From NO requirements to MAXIMUM Coverage
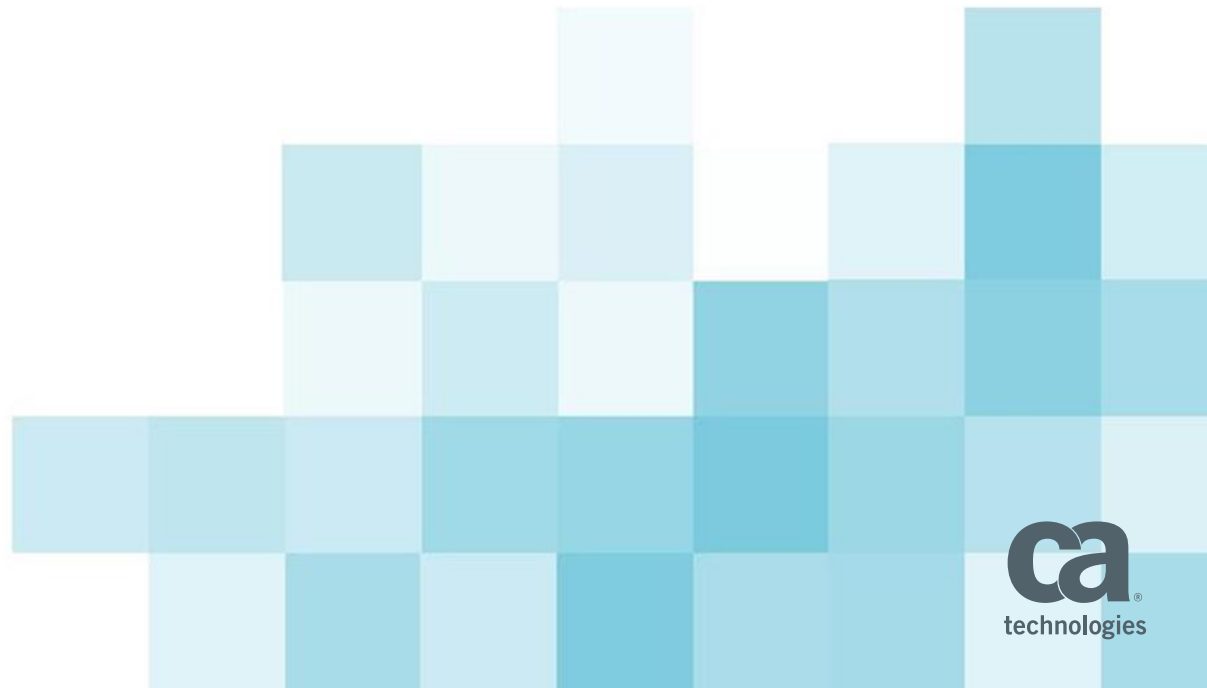
Workshop by Arno van de Velde
Solution Strategist, CA Technologies

13-07-2016

ca technologies

Workshop Agenda

INTRODUCTION AND SETUP	**15 MINUTES**
ACTIVITY 1 (...)	**20 MINUTES**
ACTIVITY 2 (...)	**20 MINUTES**
SHORT BREAK	**10 MINUTES**
ACTIVITY 3 (...)	**90 MINUTES**
DEMO & WRAP UP	**25 MINUTES**

ca technologies

Introduction & Setup

- Short Introduction
  - Name
  - Organisation
  - Position/Role
  - Testing focus

- What are your expectations for this workshop?

ca technologies

Practical Information

- NOP Commerce Website

- Can be reached on http://212.67.209.24:8010

- Check if website is reachable

- Teams as mixed as possible (2-4 people)

- Installation of ARD Software throughout the workshop (usb sticks with install file and license key).

- License can be used after the workshop

Activity 1: Test Website

- Register users on website

- Teams

- Every team is responsible for 1 part of the administrator section and the associated front end part.

- After x minute teams explain to each other:
  - Coverage
  - Amount of tests
  - Results
  - How well they've done.

Results Activity 1

- No requirements or formal documentation → hard job, time consuming, not impossible

- Coverage?

- Visibility of results

- Predictability and expected results

- ….

