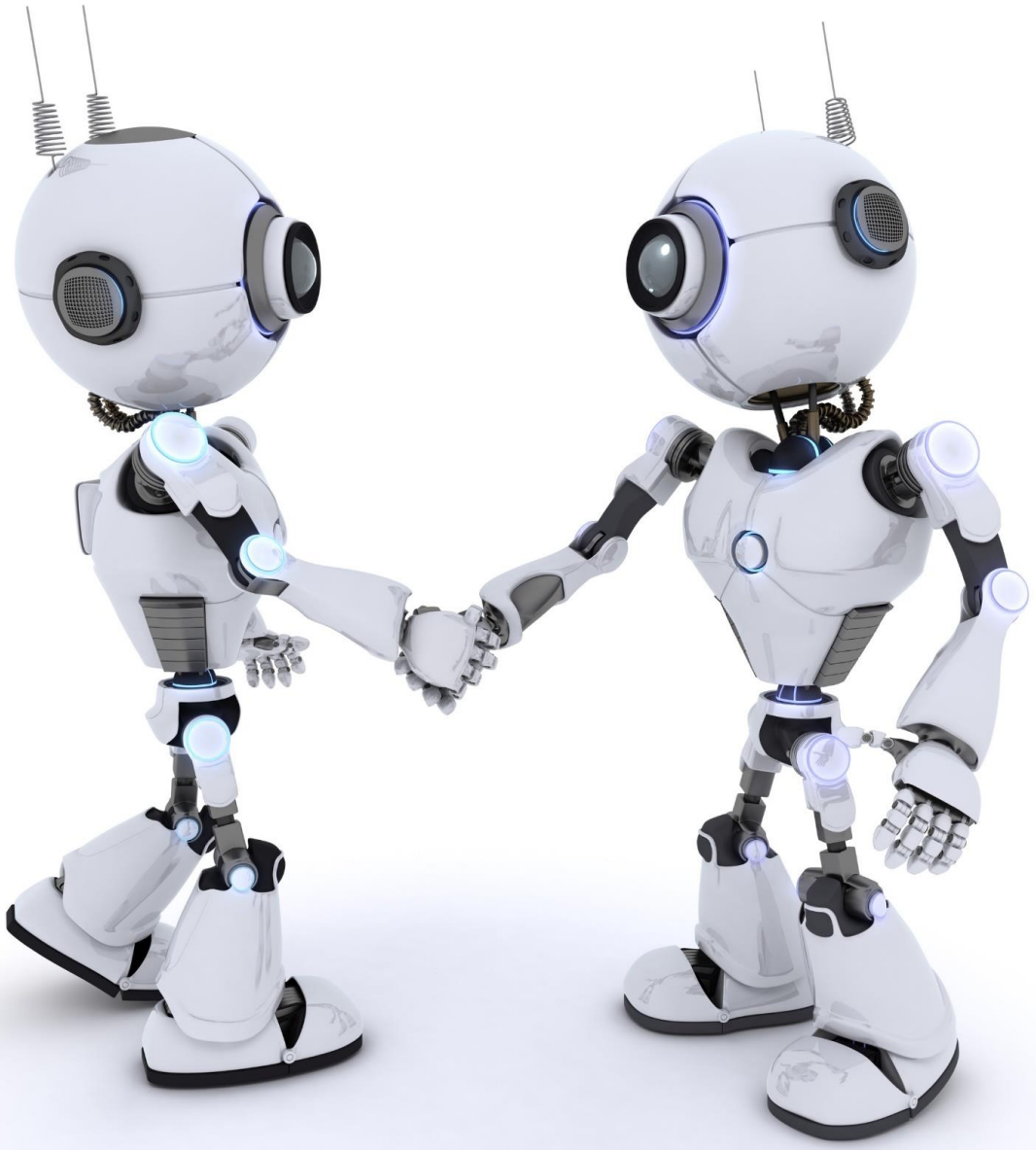


Miten tekoäly auttaa ohjelmistotestausta?

- Kari Kakkonen
- https://www.linkedin.com/in/karikakkonen/
- Dragons Out Oy
- IoT-seminaari 2024
- Kokkola
- 12.9.2024

Kari Kakkonen, Lead Testing Consultant

twitter.com/kkakkonen
dragonsout.com
act2lead.net

MORE INFORMATION

linkedin.com/in/karikakkonen/

ROLES

- Dragons Out Oy, Director of Training, Trainer and Coach
- Children's and testing author at Dragons Out Oy
- TMMi, Board of Directors
- Treasurer of Finnish Software Testing Board (FiSTB)

ACHIEVEMENTS

- **Tester of the Year in Finland 2021**
- **EuroSTAR Testing Excellence Award 2021**
- **Exemplary DevOps Instructor Award 2023 by DASA**
- ISTQB Executive Committee 2015-2021
- Influencing testing since 1996
- Ranked in 100 most influential IT persons in Finland (Tivi magazine)
- Great number of presentations in Finnish and international conferences
- TestausOSY/FAST founding member.
- Co-author of Agile Testing Foundations book
- Regular blogger in Tivi-magazine
- Growing Knowit to Testing Leader 2002-2024

SERVICES

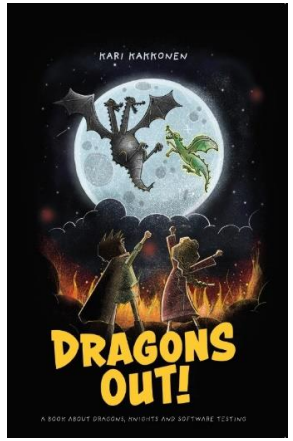
- ISTQB Advanced, Foundation, Agile Testing, AI Testing
- Knowit Quality Professional
- DASA DevOps
- Quality & Test process and organization development, Metrics, TMMi and other assessments
- Agile testing, Scrum, Kanban, Lean
- Leadership
- Test automation, Mobile, Cloud, DevOps, AI
- Quality, cost, benefits

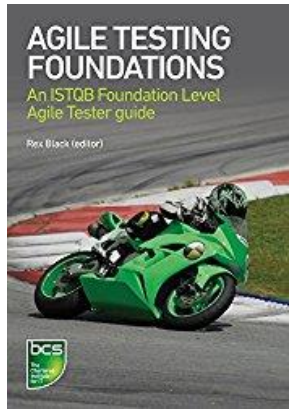
EDUCATION

- ISTQB Expert Level Test Management & Advanced Full & Agile Tester certified
- DASA DevOps, Scrum Master and SAFe certified
- TMMi Professional, Assessor, Process Improver certified
- SPICE provisionary assessor certified
- M.Sc.(Eng), Helsinki University of Technology (present Aalto University), Otaniemi, Espoo
- Marketing studies, University of Wisconsin-Madison, the USA.

BUSINESS DOMAINS

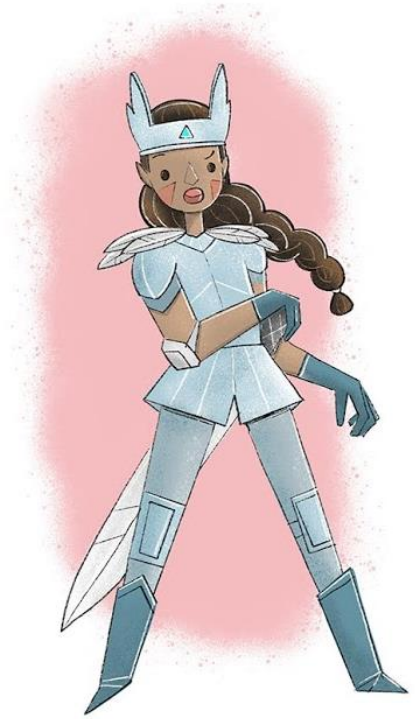
Wide spread of business domain knowledge: Embedded, industry, public, training, telecommunications, commerce, Insurance, banking, pension.





The book project ”Dragons Out!”

