The SQALE Method for Managing Technical Debt

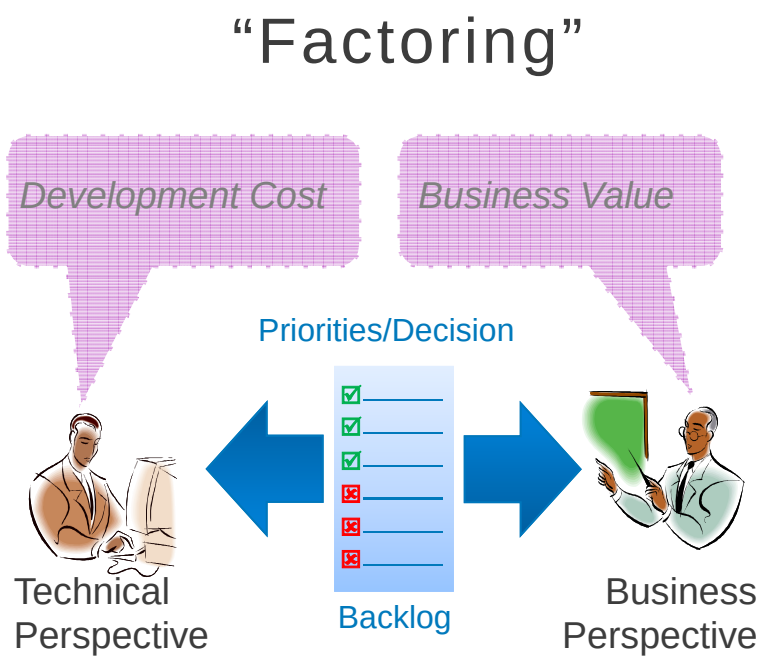
Introduction

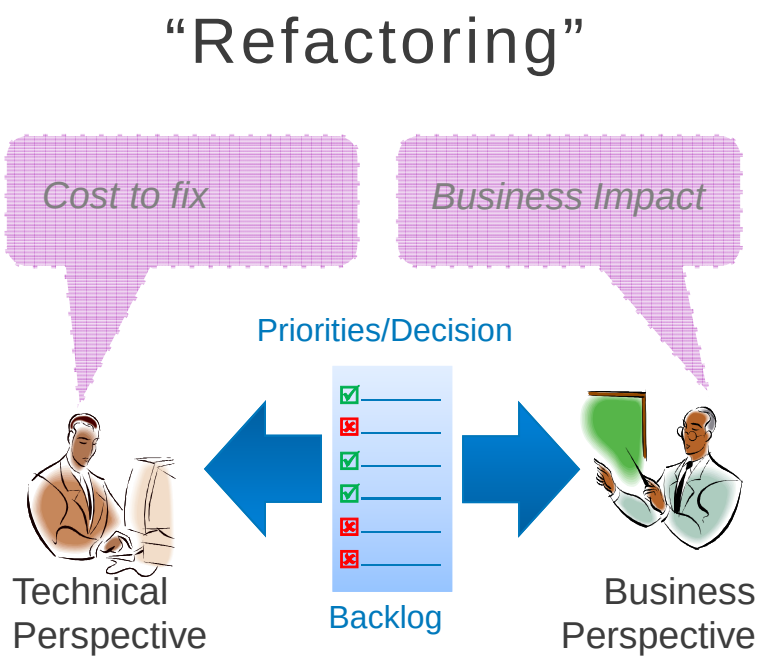
Jean-Louis Letouzey

The SQALE method

- A meta-method, a framework for managing Technical Debt
- **Open source** (www.sqale.org), no royalty
- Tool independent (as today: 4 tools)
- A very large number of users

How to prioritize?

