Quality Street: from the fancy to the mundane

Nederlandse Testdag 2015

Oktober 2015

ALTRAN

Quality Street: from the fancy to the mundane ???

Nederlandse Testdag 2015

Oktober 2015

ALTRAN

Intelligent Systems | ALTRAN

THE GLOBAL INTEGRATOR OF INTELLIGENT SYSTEMS

7 Practices

SYSTEMS ENGINEERING | **SOFTWARE ENGINEERING** | ELECTRONICS | SAFETY | SECURITY | **CONNECTIVITY** | ANALYTICS & INTELLIGENCE

Serving all industries

5 Service Lines

Systems and Product Engineering
- Turnkey Development
- Integration Partner
- Rapid Prototyping

World-in-the-Loop
- Synthetic environments
- HIL, SIL
- Virtual Engineering
- Testing

Connected World
- Connected Machines & Devices,
- Internet of Things
- Network Technology

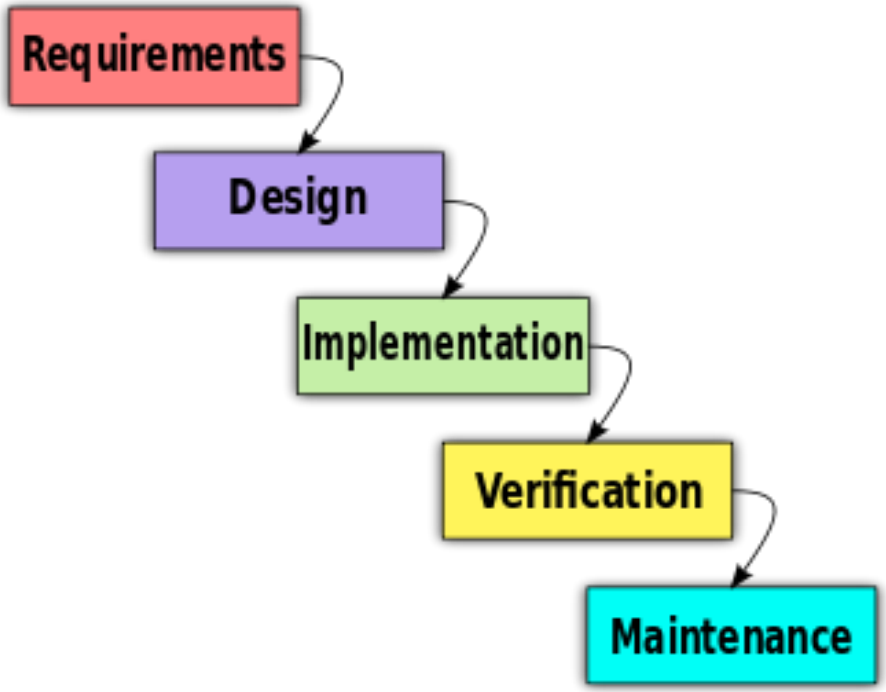
Intelligent Systems | ALTRAN

Human Systems Integration
- User Experience
- Human Factors
- Ergonomics & Design

Safe & Secure World
- Safety-Critical Sw & Syst
- Cyber security
- Certification

ALTRAN

History – still found today



This does not work for software, even though people still try.

Disadvantages:

-Complete requirements up front

-Everything in discrete phases

-Past phases become 'stale' quickly

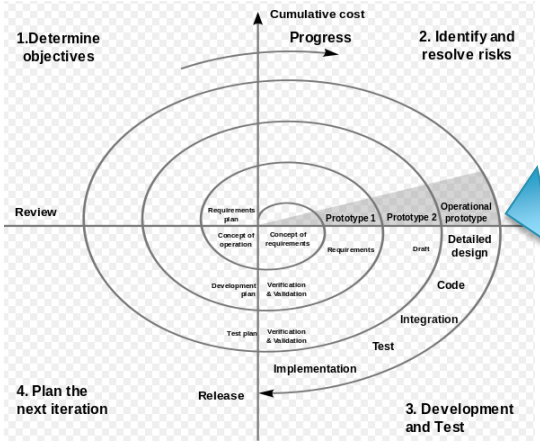
Project Engineering Processes and QA Milestones still use this!

We think like this!

Recognise and understand it.

ALTRAN

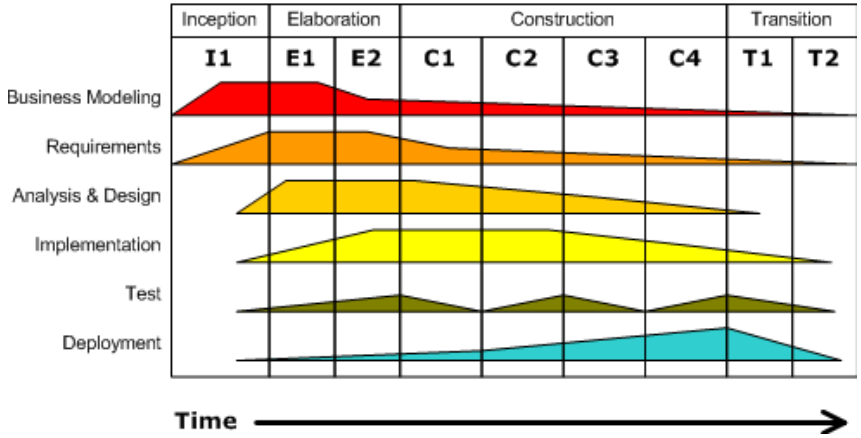
History



- Getting more cyclic, lightweight and adaptive
- After requirements elaboration is still a fairly traditional development model
- Testing is still a separate phase

