

# ENHANCING DVS EXECUTION WITH DVSBATCH

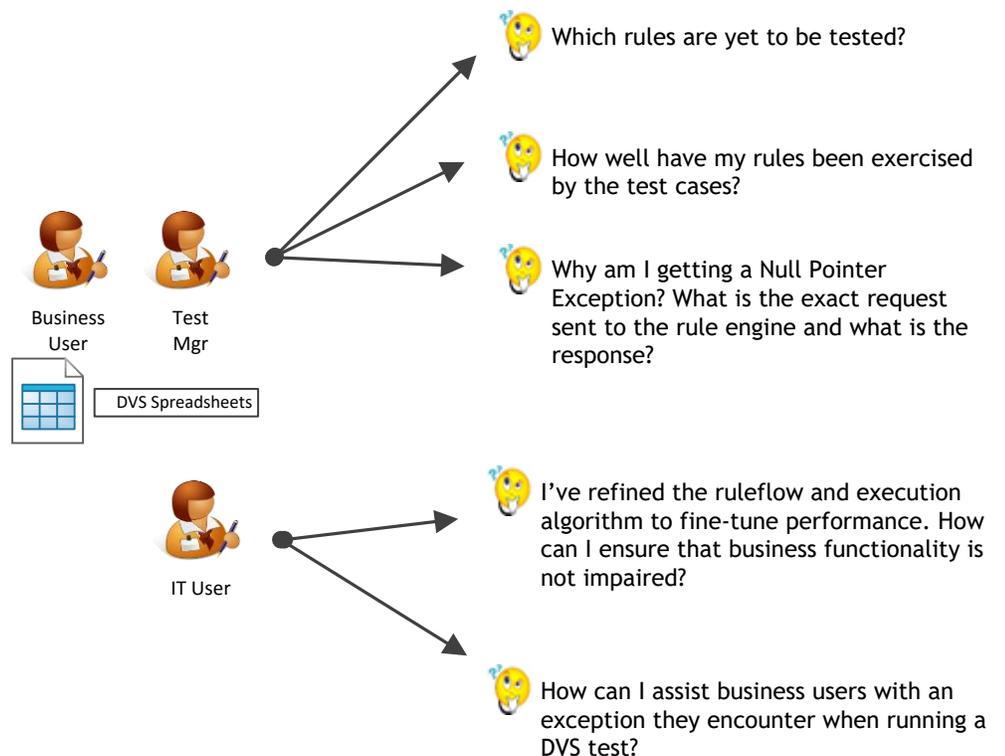
## INTRODUCTION

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.

## DVS CHALLENGES

As a business user testing tool, DVS plays a central role in the rule governance process for business rule validation. But it has the potential for more. It has the potential to be used for regression testing whenever technical refactoring or tuning is performed by IT stakeholders.

However, DVS is not free of challenges. IT and business users run into these challenges when executing DVS test cases:



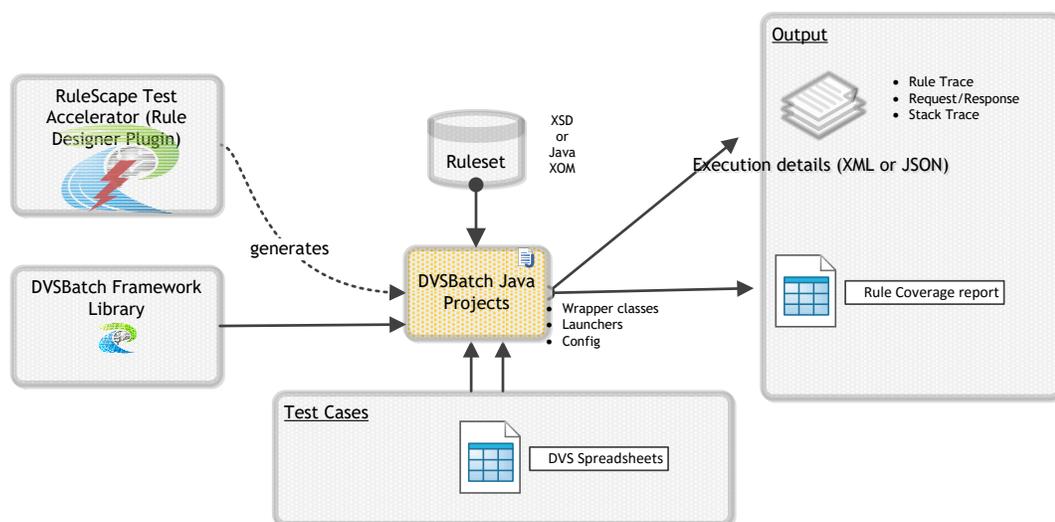
RuleScope DVSBatch is the answer to realize the execution potential of DVS and overcome its challenges.

