



专栏：5G+时代的可编程网络

面向 6G 的新型可编程网络架构研究

王旭亮，全硕，刘增义，章军
(中国电信股份有限公司研究院，北京 102209)

摘要：从业界对 6G 网络的多种业务场景入手，深入分析了各种业务场景中对 6G 网络设备所需要的动态网络能力调整需求。提出了一种具备根据网络状态管理进行闭环自优化特点的可编程网络平台架构。通过该架构设计可以最大化利用与 SDN、NFV 以及网络 AI 相关的已有技术体系，同时可以让网络管理者基于网络状态声明的方式进行网络的生命周期管理。该平台设计在未来可以为数字孪生网络以及随愿网络提供一种新的技术实现思路和方法。

关键词：6G；可编程网络；软件定义网络

中图分类号：TP393

文献标识码：A

doi: 10.11959/j.issn.1000-0801.2021266

Research on architecture of novel programmable networks for 6G

WANG Xuliang, QUAN Shuo, LIU Zengyi, ZHANG Jun
Research Institute of China Telecom Co., Ltd., Beijing 102209, China

Abstract: Starting from the industry's various business scenarios for 6G networks, and the dynamic network capability adjustment requirements for 6G network equipment were deeply analyzed. A programmable network platform architecture with the characteristics of closed-loop self-optimization based on network state management was proposed. Through this architecture design, the existing technology systems related to SDN, NFV, and network AI can be maximized and network administrators can perform network lifecycle management based on network status statements. This platform can provide a new technical realization idea and method for the digital twin network and on-demand network in the future.

Key words: 6G, programmable network, software defined network

1 引言

6G 网络将构建横跨天地的网络连接，实现从人口覆盖走向地理全覆盖，进一步驱动产业技术和生产方式向跨界协同和智能互联方向迈进。6G

应用场景将比 5G 更为广泛，将传统的 eMBB、mMTC、uRLLC 场景进行融合^[1]，将涵盖极高吞吐量 and 极低时延需求的全息通信及扩展现实体验；超高实时性及可靠性需求的人体数字孪生；超高移动性及全覆盖需求的空中高速上网；智能超连

通性、内生智能以及安全需求的新型智慧城市；基于 AI 自主运行需求的联网无人驾驶；超高带宽、超低时延和超可靠等需求的高精度智能工业；全覆盖需求的全域应急通信抢险等。以上所有对 6G 网络的指标要求都需要通过网络设备具体实现，可编程网络作为 6G 网络的三大关键能力技术之一^[2]，是满足上述网络指标的关键技术。通过网络可编程技术可以把超大带宽、超低时延以及超大并发与可靠性的具体指标组合技术要求，通过网络可编程的技术^[3]把上述指标组合通过逐层拆解和建模的方式下发到网络的控制面与转发面设备中，从而实现 6G 端到端网络对上述应用的综合承载。

6G 可编程网络最终是为数字孪生的网络自治体系服务的^[4]。数字孪生技术是指通过数字化手段将物理世界实体在数字世界建立一个虚拟实体，借此实现对物理世界实体实现动态观察、分析、仿真、控制与优化。数字孪生网络技术包括功能需求建模、网元建模、网络建模、网络仿真、参数与性能模型闭环优化、自动化测试、数据采集、大数据处理、数据分析、机器学习、故障预测以及拓扑与路由调优等，从而把网络每个阶段遇到的难题转换到数字世界求解，通过监控、预测、优化、仿真实现网络的自治能力。其中，在网络建模、网元建模、参数与性能模型闭环优化以及拓扑与路由调优等环节都需要可编程网络技术实现对相关模型与参数偏差的自我调整。例如，通过数字孪生建立一个虚拟的自优化网络环境实现对未来网络状态的走势提前预测，对可能发生的性能劣化提前干预，同时该数字孪生网络持续地对物理网络的最优状态进行优化和仿真验证，提前下发对应的运维操作自动地对物理网络进行校正。自优化网络首先基于人工智能对网络功能的演进路径进行分析和决策，包括既有网络功能的优化增强和新功能的设计、实现、验证和实施。