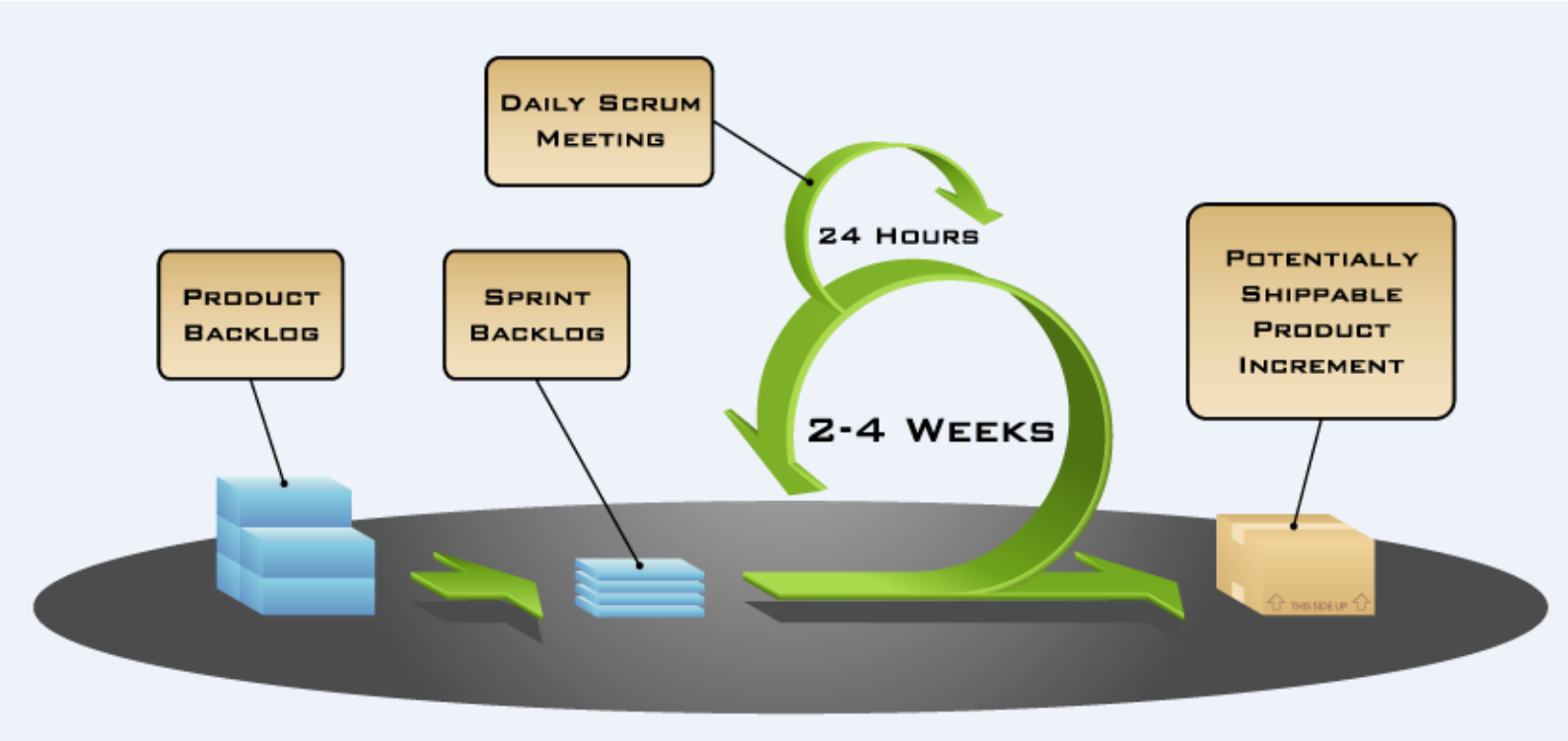
Iterative Development

Business value is delivered incrementally in time-boxed cross-discipline iterations.