Quality? Even a basic system will have more possible paths through it than any one person can think up

2145 Paths

The number of possible tests grows exponentially with every added decision
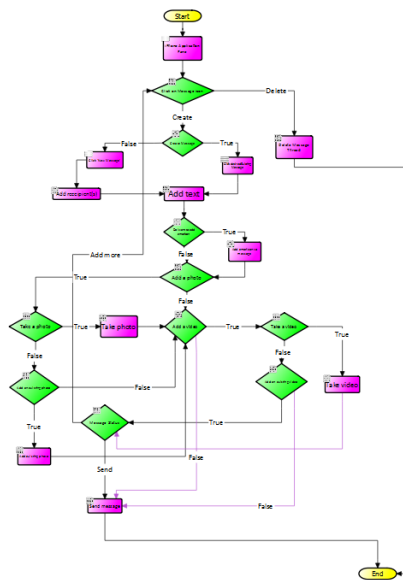
32 nodes + 62 edges = *1,073,741,824* possible routes

- Must now reckon with composite applications, with numerous moving paths

- Have distributed systems, middle way services, the internet

- As well as APIs, third party services and the Internet of Things

- **Manually and unsystematically deriving tests will cover just a fraction of this functionality**

ca technologies

Activity 2: Requirements gathering

- Each team will be given requirements (verbal)

- Write them down / document them

- Pass the written requirements to the team on your left

- Write at least 2 complete testcases/scenario's based on these written requirements. Include test data, expected result etc.

- Pass these testcases to the team on your left

- Execute tests, document them well, be prepared to give an overview on how well the team has done it.
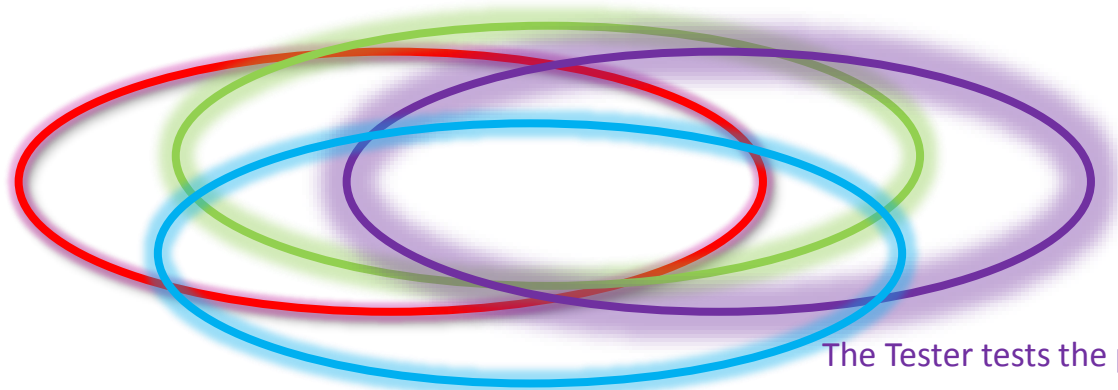
Activity 2 results

- Ambiguous requirements

- Time consuming methods of going from written requirements to testcases

- What happens if requirements change

SHORT BREAK

The Problem:

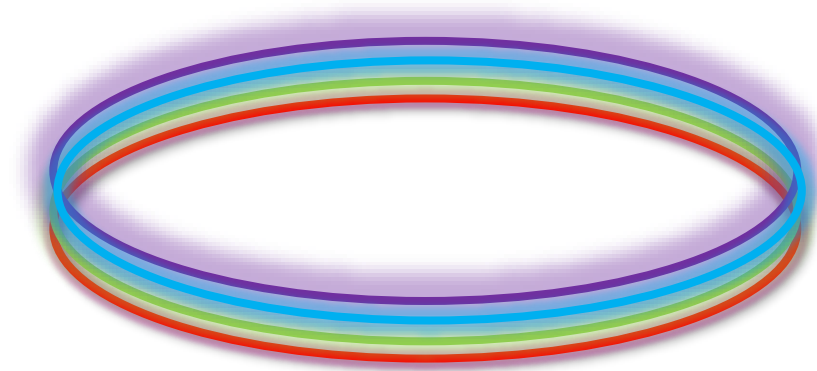A lack of clarity and vision during development

User → Business Analyst → Programmer → Tester

The Tester tests the program

The User Knows what they want   The Programmer writes the code

The further the visions diverge The analyst specifies what They greater the problem

ca
technologies

The Solution:

Clarity and Vision during development

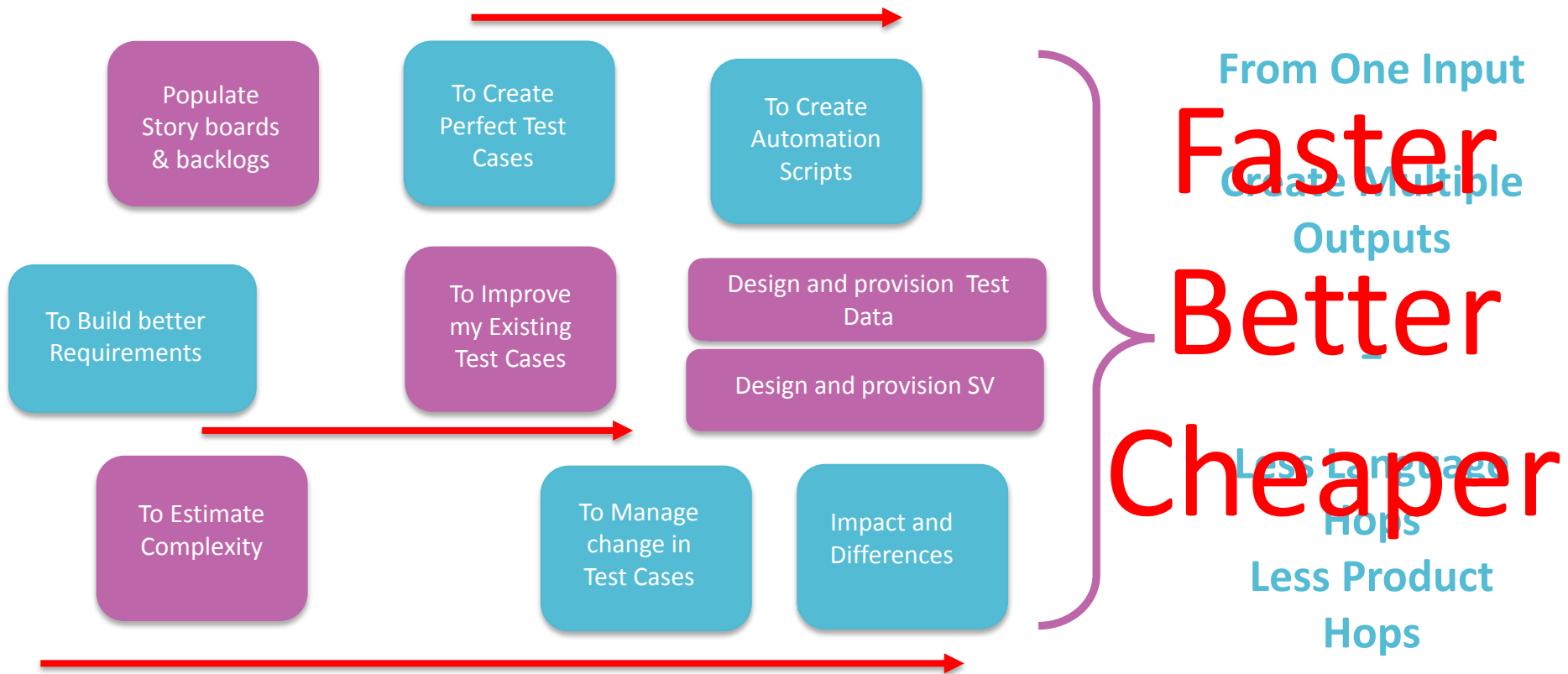| User		Business Analyst		Programmer		Tester

