

BDD your solution from git init to Kubernetes





Versioning Strategy for DevOps Implementations

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

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 ^(CC), Diego Fontdevila, Fernando Gainey, and Alejandro Oliveros

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

Characterizing Technical and Organizational Practices in the Agile Community

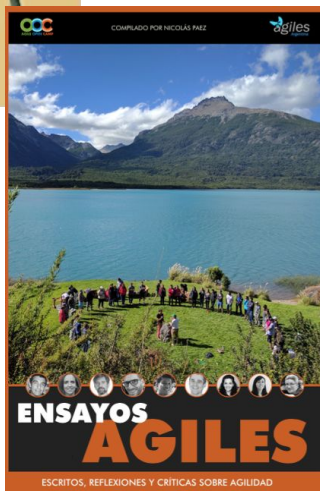
Nicolás Paez, Diego Fontdevila, Alejandro Oliveros
Departamento de Ciencia y Tecnología
Universidad Nacional de Tres de Febrero
nicopaez@computer.org, dfontdevila@untref.edu.ar, aoliveros@untref.edu.ar

HELENA Study: Initial Observations of Software

Development Practices in Argentina

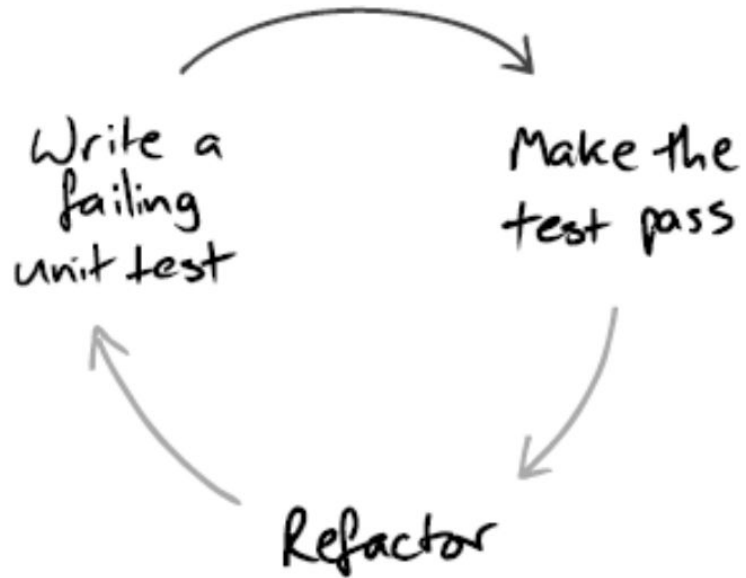
Nicolás Paez, Diego Fontdevila and Alejandro Oliveros

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



About us

Test-Driven Development

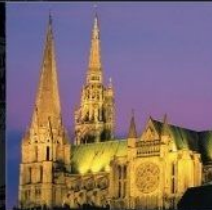


The Addison-Wesley Signature Series

TEST-DRIVEN DEVELOPMENT

BY EXAMPLE

KENT BECK



A KENT BECK
SIGNATURE
BOOK



Software Engineering Radio

The Podcast for Professional Software Developers

[Home](#)[About SE-Radio](#)[Team](#)[Advertise](#)[Recognition](#)[Contact](#)[Legal](#)

Episode 167: The History of JUnit and the Future of Testing with Kent Beck

Filed in [Episodes](#) by [SE-Radio](#) on September 26, 2010 • [28 Comments](#)

Recording Venue: Skype

Guest: Kent Beck



Premise #0:

The result is a consequence of the process

Premise #1:

Development is guided by tests

Premise #2:

The process is iterative and
the increments are small

Premise #3:

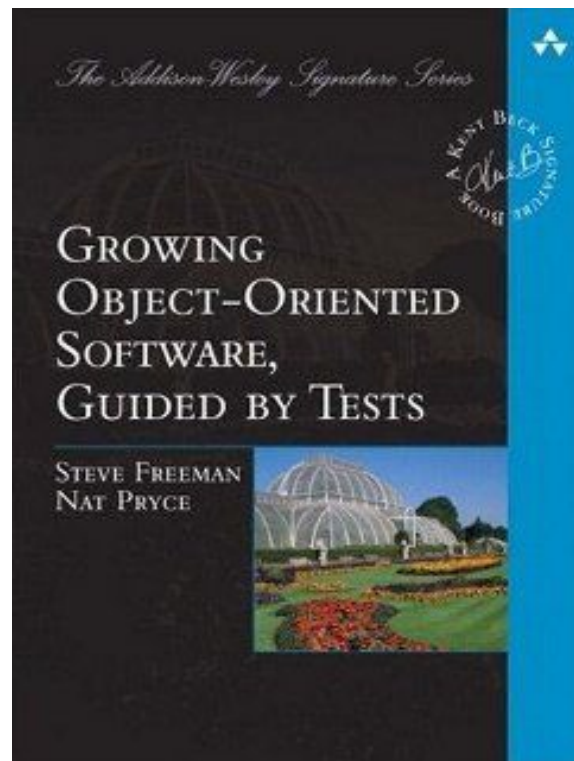
Continuous integration, verification and
delivery

Premise #4:

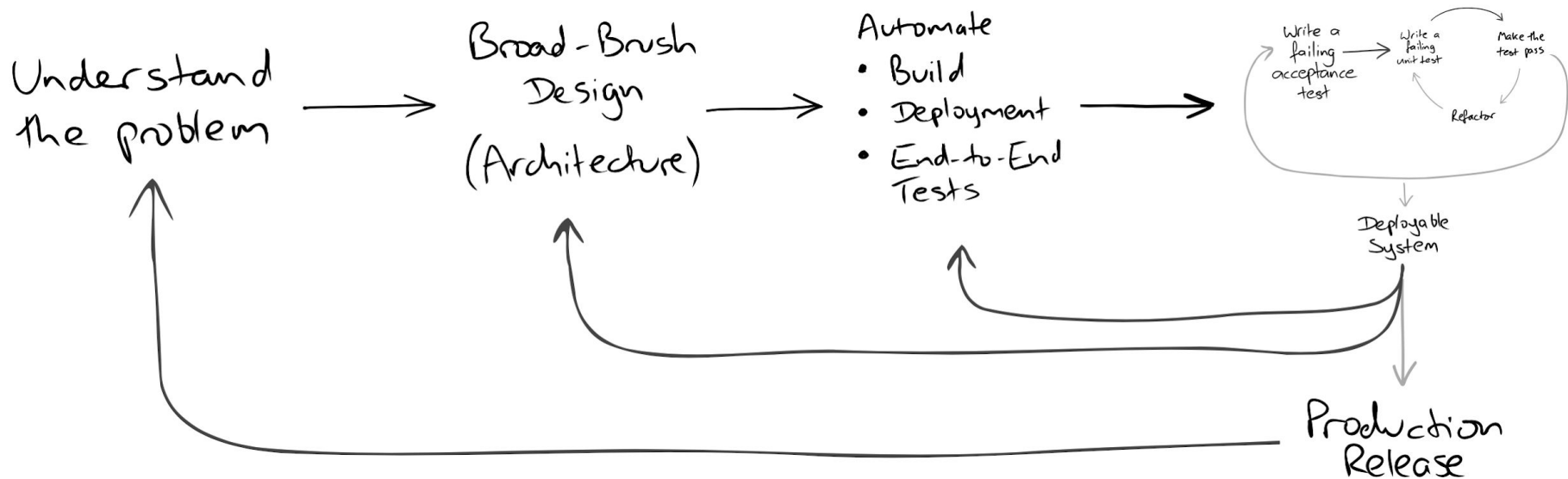
Keep It Simple, Stupid (KISS)