- Mission
  - “Software testing brought to children”
- Book
  - Author Kari Kakkonen
  - Illustrator Adrienn Széll
  - Text and illustration rights Dragons Out Oy
  - In Finnish, English, Polish, French and growing
  - For ages of 10-99
- Free “Dragon lesson in software testing” presentation under Creative Commons –license
  - Translated to 20 languages!
- More info: www.dragonsout.com
- Also other books coming out



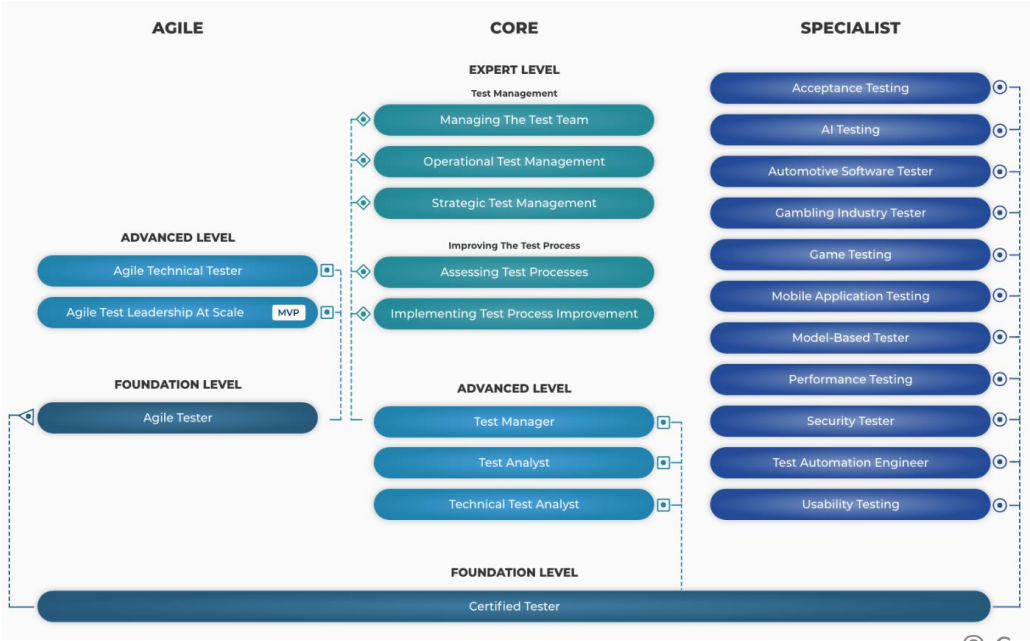
ACT 2 LEAD as a book

- Easy to read - chapters can be read in any order.
- Structure: questions, answers and cases.
  - 34 main chapters = questions, see next page.
- 270+ pages, in Finnish and soon in English (softback, e-book).
- For people like CxO, director, head of, manager, product owner, designer, developer, test manager, tester and student.
- Teaches to lead testing, not to test.

Buy the book: https://bit.ly/act2lead-book

ISTQB GLOBAL PRESENCE

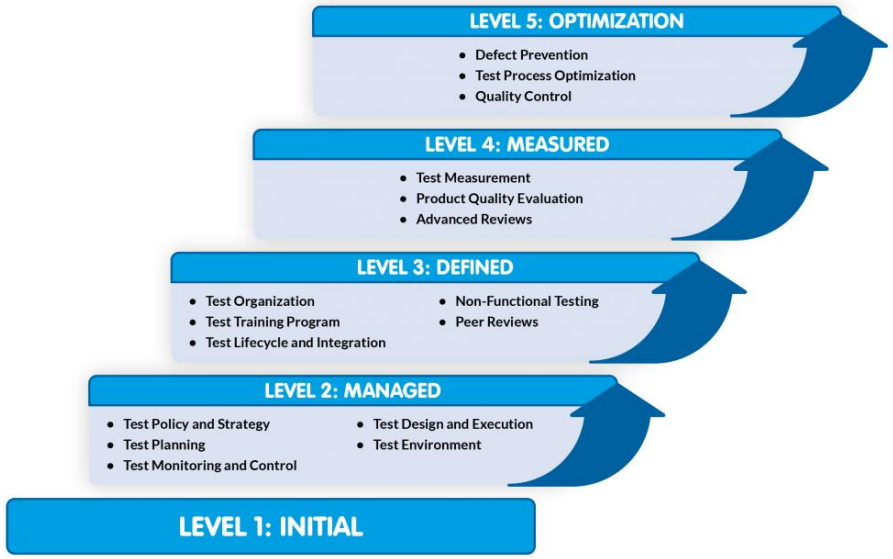
- Number of exams administered: over **1,2 million**
- Number of certifications issued: **845,000**
- **In 130 countries**



*Countries covered by **Member Boards and Global Exam Providers***

*Countries covered by **Global Exam Providers***

TMMi for test improvement in all kinds of testing, including agile and DevOps

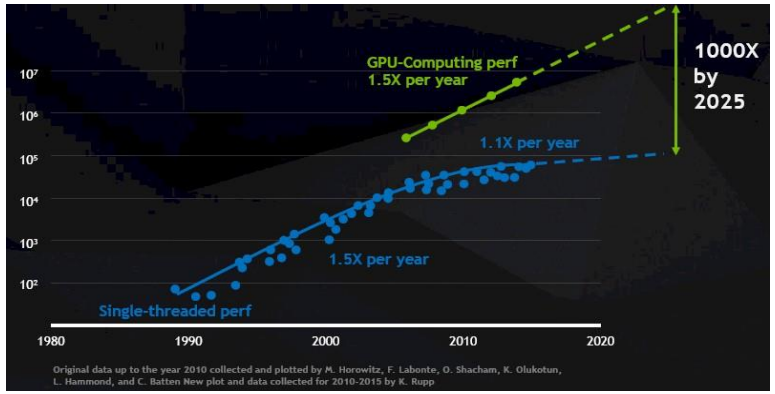


Agenda

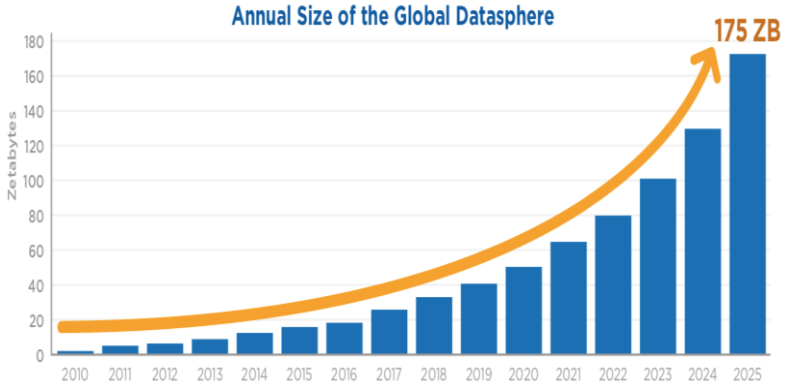
- Artificial Intelligence (AI) in short
- AI and testing tools

Why right now?
Four drivers behind AI revolution

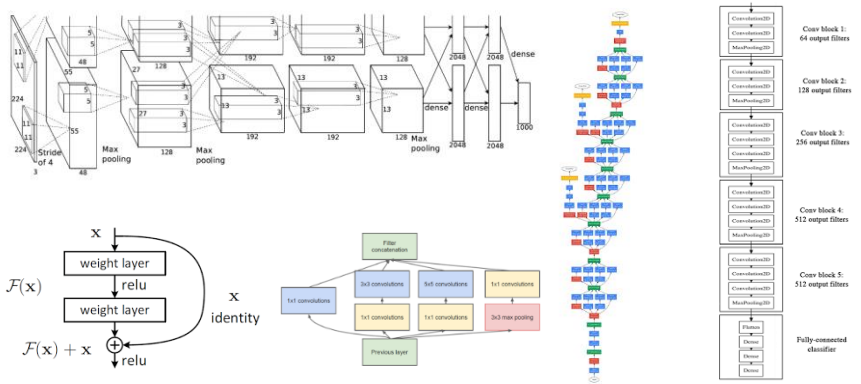
Computation growth due to general purpose GPUs



