

# From Strategy to Techniques

Tim Koomen

Dutch Testing Conference 2010



From Strategy to Techniques

## The gap ... no follow-up on strategy

- ▶ Do you recognize this?
- ▶ *"I don't know how the testers make their choices what to test and how to test ..."*
- ▶ Test manager has agreed test strategy on paper, then:
  - ▶ Testers decide for themselves what to test and how thoroughly, or ...
  - ▶ Only 1 technique is used by everyone, or ...
  - ▶ No techniques at all

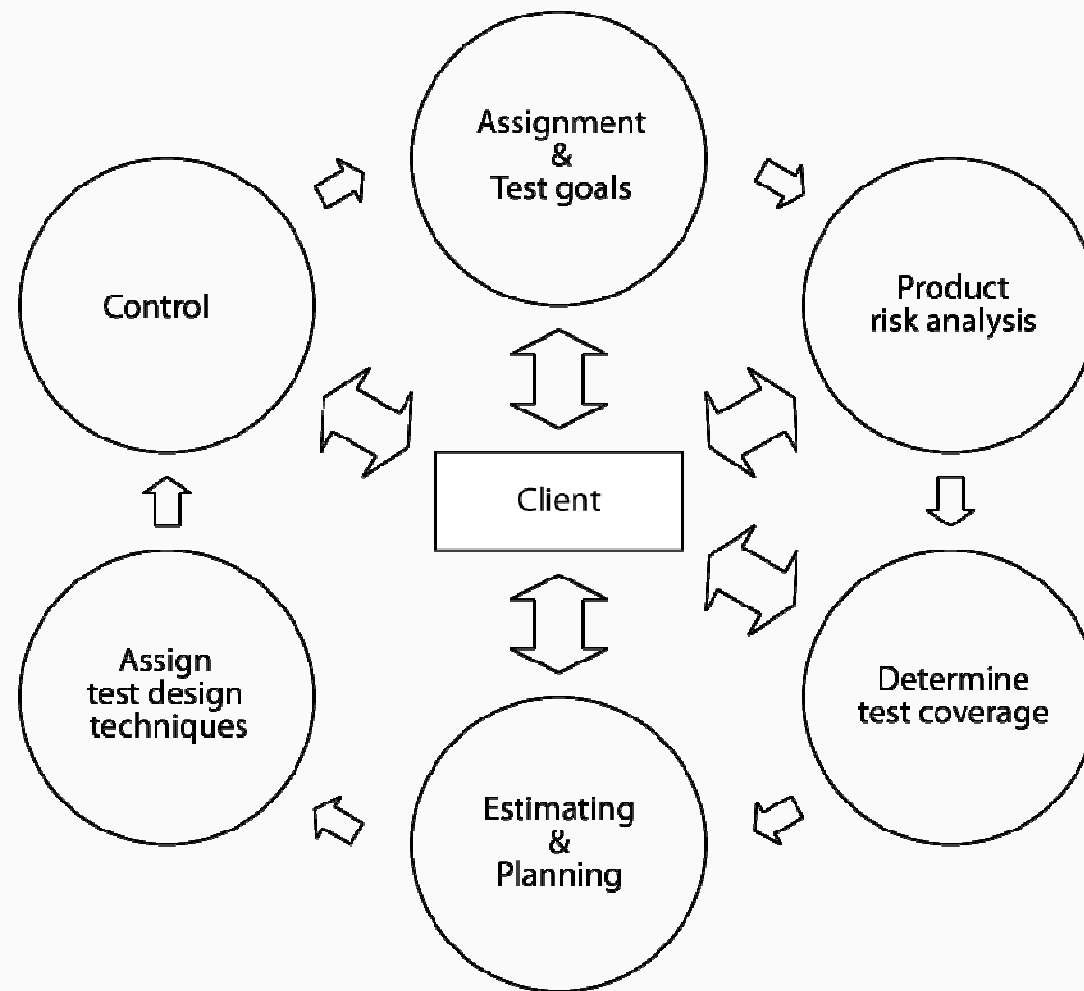