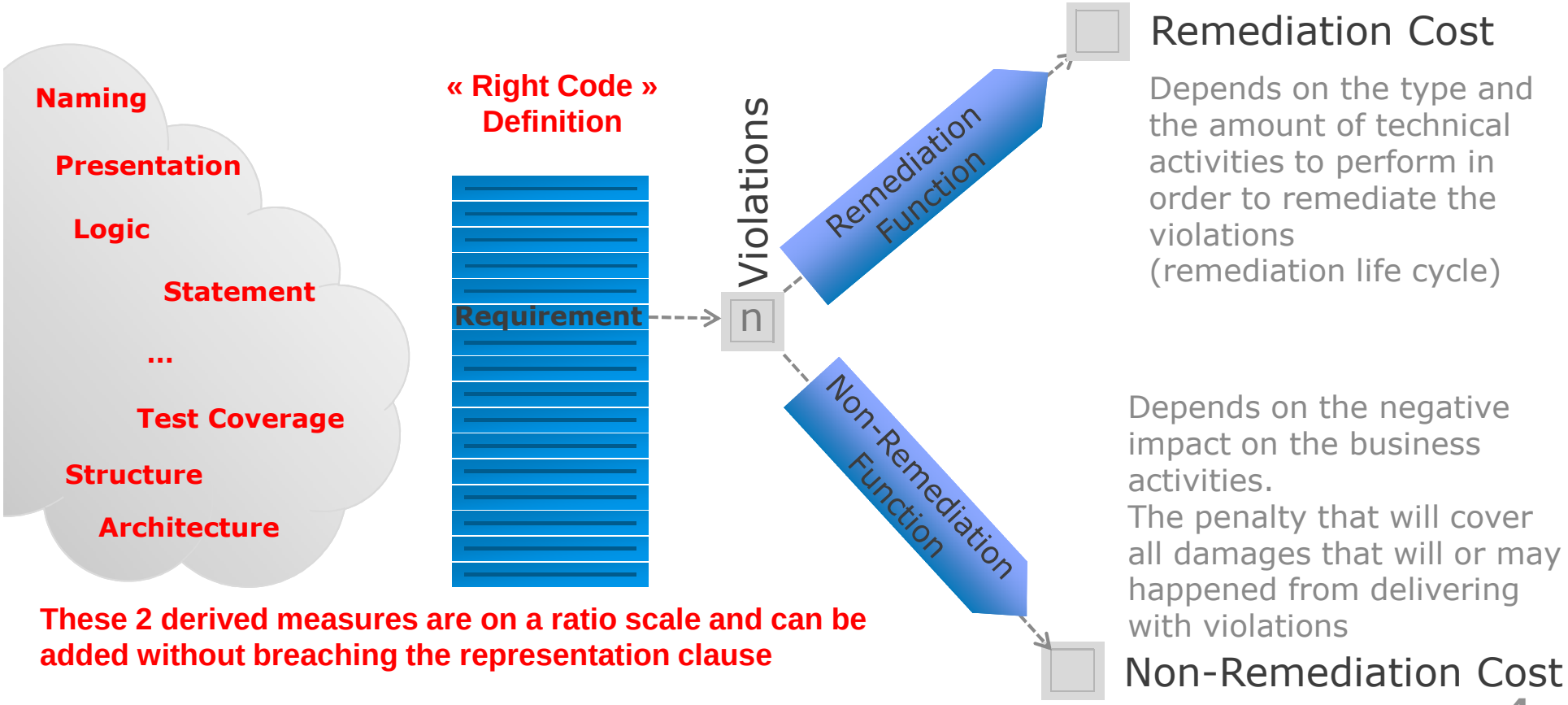




- Major aspects
  - Dev cost
  - Business Value
  - Dependencies

SQALE: 2 Estimation Models

- Estimation models transform tool findings in costs
  - One for the Technical Perspective > Technical Debt (principal)
  - One for the Business Perspective > Business Impact (interest)

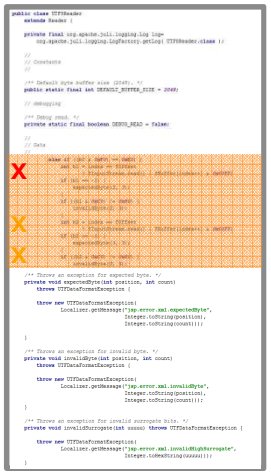


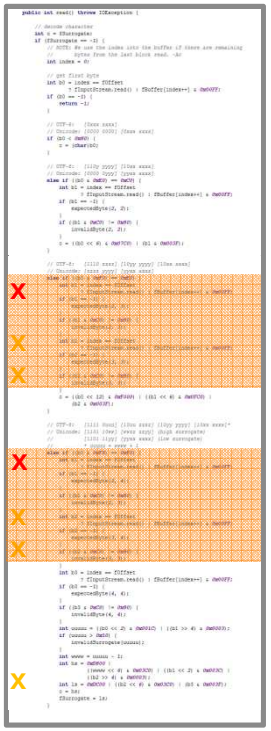
These 2 derived measures are on a ratio scale and can be added without breaching the representation clause

Optimised improvement roadmap

4.382 issues! Details for 2 files
In what order should I make corrections?

Rule	Impact	Violations	
Methods should not be too complex (v(G<15)	High (500)	1	
No Duplicated block	High (500)	3	
Magic number should not be used	Major (50)	7	x
Source code should be correctly indented	Minor (1)	187	
if/else/for/while/do statements should always use curly braces	Very High (5,000)	3	x