The rise of Big data



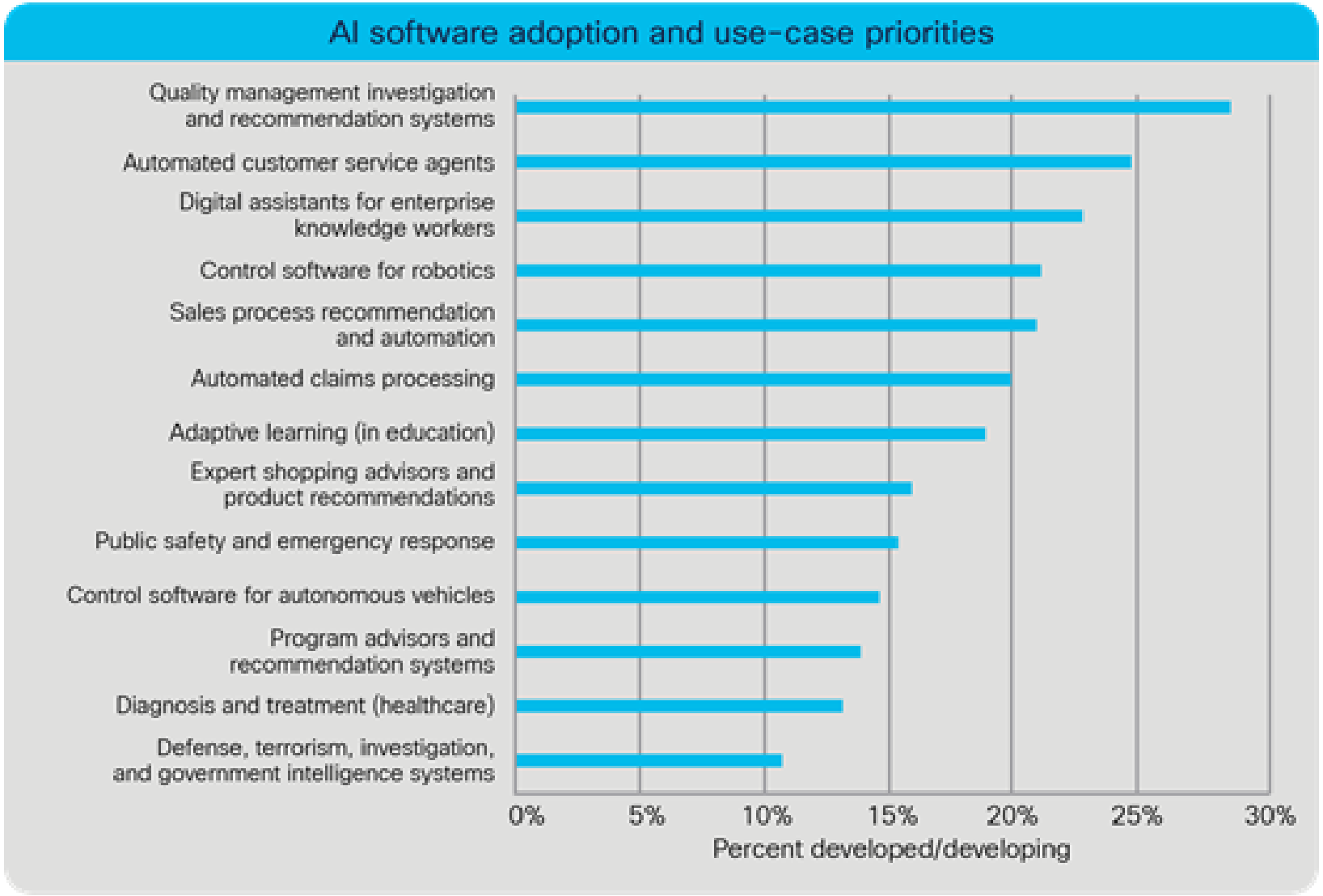
Community based achievements in Deep learning



Open source tools and frameworks

TensorFlow, scikit learn, Keras, Spark MLlib, H2O AutoML, PyTorch

Quality management as top use-case for AI



Source: AI software platform adoption survey, IDC, February 2019 [Percent of respondents: N=505]

Artificial intelligence == machine learning?

- When talking about AI, it is important to ask if we are talking about modelling humans or how to work in ideal way.
- Machine learning is a sub-category of AI, which considers algorithms that enable AI to learn ideal way to work.

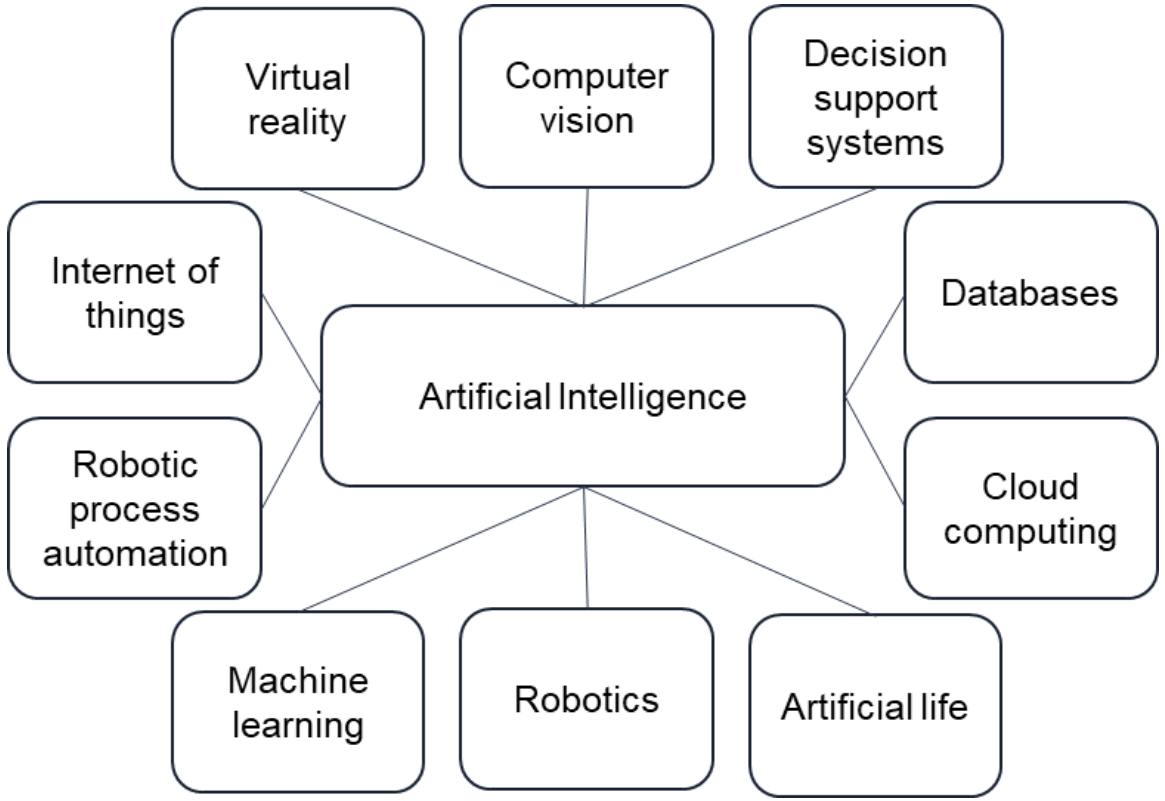


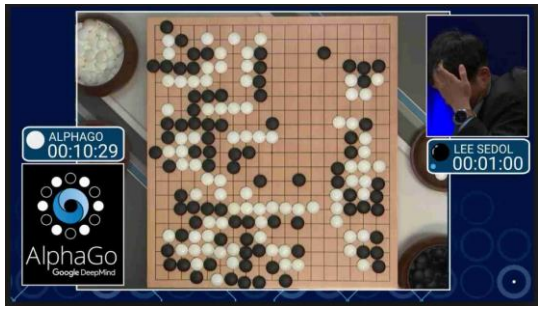
Figure: Example of concepts related to AI.

Three methods of machine learning

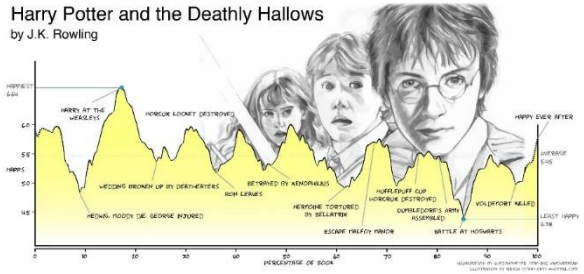
Machine learning methods

Supervised Learning
- The aim is to learn mapping from input to output.
- Output is known by supervisor.

Reinforcement Learning
- Seemingly good solutions are encouraged to develop further.



Unsupervised Learning
- The aim is to find ”something interesting”.
- Output is not known (known also as density estimation in statistics).



AI literature review[1]

1 Reagan, A., *et al.*, The emotional arcs of stories are dominated by six basic shapes, *EPJ Data Science* **(2016)**.

Anomaly detection

Tester’s objectives

- Can I find defects faster from lots of test runs?
- Which automatic defect reports / crash reports are actual defects?
- Are there duplicate reports?
- Automation to defect reporting?

Anomaly detection

- Detecting:
  - abnormal behaviour
  - new kind of customers
  - etc.