# Risk-driven testing



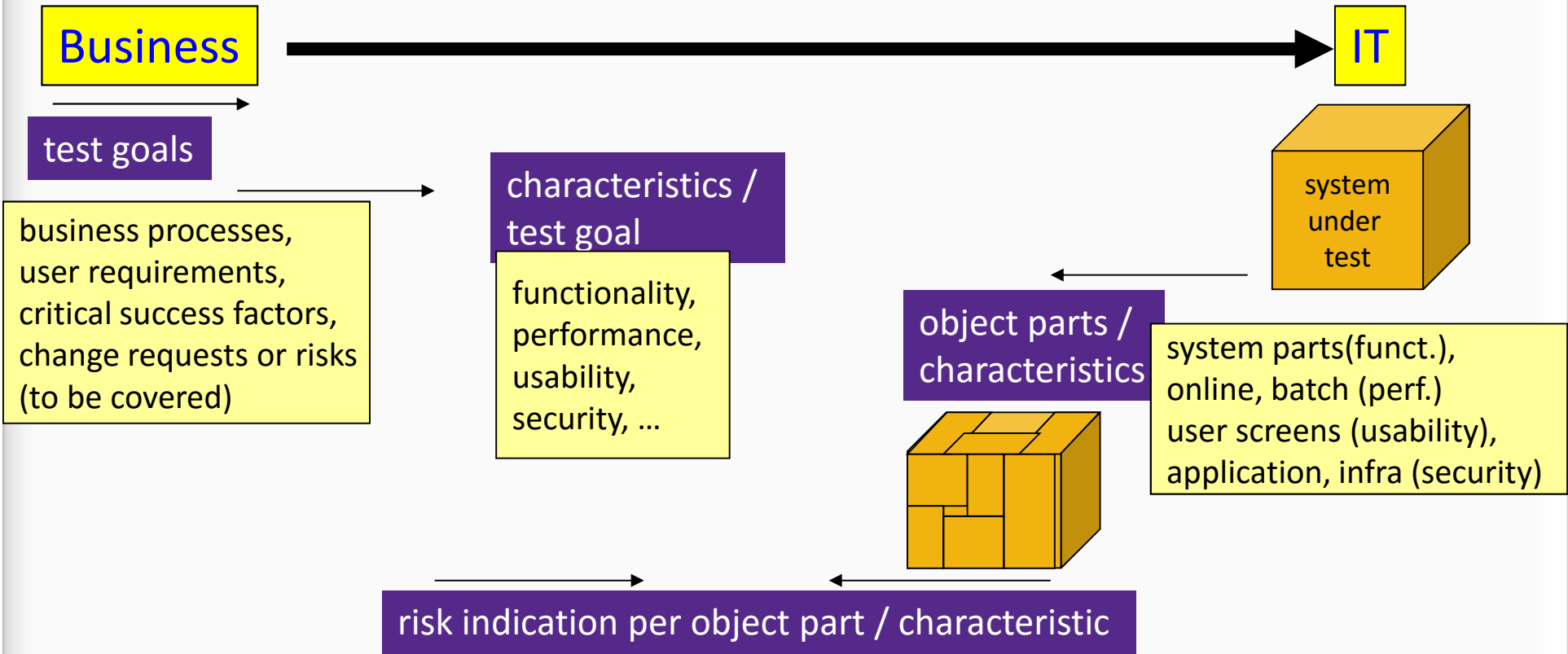
Client



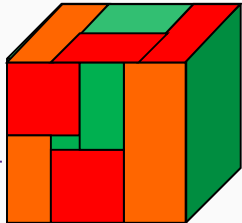
# Steps of risks-driven testing



# PRA: from business to IT



		Chance of failure		
		High	Medium	Low
Damage	High	A	B	B
	Medium	B	B	C
	Low	C	C	C



From Strategy to Techniques

# PRA overview

Characteristic / object part	PRA-Risk Class
Functionality	
- subsystem 1	B
- subsystem 2	A
- total	C
security	
- infrastructure	B
- application	B
performance	
- online	B
- batch	C
suitability	A
manageability	C

