UNLEASH THE POWER OF DVS WITH DVSBOOST

INTRODUCTION

The importance of rule testing cannot be overstated. Business rules are at the very heart of key decisions that operate your business and it is imperative that these business rules be thoroughly tested prior to production deployment. Rulesets should be tested independent of the applications code that invokes them. Business users should be able to validate rules without reliance on IT staff.

In recognition of these requirements, Decision Validation Services (DVS) is supplied by IBM as an Operational Decision Manager (ODM) module that enables ruleset testing for developers, business users, and QA engineers. Testing using DVS typically involves creating scenarios (in Excel) that represent real or fictitious use cases to validate the behavior of your rules. These scenarios can be executed by a business user directly from Decision Center.

As a business user testing tool, DVS plays a central role in the rule governance process. It has the potential not only to be used for rule validation by business stakeholders throughout the life of the rule application, but also for regression testing whenever technical refactoring or tuning is performed by IT stakeholders.

DVS LIMITATIONS

The inherent potential of DVS is not realized as it lacks some critical capabilities that inhibit its adoption through the project lifecycle.

Business users frequently run into challenges and ask these questions:

- Invalid Excel input data leads to nasty Java exceptions that are detected only during test execution. **How do I validate** my input Excel data?
- The data relating to a scenario spans across multiple tabs. **How can I easily navigate** across tabs for a specific scenario?
- Data records are often shared across scenarios. Can I identify **which scenarios will be impacted** when I modify a data record?
- My business object model has evolved. **How can I apply these changes** to existing DVS files?
- The development team is changing the execution object model, ruleflow and other details for performance reasons.

How can I ensure that business functionality has not been compromised?

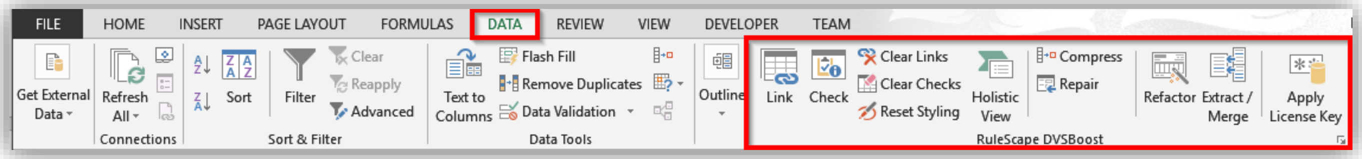
My DVS file has been corrupted due to copy and paste from other DVS files. **How can I repair** this file?

Can I remove unused data records that clutter my DVS spreadsheets?

Over time, as I create several DVS files, **how can I reorganize** the scenarios contained in them?

DVSBoost provides the capabilities to empower a business user to deal with all these issues.

RULESCAPE DVSBOOST



RuleScape DVSBoost, designed to be used in conjunction with DVS, is an Excel Add-in which seamlessly integrates with DVS and provides a number of capabilities that enhance the usability and management of DVS assets. DVSBoost unlocks the full promise and potential of DVS by overcoming its limitations in the areas of:

1. Refactoring and managing existing DVS assets
2. Navigation and viewing of scenarios
3. Validation of scenario data

In addition, lack of a test coverage report during test execution is a major limitation of DVS. This is rectified by ***RuleScape DVSBatch***, and is described in a separate document.

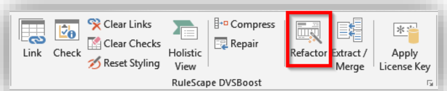
REFACTORING AND MANAGING EXISTING DVS ASSETS

As is said: “the only constant is change”! As anybody who has done ODM development for any period of time will attest, the business object model (BOM) evolves over time. This evolution is rapid during the inception of a project and during subsequent big releases, but rarely ever stops. When changing the BOM, the ability to use existing DVS test cases to perform regression testing is invaluable.

When your BOM is modified, you have to generate a new DVS template file; but what about the existing DVS spreadsheets? Bluntly put, the reality is that prior DVS assets cannot be used after any non-trivial BOM change.

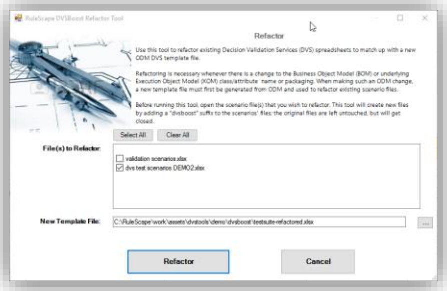
There are several types of refactoring that can invalidate existing DVS assets. Some of these changes, such as changing dynamic domain values, can even be initiated by business users with no involvement of IT. This makes it even more critical for business users to have the refactoring tools.