Techniques

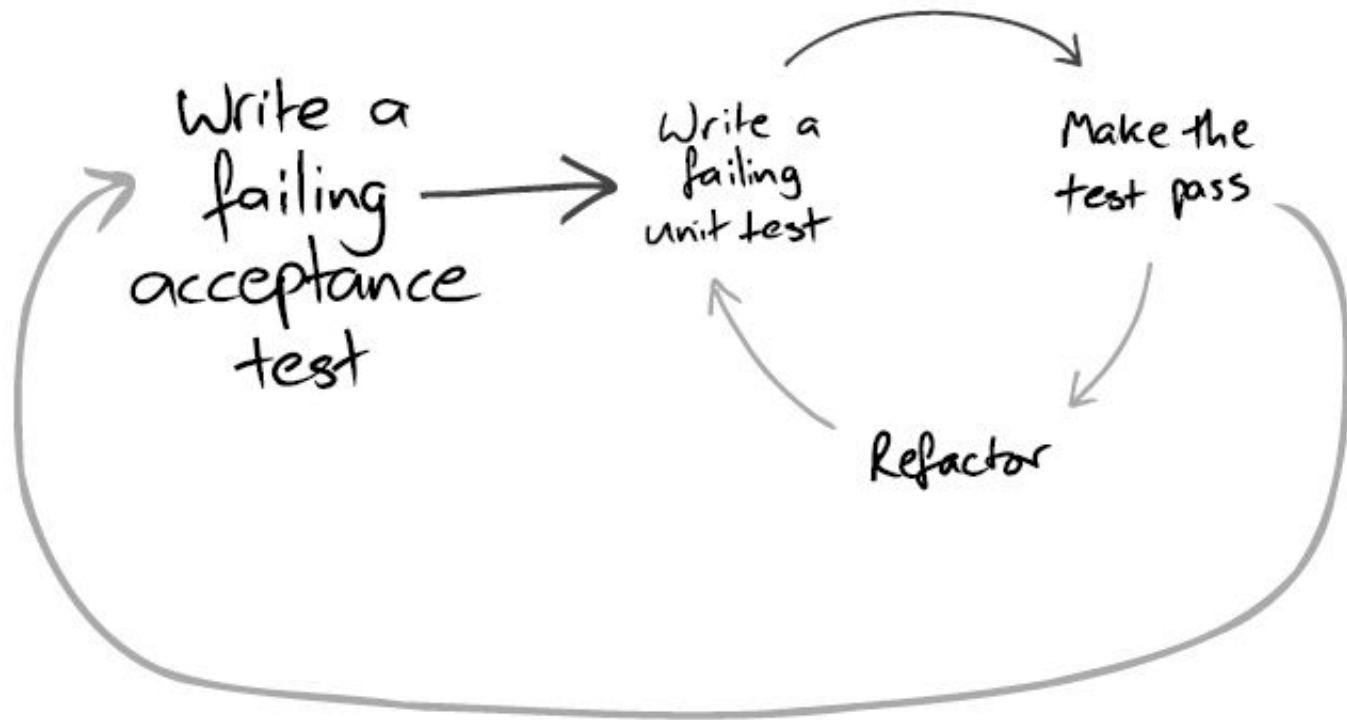
- BDD / TDD
- Continuous Integration / Continuous Delivery
- Walking Skeleton
- Hexagonal Architecture



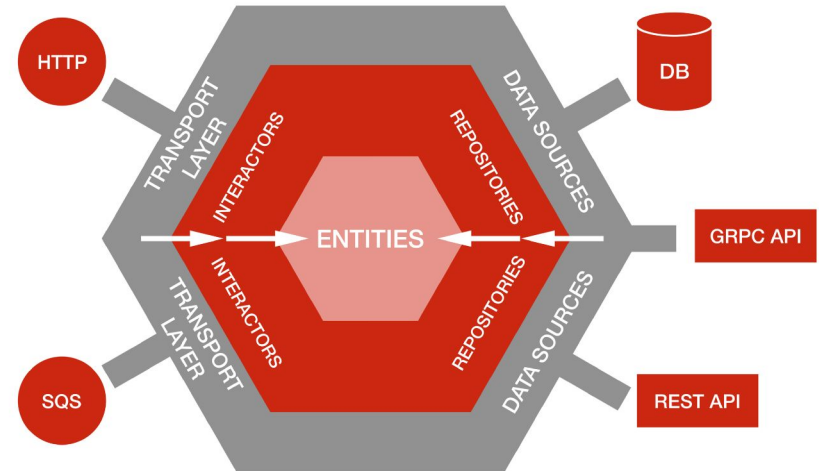
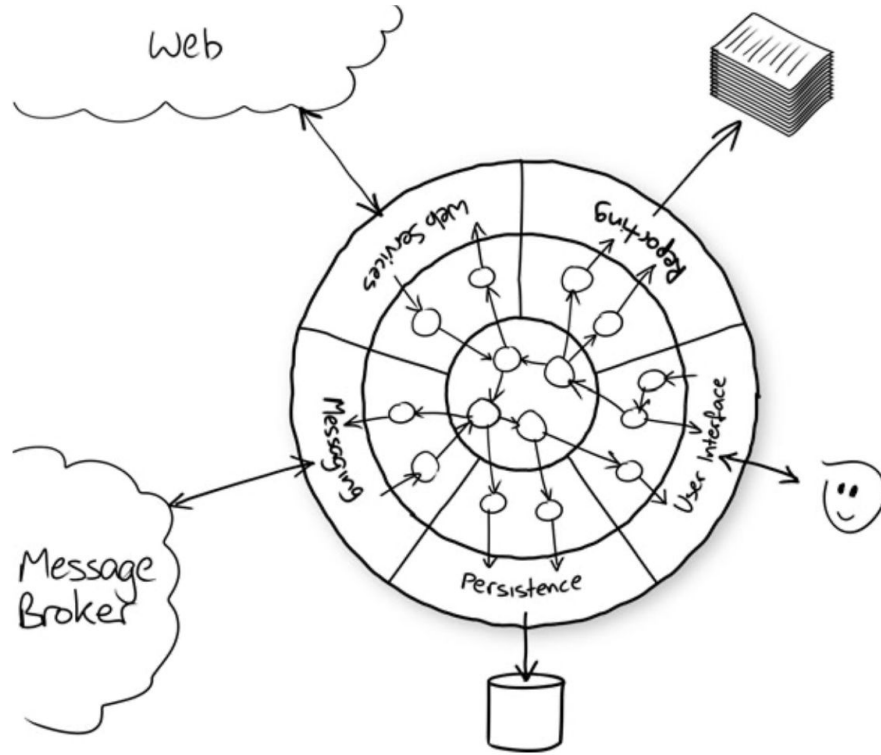
High level process