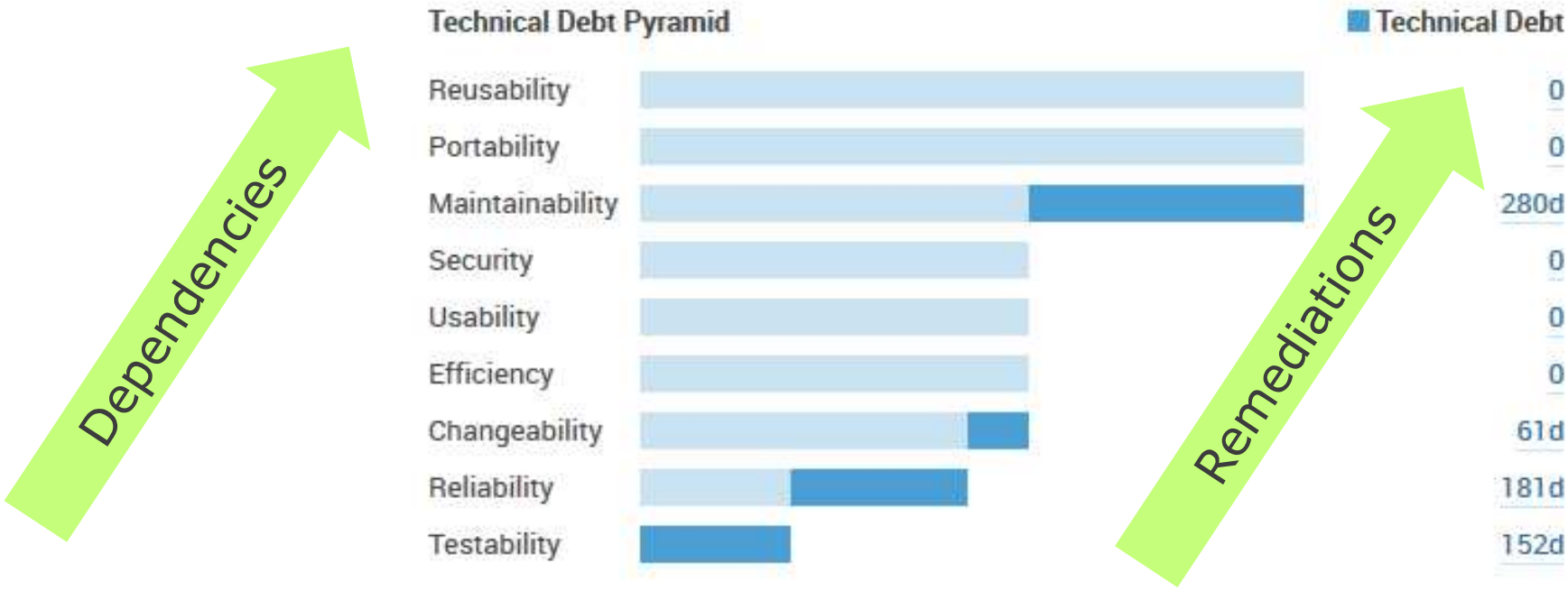
Prioritization depends on

- Refactoring Costs
- Potential Impacts
- Issue Dependencies

Paying back your Technique Debt - 1

SQALE supports 3 strategies

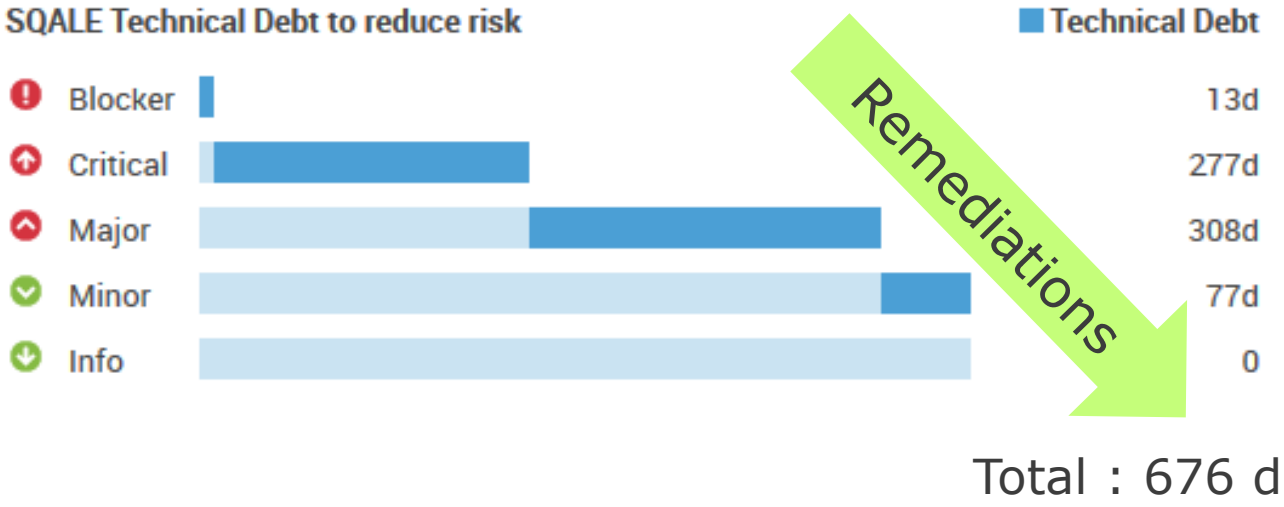
- The SQALE pyramid defines a **logical** priority (technical dependencies) for remediation actions. This is a technical perspective



Total : 676 d

Paying back your Technique Debt - 2

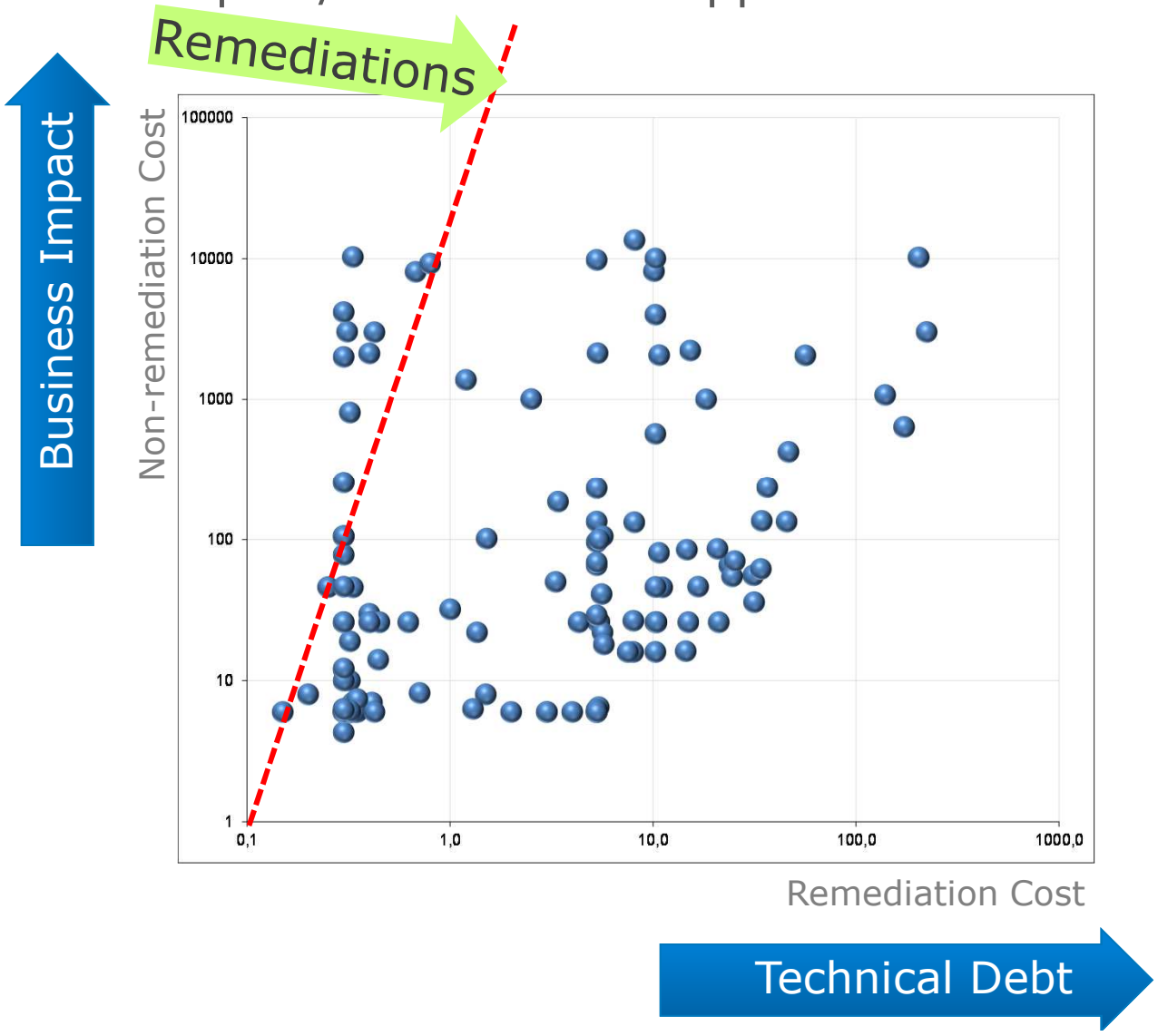
- The business perspective upon « Severity », « Impact »



