Long Running Processes

What is “process”?

A process can be described as a set of activities that are performed in a certain sequence as a result of internal and external triggers.

- The most basic process control is: if-then
- More complex processes include state machines

What is “long running process”?

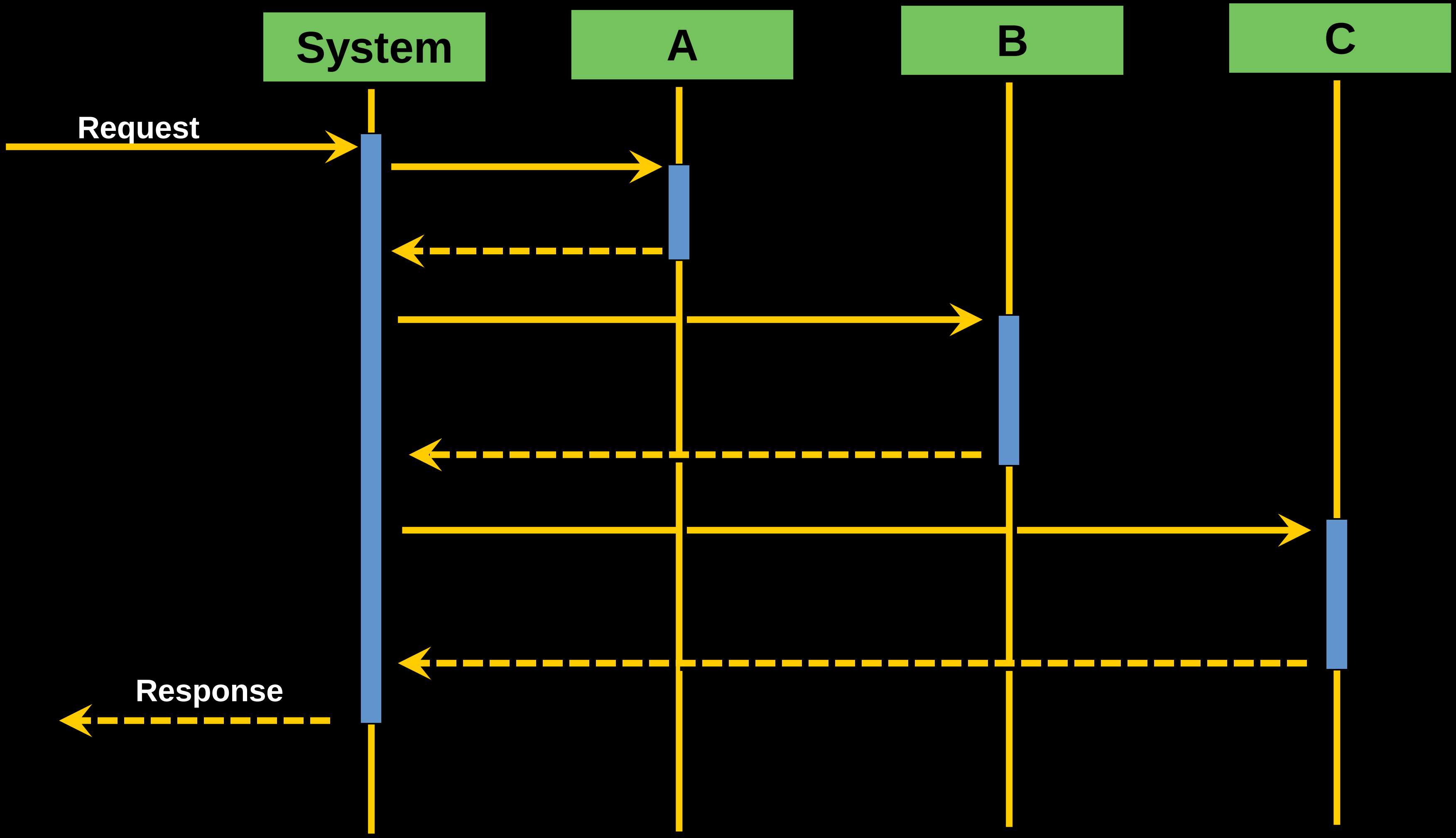
A long running process is a process whose execution lifetime exceeds the time to process a single external event or message.