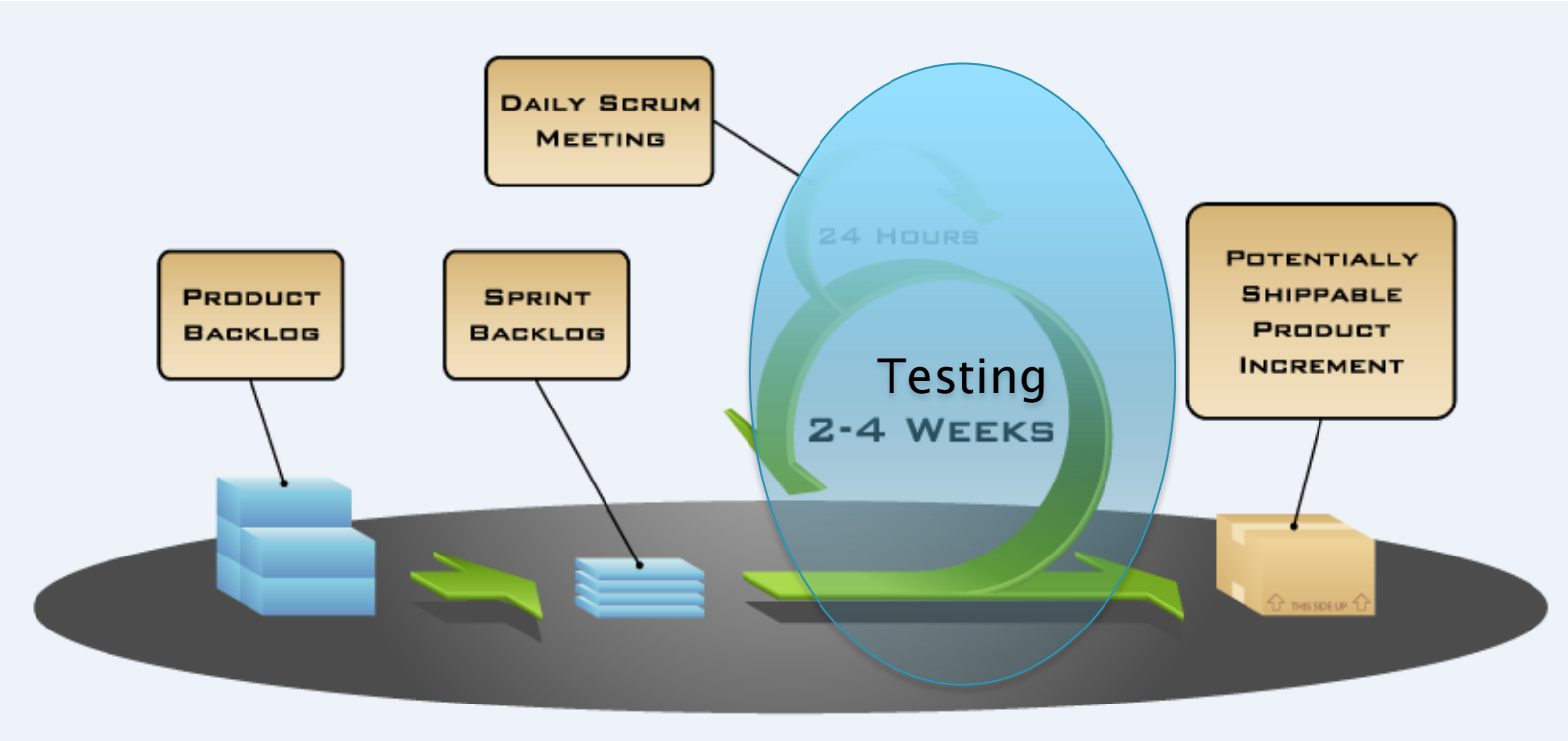
Today

Move a largely creative, “in-head” profession into tangible results with the possibility of fast feedback: Agile



