CMPT 473
Software Testing, Reliability and Security

Intro to Software Quality

Nick Sumner

Today

- Quiz
- Administrivia
- Overview
- Intro to Software Quality

Course Website

- http://www.cs.sfu.ca/~wsumner/teaching/473/
  - Policies
  - Schedule
  - Reading
  - Assignments
  - Exam/Course Project
  - …

People

- Instructor: Nick Sumner
  - Office Hour: Wednesday 3:30-4:30
  - Office: TBD

People

- Instructor: Nick Sumner
  - Office Hour: Wednesday 3:30-4:30
  - Office: TBD

- TAs: Poornima
  Kumar Shivam

Grading

- 45% Exercises
- 30% Exam
- 25% Quizzes (Dropping the worst 2) & Homework

Exercises

- 5ish Exercises

Exercises

- 5ish Exercises
- Practical experience learning how to use tools & techniques

Exercises

- 5ish Exercises
- Practical experience learning how to use tools & techniques
- Exercises may be turned in late with a 20% penalty for each day.

Exams

- Two options for a final exam:

Exams

- Two options for a final exam:
    1) Written final exam

Exams

- Two options for a final exam:
  1) Written final exam
  2) Practical research project based final
     - Requires meeting with me by the middle of the semester to get a project approved (and to commit to a project)

Exams

- Two options for a final exam:
  1) Written final exam
  2) Practical research project based final
     - Requires meeting with me by the middle of the semester to get a project approved (and to commit to a project)
- Why two options?

Reading

- Not *required* to purchase any books for the course

Reading

- Not *required* to purchase any books for the course

- Books available online or through library for reading assignments

Reading

- Not *required* to purchase any books for the course
- Books available online or through library for reading assignments
- Still need to complete reading assignments before class for the day they are assigned