6G 可编程网络可以实现数字孪生网络对网

络业务模型、网元模型、网络状态感知、网络资源动态调整以及网络故障快速处理，是连接数字孪生网络与现实网络的重要桥梁。6G 可编程网络包括网络业务模型的可编程、各类网元设备的可编程、端到端网络状态精确感知、网络资源的动态调整以及网络故障的快速定位与处理。网络业务模型的可编程是通过网络功能虚拟化编排器（network function virtualization orchestrator，NFVO）承载和实现的。网元设备的可编程包括控制面的可编程以及转发面的可编程。网元控制面可编程主要由软件定义网络（software defined network，SDN）控制器实现，SDN 控制器接收到 NFVO 的业务编排指令后，会把相应的网络配置、拓扑配置、路由配置、运维策略等参数下发到已注册到该控制器的网元设备上。其中，虚拟网元的场景还需要 NFVO 配合虚拟网络功能管理器（virtual network function manager，VNFM）和虚拟基础设施管理器（virtual infrastructure manager，VIM）完成其创建和初始化过程^[5]。网元设备转发面的可编程分为两种情况：第一，虚拟网元设备主要承载在 x86 芯片和 ARM 芯片平台上，以开源设备 FD.io 里的矢量包处理机（vector packet processor，VPP）为例，主要是通过开发相应芯片的驱动程序实现对芯片 SDK 的调度，从而实现对数据包的各类转发操作；第二，物理网络设备，即白盒交换机，主要是通过设备厂商或第三方开发应用驱动程序调度可编程交换芯片^[6]在硬件平台实现数据包转发流水线的定制化开发。端到端网络状态精确感知当前主要依赖 Telemetry 技术^[7]对端到端的网络设备进行逐个的状态数据采集。虚拟网络资源的动态调整依赖 NFV MANO 技术，物理网络资源的调整依赖 SDN 控制器根据业务负载进行多设备间的流量调度。网络故障定位与处理可以通过 SDN 控制器实现，更多的情况下仍需要大量依赖网元设备厂商提供的网关系统实现。

为了系统化地实现 6G 网络的可编程网络总



体目标,本文提出一个对网络全要素具备灵活感知、由真实网络状态模型驱动、可实现全自动闭环控制的可编程网络整体架构。该架构包括三大部分内容:第一,目标网络状态声明和当前网络真实状态。通过循环比较这两个状态的信息,实现可编程网络的全自动闭环控制。目标网络状态由管理员手动填写模板生成或由数字孪生模拟系统下发,当前网络真实状态由可编程网络管控系统根据当前网络的状态数据动态生成。第二,可编程网络的管控系统包括4个基本模块,负责把当前网络状态调整到目标网络状态,分别为网络业务编排器、网络控制器、网络状态采集器以及网络故障分析与处理器。第三,各类网元设备,主要包括虚拟网元设备和物理网元设备。

通过本文在网络全要素感知、网络状态模型驱动以及全自动闭环控制的可编程网络框架方面的研究探索,希望在6G网络架构设计与关键技术探索等方面提供技术参考。

2 相关研究

本文提出的面向声明式的目标网络状态,由可编程网络框架中的多组件互相配合实现的思想借鉴了部分云原生技术中的声明式应用程序接口(application programming interface, API)的理念,同时考虑到业界也有相关的研究成果,下面将通过举例类比的方式,分析本研究与其他相关研究的区别。

NetEgg^[8]是一种面向网络管理员期望策略由SDN控制器自动化生成网络控制策略的编程框架。该文重点介绍了在防火墙场景中,根据管理员提出的网络访问控制策略由控制器自动生成防火墙相应配置代码并配置下发的全实现流程。这与本文提出的根据网络目标状态,通过可编程的方式实现网络自治的目标是有一定关联性的,但是NetEgg更侧重于某个管理员期望的具体网络行为如何转化为网元的具体配置命令,且属于一

种不具备闭环自动化的代码生成工具,相比本文提到的可编程网络平台在端到端网络服务层面进行网络自治存在较大的差异。

YANG认为,为了可编程开发无数个面向特定领域的网络控制器^[9],不如在已有的各类控制器之上做统一的能力扩展。提出在多种控制器系统之上建立一个分布式的抽象文件系统,通过该文件系统把网络的配置和状态通过文件的形式提供给上层应用(用户空间进程、cron作业、命令行实用程序等)读写调用。从而让研究者投入更多的精力到上层的高价值应用开发上。本文提出的可编程框架中,为了进一步摆脱以Netconf/YANG等传统网络的束缚,最大化挖掘可编程网络的维度和潜力,控制器与虚拟网元之间也采用基于数据库文件方式的管理方式。通过在虚拟网元转发面与控制器之间编写NoSQL数据库文件进一步提升控制器对集群化的虚拟网元并发管理能力以及管控指令的执行效率等。

