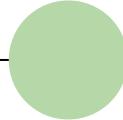


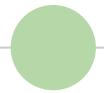
Untangling The API Spaghetti With BPMN Diagrams



Mourjo Sen

October 9, 2025

<https://mourjo.me>



Welkom! ನಮಸ್ಕಾರ !



Welkom! ನಮಸ್ಕಾರ !

- Born and brought up in Kolkata
- Living in Amsterdam since 2023



Welkom! ನಮಸ್ಕಾರ !

- Born and brought up in Kolkata
- Living in Amsterdam since 2023
- 10 years as a software engineer
- Currently @ booking.com

Welkom! ନମ୍ବାର !

- Born and brought up in Kolkata
- Living in Amsterdam since 2023
- 10 years as a software engineer
- Currently @ booking.com
- Writing and speaking in the last 5 years

ARTICLE
Applying Flow Metrics to Design Resilient Microservices
Mourjo Sen
ARCHITECTS ARCHITECTURE & DESIGN

InfoQ

Unlocking observability: Structured logging in Spring ...

10 min read · Mar 26, 2024 · [View story](#)

13.6K Views

Load Shedding in Web Services

10 min read · Sep 4, 2020 · [View story](#)

13.3K Views

<https://mourjo.me>

www.youtube.com › watch

Property Based Testing to Preserve Invariants - Fachtalk mit ...

Mourjo Sen spricht in unserem LEGACY & INNOVATION Track beim techCamp über Property-based Testing (PBT). Eine Methode, um die...
YouTube · intersoft GmbH · 1 month ago

www.youtube.com › watch

Building Resilient Services in Clojure

Abstract Building resilient web services in Clojure. By employing strategies like circuit breakers, load shedding, graceful restarts among ...
YouTube · reclojure · 21 déc. 2020

Writing a fair task execution framework — Mourjo Sen

YouTube · IN/Clojure
4 févr. 2019



Monoliths → Microservices

Monoliths → Microservices



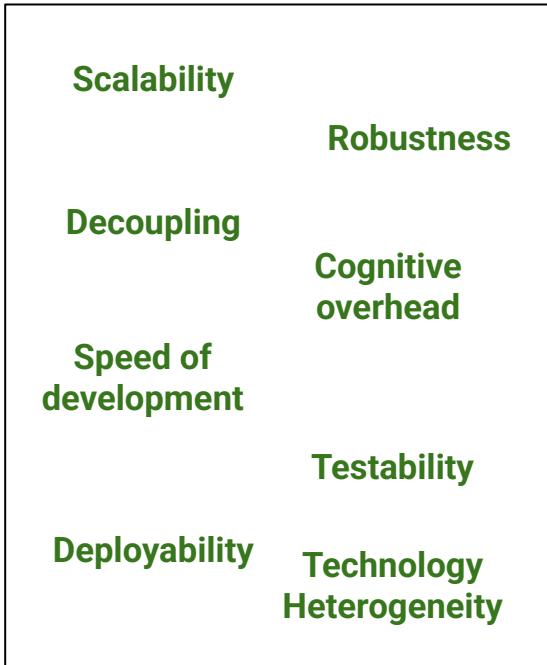
Monoliths → Microservices



Monoliths → Microservices



Monoliths → Microservices





Fragmentation





Fragmentation

- More pieces make it harder to see the big picture





Fragmentation

- More pieces make it harder to see the big picture
- The product has to work as a whole





Fragmentation

- More pieces make it harder to see the big picture
- The product has to work as a whole
 - Not adaptable to change
 - Not easily understandable