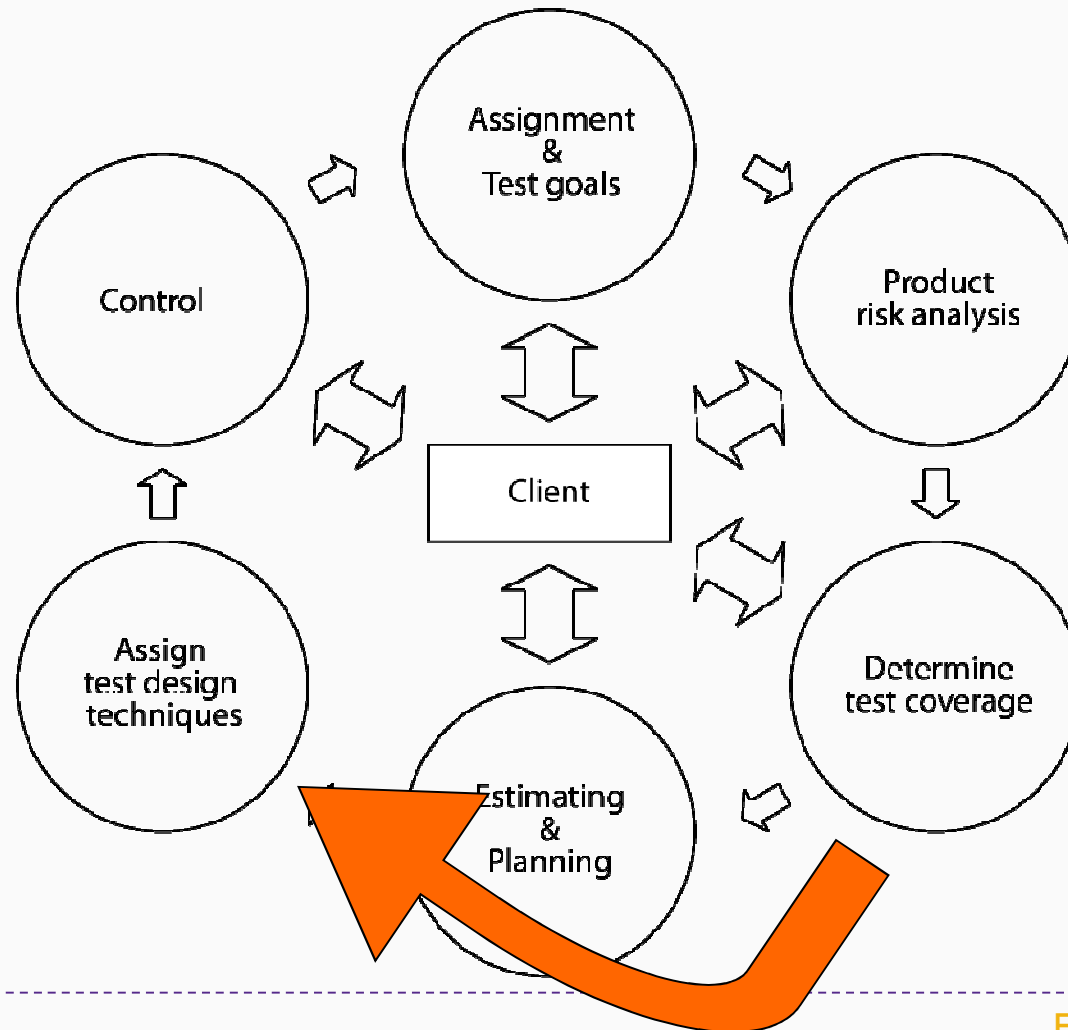
# Test strategy MasterTestPlan

Characteristic/ object part	PRA -RC	Review	DT	ST	UAT	PAT
functionality						
- subsys 1	B	●	●●	●●	●	
- subsys 2	A	●	●	●●●	●●	
- total	C			●	●●	
security						
- infrastructure	B	●				●●
- application	B	●		●	●●	
performance						
- online	B			I		●●
- batch	C				I	S
suitability	A	●			●●●	
manageability	C					●

