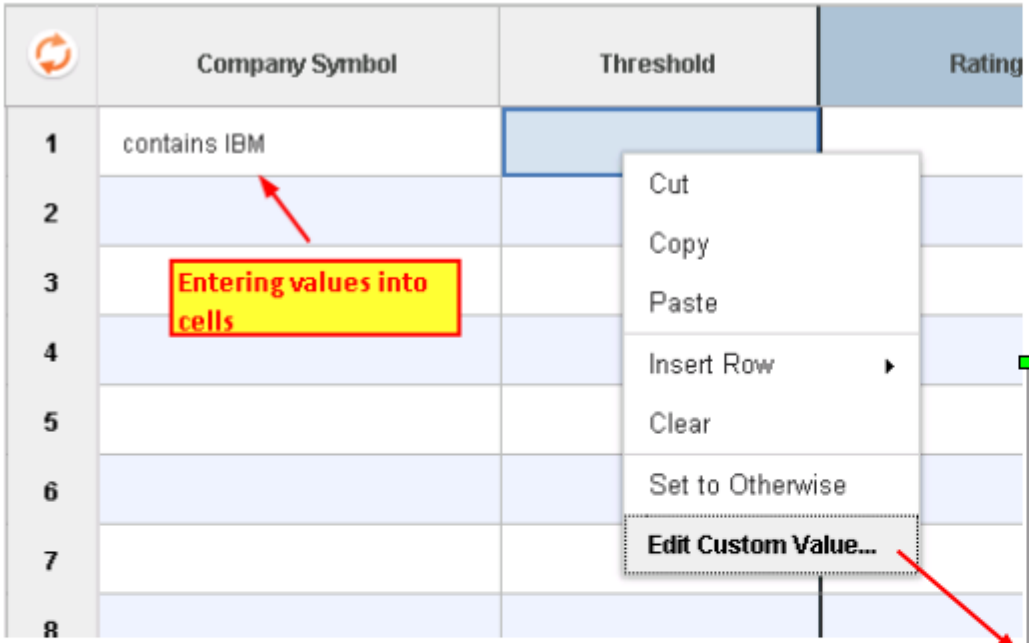


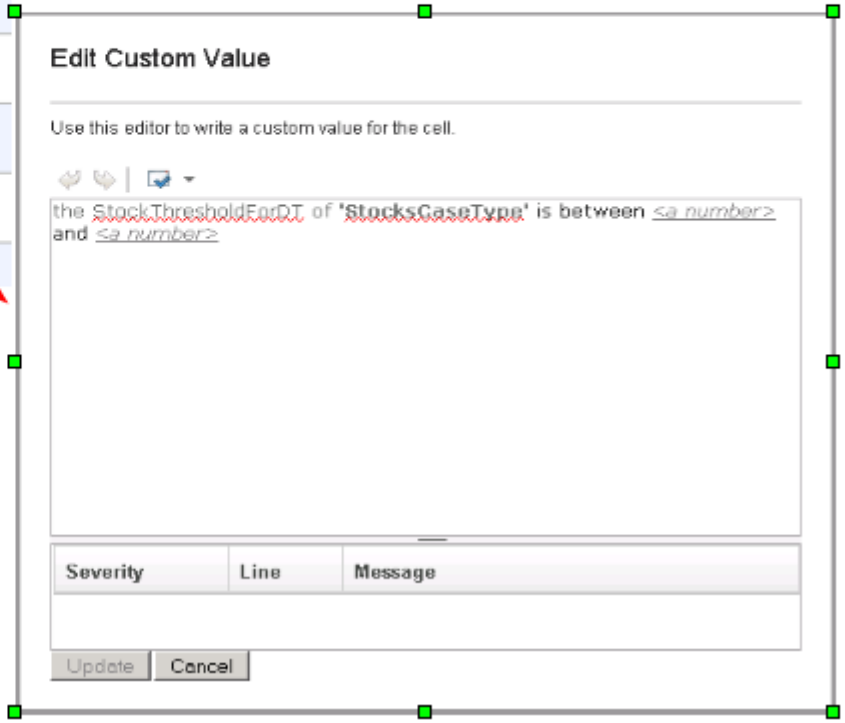
Authoring rules in Case Builder : Table-based rules – Example Contd.

- Condition Columns:
  - **Company Symbol:** the StockSymbol of StocksCaseType contains <a string>
  - **Threshold:** StocksPrice is between <min> and <max>
- Action Columns:
  - **Rating:** set the StockRating of StocksCaseType to <a RatingOptions>
  - **Validated:** make it <a boolean> that StocksCaseType is Validated



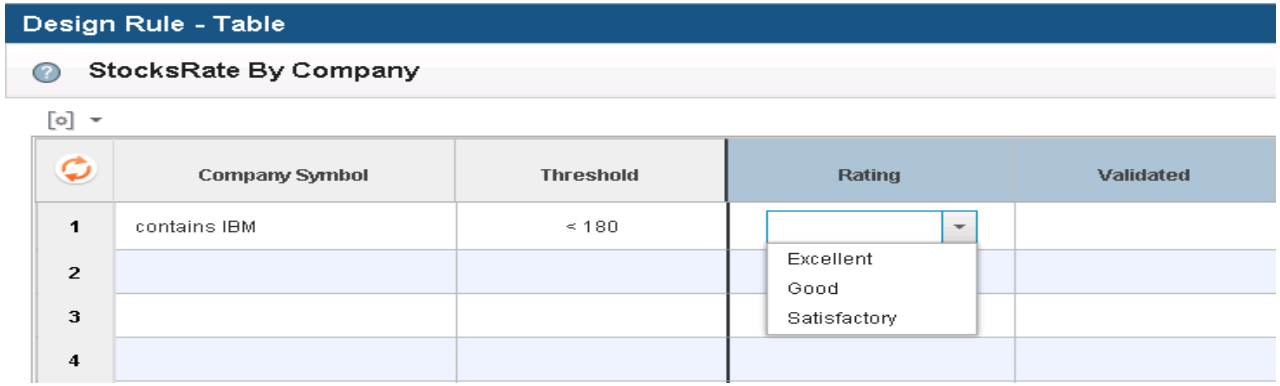
Authoring rules in Case Builder : Table-based rules – Example – Populating Rows



