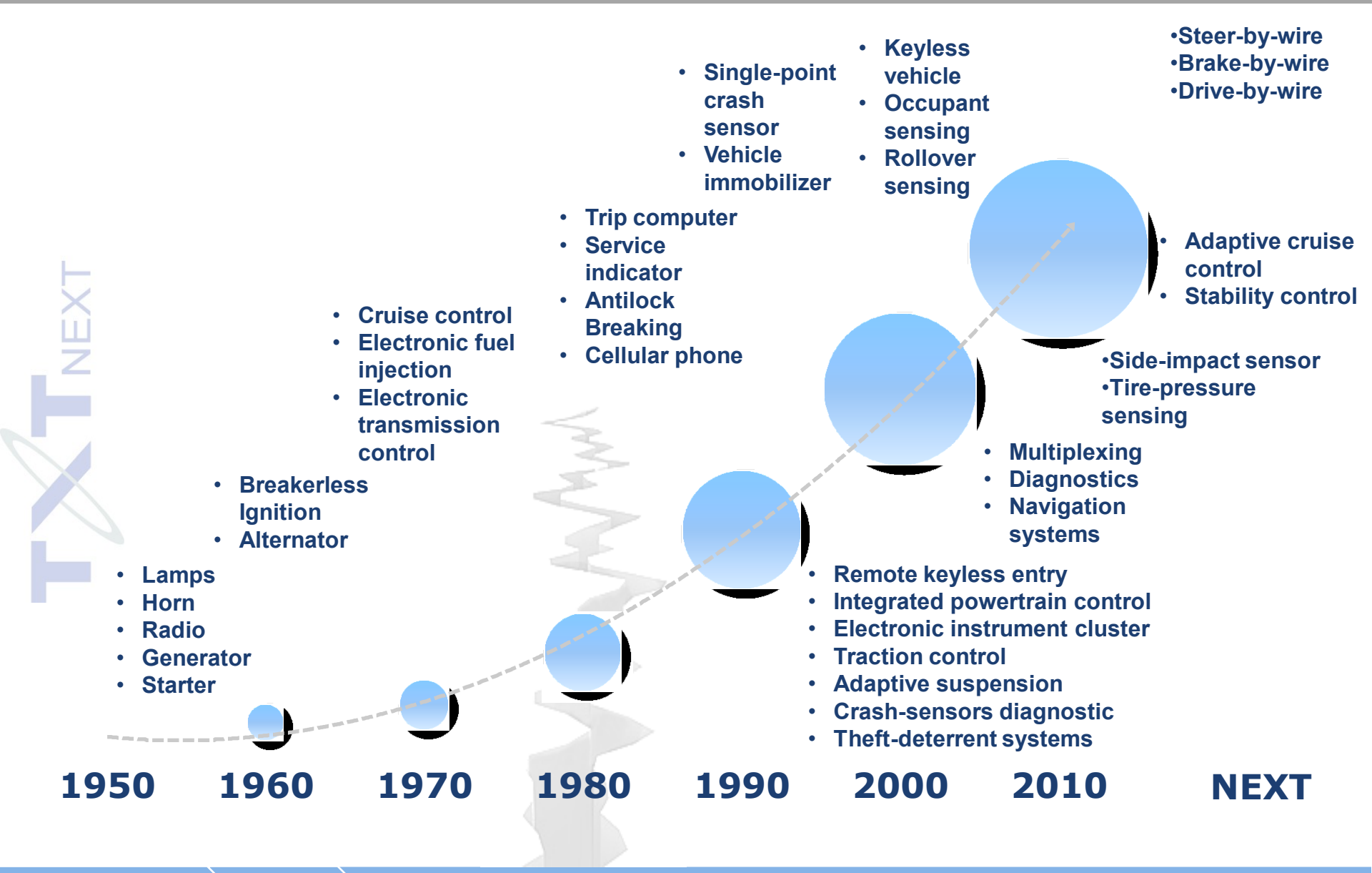
Improve your process, don't forget old lessons, to reach your safety goals.

