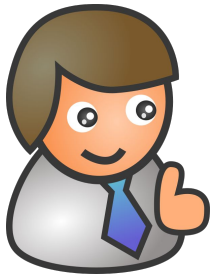
Introduction to CSPA Modules

Romain Tailhurat (INSEE)
Trygve Falch (SSB)

"Timeline-overview"



ArchPat Group Scope

Explore

[Placeholder: SSB VTL Example]

Design

Build

[Placeholder: ONS Collection Example - hosting in cloud env]

Assemble/deploy

[Placeholder: possible SCB demo of process designer]

(Production)

Configure

Use

Archetypes

Archetype A - Bundled, P2P

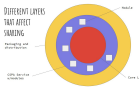
Archetype B - Bundled, Orchestration

Friction points

- "My data layer is Postgre not MySQL"
- "I'm using an event broker, I can't use your REST endpoint"
- "I don't have an ESB!"

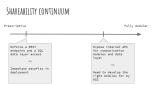
→ Modules to the rescue!

Cost of shareability



Different layers that affect sharing

Shareability continuum



[Placeholder: Containerization slides]

Demo

[Based on Developed and Deployed Services - Configure for CPI and GDP]

Demo

CSPA challenges today

- Re-usability of services is difficult in its current state
- Different NSI’s have different investments in different technology platforms
- CSPA lacks good guidance for how services should be implemented and made shareable.

Friction points

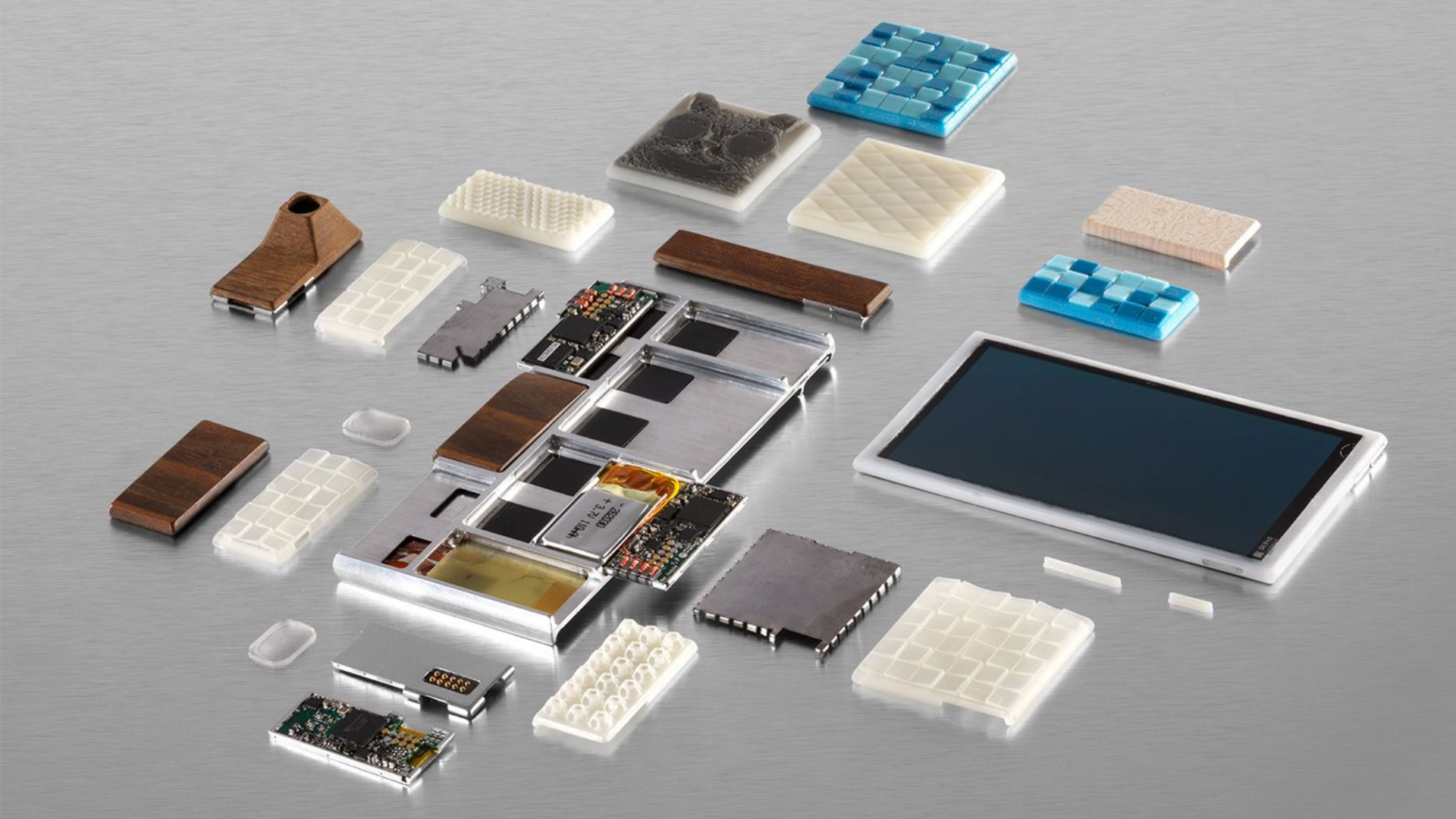
- “My data layer is Postgre not MySQL”
- “I’m using an event broker, i can’t use your REST endpoint”
- “I don’t have an ESB!”

