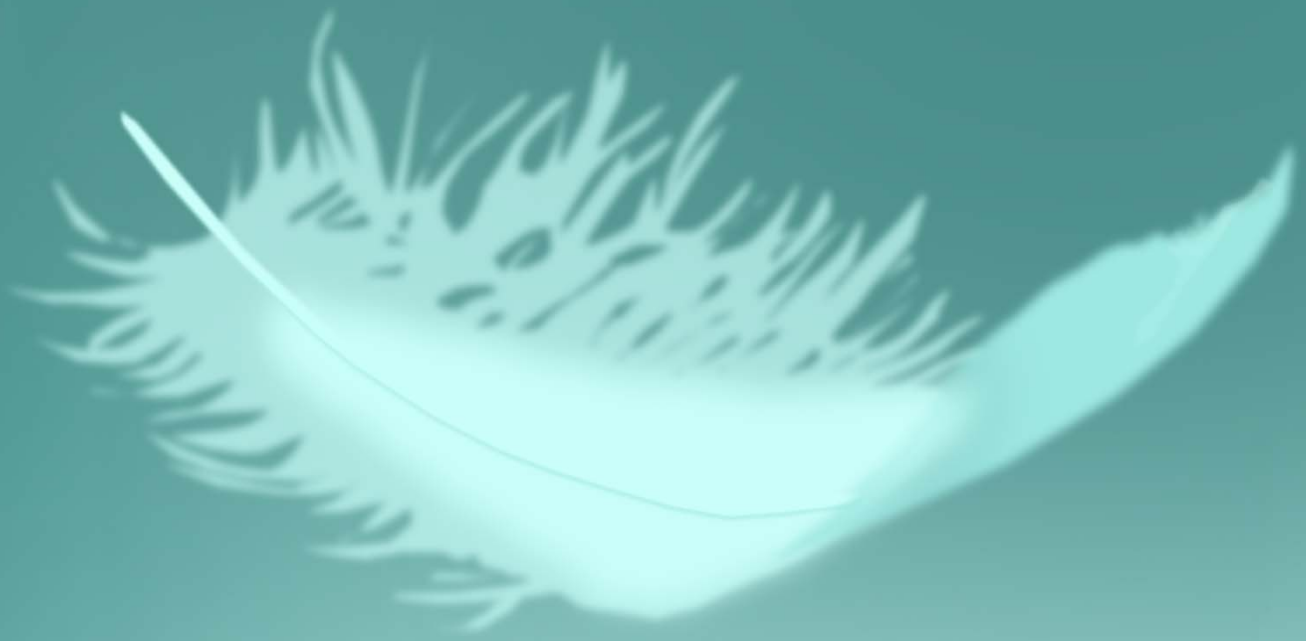
BREDEX
Software-Entwicklung und Beratung



Fighting through the jungle

A checklist for test automation

 Version 2.3

Agenda

- **Introduction to Bredex**
- **Why a checklist can help**
- **Areas to consider**

  Strategy

  Environment

  Skills

  Tools

  Economics

Bredex GmbH

- Founded in 1987
- Software development and consulting
- End-user projects in various industries → GUIs
- Since 1995: specialty in Java
- Recurring question:

→ How should we test?

BREDEX
Software-Entwicklung und Beratung

TESTING

I FIND YOUR LACK OF TESTS DISTURBING.

How we got into testing

- **What we wanted:**

  Automated functional tests

  Without writing or maintaining code

  Independence from a running application

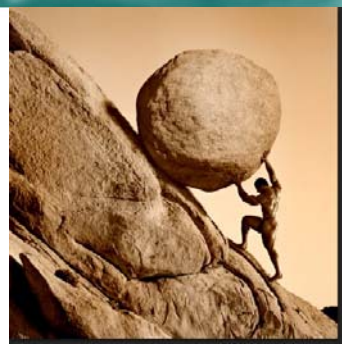
  Support for modularity, abstractness and reusability