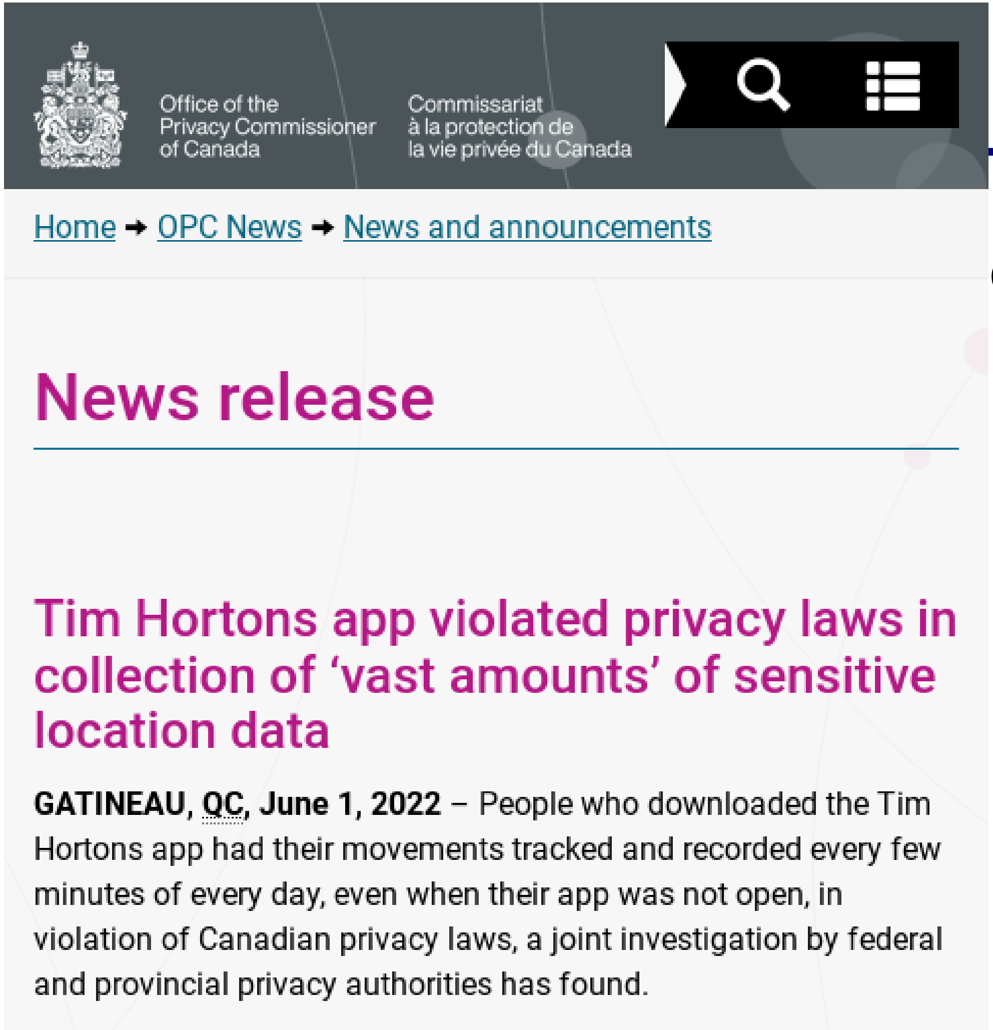
End of the Administrivia

Q&A

What is Software Quality?

What is Software Quality?

- Bad software costs developers and users ***billions*** of dollars a year, but why?
  - What is `good' software?
  - What is `bad' software?

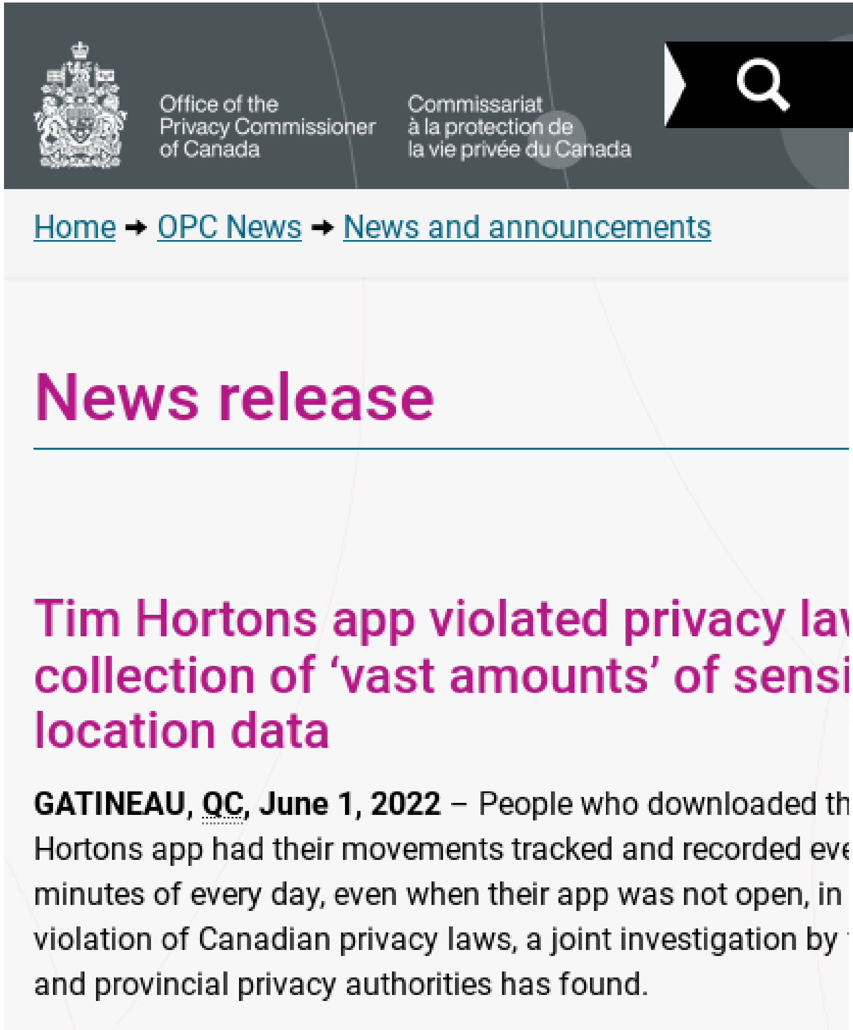


News release

Tim Hortons app violated privacy laws in collection of 'vast amounts' of sensitive location data

