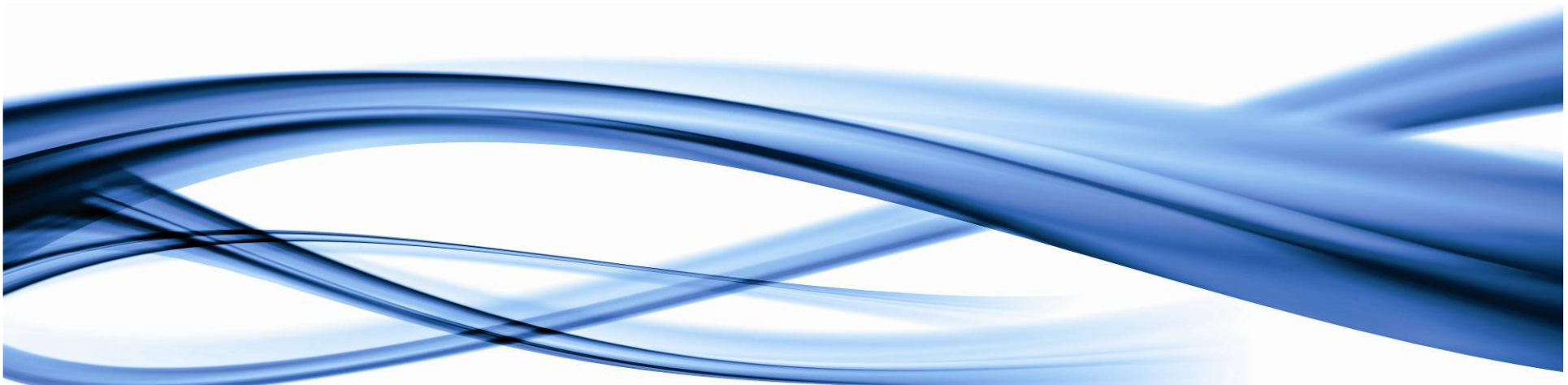


**ANZTB  
Test 2011 Conference  
-  
Geek's Corner Session 2**



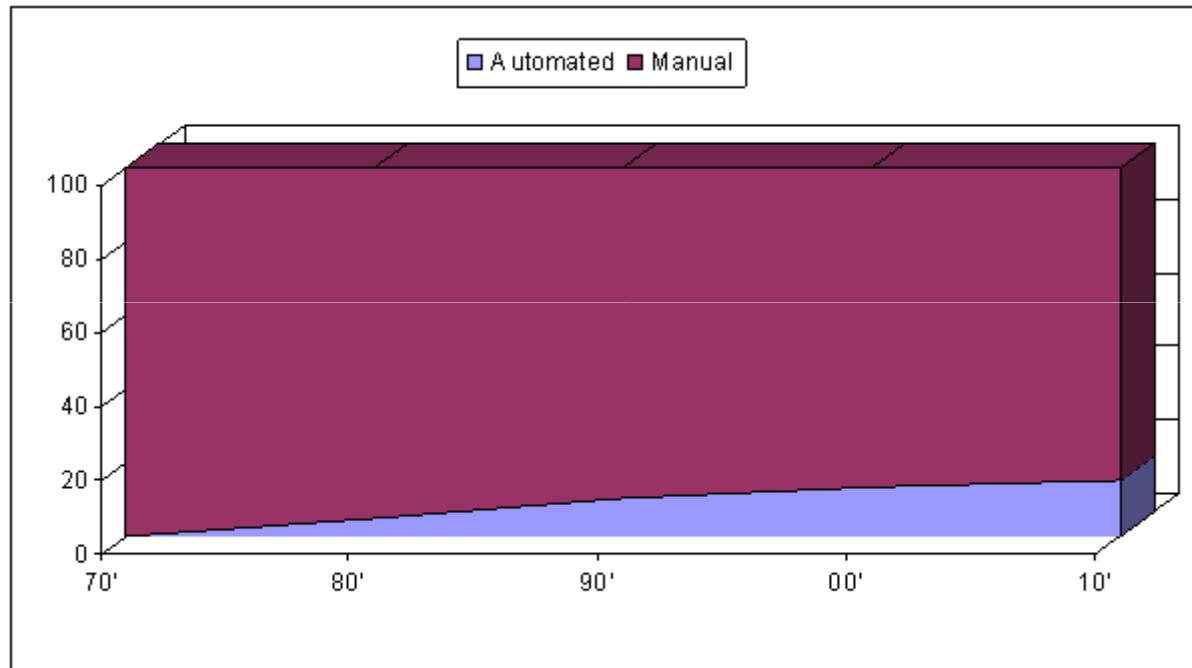
# Testing Beyond Today



**Gartner™ believes “that  
model-driven testing approaches  
will become the dominant format  
in five years”**

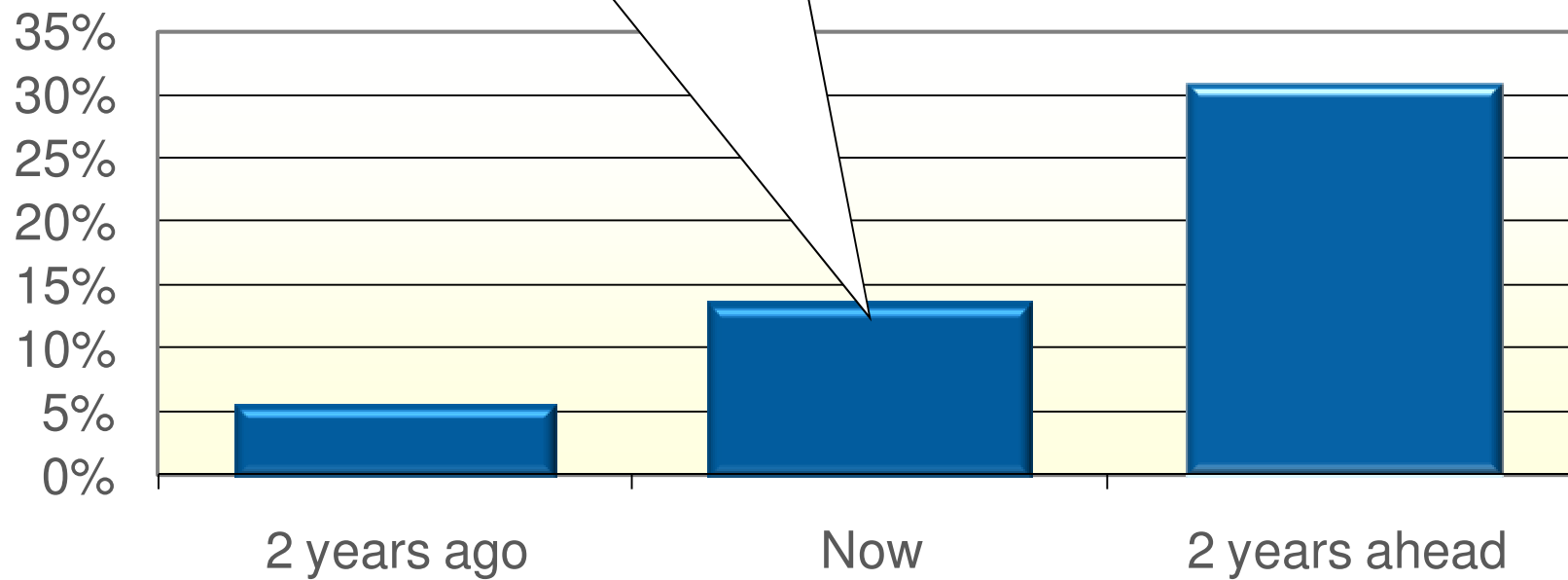
Source: Gartner™ Cool Vendors in Application Development, 2010.