- Many other friction points related to information architecture, etc.

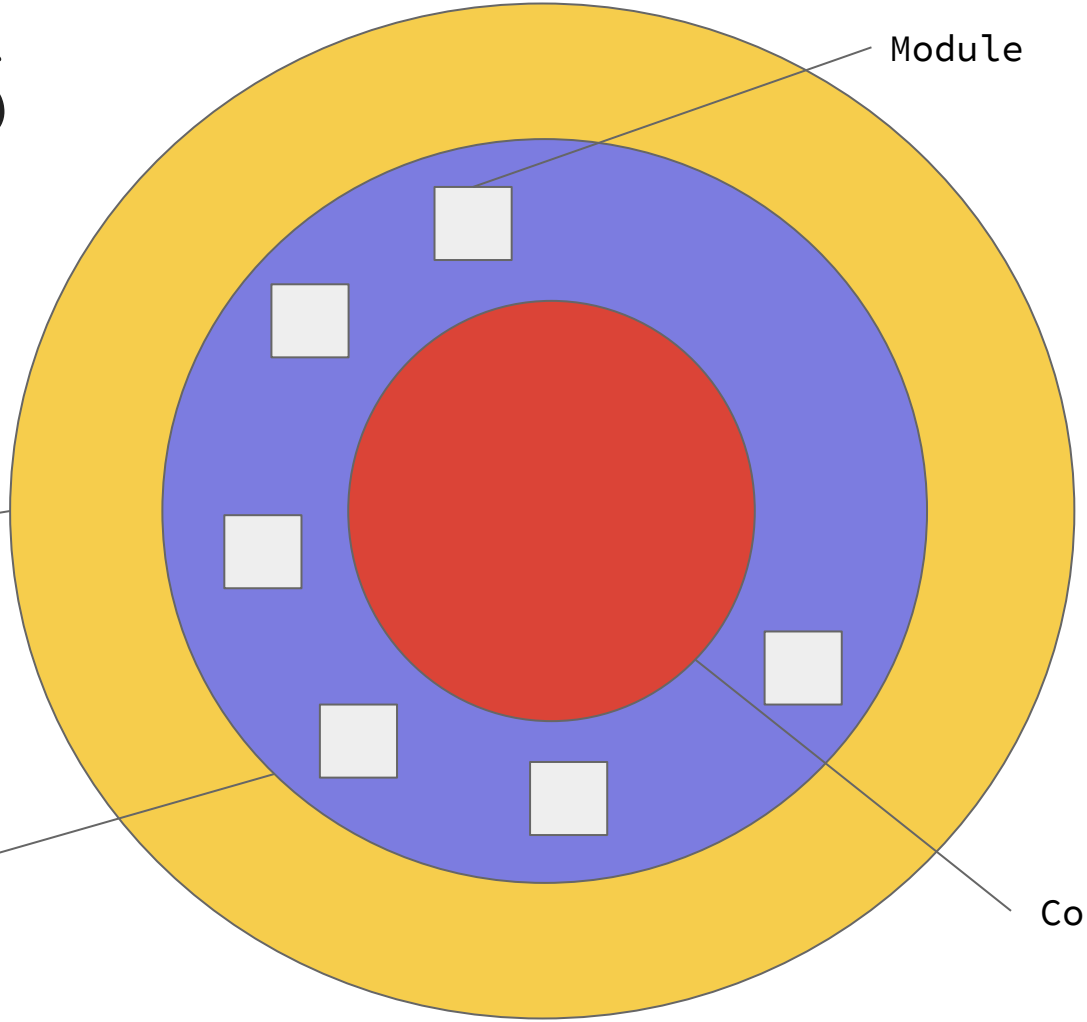
⇒ CSPA Modules to the rescue!