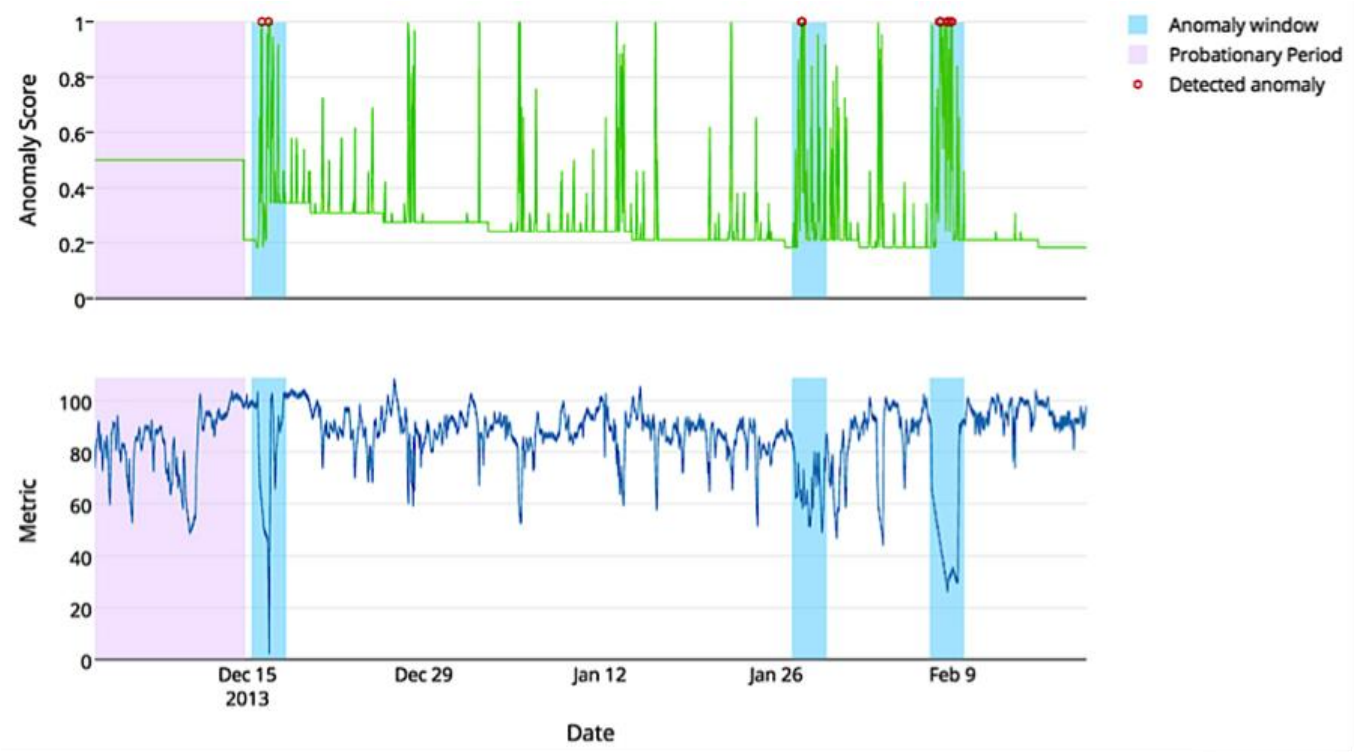


Figure: Example of time series

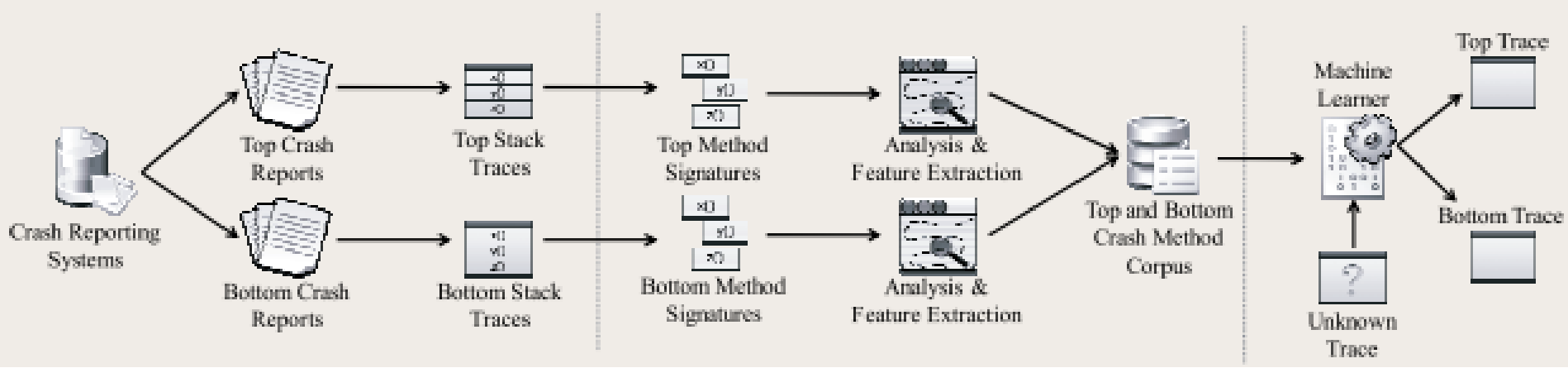
Large project defect classification

- NLP can be used to analyze text within different defect reports to identify areas of affected functionality
- Clustering algorithms such as SVM are used to define defect categories
- Text similarity metrics are used to identify similar or duplicate defects
- Particularly useful for automated defect reporting systems
- Case: MS Windows and Firefox and on large projects with many software engineers

Source: STA Consulting

Defect prioritization

- ML models can be trained to identify those defects most likely to cause critical system failures from automatically generated defect reports



Ref: Kim, D.; Wang, X.; Kim, S.; Zeller, A.; Cheung, S.C.; Park, S. (2011). "Which Crashes Should I Fix First? Predicting Top Crashes at an Early Stage to Prioritize Debugging Efforts," in the IEEE Transactions on Software Engineering, volume 37

