

# Distributed Transactions - Learning Materials

## Articles & Blogs

- [Saga Pattern: How to Implement Business Transactions Using Microservices](#) - Comprehensive guide on Saga implementation
- [Pattern: Transactional Outbox](#) - Chris Richardson's definitive guide
- [Life Beyond Distributed Transactions](#) - Pat Helland's influential paper on scalable systems

## Videos

- [Distributed Transactions are Dead](#) - Sergey Bykov on modern alternatives
- [Saga Pattern Explained](#) - ByteByteGo's visual explanation
- [Outbox Pattern with Debezium](#) - Gunnar Morling's deep dive

## Technical Documentation

- [Debezium Outbox Event Router](#) - Official CDC implementation guide
- [AWS Step Functions for Saga](#) - AWS implementation pattern
- [Temporal.io Saga Tutorial](#) - Workflow-based Saga implementation

## Key Concepts to Explore

- Two-phase commit (2PC) and three-phase commit (3PC)
- Saga orchestration vs choreography
- Compensation and rollback strategies
- Eventual consistency and BASE properties
- Idempotency and exactly-once semantics