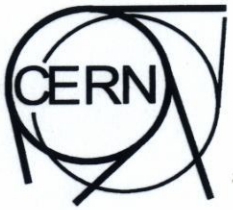
Proposal for a LVL2 Trigger Quality and Benchmarking Environment

Julius Hrivnac
and Matthias Sessler



Trigger Algorithms

- should perform well for physics (efficiency vs. rate)
- should minimise hardware requirements

=> optimal compromise between physics performance and use of system resources desired

Today

- **Trigger algorithms tested in two different environments:**
  - **quality tested in ATRIG**
  - **execution time measured in standalone C programs**
- **LVL2 testbed reference software aims to be as close as possible to real trigger, therefore no quality studies with "truth" information foreseen**

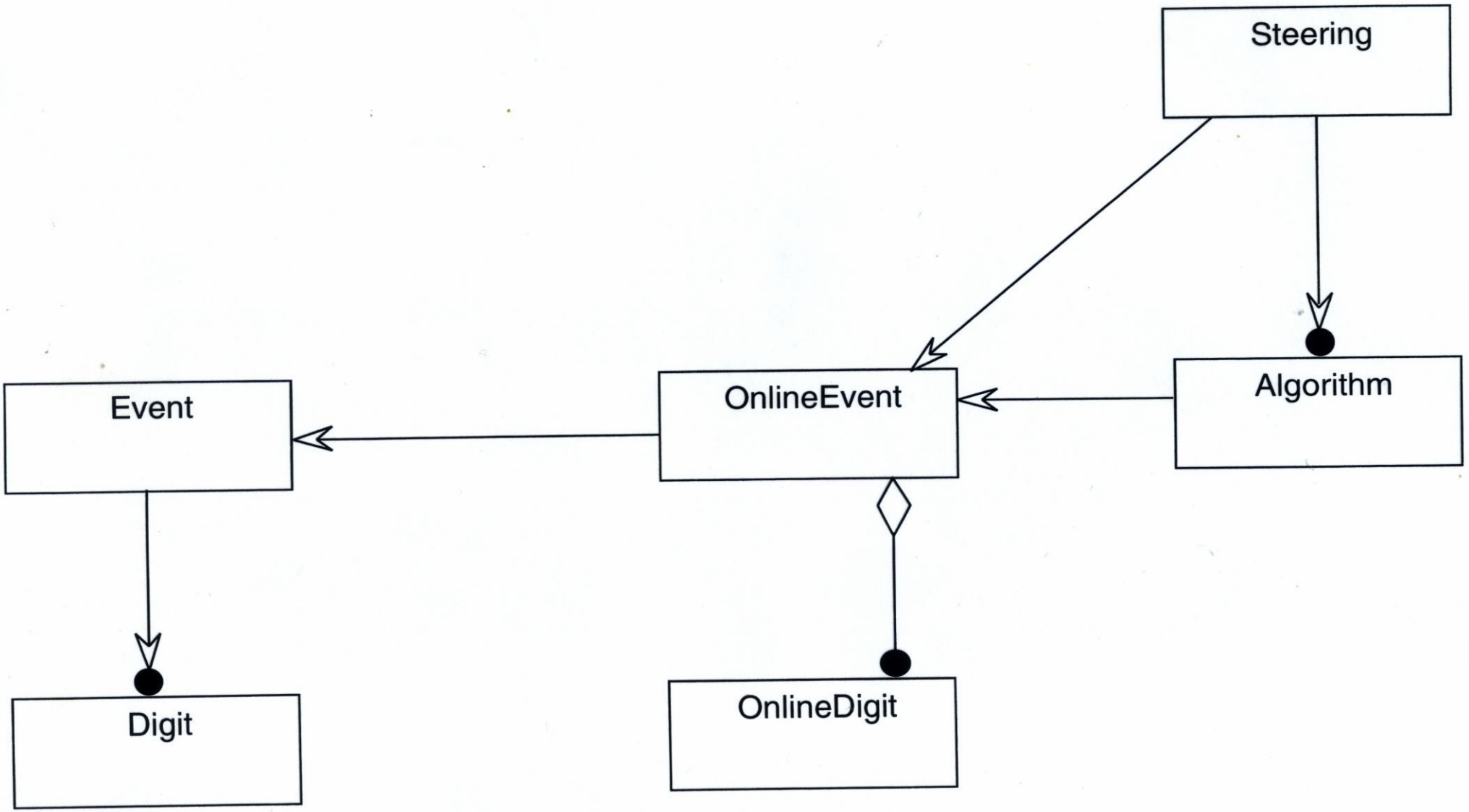
=> difficult to quantify algorithm speed-quality tradeoff

