

Paris, 16-18 October 2018



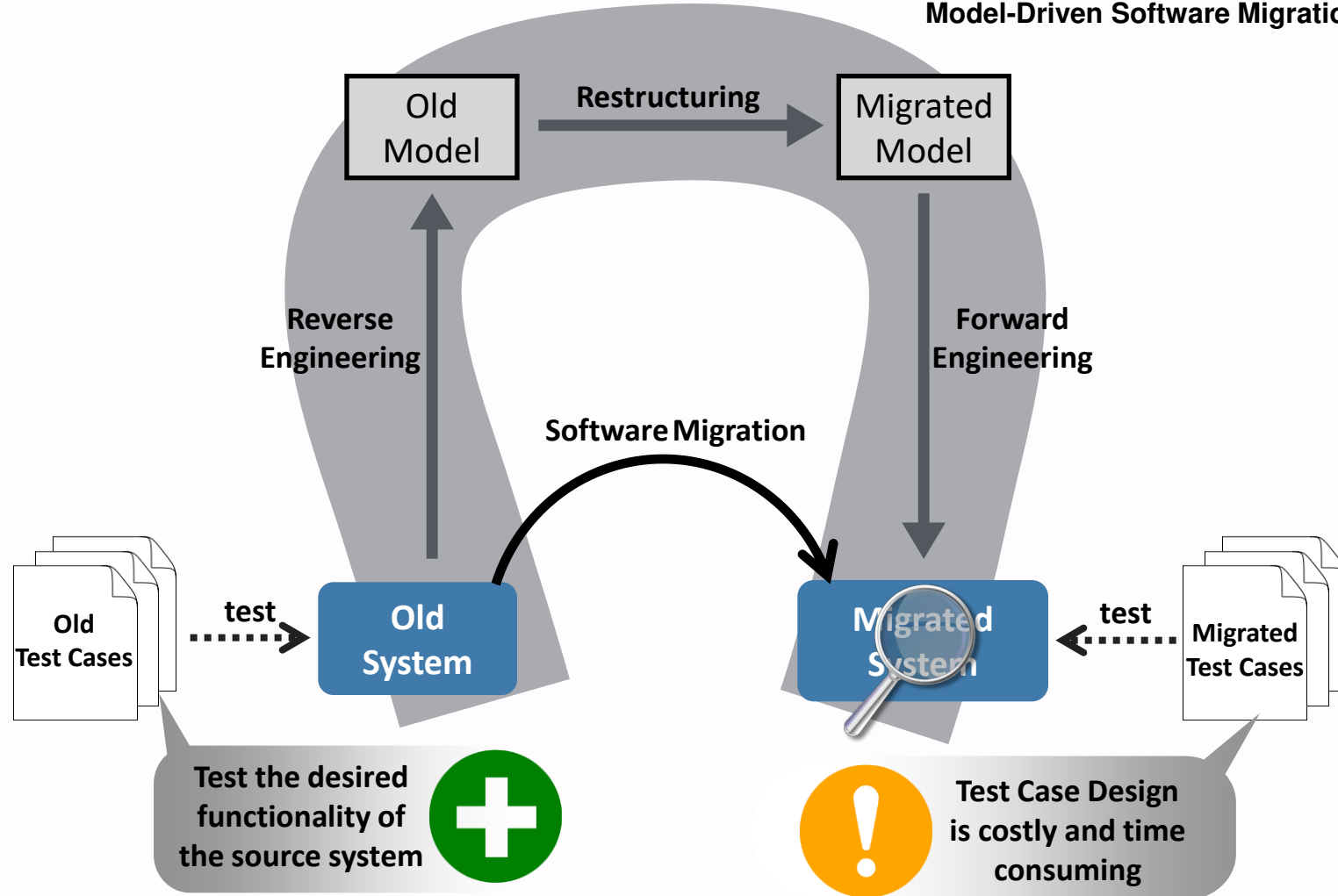
Organizer: **TESTING**
SOLUTIONS
& **SERVICES** 

