服务定制网络体系架构的设计与思考

黄韬[1,2]，张晨[2]，肖玉明[2]，余水[3]，刘韵洁[1,2]
（1. 北京邮电大学网络与交换国家重点实验室，北京 100876；
2. 网络通信与安全紫金山实验室，江苏 南京 211111；3. 悉尼科技大学计算机学院，悉尼 NSW2007）

摘 要：系统性阐述了服务定制网络（SCN）新型网络架构，可为互联网应用提供一种全新的网络底层能力与使用方式。在 TCP/IP 网络架构中，网络为应用提供“尽力而为”的服务质量，而 SCN 架构转换了应用与网络两者之间的主客体关系，允许应用“按需定制”网络的服务质量。以应用视角出发，挖掘了“可声明”“细粒度”“端到端”三大能力内涵，由此推演了 SCN 的总体设计思路，并给出了一种具象的 SCN 体系架构和一种可行的 SCN 系统实现。SCN 未来可应用于远程工控、增强现实等人机物全场景，为网络即服务（NaaS）的实现提供一种新颖、实用、理想的手段。

关键词：未来网络体系架构；服务定制网络；网络即服务；确定性网络
中图分类号：TP393
文献标志码：A
DOI: 10.11959/j.issn.1000−436x.2024017

Design and research of service customized networking architecture

HUANG Tao[1,2], ZHANG Chen[2], XIAO Yuming[2], YU Shui[3], LIU Yunjie[1,2]
1. State Key Laboratory of Networking and Switching Technology, Beijing University of Posts and Telecommunications, Beijing 100876, China
2. Purple Mountain Laboratories, Nanjing 211111, China
3. School of Computer Science, University of Technology Sydney, Sydney NSW2007, Australia

Abstract: A new network architecture of SCN (service customized networking) was systematically expounded, which provided a novel network underlying capability and usage method for Internet applications. TCP/IP network architecture served the application with “best effort” quality of service, while SCN converted the subject-object relationship between the application and network that allowed the application to “on-demand customize” the QoS of networks. From the perspective of applications, three major connotations “declarable”, “fine-grained” and “end-to-end” were excavated to deduce the overall design of SCN, where a concrete SCN architecture and a feasible SCN system realization were present. In future, SCN can be applied in scenarios of man-machine-things, such as remote industrial control, and augment reality, to provide a novel and ideal means for NaaS (network as a service).

Keywords: future network architecture, service customized networking, NaaS, deterministic network

0 引言

1974 年，Cerf 与 Kahn[1]开创性地提出了一种跨异构分组交换网络的统一通信方法，其思想后续进一步发展为网际互连协议（IP, Internet protocol）[2]和传输控制协议（TCP, transmission control protocol）[3]的标准化基础，并在不断的实践与修订中形成了一整套的 TCP/IP 协议族[4]。如今互联网已渗透人类生活工作的各个方面，现有各式各样的互联网应用，无一例外都构建在 TCP/IP 网络架构之上。本文将系统性阐述一种名为服务定制网络（SCN, service customized networking）的新型网络架构，并以此力求为互联网应用提供一种全新的网络底层能力与使用方式。在 TCP/IP 网络架构中，网络为应用提供“尽力而为”的服务质量，即网络会尽可能地将应用间通信数据传送至目的端，但对其传送速率、

收稿日期：2023−10−19；**修回日期：**2023−12−13
基金项目：国家自然科学基金资助项目（No.62171046, No.92267301）
Foundation Items: The National Natural Science Foundation of China (No.62171046, No.92267301)

时延不做任何形式的保障承诺[5]。虽然“尽力而为”的设计在极大程度上简化了网络，但应用却只能被迫改变自身行为去适应网络，或做额外努力去弥补网络不足，如流视频应用会根据当前传送速率来改变清晰度[6]，同时会通过应用侧缓存来平滑时延波动。

实际上，自实时语音类应用出现[7]，业界就开始持续地关注如何改善 TCP/IP 网络的服务质量（QoS, quality of service）[8-16]。不过从本质上而言，上述服务质量增强的网络技术均是对“尽力而为”网络架构的局部改进，其中大部分能力只作用于运营商网络内部，并未被终端侧、应用侧所接受和使用。与之相对的是，在 SCN 架构中，期望能够实现应用“按需定制”网络服务质量，即应用可对网络提出带宽、时延、抖动、丢包等需求，网络需在能力允许条件下提供相应水平的服务质量保障。一种情况是网络成功响应需求，并分配好传送路径与资源，等待应用发送数据；另一种情况是网络无法响应需求，则需与应用重新协商需求。

从“尽力而为”到“按需定制”，意味着应用与网络两者之间的主客体关系发生转换，如图 1 所示。“尽力而为”体现的是“网络为中心”的思想，网络以“中立性”视角看待所有应用，而应用只能被动地接受网络向其提供的服务质量；“按需定制”体现了“网络即服务（NaaS, network as a service）”的思想，应用可主动要求网络向其提供所需服务质量，而网络则需以“差异化”的方式来对待不同应用。

上述转换关系将深刻地影响未来网络的技术与业务发展。传统消费型互联网以人–机或人–人场景为主，由于消费者对服务质量的低敏感与高容忍，“尽力而为”引发的问题并不明显。未来随着“人机物”场景的多元化，生产型互联网[17-18]对“按需定制”的需求愈发明确，均无法接受“尽力而为”的服务质量，仅靠应用自身的优化效果杯水车薪。同时，随着新型人–机、人–人场景的涌现[19-20]，为满足视、听、触、嗅、味等感官间的差异性，也亟须一种可“按需定制”的网络，改变“特定场景特定优化”的无奈现状。

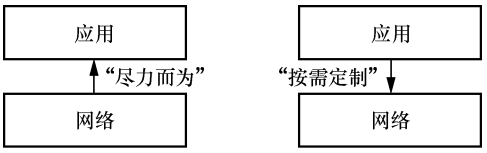

图 1 应用与网络关系转换

服务定制网络“按需定制”思想的已提出近 10 年[21]，期间得益于云计算、软件定义网络、确定性网络、算力网络等技术的成熟与发展，目前 SCN 已形成较系统的总体框架与技术突破。本文将力争在有限篇幅中对 SCN 体系架构进行系统性阐述：分析“可声明”“细粒度”“端到端”等按需定制的内涵；从“目标推演”“客观约束”两方面介绍 SCN 总体设计；从“组网模型”“数据包头”“协议交互”三方面介绍一种具象的 SCN 体系架构；从“数据面”“控制面”“业务面”三方面介绍一种可行的 SCN 实现；以远程控制、算力网络、增强现实为例介绍 SCN 使用方式；展望后续研究方向。

1 “按需定制”的能力内涵

本节首先界定应用和网络的形态与边界。如图 2 所示，应用是一段计算机程序，它在被实例化之后运行自身的业务处理逻辑，当其需要与本主机中或其他主机上的应用进行通信时，则调用网络能力来

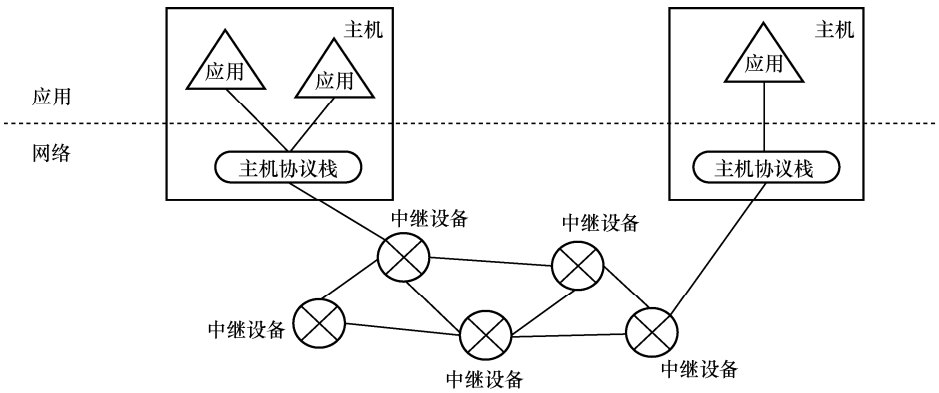

图 2 应用与网络的形态与边界

传递数据；网络是为实现应用间访问的一系列软硬件实体，包括主机中专用的协议栈程序（负责将应用所要传递的数据封装为数据包并进行发送或接收）与网卡，以及中继设备（用于交换数据包至目的地）与链路。图 2 中虚线代表应用和网络间的边界，同时也是应用调用网络能力的接口。

TCP/IP 协议栈使用 Socket[22]作为应用和网络间的接口，源应用通过指定目的应用所在节点的 IP 地址以及目的应用使用的端口号，即可通过网络将数据包传送过去。中继设备根据数据包中目的 IP 的可达性进行转发而不区分任何应用，这导致传送速率、时延等指标的随机波动，即“尽力而为”，主机侧协议栈为应对这种情况通过 TCP 不断观测上述指标的波动，并通过拥塞控制算法适应性地调整数据包的发送行为以获得更优效果。

为优化特定应用在中继设备间的传送效果，有一种通用范式得到了广泛使用：在入口的中继设备处对应用特征进行识别，如五元组[23]、深度包检测（DPI, deep packet inspection）[24]等；然后将应用相关的数据包导入隧道路径，如多协议标签交换流量工程（MPLS-TE, multi-protocol label switch traffic engineering）[10]、分段路由（SR, segment routing）[16]等，以获得更优的传送速率。该范式可概括为网络感知应用[25]或者“可感知应用的网络”。该范式也可以进一步拓展其边界：当主机与中继设备属于同一管理主体时，网络感知应用的位置由入口中继设备上移至主机协议栈，由主机协议栈将应用相关数据包直接导入隧道路径，如基于 IPv6 的分段路由（SRv6, segment routing IPv6）[26]、SR over 用户数据报协议（UDP, user datagram protocol）[27]。此外，也有工作尝试探讨拥塞与路径的联合控制问题[28]。

无论边界起始于中继设备或主机协议栈，上述范式均体现了“网络为中心”的思想，因此多被运营商用于底层承载网的内部优化。若从应用视角来看，其在技术层面至少存在如下问题。SCN 能力需求与现有技术不足分析如表 1 所示。