With DVSBoost, business users are enabled to refactor DVS files.



Use ***DVS Refactor Tool*** to automatically refactor DVS scenario files whenever there is a BOM/XOM evolution. DVSBoost applies sophisticated refactoring logic to multiple DVS scenario files and can handle all your typical refactoring requirements, including:

- ➢ Add, remove or rename BOM attributes
- ➢ Change the underlying XOM (Java or XSD) of an existing BOM attribute
    - Change data type of an attribute
    - Change package (or namespace) structure
    - Change XOM class name or attribute name
- ➢ Add, remove or modify enumerated values or dynamic domain values
- ➢ Add, remove or rename ruleset parameters



Use this tool to refactor existing Decision Validation Services (DVS) spreadsheets to match up with a new ODM DVS template file.

Refactoring is necessary whenever there is a change to the Business Object Model (BOM) or underlying Execution Object Model (XOM) class/attribute name or packaging. When making such an ODM change, a new template file must first be generated from ODM and used to refactor existing scenario files.

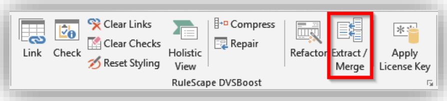
Before running this tool, open the scenario file(s) that you wish to refactor. This tool will create new files by adding a "dvsboost" suffix to the scenarios' files: the original files are left untouched, but will get closed.

File(s) to Refactor:
☐ validation scenarios.xlsx
☑ dvs test scenarios DEMO2.xlsx

New Template File: C:\RuleScape\work\assets\dvstools\demo\dvsboost\testsuite-refactored.xlsx

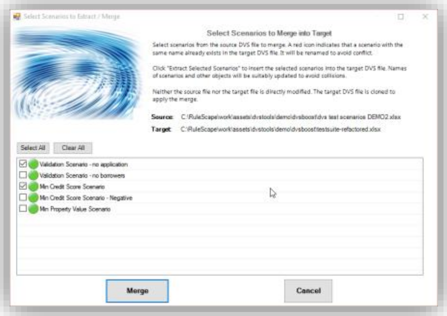
Rules are designed to accommodate change. As rules change, new DVS scenarios may be added or existing ones may be modified. Over time, multiple business users may build test scenarios across multiple files. As the number of scenarios and DVS spreadsheets keep rising, there usually comes a time during the lifecycle of the project when you will want to do some housekeeping on these assets. For instance, a business user may wish to merge one or more scenario files, or may want to extract a subset of scenarios from an existing file to a new file.

DVSBoost provides tools to ***Repair*** a damaged DVS spreadsheet and ***Compress*** to remove all irrelevant records.



Use ***Extract / Merge*** to manage scenarios across multiple DVS spreadsheets. With this tool, you can:

1) Merge scenarios from several source DVS spreadsheets into a target spreadsheet.
2) Merge a subset of scenarios from a source DVS spreadsheet into a target spreadsheet.
3) Extract a subset of scenarios from a source DVS spreadsheet.



Select Scenarios to Merge into Target

Select scenarios from the source DVS file to merge. A red icon indicates that a scenario with the same name already exists in the target DVS file. It will be renamed to avoid conflict.

Click "Extract Selected Scenarios" to insert the selected scenarios into the target DVS file. Names of scenarios and other objects will be suitably updated to avoid collisions.

Neither the source file nor the target file is directly modified. The target DVS file is cloned to apply the merge.

Source: C:\RuleScape\work\assets\dvstools\demo\dvsboost\dvs test scenarios DEMO2.xlsx
Target: C:\RuleScape\work\assets\dvstools\demo\dvsboost\testsuite-refactored.xlsx

☑ Validation Scenario - no application
☐ Validation Scenario - no borrowers
☑ Min Credit Score Scenario
☐ Min Credit Score Scenario - Negative
☐ Min Property Value Scenario

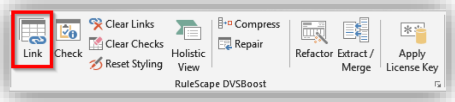
This tool intelligently merges only the relevant rows. Irrelevant data records are not merged.