10 Automotive SPIN Italy Workshop

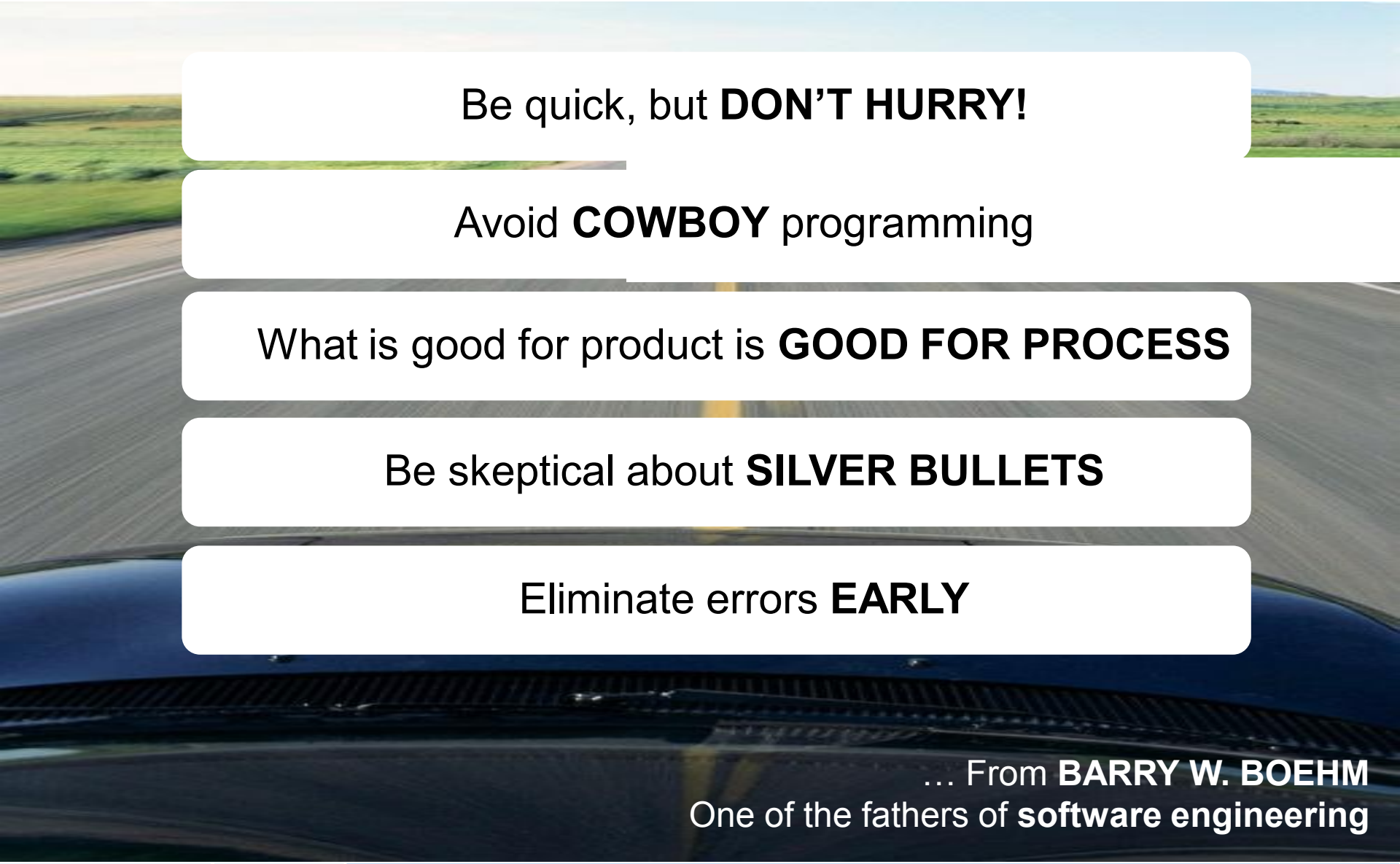
25 October 2012, Milan

Matteo Carlo Crippa

On-board automotive electronics history