## RULESCOPE DVSBATCH

RuleScope DVSBatch is a tool that executes some or all of your DVS scenario files and reveals insightful execution details that help you devise better test cases and identify areas which need more testing. In addition, it helps you leverage your DVS test scenarios as a means of regression testing whenever a major change has been made to the rule foundation.

DVSBatch is packaged as an Eclipse plug-in Test Accelerator which generates the Java projects for you to execute the tool. Using the tool is as simple as placing your DVS scenario files in a configurable folder and running the prepackaged launcher.

At the core of the functionality is the **Coverage Report** generated by DVSBatch which provides rich details relating to how well your tests cover your rules. This helps you to decide which rules now need to be your focus of testing. Besides the coverage report, DVSBatch can be configured to log the inputs, outputs, rule traces and exception stack traces for your test run. These can serve as a baseline for future regression tests.



DVSBatch recognizes and supports two distinct scopes of execution:

- 1) **Regression** Testing, which entails running all your DVS test scenarios against the latest ruleset and assessing its validity by examining the failed scenarios.

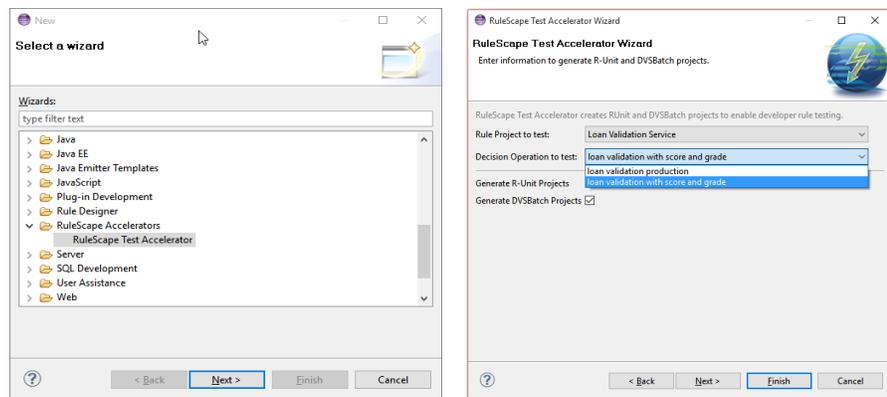
- 2) **Triage** Testing, which involves IT staff running selected DVS test cases (typically just 1) that are not yielding the results expected by business users.

Additionally, DVSBatch recognizes that there are situations where some of the rules may not be in scope for testing purposes; for instance, rules that will only become effective next year. As part of its configuration, DVSBatch allows you to identify out-of-scope rules that you want to ignore for test coverage reporting. DVSBatch brings this into consideration when calculating the *effective* coverage.

DVSBatch is very simple to use. The straight-forward steps to using DVSBatch tool are:

1. Generate pre-configured DVSBatch projects using **Test Accelerator Plug-in**.
2. Copy your DVS files into **DVSBatch project** and add additional configuration parameters if necessary.
3. Execute the tool using one of the provided launchers and gain insights through the **Excel Coverage Report** and other execution logs.

## TEST ACCELERATOR PLUG-IN



Test Accelerator, an Eclipse plug-in that extends ODM Rule Designer, provides a simple wizard that generates preconfigured DVSBatch projects based on the rule project (and decision operation, if necessary) you've selected to test.

