



Don't worry, be happy

Independently Developing &
Rigorously Testing your application in isolation

Iñaki de Tejada & Alessandro Vermeulen
20th June 2024



do your thing

Iñaki De Tejada



Alessandro Vermeulen





Content

- Context
- Why: 3 problems to solve
- Target: A state of independence and safety
- How to get there?
- Design: Prefer industry standards.
- What: The Conformance TestKit.
- Missing the chaining of contract progress?
- Demo

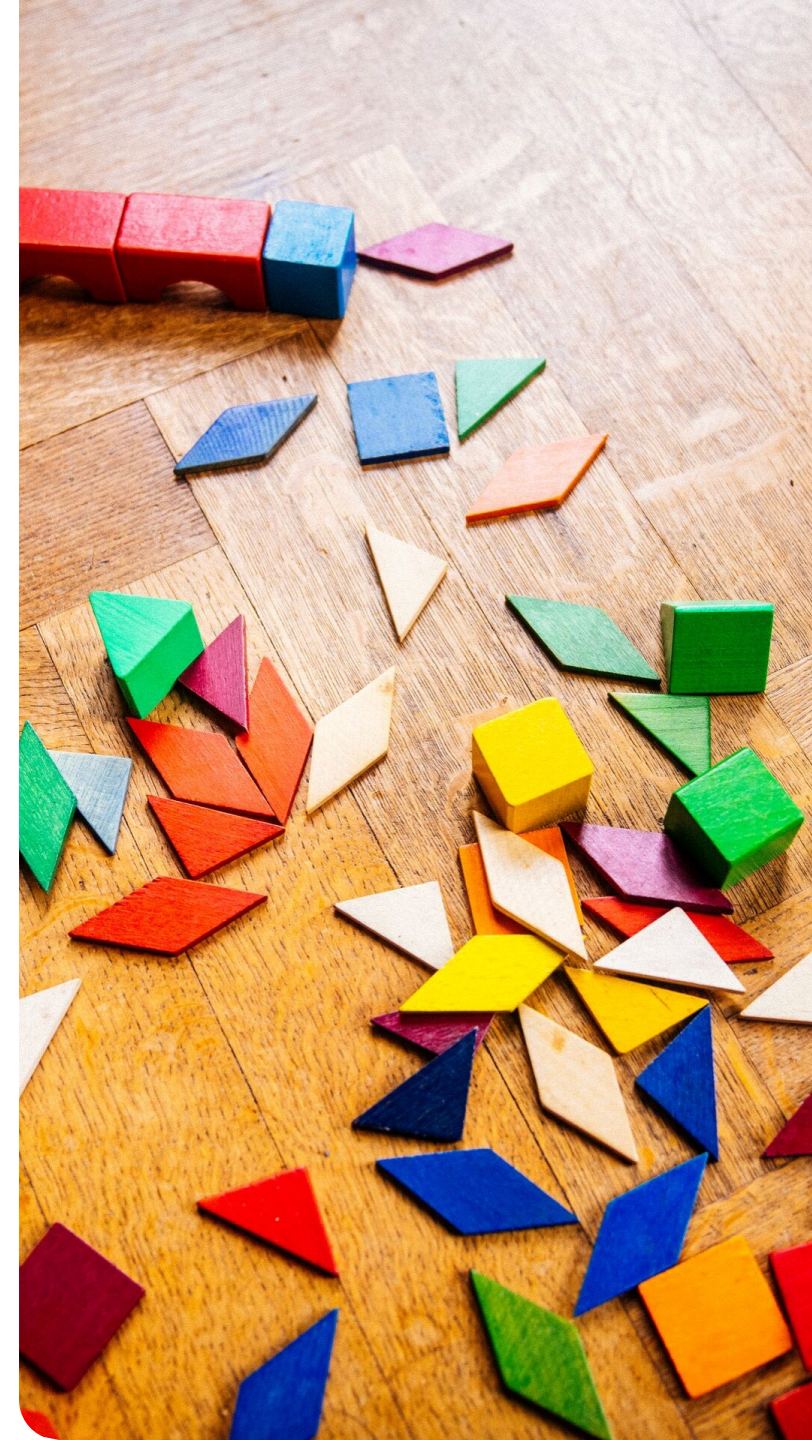


ING is quite large

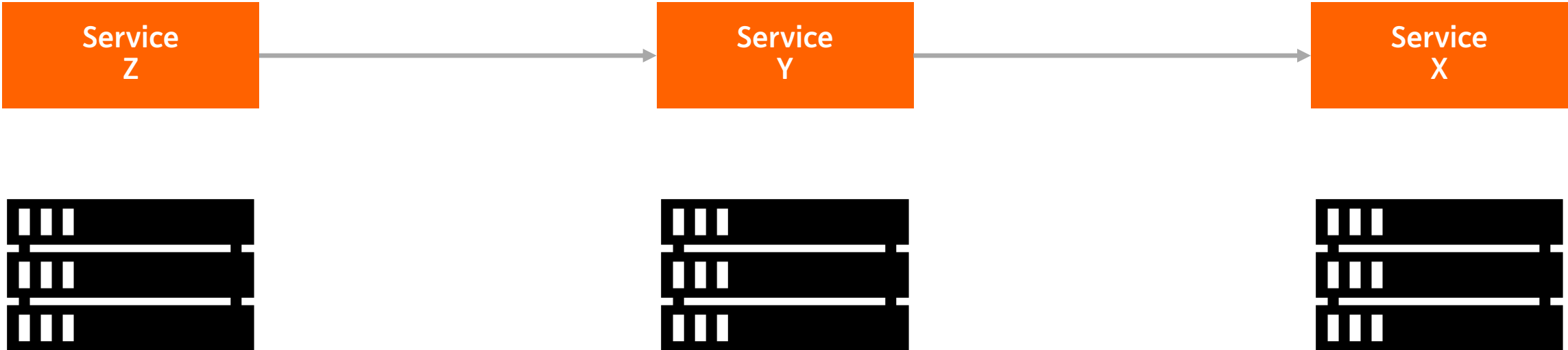
- ING is a big organisation, not only the banking side but also on the IT side:
 - 900+ teams connecting and collaborating
 - 1800+ running services in production in the mesh
 - 15k+ endpoints
 - Cross-country & cross-continent
 - Development of a new feature depends on sequential implementation of features
- Migrations are a fact of life
 - From Continuous Delivery to Continuous Migration
- Already used to governing things in API Management
 - API definitions
 - Service Subscriptions

Why: 3 problems to solve.

1. Slow feature delivery
 - Quality improvement vs new feature development
2. Complicated setup, always moving:
 - Applications need to adhere to many requirements to behave as a proper citizen of the ecosystem
3. Inconsistent Quality Assurance
 - Different interpretation of requirements
 - Quality of tests doesn't cover properly, both breadth and depth



