Modeling a Distributed Intrusion Detection System Using Collaborative Building Blocks

L.A. Gunawan*, M. Vogel◊, F.A. Kraemer*, S. Schmerl◊, V. Slåtten*,
P. Herrmann*, H. König◊

*Department of Telematics, NTNU, Norway
◊Computer Science Department, BTU Cottbus, Germany

NTNU
Innovation and Creativity

Engineering Distributed Systems

- difficult task [Jennings01]
  - interdependency
  - reconfiguration

distributed Intrusion Detection System (IDS)

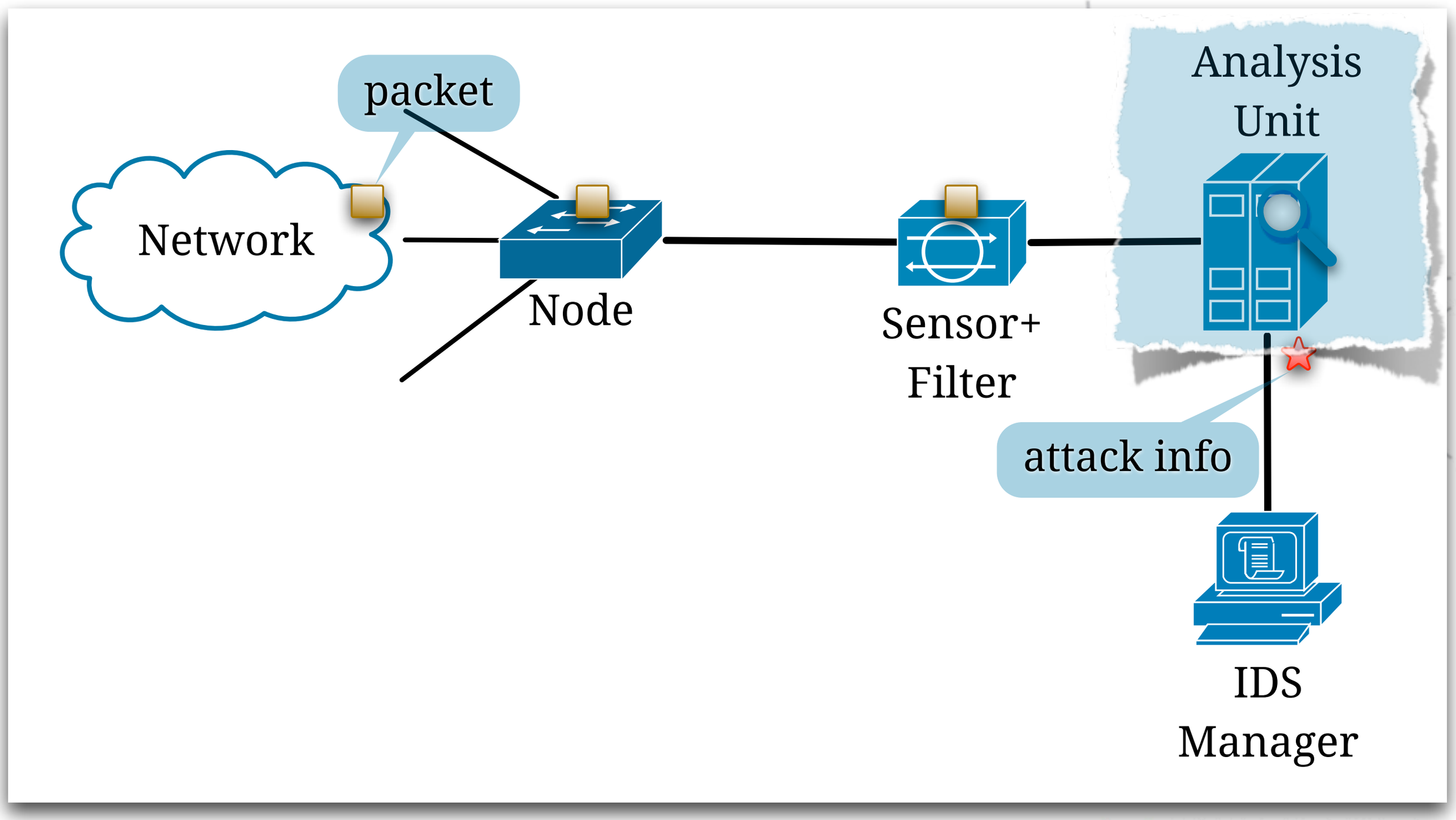
using

Collaborative Building Blocks (SPACE + Arctis)

Collaborative Building Blocks (SPACE + Arctis)

- SPACE + Arctis
  - graphical models, UML 2.x
  - formal
  - collaborative, high degree of reuse

