



Spring Boot Microservices

Beginner to Guru

Saga Coordination



Saga Coordination

- **Saga Definition** - *"A long, involved story, account, or series of incidents"*
 - Saga - Series of Steps
 - With compensating transactions
- How to define a saga in a distributed environment?
 - Two Primary Approaches for Saga Coordination
 - **Choreography** - Distributed decision making. Each actor decides next steps.
 - **Orchestration** - Centralized decision making. Central component decides next steps.



Choreography Coordination

- Distributed Decision Making - each actor determines next steps
- **Benefits**
 - Simple, loosely coupled
 - Good for simpler Sagas
- **Problems**
 - Cyclic dependencies
 - Harder to understand - logic is spread out
 - Components are more complex



Choreography Implementation

- Choreography Coordination typically implemented using events
- Each actor emits an event for the next step in the Saga
- Requires each actor to have logic about the Saga
- Each actor needs to know how to perform a compensating transaction
 - Thus each actor has more coupling to other system components



Orchestration Based Coordination

- Centralized Decision Making - step components do not decide next steps
- **Benefits**
 - Logic is centralized and easier to understand
 - Reduced coupling, better separation of concerns
- **Problems**
 - Risk of over centralization - need to maintain focus on separation of concerns



Orchestration Implementation

- Orchestration Coordination has a central component directing other actors
- Central component maintains state for the Saga
 - State Machine
 - Saga Execution Coordinator (aka SEC)
 - Event Sourcing
- Must take responsibility for the completion of the Saga
 - ie persist state to DB, use persistent message queues, etc



Conclusion

- Which to Use?
 - Choreography Coordination for smaller simple Sagas
 - Orchestration Coordination for larger more complex Sagas
- How to Implement?
 - Typically custom solution - wide variety of implementations
 - Open Source / Commercial solutions are emerging
 - Still fairly early and are maturing