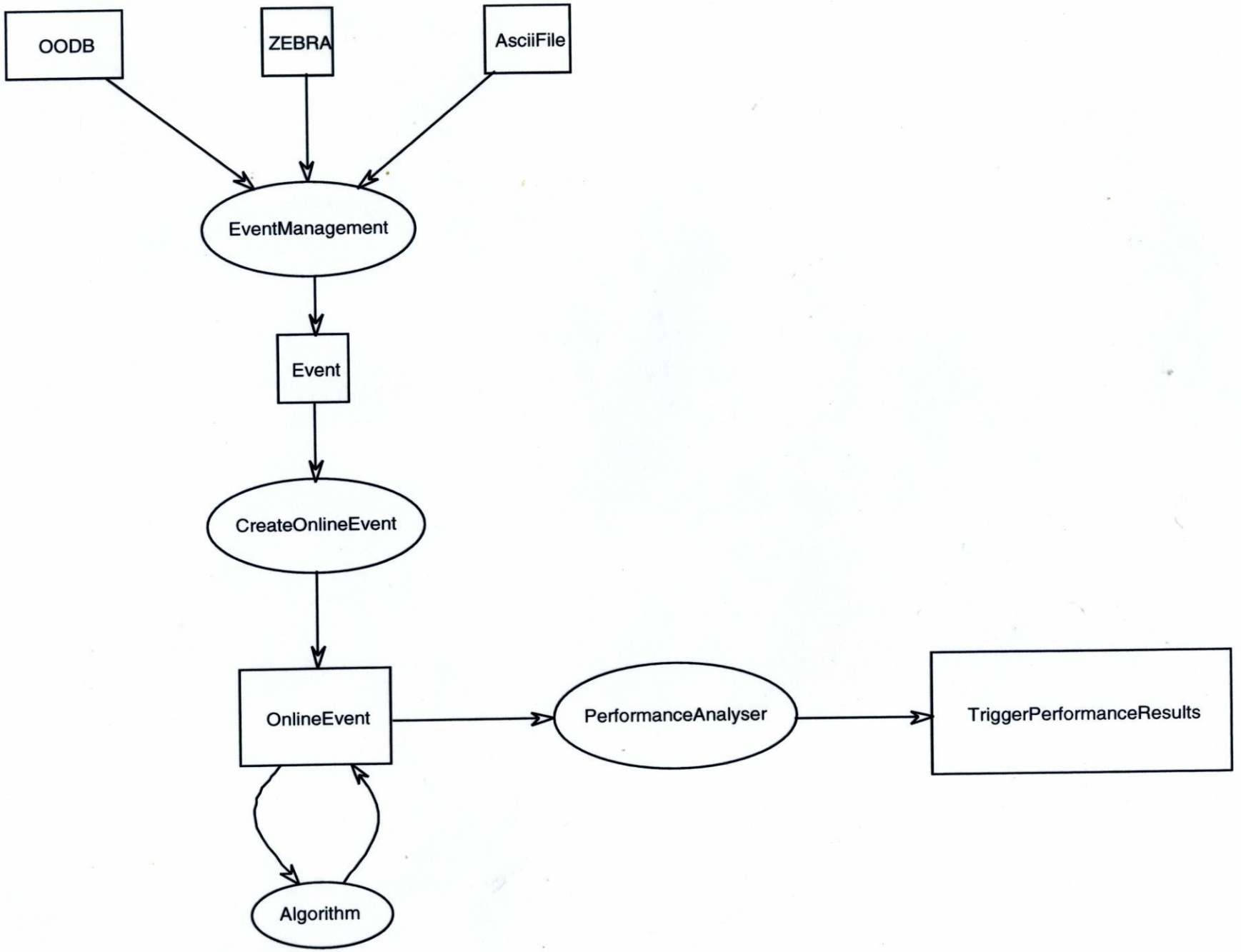
Proposal

- combine LVL2 Trigger Benchmarking and full analysis of Quality in one framework
- be prepared for different input (OODB, ZEBRA, ASCII-File)