本文提出的可编程网络框架与平台与业界的意图网络存在一定的联系,但侧重于不同的领域。根据最新的关于意图网络的调研^[10],在过去的十年时间内业界主要通过深度学习的相关技术试图进一步推进意图网络在意图定义、意图分解、意图建模以及意图执行等方面的学术意义上的进展。本文提到的可编程网络框架只需要网络管理员通过模板人工填写或由意图网络给出一个声明式的意图网络目标描述。可编程网络框架的总体控制器会协调不同控制模块以及这些控制模块间接控制的网元设备把网络初始化,并持续调整真实网络状态到期望的目标网络状态,因此可编程网络可以为意图网络服务,但不必然属于意图网络自身的一部分。按照当前技术成熟度和产业关注度来看,6G是可编程网络的一个重要应用领域,且可编程网络的技术理论已经相对完备,目前比较缺乏整体性的原型试验验证可编程网络的理论完备性和工业方面的可行性。

3 新型可编程网络架构

本文提出的面向 6G 的新型可编程网络架构借鉴了云原生领域关于声明式 API 的设计思想。网络管理员只需要一定的格式定义期望的网络状态声明,由可编程网络平台自动化闭环迭代控制真实网络系统实现上述状态。通过该可编程网络平台,能够快速实现网络按需编排、部署、资源弹性伸缩、动态路由规划与下发、网络故障自动发现与修复,使能 6G 网络满足多种业务网络承载需求。

可编程网络平台的整体架构如图 1 所示,包括目标网络状态声明、实际网络状态、网元编排器、网络控制器、网络监控器、故障分析处理器、网元设备等模块。可编程网络平台接收到目标网络状态声明通过闭环自优化机制协调网元编排器和网络控制器管理网元设备、网络链路的生命周期及配置,协调网络监控器和故障分析处理器实时监控网络运行状态并进行故障处理和优化,协同网元设备完成数据的转发,使网络的真实状态和目标网络状态声明始终保持一致。

可编程网络平台最核心的部分来源于网络服务的闭环自优化机制。网络服务闭环自优化机制架构如图 2 所示,该机制分为 4 个层级,分别是网络服务状态管理层、协调与控制服务抽象层、

协调与控制服务执行层以及网元转发设备层。网络服务状态管理层的目标是驱动实际网络服务状态不间断地向目标网络服务状态过渡。协调与控制服务抽象层的目标是根据网络服务状态自优化控制器的指令,把一个网络服务整体拆分为网络资源、网络资源关系以及网络优化策略,分别由 3 类控制器负责具体的执行。协调与控制服务执行层的目标是根据抽象层的指令,由网元编排器实现物理网元和虚拟网元的生命周期管理,由网络控制器实现网络链路的生命周期管理,由网络监控器实现对网元设备和网络链路的实时监控、触发并上报网络故障告警以及对网络调优结果进行反馈,由故障分析处理器根据抽象层事先定义的网络故障与调优处理策略进行相关故障和调优处理。网元转发设备层主要是指支持可编程交换芯片的白盒物理转发设备以及基于虚拟机或容器的虚拟网络转发设备。

3.1 网络服务状态管理层

可编程网络平台会根据网络管理员声明的网络状态自动化地指导并调整真实网络状态,即系统会周期性地对比网络服务的实际状态和预期状态,自动调整网络服务向预期状态过渡,从而保证网络服务状态的最终一致性。在这一过程中,网络服务状态自优化控制器起到了管理网络服务状态和过渡过程的关键作用。