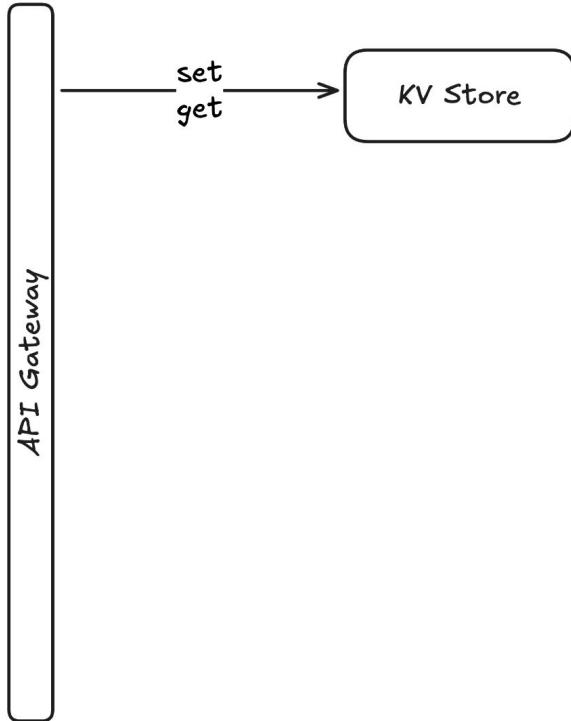


Adapting To Problem 1: Change

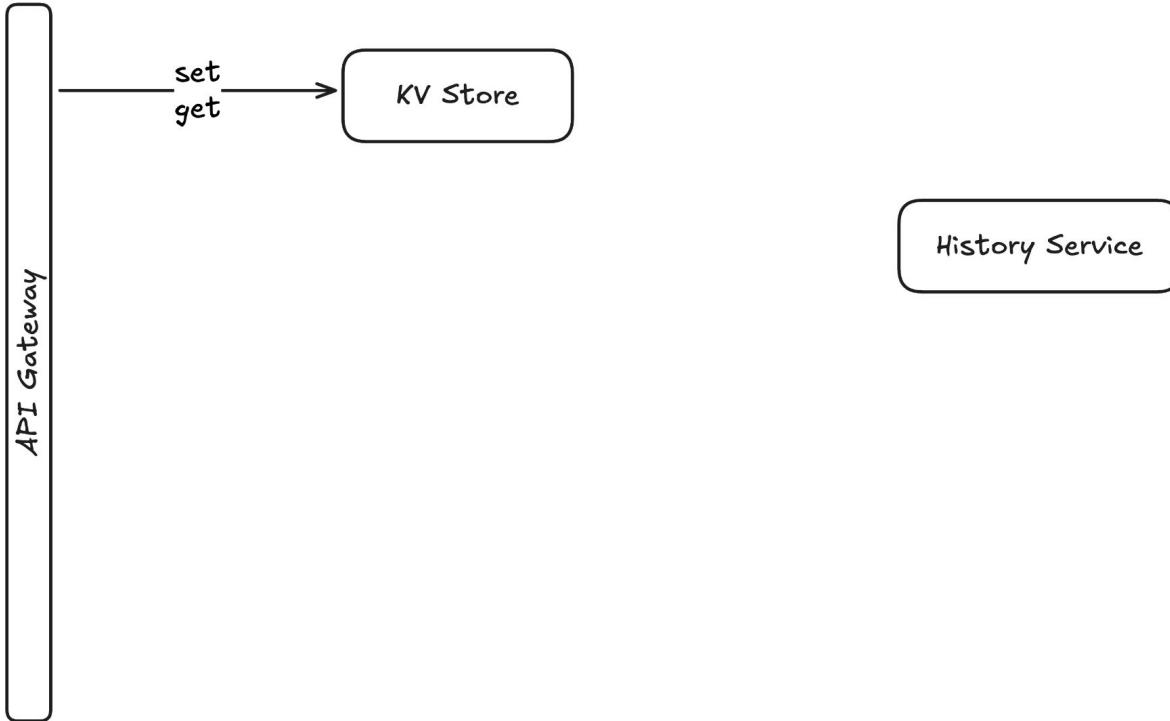


Fragments Compose In Fixed Ways

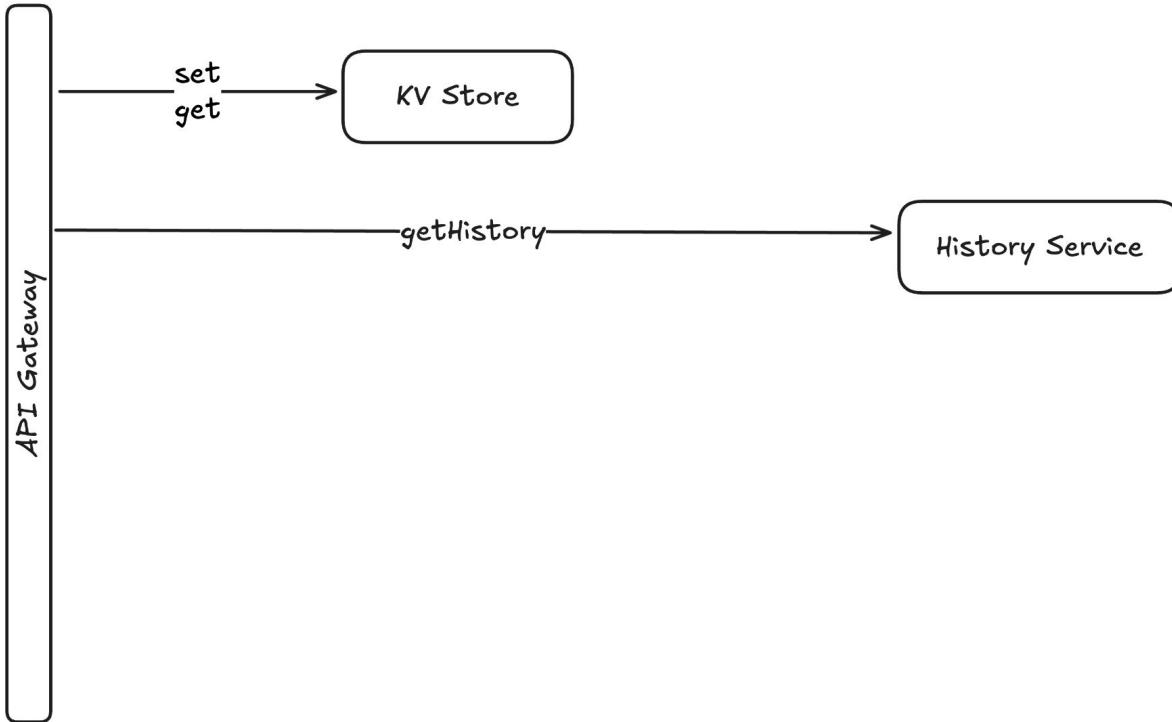
Fragments Compose In Fixed Ways



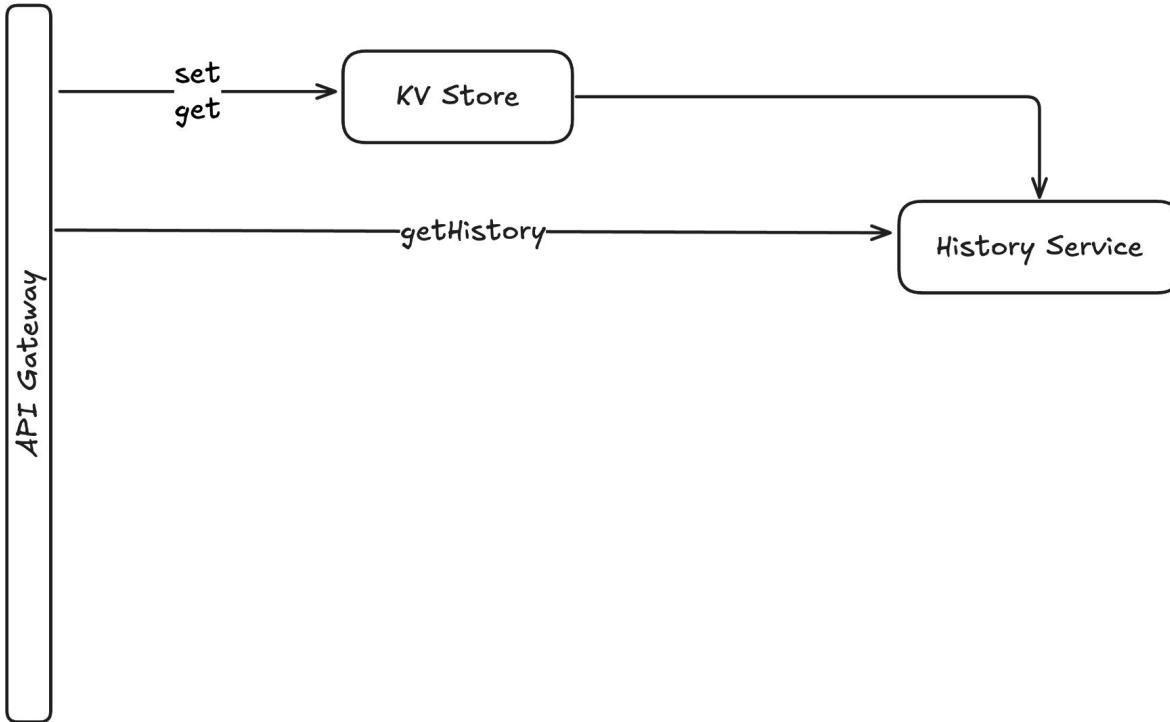
Fragments Compose In Fixed Ways



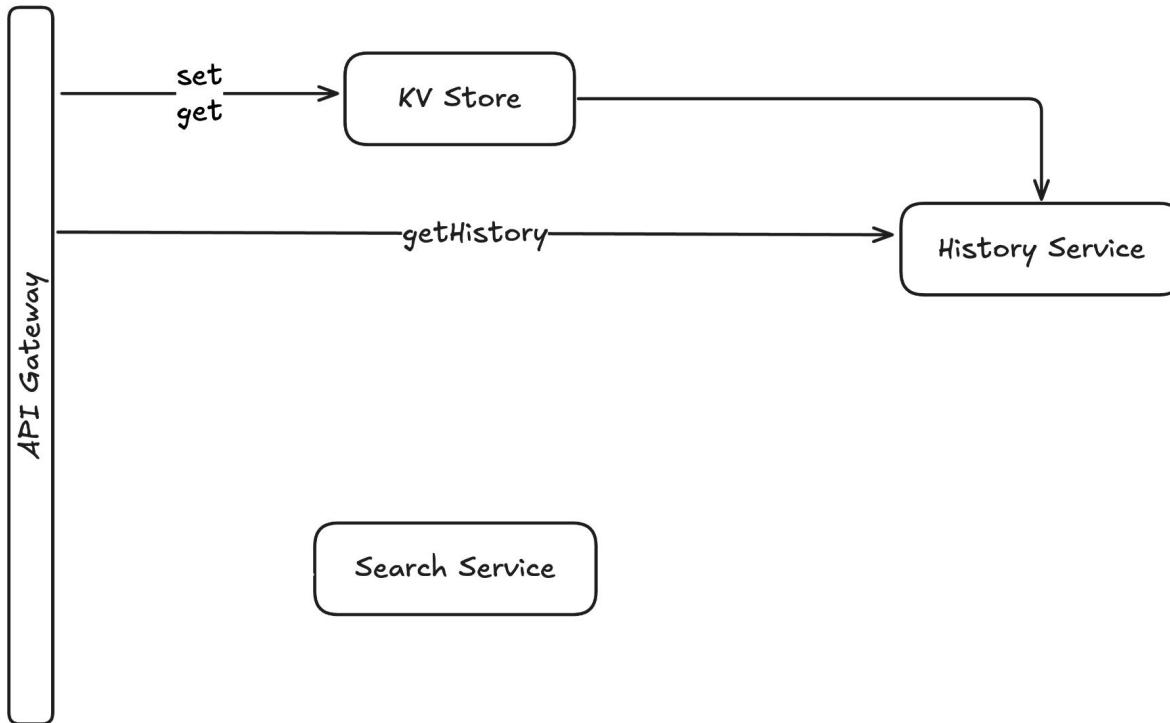
Fragments Compose In Fixed Ways



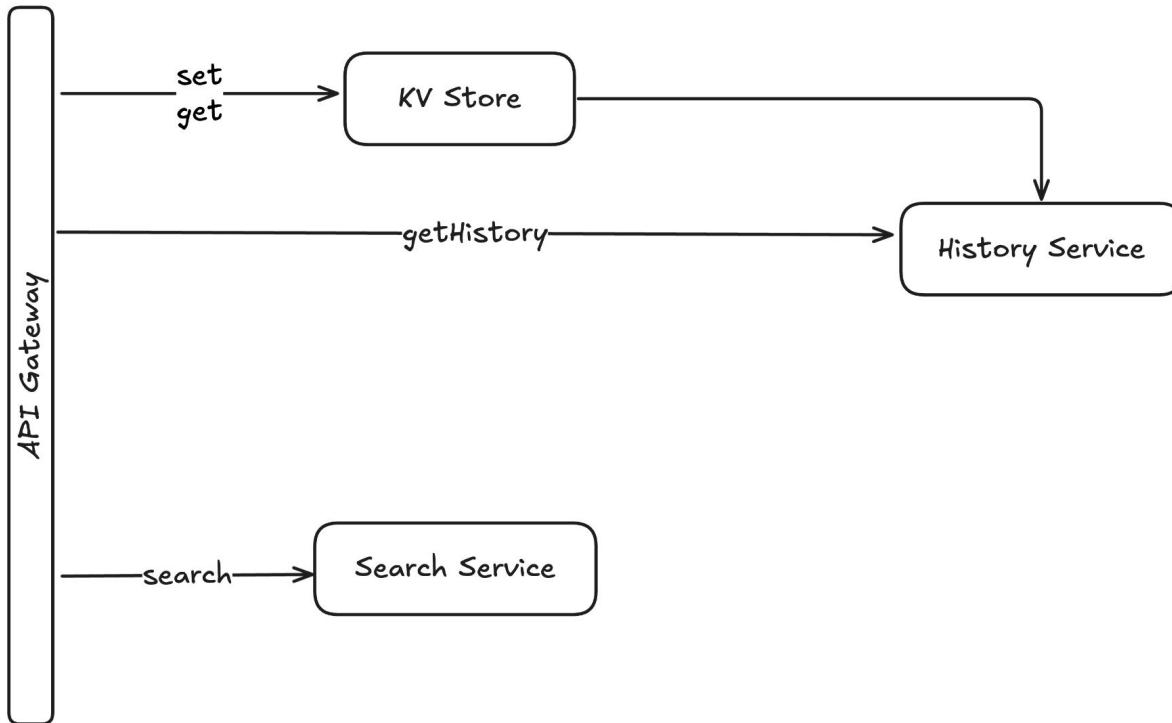
Fragments Compose In Fixed Ways



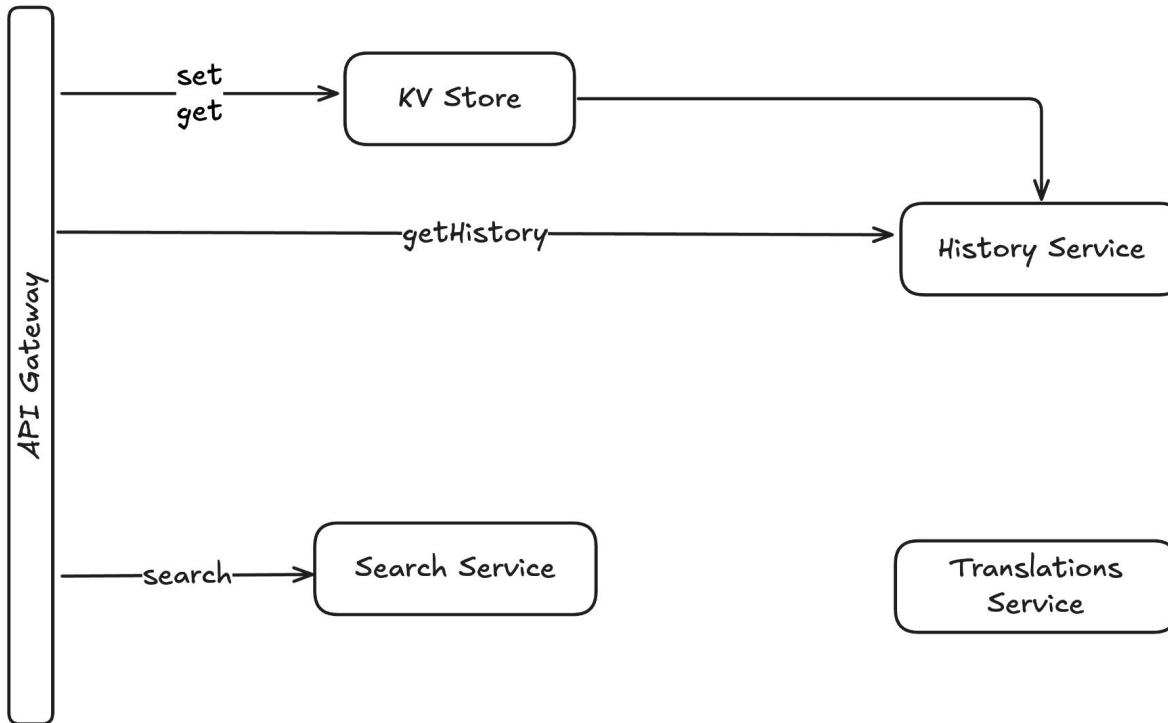
Fragments Compose In Fixed Ways



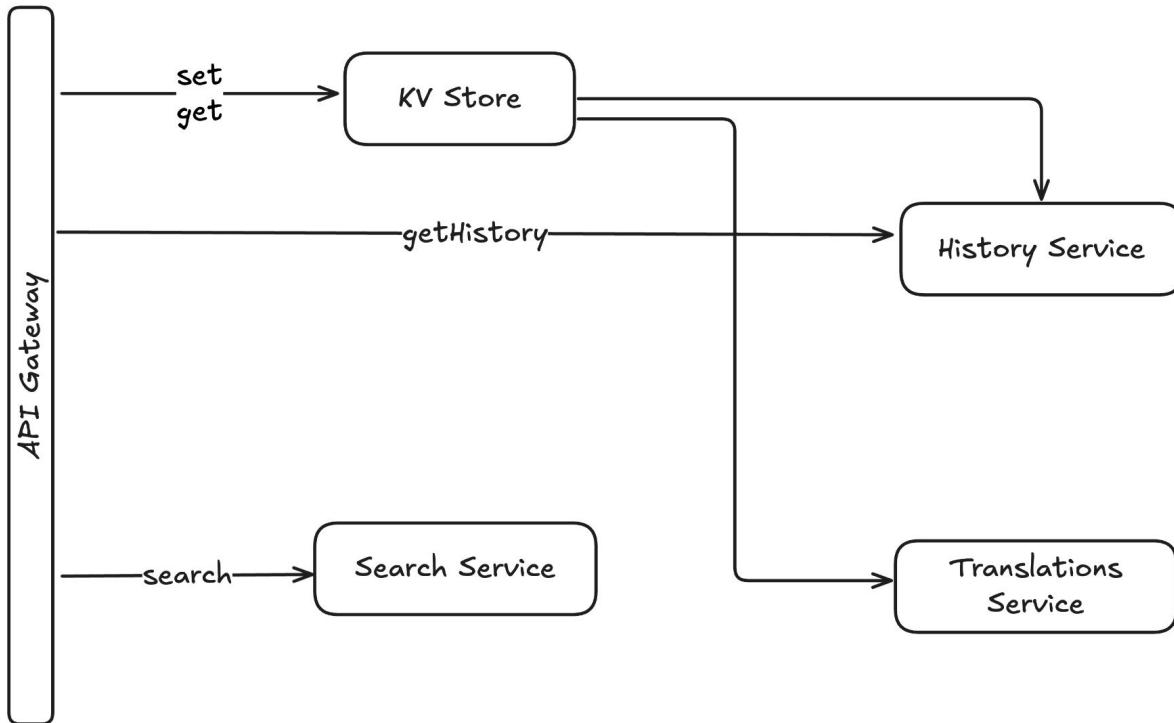
Fragments Compose In Fixed Ways



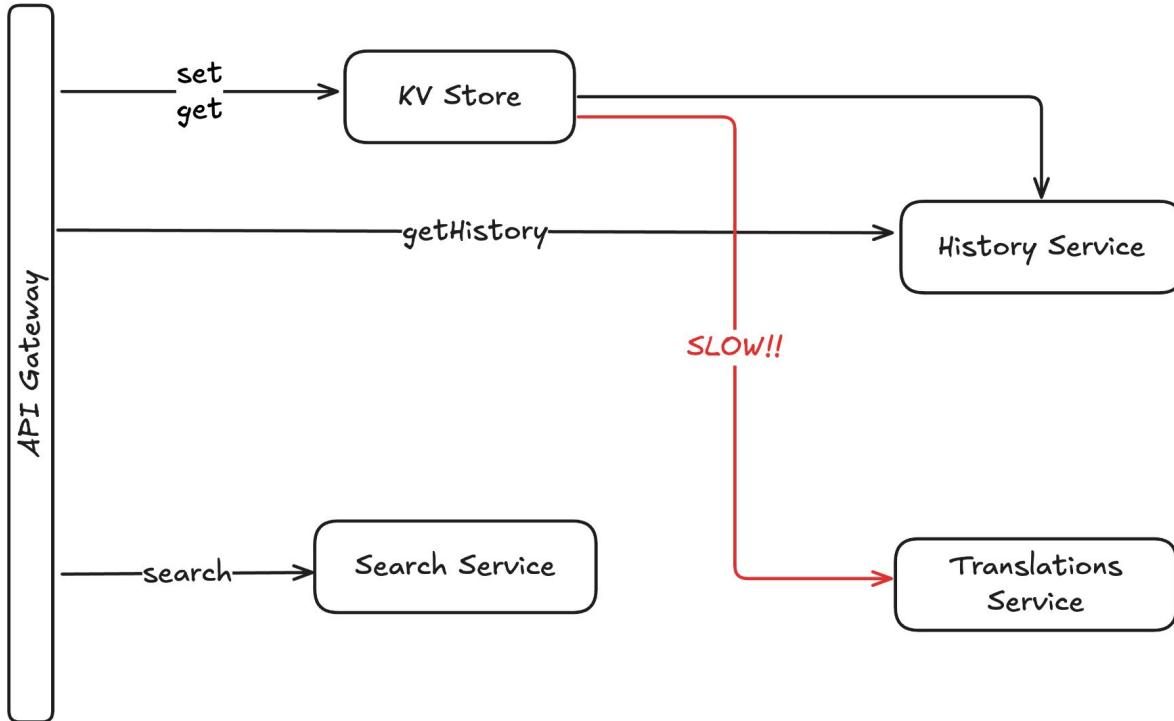
Fragments Compose In Fixed Ways



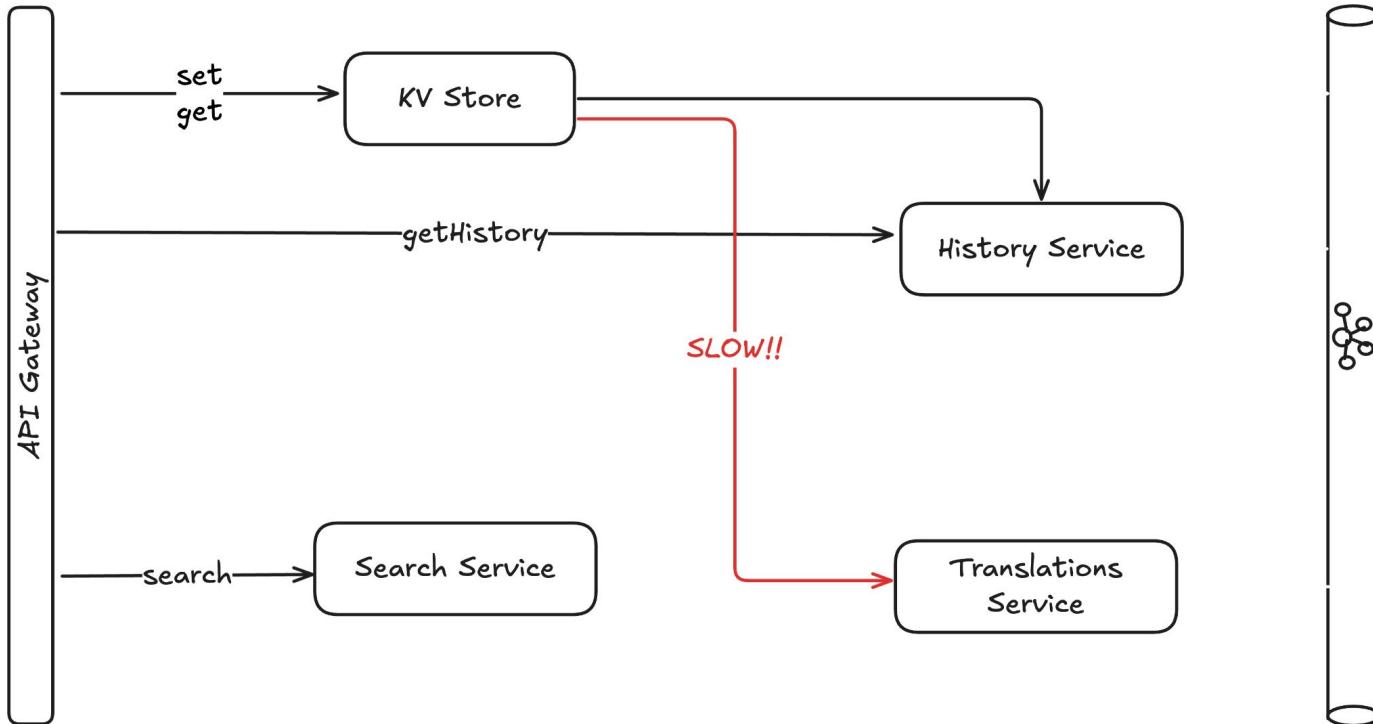
Fragments Compose In Fixed Ways



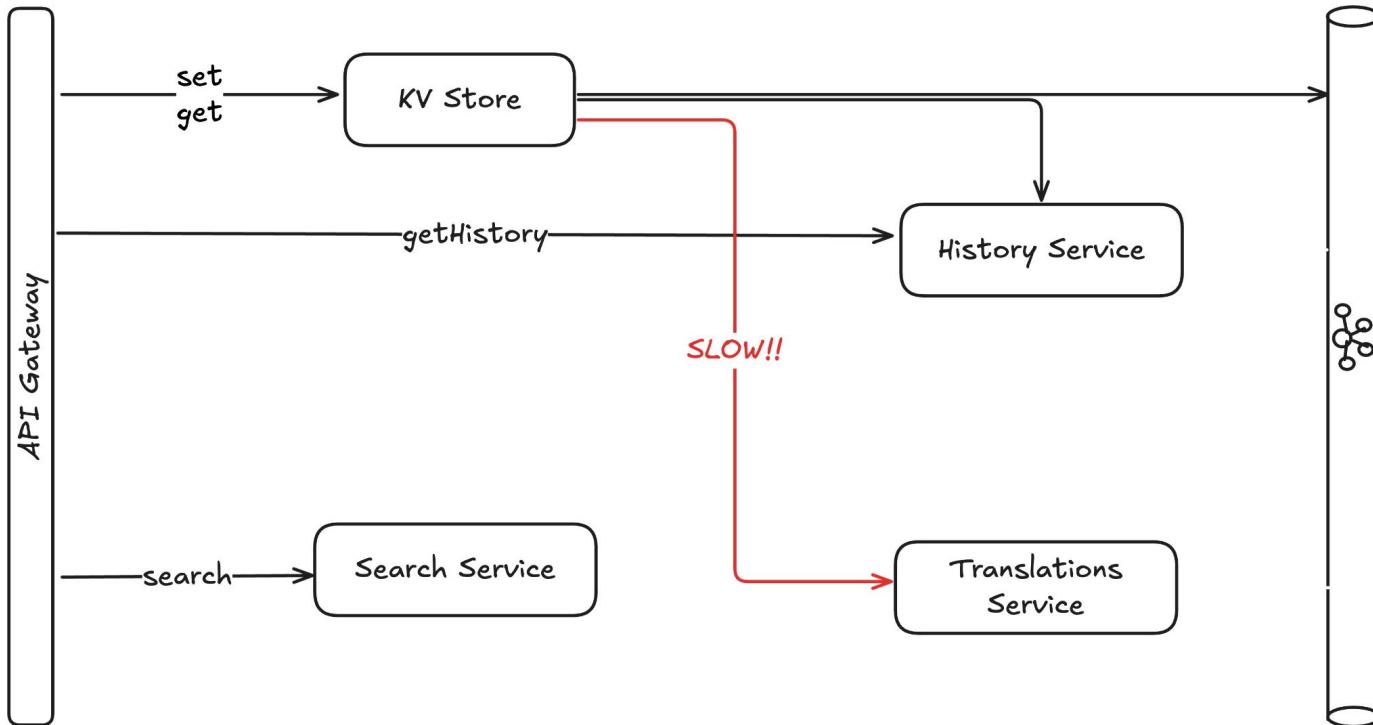
Fragments Compose In Fixed Ways



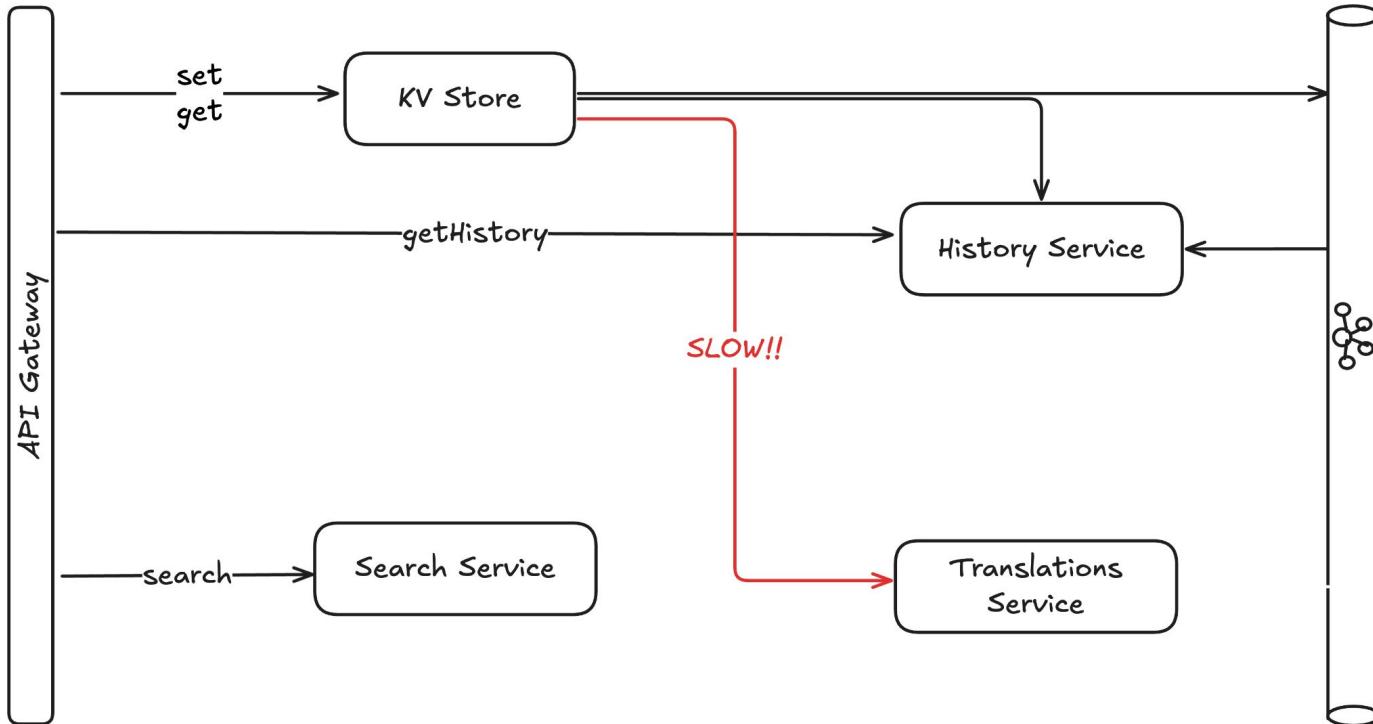
Fragments Compose In Fixed Ways



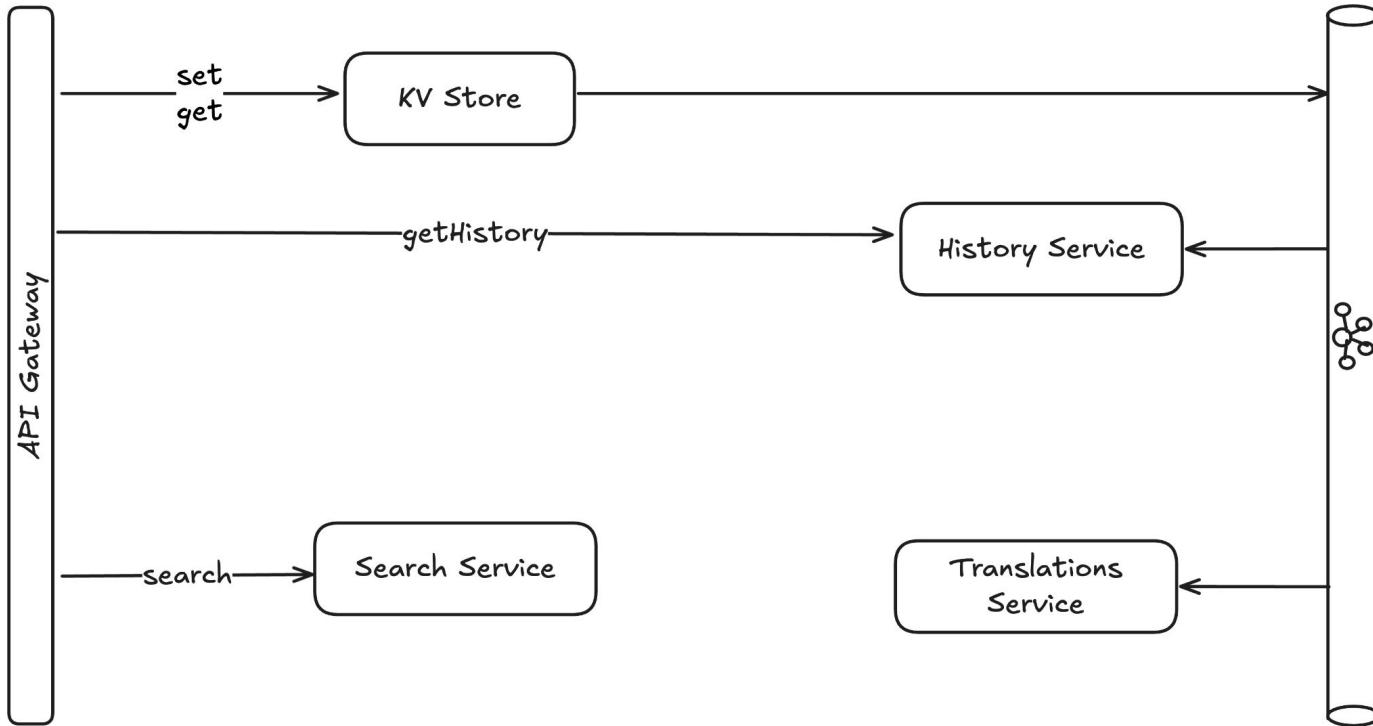
Fragments Compose In Fixed Ways



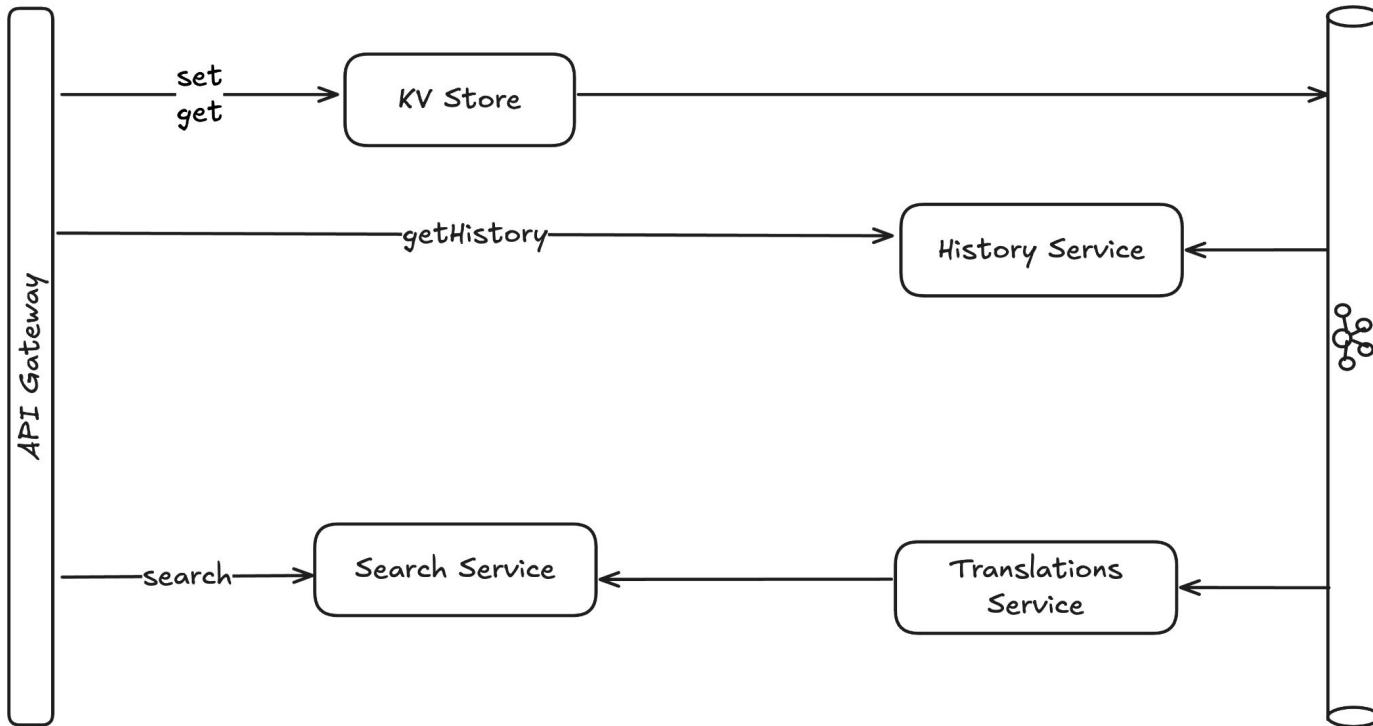
Fragments Compose In Fixed Ways



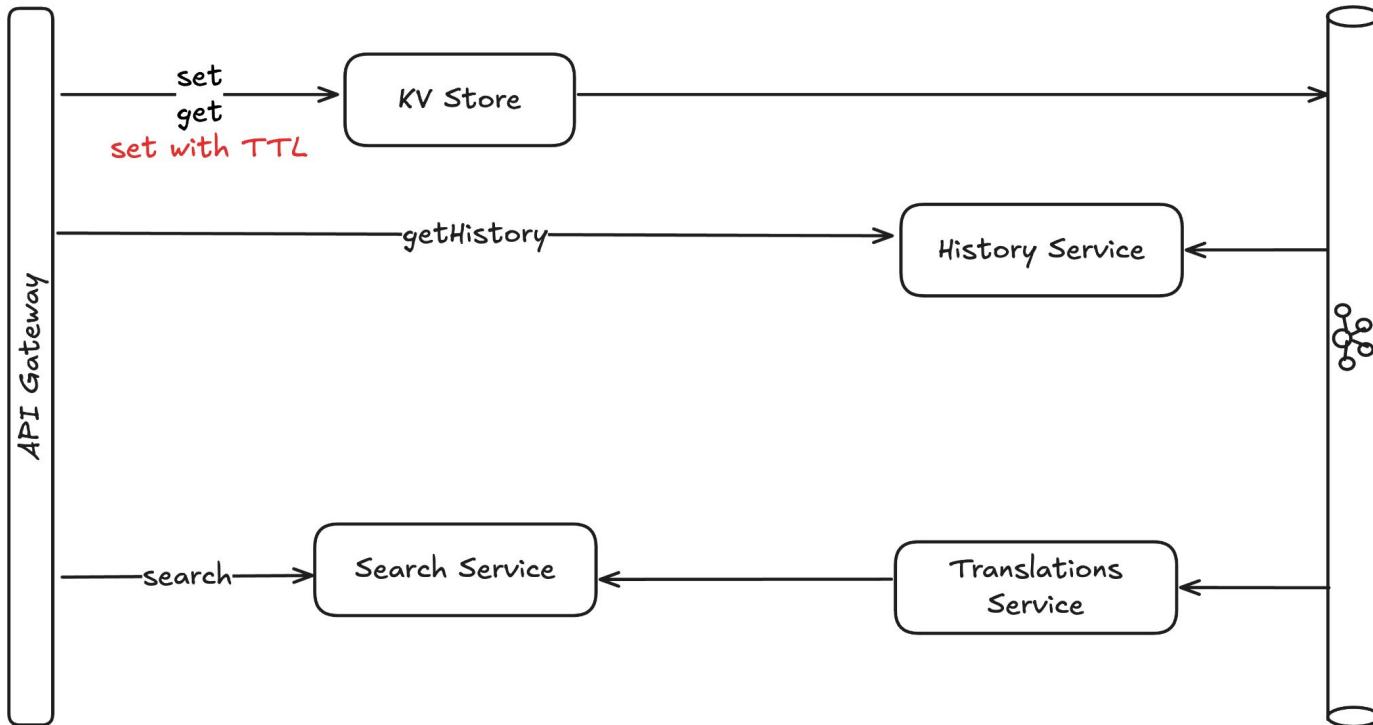
Fragments Compose In Fixed Ways



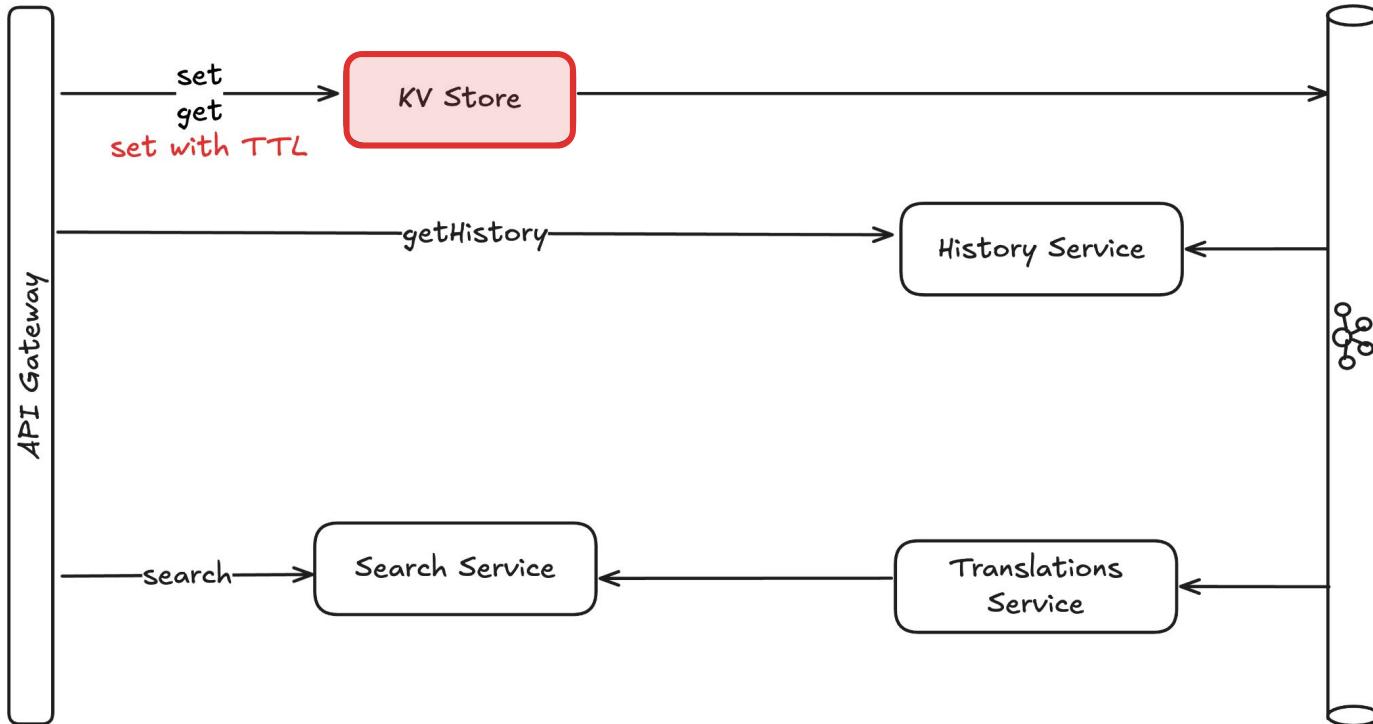
Fragments Compose In Fixed Ways



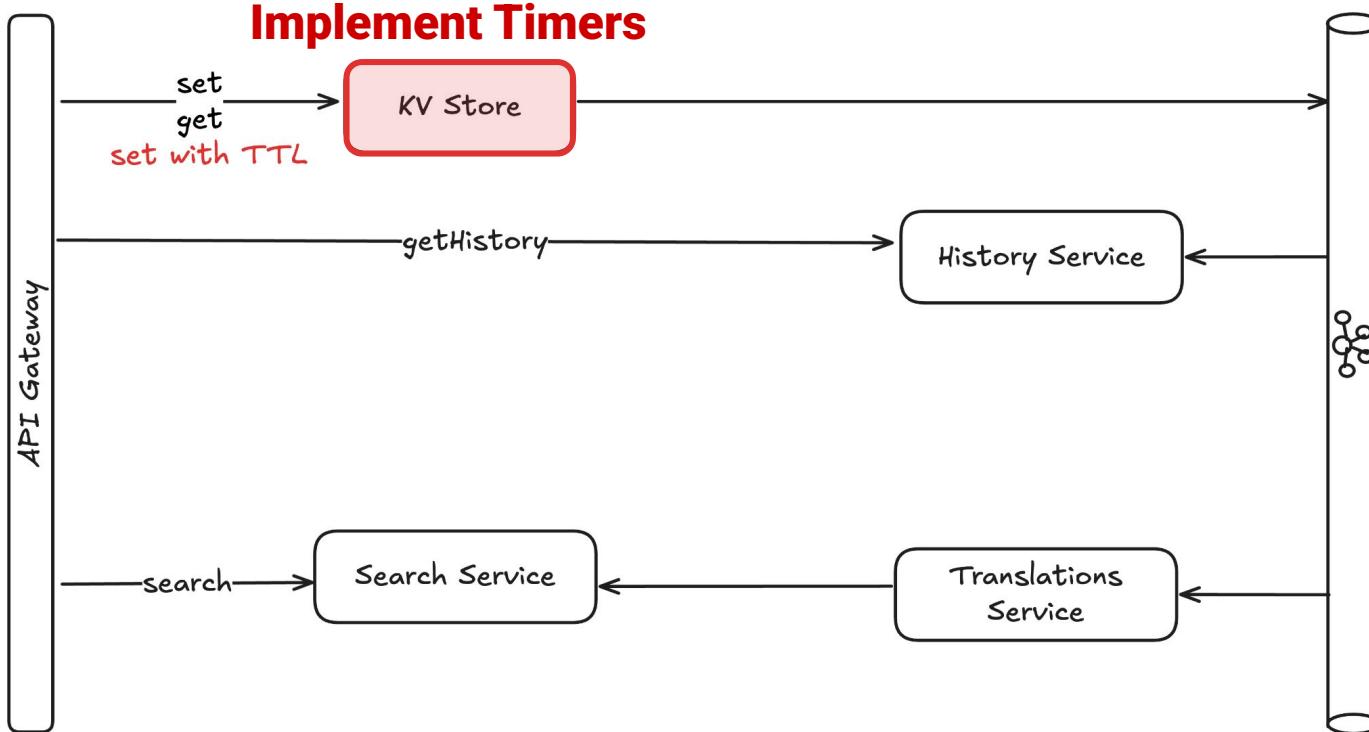
Fragments Compose In Fixed Ways