A typical application chain



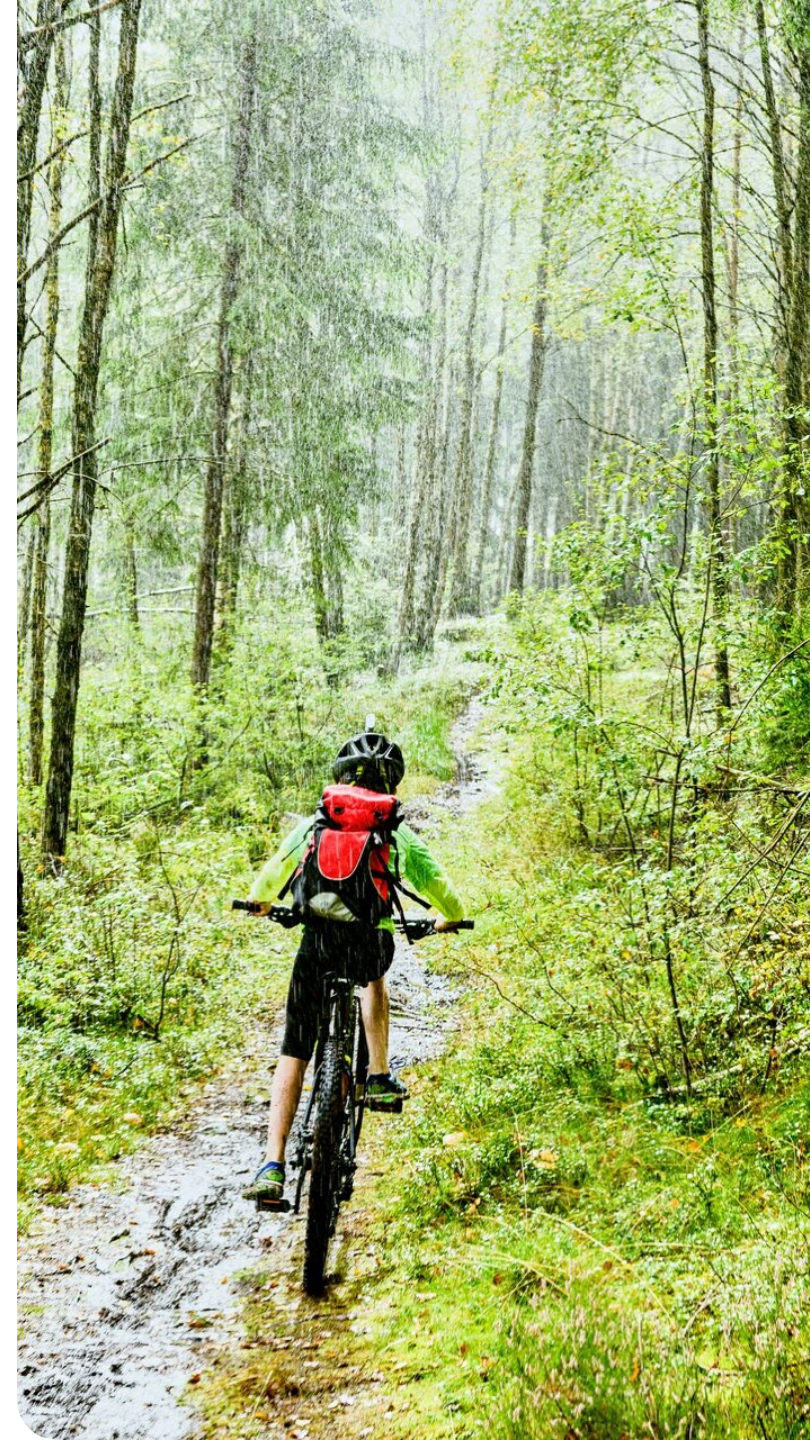


Target: A state of independence and quality.

- Develop the features of your application parallel and independently from your providers instead of sequentially
- Get started with our Service Mesh and get a feel of your app in production, knowing which standards to build your app against
- Automatically testing of your application in depth and breadth with all available evidence

How to get there?

1. A production-like environment that is easy to execute everywhere
2. A smart testing tool integrated with CI/CD & API Governance
3. A way to define requests and responses in a repeatable, yet powerful enough language