BDD / TDD Cycle



Hexagonal Architecture



Walking Skeleton / Tracer Bullet

Minimal feature, end-to-end*,

Integrated, tested and deployed

Goal:

Mitigate risks, Minimize uncertainty, set the basis

Hands-on Code

Quickstart

1. git clone
2. Check Readme
3. Run test
4. Run the app
5. Set the team name

Ruby => app.rb

Java => pom.xml

6. Run test
7. Commit & Push

Meanwhile let's see the project structure

Case Study: Advisor App

WebApi that based on user preferences
(query param) and current temperature
(external weather API) advise on what
kind of cloth to wear.

Rules:

Preference: warm or cool

Temp: (under 10); (10,20); (over 20)

Examples:

Given preference is *warm*

When temperature is under 10° C

Then take a *coat*

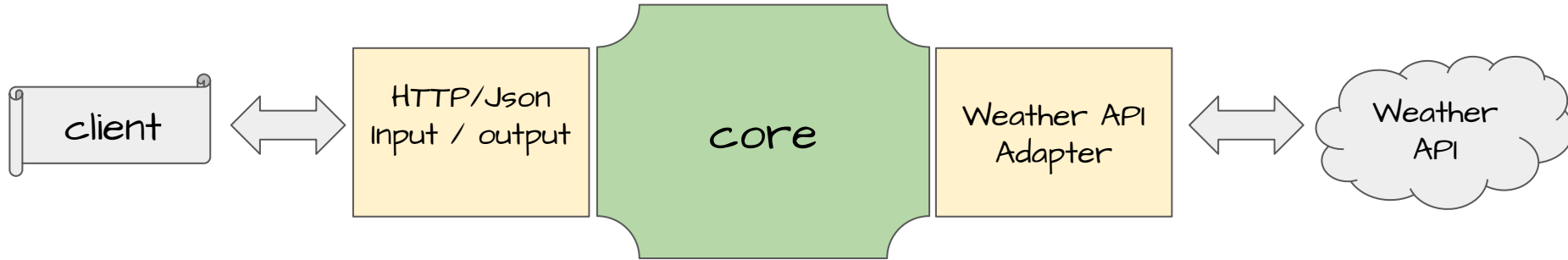
Given preference is *cool*

When temperature is under 10° C

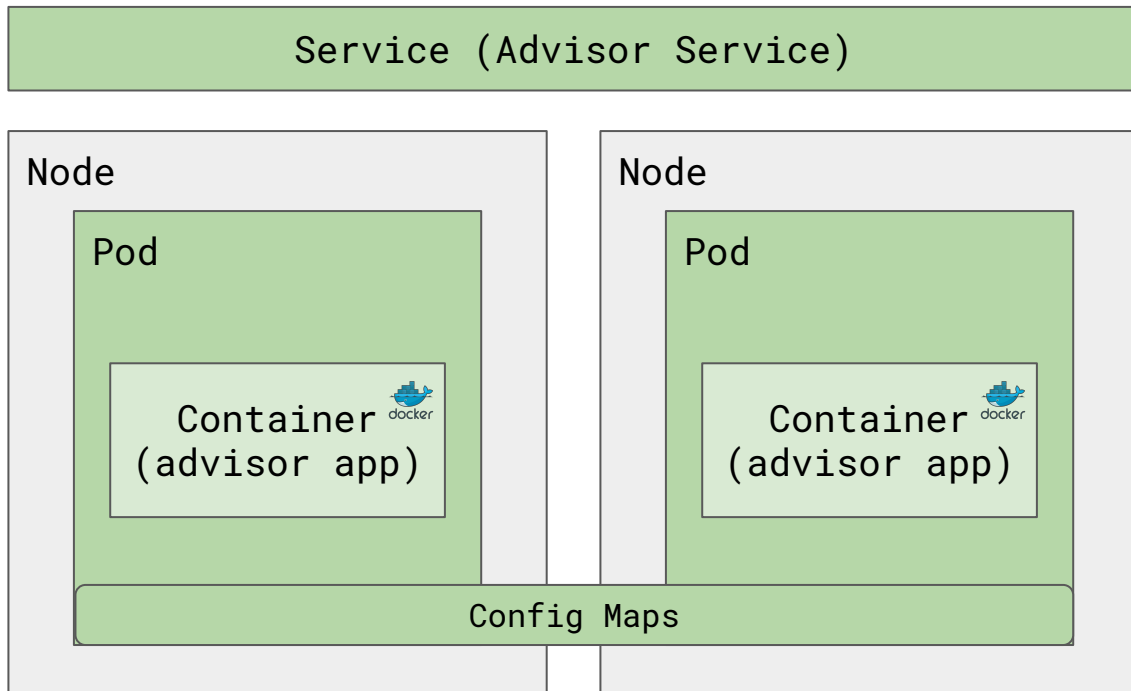
Then take a *t-shirt*



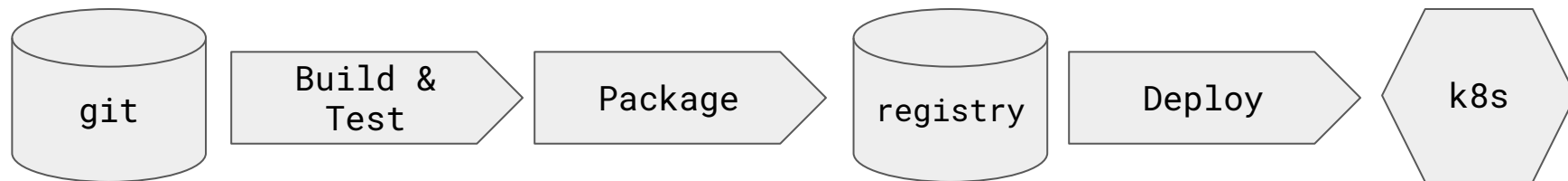
Hexagonal Architecture



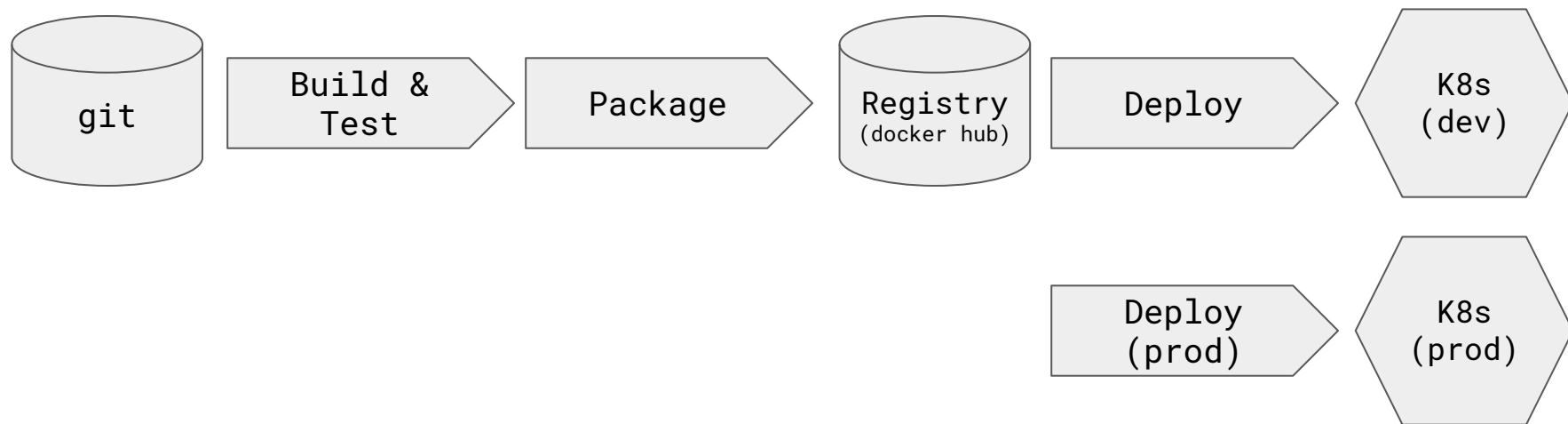
Runtime Infrastructure