1) 不好用：应用无法量化地声明对网络服务质量的需求

网络虽可提供路径开通的应用程序接口（API, application program interface），但这种带外表达需求的方式需知晓入口和出口的中继设备，应用难以提供相关信息。NetworkAPI[29]通过 SRv6 段标识（SID, segment identifier）表达应用信息与 QoS 需求，使用任播解决第一跳的选路问题，但只适用于单域场景；资源预留协议（RSVP, resource reservation protocol）[8]允许应用通过信令量化地表达带宽需求，但只能在不影响选路的条件下预约带宽，且现网中并未实际使用；差分服务代码点（DSCP, differentiated services code point）[30]允许应用标记数据包定性地表达分类与优先级，但无法具体指定带宽或时延等需求，且相关需求不被网络所信任。

表 1　　SCN 能力需求与现有技术不足分析

能力需求	现有相关技术	不足
表达服务质量需求（可声明）	API	应用无法提供入口与出口中继设备的地址信息
	NetworkAPI	只适用于单域场景
	RSVP	只能在不影响选路的前提下预约带宽，且无法定制时延等指标
	DSCP	无法量化表达带宽、时延等需求
	TAPS	无法表达带宽、时延需求
业务流级需求声明（细粒度）	Flow Label	被用于中继设备上的随机负载均衡，与 QoS 相悖
	Stream ID	通常以加密形式存在，无法被网络感知
	APN6	一定程度解决“可声明”与“细粒度”问题，但其设计初衷是避免协议交互，因此在流量准入方面存在缺陷
	CCN	业务状态在全网扩散使网络扩展性受限，并且未系统性解决 QoS 问题
跨域服务质量保障（端到端）	ALTO	作为一种辅助技术向应用提供端到端网络地图，但并不关注网络服务质量保障
	QUIC	其设计仍以底层 QoS 随机波动为前提，同时未考虑其可靠与顺序处理在端到端中所引入的速率限制与额外时延
	QUIC-SR	一定程度解决“细粒度”与“端到端”问题，但其要求应用掌握网络全部控制权，因此只能采用叠加式组网，导致无法获得底层网络的 QoS 保障
	确定性网络技术，如 IEEE802.1Qch、DIP 等	当前确定性技术只能作用于有限域（局域或广域），无法实现端到端跨域的时延、抖动保证

2) 不敢用：应用无法指定业务流级别的细粒度网络服务质量需求

应用可为不同业务流分配不同五元组，并将其告知网络以映射隧道路径，不过这面临着极大的安全风险。Flow Label[31]提供了在网络层细粒度区分业务流的潜在可能，不过却被实现用于中继设备上的随机负载均衡，与服务质量相悖；Stream ID[32]可在应用层区分业务流，但通常以加密形式存在因而无法被网络感知。内容中心网络（CCN, content centric networking）[33]采用一种面向内容的解决方案，其不关心内容所在位置，而是以内容名称进行路由转发，并通过路由器进行沿路缓存，但这将导致业务状态在全网扩散使网络扩展性受限，并且CCN未系统性解决服务质量问题。

3) 不可用：应用跨多管理主体分布时无法得到端到端的网络服务质量保障

网络跨域路由采用对等互联模型，根据运营商网间策略进行选路，缺乏应用级端到端服务质量视角。应用层流量优化（ALTO, application-layer traffic optimization）[34]通过汇集各管理主体的网络信息并向应用提供端到端的网络地图，可用于点对点/内容分发等叠加网络优化，但不关注如何保障网络服务质量。快速UDP互联网连接（QUIC, quick UDP Internet connection）[35]可实现端到端灵活的传输与拥塞控制，但其设计仍以底层服务质量的随机波动为前提，也并未考虑其可靠与顺序处理在端到端中所引入的速率限制与额外时延。确定性网络技术能够承诺带宽、时延等性能，如IEEE 802.1Qch[36]、确定性IP（DIP, deterministic IP）[37]等，但其仅作用于有限域（如局域或广域），无法提供面向互联网的端到端确定性保障能力。

面对上述问题，传输服务（TAPS, transport services）[38]抽象了一套通用的传输服务，应用不必指定传输协议而是表达服务质量需求（如有序），重点作用于主机协议栈，一定程度上解决了问题1)，但由于其设计目标需兼容现有实现，因此无法表达带宽与时延需求。应用感知型IPv6网络（APN6, application-aware IPv6 networking）[39]扩展了IPv6的数据封装，应用可在数据包中填充应用/流的标识以及服务质量需求（如带宽、时延等），重点作用于中继设备，一定程度上解决了问题1)和问题2)，但其设计初衷是避免协议交互，因此在流量准入方面存在缺陷。QUIC-SR结合了QUIC和SR思想，应用可端到端地指定隧道路径并与业务流进行映射，可作用于主机协议栈与中继设备，一定程度上解决了问题2)和问题3)，但由于其设计思想是应用拥有网络的全部控制权，因此只能采用叠加式组网而无法得到底层承载网的服务质量保障。

SCN架构“按需定制”的能力内涵即全面解决应用所面临的上述及其他可能存在的问题，提供“可声明”“细粒度”“端到端”的网络服务质量保障。

2 SCN的设计思路

2.1 目标推演

围绕“按需定制”的能力内涵，下面将继续以应用视角推演SCN的设计目标。

1) 可声明

应用对网络服务质量的需求主要包括带宽 B、时延 D、可靠度 L 以及顺序性 O 四类指标。带宽指网络传送速率，可细分为恒稳速率、速率下限和平均速率等；时延指网络的传送时延，可细分为恒稳时延、时延上限、平均时延等；可靠度指网络的丢包情况，可细分为不可丢包、丢包上限、逾期丢包等；顺序性指网络的交付顺序，可细分为严格保序、局部有序和紧急插序等。应用可定量地对上述指标进行声明，同时须指定目的应用的标识Dst_ID，因此需求可表示为

$$(\text{Dst\_ID}, B, D, L, O)$$

明确上述信息后，应用可在带内向网络发出如下请求消息以进行声明

$$\text{RQ}(\text{Dst\_ID}, \langle B, D\rangle)$$

$$\text{RQ}(\text{Dst\_ID}, \langle L, O\rangle)$$

此处将带宽、时延和可靠性、顺序性分开，是考虑到两类需求所作用的网络主体有所不同：带宽和时延主要作用于中继设备，而可靠性和顺序性则主要作用于主机协议栈。

发出上述请求消息后，应用需要网络进行消息回复以得知请求是否成功，即

$$\text{RP}(\text{Dst\_ID}, \langle B, D\rangle, \text{Success/Fail})$$

$$\text{RP}(\text{Dst\_ID}, \langle L, O\rangle, \text{Success/Fail})$$

2) 细粒度

应用间的通信会产生不同类型的消息或业务流，它们有着截然不同的网络服务质量需求。一种自然的方式是为目的应用程序中各入口函数或程序片段都分配不同的标识，这时需求可进一步表示为

$$N\ (\mathrm{Dst\_ID}_n, B, D, L, O)$$

但在上述方式中，应用需要暴露出内部业务信息，难以被应用服务商所接受。因此可将需求的表示方式转换为

$$(\mathrm{Dst\_ID}, [B, D, L, O]_n)$$

在上述方式中，应用程序只使用一个标识但可以映射一组需求，从而避免内部信息暴露。为便于不同类型消息或业务流的使用，每一个需求在经过应用请求和网络回复后，都需要对应于一个使用入口，而如何定义相关使用入口的语义，将直接决定应用可获得何种能力。

带宽和时延这类需求主要作用于中继设备，在根本上与传送路径及沿路中继设备的资源分配相关。最简单的方式是网络将满足带宽和时延需求的路径信息回复给应用，并作为应用获得带宽与时延的使用入口。但这种方式存在 2 个问题：一是将网络拓扑、路径等信息暴露给应用存在着极大的风险；二是将服务质量绑定于特定传送路径不够灵活。

为解决上述问题，本文在 SCN 设计上引入了一个核心语义——“票（Ticket）”。“票”概念的引入可实现网络和应用之间的能力映射。对于应用而言，“票”代表由带宽与时延任意组合的网络能力；对于网络而言，“票”意味着发放具有某种带宽与时延水平的服务能力。在“票”语义的抽象下，应用既不必关心也无法获知网络路径的选择或调整，“票”的实现只作用于中继设备。因此，带宽和时延的请求和回复消息改为

$$\mathrm{RQ(Dst\_ID}, <B, D>)$$

$$\mathrm{RP(Dst\_ID}, <B, D>, \mathrm{Ticket\_ID})$$

可靠度和顺序性需求主要作用于主机协议栈。虽然丢包和乱序多源于沿路中继设备的资源不足与多路径传送，但若由中继设备处理丢包和乱序则会占用其大量资源进而影响带宽和时延保障能力，另外目的应用所在主机协议栈的资源不足也会导致部分丢包和乱序，因此由主机协议栈来综合处理中继设备及自身导致的丢包和乱序是最合适的。通常的方式是以五元组表示源和目的地址、端口以及协议类型，并作为应用获得可靠度与顺序性的使用入口。不过这种方式主要存在 2 个问题：一是将应用业务流的地址和端口暴露给网络存在着极大的风险；二是将服务质量绑定于特定协议类型不够灵活。

为解决上述问题，本文在 SCN 设计上引入另外一个核心语义——“关联（Association）”。对于应用而言，“关联”代表可靠度与顺序性指标灵活组合的网络；对于网络而言，“关联”意味着有某种可靠度与顺序性需求的应用。在“关联”语义的抽象下，网络如何处理丢包或平衡带宽/时延，应用既不必关心更不需要自己实现，“关联”的实现只作用在源和目的主机协议栈。因此，可靠性和顺序性的请求和回复消息改为

$$\mathrm{RQ(Dst\_ID}, <L, O>)$$

$$\mathrm{RP(Dst\_ID}, <L, O>, \mathrm{Association\_ID})$$

在上述基础上，应用发送数据可通过 Ticket_ID 和 Association_ID 为使用入口以获得所需网络服务质量，表示为

$$\mathrm{Send(Data, Ticket\_ID, Association\_ID)}$$

3) 端到端

保障从源应用到目的应用的端到端网络服务质量要经历复杂的处理过程。通过“票”和“关联”的抽象，使应用不需要关心网络内部实现，并端到端地获得带宽、时延和可靠度、顺序性等指标的物理组合。但“票”和“关联”的实现会在彼此间产生一定影响，如“票”在中继设备上的实现会不可避免地引入一些丢包和乱序，“关联”在主机协议栈上的实现也会引入一些速率限制和额外时延，因此物理组合的实际结果会与应用端到端需求有所偏差。