**Test automation coverage below 25% even after 30 years of software testing.**



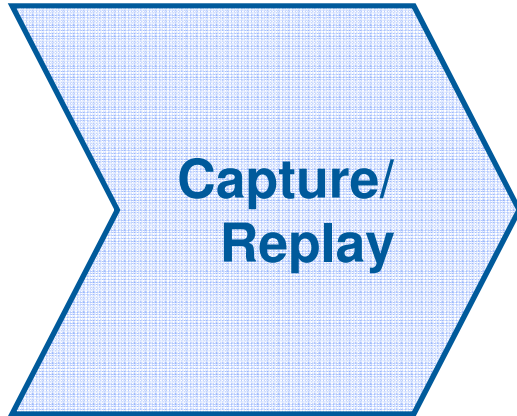
**Why? Do we need a different approach?**

**AU 13%**



Source: K.J. Ross & Associates, Industry Benchmark 2009/2010

1. Generation



2. Generation

3. Generation



Too fragile

## Why fragile?

Checkbox	Name	Reg Number	Invoice
<input type="checkbox"/>	John Smith	NSW-1234	90035560
<input type="checkbox"/>	Jim Courier	NSW-2345	90035561
<input checked="" type="checkbox"/>	John Cook	VIC-3456	90035562
<input type="checkbox"/>	Tim Reborn	VIC-4567	90035563
<input type="checkbox"/>	Dale Howard	WA-5678	90035564
<input type="checkbox"/>	Frank Rose	WA-6789	90035565



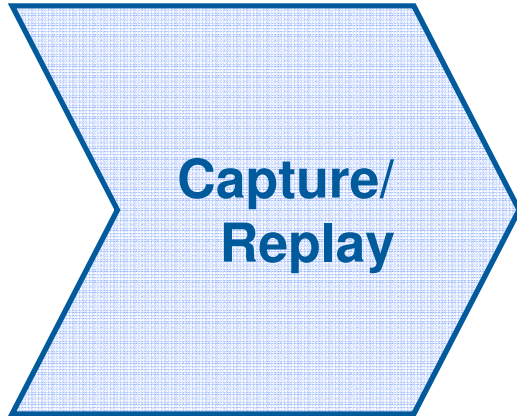
```
CheckBox Click,  
"/usr/cnt1CONTAINER/  
shellcont/  
shell[2]/chbx[1,3]"
```

Business instruction: "Select invoice for John Cook".

Code: click column #1  
row #3!