PRA

Test strategy

# The gap, how to continue?

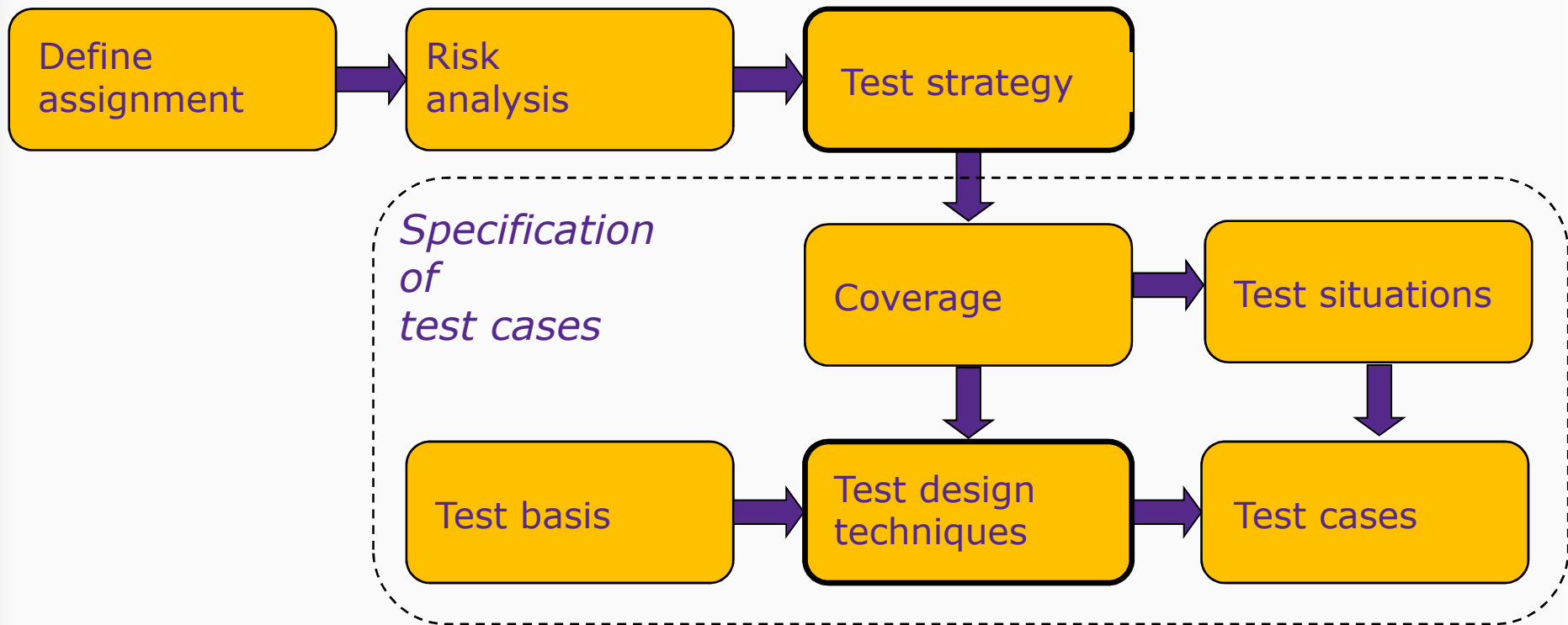




# Generic steps of a test design technique

- ▶ Identify test situations
- ▶ Create logical test cases
- ▶ Create physical test cases
- ▶ Establish the starting point
- ▶ Create the test script

# Choice of techniques...



# Building blocks of a test design technique

Test design technique	Test basis	Coverage types / basic techniques
Algorithm test	Technical design	Path coverage test depth level 2
Data Combination Test	All types of test basis	Equivalence classes and (Multiple condition coverage or Pairwise testing)
Elementary Comparison Test	Structured functional specifications	Modified condition / decision coverage
Exploratory Testing	All types of test basis	Several, according to choice
Data Cycle Test	CRUD matrix Data-integrity rules	CRUD and Decision coverage
Process Cycle Test	Structured description of business processes	Path coverage test depth level 2
Real-Life Test	Operational or load profiles	Statistically responsible simulation
Semantic Test	Input and output specifications , Business rules	Modified condition / decision coverage
Syntactic Test	Input and output specifications	Checklist
Use Case Test	Use cases	Checklist, Path coverage, Equivalence classes
<Your name> test	???	???



# Choice of a particular test design technique depends on ...

Choice	Depends on ...
What?	System characteristics to be tested
How thorough?	Coverage required x resources, time available
Based on?	Available relevant information on the system characteristics (test basis)
With what?	Coverage types that can be applied to the test basis
By whom?	Skilled tester who knows how to apply technique
Who chooses?	Knowledge of test manager about which ones to choose, plus additional choices (level of documentation/detail)!

# Assignment of techniques

Characteristic / object part	PRA -RC	Thoroughness	Type	Techniques
functionality				
- system part 1	B	●●	Funct.	Test Unit1: DCoT, SYN Test Unit2: DCoT, SYN Test Unit3: Exploratory
- system part 2	A	●●●	Funct.	Test Unit4: ECT, SYN, SEM Test Unit5: Exploratory, SYN
- total	C	●	Regression	Happy paths, DCyT
security				
- application	B	●	Authorisation test	SEM
Performance				
- online	B	I		"Stopwatch"

