Mars Mission AI - High-Level Architecture

Client Applications

Kong API GatewayPort 8000/8001

Microservices Layer

Planning ServicePort 8005

Vision ServicePort 8002

MARL Ser

Data IntegrationPort 8004

ML Models

SegFormerModel

5 RL AgentsTrained

PostgreSQLPort 5432

Redis CachePort 6379
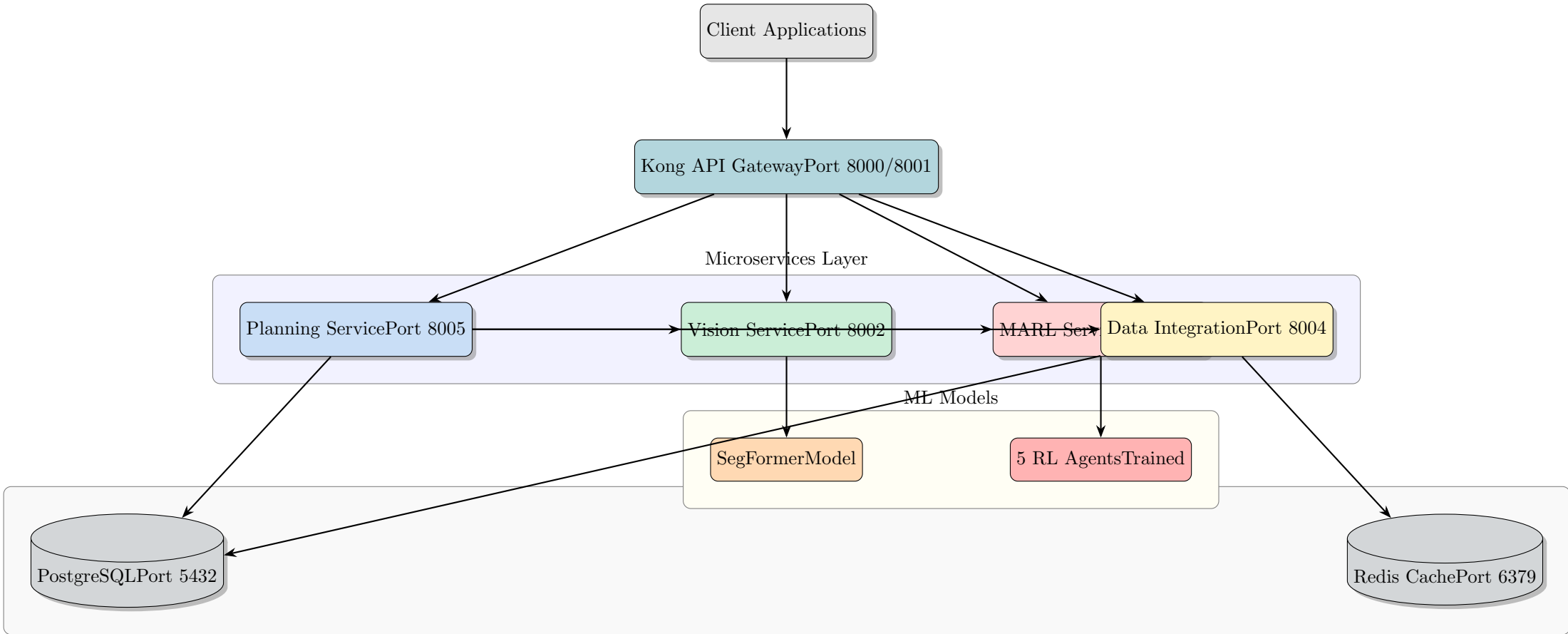
MARL System Architecture
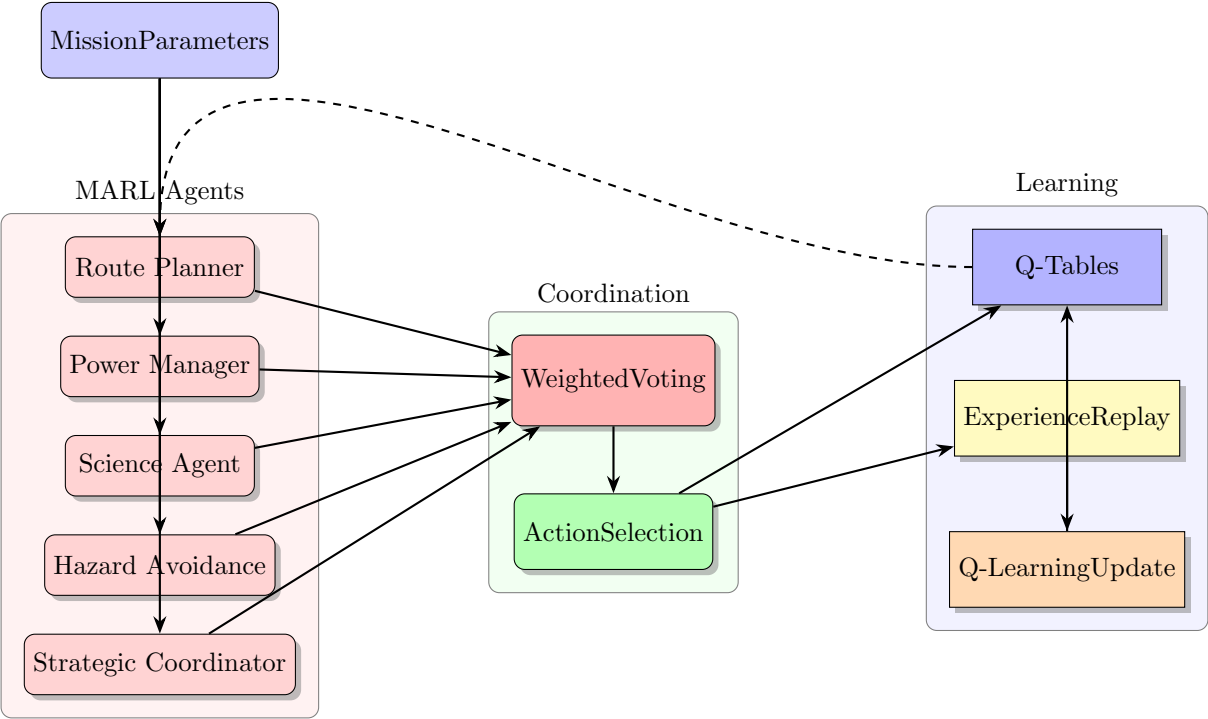
Data Flow Sequence Diagram