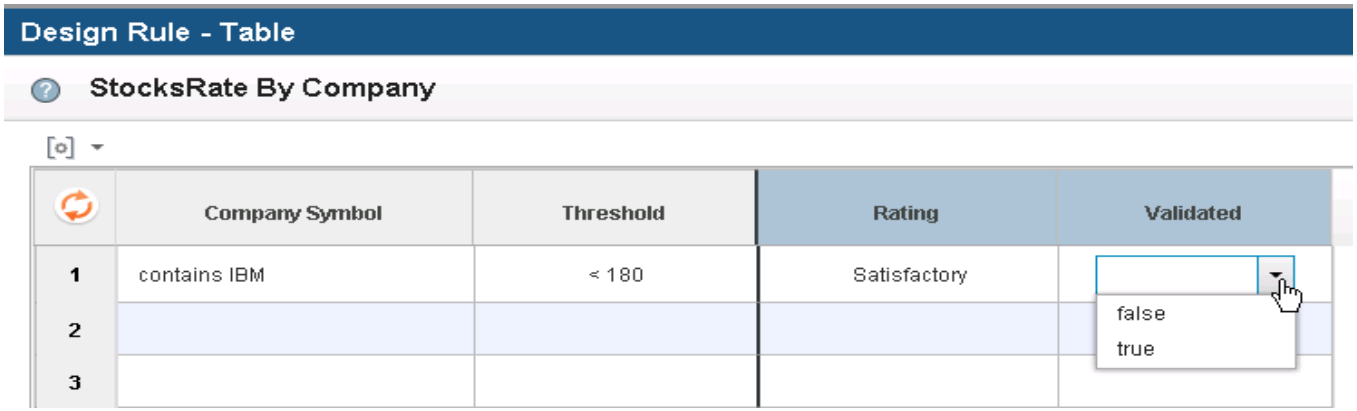


Authoring rules in Case Builder :Table-based rules – Example - Populating Rows

- Drop down menu for choice list values

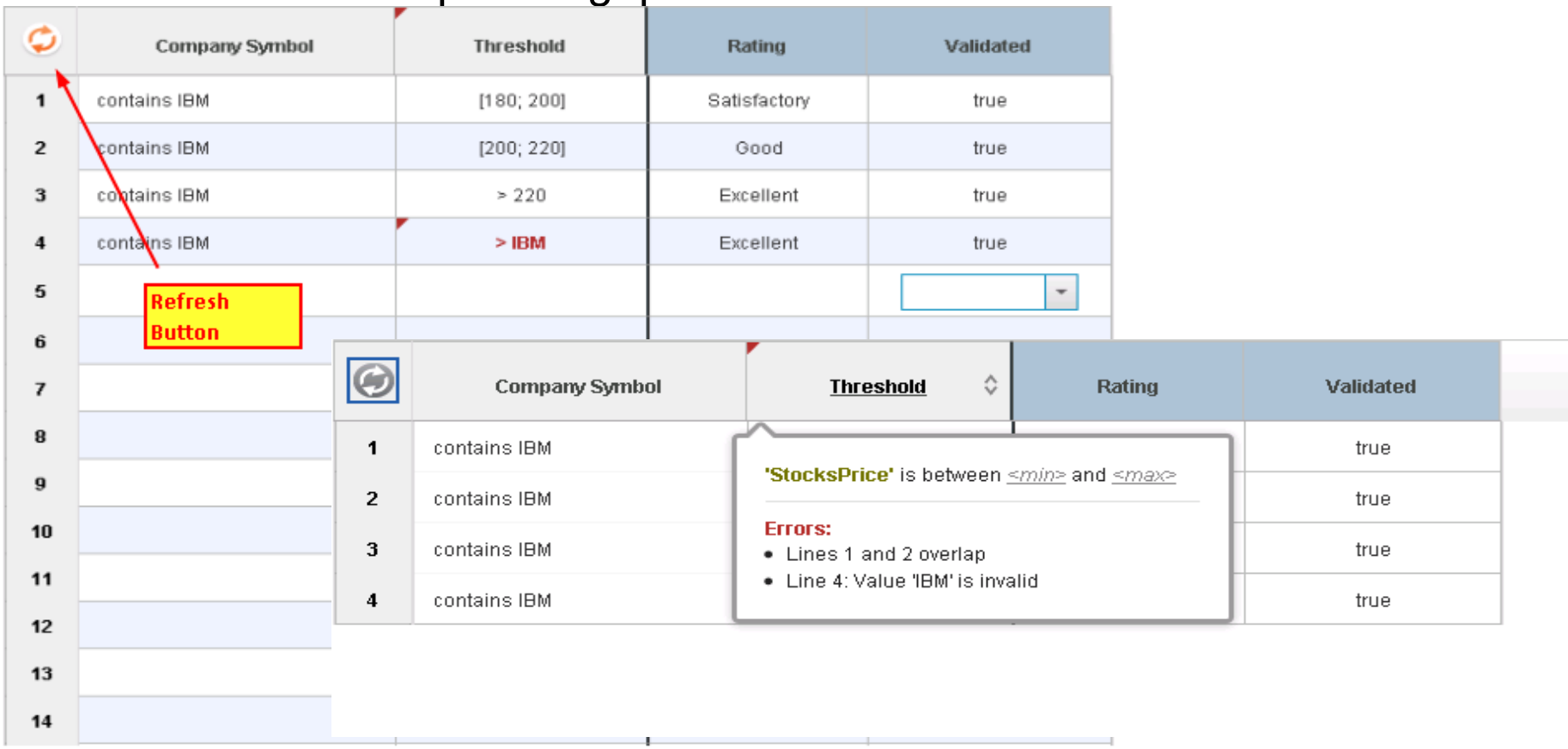


- Drop down menu for Boolean values



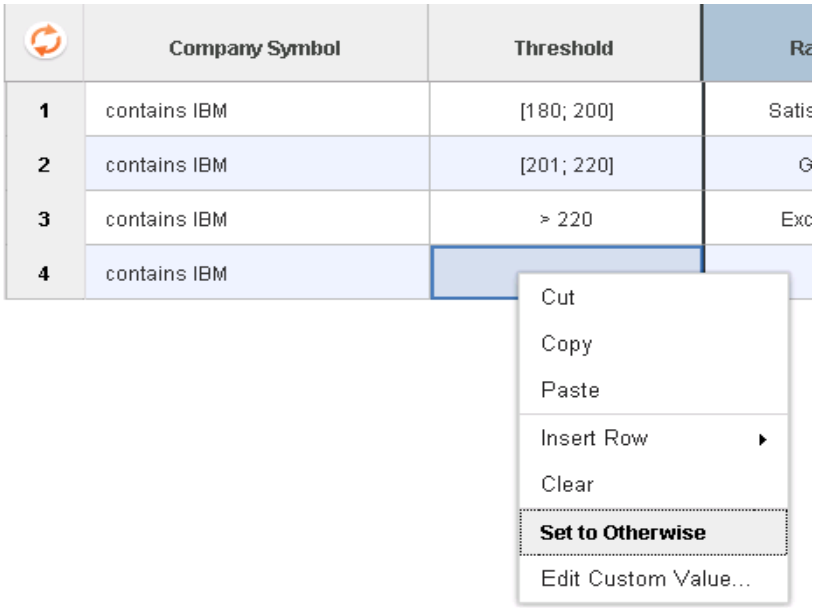
Authoring rules in Case Builder – Table-based rules – Refresh Button

- Refresh button – removes unused rows, sorts rows in ascending order, validates for overlaps and gaps

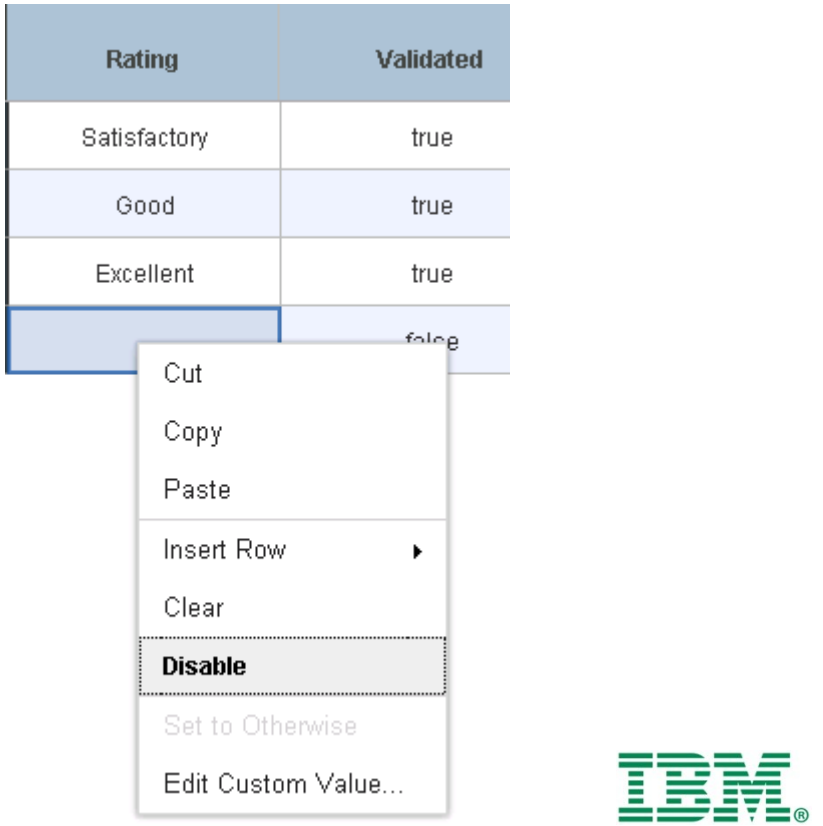


Authoring rules in Case Builder : Table-based rules – Otherwise and Disable

- Otherwise – Condition Column option – “catch all” for values not satisfying other rows, thereby resolving gaps in values



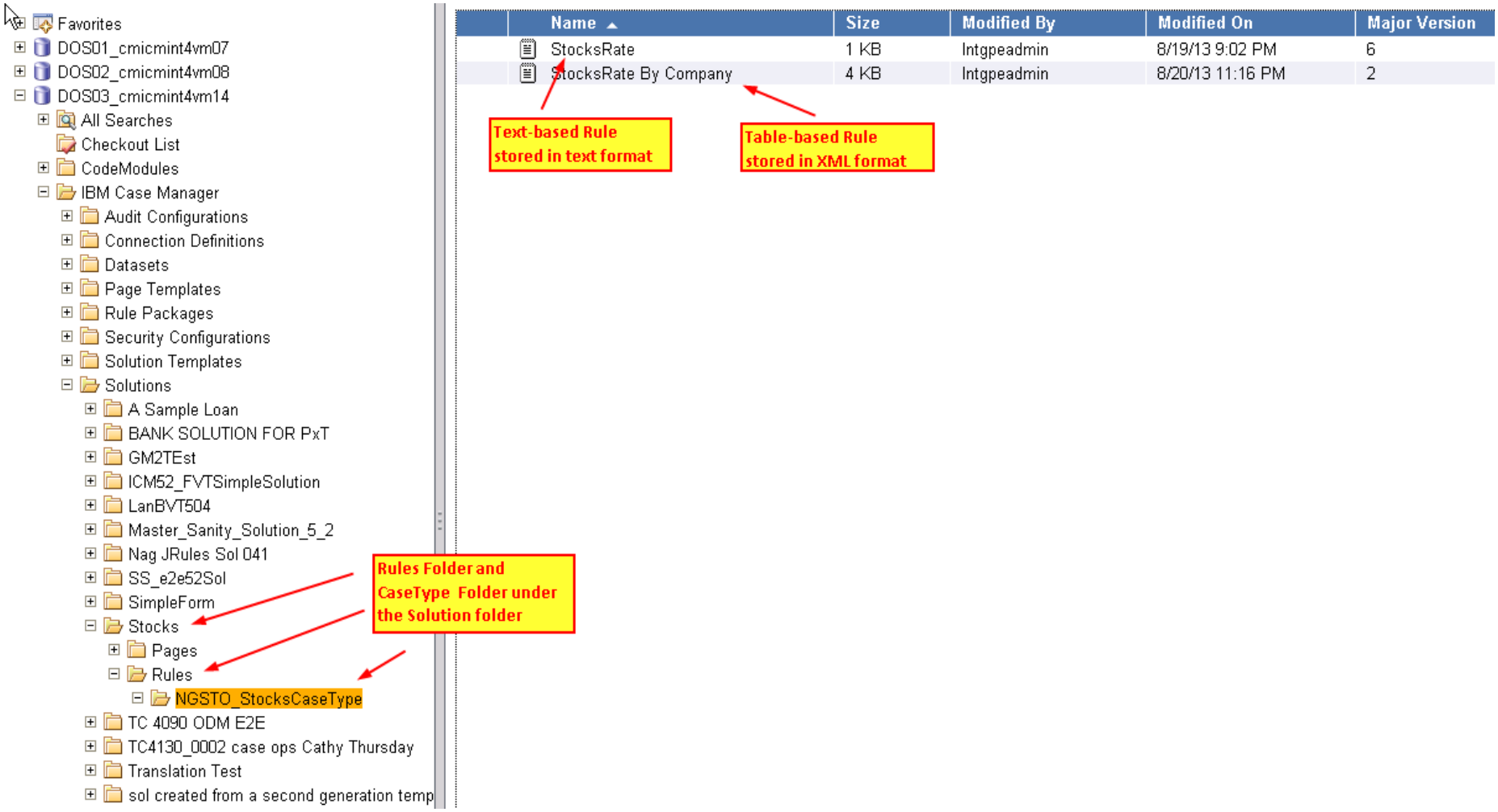
- Disable – Action Column option – for disabling a particular action