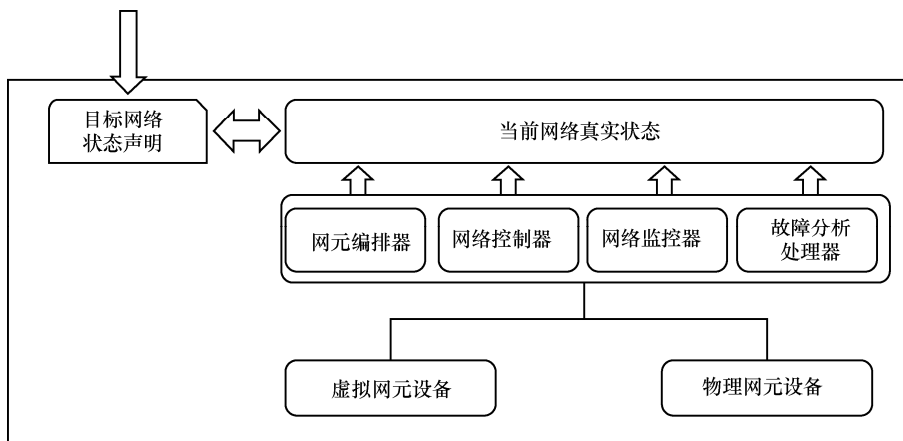


图 1 可编程网络平台的整体架构

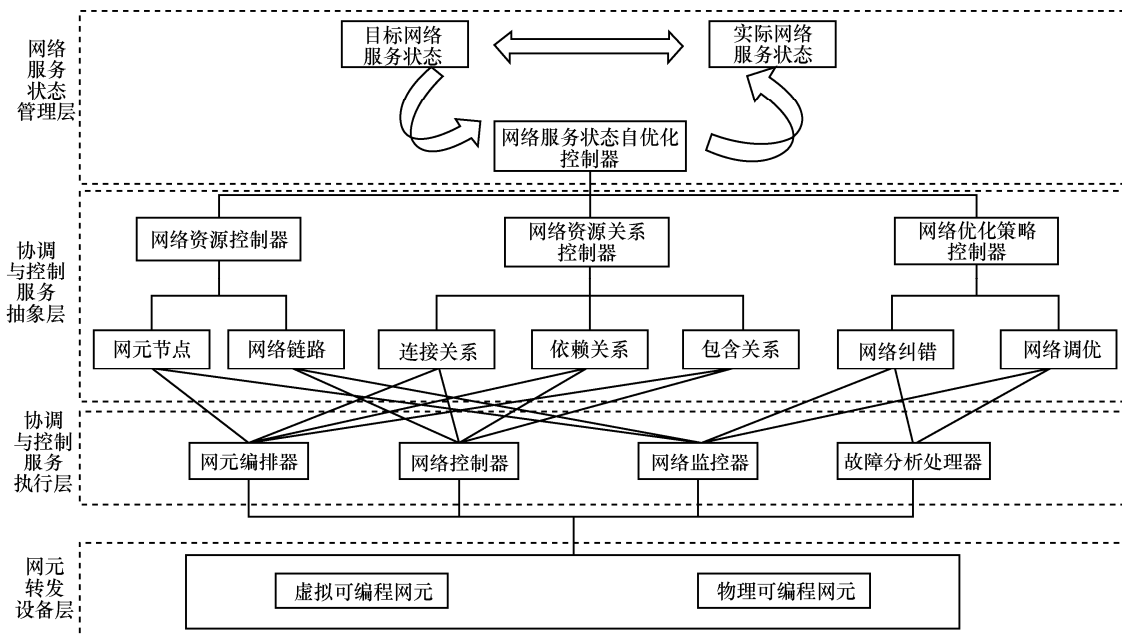


图2 网络服务闭环自优化机制架构

网络服务状态自优化控制器定义了一个框架，根据具体的功能可以再划分为网络资源控制器、网络资源关系控制器、网络优化策略控制器等，当需要增加新的控制器时，只需要按照网络服务状态自优化控制器定义的流程和规范实现对应的接口，即可将控制器纳入框架。网络服务状态自优化控制器处理流程如图3所示，主要包括以下几个步骤。