CSPA Modules

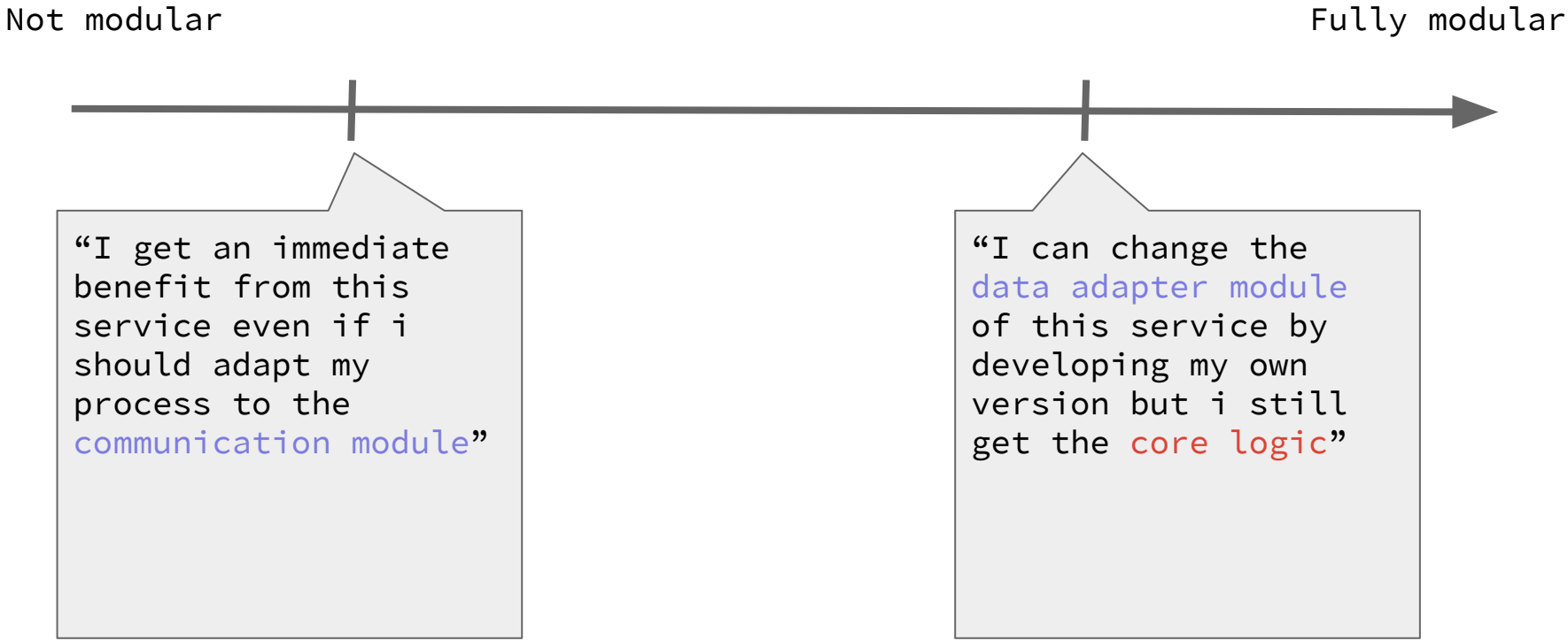
- A CSPA Module is a small piece of functionality that communicates with the core business-logic of the service, and with the outside world
- Modules are optional for a CSPA service, but it will make sharing easier if you decide to use it
- A module will be the technical “glue” for binding the CSPA Service to a local NSI’s environment while still retaining the core business logic of the Service



