Agile ORLANDO 2023

JULY 24-28

PRESENTED BY Agile Alliance®

JOIN US TODAY!

#AGILE2023

LEARN MORE

NicoPaez

Technical Practices Walkthrough for non-technical People

Versioning Strategy for DevOps Implementations

Nicolás Paez
Department of Science and Technology
Universidad Nacional de Tres de Febrero
Caseros, Argentina
nicopaez@computer.org

UNTREF UNIVERSIDAD NACIONAL DE TRES DE FEBRERO

Flipped Classroom Experience Teaching Software Engineering

Nicolás Martín Paez
Department of Science and Technology
Universidad Nacional de Tres de Febrero
Saenz Peña, Buenos Aires, Argentina
nicopaez@computer.org

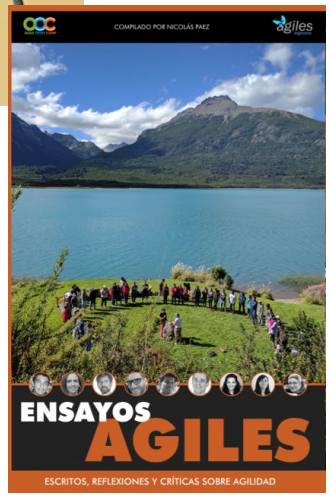
Technical and Organizational Agile Practices: A Latin-American Survey

Nicolás Paez, Diego Fontdevila, Fernando Gainey, and Alejandro Oliveros

Universidad Nacional de Tres de Febrero, Caseros, Argentina
nicopaez@computer.org,
{dfontevila, fgainey, aoliveros}@untref.edu.ar

.UBA fiuba FACULTAD DE INGENIERÍA

IEEE computer society



Software Engineering Education in the DevOps Era: Experiences and Recommendations

Nicolás Paez[1,2], **Carlos Fontela**[2]

[1]Departamento de Ciencia y Tecnología - Universidad Nacional de Tres de Febrero
Saenz Peña, Argentina

[2]Facultad de Ingeniería - Universidad de B
Buenos Aires, Argentina

nicopaez@computer.org, cfontela@

HELENA Study: Initial Observations of Software Development Practices in Argentina

Nicolás Paez, Diego Fontdevila and Alejandro Oliveros

Universidad Nacional de Tres de Febrero, Caseros, Argentina
nicopaez@computer.org, {dfontdevila,aoliveros}@untref.edu.ar

RADIO CUT

Microsoft

SOUTHWORKS
Development on Demand

What is a technical practice?

Technical practice:

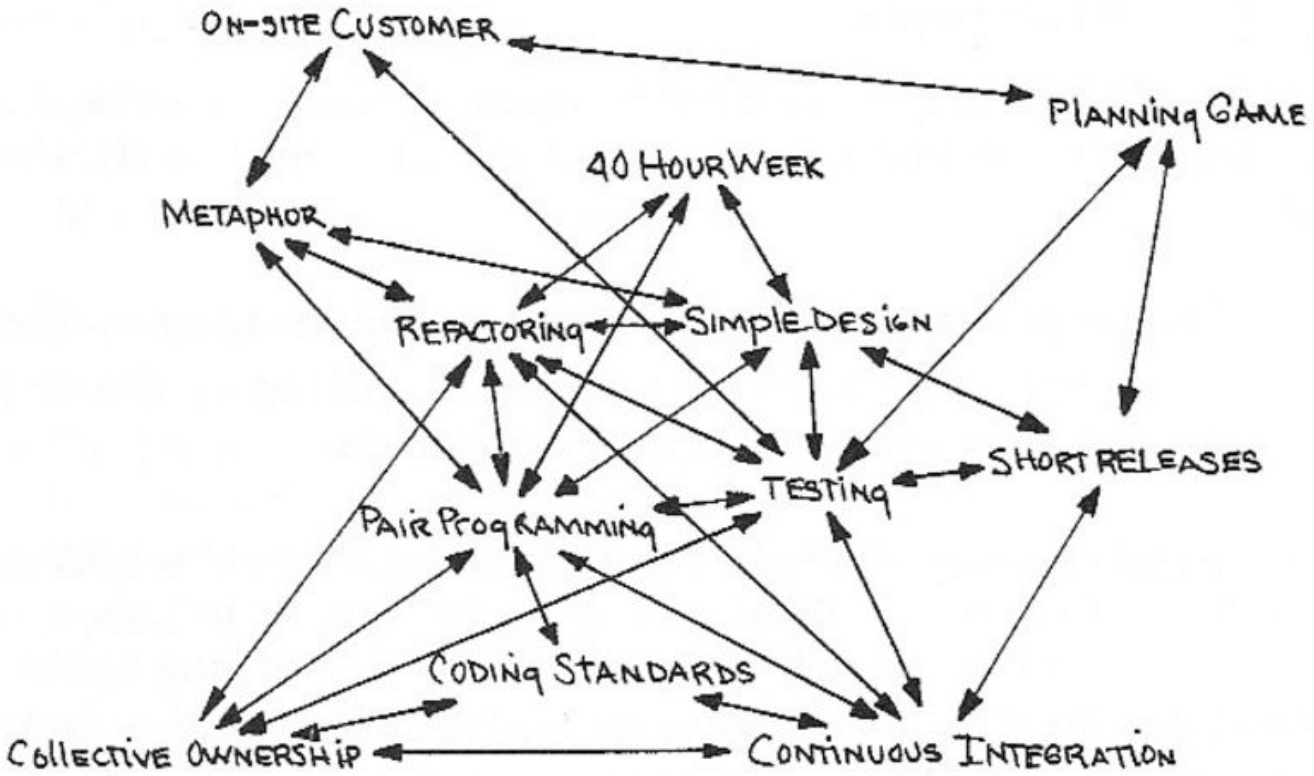
A practice that **is specific to software projects** and that is not applicable to other kind of projects

https://link.springer.com/chapter/10.1007/978-3-319-91602-6_10

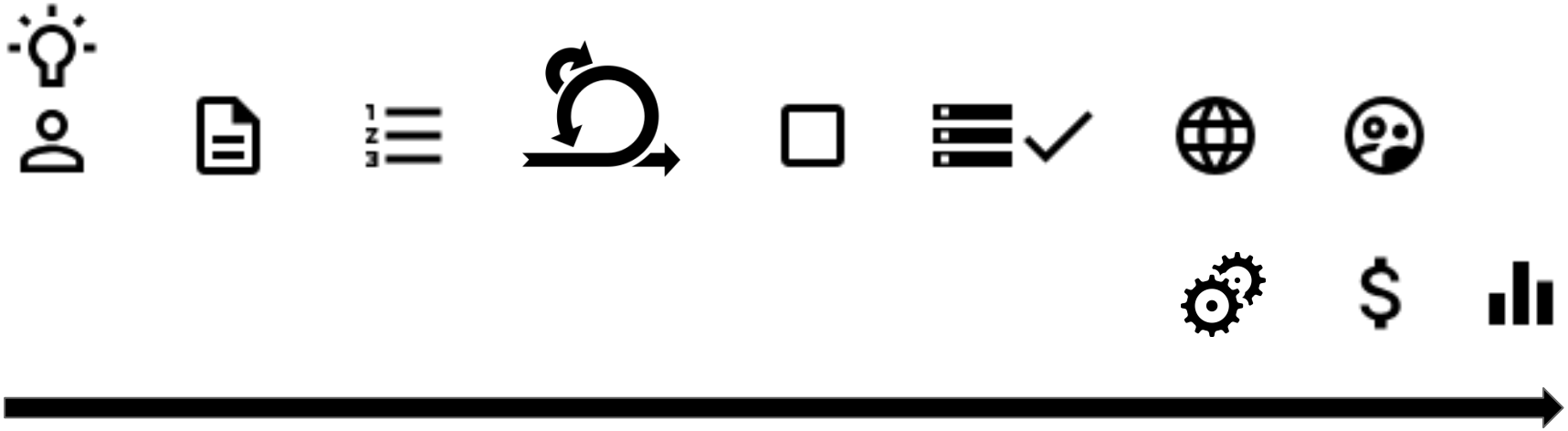
Mainstream

vs.

