

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Can software be healthy?

Jesus M. Gonzalez-Barahona

Universidad Rey Juan Carlos  
@jgbarah <http://jgbarah.github.io/presentations>

SoHeal 2019

Montreal (Canada), May 28th 2019

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

*Health,  
what is health?  
Can anyone be healthy  
at all?*

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# What do we want?



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

*Speaker: What do we want?*

*Crowd: Patience!*

*Speaker: When do we want it?*

*Crowd: Right now!!!*

Adapted from a well known joke by Eugenio (Spanish humorist).

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# The theory

*Software should behave  
according to requirements,  
be cheap to maintain,  
be easy to use,  
have good performance,  
...*

“We want software of good quality”

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# The practice

In most cases...

- Functionality: shallow verification
- Requirements: from nonexistent to incomplete
- Maintainability: very expensive
- Usability: many facets
- Performance: only a relative target

“Good enough”, depending on the stakeholder

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Improving quality



Can software be  
healthy?

Jesus M.  
Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# The quest for quality

“Traditional” approach in software engineering:

- Product quality  
(ISO 9126, CISQ)
- Process quality  
(ISO 9001, CMM)

Follow the rules, increase quality



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# CISQ (code) quality model

- reliability
- efficiency
- security
- maintainability

<https://www.it-cisq.org>

Can software be healthy?

Jesus M. Gonzalez-Barahona

# CISQ (code) quality model

| SOFTWARE QUALITY CHARACTERISTIC | CODING PRACTICES UNIT LEVEL  | ARCHITECTURAL PRACTICES SYSTEM LEVEL   |
|---------------------------------|--|--|
| RELIABILITY                     | <ul style="list-style-type: none"> <li>• Protecting state in multi-threaded environments</li> <li>• Safe use of inheritance and polymorphism</li> <li>• Resource bounds management, Complex code</li> <li>• Managing allocated resources, Timeouts</li> </ul>  | <ul style="list-style-type: none"> <li>• Multi-layer design compliance</li> <li>• Software manages data integrity and consistency</li> <li>• Exception handling through transactions</li> <li>• Class architecture compliance</li> </ul>   |
| PERFORMANCE EFFICIENCY          | <ul style="list-style-type: none"> <li>• Compliance with Object-Oriented best practices</li> <li>• Compliance with SQL best practices</li> <li>• Expensive computations in loops</li> <li>• Static connections versus connection pools</li> <li>• Compliance with garbage collection best practices</li> </ul> | <ul style="list-style-type: none"> <li>• Appropriate interactions with expensive or remote resources</li> <li>• Data access performance and data management</li> <li>• Memory, network and disk space management</li> <li>• Centralized handling of client requests</li> <li>• Use of middle tier</li> </ul> |

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Measuring quality

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# There are other motivations

What if the focus is “knowing” instead of “improving”

- comparison
- tracking
- self-awareness

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# There are other subjects

What if the people are also important?

- the builders
- the evaluators

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# The builders

Specially important in FOSS:

- diverse people working together
- different motivations, agendas...
- the sense of community

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# The evaluators

Different goals / interests  
mean  
different definitions of “good”

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# And we still have the context...

Software is not used in a vacuum:

- legalese
- support
- economy
- ecosystem
- ...



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# A bit of history



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# OpenBRR



## Business Readiness Rating™

*A Framework for Evaluating Open Source Software*

### STEERING COMMITTEE

Larry Augustin, Open Source Strategist  
Michael Goulde, Forrester Research  
Peter Kronowitt, Intel  
Murugan Pal, SpikeSource

Josh Berkus, PostgreSQL  
Marc Hedlund, O'Reilly CodeZoo  
George Pace, Prudential Financial  
Anthony Wasserman, Carnegie Mellon West (Chair)

### FOUNDING SPONSORS



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

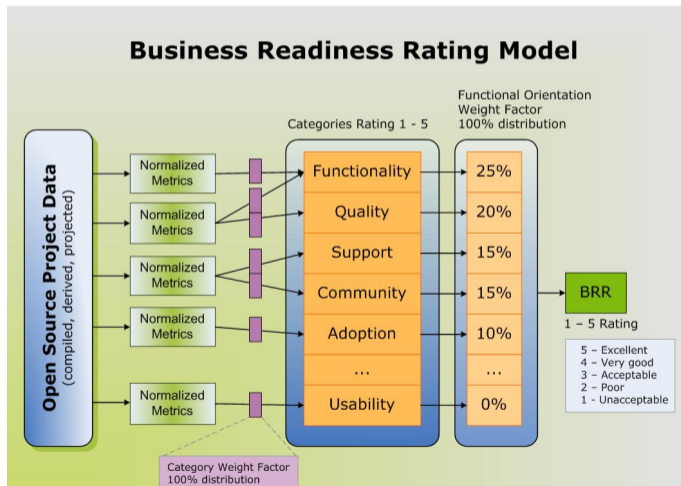
A bit of history

Software health

Some ideas

Concluding...