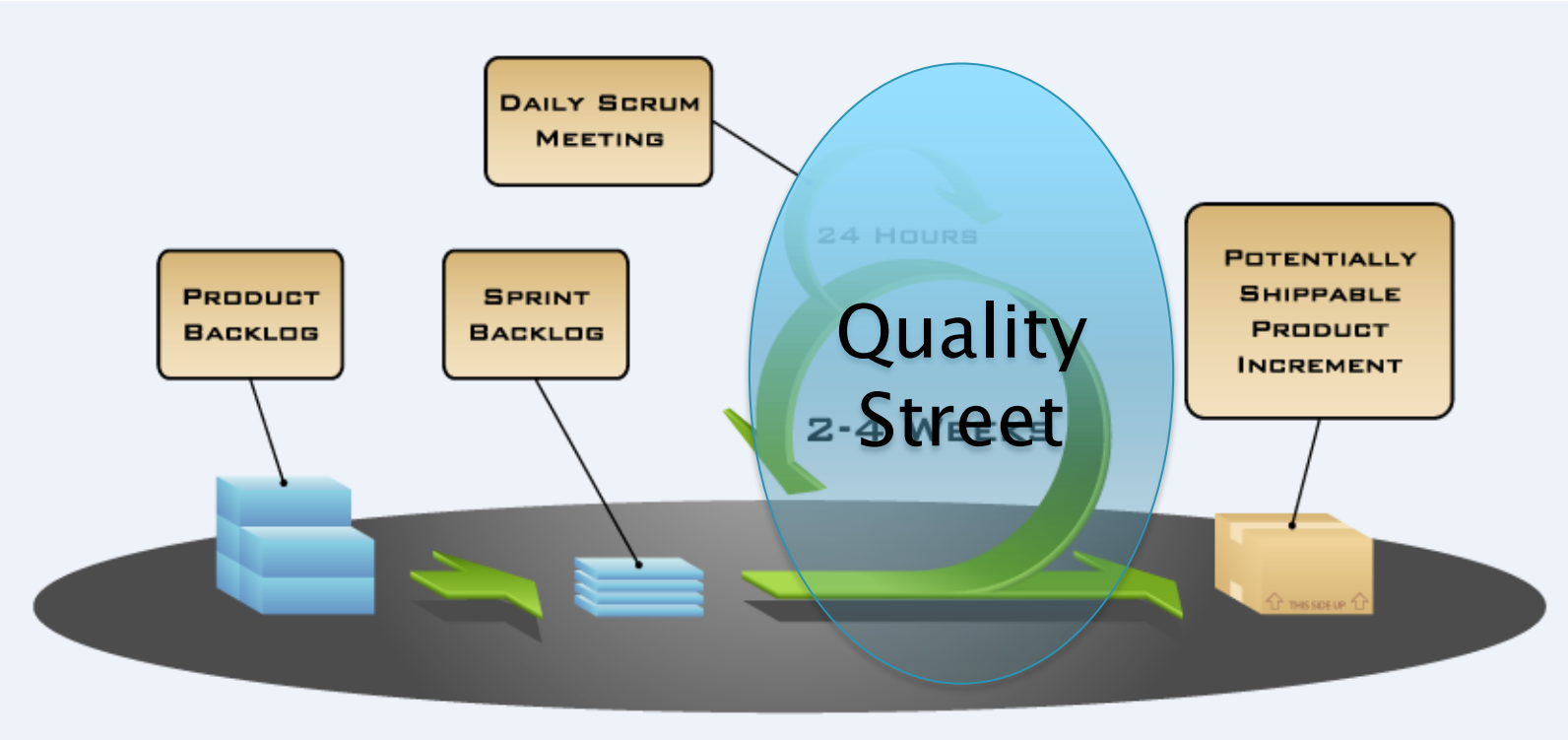
Testing activities

Testing has become integrated with development.

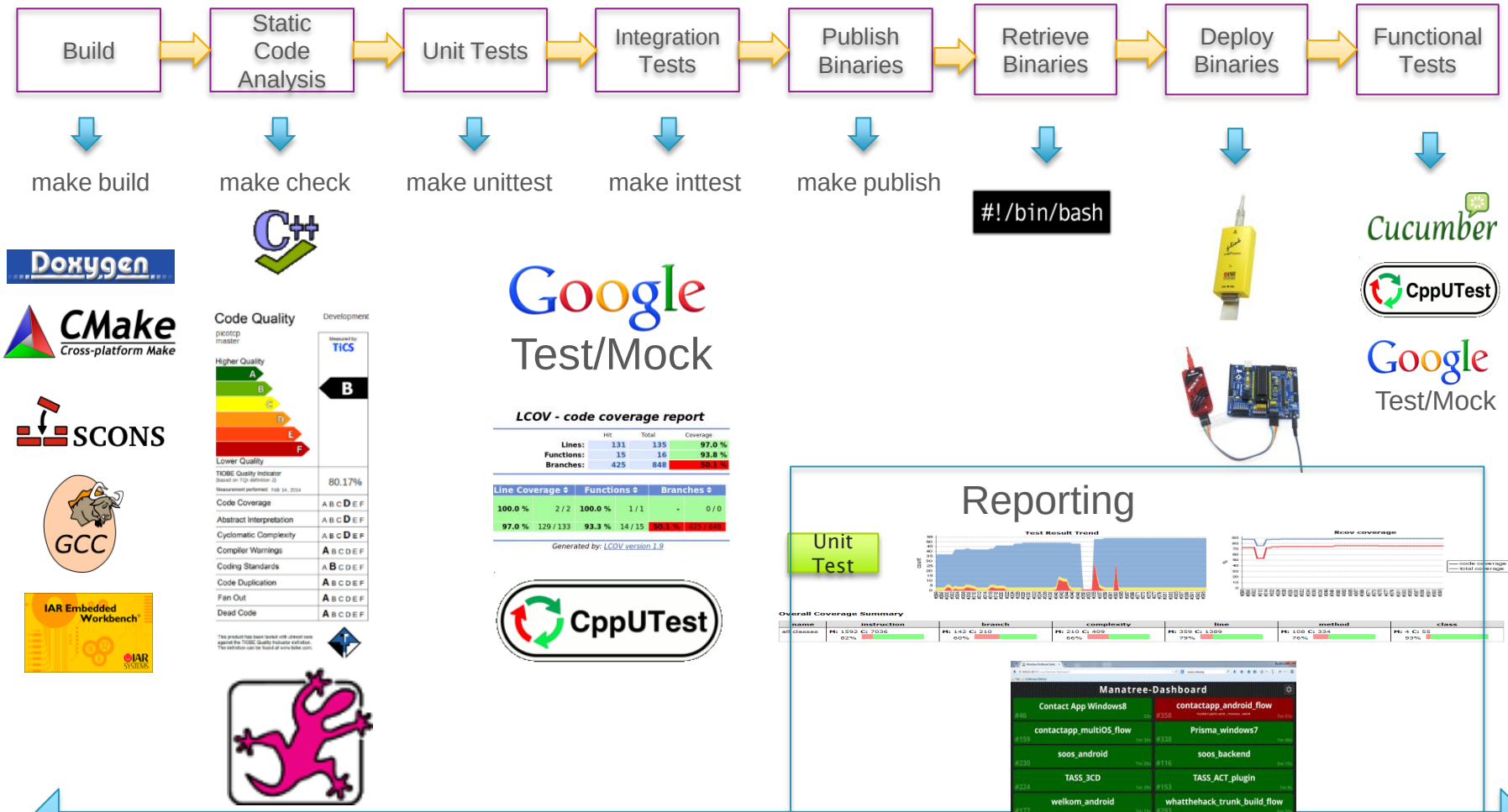


Testing activities

Automate everything: enter the Quality Street.



