

End-to-End Testing

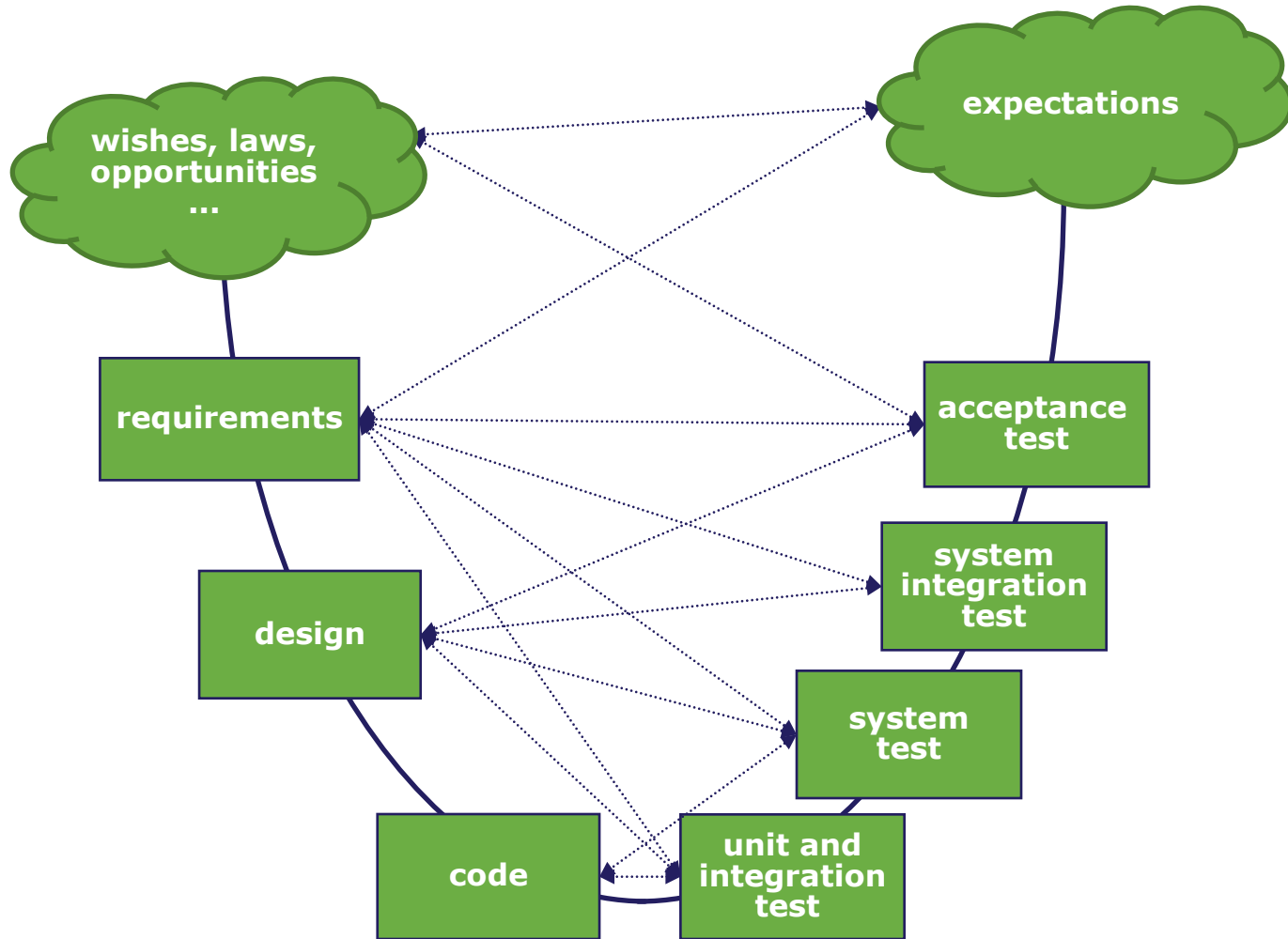
From ad hoc towards a test level,
especially in the Cloud

Ruud Teunissen

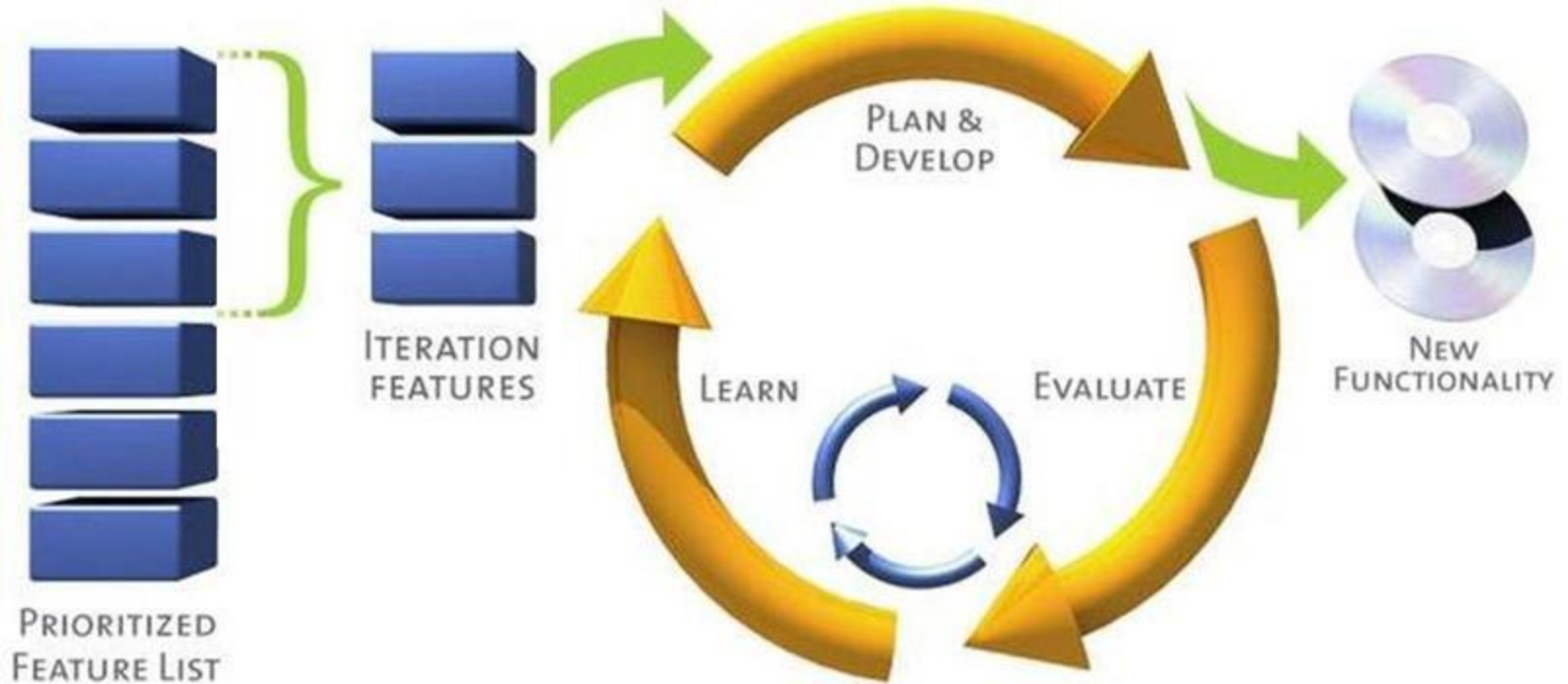
Polteq Test Services BV – The Netherlands

Software Development Life Cycle

Yesterday, Today, ...