# OpenBRR



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

Atos  Origin

QSOS



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

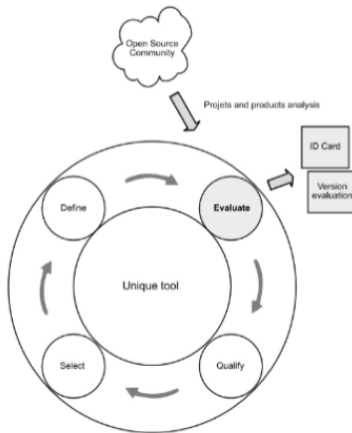
A bit of history

Software health

Some ideas

Concluding...

# QSOS



- ID card and version evaluations;
- Scoring of criteria on three major axis:-
  - Functional coverage;
  - Risks from customer perspective;
  - Risks from Atos Origin perspective;
- Weighted metrics for product scoring;

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# QSOS

## Intrinsic robustness

- Maturity
- Adoption
- Development Roadmap
- Activity
- Development independence

## Industrialised solution

- Services
- Documentation
- Quality Assurance
- Exploitability

## Integration

- Adherence to standards
- Interface with other products

## Strategy

- Licence
- Copyright owners
- Modification of source code
- Roadmap
- Sponsor

## Technical adaptability

- Modularity

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

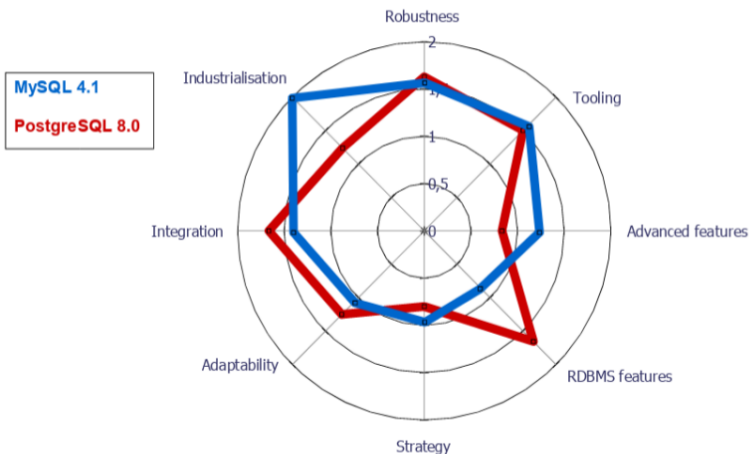
A bit of history

Software health

Some ideas

Concluding...

# QSOS



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

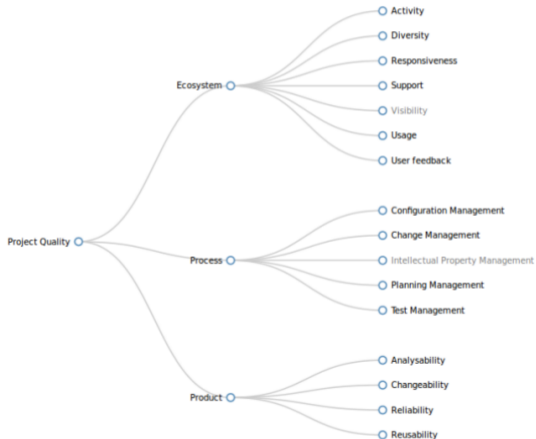
A bit of history

Software health

Some ideas

Concluding...

# Polarsys Quality Model





Can software be healthy?

Jesus M. Gonzalez-Barahona

# Polarsys Quality Model

What do we want?

