

Levels of Autonomous Network Technologies

(draft-liu-nmrg-networkless-roadmap-00)

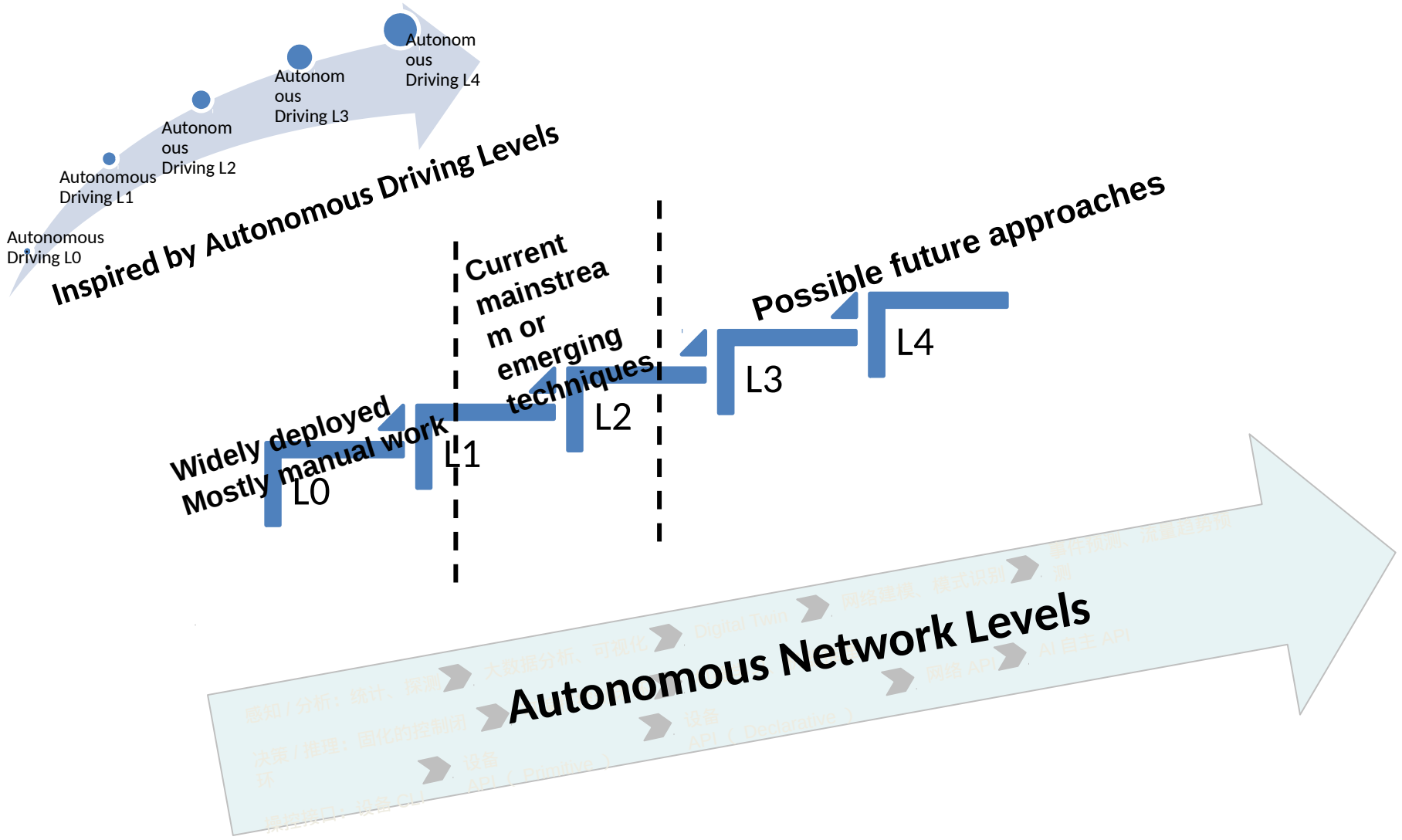
Bing Liu, Brian Carpenter

@NMRG, ietf102, July 2018

Background

- Network OPEX is always an important issue
- New network automation technologies are emerging:
 - SDN for configuration/service delivery
 - New emerging Intent-driven/based Network
 - AI for autonomous control, diagnostic etc.
- Maybe it's useful to draw an road map of how relevant technologies could evolve to the ultimate goal (named as "Networkless" in this draft)?