WHO GUARDS THE GUARDS? ON THE VALIDATION OF TEST CASE MIGRATION

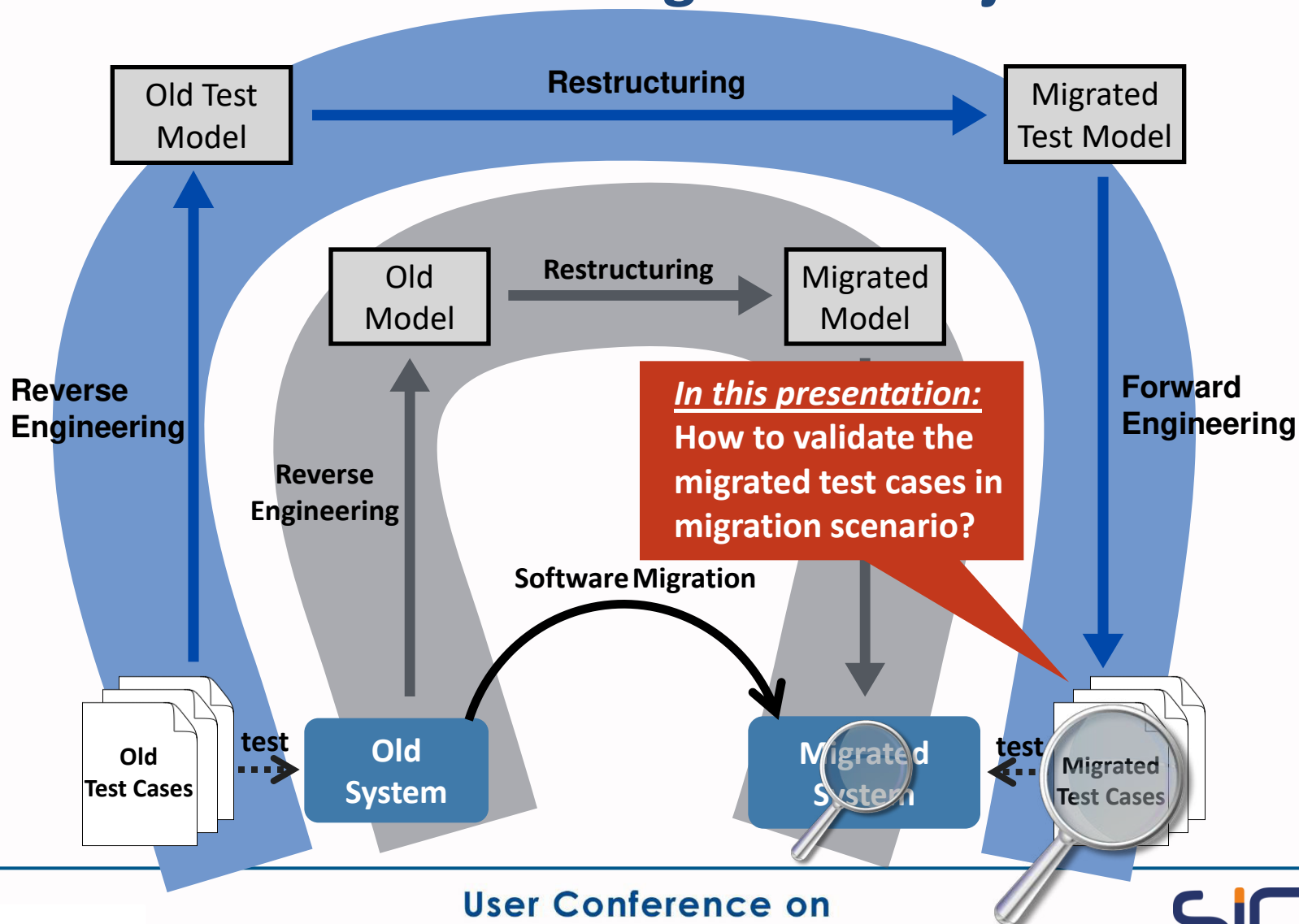
Presented by Ivan Jovanovikj

Reuse of Test Cases in Migration Projects

Model-Driven Software Migration OMG ADM



Reuse of Test Cases in Migration Projects



Validation of Test Case Migration

How to validate a test case migration?



What is a valid test case migration?



Test case migration is a process of transferring test cases into new environments **without changing their functionality**, i.e., **without changing what they test**.