The Quality Street



ALTRAN

The Quality Street

Build → Static Code Analysis → Unit Tests → Integration Tests → Publish Binaries → Retrieve Binaries → Deploy Binaries → Functional Tests

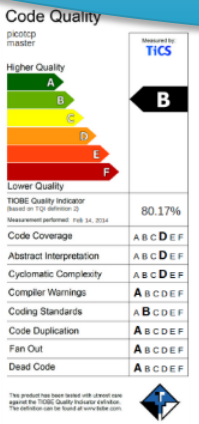
Every step incorporates testing

CMake
Cross-platform Make

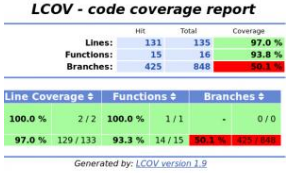
SCONS

GCC

IAR Embedded Workbench



Test/Mock



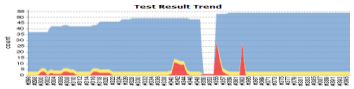
CppUTest

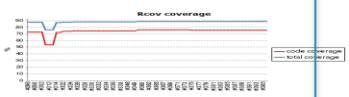
CppUTest

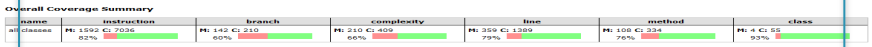
Google Test/Mock

Reporting

Unit Test



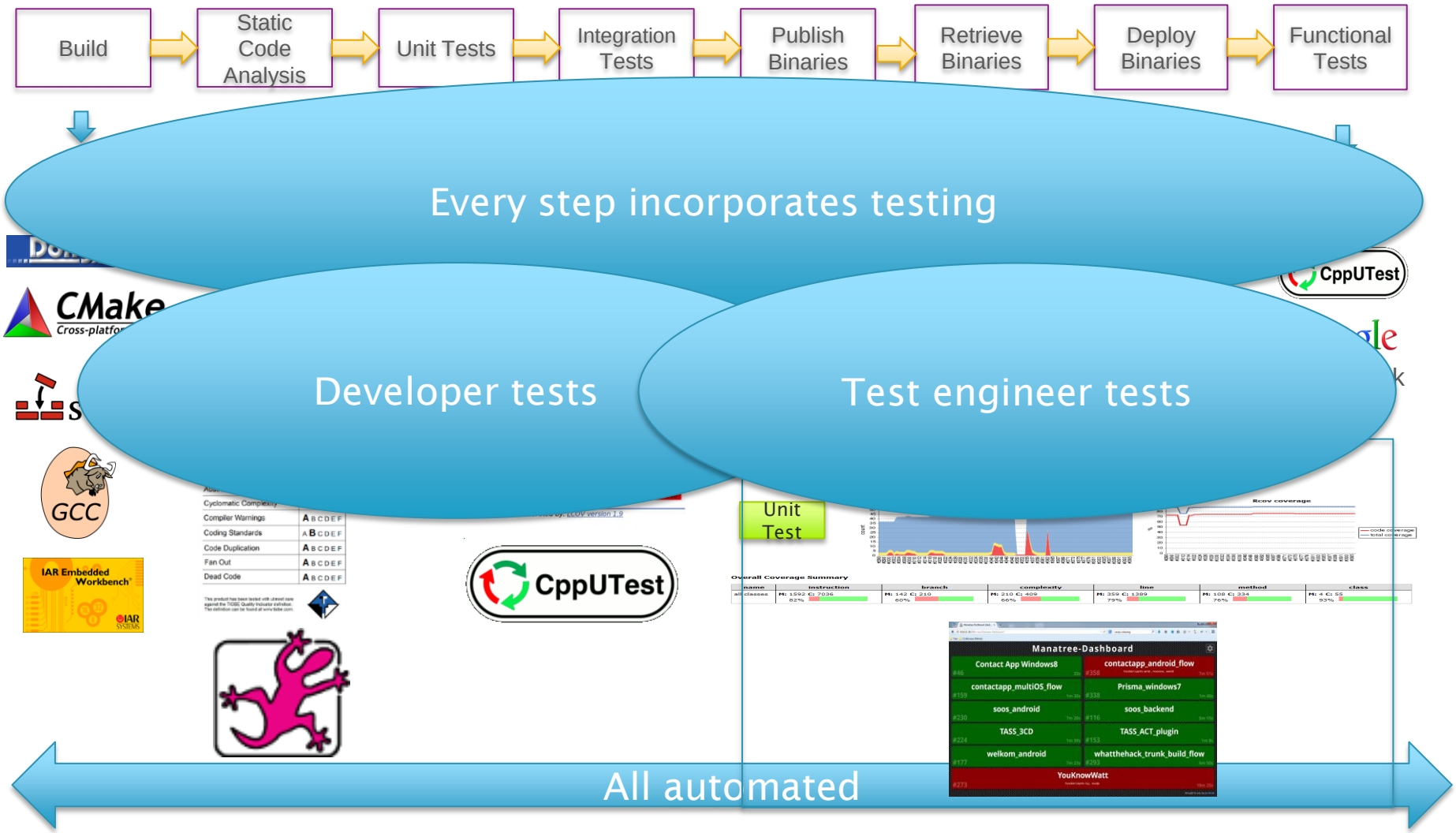




