



## Ultimate CQRS Architecture Documentation

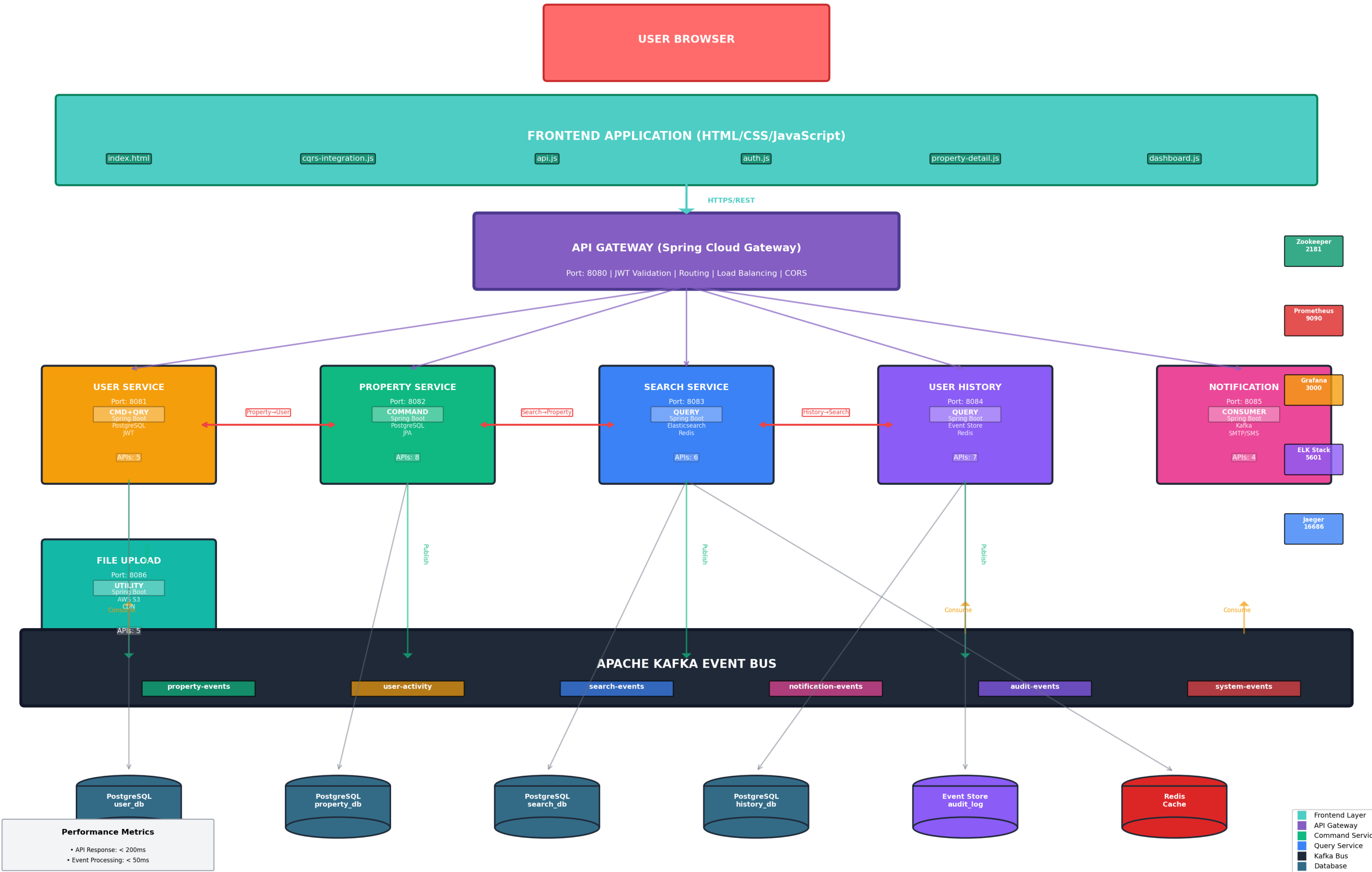
Complete Technical Specification with Detailed Flow Diagrams

Section	Description	Page
1. Microservices Architecture	Complete system with all components and connections	2
2. Sequence Diagram	Property creation flow with 32 detailed steps	3
3. Database Schema	All tables, relationships, and JSONB structures	4
4. Kafka Architecture	Event streaming with topics and consumer groups	5
5. API Documentation	All 51 endpoints with request/response examples	6