Authoring rules in Case Builder : Table-based rules – Example – Completed Rule

	Company Symbol	Threshold	Rating	Validated
1	contains AMNZ	[200; 250]	Satisfactory	true
2	contains AMNZ	[251; 300]	Good	true
3	contains AMNZ	> 300	Excellent	true
4	contains AMNZ	Otherwise	⊘	false
5	contains IBM	[180; 200]	Satisfactory	true
6	contains IBM	[201; 220]	Good	true
7	contains IBM	> 220	Excellent	true
8	contains IBM	Otherwise	⊘	false
9	Otherwise	Otherwise	⊘	false

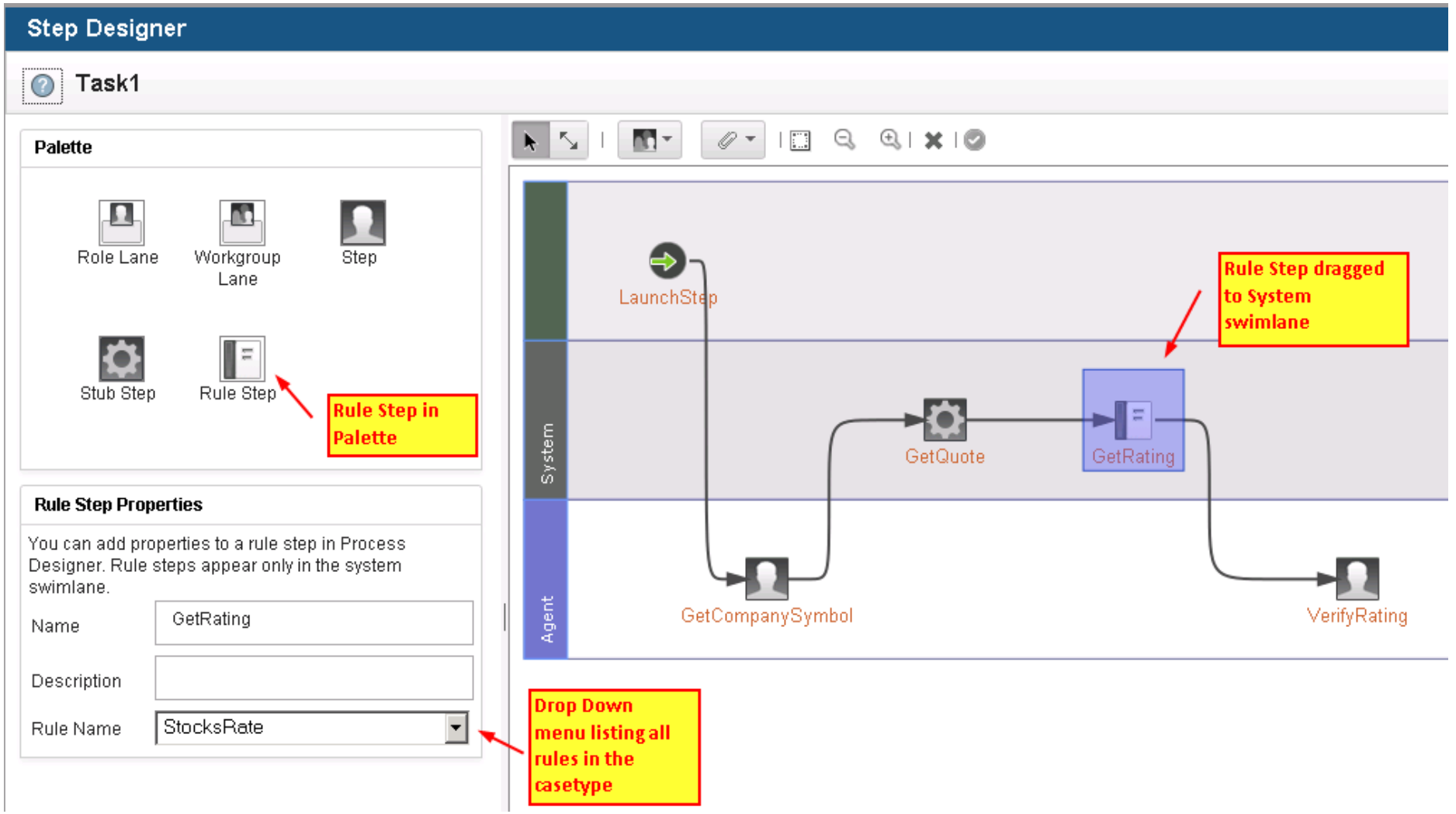
Rules – DOS structure



Course Roadmap

- Introduction
- Configuring Embedded Rules in ICM environment
- Rules Authoring in Case Builder
- ➔ Executing Rules as part of a workflow
- Moving Rules from embedded environment to IODM
- Impact to ICM 5.1.1 customers
- Troubleshooting
- Known Issues, Limitations, Best Practices
- Course Summary

Executing Rules as part of a workflow : Step Editor



Executing Rules as part of a workflow :Mapping Custom Parameters to external data sources

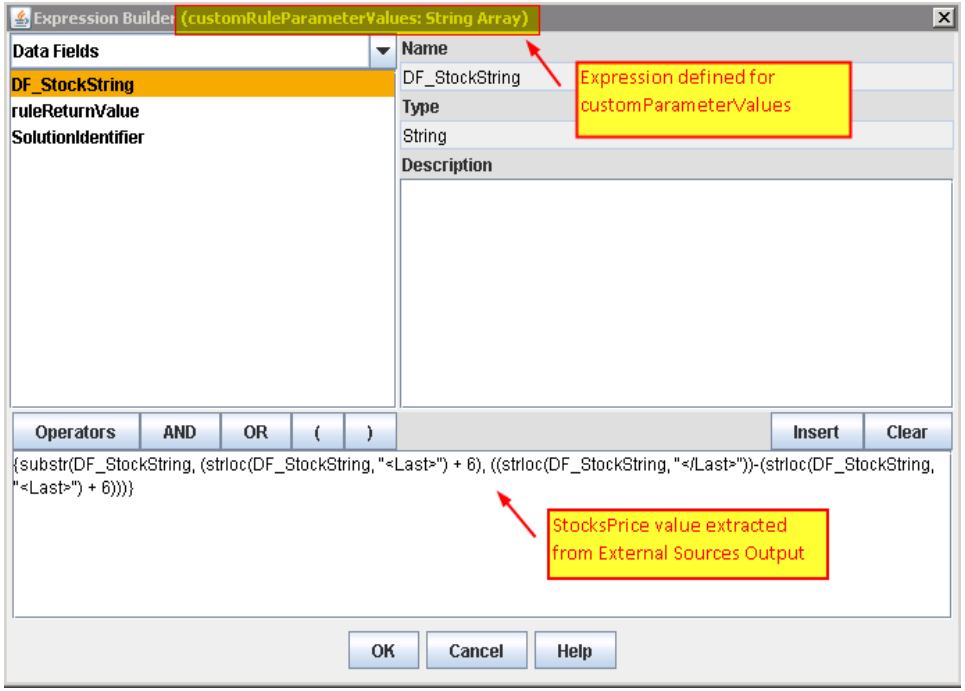
- If Custom Parameters are used in the rules, the following have to be edited in Process Designer:
  - **ruleName** – The name of the rule to be executed
  - **customRuleParameterNames** – String Array with all custom parameters defined (added when rule step is added to the workflow)
  - **customRuleParameterValues** – String Array defining values to custom parameters in the order specified in customRuleParameterNames
  - **return_value** – ruleReturnValue, holds the value of the “print” statements in the rule

Executing Rules as part of a workflow :Mapping Custom Parameters to external data sources

Operation Parameters

Name	Type		Expression
caseId	String		F_CaseFolder
ruleName	String		"StocksRate"
customRuleParameterNames	String[]		{"StocksPrice"}
customRuleParameterValues	String[]		{substr(DF_StockString, (strloc(DfF_StockStri...
return_value	String		ruleReturnValue