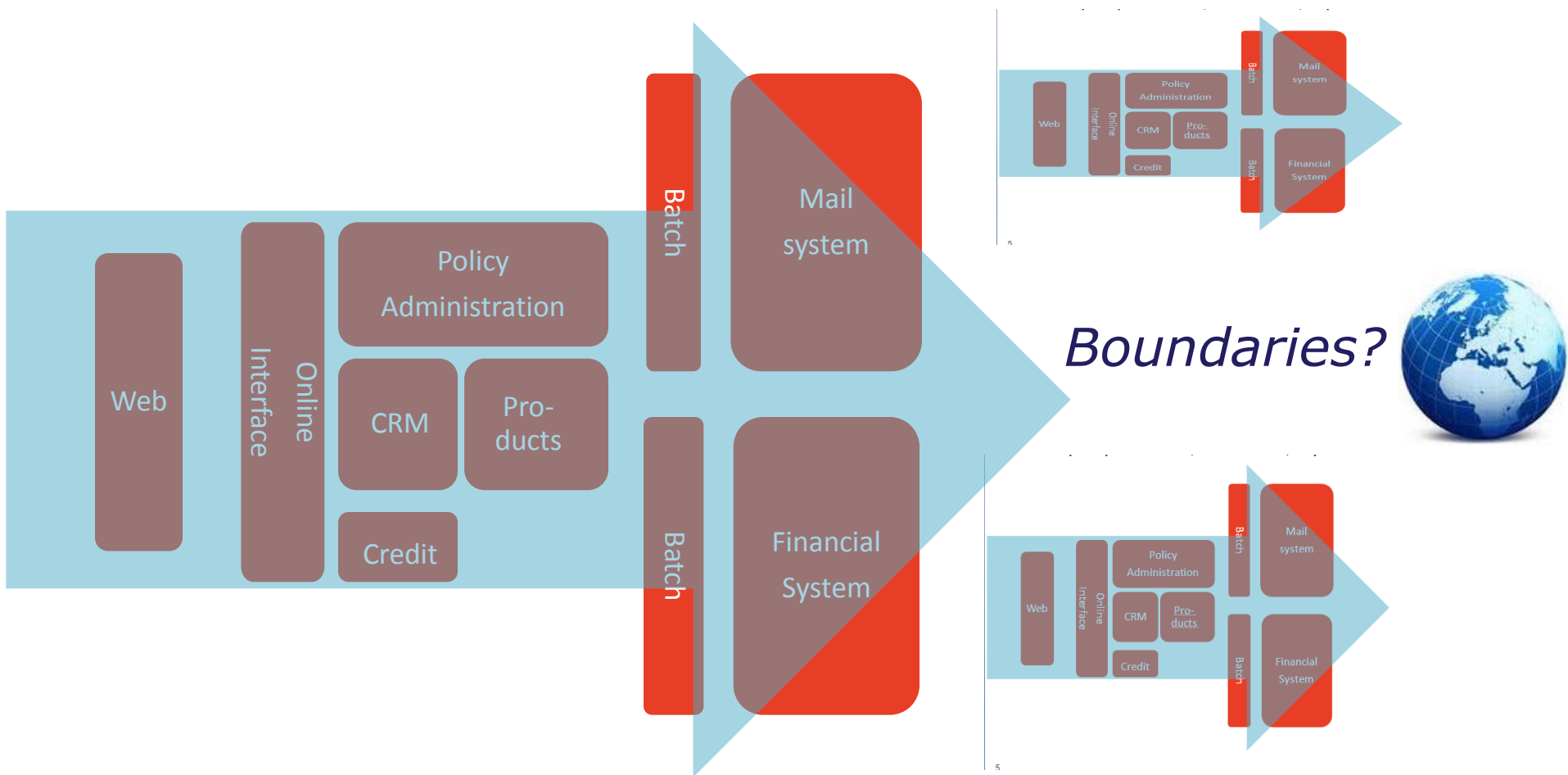


Software Development Life Cycle Yesterday, Today, ...



Landscape & architecture evolve rapidly

- End-user experience & business processes span across multiple products, features, systems and organizations



Interface test



- Goal:
testing the extent to which systems are connected correctly, if all possible data will be sent correctly

System Integration Test



- Goal:
testing the extent to which systems work together and support functionality properly

User Acceptance Test



- Goal:
testing if user requirements have been met and the system and processes are suitable

Interface test, SIT, UAT

Missing:

- Life cycle of data through the chain
- Meaning of fields
- Data synchronicity in all possible circumstances (jobs, month runs, batches, migrations)
- Error and failure handling
- Traceability
- Performance
-