IDS Components

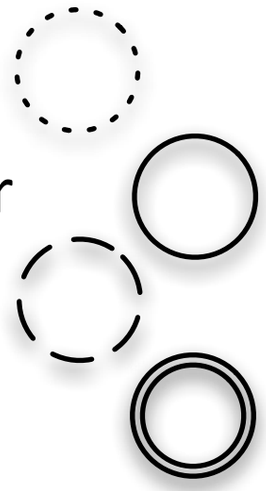


Detection Techniques

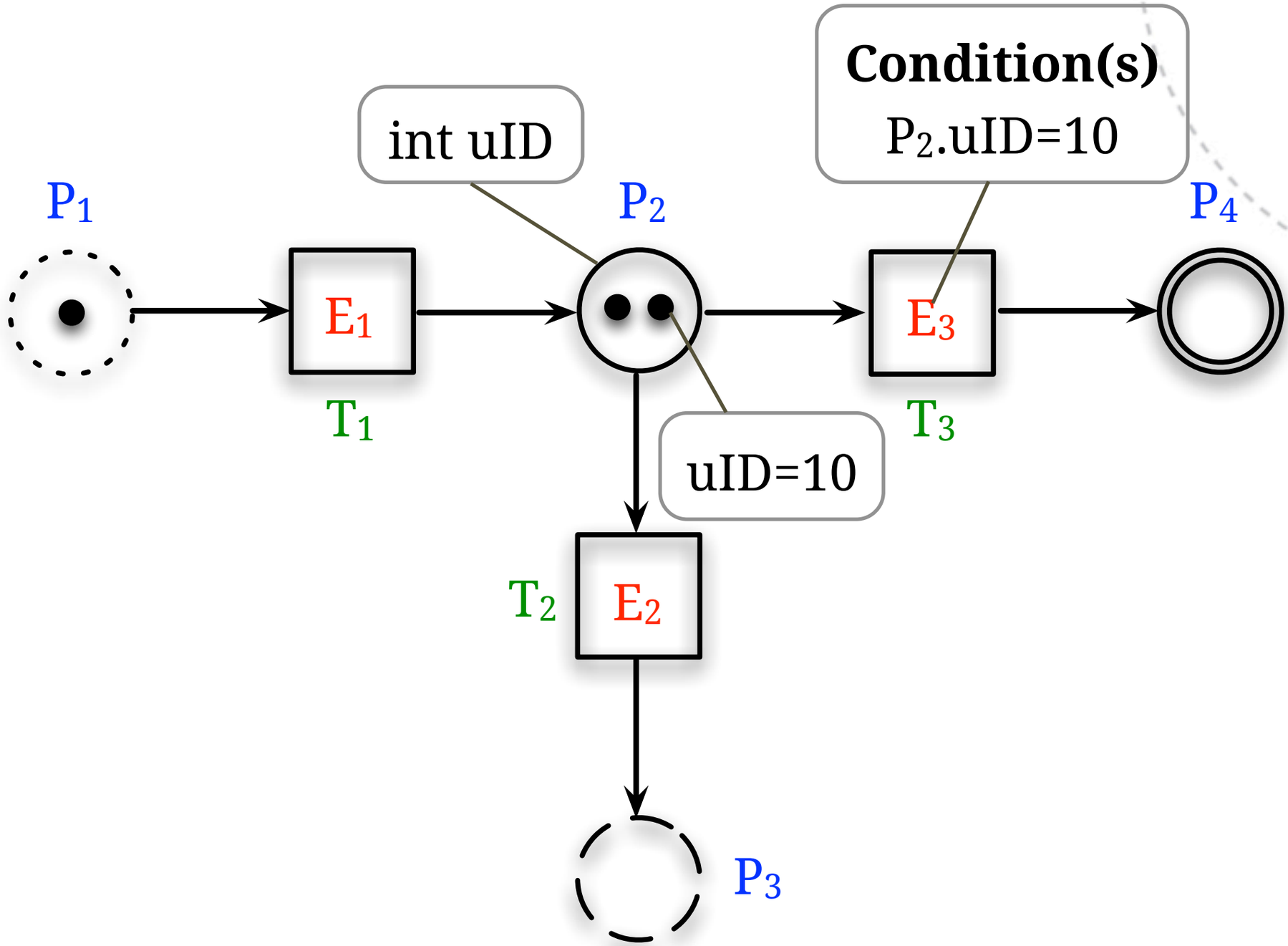
- Signature-based
  - attack patterns
  - pro: high accuracy
  - con: reactive
- Anomaly detection-based
  - abnormal behavior
  - e.g., statistical anomaly, artificial intelligence, data mining
  - pro: proactive
  - con: «normal»?

Event Description Language (EDL) Signatures

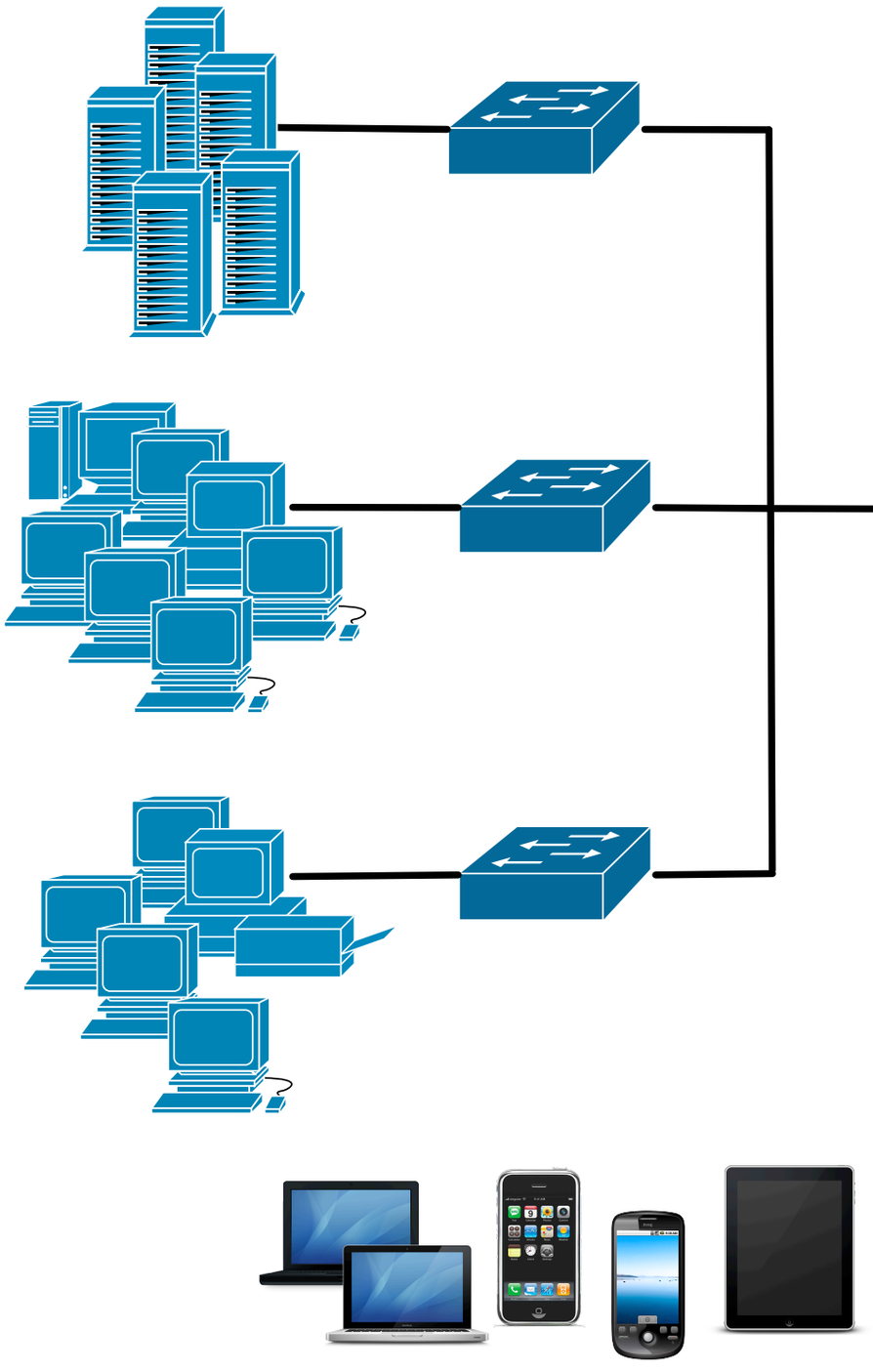
- Places
  - system states of an attack
  - four types:
    - ✓ initial
    - ✓ interior
    - ✓ escape
    - ✓ exit
- Transitions
  - state changes triggered by audit event type
- Directed edges →
- Tokens •
  - ongoing attacks

