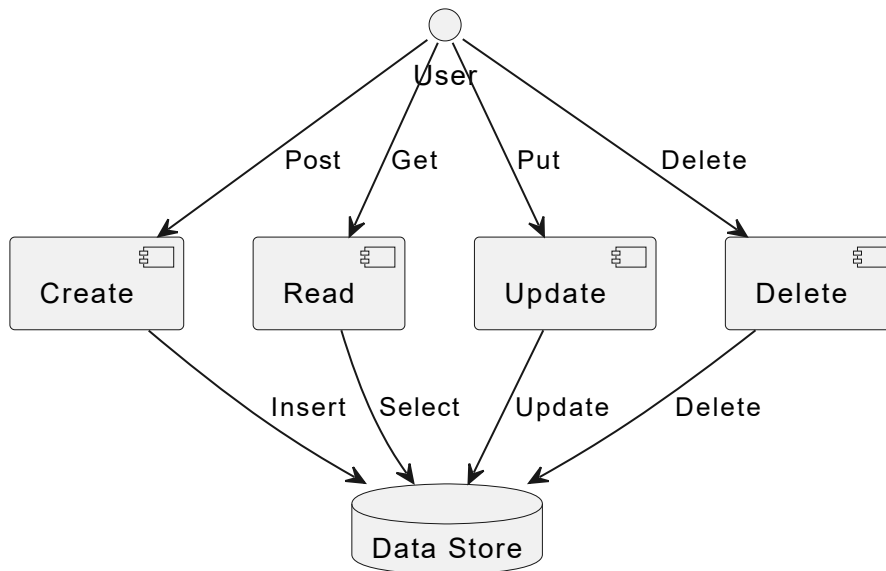# Data Access Patterns

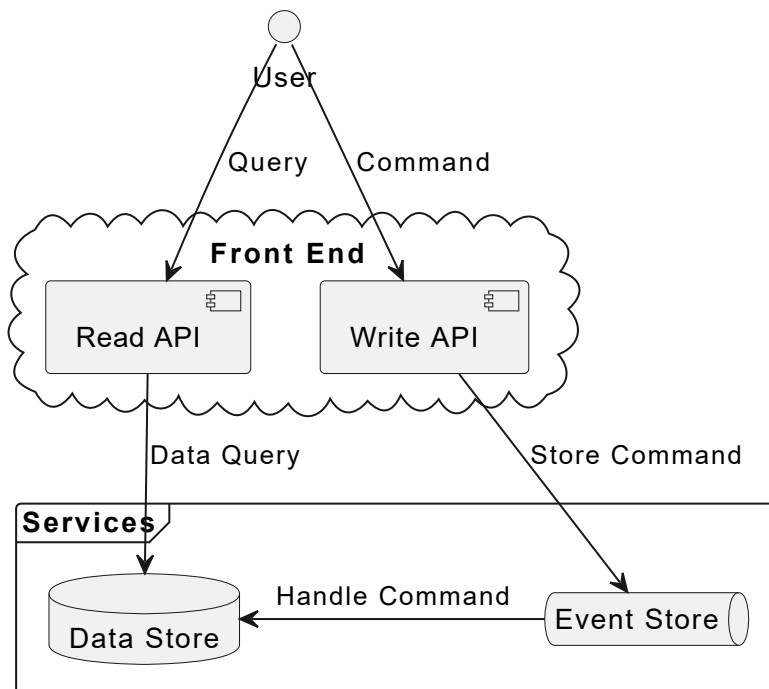## Create, Read, Update, Delete (CRUD)

