Model-Driven Test Generation with VelociRator
Abstract

Concept, code, and tests tend to drift apart in larger or longer running projects. A lot of time and money is wasted to keep them aligned and yet to a bad degree. With agile working models and higher frequency of deployment cycles this gets even worse.

Two main reasons are identified:

a. hands-on working style with insufficiently connected tools

b. missing refactoring capability for concepts and tests (compared to code where we have learned to love powerful IDEs)
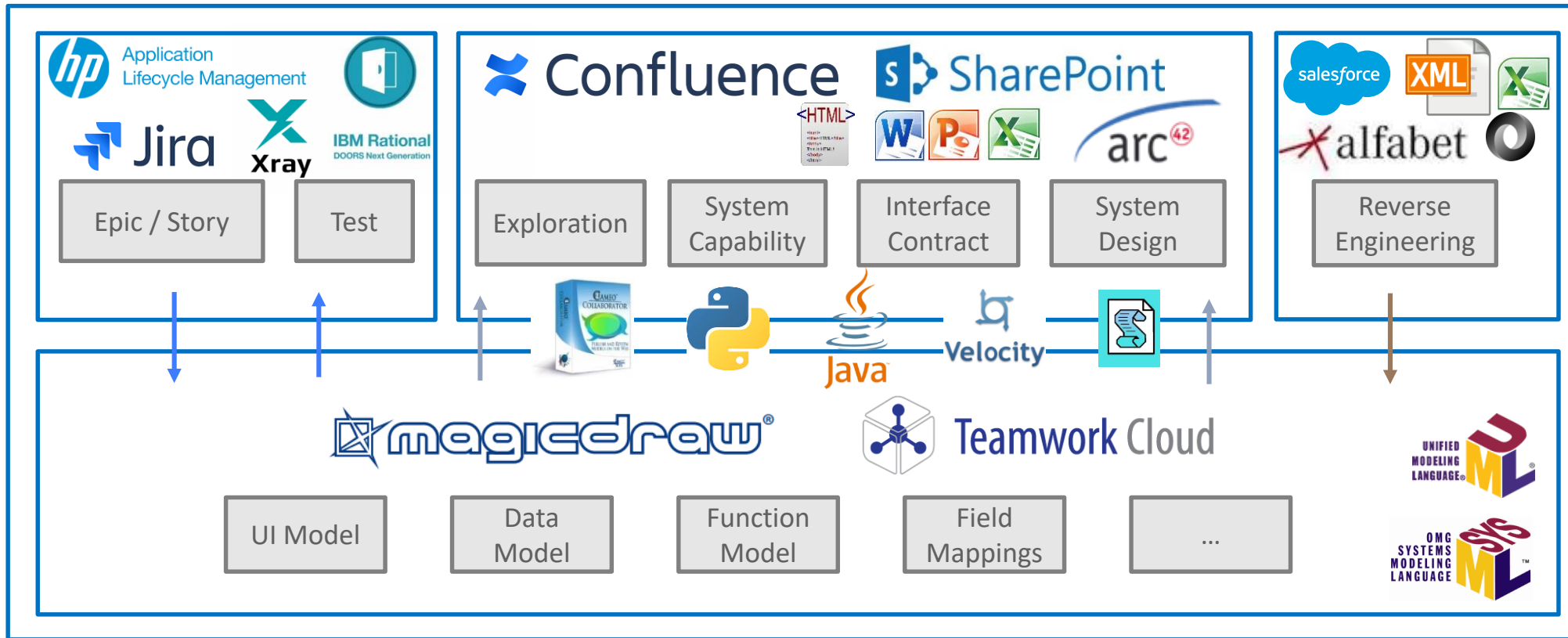
In order to keep concept and code aligned I have presented model-driven system documentation in 2018.

Today I will present model-driven test generation extending that successful alignment even further.
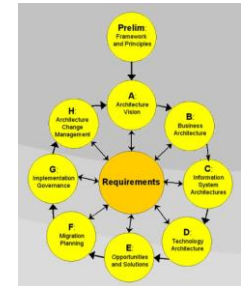
After having motivated model-driven test generation, I will present a little journey that many of us work with, probably on a daily basis. I will start with an epic, manually created by some Product Owner in a well known tracking tool named Jira. Via interface this epic is pushed to a system model in MagicDraw. There, a business analyst describes the impact of the epic on already existing use cases. Then, VelociRator, the model-driven test generator, produces tests from modeled behavior. These tests are finally pushed back to Jira and linked to the initial epic closing the loop with a neat test coverage on the epic.

An outlook on model-driven test generation for automation will give a hint on how to combine test generation and test automation. Finally, I will conclude with why model-driven test generation saved our back during COVID-19.
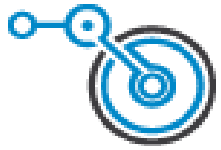
Tool Universe

PO   BIZ   DEV   OPS

TOGAF®

ZACHMAN

EAP

HP Application Lifecycle Management

Jira   Xray   IBM Rational DOORS Next Generation

| Epic / Story | Test |

Confluence   SharePoint

arc42

| Exploration | System Capability | Interface Contract | System Design |

salesforce   XML   Excel

alfabet

| Reverse Engineering |

Cameo Collaborator   Python   Java   Velocity

magicdraw®   Teamwork Cloud

UNIFIED MODELING LANGUAGE® UML

OMG SYSTEMS MODELING LANGUAGE SysML

| UI Model | Data Model | Function Model | Field Mappings | ... |

NINKIT-CONSULTING
architecture advisory services

About Me

http://www.ninkit.com/about.html
https://www.gulp.de/gulp2/home/profil/unink
https://www.xing.com/profile/Udo_Nink
https://wwww.linkedin.com/in/udonink

Dr.-Ing. Udo Nink
Diplom-Informatiker
Landsberger Str. 455
81241 München

Telefon: 0157 7162 7478
Email: udo.nink@unitc.de
Web: www.ninkit.com

„The profession of the architect is an adventurous task.

It's a job in the twilight of art and science,

on the rim of invention and memory,

between spirit for modernization and respect for tradition."

Renzo Piano, Italian architect.