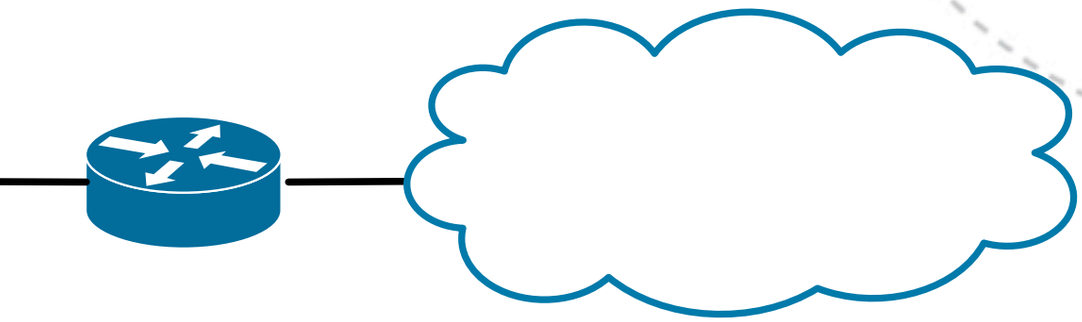


EDL Signature - an Example

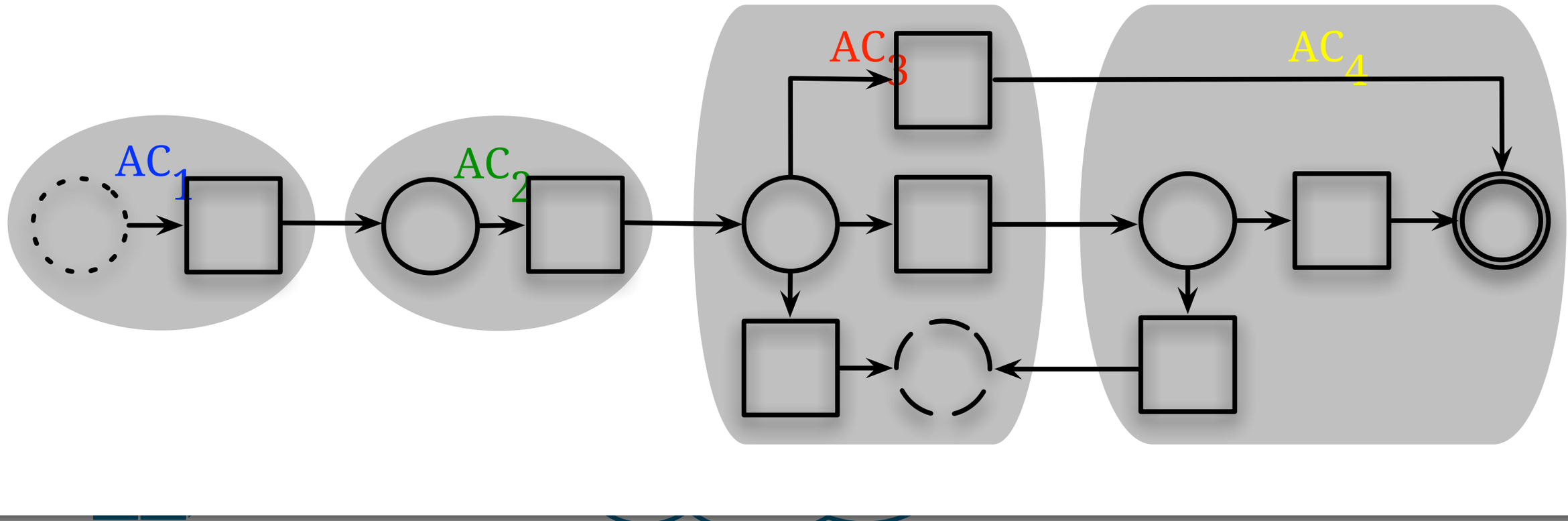


Distributed Analysis - Why?



