Monitoring within an Autonomic Network: A GANA based Network Monitoring Framework

Anastasios Zafeiropoulos, Athanassios Liakopoulos, Alan Davy, Ranganai Chaparadza

tzafeir@grnet.gr

Greek Research and Technology Network, GRNET

Outline

- Autonomic principles in network management
- The EFIPSANS project
- The GANA architecture
  - Functional Planes
  - Decision Elements
  - Hierarchical Control Loops
- EFIPSANS Monitoring Framework
  - Role
  - Interfaces
- Scenario: Traffic Monitoring and QoS Control
- Future Work

Autonomic principles in data network management

- IP networks are increasingly becoming larger and more complex to monitor or fully manage in a cost-effective way
- Heterogeneous Environments
  - constant changes in the network topology
  - diversity of the interconnected systems
  - resources discovery and management is crucial
  - predefining the monitoring scheme can be inefficient
  - coordination of different management mechanisms
- Autonomic principles in data network management are introduced in order to address complexity of information and communication systems.
- In an autonomic environment
  - the network itself can detect, diagnose and repair failures
  - adapt its behaviour according to the network environment
  - follow policies required for improving its performance and QoS

Autonomic principles in data network management

- Vision for Autonomic Network Management
  - autonomic networks whose nodes are engineered in such a way that all the traditionally so-called network management functions defined by the FCAPS (Fault, Configuration, Accounting, Performance, Security) management framework, as well as the fundamental network functions, are designed to automatically feed each other with information and effect feedback processes among the diverse functions
- FCAPS functions have to be intrinsically in-built into node architectures apart from being part of an overall network architecture - whereby traditionally, a separate management plane is engineered separately from the other functional planes of the network.
- The functional planes of an autonomic network would need to be (re)-defined, re-factored or even merged.
- Generic Autonomic Network Architecture (GANA) that is proposed in the framework of the EU FP7-EFIPSANS IP Project

The role of EFIPSANS

- **EFIPSANS** - **E**xposing the **F**eatures in **IP** version **S**ix protocols that can be exploited / extended for the purposes of designing / building **A**utonomic **N**etworks and **S**ervices.
- The EFIPSANS-IP-Project, that started in January 2008, aims at exposing the features in IPv6 protocols that can be exploited or extended for the purposes of designing or building autonomic networks and services, as necessitated by GANA.
- The EFIPSANS's long term objective is to pursue and call for the standardization of the specified autonomic behaviour specifications of DMEs/DEs for diverse networking environments, the identified exploitable IPv6 features and the new protocol and architectural extensions that will be produced by EFIPSANS.
- Website: http://www.efipsans.org

- ETSI Industry Specification Group (**ISG**) on Autonomic network engineering for the self-managing Future Internet (**AFI**) has recently been created

The GANA architecture (1)

- Clearly separates specification issues for autonomic behaviours from implementation issues of the specified autonomic behaviours
- Review existing approaches to autonomic or “somewhat autonomic” network engineering
  - CONMan, 4D, FOCALE, ANA, Haggle
- Is generic and captures in a holistic way what should be autonomic in a node and what should not, including the architecture, diversity and multiplicity of DEs of an autonomic node.
- Decisions of autonomic control can be taken by individual nodes/devices of the network without the involvement of a centralized entity but with the nodes interacting with each other in order to share knowledge and continuously adapt their behaviour to the local networking conditions.
- In additions to independent control loops implemented within the nodes/devices, GANA allows for the design of network-level control loops that control the behaviours of nodes/devices in the network via a logically centralized decision making entity or an overlay (cloud)

