A day in the life of an autonomic function

P. Peloso, L. Ciavaglia

IETF 94th

draft-peloso-anima-autonomic-function-00

Motivations

Autonomics can improve network operations
Operators need unified management functions to use autonomics and gain confidence in it.

Common management functions of AF bring

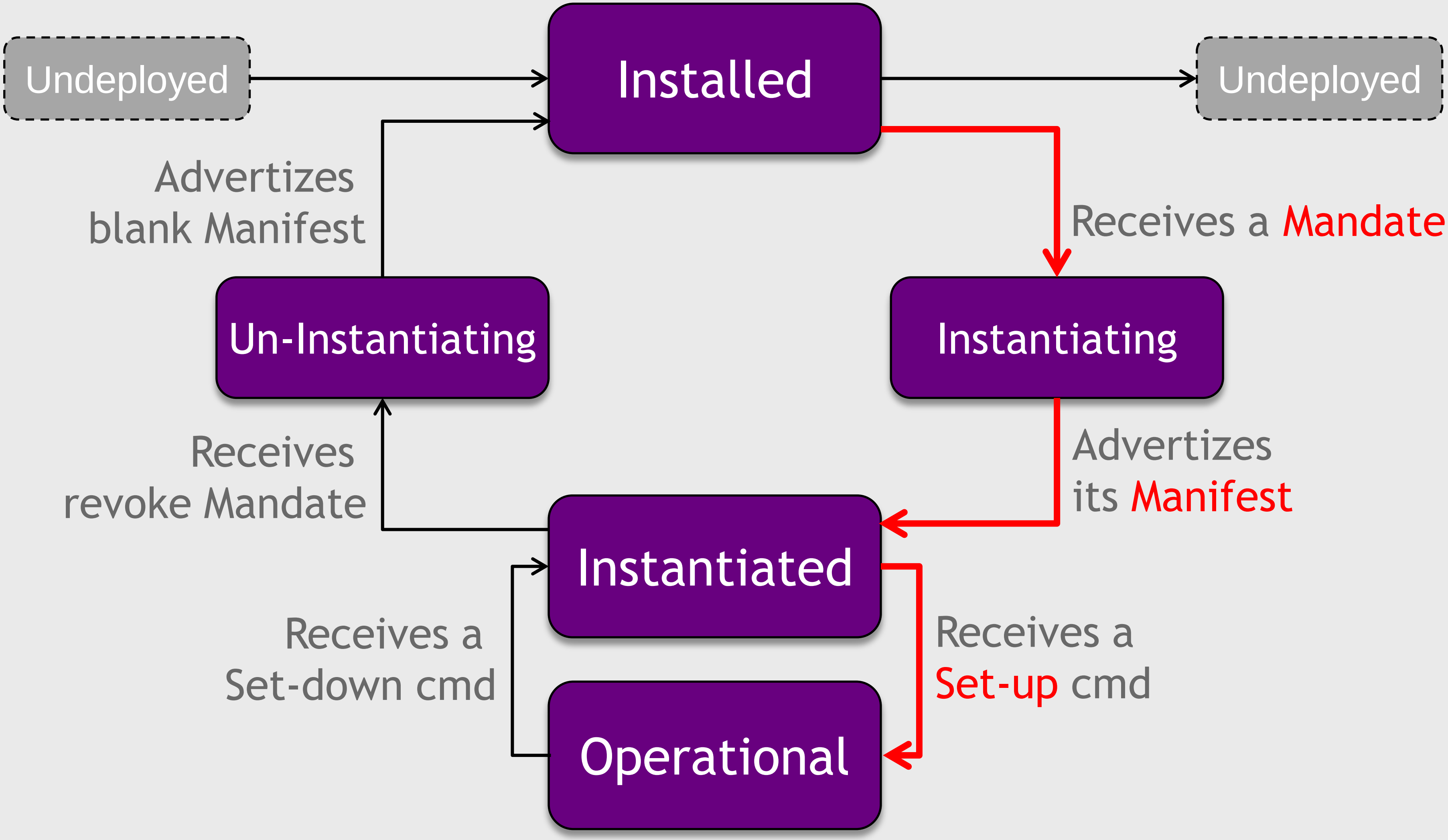
- trust in Autonomic Functions behavior
- capacity to control Autonomic Functions
- conflict avoidance mechanisms

