Self-Managed Networks with Fault Management Hierarchy

Mehmet Toy, Ph.D
Mehmet_Toy@cable.comcast.com

COMCAST

March, 2015

Outline

- Problem Statement
- Proposed Concepts and Architectures
- Self-Managing NE and Self-Managing NMS Architectures
- Self-Managing Agent Architecture
- In-band Communications
- Fault Management Hierarchy in Centralized and Distributed Networks

Problem Statement

A. So far, the industry has been focused on
   - Monitoring of network resources and services,
   - Isolating problems, and
   - Fixing them by sending technicians to the sites most of the time or
   - Downloading certain configuration files remotely for configuration related problems.

B. Networks need to identify problems by itself, fix them, and have technicians at the site of a failure only when there is a single point of hardware failure.

C. Tools for self managed networks need to be developed.

Proposed Concepts and Architectures [1-3]

A. **A concept of self-managed networks for identifying network problems and repairing them, in addition to self-configuration**
B. **Self-managed NE (Network Element) architecture, consisting of intelligent components**
C. **A centralized and distributed management architecture with intelligent systems for a self-managed network**
D. **A concept of in-band communications of failure types associated with smallest replaceable components, ESTIMATED FIX time, and fixes for failures identified by sNE, sNMS/OSS, and field technician to all related parties (i.e. sNEs, sNMS/OSS, field technician and users)**
E. **A frame format for in-band communications of failures, failure types, estimated fix time and fixes**
F. **A concept of hierarchy for self-fixing of failures in the network**

[1] M. Toy, Self-Managed Networks, Comcast internal document, November, 2012.
[2]M. Toy, "Self-Managed Networks with Fault Management Hierarchy", Procedia Computer Science, Elsevier, Nov., 2014.
[3] M. Toy, "Cable Networks, Services, and Management", ISBN: 9781118837597, J.Wiley-IEEE Press, 2015.

Autoconfiguration

- Currently CPE, interfaces and connections are auto-configured with a process, after the equipment is connected to the network, with/without SDN tools). More work is required to streamline the process and include autoprovisioning of all equipment, not just CPE.

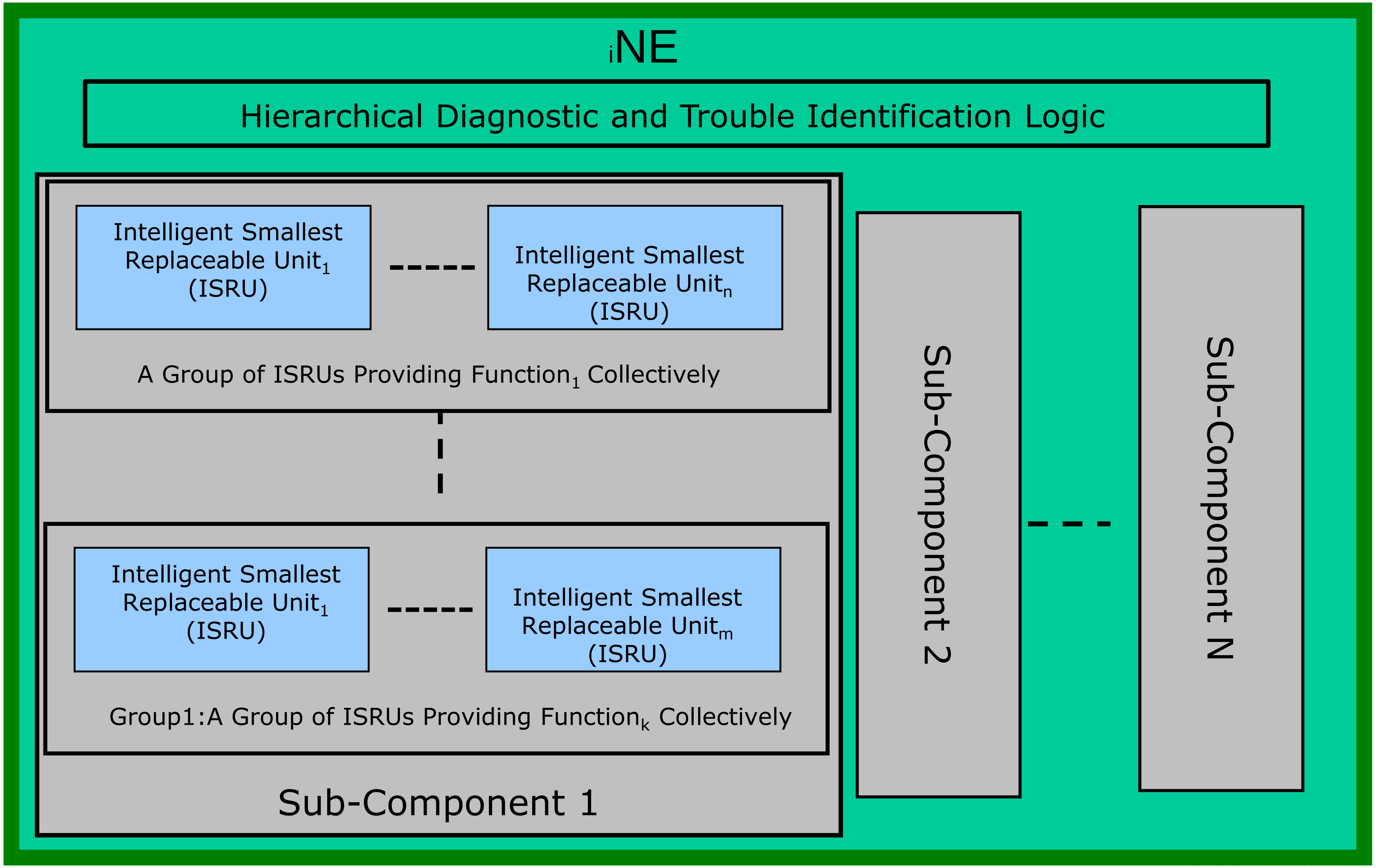
Intelligent NE (iNE)

A. **Consists of intelligent sub-components that are**
   - **Capable of periodic self-checking,**
   - **Declaring a failure when it is unable to perform its functions,**
   - **Running diagnostics and identifying whether the faulty entity is within the subcomponent or not,**
   - **Escalating the diagnostics to the next level in the hierarchy within NE.**

A. **Able to run diagnostics for a pre-defined set of sub-components, that are collectively performing a function, and finally to run diagnostics at iNE level to determine the failure.**

B. **After the failure is identified to the smallest replaceable HW (e.g. chips, connectivity between chips, backplane, etc.) and/or SW entity (e.g. kernel, log, efd, etc.), iNE determines if the failure is fixable by the iNE and communicates that to related parties.**

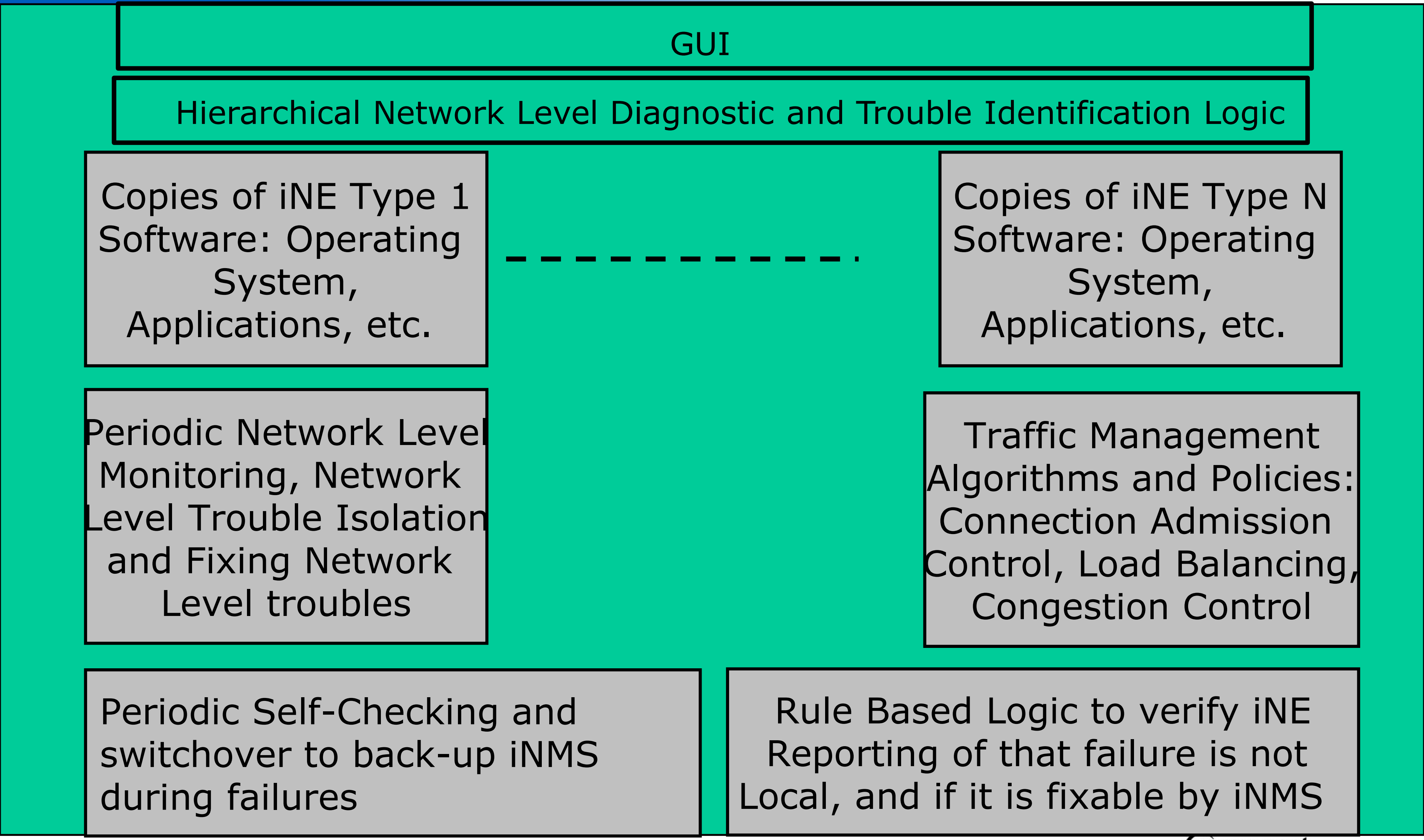
Intelligent NE (iNE)


iNE
Hierarchical Diagnostic and Trouble Identification Logic
Intelligent Smallest Replaceable Unit$_1$ (ISRU)
Intelligent Smallest Replaceable Unit$_n$ (ISRU)
A Group of ISRUs Providing Function$_1$ Collectively
Intelligent Smallest Replaceable Unit$_1$ (ISRU)
Intelligent Smallest Replaceable Unit$_m$ (ISRU)
Group1:A Group of ISRUs Providing Function$_k$ Collectively
Sub-Component 1
Sub-Component 2
Sub-Component N

Intelligent NMS/OSS (iNMS/iOSS)

A. Periodic self checking
B. Backed-up with an active iNMS/iOSS when it fails
C. Stores software modules specific to iNEs
A. Able to verify if iNE failure is not local when iNE reports that the failure is not local
B. Able to periodically monitor network, identify network level failures

Intelligent NMS



GUI
Hierarchical Network Level Diagnostic and Trouble Identification Logic
Copies of iNE Type 1 Software: Operating System, Applications, etc.
Copies of iNE Type N Software: Operating System, Applications, etc.
Periodic Network Level Monitoring, Network Level Trouble Isolation and Fixing Network Level troubles
Traffic Management Algorithms and Policies: Connection Admission Control, Load Balancing, Congestion Control
Periodic Self-Checking and switchover to back-up iNMS during failures
Rule Based Logic to verify iNE Reporting of that failure is not Local, and if it is fixable by iNMS

Self Managing Agents for sNEs

A. Each sNE will have one or more Intelligent Hardware Maintenance Agent(s) (iHMA(s)) depending on the implementation
   - To monitor hardware entities such as CPU, memory, physical ports, communication channels, buffers, backplane, power supplies, etc., and take appropriate maintenance actions during hardware failures

B. Each sNE will have one or more Intelligent Operating System Maintenance Agent (s) (iOMA (s))
   - To monitor operating system and take appropriate maintenance actions during Operating System failures

C. Each sNE will have one or more Intelligent Application Maintenance Agent (s) (iAMA (s))
   - To monitor application software and protocol software, and take appropriate maintenance actions during application and protocol software failures