# 1. Complete Microservices Architecture

# HYDERABAD INFRA - COMPLETE MICROSERVICES ARCHITECTURE

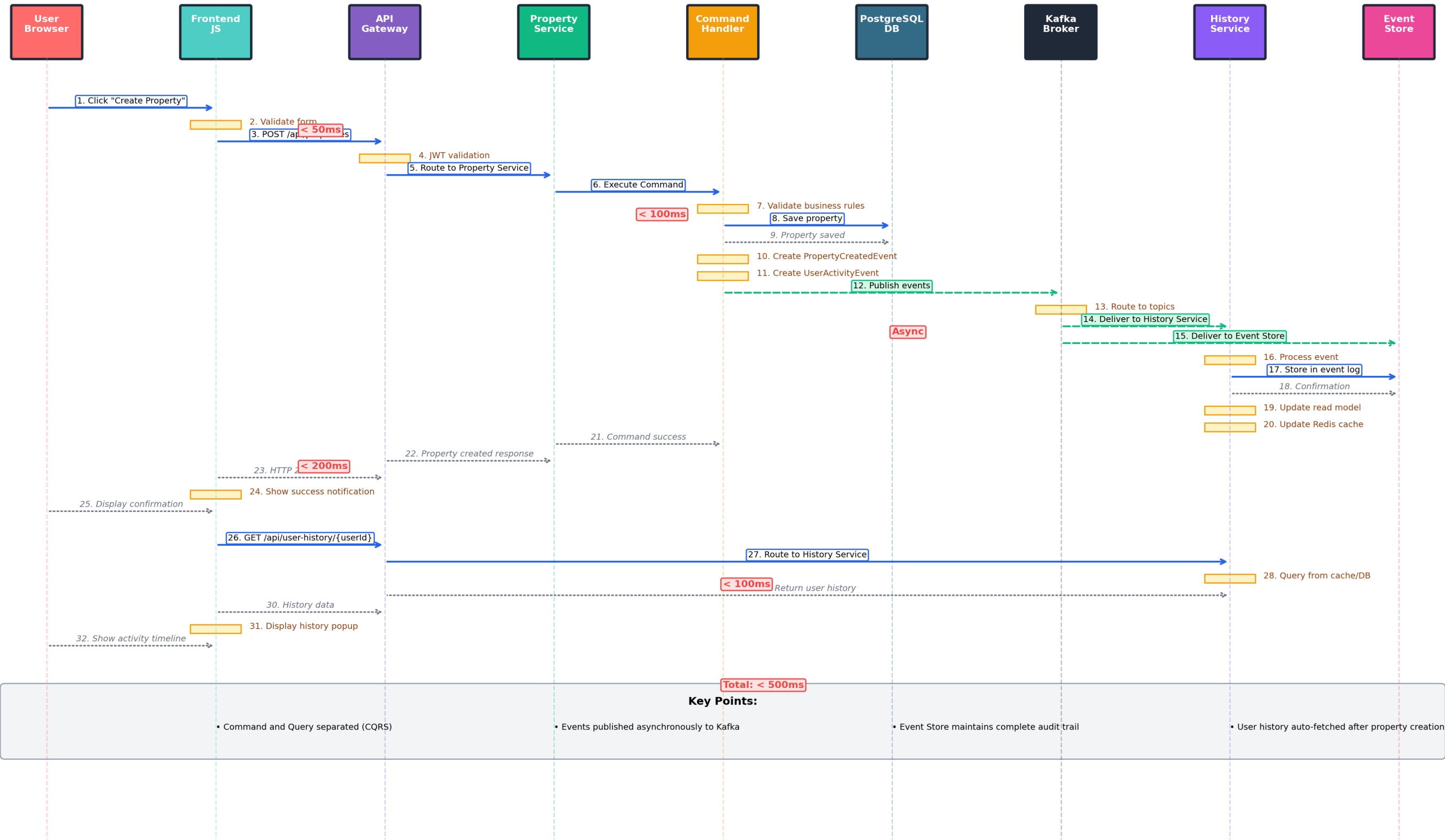
CQRS + Event Sourcing + Kafka + Microservices