13

PRA

test strategy

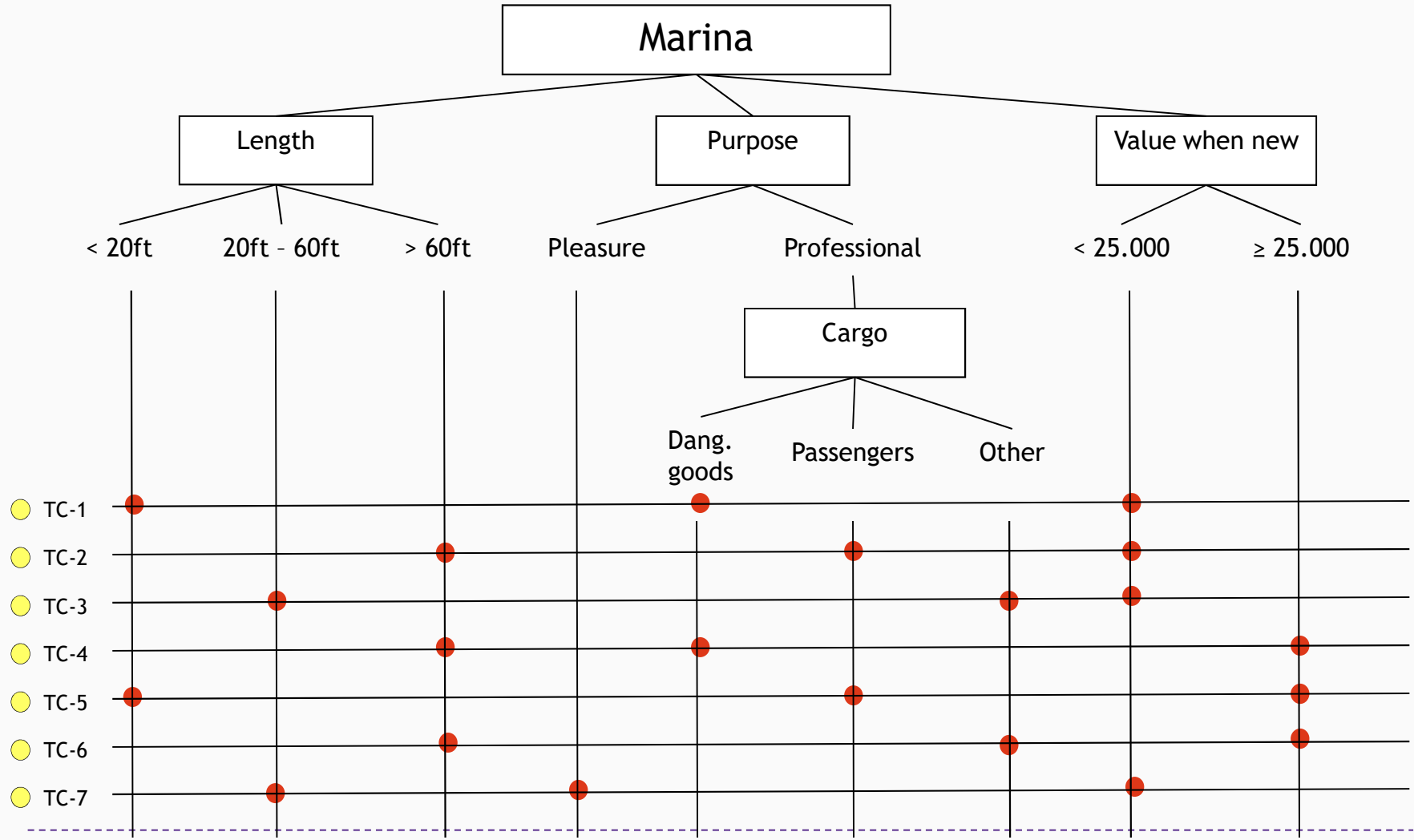
From Strategy to Techniques



# Classification Tree Method

- ▶ What to test?
  - ▶ Integral and detailed functionality
- ▶ Test basis... "Everything"
  - ▶ Formal system documentation
  - ▶ Informal documentation
  - ▶ Domain expertise
- ▶ Coverage / Basic techniques
  - ▶ Equivalence classes
- ▶ Amount of work / kind of testers
  - ▶ Light – medium, also for non-techies (users; business)
- ▶ Variations
  - ▶ Multiple Condition Coverage, Pairwise testing, BVA