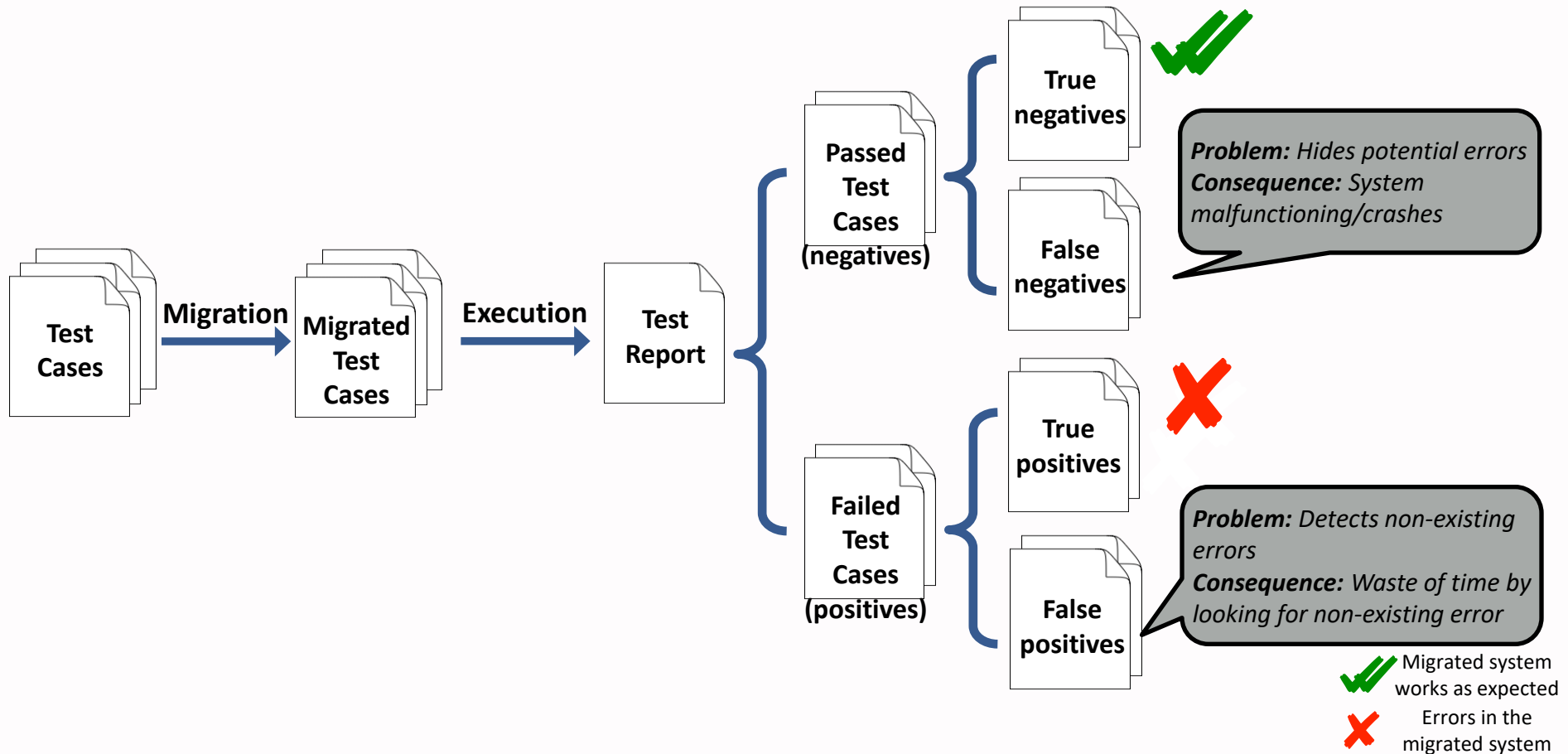
Behavioral Equivalence



How to ensure behavioral equivalence in test case migration ?



Validation of Test Case Migration



Validation of Test Case Migration

How to ensure behavioral equivalence in test case migration?



How to avoid/detect false positives and false negatives?



Constructive
approaches



Analytical
Approaches



Mutation
Testing



Bottom Line

Cost of
Test Case
Migration

+

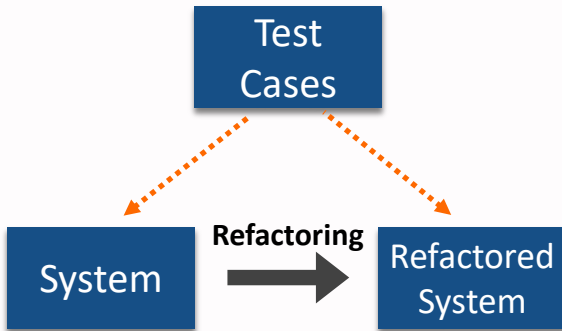
Cost of
Validation of Test
Case Migration

\leq

Cost of
Developing New
Test Cases

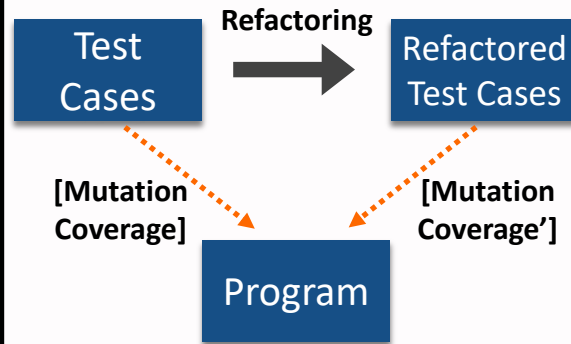
Validation of Test Case Migration

Code Refactoring



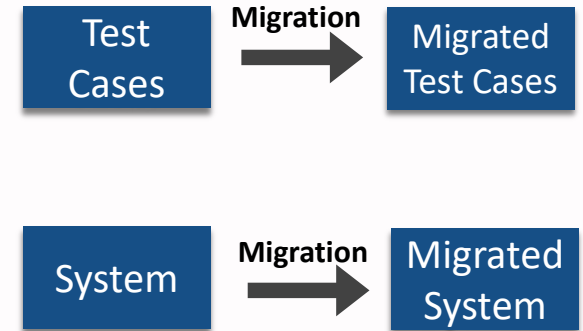
All Test Cases pre- and post-program refactoring should pass

Test Case Refactoring



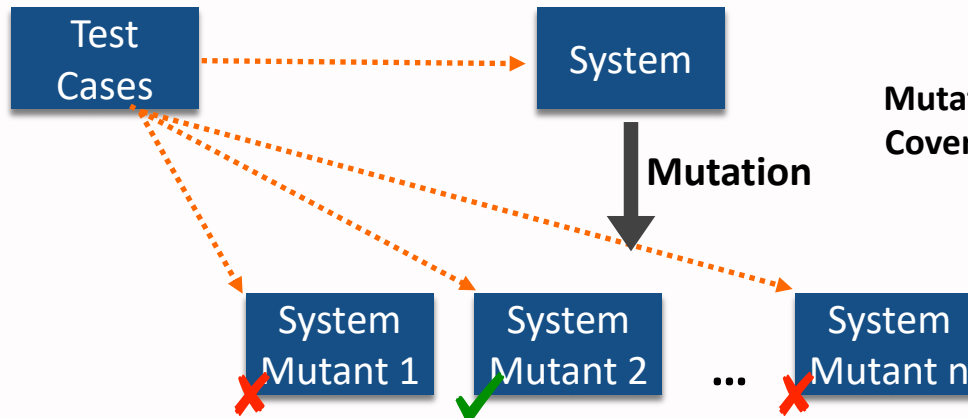
[Mutation Coverage] = [Mutation Coverage']

