IaaS on DPU（IoD）
下一代高性能算力底座

IaaS on DPU(IoD): A New Approach for Next-Gen Cloud Computing Infrastructure

技术白皮书

2024年7月

主编单位

中科驭数（北京）科技有限公司

处理器芯片全国重点实验室

中国计算机学会集成电路设计专业委员会

中科驭数 YUSUR　SKLP 处理器芯片全国重点实验室 STATE KEY LAB OF PROCESSORS, ICT, CAS　CCF 集成电路设计专委会 INTEGRATED CIRCUIT DESIGN

版权声明

前言

DPU 是当下算力基础设施的核心创新之一。如果把 CPU 比做大脑，那么 GPU 就好比是肌肉，而 DPU 就是神经中枢。CPU 承载了应用生态，提供了通用型算力；GPU 提供了高密度各类精度的算力，特别是在智算领域，对系统算力大小有决定性作用；DPU 负责数据在各种 CPU 之间、CPU 与 GPU、以及 GPU 与 GPU 之间高效流通，很大程度上决定了系统是否能协同工作。

DPU 作为数据中心的第三颗“主力芯片”，主要通过其专用处理器优化数据中心的网络、存储、安全等处理性能，助力服务器运行效率显著提升，有效降低成本。因此，在新型数据中心建设时，围绕 DPU 构建数据中心网络的基础设施，在其上挂载了各种计算、存储资源的节点，对于系统的资源弹性、运行效率、性能都大有益处。但是这种使用方式的变化，需要对现有云计算架构进行一定程度的变革，才能充分发挥出 DPU 的优势。云计算中的头部企业 AWS 与阿里云在 DPU 的应用方面也有成功案例，借助其软硬件全栈自研的优势，快速完成了云计算系统的改造工作，实现了 DPU 大规模落地部署，在降低自身运营成本的同时为客户提供更好的使用体验，并产生了可观的经济效益。这种正向循环促进了相关技术栈的快速迭代与成熟，也帮助他们发展成为云计算业务领域的领军企业。

随着众多芯片厂商投身到 DPU 技术领域后，业界对 DPU 的产品形态定义逐渐清晰，DPU 的技术标准也在不断完善。从此 DPU 不再是行业巨头的“专享”技术，基础设施与云计算相关产业参与者都在寻求一种简单高效的方法，将 DPU 的优势运用到自身业务系统之中，例如 Red Hat、VMware、Palo Alto 等公司纷纷推出相关解决方案。这些方案背后共同的本质思想是：将云计算的 IaaS 层组件从服务器侧卸载后围绕 DPU 构筑高性能算力底座，与 AWS、阿里云的技术路线不谋而合。

我们将这种思想所代表的技术路线统一归纳命名为“IaaS on DPU (IoD)”技术路线，简称 IoD。本文重点阐述了 IoD 技术的构成以及与当前主流云计算体系的融合方案，从计算、网络、存储、安全、管控等几个方面进行深度分析，论证了基于 DPU 构建云计算基础设施服务（IaaS）的性能优势与建设路径。

随着 DPU 技术的成熟，不论从功能完备性、系统稳定性还是性价比角度，DPU 均已经具备在大规模生产环境落地应用的条件。某种程度上，IoD 技术已成为下一代高性能算力底座的核心技术与最佳实践。

目录

前言 **ii**

第 1 章 云计算发展趋势 **1**

1.1 云计算系统已经成为数字世界的“操作系统” 1

1.1.1 云计算的发展历程 1

1.1.2 云计算技术特点 2

1.2 AI 产业催生高性能云计算需求 3

1.2.1 AI 技术发展概述 3

1.2.2 云计算性能对 AI 计算影响重大 4

1.2.3 主流 AI 训练的云计算支撑架构 5

1.3 IaaS on DPU(IoD) 算力底座技术路线 6

1.3.1 IoD 发展历程 6

1.3.2 IoD 技术路线解析 7

1.3.3 高性能云计算的规格定义 10

1.4 IoD 高性能云计算应用范式 13

1.4.1 “兼容并包”的公有云 13

1.4.2 “安全强大”的私有云 14

1.4.3 “小巧精美”的边缘云 15

1.4.4 “异军突起”的智算云 15

1.4.5 “电光火石”的低时延云 16

第 2 章 云计算业务模型分析 **18**

2.1 当前主流云计算体系结构 18

2.1.1 硬件部分 18

2.1.2 基础软件 19

2.1.3 云管平台 19

2.1.4 业务服务 20

2.2 计算业务分析 20

2.2.1 裸金属服务器 21

2.2.2 虚拟机 21
2.2.3 容器 22
2.2.4 GPU 服务器 22
2.2.5 应用场景与选择策略 23
2.3 网络业务分析 24
2.4 存储业务分析 25
2.5 安全业务分析 26
2.6 平台服务业务分析 27
2.6.1 数据库 27
2.6.2 中间件 27
2.6.3 服务治理 28

第 3 章 高性能云计算基础设施建设路径 29
3.1 通用算力技术分析 29
3.1.1 CPU 的计算能力发展历程 29
3.1.2 云计算卸载技术为 CPU 算力提升带来的优势 30
3.1.3 IoD 技术为 Hypervisor 卸载提供最佳支撑 32
3.2 智算算力技术分析 34
3.2.1 GPU 的计算能力发展历程 34
3.2.2 GPU 算力提升带来与网络吞吐的矛盾现状 35
3.2.3 无损网络技术为 AI 训练带来的性能提升 36
3.3 云计算网络技术分析 38
3.3.1 云计算网络是算力连通的基础 38
3.3.2 云计算网关是算力开放的门户 39
3.3.3 高性能云计算需要网络卸载进行性能提升 39
3.4 云计算存储技术分析 42
3.4.1 单一存储技术方案无法满足云计算要求 42
3.4.2 云存储需要引入新技术突破性能限制 43
3.4.3 IoD 技术可以提升存算分离架构下的处理性能 44
3.5 云计算安全技术分析 45
3.5.1 纷繁庞杂的云计算安全体系 45
3.5.2 安全处理性能提升需要异构算力加持 46

3.5.3 安全卸载技术在高性能云安全中至关重要 47
3.5.4 DPU 将成为可信计算服务中的重要组件 47
3.5.5 IoD 技术助力构建“零信任”网络 48
3.6 云计算服务治理技术分析 50
3.6.1 服务治理技术是云原生时代的重要基础 50
3.6.2 传统服务治理技术的局限性 50
3.6.3 IoD 技术带来新的服务治理模式 51
3.7 IaaS on DPU(IoD) 高性能云计算全景 51

第 4 章 高性能云计算系统架构持续演进 53
4.1 高性能云计算可观测性建设 53
4.1.1 可观测建设是云计算运维体系的关键环节 53
4.1.2 当前观测方法所面临的难题 54
4.1.3 高性能云可观测性建设建议 55
4.2 轻量级虚拟化系统演进架构革新 56
4.2.1 轻量级虚拟化技术演进路线 56
4.2.2 轻量级虚拟化技术为云计算带来新气象 57
4.2.3 DPU+ 轻量级虚拟化 = 新一代技术革命 58
4.3 “一云多芯”系统融合 59
4.3.1 “一云多芯”的应用困境 59
4.3.2 IoD 技术有助于完善“一云多芯”的服务评估体系 59

第 5 章 高性能云计算为 PaaS 服务赋能 61
5.1 高性能大数据计算服务 61
5.2 高性能中间件服务 62
5.3 高性能数据库服务 62

第 6 章 未来展望 64

第 1 章 云计算发展趋势

1.1 云计算系统已经成为数字世界的“操作系统”

1.1.1 云计算的发展历程

云计算技术的最初起源可以追溯到 20 世纪 50 年代 Christopher Strachey 发表的《Time Sharing in Large Fast Computer》论文，开启了对虚拟化技术探讨的大门。随后的 60 年代，以 IBM 与 MIT 为首的产业与学术巨头纷纷投入相关研究并在虚拟化领域取得了众多突破，最具代表性的事件是 1974 年，Gerald J. Popek 和 Robert P. Goldberg 发表论文《Formal Requirements for Virtualizable Third Generation Architectures》，提出了波佩克与戈德堡虚拟化需求（Popek and Goldberg virtualization requirements）和 I 型与 II 型虚拟化类型。

随着虚拟化技术的不断成熟与基础算力设施能力的提升，使得具备“弹性、按用计量、在线、无限”这几个云计算典型特征的业务类型逐步具备了落地应用的可行性，期间虚拟化技术领域也涌现出了 Qemu、Xen、KVM 等众多明星项目。终于在 2006 年，Google 时任 CEO Eric Schmidt 在搜索引擎大会上首次提出“Cloud Computing”概念，亚马逊在同年成立了亚马逊网络服务公司（AWS），云计算产业轰轰烈烈的发展起来。2010 年，OpenStack 项目创建，标志着云计算技术进入平民化时代，将云计算行业发展正式推向了高潮。

云计算技术的另一个分支，容器技术起源于 20 世纪 70 年代 Unix V7 引入的 chroot 工具，并在 2009 年以 LXC 形式成为 Linux 内核的容器管理器。容器技术凭借显著的轻量化优势取得快速发展并借助 CNCF 社区进行大力推广，在 2018 年发布的云原生技术定义中，容器被确立为云原生的代表技术之一。随着业务的多样化发展，云原生技术逐渐显现出强大的统治力，成为未来发展的主要方向。

伴随着云计算的蓬勃发展，当前世界上的主要算力基础设施几乎都是通过云计算技术进行管理与调度，可以说云计算技术已经成为数字世界的“操作系统”。

1.1.2 云计算技术特点

云计算的发展呈现出显著的业务驱动特征，当前 AIGC、IoT、5G/B5G、Web3.0 等行业的发展一方面要求云计算技术能为其提供融合性的底层技术支撑，能够按需以裸金属、容器或虚拟机形式承载上层业务，另一方面对云计算性能也提出了前所未有的要求。于是我们看到，OpenStack 社区涌现出大量容器相关项目，如 Zun、Magnum、Kyrur 等，CNCF 社区中的 Kubevirt、Metal[3] 等项目也逐渐成熟，这些都是为提供多模态服务类型做出的努力。同时，融合了 CPU、GPU 与 DPU 的“3U 一体”新型服务器成为当前云计算算力基础设施的主力形式，CPU 负责调度管理与运行业务进程，是通用“算力”的承载组件，GPU 负责提升大规模并行运算能力，是智算“算力”的核心引擎，DPU 负责算力集群基础设施卸载与集群的联通，三者通力合作，构成了高性能云计算的基础底座。

历史的经验告诉我们，技术的发展总是呈现出螺旋式上升的样貌。也总有人调侃，当前的问题都可以在故纸堆中找到答案。虽然异构运算并非新鲜事物，但随着单项技术的突破与不同技术领域间的融合，在当下，如图1.1所示的基于“3U 一体”的融合算力基础设施构建的融合性云计算平台，正是支撑不断爆发的上层业务应用运转的最佳实践方案。

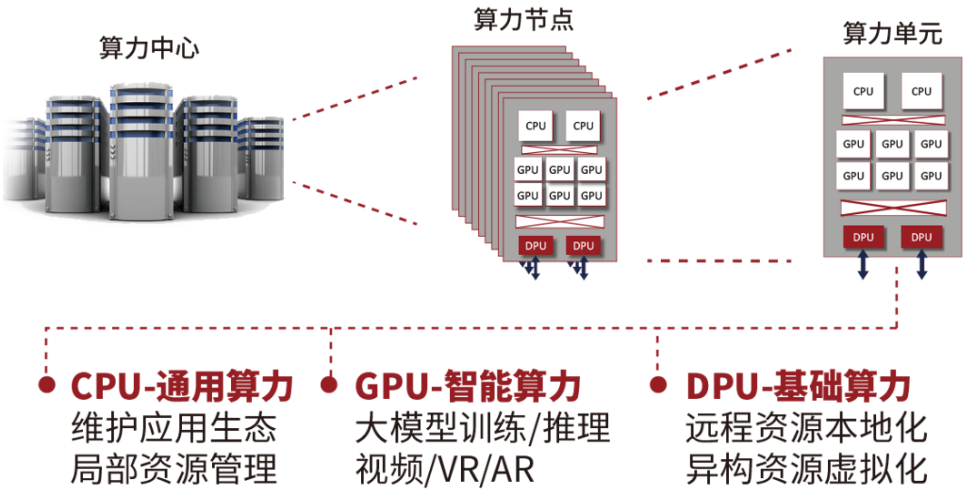

图 1.1: “3U 一体”融合基础设施

总体来说，当前云计算技术的发展呈现出如下典型特征：

- 业务承载多模化

  为了满足业务向云端平滑迁移的需求，会要求云平台能够适配业务系统的当前情况，从容器、虚拟机、裸金属中选择最佳的云上承载方式。例如对硬件设施有特

殊需求的业务需要通过裸金属承载，对操作系统有特殊需求的业务以虚拟机承载，其余业务以容器承载。

- 计算性能极致化

  在 AIGC 大爆发的背景下，上层业务系统从网络性能、存储性能、安全性能等众多方面都对云平台提出了更高的要求，百 G 级别的以太网络接入能力已经逐渐成为云计算系统的标配，400G 的无损网络接入也逐渐在行业落地。