Management of Autonomic Functions 3 phases

Installation	To dynamically install ASA to Nodes.
Instantiation	To allocate the network resources to be managed by ASA. To organize ASA in ASA domains.
Operation	To control the running of ASA. To avoid conflicts between ASAs. To share knowledge between ASAs.

Focus of today

ASA life-cycle 3 states + 2 transit ones



Undeployed

Installed

Undeployed

Advertizes blank Manifest

Receives a Mandate

Un-Instantiating

Instantiating

Receives revoke Mandate

Advertizes its Manifest

Instantiated

Receives a Set-down cmd

Receives a Set-up cmd

Operational

ASA Interactions

The Life-Cycle shows that:

Entities pertaining to the **Control of Autonomic Function** are interacting with the **ASAs** and serving **all** ASAs

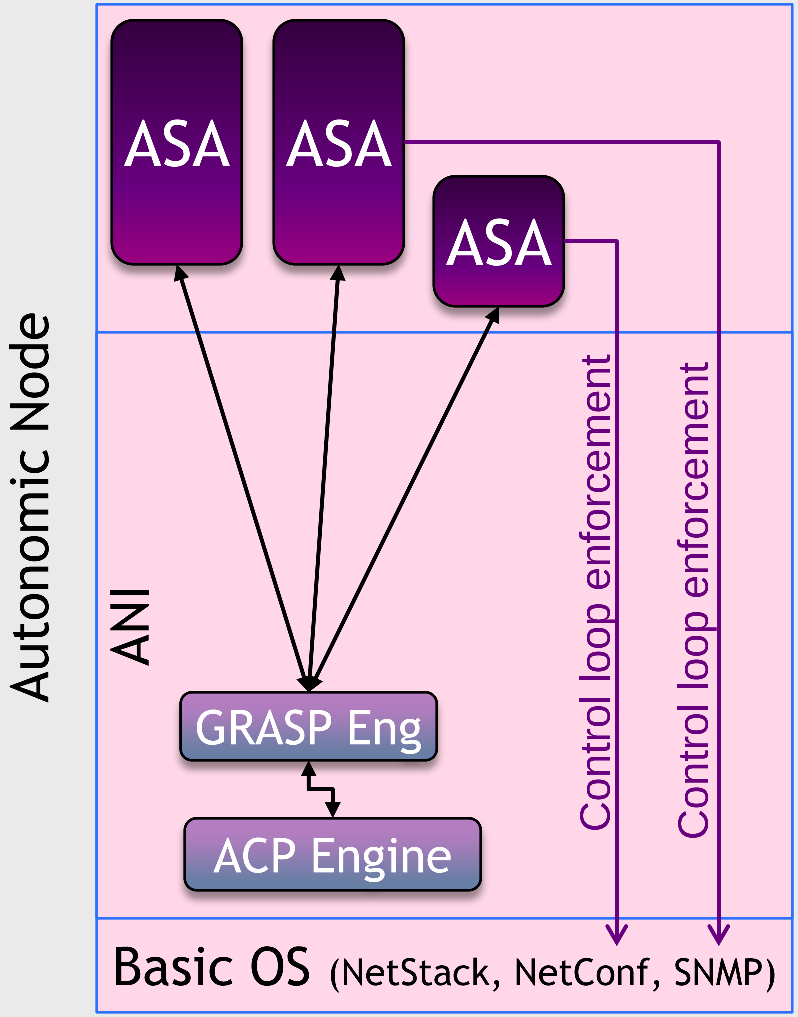
AF Mgt Function

Coordination Function

Info sharing Function

What is needed in the ANIMA ecosystem?

