In-situ (LED Component) - WP1

Michael Mock *   Ehab Qadah *

*Fraunhofer IAIS, Germany

Monday 15th January, 2018

Outline

datAcron Architecture
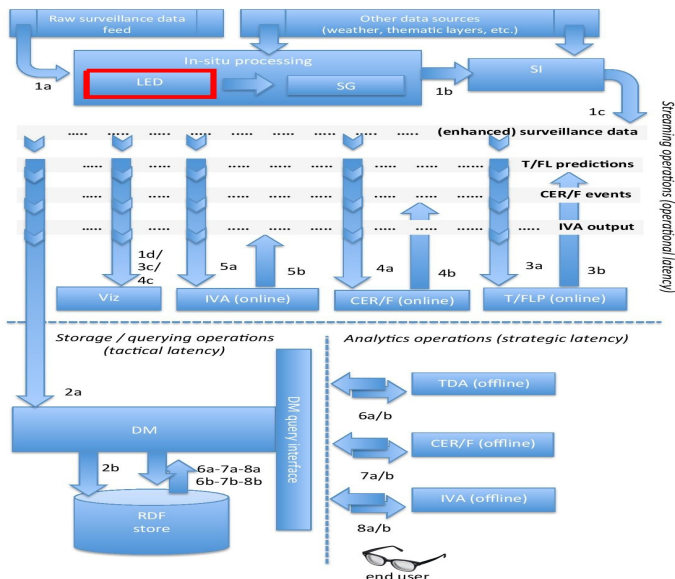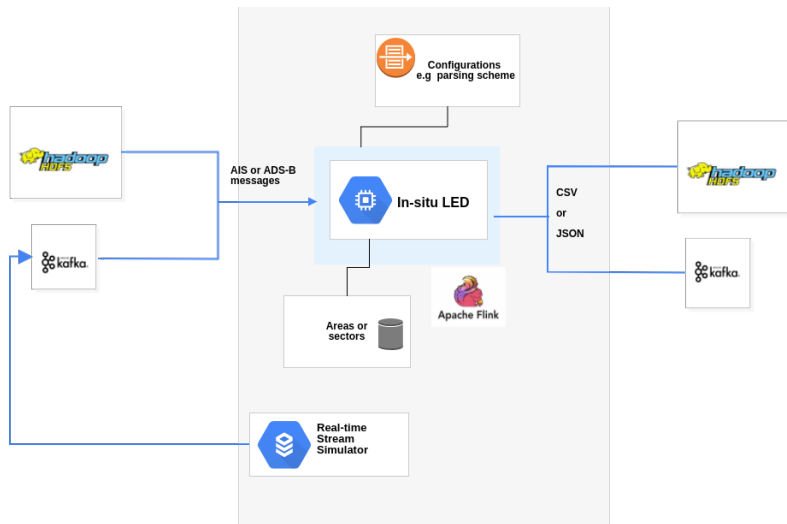
In-situ (LED Component) [1]

Overview

---

[1]github.com/ehabqadah/in-situ-processing-datAcron

In-situ LED

Functionalities

- Trajectory enrichment: computing trajectory statistics such as min/max/mean/var speed.
- Monitoring of AIS messages against areas information (detected areas)
- Derivation of low-level events for area entering and leaving

Real-time stream simulator

- ► Read raw AIS File to generate a reply stream
- ► Reconstruct the trajectories and use the actual time difference between the points to replay origin stream
- ► *delayScale* parameter to scale up/down the out stream rate

Output CSV scheme (29 fields)

```
timestamp,id,longitude,latitude,speed,
heading,msgErrorFlag,turn,course,status,
NumberOfPoints,AverageDiffTime,LastDiffTime,
MinDiffTime,MaxDiffTime,MaxSpeed,MinSpeed,
AverageSpeed,VarianceSpeed,MinLong,
MaxLong,MinLat,MaxLat,MinTurn,MaxTurn,
MinHeading,MaxHeading,isChangeInArea,
detectedAreas
```

.

Deployment

- From deployment on a single VM machine

- To deployment on the YARN datAcron cluster (10 *nodes* X 8 *cores*)

In-situ deployment on datAcron YARN cluster


```bash
#!/bin/bash

# YARN Cluster configs

numberOfTaskManagers=10
memoryPerTaskManager=15360
processingSlotsPerTaskManager=8

# Pull new code changes

git pull

# Build the project

mvn clean package

projectWorkDir=$(pwd)
.
.
.
.
.
jarFile=$(find $projectWorkDir/target/in-situ-processing*.jar)

jobName="in-situ-p:$parallelism-tm:$numberOfTaskManagers"

# Submit a Flink program to the YARN cluster

./bin/flink run -m yarn-cluster -yn $numberOfTaskManagers -ytm $memoryPerTaskManager -ys
         $processingSlotsPerTaskManager  -ynm  $jobName $jarFile > deployOnYarn.log &
```
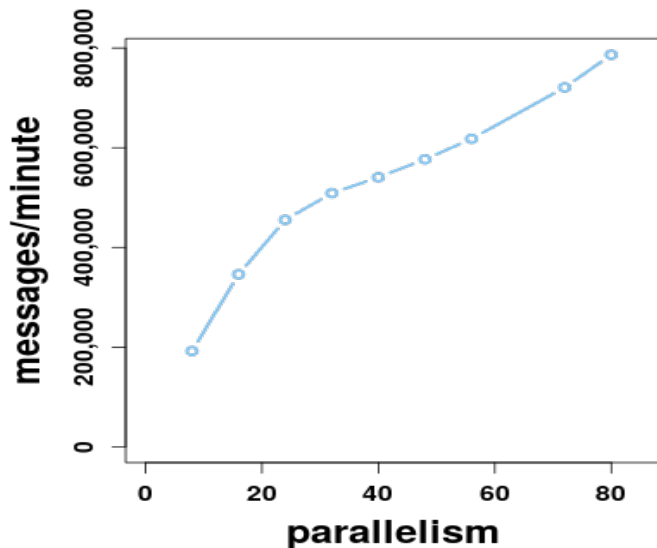
$ bash scripts/deployOnYarn.sh

In-situ Performance

Throughput on datAcron YARN cluster

Conclusion

LED in-situ processing resources/CPU consumption and delay can be neglected