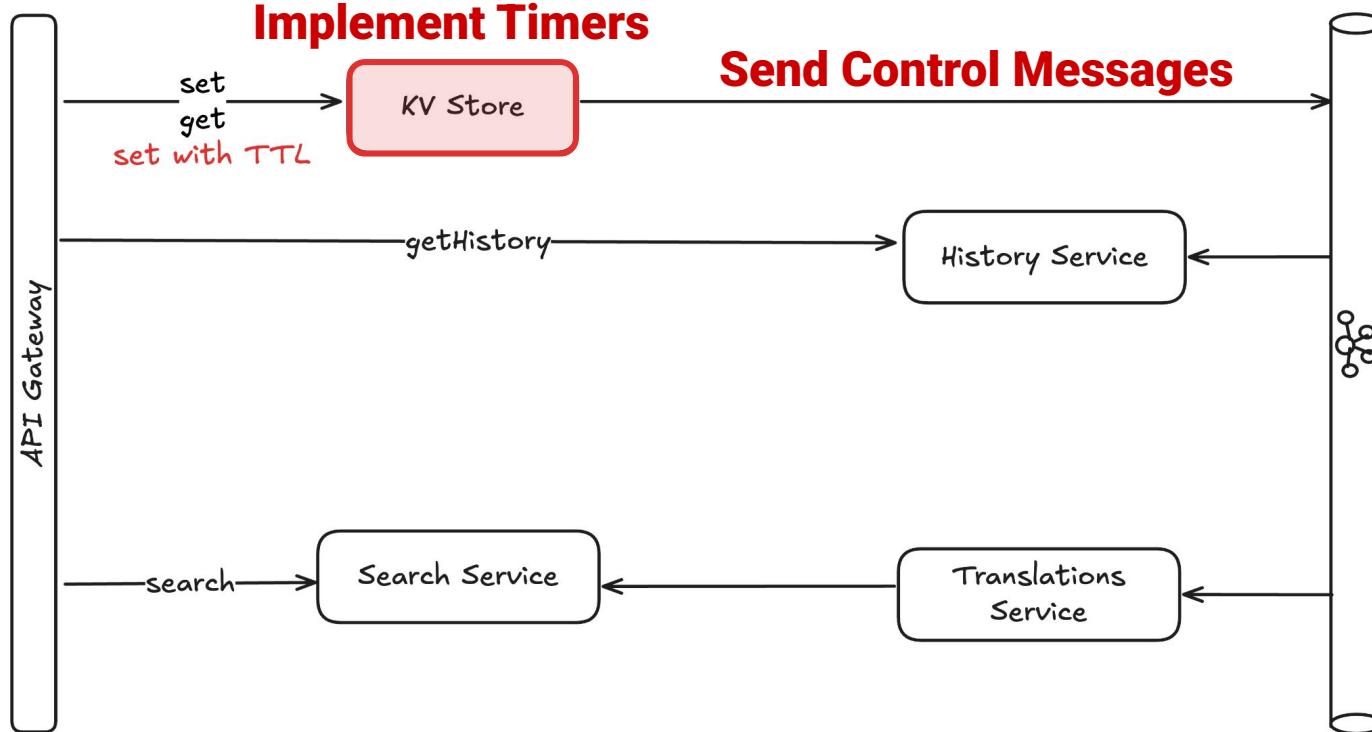
Fragments Compose In Fixed Ways



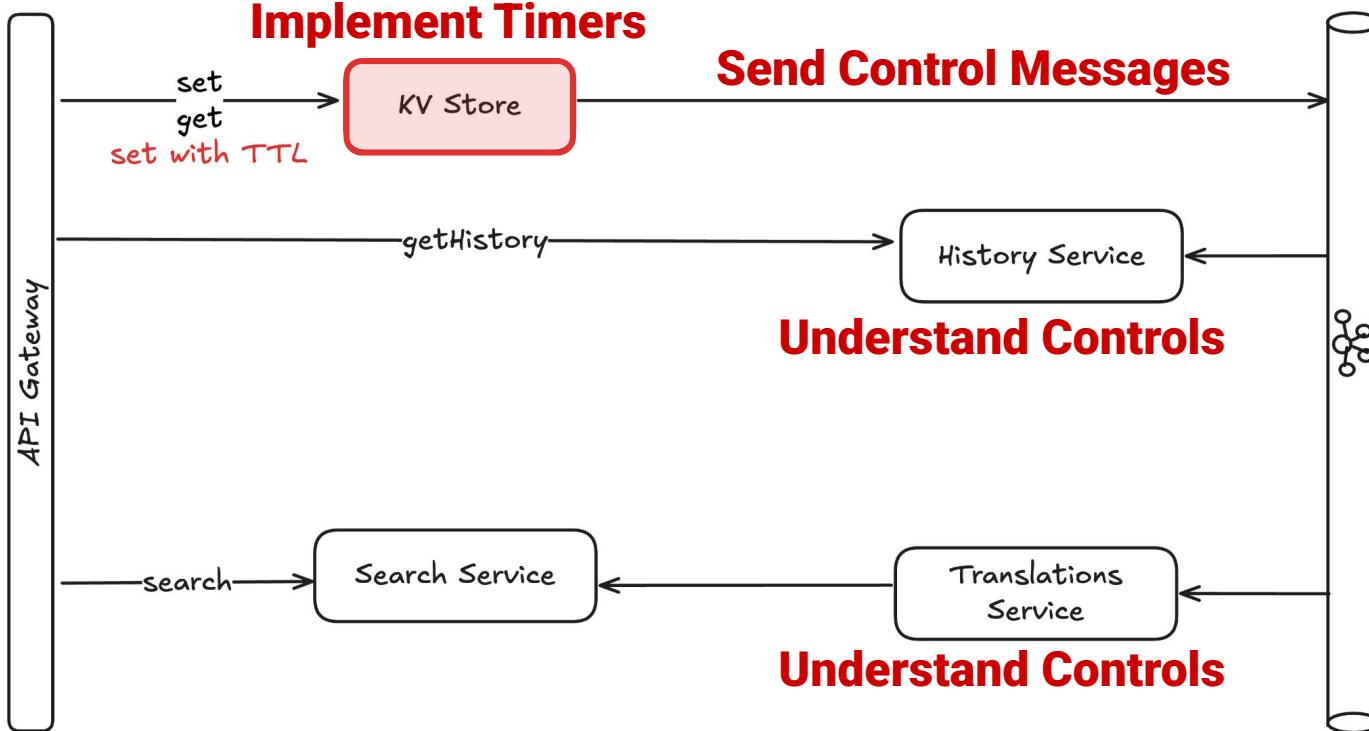
Fragments Compose In Fixed Ways



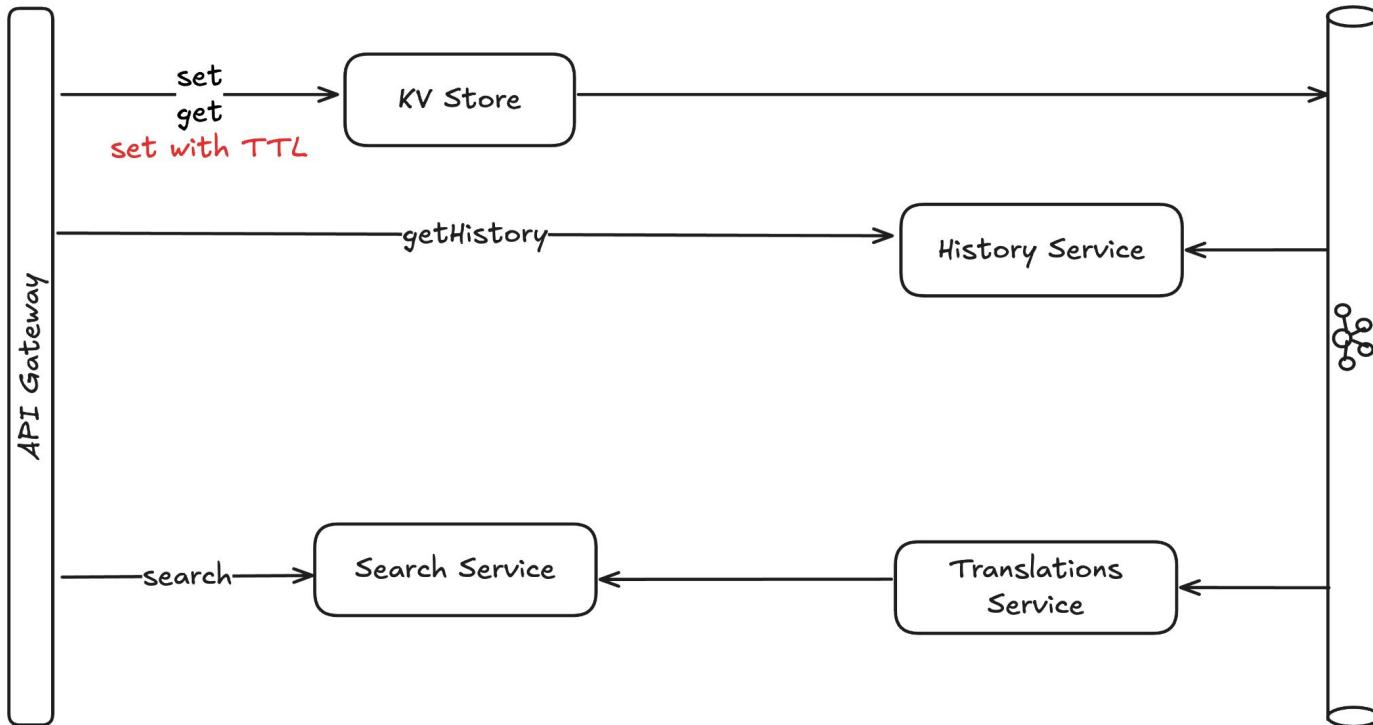
Fragments Compose In Fixed Ways



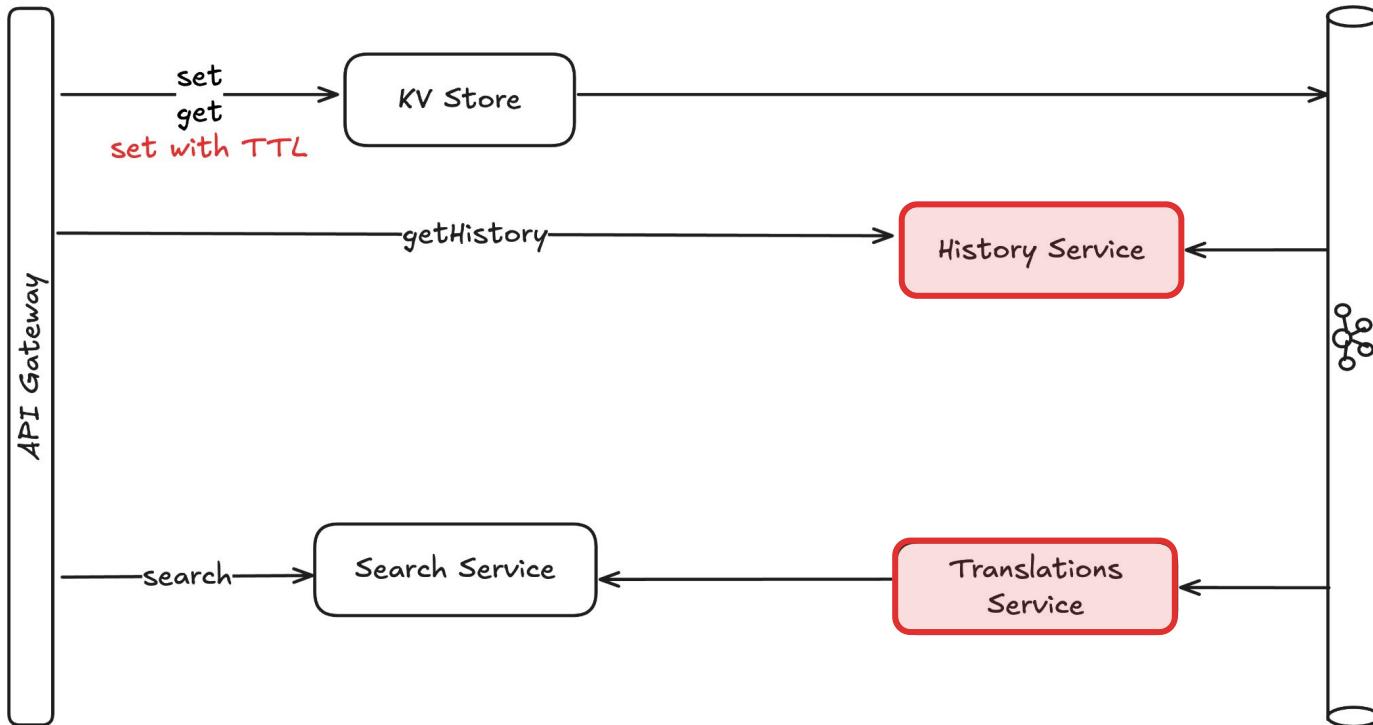
Fragments Compose In Fixed Ways



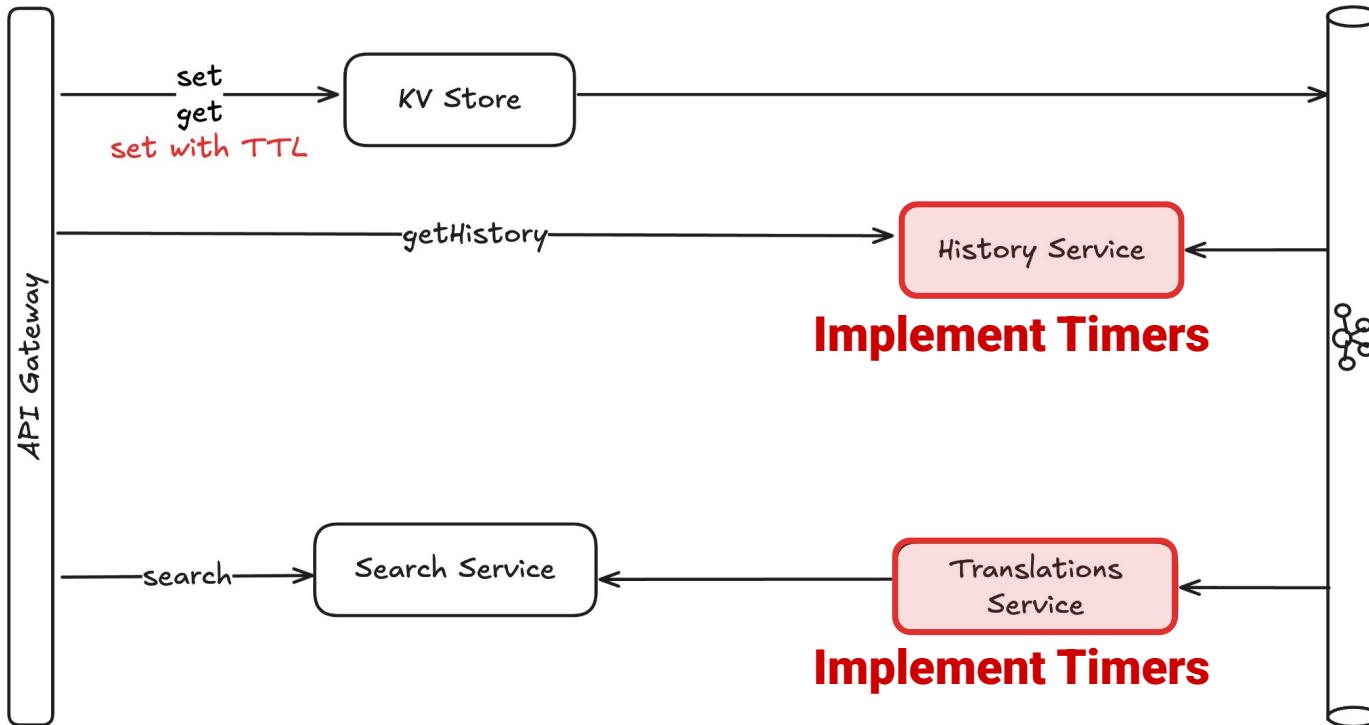
Fragments Compose In Fixed Ways



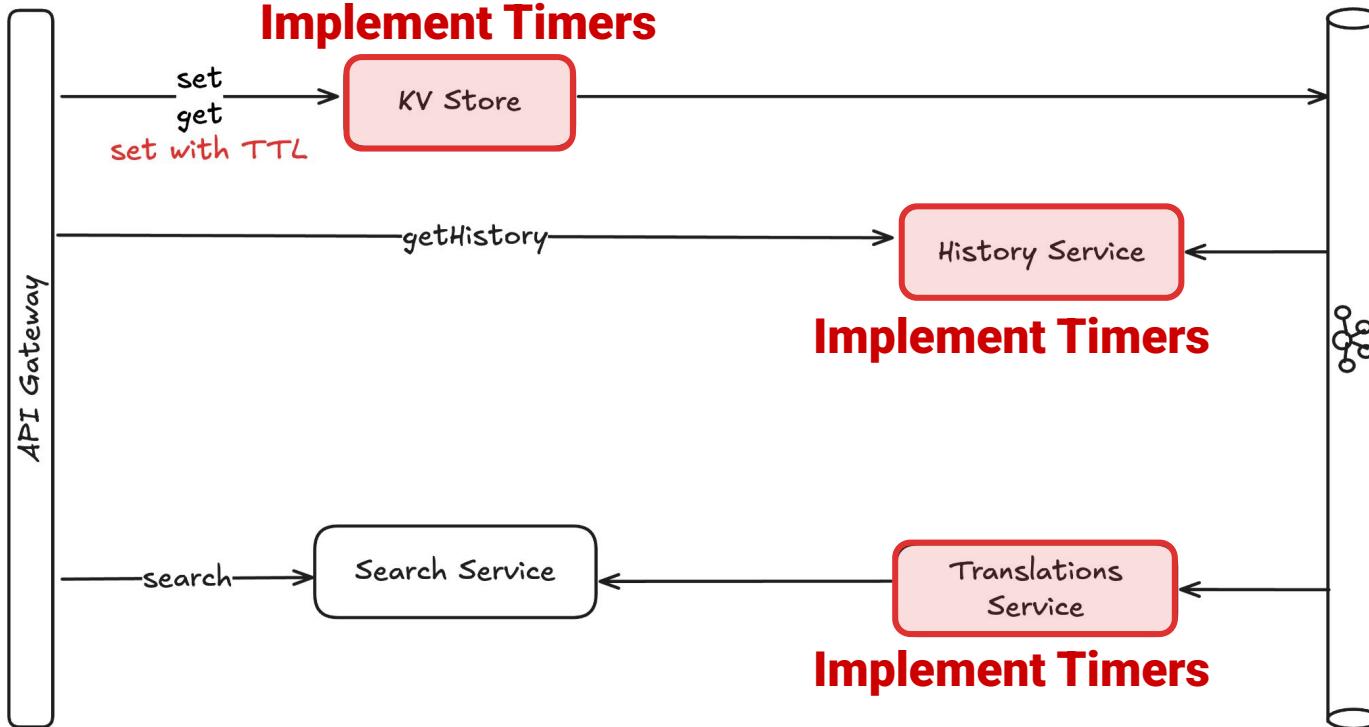
Fragments Compose In Fixed Ways



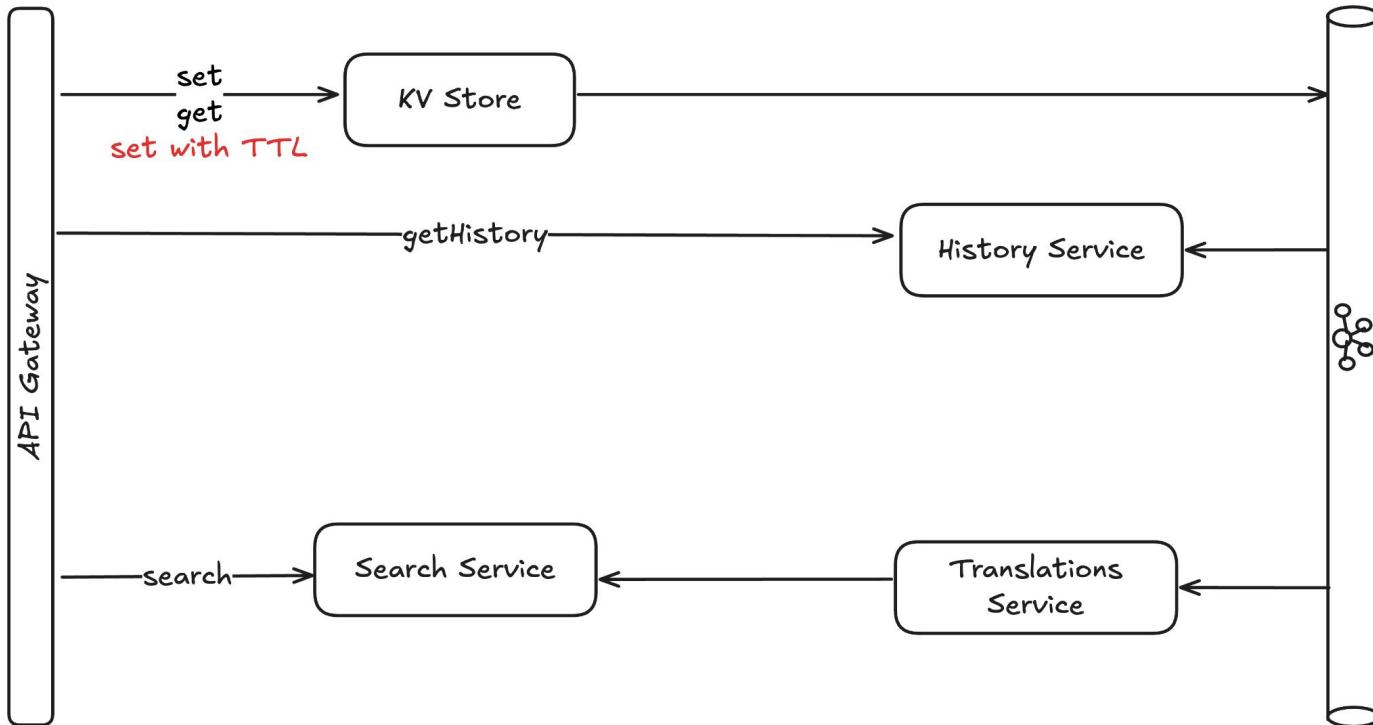
Fragments Compose In Fixed Ways



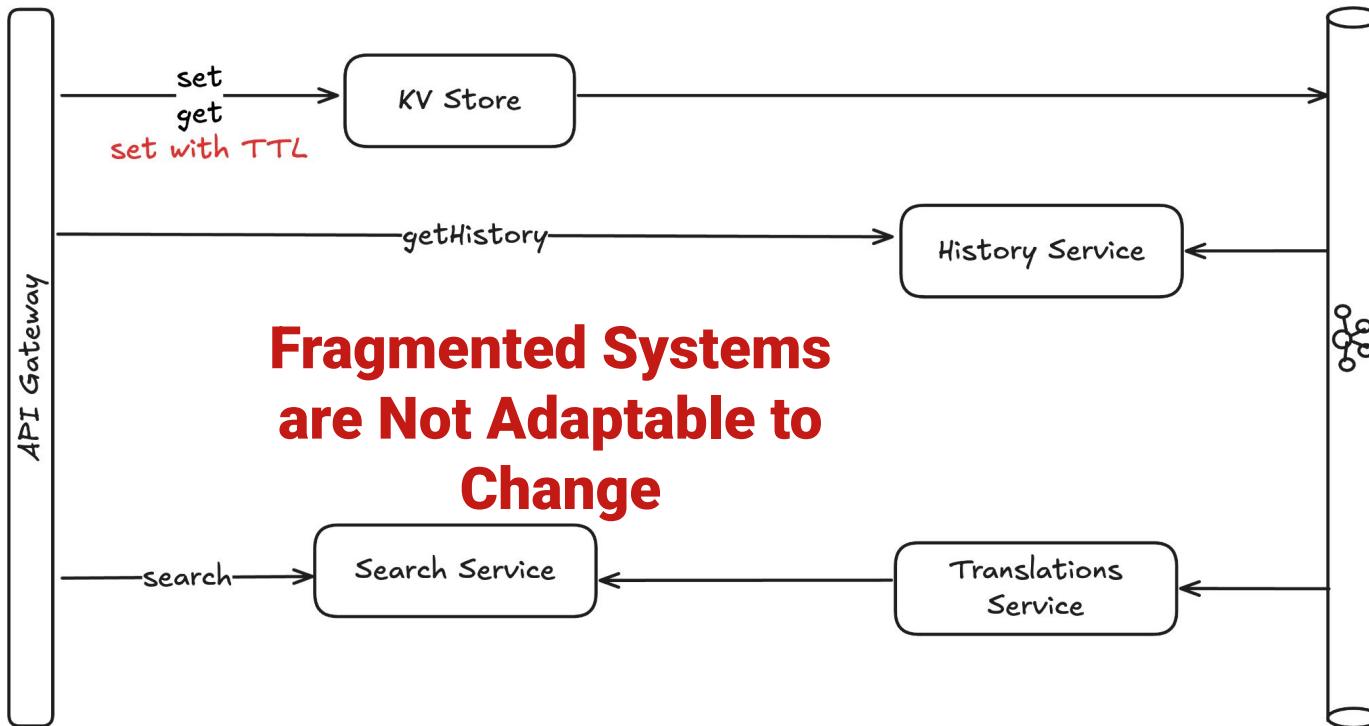
Fragments Compose In Fixed Ways



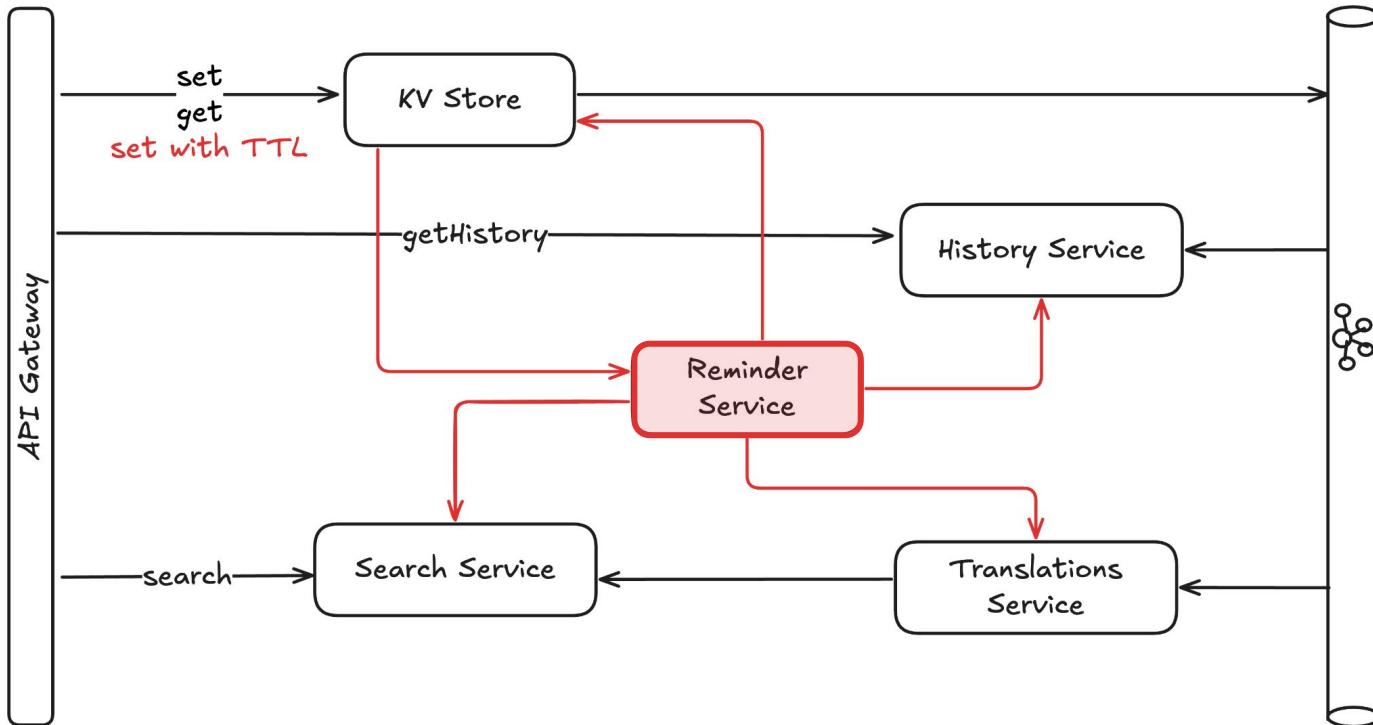
Fragments Compose In Fixed Ways



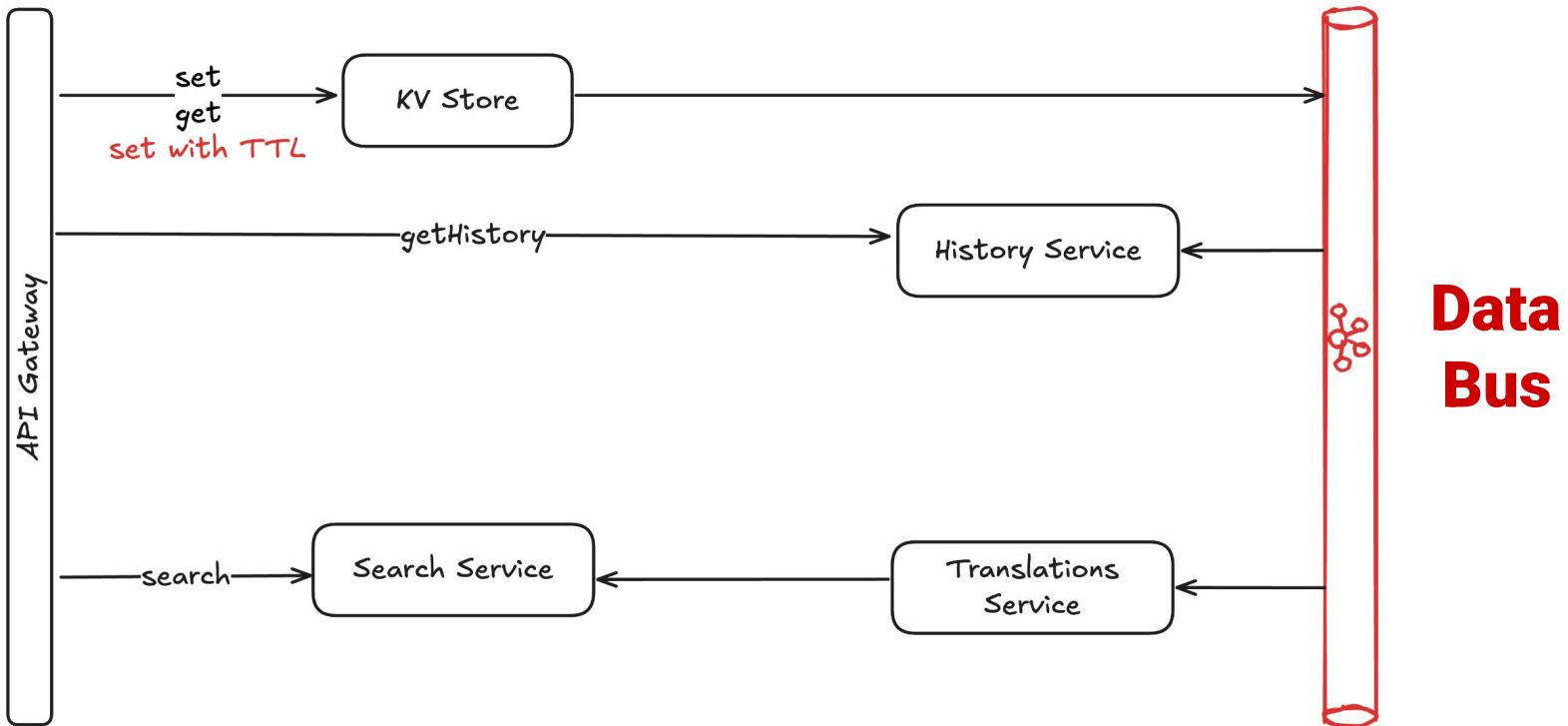
Fragments Compose In Fixed Ways



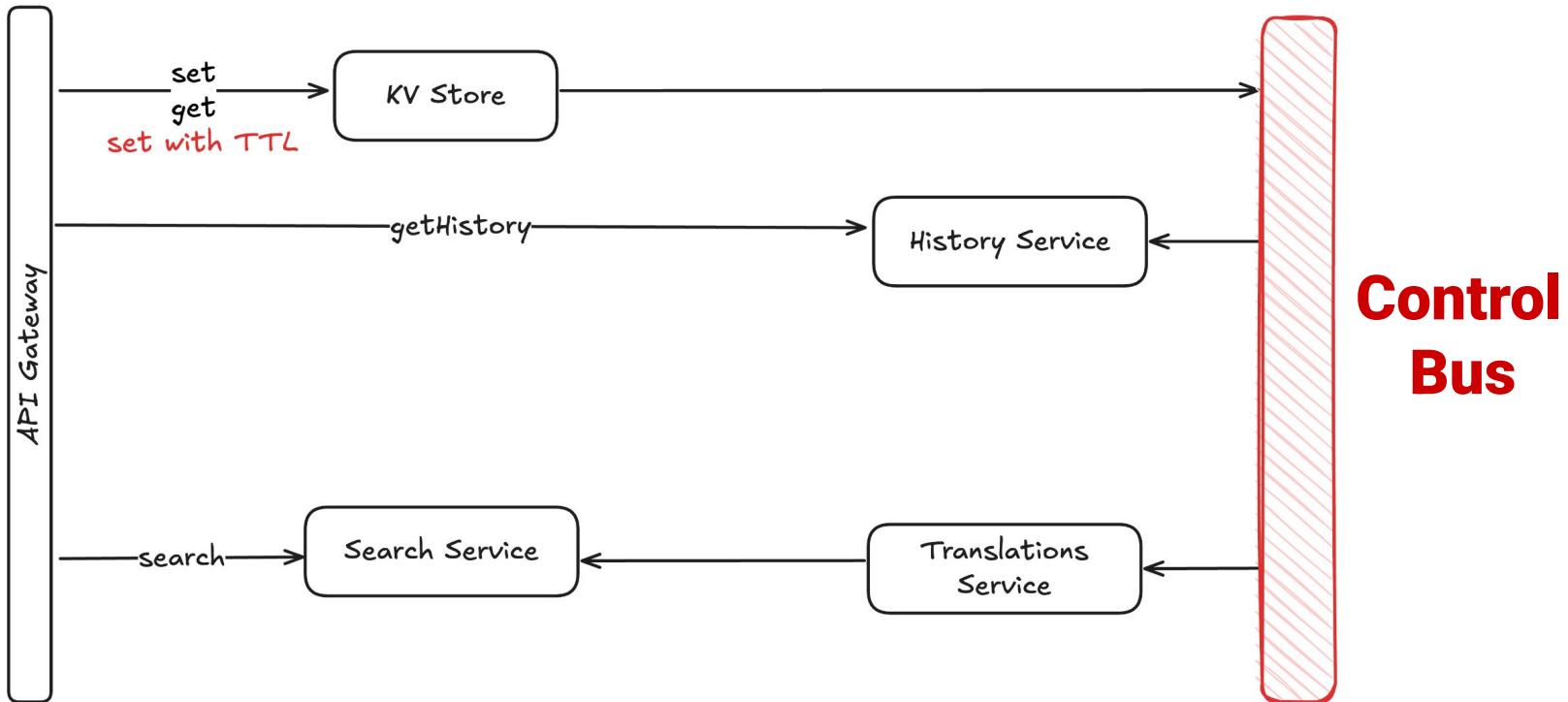
Fragments Compose In Fixed Ways



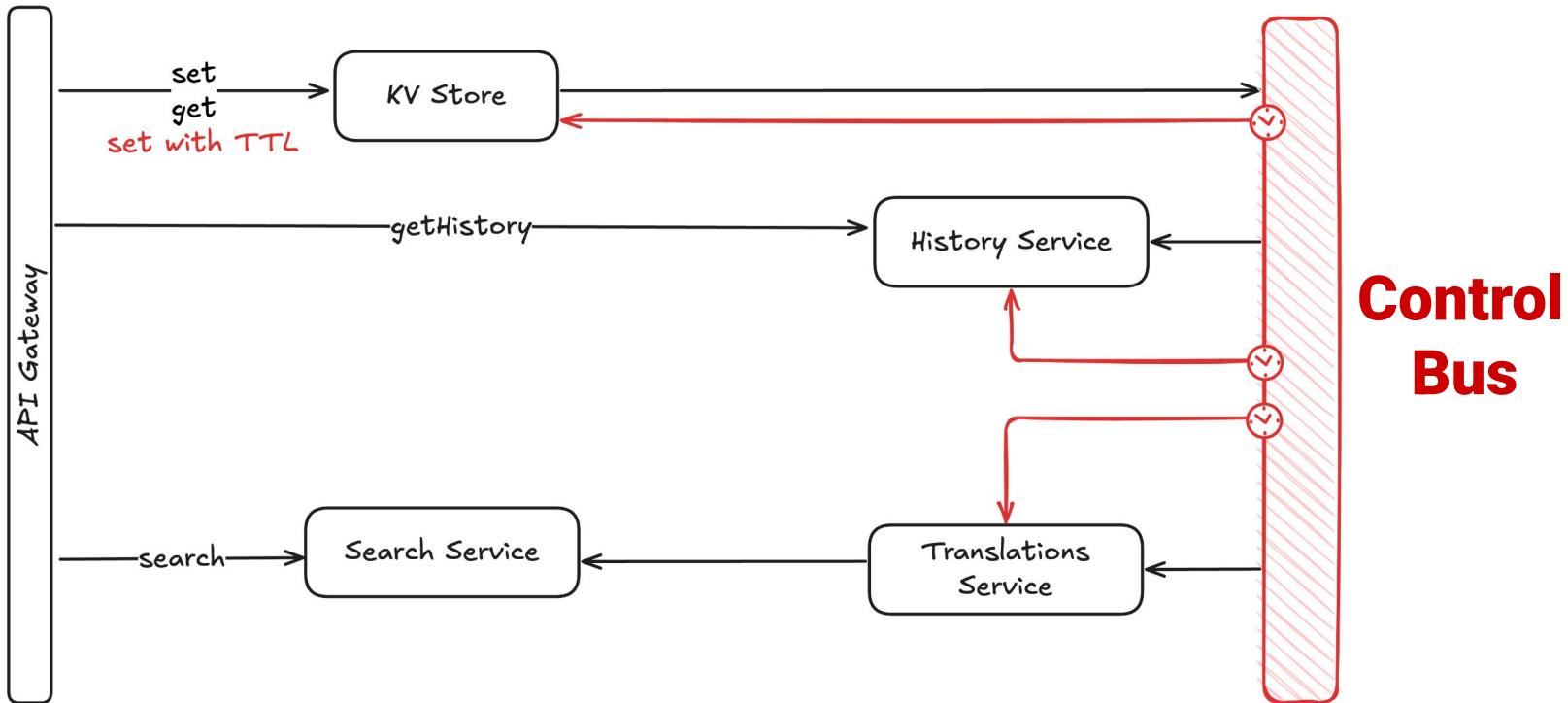
Fragments Compose In Fixed Ways



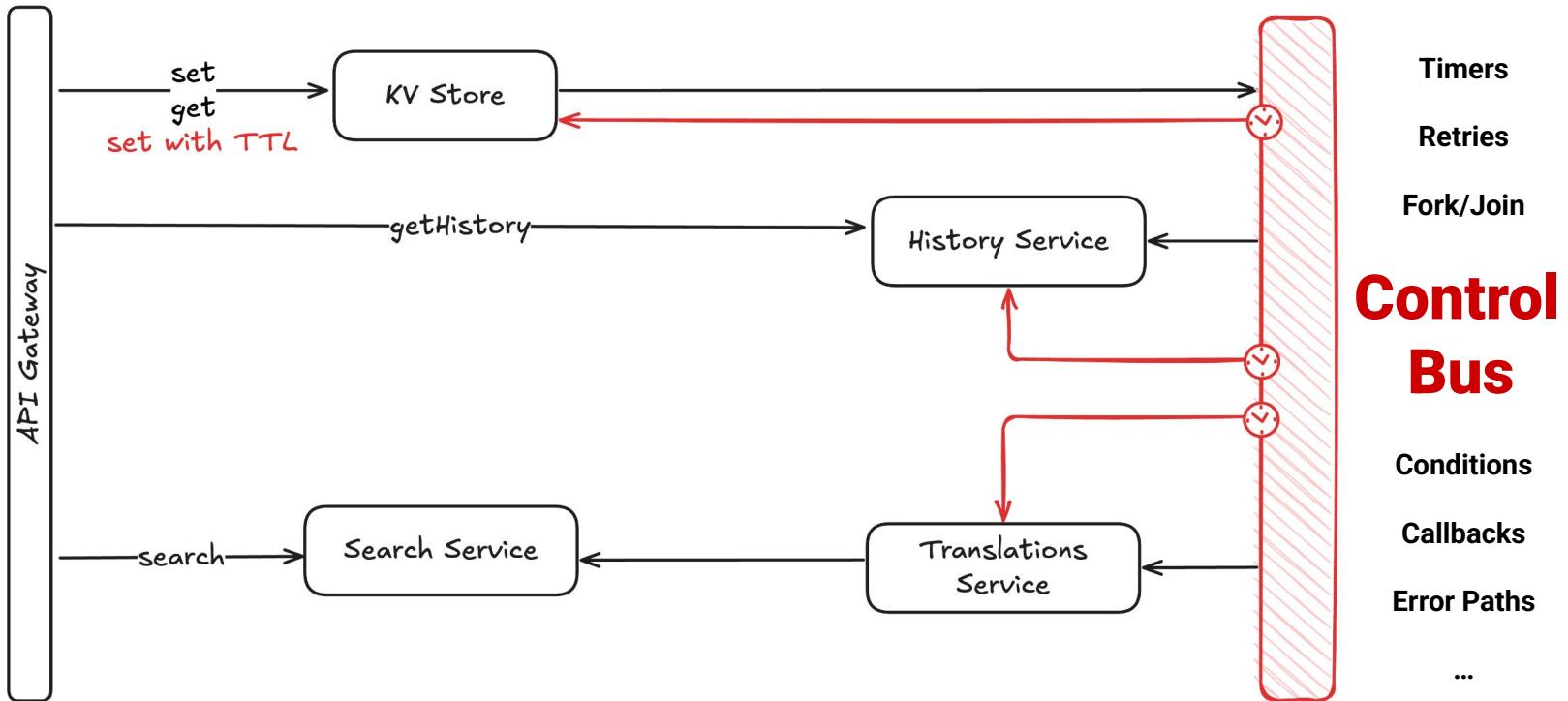
Fragments Compose In Fixed Ways



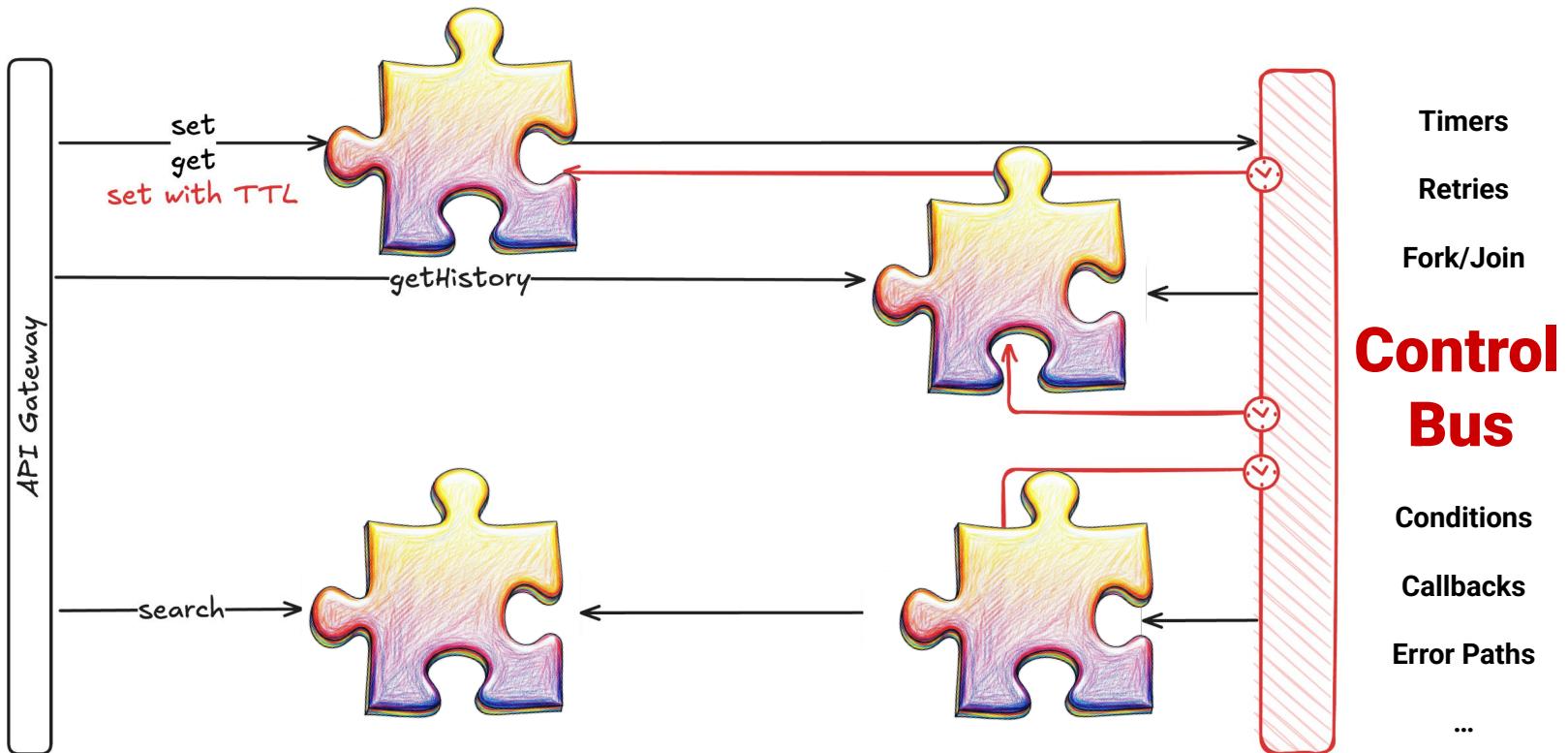
Fragments Compose In Fixed Ways



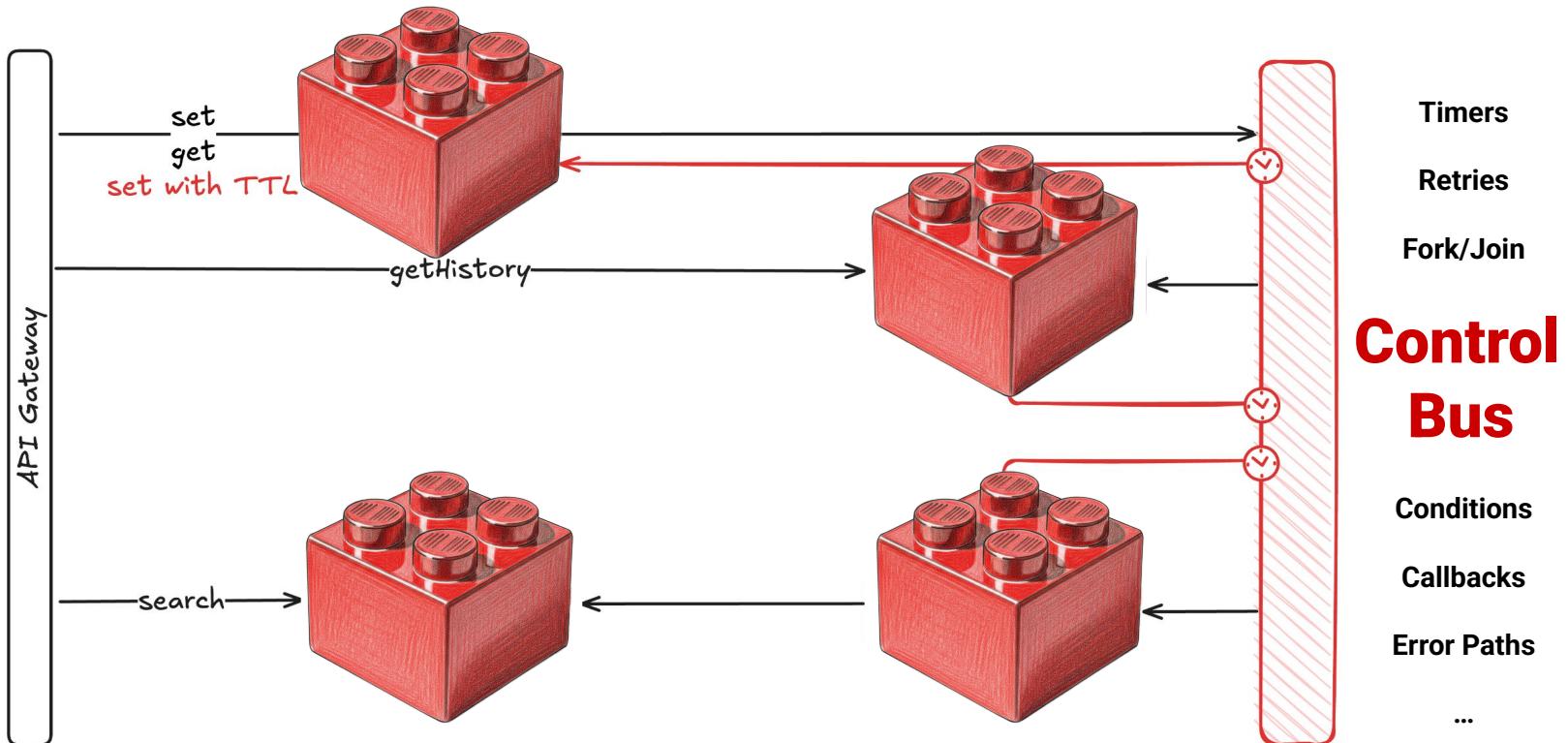
Fragments Compose In Fixed Ways



Fragments Compose In Fixed Ways



Fragments Compose In Fixed Ways



**Control
Bus**

Conditions

Callbacks

Error Paths

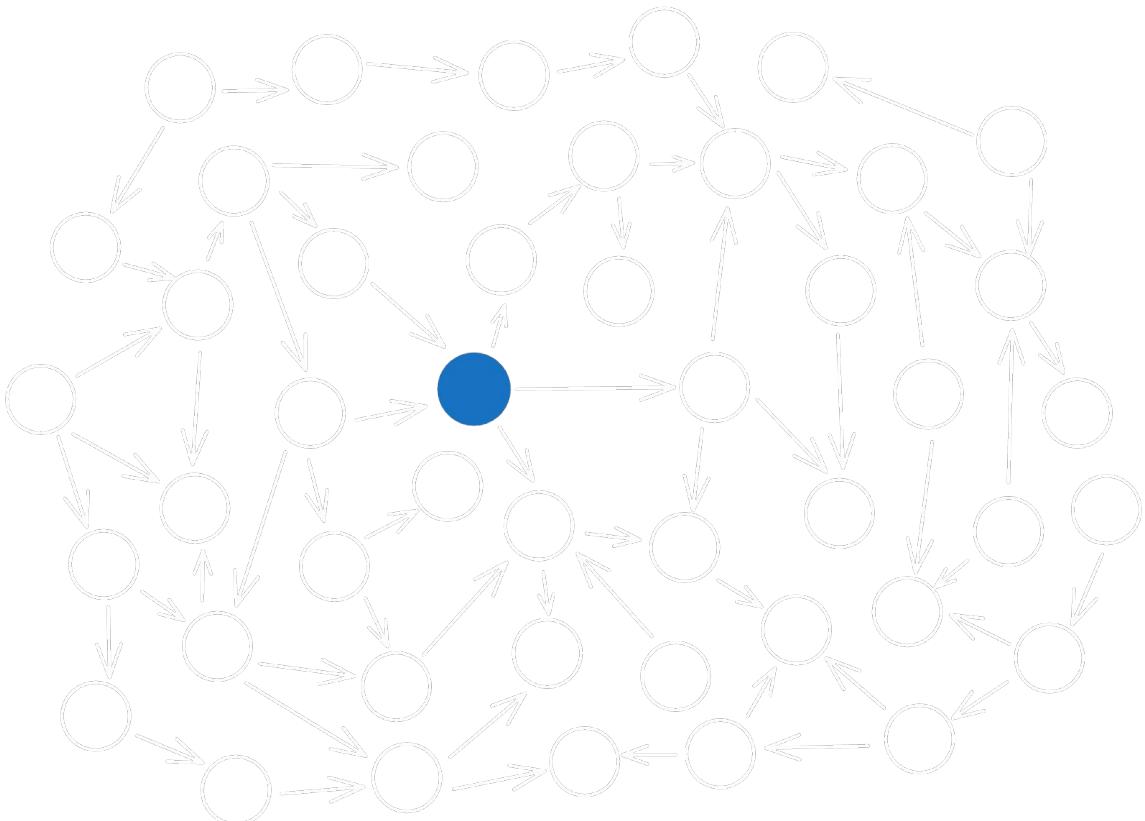
...