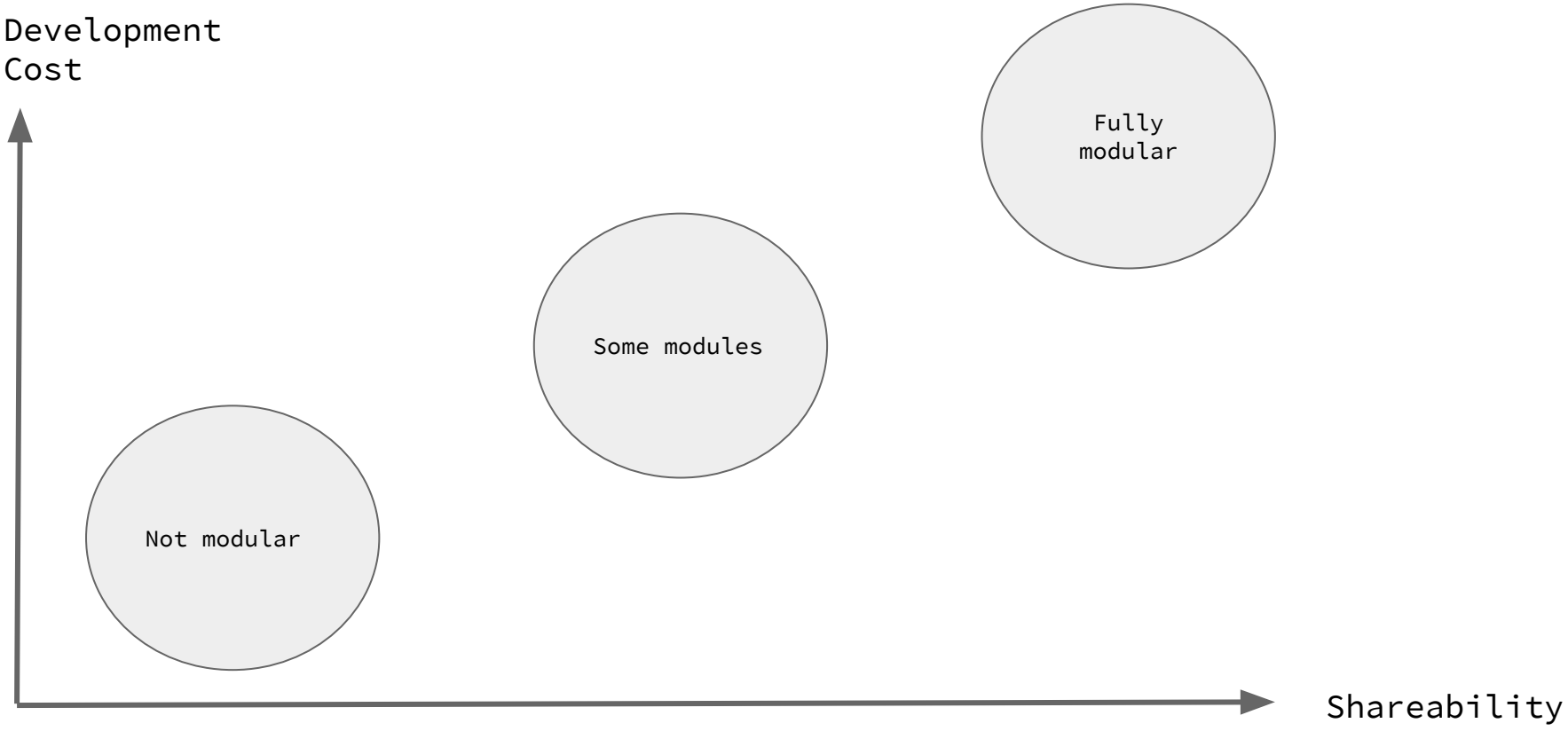
Different layers that affect sharing

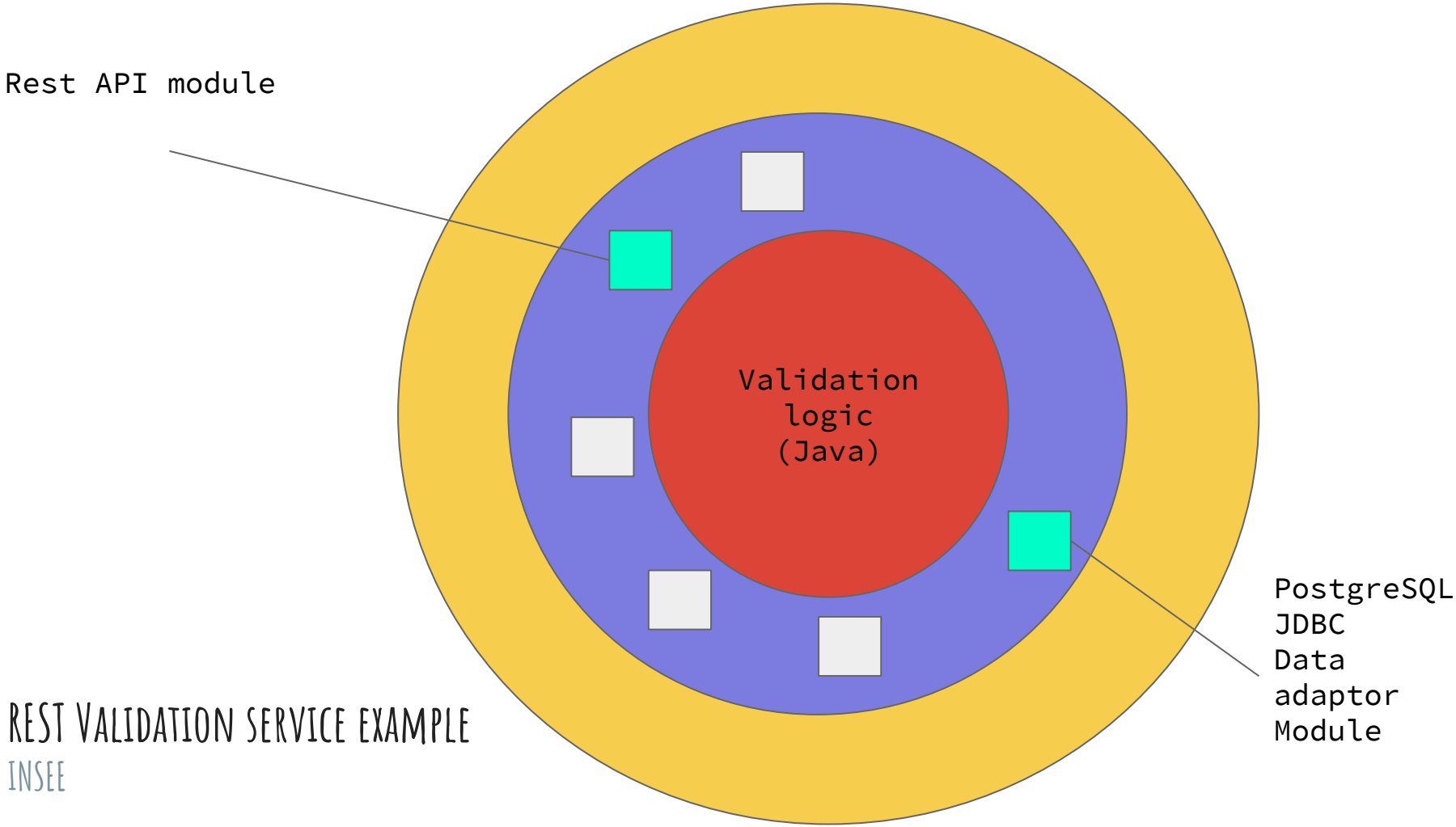


Shareability scale