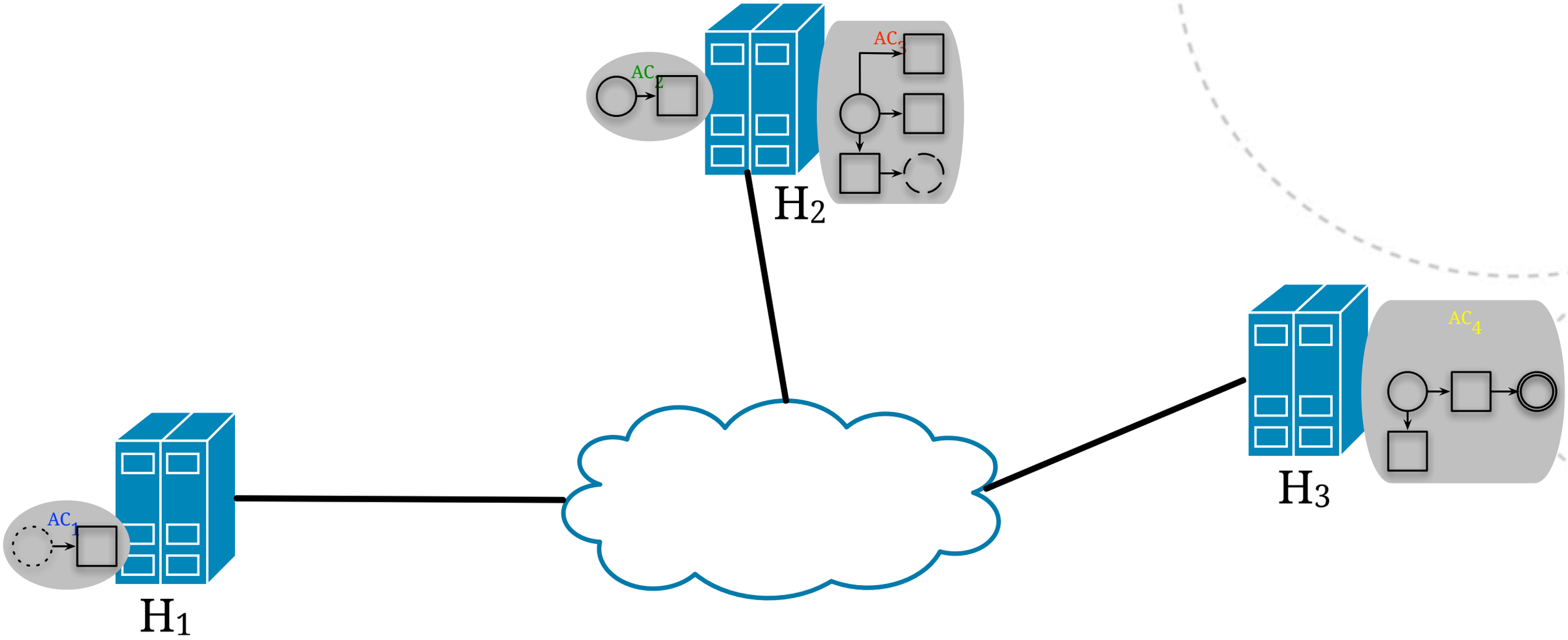


Distributed Analysis



- Atomic clusters

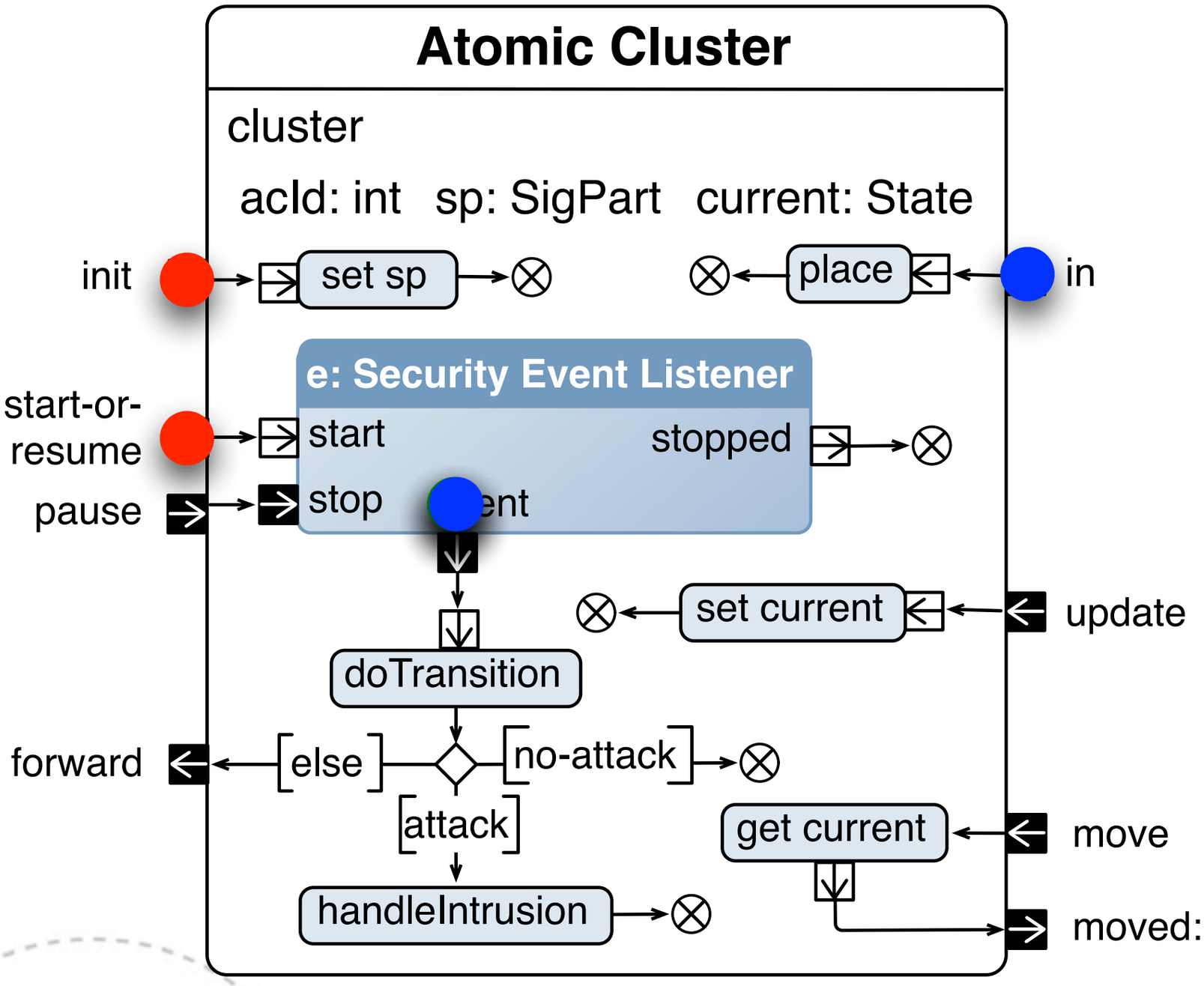
Distributed Analysis + Reconfiguration