Hard to
Problem 2: 0

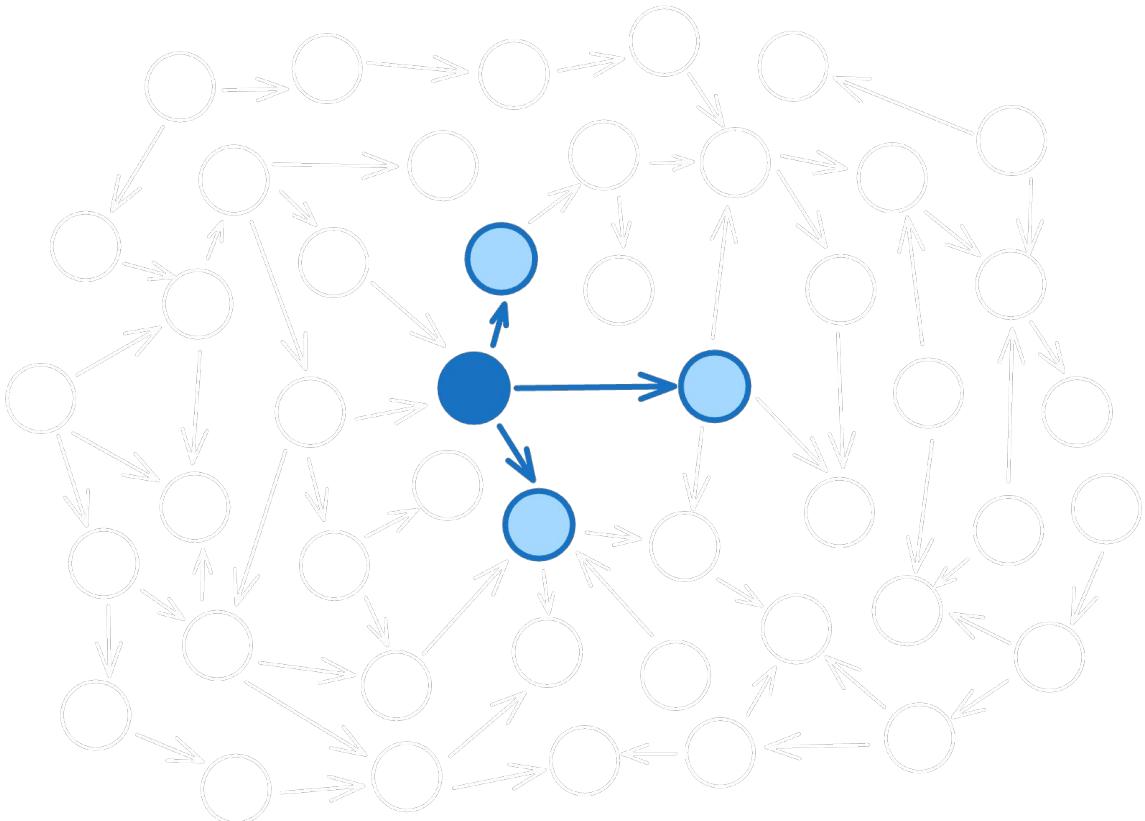
Understand

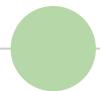


Fragmentation: Harder To Understand

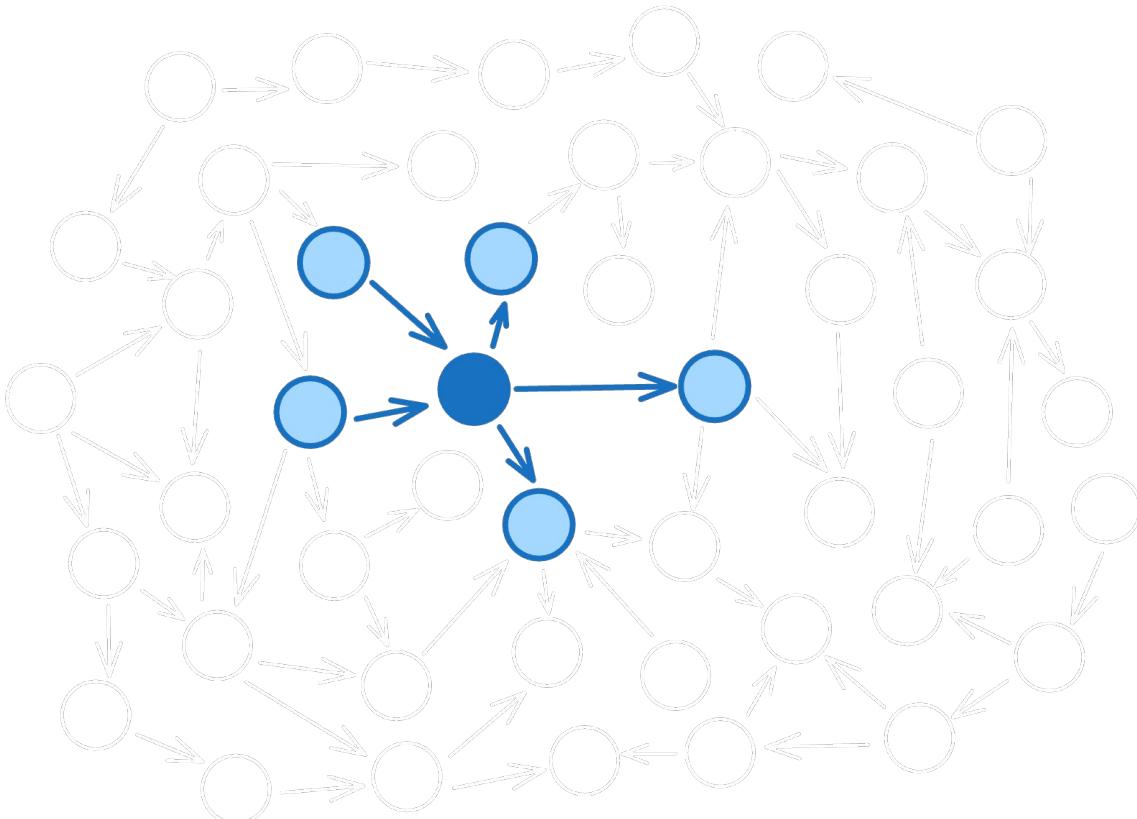


Fragmentation: Harder To Understand



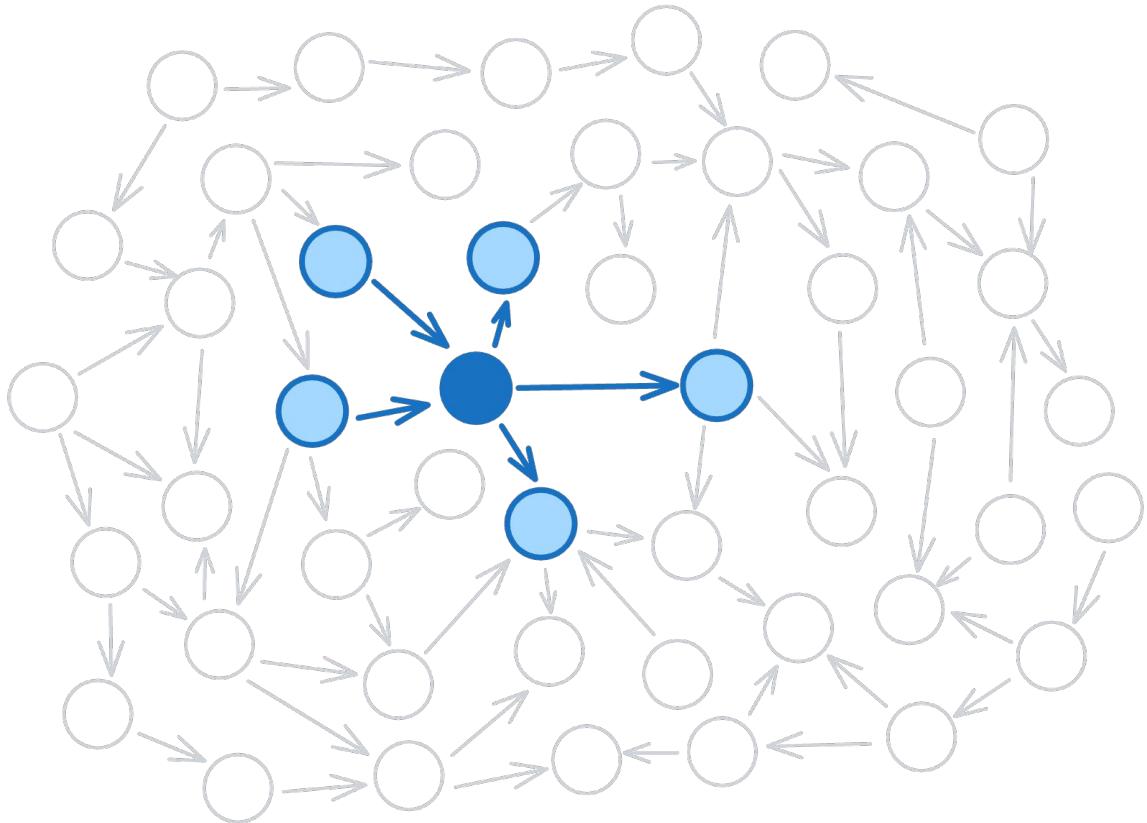


Fragmentation: Harder To Understand





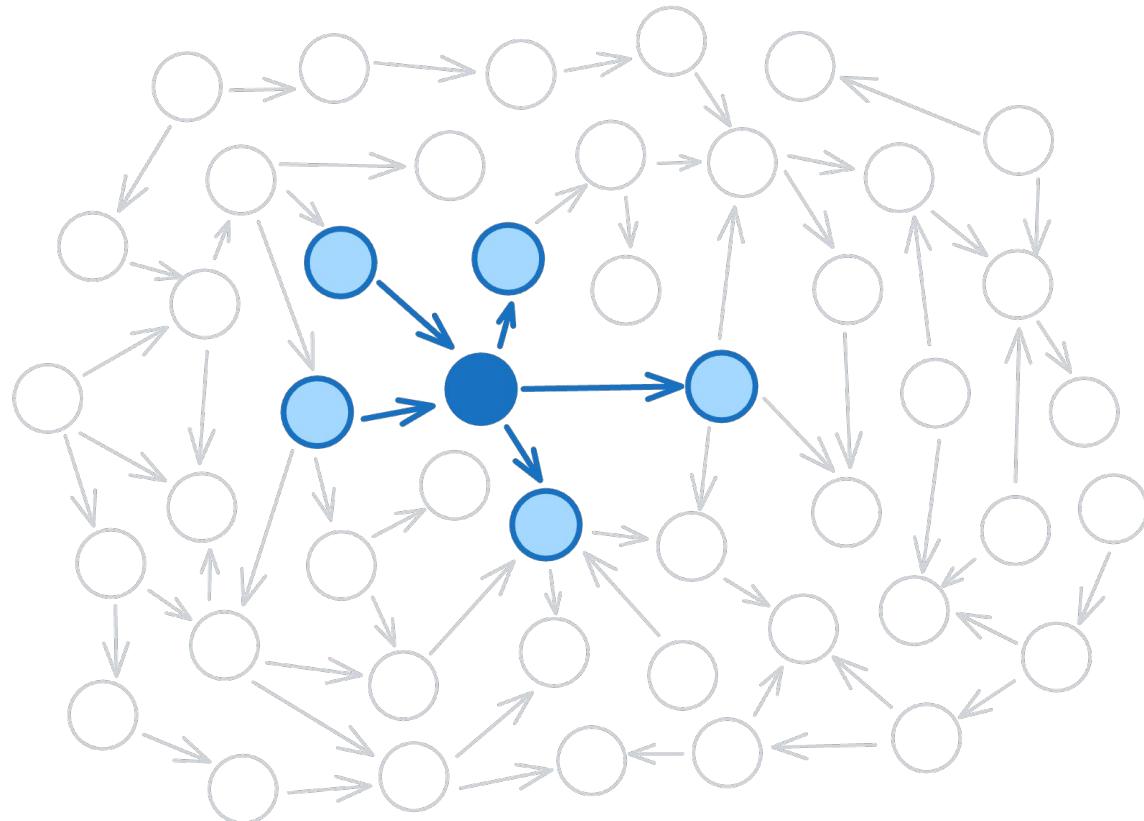
Fragmentation: Harder To Understand





Fragmentation: Harder To Understand

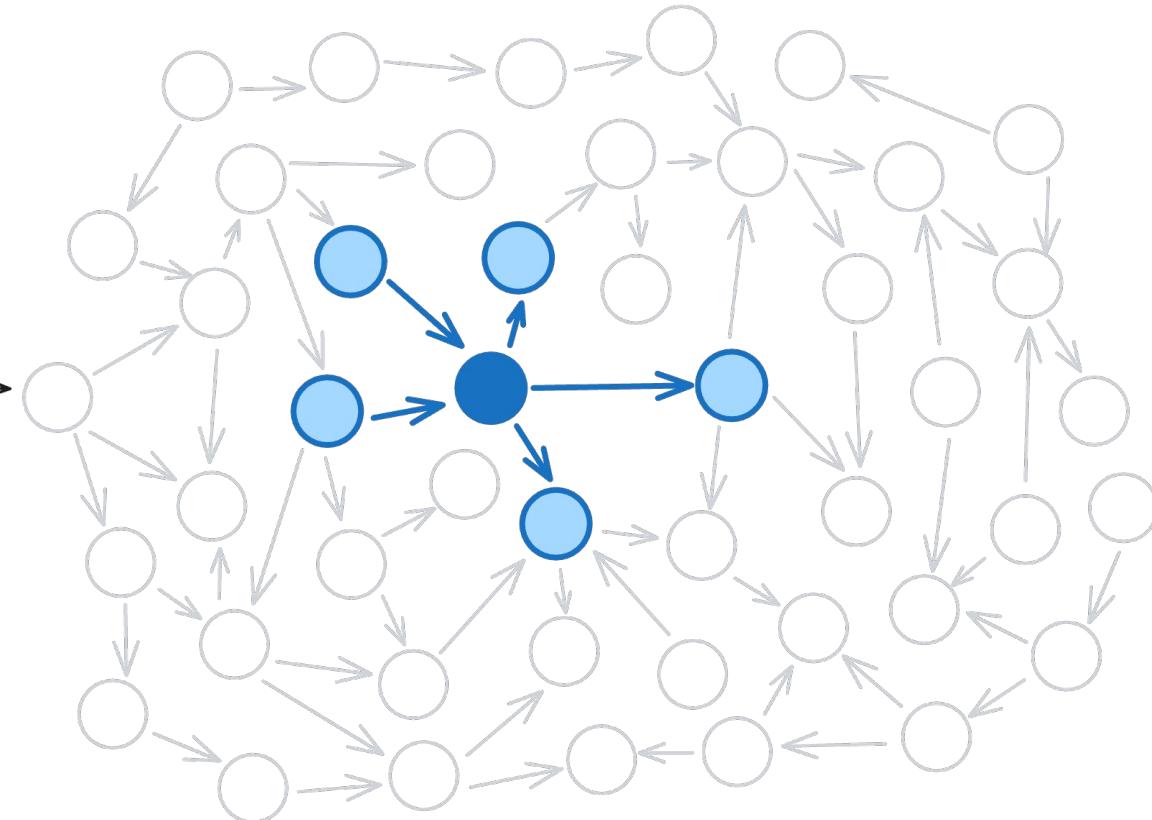
Tiny slice of the
larger picture





Fragmentation: Harder To Understand

Tiny slice of the
larger picture





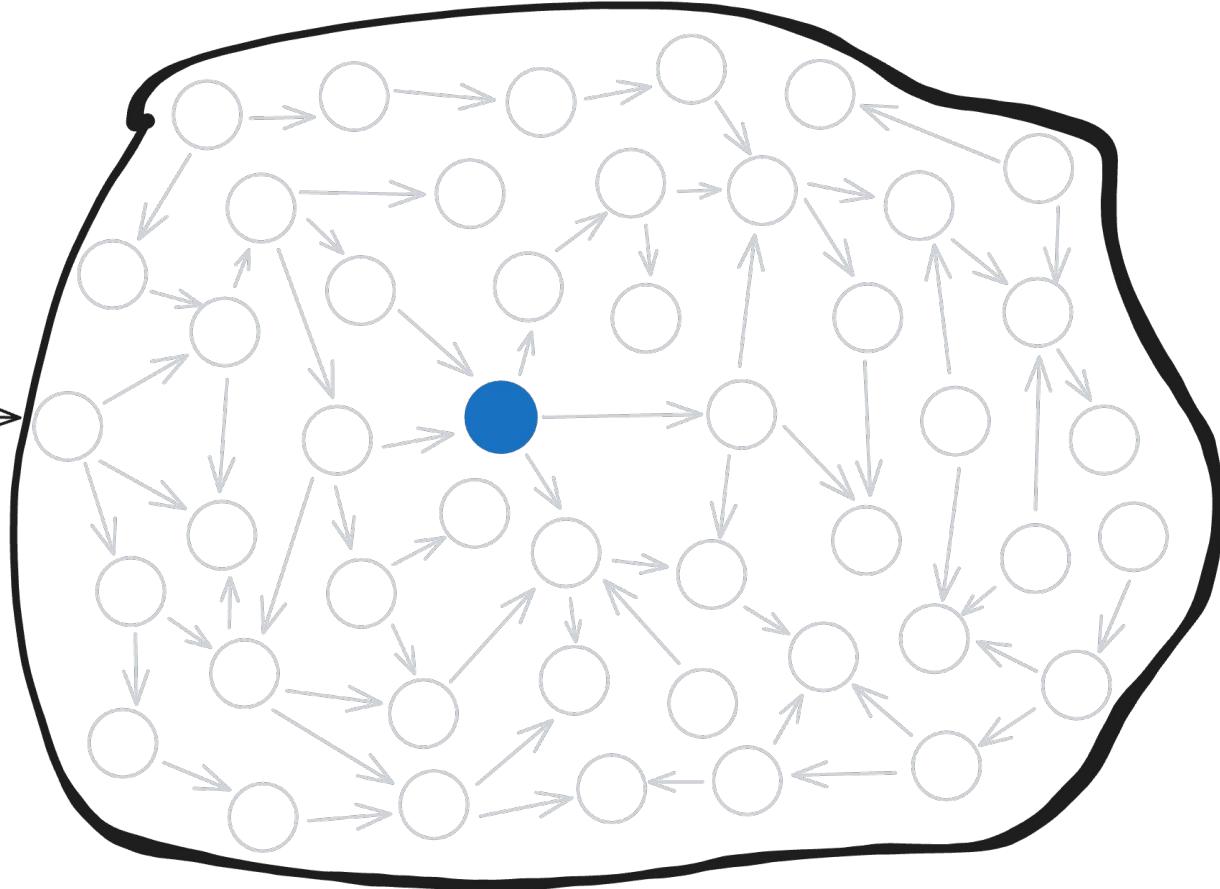
Fragmentation: Harder To Understand



**The Product Works
as One Whole**

Fragmentation: Harder To Understand

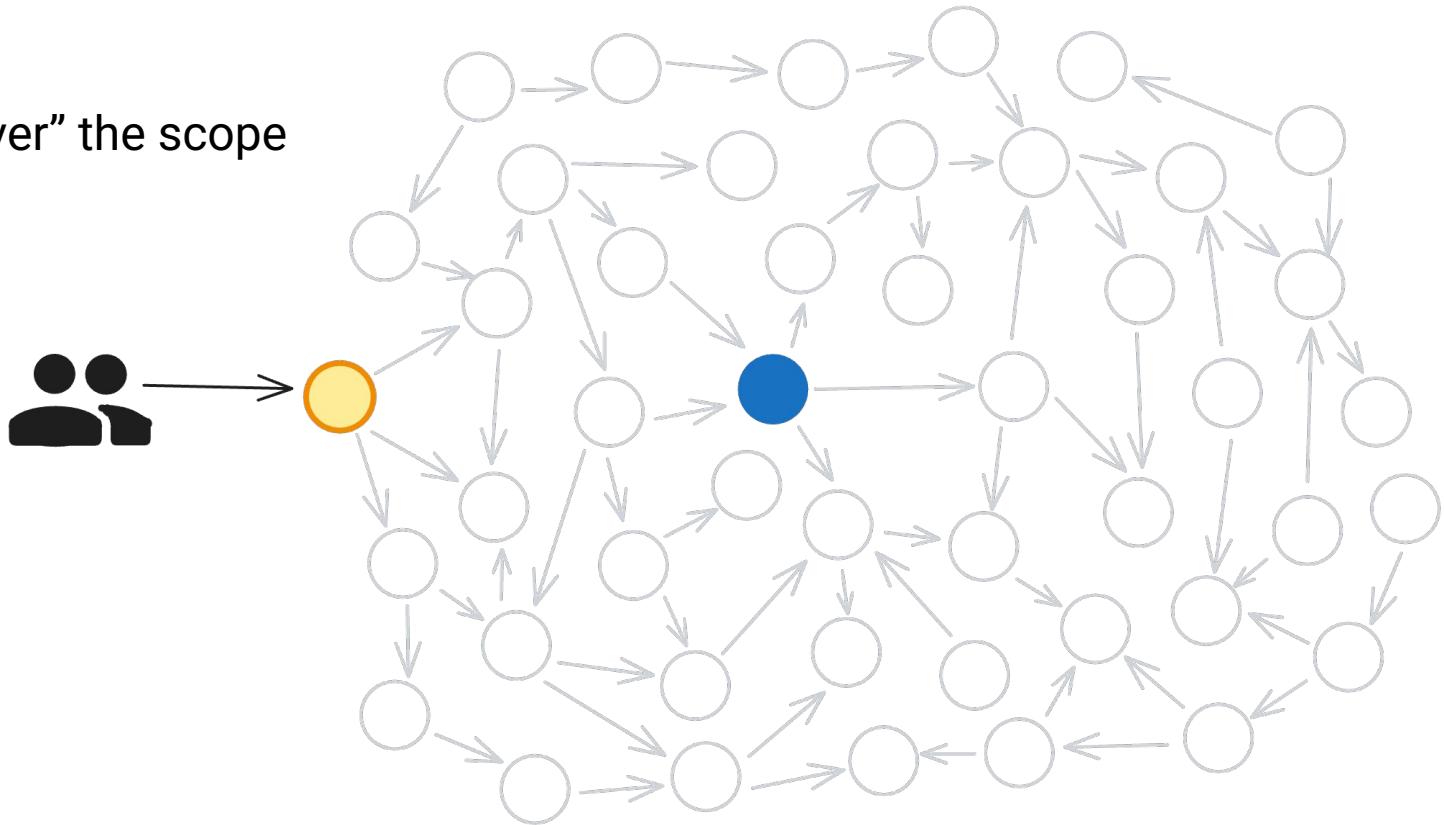
Local context is
not enough





Fragmentation: Harder To Understand

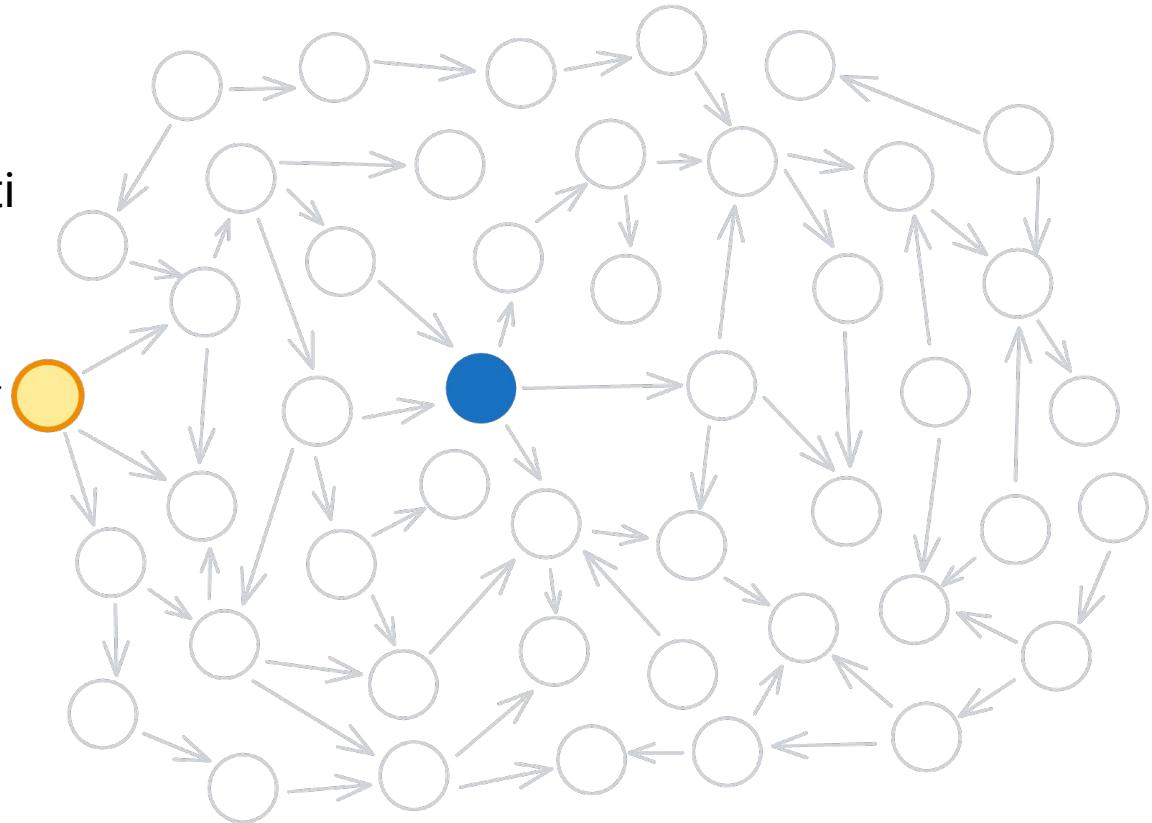
“Discover” the scope

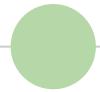




Fragmentation: Harder To Understand

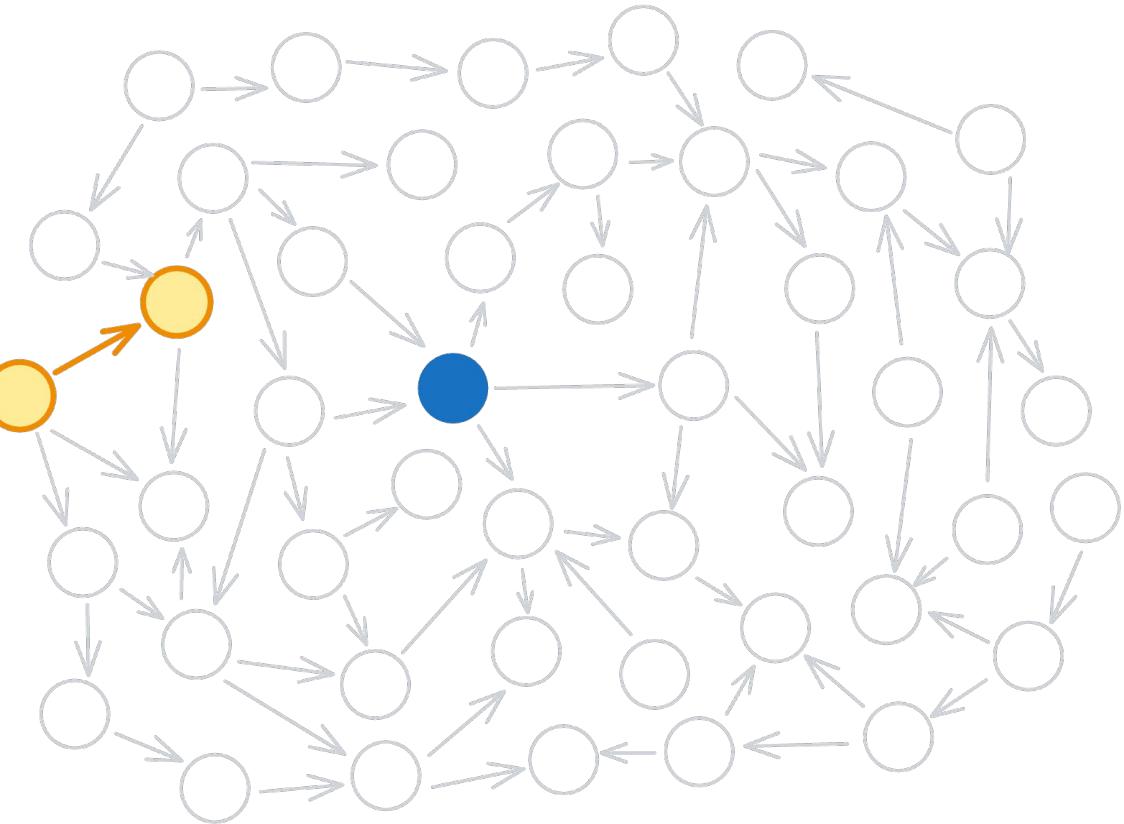
“Discover” the scope
Untangle the spaghetti

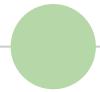




Fragmentation: Harder To Understand

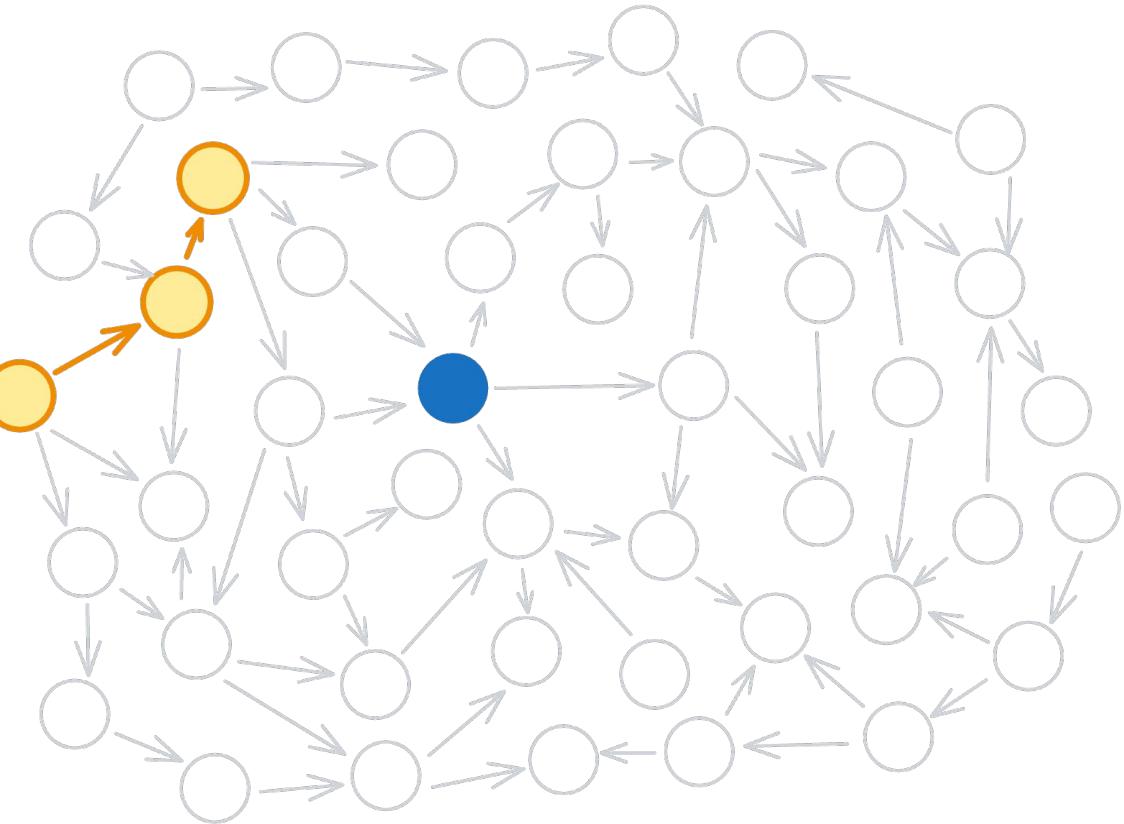
“Discover” the scope
Untangle the spaghetti





Fragmentation: Harder To Understand

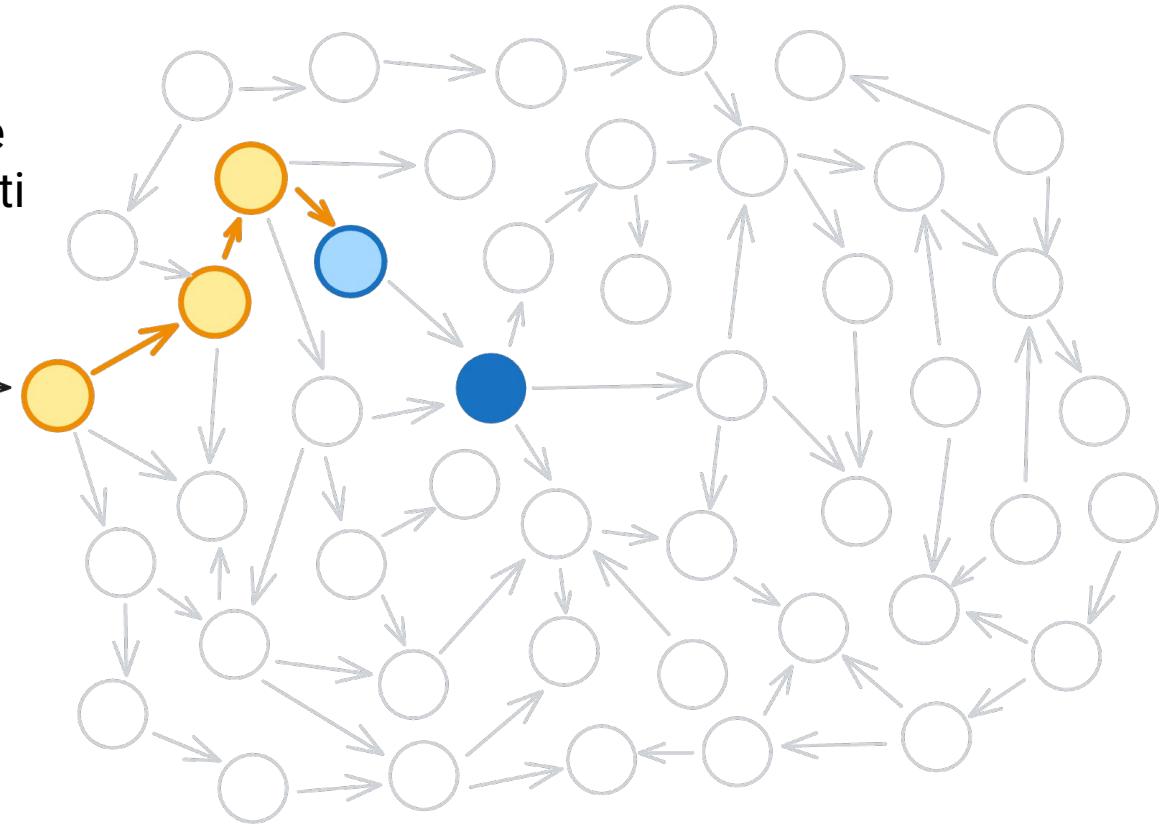
“Discover” the scope
Untangle the spaghetti





Fragmentation: Harder To Understand

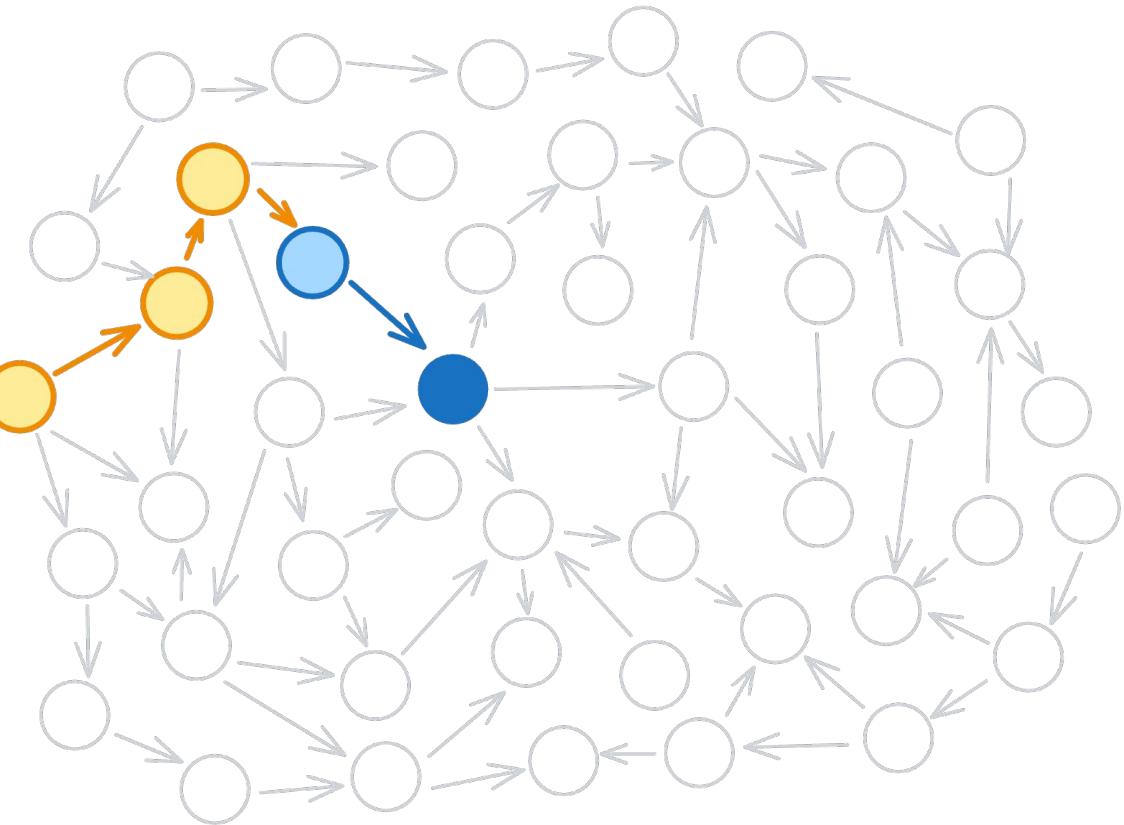
“Discover” the scope
Untangle the spaghetti