Total : 676 d

Paying back your Technique Debt - 3

- Use the impact/cost ratio to support the "ROI strategy"

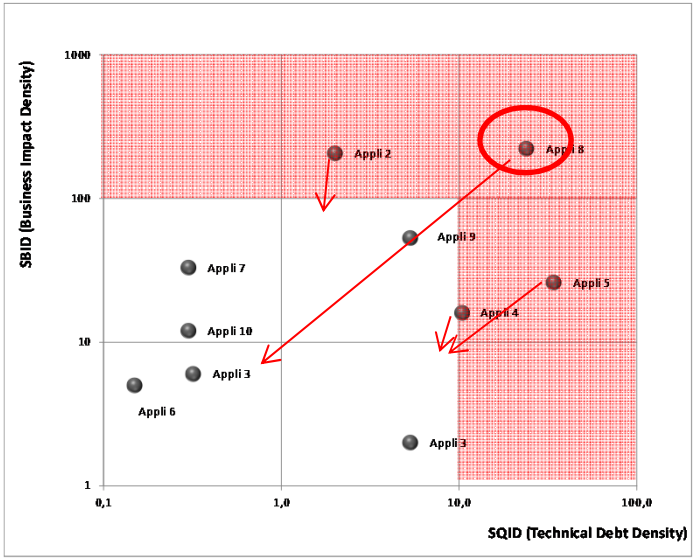


Managing TD at Portfolio level

- Because of history, software entropy, the quality of your portfolio will never be as good as you may expect.
- You will never have all the time and budget to improve all your portfolio at “A” level. **It is a question of compromise**
- A 3 step Process/Strategy

Focus on major Gaps

Operational Analysis vs **Code analysis results**



1. Establish an inventory of your portfolio and identify expected quality level for each of your applications
2. Compare their amount and type of Technical Debt against your expectation
3. Focus energy on highest gaps

How to deploy?

- Estimation models:
  - No standard for that
  - Tools default values are not precise in all contexts

- Organize workshops with experts to build their models based on their local assumptions
- Configure the tool according to the organization model

Examples

- **Assumption example:**
  - H4: Remediation Cost covers: coding, unit testing development and debug, and regression testing update.
  - H5: Developers have coding experience. They are not novices, nor are they expert coders but they know how to use their development environment effectively.
  - H7: Developers or the development team owns the code they work on. The developer understands what he/she is programming (or will get a quick support from other team members).

- **Impact levels example:**

Impact level	Typical **Potential** **Business impact of non Remediation**	Detail/ Examples	Relative Business impact
Very High	May result in bug in production if the line is executed	Potential division by zero, Dangerous cast, NPE, missing logic without commented justification	5000
High	Major additional workload for reproducing, locating and fixing a bug. Efficiency decrease	Untestable code. Unprecise error handling. Non optimal statement, Query.	500
Medium	Minor additional workload for implementing a new feature. No reliability impact on production. Impact is on ownership cost	Ununderstandable code. Coupled code	50
Low	No significant additional workload incured by delivering a non remediation. Can just be fixed later	Bad indentation. Naming rule violation	1