如果应用希望在严格意义上满足端到端需求，需要在请求消息中将带宽、时延和可靠度、顺序性合并，即

$$\mathrm{RQ(Dst\_ID}, <B, D, L, O>)$$

相应地，为在严格意义上满足应用的端到端需求，网络在回复消息中需引入新的使用入口“流”，表示为

$$\mathrm{RP(Dst\_ID}, <B, D, L, O>, \mathrm{Flow\_ID})$$

上述请求消息的处理，要求网络能够将相关指标在中继设备和主机协议栈间进行合理拆分，并分别向中继设备和主机协议栈发送带宽、时延和可靠度、顺序性的分类请求，在回复消息中分别将 Ticket_ID 和 Association_ID 与 Flow_ID 进行映射，Flow_ID 只在源应用本地有意义。应用发送数据即可将 Flow_ID 作为使用入口以获得严格意义上的端到端网络服务质量，表示为

$$\mathrm{Send(Data, Flow\_ID)}$$

本节以纯粹的应用视角推演了 SCN 的设计目

标。然而网络中的一些客观约束，将不可避免地对其相关设计目标及实现产生一定限制。

2.2 客观约束

在分析网络中的客观约束之前，本文先尝试抽象网络的本质。在物理上，网络就是一组连续的时空资源集合，用于传递应用间通信产生的信息；在逻辑上，网络是一个动态的分布式数据库，用于处理不同应用间通信的时空资源分配关系。这种分配关系在直观上体现为网络中各种“表项”，在设计时有三大原则需要明确：1) 何时形成表项？2) 形成何样的表项？3) 不同表项以何种结构进行分布？这些原则分别体现了分组交换的3个核心基础概念：“连接”“命名”与“分层”，这些核心基础概念也将对SCN的设计产生直接影响。

1) 连接

“可声明”的实现与网络中的“连接”密切相关。“有连接vs无连接”决定了表项的形成时机，是网络体系架构中最富历史性也最具争议性的话题。“有连接”由应用发送请求消息触发网络形成表项，在形成过程中应用需要阻塞数据发送。“无连接”中应用可立即发送数据而不必阻塞，表项的形成时机早于应用间通信的开始。

实际上，前文SCN“可声明”的实现在直观上采用了有连接的形式，应用发送请求消息以触发网络形成表项并响应回复消息，其优势在于表项形成后应用可获得针对性的服务质量保障，而应用所面临的客观约束是网络形成表项可能需要较长时间，其等待对应用而言可能不够理想甚至不可接受。假设网络完成同一次应用数据传送的时间，有连接为T_0而无连接为T_0'，通常$T_0<T_0'$反映了有连接在形成表项后的服务质量提升。进一步结合有连接的等待时间T，则应用在该次数据传送中得到的总体服务质量有以下2种情况：①$T_0+T<T_0'$，则仍有提升；②$T_0+T>T_0'$，则下降。

为解决上述问题，SCN在设计上使用了一个关键机制——“两阶段（TwoPhase）”，应用仍以有连接方式向网络发送请求消息，网络收到后立即发送回复消息并携带Flow_ID，应用收到后即可发送数据，此时T尚未结束仍处于第一阶段，网络将以无连接方式进行过渡处理，直至T结束进入第二阶段，网络将自动切换为有连接方式并内部更新Flow_ID和Ticket_ID/ Association_ID间的映射。“两阶段”机制结合了有连接和无连接的优势，既能有效满足情况①的需求又可部分缓解情况②的问题，“两阶段”可由应用灵活使能，若无法接受无连接的过渡处理可选禁用。因此，应用请求消息中需增加标志位，网络回复消息格式不变，表示为

RQ(Dst_ID, <*B*, *D*, *L*, *O*>, TwoPhase)

RP(Dst_ID, <*B*, *D*, *L*, *O*>, Flow_ID)

另外，“可声明”在网络中的实现还需应用说明自身的流量特征区间Feature，区间内所发数据将得到服务质量保障，超出区间的部分则不承诺质量，表示为

RQ(Dst_ID, Feature,<*B*, *D*, *L*, *O*>,TwoPhase)

RP(Dst_ID, Feature , <*B*, *D*, *L*, *O*>, Flow_ID)

2) 命名

“细粒度”的实现与网络中的“命名”密切相关。命名机制决定了表项的结构及表项间的联系，是网络体系架构的核心并本质上区分了不同的网络体系架构。命名机制使用标识来表示网络运行的必要元素，通过映射在元素间进行关联。应用发送数据时需要将标识封装进数据包，网络根据标识进行映射以实现其基本功能。

前文SCN为实现“细粒度”引入了Ticket_ID和Association_ID这2种标识，以满足应用的服务质量需求。若应用只需尽力而为，原理上不必发送请求消息而使用目的应用标识Dst_ID发送数据，但该方式将对网络形成巨大挑战。假设需要网络通信的应用及对应Dst_ID数量为A，主机与中继设备数量分别为H与R，则主机协议栈平均只需维护$\frac{A}{H}$条表项，而各中继设备要维护A条表项。庞大的应用体量与有限的中继设备资源，是网络面临的客观约束。

为解决上述问题，SCN在设计上引入了一个底层概念——“定位符（Locator）”，网络首先定位目的应用a所在主机h或所接入中继设备，并使用相关的定位符转发到h或r，再通过Dst_ID转发到a，即可将中继设备需要维护的表项开销降为H或R。“定位符”属于网络内部信息，源应用无法原生得知与目的应用相关的定位符，因而仍需额外向网络进行一次请求，为此先设想一种新的消息类型与使用入口如下

New_RQ(Dst_ID)

New_RP(Dst_ID, Locator)

New_Send(Data, Locator)

此设想中，应用需显式地使用不同方式以区分有无服务质量需求，实际上应用并不关心Locator，

而网络将 Locator 直接回复给应用也会暴露内部信息。因此 SCN 仍采用原有的消息类型与使用入口，将尽力而为视为一种任意的服务质量需求，将“定位符”视为一种“站票”。当应用在请求消息中将服务质量需求及相关参数置为任意及不可知时，网络将解析 Dst_ID 对应的 Locator 并在语义上变换为 Ticket_ID，映射 Flow_ID 并立即回复消息，应用发送数据仍可以 Flow_ID 为使用入口，表示为

RQ（Dst_ID, NA, <Any>, NA）

RP（Dst_ID, Flow_ID）

Send（Data，Flow_ID）

收到应用所发数据后，网络将 Flow_ID 反向映射为 Association_ID 及 Ticket_ID，在中继设备上将 Ticket_ID 反向映射为 Locator 并尽力而为地转发。

此处需重点说明，无服务质量的 Ticket_ID 面向目的应用，而有服务质量的 Ticket_ID 面向业务流，一种普遍认识是若应用数量为 N，则业务流为 $N\times N$ 数量级，关注于如何降低前者而非后者的表项开销是否合理？事实上，表项反映了网络“状态”，业务流的数量虽多但相关表项属于“软状态”，在通信过程中产生且只存在于沿路中继设备上，而应用数量虽少但相关表项属于“硬状态”，与通信过程无关且长期存在于全部中继设备上，两类表项的时空属性区别较大无法直接比较开销大小。另外，网络可在资源不足时拒绝有服务质量的请求，而无服务质量的请求网络则难以拒绝，因此实际开销需具体分析。

3) 分层

“端到端”的实现与网络中的“分层”密切相关。网络分层决定了表项的分布结构，是网络体系架构的经络即负责将各层次能力合为一体。网络分层通过约定各层能力以划分职责边界，通过规范层间接口以实现各层能力贯通。应用通过网络编程接口使用网络能力，不必感知也不需要关注网络在各个分层中的具体行为，在网络编程接口之上，应用可进一步结合自身通信模式差异化地实现应用层能力。

前文 SCN“端到端”的实现将 Ticket_ID 和 Association_ID 联合抽象为 Flow_ID，主机协议栈和中继设备在应用视角中被合二为一，最大限度上为应用屏蔽了端到端的全部细节，同时意味着需要网络处理好端到端的全部问题。假设从源应用到目的应用除 2 个主机协议栈外，要经过归属于 P 个管理主体的 M 个中继设备，为保障 M 个中继设备处理后的带宽与时延，网络需协同 P 个管理主体进行需求指标拆分、传送能力开通与边界设备衔接。现实中管理主体间多以对等模式互联，因缺乏应用视角而难以实现跨主体服务质量保障。

为解决上述问题，SCN 在设计上采用一种新型结构——“中转联盟”。通过第三方的网关型中继设备在不同管理主体的承载型中继设备间进行中转，网关型中继设备间形成第三方中转联盟，代理应用进行跨多管理主体的端到端规划并提供“联程票”，“联程票”被网关型中继设备映射为各管理主体提供的“逐程票”，并由承载型中继设备进行处理。“中转联盟”定义了网关与承载两类中继设备，在 SCN 分层中分别属于互联层与传送层，传送层职责是以稳定的带宽和时延传送分组，互联层职责是对多段传送的服务质量进行拼接与监测。若应用无服务质量需求，则互联层将站票映射为定位符并执行中转，而传送层可尽力而为地传送分组。

中继设备所提供的稳定带宽与时延，以应用发送数据的速率不超过其请求消息中声明的流量特征区间为前提，一旦超出网关中继设备将向发送端主机协议栈发出拥塞预警并对超出部分不再提供保障。这在 SCN 分层中属于互联层能力，分布于网关中继设备和发送端主机协议栈。

以上述为基础，发送端主机协议栈一方面得以摆脱复杂的拥塞控制，另一方面其自身实现应具备 Flow_ID 级的流量调度能力以免成为带宽或时延瓶颈。接收端主机协议栈原则上也应具备此能力并摆脱复杂的流量控制，但考虑到接收端要处理不可预测的并发，因此流量控制仍需保留以用于对发送侧进行动态反馈。而可靠度和顺序性在主机协议栈上的实现，也会不同程度地增加时延或损耗带宽，若网络无法同时满足带宽、时延和可靠度、顺序性，则需反馈应用并由应用自行决定如何平衡需求。这在 SCN 分层中属于传输层能力，分布于发送端和接收端的主机协议栈。