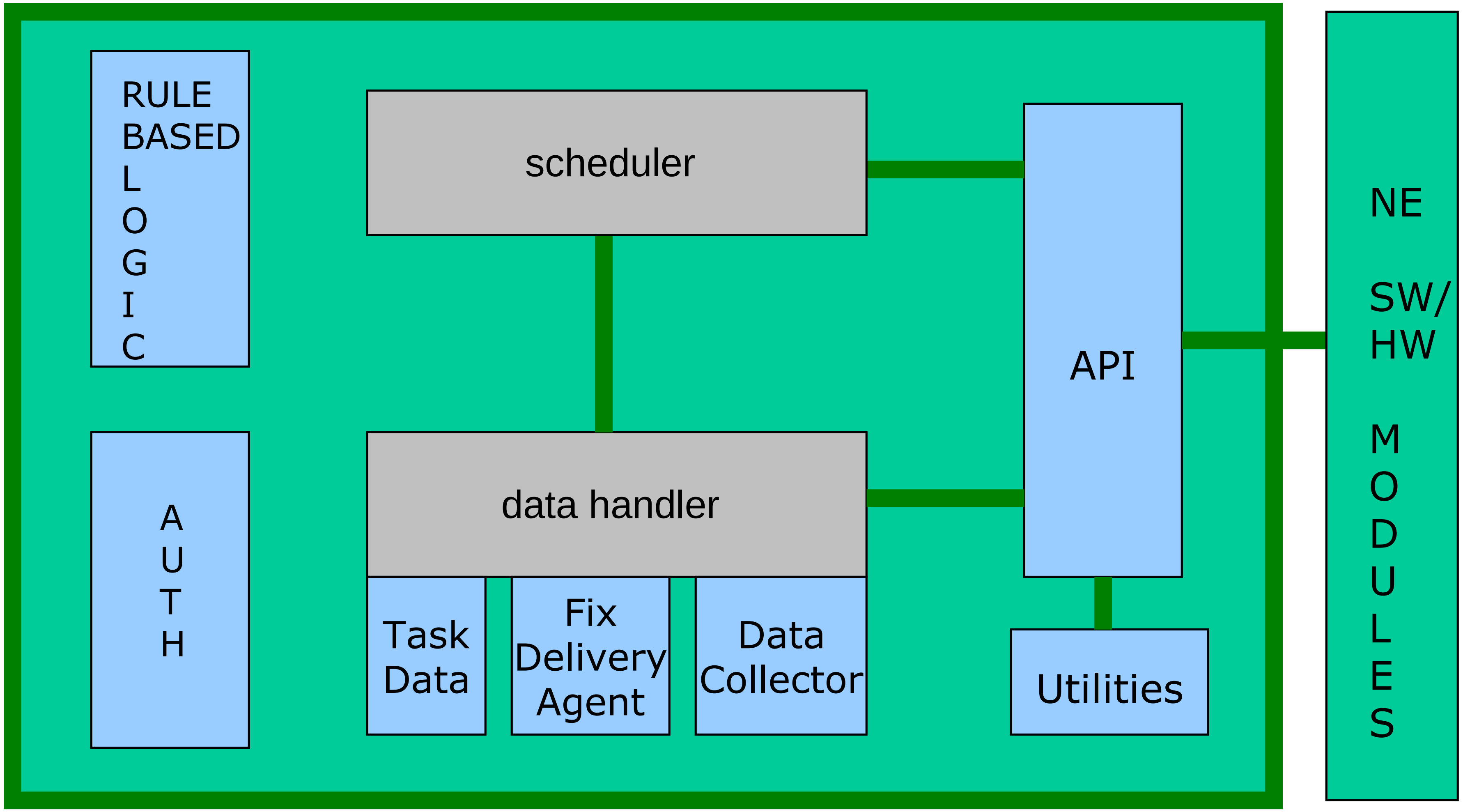
D. Each sNE will have one Intelligent Capacity Management Agent (s) (iCMA (s))
   - To monitor capacity, system load and system performance, collect measurements and take appropriate actions

Self Managing Agent Responsibilities

A. Monitor the entity that it belongs to, report a failure to all related parties (i.e. sNEs, sNMS, field technicians, etc), fix the problem, and report the fix to all related parties.

B. If the problem is determined to be not fixable after one, two, or three tries, send a message to sNMS asking for help at the site or remote fix.

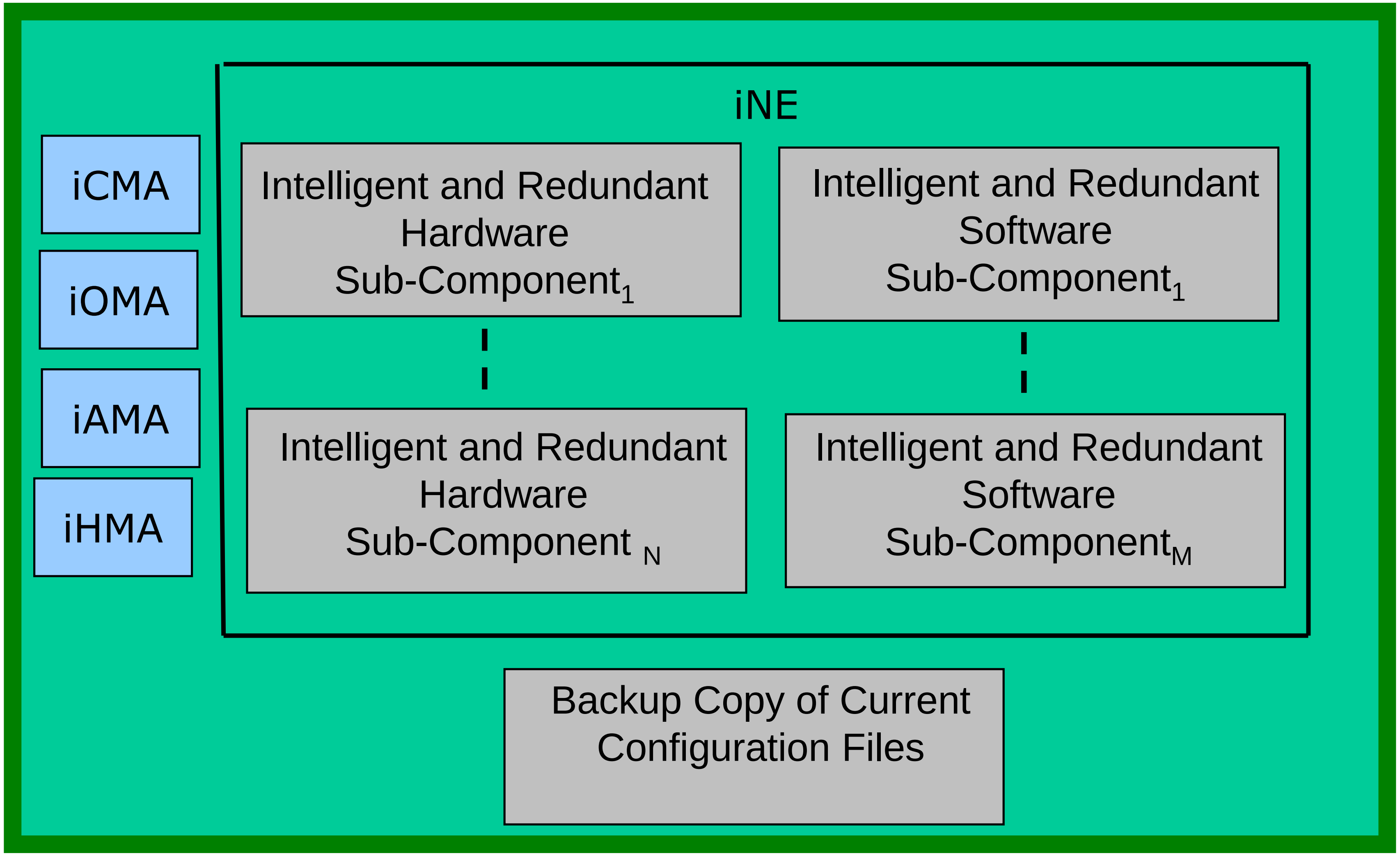
Architecture for iHMA, iCMA, iOMA, iAMA



Architecture for iHMA, iCMA, iOMA, iAMA (Cont.)

- The agent consisting of
  - *Rule Based Logic* to determine the problem and initiate/execute the fix if the problem is local to NE, initiate a test for the fix, determine if the fix procedure or a step or some of the steps are to be repeated. If the problem is not local to NE, the agent informs the NMS for its conclusion.
  - *Scheduler* to determine the priority and order of the steps with each functional entity within NE
  - *API* for the agent to interface various types of SW and HW entities within an NE
  - *Data Handler* to collect necessary data (such as appropriate measurements) for the NE, perform the tasks for the fix, and keep data associated with the task
  - *AUTH* to authenticate local user and remote user (from NMS) interface to the agent
  - *Utilities* to support various file operations
- sNMS holds a copy of the agent and remotely loads into NE when needed.

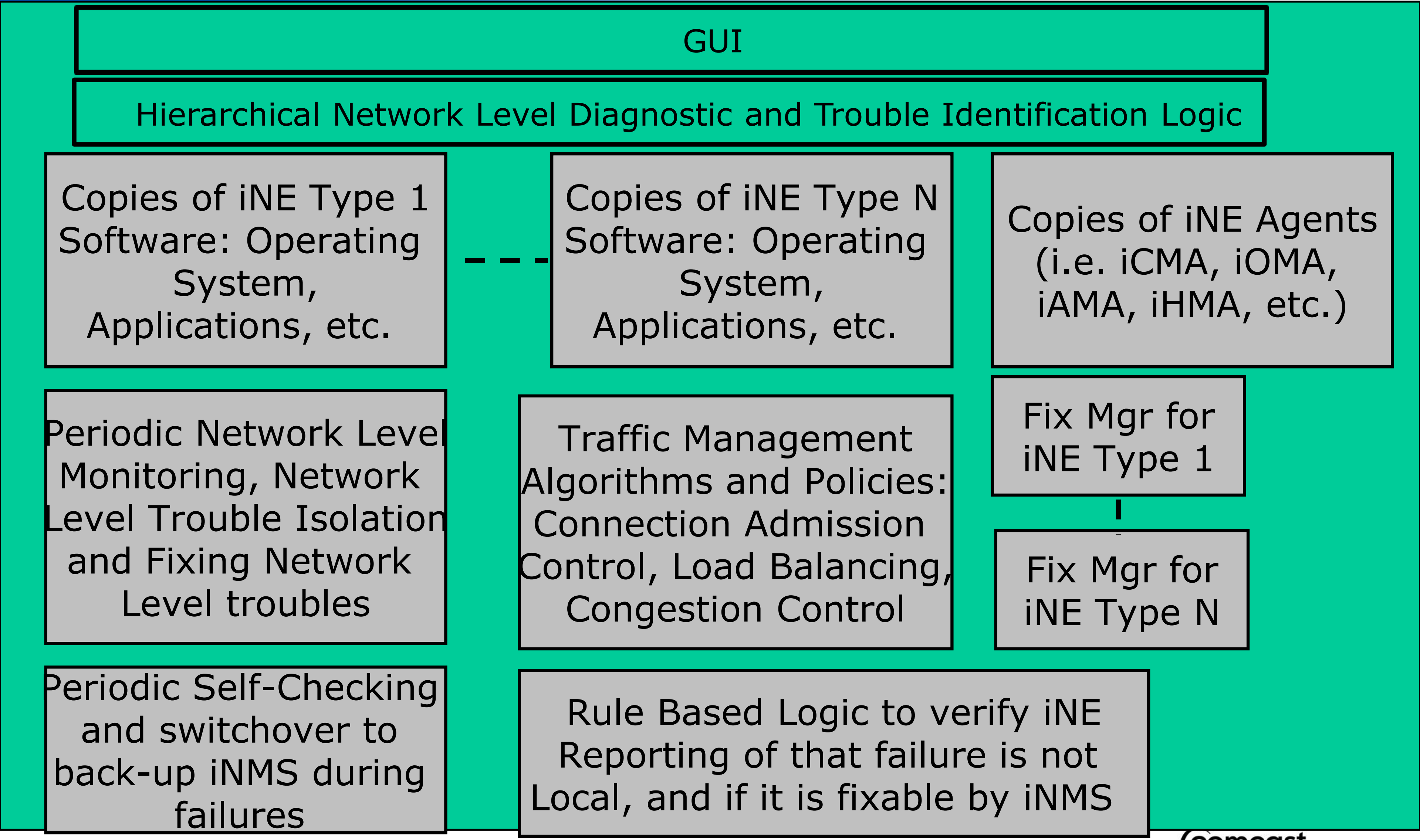
Architecture for Self-Managed NE


iCMA
iOMA
iAMA
iHMA
iNE
Intelligent and Redundant Hardware Sub-Component$_1$
Intelligent and Redundant Software Sub-Component$_1$
Intelligent and Redundant Hardware Sub-Component $_N$
Intelligent and Redundant Software Sub-Component$_M$
Backup Copy of Current Configuration Files

Self-Managing NMS/OSS Functions

A. ***C*****ollect end-to-end connection level measurements and NE level capacity measurements and store them**
B. Initiate and perform remote fixes at sNE (not fixes that can be executed locally)
C. Fix network level issues beyond the capabilities of sNEs
D. Perform Connection Admission Control (CAC), load balancing, and congestion control at network level
E. Prioritize and schedule execution of the tasks to perform repair and configuration activities that can be performed only remotely
F. Estimate the fix time for a failure that is going to repair and communicate that to related parties
G. ***S*****end info to remote clients about failures and fixes**

Self-Managing NMS


GUI

Hierarchical Network Level Diagnostic and Trouble Identification Logic

Copies of iNE Type 1 Software: Operating System, Applications, etc.