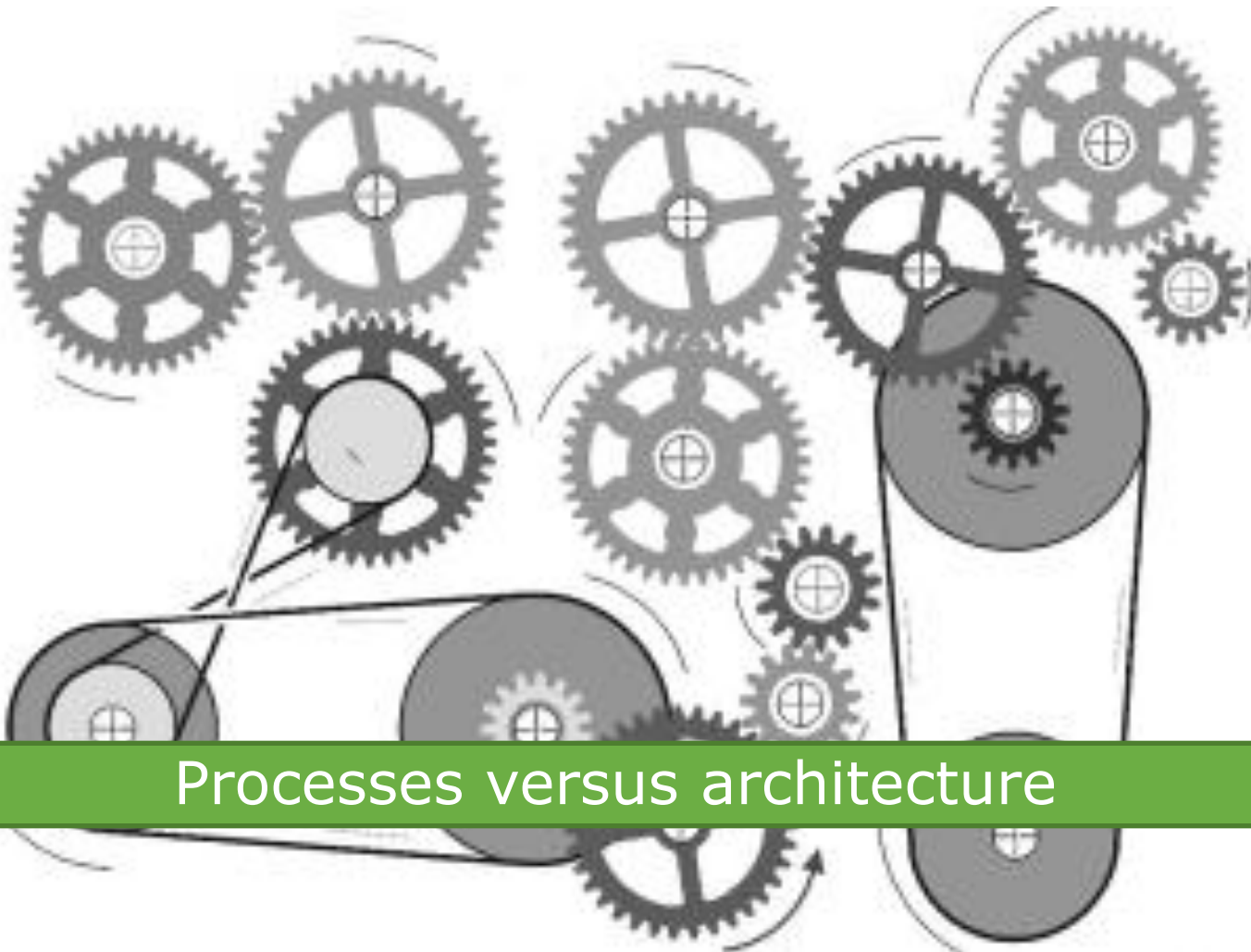
E2E test



- Goal:
 - testing if the entire IT landscape supports processes and people properly

E2E Risk Analysis

- “Helicopter view”