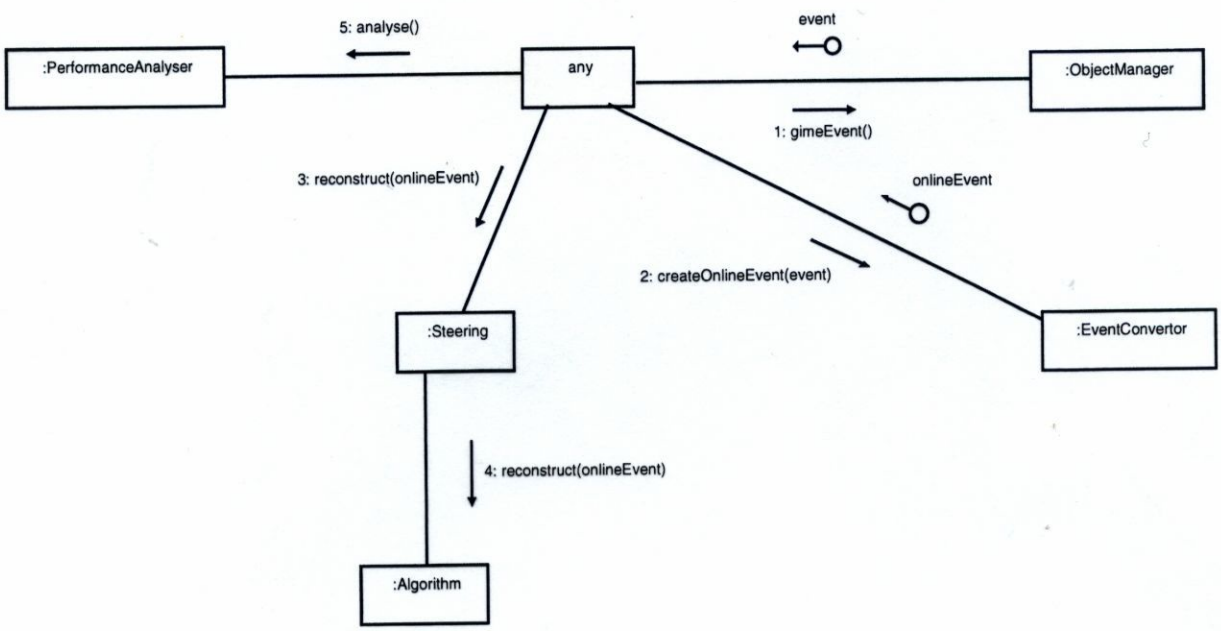
=> **Implication:**

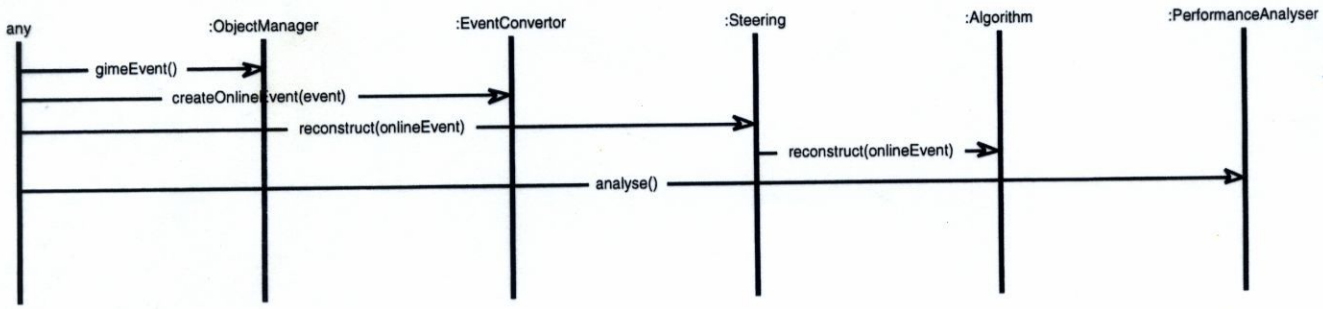
- "truth" information (including particle kinematics etc.) needed, but algorithm must not be loaded with it