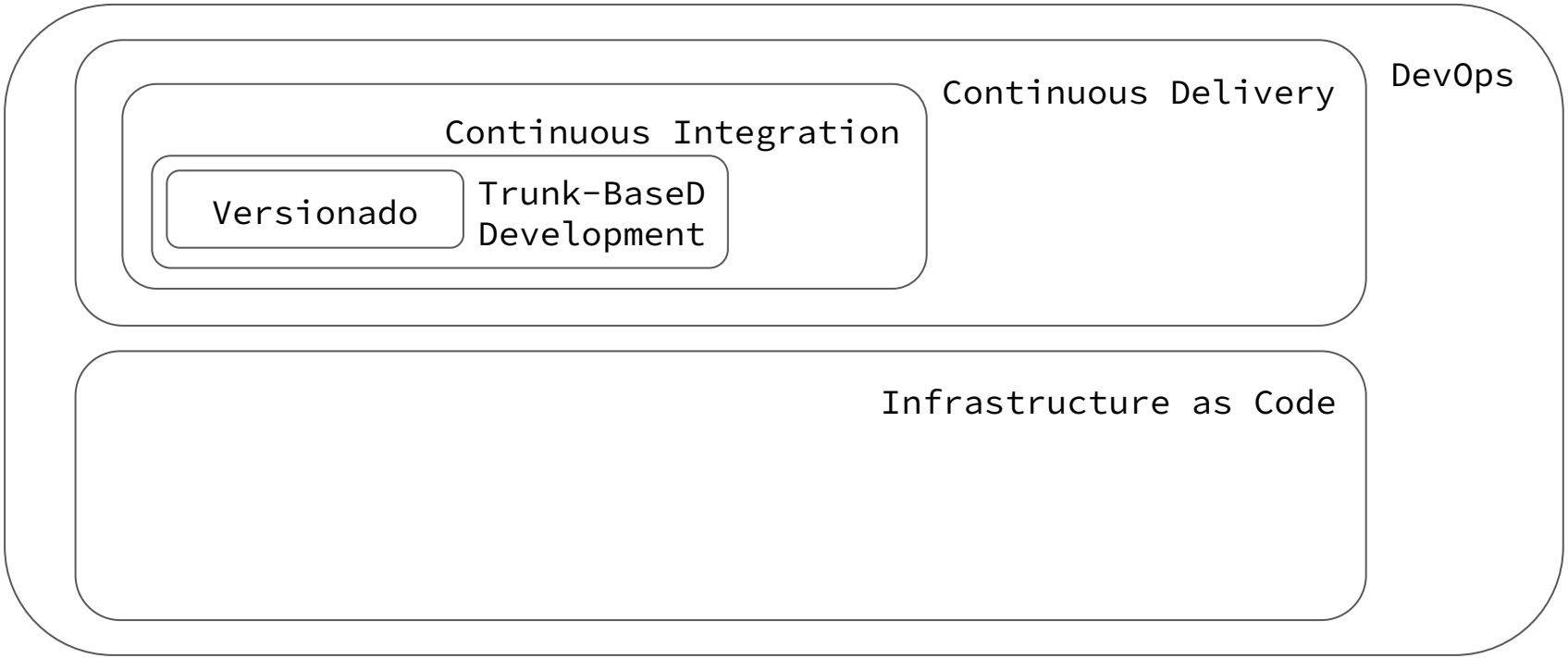
State of the Art



XP Practices, extracted from the book *Extreme Programming Explained: Embrace Change*



@inicopaez (cc) BY-SA


DevOps
Continuous Delivery
Continuous Integration
Versionado
Trunk-BaseD
Development
Infrastructure as Code

DevOps

*DevOps is a **set of practices** intended to **reduce the time** between committing a change to a system and the change being placed into normal production while ensuring high quality.*