It also performs sophisticated automatic conflict management when data records in different DVS spreadsheets have the same ID but have different data values. DVSBoost automatically renames conflicting records and consistently applies it across scenarios thereby removing collisions.

NAVIGATION AND VIEWING

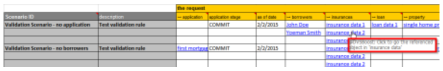
A DVS spreadsheet in essence is an Excel representation of an object graph containing the business object model instances for the input and output parameters of a ruleset. To understand a scenario of any complexity, it is important to be able to view and navigate this object graph; however, there is no easy way to navigate this object graph or to view it in its entirety in DVS.

Data records can be reused by multiple scenarios, though it is not easy to identify which scenarios use a particular data record. Without this information, you are prone to inadvertently impact scenarios when modifying a data record. Navigation and viewing are particularly thorny problems when dealing with DVS worksheets that have a large number of rows or when dealing with complex domains that have a number of classes.



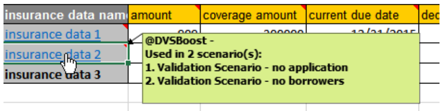
DVS Link tool is used to mitigate this - it create hyperlinks in the DVS spreadsheet. This functionality includes:

- Link and backlink a parent record with the children records on a different tab.
- Link and backlink a scenario with expected results for that scenario.
- Display which scenarios a particular record is involved in.



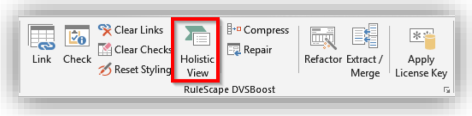
		the request						
Scenario ID	description	-- application	application stage	as of date	-- borrowers	-- insurances	-- loan	-- property
Validation Scenario - no application	Test validation rule		COMMIT	2/2/2015	John Doe	insurance data 1	loan data 1	single home pr
					Yeoman Smith	insurance data 2		
						insurance		
Validation Scenario - no borrowers	Test validation rule	first mortgag	COMMIT	2/2/2015		insurance		
						insurance data 2		
						insurance data 2		

@DVSBoost: Click to go the referenced object in 'insurance data'

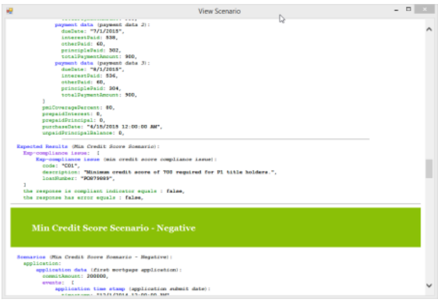


insurance data nam	amount	coverage amount	current due date	dec
insurance data 1	[illegible]	200000	12/31/2015	
insurance data 2				
insurance data 3				

@DVSBoost -
Used in 2 scenario(s):
1. Validation Scenario - no application
2. Validation Scenario - no borrowers



Use ***Holistic View*** to get a complete view of one or more scenarios. The holistic view comprises of the scenario data along with all associated objects across all the tabs (input and output).



```
            payment data (payment data 2):
              dueDate: "7/1/2015",
              interestPaid: 538,
              otherPaid: 60,
              principlePaid: 302,
              totalPaymentAmount: 900,
            payment data (payment data 3):
              dueDate: "8/1/2015",
              interestPaid: 536,
              otherPaid: 60,
              principlePaid: 304,
              totalPaymentAmount: 900,
          }
          pmiCoveragePercent: 80,
          prepaidInterest: 0,
          prepaidPrincipal: 0,
          purchaseDate: "4/15/2015 12:00:00 AM",
          unpaidPrincipalBalance: 0,
  Expected Results (Min Credit Score Scenario):
    Exp-compliance issue:  [
        Exp-compliance issue (min credit score compliance issue):
          code: "C01",
          description: "Minimum credit score of 700 required for P1 title holders.",
          loanNumber: "PC075609",
    ]
    the response is compliant indicator equals : false,
    the response has error equals : false,
```

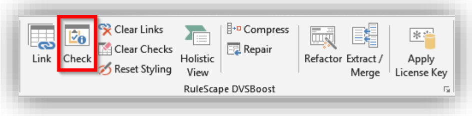
Min Credit Score Scenario - Negative

```
  Scenarios (Min Credit Score Scenario - Negative):
    application:
        application data (first mortgage application):
          commitAmount: 200000,
          events:  [
            application time stamp (application submit date):
```

This tool creates a printable HTML report for one or more selected scenarios in JSON or a more human-readable format.