Data Access Pattern - Create, Read, Update, Delete (CRUD)
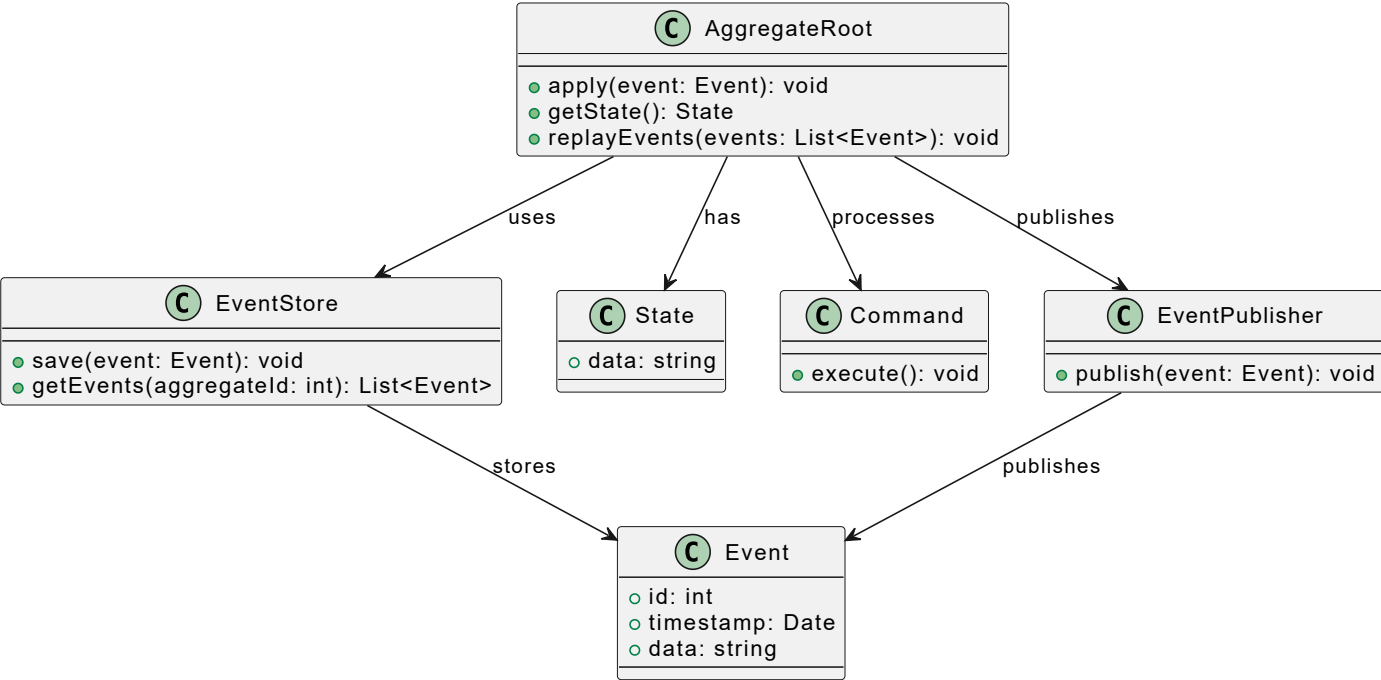

Command Query Responsibility Segregation (CQRS)

Data Access Pattern - Command Query Responsibility Segregation (CQRS)

Event Sourcing

Data Access Pattern - Event Sourcing

```
┌─────────────────────────────────────────┐
│  Ⓒ  AggregateRoot                        │
├─────────────────────────────────────────┤
├─────────────────────────────────────────┤
│  ● apply(event: Event): void             │
│  ● getState(): State                     │
│  ● replayEvents(events: List<Event>): void│
└─────────────────────────────────────────┘
```

uses   has   processes   publishes

```
┌──────────────────────────────────────┐   ┌──────────────┐  ┌────────────────┐  ┌───────────────────────────────┐
│  Ⓒ  EventStore                       │   │ Ⓒ State      │  │ Ⓒ Command      │  │ Ⓒ  EventPublisher             │
├──────────────────────────────────────┤   ├──────────────┤  ├────────────────┤  ├───────────────────────────────┤
├──────────────────────────────────────┤   │ ○ data: string│  │                │  ├───────────────────────────────┤
│  ● save(event: Event): void          │   │              │  │ ● execute(): void│  │ ● publish(event: Event): void │
│  ● getEvents(aggregateId: int): List<Event>│└──────────────┘  └────────────────┘  └───────────────────────────────┘
└──────────────────────────────────────┘
```

stores   publishes

```
┌──────────────────────┐
│  Ⓒ  Event            │
├──────────────────────┤
│  ○ id: int           │
│  ○ timestamp: Date   │
│  ○ data: string      │
└──────────────────────┘
```

Repository

Data Access Pattern - Repository

```
┌────────────────────────────────┐
│  Ⓒ  Repository                 │
├────────────────────────────────┤
├────────────────────────────────┤
│  ● add(entity: Entity): void   │
│  ● findById(id: int): Entity   │
│  ● remove(entity: Entity): void│
└────────────────────────────────┘
```

manages

```
┌──────────────────────┐
│  Ⓒ  Entity           │
├──────────────────────┤
│  ○ id: int           │
│  ○ name: string      │
└──────────────────────┘
```