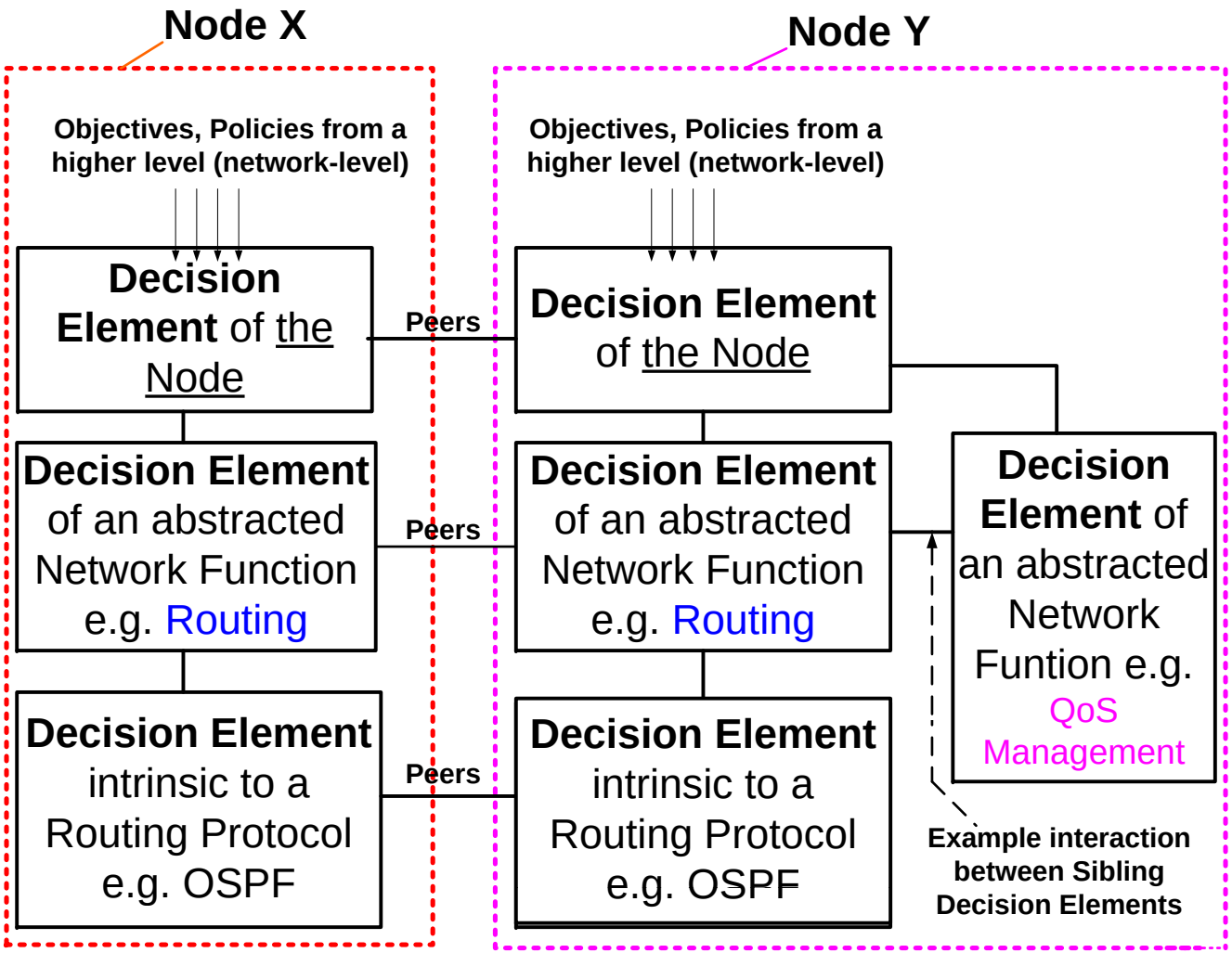
The GANA architecture (2)

- Adopts **four** functional planes from 4D but redefines them:
  - **GANA Decision Plane:** it makes all decisions driving a node's behaviour (including the behaviour of all managed entities of the node) and network-wide control, including reachability, load balancing, access control, security, and interface configuration.
  - **GANA Dissemination Plane:** it consists of mechanisms and protocols that provide a robust and efficient communication substrate that is used to exchange control information as well as any special type of information (or knowledge) that is not considered as the actual user data e.g. monitoring data.
  - **GANA Discovery Plane:** it consists of protocols or mechanisms responsible for discovering what entities make up the network or a service and creating logical identities to represent those entities. The discovery plane defines the scope and persistence of the identities, and carries out the automatic discovery and management of the relationships between them.
  - **GANA Data Plane:** it consists of protocols and mechanisms that handle individual packets based on the state that is output by the Decision Plane.