DF_StockString defined as DataField for the workflow

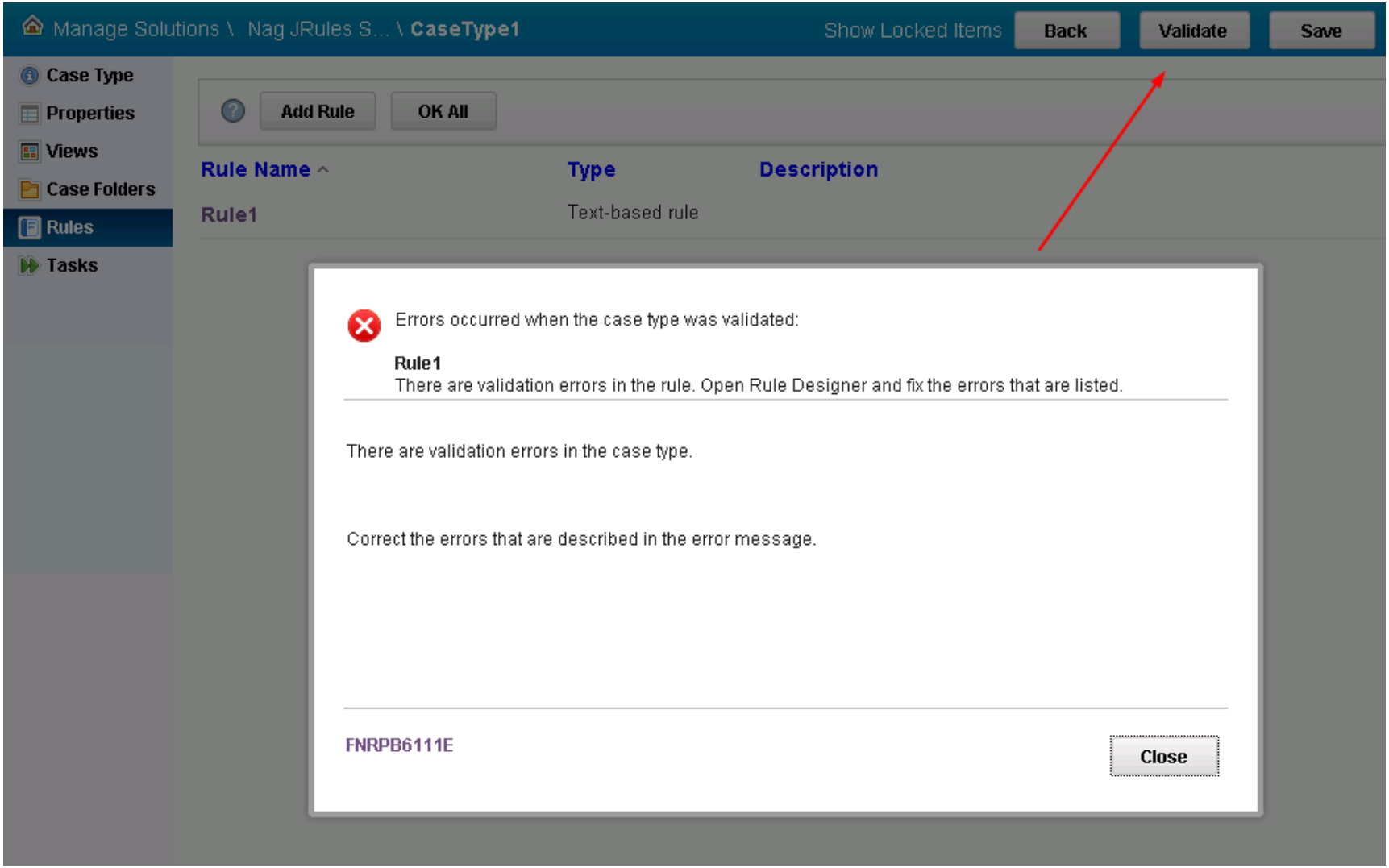


Defining custom rule parameter feeds in Process Designer

- CustomRuleParameterValues is a string array and all values have to be coverted to strings

DataType	**DataField (example)**	**Converting to string**
Boolean	DF_Bool1	convert(DF_Bool1, string)
DateTime	DF_DateTime1	timetostring (DF_DateTime1,"yyyy-mm-dd hh:tt:ss")
Float	DF_Float1	convert(DF_Float1, string)
Integer	DF_Int1	convert(DF_Int1,string)
String	DF_Str1	No need to convert as string is expected
Multi-value properties	DF_StrArray	arraytostring(DF_StrArray,"{","}",",")

CaseType Validation - Rules



Course Roadmap

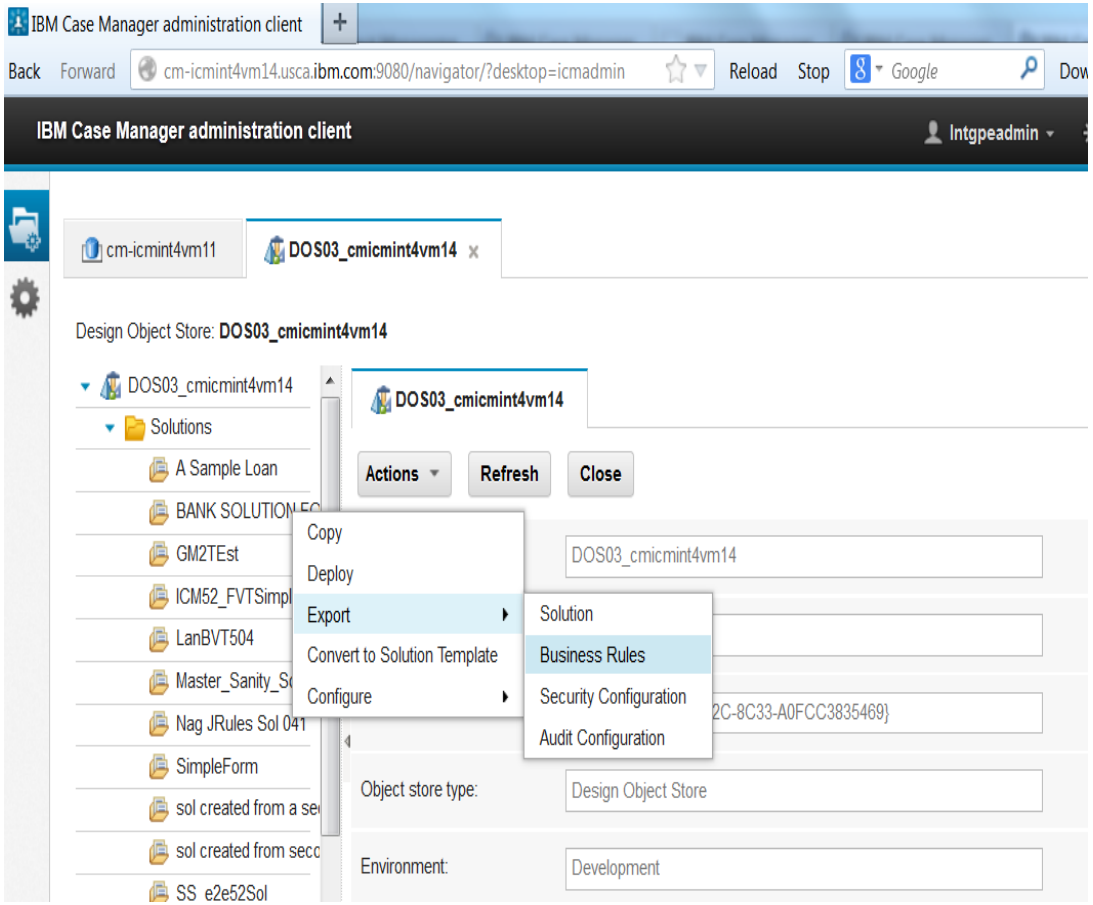
- Introduction
- Configuring Embedded Rules in ICM environment
- Rules Authoring in Case Builder
- Executing Rules as part of a workflow
- ➔ Moving Rules from embedded environment to IODM
- Impact to ICM 5.1.1 customers
- Troubleshooting
- Known Issues, Limitations, Best Practices
- Course Summary

Motivations for moving embedded rule to IODM

- Centralized governance requirements
- Change in organizational policies around Change management, version control requirements for rule artifacts
- New requirements arise that need more advanced authoring features like Decision Trees, Rule Flows, Templates
- Can be a stage by stage process, some rules remain in embedded and some moved to full IODM