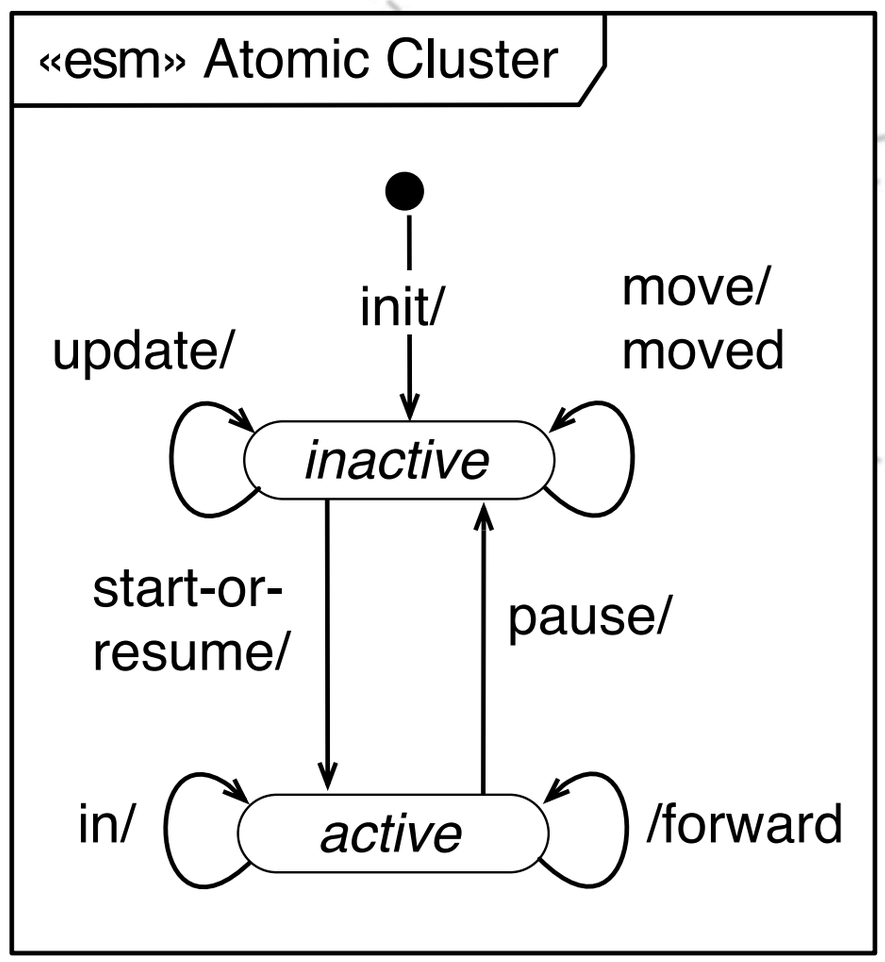


- Overload situation
  - move an atomic cluster

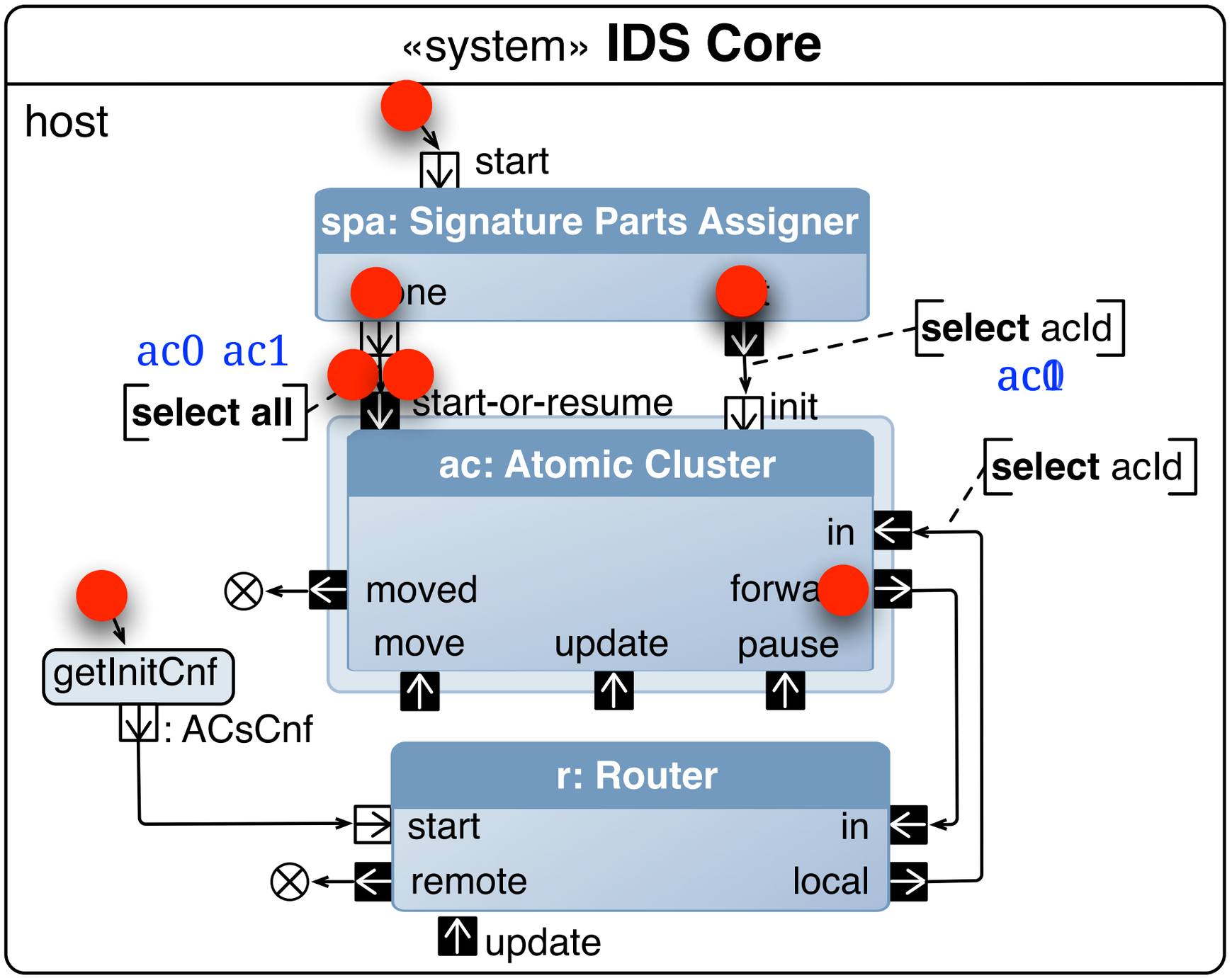
Modeling IDS Core Functionality - Atomic Cluster