## 2. Complete Sequence Diagram - Property Creation

### COMPLETE SEQUENCE DIAGRAM - PROPERTY CREATION WITH CQRS

End-to-End Flow with Event Sourcing and User History



### 3. Complete Database Schema

#### COMPLETE DATABASE SCHEMA - CQRS IMPLEMENTATION



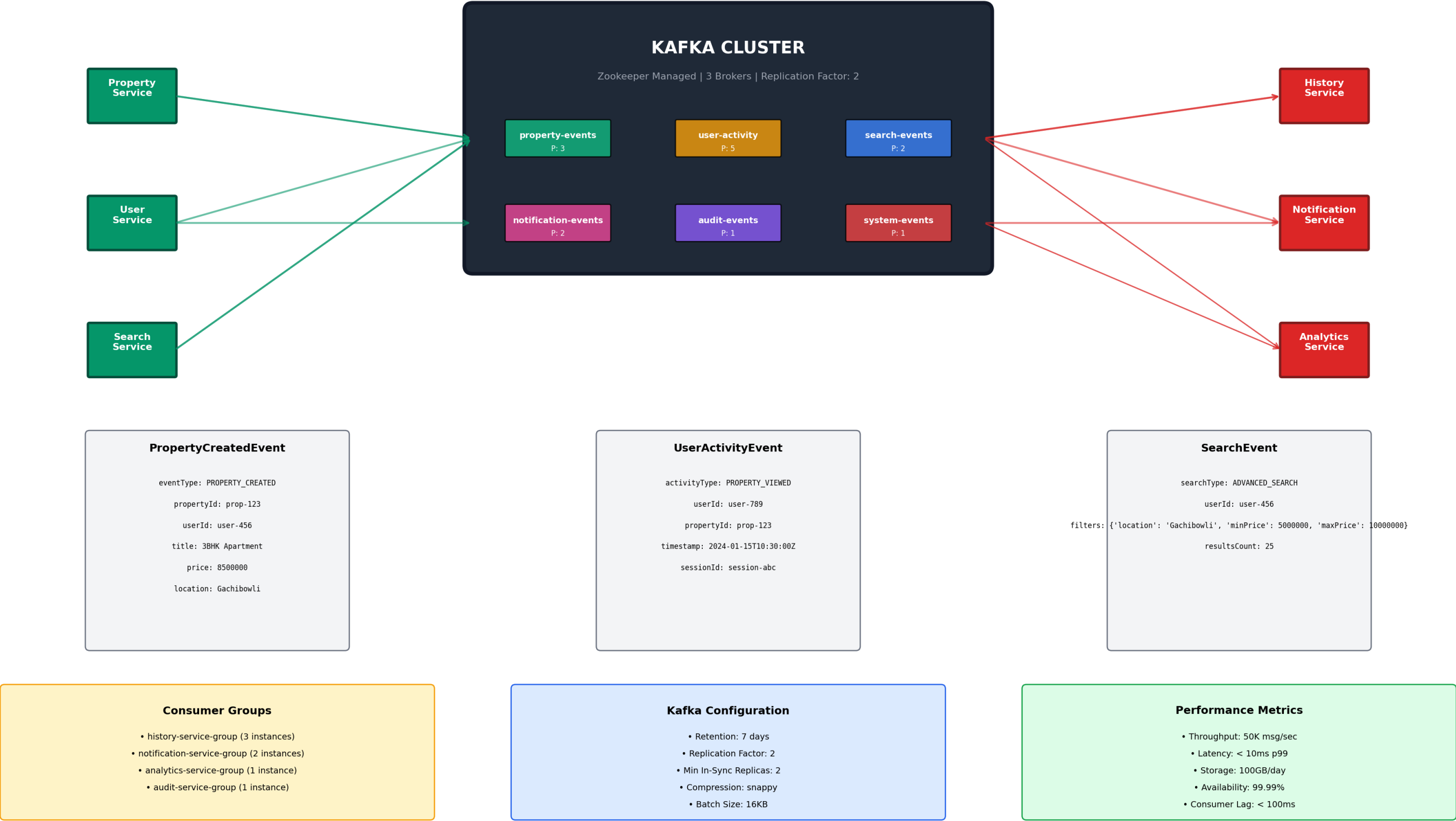
#### Database Performance Optimizations

- Indexes on foreign keys and frequently queried columns
- Connection pooling (HikariCP)
- JSONB for flexible schema and fast queries
- Redis caching for hot data
- Read replicas for query scaling
- Partitioning for event\_store table