Len Bass, SEI-CMU

https://www.amazon.com/DevOps-Software-Architects-Perspective-Engineering/dp/0134049845

DevOps

Continuous Delivery

Software engineering approach in which teams produce software in short cycles, ensuring that the software can be reliably released ***at any time*** *and, following* ***a pipeline through a "production-like environment"****, without doing so manually*

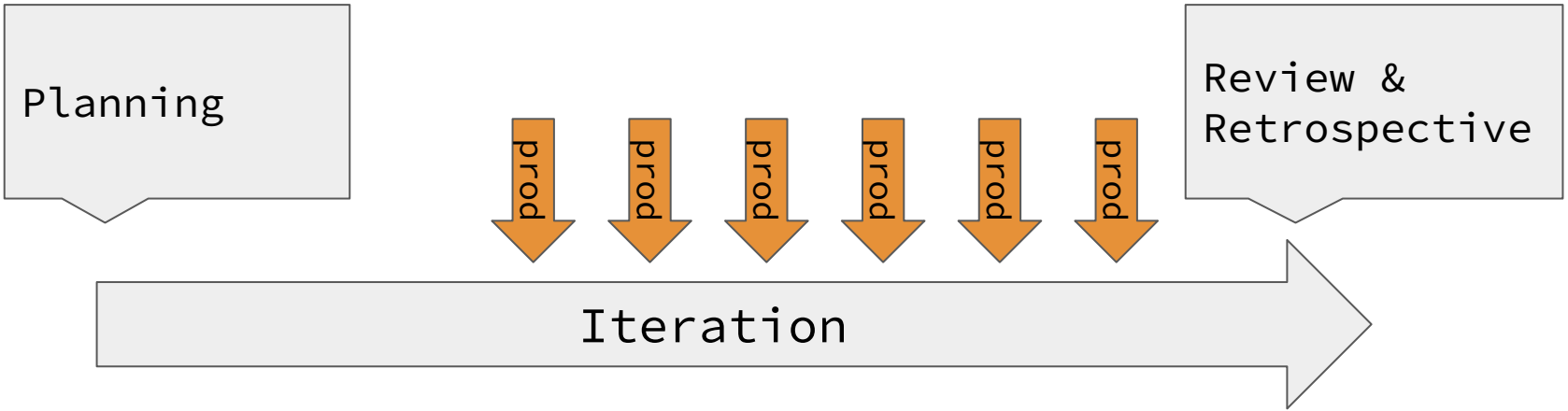
Infrastructure as Code

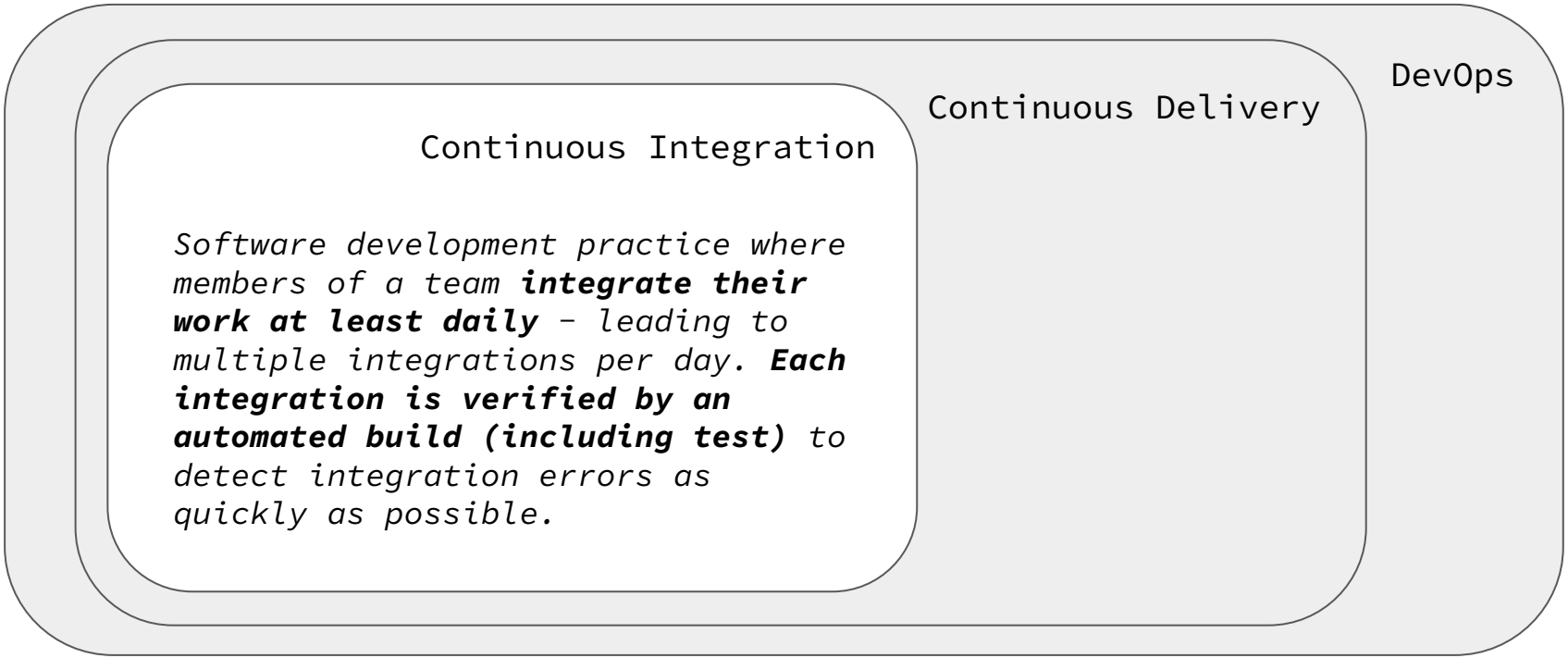
Continuous Delivery

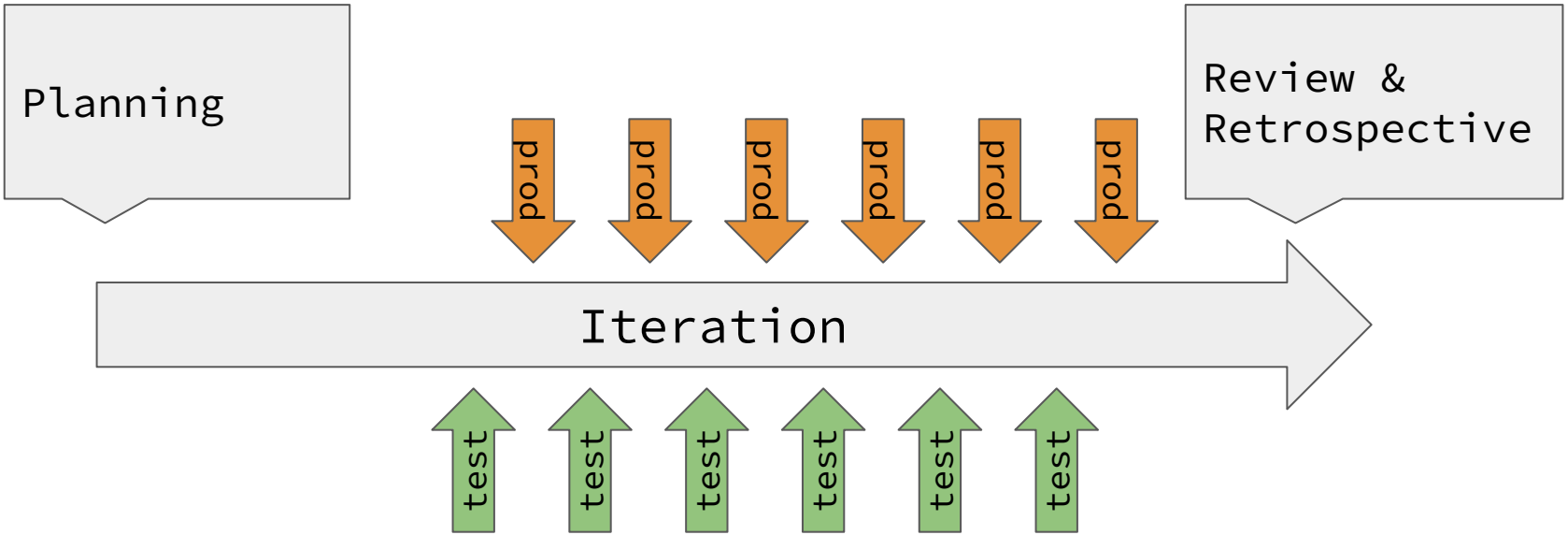
vs.