Fragmentation: Harder To Understand

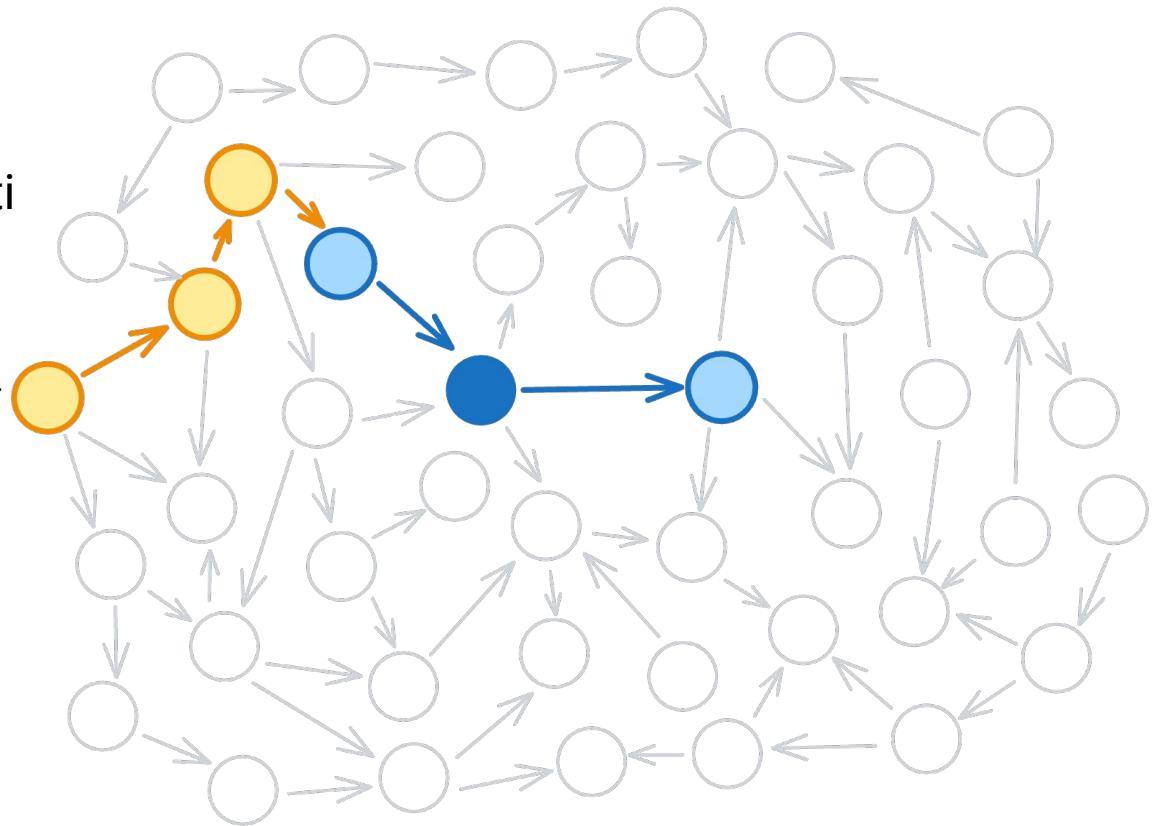
“Discover” the scope
Untangle the spaghetti





Fragmentation: Harder To Understand

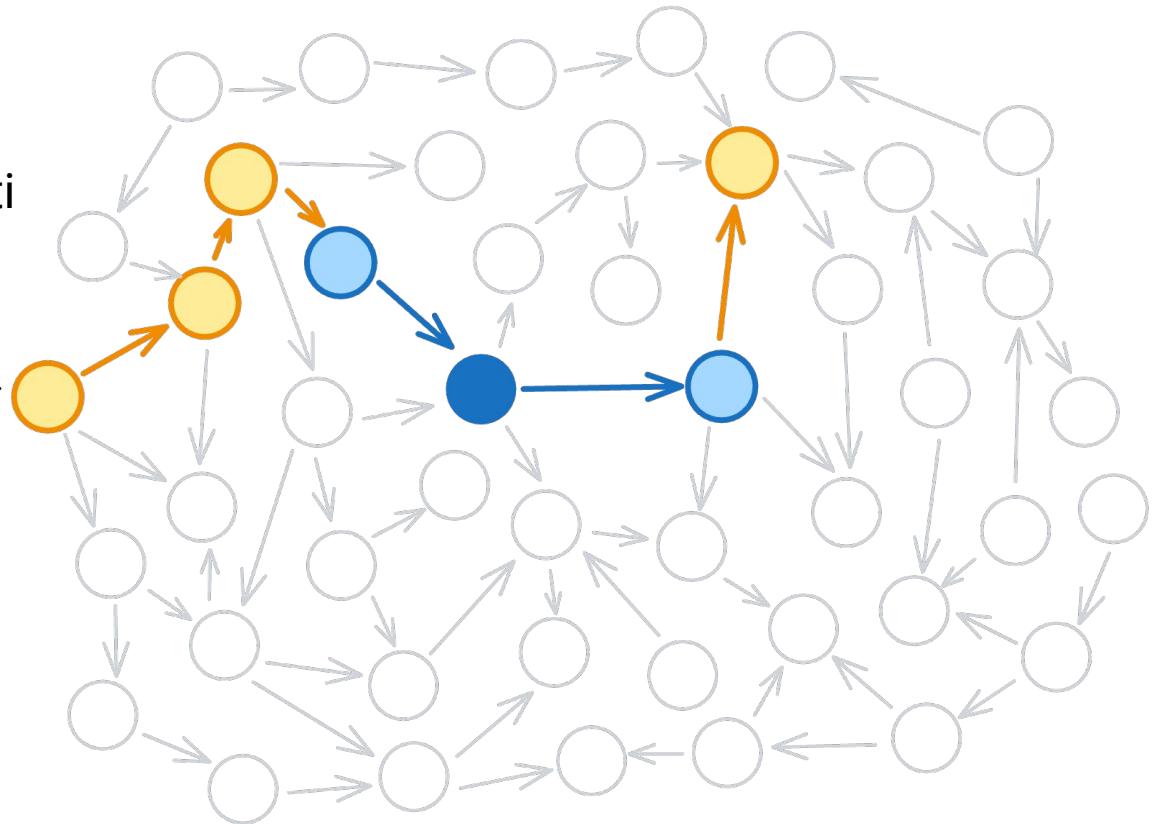
“Discover” the scope
Untangle the spaghetti

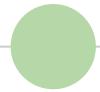




Fragmentation: Harder To Understand

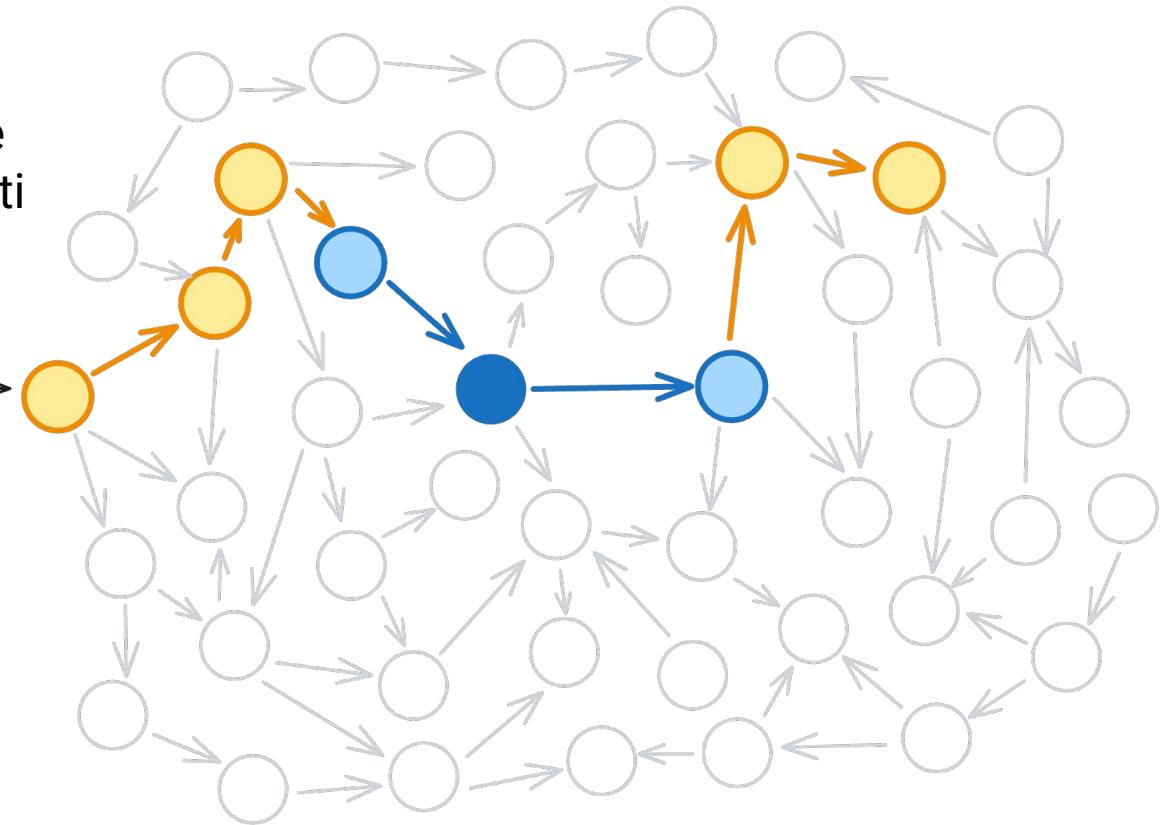
“Discover” the scope
Untangle the spaghetti

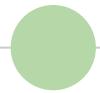




Fragmentation: Harder To Understand

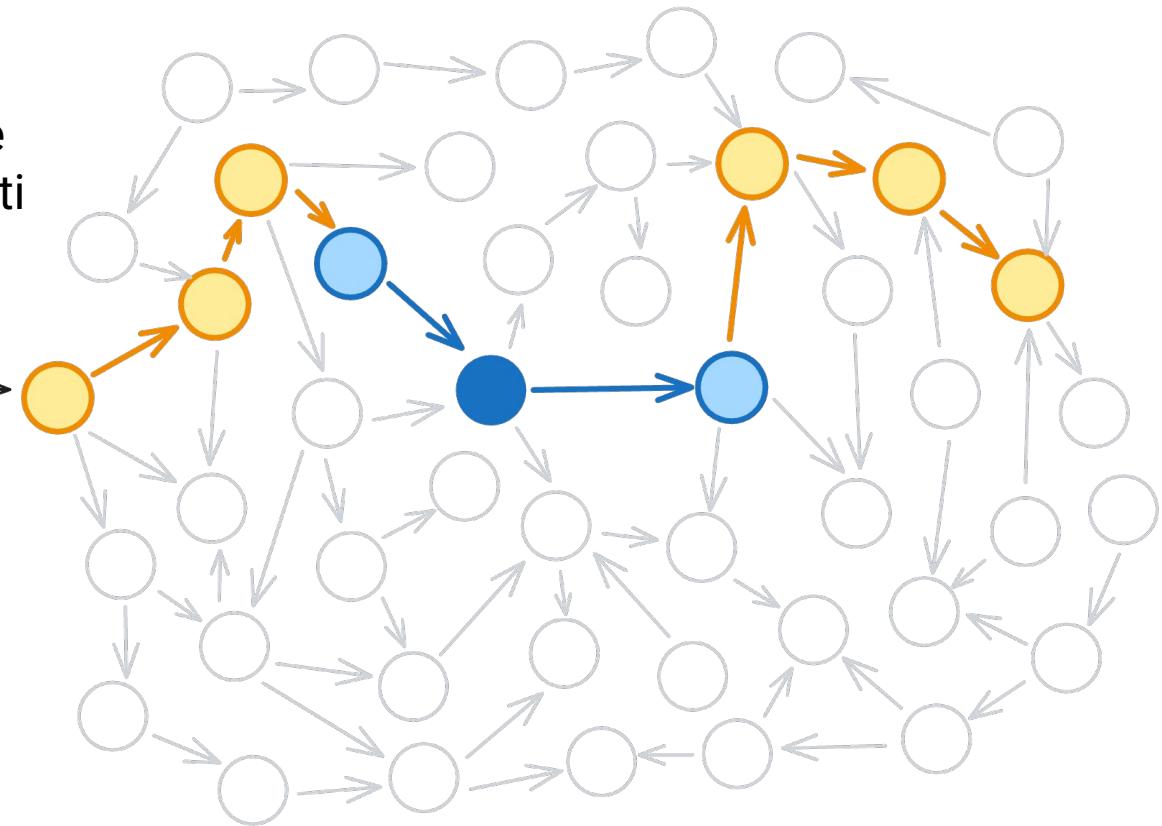
“Discover” the scope
Untangle the spaghetti





Fragmentation: Harder To Understand

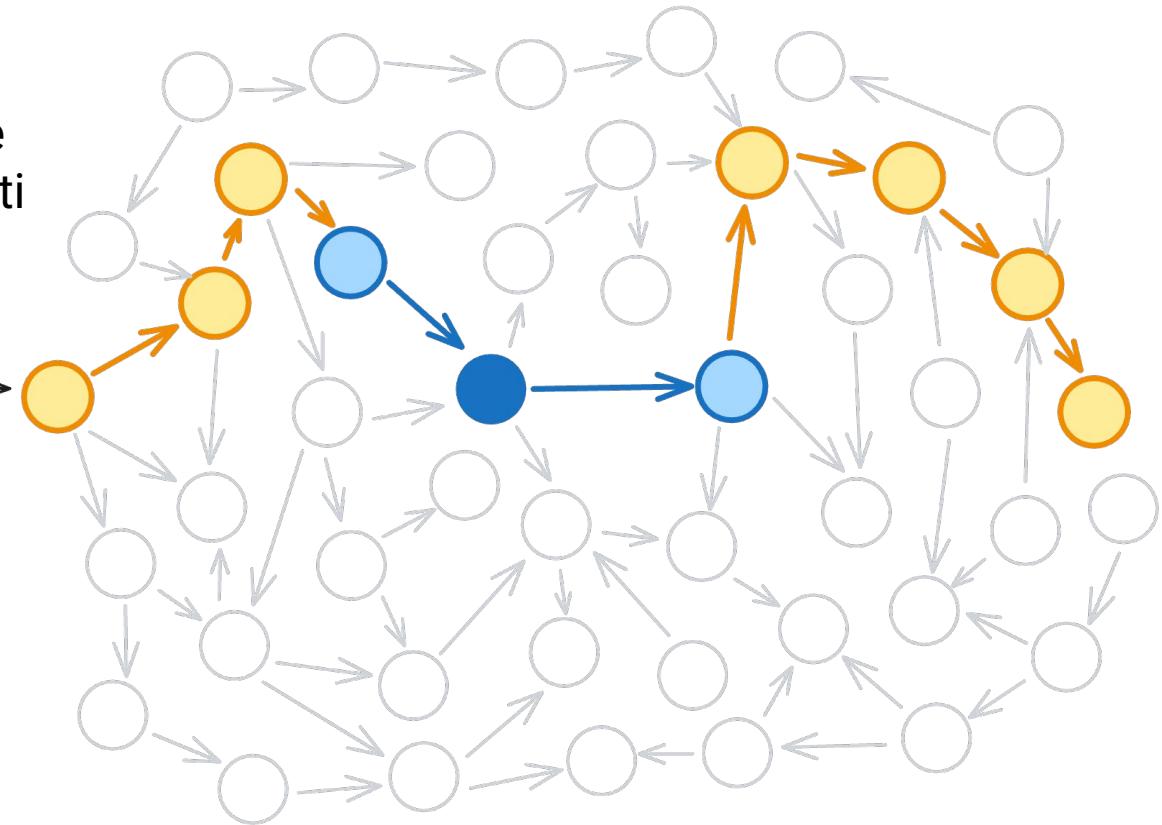
“Discover” the scope
Untangle the spaghetti





Fragmentation: Harder To Understand

“Discover” the scope
Untangle the spaghetti





Fragmentation: Harder To Understand

“Discover” the scope
Untangle the spaghetti



**A Better Mental
Model**

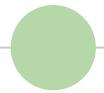


Organizational Cohesion



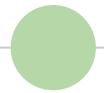
Organizational Cohesion

- Tech teams understand **the local context**
 - Microservices, requests, responses, data flow
 - Dependencies, protocols



Organizational Cohesion

- Tech teams understand **the local context**
 - Microservices, requests, responses, data flow
 - Dependencies, protocols
- Non-tech teams understand **the global context**
 - Why the user wants to do something
 - What needs to change across the product



Organizational Cohesion

- Every system design starts with **a diagram**



Organizational Cohesion

- Every system design starts with **a diagram**
- Most diagrams are historical reflections

Solving Fragmentation :

Diagram Driven Systems



Fixing Fragmentation With Diagrams



Fixing Fragmentation With Diagrams

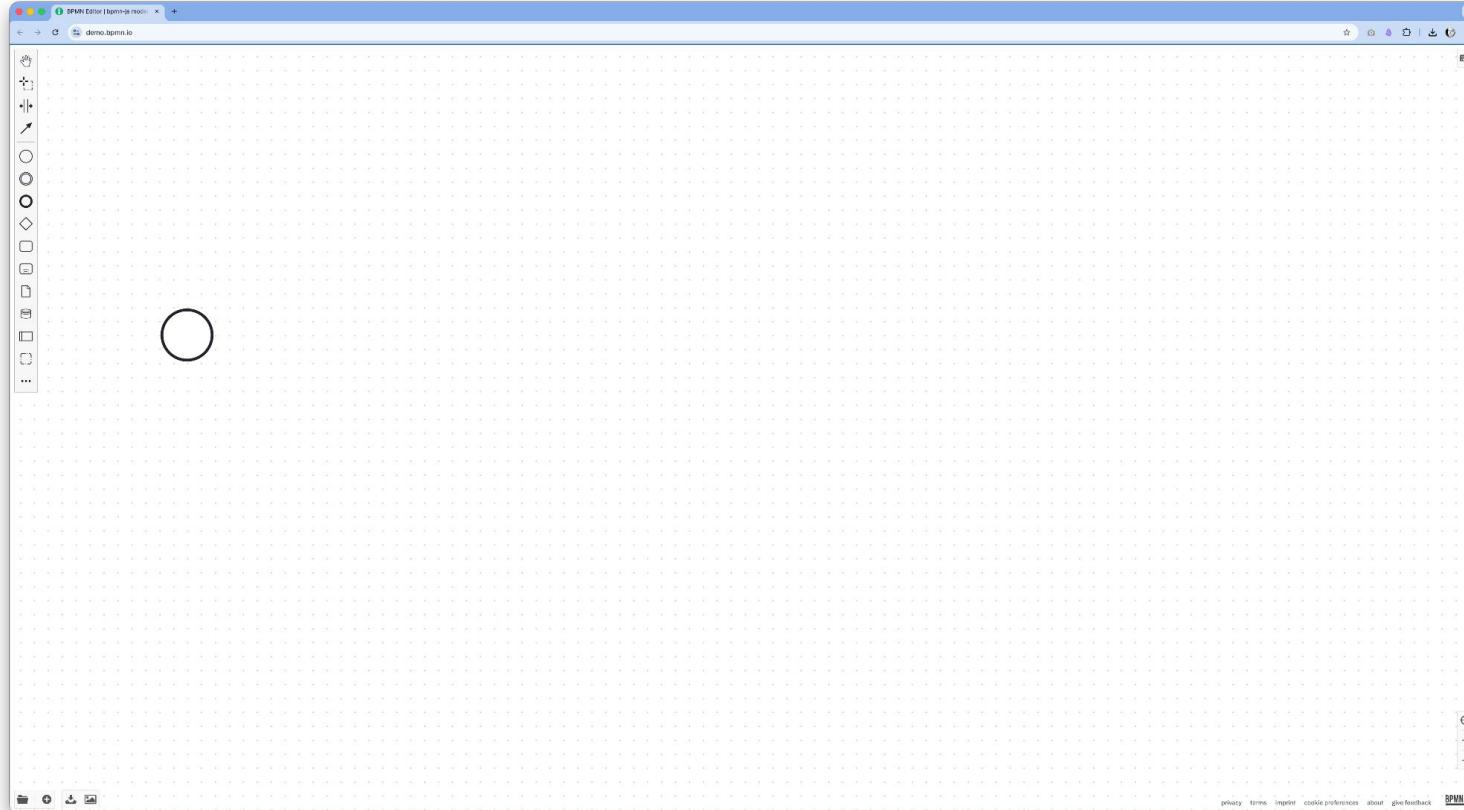
- Problem 1: Not adaptable to change
 - Fragments compose in **pre-defined ways**



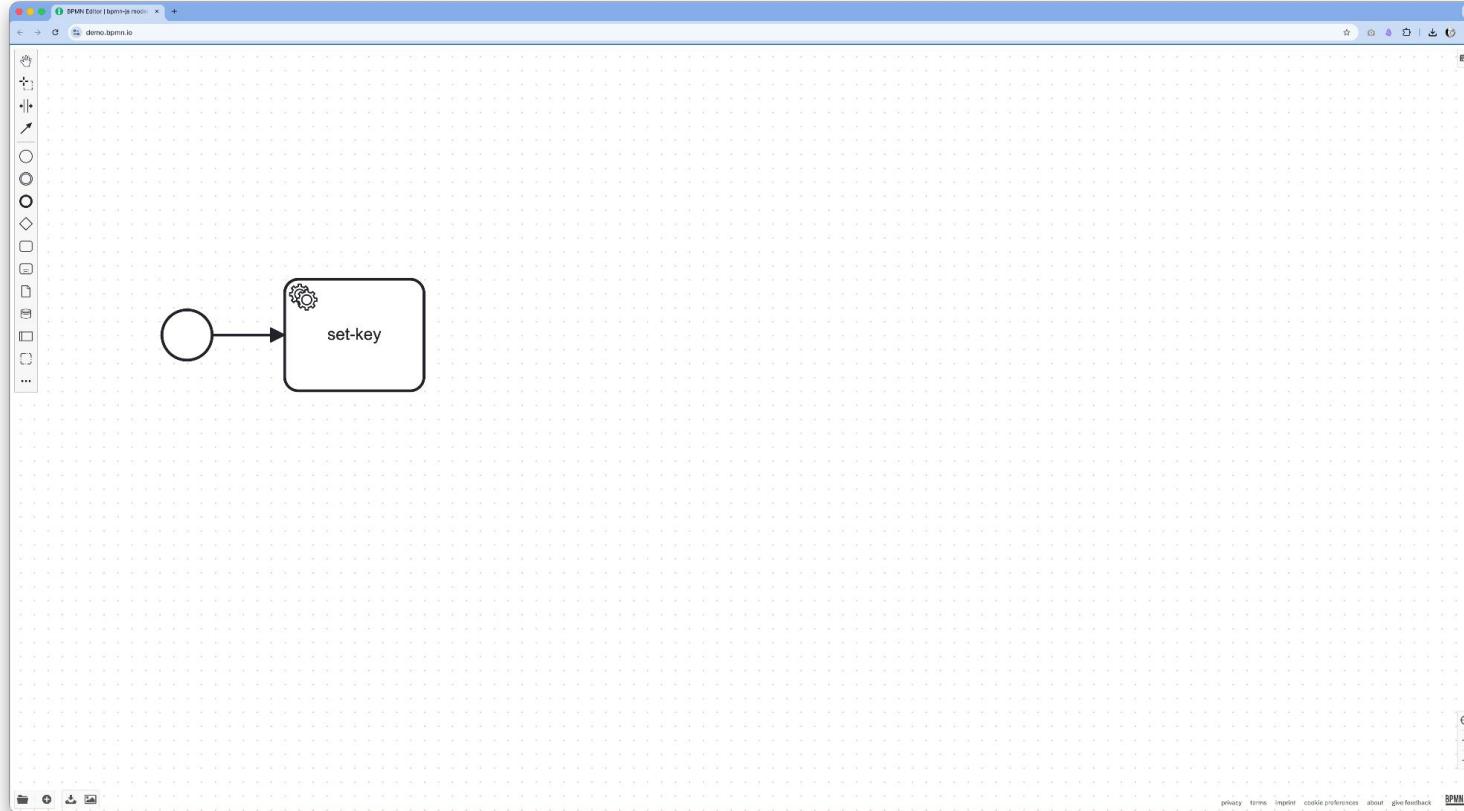
Fixing Fragmentation With Diagrams

- Problem 1: Not adaptable to change
 - Fragments compose in **pre-defined ways**
- Problem 2: Not understandable
 - **Global intent gets lost** in the local context