- 系统构成组件化

  云计算技术体系越来越庞杂，单独的封闭体系很难满足来自业务系统层出不穷的各种需求，良好的模块划分与 API 设计已经成为主流云计算系统的构成基础。“开放、可替换”模式已经成为云计算技术架构的主旋律。

1.2 AI 产业催生高性能云计算需求

1.2.1 AI 技术发展概述

人工智能（Artificial Intelligence，简称 AI）是指通过计算机技术和算法模拟人类智能的一种技术。目标是使计算机能够模拟人的思维方式和行为，让计算机可以像人类一样思考和学习，并最终实现自主决策的智能化行为。

进入 21 世纪后，互联网的普及和大数据的爆发为 AI 提供了丰富的训练材料，加速了算法的发展。2006 年加拿大 Hinton 教授提出了深度学习的概念，极大地发展了人工神经网络算法。2012 年，AlexNet 在 ImageNet 竞赛中取得突破性成果，标志着深度学习时代的到来。

当前人工智能处于深度学习和生成式 AI 大发展的时期。过去十多年基于深度学习的人工智能技术主要经历了如下的研究范式转变：从早期的“数据标注监督学习”的任务特定模型，到“无标注数据预训练 + 标注数据微调”的预训练模型，再到如今的“大规模无标注数据预训练 + 指令微调 + 人类对齐”的大模型，经历了从小数据到大数据，从小模型到大模型，从专用到通用的发展历程，人工智能技术正逐步进入大模型时代。自 2017 年 Google 提出 Transformer 模型以来，AI 大语言模型（LLM，Large Language Model）已取得飞速进展。

2022 年底，由 OpenAI 发布的基于 GPT3.5 的语言大模型 ChatGPT 引发了社会的广泛关注。在“大模型 + 大数据 + 大算力”的加持下，ChatGPT 能够通过自然语言交互完成多种任务，具备了多场景、多用途、跨学科的任务处理能力。以 ChatGPT 为代表的

大模型技术可以在经济、法律、社会等众多领域发挥重要作用，引发了大模型的发展热潮。

2024 年被称为 AGI 元年，文生视频大模型 Sora 的问世再次引爆了行业热点，在通用问题上 AI 通过自学习实现从 GPT 到 GPT-Zero 的升级，开启了 AGI 时代。

1.2.2 云计算性能对 AI 计算影响重大

随着大模型和生成式 AI 的迅速发展，大模型参数规模和数据集不断增加，2017 年到 2023 年 6 年间，AI 大模型参数量从 Transformer 的 6500 万，增长到 GPT4 的 1.8 万亿，模型规模增长超过 2 万倍。业界对智算算力的需求也水涨船高，据 Al Now《计算能力和人工智能》报告指出，早期 Al 模型算力需求是每 21.3 个月翻一番，而 2010 年深度学习后（小模型时代），模型对 Al 算力需求缩短至 5.7 个月翻一番，而 2023 年，大模型需要的 Al 算力需求每 1-2 个月就翻一番，摩尔定律的增速显著落后于社会对 Al 算力的指数级需求增长速度，即“Al 超级需求曲线”遥遥领先传统架构的 Al 算力供给，带来了 Al 芯片产能瓶颈涨价等短期市场现象。根据工信部等部委 2023 年 10 月发布《算力基础设施高质量发展行动计划》，截至 2023 年 6 月底，我国算力总规模达到 197EFLOPS，智能算力规模占比达 25.4%。按照该计划，我国 2023 年底智算算力要达到 220EFLOPS，2024 年要达到 260EFLOPS，2025 年要达到 300EFLOPS。如此庞大的智算算力需求对底层智算基础设施性能、稳定性、成本及安全性方面带来巨大技术和成本挑战。特别是智算云基础设施在算力、网络、存储、调度等方面的性能对 AI 训练过程有关键影响，是决定 AI 大模型训练质量（效率、稳定性、能耗、成本、信任等）的关键因素。底层智算云基础上设施性能对 AI 训练的质量有着重大影响，体现在多个方面：

1. 数据处理能力：千亿级模型的训练需要使用文件、对象、块等多种存取协议处理处理 PB 级规模的数据集，万亿级模型的训练处理 checkpoint 的读写吞吐性能要求高达 10TB/s。现有智算存储设施在协议处理、数据管理、吞吐性能等方面面临诸多挑战。传统智算的分布式文件存储系统仅支持百节点级别扩展，节点规模小，难以满足万卡级集群的数据吞吐性能要求。高性能云计算平台能够高效地存储和处理海量的训练数据。数据预处理、清洗和标注等步骤可以在云端高效完成，确保输入模型的数据质量，从而提升模型的准确性和泛化能力。
2. 算力支持：云计算提供了弹性且强大的计算资源，特别是 GPU 和 TPU 等加速器，能够大幅缩短 AI 模型的训练时间。大规模并行处理能力使得处理复杂的深度学习模型成为可能，这对于模型收敛速度和训练质量至关重要。

3. 分布式训练：云计算平台支持模型的分布式训练，通过多节点并行计算，可以处理更大规模的数据集和更复杂的模型，同时减少训练时间。这对于大型语言模型、图像识别模型等尤为重要。
4. 模型优化：利用云计算资源，可以进行大量的模型调优实验，比如超参数调优、模型架构搜索等，找到最优模型配置。云计算的灵活性允许数据科学家和工程师快速迭代，提高模型性能。
5. 存储与 IO 性能：高速的存储系统和优化的 IO 性能减少了数据读写瓶颈，确保训练过程中数据的快速存取，这对于大规模数据处理和模型训练至关重要。
6. 资源调度与自动化：云平台的智能资源调度能力可以根据 AI 训练任务的需求动态调整资源分配，保证计算资源的高效利用。自动化工具和服务进一步简化了模型训练流程，降低了操作复杂度。
7. 成本效益：云计算的按需付费模式降低了进入门槛，使得企业和研究机构无需前期大量投资硬件设施，就可以开展高级 AI 项目，促进了 AI 技术的普及和创新。

综上所述，云计算不仅提供了必要的基础设施来支撑 AI 训练，还通过其灵活、高效、可扩展的特性，直接促进了 AI 模型训练质量和效率的提升，推动了 AI 技术的快速发展和广泛应用。

1.2.3 主流 AI 训练的云计算支撑架构

智算云数据中心架构可划分为基础设施层、管理调度层、大模型平台层、AIGC 应用层，各层的作用说明如图1.2所示：

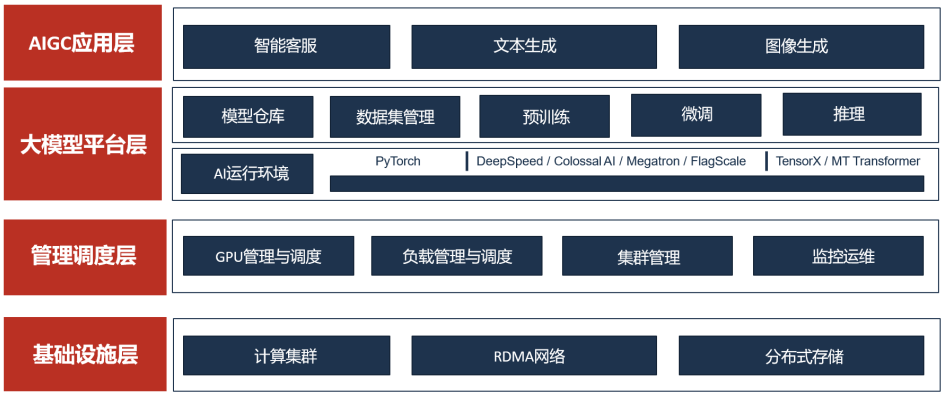

图 1.2: 智算中心架构

- 基础设施层

适度超前建设，满足面向未来客户的算力多元化需求，基于开放计算，兼顾软硬

一体协同，构建多元融合型架构，将通用 CPU 与多元异构芯片集成，融合多种算力，充分释放算力的价值。

基于领先的 AI 服务器为算力单元，支持成熟丰富的软件生态，形成高性能、高吞吐的计算系统，为 AI 训练和 AI 推理生产输出强大、高效、易用的计算力。

- 管理调度层

  硬件资源与 AI 应用松耦合，CPU 算力与 AI 算力按需配比，AI 算力资源按需调用，随需应变，显存可扩展、算力可超分。

  系统调度层一般采用云计算技术，根据资源池内算力资源使用情况，统一调度 AI 任务，AI 算力资源采用声明式申请，实现资源自动聚合，满足单机单卡，单机多卡及多机多卡不同场景要求。

- 大模型平台层

  覆盖 AI 模型开发部署全生命周期，提供预置行业算法、构建预训练大模型，推进算法模型持续升级、提供专业化数据和算法服务，让更多的用户享受普适普惠的智能计算服务。

- AIGC 应用层

  使用云计算技术作为底层支撑，利用训练过的模型对外提供 AI 服务，包括语音交互服务、文本交互服务、图像生成服务与视频生成服务等。需要满足业务系统高可用性与快速迭代等需求。

当前，主流 AI 框架主要采用云原生技术作为底层支撑，主流 AI 分布式训练框架如图1.3所示。

1.3 IaaS on DPU(IoD) 算力底座技术路线

1.3.1 IoD 发展历程

为了将算力基础设施的能力充分发挥出来，云计算系统整体架构也在不断演进。传统的 IaaS 平台组件功能全部由 CPU 算力承载，但是随着对云计算性能需求的提升以及极致利用 CPU 算力需求的发展，基于 DPU 构建 IaaS 平台的理念被提出与论证。这其中的佼佼者以亚马逊网络服务（AWS）为代表，根据披露的材料分析，自 2013 年发布 Nitro (DPU) 设备以来，AWS 的云计算服务体系逐渐改造为基于 DPU 构建并运行在 Nitro 设备中，服务器上的 CPU 算力被完全池化并以近乎 100% 的原始算力性能向客户售卖。以此为基础，AWS 构建了一整套高性能、高稳定性的云服务体系，成为全球范围内最大