System and Test Case Migration



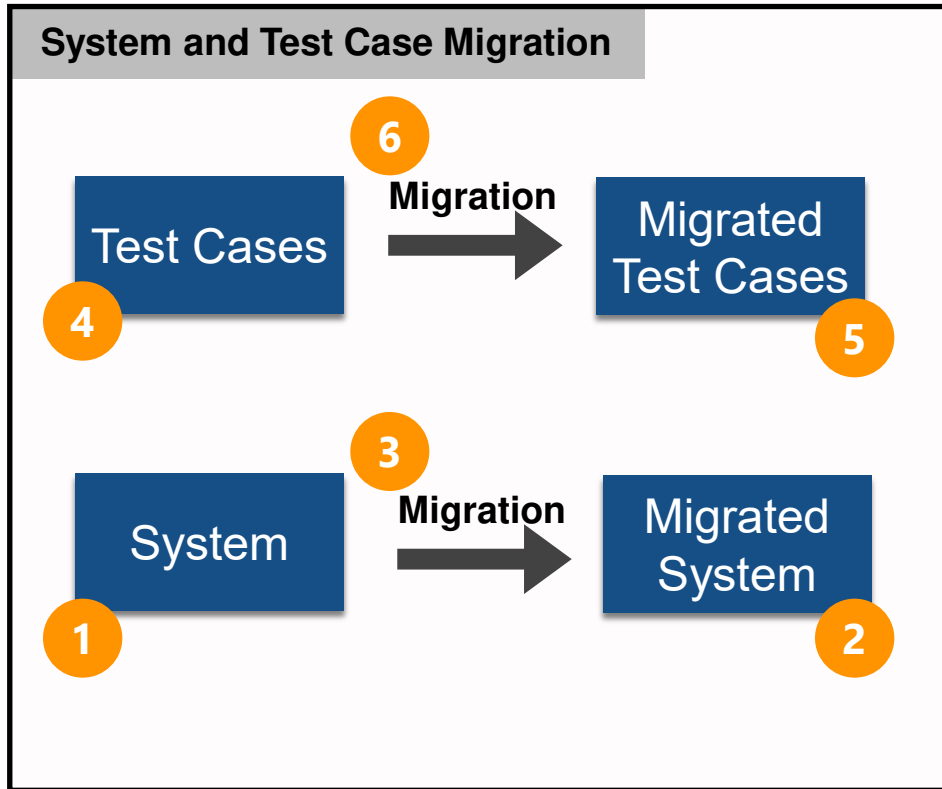
Executed against

✗ mutant killed
✓ mutant survived



$$\text{Mutation Coverage} = \frac{\text{Killed Mutants}}{\text{All Mutants} - \text{Equivalent Mutants}}$$

Mutation Analysis Scenarios



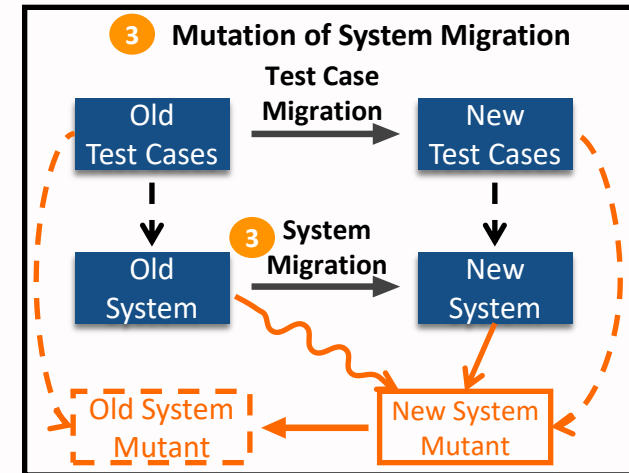
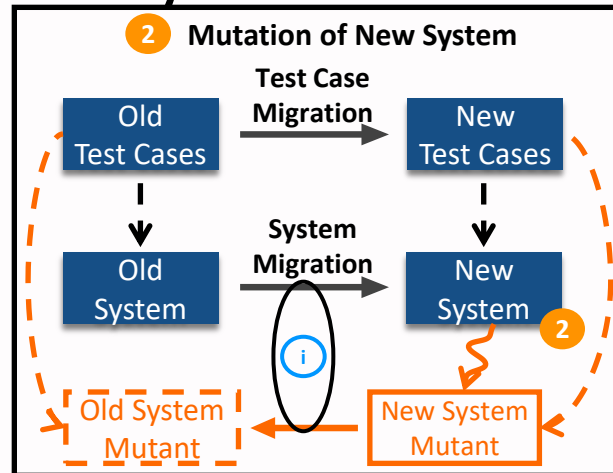
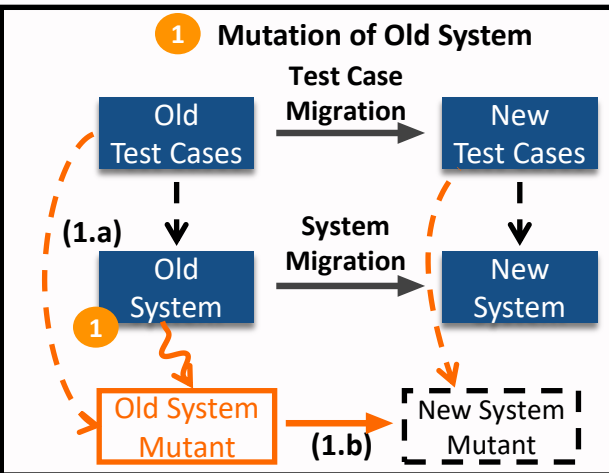
- 1 Mutation of Old System
- 2 Mutation of New System
- 3 Mutation of System Migration
- 4 Mutation of New Test Cases
- 5 Mutation of Old Test Cases
- 6 Mutation of Test Case Migration