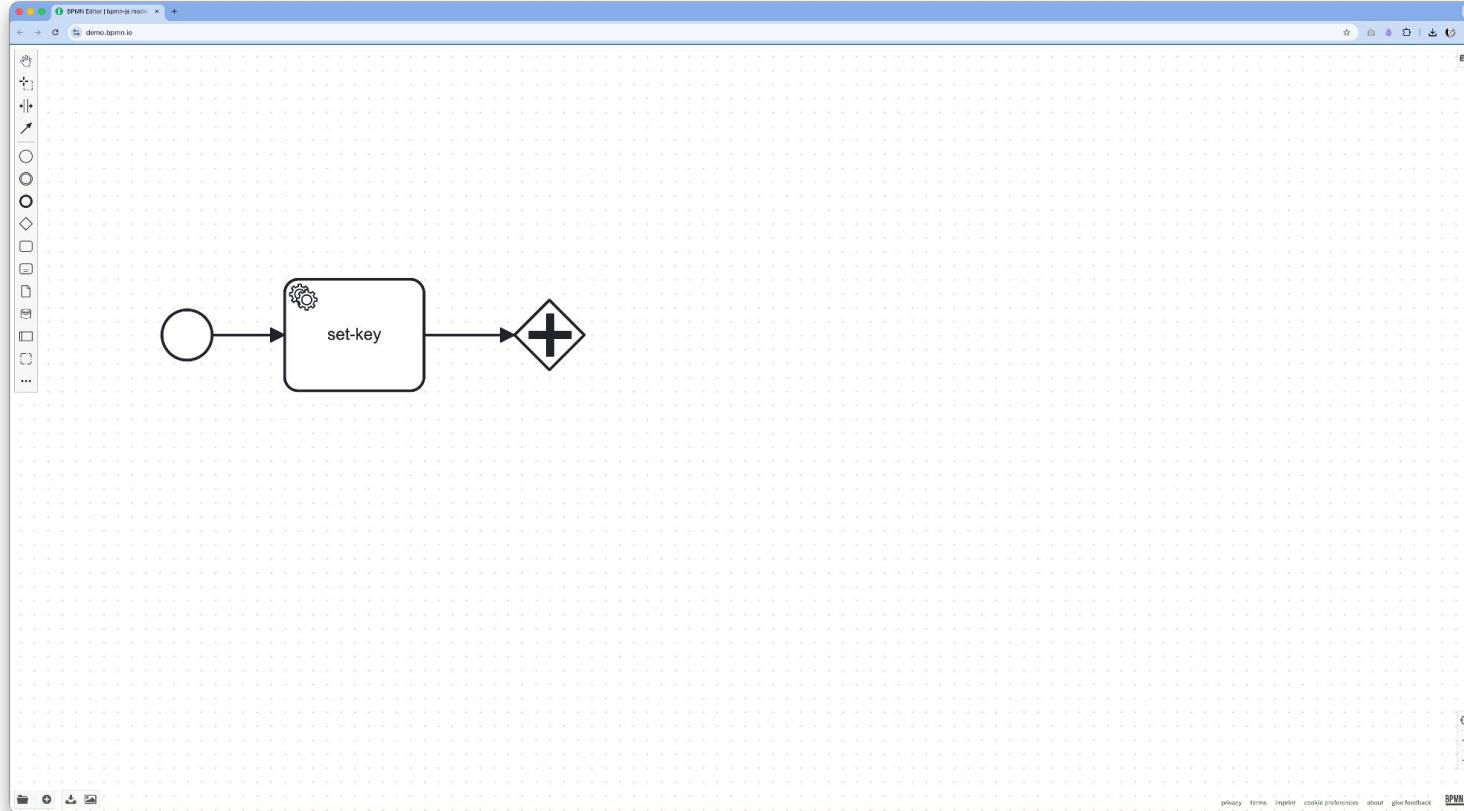
Fixing Fragmentation With Diagrams



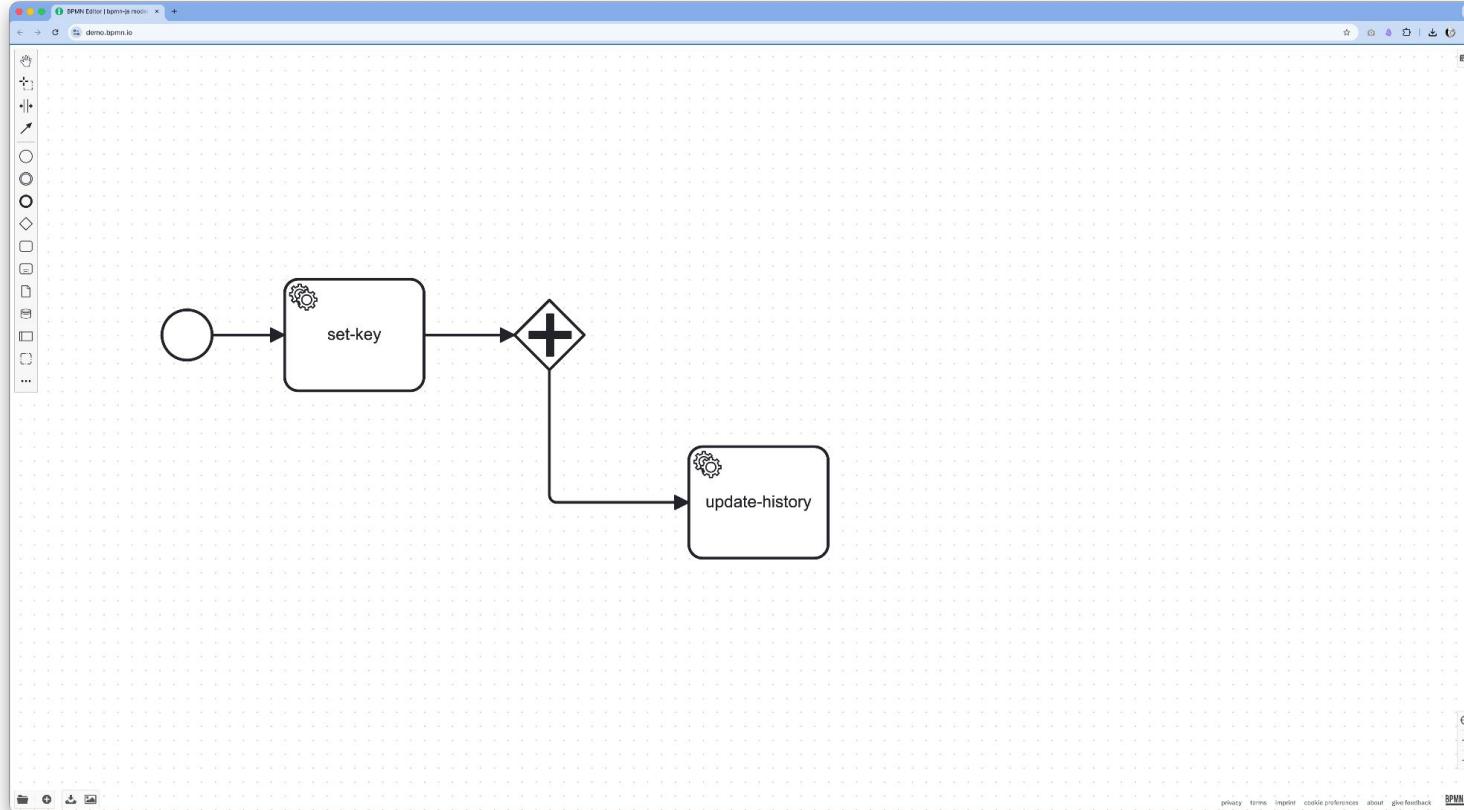
Fixing Fragmentation With Diagrams



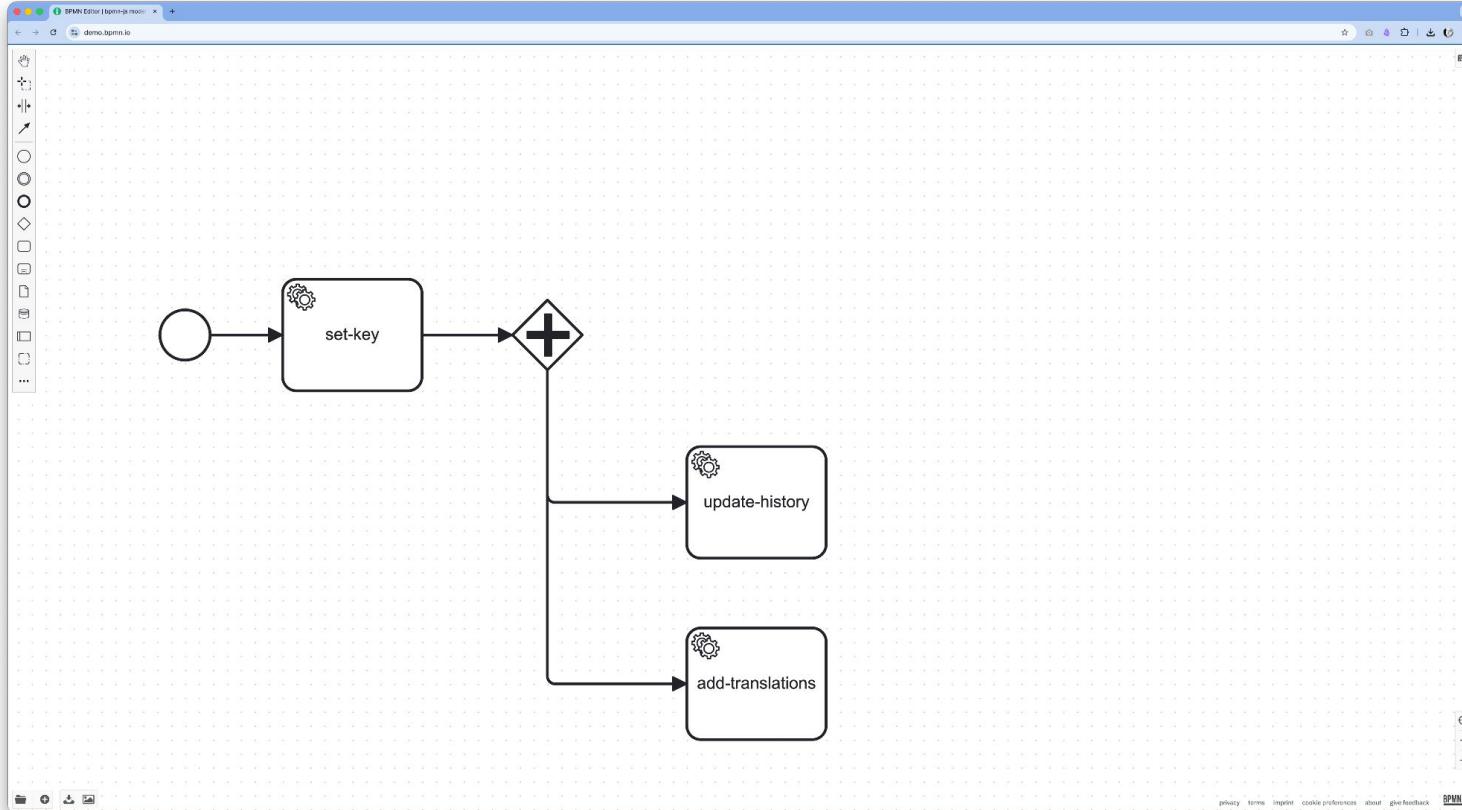
Fixing Fragmentation With Diagrams



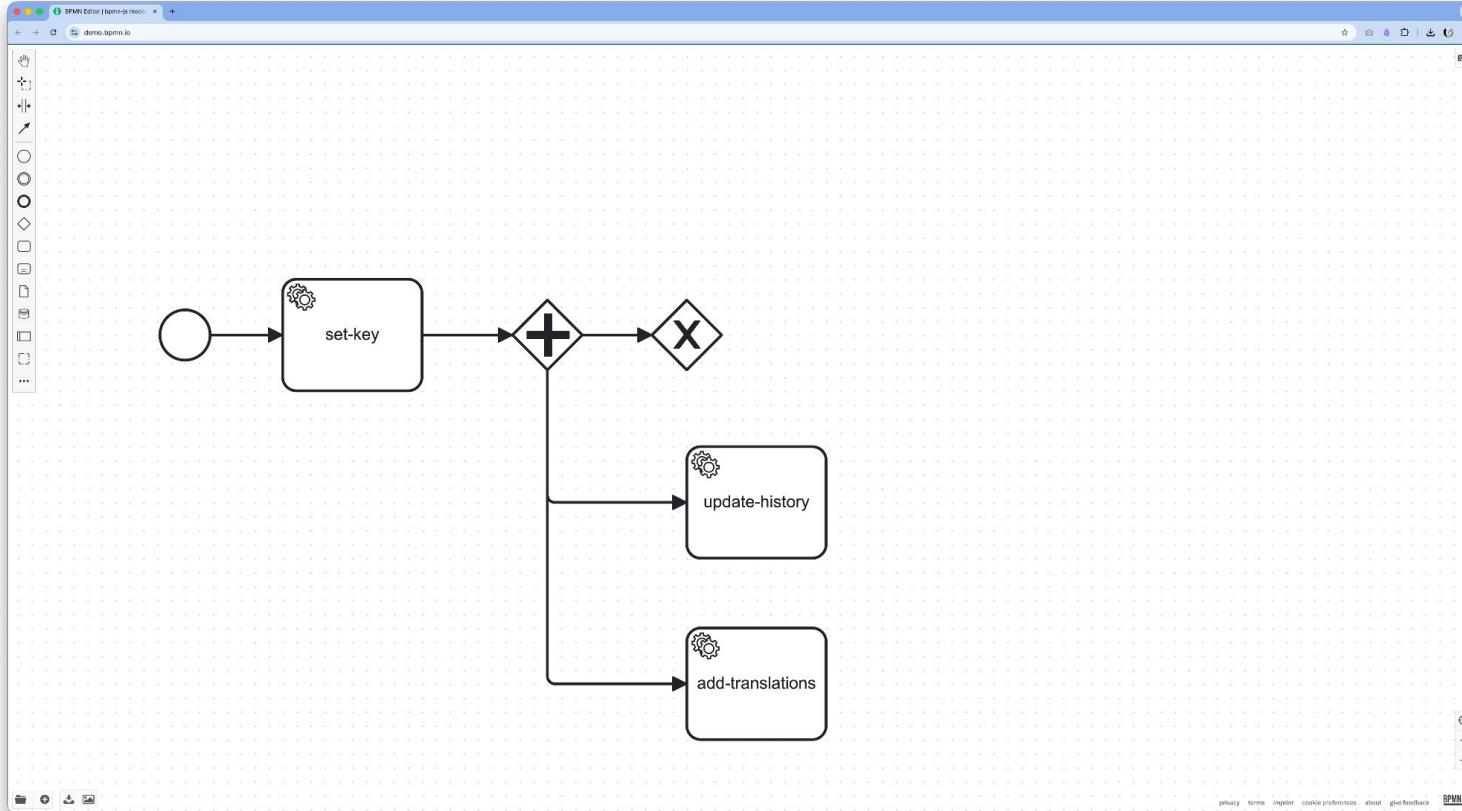
Fixing Fragmentation With Diagrams



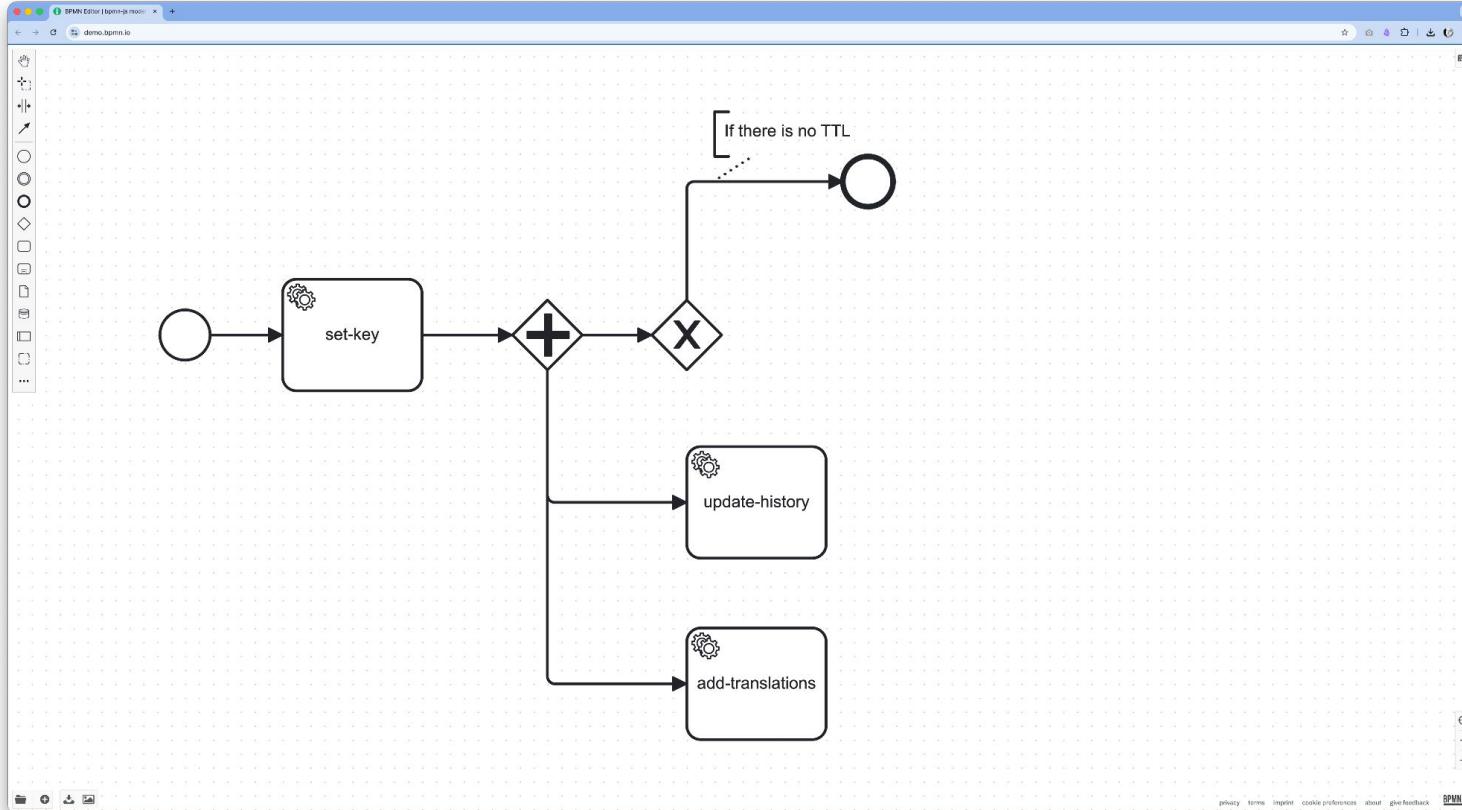
Fixing Fragmentation With Diagrams



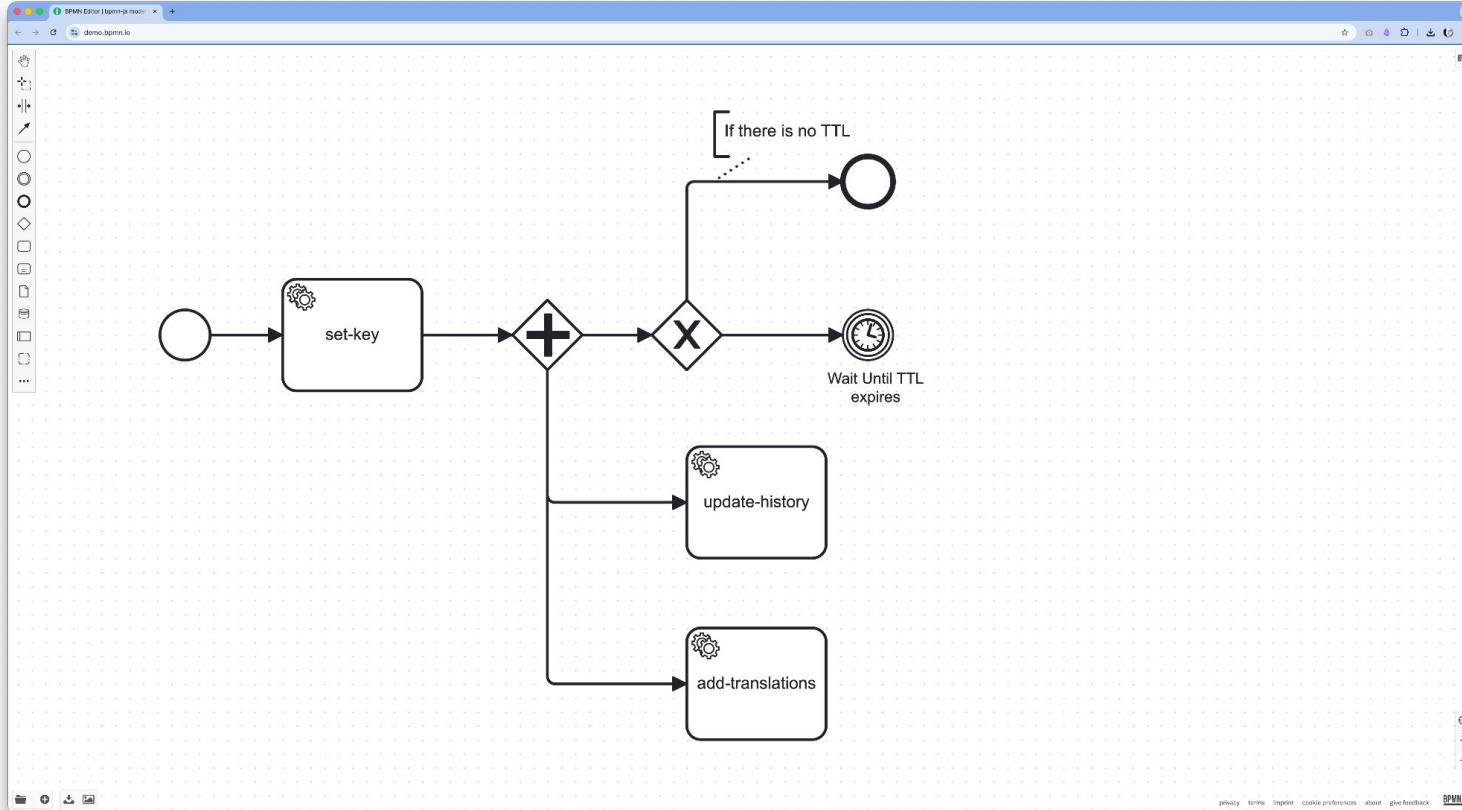
Fixing Fragmentation With Diagrams



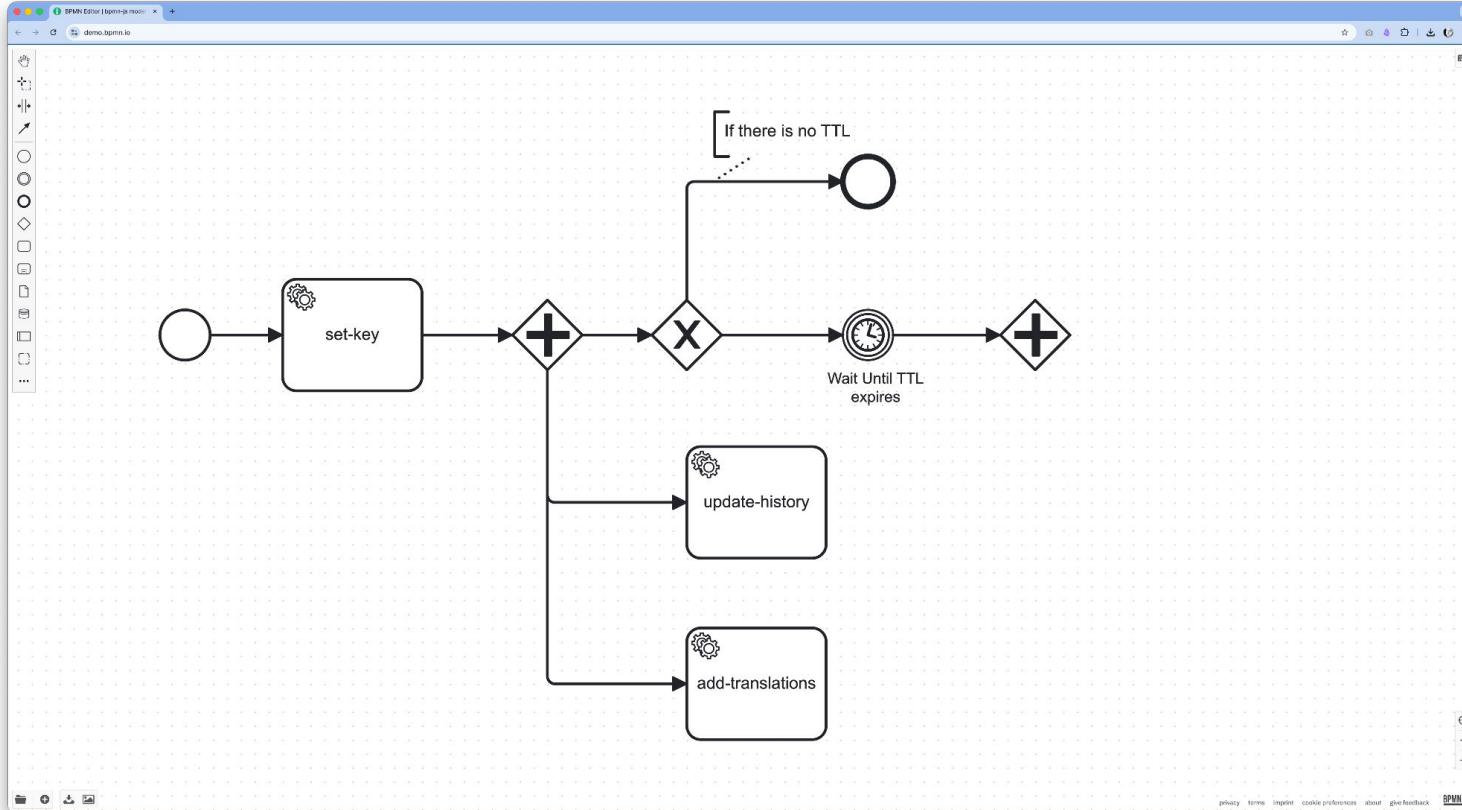
Fixing Fragmentation With Diagrams



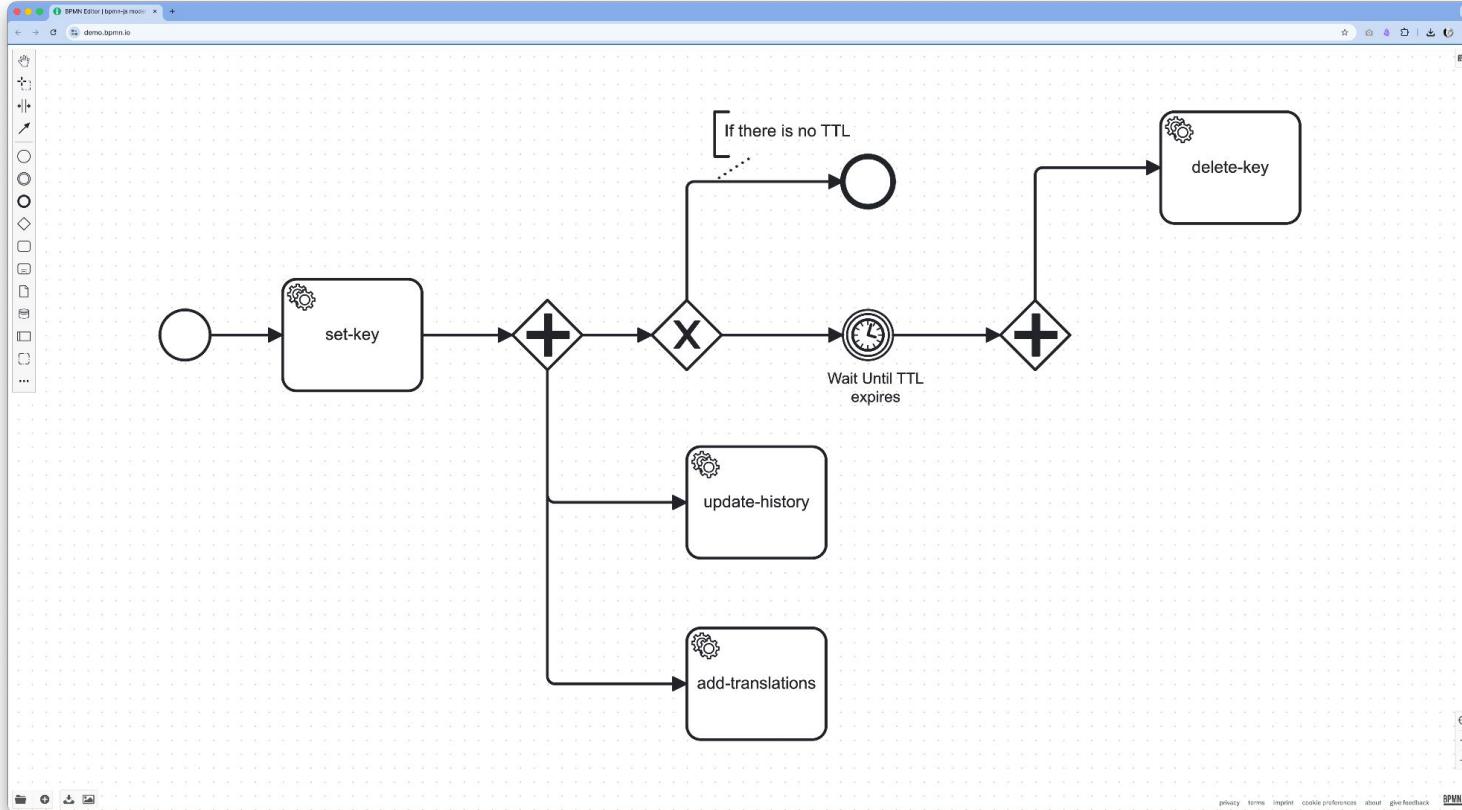
Fixing Fragmentation With Diagrams



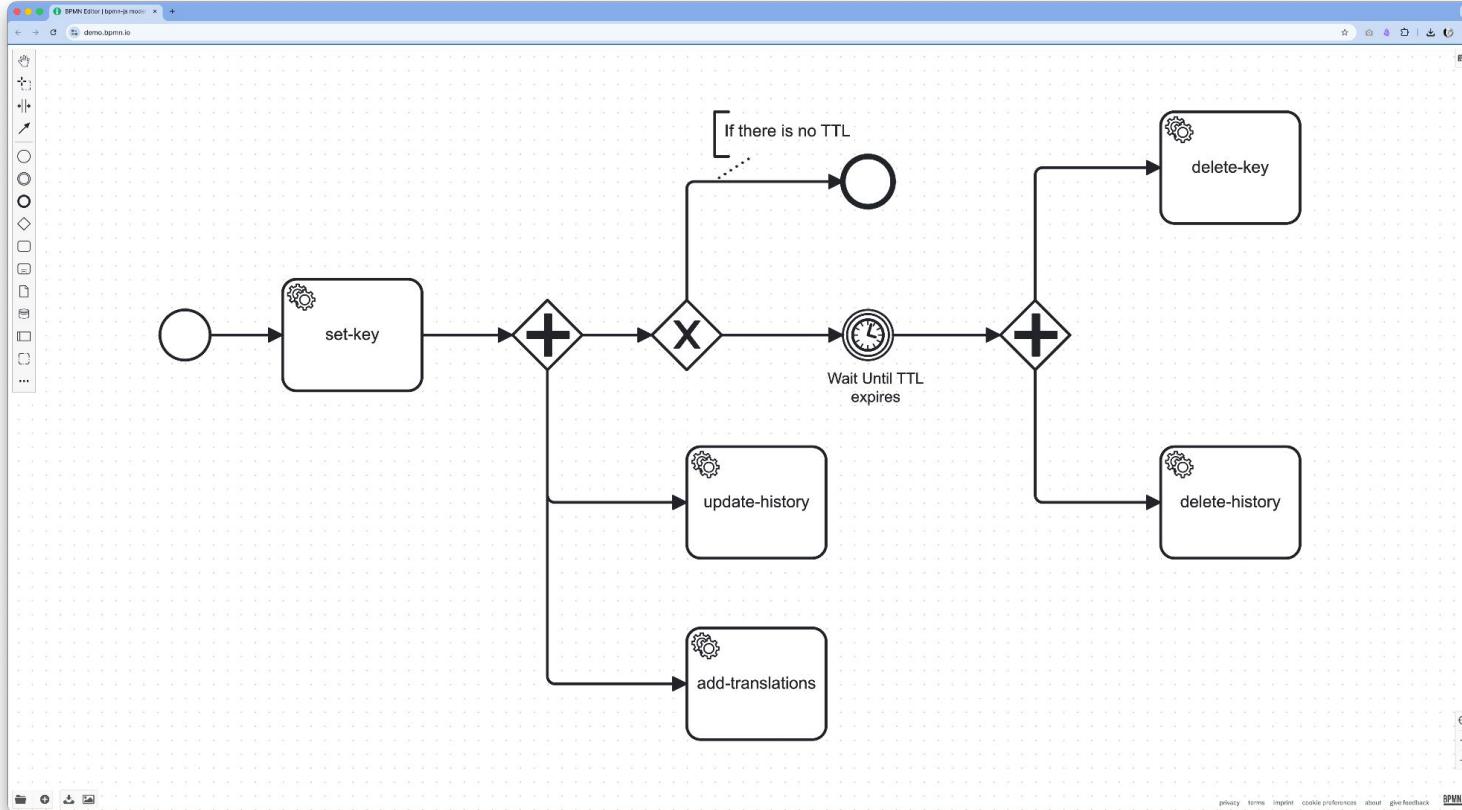
Fixing Fragmentation With Diagrams



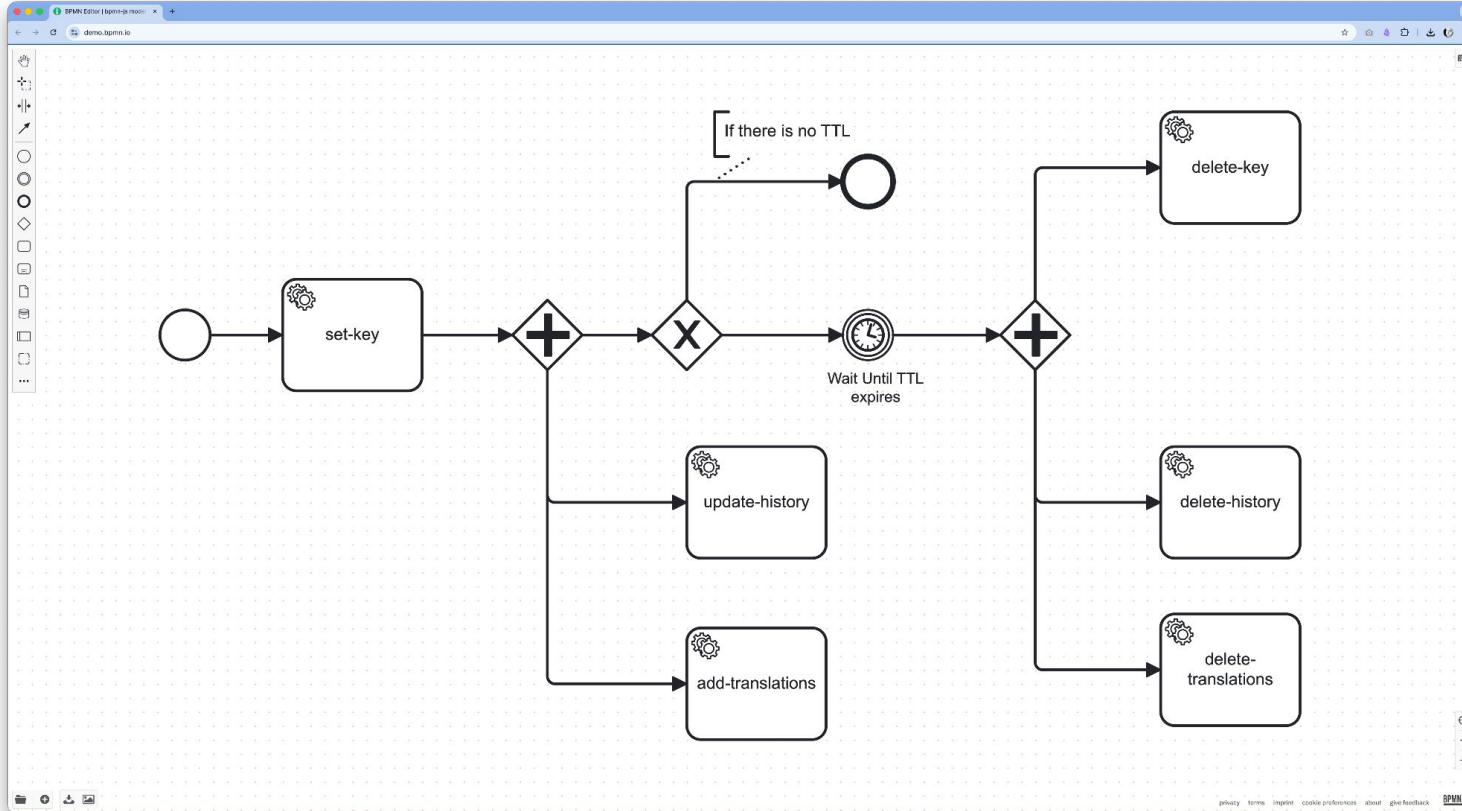
Fixing Fragmentation With Diagrams



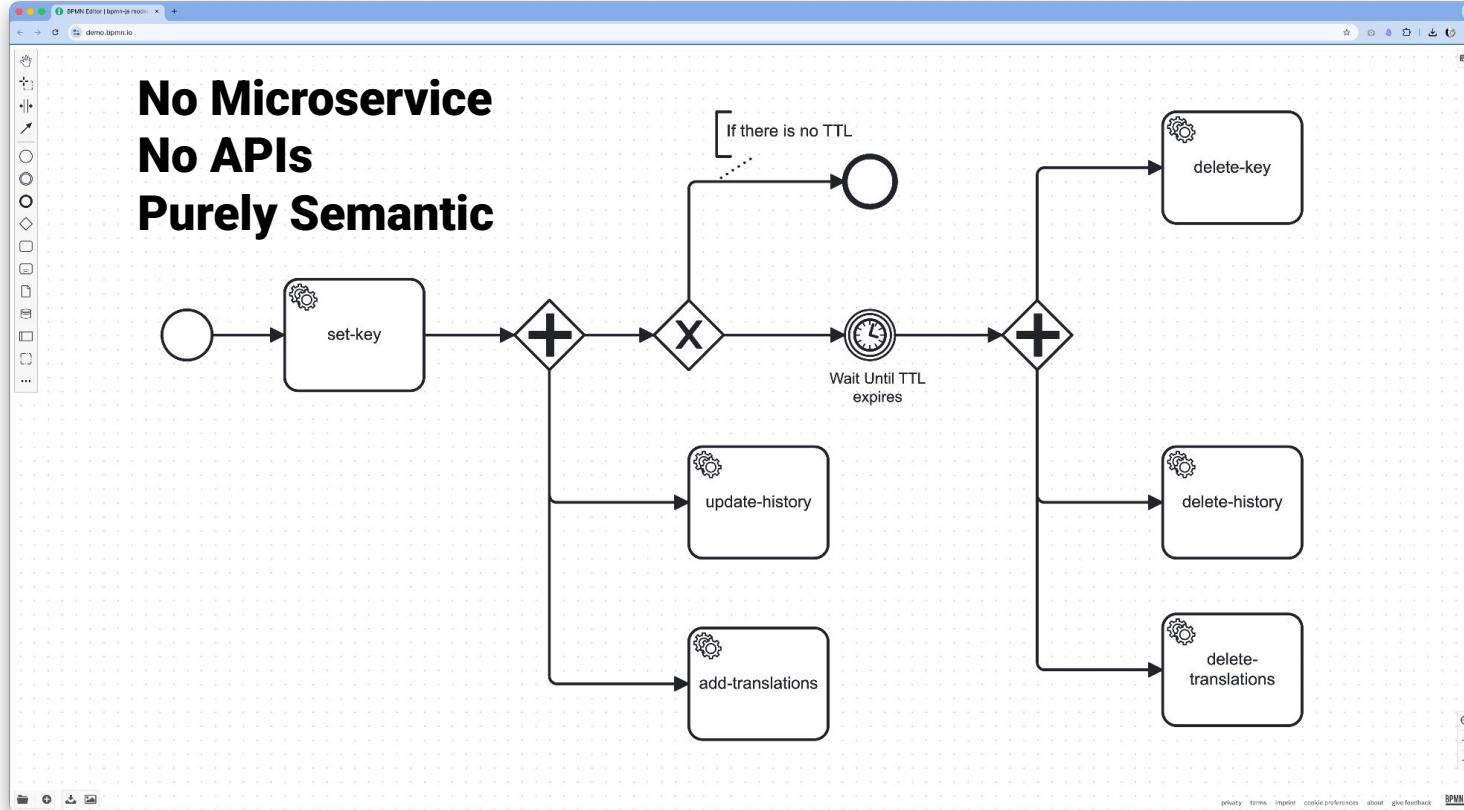
Fixing Fragmentation With Diagrams



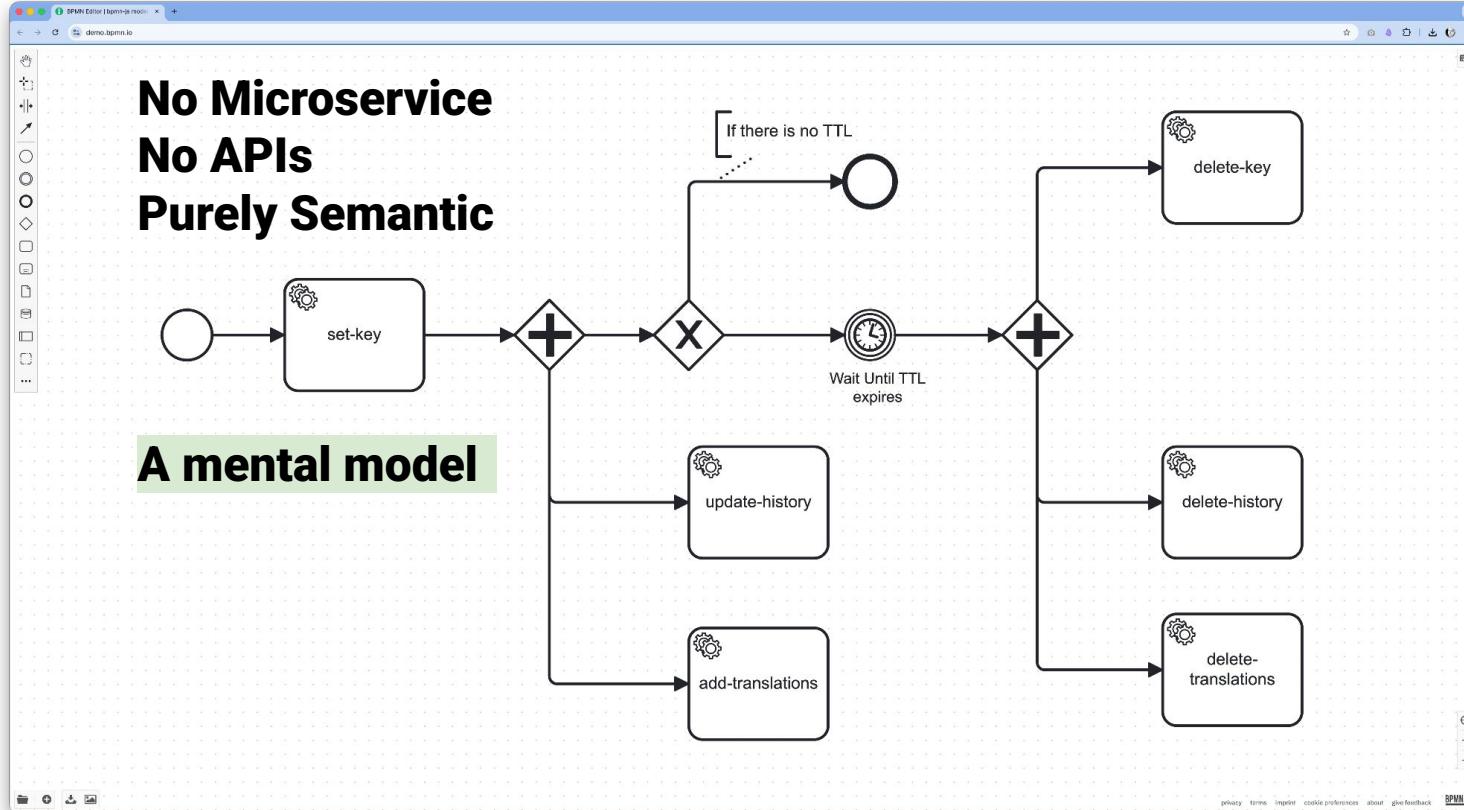
Fixing Fragmentation With Diagrams



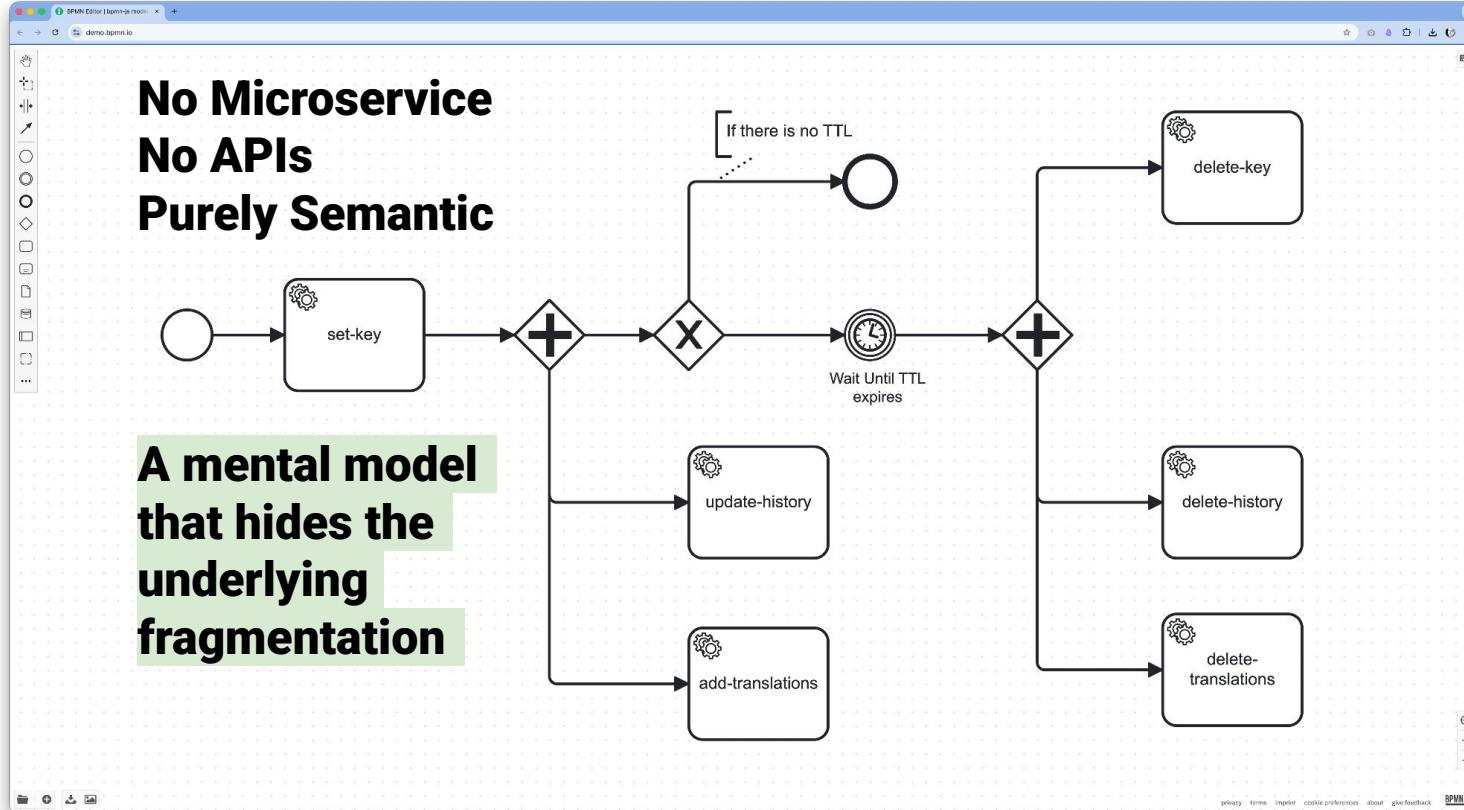
Fixing Fragmentation With Diagrams

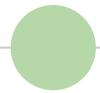


Fixing Fragmentation With Diagrams

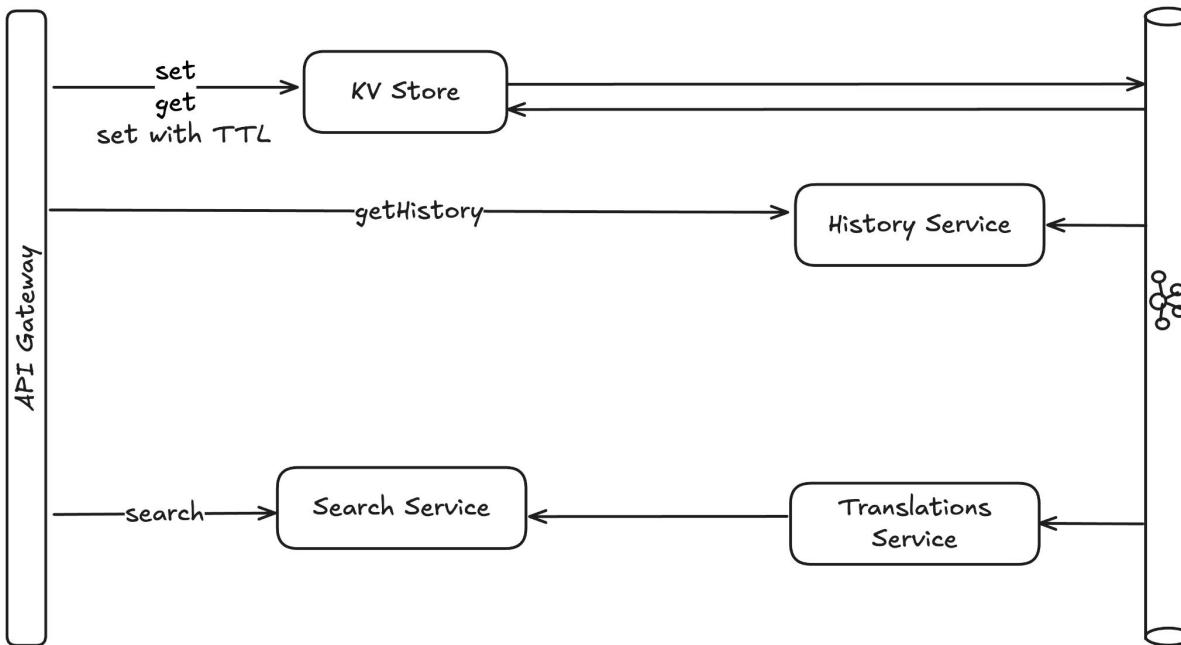


Fixing Fragmentation With Diagrams

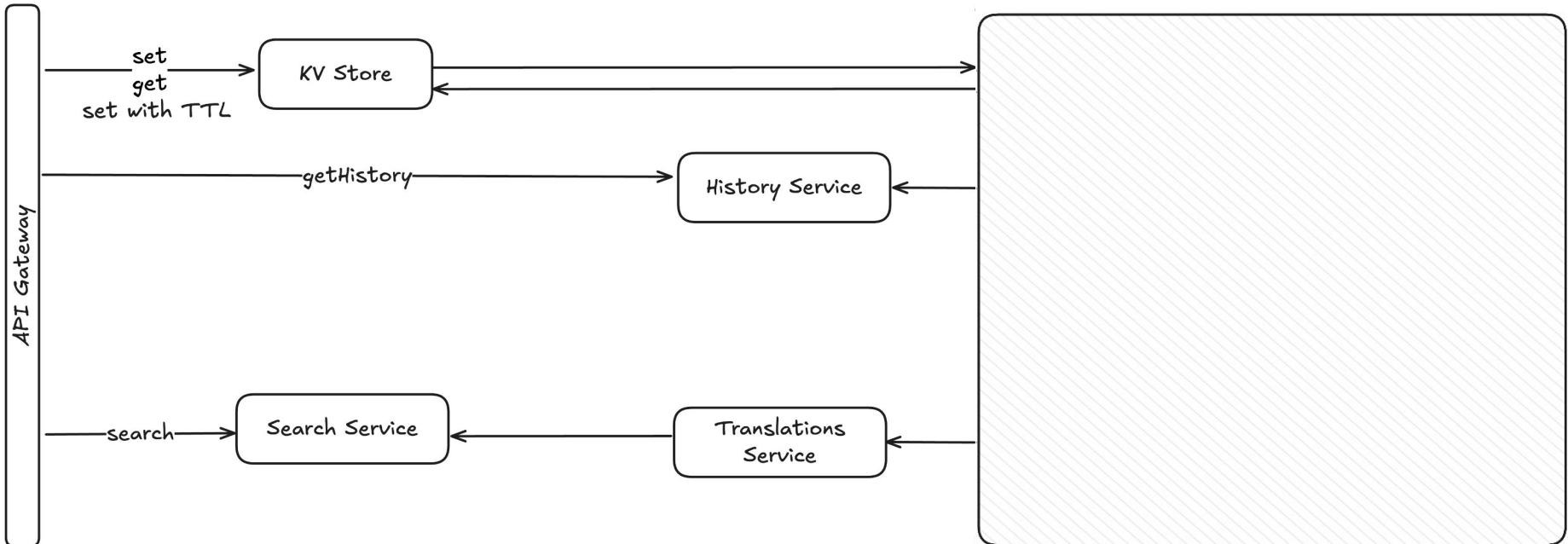




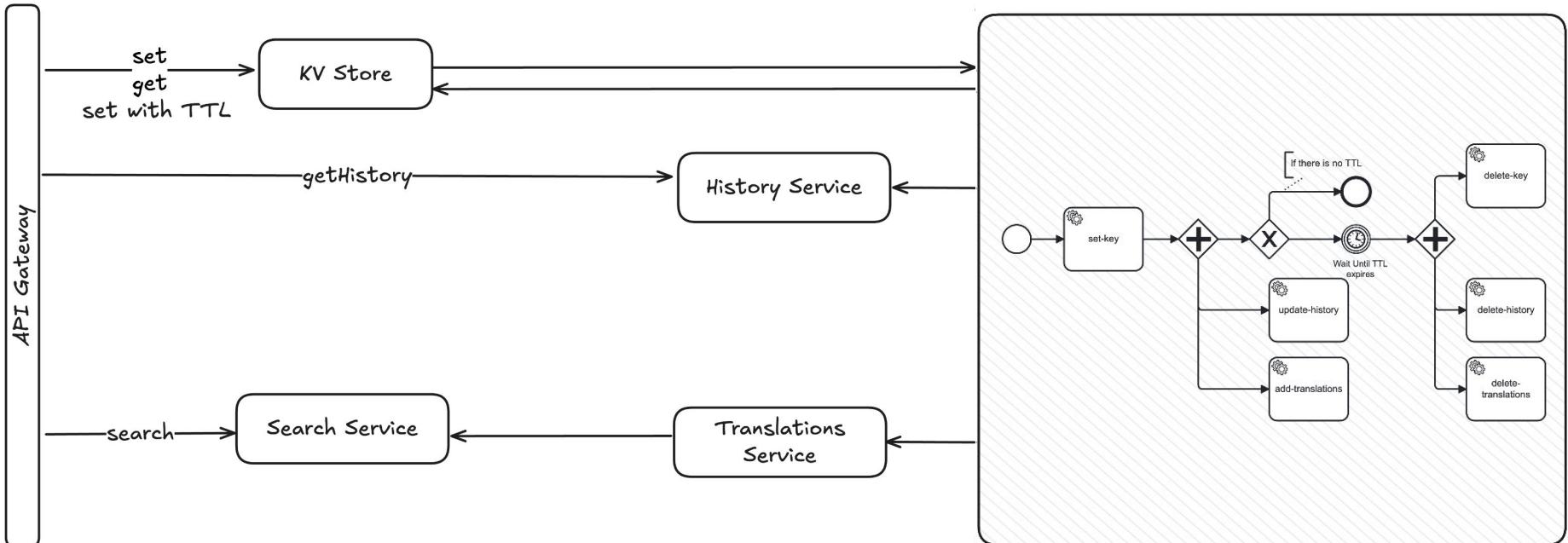
Fixing Fragmentation With Diagrams



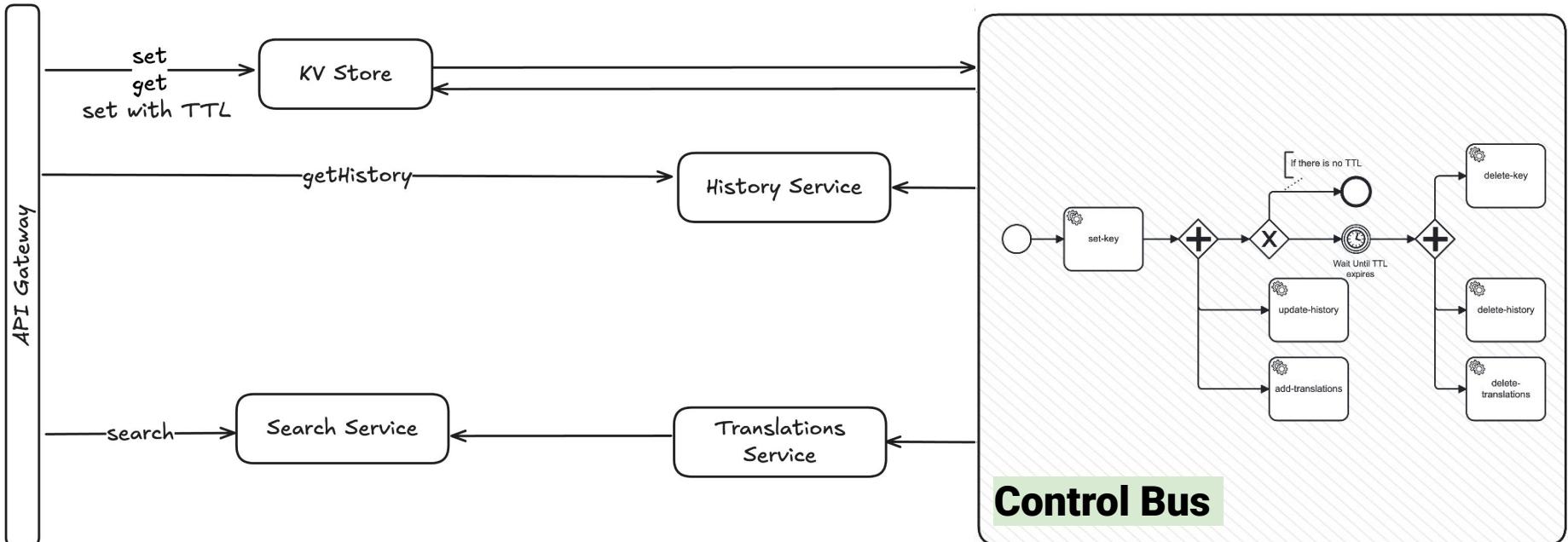
Fixing Fragmentation With Diagrams



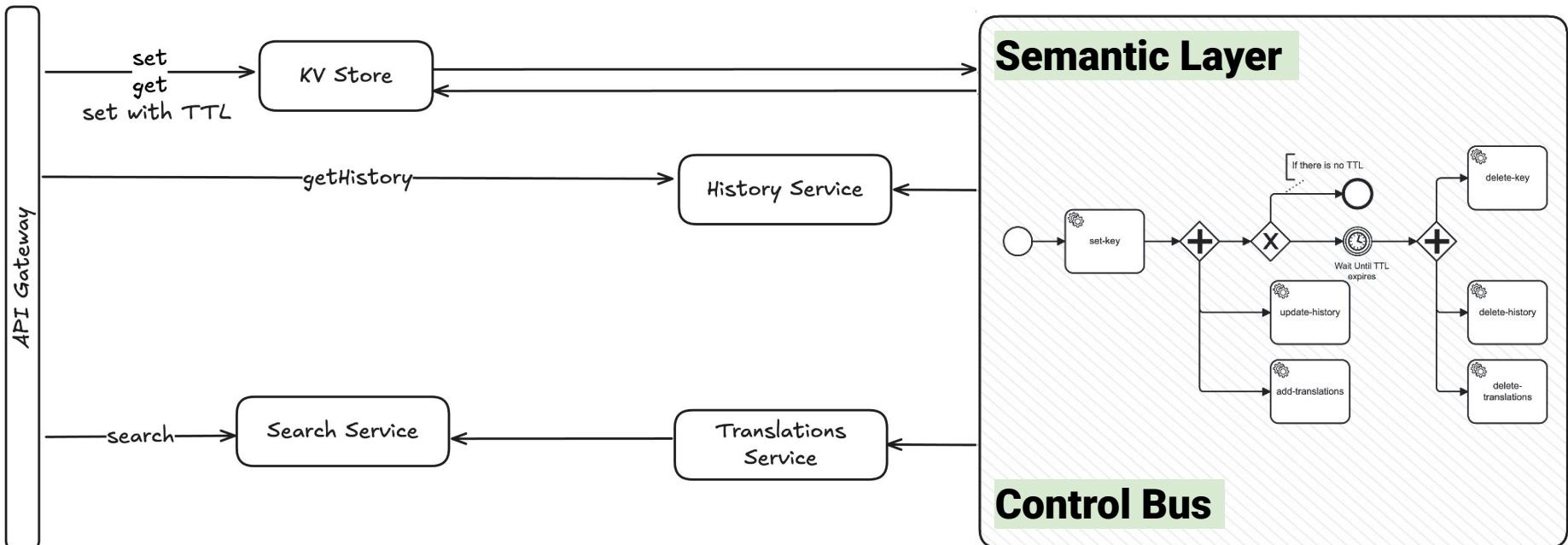
Fixing Fragmentation With Diagrams



Fixing Fragmentation With Diagrams



Fixing Fragmentation With Diagrams





Diagram

Declarative
Specification



Diagram → Orchestration

Between
Domain-Driven Microservices



Diagram → Orchestration → Glue

To solve the problems of
fragmentation