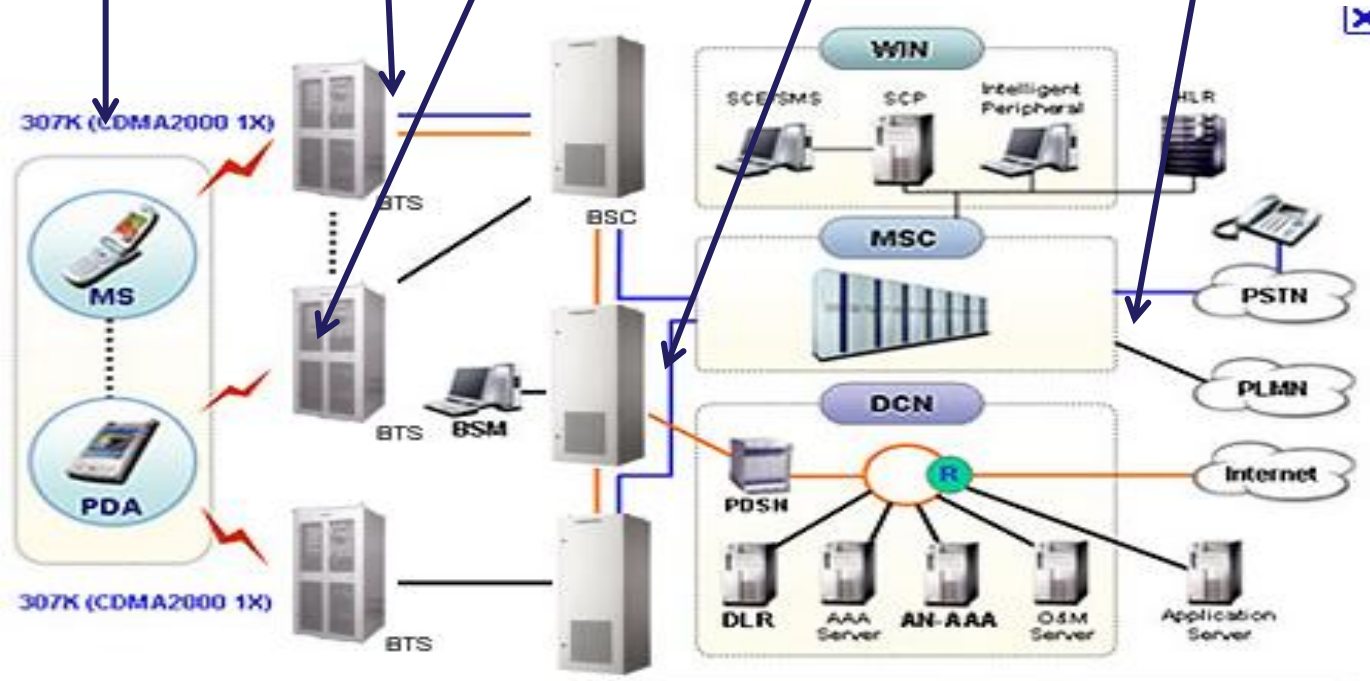


Processes versus architecture

E2E example



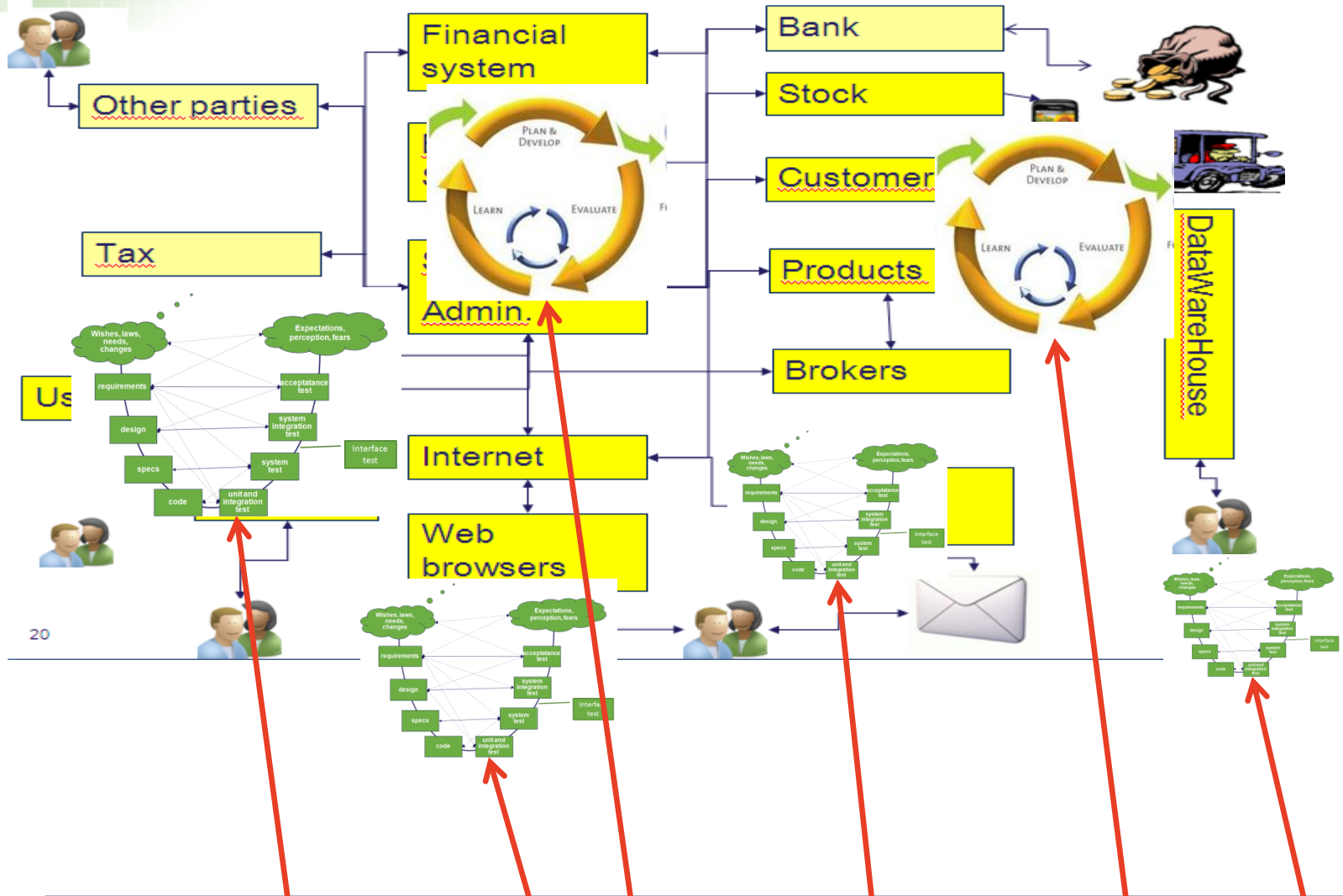
Man wants
To make a call
To wife



Typical E2E risks

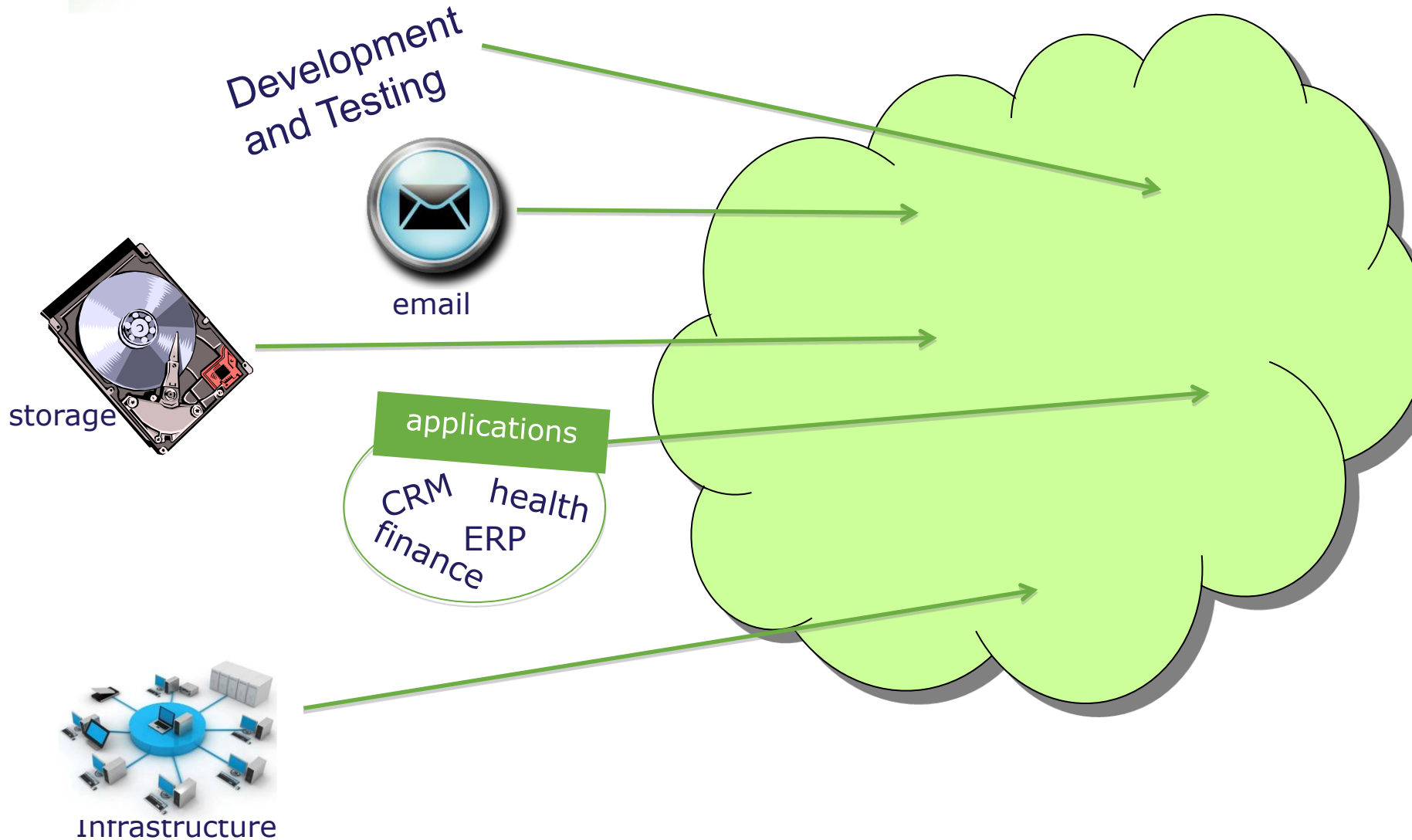
- Actuality & Correctness of data
- Data Life Cycle
- Security & Accessibility
- Defect resilience
- Suitability
- Failover
- Maintainability
- Timing
- Data pollution
- Efficiency