(1) 网络服务初始化时，控制器对服务配置

进行校验，并记录网络服务的配置和预期状态。

(2) 经过校验后，网络服务进行实际的部署操作，控制器对操作结果及状态进行数据持久化。

(3) 控制器启动监听服务，定时同步网络服务的实时状态。

(4) 控制器将网络服务的实时适配及状态进行数据持久化，更新实时数据。

(5) 当网络服务数据持久化中的状态发生变化时，控制器接收状态变更的通知。

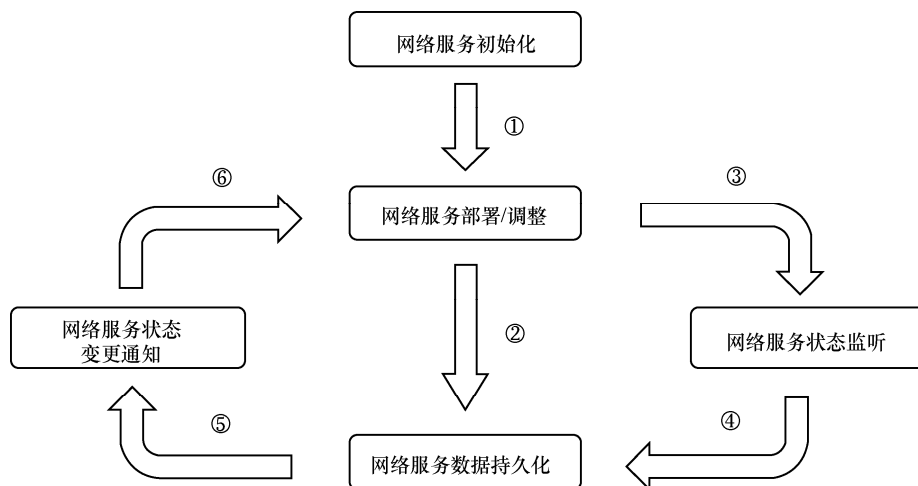


图3 网络服务状态自优化控制器处理流程

(6) 控制器根据网络服务当前状态和预期状态, 判断网络服务部署的调整策略, 使网络服务逐渐达到预期状态。

在以上各步骤中, 网络服务状态自优化控制器仅定义各个阶段的接口规范, 而不定义具体的参数及实现, 这种开放式的设计提高了控制器的可扩展性。

3.2 协调与控制服务抽象层

为了实现网络的声明式特性, 关键的一环就是网络资源的抽象化和模板化, 将网络资源的配置、关联关系、运行状态等通过统一、规范的语言描述, 从而达到整体网络可编程的效果。在本文提出的可编程网络框架中, 将网络抽象为 3 种资源: 节点、关系和策略。

“节点”是网络资源实体的抽象, 包含多种类型。它可以是一台物理交换机、一个虚拟网元, 甚至任何暴露操作接口的黑盒设备, 可以通过更改“节点”的状态部署和更改网络业务。“节点”通过以下关键属性描述和操作底层设备。

- 配置: 每种类型的节点都有各自的配置, 用于对节点进行操作。例如, 交换机节点具有“管理 IP 地址”这样的配置。
- 生命周期: 节点从创建到销毁的状态变化过程, 一般包括创建、实例化、更新、自愈、销毁等, 节点根据实际业务可以自定义生命周期及其实现方式。
- 控制器: 节点在生命周期操作中会发生状态的变化, 控制器用于维护节点的状态变化过程, 保证节点状态符合预期。

“关系”是网络拓扑和业务逻辑的抽象, 用于描述节点与节点之间的依赖关系、业务操作的先后顺序等。网络业务的下发通常涉及多个物理或虚拟网元及链路, 仅通过节点难以定义这种稍微复杂的业务场景。引入“关系”后, 可以实现节点拓扑的描述、节点依赖关系的维护、节点之间的参数传递。此外, 还可以根据实际业务自定义

关系, 例如, 自定义一个纳管关系, 在网络设备实例化后将设备连接到对应的管理系统并提交设备信息。

“策略”是针对一组节点在满足指定条件时执行的操作, 用于节点控制的自动化。对于需要根据实际运行情况而调整的业务, 通过节点和关系无法定义其调整逻辑, 策略的引入可以解决这类问题。策略包含目标节点、触发条件、执行动作 3 个要素, 通过三要素的组合可以实现诸如网络纠错、网络自动调优等自动化和智能化的业务。

基于以上 3 个基本元素, 可以将任意网络服务拆解成节点、关系、策略的组合, 即服务模板。系统通过服务模板描述、操作和维护实际的网络服务, 从而实现网络服务的可编程。

3.3 协调与控制服务执行层

网络资源经过模板化定义后, 在协调与控制服务执行层完成指令的具体执行操作, 使网络的实际状态达到声明状态。协调与控制服务执行层主要分为 4 个模块, 分别是网元编排器、网络控制器、网络监控器和故障分析处理器。