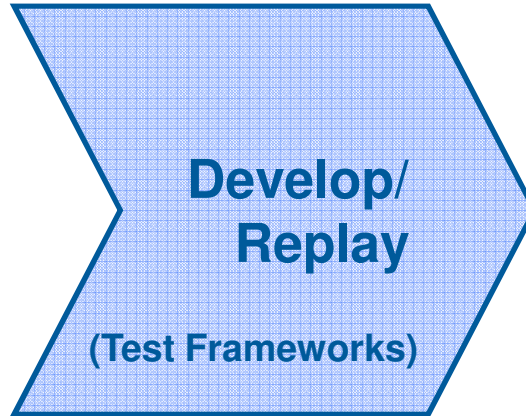
**Breaks when data or layout changes!**

# 1. Generation



Too fragile

# 2. Generation



Too costly

# 3. Generation

## Why costly?

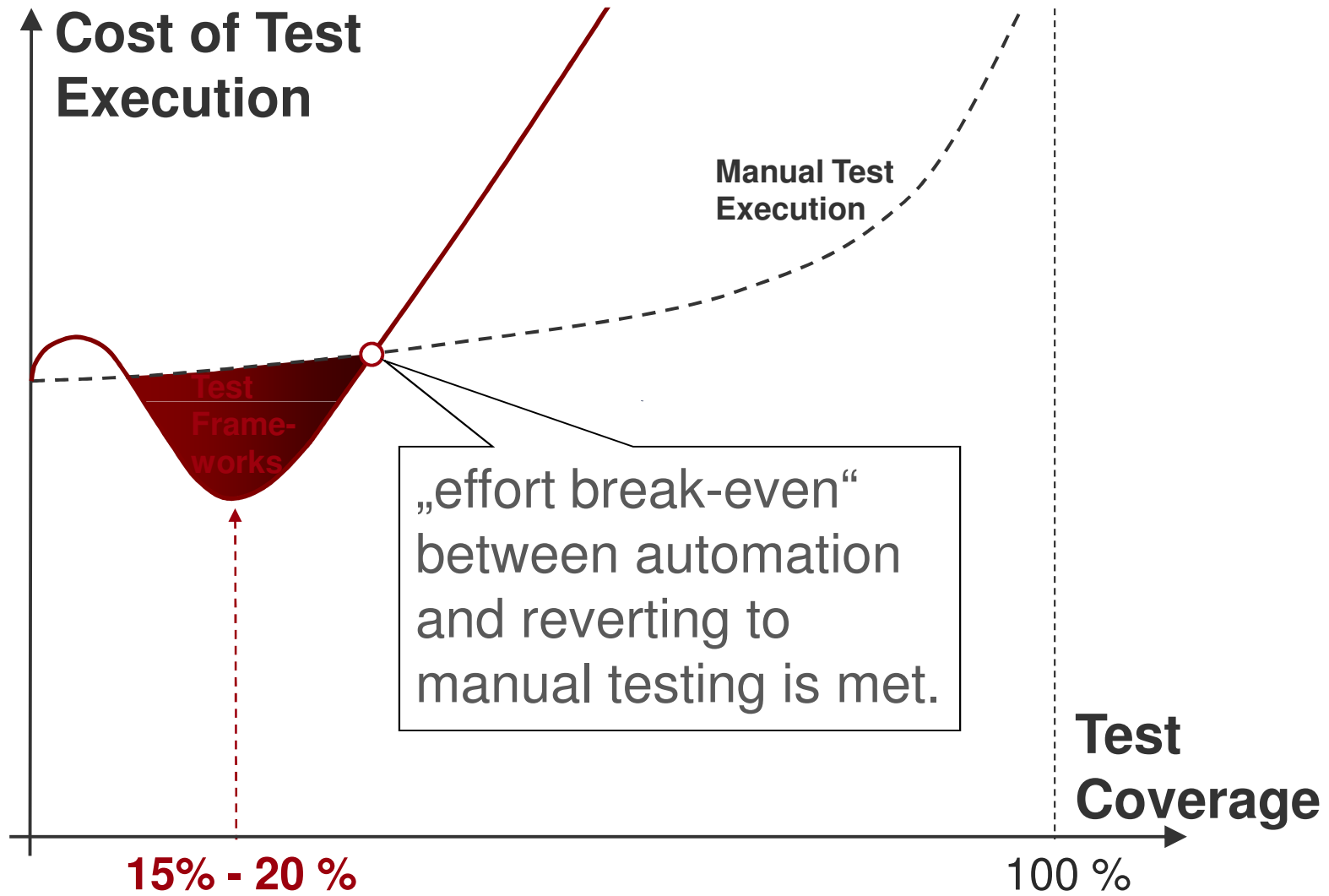


- Generalise/modularise scripts
- Extract test data
- Requires programming

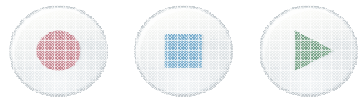
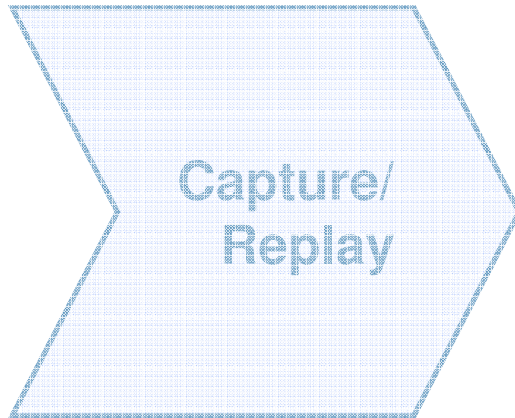
+ **Addresses fragility**, but...