All automated

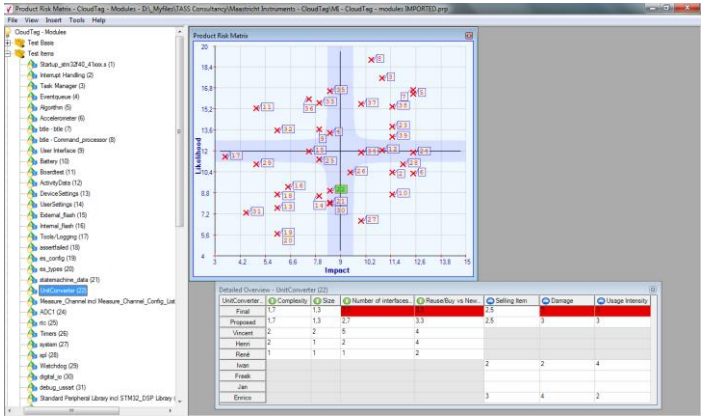
ALTRAN

The Quality Street



ALTRAN

Green-field vs. Legacy



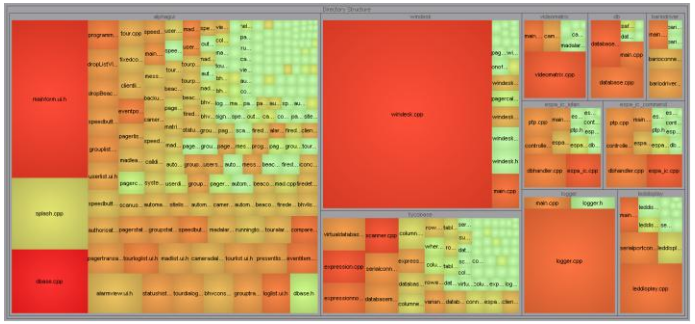
Green-field: new development

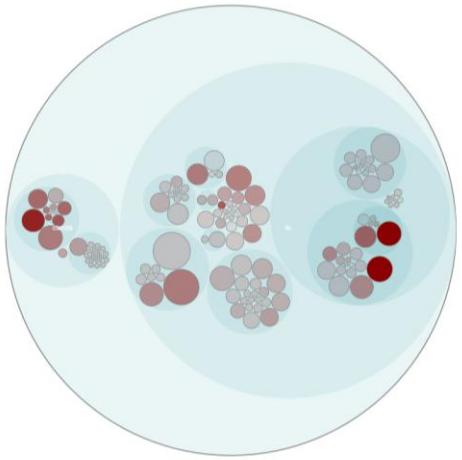
Legacy: existing code base

Where to start?

-Building the code is not a big problem

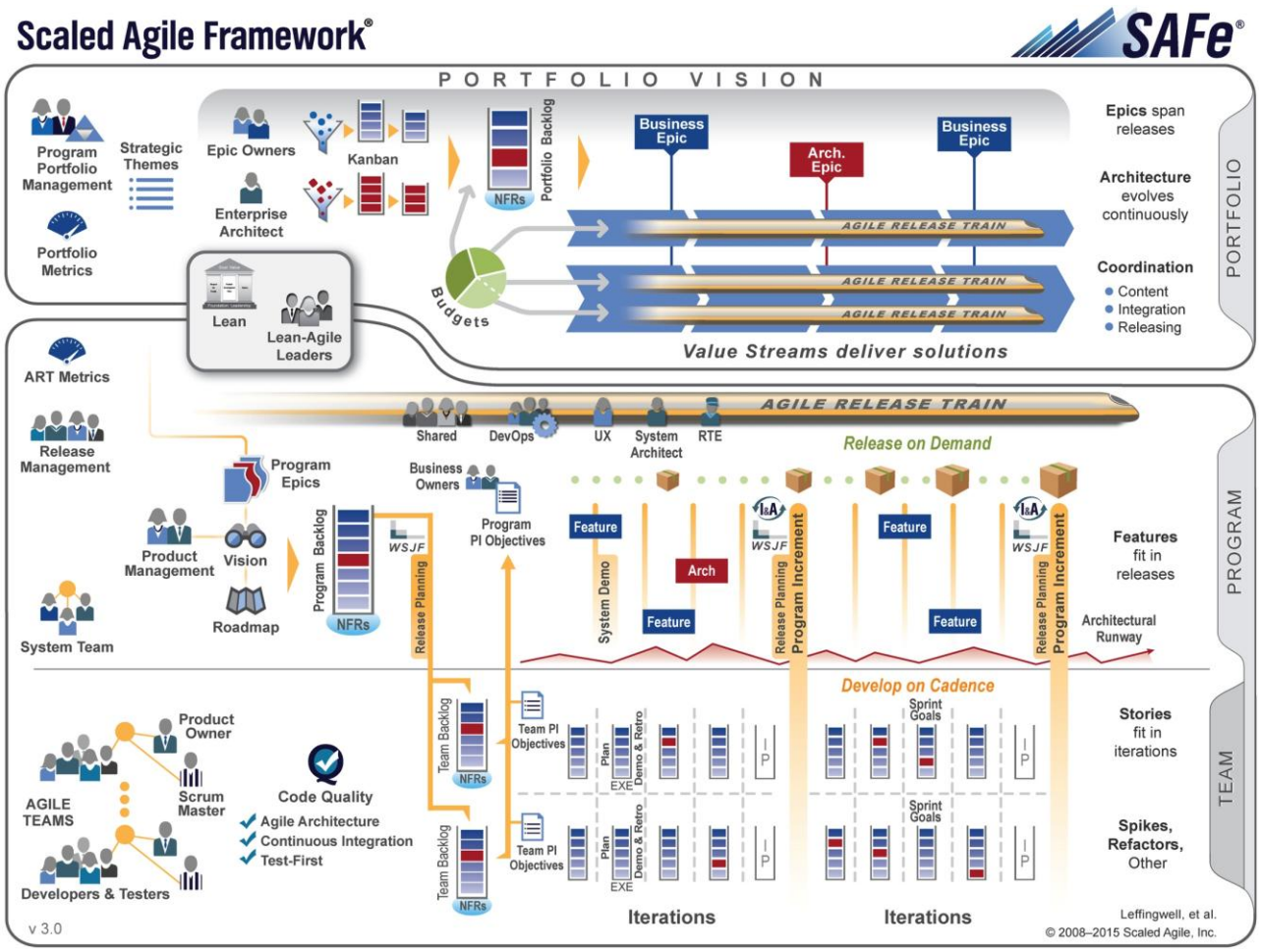
- Static analysis is quickly automated
- For the tests:
    - Use the existing code base as information source:
        - Forensic analysis
        - Risk assessment
        - Field/end user/complaints data