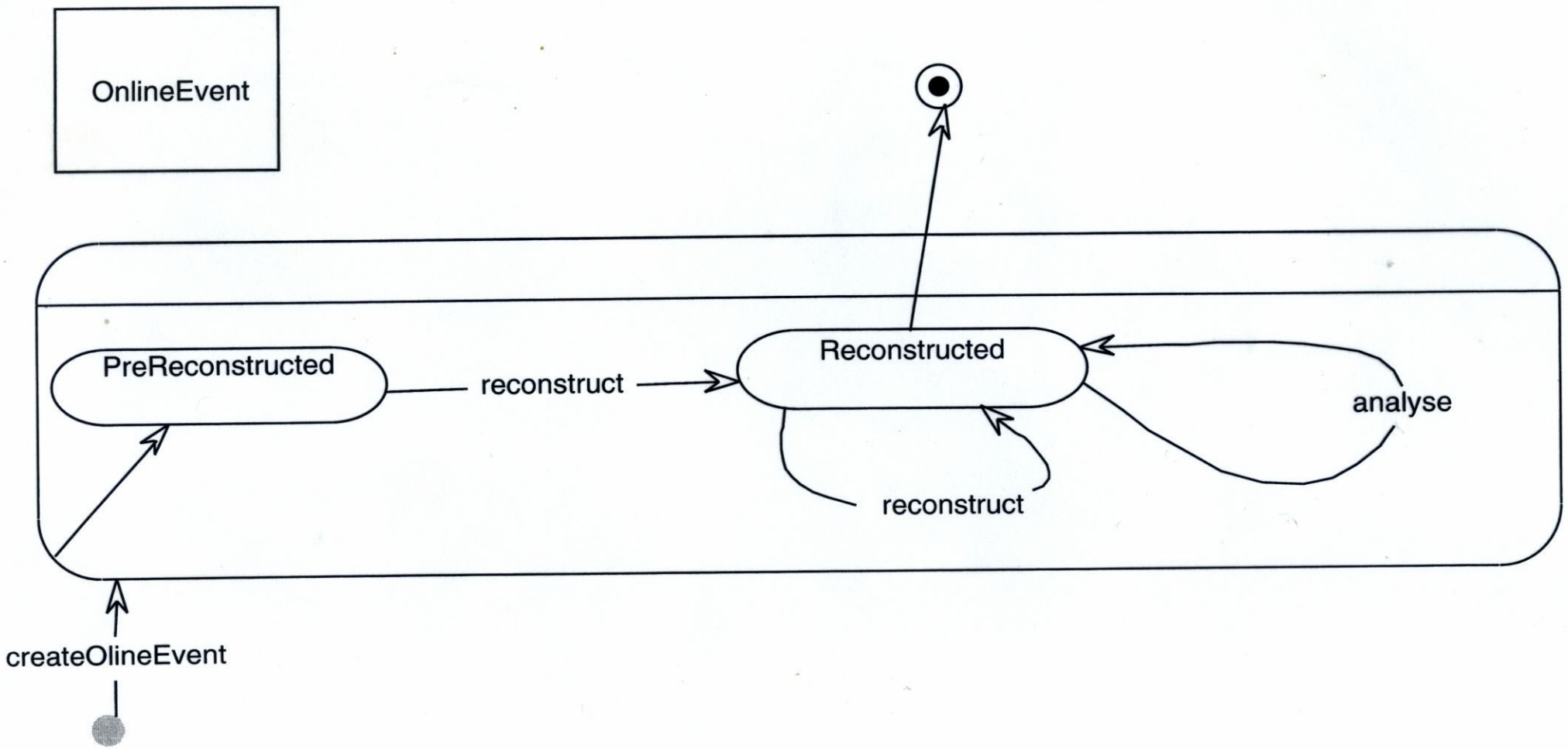

Steering
Event
OnlineEvent
Algorithm
Digit
OnlineDigit


OODB
ZEBRA
AsciiFile
EventManagement
Event
CreateOnlineEvent
OnlineEvent
PerformanceAnalyser
TriggerPerformanceResults
Algorithm


(Context)
OnlineEvent
(Scenario Name)
Reconstruct and Analyse Results.


:PerformanceAnalyser
5: analyse()
any
event
:ObjectManager
1: gimeEvent()
3: reconstruct(onlineEvent)
onlineEvent
2: createOnlineEvent(event)
:Steering
:EventConvertor
4: reconstruct(onlineEvent)
:Algorithm


any
:ObjectManager
:EventConvertor
:Steering
:Algorithm
:PerformanceAnalyser
gimeEvent()
createOnlineEvent(event)
reconstruct(onlineEvent)
reconstruct(onlineEvent)
analyse()


OnlineEvent
PreReconstructed
reconstruct
Reconstructed
analyse
reconstruct
createOlineEvent