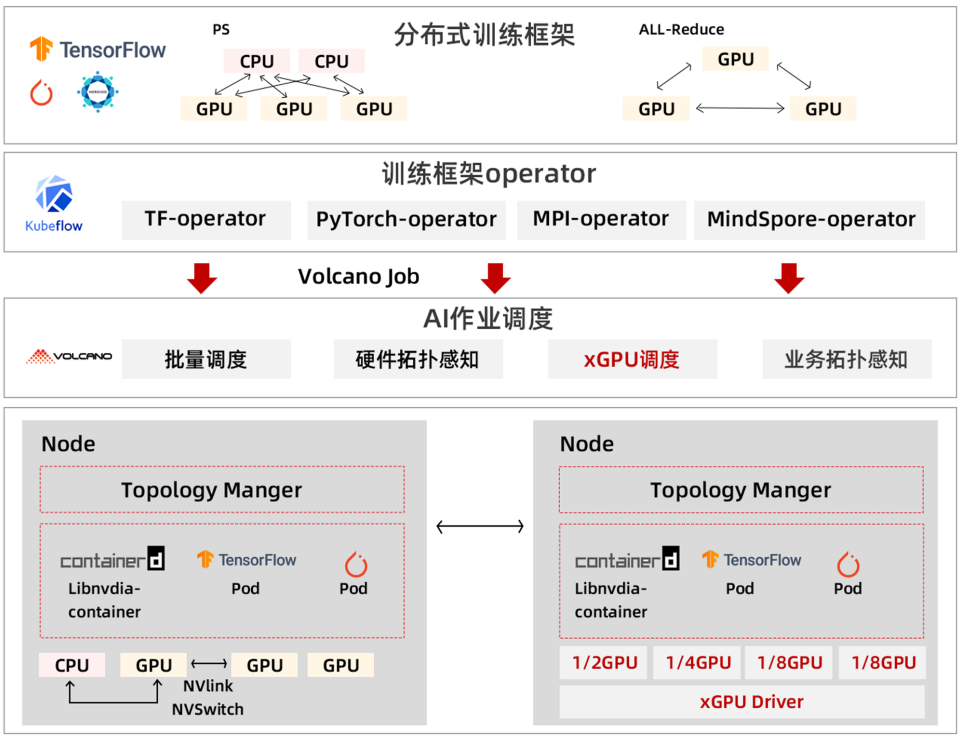

图 1.3: 主流分布式训练框架

的云服务供应商。国内阿里云也采用类似的体系，其云服务体系与其自研的 DPU 设备紧密配合，帮助阿里云取得了巨大的成功。

因此，IaaS on DPU，简称为 IoD，并非全新的概念，而是已经被业内头部企业充分论证过的技术方向，其商业价值也已经经过市场的考验。但是如 AWS 等企业的 DPU 与云平台经过高度订制化，难以简单在业内推广开来。随着 Nvidia、Intel、AMD 等芯片行业的领军企业进军 DPU 赛道后，如何探索出一条通用云计算系统与标准 DPU 产品结合的路径成为业内关注的焦点。上述芯片企业通过行业论坛或技术文章等方式发表过众多类似的解决方案，将部分 IaaS 平台能力下沉到 DPU 中。众多云计算供应商如 Red Hat、VMware 等也顺应趋势，展开了相关研究并在其产品中纳入了相关能力。

其中关键性事件是 OPI 与 ODPU 等标准化组织的成立，云厂商与 DPU 供应商纷纷参与其中探讨 DPU API 规范，DPU API 规范可以将云平台与 DPU 设备解耦，将 IoD 技术规范化并全面推向云计算行业。

1.3.2 IoD 技术路线解析

IoD 技术的核心思想是依托于 DPU 的异构运算能力，将云计算平台的基础设施组件尽可能下沉到 DPU 承载，实现节约 CPU 开销与提升 IaaS 服务性能的目的。同时，基

础设施组件下沉到 DPU 之后，可以为服务器侧运行的各种业务提供一致的网络、存储与安全底座，可以更好的将虚拟机、容器与裸金属的业务调度收敛到统一平台。如图1.4所示为 IoD 架构下的系统模型。

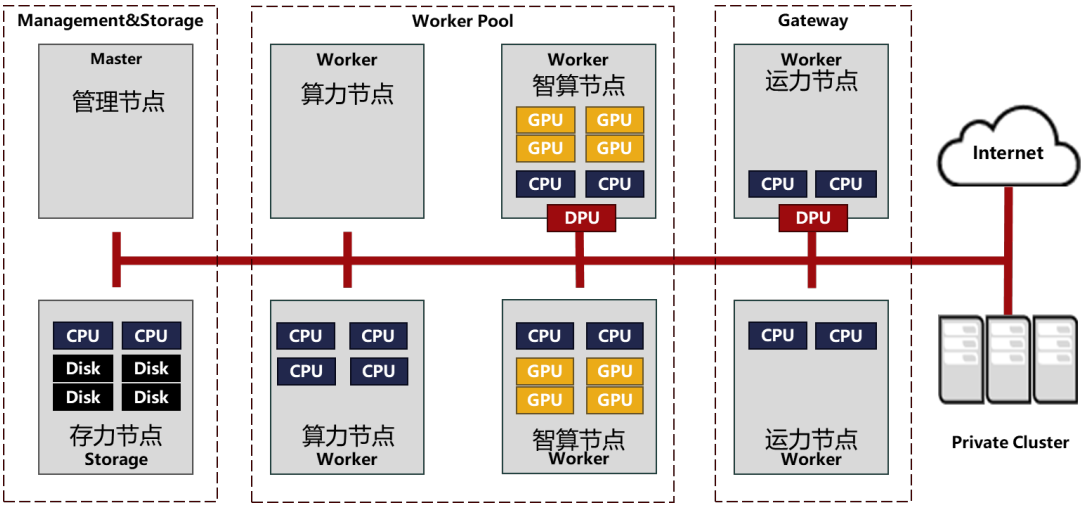

图 1.4: IoD 系统模型

当前开源领域最主流的云计算平台有 Openstack 体系与 Kubernetes 体系，虽然二者在虚拟技术和容器编排方向各有侧重，但它们可以互补使用，并且随着不断地技术迭代，二者的业务覆盖范围也有所重叠。

总的来说，Openstack 系统更注重对物理设备的模拟，对业务隔离性与复杂业务系统的支持更加友好，适合作为重点以虚拟机为主并需要复杂网络管理和多租户环境的企业级 IaaS 平台使用。它在虚拟机管理、网络配置和企业级特性方面表现出色。Kubernetes 系统则是从上层业务的架构设计与生命周期管理角度出发，提供更好的业务编排特性与抽象层次更高的网络与存储特性，拥有更加丰富的系统组件和更加灵活的插件机制，更适合作为以容器业务为主的 IaaS+PaaS 综合平台使用，尤其是在需要高效管理容器化应用和自动化运维的场景中。值得一提的是，Kubernetes 体系中提供的 Service Mesh 组件，在底层平台提供了丰富的服务治理能力，其内置的服务发现、负载均衡、业务自愈、高可用、业务跟踪、滚动发布等特性大幅简化了业务系统的架构设计难度。借助于 Kubernetes 体系更友好的插件机制，CNCF 社区发展迅速，也逐渐补足了 Kubernetes 其在虚拟化与业务隔离性方面的劣势。

从另一个方面来讲，据 Gartner 统计，新建云计算平台中选择基于 Kubernetes 构建的比例越来越高，尤其是以 AI 相关的云计算基础设施中，Kubernetes 体系占据绝对数量优势，已经成为云计算技术发展与应用最主要的方向。

由于以上原因，IoD 技术架构更推荐选择采用扩展 Kubernetes 的形式，通过众多插

件将 DPU 能力引入到云原生技术栈中，并将 Worker 节点的基础设施组件完全运行在 DPU 中。

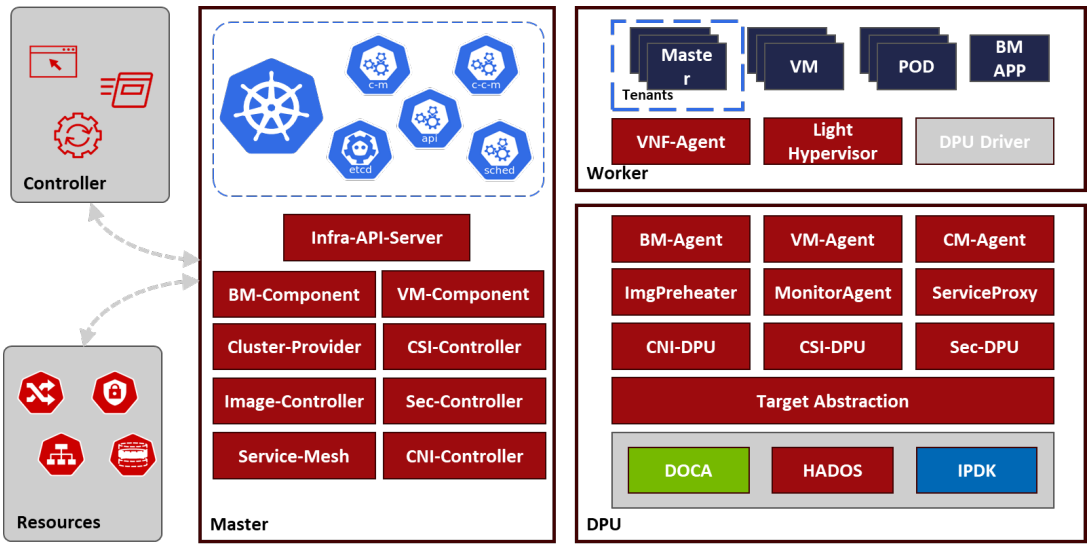

图 1.5: IoD 技术架构图

如图1.5所示，典型的 IoD 技术部署在 DPU 上的核心组件包括：

- BM-Agent：裸金属业务组件，裸金属系统盘采用 DPU 虚拟的磁盘，通过在虚拟磁盘中挂载用户镜像，可以实现裸金属业务的快速切换与业务温迁移。
- VM-Agent：虚拟机业务组件，通过监控本机虚拟机声明和实例资源，实现对服务器上所有虚机实例的管理。
- CM-Agent：容器业务组件，定期从 Kubernetes 接收新的或修改的 Pod 规范，并确保 Pod 及其容器在期望规范下运行。
- CNI-DPU：网络插件，提供高性能网络卸载方案，同时提供高性能网络接入组件、可以灵活高效对接各种外部网络。
- CSI-DPU：存储插件，提供高性能存储卸载方案，同时兼容多种存储方案。
- Sec-DPU：安全插件，提供高性能分布式安全方案，是集群网络安全策略执行的锚点。
- Service Agent：服务治理组件，可以根据业务需要通过流量劫持的方式实现服务治理功能，为虚拟机、容器以及裸金属业务提供通用的服务治理能力。
- Image-Preheater：镜像预加载组件，对通用的基础镜像进行多节点缓存，容器优先调度使用具有预热镜像的 Worker 节点，以避免其频繁拉取远端镜像。
- Target Abstraction：驱动抽象层，用来对接各种 DPU 产品，屏蔽底层差异，需要与不同 DPU 开发套件进行适配。

同时，为了将 DPU 融入进 Kubernetes 系统，IoD 体系下的 Kubernetes 平台也需要进行相应扩展，核心扩展包括：

- BM-Component：实现裸金属业务定义与生命周期管理。
- VM-Component：实现虚拟机业务定义与生命周期管理。
- Kubernetes 原生组件：实现容器业务定义与生命周期管理。
- CNI-Controller：实现网络服务定义与管理。
- CSI-Controller：实现存储服务定义与管理。
- Sec-Controller：实现安全服务定义与管理。
- Service-Mesh：服务治理组件，实现服务治理规则定义与管理。
- Image-Controller：镜像管理组件，提供容器、虚拟机、裸金属镜像统一管理与预热策略下发功能。
- Cluster-Provider：提供集群部署与 DPU 节点生命周期管理功能。
- API-Server：提供对外 API 服务，暴漏底层 IaaS 能力。

通过以上设计，已经完成了云计算 IaaS 体系与 DPU 的结合并将主要组件下沉到 DPU 系统。

类似的设计方案对 Openstack 体系也完全适用。

值得一提的是，通过前述方案中 API-Server 暴露的能力，在已经完成 IoD 基础环境搭建之后，不管是 Openstack 体系或者其他云平台体系，都可以通过简单的 API 集成，实现集群的 IoD 改造。

通过 IoD 技术，可以为云计算体系提供以 DPU 为核心构造、软硬件一体化高性能计算底座，对外提供统一管理、高可扩展性、高性能、低成本的 IaaS 服务。在硬件层面为“3U 一体”和“一云多芯”的异构算力管理提供更好的解决方案。通过对网络、存储、安全、管理等负载的卸载，释放服务器的硬件资源，实现性能加速，提升基础设施运行效率。此外，通过 IoD 的统一底座技术，可以为云计算系统提供容器、虚拟机、裸金属业务的统一调度和运维管理能力，提升运维管理效率。

1.3.3 高性能云计算的规格定义

1.3.3.1 高性能网络规格定义

在高性能云计算底座中，高性能网络需要满足一系列严格的要求：

1. 带宽（Throughput）：高性能计算集群通常需要处理大量数据传输，因此网络必须

提供极高的带宽，以确保数据可以在节点间快速流动，减少传输瓶颈。例如，在科学计算、大数据处理、深度学习训练等场景中，数据集可能达到 PB 级别，要求网络带宽至少达到百 GB 甚至更高。