3　SCN 体系架构设计

第 2 节介绍了 SCN 设计思路，其思想可概括为：应用通过“请求消息”量化声明需求，通过“票”和“关联”抽象带宽和时延、可靠度与顺序性水平，通过“流”获得端到端服务质量；网络使能“两阶段”机制降低应用等待时间，引入“定位符”概念统一应用使用入口，采用“中转联盟”结构跨越多管理主体。本节将尝试给出一种具象的 SCN 体系架构设计。

3.1　组网模型

SCN 组网模型如图 3 所示。主机协议栈位于主机中，运行传输层、互联层、传送层能力，代表应用进行数据包收发并提供可靠度和顺序性保障；子网由承载型中继设备组成，运行传送层能力，负责传送数据包并逐程保障带宽和时延；网关由网关型中继设备组成，运行互联层、传送层能力，实现子网间协同监测并保障端到端带宽和时延。以下将借鉴计算机网络分层思想，尝试定义 SCN 架构的传送层、互联层与传输层能力，并描述会话层与应用层常见模式，但对其实现方式不做限定。

1) 应用层。执行应用通信行为，使用规范语义语法构造数据。应用层功能可随应用业务逻辑灵活变化，模式上或以一次性 Request-Reply、偶发性 Notify、持续性 Streaming 为原语，或应用自定义。

2) 会话层。结合应用通信结构，使用流列表维护通信关系，实现多应用间或单应用内 *M*:*N* 通信。该层功能支持应用可选，典型模式如双向、点对点、并发、集合等，并作为库函数与应用集成。

3) 传输层。基于序号化数据包实现消息和流，保障应用间通信的可靠度和顺序性，并配合互联层保障带宽和时延。该层功能包括分段、复用、确认、重传、纠错、流控、排序等，并在多应用间共用。

4) 互联层。基于网关进行数据包统一中转，为应用间通信提供端到端带宽和时延，协同各子网并监测其服务质量。互联层功能包括接入控制、路由转发、拥塞预警、子网映射、质量监控等并规范最大传输单元。

5) 传送层。通过子网在网关间及主机−网关间传送数据包，实现逐程带宽和时延保障，不同子网归属不同管理主体并可获得相应的传送收益。传送层功能包括质量注册、路径管理、分包重组等。

SCN 体系架构实现必须包括传输层和互联层能力，各子网可采用不同传送层技术但均需通过“票”提供带宽和时延，不同应用可选地采用应用层或会话层能力，但均需通过 API 调用 SCN 能力。

3.2　数据包头

基于上述组网模型，SCN 数据包头如图 4 所示。

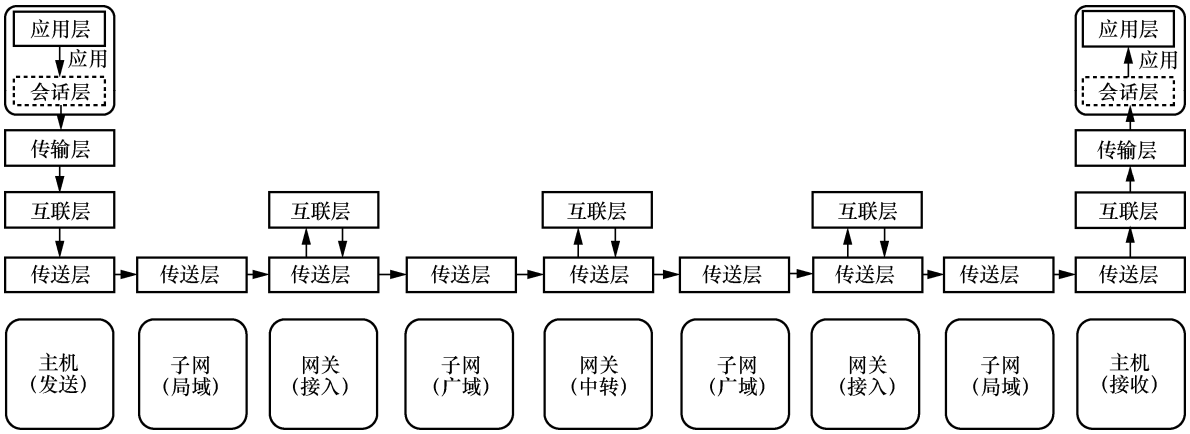

图 3　SCN 组网模型

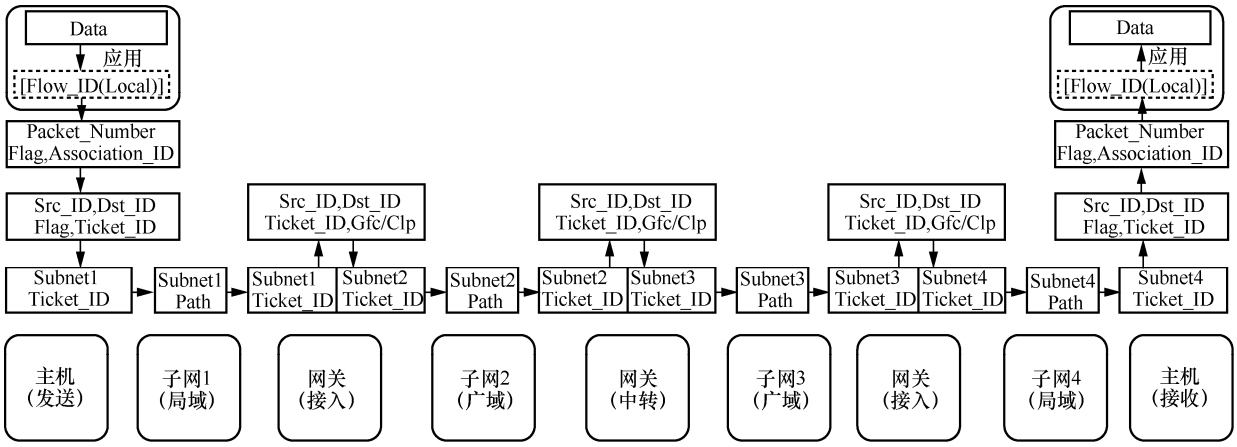

图 4　SCN 数据包头

应用通过应用层构造数据载荷，源主机协议栈逐层封装数据包头并从源主机发出，经由子网及网关传送至目的主机，目的主机协议栈反向解封数据包头，应用层完成信息解构后交付应用。下文将介绍传输层与互联层包头的必要字段及处理机制，并描述会话层处理机制，但对格式、长度、编码等不做限定。

1) 传输层包头。Association_ID 标识“关联”并实现应用内不同通信逻辑的复用与解复用，源应用和目的应用间可建立多对关联以获得不同的可靠度与顺序性。源和目的主机协议栈分别为每对关联生成一个 Association_ID 并提供给对方使用，并封装入数据包头中。Flag 用于带内区分协议消息与应用数据，并支持关联协商过程中的应用数据发送（即两阶段）。关联协商消息及协商过程中发送应用数据需使用保留 Association_ID，完成协商后协议栈自动切换为对方指定的 Association_ID。

Packet_Number 标记“关联”内数据包序号，是可靠度与顺序性保障的基础，具体可采用前向纠错、丢包重传、多发选收等实现机制或组合不同实现机制以满足应用需求。

2) 互联层包头。Ticket_ID 标识“联程票”并跨多个管理主体完成应用数据传送。源和目的主机协议栈需分别向“中转联盟”申请从自身到对方的单向 Ticket_ID，并封装入自身数据包头中。Flag 用于带内区分协议消息与应用数据。票申请消息使用保留 Ticket_ID，票激活前应用数据发送使用的 Ticket_ID 由目的应用所在网关的定位符 Gw_Locator 变换形成，票激活后主机协议栈切换为“中转联盟”更新的 Ticket_ID，网关收到数据包后根据 Ticket_ID 中转并映射相应子网的“逐程票”。

Src_ID 和 Dst_ID 标识源和目的应用，在端到端转发中保持不变。将其封装入数据包的作用为：源应用所在网关可通过 Src_ID 检验 Ticket_ID，避免应用恶意扫描 Ticket_ID 导致服务质量下降；目的应用所在网关通过 Dst_ID 映射目的应用所在主机的定位符，避免应用拒绝服务攻击主机定位符导致服务质量下降；目的主机协议栈通过 Dst_ID 找到应用；目的应用使用 Src_ID 反向发送数据。

3) 会话层结合应用通信结构维护 Flow_ID 列表，如在并发场景下为满足客户端应用的服务质量需求可能要与多个服务器应用实例间收发数据，集合场景为实现数据同步可能要在同一应用的多个实例间收发数据等。Flow_ID 本地有效不涉及数据包头。

3.3　协议交互

SCN 协议交互过程如图 5 所示。假设应用 A 与

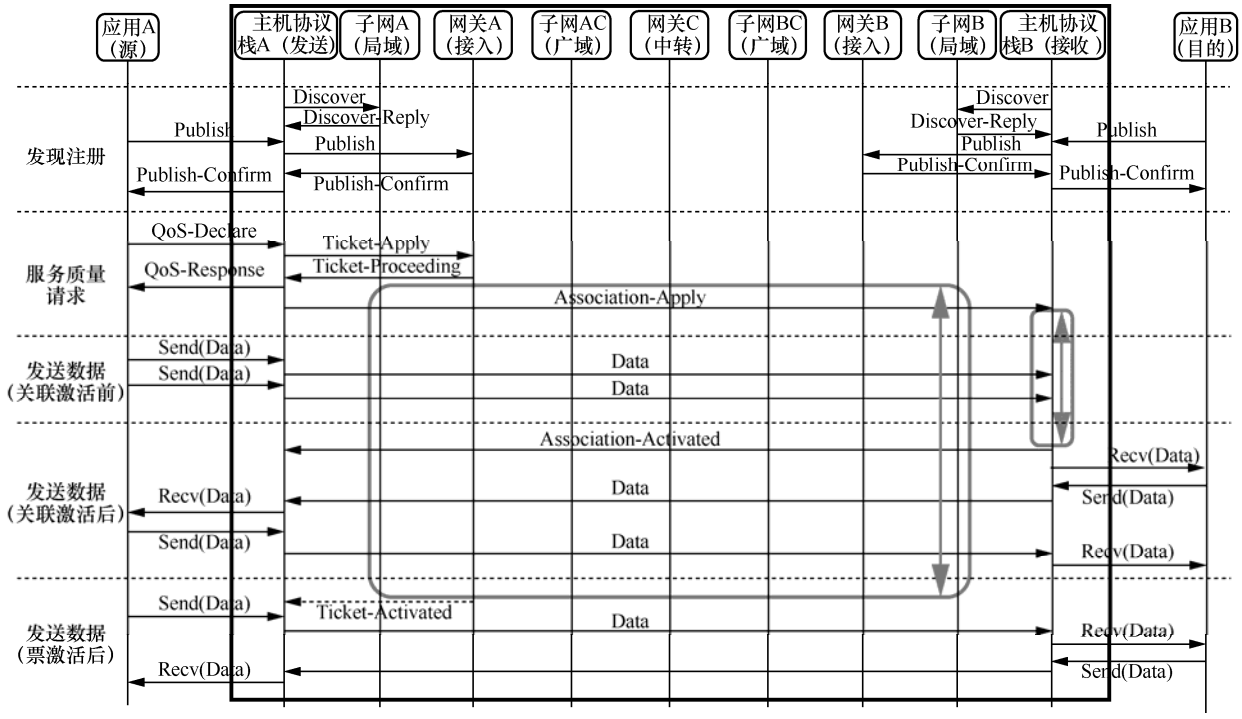

图 5　SCN 协议交互过程

应用 B 位于不同主机并接入不同子网与网关，A 通过 SCN 编程接口与 B 通信并期望得到服务质量保障，并由此触发了图 5 中一系列的协议交互。交互过程中所涉变量及其含义如表 2 所示。