(1) 网元编排器

网元编排器实现了物理网元和虚拟网元设备的生命周期管理, 包括网元资源的创建、分配、回收、销毁等, 根据协调与控制服务抽象层中网元资源以及网元资源关系的声明情况, 按照顺序执行相应的操作。

网元资源的创建: 对于物理网元主要是将其纳管, 可以分配给用户使用; 对于虚拟网元, 需要根据资源需求、网元与虚拟机的亲和性与反亲和性关系、优先级等调度策略, 选择物理机/虚拟机节点创建。

网元资源的分配: 如果是共享网元资源声明, 则根据声明中网元资源的要求以及可编程网络平台各共享网元的负载情况, 按照一定的编排策略分配资源; 如果是独享网元资源声明, 则按需创建后分配。



网元资源的回收：将运行在网元上的网络业务删除，释放资源；如果是独享网元资源，在终止业务后，还需要进行资源的销毁。

网元资源的销毁：对于物理网元将其从资源可用列表删除；对于虚拟网元，删除网元并将资源归还给物理机/虚拟机。

（2）网络控制器

网络控制器实现网络链路的生命周期管理。根据协调与控制服务抽象层定义的网络链路资源以及相关连接依赖包含关系，依次给网元设备下发配置，完成网络链路的创建、更新、优化、删除等。网络控制器架构如图 4 所示，分为 3 个层级：北向 API 层、控制层和南向下发层。

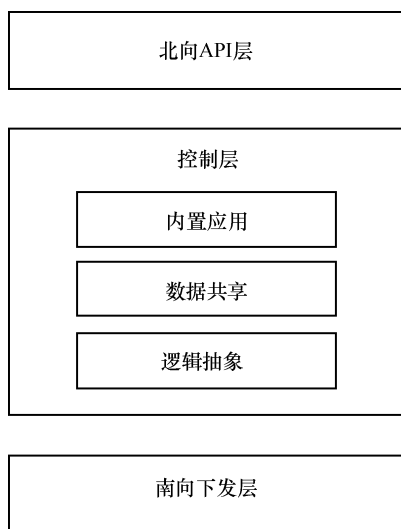


图 4 网络控制器架构

北向 API 接收业务编排器下发的业务请求；控制层包含内置应用、数据共享和逻辑抽象模块，内置应用模块负责将网络业务请求分解成二层、三层网络服务对象，并进行拓扑连接、路径计算等基础网络管理。数据共享模块通过定义标准化的数据结构、数据模型，使其在不同协议、不同网络设备间实现数据共享。抽象逻辑模块实现南向接口层和北向数据共享层之间的解耦，保证设备的适配和数据的建模可以独立演进，互不影响；南向下发层主要将网元设备能理解的配置和表项

下发到网元设备。

（3）网络监控器

网络监控器通过采集网元设备、网络链路、网络流量等关键指标、告警、日志数据，实时监控当前网络的运行状态，在出现故障时能及时触发报警机制及事件响应，将其反馈到网络故障分析处理器，保证故障的快速解决和网络状态的快速恢复，是实现可编程网络平台自愈自治能力的基础。同时，网络监控器把网元设备、网络链路等资源的使用情况实时反馈给网络编排控制器，为网络编排器进行资源编排、网络控制器实现配置下发提供依据。

（4）故障分析处理器

网络故障分析处理器是实现网络服务闭环自优化机制的关键模块。根据协调与控制服务抽象层预先定义的网络纠错和调优策略，在发生网络故障时，能够快速进行故障处理和优化，使网络服务逐渐回到预期状态。

上述策略可以根据故障节点的配置迅速拉起一个新的节点替代故障节点工作，也可以利用 AI 大数据技术，通过自动化智能化运维的方式快速解决故障。

3.4 网元转发设备层

网元转发设备层主要指物理网元和虚拟网元设备。物理网元设备是指基于可编程交换芯片的白盒设备^[11]，优点是能够保证超高的吞吐量和网络转发速度，缺点是不够灵活、过度依赖硬件厂商；虚拟网元设备是指在 ARM 或 x86 服务器平台上用虚拟机或容器的形态部署的虚拟网络功能^[12]。通过虚拟化的方式把网元软件功能与具体部署所需要的物理硬件平台解耦，可以实现快速部署、弹性扩展、业务单元的可编程迭代以及多个功能单元的灵活编排等特性。