- **What we got:**

  Our own test tool

  A tonne of experience in testing theory and practice

Motivation for this talk

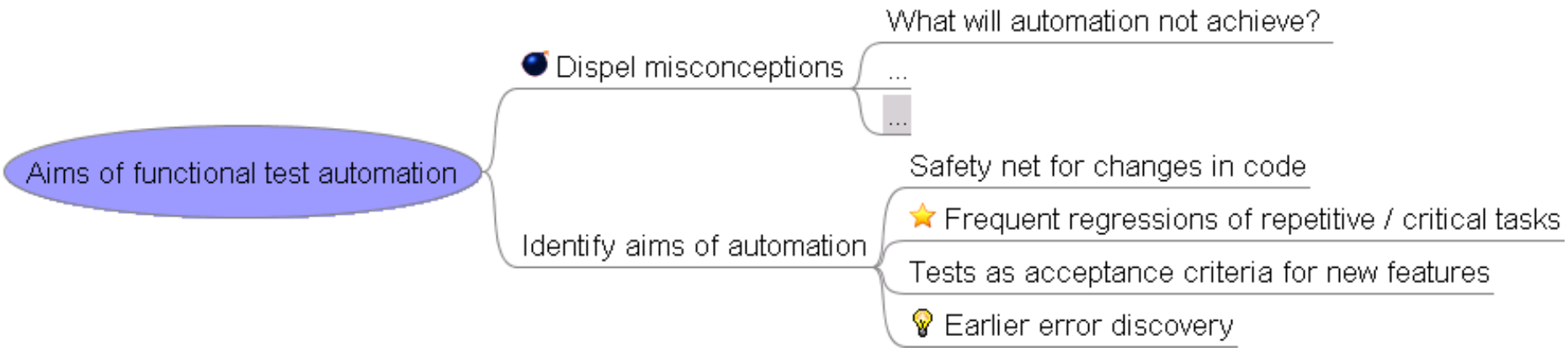


- **We meet two main groups of people:**
  1. People with no formal requirements for aut. testing
  2. People with incomplete requirements for aut. Testing
- **Requirements are hard to define at first**
  Tool often at forefront – what is important?
  Process considerations often forgotten
  Other aspects overlooked
- **Can lead to problems with automation and quality**