表 2　　交互过程中所涉变量及其含义

变量	含义
Tkt_A / Tkt_B	主机在子网中的逐程票
Tkt_GwA / Tkt_GwB	主机通过子网到达锚定网关的逐程票
Ticket_ID	联程票标识
Subnet_Ticket_ID	子网内部票标识
Tkt_AB / Tkt_BA	为主机协议栈之间通信所分配的联程票
Src_ID	源端应用标识
Dst_ID	宿端应用标识
ID_A / ID_B	应用 A / B 的标识
Loc_GwA / Loc_GwB	网关 A / B 的定位符
Fl_AB	本地流标识
Association_ID	关联标识
Asc_Random	在所需关联激活前随机生成的关联
Asc_A / Asc_B	为特定可靠/顺序性指标生成的关联

1) 发现注册

主机 A 接入子网 A 后，其主机协议栈与子网 A 交互 Discover 消息获取主机在子网中的逐程票 Tkt_A 及通过子网到达网关 A 的逐程票 Tkt_GwA，Discover 消息沿用并扩展子网 A 的发现消息格式。B 同理。

应用 A 接入主机协议栈 A 后通过 Publish 消息注册 ID_A，主机协议栈随即向网关 A 发送 Publish 消息，其互联层包头置位协议消息 Flag、Src_ID 使用 ID_A、Dst_ID 和 Ticket_ID 使用协议消息保留值、Subnet_Ticket_ID 使用 Tkt_GwA，其消息载荷中携带 ID_A 与 Tkt_A。网关 A 收到消息后记录<ID_A, Tkt_A> 的映射，向主机协议栈 A 回复 Publish-Confirm 消息，并向其他网关同步<ID_A, Loc_GwA>的映射。

2) 服务质量请求

应用 A 通过 QoS-Claim 消息声明其与应用 B 间的通信服务质量需求，参数包括 ID_A、ID_B，流量特征区间，带宽、时延、可靠度、顺序性等指标，及建路、建链的两阶段选项等。指标分类拆解由应用自行完成或借助会话层完成。

首先，主机协议栈 A 向网关 A 发送 Ticket-Apply 消息，其互联层包头置位协议消息 Flag、Src_ID 使用 ID_A、Dst_ID 和 Ticket_ID 使用协议消息保留值，其消息载荷中携带 ID_A/ID_B 等参数信息。网关 A 收到消息后解析 ID_B 所在网关的定位符 Loc_GwB，将其变换为应用 A 与 B 间的联程票 Tkt_AB 后通过 Ticket-Proceeding 消息回复主机协议栈 A，并根据带宽/时延指标激活 Tkt_AB 资源。主机协议栈 A 收到回复消息后，本地生成流 Fl_AB 并记录<Fl_AB, ID_A/ID_B, Tkt_AB>的映射，向应用 A 返回 QoS-Response 消息并允许其通过 Fl_AB 向 B 发送数据。

随后，主机协议栈 A 使用 Tkt_AB 向主机协议栈 B 发送 Association-Apply 消息，其互联层包头置位应用数据 Flag、Src_ID 与 Dst_ID 使用 ID_A 与 ID_B、Ticket_ID 为 Tkt_AB。传输层包头置位协议消息 Flag、Association_ID 由主机协议栈 A 随机生成 Asc_Random，其消息载荷中携带自身所用 Asc_A、可靠度/顺序性、两阶段选项等信息。主机协议栈 A 更新映射 <Fl_AB, ID_A/ ID_B, Asc_A/Asc_Random, Tkt_AB>并发送至网关 A，该消息经由网关 C 和 B 传送到主机 B。主机协议栈 B 本地生成流 Fl_AB 并分配 Asc_B，记录<Fl_AB, ID_A/ID_B, Asc_A/Asc_B>的映射关系，并根据可靠度/顺序性指标激活 Asc_B 资源。

3) 发送数据

应用 A 在 Asc_B 激活前即可发送数据，此时因无法处理丢包和乱序，除非应用 A 可靠度/顺序性需求均为“任意”，否则主机协议栈 B 收到数据后暂不交付应用。但主机协议栈 B 完成 Asc_B 资源激活的时间极短，其持续时间约等于 Association-Apply 消息的传送时延，而 Tkt_AB 此时尚未激活，所以应用 A 的发送速率无法保障。

主机协议栈 B 完成 Asc_B 资源激活后向主机协议栈 A 回复 Association-Activated 消息，其互联层包头置位应用数据 Flag、Src_ID 使用 ID_B、Dst_ID 使用 ID_A、Ticket_ID 使用 Tkt_BA，其传输层包头 Association_ID 使用 Asc_A，消息载荷中携带 Asc_B 替换 Asc_Random。该消息经由网关 C 和 A 传送到主机 A，主机协议栈 A 更新映射为<Fl_AB, ID_A/ID_B, Asc_A/Asc_B, Tkt_AB>，后续发送应用 A 数据时将使用 Asc_B 以获得可靠度/顺序性保障，但此时 Tkt_AB 仍未激活，所以带宽/时延仍无法保障。

网关 A 通过中转联盟激活 Tkt_AB 资源，在完成后向主机协议栈 A 回复 Ticket-Activated 消息并携带新 Tkt_AB。主机协议栈 A 收到消息后更新映射

<Fl_AB, ID_A/ID_B, Asc_A/Asc_B, Tkt_AB>，后续发送应用 A 数据时使用 Tkt_AB 以获得带宽/时延保障。

上述过程以“票激活不等待、关联激活不等待”为两阶段选项，其他选项下应用 A 首次发送数据的时间会相应延后，应用 B 类似。值得注意的是，双向数据收发在 SCN 传输层和互联层的视角中相互独立且可对应不同服务质量，应用可在应用层自由定义双向的具体实现或使用会话层相关能力。受篇幅所限，本文对票和关联释放及应用注销过程不做展开。

4　SCN 实现

第 3 节介绍了一种 SCN 体系架构的设计思路，本节将介绍科研团队设计的一种 SCN 实现方案，并按如图 6 所示的转发、控制、业务 3 个平面展开。

4.1　转发平面

1) 半开环细颗粒监管

监管实现流量在中继设备入口的处理，“半开环”指介于开环与闭环间的反馈，根据流量特征执行监管并带内反馈主机，网关可对主机超发部分自行处理，如标记优先丢弃等，该处理多发生在接入网关的用户–网络接口（UNI, user-network interface）侧（DI3）。“细颗粒”指提供业务流级别的监管能力，网关代表应用对子网带宽和时延进行测量，该处理发生在网关与子网之间（DI4 与 DI6）。

实现上可采用漏桶强行限制传输速率，但可能会改变分组到达规律而引入额外时延；采用令牌桶可允许突发存在，在流量特征区间内不改变分组到达规律并可对超出部分进行标记[40-41]。

2) 无阻塞可编程交换

交换实现从中继设备入口到出口的流量处理，可看作中继设备的空间资源调度。“无阻塞”指业务流不因交换而产生可观测的带宽损耗或时延抖动，该处理多发生在设备芯片内部通道中，也可能发生在网关集群或网关矩阵的内部通道中；“可编程”指根据数据包头灵活定义交换芯片处理逻辑。

实现上可采用包交换或信元交换实现无阻塞，其中信元交换需保持数据包完整性；采用协议无关的包处理器编程方式（P4, programming protocol- independent packet processor）实现可编程，根据不同服务质量技术自定义数据包格式和转发机制[42]。

3) 确定性多目标排队

排队实现流量在中继设备内的处理，可看作中继设备接口上的时间资源调度。“确定性”指可承诺地保证带宽或时延，“多目标”指可同时承诺时延和带宽。传统加权公平排队（WFQ, weighted fair queuing）基于令牌动态分配带宽占比，可承诺带宽但无法承诺时延；传统低时延排队（LLQ, low latency queuing）以严格优先级保