# 4. Kafka Event Streaming Architecture

## KAFKA EVENT STREAMING ARCHITECTURE

Complete Event Flow with Topics, Partitions, and Consumers



# 5. Complete API Documentation

## COMPLETE API DOCUMENTATION - ALL ENDPOINTS

User Service (8081)			
POST	/api/auth/register	(User registration)	CMD
POST	/api/auth/login	(User login + auto history)	CMD
POST	/api/auth/refresh	(Refresh JWT token)	CMD
POST	/api/auth/logout	(User logout)	CMD
GET	/api/users/{id}	(Get user profile)	QRY
PUT	/api/users/{id}	(Update user profile)	CMD
DELETE	/api/users/{id}	(Delete user account)	CMD
GET	/api/users/{id}/properties	(User properties)	QRY
POST	/api/users/{id}/property-count	(Update count)	CMD
GET	/api/users/search	(Search users)	QRY

Property Service (8082)			
POST	/api/properties	(Create property)	CMD
GET	/api/properties/{id}	(Get property details)	QRY
PUT	/api/properties/{id}	(Update property)	CMD
DELETE	/api/properties/{id}	(Delete property)	CMD
POST	/api/properties/{id}/view	(Track property view)	CMD
GET	/api/properties/user/{userId}	(User properties)	QRY
GET	/api/properties/featured	(Featured properties)	QRY
GET	/api/properties/recent	(Recent properties)	QRY
POST	/api/properties/{id}/images	(Upload images)	CMD
GET	/api/properties/similar/{id}	(Similar properties)	QRY

Search Service (8083)			
GET	/api/search/properties	(Search properties)	QRY
POST	/api/search/advanced	(Advanced search)	QRY
GET	/api/search/filters	(Get search filters)	QRY
GET	/api/search/suggestions	(Auto-suggestions)	QRY
GET	/api/search/history/{userId}	(Search history)	QRY
POST	/api/search/save	(Save search)	CMD
GET	/api/search/saved/{userId}	(Saved searches)	QRY
DELETE	/api/search/saved/{id}	(Delete saved search)	CMD

User History (8084)			
GET	/api/user-history/{userId}	(Complete history)	QRY
GET	/api/user-history/{userId}/recent	(Recent activities)	QRY
GET	/api/user-history/{userId}/properties	(Property activities)	QRY
GET	/api/user-history/{userId}/searches	(Search history)	QRY
GET	/api/user-history/{userId}/events	(Raw events)	QRY
GET	/api/user-history/{userId}/timeline	(Activity timeline)	QRY
GET	/api/user-history/{userId}/stats	(User statistics)	QRY
POST	/api/user-history/replay	(Replay events)	CMD

Notification (8085)			
POST	/api/notifications/email	(Send email)	CMD
POST	/api/notifications/sms	(Send SMS)	CMD
GET	/api/notifications/{userId}	(User notifications)	QRY
PUT	/api/notifications/{id}/read	(Mark as read)	CMD
DELETE	/api/notifications/{id}	(Delete notification)	CMD
GET	/api/notifications/preferences/{userId}	(User preferences)	QRY
PUT	/api/notifications/preferences	(Update preferences)	CMD

File Upload (8086)			
POST	/api/files/upload	(Upload file)	CMD
GET	/api/files/{id}	(Download file)	QRY
DELETE	/api/files/{id}	(Delete file)	CMD
POST	/api/files/bulk	(Bulk upload)	CMD
GET	/api/files/property/{id}	(Property files)	QRY
POST	/api/files/optimize	(Optimize images)	CMD

### Request Example

```
POST /api/properties
Headers:
Authorization: Bearer JWT_TOKEN
Content-Type: application/json
X-User-Id: user-456
X-Session-Id: session-abc

Body:
{
  "title": "3BHK Apartment",
  "location": "Gachibowli",
  "price": 9500000,
  "propertyType": "APARTMENT",
  "bedrooms": 3,
  "bathrooms": 2
}
```

### Response Example

```
HTTP/1.1 201 Created
Location: /api/properties/prop-123

{
  "success": true,
  "message": "Property created",
  "data": {
    "id": "prop-123",
    "title": "3BHK Apartment",
    "createdAt": "2024-01-15T10:30:00Z"
  }
}
```

### API Statistics

Total Endpoints: 51

Command APIs: 23

Query APIs: 28

Avg Response: < 200ms