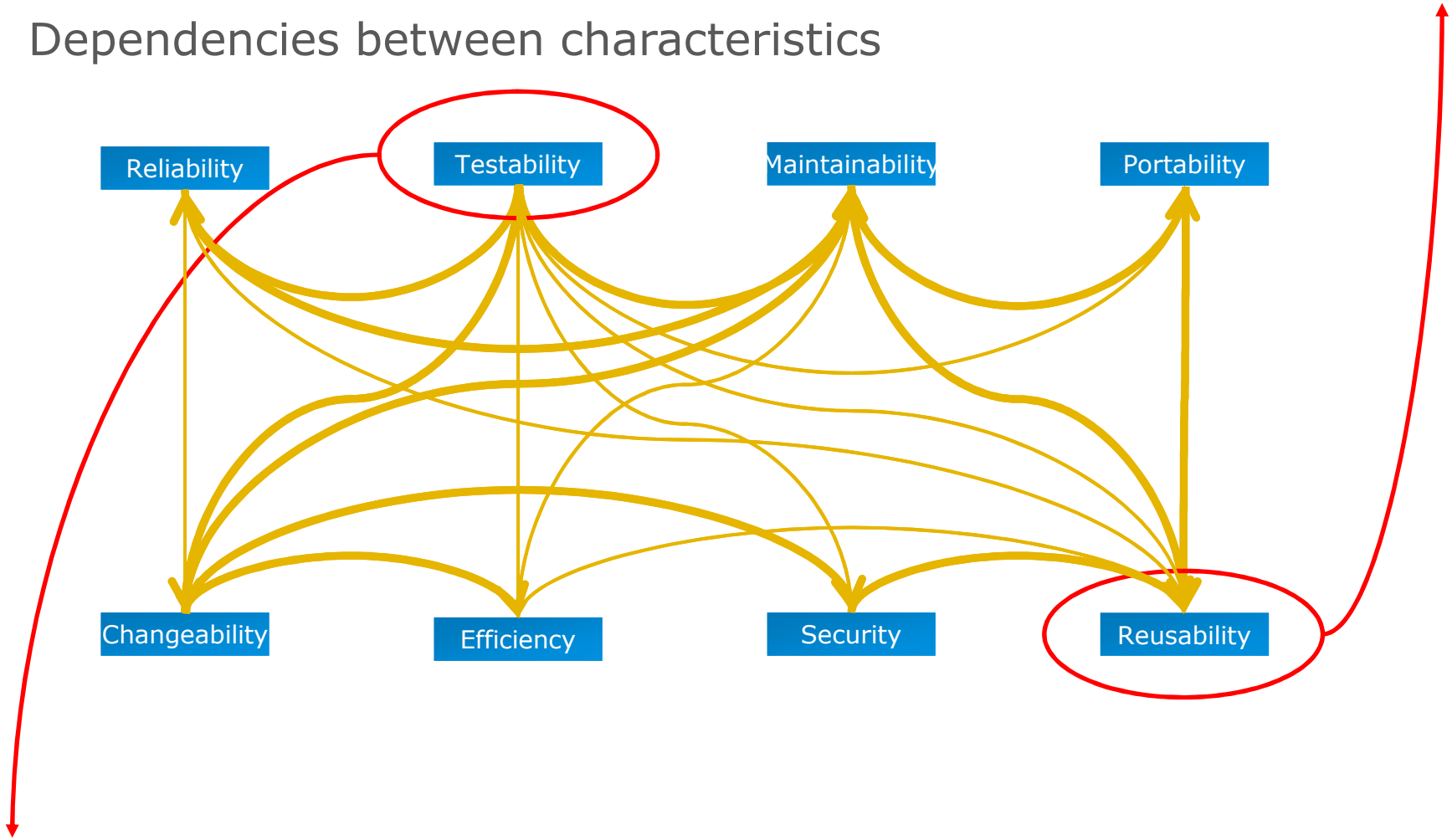
THANK YOU

QUESTIONS

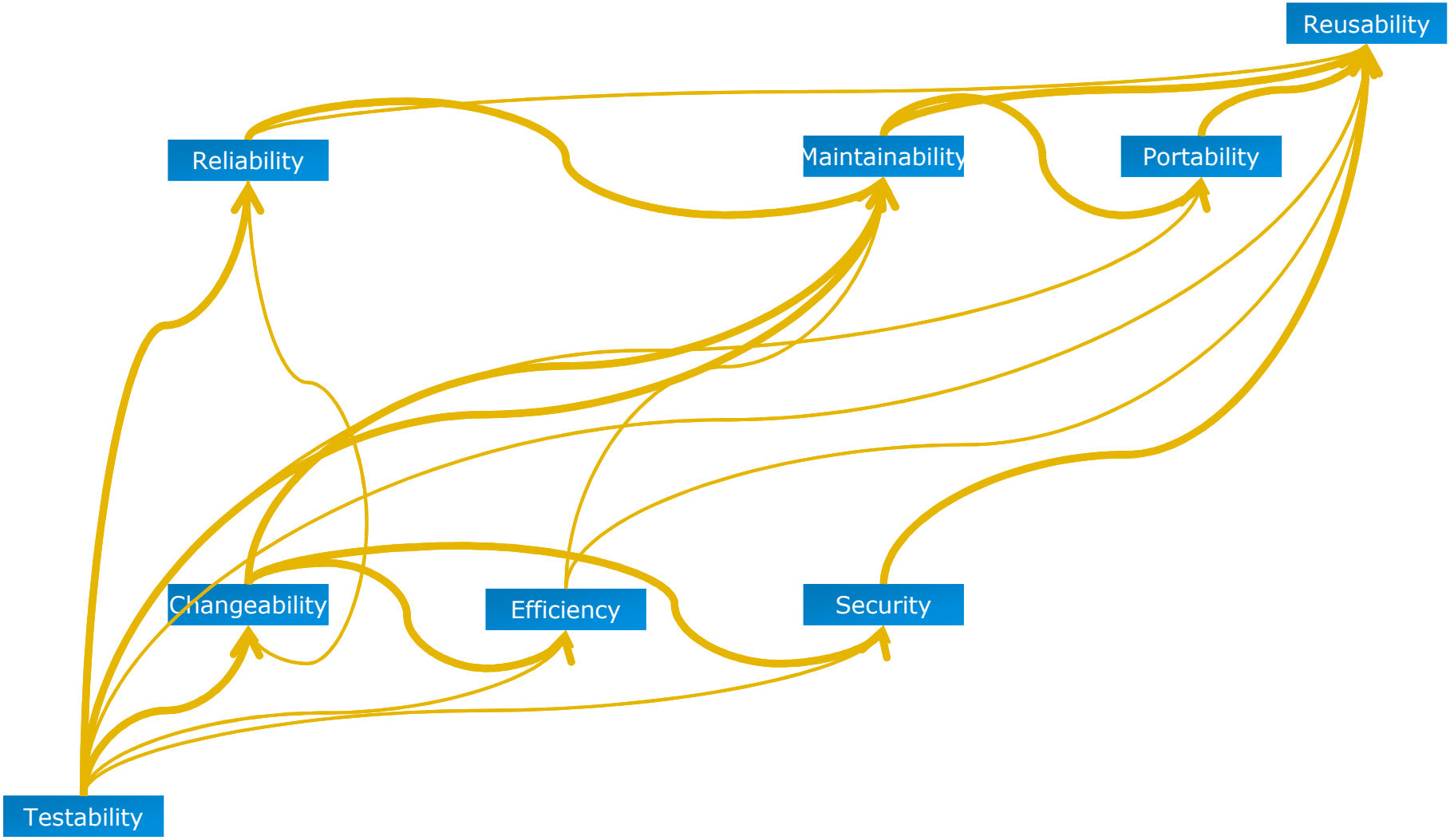
Addendum

SQALE Quality Model: Order of the Characteristics

