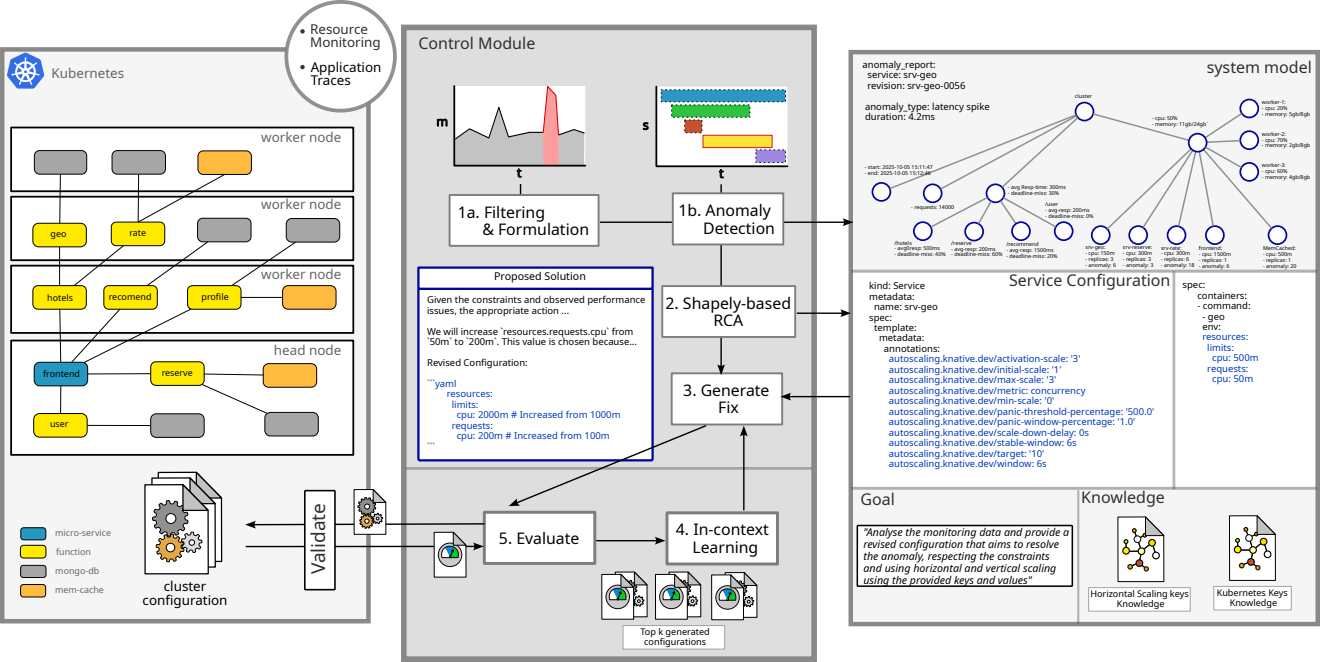
Kubernetes
• Resource Monitoring
• Application Traces
worker node
geo
rate
worker node
hotels
recomend
profile
worker node
frontend
reserve
user
head node
micro-service
function
mongo-db
mem-cache
cluster configuration
Validate
Control Module
m
t
s
t
1a. Filtering & Formulation
1b. Anomaly Detection
Proposed Solution
Given the constraints and observed performance issues, the appropriate action ...
We will increase `resources.requests.cpu` from `50m` to `200m`. This value is chosen because...
Revised Configuration:
```yaml
resources:
limits:
cpu: 2000m # Increased from 1000m
requests:
cpu: 200m # Increased from 100m
```
2. Shapely-based RCA
3. Generate Fix
5. Evaluate
4. In-context Learning
Top k generated configurations
system model
anomaly_report:
service: srv-geo
revision: srv-geo-0056
anomaly_type: latency spike
duration: 4.2ms
cluster
- cpu: 50%
- memory: 11gb/24gb
worker-1:
- cpu: 20%
- memory: 5gb/8gb
worker-2:
- cpu: 70%
- memory: 2gb/8gb
worker-3:
- cpu: 60%
- memory: 4gb/8gb
- start: 2025-10-05 15:11:47
- end: 2025-10-05 15:12:46
- requests: 14000
- avg Resp-time: 300ms
- deadline-miss: 30%
/user
- avg resp: 200ms
- deadline-miss: 0%
/hotels
- avg Resp: 500ms
- deadline-miss: 40%
/reserve
- avg resp: 200ms
- deadline-miss: 60%
/recommend
- avg resp: 1500ms
- deadline-miss: 20%
srv-geo:
- cpu: 150m
- replicas: 1
- anomaly: 6
srv-reserve:
- cpu: 300m
- replicas: 3
- anomaly: 3
srv-rate:
- cpu: 300m
- replicas: 6
- anomaly: 18
frontend:
- cpu: 1500m
- replicas: 1
- anomaly: 6
MemCached:
- cpu: 500m
- replicas: 1
- anomaly: 20
Service Configuration
kind: Service
metadata:
name: srv-geo
spec:
template:
metadata:
annotations:
autoscaling.knative.dev/activation-scale: '3'
autoscaling.knative.dev/initial-scale: '1'
autoscaling.knative.dev/max-scale: '3'
autoscaling.knative.dev/metric: concurrency
autoscaling.knative.dev/min-scale: '0'
autoscaling.knative.dev/panic-threshold-percentage: '500.0'
autoscaling.knative.dev/panic-window-percentage: '1.0'
autoscaling.knative.dev/scale-down-delay: 0s
autoscaling.knative.dev/stable-window: 6s
autoscaling.knative.dev/target: '10'
autoscaling.knative.dev/window: 6s
spec:
containers:
- command:
- geo
env:
resources:
limits:
cpu: 500m
requests:
cpu: 50m
Goal
"Analyse the monitoring data and provide a revised configuration that aims to resolve the anomaly, respecting the constraints and using horizontal and vertical scaling using the provided keys and values"
Knowledge
Horizontal Scaling keys Knowledge
Kubernetes Keys Knowledge