Autonomic Networking Infrastructure — The toolbox



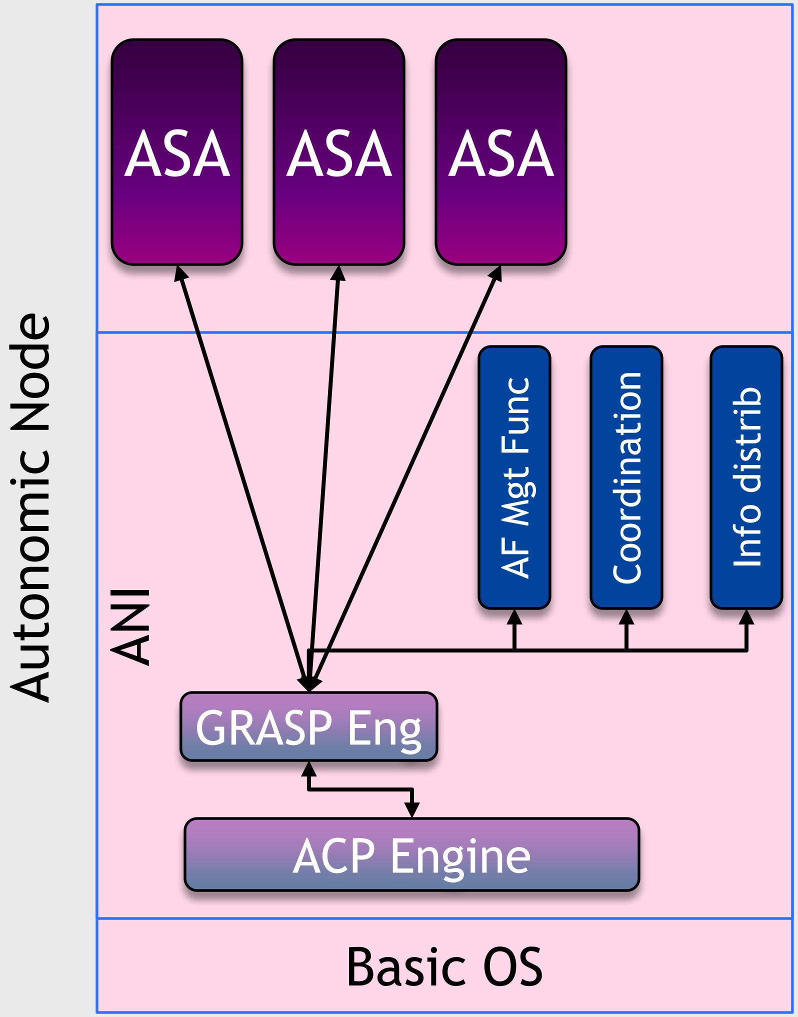
Current ANIMA picture

- GRASP preferably uses ACP
- ASA use GRASP signaling in-between them
- ASA monitor the equipment and modify its state directly using either NetConf, SNMP, call to Basic OS API…

Legend

Protocol engine

Autonomic Networking Infrastructure The toolbox



Simplest option to control ASAs:

- Same as before
- Plus use GRASP signaling between ASAs and AF Control Agents (Coordination, AF Mgt, Info Distribution)
- Hence multiple type of GRASP clients

Legend

- Protocol engine
- ANI function (serving ASAs)

Nota

The functions controlling autonomic functions NEED NOT being instantiated in each Node

Actually there even likely being instantiated in servers part of the ACP but not on network equipments like routers or switches.