Copies of iNE Type N Software: Operating System, Applications, etc.

Copies of iNE Agents (i.e. iCMA, iOMA, iAMA, iHMA, etc.)

Periodic Network Level Monitoring, Network Level Trouble Isolation and Fixing Network Level troubles

Traffic Management Algorithms and Policies: Connection Admission Control, Load Balancing, Congestion Control

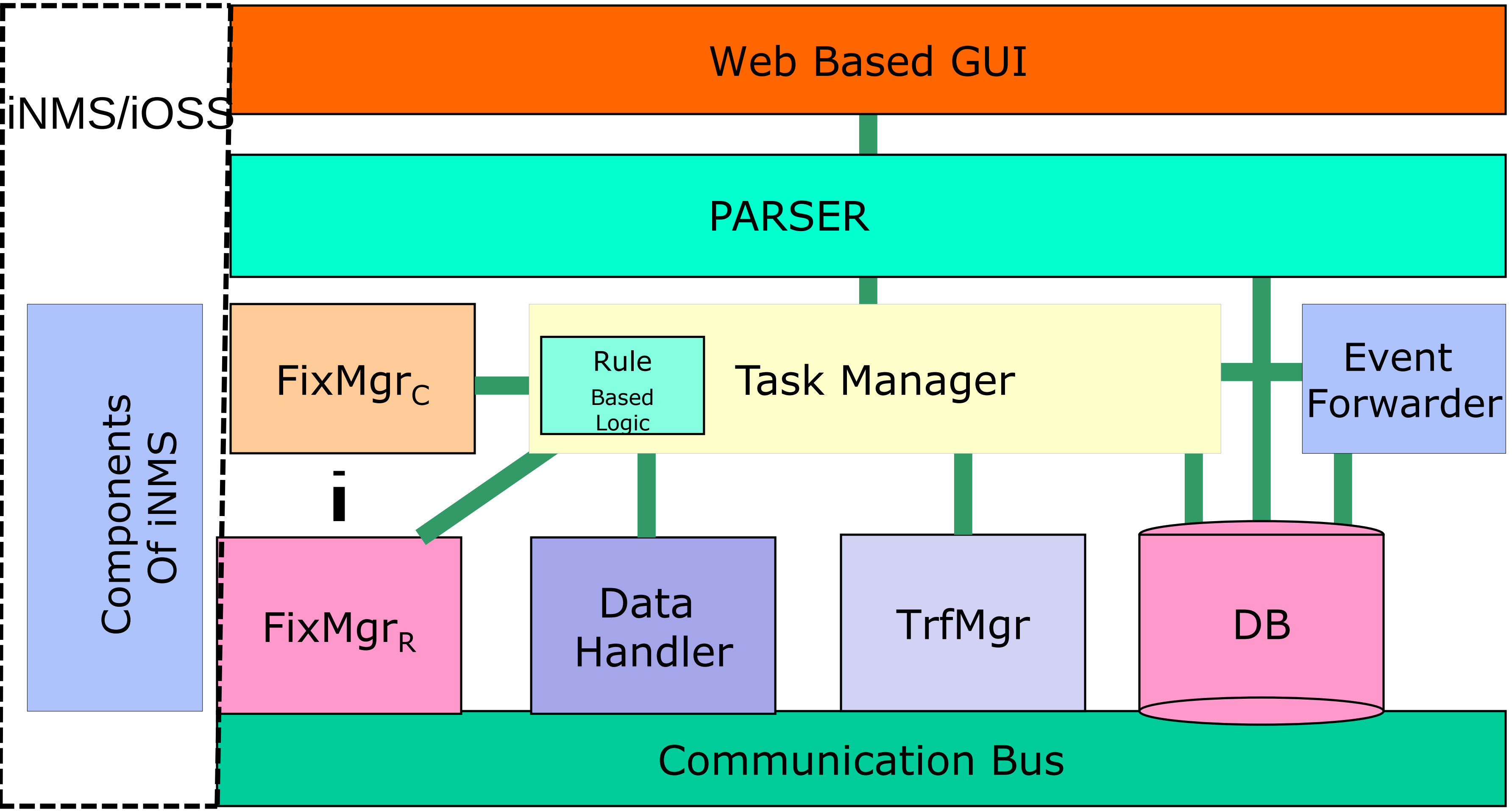
Fix Mgr for iNE Type 1

Fix Mgr for iNE Type N

Periodic Self-Checking and switchover to back-up iNMS during failures

Rule Based Logic to verify iNE Reporting of that failure is not Local, and if it is fixable by iNMS

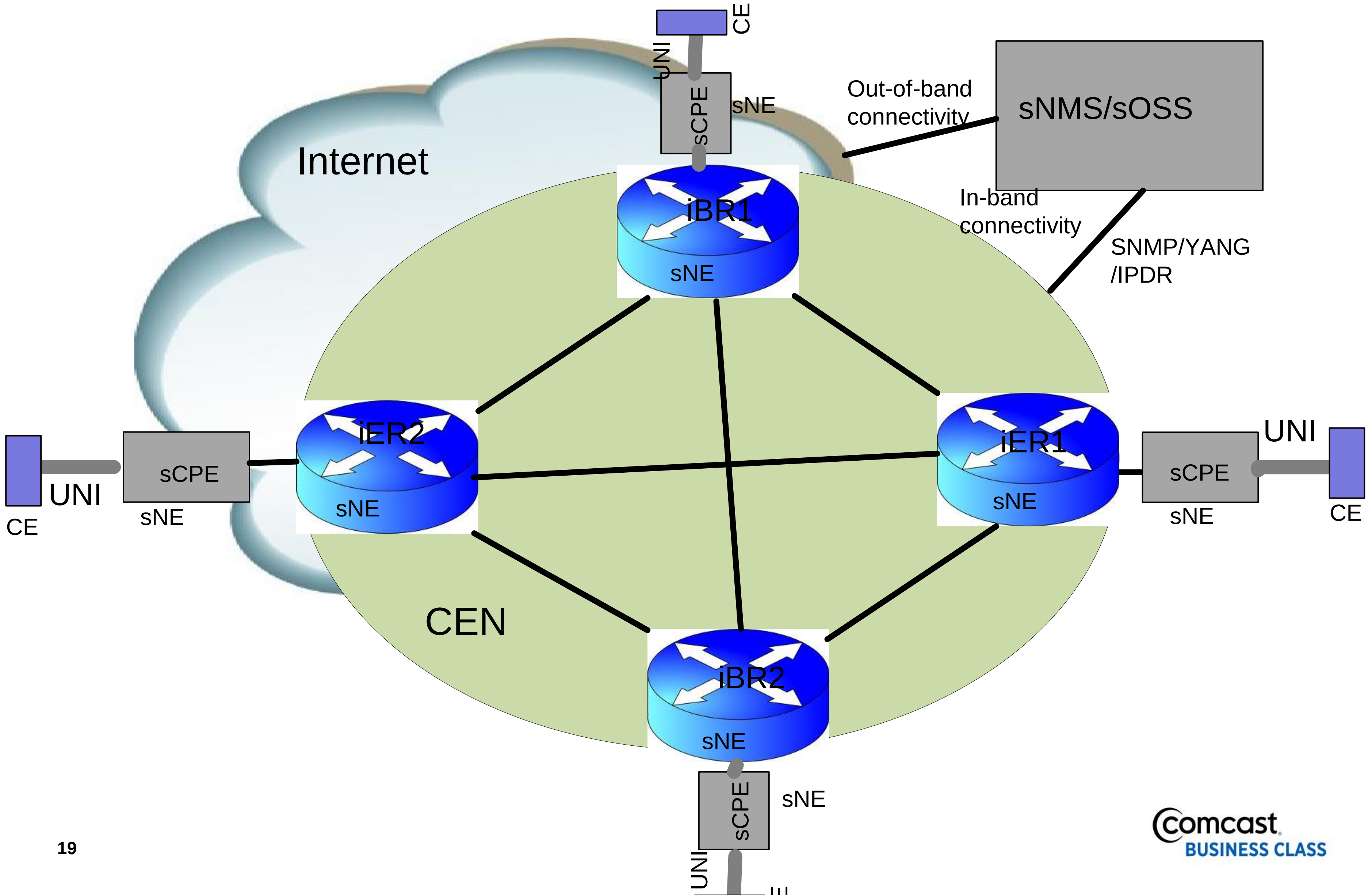
Self Managing NMS/OSS Architecture



Self-Managing NMS/OSS Architecture (cont.)

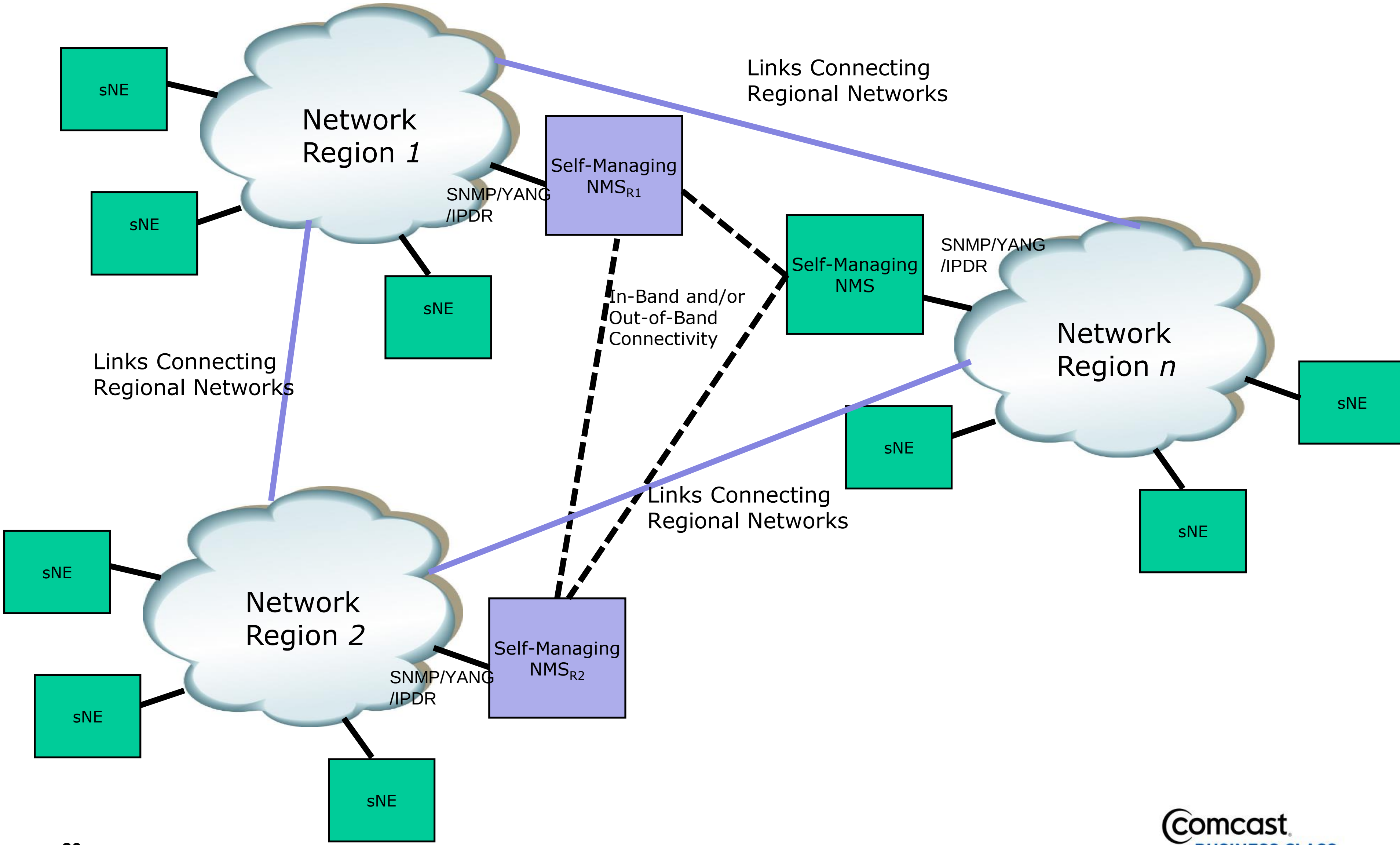
- *Web-based GUI* for human and machine interfaces
  - Login/Logout – Opens and closes sessions and does initial authentication.
  - MAP-Sends all data to and from the client and the PARSER.
    - Validates all submitted data from the client.
  - Processes chart data generated by parser, generates flash based charts and graphs, and sends it to the client.
- *Parser* to process GUI templates, Database for storing GUI events and data collected
- *Task Manager* to prioritize and schedule execution of the tasks, with *Rule Based Logic* to perform repair and configuration activities that can be performed only remotely
- *Event Forwarder* to send info to remote clients about failures and fixes
- *FixMgr* for each sNE type to initiate and perform remote fixes (not fixes that can be executed locally)
- *TrfMgr* for Connection Admission Control (CAC), load balancing, and congestion control at network level
- *Data Handler* to collect end-to-end connection level measurements and NE level capacity measurements and store them in DB to support TrfMgr
- *DB* to store all relevant data