2. 延迟（Latency）：对于需要频繁通信和数据交换的应用，网络延迟需要控制在微秒级甚至纳秒级，以保证系统的响应速度和实时性。
3. 并发连接（Concurrency）：在高负载和大规模分布式环境中，单节点需要同时处理成数万并发连接，确保每个连接都能得到及时响应。
4. 网络服务质量 (QoS): 不同类型的数据流和服务对网络资源的需求和优先级不同，QoS 功能允许网络管理员根据服务类型动态分配带宽和其他资源，确保关键应用的性能不受非关键流量的影响。
5. 冗余:（Redundancy）高性能网络应具备高度的弹性和冗余设计，即使部分组件出现故障，也能保持网络的连通性和稳定性。这意味着网络需要有多条路径和备份链路，以及自动故障检测和恢复机制。
6. 可管理性（Manageability）：网络应易于管理和监控，提供详细的性能指标和日志记录，帮助运维人员及时发现和解决问题。

1.3.3.2 高性能存储规格定义

在云计算场景下，存储处理性能直接影响着系统的整体性能和用户体验，高性能存储对于处理性能的规格定义通常包括以下关键指标和参数：

1. 吞吐量（Throughput）：吞吐量是指存储系统能够处理的数据量或信息流量。高性能存储目前主流性能在 100-400Gb/s，根据云规模的不同略有浮动。
2. IOPS（Input/Output Operations Per Second）：IOPS 是指存储系统每秒钟可以执行的输入/输出操作次数。高性能存储后端需要提供至少千万级的总 IOPS 数据处理能力，特定场景如 AIGC 应用中，单个存储前端也需要百万级的单磁盘 IOPS 能力。
3. 延迟（Latency）：存储系统的延迟是指数据请求从发起到完成所需的时间。考虑到存储系统的额外延迟开销，高性能云计算的延迟总体开销应控制在亚毫秒级（即百微秒量级）。
4. 容量（Capacity）：存储系统的容量指的是其可以存储的数据量。在高性能存储方案中，存储容量可以达到 EB 级。
5. 鲁棒性（Robustness）：高性能存储系统需要具备高可靠性和高可用性，以确保数据的安全性和持续性。这包括数据冗余、故障恢复能力、备份与恢复机制等。

6. 数据保护（Security）：高性能存储系统需要提供有效的数据保护机制，包括数据加密、访问控制、数据备份等，以确保数据的安全性和完整性。
7. 扩展性（Extendibility）：高性能存储系统应具备良好的扩展性，能够根据需求灵活扩展存储容量和性能，以适应不断增长的数据需求。
8. 融合性（Integration）：高性能存储系统通常支持多种存储访问协议，如 NFS、SMB、Object、iSCSI、FC、NVMe-oF 等，以满足不同应用场景的需求。

1.3.3.3 高性能安全规格定义

对于高性能云计算场景，传统安全设备通常部署在网络边界处，无法部署在安全计算环境中，而传统网络安全软件无论是防火墙、VPN、IPS 等产品都非常消耗服务器主机算力资源，这将严重影响服务器所承载业务应用的客户体验，也是当前计算环境的安全防护比较薄弱的一个重要原因。

1. 算力损耗（Loss-rate）：不因开启网络安全功能而导致处理高性能网络处理性能明显下降；安全计算环境开启网络安全软件功能后，服务器主机算力资源消耗小，平均算力占用率不超过 5%。
2. 吞吐量（Throughput）：吞吐量是在各种帧长的满负载双向发送和接收数据包而没有丢失情况下的最大数据传输速率，开启安全功能后，安全吞吐量可能为正常情况的 70-90%。
3. 延时（Latency）：开启安全功能后，网络延时需要控制在微秒级。
4. 会话数量（Number of session）：最大会话数量指基于防火墙所能顺利建立和保持的最大并发 TCP/UDP 会话数，对于高性能网络，最大会话数量至少为千万级。
5. 每秒新建连接数（Connection Per Second,CPS）：每秒新建连接数指一秒以内所能建立及保持的 TCP/UDP 新建连接请求的数量，每秒新建连接数通常需要几十万级。
6. 误报率（False alarm rate）：误报率是指某种类型的网络业务流量被误识别为其它类型网络业务流量在所有被测试网络业务流量样本中的占比，此指标需要接近于 0%。
7. 漏判率（Miss rate）：漏判率是指网络业务流量中预期应该被识别出来的业务类型没有识别到的网络业务流量占总网络业务流量样本的百分比，此指标接近于 0%。
8. 识别准确率（Identification accuracy）：识别准确率是指测试用的网络业务流量样本中被准确识别的比例。此指标识别准确率接近 100%，至少要求在 95% 以上。

9. 隧道会话数（Number of IPSec tunnels）：最大 IPSec 隧道会话数量指 IPSec 隧道会话所能顺利建立和保持的最大并发会话数，IPSec 隧道会话数量至少为数万级到数十万级。
10. 每秒新建 IPSec 会话数（IPsec Connection Per Second）：每秒新建连接数指一秒以内 IPSec 所能建立及保持的 IPSec 隧道会话的数量，至少要求在几千或数万级。

1.4 IoD 高性能云计算应用范式

1.4.1 “兼容并包”的公有云

公有云服务是最典型的云计算应用场景，通过互联网将算力以按需使用、按量付费的形式提供给用户，包括：计算、存储、网络、数据库、大数据计算、大模型等算力形态。

基础设施能力的提升会为公有云服务商带来很多优势：

- 拓展用户宽度：云计算服务的性能是对部分客户至关重要，云计算服务的网络带宽、存储性能、响应时间等往往成为客户是否选择一家云厂商的关键因素，因此更高的性能有助于云计算厂商有效的拓展用户宽度。
- 降低客户成本：更高的性能意味着云计算客户可以用更短的时间完成任务，直接关系到客户的使用成本，高性能云计算平台可以帮助云服务商在价格竞争中取得身位领先。
- 极致资源利用：通过卸载技术降低 CPU 负荷意味着相同集群规模可以提供更多的可售卖资源，因此高性能云底座将直接关系到云服务商的生产能力。
- 提升 ROI：IoD 技术不仅仅可以提升算力资源池的服务性能，在网络资源池、安全资源池与存储后端等领域也可以通过性能提升为云服务商带来更高的经济效益。

整体看来，部分公有云厂商在选定技术路线后会采用自研 DPU 的方式来获得更高的业务定制性，但芯片研发的巨额资金投入也带来了巨大的不确定性。其余大部分云服务厂商会选择引入硬件供应商的设备来构筑自己的技术体系，此时 DPU 设备的规范性、可定制能力以及服务支持能力将成为至关重要的因素。

1.4.2 “安全强大”的私有云

私有云是仅为单一组织或企业专用的一种云计算环境，相对于公有云，它提供了更高的控制权、隐私性和定制化能力。私有云一般部署在企业内部的自有数据中心（本地私有云），也可以托管在第三方服务提供商的数据中心（托管私有云）。由于其承载的业务范围相对固定，因此除了个别应用类型为，私有云对性能的需求往往聚焦在某个方向，并不像公有云需要全方位的性能提升。但是，私有云的应用对于运维隔离、安全管控等需求更为强烈，IoD 技术也将为私有云带来诸多好处：

- 运维隔离：通过 IoD 技术，云平台的基础设施层与业务运行环境做到了最大限度的隔离，并且各种基础设施能力仅通过虚拟设备形式对业务系统呈现，最大限度的完成了运维与业务的解耦部署。
- 高安全性：借助 DPU 的能力，可以更好的实施“分布式防火墙”与“零信任”网络方案，并且通过 DPU 参与到数据收发路径的方式，能够更方便的实现集群业务监控。
- 性能提升：通过定向的性能提升，能够帮助私有云延续老式设备的服役周期，保护既有投资。
- 节能减排：通过 IoD 技术提升集群整体性能，可以用更少的设备与能耗提供同等算力，帮助客户实现节能减排的目标。

IoD 技术对于私有云建设的优势非常明显，但是目前在运行的私有云改造确面临着诸多问题，涉及适配改造、业务迁移等方面，典型的建设方案有：

1. 新建集群并逐步完成业务迁移与 IoD 集群扩容，此方案要求新建的 IoD 集群能够与源集群较好的适配与互通，能够实现安全方案的平滑迁移以及能够共享存储系统。此方法优势是迁移过程较平滑，但是整体项目实施周期可控性较低，迁移启动时无法充分验证系统对上层业务需求的支持情况。
2. 推动当前云平台完成 IoD 业务改造并确保同一平台同时支持 DPU 服务器与非 DPU 服务器同时存在的情况。此方案的优势是可以保持云平台的一致性，在前期业务改造与论证阶段完成尽可能多的业务验证，完成平台改造后的迁移风险较小，但是存在前期资源投入大的缺点。

1.4.3 “小巧精美”的边缘云

边缘云是将计算、存储和网络资源部署在靠近用户、设备或数据源的位置，以提供低延迟、高带宽和实时处理能力的云计算服务。这些资源通常位于电信基站、商业园区、区域数据中心或本地服务器等边缘设备上。具有规模小，部署环境受限等特点，优势是能够减少数据传输的延迟，提高响应速度，优化带宽使用，增强数据隐私和安全性。IoD 技术对于边缘云的发展来说也具有重大意义：

- 空间节约：由于边缘云的部署方式往往受空间限制较大，集群规模很小，因此借助 IoD 技术，不仅可以将工作节点组件部署在 DPU 上，还可以将云平台管理组件也运行在 DPU 中，进一步减少边缘集群服务器数量，实现对物理空间的节约。
- 定制性强：边缘云部署的业务往往具有很强的定制性，借助 DPU 的高度可编程特性，可以对实现对特定类业务的优化处理。例如 5G MEC 系统可以借助 DPU 实现更高的 UPF 数据转发性能与 SD-WAN 接入能力，视频监控边缘云系统中可以实现视频数据包的预处理等。
- 性能提升：DPU 的网络与存储卸载能力对边缘云性能提升大有帮助，同时大量边缘部署的应用对系统时延较为敏感，DPU 系统的低时延能力也可以帮助边缘云系统应对更多的业务挑战。

当前还处于边缘云业务大规模部署的初期阶段，此时正是边缘云技术体系引入 DPU 应用的最佳时机，但是同样面临的最大挑战是需要 DPU 系统对不同边缘云应用需求的优化与增强，对 DPU 的可编程能力与服务厂商的定制研发支撑能力具有很强的要求。

1.4.4 “异军突起”的智算云

智算云平台可以为大模型、生成式 AI 提供 IaaS、PaaS、SaaS 等多个层面的云服务，同时满足 AI 训练和推理服务两种业务需求。智算云可以以公有云或私有云等各种形式呈现，但由于其专门为 AI/HPC 应用设计，在整体架构上有自己的独到之处，总体架构如图1.6所示：

基础设施层多采用 CPU+DPU+GPU 3U 一体异构算力架构，提供通用算力和智算算力，满足多种算力需求。其中 CPU 多采用 X86 和 ARM 两种处理器架构，LoongArch，Alpha 等架构也逐渐开始进入智算算力视野。GPU 的引入可以良好的支持人工智能的推理和训练业务，满足智算业务通用性需求。网络层硬件采用 DPU 系列产品，通过将智算的计算、存储、网络、安全、管理等卸载到 DPU 硬件层处理，实现在超高带宽、超

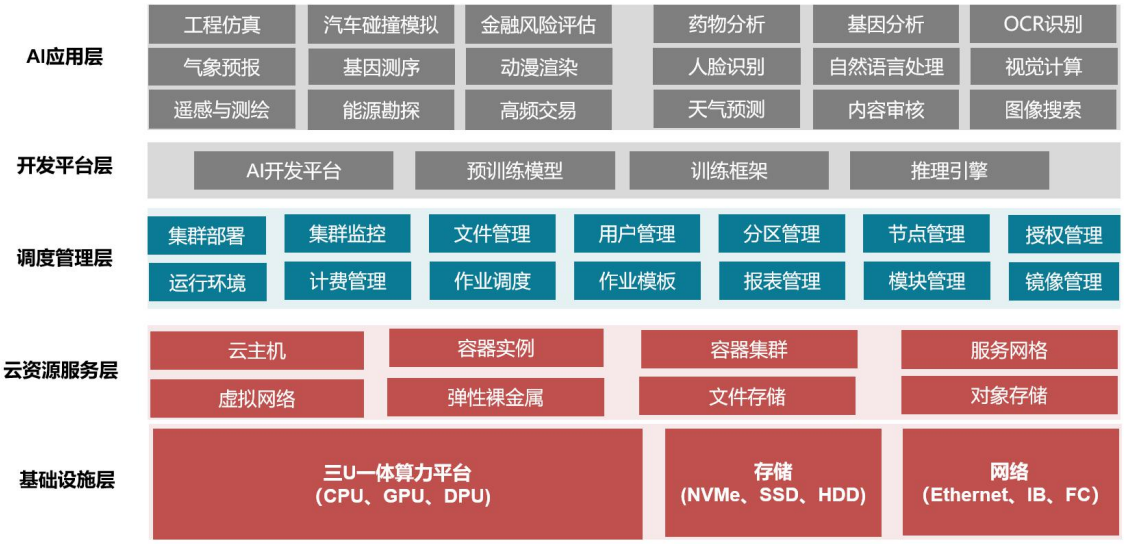

图 1.6: 智算云架构图

低延迟的网络环境中发挥极致效能，同时 DPU 为多租户智算云业务提供安全隔离保护，良好地支撑了 AI 人工智能的 GDR 和 GDS 场景下的推理和训练业务，保证了智算云平台所有业务及数据安全、稳定、可靠的运行。