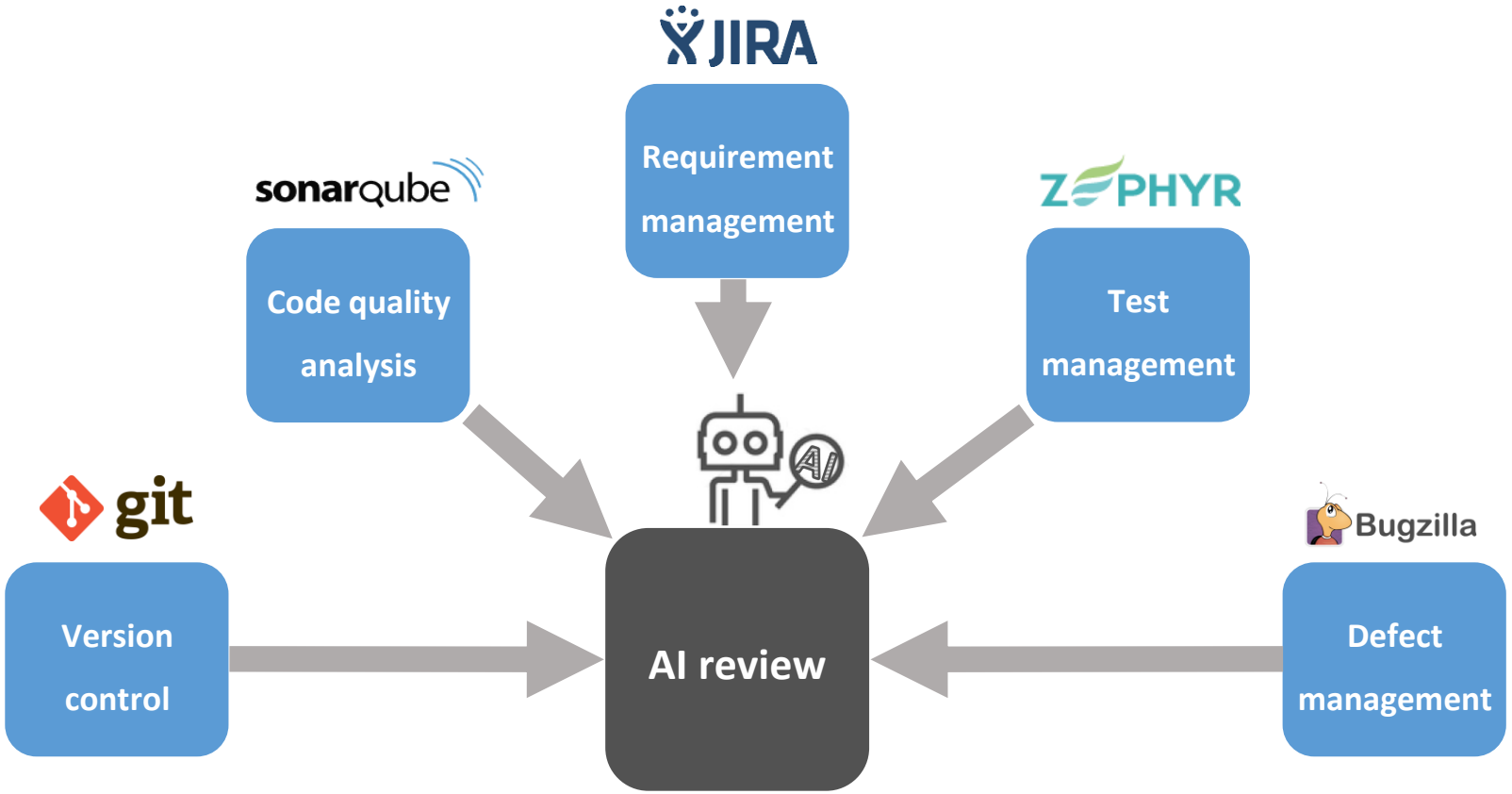
Source: STA Consulting

Prediction

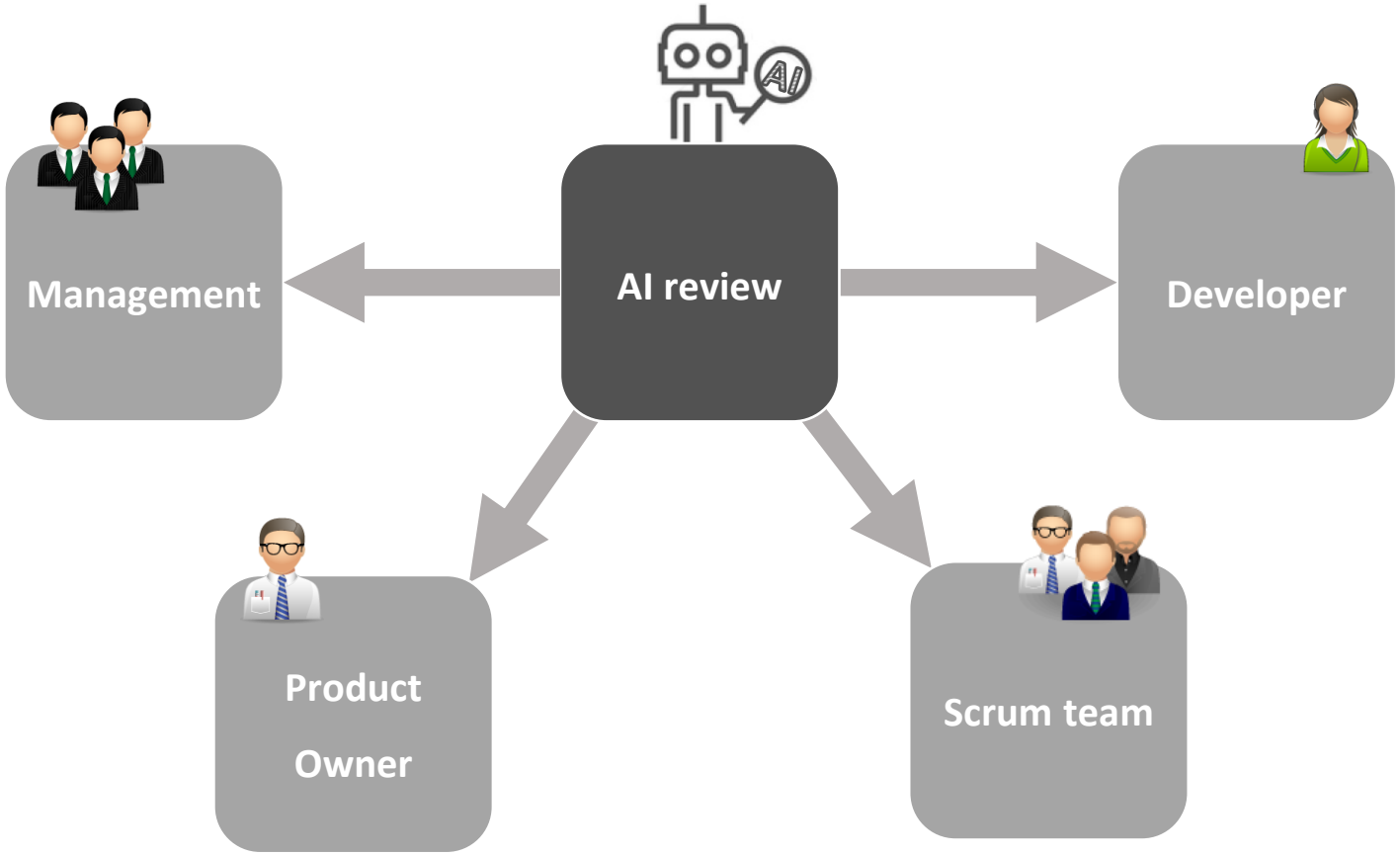
Tester’s objectives

- Where should I start testing?
- Which tests will give me results fastest?
- Do I need all my 10 000 test cases / scripts?
- Which parts of software are likely to have defects?

AI code quality metrics



AI code quality metrics



AI code quality metrics

SCRUM TEAM 

- Goal is to understand technical quality and test coverage of modules. Also, goal is to guide and focus testing:
  - Focusing technical debt reduction
  - Found defects per module
  - Guide for test coverage in automating system level testing.

Feature No.	Used modules										Number of tests	Priority
1	1	2	4	5	7	8	10				8	5
2	1	3	6								4	10
3	1	2	3	4	5	6	7	8	10		10	2
4	1	4	5	6							5	7
5	1	2	3	4	5	6	7	10			9	3
6	1	2	4	5	6	7	9	10			9	1
7	1	2	4	8	10						6	9
8	1	4	5	6	8						6	8
9	1	4	6	8	9						6	6
10	5	8	9								4	4

High probability of a defect
Normal probability of a defect

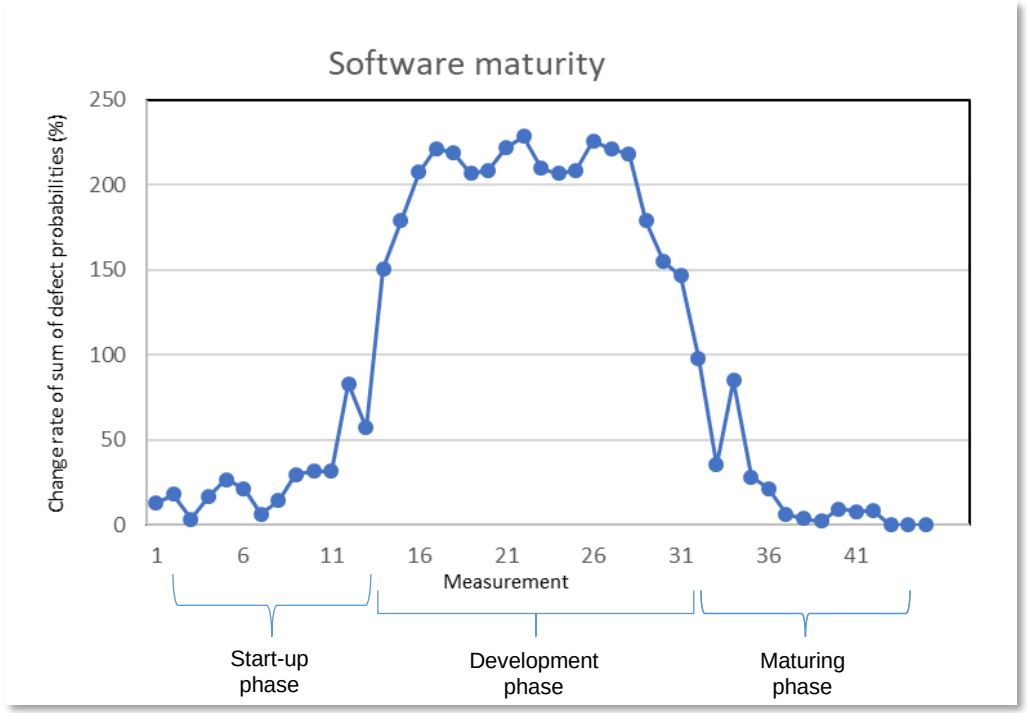
Module	Developer	Benchmark (order.numb.)	Relative risk	Number of reported bugs
Module_1	Jaakko	50	58	4
Module_2	Jaakko	4	5	1
Module_3	Mikko	180	95	7