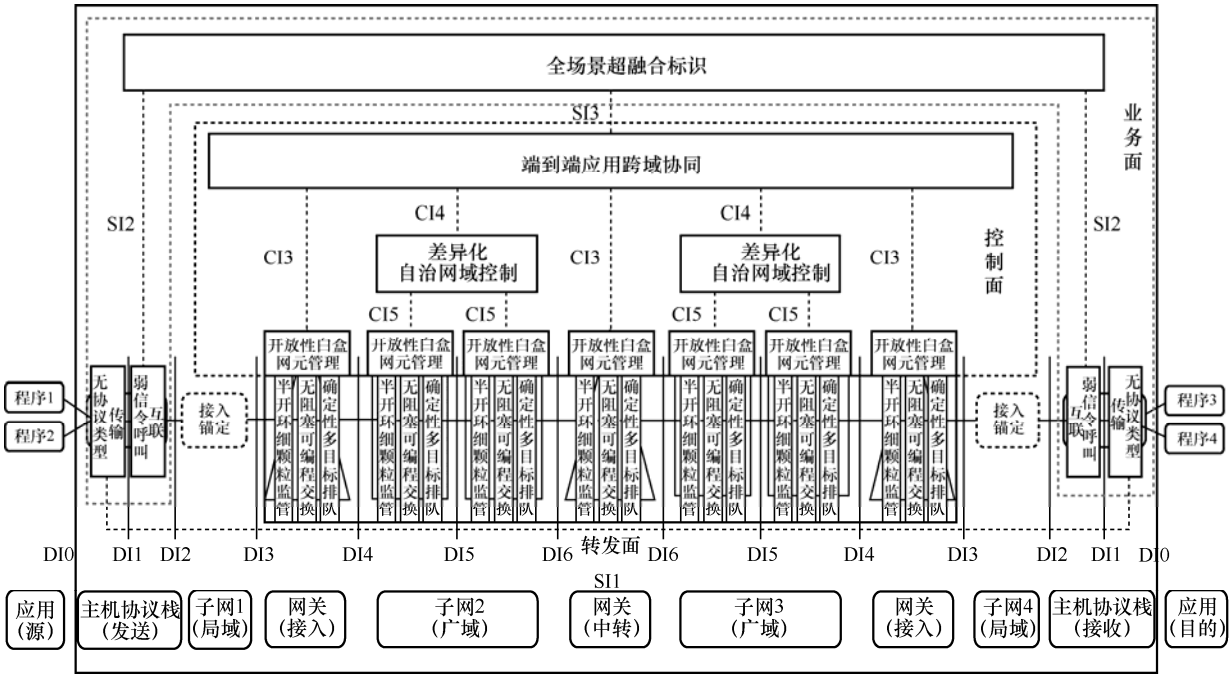

图 6　一种 SCN 的设计实现

障低时延，但受限于先入先出（FIFO, first in first out）机制无法同时承诺大带宽与低时延，也无法承诺时延抖动。

实现上可采用时间片调度或时隙调度[36, 43]，满足周期性业务的精准时间排队；可采用网络演算和预算门控动态规划路径与沿路资源，满足非周期性业务的截止时间排队。

4.2 控制平面

1) 开放性白盒网元管理

网元管理面向单个中继设备，位于子网中继设备或网关内部。“开放性”指中继设备提供管控 API，“白盒”指网元操作系统与交换芯片间实现标准化操作。基于本地的邻居维护、拓扑生成、基础路由等功能，中继设备通过管控 API 与远端交互上报状态与测量数据并实现路径与资源调优（CI3 或 CI5）。网元管理可部分代理远端处理逻辑以缓解其压力，但要求设备本地实现分布式缓存与优化机制。

实现上可采用远端逻辑严格优先的处理机制，本地逻辑作为缺省可在远端失联或表项失效时实现保护；也可重新定义路由信息库（RIB, routing information base）实现深度的控制优化。

2) 差异化自治网域控制

网域控制面向属于同一管理主体的一组中继设备，作用于子网内部。“差异化”指可提供多种带宽和时延水平，“自治”面向一个路由域或多个路由域形成的管理域。各子网可采用不同技术实现控制与转发（CI5 与 DI5），但对外均需以票的形式提供服务，当内部路径出现问题后应主动切换路径并维持服务水平不变。接入子网的体制繁杂，若具备定制能力则可通过票提供服务，否则只能预估其带宽和时延水平并引入风险系数。

实现上可采用全集中式的控制结构，以悲观锁方式控制表项生成但并发度较低；或采用集中与分布结合的控制结构，以乐观锁平衡并发度和一致性。

3) 端到端应用跨域协同

跨域协同面向多个管理主体，作用于网关及各子网。“端到端”指跨多子网规划传送路径，“应用”指可完全代表应用需求且子网中立。中转联盟收到网关上报的应用带宽时延需求（CI3）后，完成全局规划并通知子网开通逐程票（CI4），通知网关（CI3）拼接生成联程票（DI4 与 DI6）。全局规划可采用带宽取最小、时延取累加的方式分解需求指标并求解匹配的子网列表，接入子网若不支持定制，规划时以其引入风险系数后的带宽时延为依据。

实现上可采用南北向协同方式，适用于有权威的中转联盟；或采用南北向与东西向协同方式，适用于无权威或弱权威中转联盟但需实现拜占庭容错。

4.3 业务平面

1) 无协议类型传输

传输的核心是实现应用间通信的关联，作用于主机协议栈。“无协议类型”指应用不再感知传输协议类型而通过提出可靠度和顺序性需求以获得相应服务质量，并可灵活组合需求指标以构建传统传输协议类型的能力超集。传输支持应用以各种形态沙箱接入（DI0），通过协议交互实现多应用间关联（SI1），通过主机内存池管理为关联准备缓冲区资源以保障可靠度与顺序性，通过主机时间片调度控制传输自身引入的带宽和时延。

实现上可采用传输需求适配机制对接现有多传输协议类型，兼容性强但只能实现部分传输需求组合；或重构传输层，以模块化机制支持可靠度与顺序性的任意组合，对于长尾需求可按需实现。

2) 弱信令呼叫互联

互联的核心是实现应用间通信的票，作用于主机协议栈。“弱信令呼叫”指应用通过类似用户信令的协议消息触发网络路径开通与资源分配但不知晓网络内部信息，可不必等待呼叫建立并分离处理中继设备与主机协议栈（DI1），应用使用统一方式获得定制化或尽力而为的服务质量。“互联”支持主机接入各类子网（DI2）并通过子网锚定网关（DI3），并使用联程票获得端到端带宽时延保障，支持与网关间的拥塞预警互联控制自身发送速率，同一子网内主机可点对点互联以免绕行网关（DI2）。

实现上可采用叠加于 IP 之上的方式并基于隧道对 IP 接入子网进行穿透；或采用独立于 IP 的方式直接作用于以太网之上，接管网卡并调度数据包收发。

3) 全场景融合标识

标识的核心是实现应用的基础命名，作用于主机协议栈与网关。“全场景融合”指应用可针对人

机物场景在程序中使用标识对外通信：人场景中程序代表用户身份，不同身份对应不同标识而同一用户终端的多个程序可使用相同标识；机场景中程序提供服务/内容，不同服务/内容使用不同标识而同一服务/内容的多实例/副本可使用相同标识；物场景中程序对应设备/编号，不同类型设备使用不同标识而相同类型不同编号设备可使用相同标识。标识由主机协议栈注册发布（SI2）并映射其所在网关的定位符，通信时反向解析该映射并通知中转联盟（SI3）进行全局规划提供联程票。

实现上可改造域名体系以映射各场景标识与网关定位符。虽然标识在不同场景语义不同，但数据发送时会被统一映射为票和关联，网络可根据场景约束实现动态的访问策略。

5　SCN 应用场景示例

本节总体描述了 SCN 在远程控制、算力网络、增强现实三类场景中的应用，并针对各场景中业务对网络能力的调用方式展开介绍。

5.1　远程控制

以工业遥操作、远程手术为代表的远程控制，对速率、时延、丢包有严格要求。假设某企业工厂 1、工厂 2、工厂 3 分别支持远程运动控制、远程人工维修、车间生产运行功能，其网络拓扑与应用分布如图 7 所示。

1) 同城工厂 1、工厂 2 间的运动控制。集中化可编程逻辑控制器（PLC）要以恒稳速率、恒稳时延、不可丢包、严格保序地控制精密工序运动部件；运动部件需反馈受控状态以便 PLC 及时监测故障，其时延需求不超过上限即可。产线长期以该模式运行，可容忍建路和建链的等待时间。

2) 跨城工厂 2、工厂 3 间的人工维修。操作台需以恒稳时延、不可丢包、严格保序地控制维修机械臂，人工维修虽非以固定周期操作但为万无一失仍需恒稳速率；机械臂会持续回传现场视频以便实时交互，要求网络保障其速率下限与时延上限，在不卡顿前提下可允许丢包，为实现连贯播放发生丢包时可局部有序地向应用交付。维修工作定期开展且每次持续较长时间，可容忍建路和建链的等待时间。

上述场景中，网关可由企业自行建设，主机协议栈中的传输与互联需要重构。各部件通过表 3 所示方式调用 SCN 能力并获得所需服务质量。为验证 SCN 在远程控制场景中的应用效果，本文基于未来网络试验设施 CENI（China environment for network innovation）长三角网络，联合华为、宝武集团开展远程云化 PLC 确定性控制试验，在经过 9 跳设备、物理链路 300 km 的情况下，对于不同网络拥塞情况（链路存在微突发流量，即每 800 ms 发送 5 万、10 万、15 万个报文），均可按照确定的时延（<4 ms）与抖动（<10 μs）进行数据传输，满足云化 PLC 对跨广域时延与抖动的指标要求，如图 8 所示。

5.2　算力网络

算力网络旨在实现跨地区的算力协同，对速率、时延、丢包等有要求。假设有 3 个核心数据中心和 2 个边缘数据中心互联以实现分布式训练与层级化推理，其网络拓扑与应用分布如图 9 所示。