云资源服务层提供裸金属服务器、虚拟机、容器、服务网格等各类智算云平台资源服务，大多采用 Kubernetes 的云原生容器化应用全生命周期管理，提供高扩展、高性能容器应用管理服务。采用 IoD 技术可以将容器的基础网络与存储等能力卸载到 DPU 硬件上，实现了超高性能的容器云业务环境。

调度管理层提供弹性灵活的云原生资源管理和调度能力，IoD 技术可以为云原生智算资源管理和调度平台提供 GPU 池化和对容器、虚拟机、裸机的统一管理调度能力，配合 AI 调度管理平台，实现 DPU 、GPU、CPU 资源和裸金属、虚拟机、容器等各类云服务的智能负载调用，为智算各个业务场景合理调度、分配资源，实现资源最大化利用和灵活调度。

开发平台层为各类 AI 业务应用提供开发框架、预训练框架、训练框架、推理引擎等基础服务。

AI 应用层是指智算云平台上承载的各种智算应用服务。

1.4.5 “电光火石”的低时延云

随着技术升级和业务创新，众多行业对网络时延也提出了更高的要求，越来越多的行业对时延要求从毫秒下降到微秒，比如证券领域的极速交易场景对时延要求下探到

亚微秒级，工业控制协议中约 15% 核心生产环节要求时延不高于 1ms，自动驾驶场景下远程取消操作的要求时延不高于 3ms。更低的时延，在产业中，意味着更高的收益、更高的精度、更高的安全性。因此，各个行业对于时延性能的追求和探索永无止境。由于传统云平台时延性能不能满足核心交易等时延敏感业务要求，因此现阶段，金融领域使用的低时延网络和服务器，普遍使用裸金属部署，存在诸多痛点，包括物理资源容量受限、资源使用成本高、交付效率难以满足业务调整、虚拟化带来的时延损耗大。

低时延云平台是构建在超低时延网络服务器集群上，集成了超低时延技术，具备超低时延、自主可控、业务深度调优、高性能异构加速等核心特性，为金融极速交易、人工智能、工业控制、边缘计算等时延敏感场景提供超低时延的云计算服务。

通过 IoD 技术体系的异构算力管理能力，将低时延传输能力纳入云平台管理与调度，可以更好的支撑低时延云场景的业务需求。

第 2 章 云计算业务模型分析

2.1 当前主流云计算体系结构

当前主流云计算体系结构是一个多层次、高度集成的架构，旨在提供弹性的资源分配、自动化管理以及丰富的服务选项。这一架构通常可以从四个核心层面来描述：硬件部分、基础软件、云管平台（云管理平台）以及业务服务。下面是对这四个方面的详细说明：

2.1.1 硬件部分

硬件部分构成了云计算的物理基础，包括服务器、存储设备、网络设备（交换机、路由器等）以及可能的专用硬件（如 GPU 服务器、FPGA 加速器等）。

在现代云计算数据中心中，服务器通常采用高性能、高密度的设计，支持大规模横向扩展。

- 高密度服务器：为了提高数据中心的空间利用率，现代云数据中心倾向于使用高密度服务器，如刀片服务器，它们能在有限的空间内提供更多的计算能力。
- 异构计算：除了传统的 CPU 服务器，云计算平台还会部署 GPU 服务器、FPGA 服务器和 TPU（针对特定工作负载，如机器学习、图形渲染和高性能计算）来提升特定应用的处理效率。

存储设备涉及分布式存储系统，以提供高可用性和数据冗余。

- 分布式存储系统：如 Ceph、GlusterFS、HDFS，通过多副本、纠删码等机制确保数据的高可用性和容错性。
- 全闪存阵列：为了提升 I/O 性能，越来越多的云服务采用全闪存存储解决方案，提供低延迟、高吞吐的存储服务。

网络方面，SDN（Software-Defined Networking，软件定义网络）技术被广泛应用，实现网络资源的灵活配置和管理。

- SDN（软件定义网络）：通过分离控制平面和数据平面，实现网络资源的灵活配置和自动化管理，如 OpenFlow 协议和控制器如 OpenDaylight。
- NFV（网络功能虚拟化）：将传统网络设备功能（如防火墙、负载均衡器）转变为软件形态，运行在通用服务器上，提高灵活性和可扩展性。

2.1.2 基础软件

基础软件层主要包括操作系统（如 Linux）、虚拟化技术（如 KVM、Xen、Hyper-V）、容器技术（如 Docker）、以及分布式系统基础组件（如分布式文件系统、数据库、消息队列等）。虚拟化技术允许在单个物理服务器上运行多个虚拟机，提高硬件资源的利用率。容器技术则进一步提升了资源利用效率和应用部署的灵活性。此外，微服务架构和无服务器计算（Serverless Computing）也是现代云基础软件的重要组成部分，它们促进了服务的快速开发和部署。

1. 操作系统
   - 轻量化 OS：针对云环境优化的轻量级操作系统，如 CoreOS、ContainerOS，减少不必要的服务，更适合容器运行环境。
   - 容器运行时：Docker、rkt 等，提供容器的创建、运行、管理和镜像分发能力。
2. 虚拟化技术
   - Type 1 Hypervisor：直接运行在硬件之上的虚拟化层，如 Xen、KVM，提供接近物理机的性能。
   - Type 2 Hypervisor：运行在操作系统之上的虚拟化层，如 VirtualBox、VMware Workstation，更适合桌面虚拟化和开发测试环境。
3. 分布式系统基础组件
   - 分布式数据库：如 Cassandra、MongoDB，设计用于处理大规模数据集，提供高可用性和水平扩展性。
   - 消息队列：如 Kafka、RabbitMQ，用于解耦服务、异步处理和高并发处理。

2.1.3 云管平台

云管理平台是云计算体系的核心，负责资源的自动化管理、监控、计费、安全控制以及服务交付。这一层常见的平台包括 OpenStack、AWS CloudFormation、Azure Resource Manager、Google Cloud Console 等。云管平台提供了一个统一的界面或 API，使得用户可以轻松创建、配置、监控和销毁各种云资源，如虚拟机、容器、数据库实例、负载均衡器等。此外，它还负责资源调度、故障恢复、资源计量计费等功能，确保服务的高可用性和经济性。

1. 资源调度与编排
   - Kubernetes：目前最流行的容器编排平台，负责容器应用的部署、扩展、维护。

- OpenStack：开源云平台软件，提供 IaaS 服务，包括计算、存储、网络等资源的管理。

2. 监控与日志管理
   - Prometheus + Grafana：Prometheus 负责收集指标，Grafana 用于数据可视化，共同实现全面的监控。
   - ELK Stack：Elasticsearch、Logstash、Kibana 组成的日志分析平台，用于日志收集、分析、展示。
3. 安全与合规
   - IAM（Identity and Access Management）：管理用户身份和权限，确保只有授权用户能访问相应资源。
   - 安全组与网络 ACL：提供网络层面的安全控制，限制进出流量，增强云环境的安全性。

2.1.4 业务服务

业务服务层直接面向最终用户或开发者，提供一系列可直接消费的云服务，包括但不限于：

- 计算服务：如弹性计算服务（EC2, ECS）、裸金属服务（BMS）函数即服务（Lambda, Azure Functions）。
- 存储服务：对象存储（S3, Azure Blob Storage）、块存储、文件存储等。
- 数据库服务：关系型数据库服务（RDS）、NoSQL 数据库、数据仓库等。
- 网络服务：VPC、EIP、VPN、负载均衡、CDN、DNS 管理等。
- 安全服务：身份与访问管理（IAM）、防火墙、安全组、数据加密服务等。
- 开发者服务：持续集成/持续部署（CI/CD）、API 网关、消息队列等。
- 数据分析与 AI 服务：大数据处理、机器学习平台、数据湖等。

这些服务旨在降低企业构建和运行应用程序的技术门槛，加速产品上市时间，同时提供按需付费的灵活性，帮助企业根据实际需求动态调整资源规模。

2.2 计算业务分析

云计算基础设施的多样化为不同规模和需求的企业提供了灵活的选择，其中，裸金属服务器、虚拟机、容器和 GPU 服务器作为四大核心服务形态，各自拥有独特的性能

特点和应用场景，满足了从基础计算到高性能计算、从轻量级应用到大规模数据处理的广泛需求。

2.2.1 裸金属服务器

性能特点:

- 极致性能与低延迟：裸金属服务器直接运行在物理硬件之上，消除了虚拟化层的开销，提供了接近硬件极限的性能。这使得它们成为对计算性能和低延迟有极高要求应用的理想选择，如高频交易系统、大规模数据库和高性能计算（HPC）场景。
- 资源独享：与虚拟机不同，裸金属服务器的资源（CPU、内存、存储、网络）完全为单一用户所用，避免了资源竞争，确保了性能的稳定性和可预测性，适合对资源隔离性有严格要求的应用。
- 高度可定制与扩展：用户可以根据特定需求选择和配置硬件，如特定型号的 CPU、内存大小、存储类型和网络配置，以及添加 GPU 等特殊硬件，以满足特定应用的优化需求。

衡量指标:

1. CPU 基准测试：使用 SPEC CPU Benchmark Suite 等工具评估处理器的整数和浮点运算性能。
2. 内存带宽测试：通过 Stream Benchmark 测试内存读写速度，反映大块数据操作的效率。
3. 存储 I/O 性能：使用 fio 工具测量磁盘读写速度和 IOPS（每秒输入输出操作），评估存储系统的响应能力。
4. 网络吞吐与延迟：使用 iperf 或 netperf 工具测试网络接口的最大吞吐量和数据包往返时间。

2.2.2 虚拟机

性能特点:

- 资源灵活分配与管理：虚拟机能够在一台物理服务器上创建多个独立的运行环境，每个环境都拥有自己的操作系统、内存、CPU 份额和存储。这使得资源的分配和回收变得非常灵活，适合快速开发和测试环境的搭建。

- 隔离与安全性：虽然不如裸金属服务器，但虚拟化层提供了基本的隔离能力，防止一个虚拟机的崩溃或攻击影响到其他虚拟机，提升了整体环境的安全性。
- 迁移与灾备：虚拟机易于迁移和备份，为业务连续性和灾难恢复提供了便利。

衡量指标:

1. 虚拟化开销评估：比较虚拟机与裸金属服务器在相同工作负载下的资源使用和执行时间，评估虚拟化层引入的性能损耗。
2. 资源调度效率：观察 CPU 的争用率、内存页交换频率，以及在资源紧张时的性能稳定性。
3. 热迁移能力：测试虚拟机在不同物理主机间迁移的速度和业务中断时间，评估云平台的灵活性。
4. 动态资源调整响应：测量增加或减少 vCPU、内存资源时，虚拟机性能的变化情况。

2.2.3 容器

性能特点:

- 轻量级与快速启动：容器共享宿主机的操作系统内核，无需额外的操作系统层，启动速度极快，资源占用小，适合快速部署和扩展微服务架构。
- 高度可移植性：容器镜像标准化，便于跨平台、跨环境部署，提高了开发到生产的效率和一致性。
- 资源利用率高：相比于虚拟机，容器在资源使用上更为高效，能够支持更密集的部署，降低资源成本。

衡量指标:

1. 启动时间：容器从创建到就绪的平均时间，反映容器的快速响应能力。
2. 资源利用率：比较容器与虚拟机在同一工作负载下的 CPU、内存使用效率。
3. 网络性能：容器网络模型（如 Docker bridge、Kubernetes CNI）的吞吐量和延迟，影响服务间的通信效率。
4. 隔离与安全性：通过 cgroups 和 namespace 等机制评估容器间的隔离程度和安全风险。