# Example - logical test cases CTM



# Pairwise testing

- ▶ What to test?
  - ▶ Integration or regression, with many parameters
- ▶ Test basis
  - ▶ Same as equivalence classes
- ▶ Coverage / Basic techniques
  - ▶ Very effective, more thorough than equivalence classes
- ▶ Amount of work / kind of testers
  - ▶ Free tooling!
- ▶ Variances
  - ▶ N-wise testing



## Some parameters ...

- ▶ Parameters:
  - ▶ Trip: one-way, return
  - ▶ Class: 1st , 2nd
  - ▶ Payment: credit card,debit card, cash
  - ▶ Discount: yes, no
  - ▶ Place: lower, upper, standing, roof
  - ▶ Train type: ICE, slow train, diesellocc, intercity, steam
  - ▶ Age: <18, 18-65, 65+
  - ▶ Catering: coffee, food, no
  - ▶ Color: red, yellow, blue
- ▶ All combinations: 12960 ...

	Trip	Class	Payment	Discount	Place	Train type	Age	Catering	Color
1	one-way	1st	debit card	yes	upper	steam	65+	coffee	red
2	return	2nd	credit card	no	standing	intercity	<18	no	blue
3	return	2nd	cash	yes	lower	dieselloc	18-65	food	yellow
4	one-way	1st	debit card	no	standing	slow train	<18	food	red
5	one-way	1st	credit card	no	standing	steam	18-65	coffee	yellow
6	return	1st	cash	no	lower	slow train	65+	coffee	blue
7	return	2nd	debit card	yes	upper	slow train	18-65	no	yellow
8	one-way	2nd	cash	yes	lower	steam	<18	no	red
9	one-way	2nd	credit card	yes	roof	ICE	65+	food	blue
10	return	1st	cash	no	upper	ICE	<18	coffee	red
11	one-way	2nd	debit card	yes	lower	intercity	65+	coffee	yellow
12	one-way	1st	debit card	yes	standing	ICE	65+	no	yellow
13	one-way	1st	credit card	no	standing	dieselloc	18-65	coffee	red
14	one-way	1st	debit card	no	lower	ICE	18-65	no	blue
15	one-way	2nd	debit card	no	upper	dieselloc	65+	food	blue
16	return	1st	debit card	no	roof	slow train	<18	no	yellow
17	return	1st	cash	yes	roof	intercity	18-65	food	red
18	return	2nd	credit card	yes	upper	dieselloc	<18	no	blue
19	return	1st	debit card	yes	upper	intercity	65+	coffee	red
20	return	2nd	credit card	yes	roof	steam	65+	food	blue
21	one-way	2nd	cash	yes	standing	slow train	65+	coffee	blue
22	return	1st	credit card	no	roof	dieselloc	65+	coffee	red
23	one-way	1st	credit card	no	lower	slow train	18-65	food	red

# Pitfalls ... techniques not used

## ▶ Situation

- ▶ test basis not available, unsuitable, unstable or a mix ...
- ▶ (test documentation) takes too much time, too difficult
- ▶ our own “technique” achieves 100% (requirement / scenario / interface) coverage
- ▶ we’re in a maintenance situation ...

## ▶ Counter measures

- ▶ in most cases, a suitable technique exists ...
- ▶ break up in smaller pieces (test units)
- ▶ use light-weight test documentation
- ▶ training



# Choices in elaboration of test design

## ▶ How far to go?

- ▶ Break down system in smaller parts (test units)

- ▶ Choose somewhere between:

← ↑  
A) describe in abstract terms, even only choice of a technique.  
Leave rest to the tester to work out during execution

...

↓  
Z) describe to the highest level of detail what the tester needs to do exactly