E2E versus systems and projects

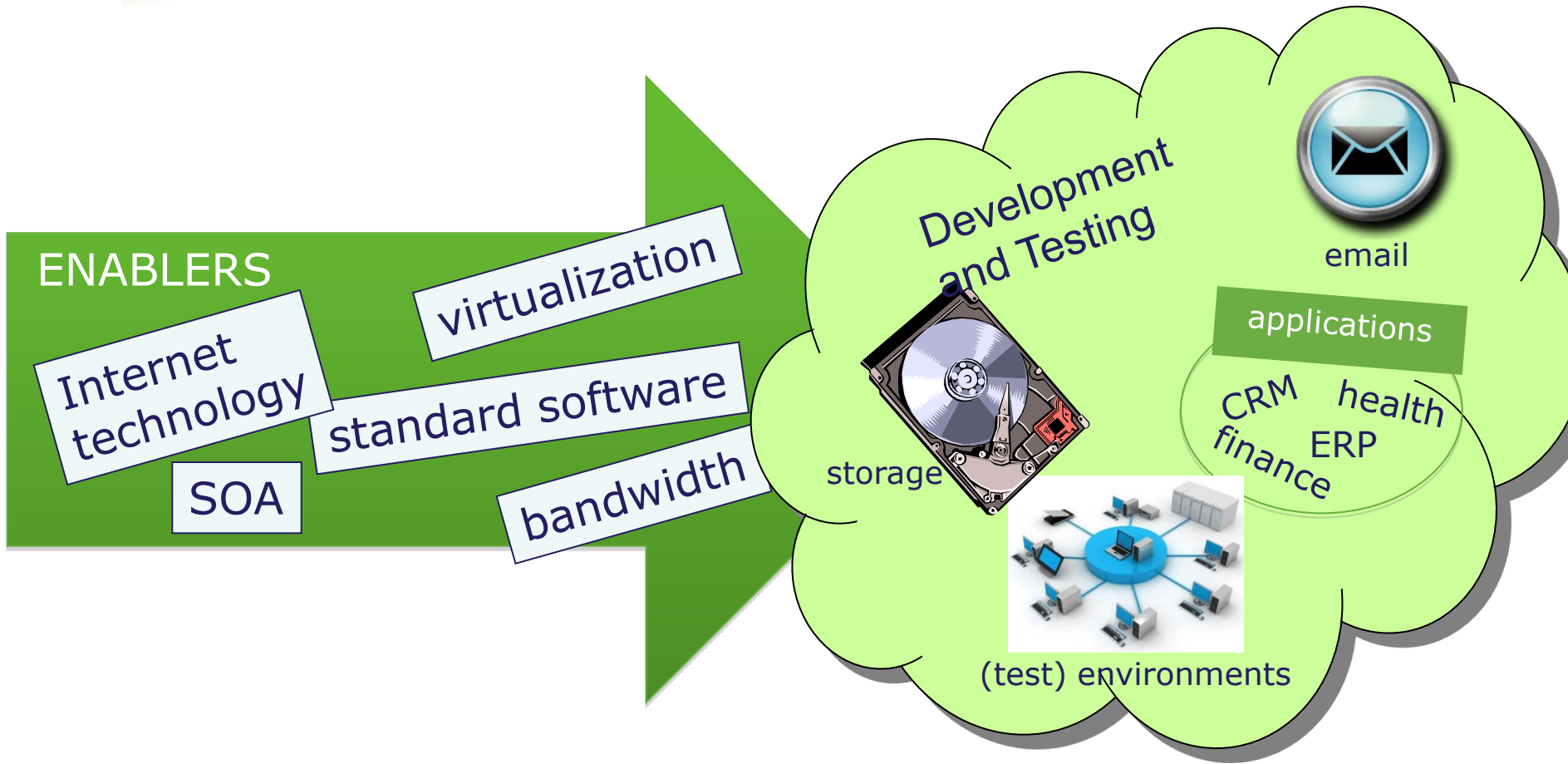


Continuous Integration E2E Testing

What's going into the Cloud...



Enablers for Cloud Computing



Why?



DRIVERS

flexibility

grid

\$ € £

bring your own device

accessibility



Development and Testing

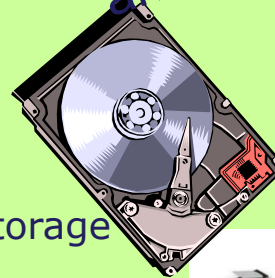


email

applications

CRM health
finance ERP

storage







(test) environments

Cloud Computing according to NIST

Essential characteristics

- ✓ On-demand service
- ✓ Broad network access
- ✓ Resource pooling
- ✓ Rapid elasticity
- ✓ Measured service