2.2.4 GPU 服务器

性能特点:

- 并行计算加速：GPU（图形处理器）拥有数千个核心，特别适合执行高度并行的任务，如深度学习训练、大规模数据分析、科学计算和 3D 渲染，相比 CPU 能显著缩短计算时间。
- 高带宽显存：GPU 配备有高带宽内存（如 HBM2、GDDR），适合处理大规模数据集，减少内存访问瓶颈。
- 能效比：在处理特定类型的工作负载时，GPU 相比 CPU 展现出更高的能源效率，有利于降低长期运营成本。

衡量指标:

1. 浮点运算性能：通过 FP32、FP16、INT8 等不同精度的 TensorFLOPS（TFLOPS）衡量 GPU 的计算能力。
2. 显存带宽：衡量 GPU 内存的数据传输速度，对处理大型数据集至关重要。
3. 并行处理效率：以深度学习为例，测量每秒处理图像数量（Images Per Second, IPS）或模型训练时间，评估 GPU 加速效果。
4. 功耗与散热：考虑 GPU 服务器在高负载下的能源消耗和散热需求，评估其在数据中心的运维成本。

2.2.5 应用场景与选择策略

在实际应用中，往往会根据业务特点选择使用不同的云计算服务：

- 裸金属服务器：适合对性能和安全性有极端要求的场景，如核心数据库、大规模数据分析、金融交易系统、高性能计算等。
- 虚拟机：适合需要灵活资源分配、快速部署和低成本试错的场景，如开发测试环境、网站托管、轻量级应用部署。
- 容器：适合微服务架构、持续集成/持续部署（CI/CD）流程、快速迭代的软件开发，以及需要快速扩展和高密度部署的场景。
- GPU 服务器：针对深度学习、科学计算、3D 图形渲染、大数据分析等高度并行计算需求，以及对计算效率和能效比有特殊要求的应用。

综上所述，选择合适的云服务形态需综合考虑业务需求、性能要求、成本预算和运维能力。随着技术的不断进步，如智能调度、自动化运维、云原生技术的发展，未来云服务将更加灵活、高效，更好地服务于多样化的业务场景。

2.3 网络业务分析

随着云计算的飞速发展，云计算网络技术也在不断演进，初始的传统三层网络逐步被更适合大二层网络的 Spine-Leaf 架构替代。经典的大二层网络经历设备虚拟化方案、L2 over L3 方案和 Overlay 方案后，现在已经演进为 VPC（Virtual Private Cloud）网络。从核心技术来看，云计算网络最看重对网络编址和网络性能的优化，以满足云计算对高性能和灵活性的要求，这其中 SDN（Software Defined Network）发挥着至关重要的作用。

SDN 技术是对传统网络架构的一次重构，其核心思想是通过控制面与数据面的分离，将网络的管理权限交给控制层的控制器软件，再通过南向协议通道，统一下达指令给数据转发层设备，网络控制与数据转发的充分解耦，实现云计算网络控制的集中化、自动化和可编程性。

SDN 网络架构通常包含三个关键层次：

- 应用层：SDN 的最上层，承载着云上各种网络应用和服务，允许开发人员根据具体业务需求创建自定义的网络应用程序。同时提供了与控制层交互的 API 接口，使得应用能够向 SDN 网络发出指令、获取网络状态信息以及实时调整网络行为。
- 控制层：SDN 的核心，包含 SDN Controller，负责收集全局网络视图，将来自应用层的需求转化为具体的网络配置指令，并将这些指令传递给底层的网络设备。SDN Controller 通过北向接口与应用层通信，通过南向接口与基础设施层通信，实现对网络行为的集中控制。
- 基础设施层：包含了网络中的实际设备，如交换机、路由器等，这些设备既可以是物理设备，也可以是虚拟设备。基础设施层负责执行来自控制层的指令，将其翻译为底层设备的配置，实现数据的实际传输和处理。

云计算网络重点实现 SDN 的控制层和基础设施层，控制器、网关和虚拟交换机协同工作，构成一个集成的云网络系统。其中网关和虚拟交换机作为承载网络传输的基础设施，其性能很大程度决定了整个云网络的整体性能。

值得特别指出的是网关作为连接不同网络环境的关键组件，其功能和形态也经历了从基础的数据转发到软件定义与高性能网关的转变，以适应云环境的复杂性和动态性。在云网络的早期阶段，网关主要基于传统的硬件设备，负责不同网络之间的数据包转发、协议转换和安全控制。这些设备固定功能，配置复杂，且扩展性和灵活性有限，难以适应云计算环境的快速变化和动态需求。随着 NFV（Network Function Virtualization）概念的提出，网关开始向虚拟化方向发展。NFV 允许将传统网关的功能（如路由、防

火墙、负载均衡等）以软件的形式运行在通用服务器上，从而提高了资源利用率、降低了硬件成本，并增强了灵活性和可扩展性。第一代 NFV 网关虽然实现了功能的虚拟化，但在性能、可管理性和集成度上仍有待提升。SDN 技术的引入，使得网络控制层面与数据转发层面分离，网关技术也随之升级。SDN 集成网关能够与 SDN 控制器配合，实现网络策略的集中控制和动态配置，提高了网络的自动化水平和响应速度。SDN 网关不仅能够执行传统网关的功能，还能动态适应网络拓扑变化，优化数据路径，为云环境提供了更高的灵活性和可编程性。

随着网络流量的持续增长，网络数据包处理效率的问题开始凸显，对应的云计算网络技术也在持续演进。虚拟交换机经过 Linux 内核交换机、用户态 DPDK 交换机阶段，现在通过 DPU、智能网卡等设备的硬件卸载能力，来进一步提升虚拟交换能力。传统虚拟交换机的大部分数据包处理任务（如封包转发、VLAN 标记/去标记、流量控制等）依赖于主机 CPU。随着虚拟机数量的增长，网络流量的增加，CPU 负担加重，成为性能瓶颈。DPU 通过将这些网络处理任务从 CPU 卸载到专门的硬件设备上，显著减轻了 CPU 的压力，提高了整体系统性能和效率。另外 DPU 等设备通常配备高性能的网络接口和加速引擎，能够以硬件加速的方式处理网络数据包，相比软件实现，其处理速度更快，延迟更低，吞吐量更高。

2.4 存储业务分析

在云计算业务模型中，存储性能要求在不同场景下扮演着关键角色。大数据分析需要高吞吐量、低延迟和良好的扩展性；人工智能应用则侧重于高速数据读写和一致性；在线交易场景要求低延迟、高并发读写和数据可靠性；而多媒体存储与流媒体场景则需要高带宽、低延迟和数据稳定性。这些场景下的存储性能需求突显了各自的重点，包括吞吐量、延迟、可靠性和扩展性等关键指标。因此，在设计和优化存储系统时，必须针对特定业务场景的需求进行有针对性的考量，以确保系统能够满足不同场景下的性能要求。

- 大数据分析场景：在大数据分析场景下，存储系统需要具备高吞吐量和低延迟的特性，以支持快速的数据读取和处理。同时，大数据分析通常涉及大规模数据的并行处理，因此存储系统需要具备良好的扩展性和并发处理能力。
- 人工智能应用场景：对于人工智能应用，存储系统需要具备高速的数据读取和写入能力，以支持大规模的数据训练和推理过程。低延迟和高 IOPS 对于实时推理

和训练任务至关重要，同时数据的一致性和可靠性也是关键指标。

- 在线交易场景：在线交易场景对存储系统的响应速度和数据一致性要求较高。存储系统需要具备低延迟、高并发读写能力，以确保交易数据的实时性和准确性。同时，数据的持久性和可靠性也是关键考量因素。
- 多媒体存储与流媒体场景：在多媒体存储和流媒体场景下，存储系统需要具备高带宽、高吞吐量和低延迟的特性，以支持大规模的多媒体数据的存储和传输。同时，数据的稳定性和可靠性对于保障媒体数据的完整性至关重要。

2.5 安全业务分析

随着云计算技术的快速发展，高性能云计算对网络架构和弹性提出了更高要求，基础设施暴露了更大的攻击面，业务数据更加集中，数据价值越来越高，黑客攻击越来越多，企业面临着更加复杂的安全威胁。高性能云计算改变了传统数据中心的网络和业务模型，同时给网络安全建设带来了巨大的挑战，需要高性能、弹性的分布式安全防护体系。

传统安全方案采取集中式部署方式，可以有效防御南北向网络攻击，无法应对东西向流量的网络攻击，无论是从成本还是机房空间等其他方面考虑无法部署到计算环境中。攻击者一旦进入网络，通过内部网络横向发起扩散攻击，入侵更多的主机，扩散范围包括 VPC 之间和 VPC 内部各个主机、虚机、容器等。为了解决云计算东西向网络攻击，客户通常采用基于网络引流模式的安全资源池，或者采用基于代理模式的独立虚拟防火墙进行防护，这两种旁路安全防护方式，在防护效率、复杂度、覆盖度等多方面面临如下挑战：

- 网络引流路径长，产生额外开销，防护效率低。
- 需要操作交换机，网络操作复杂，出错风险增加。
- 无法对跨虚拟机、容器的流量进行隔离防护。
- 服务器处理防火墙、加解密等安全功能性能低，尤其是国产化服务器平台。
- 需要增加额外的服务器，占用机房物理空间，综合成本增加。

通过上述安全性能分析，针对薄弱的安全计算环境，云计算需要一种新的网络安全防护体系，这种防护体系可以面向租户进行贴身防护，满足云内虚机/容器安全隔离，具有高安全、高性能、分布式部署能力，可以与第三方安全软件控制面进行适配，具有弹性扩展的能力。安全防护体系不应该依赖于业务服务器算力，不影响业务应用的效率，

不增加额外的机房空间。

2.6 平台服务业务分析

2.6.1 数据库

通常我们希望高性能数据库应具备高并发处理能力、低延迟响应、高 TP/AP 性能、低资源使用率、高可靠性、可扩展性、数据安全等关键特性。在云计算业务场景下，分布式数据库的部署通常具备更好的资源弹性，以及采用“存算分离”的部署方式，相较于传统的“存算一体”架构，这对网络提出了更高的性能要求。

分布式数据库的业务场景分为计算密集型、IO 密集型和网络密集型，分别容易造成计算瓶颈、IO 瓶颈和网络瓶颈，传统方案中，一般都是通过增加相对应的资源来解决相应的业务瓶颈，这无疑增加了相应的建设成本。

继续细分业务场景，我们简单分为 OLTP 场景和 OLAP 场景，OLTP 场景下，我们对数据库的高并发处理能力、低延迟响应、高 TP 性能提出了较高的要求。OLAP 场景下，我们则对高 AP 性能、低资源使用率提出了较高的要求。

再继续下沉到 IaaS 基础能力上，高并发处理能力，需要高计算处理能力、高带宽、高 IO 性能（高带宽/高 IOPS）；低延迟响应需要网络和 IO 的低延迟；资源使用，一方面依赖数据库本身的功能实现，以及任务本身，另一方面也取决于 IaaS 整体架构对于资源使用的优化能力。

2.6.2 中间件

云计算场景下，中间件服务通常都是分布式的部署方式，服务状态或持久化数据通常存储于云盘或分布式文件系统中，同样需求更高的网络和 IO 性能。高性能中间件应具备高吞吐量、低延迟、高可用性、可扩展性、灵活性、易用性、容错性等多个特性。

从有无状态或持久化数据的角度上，我们可简单将中间件分为有状态中间件和无状态中间件。有状态中间件，由于通常有大量的数据从数据盘进行读写，因此对 IO 性能需求较高；无状态中间件没有数据落盘，通常更偏向网络型应用场景，更依赖网络性能。同时，两者一般都有低延迟的性能需求，因此需要依赖网络和 IO 的低延迟。

2.6.3 服务治理

伴随着云原生技术的发展，微服务架构已经成为现代软件开发的主流架构。在微服务架构中，由于原来的单体应用被拆分为众多的微服务组件，一次业务请求通常需要多级链式调用才能完成处理。因此，通过服务治理技术在众多微服务容器间完成业务调度的工作就显得尤为关键。当前许多开发框架均能提供服务治理的技术栈并被广泛适用，如 SpringCloud、Dubbo 等。但是一般都存在 Setup 困难，绑定开发语言，增加额外组件与业务逻辑侵入等问题，需要开发者深入了解所使用的框架特性与编程模式，才能取得较好的服务治理效果。相比之下，Service Mesh 技术具有明显的优势，业务开发人员只需关注自己的业务逻辑，其余像服务注册与发现、动态路由与业务追踪等功能，全部通过 Service Mesh 体系的边车容器完成。但是，Service Mesh 体系中边车容器的引入，造成了整体通信链路的性能降低，这又成为了新的痛点。