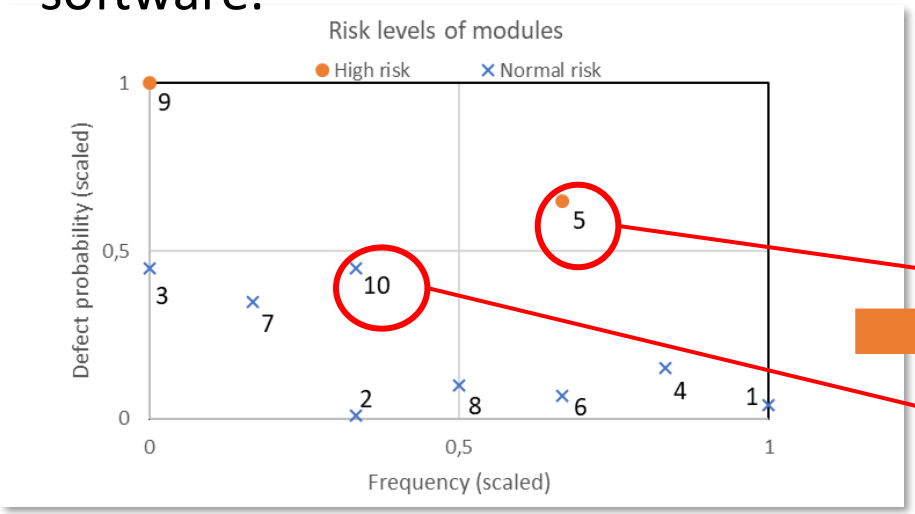
Requires that there is link from components (version control) and tests (test management) to feature descriptions and from reported defects to modules (observation control)

AI code quality metrics

MANAGEMENT

- Goal is to provide overview:
  - How fast the development is advancing
  - Maturity for release
  - Quality and technical debt of the software
  - Defect probabilities
  - Risks related to modifications of the software.





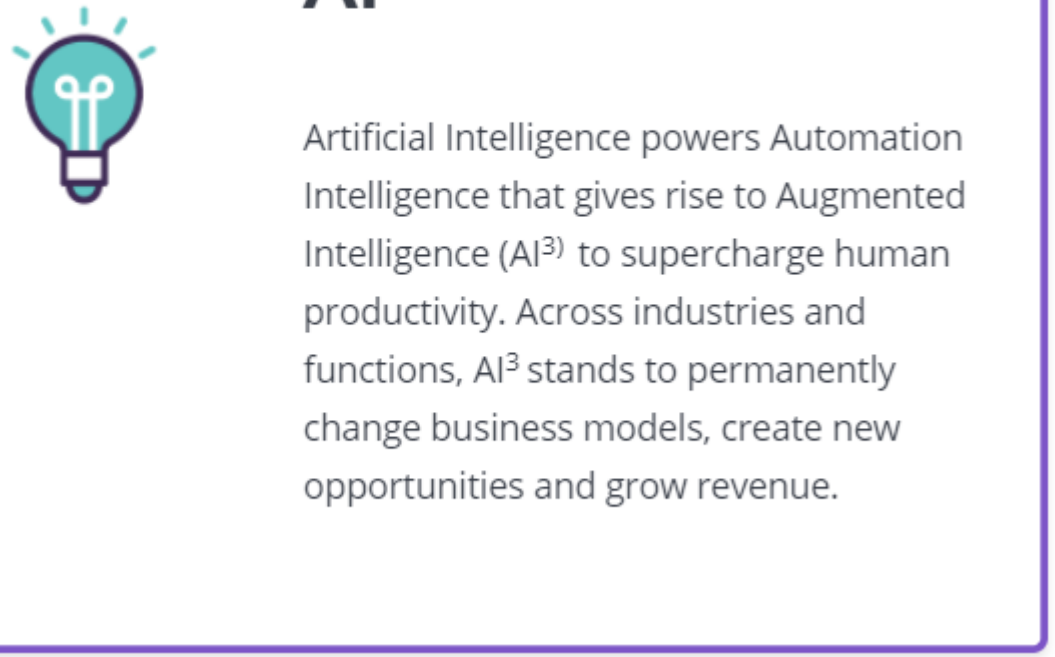
Interpretation:

- *Module 5* has high defect probability and is related to five most important features of the application. This results into high risk in release at this moment.
- Defect probability *of module 10* has fallen to acceptable level.

Predict quality issues

- Go through functionalities
- Identify risky areas
- Focus testing

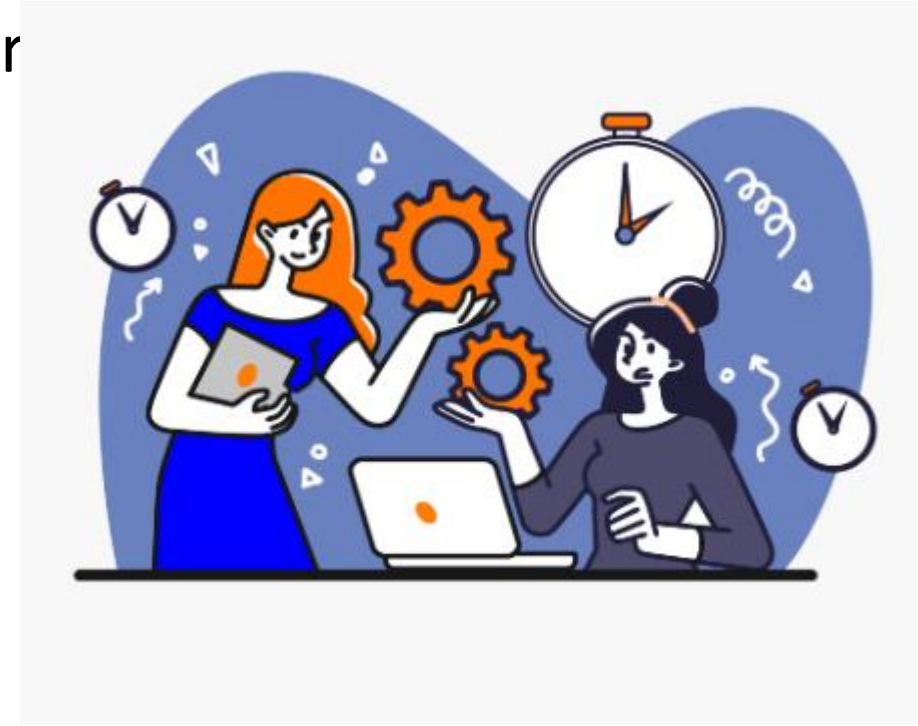
- Case: Eggplant
- https://www.eggplantsoftware.com/



Select most suitable test cases

- Analyze risk in commits to the code
- Select the tests that test the risky areas
- Reduce number of test cases needed to r