- Long running means that multiple external events/triggers are handled by the same process instance – is **Stateful**
- Derived from “long-lived transactions” work in the late 80’s and early 90’s

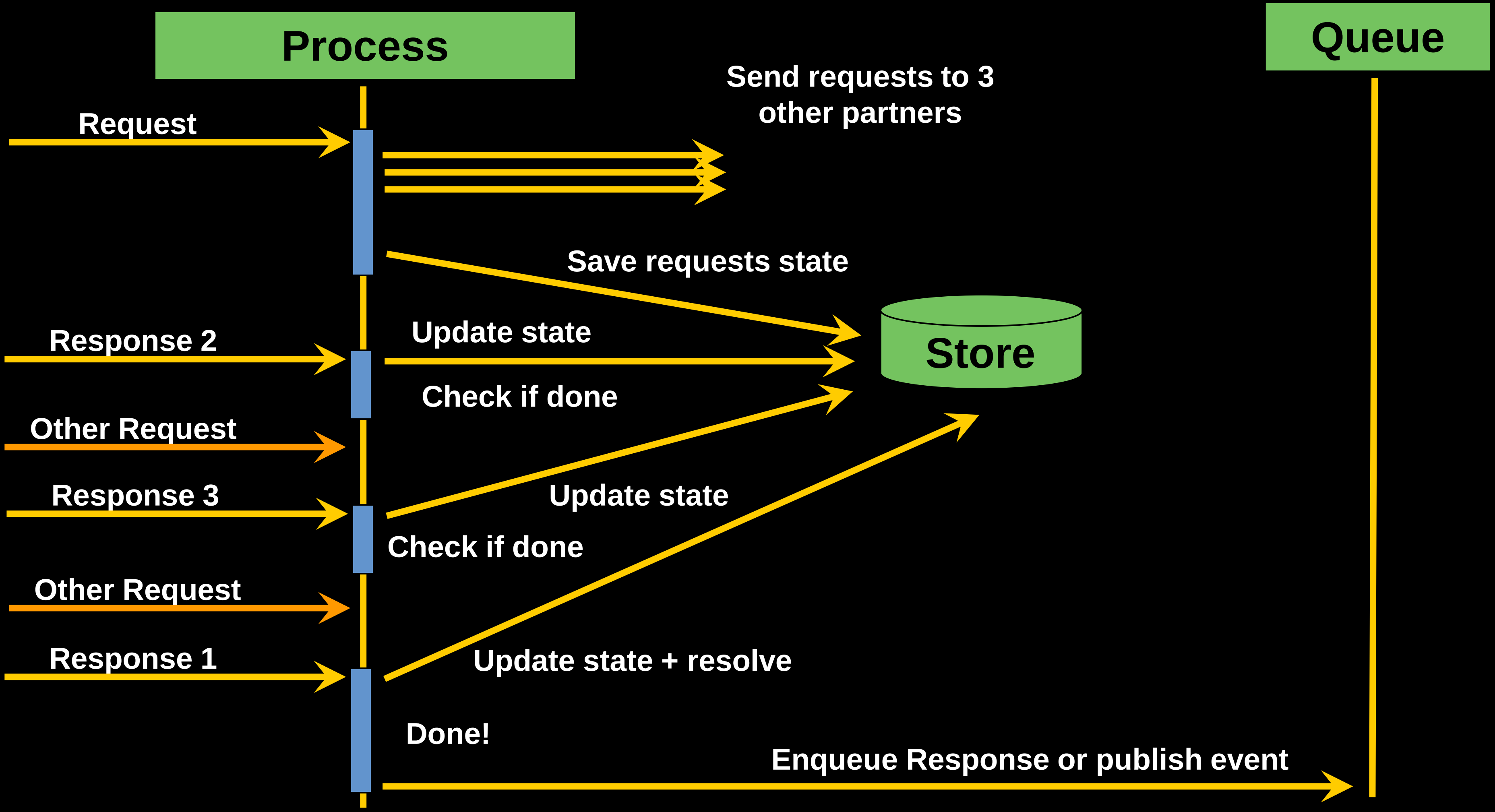
Why use long running process?

- Long running processes provide a state management facility that enables a system to encapsulate the logic and data for handling an external stream of events.
- It’s just good OO programming.