PACT
FOUNDATION



JUnit

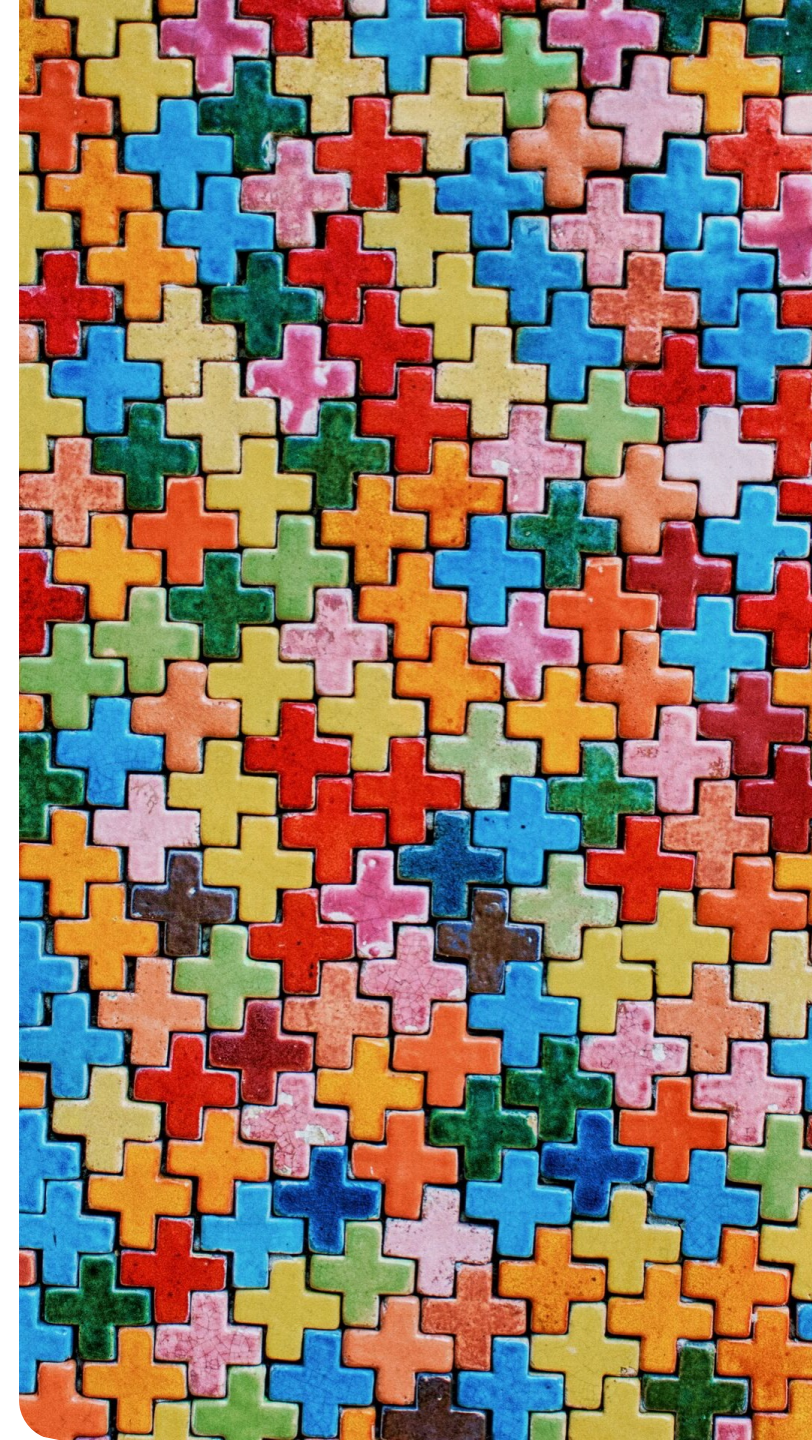


Design: Prefer industry standards.

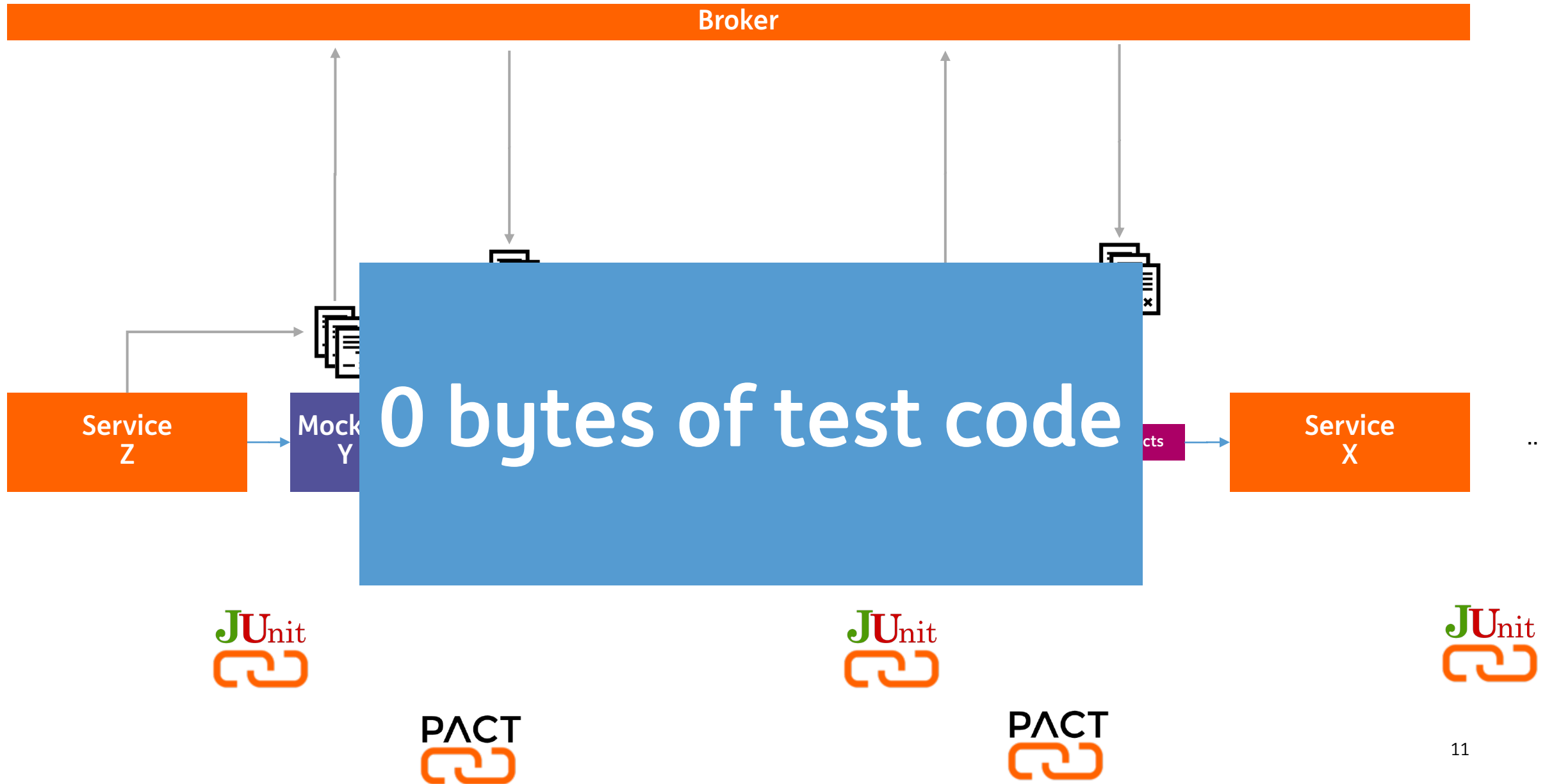
- Consumer Driven Contract testing with PACT
- Containerisation:
 - Docker images
 - docker compose
- Re-use existing tools like:
 - Performance: Gatling
 - Security: Burp suite
 - TestReport: Surefire xml
 - Pact Verification: Pact-verifier-cli
 - Network emulation: Linux `tc`
 - Observability: OpenTelemetry & Grafana