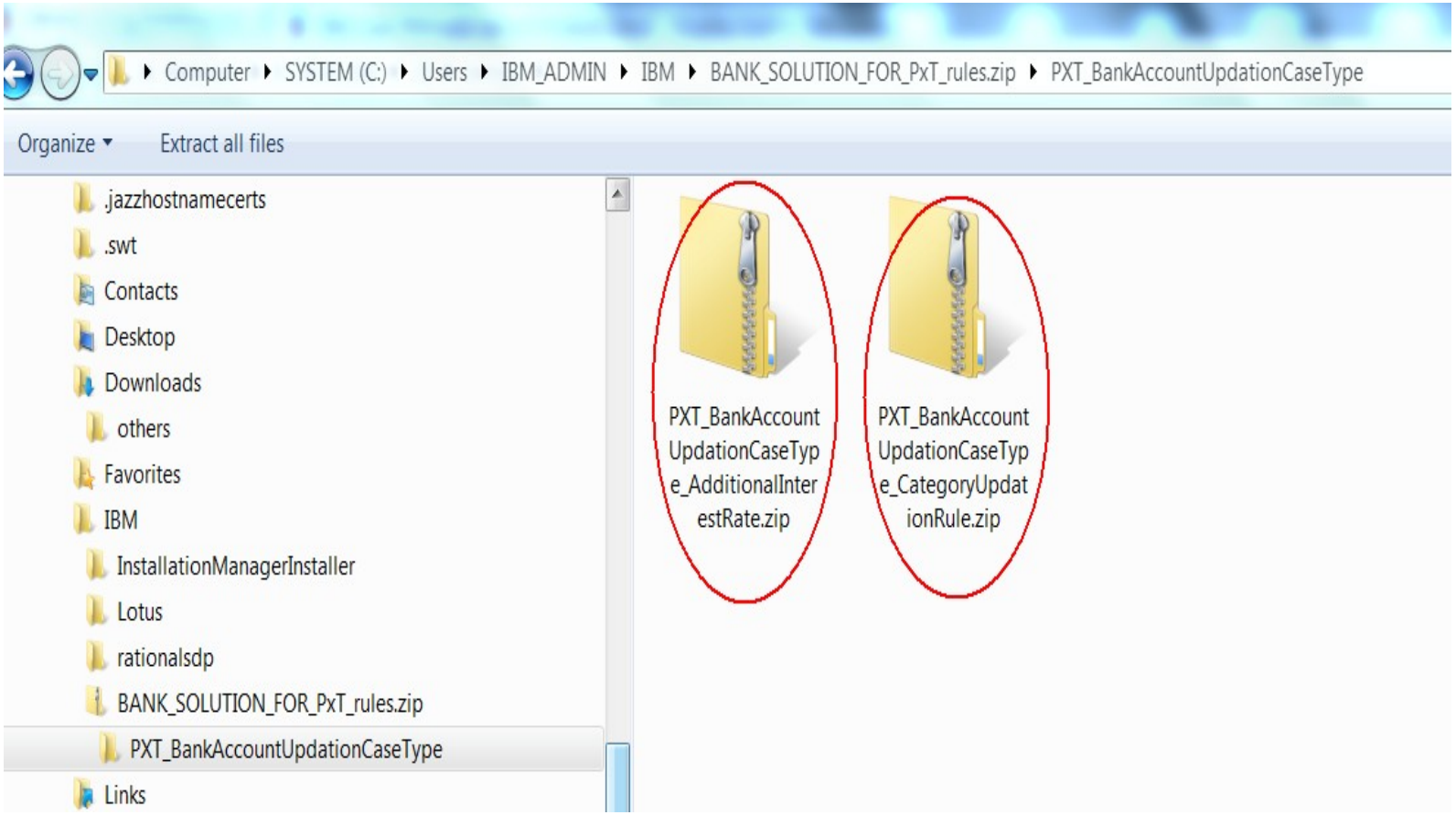
Exporting embedded rules as rule projects using ICM Administration Client

- All rules in a solution are exported as a zip file from ICM Administration Client
- Pre-requisite : Solution should have been successfully deployed before export is invoked



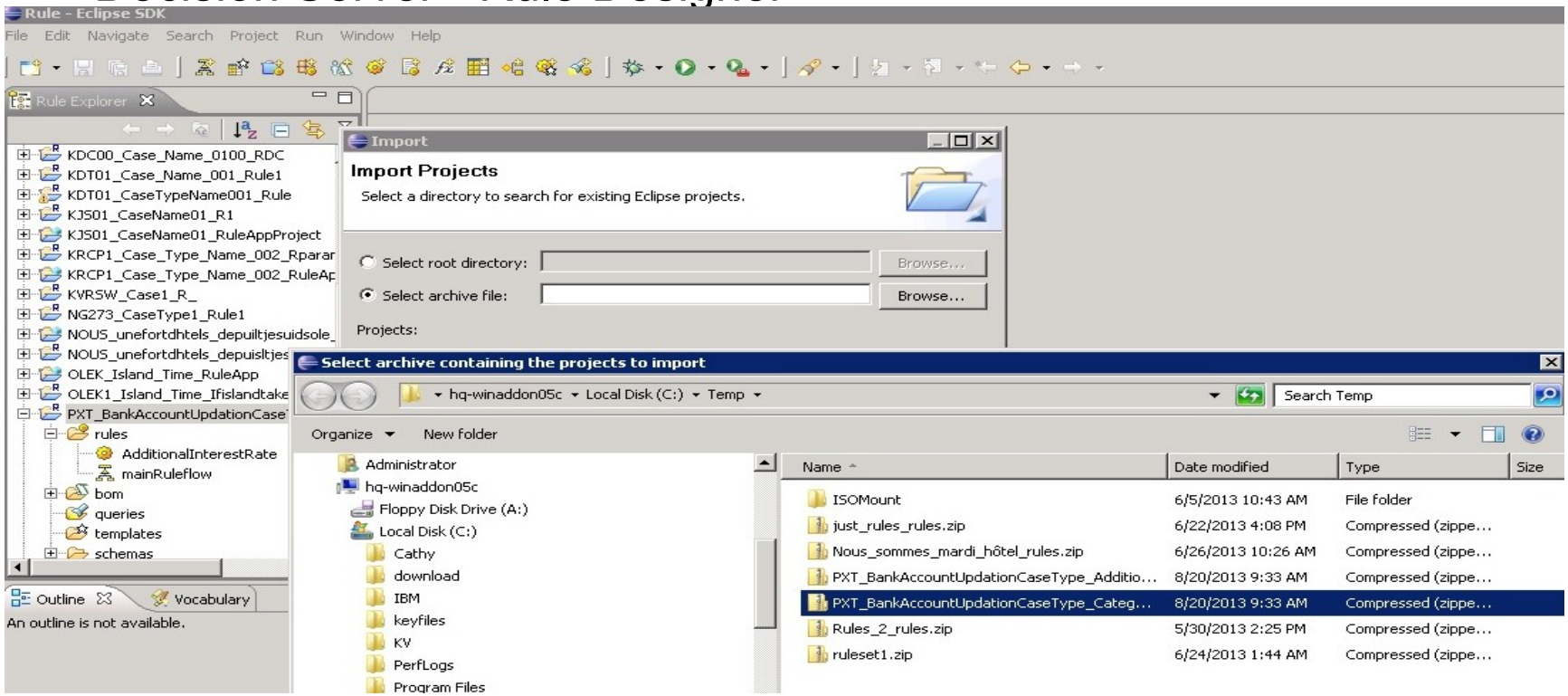
Structure of rule export output

- Output zip contains multiple rule project zip files, one per rule



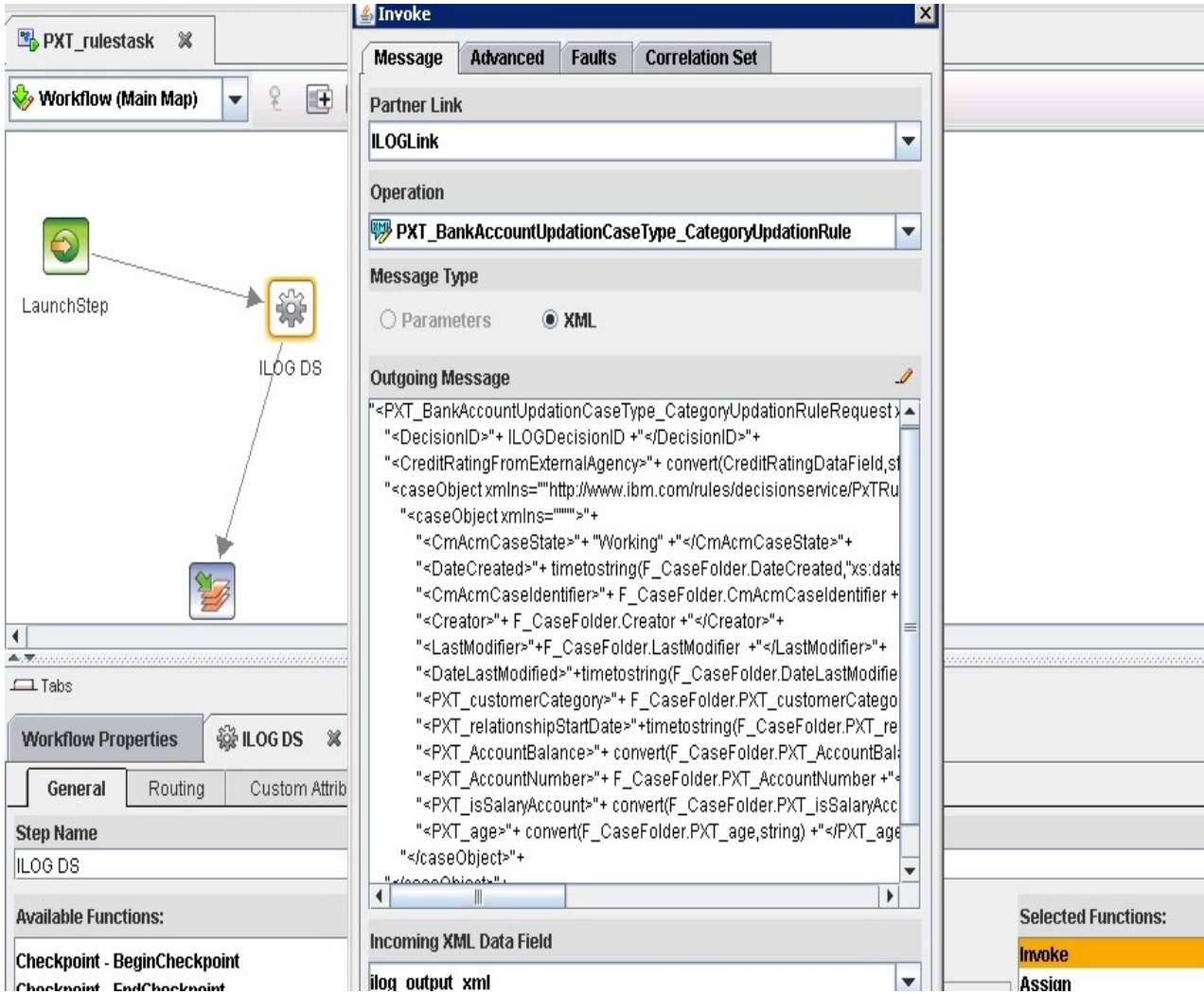
Importing to IODM

- Individual rule project zip files can be imported to IODM products
  - Decision Center
  - Decision Server - Rule Designer