Self-Managed Network Architecture-Example



Comcast BUSINESS CLASS

Distributed Self-Managed Network Architecture

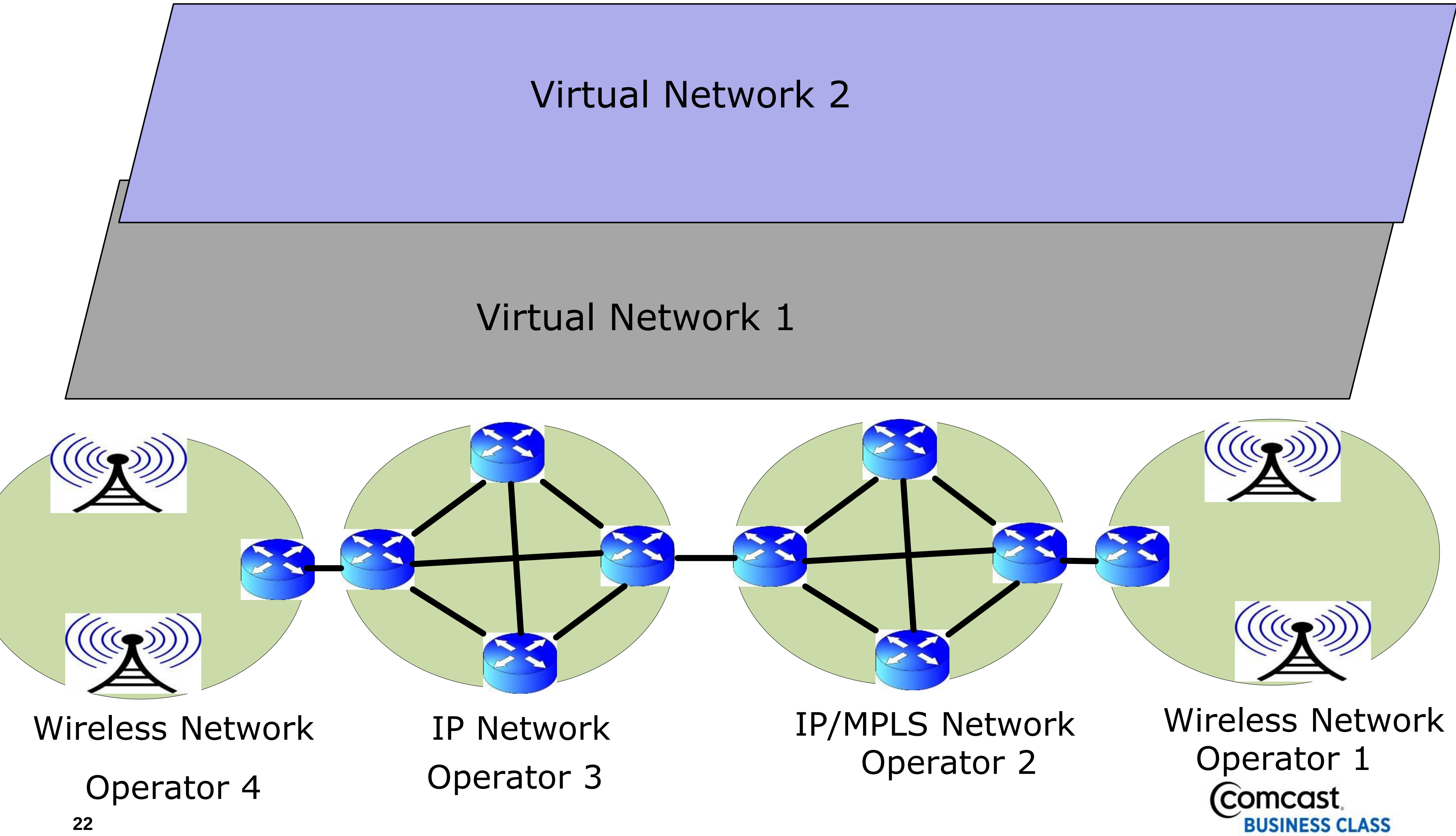


Comcast BUSINESS CLASS

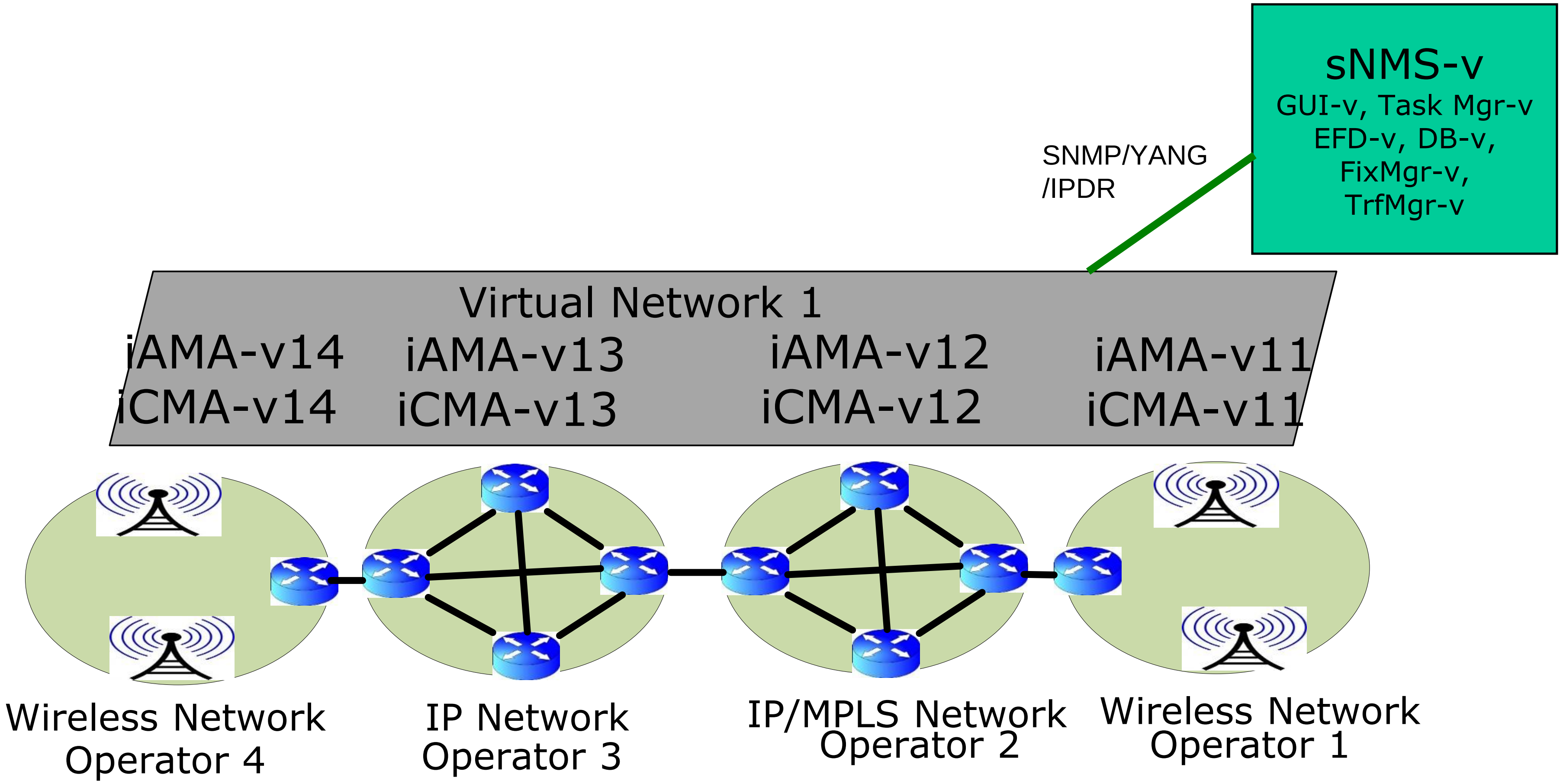
Distributed Self Managed Network Architecture

A. In distributed architecture, the network is divided into sub-networks (I.e. regional networks), where each sub-net has its own Self-Managing NMS_R
B. Self-Managing NMS_R provides all the centralized management functions for its own subnet and informs Self-Managing NMS about its activities.
C. End-to-end network level activities beyond region boundaries will be left to sNMS. These activities can be Connection Admission Control (CAC), load balancing, and congestion control at network level.

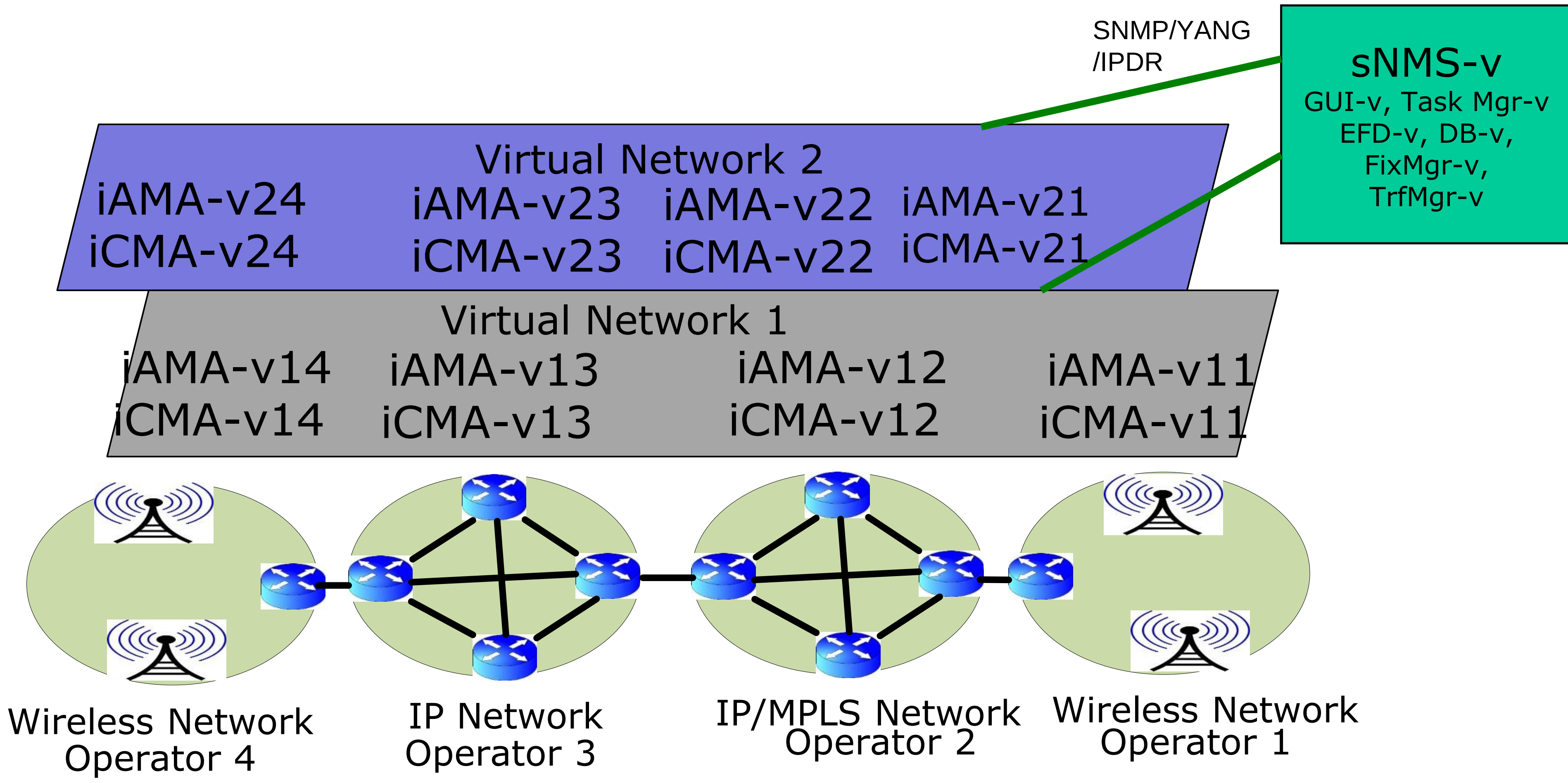
Virtual Networks



Centralized Self-Managed Virtual Networks



Centralized Self-Managed Virtual Networks (Cont.)