Assumptions

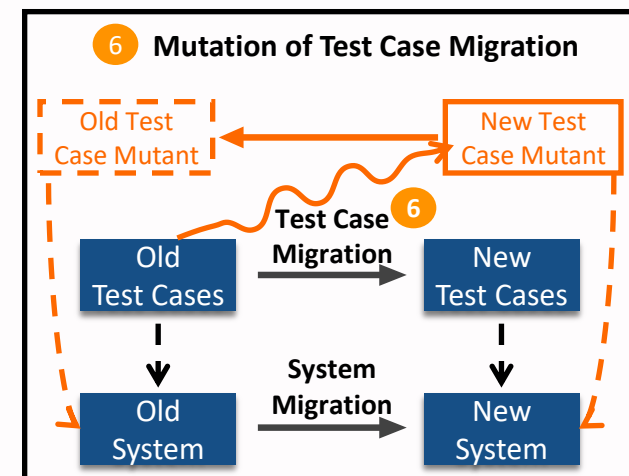
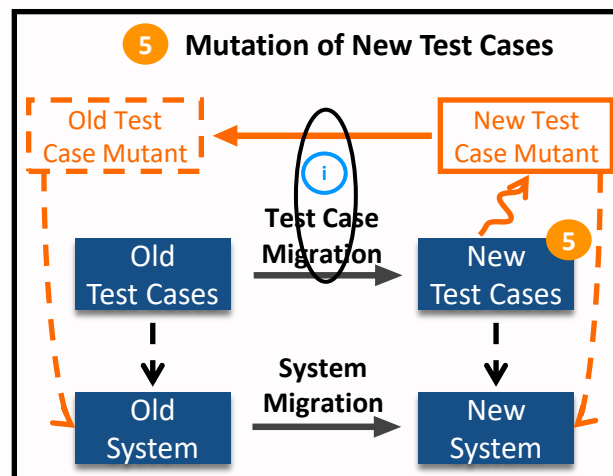
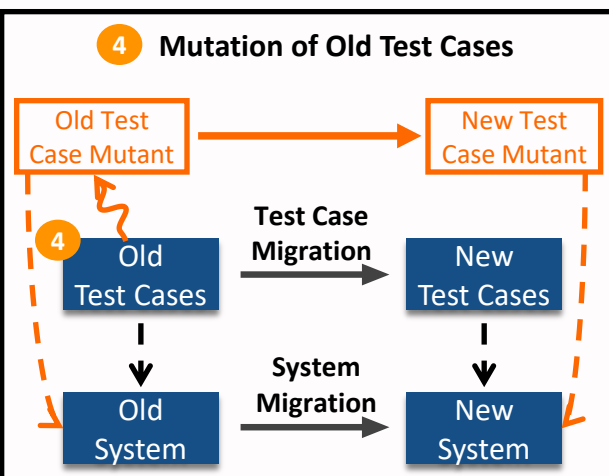
Indications

Mutation Analysis Scenarios

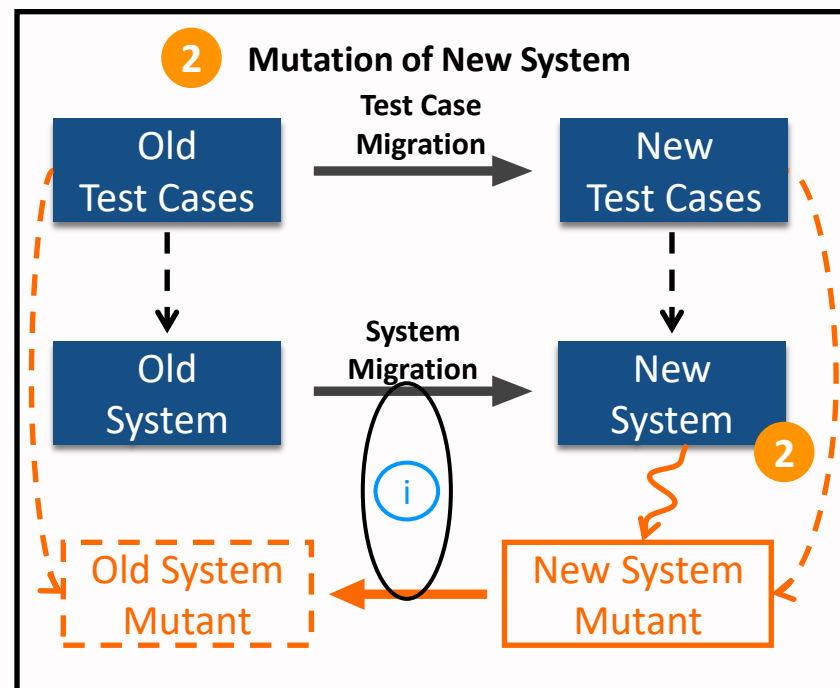
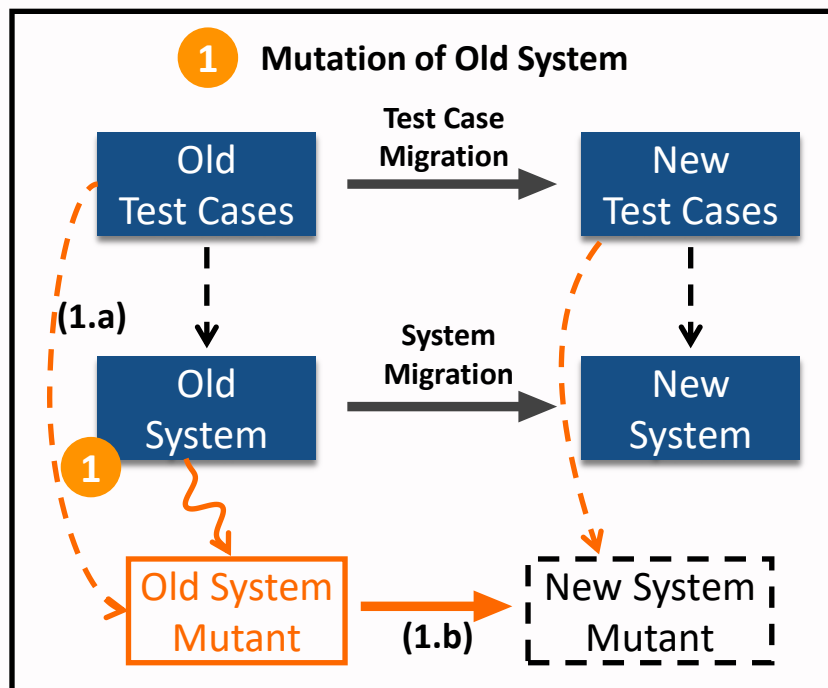
System Mutation