Lessons learned



Be quick, but **DON'T HURRY!**

Avoid **COWBOY** programming

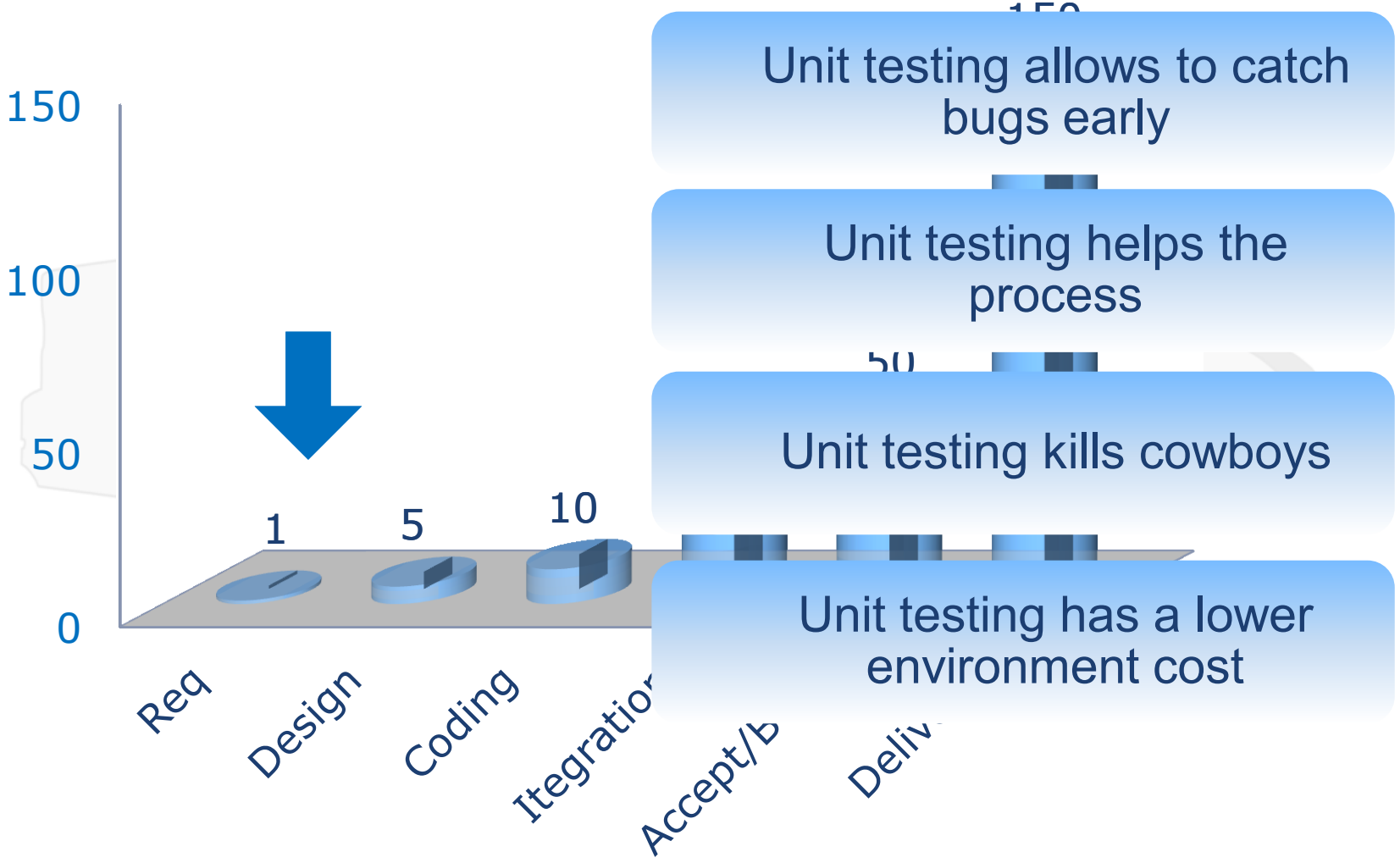
What is good for product is **GOOD FOR PROCESS**