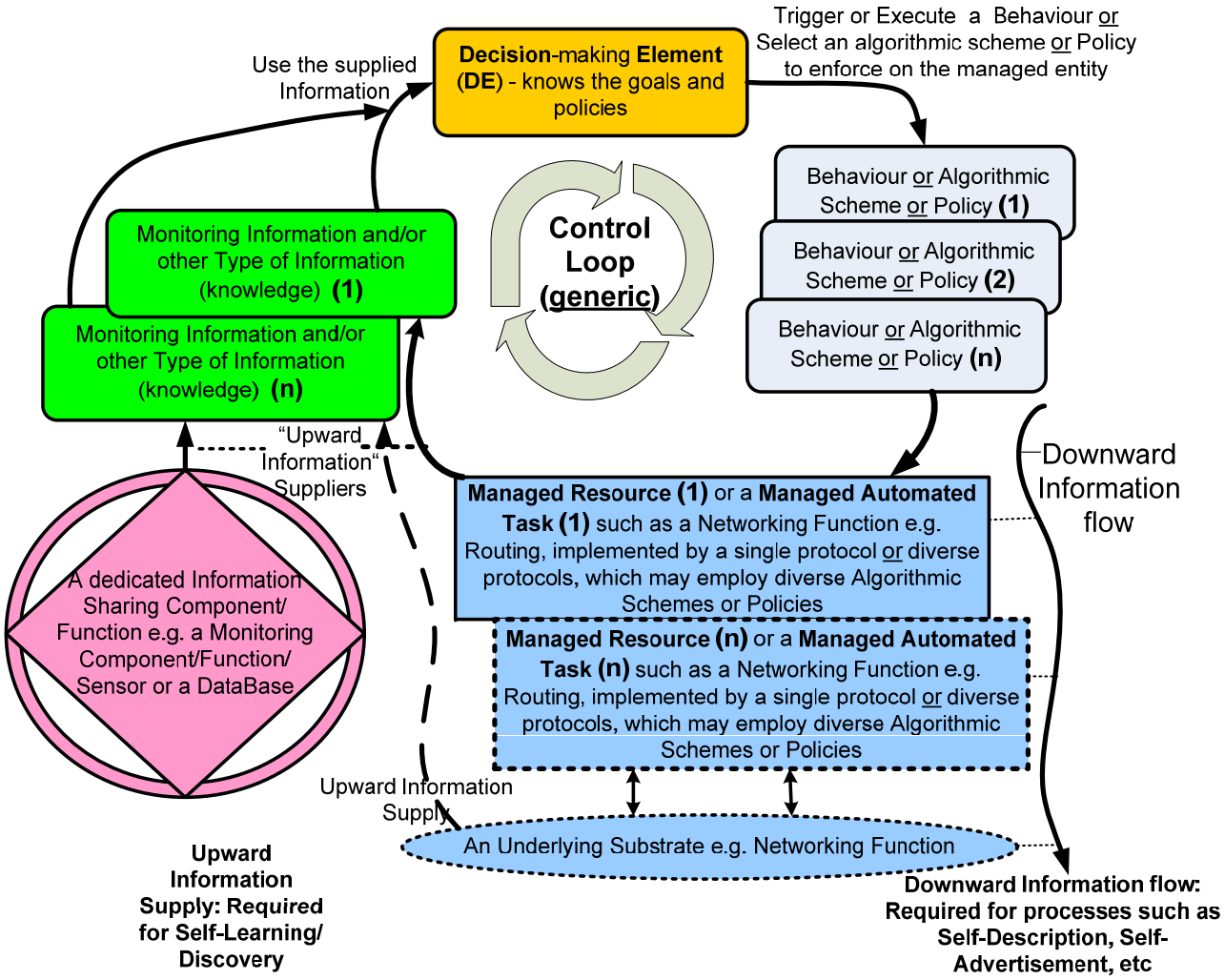
Decision Elements in GANA (1)

- **Decision-Making-Element (**DME/DE) : a concept that is associated with some concrete resource(s)/element(s) that the DME manages, implements and drives its control loop based on its continuous learning cycle, whereby information or views are being continuously exposed by its managed resource(s)/element(s), together with information coming from other required or potential information suppliers of the DME, such as the environment in which the device hosting the DME is operating.
- DEs that drive control loops of an autonomic node/network follow the same hierarchical approach.
- DEs have the ability to form peers with other DEs
  - *Hierarchical relationships:* a lower level DE(s) is managed by its upper level DE
  - *Peering relationships:* communication between DEs for exchanging information
  - *Sibling relationships:* the entities are created or managed by the same upper level DE