The Test Accelerator generates all the scaffolding necessary for you to execute DVSBatch, which includes framework libraries, preconfigured Java projects and launchers to execute DVSBatch.

## DVSBATCH PROJECT CONFIGURATION AND EXECUTION

The Test Accelerator generates pre-configured DVSBatch projects that are ready to execute. However, you can easily reconfigure test execution using the supplied configuration file. Common configuration changes include:

- Logo to be used in reports
- Identification of out-of-scope rules. These are rules that are present in the ruleset, but are not in scope for current testing and should be excluded for coverage purposes.
- Location of folders for Regression DVS tests, Triage DVS tests and reports.

Several launchers, for different test scopes and with differing logging facilities, are provided in the DVSBatch project. Once DVSBatch has been configured, execution of either Regression Test or Triage Test is simply a matter of running these launchers. The test execution generates log files and a coverage report in the *reports* folder.

## COVERAGE REPORT

DVSBatch generates an Excel Coverage Report that provides you with deep insights relating to the test execution. This report has four worksheets:

1. Execution Summary
2. Tested Rules
3. Untested Rules
4. Rule Package Statistics

---

## EXECUTION SUMMARY

The Execution Summary report tells you at a glance how well your tests ran. You can gauge the validity of your ruleset through the test success rate, and gauge performance through the total execution time. In addition, the Rule Coverage Summary gives you a snapshot of how many rules were not tested.

TEST COVERAGE REPORT	
<b>Test Execution Statistics</b>	
Report creation date:	9/17/2015 11:28
Test execution started on:	Thu Sep 17 11:28:19 EDT 2015
Test execution ended on:	Thu Sep 17 11:28:25 EDT 2015
Test execution duration (secs):	5.87
Tested ruleset:	C:\samples\loan_validation_with_score_and_grade.jar
Ruleset generated on:	Thu Sep 17 10:19:01 EDT 2015
Number of test files processed:	3
Number of test scenarios:	8
Number of failed scenarios:	4
Test success rate (%):	<b>50.00%</b>
<b>Rule Coverage Summary</b>	
Total number of rules:	74
Number of 'real' rules (without ignored rules):	74
Number of rules tested:	23
Number of rules not tested:	51
Coverage (%):	<b>31.08%</b>
Number of out-of-scope rules:	24
Effective coverage (%):	<b>46.00%</b>
<span>Execution Summary</span>   <span>Tested Rules</span>   <span>Untested Rules</span>   <span>Rule Package Stats</span>   <span>+</span>	

## TESTED RULES

The Tested Rules worksheet depicts how well a rule has been tested and provides details on which scenarios tested each of the rules. This information is valuable in determining which test conditions a particular rule has been tested under. This information can guide you in devising more significant tests for a rule.

	Rule #	Rule Name	Rule Package	Total Executions	Executed in Scenarios
1					
2	1	bankruptcyScore 1	computation	2	Chapter 1 Scenario - PASSED
3					Chapter 11 Recent Scenario - PASSED
4					
5	2	bankruptcyScore 4	computation	1	Chapter 11 Old Scenario - PASSED
6					
7	3	initialCorporateScore	computation	5	Chapter 1 Scenario - PASSED
8					Chapter 11 Old Scenario - PASSED
9					Chapter 11 Recent Scenario - PASSED
10					High Income Scenario - FAILED
11					Low Income Scenario - PASSED
12					
13	4	neverBankruptcy	computation	2	High Income Scenario - FAILED
14					Low Income Scenario - PASSED
15					
16	5	rate 4	computation	5	Chapter 1 Scenario - PASSED
17					Chapter 11 Old Scenario - PASSED
18					Chapter 11 Recent Scenario - PASSED
19					High Income Scenario - FAILED

## UNTESTED RULES

Untested Rules worksheet lists out rules that have not been tested by any of the scenarios in this test run. When your entire set of DVS scenarios is run as a regression test, this list reveals which rules are yet to be tested and serves as a guide for helping you determine the area of test focus.