What: The Conformance TestKit

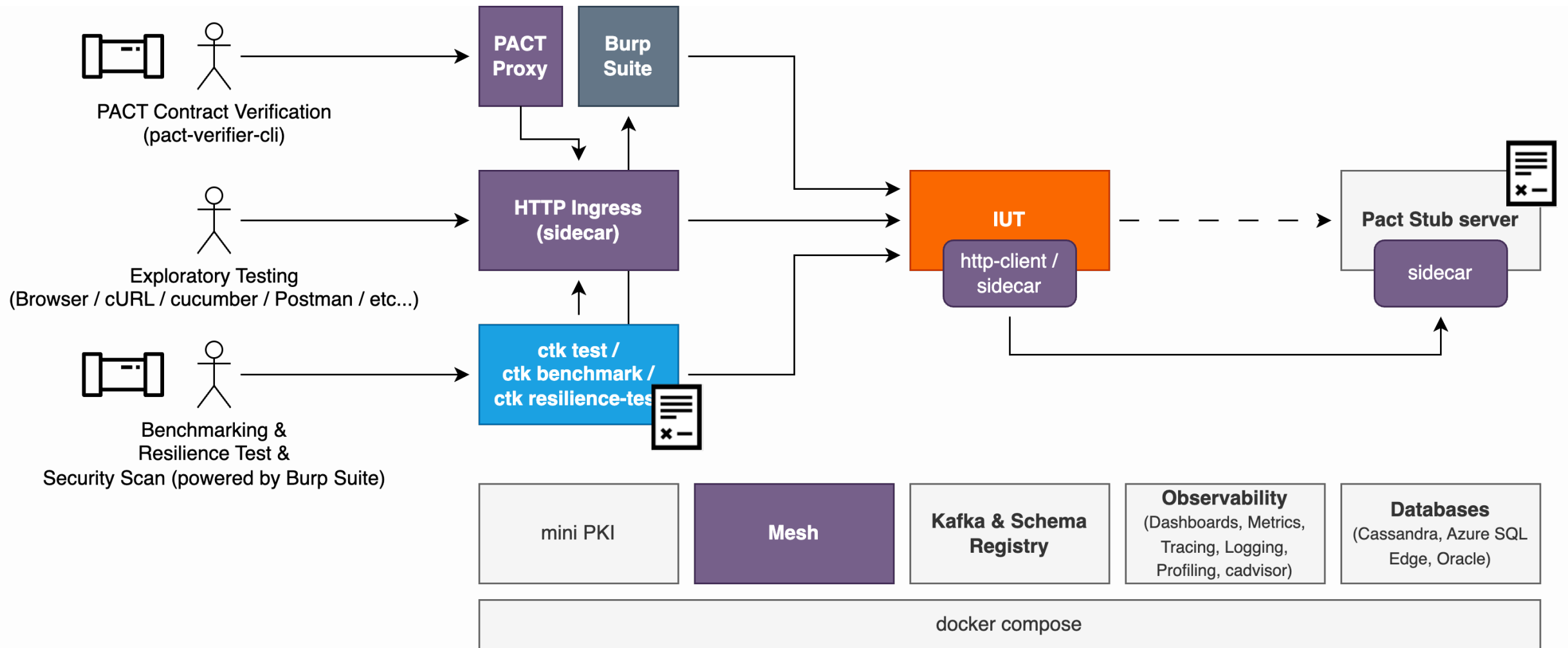
- CLI: Mini environment wiring everything together using Docker Compose
- Docker Compose Definitions with pluggable services
- Smart test tool, verification of:
 - Conformance
 - Performance
 - Security
 - Resilience
- Pact verification mechanism, connected with a broker for proper visibility of the integration between services