Changes required in solution for migrated rules

- Remove rule steps, bring in web services invocation methods
- You can choose to move only few of the rules to web services invocation
- Use outgoing type as “XML message” to pass info to IODM web service



Sample outgoing XML message

"<PXT_BankAccountUpdationCaseType_CategoryUpdationRuleRequest
xmlns=""http://www.ibm.com/rules/decisionservice/PxTRuleApp/PXT_BankAccountUpdationCaseType_CategoryUpdationRule"">"+

"<DecisionID>"+ **ILOGDecisionID** +"</DecisionID>"+

"<CreditRatingFromExternalAgency>"+ **convert(CreditRatingDataField,string)** +"</CreditRatingFromExternalAgency>"+

"<caseObject xmlns=""http://www.ibm.com/rules/decisionservice/PxTRuleApp/PXT_BankAccountUpdationCaseType_CategoryUpdationRule/param"">"+

"<caseObject xmlns="""">"+

"<CmAcmCaseState>"+ **"Working"** +"</CmAcmCaseState>"+

"<DateCreated>"+ **timetostring(F_CaseFolder.DateCreated,"xs:datetime")** +"</DateCreated>"+

"<CmAcmCaseIdentifier>"+ **F_CaseFolder.CmAcmCaseIdentifier** +"</CmAcmCaseIdentifier>"+

"<Creator>"+ **F_CaseFolder.Creator** +"</Creator>"+

"<LastModifier>"+**F_CaseFolder.LastModifier** +"</LastModifier>"+

"<DateLastModified>"+**timetostring(F_CaseFolder.DateLastModified,"xs:datetime")** +"</DateLastModified>"+

"<PXT_customerCategory>"+ **F_CaseFolder.PXT_customerCategory** +"</PXT_customerCategory>"+

"<PXT_relationshipStartDate>"+**timetostring(F_CaseFolder.PXT_relationshipStartDate,"xs:datetime")** +"</PXT_relationshipStartDate>"+

"<PXT_AccountBalance>"+ **convert(F_CaseFolder.PXT_AccountBalance,string)** +"</PXT_AccountBalance>"+

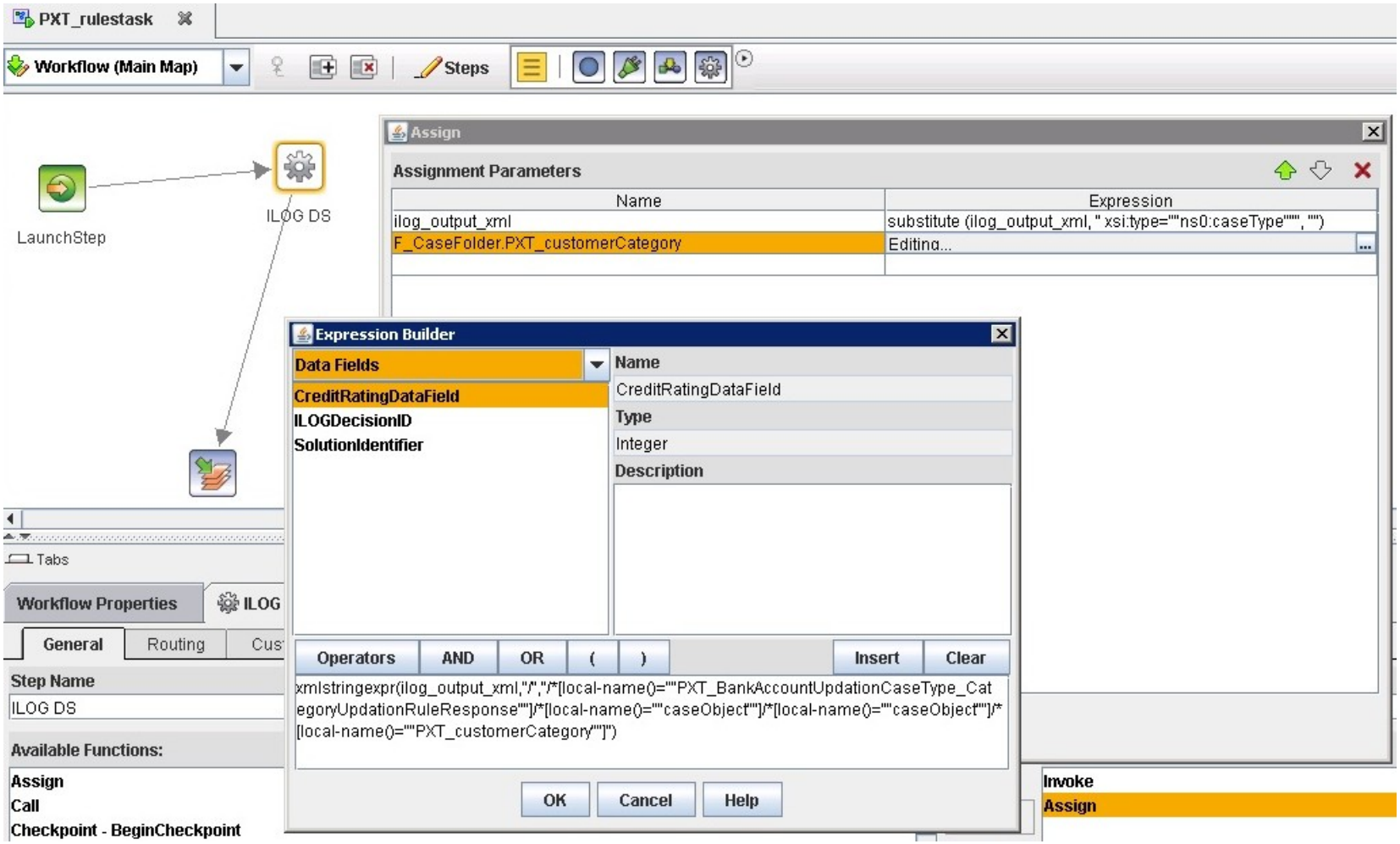
"<PXT_AccountNumber>"+ **F_CaseFolder.PXT_AccountNumber** +"</PXT_AccountNumber>"+

"<PXT_isSalaryAccount>"+ **convert(F_CaseFolder.PXT_isSalaryAccount,string)** +"</PXT_isSalaryAccount>"+

"<PXT_age>"+ **convert(F_CaseFolder.PXT_age,string)** +"</PXT_age>"+

"</caseObject>"+ "</caseObject>"+ "</PXT_BankAccountUpdationCaseType_CategoryUpdationRuleRequest>"