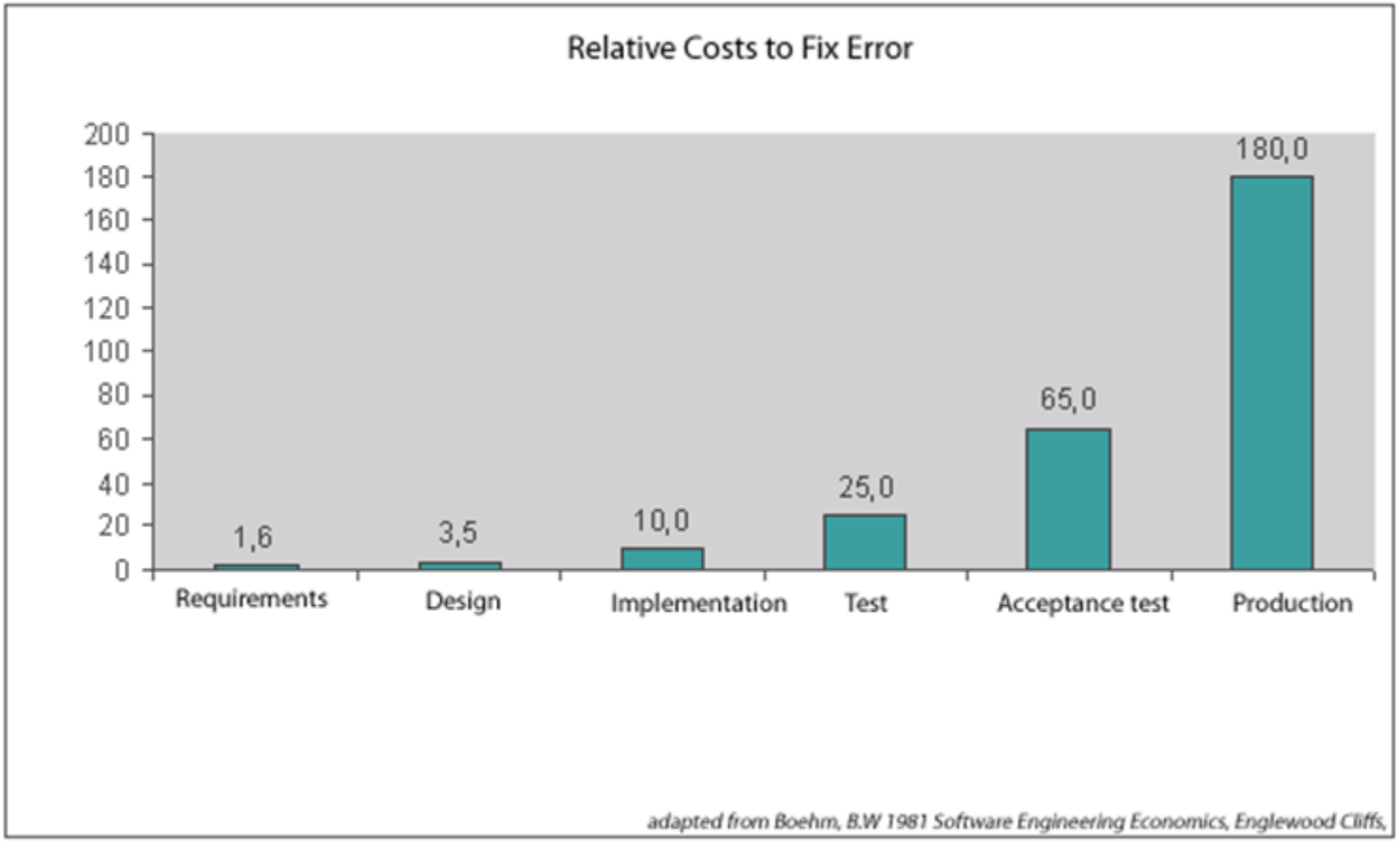
A checklist to work with

- **Are you ready for test automation?**
  - Strategy, organization, process
  - Environment
  - Skills
- **Tool**
- **Economic considerations**

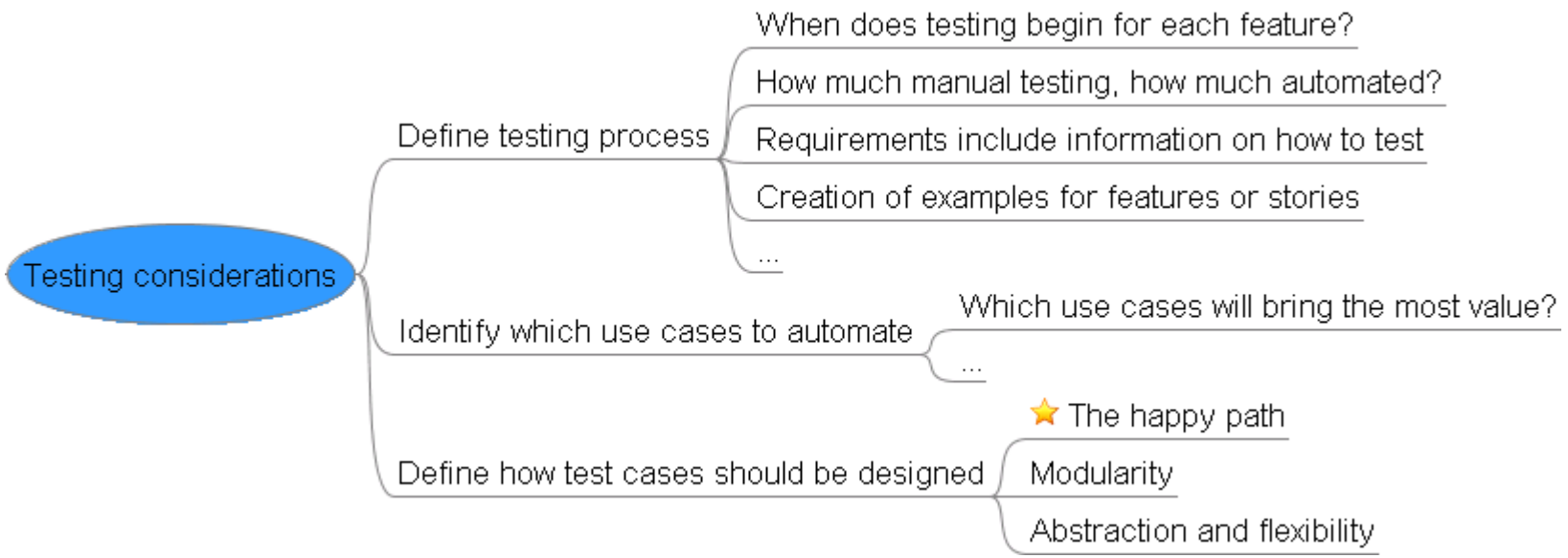
Strategy, organizaztion and process (1/5)