A Level-by-Level Roadmap



Two Dimensions of the Levels

- Final Effect in Operators' Perspective

- Self-Organization

- Bottom line: various Plug-n-Play techniques (joining in the domain, get connect with NMS, build adjacencies/routes etc.)

- Self-Configuration

- Bottom line: NMS/Controller automatically pushes configurations to devices

- Self-Optimization

- Bottom line: ECMP, TE etc.

- Self-Diagnostic

- Bottom line:

- Self-Healing

- Bottom line: VRRP, FRR etc.

- Key capabilities of Network Systems

- Network perception

- Bottom line: various measurement, telemetry and network visualization techniques

- Decision/Reasoning

- Bottom line: protocol-based control loops, machine learning based self-decision

- Operational interface

- Bottom line: primitive device API for software control/management

Effect (in perspective of Administrators)						Key Capabilities (of the network system)			
	Self-Organization	Self-Configuration	Self-Optimization	Self-Diagnostic	Self-Healing	Intent	Operation Interface	Decision/Reasoning	Perception/Analysis
L5	Self-Construction of Network Topologies (for wireless network or overlay virtual networks)	Network Self-Orchestration (Admins/Apps only input highly abstracted service request (e.g., build a wireless backhaul network), then the network would deduce all configurations.)	Autonomous Optimization (The network generates optimization policies by itself, and keep the performance at the best level; Meanwhile, achieve balance between performance and cost.	Fault Prediction	Fault Avoiding According to the prediction, avoid the fault by backup, adjust traffic etc.	Business Intent Natural language style formation to express the requirements; The network autonomously interpret the technical requirements to the network.	Machine-native Autonomous API The machines would autonomously construct the content of the APIs to fulfill the need of collaboration between modules.	?	Network Event Prediction Traffic Trend Prediction
L4	Network Architecture and NE roles Self-identification (E.g.: autonomically identify topology characteristics and divide network layers; autonomically identify roles such as access gateway, aggregation gateway, core gateway etc.)	NE Configs Auto-Compiling Admins design network architecture and solutions, the network autonomically compile detailed NE configs. All detailed configs are hosted by software. More and more machine-native configs rather than human interfaces.	Comprehensive SLA/QoS Self-Optimization The network autonomically optimize delay, bandwidth etc. according to admin or App's requirements; The network autonomically achieve measurement according to the optimization goal.			Precise Fault Location	Programmable Healing Admin can set specific healing policies based on a set of general and abstracted rules of dealing with fault.	Network Intent Describe the network architecture/solution s/policies in a certain abstracted formation (e.g. program language).	Network-level Declarative API User/Admin oriented declarative API, to make the network be called as a service.
L3	Network Areas Self-Division and Key NEs election (E.g.: IGP Area self-division; controller election)							NE Intent Describe the NE-level policies such as config policies, config goal, optimization goal in a certain way that the NE can directly interpret it.	NE-level Declarative API Controller oriented NE-level declarative API
L2	NE Plug-N-Play (NEs automatically get connected with the NMS, current solutions includes DCN, Anima ACP, ZeroTouch etc.)	NE Configs Auto-delivery (Admins design detailed configs of each NE, NMS/Controller automatically deliver the configs.)	Auto Traffic Load Balance Controller dynamically adjust paths to achieve balanced traffic load, according to specific algorithms; NE can achieve port-based load balance locally	Automatic Data Analysis Software collects data around the whole network, and use data mining/machine learning and decision tree to aggregate alarms and analyze the cause.	Protocol-based Healing (Fixed healing functions built into NEs, such as BFD, FRR etc.)	N/A	NE-level Primitive API Controller oriented NE-level API containing detailed configurations. (E.g. Openflow, Netconf/YANG)	Programmable Control Loops Algorithms (in Controller) for specific functions and scenarios (might embedded some Machine Learning capabilities.)	Big Data Analysis Telemetry, Network Visualization, and logs analysis etc.

Open Discussion

- Is it valuable to do such a work?
- The two dimensions are reasonable?
- Any thing missing in the “Self-X” part; and also the “Key Capabilities” part?
- Other specific comments

Next step

- Feedback are highly needed
 - This work cannot be done by only few people
 - Reaching consensus of the industry in the goal
- Might consider a more intuitive result for each level

Comments are highly welcomed

Thank you!

leo.liubing@Huawei.com
brian.e.carpenter@gmail.com

IETF102, Montreal