Be skeptical about **SILVER BULLETS**

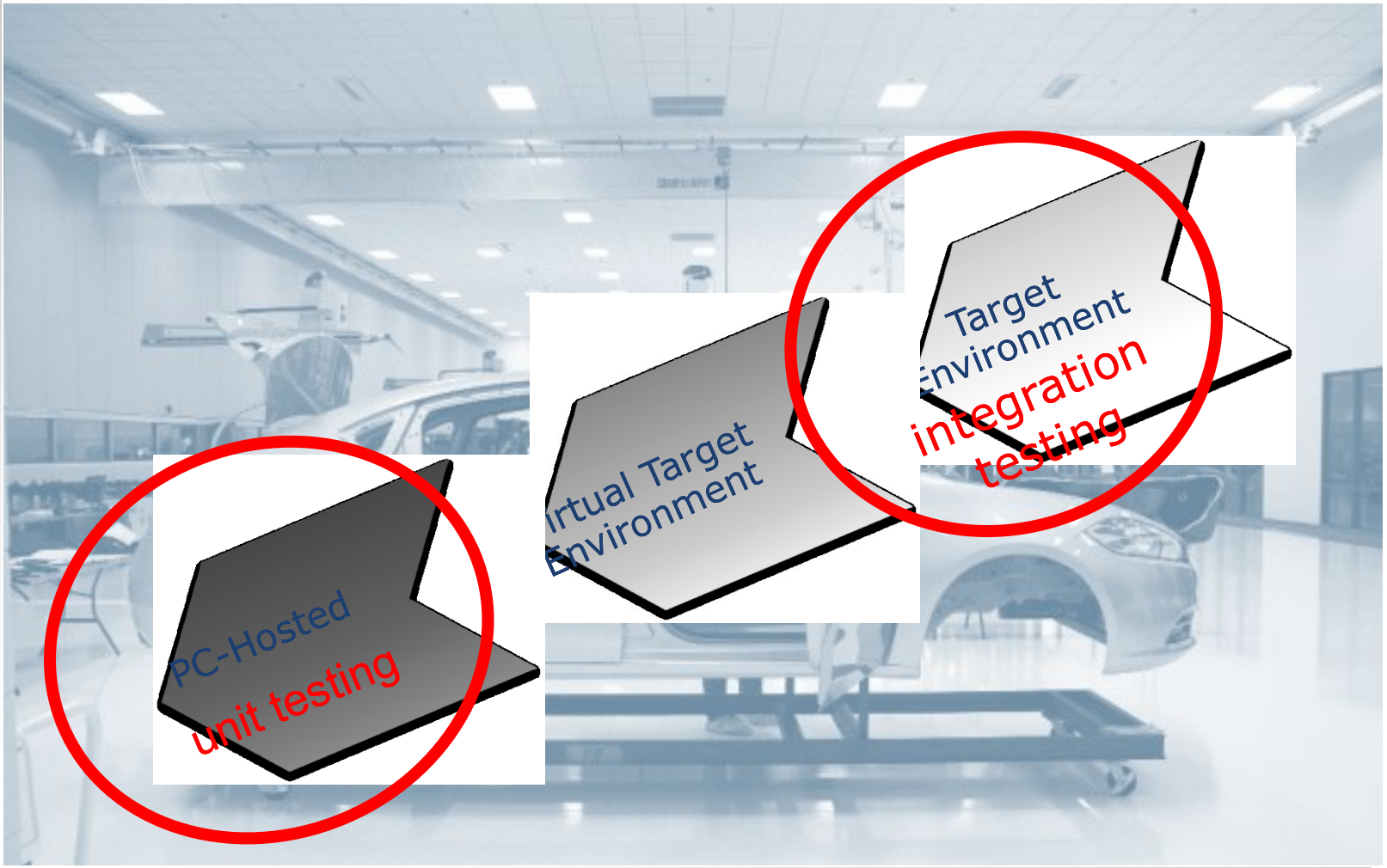
Eliminate errors **EARLY**

… From **BARRY W. BOEHM**
One of the fathers of **software engineering**

Focus on unit testing



Test environment costs



ISO 26262-6 Chapter 9

Requirement-based testing at unit level is central to ISO 26262 compliance, as it is highly recommended for all ASIL levels.

Table 10 - Methods for software unit testing

The standard also makes provisions for testing based on the structure of the code.

Table 12 - Structural coverage metrics at the software unit level

ISO 26262 strongly recommends the usage of code coverage as a metric to measure whether sufficient testing has been performed on a given unit.

Par. 9.4.5

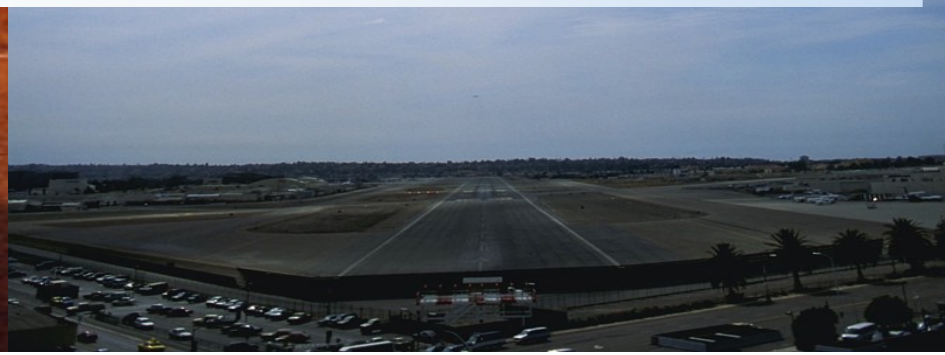
The goal is clearly intended to be 100% coverage based on the coverage criteria selected.