- All ACs are executed in a single host





IDS Core System



Atomic Cluster Manager

host

hostId: String m: MoveInfo

otherhosts [*]

move

move Completed

start

set m

select all : ac.active

pi: Pause Handler

reqPause

paused

resumed

reqResume

pr: Pause Handler

remotePause

remoteResume

spa: Signature Parts Assigner

start

finished

out

select acId

init

pause

ac: Atomic Cluster

start-or-resume move

moved

in

select all : id **in** r.localList

forward update

select acId

setState

completed

ms: Move Handler

move

newCnfLocal: ACsCnf

getState

in

r: Router

start

local remote update

mr: Move Handler

moveRemote

newCnfRemote

mt: Move Handler

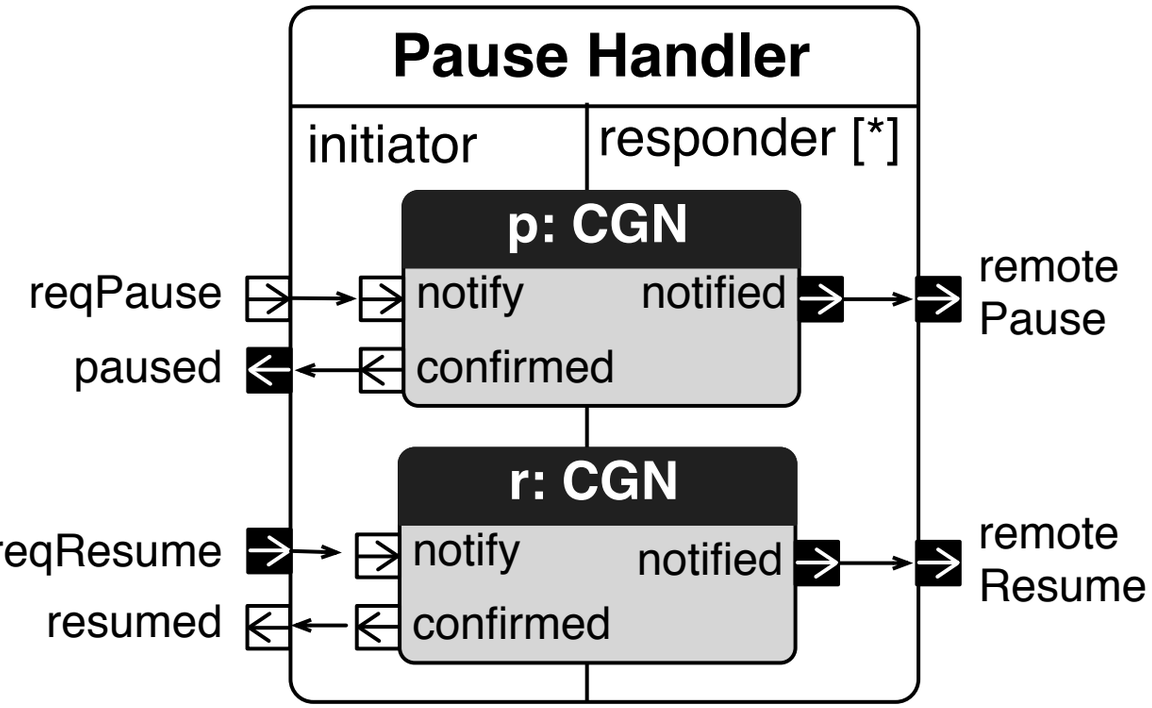
newCnfRemote

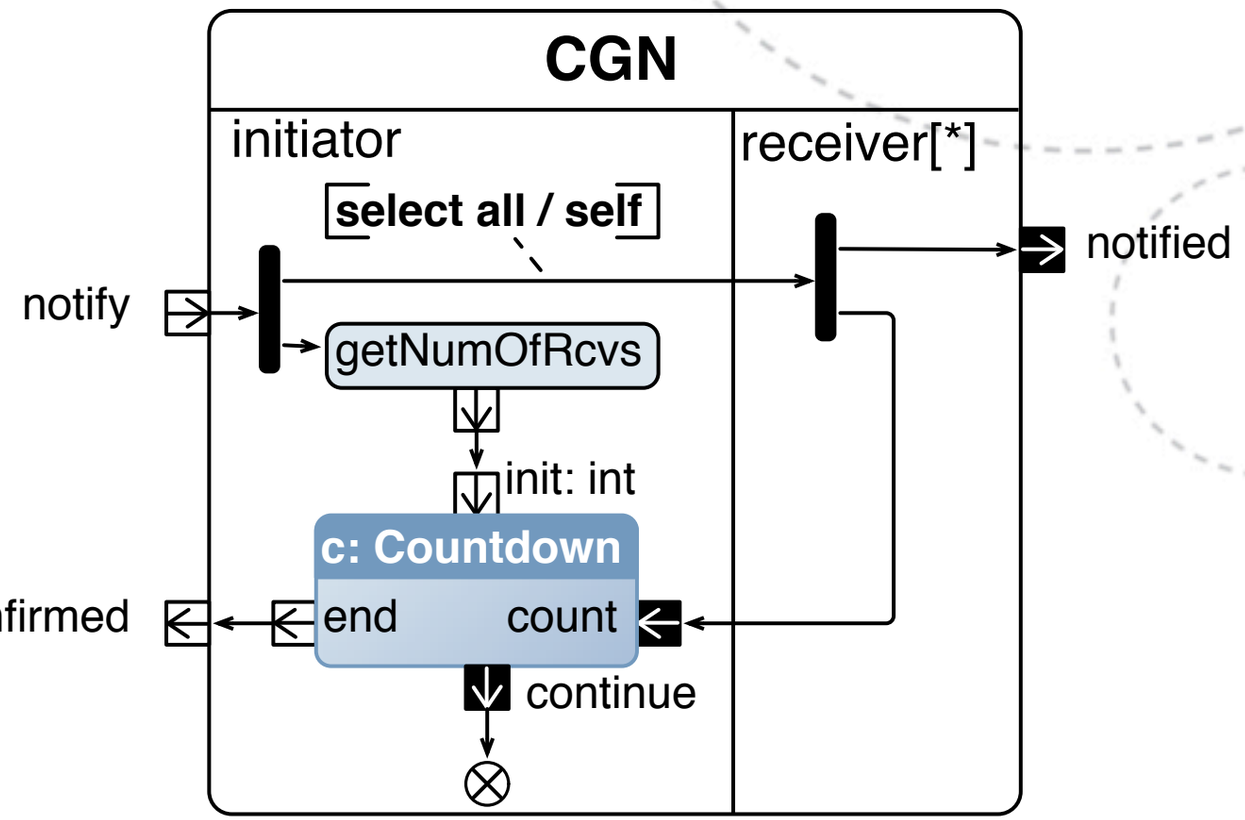
select acId

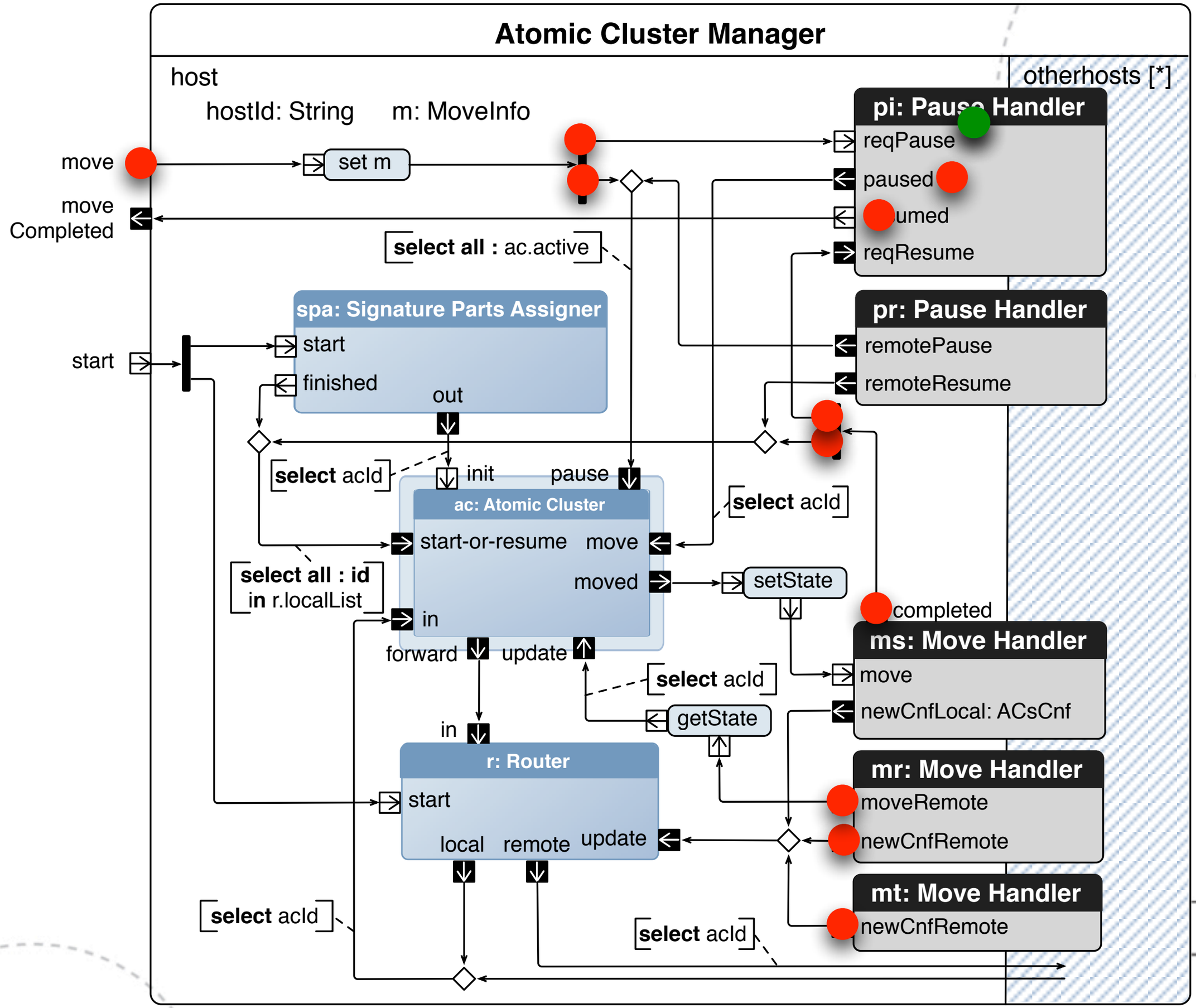
select acId

select acId

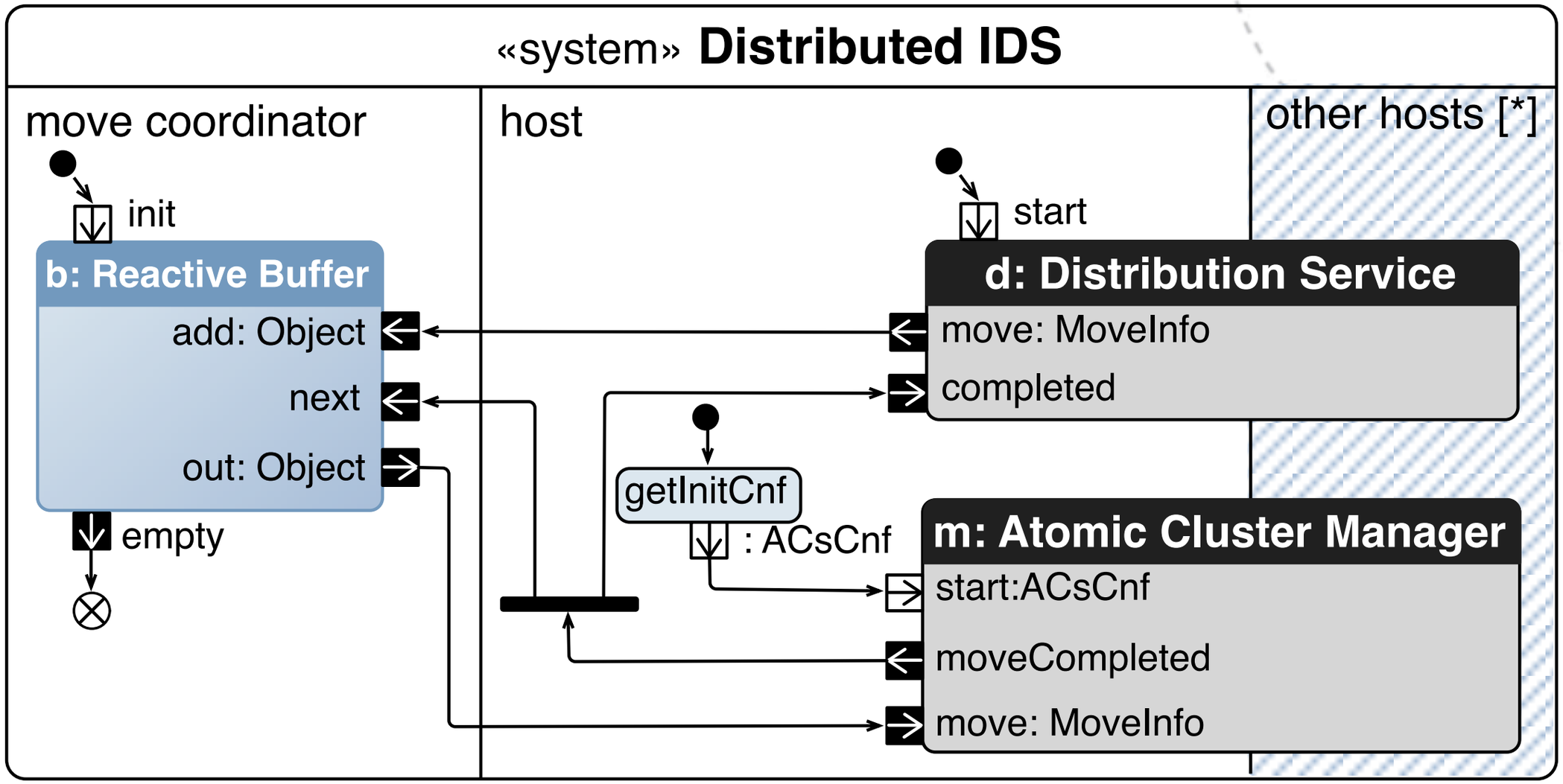
Group Communication






Atomic Cluster Manager
host
hostId: String
m: MoveInfo
otherhosts [*]
move
move Completed
set m
pi: Pause Handler
reqPause
paused
umed
reqResume
select all : ac.active
spa: Signature Parts Assigner
start
finished
out
pr: Pause Handler
remotePause