Self-Managed Virtual Networks (VNs)

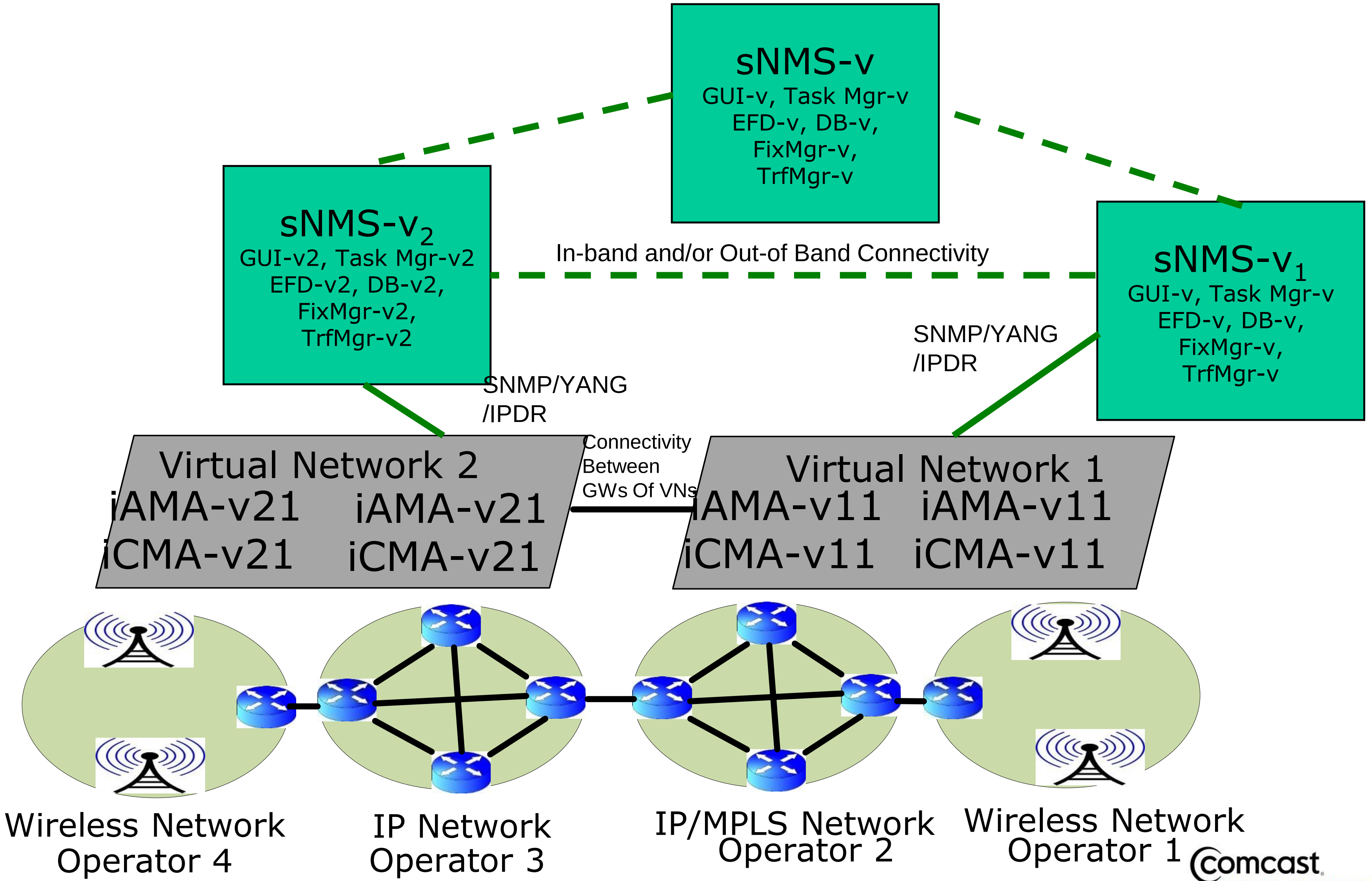
A. In addition to all components of self managed network infrastructure, we need to add self managed virtualized network infrastructure

B. Each VN will have one or more Intelligent Application Maintenance Agent (s) (iAMA-v (s))
   - To monitor application software and protocol software, and take appropriate maintenance actions during application and protocol software failures

C. Each VN will have one Intelligent Capacity Management Agent (s) (iCMA-v(s))
   - To monitor capacity, and system load and performance, collect system measurements and take appropriate actions

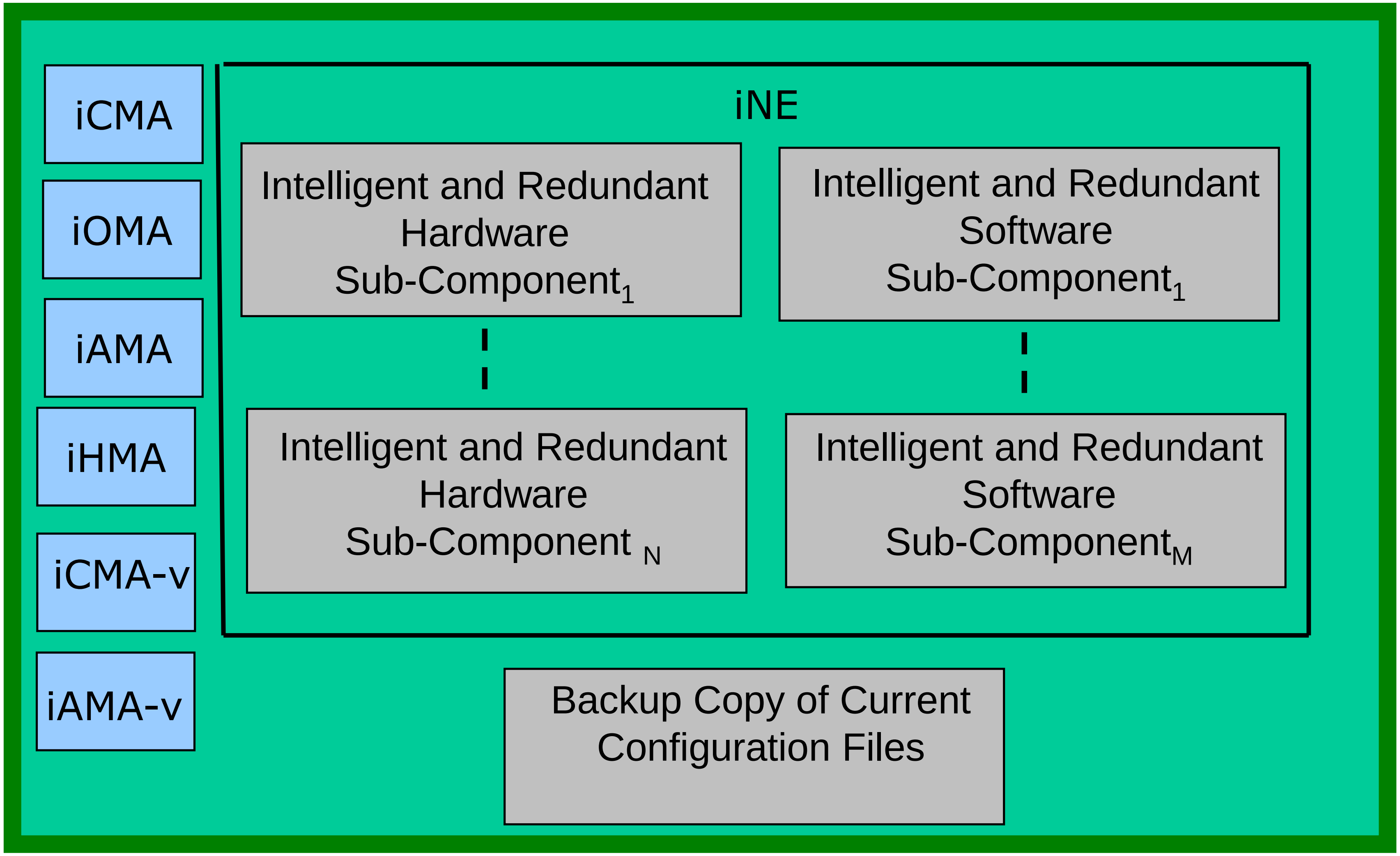
D. Each agent will monitor the entity that it belongs to, report failure to the centralized controller (sNMS-v), fix the problem, and report the fix to the centralized controller. If the problem is determined to be not fixable after one, two, or three tries, send a message to the controller (sNMS-v) asking for help at the site.

Distributed Self-Managed Virtual Networks



Comcast BUSINESS CLASS

Architecture for Self-Managed NE Supporting Virtualization

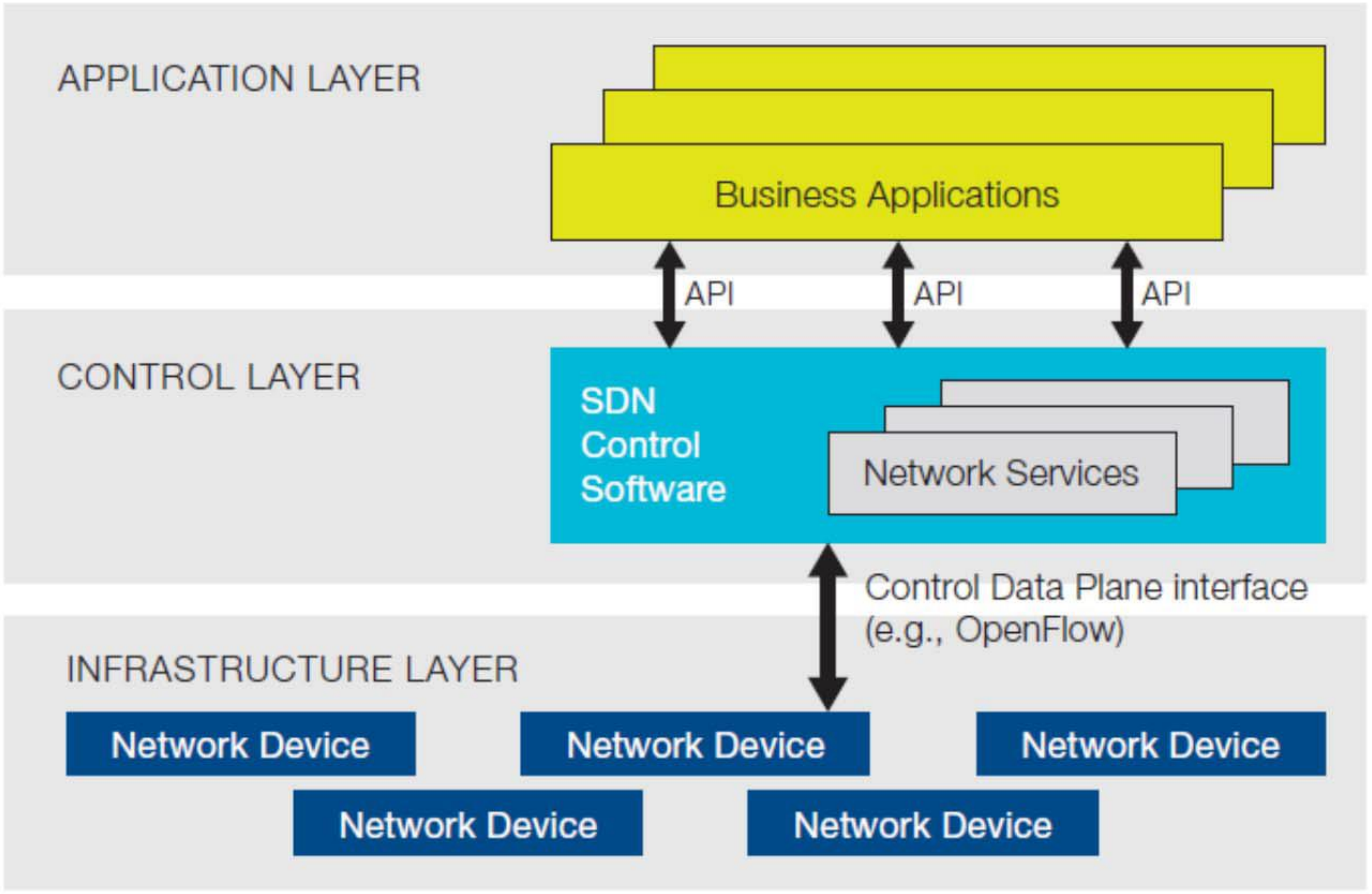


iCMA
iOMA
iAMA
iHMA
iCMA-v
iAMA-v
iNE
Intelligent and Redundant Hardware Sub-Component$_1$
Intelligent and Redundant Software Sub-Component$_1$
Intelligent and Redundant Hardware Sub-Component $_N$
Intelligent and Redundant Software Sub-Component$_M$
Backup Copy of Current Configuration Files