Diagram → Orchestration → Glue

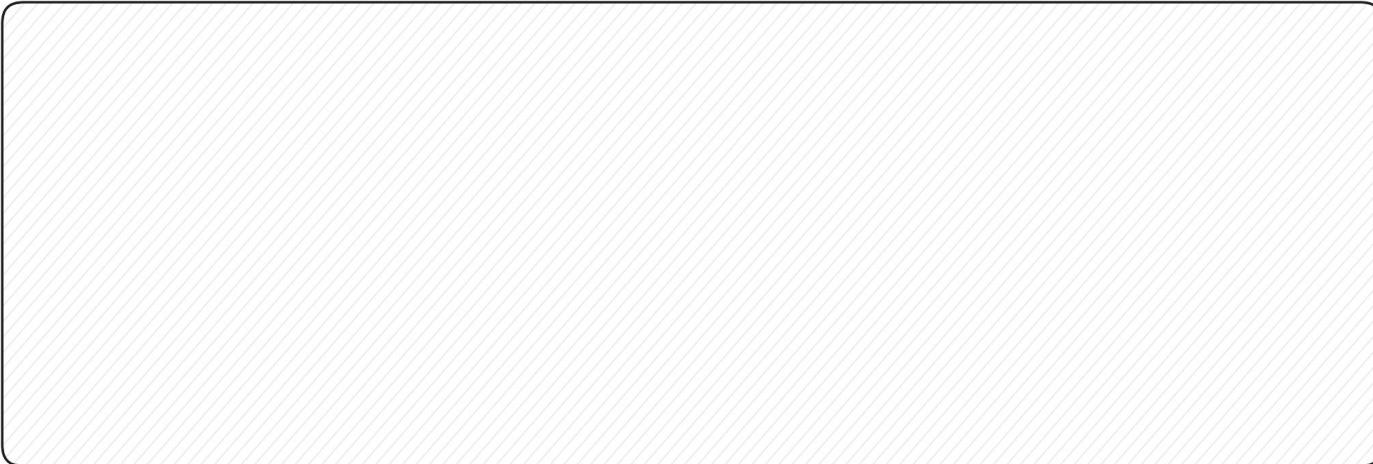


Diagram → Orchestration → Glue

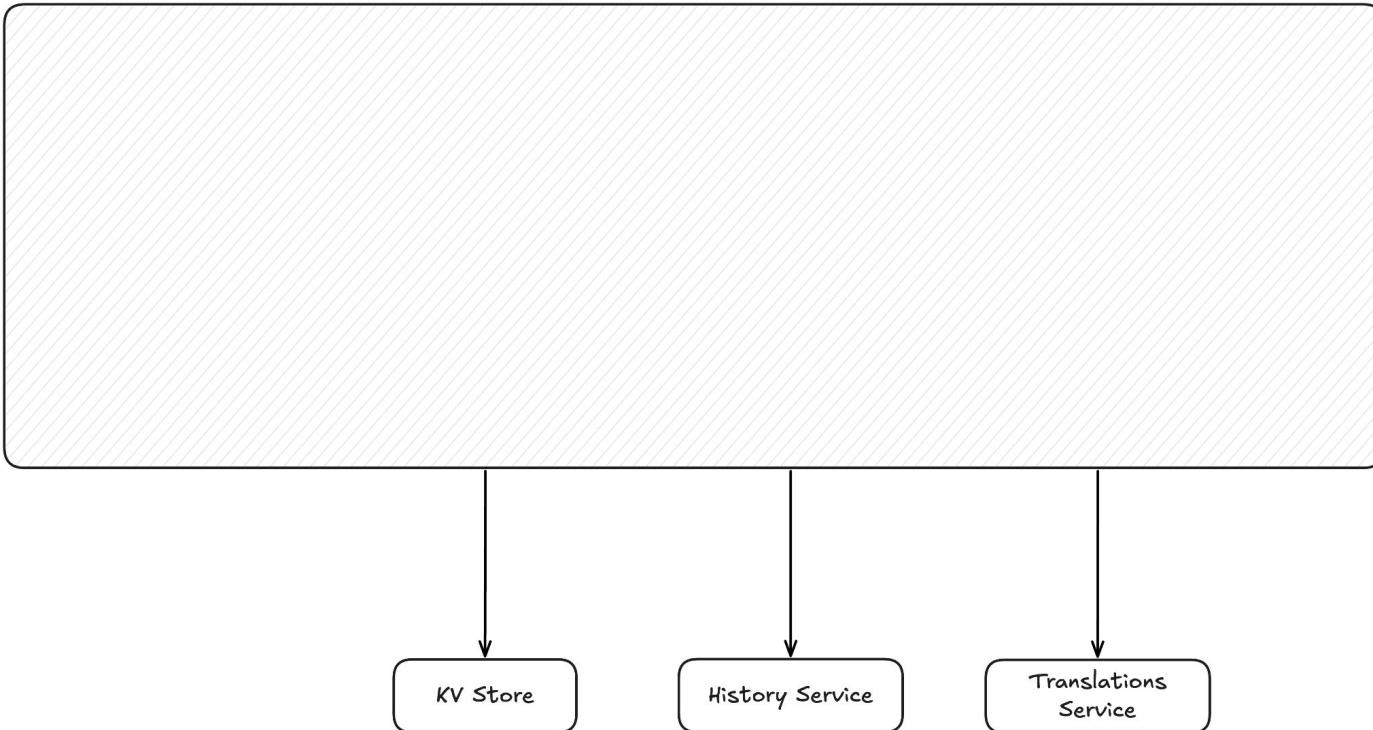


Diagram → Orchestration → Glue

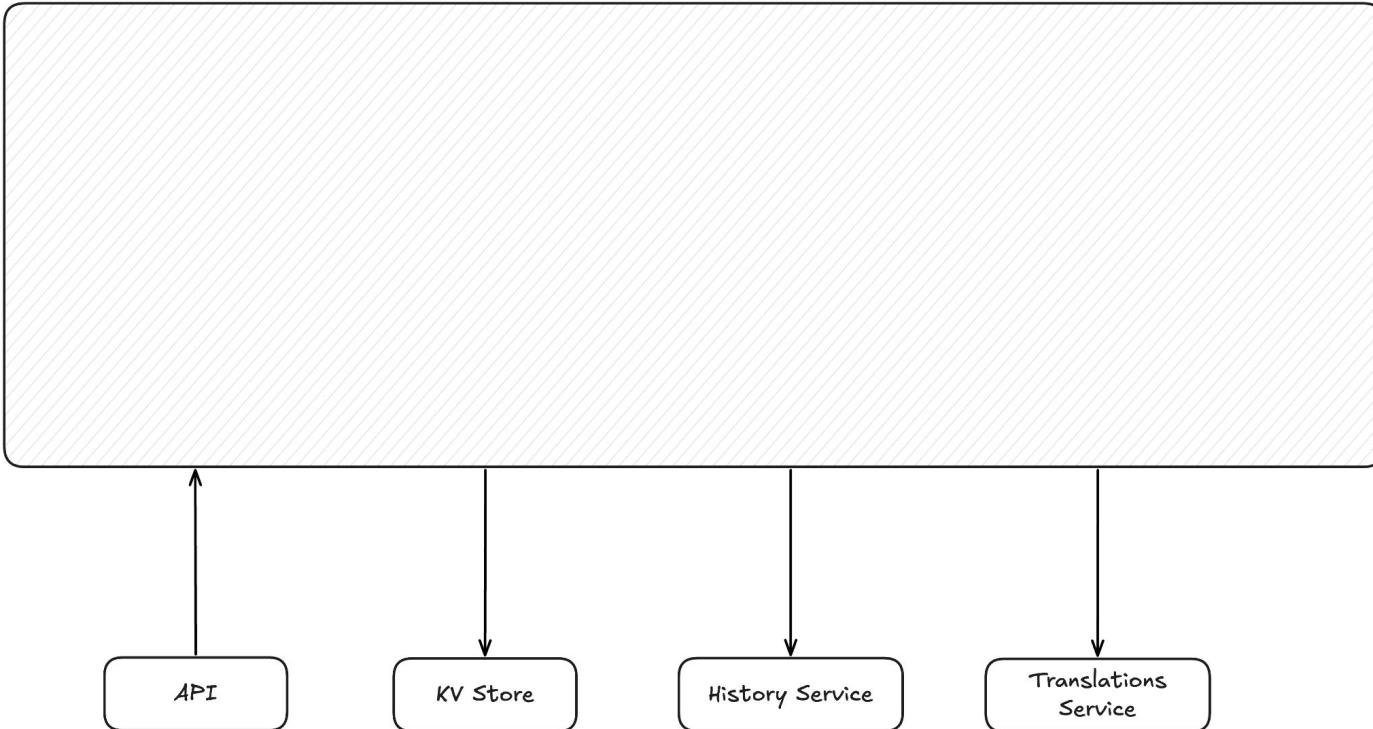


Diagram → Orchestration → Glue

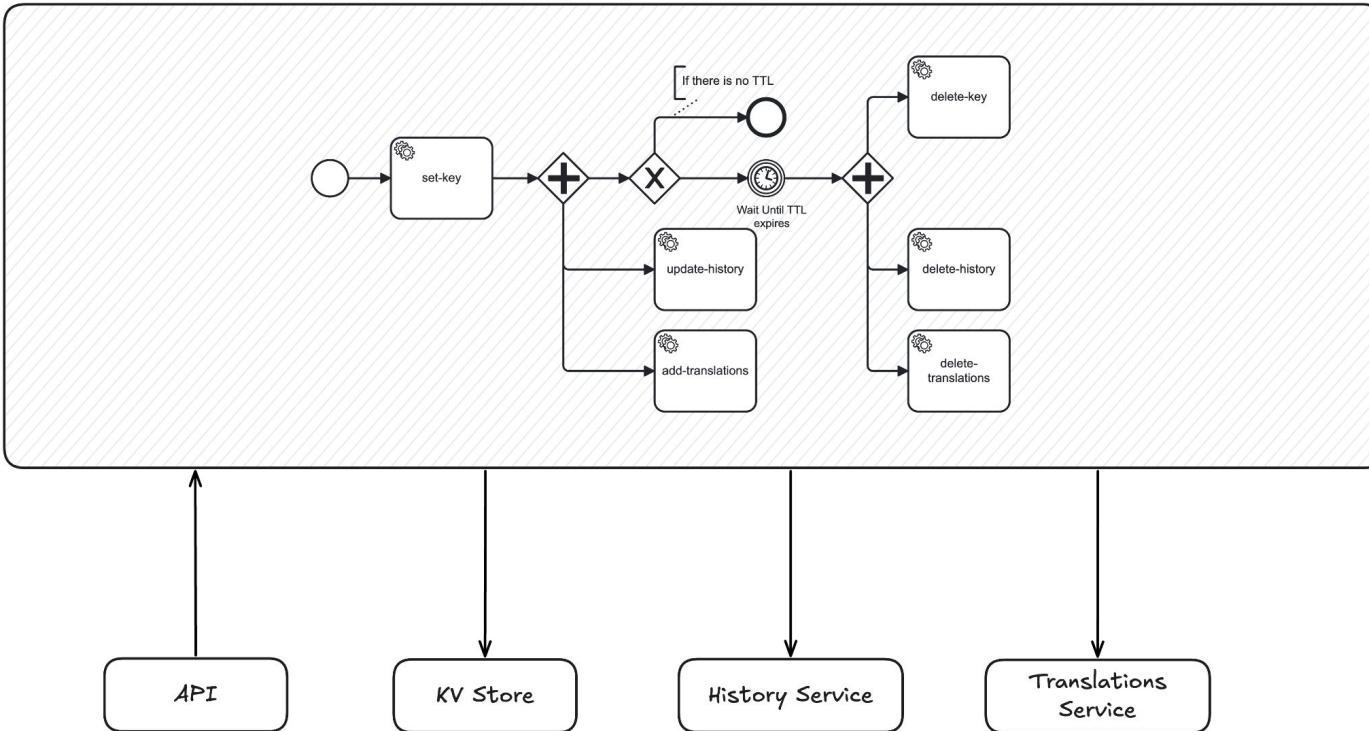


Diagram → Orchestration → Glue

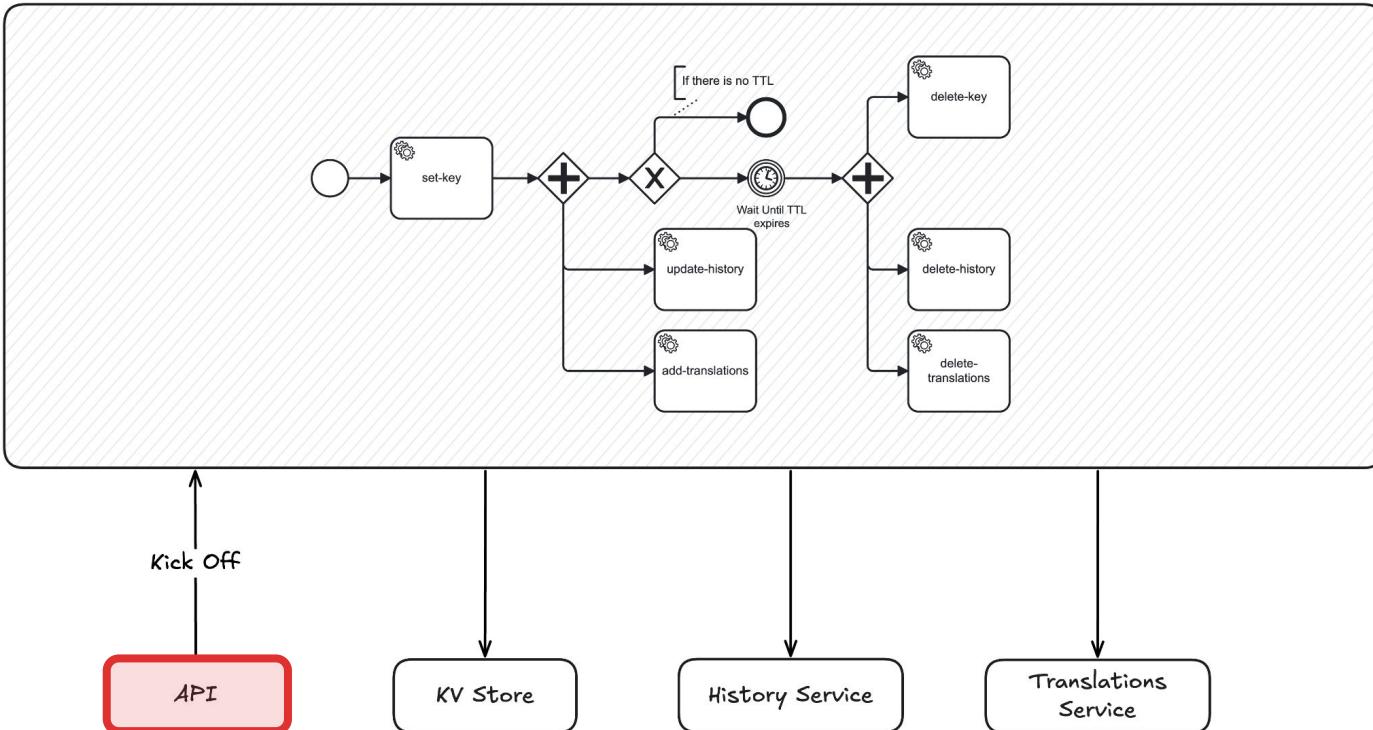


Diagram → Orchestration → Glue

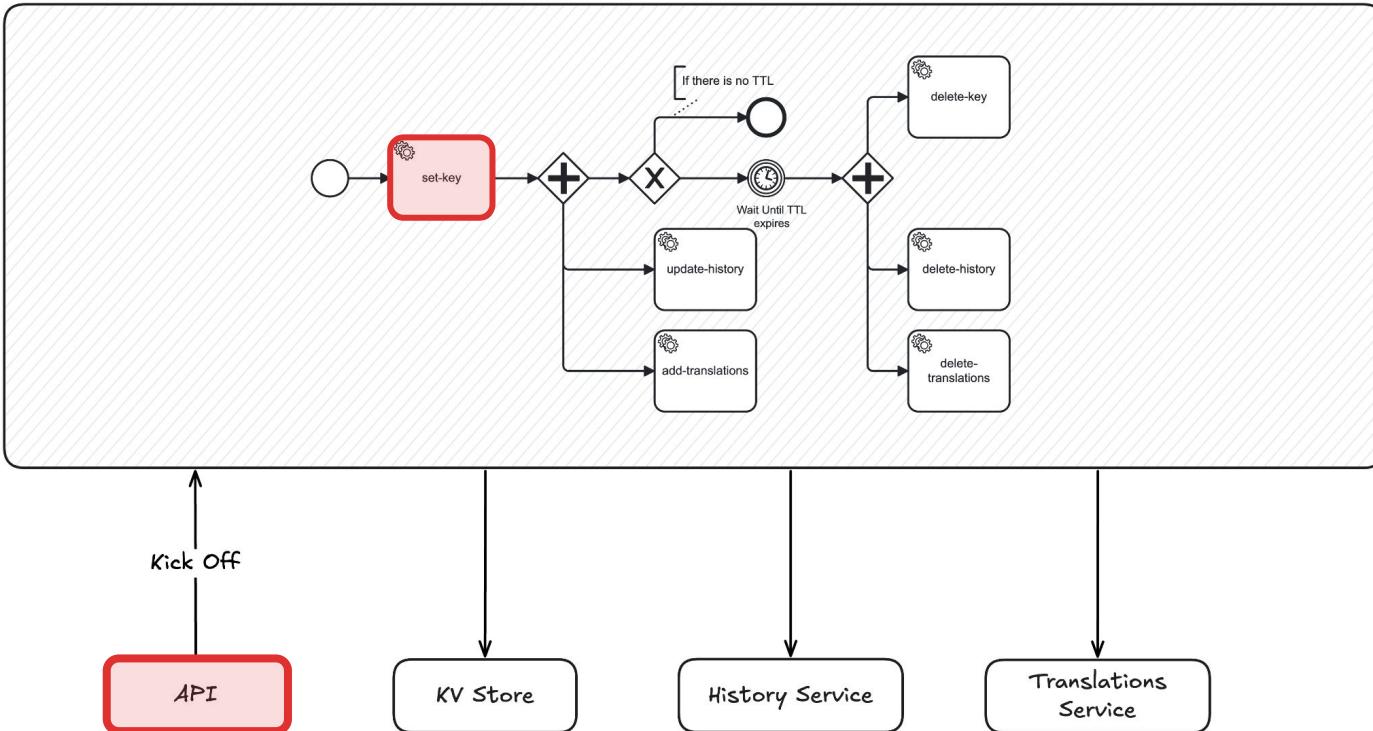


Diagram → Orchestration → Glue

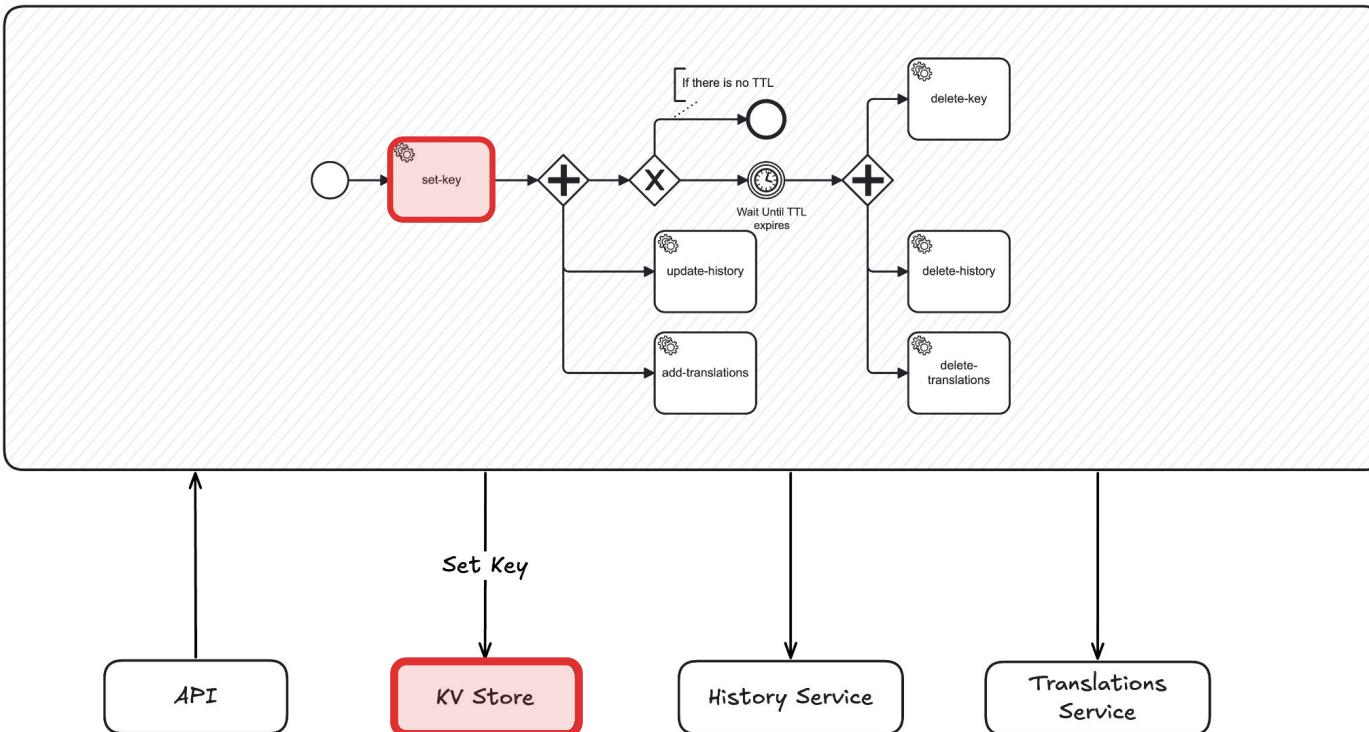


Diagram → Orchestration → Glue

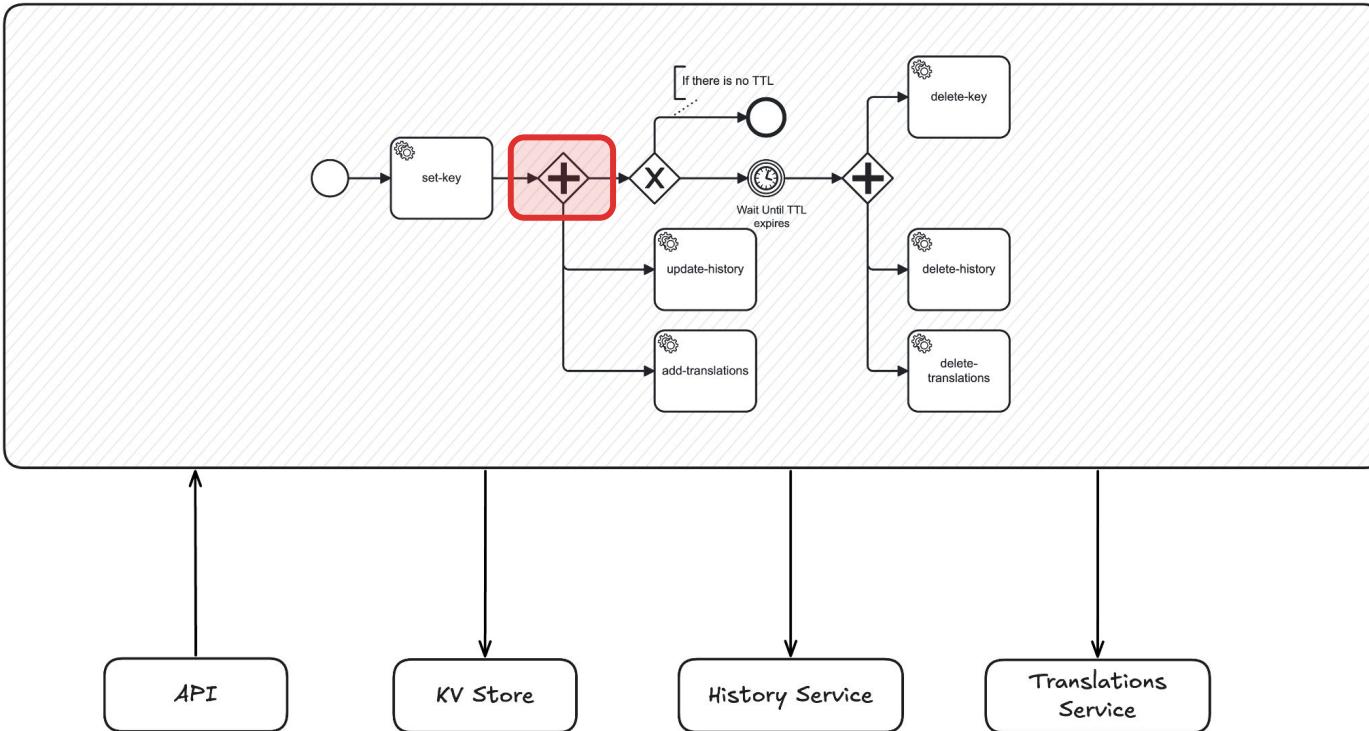


Diagram → Orchestration → Glue

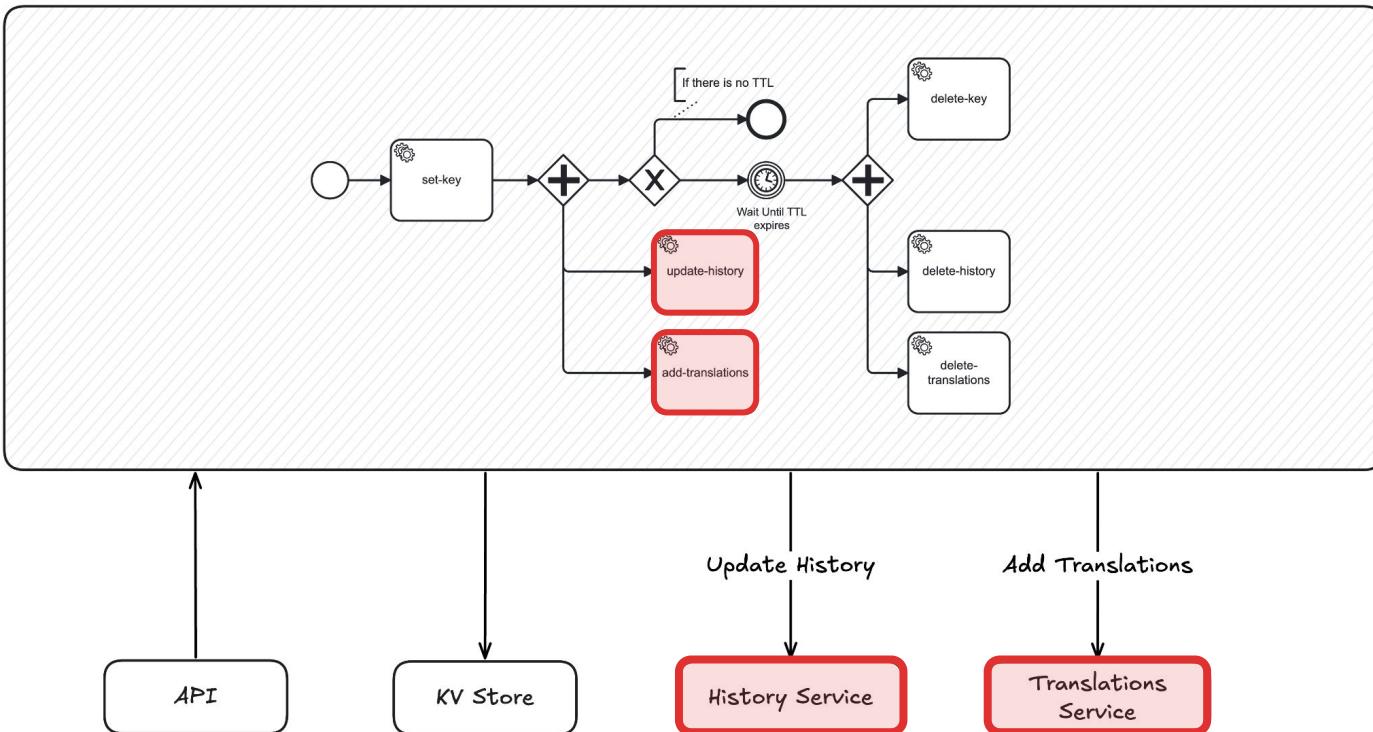


Diagram → Orchestration → Glue

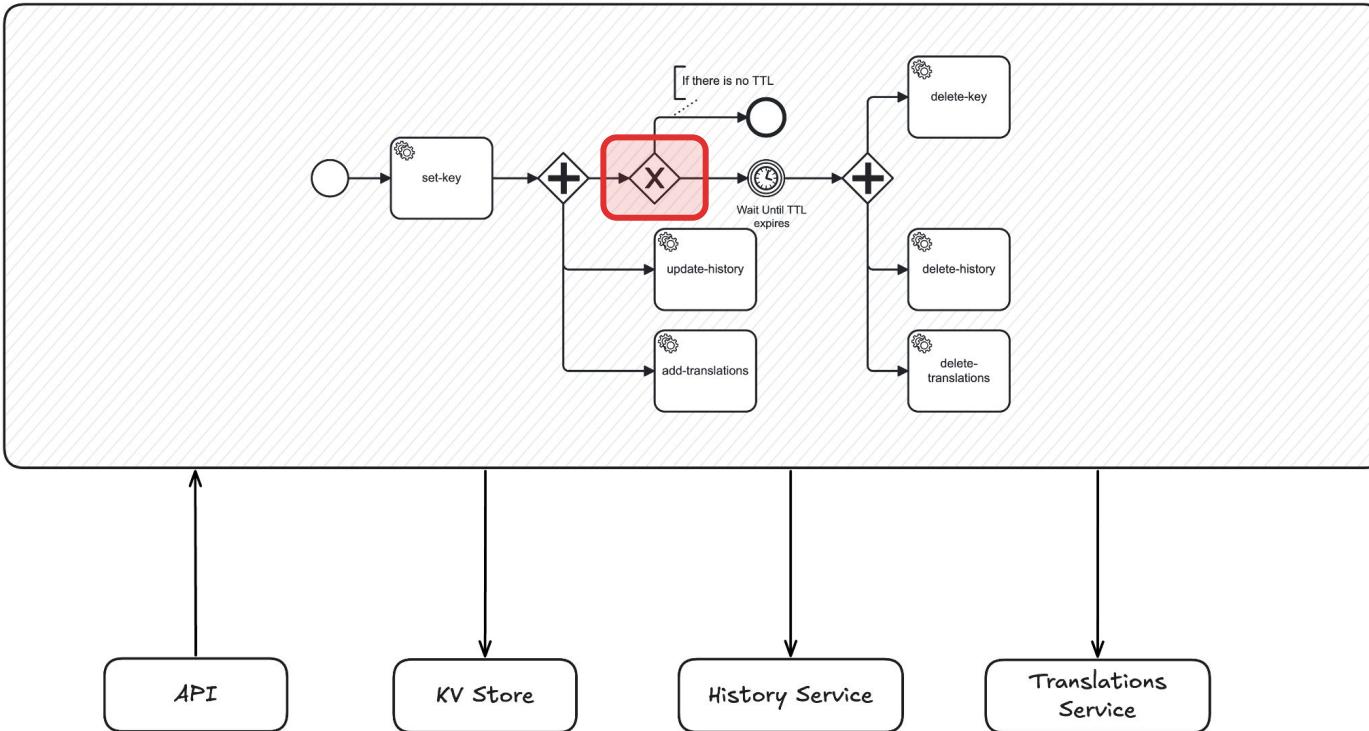


Diagram → Orchestration → Glue

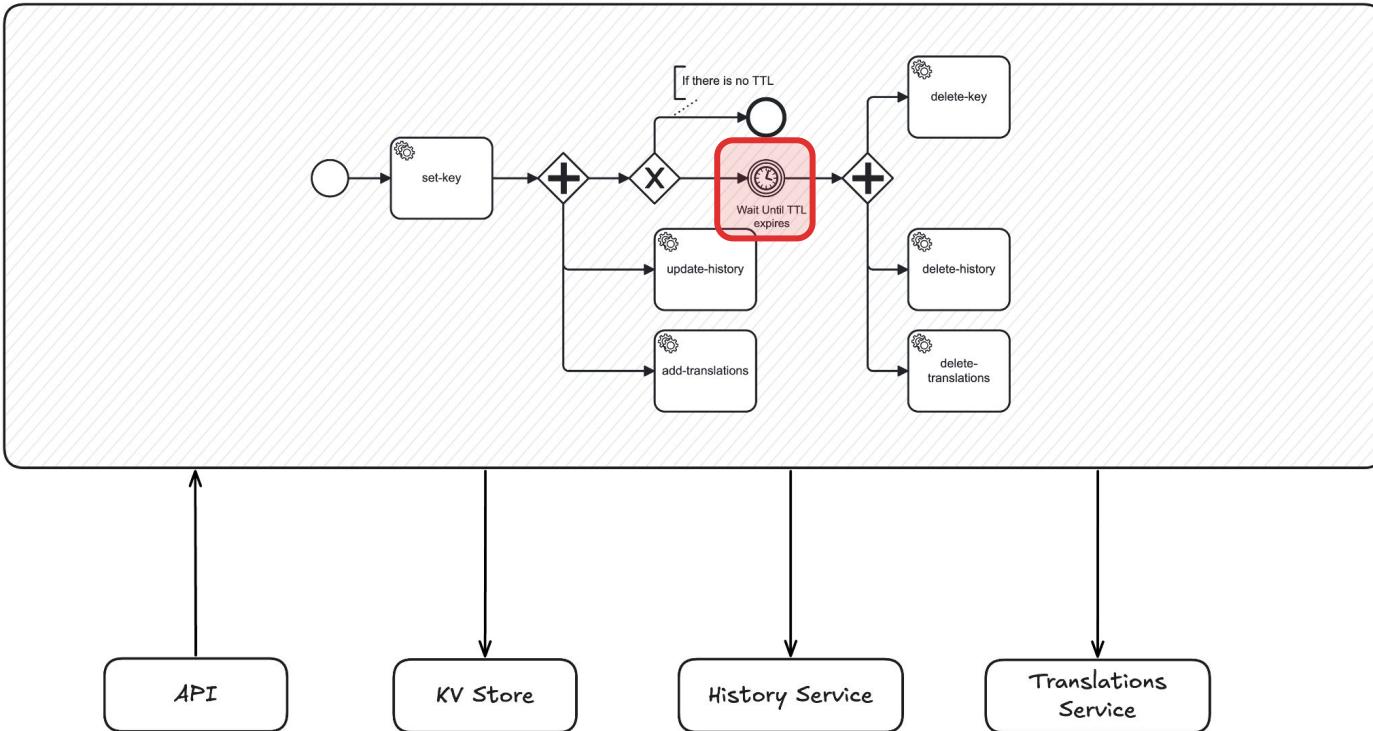


Diagram → Orchestration → Glue

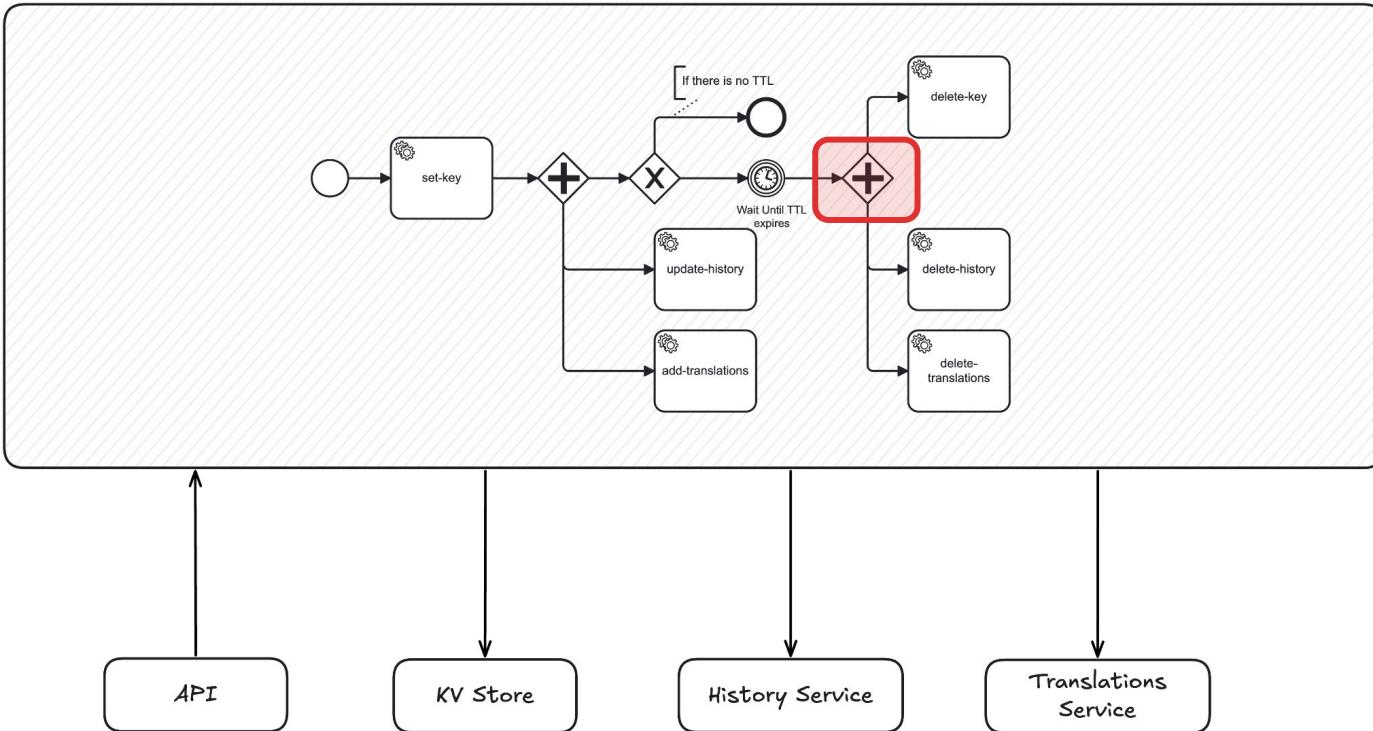


Diagram → Orchestration → Glue

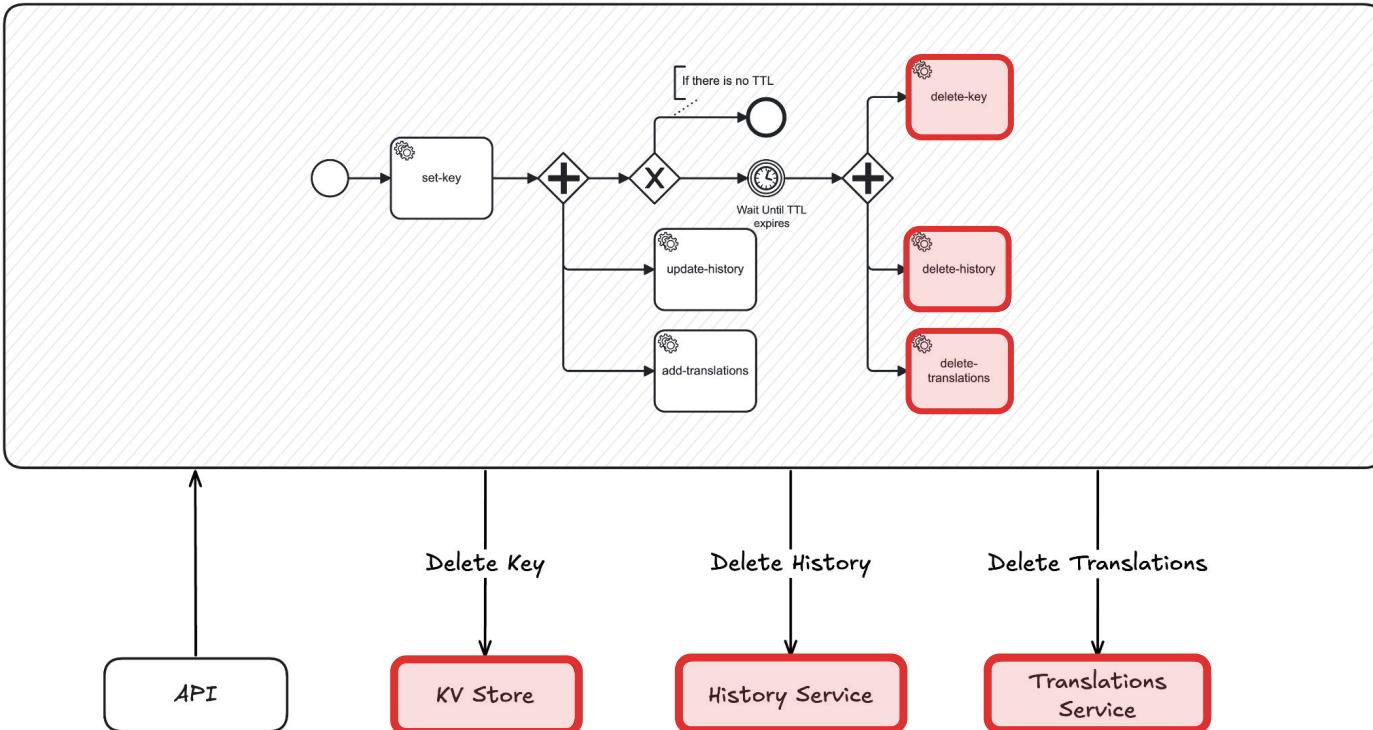




Diagram → Orchestration → Glue

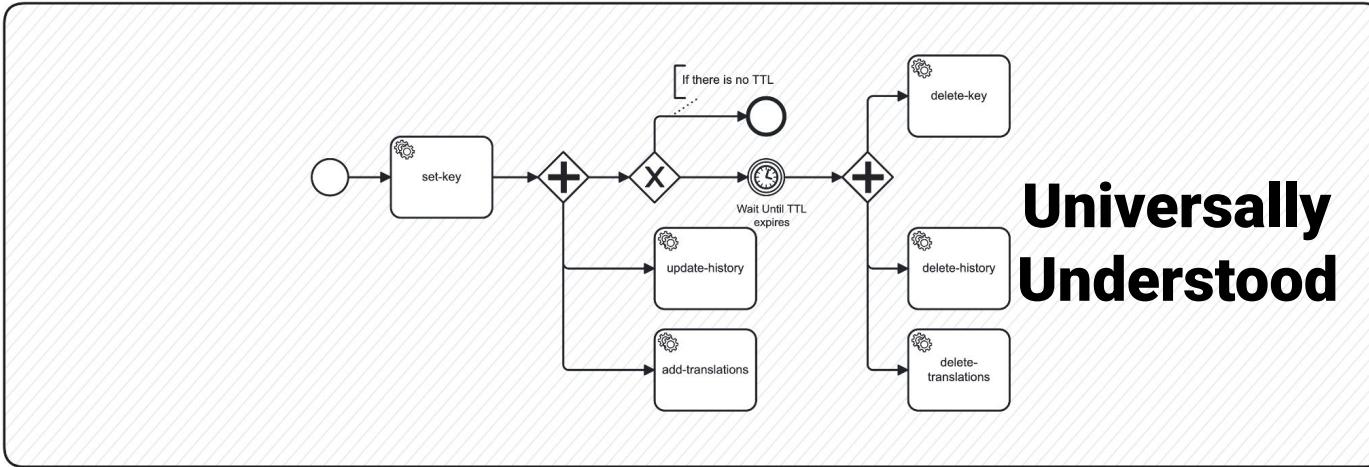




Diagram → Orchestration → Glue

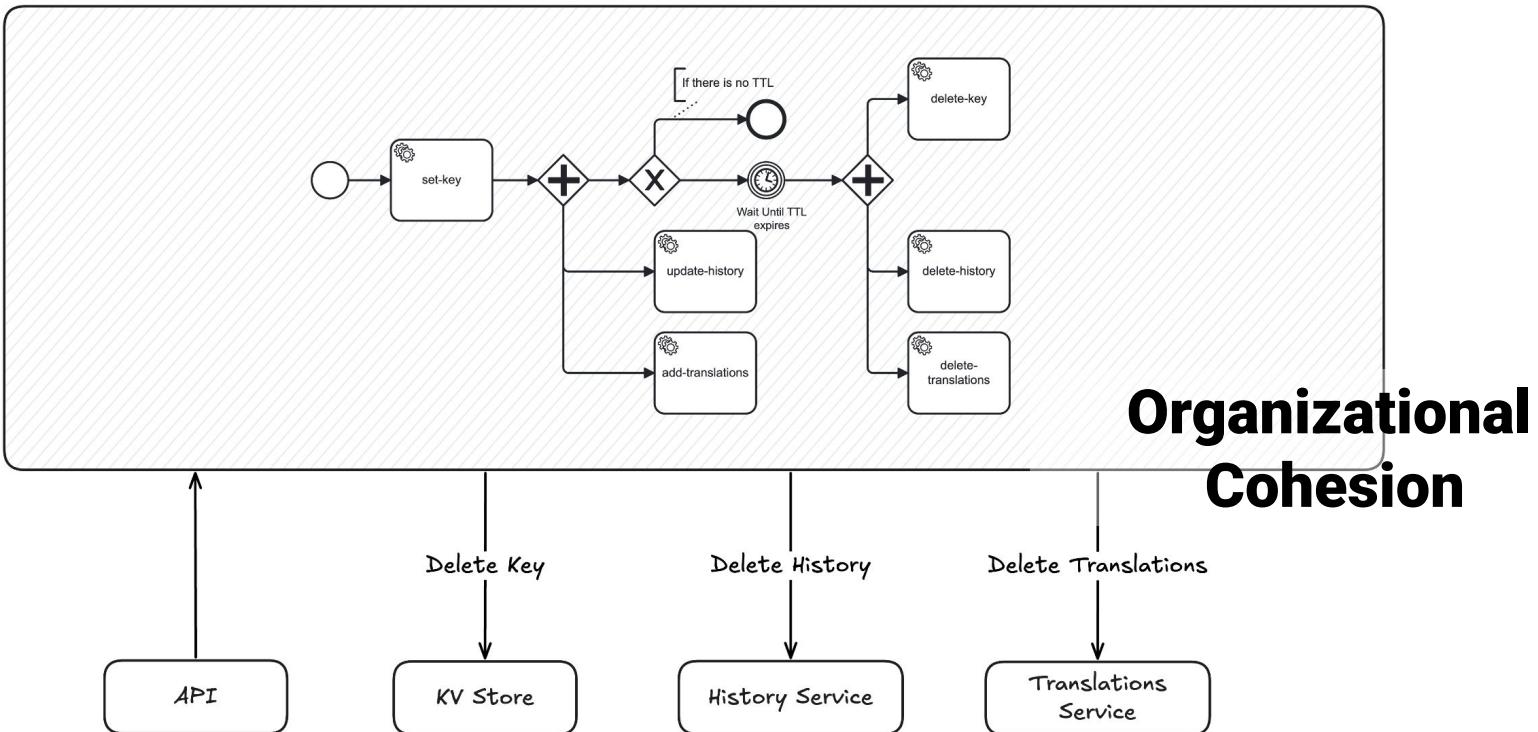
KV Store

History Service

Translations
Service

**Dedicated
Microservices**

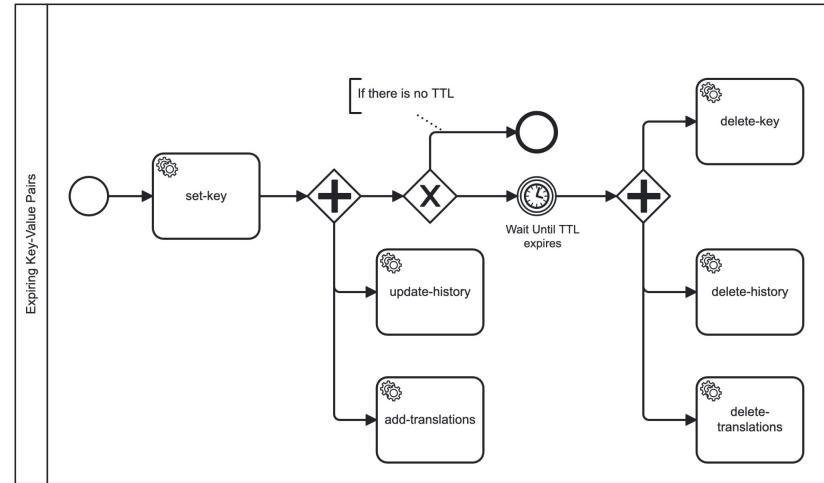
Diagram → Orchestration → Glue





What is BPMN?

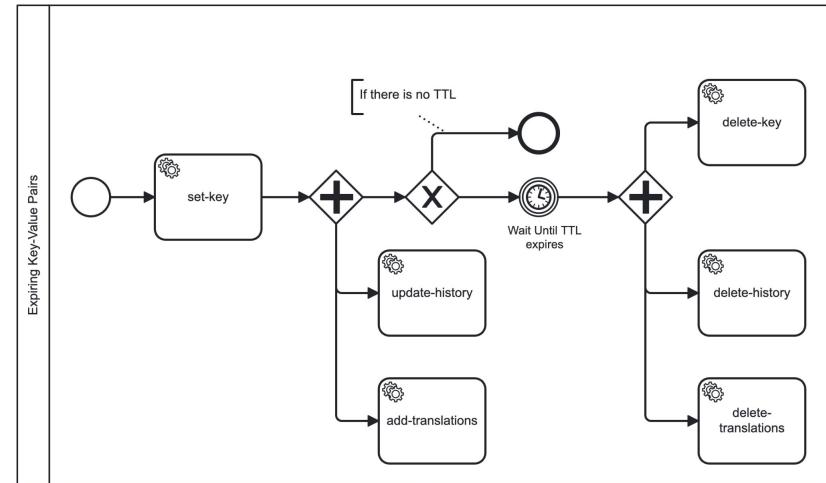
- Business Process Modeling Notation
 - Open standard - bpmn.org
 - Stored as XML





What is BPMN?

- Business Process Modeling Notation
 - Open standard - bpmn.org
 - Stored as XML
- Implemented by orchestration engines
 - Flowable
 - Camunda
 - ...





Focusing on Essential Complexity



Focusing on Essential Complexity



- Historical key-values
- Searching translations
- Expiring Keys