虽然虚拟网元网络拥有上述诸多的优势，但是相比传统的硬件网络设备在单个设备转发性能、专业硬件级可靠性能力以及设备长时间稳定

性等方面仍然存在较大的差距,这也是长期阻碍虚拟网元实现大规模商用部署的关键因素。针对虚拟网元设备在性能、可靠性、稳定性方面的劣势,通过本文提出的可编程网络平台,利用统一的网元管控面高并发分布式数据库、虚拟网元集群化部署以及按需预配置的不同业务单元的多副本机制不仅可以弥补一般虚拟网元的劣势,还可以实现虚拟网元的高性能、高可靠以及高稳定的电信级指标要求。

网元转发设备层的整体架构如图 5 所示,由虚拟网元集群和分布式数据库组成。虚拟网元集群根据虚拟网元和代理的承载方式可以分为虚拟机集群和容器集群,其中,虚拟网元负责数据包的转发,代理负责编程数据平面网络功能;分布式数据库用于实现配置的持久化存储。当控制器下发配置时不直接调用虚拟网元的北向 API,而是写入分布式数据库中,代理通过实时监听分布式数据库发现配置的改动,将最新配置应用到虚拟网元,实现配置的下发。

上述架构通过引入分布式数据库,进一步解耦控制面和数据的数据模型语言约束(如 YANG 模型),简化了控制器管理虚拟网元的方式,提升

了管控指令的执行效率和运维效率。当网元因故障、负载需要迁移、扩容时,可以根据数据库中持久化存储的配置内容快速拉起虚拟网元实例,提高转发层的可用性和稳定性;同时,利用虚拟网元集群化部署带来的弹性伸缩、灵活部署等特点,结合分布式数据库高并发特性,通过多实例的方式提高转发层的整体转发性能,通过多副本的方式提高网元的可靠性。

4 结束语

本文从当前业界提出的 6G 网络目标场景入手,根据上各种场景中对 6G 网络多维度的动态实现需求出发,提出了一种面向网络状态管理具备闭环自优化能力的可编程网络平台架构整体设计方案,基于此平台研发的网络设备更容易满足未来对 6G 网络设备多维度网络指标的要求。经过初步逻辑推理论证,该平台除了有效地整合 SDN、NFV 以及网络智能化运维的能力之外,基于云原生声明式状态管理的理念,使用面向网络状态的管理方式驱动网络的闭环自优化处理,使能未来 6G 网络的编排与实现具备复杂系统内多维度协调、自动化调整以及自我进化的能力。

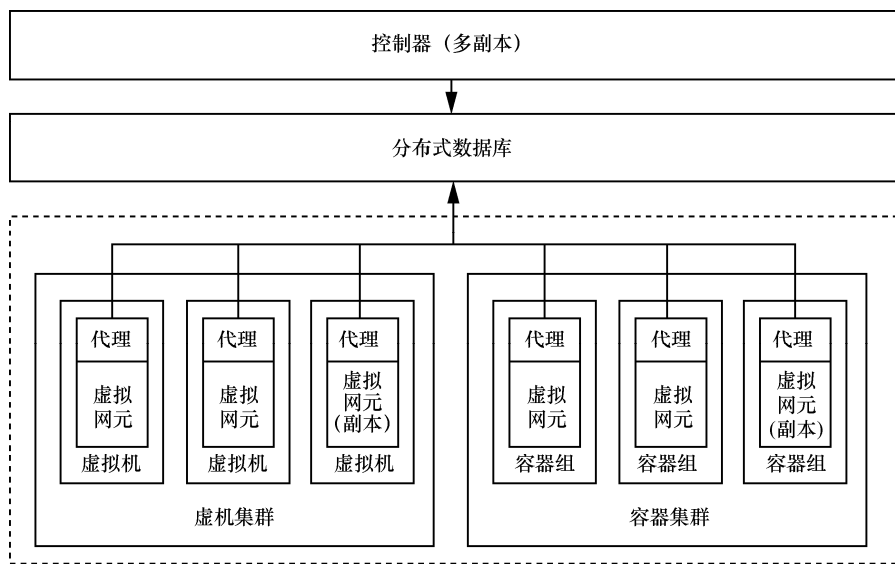


图 5 网元转发设备层的整体架构



未来笔者会根据这套可编程网络平台的设计理念, 深入探索并实现相应的原型系统。结合运营商在边缘计算、物联网以及家庭云方面的场景化需求, 验证并补足可编程网络的设计与方法。在可预计的未来希望这套可编程网络平台可以为随愿网络以及数字孪生网络提供网络可编程服务。