Test Case Mutation



Mutation Analysis Scenarios



Artefacts		Activities	
	Mutant (System or Test Case)		Mutant generation
	Reverse engineered Mutant		Mutant migration
	Migrated Mutant		Reverse engineering of Mutant
			Test execution against system mutant
			Test execution against system

Mutation Analysis Scenarios

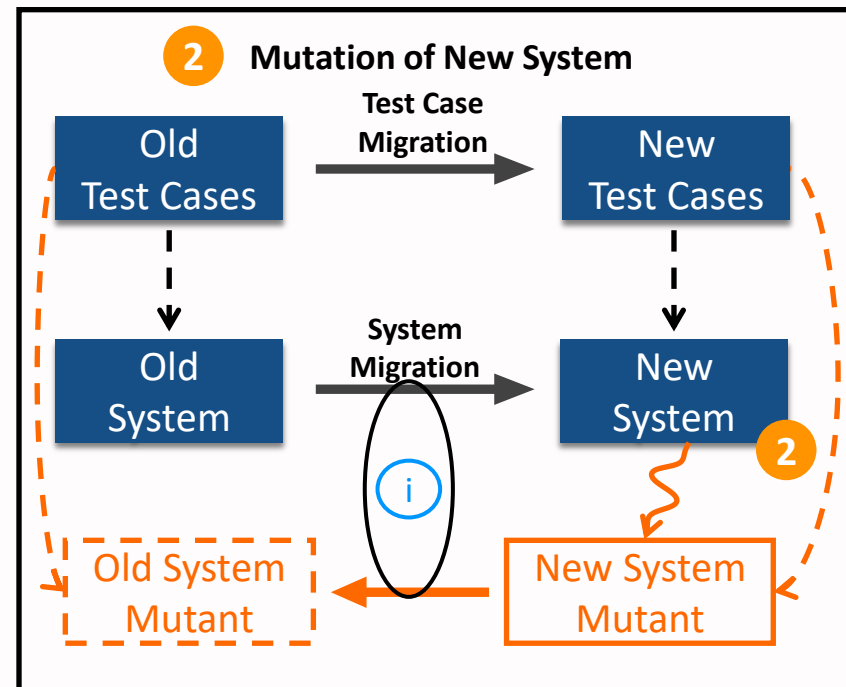
Assumptions

Suitable mutation framework exists
Mutant reverse engineering is possible

Indications

```

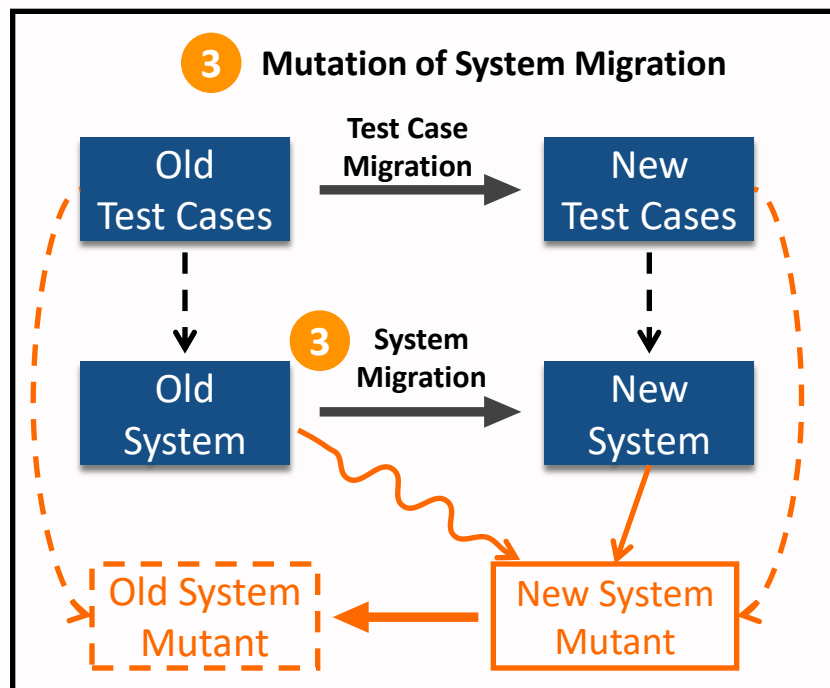
if migrated system mutant is killed then
  if old system mutant is killed then
    Expected case
  else
    if old system mutant is equivalent then
      No indication
    else
      Scenario 1a should be revisited
else //migrated system mutant NOT killed
  if old system mutant is killed then
    At least one migrated test case
    is a false negative
  else
    if old system mutant is equivalent then
      No indication
    else
      Scenario 1a should be revisited
  
```











Artefacts	
	Mutant (System or Test Case)
	Reverse engineered Mutant
	Migrated Mutant