The closer the vision means the user gets a quality product

There are less bugs and the product is delivered faster

ca
technologies

CA-Agile Requirements Designer – CA TDM – CA SV

Populate Story boards & backlogs

To Create Perfect Test Cases

To Create Automation Scripts

To Build better Requirements

To Improve my Existing Test Cases

Design and provision Test Data

Design and provision SV

To Estimate Complexity

To Manage change in Test Cases

Impact and Differences

From One Input

Create Multiple Outputs

Less Language Hops

Less Product Hops

Faster Better Cheaper

ca technologies

All testing is at least implicitly based on models

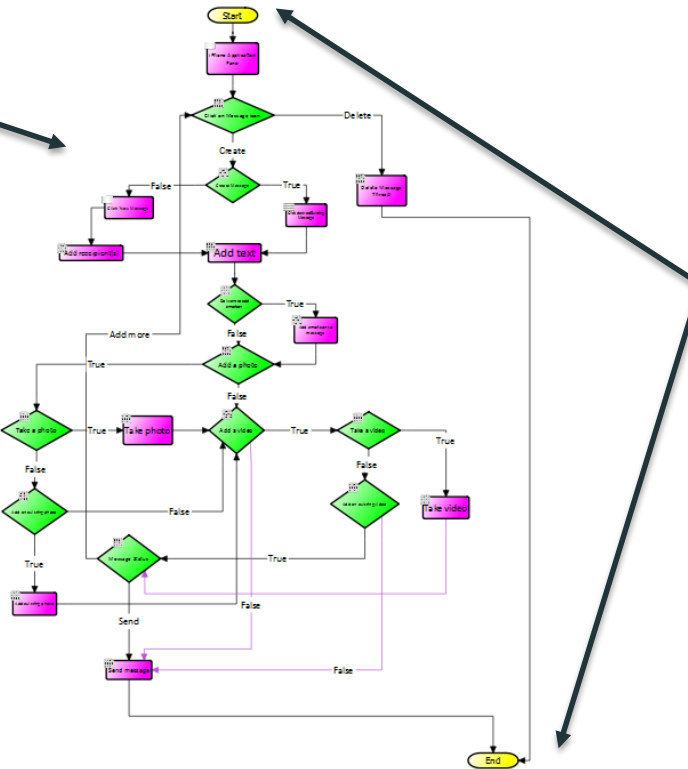- E.g. thinking up negative tests or connecting up Behavior-Driven Requirements or user stories

Build model

| Explore (sources) | Judgement | Create Model | | Test Suite |

Knowledge
Requirements
Source code

Models not adequate

Mentally
Written down
Explicit
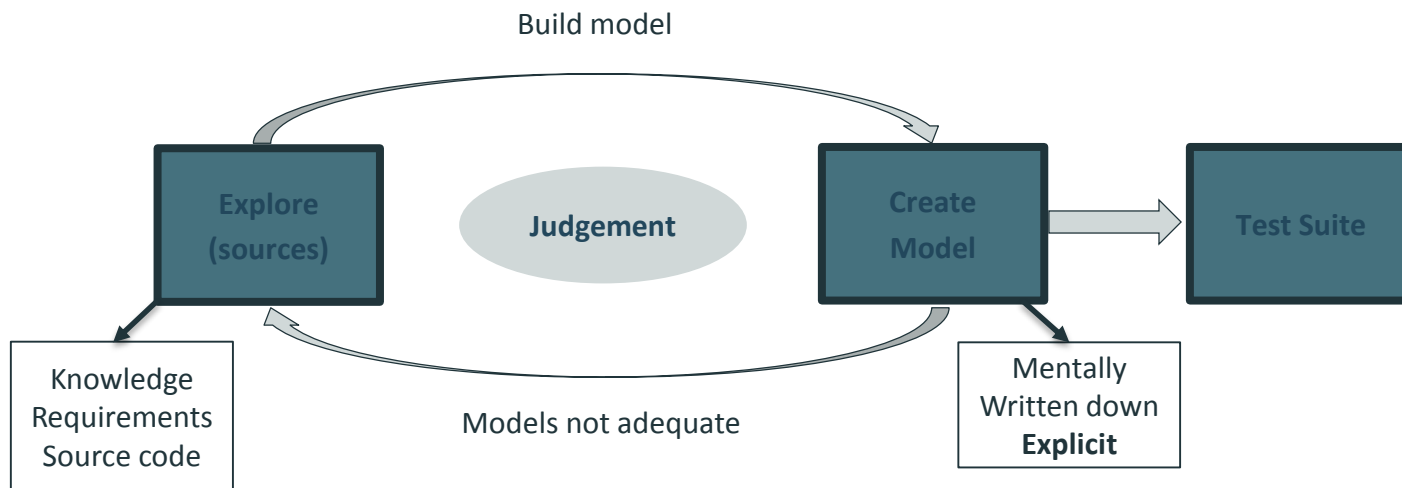
A new model for Testing, EuroStar - Paul Gerrard (2015)

ca
technologies

# Model example

Models can be represented as flow charts.