Strategy: Earlier error discovery (2/5)



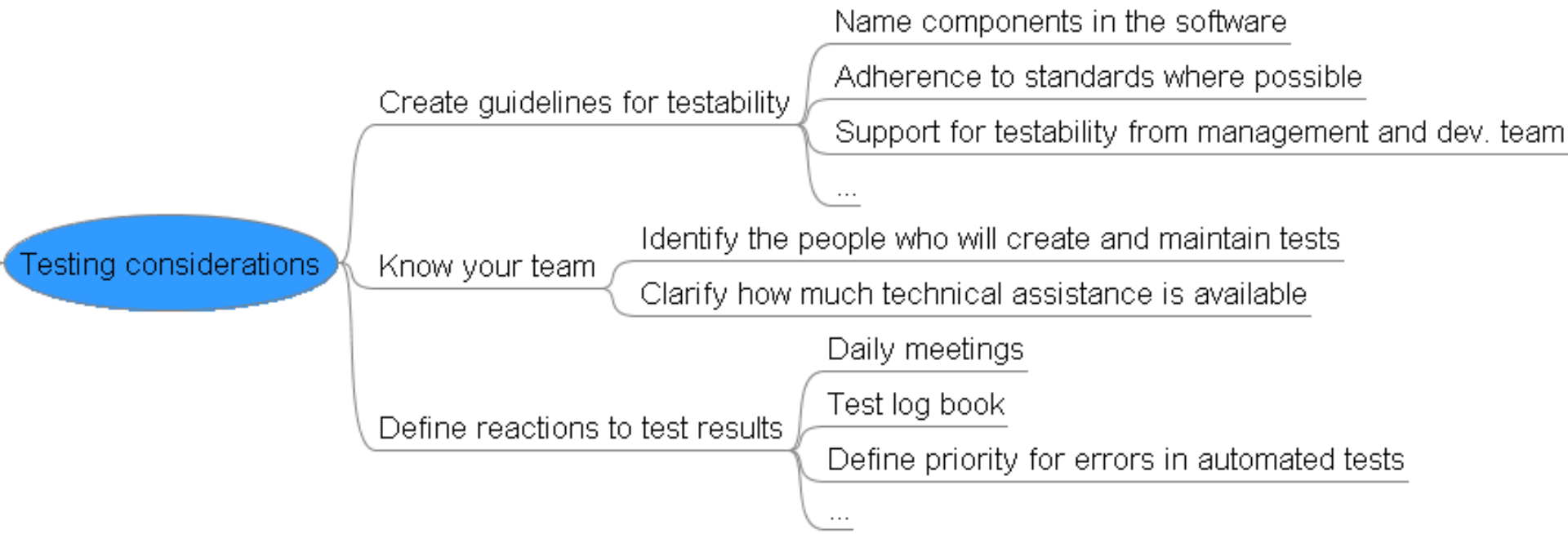
Strategy: Test process (3/5)