GATINEAU, QC, June 1, 2022 – People who downloaded the Tim Hortons app had their movements tracked and recorded every few minutes of every day, even when their app was not open, in violation of Canadian privacy laws, a joint investigation by federal and provincial privacy authorities has found.

d users ***billions*** of dollars a year, but



Office of the Privacy Commissioner of Canada

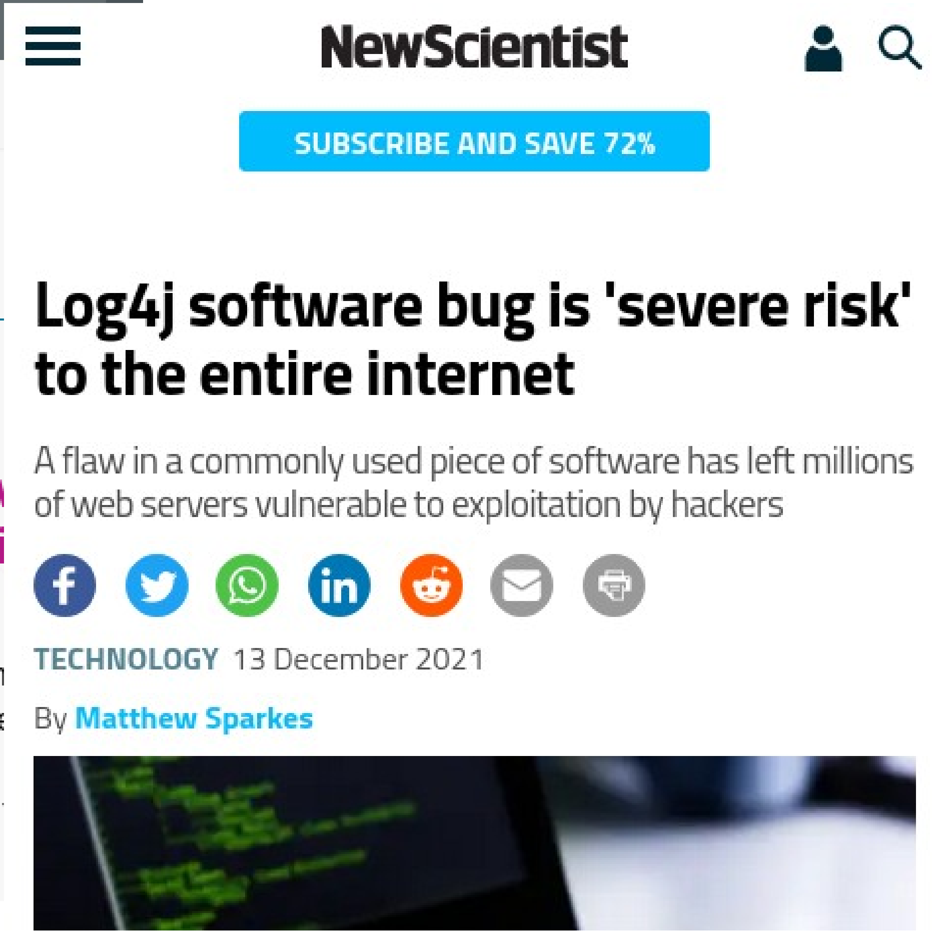
Commissariat à la protection de la vie privée du Canada

Home → OPC News → News and announcements

News release

Tim Hortons app violated privacy la collection of 'vast amounts' of sensi location data

GATINEAU, QC, June 1, 2022 – People who downloaded th Hortons app had their movements tracked and recorded eve minutes of every day, even when their app was not open, in violation of Canadian privacy laws, a joint investigation by and provincial privacy authorities has found.