因此，在云计算场景下，高性能服务治理是在微服务时代必须要解决的核心问题，一方面要能够提供兼具高吞吐与低时延性能的方案，另一方面也需要尽量减少资源消耗。

第 3 章 高性能云计算基础设施建设路径

3.1 通用算力技术分析

3.1.1 CPU 的计算能力发展历程

CPU 计算能力的发展大致可分为以下几个阶段:

1. 早期发展 (1971-2000): 从 1971 年英特尔推出首款商用微处理器 4004 开始,CPU 性能主要通过提高时钟频率和改进微架构来提升。这个时期遵循摩尔定律, 晶体管数量大约每 18-24 个月翻一番。
2. 多核时代 (2000-2010): 单核心频率提升遇到瓶颈后, 处理器厂商转向多核设计。这一时期出现了双核、四核等多核心处理器, 通过并行计算提高性能。
3. 异构计算与专用处理器 (2010-2015): 除了继续提高核心数量, 处理器开始整合 GPU 等专用单元, 形成异构计算架构。同时, 针对特定应用如 AI 的专用处理器开始出现。
4. 近 10 年的发展 (2015-2024):
    - 核心数量持续增加: 消费级 CPU 从四核发展到现在的 16 核甚至更多, 服务器 CPU 则已达到 64 核甚至 128 核。
    - 制程工艺不断进步: 从 22nm、14nm, 到现在主流的 7nm、5nm, 甚至已有 3nm 制程的 CPU 问世。更先进的制程带来了更高的能效比和更强的性能。
    - 架构创新: 各大厂商不断推出新的 CPU 架构, 如英特尔的 Skylake、AMD 的 Zen 等, 通过优化指令集、缓存结构等提升性能。
    - 异构计算:CPU 在同一芯片上越来越多地集成了 GPU、AI 加速器和安全处理器等专用组件。
    - 3D 封装技术: 如 AMD 的 3D V-Cache 技术, 通过堆叠更大容量的缓存来提升性能。
    - 大小核设计: 借鉴移动处理器的设计理念, 将高性能核心和高能效核心结合, 如英特尔的 Alder Lake 架构。
    - AI 加速: 集成专门的 AI 处理单元, 以应对日益增长的 AI 计算需求。
    - 安全性增强: 加入更多硬件级安全特性, 如英特尔的 SGX、AMD 的 SEV 等。

在具体性能方面, 以消费级 CPU 为例,2014 年的顶级 CPU(如英特尔 i7-4790K) 单核性能约为 2500 points(Passmark 单核测试), 多核性能约为 10000 points。而 2024 年的顶级 CPU(如 AMD Ryzen 9 7950X) 单核性能已达到 4000+ points, 多核性能超过 50000 points,10 年间性能提升了约 5 倍。

服务器 CPU 的进步更为显著。2014 年的高端服务器 CPU(如 Intel Xeon E5-2699 v3) 拥有 18 核心,36 线程, 多核性能约为 24000 points。而 2024 年的顶级服务器 CPU(如 AMD EPYC 9654) 拥有 96 核心,192 线程, 多核性能超过 150000 points, 性能提升超过 6 倍。

除了原始计算能力的提升, 现代 CPU 还在功耗效率、特定任务加速 (如 AI、加密等)、安全性等方面取得了巨大进步。例如, 支持 AVX-512 指令集的 CPU 在某些特定计算任务上可以获得数倍于传统指令集的性能提升。

然而，CPU 的发展也遇到了一些瓶颈, 主要包括以下几个方面:

1. 摩尔定律放缓: 传统上, 晶体管数量每 18-24 个月翻倍的摩尔定律已经难以为继。制程工艺进入纳米级后, 面临量子效应等物理极限。
2. 功耗墙: 随着晶体管密度的提高, 单位面积的功耗不断增加, 散热成为严重问题。这限制了时钟频率的进一步提升。
3. 存储墙: 内存访问速度的提升远落后于 CPU 速度, 造成 CPU 经常处于等待数据的状态, 影响整体性能。
4. 指令级并行 (ILP) 瓶颈: 单核中能够并行执行的指令数量已接近极限, 难以通过增加流水线深度等方法获得显著性能提升。
5. 多核扩展性问题: 增加核心数量面临软件并行化的挑战, 以及核心间通信和同步开销增加的问题。
6. 制造成本上升: 先进制程研发和生产线建设成本呈指数级增长, 限制了性能提升的经济可行性。

展望未来,CPU 的发展可能会继续朝着以下方向前进: 更先进的制程 (如 2nm、1nm)、更多的核心数、更高效的异构计算、更先进的 3D 封装技术, 以及可能的新型计算范式 (如量子计算辅助单元) 等。

3.1.2 云计算卸载技术为 CPU 算力提升带来的优势

从技术角度分析，云计算卸载技术可以为 CPU 算力提升带来显著优势，主要体现在卸载基础服务释放 CPU 资源以及卸载后的性能提升两个方面。

1. 卸载基础服务，释放 CPU 资源 卸载技术将网络、存储、安全、管理等基础服务

从主 CPU 转移到专门的硬件 DPU 上，可以带来如下优势：

- 减少 CPU 负载：

  网络处理：如 TCP/IP 协议栈处理、数据包分类、负载均衡等耗费大量 CPU 周期的任务被卸载。

  存储操作：如数据压缩、加密、RAID 计算等 I/O 密集型任务被转移。

  安全功能：如加密/解密、防火墙、入侵检测等计算密集型安全任务被卸载。

  虚拟化管理：Hypervisor 的部分功能，如设备模拟、内存管理等被转移。

- 提高 CPU 效率：

  减少上下文切换：基础服务由专门硬件处理，减少了 CPU 频繁在用户态和内核态之间切换。

  降低中断处理开销：大量 I/O 中断被卸载硬件直接处理，减轻了 CPU 的中断处理负担。

- 优化资源分配：

  更多 CPU 资源可以分配给核心业务应用，提高应用性能和响应速度。

  支持更高的虚拟机或容器密度，提升整体计算资源利用率。

- 简化 CPU 调度：

  减少了 CPU 需要处理的任务类型，简化了调度算法，提高调度效率。

2. 基础服务卸载后性能提升

   将基础服务卸载到专门的硬件不仅释放了 CPU 资源，还能在性能上超越纯 CPU 处理：

   - 硬件加速：

     DPU 可以为特定任务提供高度优化的处理能力。网络处理可以实现线速性能，大幅超越 CPU 软件处理。

   - 低延迟处理：

     卸载硬件直接与 I/O 设备交互，绕过了操作系统内核，显著降低延迟。例如，RDMA 技术可以实现微秒级的网络延迟。

   - 一致性能：

     卸载处理不受主机 CPU 负载波动影响，提供更稳定、可预测的性能。特别适合需要 QoS 保证的关键业务应用。

   - 能效提升：

     专用硬件通常比通用 CPU 更节能，特别是在处理特定任务时。整体系统功耗

降低，提高了数据中心的能源效率。

- 安全性增强：

  安全功能在独立硬件上运行，提供了更好的隔离性和防御能力。例如，加密操作在专用硬件中进行，降低了密钥泄露的风险。

总结来说，云计算卸载技术不仅释放了宝贵的 CPU 资源，还通过专门的硬件加速实现了更高的处理性能。这种方案在提高系统整体效率、降低延迟、增强安全性和改善可扩展性等方面都表现出明显优势。随着卸载技术的不断发展，我们可以期待看到更多创新应用。

3.1.3 IoD 技术为 Hypervisor 卸载提供最佳支撑

在现代云计算环境中，虚拟化技术扮演着至关重要的角色。其中，计算系统虚拟化的核心通常是基于 KVM-QEMU 架构的 Hypervisor 系统。这种虚拟层平台通过 Hypervisor 为上层业务提供了灵活且高效的虚拟机环境。

然而，传统的 Hypervisor 系统存在一个显著的缺点：它需要占用主机相当大比例（通常为 10% ~ 20%）的 CPU 计算资源。这种资源占用不仅降低了系统的整体效率，还限制了可用于实际业务的计算能力。为了解决这个问题，IoD 技术了一种创新的解决方案，主要指的是将 Hypervisor 的部分功能卸载到 DPU 上。

这种卸载策略的核心思想是将 Hypervisor 分为前端和后端两个部分：

1. 前端（运行在主机侧）：
   - 维护逻辑 CPU 的上下文同步
   - 与内核中的 KVM 模块交互，完成虚拟机 vCPU 和内存的分配
   - 与后端的设备模型交互，确定虚拟机 IO 和内存的映射关系
2. 后端（运行在 DPU 上）：
   - 完成虚拟设备的初始化
   - 在虚拟机生命周期的各个阶段（如启动、关闭）与前端协同，管理资源的分配和释放
   - 与 Libvirt 交互，实现虚拟机的全生命周期管理

如图3.1所示，这种架构设计带来的好处是显著的：

- Hypervisor 预留的计算资源可以降低到接近” 零”
- 提高了系统的整体效率和性能

为了进一步优化主机 CPU 资源的利用率，可以考虑对 Host OS 进行精简：

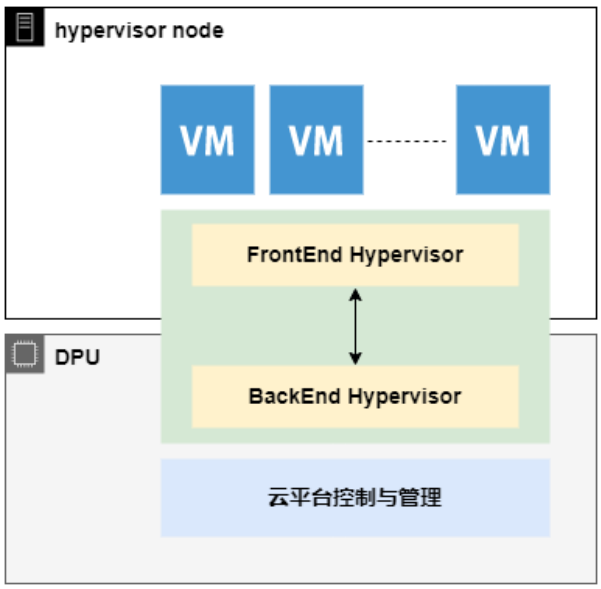

图 3.1: IoD 技术 Hypervisor 卸载方案

- 大部分 IO 操作都由 DPU 直接传递给虚拟机，Host OS 几乎不需要管理 IO 硬件设备
- Host OS 的主要任务简化为支持运行虚拟机 vCPU 线程和管理服务代理程序
- 采用更为精简的初始化系统，进一步减少 Host OS 的资源占用

这种优化策略的最终目标是实现主机 CPU 资源占用接近” 零” 的理想状态。此外，这种架构还为未来的发展提供了更多可能性：

- 更好的资源隔离：DPU 可以提供更强的安全隔离，减少潜在的安全风险
- 灵活的资源调度：可以根据负载动态调整 DPU 和主机之间的任务分配
- 简化管理：集中化的 DPU 管理可以简化整个数据中心的运维工作
- 性能优化：通过专门的硬件加速某些虚拟化功能，可以进一步提升性能

目前阶段，在 IoD 技术体系中，采用的方式是在服务器侧运行一组轻量级组件，一方面响应 DPU 的业务事件，辅助完成与 KVM、LXC 等系统交互，实现云计算业务调度。另一方面辅助将服务器侧文件系统透传给 DPU，帮助下沉的云管系统完成对服务器侧的业务监控。通过这种方式，可以满足云业务平台下沉 DPU 的功能需求。此方法的优势是可以用最小的改造成本完成业务卸载，劣势是牺牲了部分处理性能。

长远来看，更具优势的方案是通过对 Linux 内核的扩展来辅助完成业务下沉功能，建议通过增加内核线程来对 DPU 的请求快速响应，另外通过 eBPF 机制完成对服务器操作系统事件的监控。这种方案可以提供最优的性能，但是需要对 Linux 内核做出较大的改动，相信随着 DPU 技术规范的完善，后一种方案会快速成熟起来。