Independently develop and verify the chain



What: The Conformance TestKit





Demo

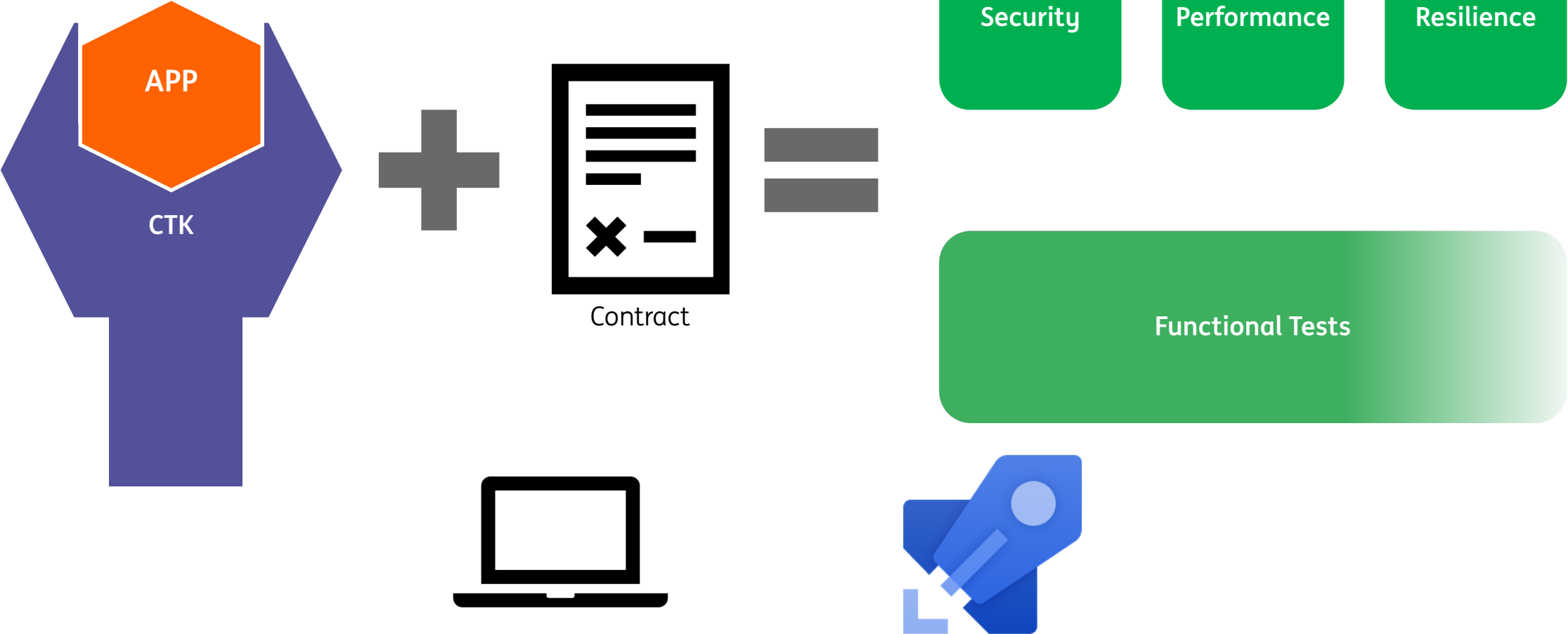
CTK + PACT validation



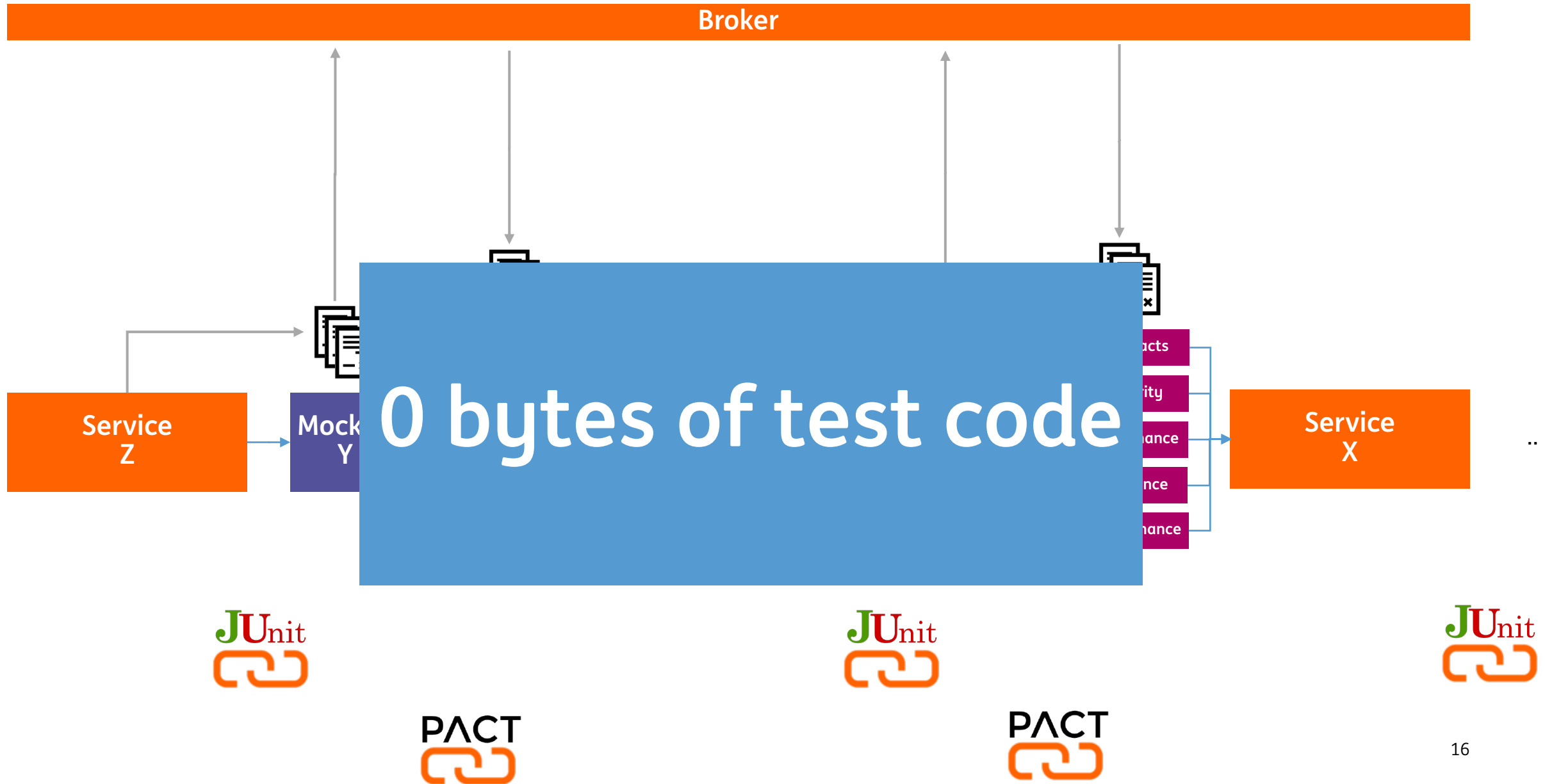
Now squeeze those oranges packs

- **Conformance:** Generate valid requests to test conformance
- **Performance:** Re-use request and response definition automatically generating benchmark validating SLA (latency @ throughput)
- **Security:** Translate into Burp Suite compatible flavour
- **Resilience:** Resilience test by screwing with downstreams during benchmark

Don't worry, Be Confident



Full coverage across the chain



A night sky with a rainbow and a bright light source. The rainbow is a vibrant arc of colors spanning across the upper half of the frame. Below it, a bright, yellowish-white light source, possibly the sun or moon, is partially obscured by clouds. The sky is dark blue with scattered white clouds. In the bottom right corner, a small, dark silhouette of a gazebo or pavilion is visible. The overall scene is a dramatic and colorful night sky.