NewScientist

SUBSCRIBE AND SAVE 72%

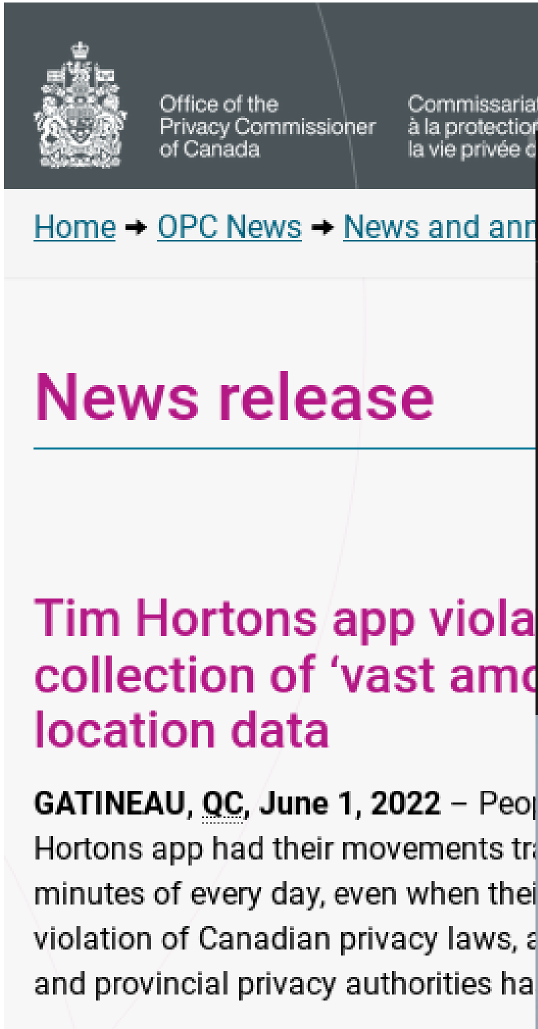
Log4j software bug is 'severe risk' to the entire internet

A flaw in a commonly used piece of software has left millions of web servers vulnerable to exploitation by hackers