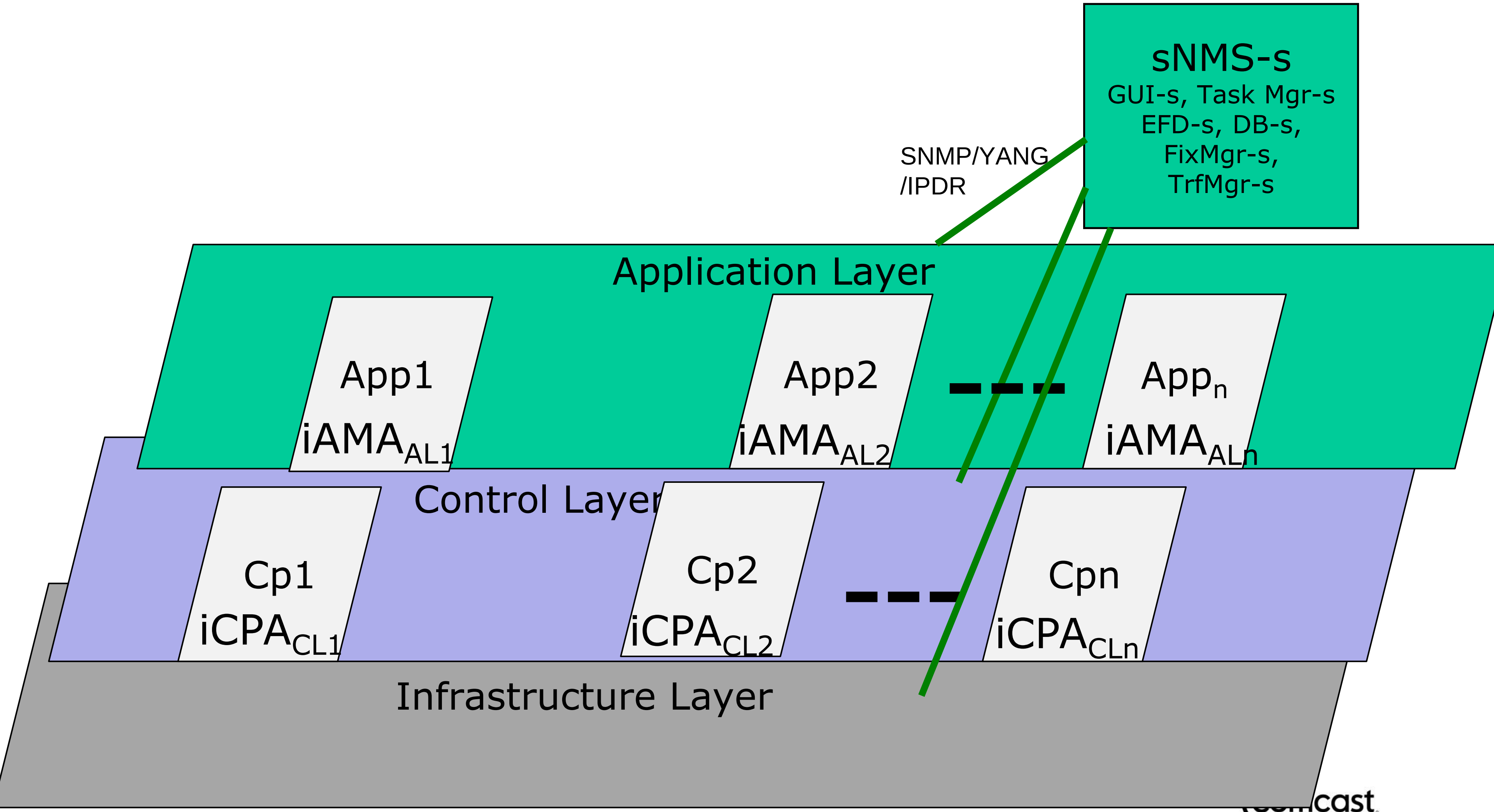
Distributed Self-Managed VN Architecture

- In distributed architecture, each VN will have its own $sNMS_v$
- $sNMS_v$ provides all the centralized management functions for its VN and informs sNMS about its activities.
- End-to-end network level activities beyond VN boundaries will be left to sNMS. These activities can be end-to-end Connection Admission Control (CAC), load balancing, and congestion control.

Software Defined Network (SDN) Architecture



Centralized Self Managed SDN-I



Self-Managing Agents for Layers of SDN

A. In addition to all components of self managed network infrastructure, we need to add self managed Control and Application Layers of SDN

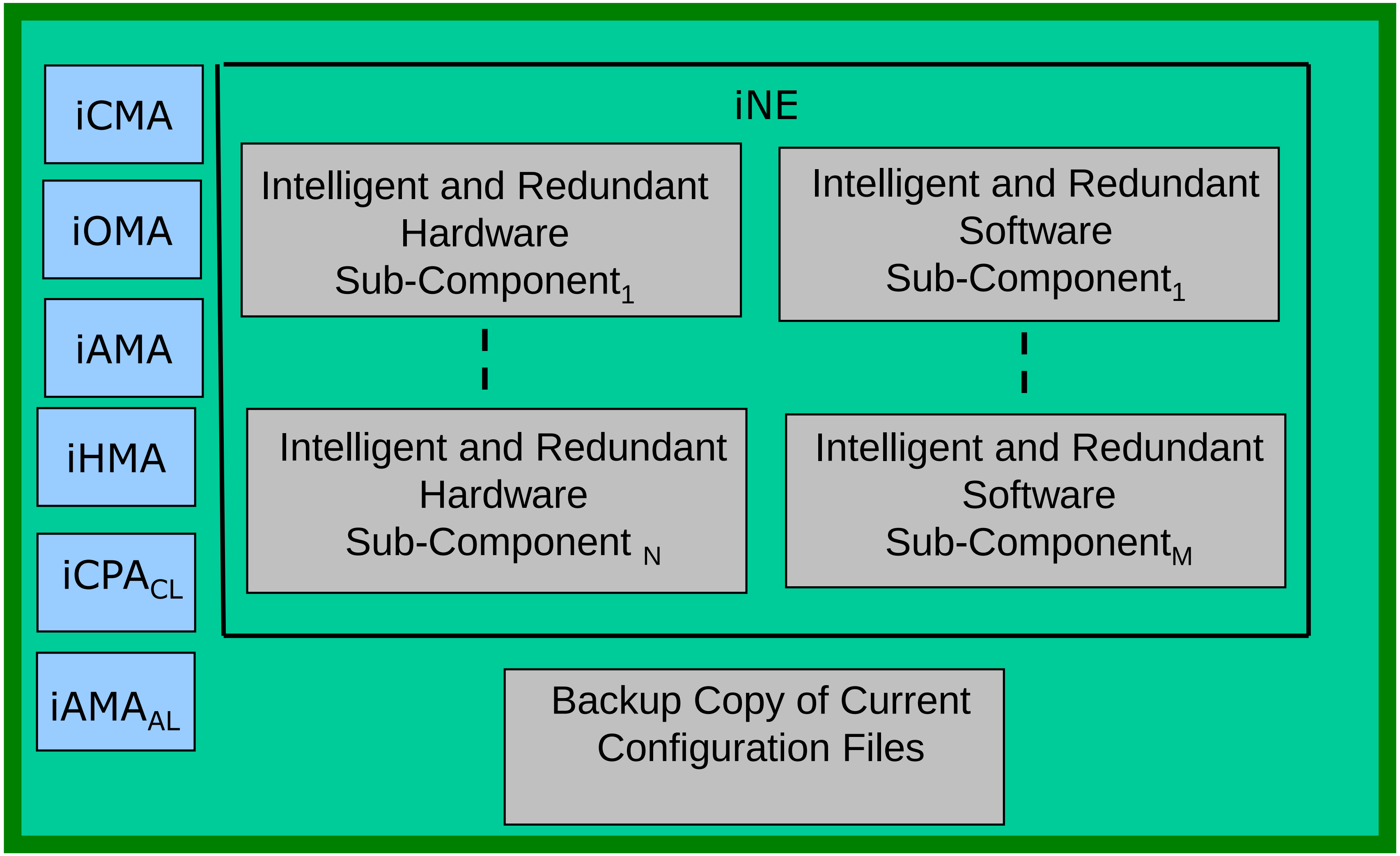
B. Each Application will have an Intelligent Application Maintenance Agent (s) (iAMA$_{AL}$ (s))

- **To monitor application software, and take appropriate maintenance actions during application software failures**

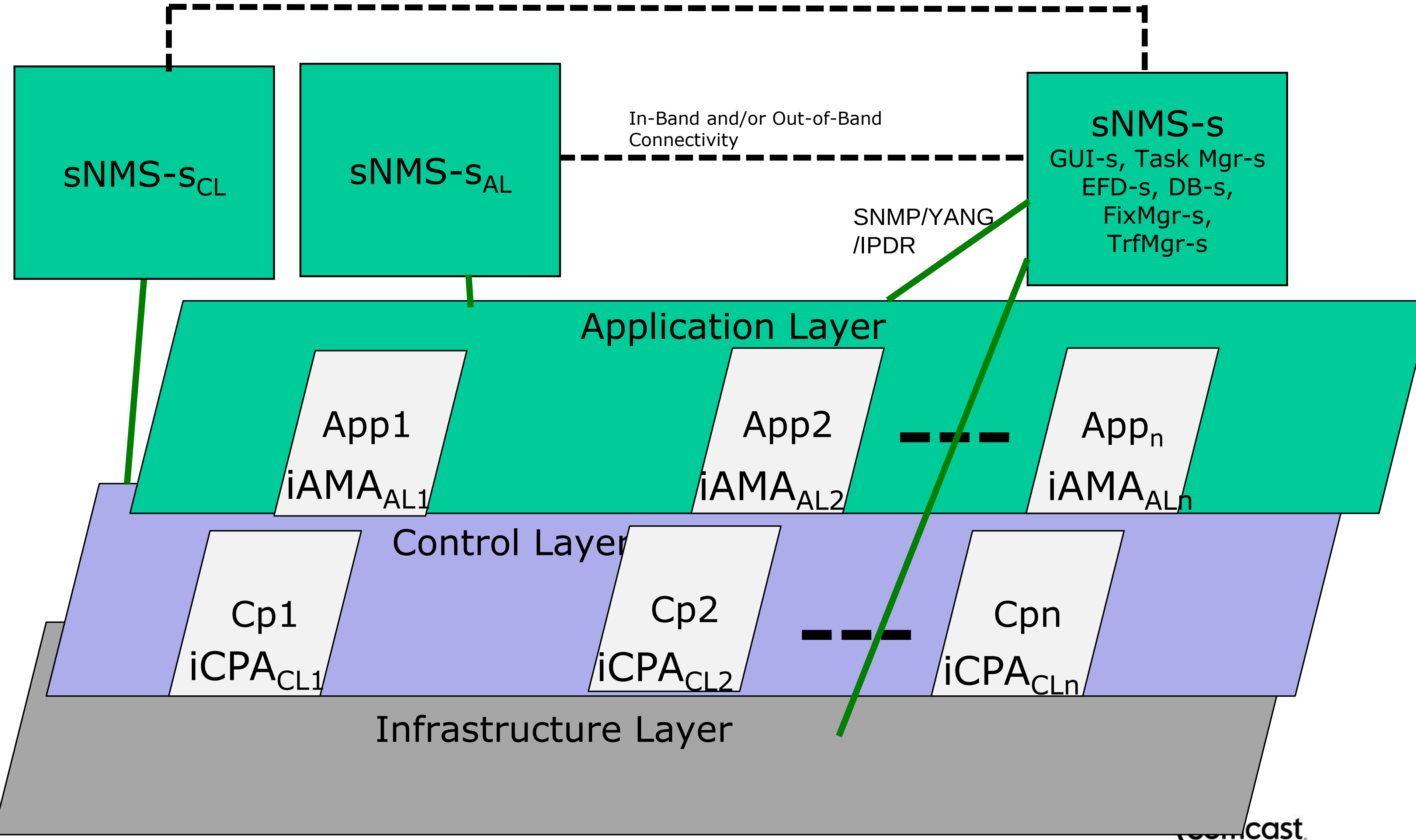
C. Each Control Layer Protocol will have one Intelligent Control Protocol Management Agent (s) (iCPA$_{CL}$ (s))

- **To monitor control protocol software, and take appropriate maintenance actions during the software failures**

Architecture for Self-Managed NE Supporting SDN


iCMA
iOMA
iAMA
iHMA
iCPA$_{CL}$
iAMA$_{AL}$
iNE
Intelligent and Redundant Hardware Sub-Component$_1$
Intelligent and Redundant Software Sub-Component$_1$
Intelligent and Redundant Hardware Sub-Component $_N$
Intelligent and Redundant Software Sub-Component$_M$
Backup Copy of Current Configuration Files

Centralized Self-Managed SDN-II



Distributed Self-Managed SDN (Cont.)