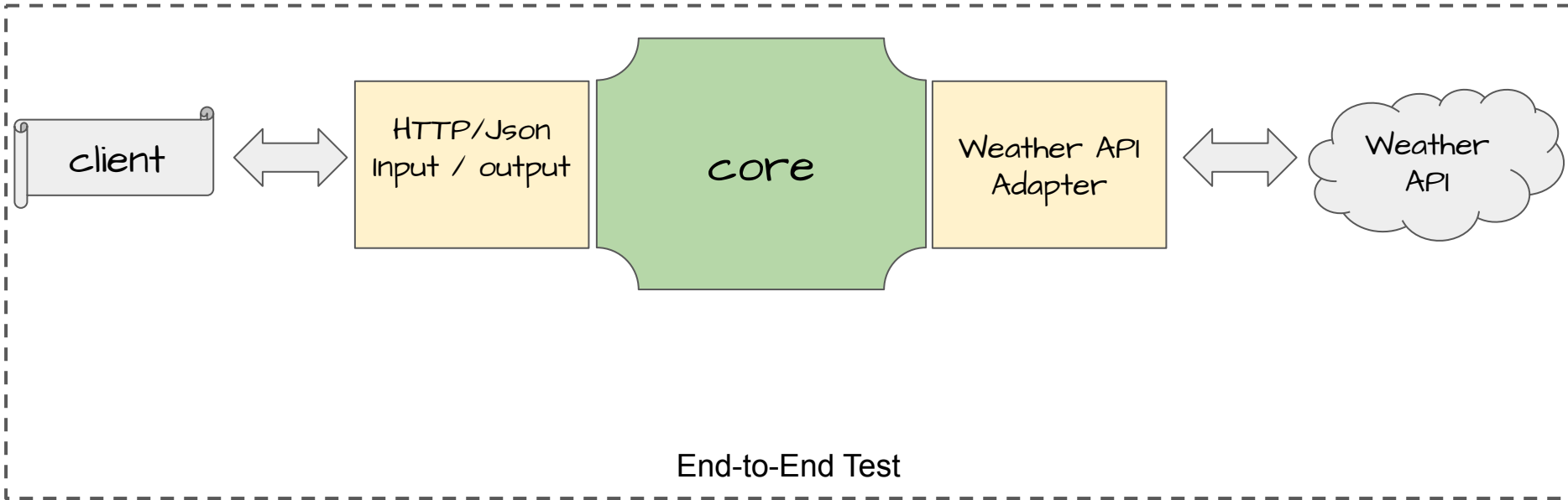
Deployment Pipeline



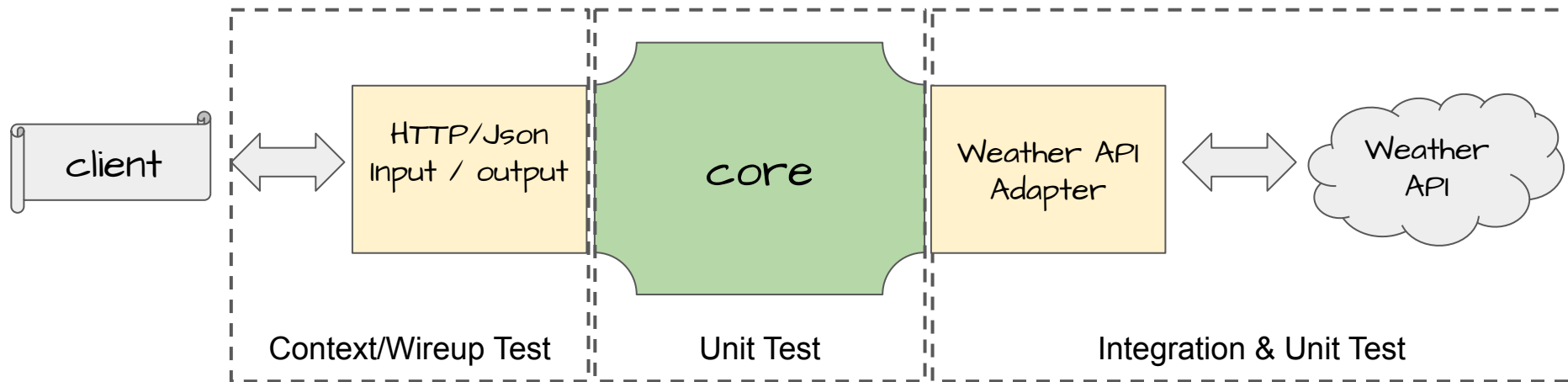
Deployment Pipeline



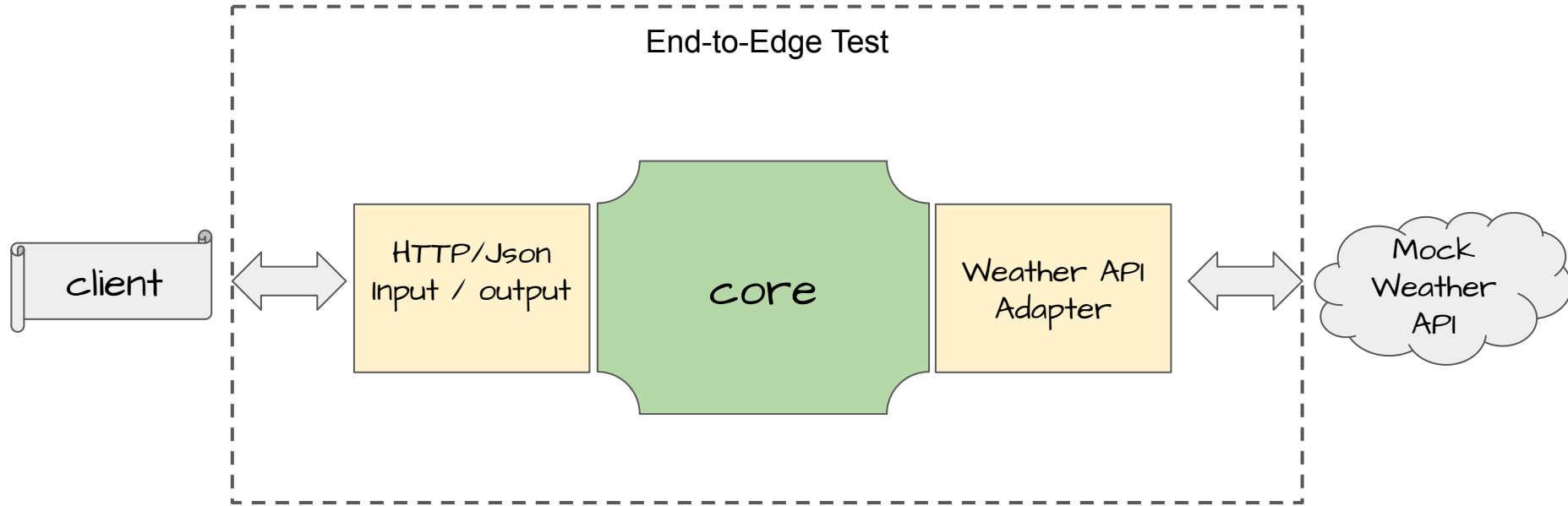
Hexagonal Architecture



Hexagonal Architecture



Hexagonal Architecture



The End

nicopaez@computer.org

@inicopaez

blog.nicopaez.com

Feedback form:

<http://nicopaez.com.ar/wbfgtkf>

