

Software Quality Models

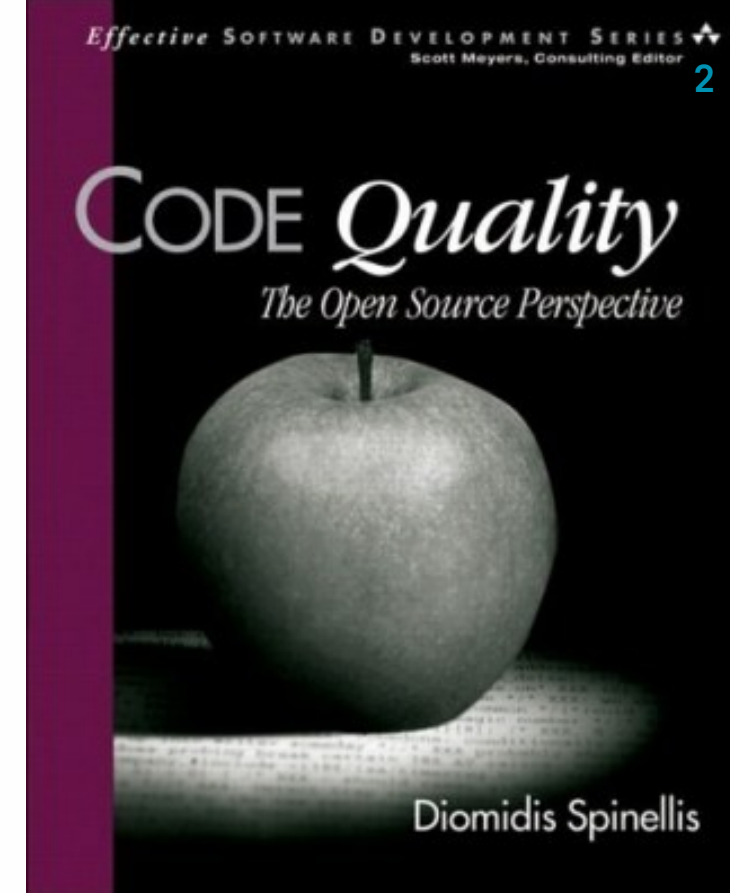
Software Evolution – L3T1

Dr. Vadim Zaytsev aka @grammarware, February 2021

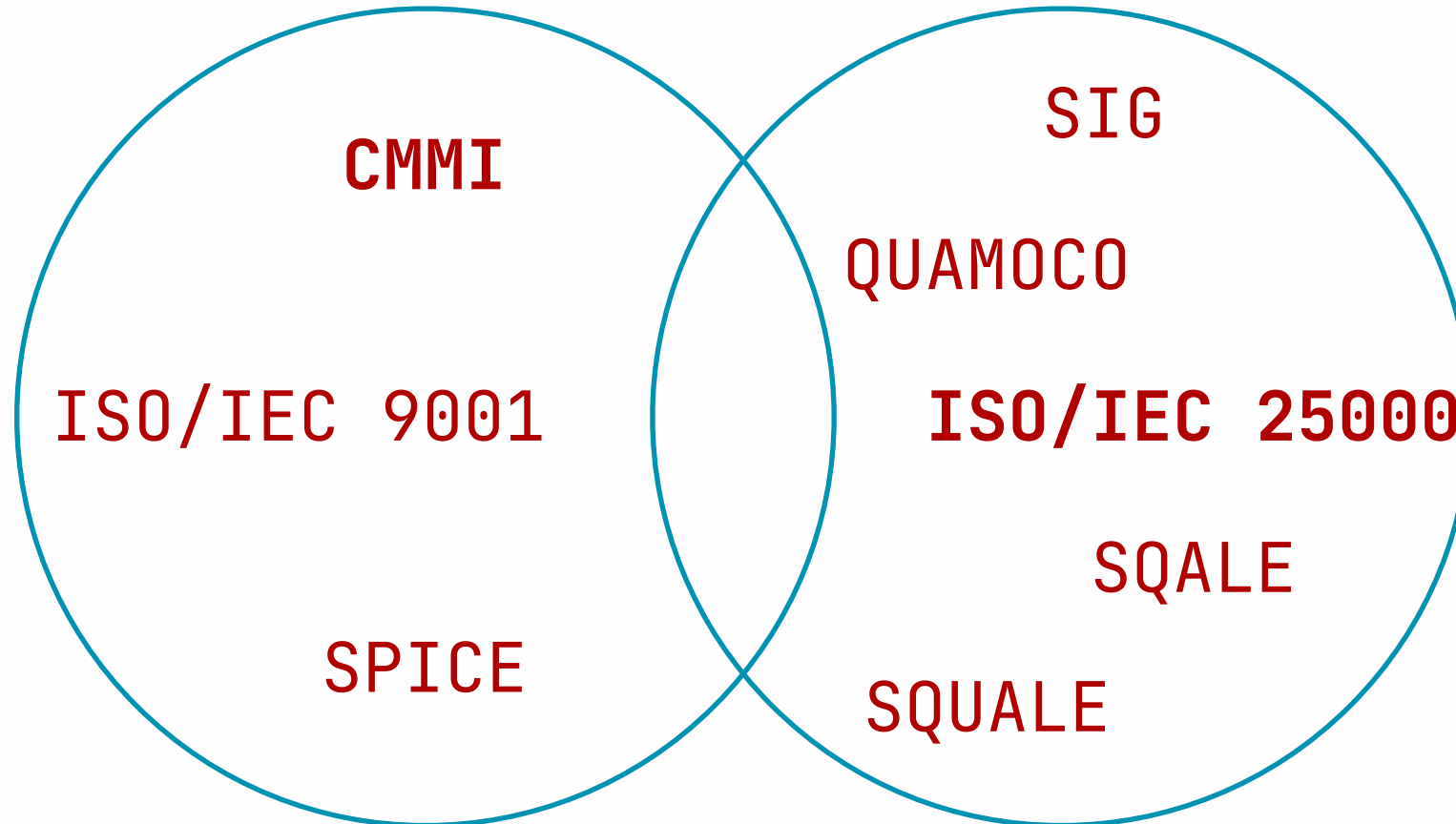


Code Quality

- Quality in use
 - bugs that manifest
- External quality
 - benchmarked
- Internal quality
 - examine, not run
- Process quality
 - accident?



Quality Models



Portability

Maintainability

Security

Suitability

System Quality

Reliability

Efficiency

Compatibility

Usability

Adaptability
Installability
Replaceability

Modularity
Reusability
Analysability
Modifiability
Testability

Confidentiality
Integrity
Accountability
Authenticity

Completeness
Correctness
Appropriateness

System Quality

Maturity
Availability
Fault tolerance
Recoverability

Time behaviour
Resource utilisation
Capacity

Co-existence
Interoperability

Recognisability
Learnability
Aesthetics
Accessibility

Capability Maturity Model Integration

5: Optimising

improving with experiments

4: Measured

measured, controlled, quantitatively managed

3: Defined

proactive, organisation-specific, tailorable

2: Managed

reactive, project-specific

1: Initial

unpredictable, reactive,
poorly controllable

Conclusion

- Product quality models
 - e.g., [SQuaRE](#)
- Process quality models
 - e.g., [CMMI](#)
- Q&A Sessions @ Canvas
 - ⇒ v.zaytsev@utwente.nl
 - ⇒ <https://discord.gg/n7VQAPNBPD>