1) 分布式训练中参数服务器（PS, parameter server）与工作节点（worker）间的多轮次梯度传播

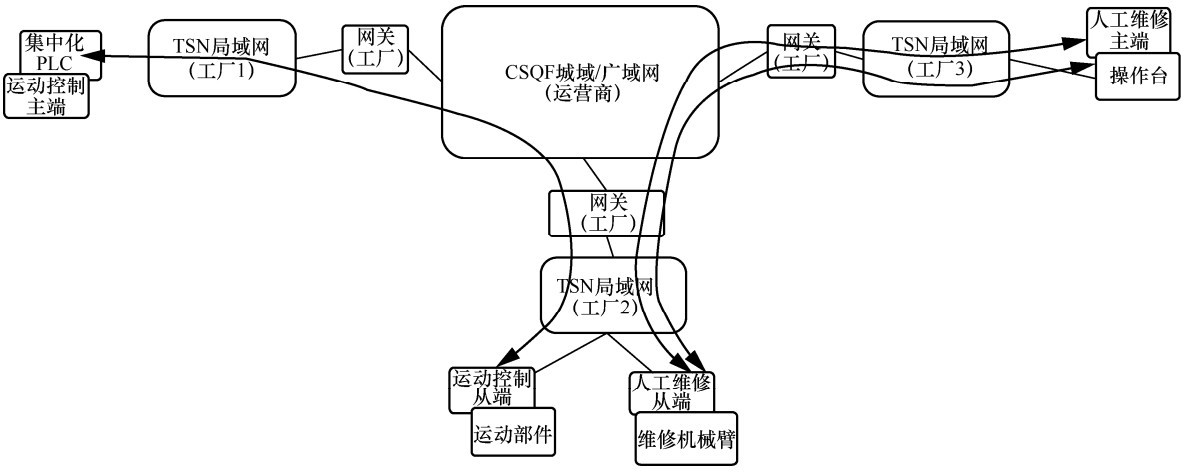

图 7　远程控制网络拓扑与应用分布

表 3　　SCN 应用用例

业务场景		API 调用方式
远程控制	运动控制	PLC：Qos_Claim(<Mc_Master, Mc_Slave>, <恒稳速率、恒稳时延、不可丢包、严格保序>, <建路容忍，建链容忍>)
		运动部件：Qos_Claim(<Mc_Slave, Mc_Master>, <恒稳速率、时延上限、不可丢包、严格保序>, <建路容忍，建链容忍>)
	人工维修	操作台：Qos_Claim(<Hm_master, Hm_Slave>, <恒稳速率、恒稳时延、不可丢包、严格保序>, <建路容忍，建链容忍>)
		机械臂：Qos_Claim(<Hm_Slave, Hm_master>, <速率下限、时延上限、丢包上限、局部有序>, <建路容忍，建链容忍>)
算力网络	分布式训练	worker：Qos_Claim(<worker, PS_Slave>, <平均速率、平均时延、不可丢包、严格保序>, <建路容忍，建链容忍>)
		主 PS：Qos_Claim(<PS_Master, PS_Slave>, <平均速率、平均时延、不可丢包、严格保序>, <建路容忍，建链容忍>)
		从 PS：Qos_Claim(<PS_Slave, Worker>, <平均速率、平均时延、不可丢包、严格保序>, <建路容忍，建链容忍>)
		Qos_Claim(<PS_Slave, PS_Master>, <平均速率、平均时延、不可丢包、严格保序>, <建路容忍，建链容忍>)
	实时性推理	边缘推理：Qos_Claim(<Infer_Coarse, Infer_Precise>, <速率下限、时延上限、逾期丢包、局部有序>, <建路不容忍，建链容忍>)
		核心推理：Qos_Claim(<Infer_Precise, Infer_Coarse>, <速率下限、时延上限、不可丢包、严格保序>, <建路不容忍，建链容忍>)
增强现实	AR 直播	主播程序：Qos_Claim(<Person_A, Person_B>, <速率下限、时延上限、丢包上限、局部有序>, <建路不容忍，建链不容忍>)
		观众程序：Qos_Claim(<Person_B, Person_A>, <速率下限、时延上限、不可丢包、严格有序>, <建路容忍，建链容忍>)
	情境识别	捕捉程序：Qos_Claim(<Person_A, Context_Cognition>, <速率下限、时延上限、逾期丢包、紧急插序>, <建路不容忍，建链不容忍>)
		识别程序：Qos_Claim(<Context_Cognition, Person_A>, <速率下限、时延上限、逾期丢包、局部有序>, <建路不容忍，建链不容忍>)

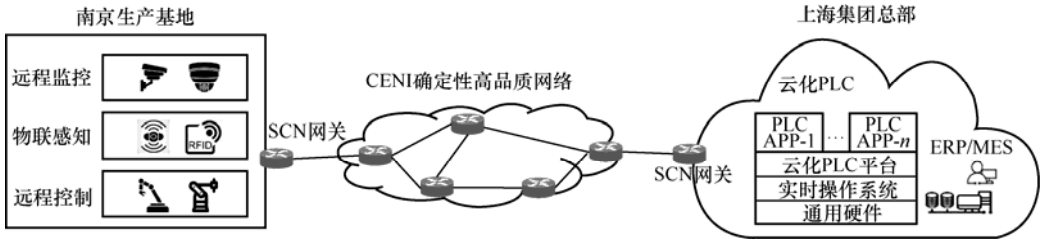

图 8　远程云化 PLC 确定性控制试验

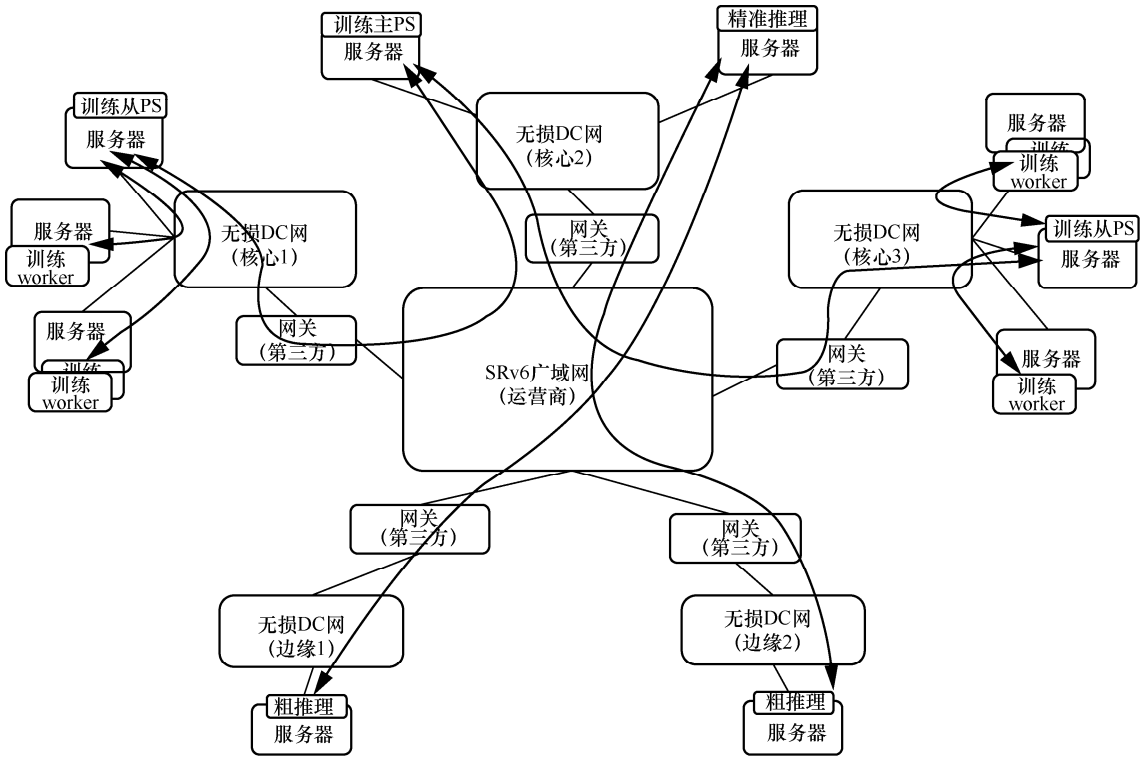

图 9　算力网络拓扑与应用分布

与参数更新将产生大量数据收发。千亿大模型在核心数据中心内可通过声明平均速率、平均时延、不可丢包、严格保序需求来获得数据中心（DC, data center）局域大带宽；万亿大模型可能要协同多 DC 并引入主从参数服务器结构，通过相同需求获得 DC 间的广域大带宽。大模型训练时间较长，则可容忍建路和建链的等待时间。

2) 实时性推理需在核心和边缘 DC 间进行协同。低精度推理模型分布于各边缘 DC 以实时响应客户端推理请求，若精度不足，可调用核心 DC 高精度推理模型进行二次推理，并通过云边之间的回源加速以保障客户端体验。调用的及时性可通过速率下限和时延上限来满足，边−云上行可通过逾期丢包、局部有序加速数据接收，云−边下行需采用不可丢包、严格保序确保正确响应。若协同推理使用短连接，则无法容忍建路的等待时间。

上述场景中，网关可由算网第三方建设，主机协议栈可叠加于 IP 之上或重构。各节点与程序通过表 3 所示方式调用 SCN 能力并获得所需服务质量。为验证 SCN 在算力网络中的应用效果，本文基于自主搭建的长三角网络（在南京、泰州、南通、上海节点各部署一台 DIP 设备），开展了跨广域云原生存储试验，完成南京本地数据库通过广域网向上海远端数据库进行读写（I/O, input/output）操作“写”操作，实现单向 600 km 的 I/O 操作响应时延低于 13 ms，每秒读写次数（I/OPS, input/output per second）达到试验所用固态硬盘（SSD, solid state disk）能力上限的 1/3，从而可支持对 IOPS 要求不是特别高的跨广域存储场景，如图 10 所示。

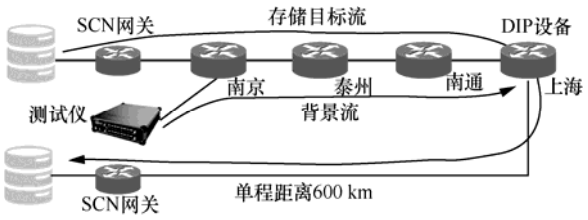

图 10　跨广域云原生存储试验

5.3　增强现实

增强现实将数字信息融入现实情境，对速率、时延等需求较敏感。假设某旅游主播通过增强现实眼镜全球直播其第一视角，并通过情境识别的辅助提示进行解说，其网络拓扑与应用分布如图 11 所示。

1) 以第一视角在主播与观众间直播景点实景。主播到观众持续传送实景视频，要求网络保障其速率下限与时延上限，在不卡顿的前提下可允许丢包，为实现连贯播放，丢包时可局部有序地向应用交付。考虑到观众进入直播间后需尽快开屏，所以无法容忍建路和建链的等待时间；观众对主播进行礼物打赏，可通过速率下限和时延上限满足互动实时性，通过不可丢包和严格顺序保障支付一致性。若礼物打赏使用短连接则无法容忍建路的等待时间。

2) 情境识别发生在增强现实眼镜与服务器间。主播第一视角中出现某地标后将相关图片上传边缘云，需要网络保障其速率下限与时延上限，若在主播移出该区域前无法完成重传则逾期丢包，另外可通过紧急插序优先进行隐私保护；边缘完成地标识别并获得讲解词后回复响应，要求网络保障其速率下限与时延上限，若在主播移出该区域前无法完成重传则逾期丢包，另外可通过局部有序加速数据接收。

上述场景中，网关可由运营商建设，主机协议栈叠加于 IP 之上。不同程序通过表 3 所示方式调用 SCN 能力。针对增强现实场景的试验验证将在后续工作中展开。

6　结束语

本文系统性地分析了“按需定制”内涵并阐述了 SCN 体系架构，但仍存在如下问题以待后续研究。1)子网如何通过提供服务质量获得收益？非收益型子网提供何种水平的服务质量？2)如何定义网