Integration example

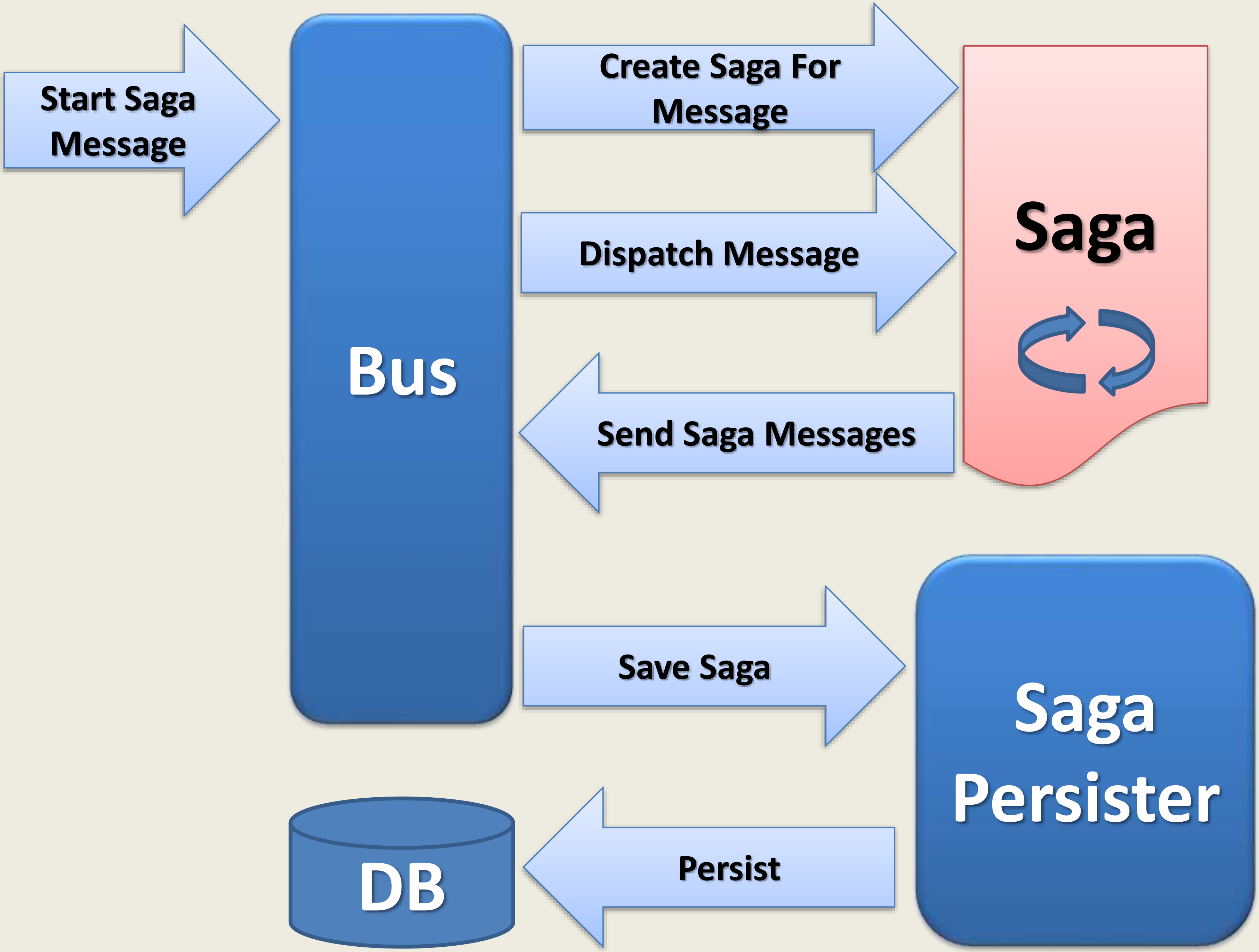


Long running process implementation



Sagas

- Triggers are messages
- Similar to message handlers
  - Can handle a number of different message types
- Different from message handlers
  - Have state, message handlers don’t