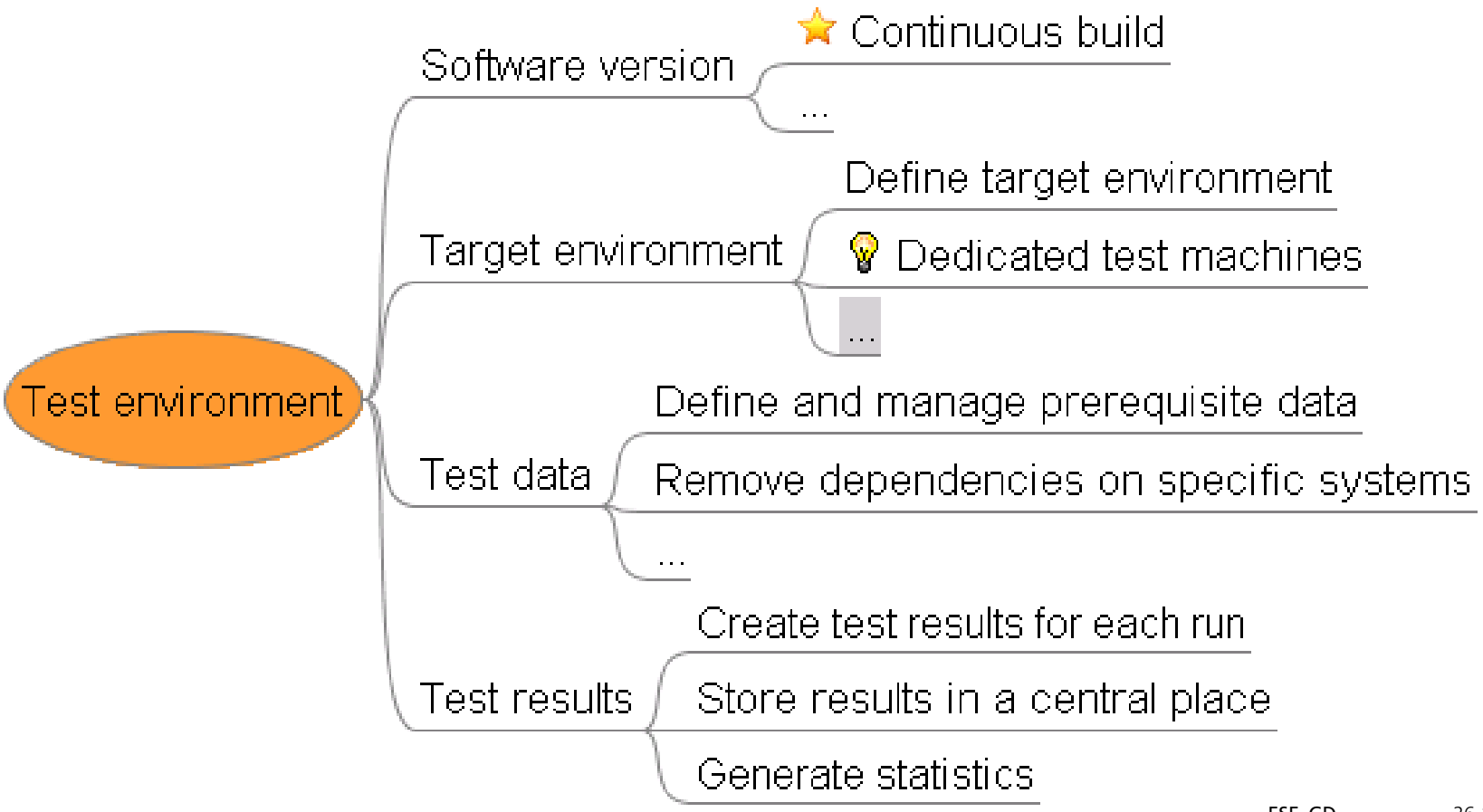
Strategy: Automation success (4/5)

- **Value from tests**
  - High regressions
  - Easy to automate
  - Easy to maintain
  - Critical paths
  - → Reach general safety net quickly
- **The happy path**
  - Best case scenario through the use case
  - Checks basic functionality
  - Other paths will use same modules