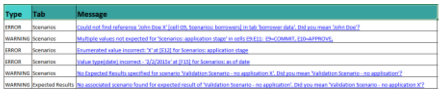
VALIDATION OF SCENARIO DATA

It is quite easy to make errors when entering data, such as using an invalid name for associated objects resulting in "dangling" references, or not entering the right data type, or omitting a mandatory field, just to name a few.

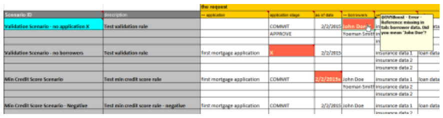


Use the ***Check*** tool to validate the scenario data in the currently open DVS file. This tool sweeps through all the worksheets in the DVS file and performs the following data checks:

- Validate that the DVS spreadsheet is structurally sound and that all reference worksheets are present.
- Validate that there are no dangling references.
- Validate that each scenario has a corresponding expected result.
- Validate that a cell value is a valid enumerated value.
- Validate that a cell data element is of the right data type.
- Validate that mandatory values are provided.
- Validate that multiple values are not provided where a single value is expected.



Type	Tab	Message
ERROR	Scenarios	Could not find reference 'John Doe X' [cell G9, Scenarios: borrowers] in tab 'borrower data'. Did you mean 'John Doe'?
WARNING	Scenarios	Multiple values not expected for 'Scenarios: application stage' in cells E9:E11: E9=COMMIT, E10=APPROVE,
ERROR	Scenarios	Enumerated value incorrect: 'X' at [E12] for Scenarios: application stage
ERROR	Scenarios	Value type(date) incorrect - '2/2/2015x' at [F15] for Scenarios: as of date
WARNING	Scenarios	No Expected Results specified for scenario 'Validation Scenario - no application X'. Did you mean 'Validation Scenario - no application'?
WARNING	Expected Results	No associated scenario found for expected result of 'Validation Scenario - no application'. Did you mean 'Validation Scenario - no application X'?



		the request					
Scenario ID	description	-- application	application stage	as of date	-- borrowers		
Validation Scenario - no application X	Test validation rule		COMMIT	2/2/2015	John Doe	in	
			APPROVE		Yoeman Smit	in	
						in	
Validation Scenario - no borrowers	Test validation rule	first mortgage application	X	2/2/2015		insurance data 1	loan data
						insurance data 2	
						insurance data 2	
Min Credit Score Scenario	Test min credit score rule	first mortgage application	COMMIT	2/2/2015x	John Doe	insurance data 1	loan data
					Yoeman Smit	insurance data 2	
						insurance data 2	
Min Credit Score Scenario - Negative	Test min credit score rule - negative	first mortgage application	COMMIT	2/2/2015	John Doe	insurance data 1	loan data
						insurance data 2	

@DVSBoost - Error - Reference missing in tab: borrower data. Did you mean 'John Doe'?

Upon invoking ***Check***, all the data errors are identified and reported in a *DVSBoost Report* worksheet. This report lists all detected validation issues (ERROR or WARNING) along with suggested corrective action. Clicking on the validation issue text navigates you directly to the erroneous cell.

RULESCAPE DVSBOOST BENEFITS

DVSBoost seamlessly integrates with DVS and unlocks its full potential and promise. With DVSBoost, the rule validation function can truly be accomplished by business stakeholders with minimal involvement from IT staff.

- ✓ Put the business users in control of the test scenarios by allowing them to easily navigate even complex object models through hyperlinks.
- ✓ Prevent accidental scenario modifications by enabling business users to easily identify which scenarios use a data record and comprehend the impact of data changes across multiple scenarios.
- ✓ Through automatic refactoring, enable the continued use of DVS scenario files even after the business object model significantly evolves.
- ✓ Prevent clutter from building up in scenario files by identify and removing orphan data records that are not used in any scenarios.
- ✓ Enable testers to easily verify or share a test case with other teams by generating a report containing the complete scenario.
- ✓ Improve efficiency by allowing testers to identify and correct any input data errors prior to submitting the data for execution.
- ✓ Enable business users to safely create a variation of a scenario by allowing them to extract that scenario to a separate file.
- ✓ Allow users to structurally repair DVS files that have been accidentally corrupted.
- ✓ Improve effectiveness of business users by enabling them to perform housekeeping activities (merging or splitting) DVS files.