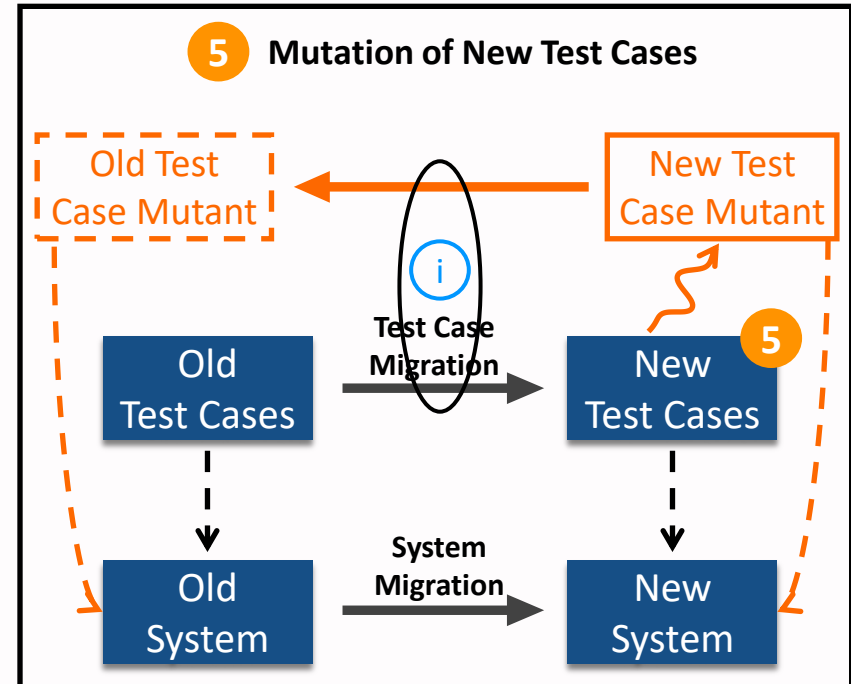
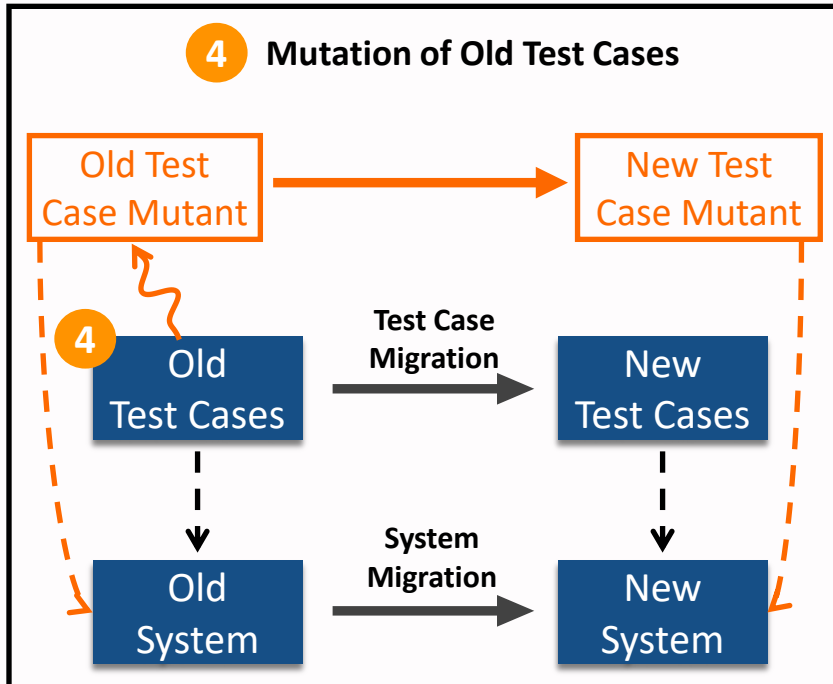
Activities	
	Mutant generation
	Mutant migration
	Reverse engineering of Mutant
	Test execution against system mutant
	Test execution against system

Mutation Analysis Scenarios



Artefacts		Activities	
	Mutant (System or Test Case)		Mutant generation
	Reverse engineered Mutant		Mutant migration
	Migrated Mutant		Reverse engineering of Mutant
			Test execution against system mutant
			Test execution against system

Mutation Analysis Scenarios



Artefacts		Activities	
	Mutant (System or Test Case)		Mutant generation
	Reverse engineered Mutant		Mutant migration
	Migrated Mutant		Reverse engineering of Mutant
			Test execution against system mutant
			Test execution against system