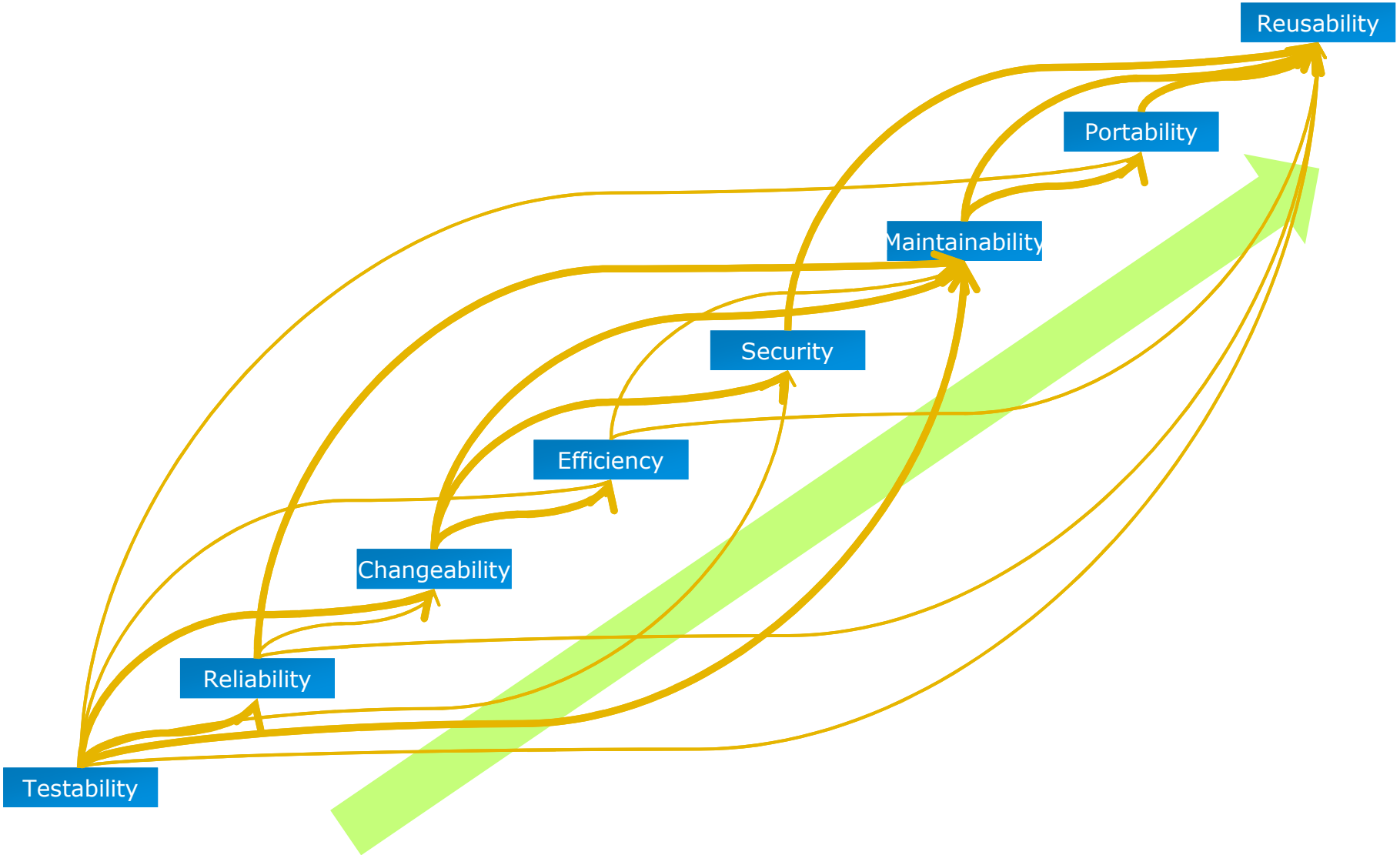
- Dependencies between characteristics



SQALE Quality Model: Order of the Characteristics

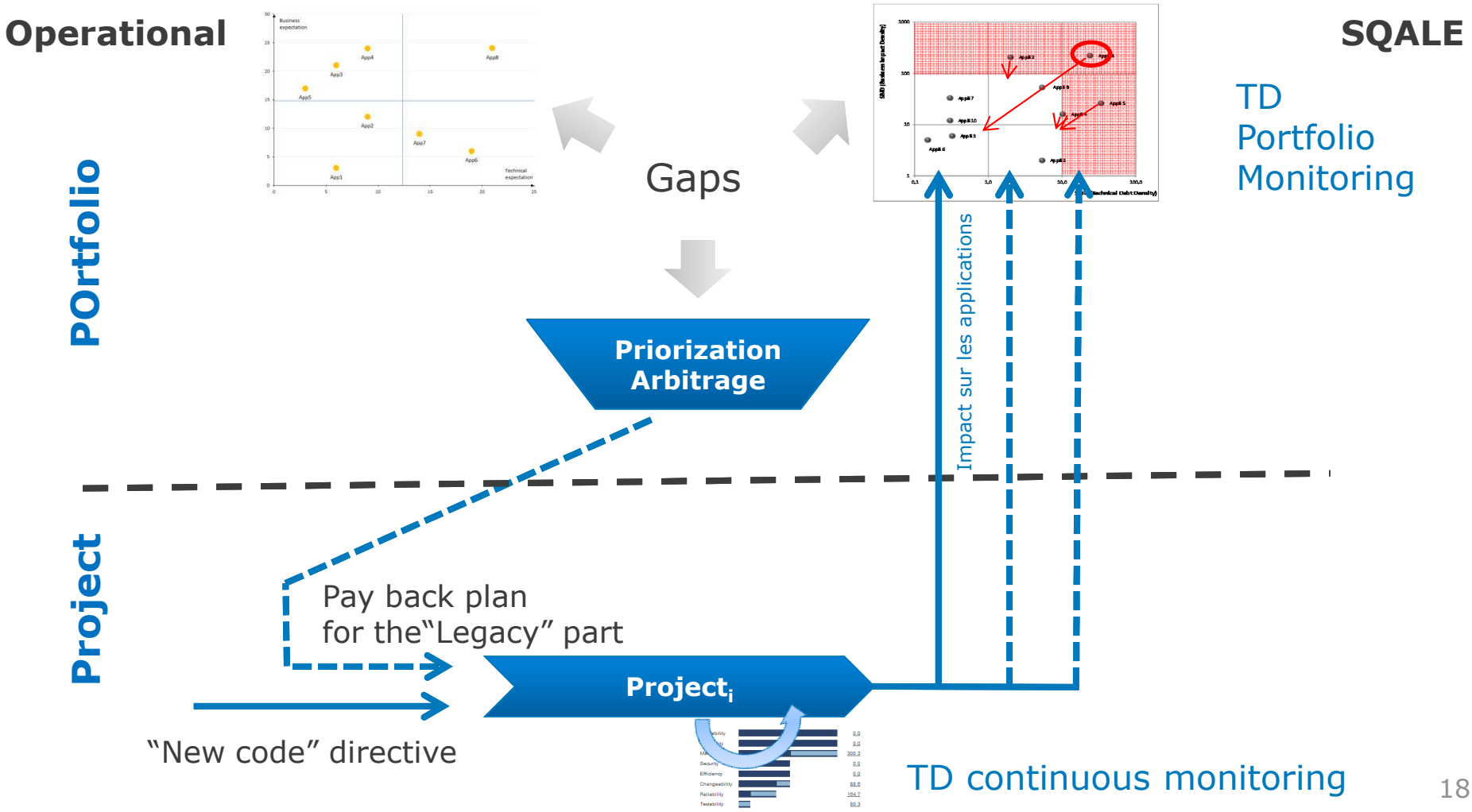