Continuous Deployment

https://gitlab.com/nicopaez/jobvacancy/-/pipelines/931857712

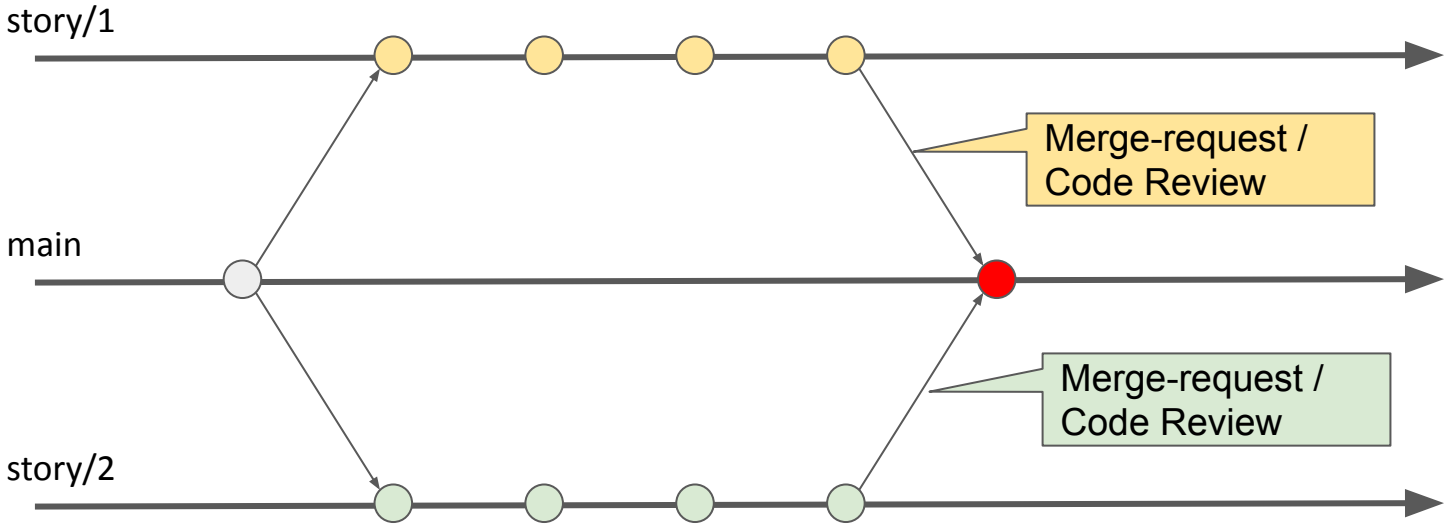

Planning
prod
prod
prod
prod
prod
prod
Review &
Retrospective
Iteration


DevOps
Continuous Delivery
Continuous Integration
*Software development practice where members of a team **integrate their work at least daily** - leading to multiple integrations per day. **Each integration is verified by an automated build (including test)** to detect integration errors as quickly as possible.*