Client   Gateway   Planning   Data   MARL

POST /plan
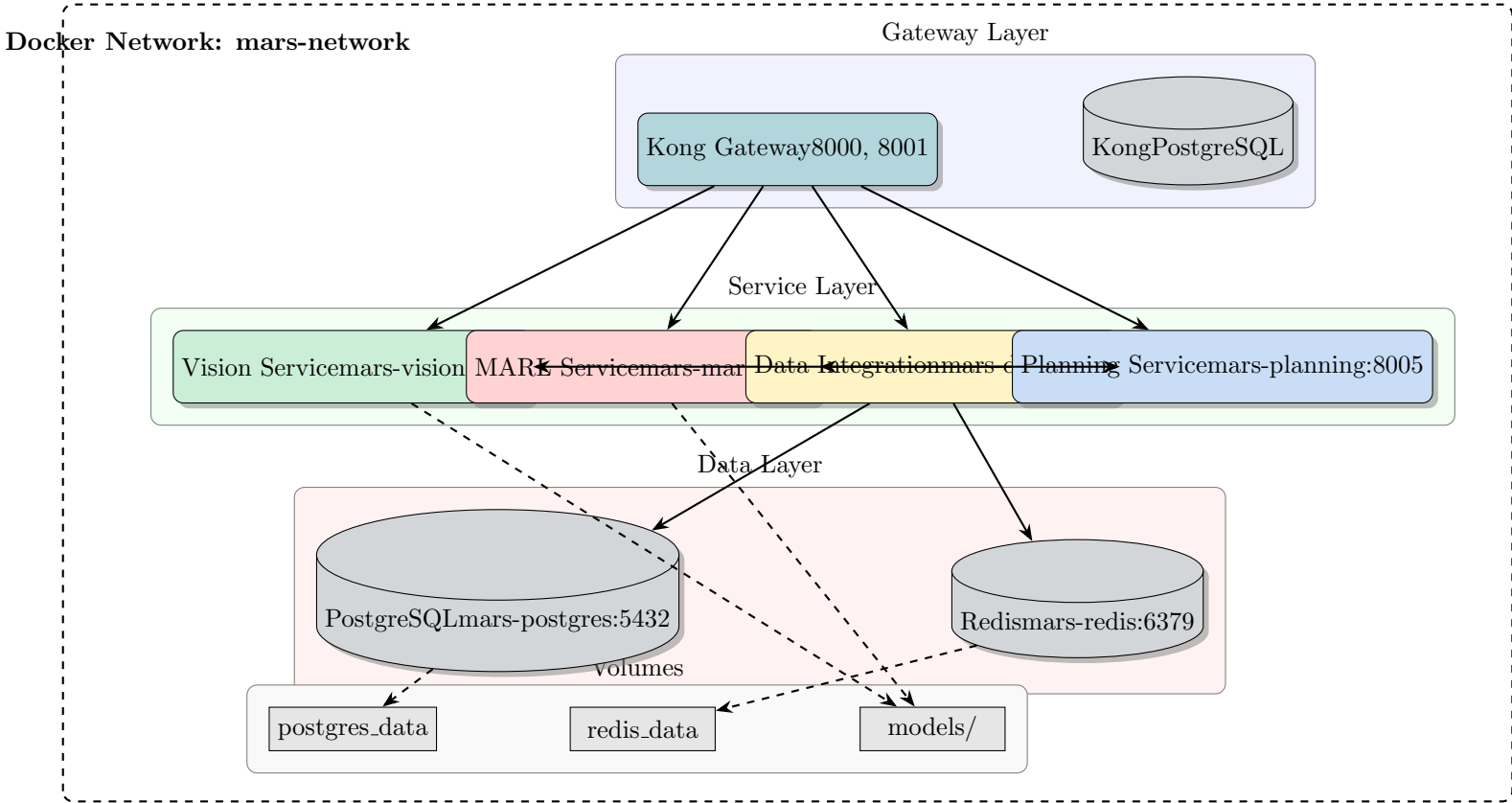
forward

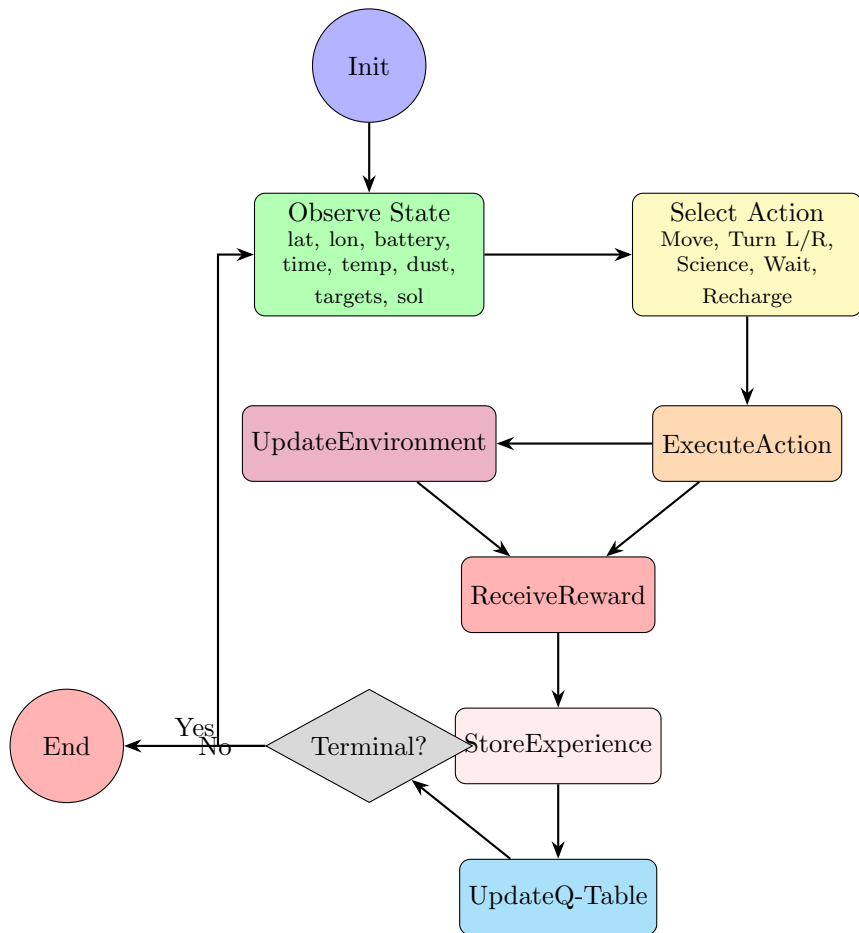GET /integrated

data

POST /optimize

optimized plan

response
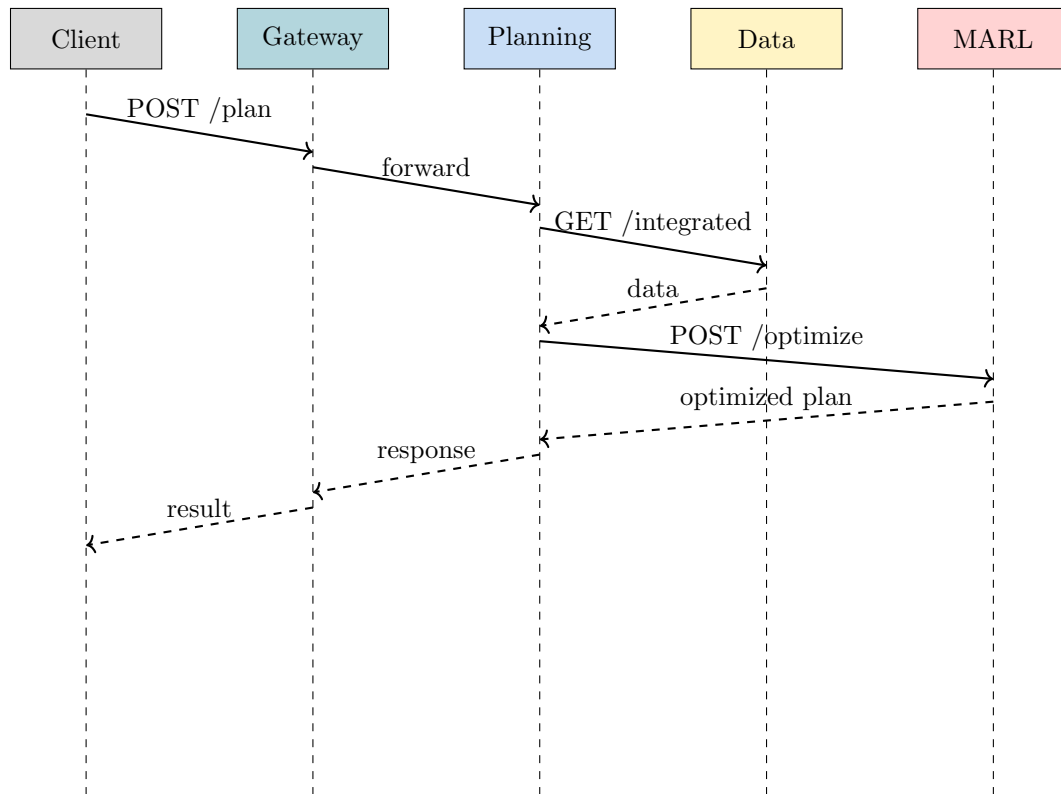
result

1. Client requests
2. Gateway routes
3. Planning gets data
4. MARL optimizes
5. Return result

Docker Deployment Architecture

Docker Network: mars-network

Gateway Layer

Kong Gateway8000, 8001

KongPostgreSQL

Service Layer

Vision Servicemars-vision | MARS Servicemars-mar | Data Integrationmars-c | Planning Servicemars-planning:8005

Data Layer

PostgreSQLmars-postgres:5432

Redismars-redis:6379

Volumes

postgres_data

redis_data

models/

MARL Agent State-Action Space

MARL Training Performance

Average Reward

Target: 782.3

Training Progress

Convergence~300 episodes

Legend: Average Reward, Final Performance

Episode

Service Response Time (95)