Demo

Benchmark + Observability + Profiling

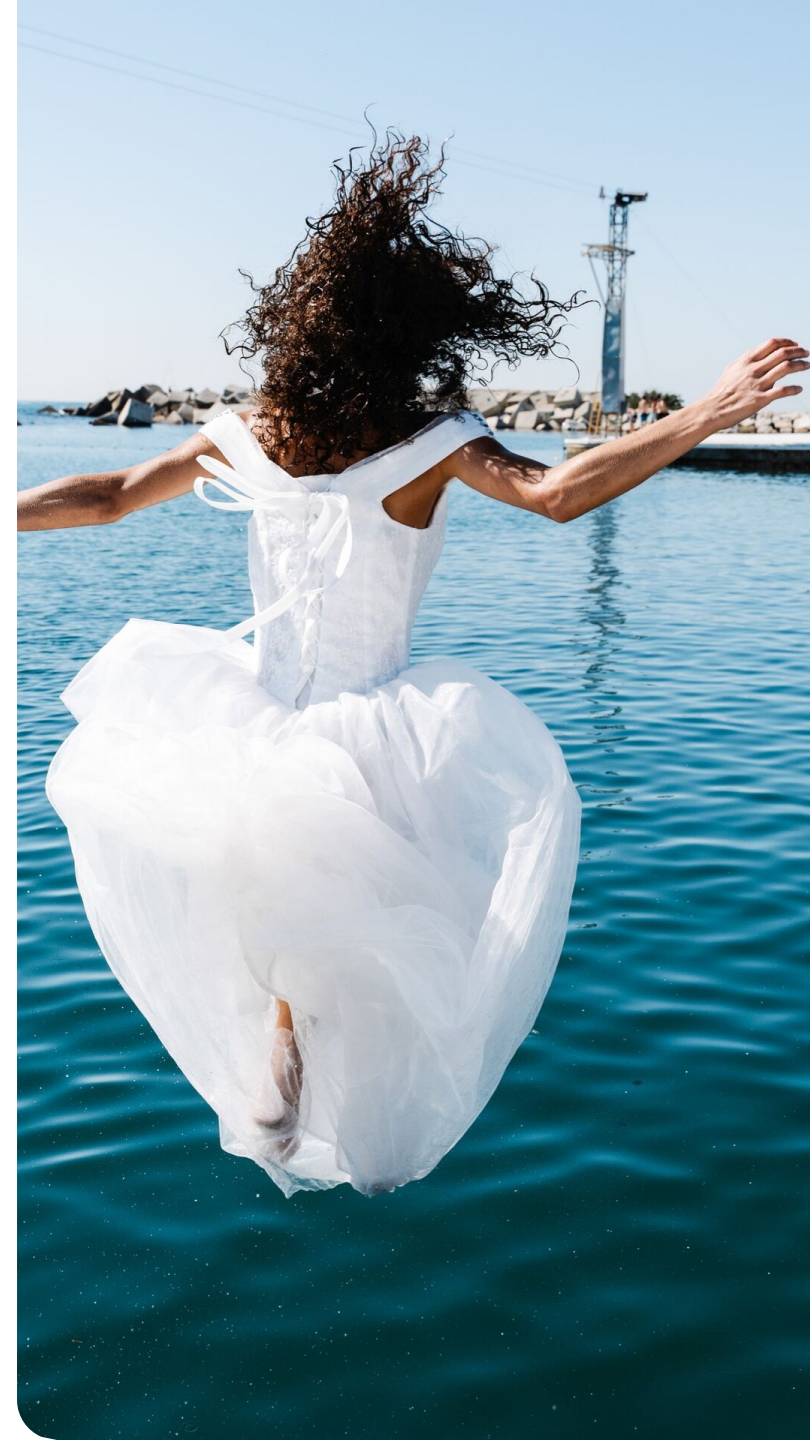
Additional features

- OotB Profiling with Pyroscope
- L4 and L7 Fault & Latency simulation
- Baseline comparisons
- Repeatable Resilience testing
- Native Image GraalVM assistance
- AutoTune
- ...



Conclusion

- High quality with little effort
- Consumer Driven Contract Testing is beneficial to both the consumer and to the Provider!
 - Development can happen in parallel
- Peace of mind is achievable
 - Git commit, push, go home



A serene winter night scene. The foreground is a snowy slope dotted with numerous evergreen trees heavily laden with snow. In the background, a small wooden cabin is nestled among the trees. The sky is a deep, dark blue, filled with a multitude of stars and a prominent, bright, orange-yellow full moon. The overall atmosphere is peaceful and magical.

Thank you

Ask us anything



do your thing