remoteResume
start
select acId
init
pause
ac: Atomic Cluster
start-or-resume
move
moved
in
select all : id
in r.localList
forward
update
select acId
getState
setState
completed
ms: Move Handler
move
newCnfLocal: ACsCnf
in
r: Router
start
local
remote
update
mr: Move Handler
moveRemote
newCnfRemote
mt: Move Handler
newCnfRemote
select acId
select acId
select acId

The Distributed IDS Model



Validation

- cTLA (compositional Temporal Logic of Action)
- model checker
  - hidden from user
  - simulation
  - error trace
- incremental verification -- block by block (ESM)
  - small state space
- challenge: group communication type block

Evaluation

- Reuse
  - other cases > 70%
- metrics:
  - number of nodes & edges (≈ line of codes)
  - two types of experts: IDS, communication

Metrics

Building Block	Complexity n	Reused	Experts
Distributed IDS [Fig. 6]	39		C
Reactive Buffer	39	✓	
Distribution Service	50		C
One	4	✓	
Load Monitoring	37		C
Component Monitor	10		C
Find Host With Minimal Load	32		C
Collected Group Response	26	✓	
Countdown	18	✓	
Atomic Cluster Manager [Fig. 7]	86		C
Router	29		I + C
Distribute Signature Parts	23		I + C
Iterator	22	✓	
Atomic Cluster	52		I + C