Service	Time (ms)
Vision	200ms
MARL	85ms
Data	320ms
Planning	1200ms
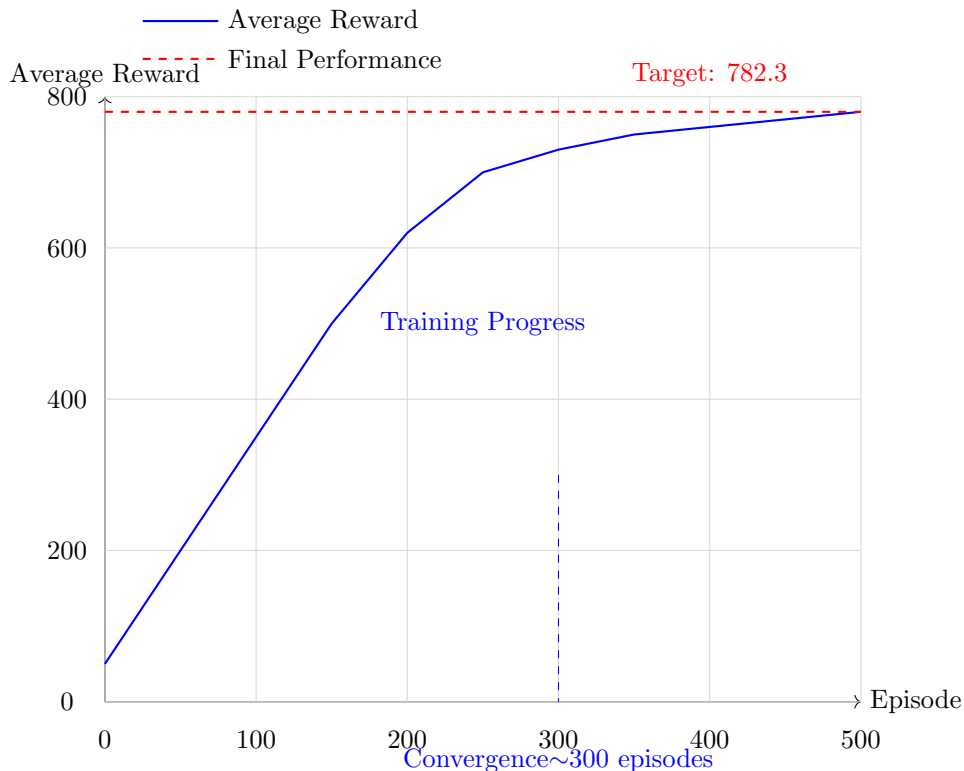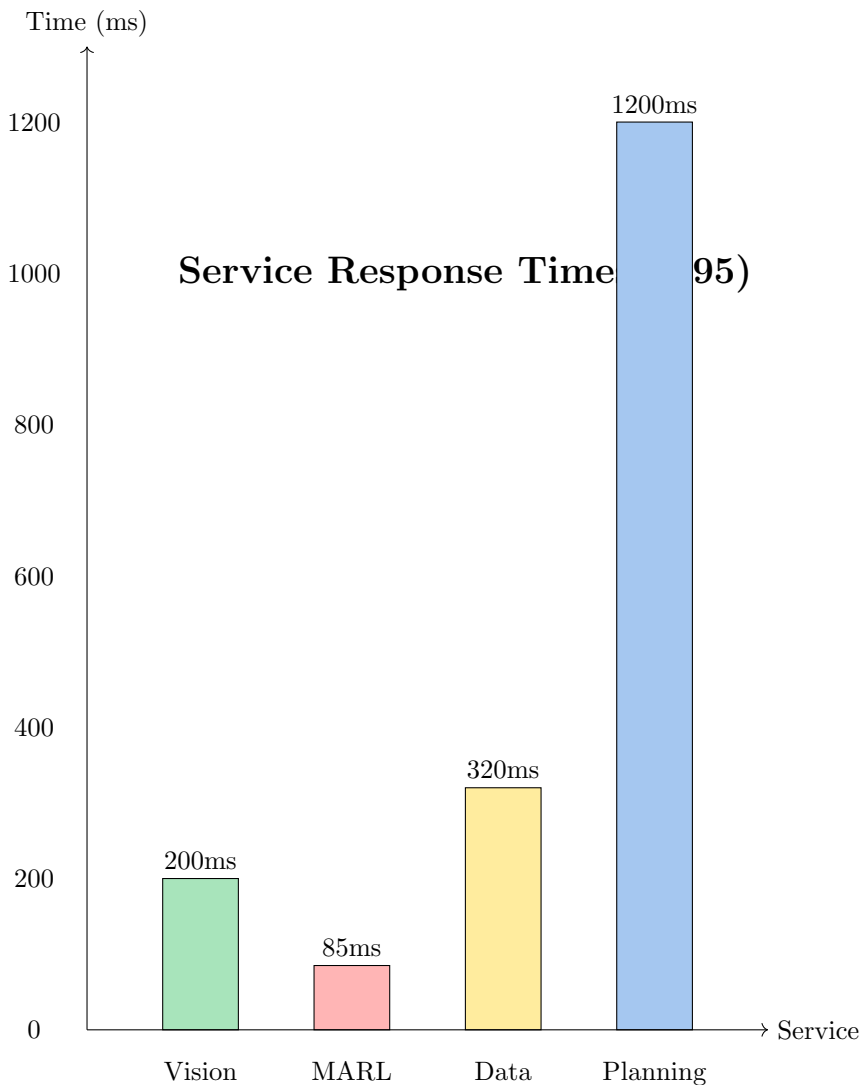
Microservices Communication Pattern

User

HTTP/REST

Synchronous
REST calls

API GatewayLoad Balancer

Async
messaging

MessageQueue(Optional)

VisionService

ServiceDiscovery(DNS)

PlanningService

MARLService