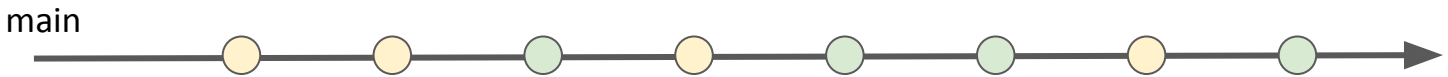

Planning
Review &
Retrospective
prod
prod
prod
prod
prod
prod
Iteration
test
test
test
test
test
test

feature branches vs trunk-based

Feature-branches, merge-request



Trunk-based Development



Trunk-based Development

Small Increments / Releases

Test-Automation

Pair/Mob Programming

Feature Toggles

Trunk-based Development

Small Increments / Releases

Test-Automation

Pair/Mob Programming

Feature Toggles

Decouple Release and Deploy

A/B Testing

Testing in Production

Canary Releases

Continuous Delivery

Strengthening the Case for Pair Programming

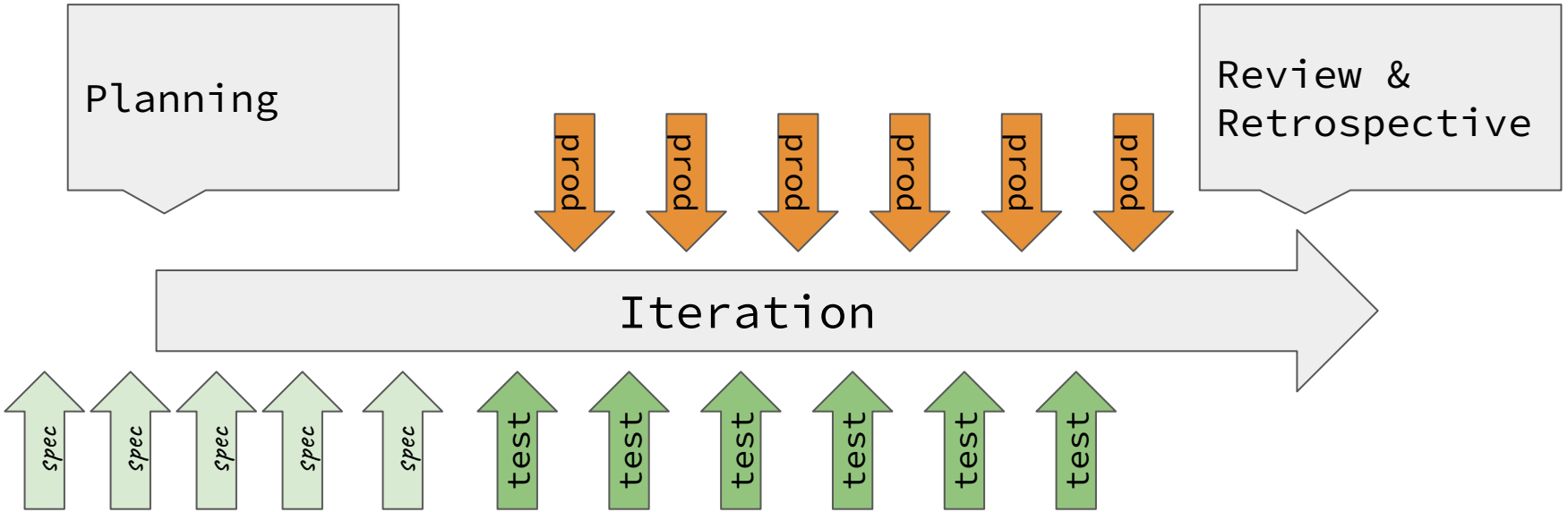
The software industry has practiced pair programming—two programmers working side by side at one computer on the same problem—with great success for years. But people who haven't tried it often reject

Laurie Williams, *North Carolina State University*
Robert R. Kessler, *University of Utah*
Ward Cunningham, *Cunningham & Cunningham*
Ron Jeffries