Deployment models

-  private cloud
-  community cloud
-  public cloud
-  hybrid cloud

Service Models

Software as a Service

Platform as a Service

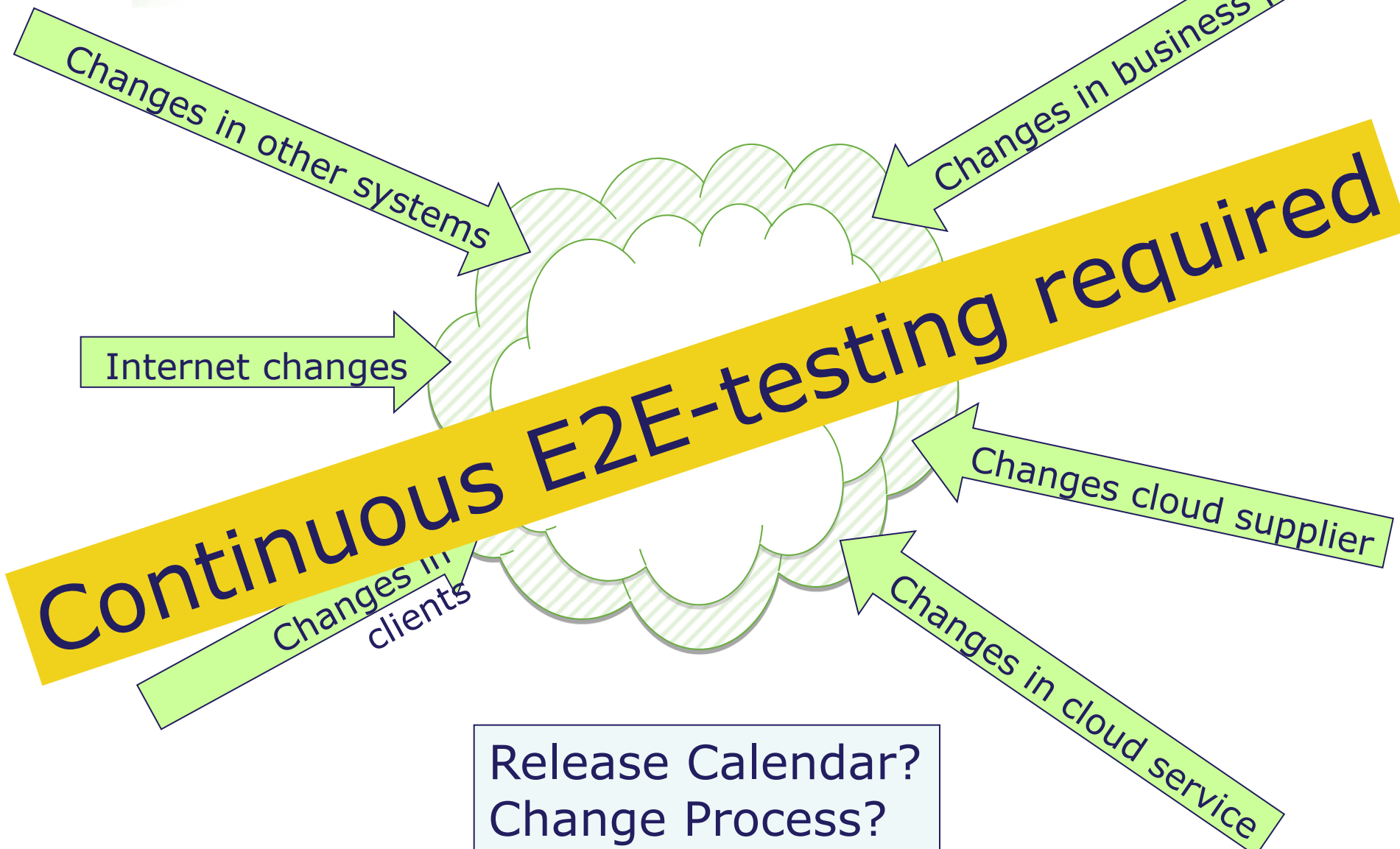
Infrastructure as a Service

*aaS

US: National Institute of Standards and Technology

<http://www.nist.gov>

Change, Continuity and E2E-testing



E2E Risk mitigations

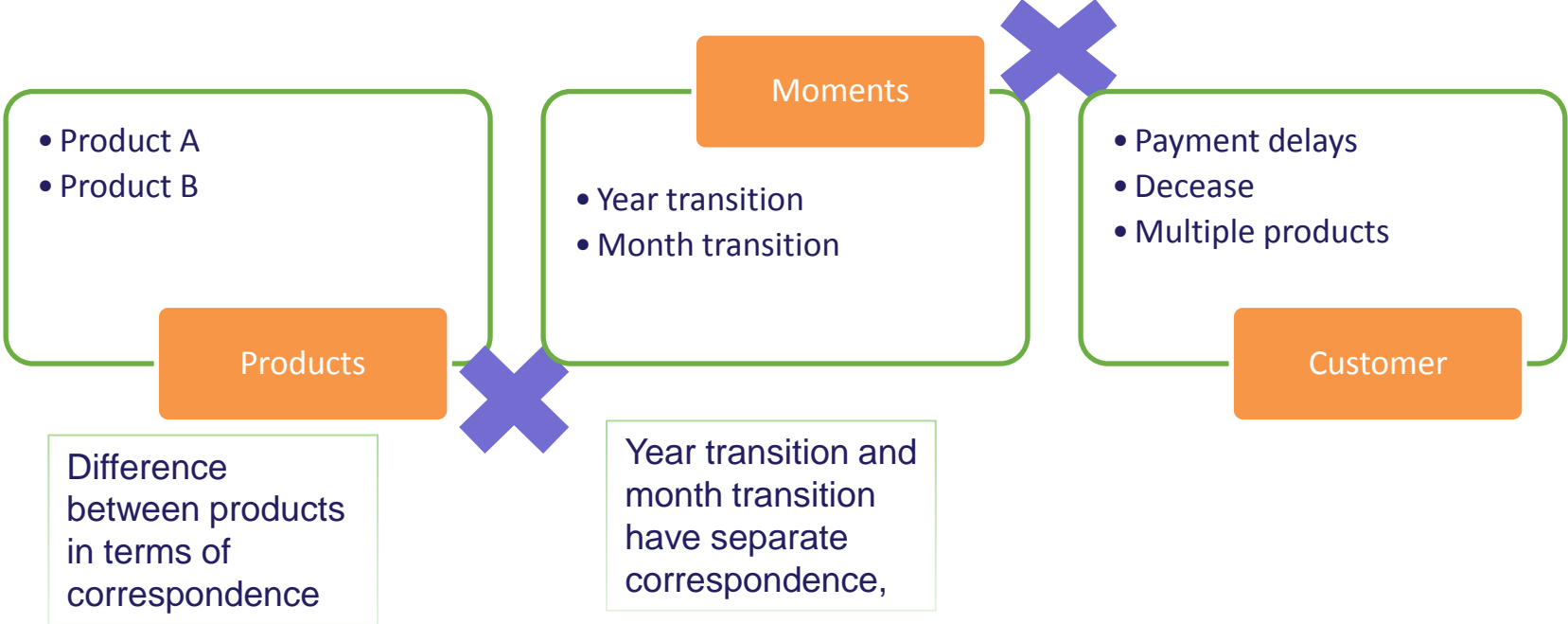
- Standardization
- Certification
- Local Testing (ST, SIT, UAT)
- Operational quality

- E2E-testing

E2E Test Conditions

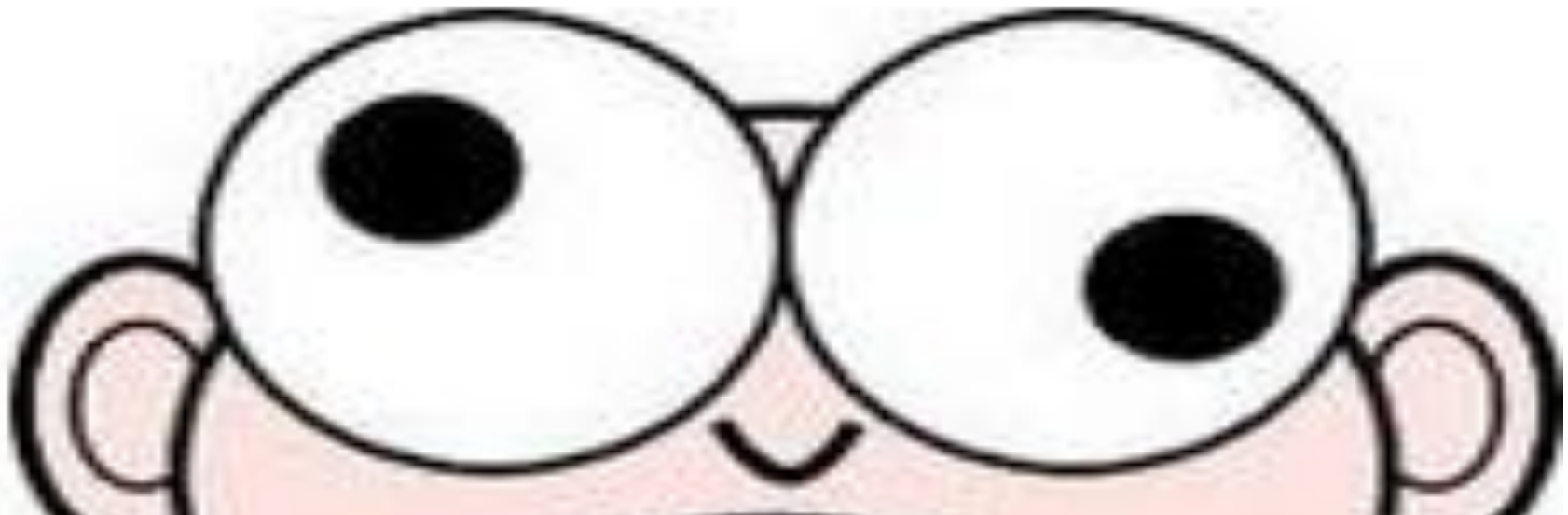
- Infrastructure elements:
 - Systems
 - Interfaces
 - Batches
 - Files
- Processes
- Critical factors
 - Special events
 - Special circumstances

