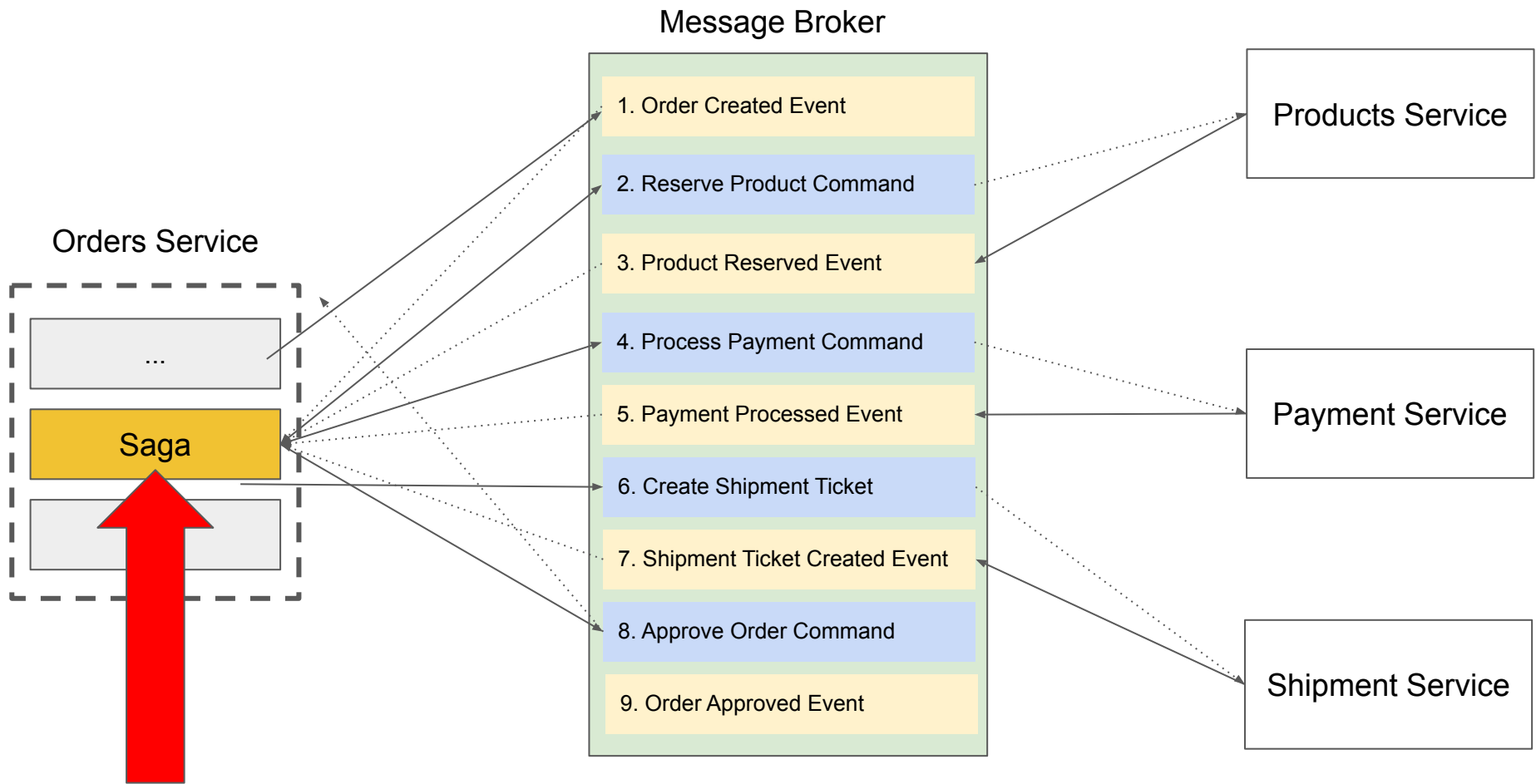
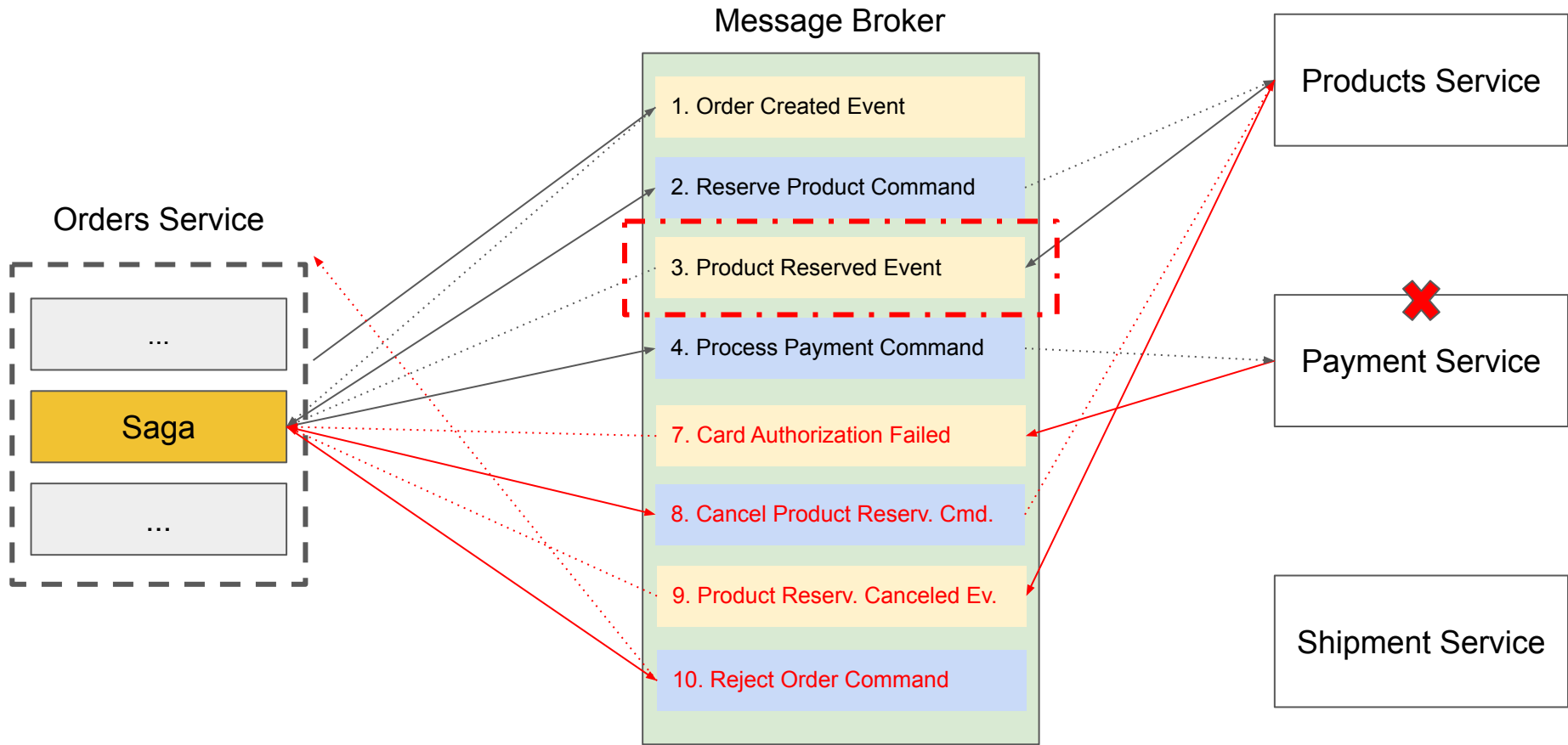


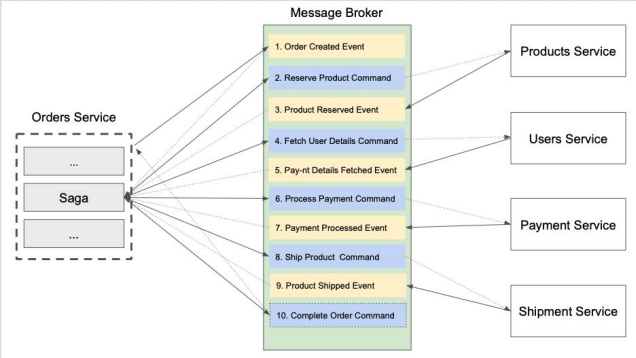
SAGA

Design Pattern

Orchestration-Based Saga







Orchestration-based Saga

Benefits:

- Good for complex workflows involving many Microservices,
- Easier to control the flow of activities,
- Doesn't introduce cyclic dependencies, because the orchestrator unilaterally depends on the saga participants,
- Saga participants don't need to know about commands for other participants. Clear separation of concerns simplifies business logic.

Drawbacks:

- Additional design complexity requires an implementation of a coordination logic.
- There's an additional point of failure, because the orchestrator manages the complete workflow.