Improving quality

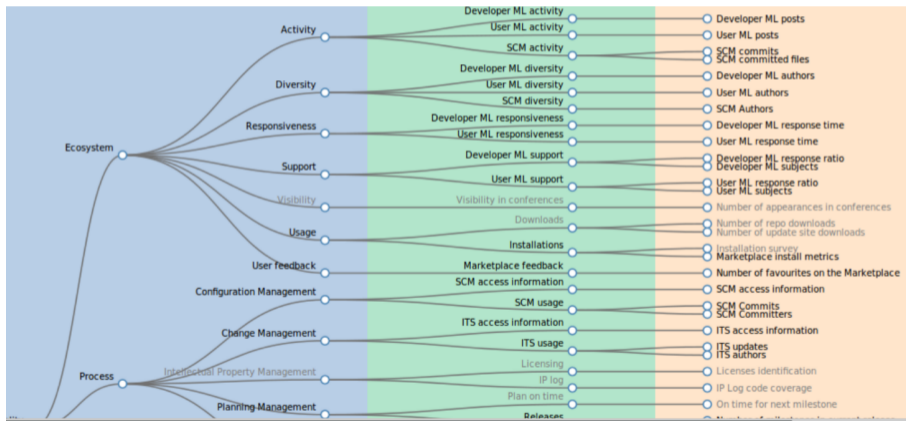
Measuring quality

A bit of history

Software health

Some ideas

Concluding...



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Software health



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

*“A set of **characteristics** of a software **project** and its **context** determining its capability for producing **software of good quality**, according to certain **criteria**”*

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# What is software health?

A concept applied to a **project**

- Criteria to define quality
- Characteristics that allow for that quality
- Time spot for measuring

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# What is software health?

A concept applied to a **project context**

- Important issues happen in the surroundings
- Examples: training, business, use
- Interrelations in large ecosystems (competition, cooperation: coevolution)

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# The context: ecosystem

A project in a certain ecosystem:

- Health linked to the health of the ecosystem  
Mutual impact (positive or negative)
- Example: attraction of new developers
- Example: common modules
- Example: availability of skills
- Example: marketing and usability

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Measuring software health

- Quantify quality criteria
- Find indicators that summarize criteria
- Find values for them that characterize health
- Track their evolution

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Example

- Criteria for quality: minimize unfixed errors
- Indicator: unfixed bug reports
- Healthy value:  $X$  unfixed bug reports per KLoC
- Alarm when number below  $X$



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# The causes for health

*The really interesting matter is to know the causes for variation in indicators*

Example: unfixed bug reports are minimized by good code review

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...



# Some ideas

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# On the shoulders of giants

Systems are composed of many modules:

- Dependencies matter
- Overall health dependent all components
- In some cases, dependent on the most unhealthy component
- Projects and communities: interdependent

Assessing the overall health of a complete system

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Making decisions for tomorrow

Many systems are in production for many years:

- Prediction on future health
- Not all aspects are equally relevant (example: fixing bugs vs. new functionality)
- Important: understanding dynamics (extending past to future is not good enough)

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Integrating metrics with development

Can health be yet another factor to consider?

- It could be an indicator for every stakeholder
- Computed frequently, so that it is up to date
- Published widely, so that everyone is aware

Include health in the data for decision making

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Working with stakeholders

- Builders
- Integrators
- Users

Health for different actors  
for different purposes

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

CHA<sup>O</sup>SS

The logo for CHAOSS features the word "CHA" in a bold, black, sans-serif font, followed by a stylized letter "O" composed of five colored segments (blue, red, green, purple, and pink) arranged in a circle, and finally the word "SS" in the same bold, black, sans-serif font.

<http://chaoss.community>

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Understanding dynamics

How do specific actions impact on the health model for a software development system?



Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Towards a new research framework

Define health conditions

Find out how to measure indicators of health

Study deviations from healthy conditions

Learn how to help to go back to healthy

Include all of this in the development process

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Simple example

Health condition: no regressions

Indicators: tests failing

Deviations: old errors appear

Mitigation: automatic testing

Continuous integration system

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Beyond opinions

Evidence that the indicator shows deviation from healthy condition

Evidence of mitigation:

- condition go back to healthy
- indicator go back to normal

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

# Concluding...

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

Can we do this  
in non-trivial cases?

Can software be healthy?

Jesus M. Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...

Software health may provide  
a good framework  
for structuring research,  
producing useful analysis,  
and producing actionable outputs

Can software be  
healthy?

Jesus M.  
Gonzalez-Barahona

What do we want?

Improving quality

Measuring quality

A bit of history

Software health

Some ideas

Concluding...



©2019 Jesus M. Gonzalez-Barahona.

Some rights reserved. This document is distributed under the terms  
of the Creative Commons License "Attribution-ShareAlike 4.0",  
available in

<http://creativecommons.org/licenses/by-sa/4.0/>

This document (including source) is available from  
<https://jgbarah.github.io/presentaciones>