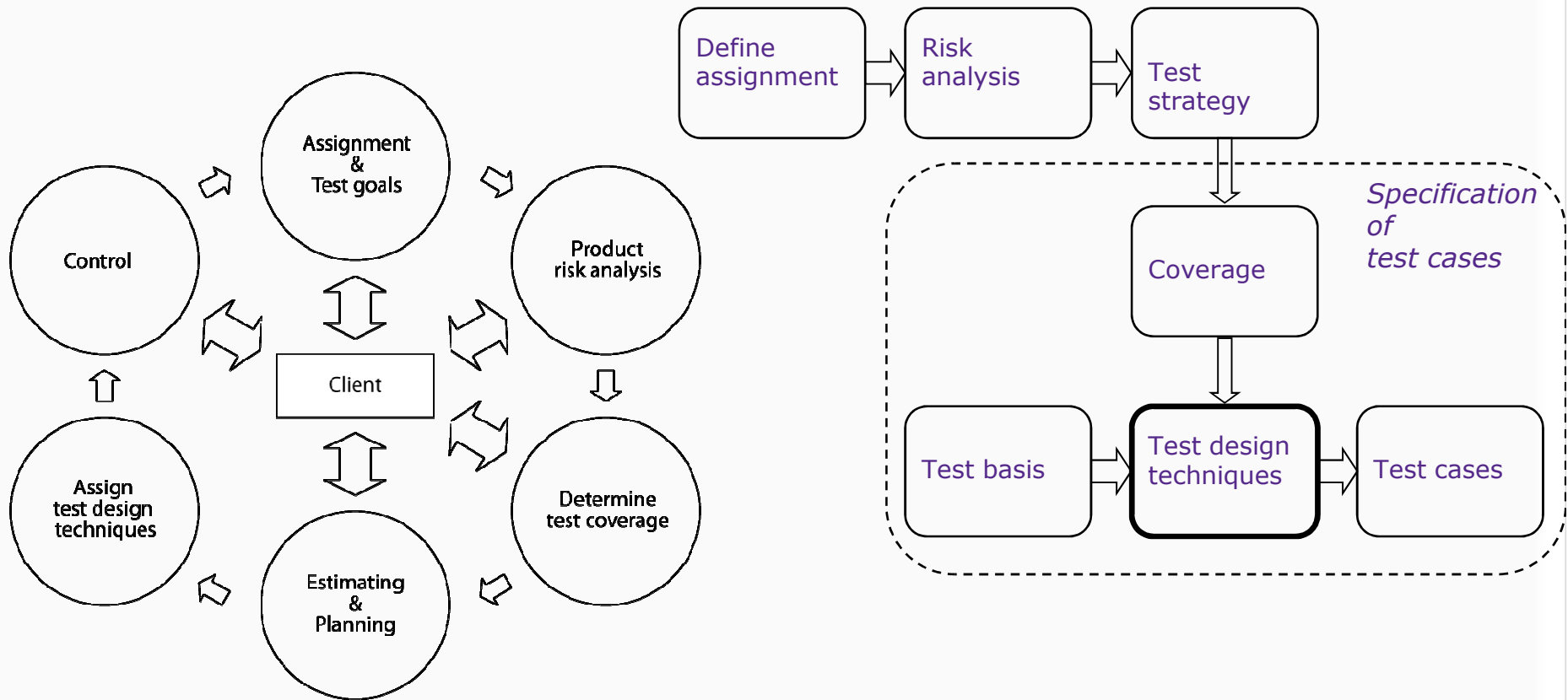
## ▶ In what form?

- ▶ Text document, spreadsheet, database, test management tool (or mix)