TECHNOLOGY 13 December 2021

By Matthew Sparkes

r, but



Office of the Privacy Commissioner of Canada

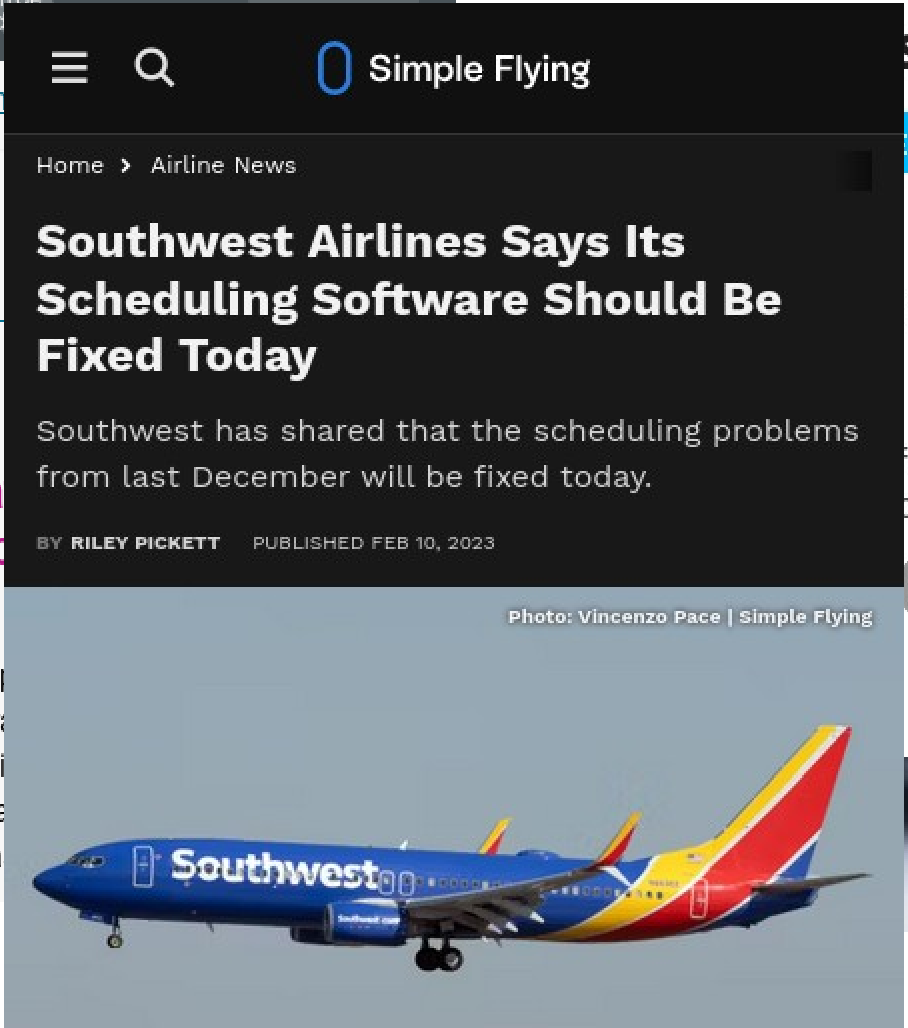
Commissariat à la protection de la vie privée d

Home → OPC News → News and ann

News release

Tim Hortons app viola collection of 'vast amo location data

GATINEAU, QC, June 1, 2022 – Peo Hortons app had their movements tr minutes of every day, even when thei violation of Canadian privacy laws, a and provincial privacy authorities ha



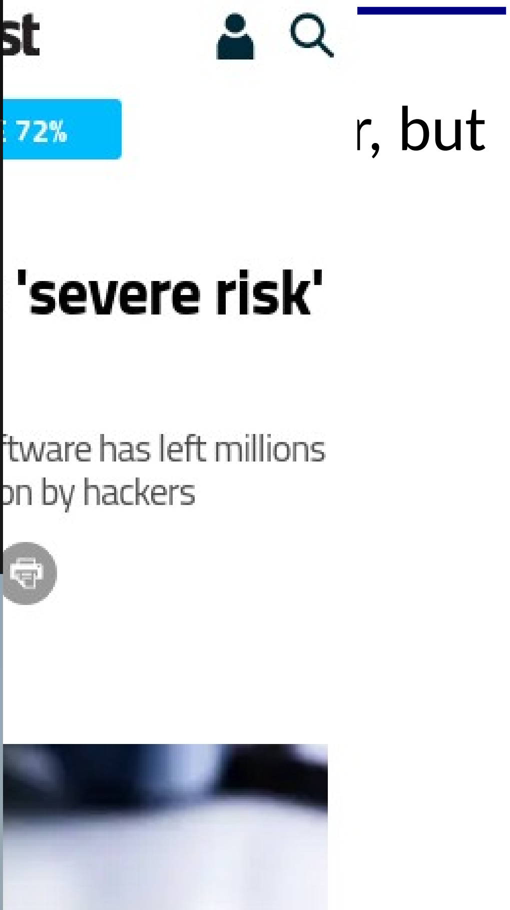
Home › Airline News

Southwest Airlines Says Its Scheduling Software Should Be Fixed Today

Southwest has shared that the scheduling problems from last December will be fixed today.

BY **RILEY PICKETT** PUBLISHED FEB 10, 2023

Photo: Vincenzo Pace | Simple Flying



st

72%

r, but

'severe risk'

ftware has left millions on by hackers

What is Software Quality?