Decision Elements in GANA (2)



Hierarchical Control Loop (HCL)



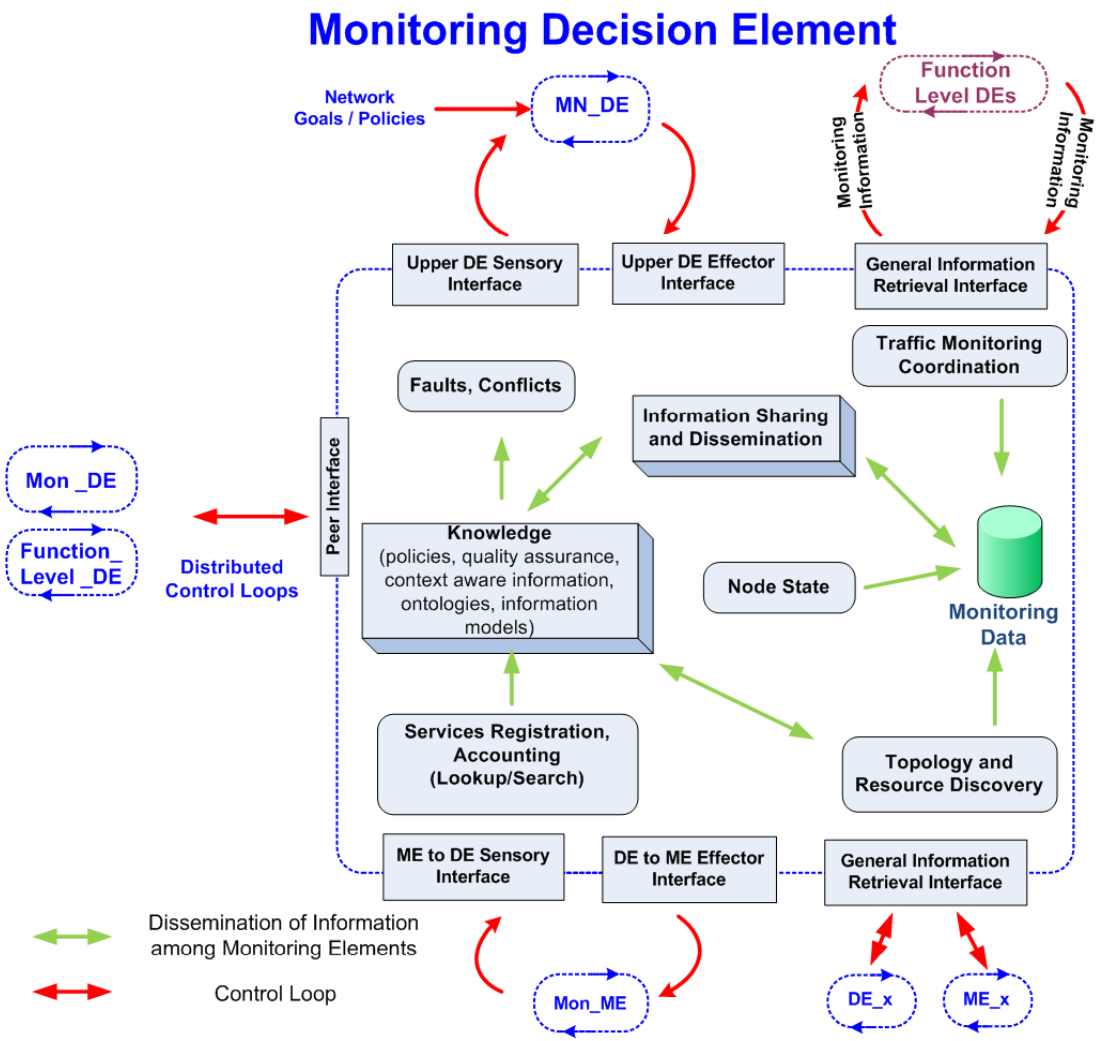
Control Loops in GANA

- In order for a node to control its behaviour in an autonomic manner, and thus take decisions about most of its functionalities, specific control loops have to be defined in a hierarchical approach
  - ***Network Control Loop*:** It manages and determines the overall behaviour of the network according to policies imposed by a third party ("manager") or co-operatively by the components of the network itself.
  - ***Node Control Loop*:** It manages and determines the overall behaviour of the node, applying specific rules to the node or adjusting its behaviour according to the commands imposed by the Network Control loop.
  - ***Functions Control Loop*:** It abstracts a particular Networking Function(s) and its associated mechanisms and controls all the protocols and mechanisms/algorithms collectively abstracted by a particular "Networking Function"
  - ***Protocol Control Loop*:** It can be used to realize autonomic driven protocols and control their corresponding operation and functionalities.
- Decision-making Elements that drive Control Loops follow the same *Hierarchy* consisting of four levels of autonomicity