Using XPath expressions in Assign function



Course Roadmap

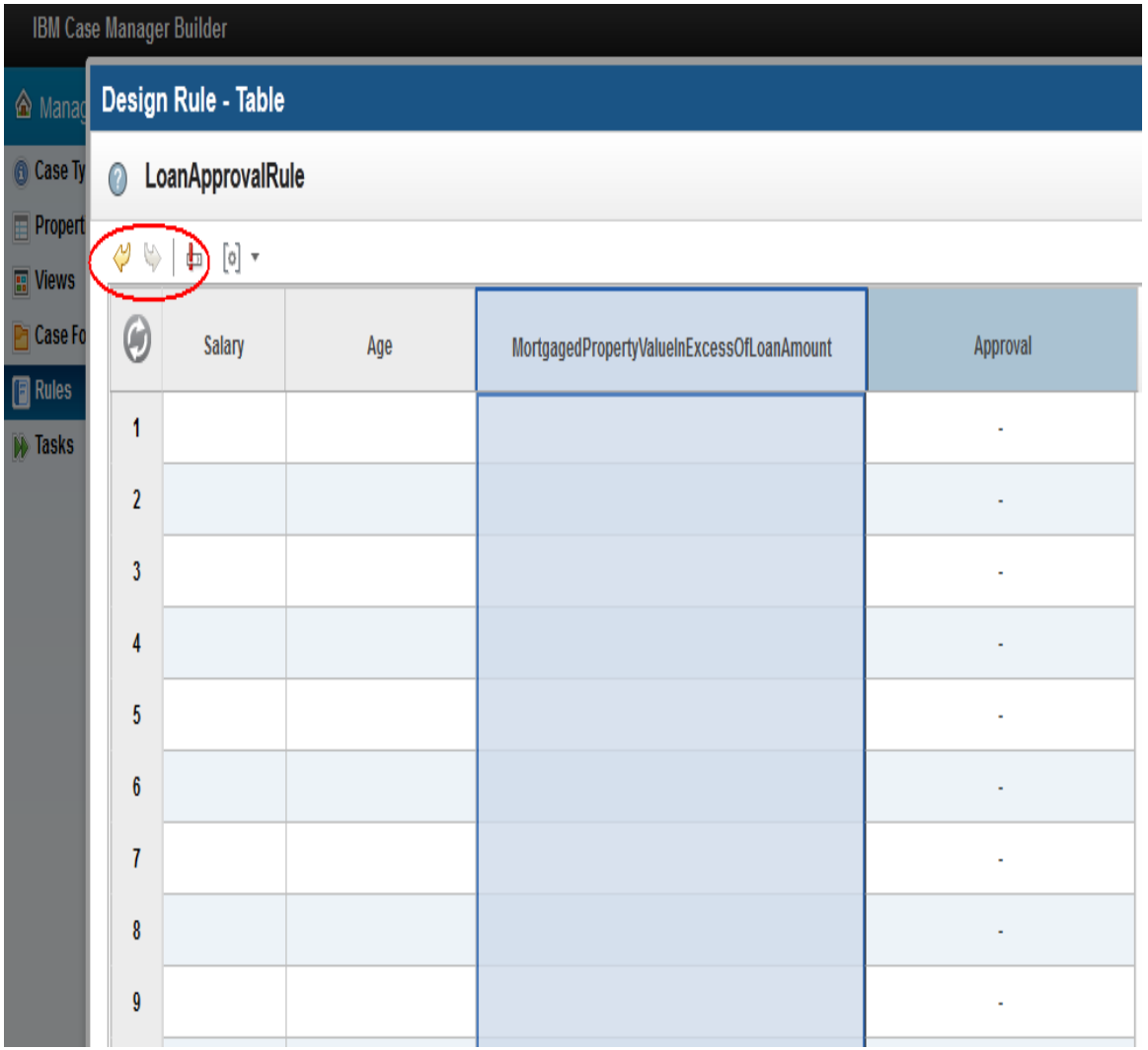
- Introduction
- Configuring Embedded Rules in ICM environment
- Rules Authoring in Case Builder
- Executing Rules as part of a workflow
- Moving Rules from embedded environment to IODM
- ➔ Impact to ICM 5.1.1 customers
- Troubleshooting
- Known Issues, Limitations, Best Practices
- Course Summary

Impact to ICM 5.1.1 customers

- ILOG 7.1 (or IODM) no longer bundled with ICM
- Web services invocation methodology still exists in CPE.
- Customer has purchased ILOG 7.x separately
  - Continue to use ILOG 7.x as is (or upgrade to latest IODM version), without any change to task processes
- Customer has been using bundled ILOG 7.x
  - Procure a license of IODM 8.5, continue to use web services invocation methodology
  - Move to Embedded Rules
    - Manual effort involved.
    - Make an assessment of impact due to functional differences, per client basis.

Embedded Rules Improvements – ICM 5.2.1

- Table-based rules have been improved
  - Undo
  - Redo
  - Preconditions



Embedded Rules Improvements – ICM 5.2.1

- Preconditions enable you to test data before executing rules inside a decision table
- If the preconditions are not satisfied, the rules in the table are not executed
- Use preconditions to declare variables that can be used in the decision table rules

Course Roadmap

- Introduction
- Configuring Embedded Rules in ICM environment
- Rules Authoring in Case Builder
- Executing Rules as part of a workflow
- Moving Rules from embedded environment to IODM
- Impact to ICM 5.1.1 customers
- ➔ Troubleshooting
- Known issues, Limitations, Best Practices
- Course Summary

Troubleshooting - Log Locations

- Rule Deployment errors
  - Deployment log in CPE
  - p8_server_error.log
  - System.out on the AppServer profile where CPE is running
- Rule Runtime errors
  - pesvr_system.log
  - System.out on the AppServer profile where CPE is running
- Web Services invocation details
  - ws*.log file in the CPE logs location

Troubleshooting - Tracing

- Tracing can be enabled for the ICM_RuleOperations Java component
  - Trace entries are written to pesvr_trace.log
  - Enable TRACE_CM flag (component manager tracing flag)
  - Details about properties that are updated by rule execution
  - Details about properties that had a null value before execution
  - Output of all print statements in the rule
  - Results in tracing of rule execution APIs of IODM SDK as well.

Troubleshooting - Case properties not visible in Rule Editor

- Exception in logs

```
Exception created : java.lang.NoSuchMethodError: ilog/rules/xml/schema/IlrXsdSchemaReader.<init>(Z)V
        at com.ibm.rules.sdk.builder.XmlSchema.computeSchemaInfo(XmlSchema.java:304)
        at com.ibm.rules.sdk.builder.XmlSchema.getNamespaceResolver(XmlSchema.java:230)
        at com.ibm.rules.sdk.builder.internal.ombuilders.XsdBuilder.createModels(XsdBuilder.java:97)
```

- Reason : ILOG 71 and ICM 52 or CPE are co-existing in same WebSphere profile
- Suggestion : Not supported configuration, remove ILOG 71 from the profile

Troubleshooting - Boolean case properties not visible

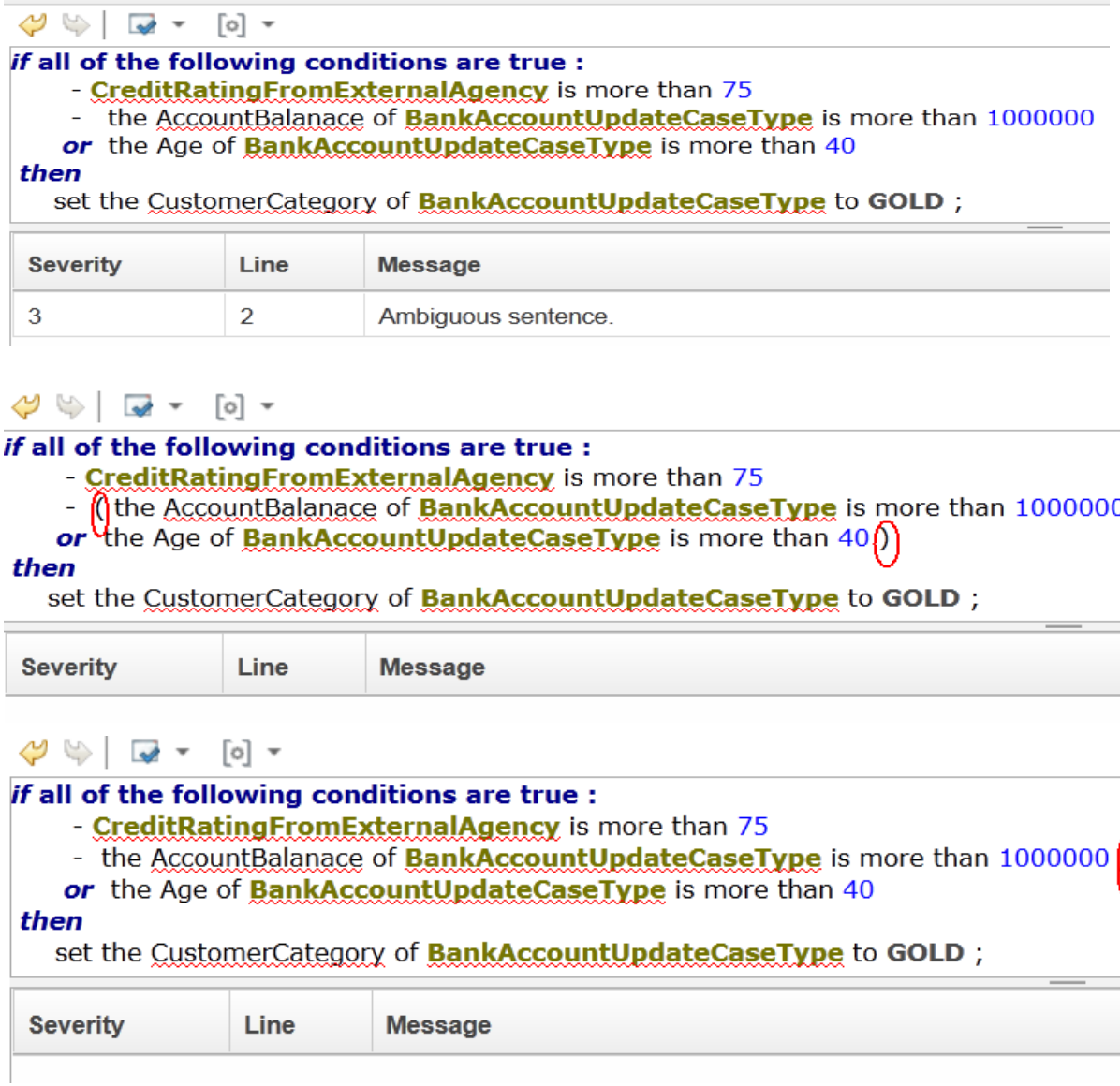
- Symptom : String/Integer/float/Date properties are visible, but boolean case properties are not
- Reason : The way IODM handles boolean properties during authoring is quite different.
- Suggestion : Review the following example
  - If the value of a Boolean property is true, set the value of another Boolean property to false:

    if CaseTypeName is BooleanProperty1

    then make it false that CaseTypeName is BooleanProperty2

Troubleshooting - Ambiguous Sentence in Rules

- Symptom : Rule Editors show 'ambiguous sentence' errors
- Reason : Parser is not able to figure out condition after **or** is a separate 3rd condition or it is part of earlier section.
- Suggestion : Add extra parenthesis or comma to resolve ambiguity.