Focusing on Essential Complexity



- Historical key-values
- Searching translations
- Expiring Keys



- Repeated implementations
- Overuse of one technology
- Prolonged discovery

Focusing on Essential Complexity



- Historical key-values
- Searching translations
- Expiring Keys

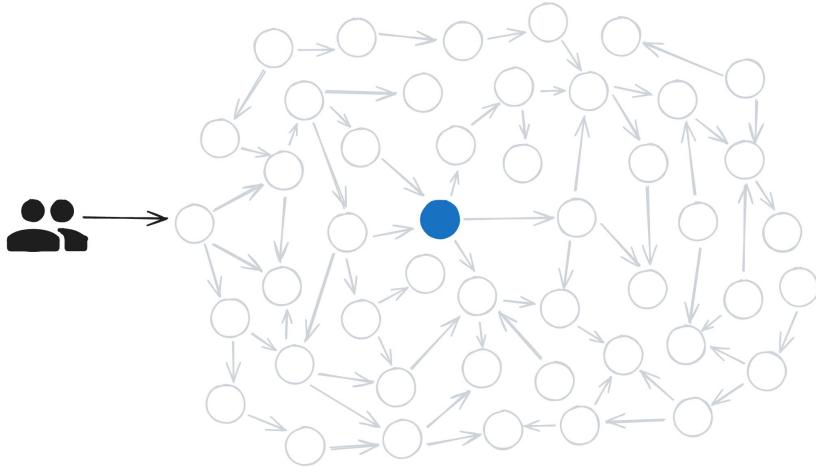


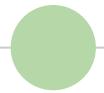
Who Pays the **Cost of Fragmentation** ?



Who Pays the Cost of Fragmentation ?

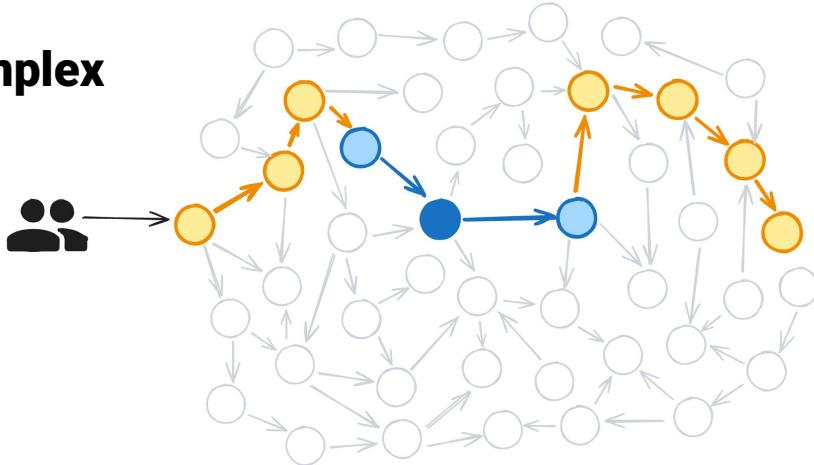
- Microservices gave us a lot of benefits
 - But it also fragmented our systems





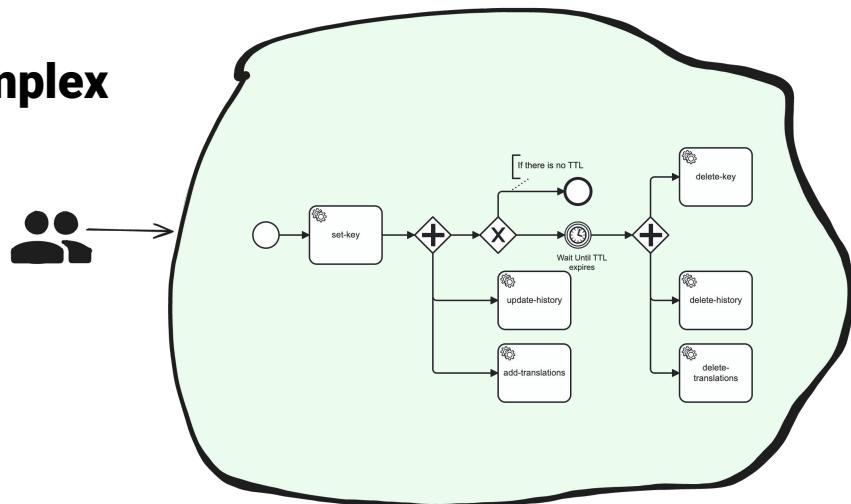
Who Pays the Cost of Fragmentation ?

- Microservices gave us a lot of benefits
 - But it also fragmented our systems
- Fragmented systems are **accidentally complex**
 - Difficult to change
 - Harder to understand



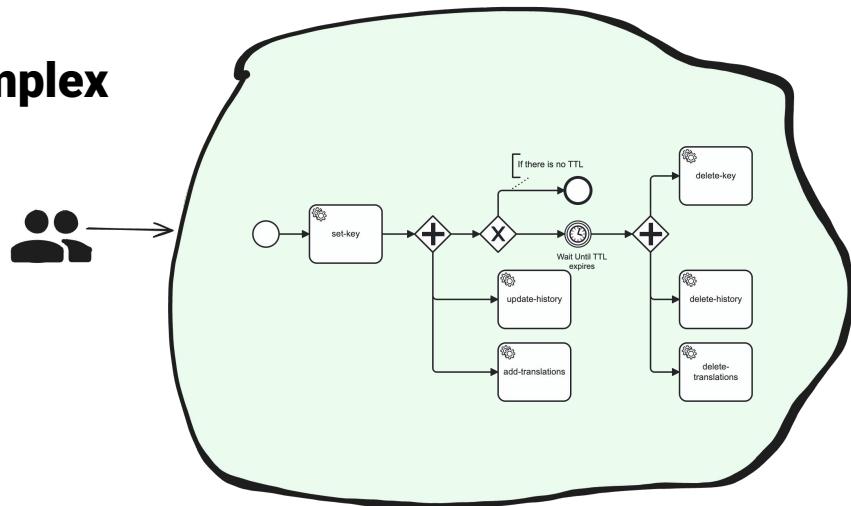
Who Pays the Cost of Fragmentation ?

- Microservices gave us a lot of benefits
 - But it also fragmented our systems
- Fragmented systems are **accidentally complex**
 - Difficult to change
 - Harder to understand
- BPMN diagrams act as the glue
 - More **adaptable to change**
 - A **better mental model**



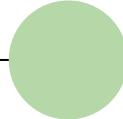
Who Pays the Cost of Fragmentation ?

- Microservices gave us a lot of benefits
 - But it also fragmented our systems
- Fragmented systems are **accidentally complex**
 - Difficult to change
 - Harder to understand
- BPMN diagrams act as the glue
 - More **adaptable to change**
 - A **better mental model**
- Brings humans and systems together
 - Organizational cohesion



Thank You!

Untangling The **API Spaghetti** With
BPMN Diagrams



Mourjo Sen

October 9, 2025

<https://mourjo.me>