- In distributed architecture, each layer will have its own Self-Managing NMS_S
- Self Managing NMS_S provides all the centralized management functions for its own layer and informs sNMS about its activities.
- End-to-end network level activities and activities involving multiple layers beyond a given layer will be left to Self-Managing NMS. These activities can be Connection Admission Control (CAC), load balancing, and end-to-end congestion control.

In-band Communications

A. A common message format for communicating failures and operational status whether it is LOS,AIS, LOF, RDI or operating system issues or protocol issues or hardware issues.

B. Communicating Failure Type to all related parties (i.e. sNEs, sNMS/sOSS, field technicians, etc.).

C. Indicating who will fix the failure, whether sNE or sNMS/sOSS or field technician.

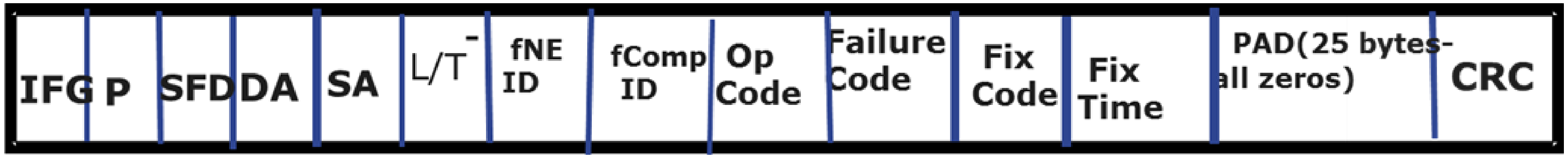
D. Indicating the time interval to repair. This will help sNEs, sNMS/sOSS, field technicians, and users to decide what to do during failures (e.g. store traffic or reroute traffic).

E. Communicating operational status of the failed component after repair.

Estimated Fix Time and Fix Communications

- Failure type, estimated fix time and failure status are communicated among Self Managing NEs, Self Managing NMS/OSS, and field technicians
- This should allow NEs to divert traffic from the failed NE(s), store messages of failed NE(s) if needed, warn Network Operations and users for the failure and allow them to prepare for the outage by indicating estimated fix time.
- The estimated fix time can be adjusted by field technician and NMS if needed, and communicated to NEs, NMS and users.

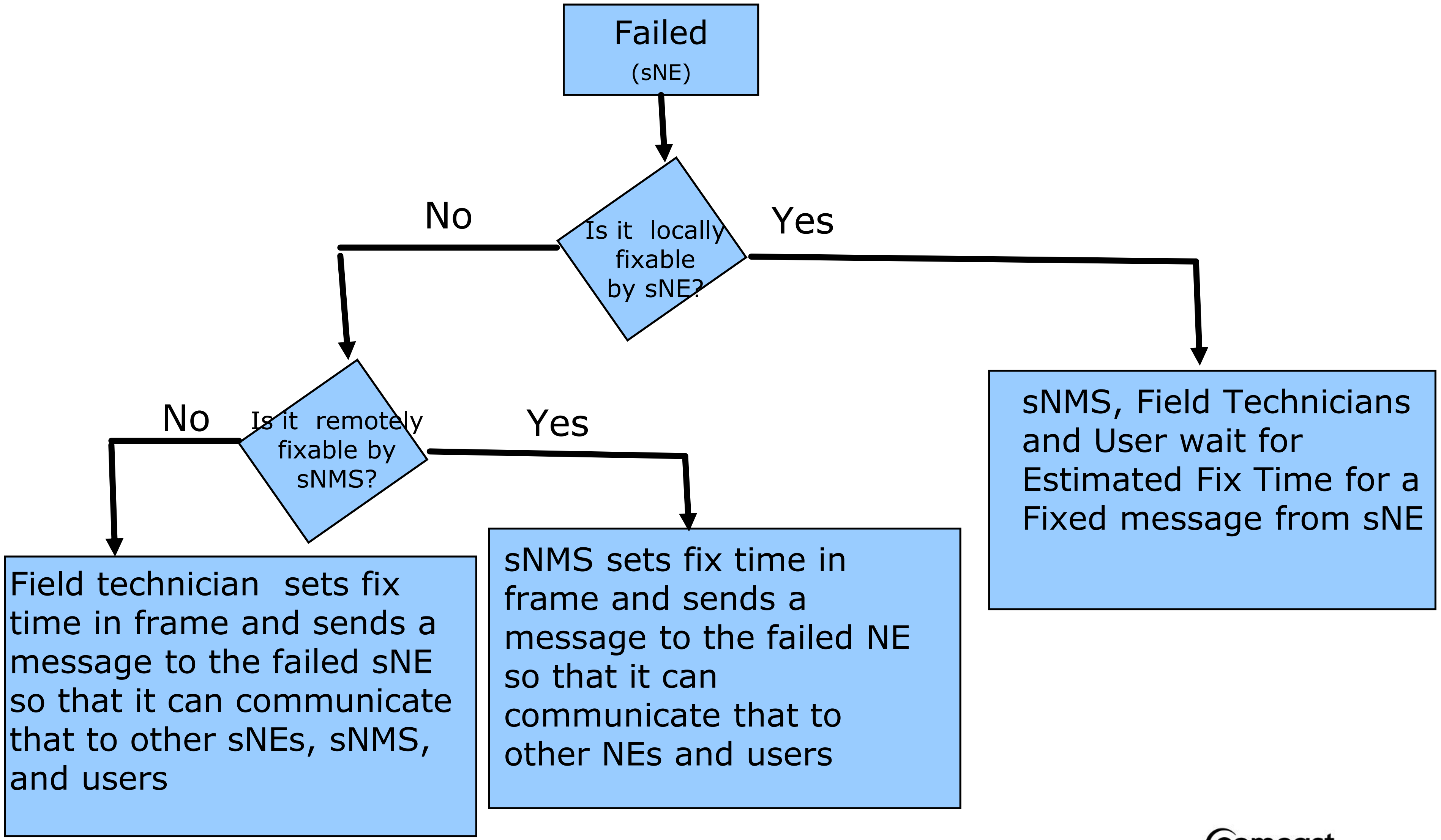
Frame for in-Band Communications of Failures, Fixes and Estimated Fix Times in CEN



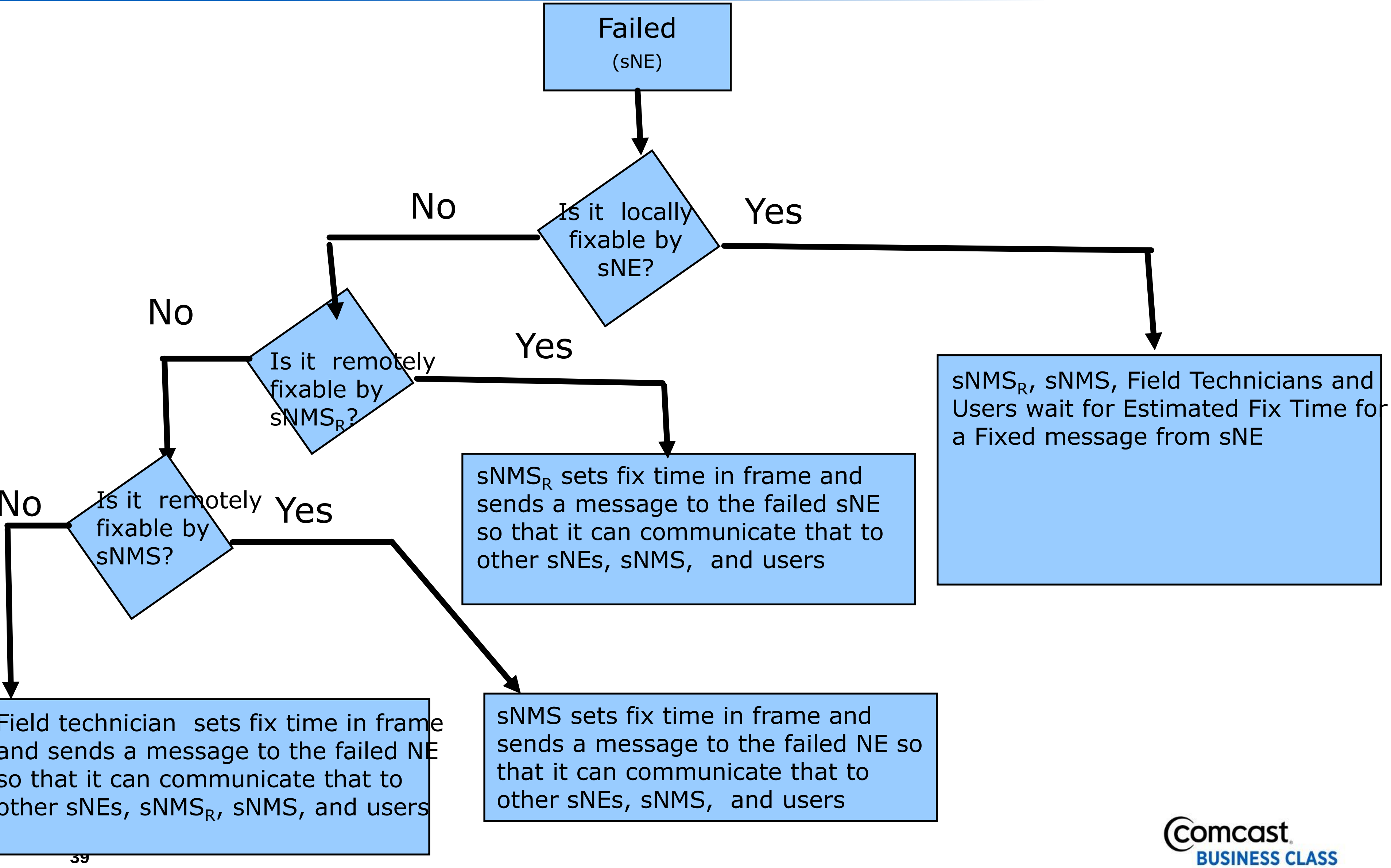
- IFG: Interframe Gap, 12 bytes
- P/SFD (Preamble/Start of Frame Delimiter)-8 Bytes (P-7 bytes, SFD-1 byte)
- L/T (Length/Type) : Length of frame or data type , 2 bytes (0x8808)
- CRC: 4 bytes
- DA: 01:80:C2:00:00:02 (6 bytes)-Slow protocol multicast address
- fNE ID: 6 bytes, Failed NE Identifier
- fComp ID: 4 bytes, Failed Component Identifier
- Op Code: 2 bytes-0x0202 for Disabled and 0x0303 for Enabled status
- Failure Code (ITU list+): 4 bytes
- Fix Code: 1 byte identifying fixing entity, sNE (x00), sNMS (x01), sNMS$_R$ (x02), field technician (x07), unidentified entity or inconclusive diag(x08)
- Fix Time in seconds: 4 bytes indicating fix time by sNE, or sNMS$_R$, or sNMS, or field technician

Comcast BUSINESS CLASS

Fault Management Hierarchy for Centralized Networks


Failed
(sNE)
Is it locally fixable by sNE?
No
Yes
sNMS, Field Technicians and User wait for Estimated Fix Time for a Fixed message from sNE
Is it remotely fixable by sNMS?
No
Yes
Field technician sets fix time in frame and sends a message to the failed sNE so that it can communicate that to other sNEs, sNMS, and users
sNMS sets fix time in frame and sends a message to the failed NE so that it can communicate that to other NEs and users

Fault Management Hierarchy for Distributed Networks


Failed
(sNE)
Is it locally fixable by sNE?
No
Yes
Is it remotely fixable by sNMS$_R$?
No
Yes
Is it remotely fixable by sNMS?
No
Yes

sNMS$_R$, sNMS, Field Technicians and Users wait for Estimated Fix Time for a Fixed message from sNE