- Bad software costs developers and users ***billions*** of dollars a year, but why?
  - What is `good' software?
  - What is `bad' software?

  - Try to identify 5-6 most important characteristics of `quality' software. (and examples)

It is a Matter of Perspective

Your *role* relative to the software changes what is important to you.

- End Users

It is a Matter of Perspective

Your *role* relative to the software changes what is important to you.

- End Users
  - Does the software have all desired features?

It is a Matter of Perspective

Your *role* relative to the software changes what is important to you.

- End Users
  - Does the software have all desired features?
  - Can it reliably produce correct results for good input?

It is a Matter of Perspective

Your *role* relative to the software changes what is important to you.

- End Users
  - Does the software have all desired features?
  - Can it reliably produce correct results for good input?
  - Does it securely & gracefully handle bad input?

It is a Matter of Perspective

Your *role* relative to the software changes what is important to you.

- End Users
  - Does the software have all desired features?
  - Can it reliably produce correct results for good input?
  - Does it securely & gracefully handle bad input?
  - It it easy to use?

It is a Matter of Perspective

Your *role* relative to the software changes what is important to you.

- End Users
  - Does the software have all desired features?
  - Can it reliably produce correct results for good input?
  - Does it securely & gracefully handle bad input?
  - It it easy to use?
  - Is it responsive?

It is a Matter of Perspective

Your *role* relative to the software changes what is important to you.

- End Users
  - Does the software have all desired features?
  - Can it reliably produce correct results for good input?
  - Does it securely & gracefully handle bad input?
  - It it easy to use?
  - Is it responsive?
  - Does it integrate well with other software?

It is a Matter of Perspective

Your *role* relative to the software changes what is important to you.

- End Users
  - Does the software have all desired features?
  - Can it reliably produce correct results for good input?
  - Does it securely & gracefully handle bad input?
  - It it easy to use?
  - Is it responsive?
  - Does it integrate well with other software?

What are the consequences of missing these goals?

It is a Matter of Perspective

- Operational / Deployment

It is a Matter of Perspective

- Operational / Deployment
  - Is the software secure from attacks that may compromise the IT infrastructure?

It is a Matter of Perspective

- Operational / Deployment
  - Is the software secure from attacks that may compromise the IT infrastructure?
  - Does the software appropriately use resources?
    (CPU, Memory, Disk Space, Bandwidth, ...)

It is a Matter of Perspective

- Operational / Deployment
  - Is the software secure from attacks that may compromise the IT infrastructure?
  - Does the software appropriately use resources?
    (CPU, Memory, Disk Space, Bandwidth, …)

What are the consequences of missing these goals?

It is a Matter of Perspective

- Developers

It is a Matter of Perspective

- Developers
  - How easy is the software to adapt to changes in requirements?

It is a Matter of Perspective

- Developers
  - How easy is the software to adapt to changes in requirements?
  - Can the software be easily adapted to other systems?

It is a Matter of Perspective

- Developers
  - How easy is the software to adapt to changes in requirements?
  - Can the software be easily adapted to other systems?
  - How easy is the software to inspect and understand?

It is a Matter of Perspective

- Developers
  - How easy is the software to adapt to changes in requirements?
  - Can the software be easily adapted to other systems?
  - How easy is the software to inspect and understand?
  - Can components be easily examined gauged for quality?

It is a Matter of Perspective

- Developers
  - How easy is the software to adapt to changes in requirements?
  - Can the software be easily adapted to other systems?
  - How easy is the software to inspect and understand?
  - Can components be easily examined gauged for quality?

What are the consequences of missing these goals?