Minimal control of ASA

Control when it runs

(and how it runs)

Know what it does to the network

Decide which equipments are under the ASA control

(Or vice-versa which ASAs control an equipment)

Control when an ASA runs

NEED

- On request Start and Stop the execution of ASA

SOLUTION

- Send a START command
- Send a STOP command

Control when an ASA runs

NEED

- On request Start and Stop the execution of ASA

SOLUTION

- Send a START command
- Send a STOP command

IN ANIMA

- Add to GRASP imperative commands type of message

Know what an ASA does to the network

NEED

- Know which network resources are modified by ASA control loop
- Know which network resources are monitored by ASA control loop

SOLUTION

- Disclose an ASA Manifest at ASA bootstrap time

Know what an ASA does to the network

NEED

- Know which network resources are modified by ASA control loop
- Know which network resources are monitored by ASA control loop

SOLUTION

- Disclose an ASA Manifest at ASA bootstrap time

IN ANIMA

- Disclose Manifest with GRASP Discovery messages

Decide which ASA control which equipment

NEED

- Give instructions to ASA during bootstrapping

SOLUTION

- Send a Mandate to ASA before end of bootstrap

Decides which ASA control which equipment

NEED

- Give instructions to ASA during bootstrapping

SOLUTION

- Send a Mandate to ASA before end of bootstrap

IN ANIMA

- Specify Intent formats compatible with Mandate and identify proper message in GRASP to convey Intent

Conclusion

Minimal control of ASA

Control when it runs

Start/Stop

Know what it does to the network

Manifest

Decide which equipments
are under the ASA control

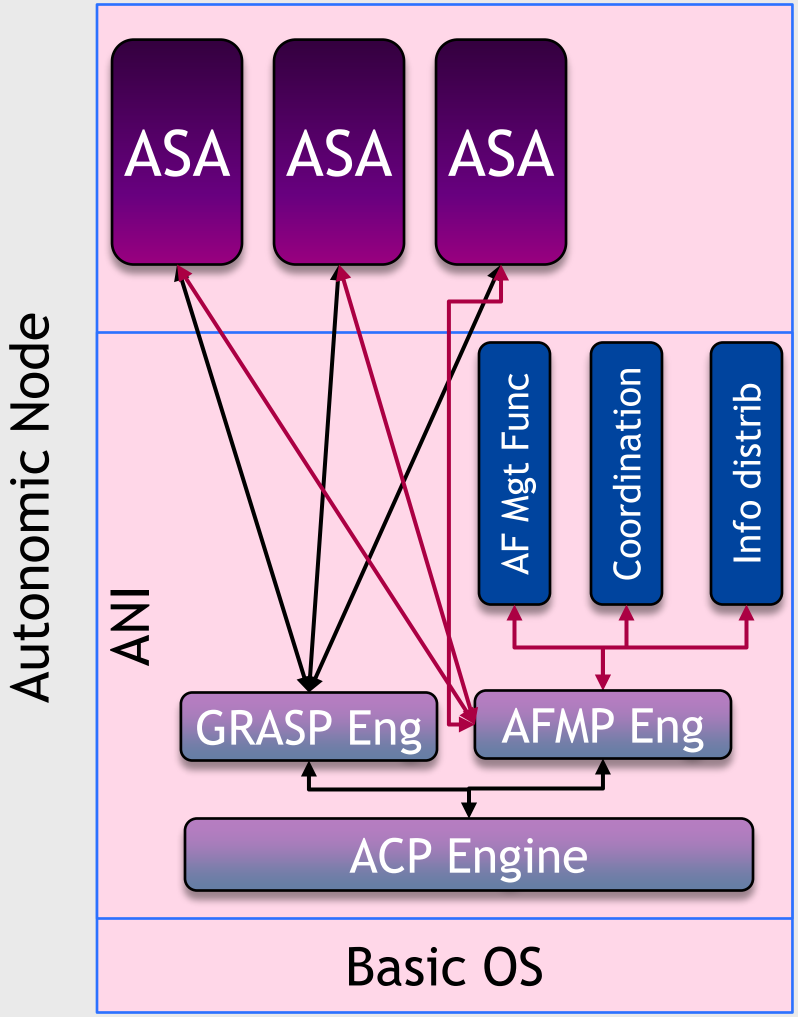
Mandate

Conclusion

Can we design a solution that oversees the operators trust in it ?

