

Setting the Scene

Reference: draft-arkko-arch-virtualization-01

*Arkko, Tantsura, Halpern, Varga
+ numerous contributors*

Starting Point

- There are a number of existing (and evolving) tools
- Virtual networks, network function virtualization, software-defined networking, service chaining, data models, traffic engineering, MPLS, QoS mechanisms, deterministic networking tools, orchestration, service-based architectures, application middleware, data center networking tools, ...
- Our day jobs are about building systems out of these lego bricks
- ... and we will also be providing “slices” in 5G networks
- Are these the same thing, or different, and if latter, how?

General Goals

- **Separation of concerns**
 - Providing tailored services
 - Separation (of traffic, security, resources, ...)
 - Resource allocation/reservation
- Independent **technology evolution**
- Ability to **benefit from modern IT** technology practises (cloud, virtualization, ...)

Ok, can this be achieved with what we have? What do we have?

Virtualization, Slicing & Protocols

- **Virtualization does not generally affect TCP/IP or applications**
- **Some exceptions to this**, when assumptions made somewhere are broken due to virtualization, leading to a need to add information to application protocols:
 - E.g., early HTTP versions assumed that 1 server = 1 website
 - With virtual hosting, modern HTTP versions carry intended web site name inside the protocol

Virtualization & Slicing Tech @ IETF

- **Instance selection** at lower layers
- Provider-based **VPNs**
 - MPLS, L2-3VPN, NVO3, ...
 - Traffic engineering, e.g., TEAS WG
- **Service chaining** — SFC WG's NSH
- **Management frameworks** — e.g., NETCONF, YANG
- **Data models** — e.g., L2SM, L3SM

Architectural Observations 1

- Trend: **Increasing role of software**
 - In many cases, this replaces the need for protocol mechanisms
- Trend: **Centralization of functions** — makes things easy
 - Still need to work even if the “center” is down
- Observation: **Stark complexity contrast** between selection/packet processing/networking and orchestration/creation/management
 - “Execution Plane” vs. “Creation Plane”
 - Example: 5G slice selection (NSSF selects and redirects to appropriate AMF) vs. actually setting up the slices

Architectural Observations 2

- Question: **Tailored vs. general-purpose networking**; what are the economics of special-purpose treatment and QoS?
 - It is possible that the industry at times gets over-excited about offering everyone added value... there's also a great economic benefit to bulk
 - Tuning one infrastructure to server multiple different categories of customers is fine, however
- Question: **What needs require something new?**
 - There are plenty of QoS tools, virtualisation platforms, orchestration mechanisms, and data models or other descriptions at varying levels. What's missing?
- Observation: new systems (like 5G) have specific goals, but ultimately, those **goals are fulfilled through a combination of the current tools and (some) new mechanisms** or enhancements... not through redesigning everything

Architectural Observations 3

- Advice: Think about **data model layering**! E.g., service vs. network/device data models
 - May need appropriate tools for different layers
 - And there are multiple tools, YANG, Tosca, ...
- Advice: Think about what is needed to for a working, **interoperable** system that maps layers of models to each other.
 - Merely the models + magic software? Or common software? Or common specification or data that the software can do its magic?

Architectural Observations 4

- Advice: **General over specific** — does it make sense for IETF to do general designs or designs for someone's specific requirements at specific time?
 - IETF probably wants to do tools that work across industry as opposed to only for 5G (no matter how important it is)
- Some **terminology and conceptual alignment across industry** would also be useful, e.g., to know what words to use in SLAs...

Some 5G Slicing Use Cases

- Many **simple cases** (QoS, access to specific networks, etc)
 - Existing tools often sufficient for these
- Everything runs on top of virtualisation and cloud platforms
 - Mobile networks running on the same tech as other applications
- One interesting case that demands interest tech is **servicing a factory that requires very low-latency network** between its machines
 - May need to build a separate instance of 5G core on site, using virtualisation, cloud, orchestration and other similar tools + hardware on site

Some 5G Slicing Use Cases

- Another interesting case: for <these users>, **run a completely new version of 5G** core network
 - Important for evolving tech quickly
 - Also useful for using competing suppliers or providers
- Some more demanding cases
 - Have the ability to control resources for a slice in **both radio and core network**
 - Provision a network (slice) **across administrative boundaries**

Thoughts for Going Forward

- Consider **all the technology!**
 - Virtualization, separation, resources, management & orchestration; in and outside IETF
- Find the **concrete missing things** that still need doing
 - Connect the top-down and bottom-up way of thinking
- There are relevant existing and new topics to work on at the IETF:
 - **Data model** development — e.g., work on commonly used data models at several layers of abstraction
 - End-to-end, heterogeneous networks, **cross-domain**
 - **How our different pieces fit together**

Reading List

- draft-irtf-nfvrg-gaps-network-virtualization

Good summary of the various technologies, plus a discussion areas that need further development

- RFC 8309

Service models explained

- draft-geng-coms-architecture

Architecture view to slicing as an orchestrator function, how slicing relates to NFV, etc.

- draft-netslices-usecases & draft-qiang-coms-use-cases

Use cases

- draft-wu-model-driven-management-virtualization

Model drive-management and layered data models

- draft-bryskin-teas-use-cases-sf-aware-topo-model

Good example of enhancing IETF-based management data models with additional information

- ...