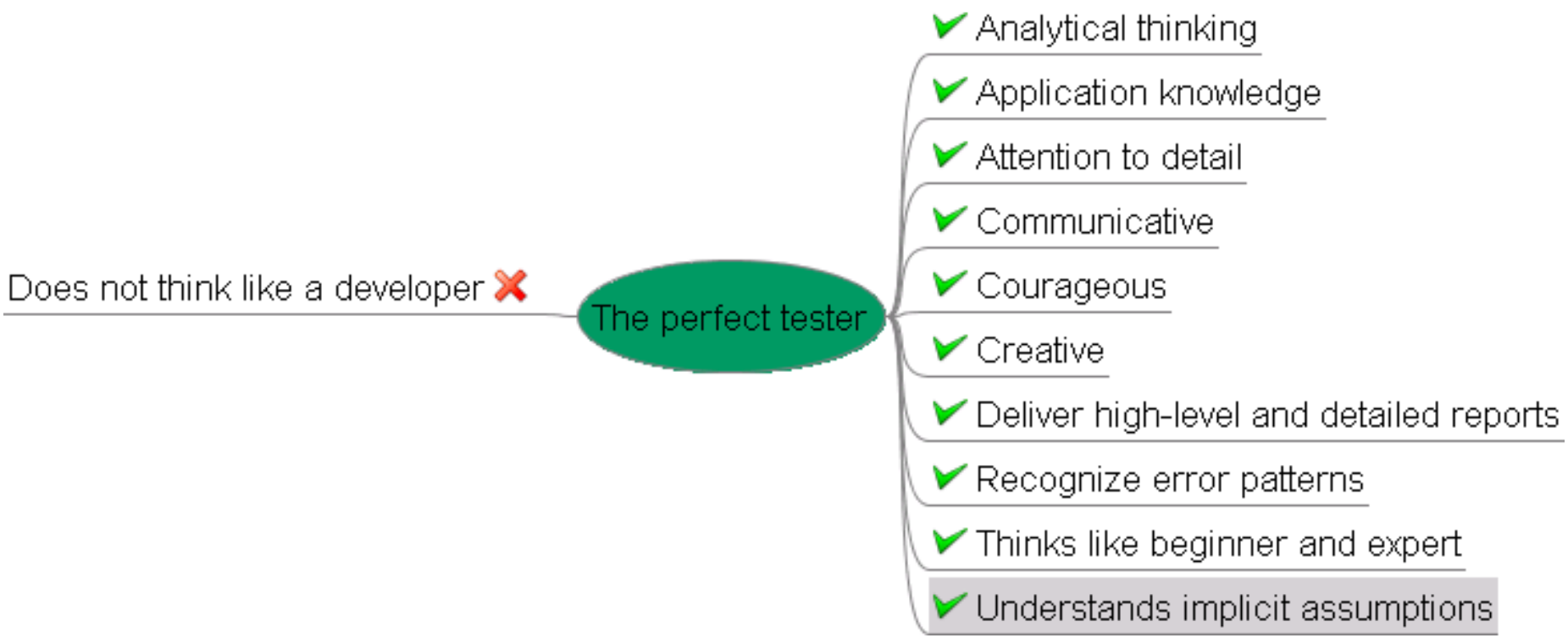
Strategy: Test Process (5/5)



Test environment (1/1)



Tester skills: Only the best can test! (1/1)



The right tool (1/8)

Technical details	1	2	3
Supports platform and technology to test	✓		
Supports application start mechanism	✓		
Extensible in standard language		✓	

Tool: Test creation and maintenance (2/8)

	1	2	3
Tests are written in easy-to-manage and readable format			
A standard library of actions is included			
Test creation is independent of a running app.			
Easy modularization and reuse of modules			
Easy parameterization of data			
Good support for test design and structure			

Tool: Test format (3/8)

- **Advantages of code**

  Powerful – can do anything

- **Disadvantages of code**

  Wrong view of application (not black box)