UNTESTED RULES			
	Rule #	Rule Name	Rule Package
1			
2	1	bankruptcyScore 2	computation
3			
4	2	bankruptcyScore 3	computation
5			
6	3	salary2score 1	computation
7			
8	4	salary2score 2	computation
9			
10	5	salary2score 4	computation
11			
12	6	salary2score 5	computation

## RULE PACKAGE STATS

The Rules Package Stats worksheet displays test coverage details at the rule package level. For each rule package, the calculated effective coverage rate is helpful in determining which package needs attention from testers.



Rule Package	Rules Tested	Total Rules	Rules Out of Scope	Test Coverage	Effective Coverage
ROOT:	23	74	24	31.08%	46.00%
computation:	8	39	23	20.51%	50.00%
eligibility:	5	15	1	33.33%	35.71%
insurance:	5	14	0	35.71%	35.71%
validation:	5	6	0	83.33%	83.33%
borrower:	5	5	0	100.00%	100.00%
loan:	0	1	0	0.00%	0.00%

## EXECUTION LOGS

DVSBatch generates text files containing the rule trace and execution details for any failed test case, but can be configured to do so for all test cases.

The execution details of each scenario is placed in a separate, easily identifiable text file. It not only contains the test results identifying any discrepancies between expected and actual outcomes, but also logs the request data and the entire observed response in a human-readable format.

```
-----
TEST RESULTS
-----
Error when processing scenario: 'High Income Scenario'

**Test 1:the grade equals - PASSED
**Test 2:the loan report is approved equals - PASSED
**Test 3:the loan report is insurance required equals - PASSED
**Test 4:the messages of the loan report equals (unordered) - FAILED

Expected:
[
  Low risk loan,
  Congratulations! Your loan has been approved
]

Actual:
[
  Average risk loan,
  Congratulations! Your loan has been approved
]

**Test 5:the monthly repayment of the loan report equals - PASSED
-----

OBSERVED RESPONSE
-----

**report:

Report:
  approved: true
  borrower:
    Borrower:
      age: 65
      birthDate: 1950-01-01
      creditScore: 200
      firstName: John
      lastName: Smith
      spouse:
      ssn:
        SSN:
          areaNumber: 123
          digits: 9
          fullNumber: 123-12-1234
          groupCode: 12
          serialNumber: 1234
          yearlyIncome: 200000
          zipCode: 12345
      insurance: 2%
      insuranceRate:
        double: 0.02
      insuranceRequired: true
```

Rule trace lists all the rule tasks and rules invoked during execution of the test case. This information is valuable as a baseline, particularly for complex scenarios, to determine which rule is failing to run for the same test scenario on a future date.

```
High Income Scenario - FAILED - RULETRACE.txt
-->loanvalidation
-->loanvalidation>initResult
-->loanvalidation>validation
validation.borrower.checkSSNdigits
validation.borrower.checkZipcode
validation.borrower.checkAge
validation.borrower.checkSSNareanumber
validation.borrower.checkName
-->loanvalidation>computation
computation.initialCorporateScore
computation.neverBankruptcy
computation.salary2score 8
computation.rate 4
computation.repayment
-->loanvalidation>eligibility
eligibility.grade 6
eligibility.approval
-->loanvalidation>insurance
insurance.unvalidated action rule
insurance.insurance 10
insurance.defaultInsurance
```

## RULESCAPE DVSBATCH BENEFITS

DVSBatch has utility for managers, business users and IT staff.

- ✓ Provides managers a means to track test progress, gauge robustness of the ruleset and identify any areas of weakness.
- ✓ Guides business users in identifying rules that should be the focus of their testing.
- ✓ Provides IT staff and business users a means to leverage DVS test suites for regression testing.
- ✓ Improves efficiency by providing a mechanism to view actual responses when running DVS scenarios without having to go to the Decision Warehouse.
- ✓ Simple to use and easily configurable.
- ✓ Rich set of reports and logs.