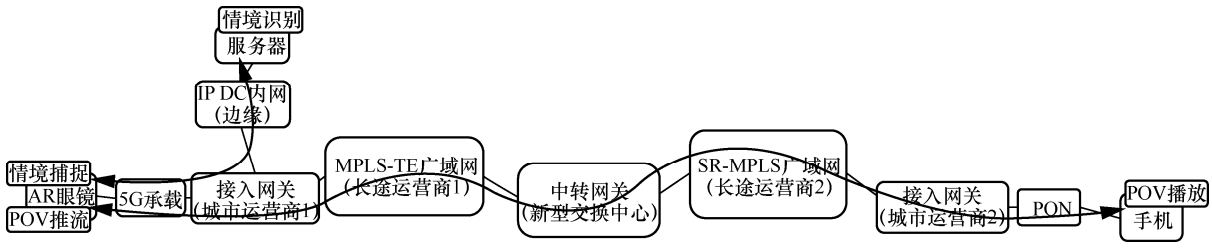

图 11　增强现实网络拓扑与应用分布

关数据包头与协议消息的格式、长度与编码？如何提升系统的可扩展性？3)对于主机协议栈如何实现本地以及与网关间的流量控制？如何实现主机协议栈间的流量控制？4)网络安全作为应用对网络的另一类需求应如何设计？5)计算存储作为应用除网络外的另一类需求如何与网络联合优化？

过去数十年间，应用能够得到的底层网络能力和应用使用网络的方式没有发生过任何实质性的变化。如果说软件定义网络（SDN, software defined network）[44]已经彻底改变了网络运营者管理网络的方式，笔者相信 SCN 未来将深刻改变应用使用网络的方式，并为 NaaS 的实现提供新颖、实用、理想的手段。

参考文献：

[1] CERF V, KAHN R. A protocol for packet network intercommunication[J]. IEEE Transactions on Communications, 1974, 22(5): 637-648.

[2] POSTEL J B. RFC 791: Internet protocol[S]. 1981.

[3] FLOYD S. RFC 793 transmission control protocol[S]. 2001.

[4] FOROUZAN B A. TCP/IP protocol suite[M]. New York: McGraw-Hill Higher Education, 2002.

[5] CLARK D D. Designing an internet[M]. Massachusetts: MIT Press, 2018.

[6] WU D P, HOU Y T, ZHU W W, et al. Streaming video over the Internet: approaches and directions[J]. IEEE Transactions on Circuits and Systems for Video Technology, 2001, 11(3): 282-300.

[7] HANDLEY M, SCHULZRINNE H, SCHOOLER E, et al. RFC2543: SIP: session initiation protocol[J]. Encyclopedia of Internet Technologies & Applications, 2015, 58(2): 1869-1877.

[8] ZHANG L, BERSON S, HERZOG S, et al. RFC 2205: resource reservation protocol (RSVP)——version 1 functional specification[S]. 1997.

[9] ROSEN E, VISWANATHAN A, CALLON R. RFC 3031: multiprotocol label switching architecture[S]. 2001.

[10] AWDUCHE D, MALCOLM J, AGOGBUA J, et al. RFC 2702: requirements for traffic engineering over MPLS[S]. 1999.

[11] AWDUCHE D, BERGER L, GAN D, et al. RFC 3209: RSVP-TE: extensions to RSVP for LSP tunnels[S]. 2001.

[12] ANDERSSON L, CALLON R, DANTU R, et al. RFC 3212: constraint-based LSP setup using LDP[S]. 2002.

[13] KATZ D, KOMPELLA K, YEUNG D. RFC 3630: Traffic engineering (TE) extensions to OSPF version 2[S]. 2003.

[14] LE ROUX J L, VASSEUR J P, BOYLE J. RFC 4105: requirements for inter-area MPLS traffic engineering[S]. 2005.

[15] AYYANGAR A, VASSEUR J P. RFC 5151: inter-domain MPLS and GMPLS traffic engineering--resource reservation protocol-traffic engineering (RSVP-TE) extensions[S]. 2008.

[16] GINSBERG L, DECRAENE B, LITKOWSKI S, et al. RFC 8402: Segment routing architecture[S]. 2018.

[17] PETER O, PRADHAN A, MBOHWA C. Industrial Internet of things (IIoT): opportunities, challenges, and requirements in manufacturing businesses in emerging economies[J]. Procedia Computer Science, 2023, 217: 856-865.

[18] GUO H, WANG F, ZHANG L J, et al. A hierarchical optimization strategy of the energy router-based energy Internet[J]. IEEE Transactions on Power Systems, 2019, 34(6): 4177-4185.

[19] CHENG R Z, WU N, VARVELLO M, et al. Are we ready for metaverse? a measurement study of social virtual reality platforms[C]// Proceedings of the 22nd ACM Internet Measurement Conference. New York: ACM Press, 2022: 504-518.

[20] VALASKOVA K, VOCHOZKA M, LÂZÂROIU G. Immersive 3D technologies, spatial computing and visual perception algorithms, and event modeling and forecasting tools on blockchain-based metaverse platforms[J]. Analysis and Metaphysics, 2022, 21: 74-90.

[21] 刘韵洁, 黄韬, 张娇, 等. 服务定制网络[J]. 通信学报, 2014, 35(12): 1-9.
LIU Y J, HUANG T, ZHANG J, et al. Service customized networking[J]. Journal on Communications, 2014, 35(12): 1-9.

[22] KALITA L. Socket programming[J]. International Journal of Computer Science and Information Technologies, 2014, 5(3): 4802-4807.

[23] BAGNULO M, MATTHEWS P, BEIJNUM I V. Stateful NAT64: network address and protocol translation from IPv6 clients to IPv4 servers[S]. 2011.

[24] DERI L, FUSCO F. Using deep packet inspection in cybertraffic analysis[C]//Proceedings of the 2021 IEEE International Conference on Cyber Security and Resilience (CSR). Piscataway: IEEE Press, 2021: 89-94.

[25] LACHOS D, XIANG Q, ROTHENBERG C, et al. Towards deep network & application integration: possibilities, challenges, and research directions[C]//Proceedings of the Workshop on Network Application Integration/CoDesign. New York: ACM Press, 2020: 1-7.

[26] LEDDY J, VOYER D, MATSUSHIMA S, et al. RFC 8986: segment routing over IPv6 (SRv6) network programming[S]. 2021.

[27] FANG K, LI Y, CAI F. Segment routing over UDP[S]. 2020.

[28] WANG S, GAO K H, QIAN K, et al. Predictable vFabric on informative data plane[C]//Proceedings of the ACM SIGCOMM 2022 Conference. New York: ACM Press, 2022: 615-632.

[29] MIYASAKA T, HEI Y, KITAHARA T. NetworkAPI: an in-band signalling application-aware traffic engineering using SRv6 and IP anycast[C]//Proceedings of the Workshop on Network Application Integration/CoDesign. New York: ACM Press, 2020: 8-13.

[30] NICHOLS K, BLAKE S, BAKER F, et al. RFC 2474: definition of the differentiated services field (DS field) in the IPv4 and IPv6 headers[S]. 1998.

[31] AMANTE S, CARPENTER B, JIANG S, et al. RFC 6437: IPv6 flow label specification[S]. 2011.

[32] BELSHE M, PEON R, THOMSON M. RFC 7540: hypertext transfer protocol version 2 (HTTP/2)[S]. 2015.

[33] KUMAR S, TIWARI R, OBAIDAT M S, et al. CPNDD: content placement approach in content centric networking[C]//Proceedings of the 2020 IEEE International Conference on Communications (ICC). Piscataway: IEEE Press, 2020: 1-6.

[34] KIESEL S, PREVIDI S, ROOME W, et al. RFC 7285: application-layer traffic optimization (ALTO) protocol[S]. 2014.

[35] IYENGAR J, THOMSON M. RFC 9000: QUIC: a UDP-based multiplexed and secure transport[S]. 2021.

[36] LEONARDI L, BELLO L L, PATTI G. Performance assessment of the IEEE 802.1Qch in an automotive scenario[C]//Proceedings of the 2020 AEIT International Conference of Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE). Piscataway: IEEE Press, 2020: 1-6.

[37] WANG S, WU B W, ZHANG C, et al. Large-scale deterministic IP networks on CENI[C]//Proceedings of the 2021 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). Piscataway: IEEE Press, 2021: 1-6.

[38] PAULY T, TRAMMELL B, BRUNSTROM A, et al. An architecture for transport services[S]. 2018.

[39] PENG S P, MAO J W, HU R Z, et al. Demo abstract: APN6: application-aware IPv6 networking[C]//Proceedings of the 2020 IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS). Piscataway: IEEE Press, 2020: 1330-1331.

[40] HEINANEN J, GUERIN R. RFC 2697: a single rate three color marker[S]. 1999.

[41] HEINANEN J, GUÉRIN R. RFC 2698: a two rate three color marker[S]. 1999.

[42] HAUSER F, HÄBERLE M, MERLING D, et al. A survey on data plane programming with P4: fundamentals, advances, and applied research[J]. Journal of Network and Computer Applications, 2023, 212: 103561.

[43] CRACIUNAS S S, OLIVER R S, CHMELÍK M, et al. Scheduling real-time communication in IEEE 802.1Qbv time sensitive networks[C]//Proceedings of the 24th International Conference on Real-Time Networks and Systems. New York: ACM Press, 2016: 183-192.

[44] GOSWAMI B, KULKARNI M, PAULOSE J. A survey on P4 challenges in software defined networks: P4 programming[J]. IEEE Access, 2023, 11: 54373-54387.

[作者简介]

黄韬（1980− ），男，重庆人，博士，北京邮电大学教授，主要研究方向为路由与交换、软件定义网络、网络试验设施等。

张晨（1992− ），男，辽宁锦州人，网络通信与安全紫金山实验室研究员，主要研究方向为计算机网络、软件定义网络等。

肖玉明（1992− ），男，江苏常州人，博士，网络通信与安全紫金山实验室研究员，主要研究方向为网络体系架构、光传送网络、确定性网络等。

余水（1970− ），男，博士，悉尼科技大学教授，主要研究方向为网络科学、大数据、系统建模等。

刘韵洁（1943− ），男，山东烟台人，中国工程院院士，主要研究方向为未来网络体系架构、网络融合与演进等。