

In-situ (LED Component) - WP1

Michael Mock * Ehab Qadah *

*Fraunhofer IAIS, Germany

Friday 12th January, 2018

Outline

datAcron Architecture

Overview

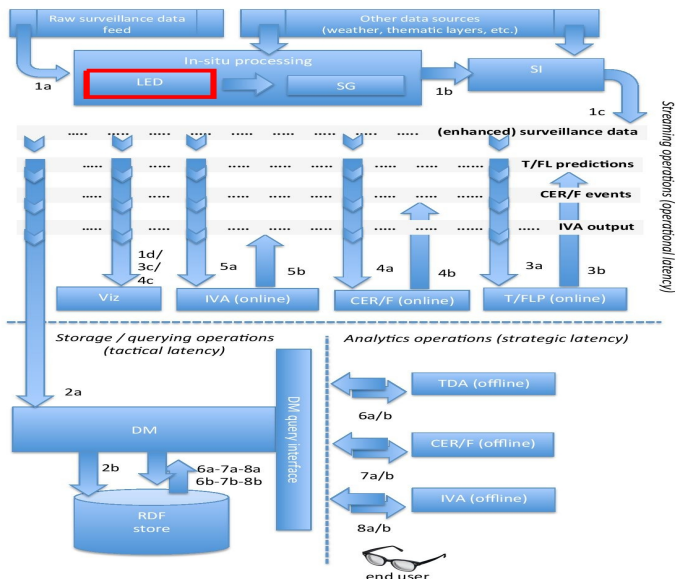
Output Scheme

Deployment and Integration

Performance on YARN cluster

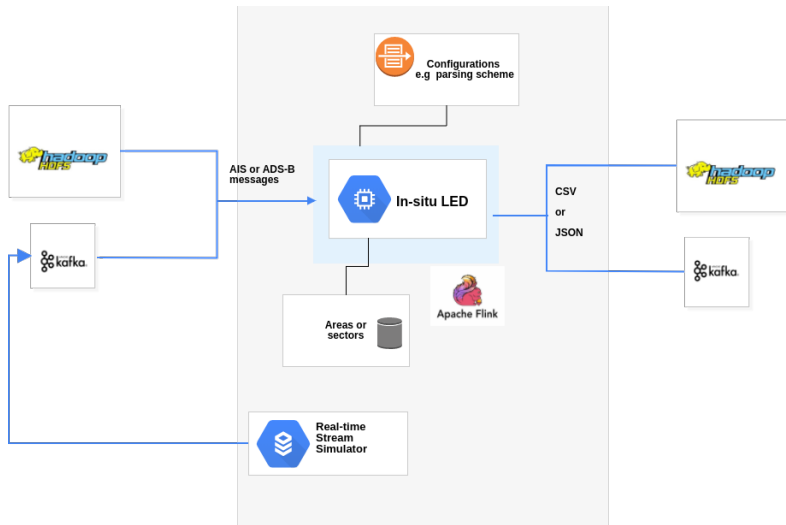
Conclusion

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 In-Situ
- ▶ -e.g., min/max/mean/var speed, acceleration
- ▶ -Deriving events against context information
- ▶ -Monitoring of ADS-B messages against sector information
- ▶ -Derivation of high-level events for sector entering and leaving
- ▶ -High-level events will be provided as real-time stream for applications

Real-time stream simulator

- ▶ Given a set of k real-time streams of events
 $S = \{s_1, s_2, \dots, s_k\}$.
- ▶ Each stream $s_i = \langle e_1, e_2, \dots, e_t, \dots \rangle$ is an evolving time-ordered sequence of events.
- ▶ Each event is defined as a tuple of attributes
 $e_t = (type, \tau, id, a_1, a_2, \dots, a_n)$ where $type \in \Sigma$ (i.e., event types).
- ▶ A user-defined pattern (i.e., complex event of interest) P expressed as sequence of event types.
- ▶ Goal: the main objective is to predict the pattern P completion with certain probability in the future over each stream s_i given the current time event e_t .

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

migration from VM version to true cluster (including cluster details
10 nodes, 8 cores each)

In-situ deployment on datAcron YARN cluster

```
#!/bin/bash
set -x

# YARN Cluster config
numberOfWorkers=10
memoryPerTaskManager=15360
processingSlotsPerTaskManager=8

# Pull new code changes
git pull

# Build the project
mvn clean package

# Deploy to YARN cluster
projectWorkDir=$(pwd)

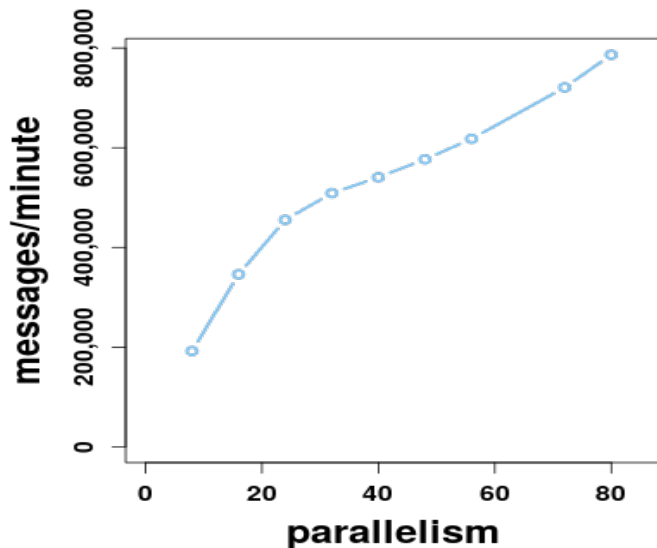
jarFile=$(find $projectWorkDir/target -name "in-situ-processing*.jar")

jobName="in-situ-p:parallelism-$numberOfWorkers"

# Submit a Flink program to the YARN cluster
./bin/flink run -n yarn-cluster -yn $numberOfWorkers -ytm $memoryPerTaskManager -ys $processingSlotsPerTaskManager -yjm $jobName $jarFile > deployOnYarn.log
```

In-situ Performance

Throughput on datAcron YARN cluster



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