Standardized

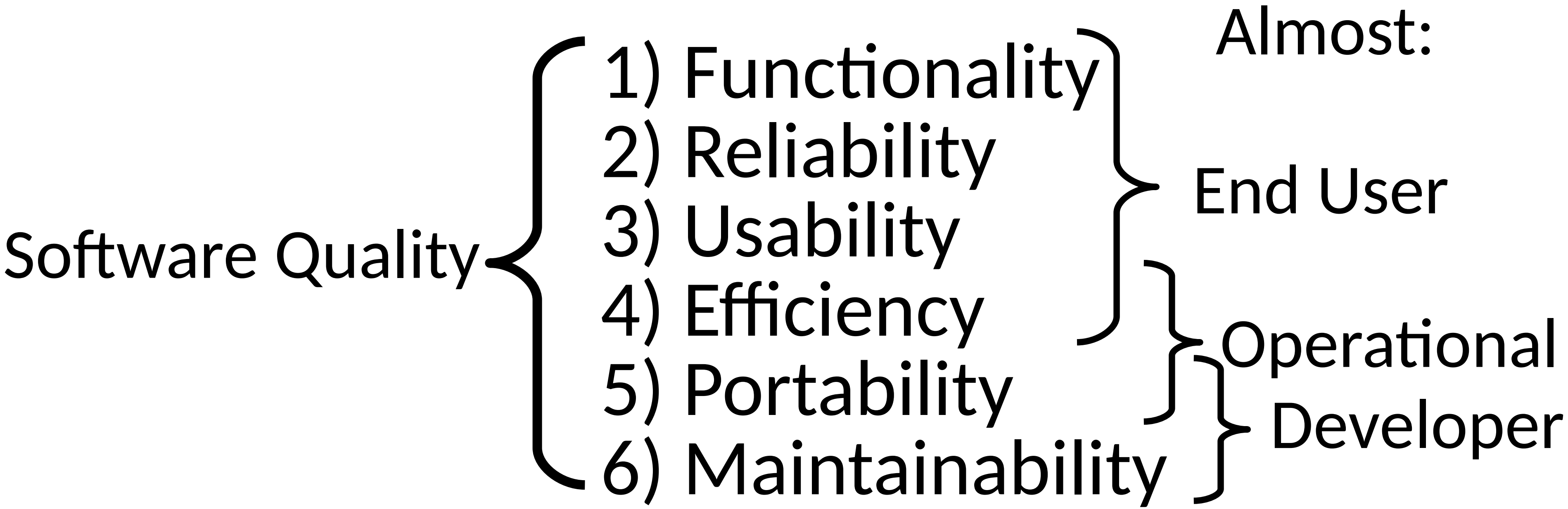
- ISO/IEC 9126 groups them as:

Software Quality

1) Functionality
2) Reliability
3) Usability
4) Efficiency
5) Portability
6) Maintainability

Standardized

- ISO/IEC 9126 groups them as:



Software Quality

1) Functionality
2) Reliability
3) Usability
4) Efficiency
5) Portability
6) Maintainability

Almost:

End User

Operational

Developer

Going Deeper

- Reliability – Doesn't just mean not having faults!

Going Deeper

- Reliability – Doesn't just mean not having faults!
  - Avoid failure resulting from software faults
  - Maintain performance in the face of faults or attacks
  - Reestablish performance and data after a failure

Going Deeper

- Usability- Beyond adequate to polished

Going Deeper

- Usability- Beyond adequate to polished
  - Helps user understand whether the program meets their needs
  - Features & uses are easy to learn
  - Easy to operate & control
  - Liked!

Going Deeper

- Maintainability- Makes developers lives easier

Going Deeper

- Maintainability- Makes developers lives easier
  - Defects are easy to identify
  - Changes are easy to understand & don't affect other components
  - The software is easy to test

Measuring Quality

How Can We Measure Quality?

These characteristics may be vague or ill defined.

So how do we do it?

How Can We Measure Quality?

These characteristics may be vague or ill defined.

So how do we do it?

- Planning
  - Decide what criteria are most important
  - Form a plan to assess them, directly or indirectly

How Can We Measure Quality?

These characteristics may be vague or ill defined.

So how do we do it?

- Planning
- Process
  - Code reviews help to improve maintainability & reduce bugs
  - Regular monitoring

How Can We Measure Quality?