Table 12 - Structural coverage metrics at the software unit level

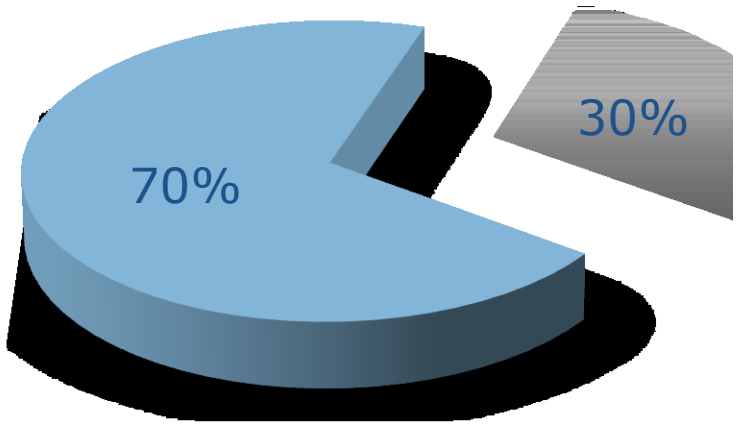
ISO 26262-6 and DO-178B convergence

Looking at ISO 26262-6 we discovered a strict convergence to the aerospace DO-178B.