- Testing becomes a **parallel software development project**

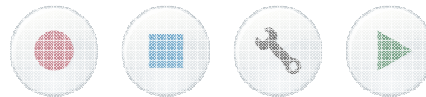
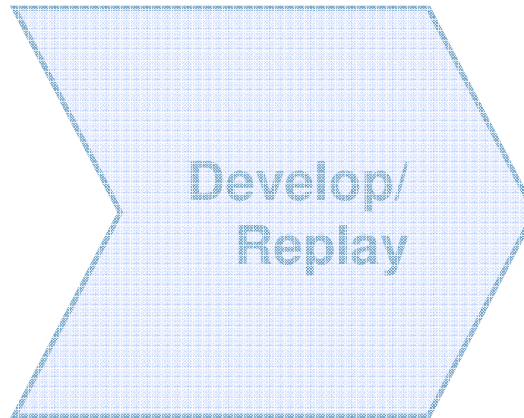
- **Too costly**



1. Generation



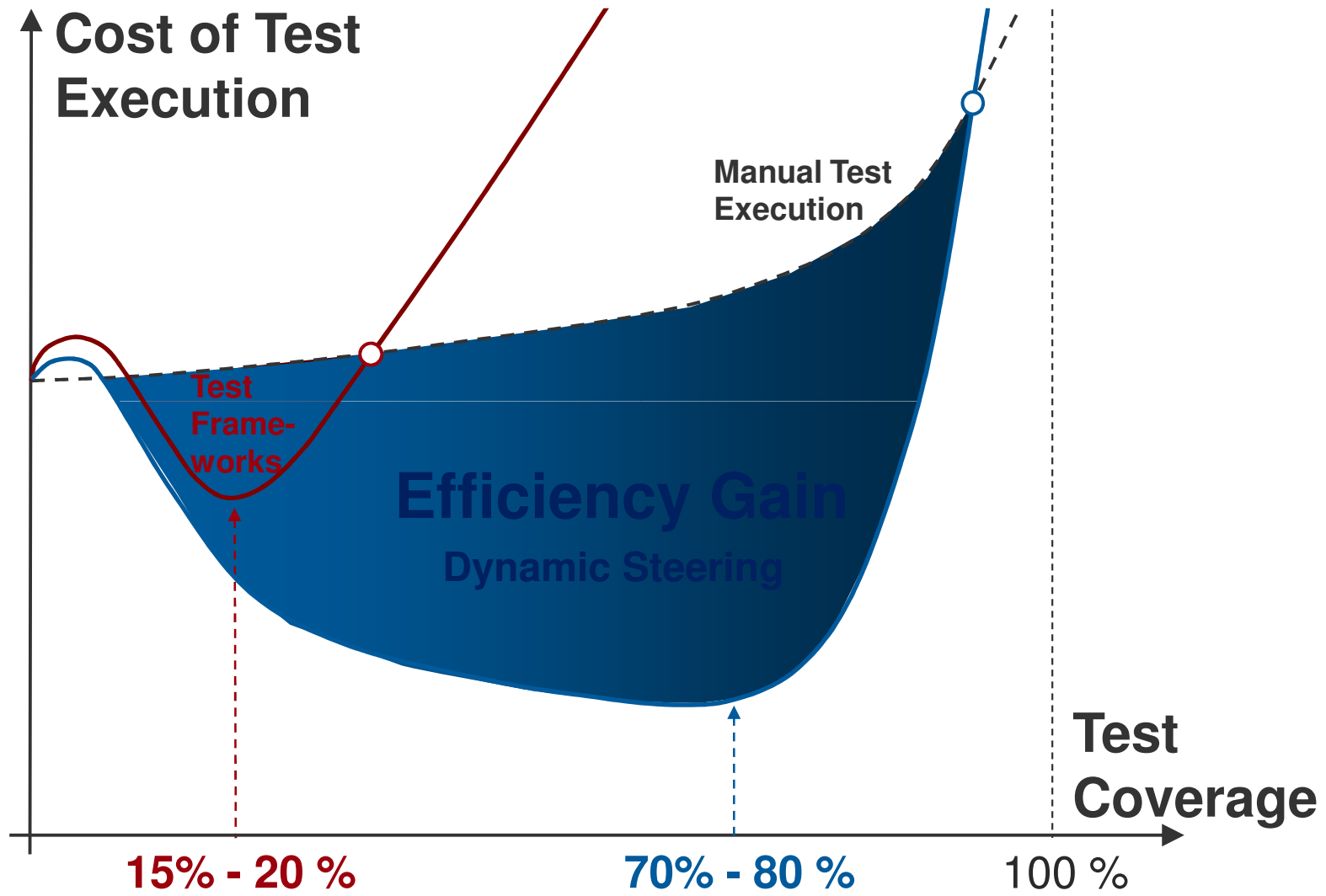
2. Generation



3. Generation



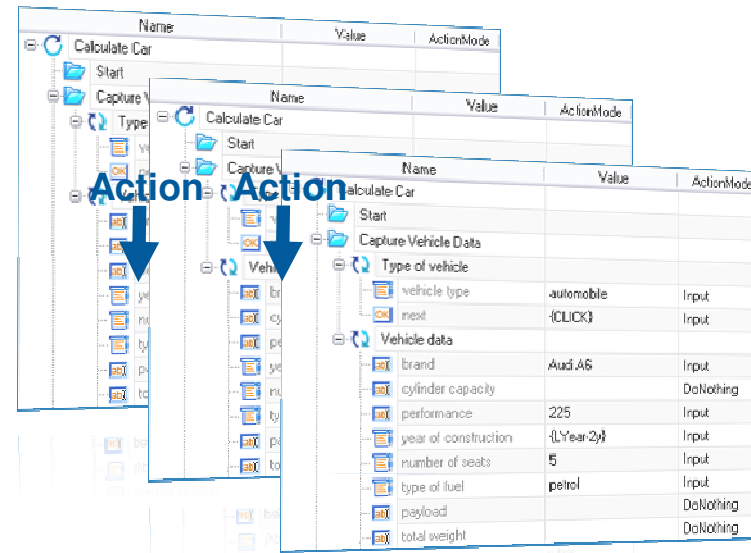
**Stable tests** only become reality if the **test automation** is being generated **dynamically** from the **business perspective** of an application at runtime and not vice versa!





Dynamic Steering: How does it work?

# Cognitive Test Cases



## System Under Test (SUT)

<b>type of vehicle</b>	automobile
brand*	<input type="text" value="BMW"/>
performance*	<input type="text" value="150 KW "/>
year of construction	<input type="text" value="2010"/>



**Business**  


---

**Technology**

TC 2 Insurance Calculation Automobile			
Type of vehicle			
Type of Vehicle	automobile		Input
Next	{CLICK}		Input
Vehicle data			
Brand	Fiat Punto		Input
Performance	44		Input
Year of Construction	2002		Input
Number of Seats	5		Input
Type of Fuel	petrol		Input
Original Price	15000		Input
License Number	TE-101 ST		Input
Way of Use	private		Input
Mileage per Year	19000		Input
Next	{CLICK}		Input
Insurant			
Product choice			
Check Premium tax net			
Offer number			DoNothing
AccountList			Verify
Action			
Column	Premium net		
Row	Premium tax		
Value	132,00		
Restart			DoNothing
Check Premium tax gross			

- **Cognitive:** Plain English - not scripts!
- Requires only **business SME** skills
- **Highly stable** from a technical and business point of view
- **Independent from technologies:** HTML, Java, .NET, SAP, Manual execution, ...