  Time spent on automation code is time not spent testing or developing

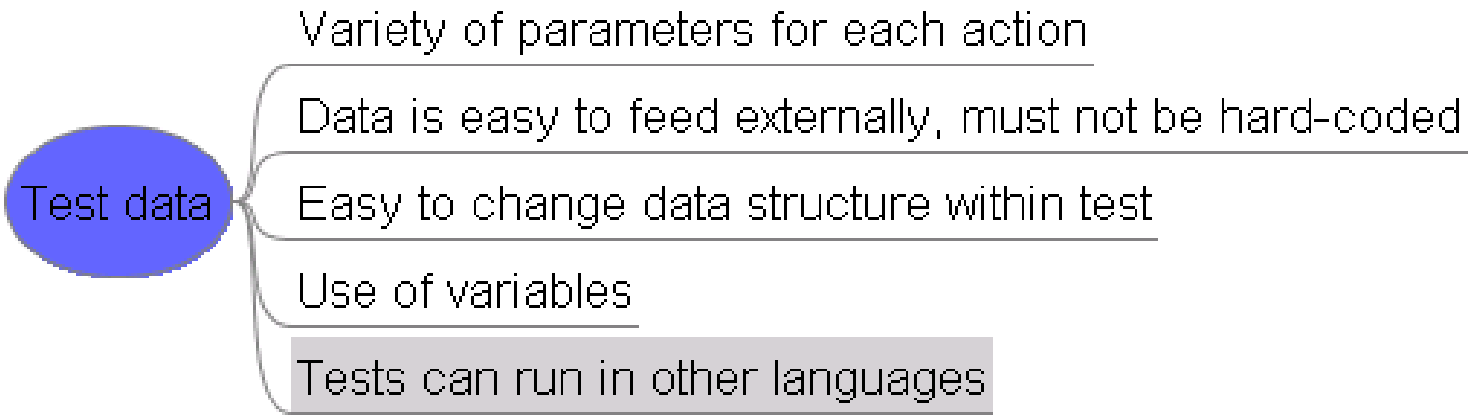
- **Writing automated tests in code = writing whole framework**

  Consumes time and resources

Tool: Test actions (4/8)

	1	2	3
Presented as meaningful, functional operations			
Maximal reusability in different situations			
Explicit synchronization actions available			
Numerous check actions available			
Synchronization and checks easy to add to test			
Easy combination of actions to make modules			
Support for complex actions (drag&drop, tree and table actions, use of native dialogs…)			