Critical factors



E2E Test Design

- 20 processes
 - 20 systems
 - 30 interfaces
 - 20 batches
 - 10 critical factors
-
- # test cases = $20 * 20 * 30 * 20 * 10 = 2.400.000$



E2E Test Design & Execution

- Preventing the worst
- Focussed on risks (per change)
- Coherence between E2E-processes and architecture
- Periodic monitoring of basic quality

E2E Test Design Techniques

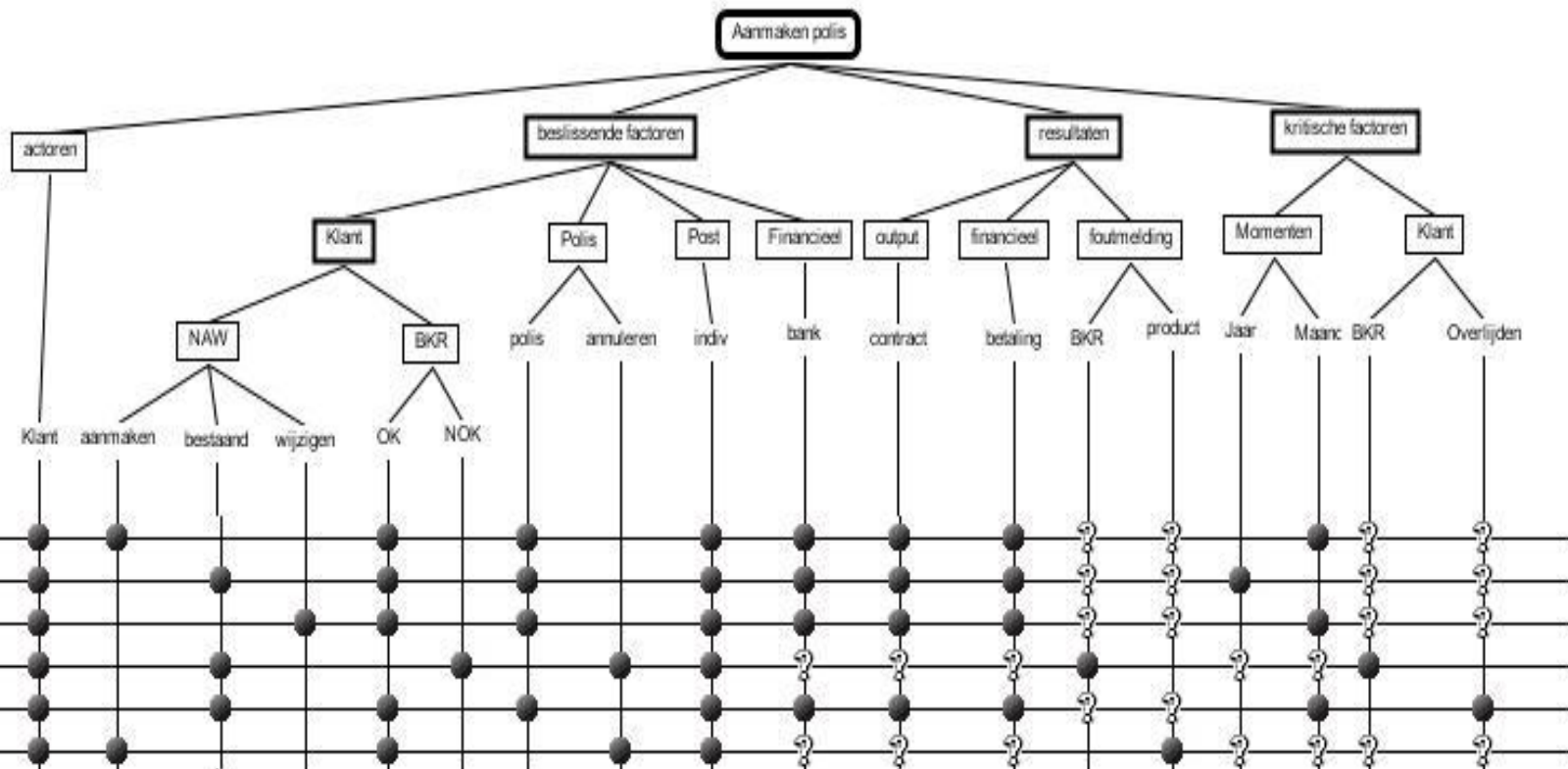


Technique

Application to E2E

Classification tree
Ideal for E2E!

Capture actors and factors, based on equivalence classes. Graphical description of situations to test. Easy to update and change.



Testcase	actoren	beslissende factoren	resultaten	kritische factoren
Testcase 1	●	●	●	●
Testcase 2	●	●	●	●
Testcase 3	●	●	●	●
Testcase 4	●	●	●	●
Testcase 5	●	●	●	●
Testcase 6	●	●	●	●

E2E Test Design Techniques



Technique

Application to E2E

Classification tree
Ideal for E2E!

Capture actors and factors, based on equivalence classes. Graphical description of situations to test. Easy to update and change.

Data Cycle

Data life cycle through the E2E landscape (Create, Read, Update, Delete), including checks for “impossibilities”

E2E Test Design Techniques



<i>Technique</i>	<i>Application to E2E</i>
Classification tree <i>Ideal for E2E!</i>	Capture actors and factors, based on equivalence classes. Graphical description of situations to test. Easy to update and change.
Data Cycle	Data life cycle through the E2E landscape (Create, Read, Update, Delete), including checks for “impossibilities”
Statistical Usage	Simulate the use of the E2E landscape based on Operational profiles (how the system is used) and Load profiles (how the system resources are used)

E2E Test Design Techniques



Technique

Application to E2E

Classification tree
Ideal for E2E!