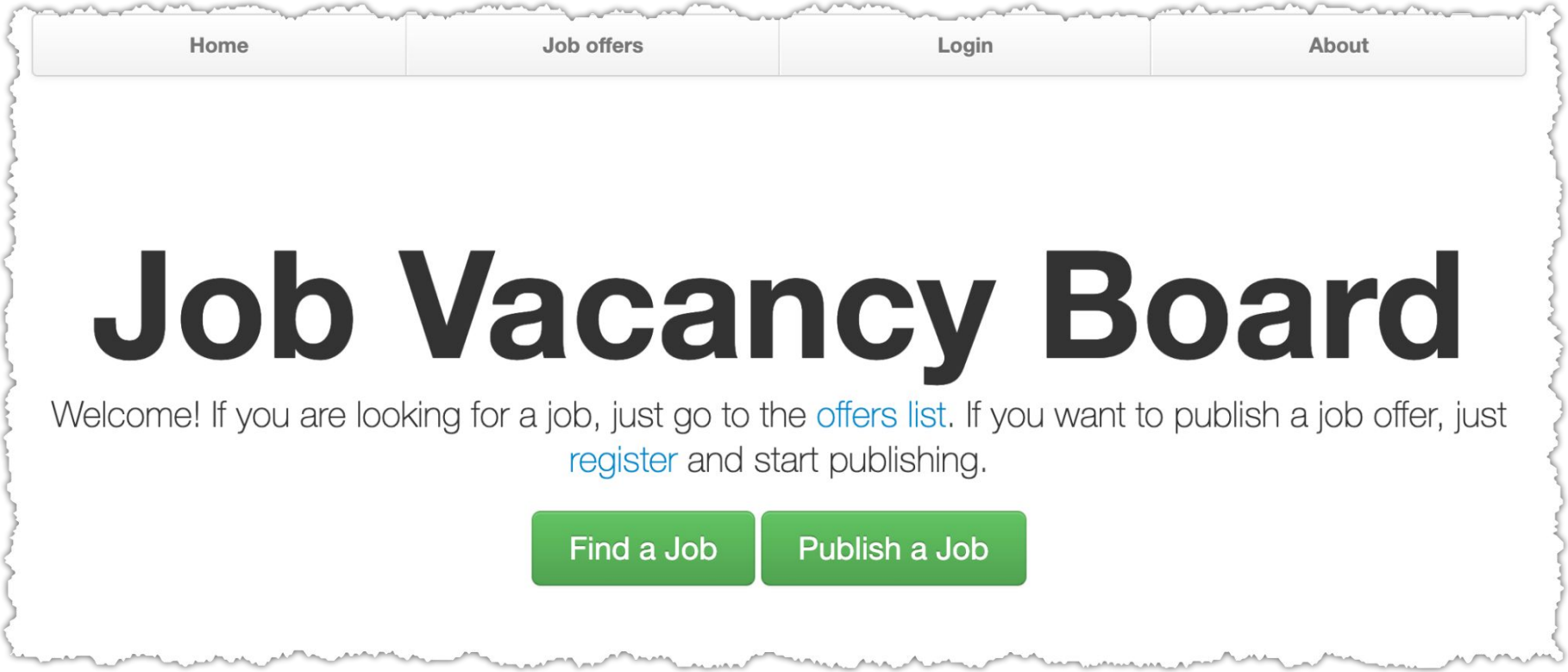
For years, programmers in industry have claimed that by working collaboratively, they have produced higher-quality software products in shorter amounts of time. But their evidence was anecdotal and subjective: "It works" or "It feels right." To validate these claims, we have gathered quantitative evidence showing that pair programming—two programmers working side by side at one computer on the same

https://ieeexplore.ieee.org/document/854064

Feature Toggles (showtime!)


Planning
Review &
Retrospective
prod
prod
prod
prod
prod
prod
Iteration
spec
spec
spec
spec
spec
test
test
test
test
test
test

Meet the JobVacancy



https://jobvacancy.com.ar/

Practice:
Specification By Example

Fin

Nicolás Paez
nicopaez@ieee.org
@inicopaez
blog.nicopaez.com

Agile Alliance PRESENTS

Agile ORLANDO JULY 24-28 2023

Join Agile Alliance today!

Become an Agile Alliance member and help support our non-profit mission, while gaining access to valuable benefits like online events, in-person conference discounts, and event session videos.

#AGILE2023

```
29a697c32fcaabcc25649103a0239c7d2bc5dc63 => scenarios

315cd52e9a785d8d1b5c2d7ccc3918f99cf92c64 => steps
```