# How to implement?

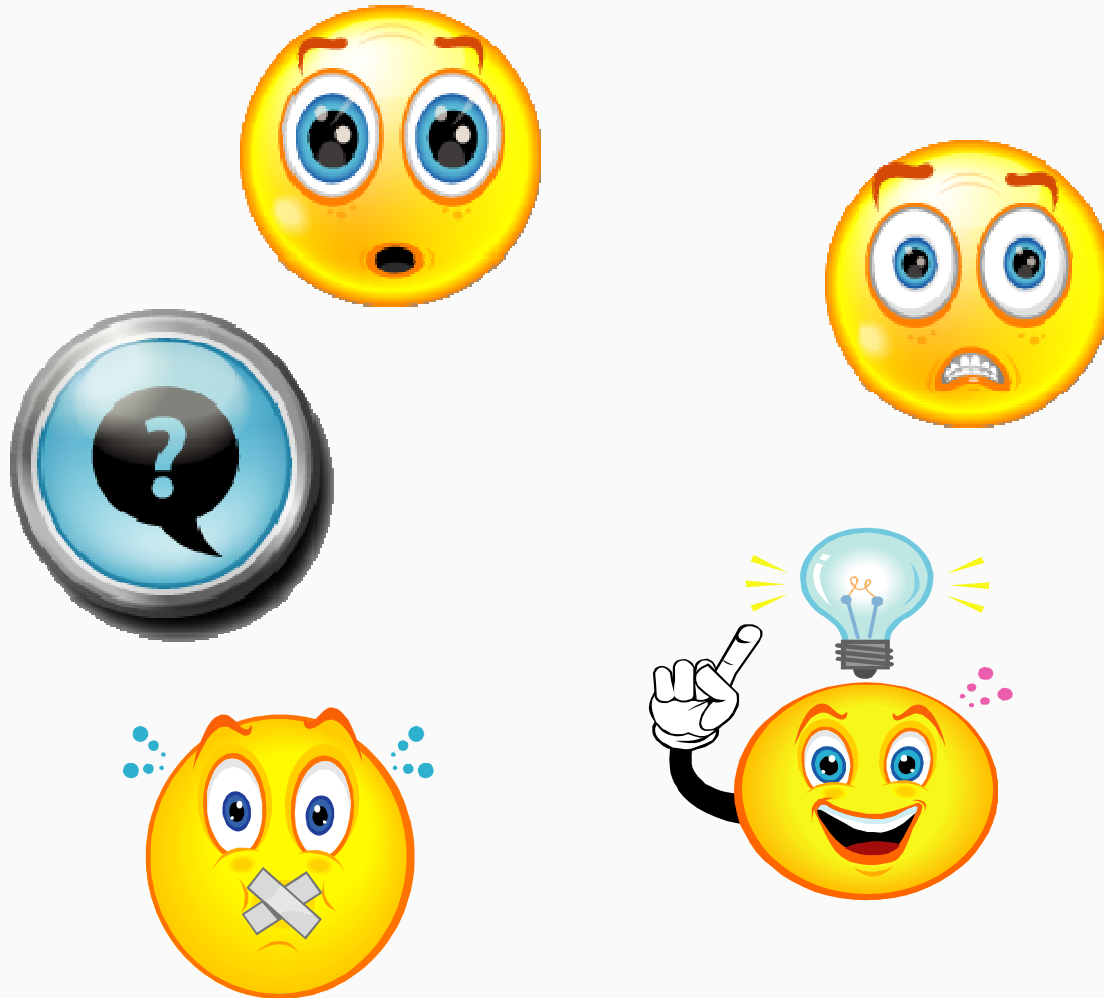
- ▶ Involve the (professional) testers
- ▶ Translate current test design approaches to test design techniques ...
- ▶ Arrange expertise on test design techniques
- ▶ Pilot
- ▶ Monitor

# From strategy to techniques ...



# Finally ...

- Questions?
- Fragen?
- Pregunta?
- Fråga?
- Vragen?





E.	<a href="mailto:info@timkoomen.nl">info@timkoomen.nl</a>
M.	+31 (0)6 34139260
I.	<a href="http://www.timkoomen.nl">www.timkoomen.nl</a>





# Appendices

- ▶ The following slides are NOT part of the presentation

# Basic techniques / coverage types

Coverage type	Description	Variations
Paths	Coverage of the variations in the process in terms of combinations of paths.	Test depth level N
Decision points	Coverage of the various possibilities within a decision point with the purpose of arriving at the outcomes of TRUE and FALSE.	(Modified or Multiple) Condition/decision
Equivalence classes	The value range of a parameter is divided into classes in which different system behaviour takes place.	--
Pairwise testing	Pairwise testing combines all the possibilities from every set of 2 parameters.	N-wise testing
Boundary value analysis	Boundary value analysis tests the boundary value itself plus the value directly above it and directly below it.	Boundary plus one or plus two values around
CRUD	Coverage of all the basic operations (Create, Read, Update, Delete) on all the entities.	--
Operational profiles	Simulation of the realistic use of the system by carrying out a statistically responsible sequence of transactions	--
Right paths/Fault paths	Coverage of the right path and the fault path	Only right paths or only fault paths
Checklist	Checking off an unstructured list.	--

# Overview test design techniques

Quality characteristic	Test design technique		
	• / light	•• / average	••• / thorough
Manageability	CKL	DCT	DCT
Security	CKL	SEM	Penetration test
Usability	UCT	UCT, PCT*	RLT
Continuity		RLT	RLT
Functionality - integration	DCT	DCT, DCyT, PCT	DCT
Functionality – detail	DCT	DCT, ECT	DCT + BVA, ECT + BVA, DTT
Functionality – validations	SYN	SYN, SEM	
User-friendliness	SYN	SYN, UCT, PCT	Usability-test (lab?)
Infrastructure		RLT	
Suitability	PCT t.size 1, UCT	PCT	PCT t.size 3
Performance		RLT	
Portability	CKL, random sample funct.tests / environm.	Functional regression, important environments	All functional tests All environments