Security Event Listener	14		I + C
Pause Handler	13		C
Conf. Group Notif. [Fig. 8(a)]	21	✓	
Countdown	18	✓	
Move Handler [Fig. 8(b)]	27		C
Conf. Group Notif. [Fig. 8(a)]	21	✓	
Countdown	18	✓	
Total complexity	599		

Building Block	Complexity n	Reused	Experts
IDS Core [Fig. 4]	34		I + C
Distribute Signature Parts	23		I + C
Iterator	22	✓	
Atomic Cluster [Fig. 5(a)]	52		I + C
Security Event Listener	14		I + C
Router	29		I + C
Total complexity	174		

NTNU
Innovation and Creativity

Evaluation

- Reuse
  - other cases > 70%
- metrics:
  - number of nodes & edges (≈ line of codes)
  - two types of experts: IDS, communication

- reduction of the overall development effort: ± 50%

Current/Future Work

- Verification of IDS specific properties:
  - All AC instances in all hosts are in state inactive, when a handover of analysis function is in progress
  - An EDL token is never routed to an inactive AC instance
  - Only one move is in progress at a time
  - Every AC is assigned to at most one host at all times
  - An AC is always eventually assigned to a host
- Distributed IDS model implements IDS Core model
- Real execution environment:
  - a host may crash