参考文献:

- [1] IMT-2030 推进组. 6G 总体愿景与潜在关键技术白皮书[R]. 2021.
IMT-2020 Promotion Group. White paper on 6G overall vision and potential key technology[R]. 2021.
- [2] 中国移动研究院. 6G 网络架构愿景与关键技术展望白皮书[R]. 2021.
China Mobile Research Institute. White paper on 6G network architecture vision and key technology outlook[R]. 2021.
- [3] 紫金山实验室. 未来网络白皮书[R]. 2021.
Purple Mountain Laboratories. White paper on future network[R]. 2021.
- [4] 孙滔, 周铨, 段晓东, 等. 数字孪生网络(DTN): 概念、架构及关键技术[J]. 自动化学报, 2021, 47(3): 569-582.
SUN T, ZHOU C, DUAN X D, et al. Digital twin network (DTN): concepts, architecture, and key technologies[J]. ACTA Automatic Sinica, 2021, 47(3): 569-582.
- [5] 王旭亮, 刘增义, 胡雅婕, 等. 基于 NFV MANO 的边缘计算多种智能化部署方案研究[J]. 电子技术应用, 2019, 45(10): 19-24, 28.
WANG X L, LIU Z Y, HU Y J, et al. Research of various intelligent multi-access edge computing deployment solutions based on NFV MANO Chinese Full Text[J]. Application of Electronic Technique, 2019, 45(10): 19-24, 28.
- [6] 佚名. AX 系列: 可编程以太网交换芯片[J]. 世界电子元器件, 2010.
Anonymous. AX series: programmable ethernet switching chip[J]. Global Electronics China, 2010.
- [7] PAOLUCCI F, SGAMBELLURI A, CUGINI F, et al. Network telemetry streaming services in SDN-based disaggregated optical networks[J]. Journal of Lightwave Technology, 2018, 36(15): 3142-3149.
- [8] YUAN Y F, ALUR R, LOO B T. NetEgg: programming network policies by examples[C]//Proceedings of the 13th ACM Workshop on Hot Topics in Networks. New York: ACM Press 2014.
- [9] MONACO M, MICHEL O, KELLER E. Applying operating system principles to SDN controller design[EB]. 2015.
- [10] ZEYDAN E, TURK Y. Recent advances in intent-based networking: a survey[C]//Proceedings of 2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring). Piscataway: IEEE Press, 2020: 1-5.
- [11] 黄志兰, 樊勇兵, 陈楠, 等. overlay SDN 实现异构兼容的关键技术[J]. 电信科学, 2016, 32(11): 112-118.
HUANG Z L, FAN Y B, CHEN N, et al. Key technologies of heterogeneous compatibility in overlay SDN Chinese Full Text[J]. Telecommunications Science, 2016, 32(11): 112-118.
- [12] 魏亮, 李蓉, 吴正东, 等. 基于 SDN/NFV 的未来网络实验平台[J]. 电信科学, 2017, 33(4): 39-47.
WEI L, LI R, WU Z D, et al. Future network experiment platform based on SDN/NFV Chinese Full Text[J]. Telecommunications Science, 2017, 33(4): 39-47.

[作者简介]



王旭亮 (1986-), 男, 中国电信股份有限公司研究院高级工程师, 主要研究方向为云网融合、云数据中心网络和边缘计算等。



全硕 (1991-), 男, 中国电信股份有限公司研究院工程师, 主要研究方向为云网融合、云计算与大数据和云网运营等。



刘增义 (1992-), 男, 中国电信股份有限公司研究院工程师, 主要研究方向为边缘计算、云网融合。



章军 (1973-), 男, 中国电信股份有限公司研究院高级工程师, 主要研究方向为云网融合技术、5G 运营管理、5G 无线控制器技术、边缘计算。