Mutation Analysis Scenarios

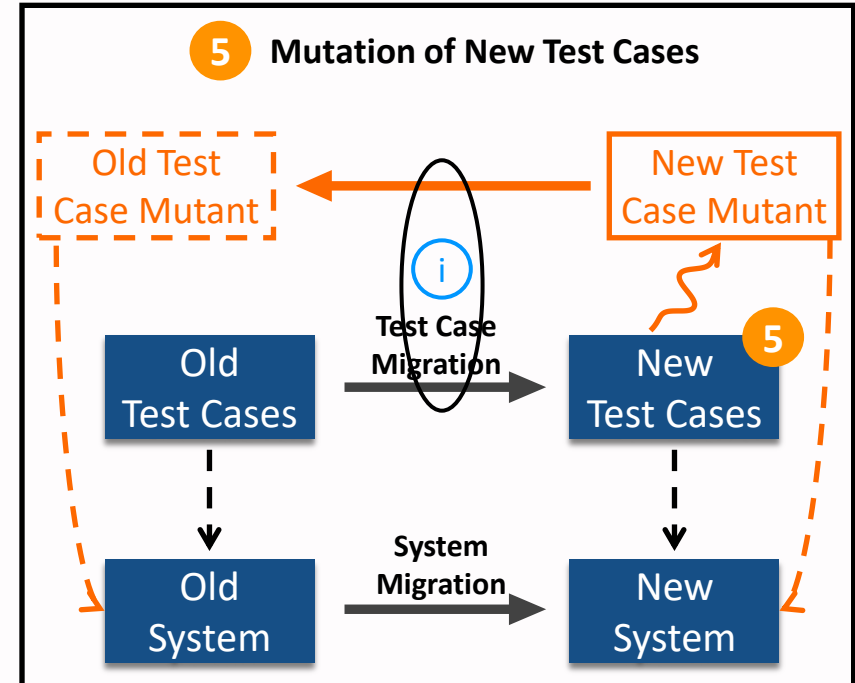
Assumptions

Suitable mutation framework exists
Mutant reverse engineering is possible

Indications

```

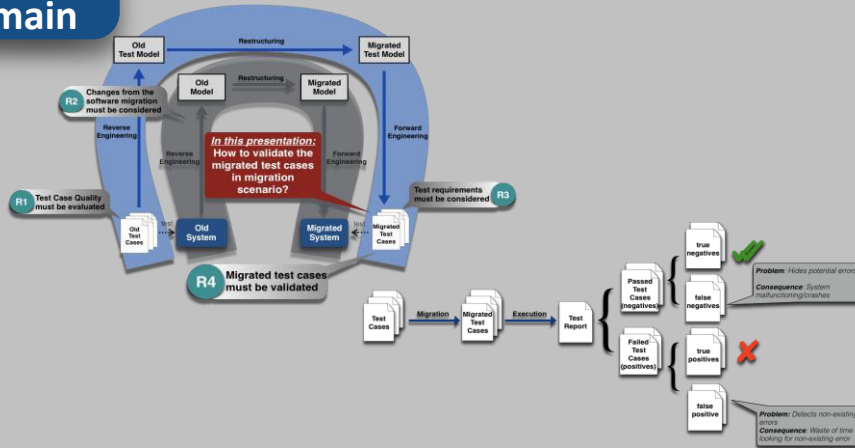
if old test case mutant fails then
    Expected case
else
    if old test case mutant is equivalent then
        No indication
    else
        Bad smell for test case migration
    
```



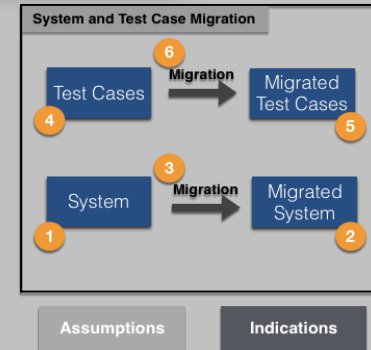
Artefacts	
	Mutant (System or Test Case)
	Reverse engineered Mutant
	Migrated Mutant

Activities	
	Mutant generation
	Mutant migration
	Reverse engineering of Mutant
	Test execution against system mutant
	Test execution against system

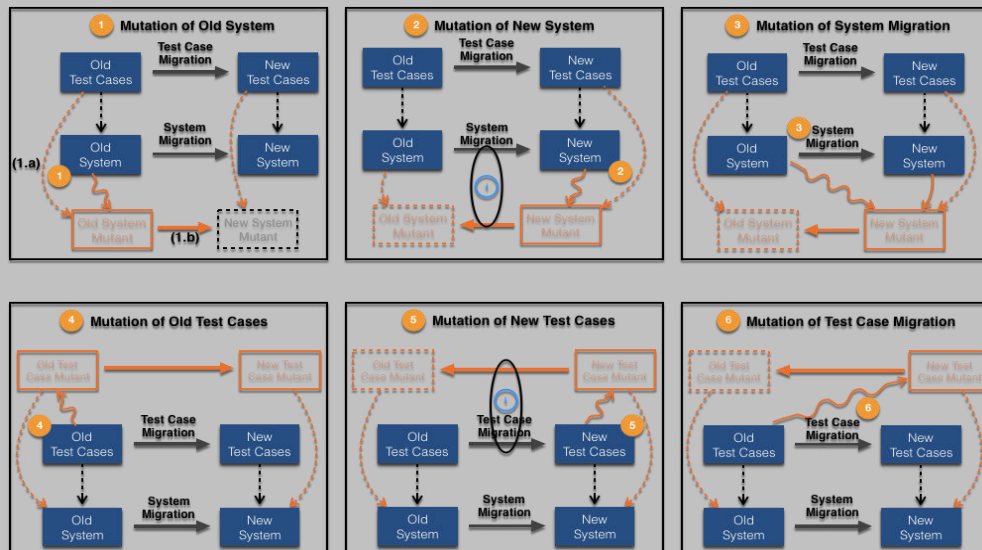
Problem Domain



Solution Idea



- 1 Mutation of Old System
- 2 Mutation of New System
- 3 Mutation of System Migration
- 4 Mutation of New Test Cases
- 5 Mutation of Old Test Cases
- 6 Mutation of Test Case Migration



**Comprehensive
discussion on the
application**

Thank you for your attention



Software Innovation Campus

Paderborn University

Fürstenallee 11

33102 Paderborn

Ivan Jovanovikj

Tel.: (05251) 60-6841

ivan.jovanovikj@sicp.uni-paderborn.de

<https://www.sicp.de/>

**User Conference on
Advanced Automated Testing**