Methods, Tools and Metrics look very similar



From avionic experience

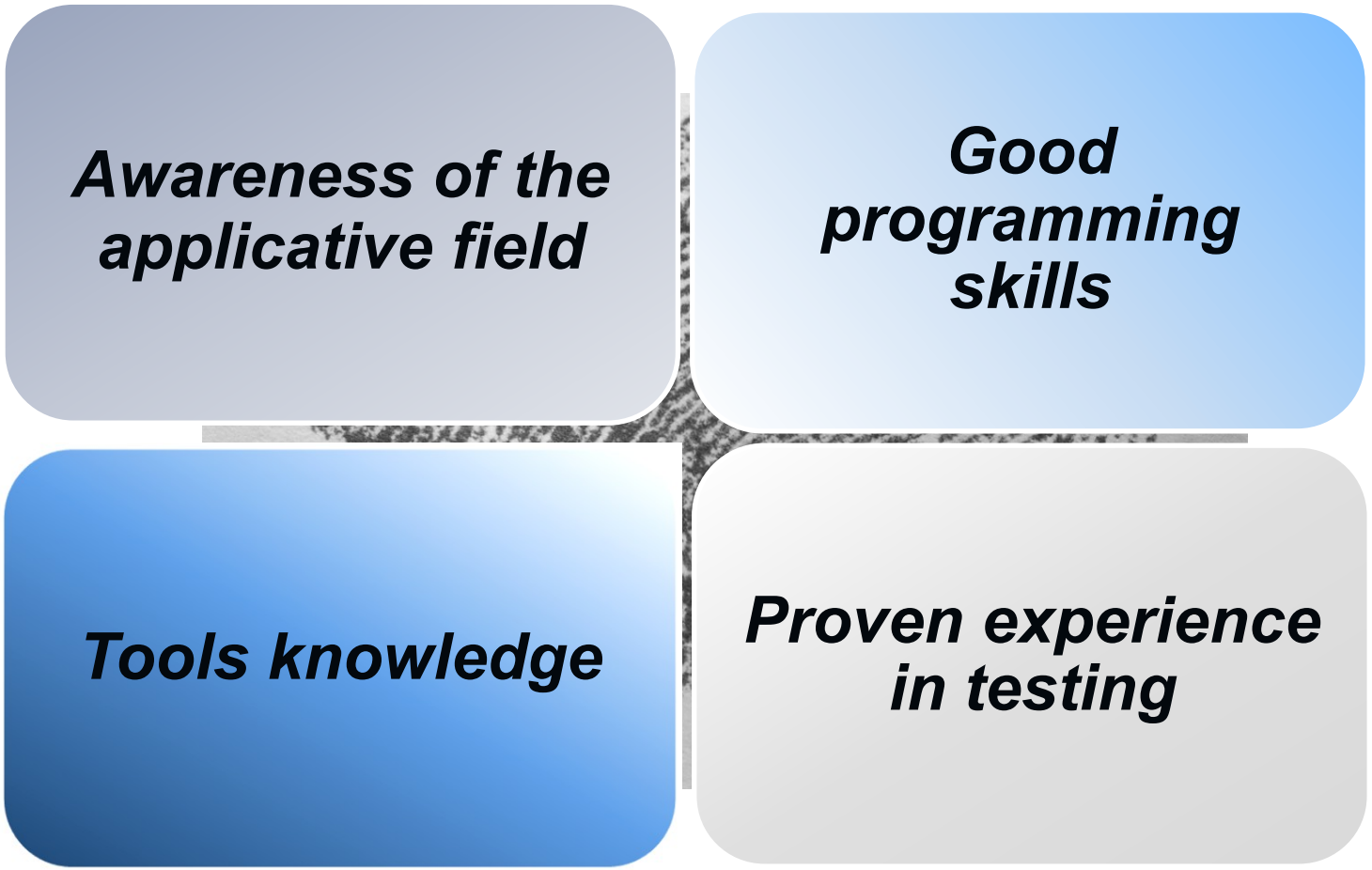


Control your process

Speed up the integration phase

Save up to 30%

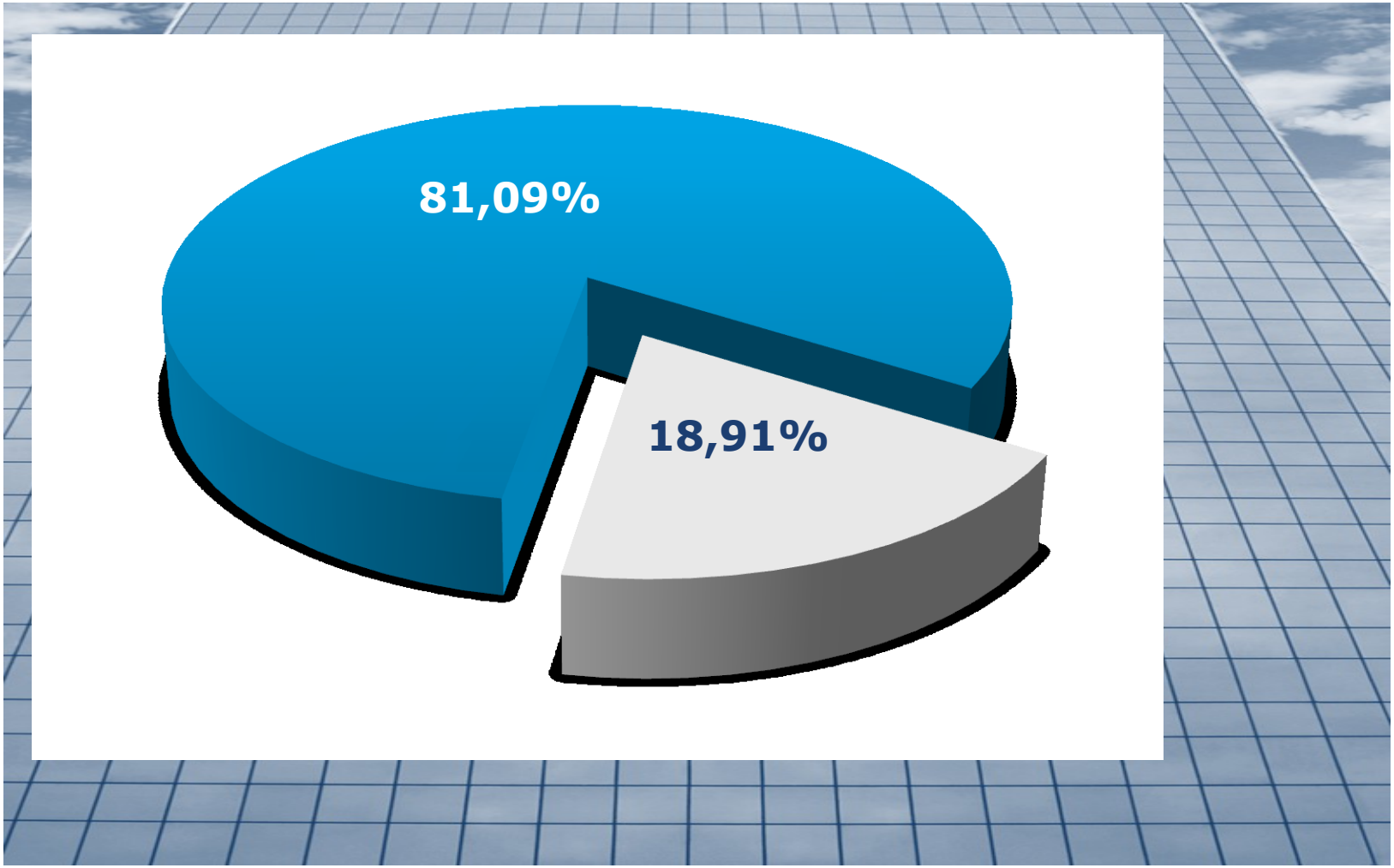
Skills at unit testing



Unit test phase process

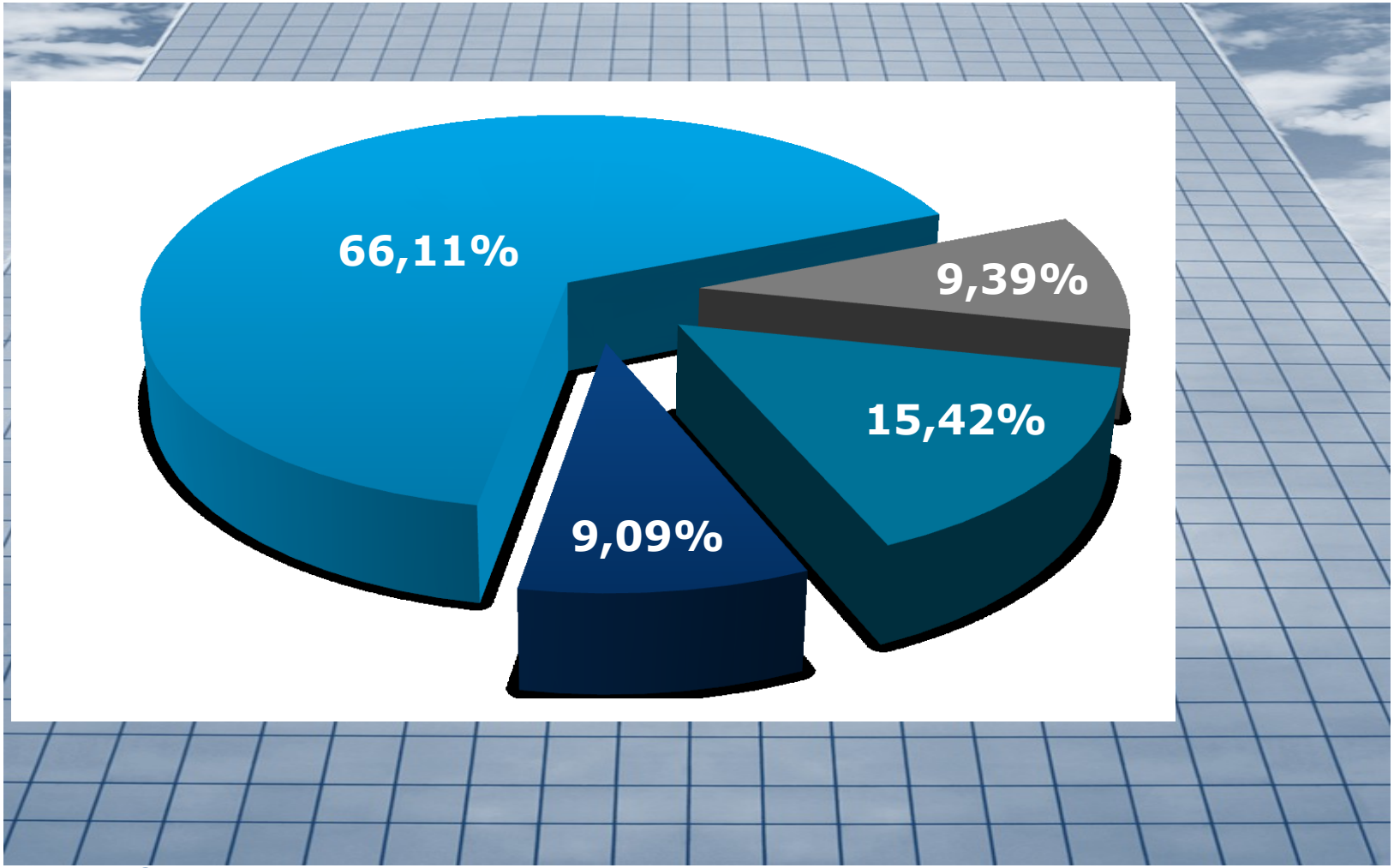
TEST ENVIRONMENT DESCRIPTION	• Scaffolding composition
TEST DESIGN	• Test execution description
TEST TRACEABILITY	• Trace test sequences Vs. software design and requirements
TEST SEQUENCE	• Versioned test sequence with verification vectors
TEST RESULT	• Execution records and results
TEST COVERAGE RESULT	• Test coverage reports and explanations