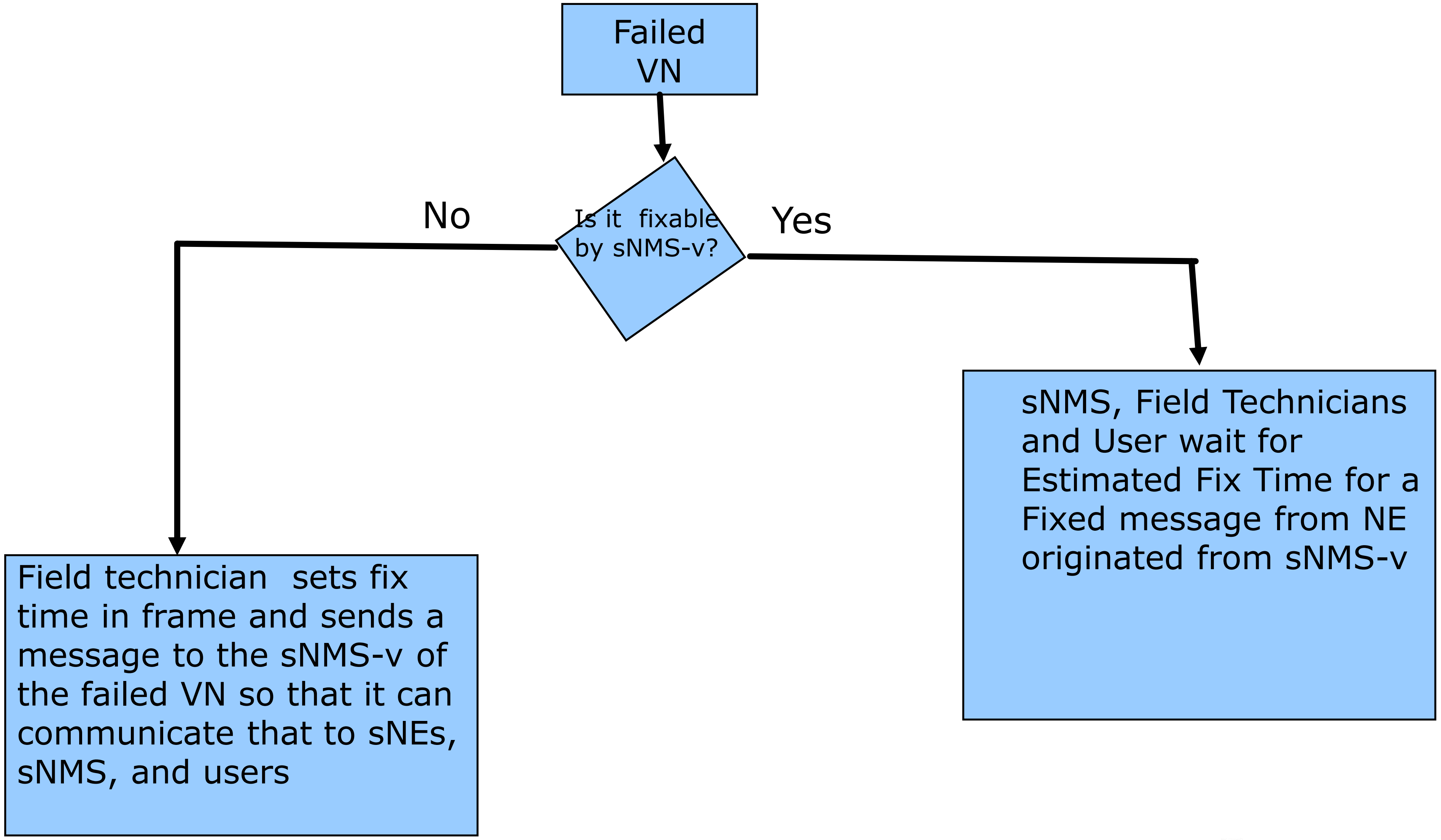
sNMS$_R$ sets fix time in frame and sends a message to the failed sNE so that it can communicate that to other sNEs, sNMS, and users

sNMS sets fix time in frame and sends a message to the failed NE so that it can communicate that to other sNEs, sNMS, and users

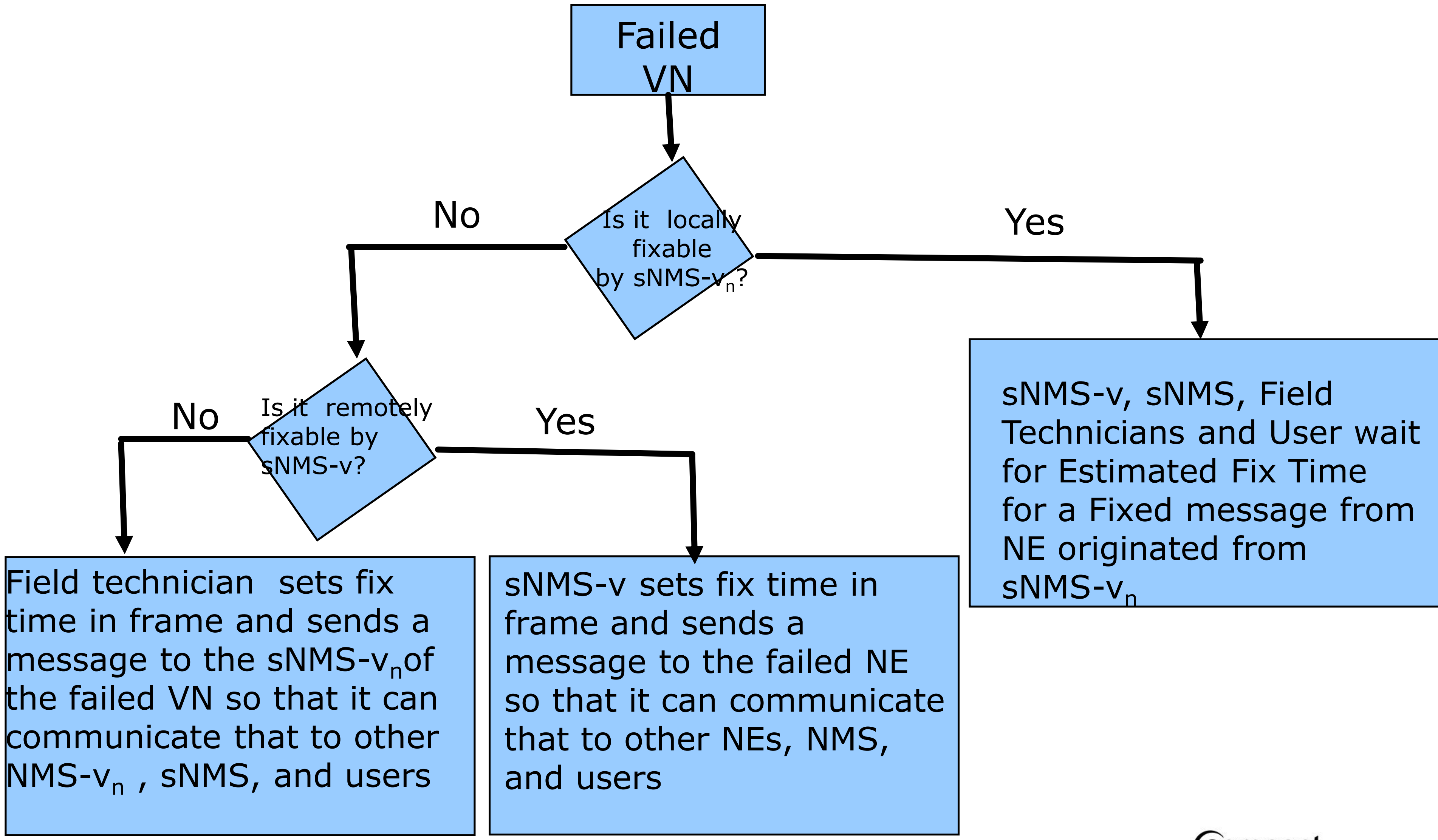
Field technician sets fix time in frame and sends a message to the failed NE so that it can communicate that to other sNEs, sNMS$_R$, sNMS, and users

Comcast BUSINESS CLASS

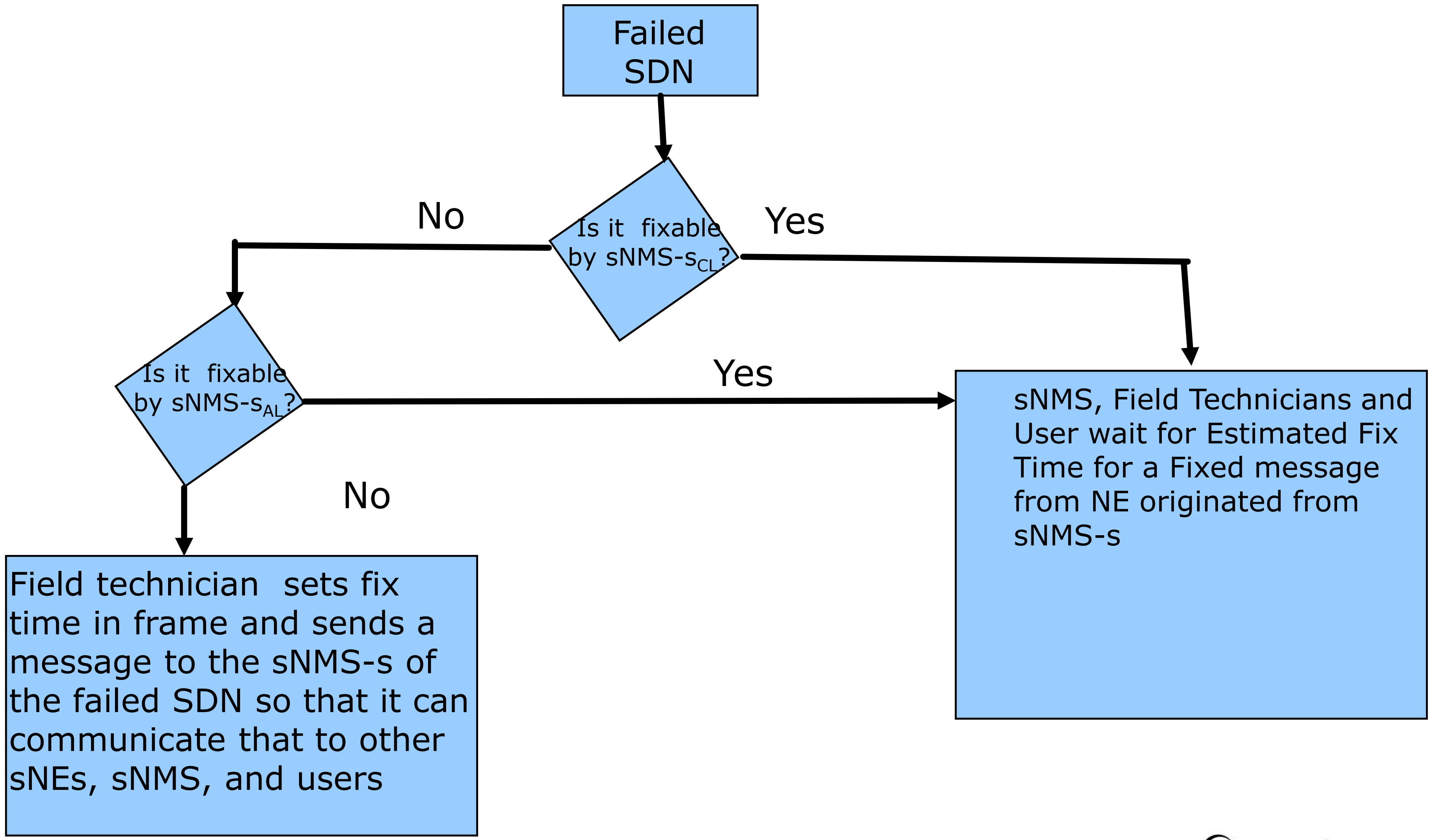
Fault Management Hierarchy for Centralized VNs


Failed VN
Is it fixable by sNMS-v?
No
Yes
Field technician sets fix time in frame and sends a message to the sNMS-v of the failed VN so that it can communicate that to sNEs, sNMS, and users
sNMS, Field Technicians and User wait for Estimated Fix Time for a Fixed message from NE originated from sNMS-v

Fault Management Hierarchy for Distributed VNs


Failed VN
Is it locally fixable by sNMS-v_n?
No
Yes
Is it remotely fixable by sNMS-v?
No
Yes
sNMS-v, sNMS, Field Technicians and User wait for Estimated Fix Time for a Fixed message from NE originated from sNMS-v_n
Field technician sets fix time in frame and sends a message to the sNMS-v_n of the failed VN so that it can communicate that to other NMS-v_n , sNMS, and users
sNMS-v sets fix time in frame and sends a message to the failed NE so that it can communicate that to other NEs, NMS, and users

Fault Management Hierarchy for Centralized SDNs



Failed SDN
Is it fixable by sNMS-s$_{CL}$?
No
Yes
Is it fixable by sNMS-s$_{AL}$?
Yes
No
sNMS, Field Technicians and User wait for Estimated Fix Time for a Fixed message from NE originated from sNMS-s
Field technician sets fix time in frame and sends a message to the sNMS-s of the failed SDN so that it can communicate that to other sNEs, sNMS, and users

Thank you!