Start and end points of the system.

Each process defines the state of the system with transitions to other states through actions.

Extracting test cases

- Each path through the model represents a test case

- Because the flowchart is mathematically precise, paths can be generated automatically using algorithms

2145 Paths
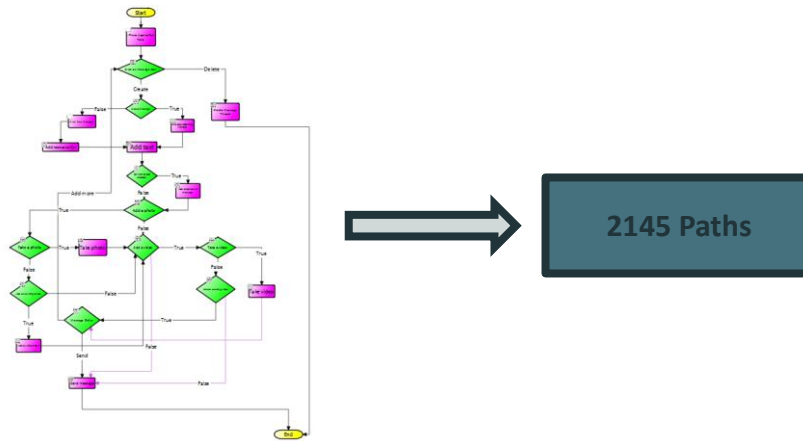
Optimization / Coverage

- Coverage: the measure of how much functionality is being covered by a set of test cases

- Combinatorics will condense the paths down while retaining coverage

Edge Pairs	→	19 Paths
In Out Edges	→	9 Paths
Edges	→	5 Paths
Nodes	→	3 Paths

ca
technologies
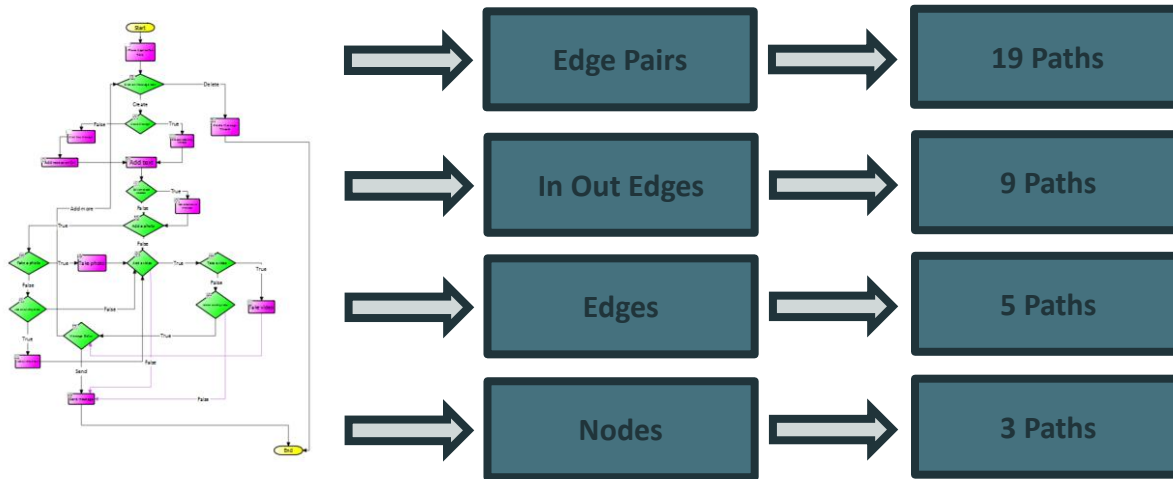
Activity 3: Use ARD

- Open up example.

- Use your own written requirements / testcases to build up a model

- Use document provided to import 'Discounts' and model that

ca
technologies

Demo and Wrap Up

- Demo of "Reactive Automation"

- CA Agile Requirements Designer Youtube (LINK)

- Testnet October event with focus on Automation with ARD by Marcel Mersie (Bartosz) supported by CA.

- For questions contact me at Arno.vandevelde@ca.com

technologies

Arno van de Velde
Solution Strategist
Arno.vandevelde@ca.com

@CAinc

Slideshare.net/cainc

nl.linkedin.com/in/arnovandevelde

CA.com