SQALE Quality Model: Order of the Characteristics

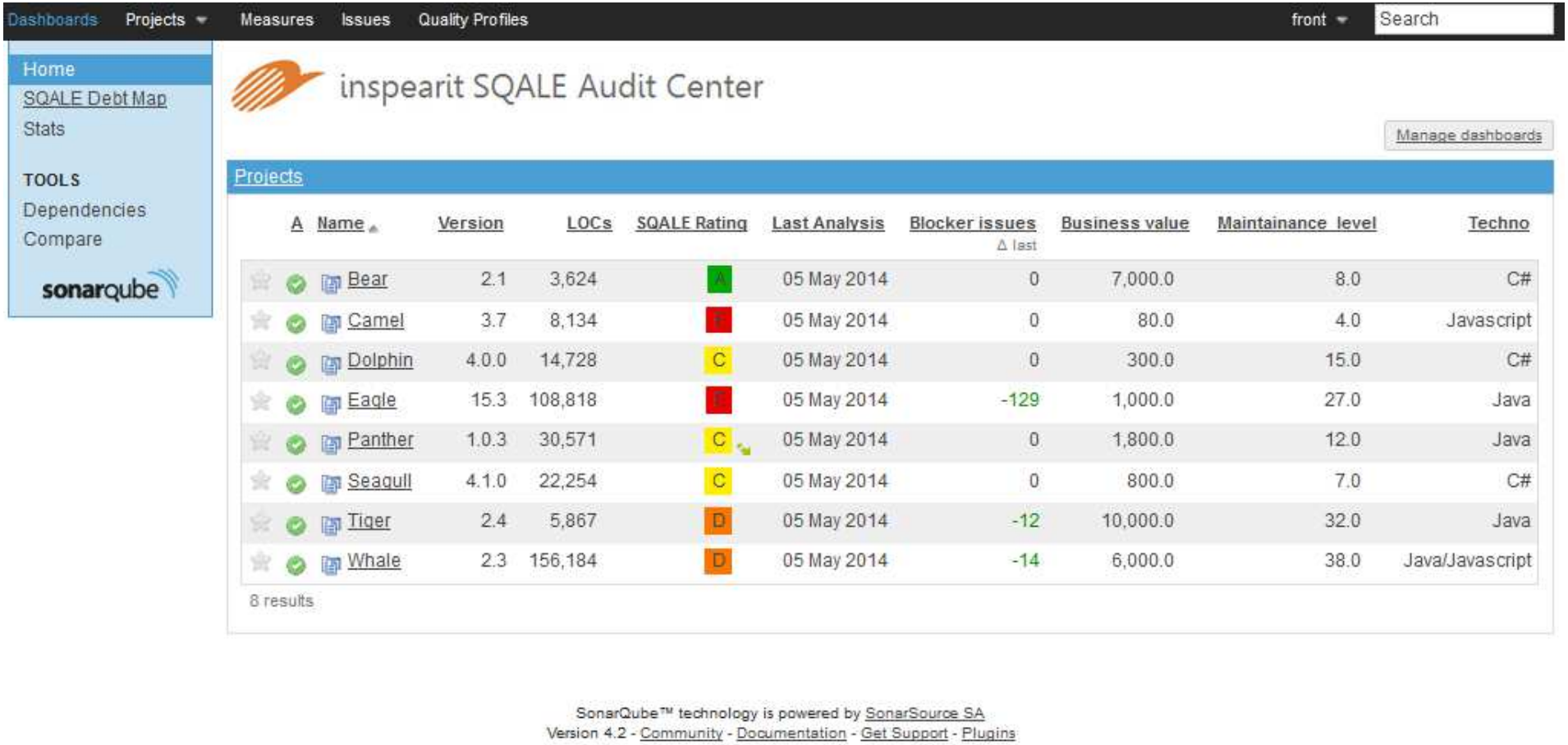


Portfolio management with SQALE

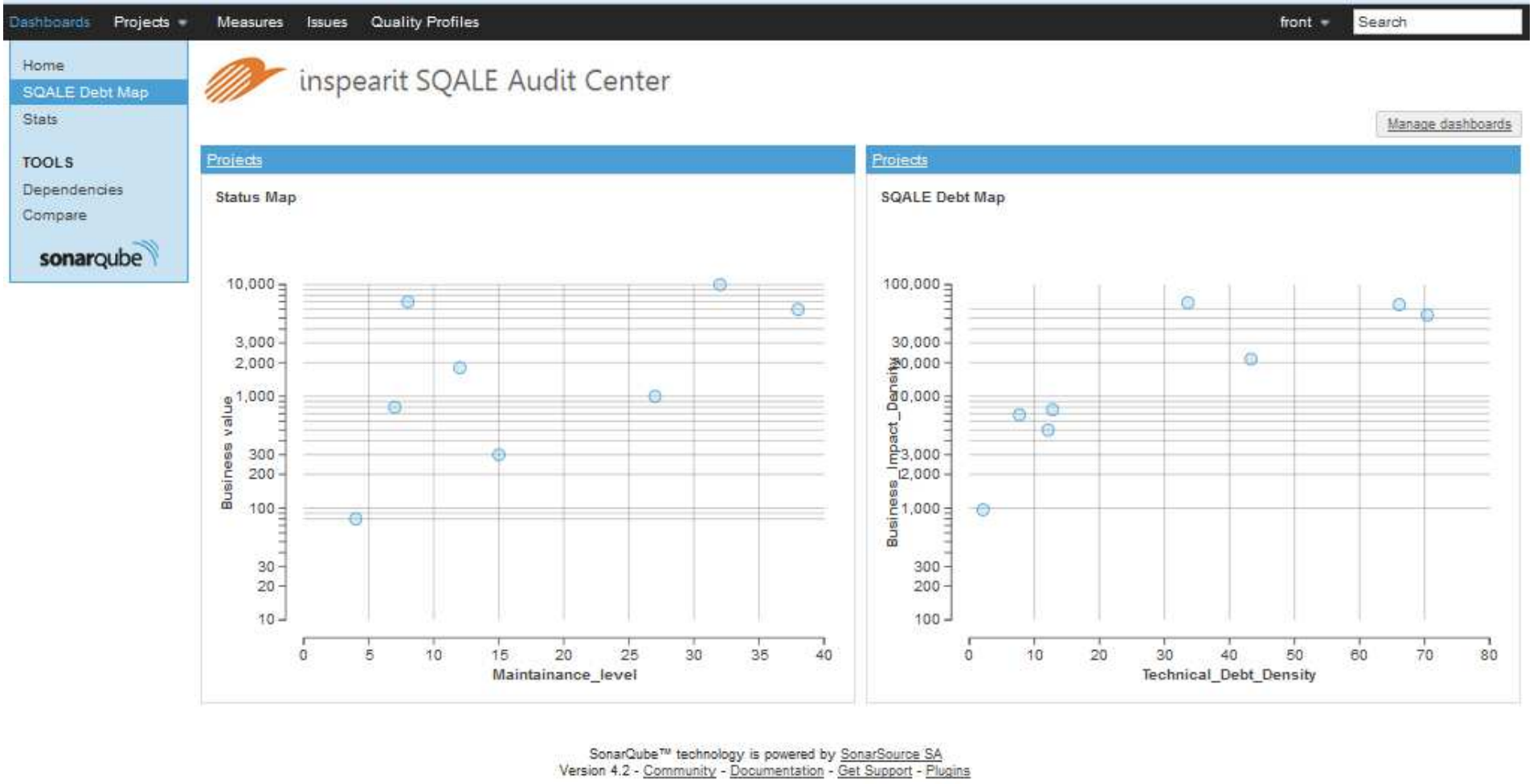
