Learning with Nephio R1 - Episode 2 - Why Nephio?

Author: John Belamaric

Prerequisites: Episode 1 - Series Introduction

Slides: 2023-07 Nephio R1 Concepts Series - Episode 2.pdf

Video: https://youtu.be/dQKlT4ik198

Why Nephio?

In the enterprise space, the emergence of the API-driven, on-demand public cloud has dramatically changed how applications are built, deployed, and managed. The resulting evolution of “cloud native” tools and techniques, like containerization, Kubernetes, and the CNCF ecosystem, has increased developer agility while improving utilization of the underlying infrastructure.

The development of multi-access edge computing (MEC) promises to extend this advantage to the edge, opening up new opportunities for communication service providers (CSPs). By placing specific workloads closer to the user, new applications that depend on low-latency networking and user locality are enabled. By supporting containerized 5G network functions (NFs), and the ability to run additional instances of those functions on the same hardware, MEC also creates the possibility for compelling use cases such as bespoke networks for enterprises or other customers. Use of an API-driven, on-demand MEC can enable these new use cases while also ensuring efficient utilization of the hardware and other infrastructure.

However, building and operating scalable edge applications, including 5G networks, across distributed geographical locations is complex. Sites vary in nearness to the user in terms of geography and latency, and in terms of the availability, capacity, and cost of compute, network, and storage. We can simplify the problem by categorizing the sites into tiers based upon resource availability, cost, nearness to the user, latency, bandwidth, and available hardware, but even then the categorization will vary across CSPs and use cases. Depending on how far to the edge these tiers extend, tiers can range in number of sites from less than 100 to over 100,000. Provisioning application and network function workloads across these sites requires choices in the placement of workloads, configuration of the workloads and necessary infrastructure, as well as continuous management of the life cycle, health and scaling of those workloads. Increasing the complexity is the interrelated nature of some workloads - in particular network functions - where configuration changes to one workload necessitate changes to many related workloads.

Each layer of the stack - compute, storage, intrasite networking, inter-site networking, workloads, and configuration of those workloads - are managed by different systems and different teams. This exacerbates the complexity concerns. Existing methods require teams to coordinate ahead of time to determine the overall design and even the specific inputs to render workloads at each site. If there are 100 inputs to 20 workloads across each of 10,000 sites, there is a need agree up front on 20,000,000 inputs. Existing infrastructure-as-code techniques are not designed to handle this level of scale and complexity, so new techniques are needed.

Nephio proposes three basic pillars to manage this complexity:

- **Consolidate on a single, unified platform for automation**
  - Across infrastructure, workloads, workload configs, vendors and deployment tiers.
  - Integration with existing tooling, but via a common mechanism is critical.
- **Declarative configuration with active reconciliation** to support day one and day two.
  - Break large topologies into smaller components with autonomous controllers.
  - High-level eventual state emerges from interaction of lower-level component states
  - Distribute state (intent) across geography for resilience.
- **Configuration that can be cooperatively managed** by machines and humans.
  - Reduce the coordination overhead between teams by enabling independent decision making
  - Machine-manipulable configuration is fundamental to automation.

Each of these will be explored in depth in later articles.

About the Nephio Community

Nephio is an open source project of the Linux Foundation. We do our best to be a friendly, welcoming community. We have several different "special interest groups" (SIGs) that meet regularly, as well as mailing lists and a Slack instance. Please join us!

- Learning page - https://nephio.org/learn
- Site - https://nephio.org/, https://nephio.org/about/
- Wiki - https://wiki.nephio.org/
- Docs - https://github.com/nephio-project/docs
- Blog Postings - https://nephio.org/blog/
- Slack - https://nephio.slack.com/ (join via this invitation)
- Project Github - https://github.com/nephio-project
- Project email distro - https://lists.nephio.org
  - nephio-tsc (for TSC members and interested parties)
  - nephio-dev (for all)
- SIG lists: sig-netarch, sig-automation, sig-release