## Manual (PROSE)

**Testfall KNU01**

Eintrag als User „alpha“, dieser hat Passwort „xyz12“  
Datum der Aktion auf den ersten Montag nach dem Tagesdatum setzen.  
Weiter klicken.

Nach der Aktion prüfen, ob die erwarteten Bildschirme sichtbar werden.

<Hier wird die nächste Aktion beschrieben>

<Hier wird die nächste Aktion beschrieben>

<Hier wird die nächste Aktion beschrieben>

<Hier wird die nächste Aktion beschrieben>

<Hier wird die nächste Aktion beschrieben>



...and...

## Automated (CODE)

```
* check all par and aggregate actionText
With pa
  If useDefinition Then
    .useDefinition
    useType = LCase(.definition.ObjectNameName)
    If useType = OR_TABLE Then
      * create a new table-ref to fetch actionText
      Set paCtrl = paramAction(TABLE_IDENTIFIER)
      If paCtrl Is Nothing Then
        Set paCtrl = paramAction(TREE_IDENTIFIER)
      End If
      If paCtrl Is Nothing Then
        * try to use standard-text
      Else
        * use specific translation
        Set myTable = New cSAPTableRef
        With myTable
          .paramAction = paCtrl
          .action = paramAction(TABLE_ACTION)
          .col = paramAction(TABLE_COL)
          .col1 = paramAction(TABLE_COLUMN)
          .Format = paramAction(CELL_FORMAT)
        End With
        txt = GetFinalTranslation(.GetActionText)
      End With
      If (txt = C_NULL) Then
        * use standard action text
      End If
    End If
  End With
```

- In the **traditional world** you maintain **two sets of documentation**.

manual

automated



Vehicle Data



Name	Value	ActionMode
Calculate Car		
Start		
Capture Vehicle Data		
Type of vehicle		
vehicle type	automobile	input
next	(CLICK)	input
Vehicle data		
brand	Audi A6	input
cylinder capacity		DoNot
performance		input
year of construction	{Year-2y}	input
number of seats	5	input
type of fuel	petrol	input
payload		DoNot
total weight		DoNot
type of vehicle		input
brand		input
cylinder capacity		input
performance		input
year of construction		input
number of seats		input
type of fuel		input
payload		input
total weight		input

Name	Value	ActionMode
Calculate Car		
Start		
Capture Vehicle Data		
Type of vehicle		
vehicle type	automobile	Input
next	(CLICK)	Input
Vehicle data		
brand	Audi A6	Input
cylinder capacity		DoNothing
performance		Input
year of construction	{Year-2y}	Input
number of seats	5	Input
type of fuel	petrol	Input
payload		DoNothing
total weight		DoNothing

Specification often in MS Word, MS Excel, etc.