- A 2 levels and 2 speeds system



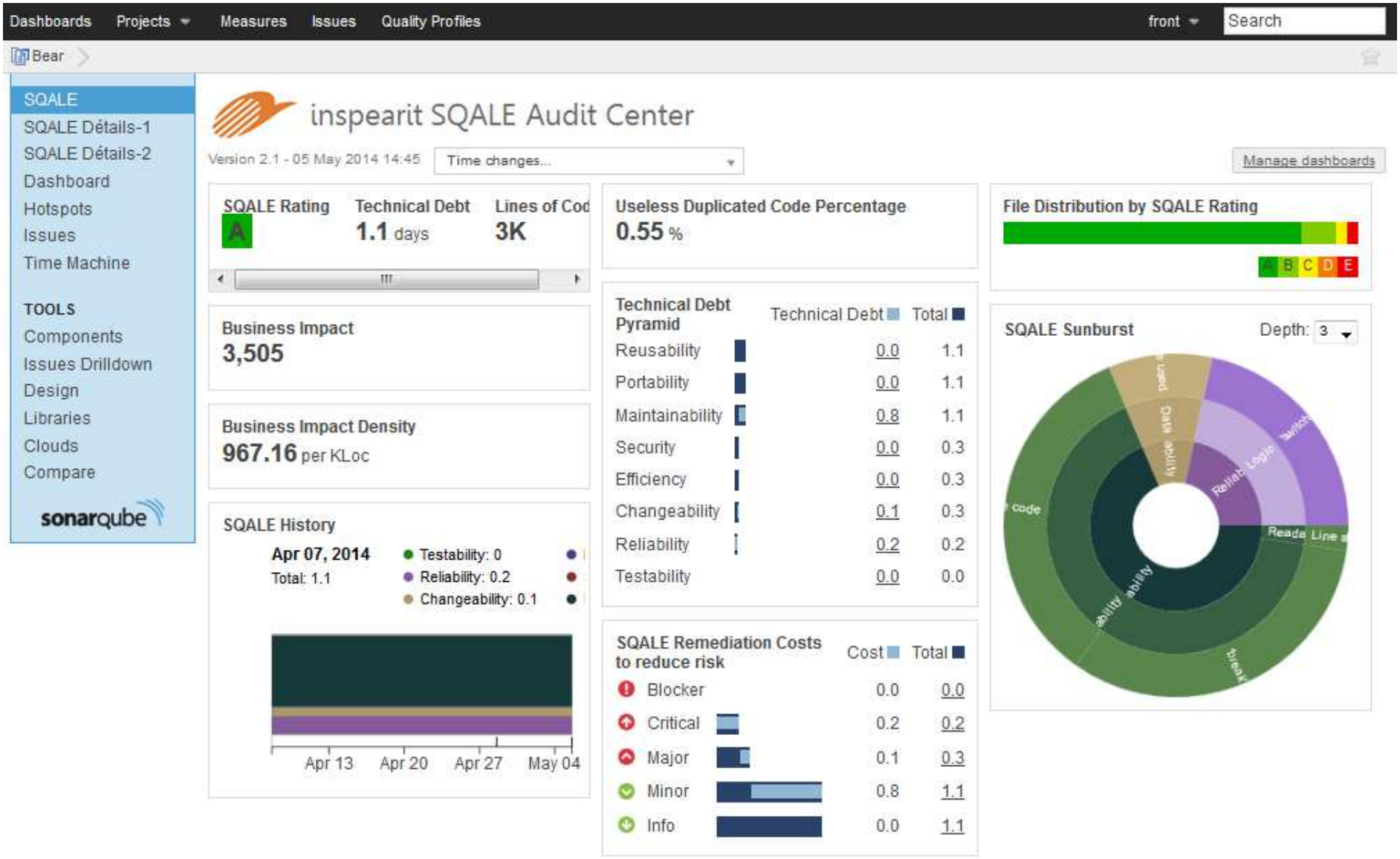
Screenshots -1



Screenshots - 2



Screenshots - 3



Screenshots - 4