These characteristics may be vague or ill defined.

So how do we do it?

- Planning
- Process
- Testing!!
  - Show that you meet the minimum agreed requirements
  - Observe defects and performance issues.

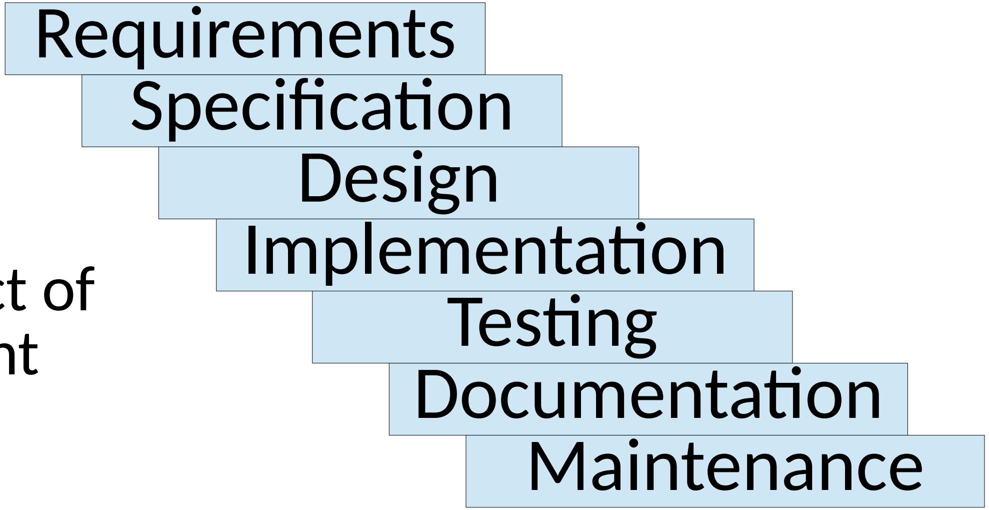
Quality Processes

Quality as a Process

The quality of software comes from the quality of the process used to build it.

Quality as a Process

The quality of software comes from the quality of the process used to build it.


Requirements

Specification

Design

Implementation

Testing

Documentation

Maintenance

In *every* aspect of development

Quality as a Process

How can we assist the process?

Quality as a Process

How can we assist the process?

- Synthetic- Tools and techniques structured to create better software
- Analytical- Tools and techniques that measure the quality of software

Quality as a Process

How can we assist the process?

- Synthetic- Tools and techniques structured to create better software
- Analytical- Tools and techniques that measure the quality of software

- Manual- Driven interactively by developers
- Automated- Function w/o developer intervention

Quality as a Process

	Synthetic	Analytical
Manual	• **design methodologies** • **prototyping** • **coding standards** • **templates** • **documentation standards**	
Automated		

Quality as a Process

	Synthetic	Analytical
Manual	• design methodologies • prototyping • coding standards • templates • documentation standards	• **walk-throughs** • **inspections** • **audits**
Automated		

Quality as a Process

	Synthetic	Analytical
Manual	• design methodologies • prototyping • coding standards • templates • documentation standards	• walk-throughs • inspections • audits
Automated	• **program generators** • **compilers** • **development environments**	

Quality as a Process

	Synthetic	Analytical
Manual	• design methodologies • prototyping • coding standards • templates • documentation standards	• walk-throughs • inspections • audits
Automated	• program generators • compilers • development environments	• **model checkers** • **program verifiers** • **program checkers** • **unit testing** • **integration testing** • **system testing**

Quality as a Process

Integrating these into the process is a core part of the quality plan

Quality as a Process

Integrating these into the process is a core part of the quality plan

- Set goals
- Establish practices
- Measure progress

So What Will We Do?

- We'll mostly consider how analytical tools and techniques can help ensure quality.

So What Will We Do?

- We'll mostly consider how analytical tools and techniques can help ensure quality.
- Starting with **TESTING**.