Start Saga Message

Bus

Create Saga For Message

Dispatch Message

Send Saga Messages

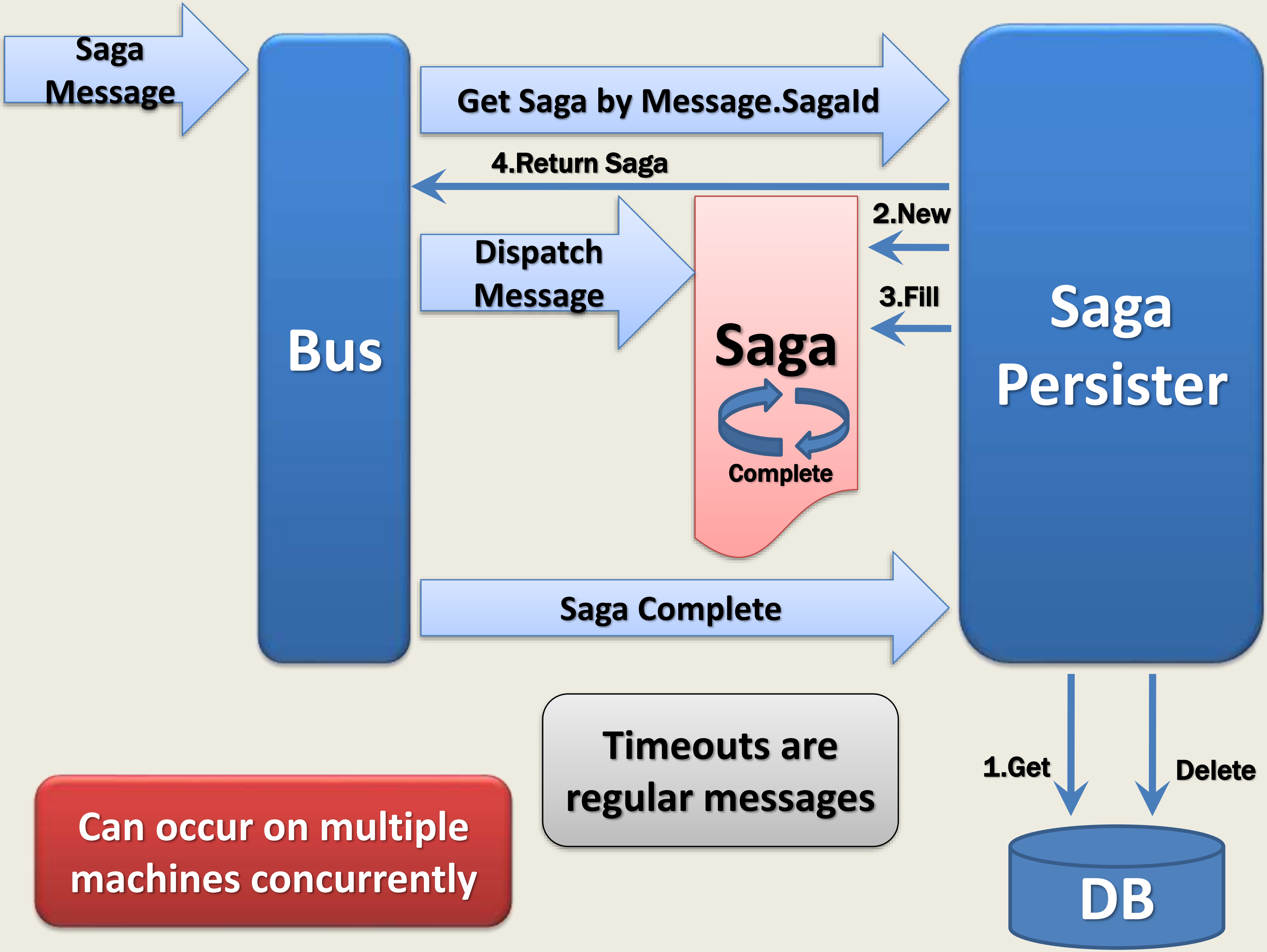
Saga

Save Saga

Saga Persister

Persist

DB



Saga Message

Bus

Get Saga by Message.SagaId

4.Return Saga

Dispatch Message

2.New

3.Fill

Saga

Complete

Saga Persister

Saga Complete

1.Get

Delete

DB

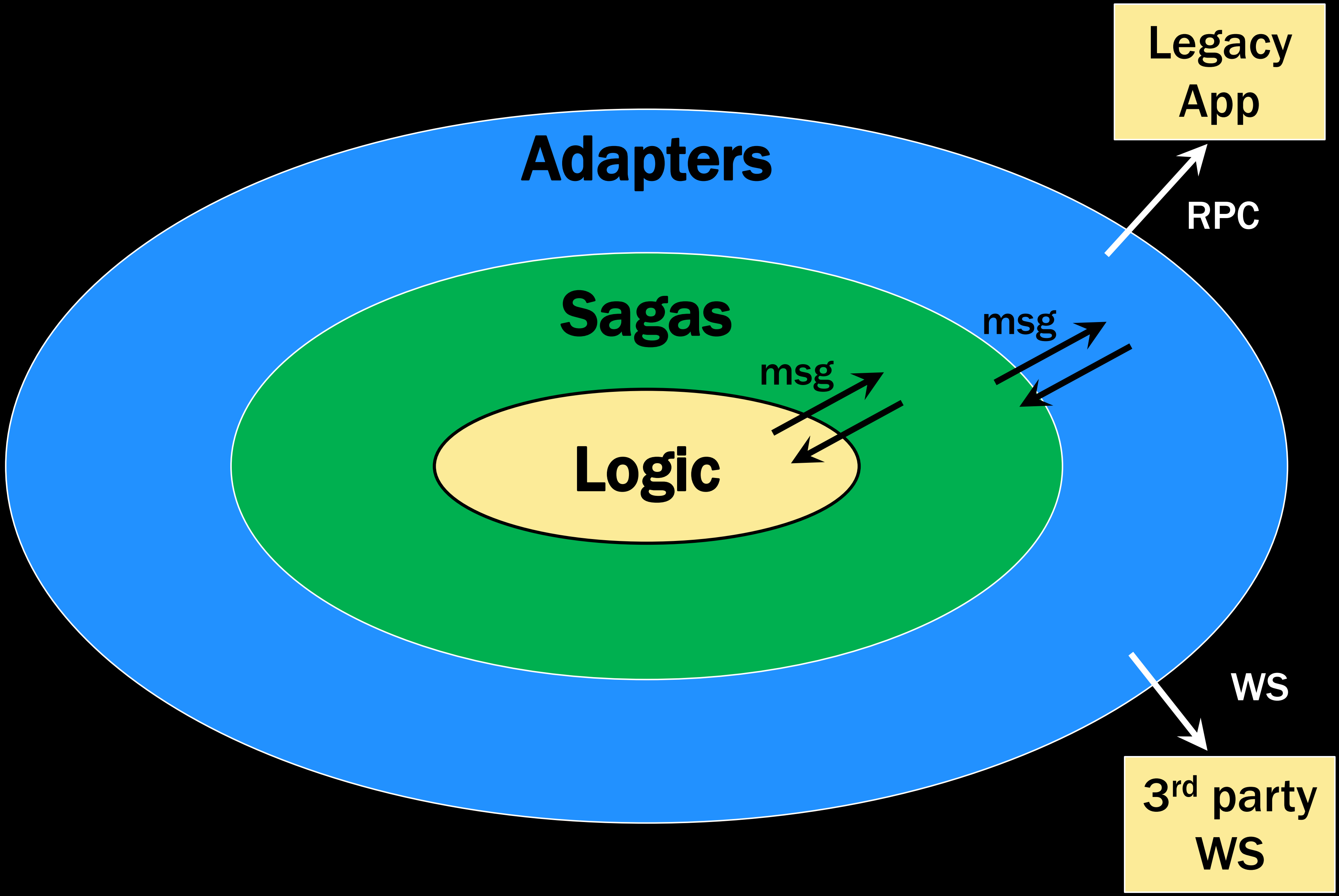
Timeouts are regular messages

Can occur on multiple machines concurrently

The hard part

- The easy part is using the building blocks
- The hard part is analyzing the business processes to identify what the steps should be.
- When interacting with legacy systems, use a saga to manage the flow, a separate adapter for the integration.

Similar to Hexagonal Architecture



Workflow & Orchestration

- **Orchestration is not a service by itself.**

- Divide up workflows/orchestrations along service boundaries
  - Events are published at the end of the sub-flow in a service
  - Events trigger a sub-flow in other services

- Sagas can be used for CEP/ESP:

  complex event processing, event-stream proc.

Summary

- Use messaging building blocks to support long running processes.
- Unit testing is critical for time-bound processes
- Keep service boundaries explicit

Questions?