Capture actors and factors, based on equivalence classes. Graphical description of situations to test. Easy to update and change.

Data Cycle

Data life cycle through the E2E landscape (Create, Read, Update, Delete), including checks for “impossibilities”

Statistical Usage

Simulate the use of the E2E landscape based on Operational profiles (how the system is used) and Load profiles (how the system resources are used)

Exploratory

Necessary due to a lack of documentation, time, data, environments ... and the learning process



E2E Test Design Techniques



Technique	Application to E2E
Classification tree <i>Ideal for E2E!</i>	Capture classes. Easy to use. ... based on equivalence ... of situations to test. ... ge.
Data Cycle	Data life cycle (Create, Read, Update, Delete) "impossible" to test. ... E2E landscape (Create, ... including checks for ...
Statistical Usage	Simulate real world usage. Operate the system under load. Load profiles (how the system is used). ... E2E landscape based on ... the system is ...
Exploratory	Necessary to find edge cases. ... data, ... process



Looking for scenarios that impact 'other' systems, applications, processes!

E2E Test in a multi-organizational world

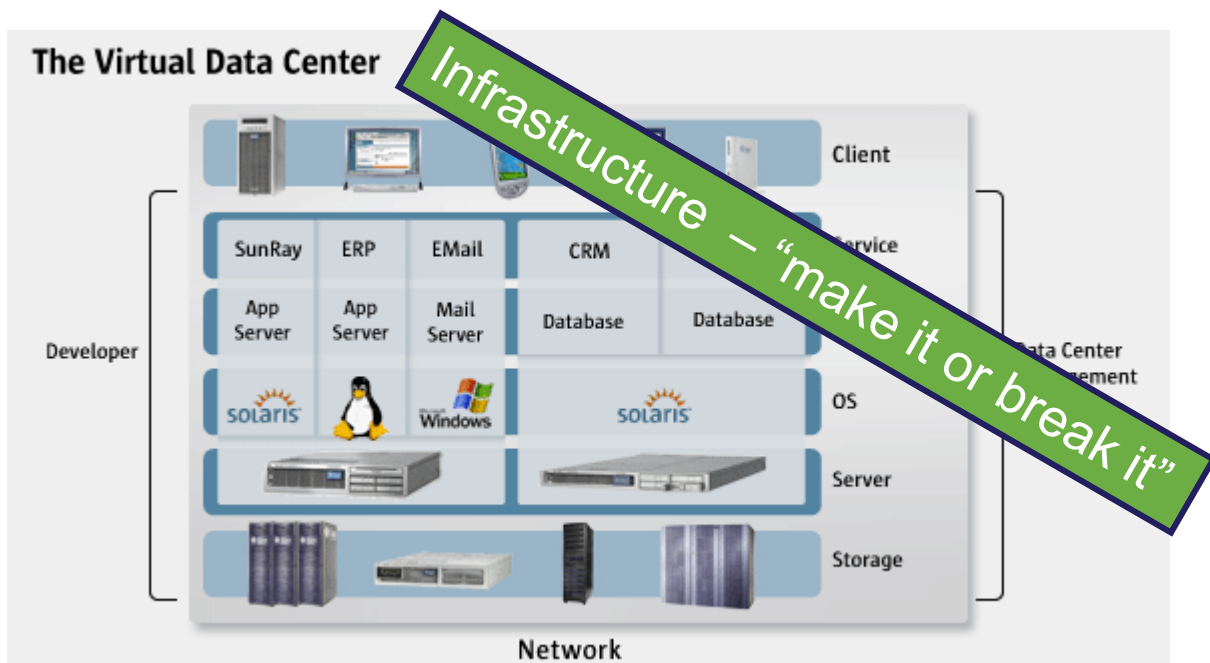
- Local E2E-testing (per project/ organization)
- Overall E2E-testing (domain, shared interest and effort)

E2E Test Environment & Data

- “Production like”
 - Virtualization
- Production
 - Taboo
- Test data management
 - Synchronization including time travel!



- *Syntax*
- *Semantics*
- *Data pollution*
- *Live cycle*
- *Error handling*
- *Traceability*
- *Access*
- *Security*
- *Timeliness*
- *Speed*



E2E Test Tools & Automation

- Test tools to enable automated (services) testing
- Test automation to enable frequent execution of E2E regression test
- Test tools to reduce dependency



E2E Issue Management

- Registering an issue is not enough!
 - You need to drill down in the E2E chain...
- Determine the **root cause** of the problem
and
Define the **required** resolution
- E2E-management (?)



E2E Test Team

- Supertester
- E2E-panel of experts

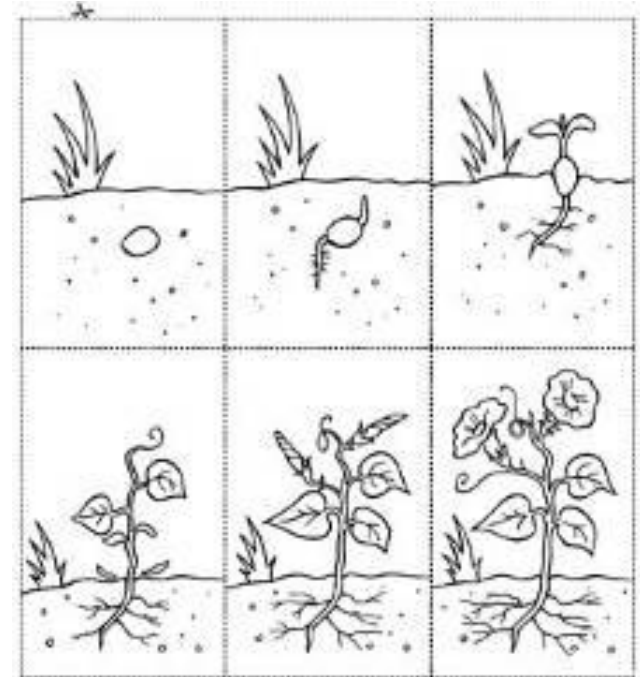


- Testers
 - Testers
 - Users
 - Operators
 - Customers



'Enablers' for E2E

- "Helicopter view"
- E2E Risk Analysis & Test Design
- Test environments
- Shared responsibility view
- Test tools & automation





Tak for din
opmærksomhed,
spørgsmål, kommentarer,
...

About the speaker

Ruud Teunissen

Polteq Test Services, The Netherlands

In the testing world since 1989, Ruud Teunissen has held numerous test functions in different organizations and projects: tester, test specialist, test consultant, test manager, etcetera. Ruud is co-author of Software Testing - A Guide to the TMap® Approach and is a frequent speaker at (inter)national conferences and workshops. He was a member of the program committee for Quality Week Europe and EuroSTAR. Ruud is currently International Test Consultant at Polteq Test Services BV.



TPA® and TMap® are registered trademarks of Sogeti