



# SQALE Your Future Software Costs

Software Quantitative Assessment based on Lifecycle Expectations

## Quality Counts!

Today's industry involves many decisions where information about the software quality is vital to guarantee a successful future.

The software quality must be managed; to know what you have, what you want and how to get there!

Software quality includes both *external* and *internal* aspects.

- External quality is defined by functionality, usability, amount of faults and performance – usually addressed and under control
- Internal quality is about testability, reusability, maintainability, changeability, etc – often overlooked since it is less visible, harder to measure and has a long term impact on the life cycle cost

## SQALE

SQALE objectively evaluates internal quality in order to understand the life-cycle costs. SQALE delivers true value by:

- Assessing and comparing software quality between different version, subcontractors, applications, subsystems, etc.
- Strengthening own quality management processes to define and follow up quality goals.

### Det Norske Veritas

Det Norske Veritas, DNV, is a foundation with headquarter in Norway and an international organisation with 9000 employees serving customers in more than 100 countries.

DNV in Sweden offers Consultant services, Certification, Classification and Training.

DNV Certifies and Classifies systems, companies, organizations and individuals within IT, development, quality, energy and environment according to standards such as ISO 9000, CMMI and SPICE.

DNV Consultants help customers to increase their performance through better processes and less risk by establishing modern methods and standards for Software Engineering, Risk Management and Quality.

# SQALE brings added value

Code analysis is not new but SQALE delivers results through:

- A Quality Model based on a life cycle perspective – quality characteristics are defined and linked to the software lifecycle
- An Analysis Model based on a unique aggregation method – SQALE aggregates all static analysis result in quality indexes based on remediation cost

SQALE gives an effective quality analysis and shows remediation cost, valuable in numerous situations.

Refactoring decisions	Control outsourced product quality
Improvement activities follow-up	Evaluate Open Source projects
Due diligence	Choose between these two similar products

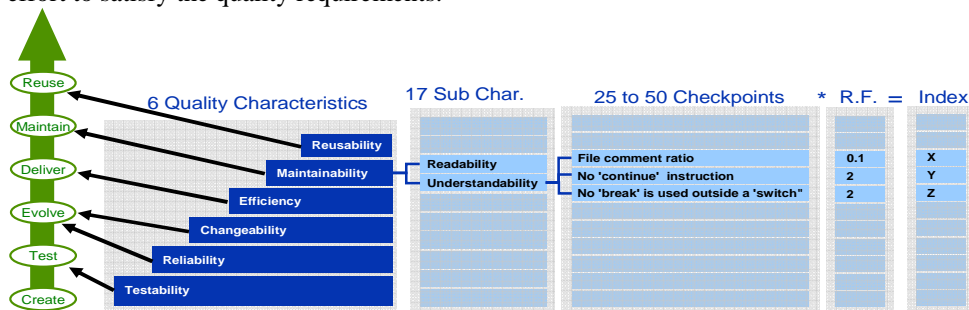
Examples of SQALE usage

## Quality Model

The Quality Model is made up by six Quality Characteristics, layered based on the software life cycle. Associated to each Quality Characteristic there is a set of Sub-Characteristics. Each Sub-Characteristic is defined through at least one Quality Checkpoint. Each Quality Check point sets the quality requirement to be satisfied.

## Analysis Model

Analyzing the software quality is measuring the distance to achieve the quality targets. To measure this distance, a Remedy Factor is introduced to compensate for the specific remedy for each Checkpoint. This results in an index that corresponds to the remediation effort to satisfy the quality requirements.

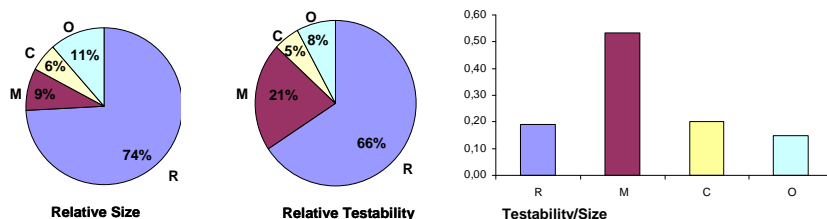


SQALE Quality and Analysis Model

## Reporting

The SQALE reporting is compact, graphical. Different stakeholders receive relevant information based on the same data e.g:

- Management gets feedback on Quality Characteristics and Life Cycles Costs
- Developers gets feedback on adherence to Quality Checkpoints



Comparing the non compliance distribution related to Testability between **Reused**, **Modified**, newly **Created**, and **Outsourced** source code files.

## SQALE in Brief

SQALE was developed by DNV in 2006 based on ISO 9126 and other quality work by Boehm, McCall et al. While ISO 9126 covers both external and internal quality aspects SQALE is focusing on the internal aspects.

## Experiences

DNV has supported several organizations using SQALE

## Telecom case

(C/C++, 4,7 MLOC)

Joint process & product audit to evaluate two sites with common products.

## Automotive case

(C++, 4,5 MLOC)

Joint product & process evaluation on one of their subcontractor's site

Support to build C-code quality model that will become a standard requirement for all subcontractors.

## Energy Case

(Java, 0,5 MLOC)

Development of their Quality Model

Support to develop internal platform that automatically evaluates applications against Quality Model

## References & Further reading

ISO, International Organization for Standardization, «9126-1:2001, Software engineering – Product quality, Part 1: Quality model», 2001

E.A. Karlsson, Software Reuse – A holistic Approach, New York: John Wiley Ltd. 1995

Boehm, B. W., Brown, J. R., Kaspar, H., Lipow, M., McLeod, G., and Merrit, M., Characteristics of Software Quality, North Holland, 1978

McCall, J. A., Richards, P. K., and Walters, G.F., Factors in Software Quality, The National Technical Information Service, 1977

Letouzey&Coq, The "SQALE" Models for Assessing the Quality of Software Source Code, Whitepaper, 2009

## Links

[www.detnorskeveritas.se/SQALE](http://www.detnorskeveritas.se/SQALE)

[www.sqale.org](http://www.sqale.org)