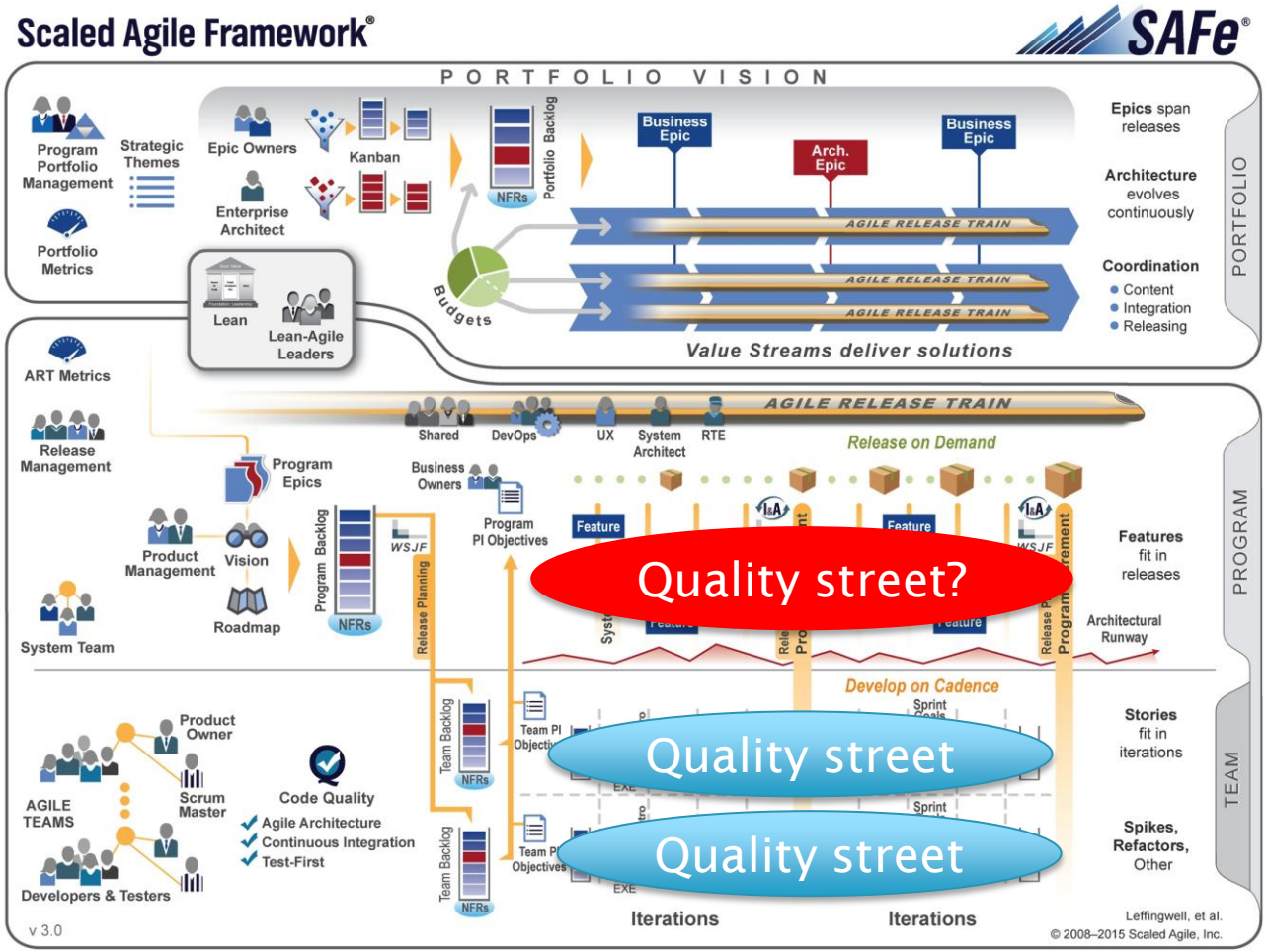


ALTRAN

Today


Scaled Agile Framework®
SAFe®
PORTFOLIO VISION
Program Portfolio Management
Strategic Themes
Portfolio Metrics
Epic Owners
Kanban
Enterprise Architect
NFRs
Portfolio Backlog
Business Epic
Arch. Epic
Business Epic
AGILE RELEASE TRAIN
AGILE RELEASE TRAIN
AGILE RELEASE TRAIN
Budgets
Lean
Lean-Agile Leaders
Value Streams deliver solutions
Epics span releases
Architecture evolves continuously
Coordination
Content
Integration
Releasing
PORTFOLIO
ART Metrics
Release Management
Shared
DevOps
UX
System Architect
RTE
AGILE RELEASE TRAIN
Release on Demand
Program Epics
Business Owners
Program PI Objectives
Product Management
Vision
Roadmap
Program Backlog
NFRs
WSJF
Release Planning
Feature
System Demo
Arch
Feature
Feature
Feature
I&A
Program Increment
Features fit in releases
Architectural Runway
System Team
PROGRAM
Develop on Cadence
Product Owner
Scrum Master
AGILE TEAMS
Developers & Testers
Code Quality
Agile Architecture
Continuous Integration
Test-First
Team Backlog
NFRs
Team PI Objectives
Plan
Demo & Retro
EXE
I
P
Sprint Goals
Stories fit in iterations
Spikes, Refactors, Other
TEAM
Iterations
Iterations
v 3.0
Leffingwell, et al.
© 2008–2015 Scaled Agile, Inc.