Unit testing metrics: successful test sequences



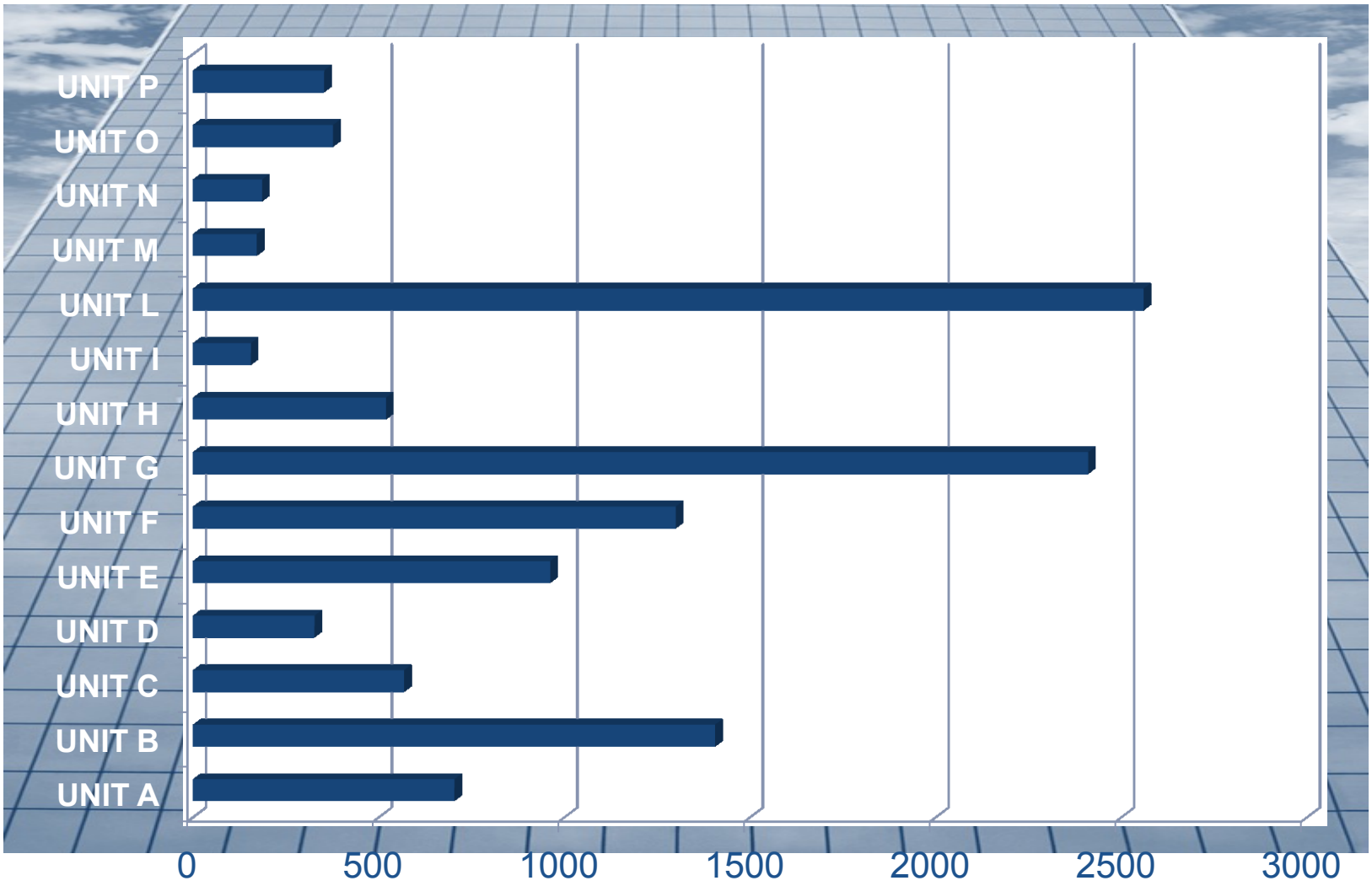
Source: TXT avionic software

Unit testing metrics: error type



Source: TXT avionic software

Unit testing metrics: SLOC/BUG



Source: TXT avionic software

What's more in unit testing

Documentation	Coverage	Traceability	Impact analysis
• Automatic test execution • Automatic document generation	• White box testing • Coverage analysis	• Complete traceability	• From coverage code impact figure

Conclusions

Unit testing is highly recommended from ISO 26262

Corrective actions, for bugs found in unit testing phase, have a lower cost

Unit testing decreases time and risk for integration phase

The effort of unit testing can help the whole process providing metrics and measures

Unit testing is reusable