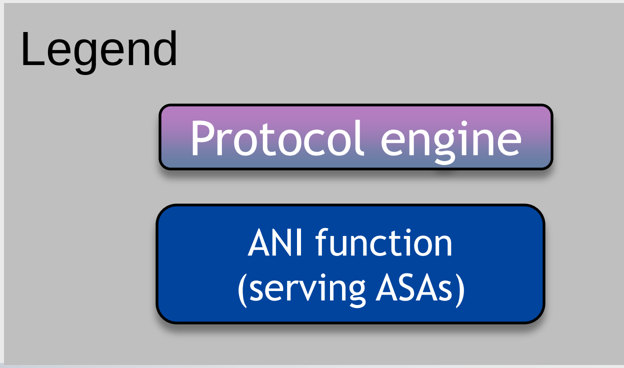
Appendix

Autonomic Networking Infrastructure The toolbox

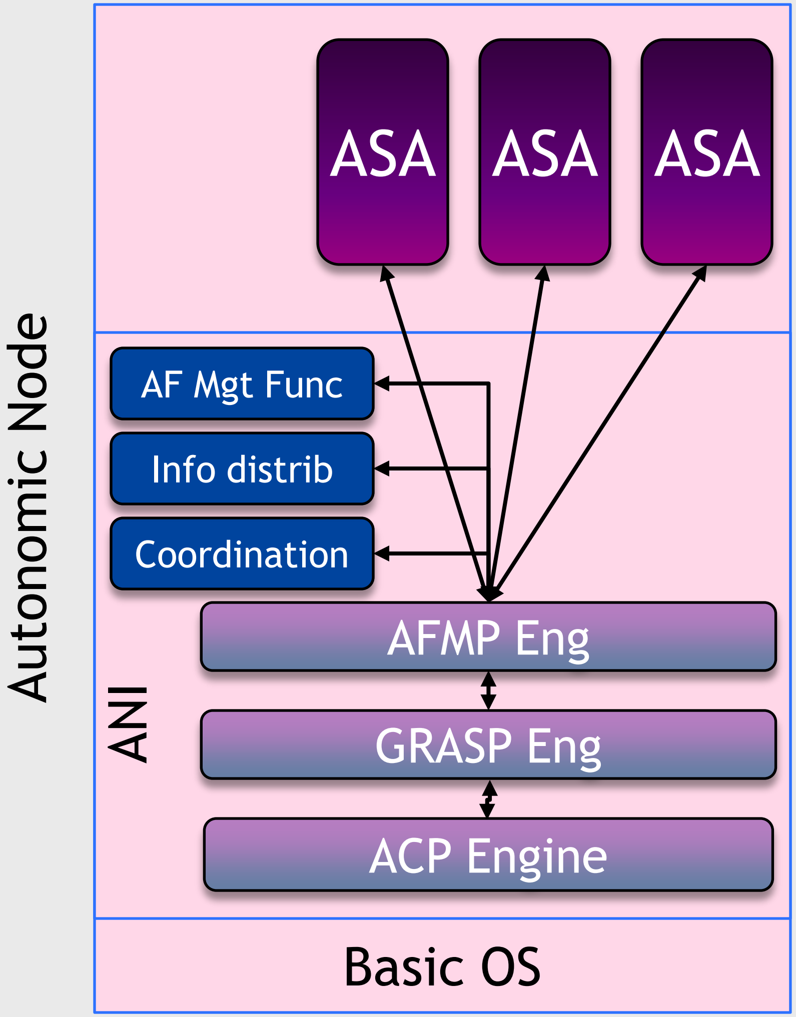


Other option

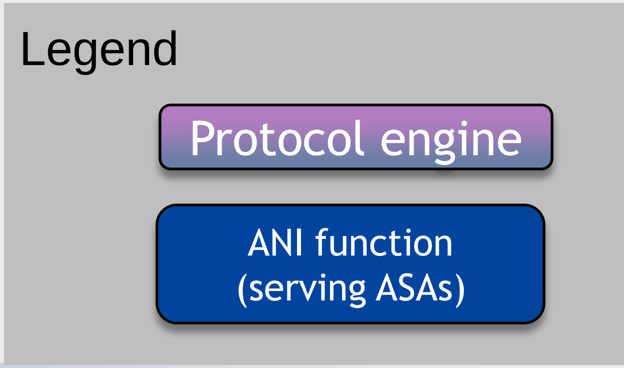
- Segmenting between ASA-ASA and ASA-ANI function



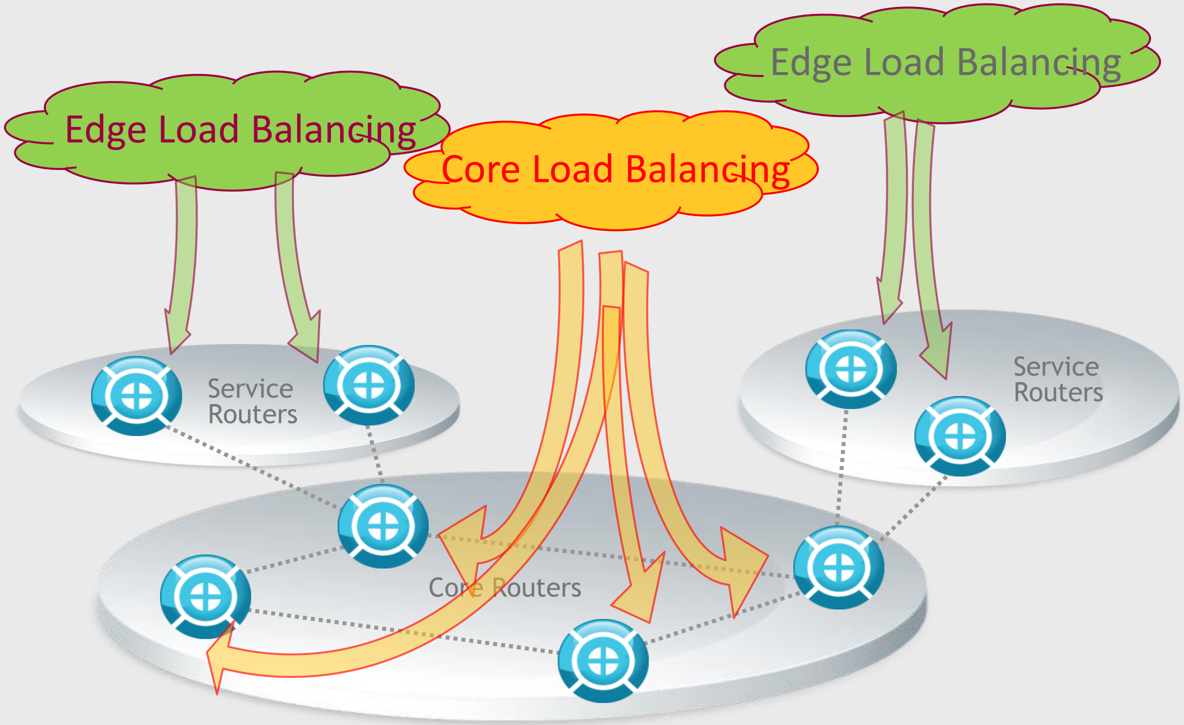
Autonomic Networking Infrastructure The toolbox



In case GRASP should absolutely not care about semantics of what is carried and roles of its clients



Deployment examples of AF


Edge Load Balancing
Core Load Balancing
Edge Load Balancing
Service Routers
Service Routers
Core Routers