Today - tomorrow



What happens to testing?

Bottom layer covered, but…

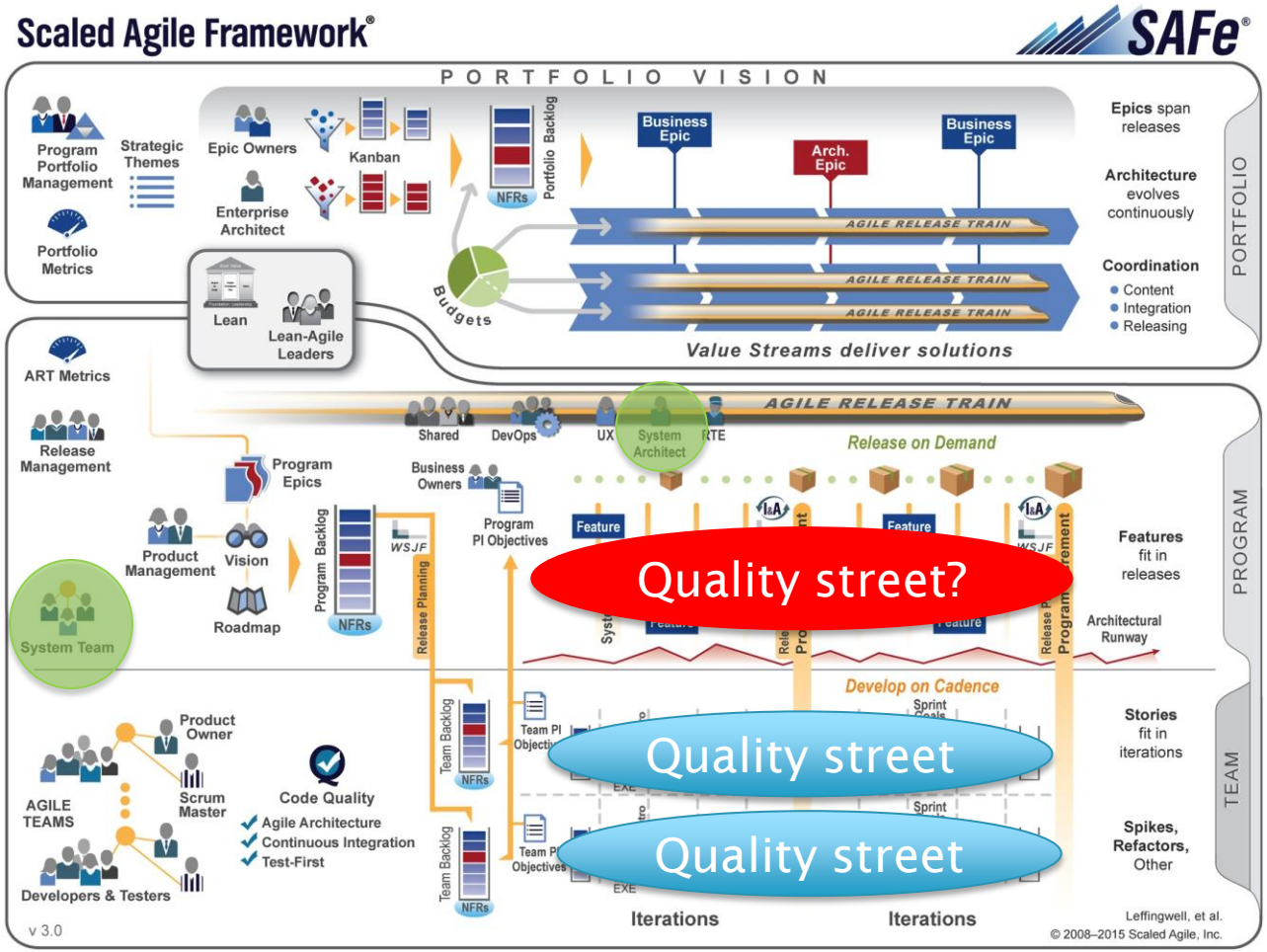
What about systems?

How to synchronise *in-between* releases?

How to enable early/daily testing?

ALTRAN

Today - tomorrow



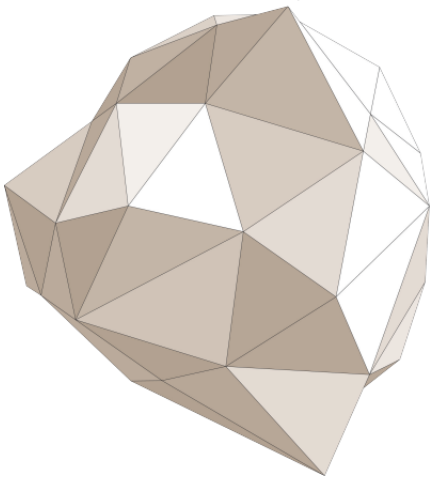
When can the “system team” actually test?

Alignment required between program increments…

I want daily insight at system level!

ALTRAN

INNOVATION MAKERS



altran