EFIPSANS Network Monitoring Framework

- Monitoring information is a fundamental part of a wide number of network functionalities, such as QoS Management, Routing, and Mobility.
  - great deal of operational overhead involved in the configuration / re-configuration / optimisation of these operations
- Monitoring Framework based on GANA principles
- The monitoring framework describes the basic functions needed for management of monitoring activities within an autonomic network.
- The Monitoring Decision Element is responsible for the configuration of the monitoring protocols and mechanisms
  - other function level Decision Elements within a node and within the network can be guaranteed that relevant and sufficiently accurate information is available to drive their control loops.
  - is responsible to orchestrate Monitoring Managed Entities (Mon_MEs) so as the node level monitoring policies are fulfilled and specific functions are realised.
- The Monitoring Decision Element and the associated monitoring protocols and mechanisms play the role of information suppliers to other Decision Elements within the node and the network as a whole.

EFIPSANS Network Monitoring Framework



Monitoring Decision Element (1)

- Traffic Monitoring Coordination
  - manage mechanisms for traffic monitoring and schedule monitoring measurements in such a way that they do not degrade other services,
  - activate active or passive measurements in the autonomic node in accordance to provided services and network context and
  - store and publish any historical data collected via measurements for later analysis
- Information Sharing and Dissemination
  - collect information about stored data in node and network level and make it available to other Decision Elements or Managed Entities
  - monitoring data may either be distributed across the network or stored in a centralized database
- Topology Discovery
  - collect information necessary for creating the topology of the network and publish topology data in a appropriate format to other Decision Elements or Managed Entities
- Resource Discovery
  - collect information regarding the available resources in network level (very useful in ad hoc networks where nodes continuously join/leave the network and allocation and negotiation of different resources is dynamic).

Monitoring Decision Element (2)

- Monitor the state of the node
  - monitor the available node resources and specific metrics such as available power, storage capacity and status of interfaces
- Provide context-aware information
  - process any collected data according to the specified GANA ontology in order to enable reasoning over the available information and extract meaningful events.
  - efficiently sense changes in the network and the provided services and proceed to corrective actions (self-healing).
- Fault and conflict management
  - diagnose a) faults in node and network level, b) violation of guarantees for predefined performance metrics and c) firewalling and security alarms
- Accounting and registration to network services
  - identify services supported in the autonomic network, enable subscription to services and access to specific information and provide related information to other Decision Elements

Traffic Monitoring and QoS Control (1)

- We focus on autonomicity as a feature of traffic monitoring, coupled with Quality of Service (QoS) management functions of an ingress edge router, complying with GANA principles.