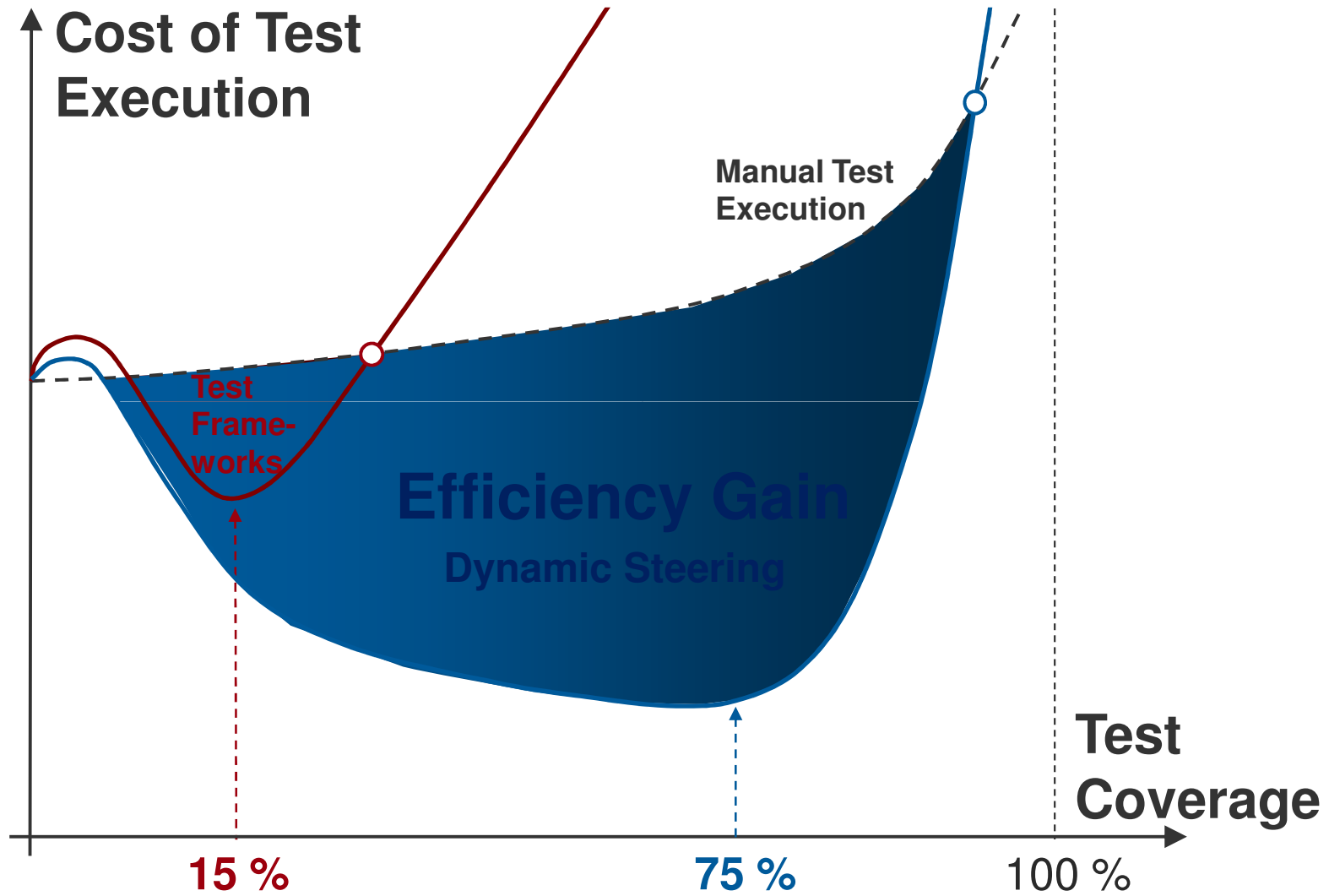


Automated test cases in form of a script (program code)



- One world: **manual** and **automated** test cases are maintained in a business language – **treated identically.**

# Dynamic Steering - Key Benefits



## Dynamic Steering - Key Benefits continued...

- **Even automated test cases are cognitive** – no scripts!  
Not just hidden but actually no scripts!
- **Only one set of test case documentation** – regardless if executed manually or automatically!
- **Technology independent** – write once, run everywhere
- **Eliminates maintenance** inherent in G1 & G2 technologies
- **Reduces cost** of software testing.

**Gartner**<sup>®</sup>



**GARTNER** calls it the

**“Paradigm shift for the software test --  
how to avoid the maintenance trap and  
elevate test automation to a business  
discipline!”**

Watch the GARTNER video at  
[www.tosca-testsuite.com](http://www.tosca-testsuite.com)