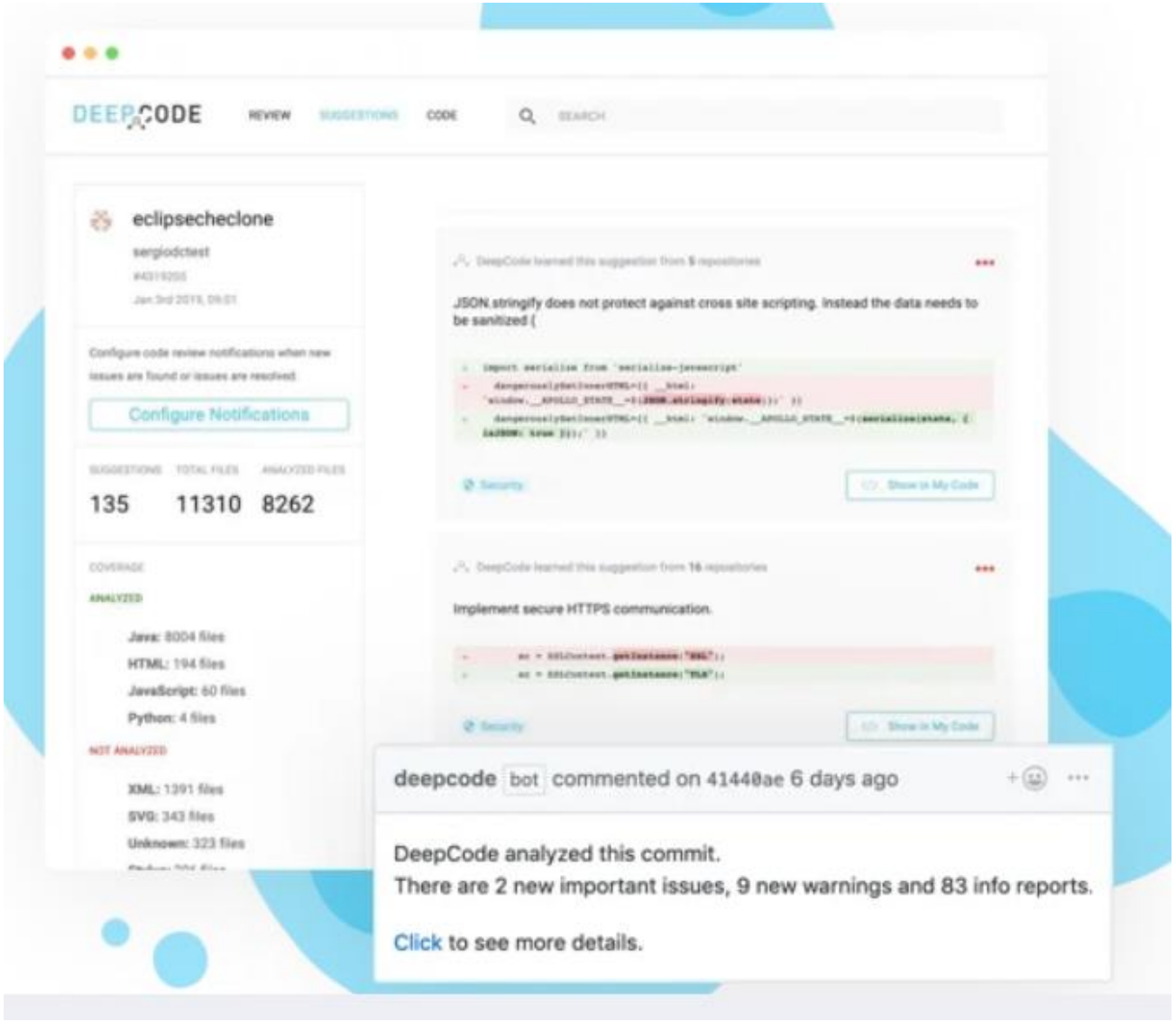
- Case: Appsurify
- https://appsurify.com/



Code analysis using AI

- Power of open source developers
- Machine learning algorithms learn from the community automatically
- Analyse code and propose improvements

- Case: DeepCode
- https://www.deepcode.ai/



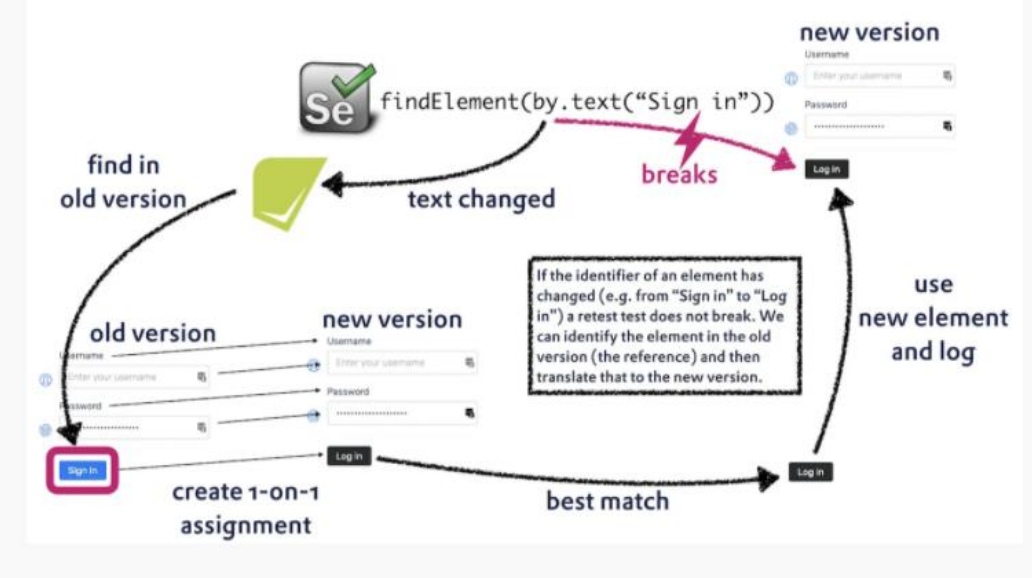
Fault-tolerant test automation

Tester’s objectives

- How can I get my regression test cases pass more often when the software under test changes?
- Can I use test automation scripts from another similar project?
- Can I test some generic test without my own scripting?

AI with regression testing

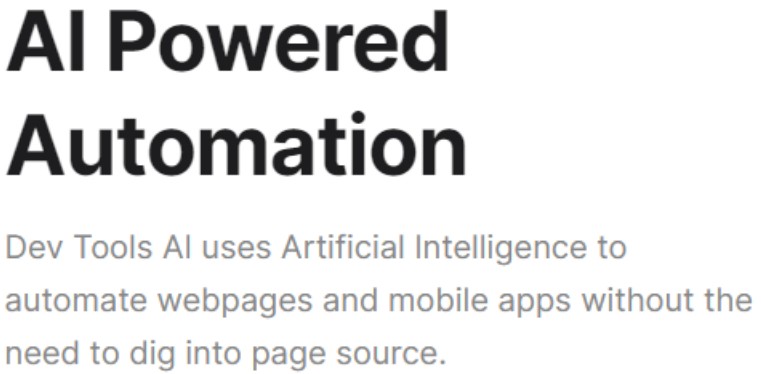
- AI-based test generation
- Checking functional differences
- Checking visual differences
- Automatically learning the test automation
- For: regression testing after changes



- Case: retest
- https://retest.de/ai-based-test-generation/

AI library helps pass Python test scripts

- Python-editor PyCharm can use Dev Tools AI library
- Asks which element to click on the screen and then keeps it running
- Suggests how to fix semiautomatically a changed test script that doesn't pass any more
- Reduces the need for test script maintenance
- Supports e.g. Cypress, Selenium, Appium, Playwright
- https://www.dev-tools.ai/



Intelligent testing

- Build test through interface backed by ML
  - Plain English
  - New life for “record-playback”
- ML algorithms maintain the tests

- Case: Functionize
- https://www.functionize.com/

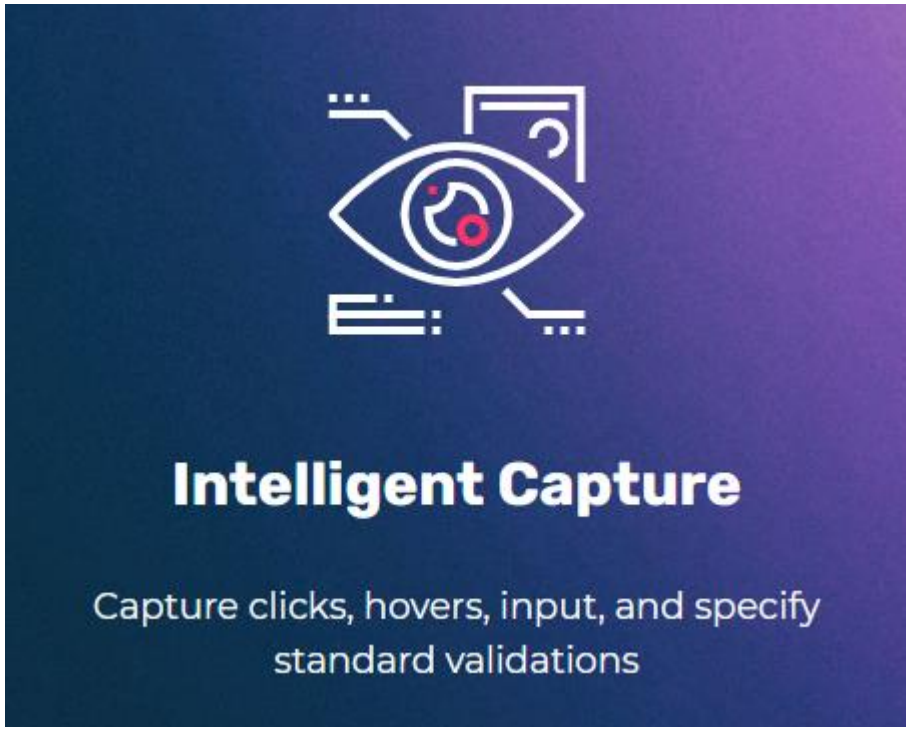


Image recognition with AI for open source test automation

- Supports test automation object recognition
- Helps with typical GUI recognition problems for test automation
- Can be used as a support library for existing test automation