Troubleshooting - Null Pointer exceptions during rule execution

- Exception in logs

```
Caused by: ilog.rules.engine.IlrUserRuntimeException: null
at call to 'mainRuleflow flow task body'
at call to 'execute'
at ilog.rules.engine.sequential.IlrWorkingMemoryTupleIterator.iterate(IlrWorkingMemoryTupleIterator.java:124)
at
ilog.rules.engine.sequential.IlrSEQBaseExecTask.executeBodyOnWorkingMemory(IlrSEQBaseExecTask.java:161)
at ilog.rules.engine.sequential.IlrSEQBaseExecTask.executeBody(IlrSEQBaseExecTask.java:123)
at ilog.rules.engine.sequential.IlrSEQBaseExecTask.runBody(IlrSEQBaseExecTask.java:80)
```

- Reason : One or more case properties that are retrieved in the rule does not have a value set
- Suggestions
  - Make sure that all case properties retrieved in rule have a value set
  - Make use of defensive rules programming techniques to avoid Null Pointer Exception

Troubleshooting - Rules Pool Size exceeded

- Sample Exception in logs

  [NG101_Task1:4F4F3DF1C7FA514B83C2260F889B892F:Workflow:executeRule] FAILED.; Exception: ilog.rules.res.xu.pool.IlrPoolException: The pool is full.

  at com.ibm.rules.res.xu.pool.internal.PoolImpl.waitNotFull(PoolImpl.java:179)

  at com.ibm.rules.res.xu.spi.internal.XUConnectionManager.allocateConnection(XUConnectionManager.java:84)

  at ilog.rules.res.xu.cci.IlrXUConnectionFactory.getConnection(IlrXUConnectionFactory.java:95)

  at com.ibm.rules.res.xu.client.internal.XUClient.createRuleEngineSession(XUClient.java:111)

  at ilog.rules.res.session.impl.IlrStatefulSessionBase.<init>(IlrStatefulSessionBase.java:105)

- Tune the rules max pool size (default is 100) by adding ***-DrulesMaxPoolSize=<maxSize>*** JVM argument to CPE AppServer profile and restart CPE AppServer.
- Tuning Guideline : Number of expected parallel rule executions in a CPE server

Course Roadmap

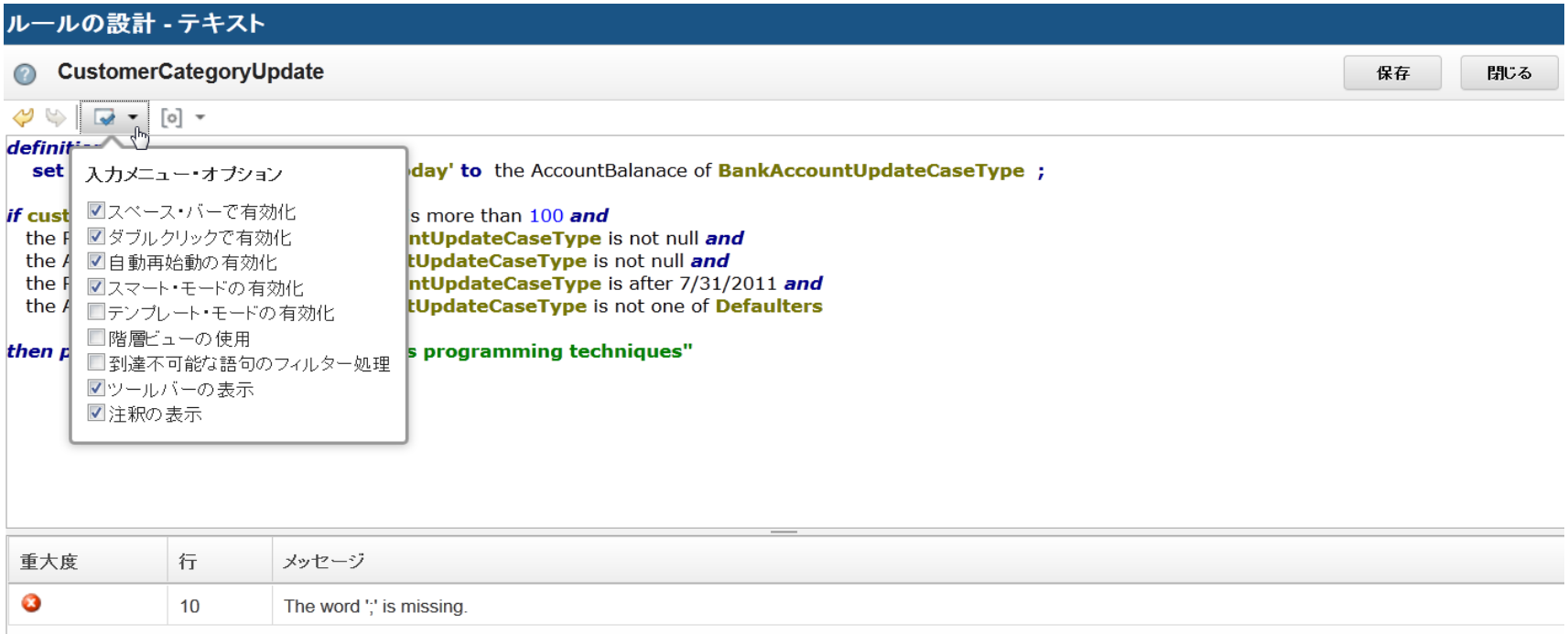
- Introduction
- Configuring Embedded Rules in ICM environment
- Rules Authoring in Case Builder
- Executing Rules as part of a workflow
- Moving Rules from embedded environment to IODM
- Impact to ICM 5.1.1 customers
- Troubleshooting
- ➔ Known Issues, Limitations, Best Practices
- Course Summary

Known Issues

- Limited support for multi valued case properties & multi valued custom parameters in rule authoring
- Limited Arabic & Hebrew rules support – Rule authoring(syntax) supported in Hebrew/Arabic, but recommended to do authoring from an English (or other supported rule persistence locale) environment.
- Traditional Chinese (zh_TW) is not supported – the option is present in the drop-down menu of “Configure Business Rules” task in CMAC, but the rule defaults to English
- There are known issues with embedded Decision Tables in non-English rule persistence locales.

Known Issues – using different locales for browser & rule persistence

- Mix of both locales in Rule editor
  - Syntax & errors in rule persistence locale
  - UI options in browser locale



Limitations

- Embedded Rules wont be able to handle certain characters in case property display names. Examples : <,>,” , (,) etc are not supported in case property display names for rules authoring
- Integer ChoiceLists are not supported in rule authoring
- There is no back-referencing in CaseBuilder from Rules to referred Case Properties or ChoiceLists
- Rule Editors are not fully accessible.

Best Practices

- Use completion menus to avoid errors/issues
- Make sure that case properties that are retrieved in a rule have values set before rule executes
- Use defensive rules programming techniques
  - Null value checks for Date/String case properties
  - Make use of restrictions in **Definitions** section to enforce implicit null value checks

```
definitions
   set 'customerAccountBalanceAsOfToday' to  a number in { the AccountBalanace of BankAccountUpdateCaseType } ;

if customerAccountBalanceAsOfToday is more than 100 and
   the RelationshipStartDate of BankAccountUpdateCaseType is not null and
   the AccountHolderName of BankAccountUpdateCaseType is not null and
   the RelationshipStartDate of BankAccountUpdateCaseType is after 7/31/2011 and
   the AccountHolderName of BankAccountUpdateCaseType is not one of Defaulters

then print "Example for Defensive rules programming techniques" ;
```

Course Roadmap

- Introduction
- Configuring Embedded Rules in ICM environment
- Rules Authoring in Case Builder
- Executing Rules as part of a workflow
- Moving Rules from embedded environment to IODM
- Impact to ICM 5.1.1 customers
- Troubleshooting & Logging
- Known Issues, Limitations, Best Practices
- ➔ Course Summary

Course Summary

You have completed this course and can:

- Describe the salient features of new Embedded Rules functionality
- Configure ICM to enable Embedded Rules features
- Author Text-based & Table-based business rules from Case Builder, use the rules in Tasks and verify rule execution results from Case Client
- Describe the steps involved in migrating Embedded Rules to full IODM product.
- Explain how ICM 5.1.1 customers may be impacted by this new feature
- Troubleshoot rules related issues in your ICM environment.
- Describe Known Issues, Limitations, Best Practices in rules functionality.

Product Help/Documentation/Resources

- Case Manager 5.2 Information Center
  - http://pic.dhe.ibm.com/infocenter/casemgmt/v5r2m0/index.jsp
- P8 5.2 Information Center
  - http://pic.dhe.ibm.com/infocenter/p8docs/v5r2m0/index.jsp
- IODM 8.5 Information Center
  - http://pic.dhe.ibm.com/infocenter/dmanager/v8r5/index.jsp

Questions