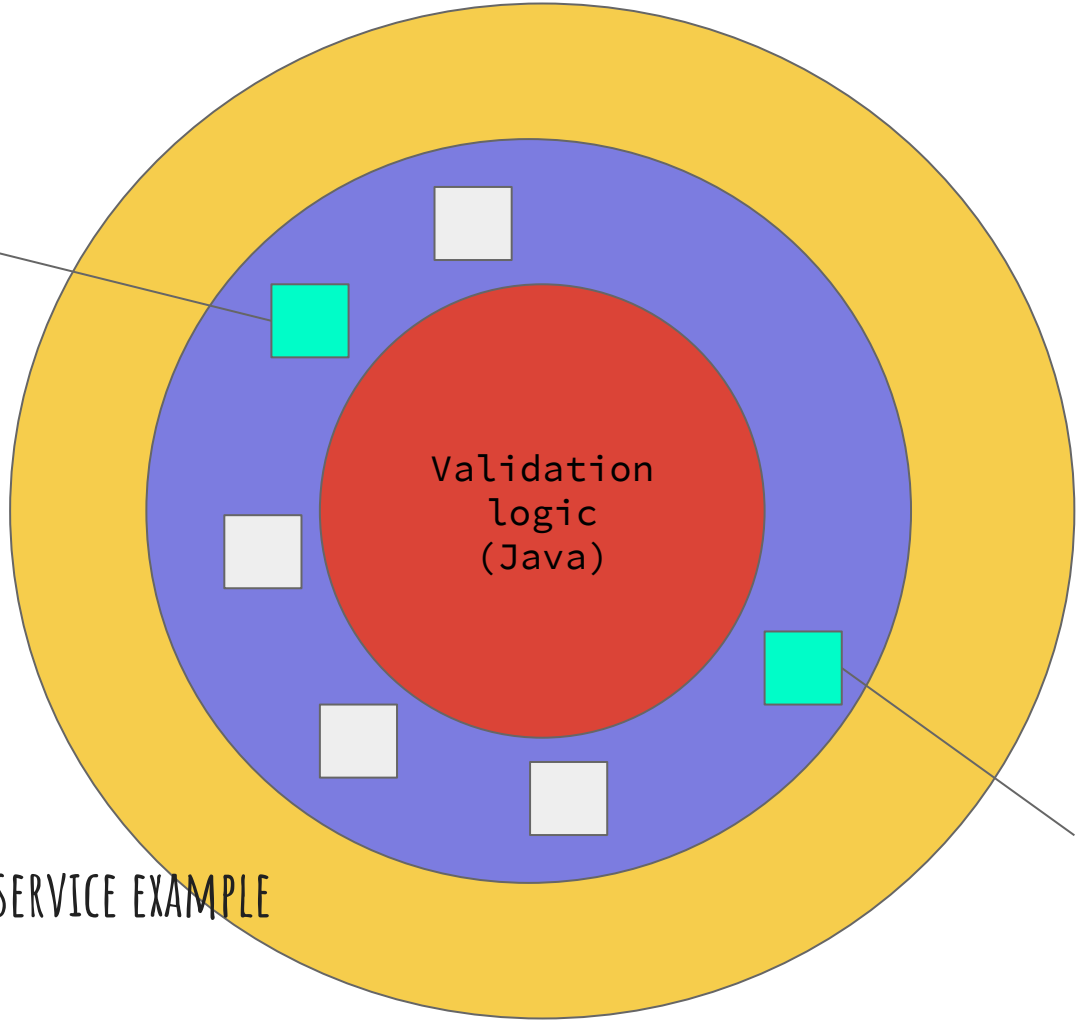
Development cost of shareability





REST Validation service example

INSEE



Event driven Validation service example

SSB

How to make modules happen ?

Software engineering principles

+

Open-source Collaboration