- Cases:
  - ImageHorizonLibrary is a cross-platform library for Robot Framework.
  - Eficode, Oulu University and Business Finland
  - https://www.eficode.com/projects/testomat

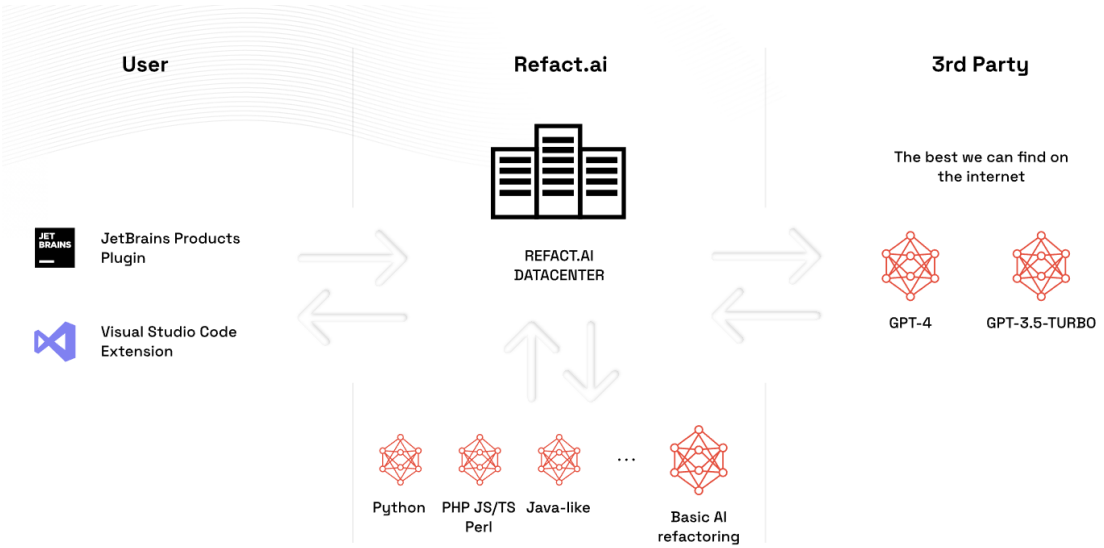
Faster test automation scripting with AI assistants

Tester’s objectives

- Can LLMs / Generative AI help me?
- I want to produce test scripts faster
- I want ideas for my test cases / scripts

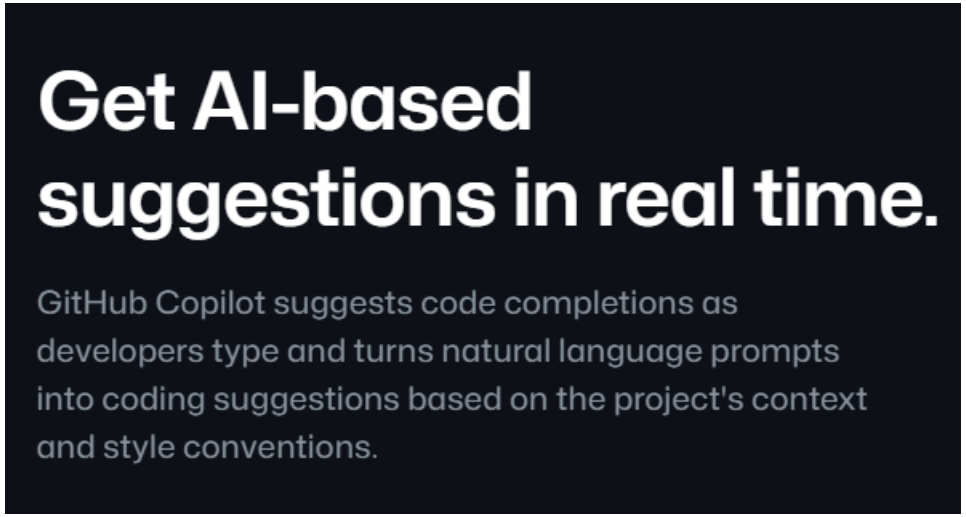
Python enhanced with ChatGPT

- Python-editor PyCharm has Refact.ai library that uses ChatGPT
- Understands Python and Robot Framework test automation setup
- Refact.ai is labelled as an AI coding assistant software
- https://refact.ai/



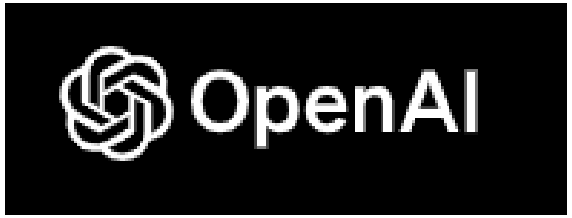
Github Copilot

- Assist test automation script creation
- Suggest what the user might want
  - What kind of code
  - What kind of test script
- User reviews and approves suggestions
- Based on Generative AI
- https://github.com/features/copilot



Generative AI helps in test creation

- Derive tests from specifications
- Create tests based on user input
- Enabled by good prompt engineering
- Challenge: repeated teaching
- User reviews AI proposals and uses in their tests
- https://chat.openai.com/ (ChatGPT)
- https://gemini.google.com/
- https://copilot.microsoft.com/ (Bing)



Autonomous testing

Tester’s objectives

- Can I have some tests done before I start my test scripting?
- Can I leave testing to AI?
- Are there some default tests that fit my product testing needs?

AI-powered mobile test automation platform

- “World’s first”
- Go automatically through mobile apps
- Use general test cases
- Reinforcement learning

- Case: test.ai
- https://www.test.ai/

Test automation assistants and bots

- Assist extending existing test automation sets
  - With e.g. Robot Framework, Playwright, Selenium, Cypress
- Use tests as self-driving test-bots
  - Exploring web apps
  - Regression testing
- Autonomous testing
- https://wopee.io/

Resources

Promoting and teaching AI and testing

- Trainings and certifications
  - https://www.istqb.org/certification-path-root/ai-testing.html
- Standards
  - ISO
    - AI/ML standard https://www.iso.org/standard/74438.html
    - Testing of AI-based systems https://www.iso.org/standard/79016.html
  - IEEE
    - What is AI Software Testing https://ieeexplore.ieee.org/document/8705808
    - AI testing perspectives https://ieeexplore.ieee.org/document/9514942
    - Application of AI in Testing https://ieeexplore.ieee.org/document/9676244

Some more research notes

- AI in Testing: Impact, Problems, Challenges and Prospect
  - https://www.researchgate.net/publication/357876318_Artificial_Intelligence_in_Software_Testing_Impact_Problems_Challenges_and_Prospect
- Utilizing AI in Software Testing
  - https://www.theseus.fi/handle/10024/263992
- AI Applied to Software Testing
  - https://dl.acm.org/doi/10.1145/3616372
- AI Applied to Testing: A Literature Review
  - https://ieeexplore.ieee.org/abstract/document/9141124
- ChatGPT helps testing
  - https://www.linkedin.com/pulse/gpt-4-sdlcs-secret-weapon-reinventing-testing-phase-andy-abbott/

Conclusion

- AI-enabled testing is already a reality
- Many companies are enhancing their solutions with AI
- New companies are set up around AI
- AI can provide simplicity to complex software development projects.
- With Generative AI, manual and automated test script creation is more productive
- With AI, testing activities can be focused on high-risk areas.
- With AI, test automation becomes more autonomous
- Testers are freed to create new tests

Any questions?

Follow and share the Kari's testing book projects:

- https://www.dragonsout.com
- https://www.act2lead.net/

Social media

- Kari https://www.linkedin.com/in/karikakkonen/
- Dragons Out https://www.facebook.com/DragonsOutOy

Ask questions:

kari.kakkonen@dragonsout.com