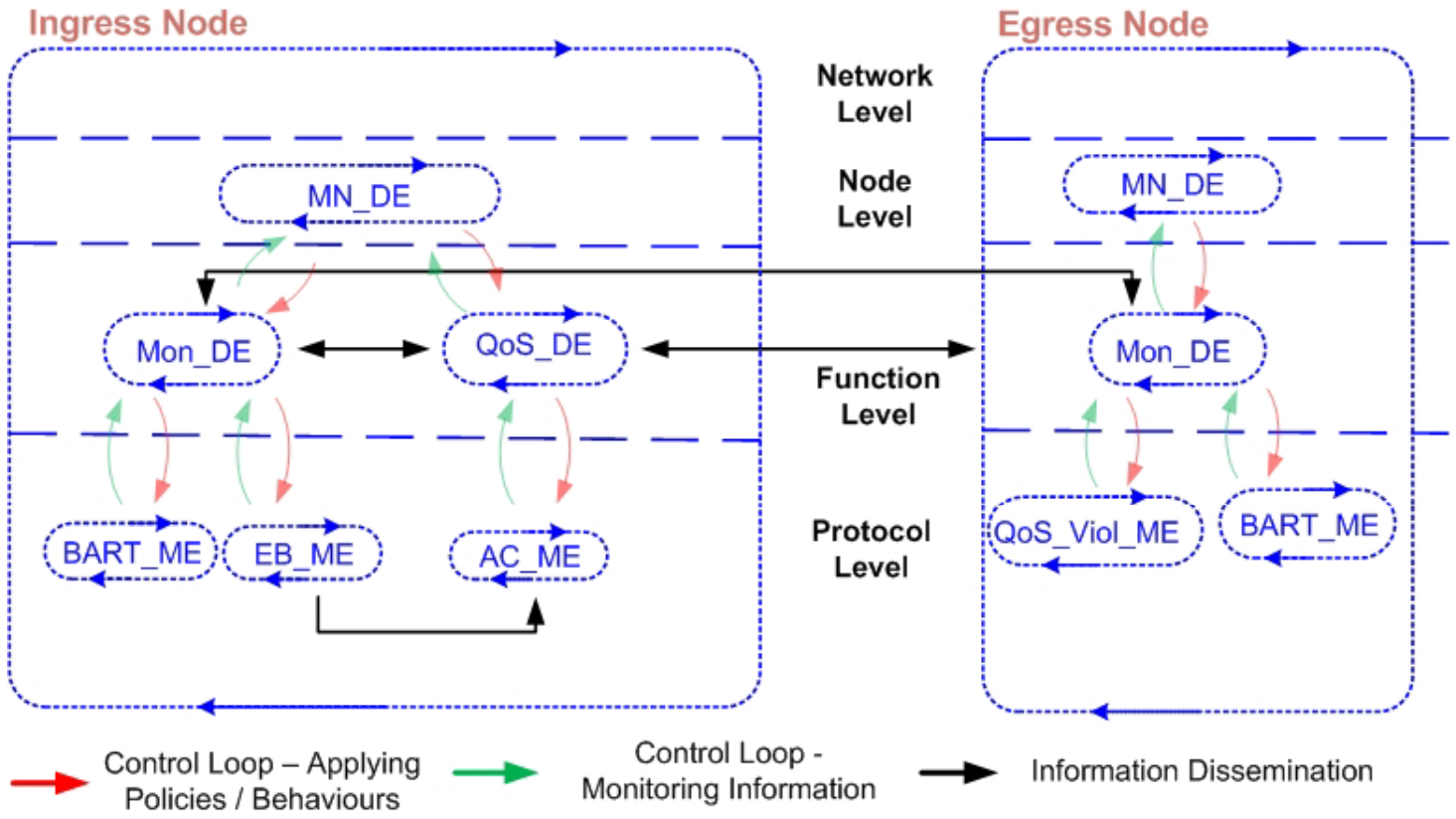
- As network and traffic conditions continuously change, monitoring protocols and mechanisms must be appropriately re-configured in order to facilitate the efficient QoS management of the autonomic network.

- The objective of QoS control at the ingress point within a DiffServ domain is to ensure that the traffic admitted to the network is appropriately classified, policed and shaped to assure performance guarantees.

Traffic Monitoring and QoS Control (2)

- Quality of Service DE (QoS_DE)
  - configure the mechanisms - such as queue management, queue scheduling, marking, policy, admission control, etc - to support service guarantees provided by the network
- Effective Bandwidth ME (EB_ME)
  - collects a packet trace from the network, performs a number of processing activities on it and reports an estimation of effective bandwidth for a particular QoS target of packet delay
- Bandwidth Availability in Real-time ME (BART_ME)
  - The BART_ME on the destination node estimates the amount of available bandwidth along the path between the two nodes
- Admission Control ME (AC_ME)
  - ensures that traffic admitted into a node or network will not violate specified QoS performance guarantees
- QoS Violation ME (QoS_V_ME)
  - produces estimations of performance violations for traffic exiting the network through an egress router. The violations are estimated by analysing short packet traces and results are compared with particular QoS targets.

Traffic Monitoring and QoS Control (3)



Future Work

- Implement the described scenario in the EFIPSANS testbed
- Use of Overlay Networks for efficient network monitoring and management
- ONIX framework – Overlay Networks for Information Exchange
- Extend the work on Context Awareness – Combine GANA Ontology with existing traffic monitoring ontologies
- Evolve GANA and specify interfaces

Thank you for your attention!

Questions?