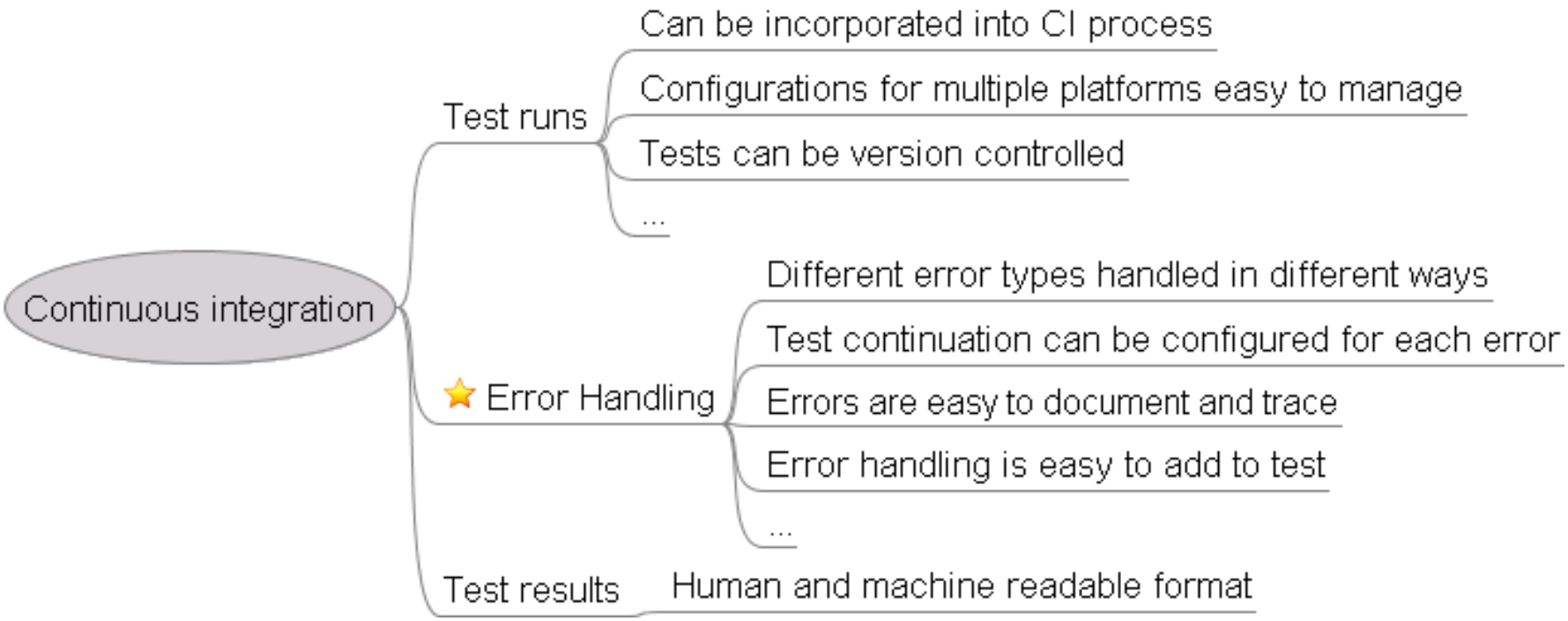
Tool: Test data (5/8)



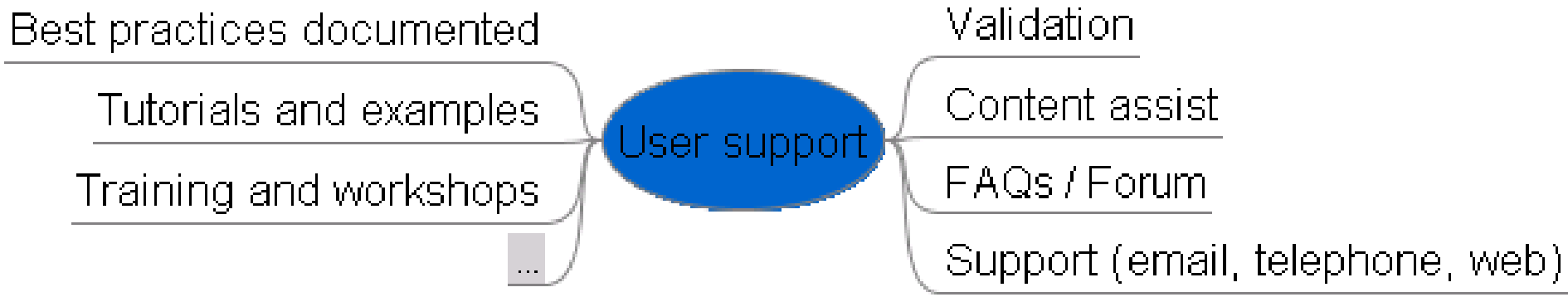
Tool: Object recognition and mapping (6/8)

	1	2	3
Objects not recognized based on coordinates			
Recognition uses a variety of attributes			
Symbolic names link real objects to test			
Object map maintenance is done centrally			
No test data is used in the object map			

Tool: Continuous integration (7/8)



Tool: User support (8/8)



The economic perspective (1/1)

- **Costs caused**

  Definition of process and requirements (one off)

  License and training costs (one off)

  Test creation and maintenance (continuous)

- **Cost benefits – continuous**

  Customer acceptance – reduction in warranty phase

  Reduction in bug-fixing costs

  Reduction of manual test phase

Summary

- **Costs of automated testing can be offset:**
  - Right process
  - Right people
  - Right tool(s)
  - Focus of resources on right areas
- **Use checklist to get it right**

Thank you!

- Questions?