总的来说，基于 DPU 的 Hypervisor 卸载技术代表了虚拟化技术的一个重要发展方向，有望在提高资源利用率、改善性能和简化管理等方面带来显著的进步。

3.2 智算算力技术分析

3.2.1 GPU 的计算能力发展历程

追溯 GPU 的历史，要从图形显示控制器说起。世界上第一台个人电脑 IBM5150 于 1981 年由 IBM 公司发布，这台 PC 搭载了黑白显示适配器（monochrome display adapter，MDA）和彩色图形适配器（color graphics adapter，CGA），这便是最早的图形显示控制器。1999 年，NVIDIA 公司在发布其标志性产品 GeForce256 时，首次提出了 GPU 的概念。它整合了硬件变换和光照（transform and lighting，T&L）、立方环境材质贴图和顶点混合、纹理压缩和凹凸映射贴图、双重纹理四像素 256 位渲染引擎等，并且兼容 DirectX 和 OpenGL，被称为世界上第一款 GPU。

2011 年 TESLA GPU 计算卡发布，标志着 NVIDIA 将正式用于计算的 GPU 产品线独立出来。凭借着架构上的优势，GPU 在通用计算及超级计算机领域，逐渐取代 CPU 成为主角。如图3.2总结了 GPU 技术的发展历程。

时间	80年代	80年代末	90年代初	90年代后期	2004~2010	2011~至今
类型	图形显示	2D加速	部分3D加速	固定管线	统一渲染	通用计算
相关标准	CGA,VGA	GDI,DirectFB	OpenGL(1.1~4.1)			CUDA, OpenCL(1.2~2.0)
代表产品	IBM 5150	86C911	Glint200SX	GeForce256	G80	Tesla
基本特征	光栅生成器	2D图元加速	硬件T&L	Shader功能固定	多功能shader	与图形处理无关的科学计算

图 3.2: GPU 发展历程

GPU 的并行处理结构非常适合人工智能计算，但传统的基于流处理器的 GPU，其流处理器一般只能处理 FP32/FP64 等精度的运算，而 AI 计算的精度要求往往不高，INT4/INT8/FP16 往往可满足绝大部分 AI 计算应用。针对 AI 应用，NVIDIA 设计了专用的 Tensor Core 用于 AI 计算，支持 INT4/INT8/FP16 等不同精度计算，RTX 2080 集成了 544 个 Tensor Core，INT4 计算能力可达 455 TOPS。

基于 NVIDIA GPU 的 AI 应用绝大多数情况下应用在服务器端、云端，基于 GPU 的 AI 计算往往具有更好的灵活性和通用性，在数据中心、云端等环境下具有更广泛的适用性。与之相对应的，在分布式应用领域 AI 计算更倾向于独立的面向特定应用领域的专用芯片，而不依赖于 GPU，如手机、平板等移动端 SOC 都集成了专用的 NPU IP。

过去 8 年英伟达 GPU 算力的发展如图3.3所示：

Year	2016	2018	2020	2022	2024
GPU	Pascal	Volta	Ampere	Hopper	Blackwell
Type	P100	V100	A100	H100	B100
Peak Teraflops	19	130	620	4,000	20,000
Training And Inference Precision	FP16	FP16	FP16	FP8	FP4

图 3.3: Nvidia 产品演进

八年中 GPU 性能却提高了 1,000 多倍。现在可以使用 Blackwell 系统在十天左右的时间内训练出具有 1.8 万亿个参数的大型模型，比如 GPT-4。两年前使用最先进的 Hopper GPU 很难在数月内训练出数千亿级参数的模型。

GPU 的未来趋势有 3 个：大规模扩展计算能力的高性能计算（GPGPU）、人工智能计算（AIGPU）、更加逼真的图形展现（光线追踪 Ray Tracing GPU)。

3.2.2 GPU 算力提升带来与网络吞吐的矛盾现状

由于 GPU 算力提升与网络带宽发展的不匹配，在实际使用中，引发了诸多矛盾点，主要体现在以下几个方面：

1. 带宽限制：随着 GPU 算力的增长，其处理数据的速度远超于传统网络接口的数据传输速度。这意味着，即使 GPU 能够迅速完成计算任务，也可能因等待数据从网络或存储系统中传输而处于空闲状态，造成算力浪费。
2. 数据局部性问题：在分布式计算环境中，数据需要在不同节点间传输以供计算。GPU 算力的提升使得数据处理速度加快，但频繁的数据移动会占用大量网络资源，影响网络的总体吞吐量，尤其是在涉及大数据集的应用中。
3. 通信开销增加：在并行计算和分布式训练场景中，GPU 之间的通信成为性能的关键因素。随着 GPU 算力提升与使用 GPU 应用的规模脏张，需要同步和交换的信息量增大，如果没有高效的数据交换机制，网络通信延迟和带宽不足会成为瓶颈。

AI 大模型的爆发给云原生智算基础设施带来巨大的技术挑战，带动智算云底座网络和存储架构向大带宽、低延迟方向演进：

1. 超高算力需求

   GPT3 175B 参数，算力需求 3640PFlop/s-day。据 ARK Invest 预测，GPT-4 参数量最高达 15000 亿个，则 GPT-4 算力需求最高可达 31271 PFlop/s-day。

2. 超大规模组网

   AI 大模型/超大模型训练等智算业务场景需要同时使用数千或数万个 GPU 卡训练，AI 服务器集群规模达到 10 万 +。

3. 超高带宽

   单台 AI 服务器内多块 GPU 卡间通信，千亿参数规模的 AI 模型 AllReduce 通信数据量会达到 100GB+；多台 AI 服务器之间流水线并行、数据并行及张量并行等网络通信带宽需求会达到 100GB+。

4. 超低时延

   大规模智算数据同步，要求网络时延 us 级。以 1750 亿参数规模的 GPT-3 模型训练为例，当动态时延从 10us 提升至 1000us 时，GPU 有效计算时间占比将降低接近 10%，当网络丢包率为千分之一时，GPU 有效计算时间占比将下降 13%。

5. 高性能存储

   自然语言处理模型 GPT-1 到 GPT-3，模型参数规模从 1.17 亿发展到 1750 亿，模型层数从开始的个位数逐步发展到成百上千，需要的预训练数据量也从最初的 5GB 发展到 45TB，原始数据集也达到 PB 级。为满足大模型对存储的性能和容量需求，外置存储进一步升级为“性能型存储 + 容量型存储”。

6. 高效调度

   AI 算力设施虚拟化管理，提升集群利用率；拥塞/故障快速切换，最大程度降低负载不均和抖动，提升计算并行能力；在 AI 算力资源不变的情况下，通过精准的调度控制，实现更高的 AI 算力性能。

3.2.3 无损网络技术为 AI 训练带来的性能提升

AI 应用对网络的需求极为严苛，当前主要通过无损网络（IB、RoCE）承载 RDMA 应用，尤其是通过 GDS、GDR 技术实现 GPU 之间以及 GPU 与后端存储之前的高效互联。这种实现方式能够给智算集群的网络处理性能带来质的飞跃。

1. 提升数据传输速度

   通过采用无损网络技术，如 RoCE、InfiniBand 等，其协议特性相比传统网络，能够实现更低的网络时延和更高的数据吞吐量，这是得 AI 训练过程中数据交换更快，从而加速模型参数的同步，减少训练周期。

2. 减少通信开销

   通过流量控制、拥塞管理、负载均衡等机制，减少数据包丢失、传输错误和重传

所造成的额外开销，是得整个训练过程更加高效。

3. 提高算力资源利用率

   通过 RDMA 网络内核旁路和零拷贝的特性，避免数据传输过程中 CPU 算力的消耗，计算节点可以更长时间地处于活跃计算状态，而不是等待数据传输，从而提升 CPU、GPU 等昂贵计算资源的使用效率。

4. 增强大规模并行计算的稳定性

   在大规模分布式训练中，无损网络能够更好地管理网络拥塞，确保数据包的顺序传输，这对于保持模型训练的收敛性和一致性至关重要。

5. 支持更大模型和数据集

   无损网络技术提高了网络的带宽和效率，使得处理更大模型和更大数据集成为可能，这对于提升 AI 模型的精度和泛化能力是非常关键的。

6. 降低能耗

   由于减少了数据传输的重试和等待时间，无损网络技术在提高效率的同时，也有可能降低数据中心的能耗，符合绿色计算的趋势。

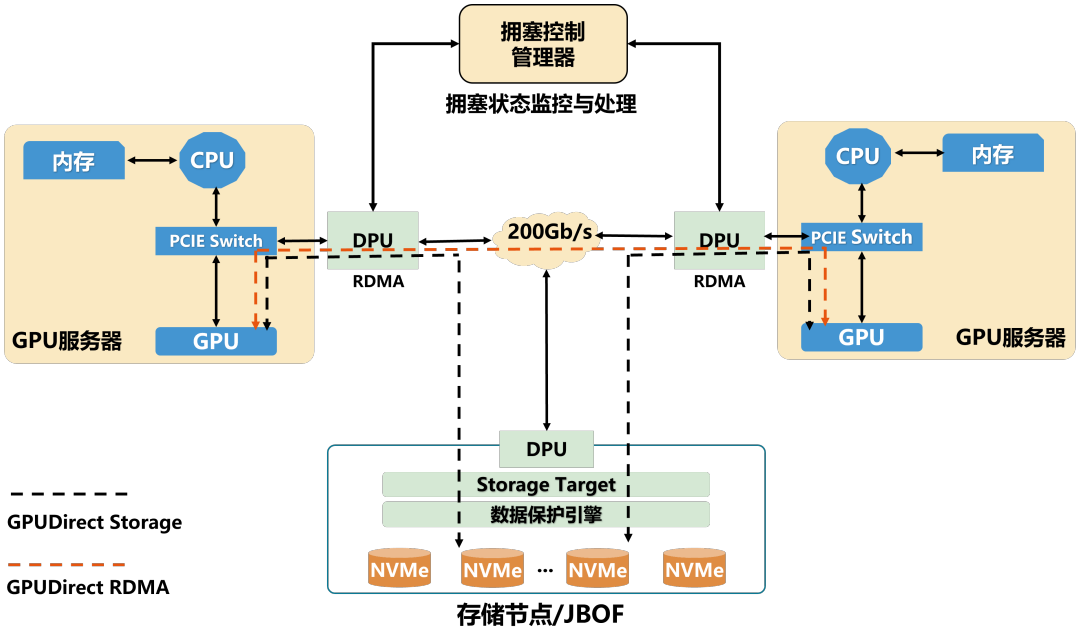

图 3.4: IoD 技术辅助改善拥塞控制算法

在无损网络中，DPU 担任了至关重要的角色，作为网络接入点设备，DPU 实现了 RDMA 协议栈与拥塞处理技术的硬件卸载，大幅提升了网络性能。由于拥塞处理的复杂性，现在业界在重点探索软件定义拥塞控制的新型解决方案，如图3.4所示，IoD 技术可以在 DPU 侧通过软件定义的方式实现网络拥塞状态的监控与拥塞处理控制，将网络处理与上层业务解耦，为整个拥塞处理机制提供更好的灵活性。

3.3 云计算网络技术分析

3.3.1 云计算网络是算力连通的基础

云计算网络是算力连通的基础，在云计算架构中扮演着至关重要的角色。云计算网络不仅连接着云数据中心内部的各种计算、存储和网络资源，还负责将这些资源与云用户、云服务以及其他云数据中心相连接。其核心在于通过网络架构与技术实现资源池化和算力的动态分配。这一过程不仅涉及到计算资源的高效利用，还涵盖了网络资源的优化配置，确保数据和计算任务可以在不同地理区域和不同层级的计算节点间流畅传输和执行。其中关键的技术点包括：

- 网络架构

  网络架构是数字世界的信息流动基础，它定义了数据如何在不同节点间传输、交换和存储。随着数据量的爆炸性增长和计算需求的多元化，传统的网络架构面临前所未有的挑战。一方面，数据的实时性、可靠性和安全性要求网络具备更高的带宽、更低的延迟和更强的稳定性；另一方面，计算任务的复杂性和多样性要求网络能够灵活适应，实现资源的高效分配和智能调度。

- 算力分布

  算力分布则是将计算资源按照需求和场景合理布局的过程。在过去，计算资源往往集中在少数几个大型数据中心，这种集中式布局虽然有利于资源的统一管理和大规模计算，但也带来了高延迟、高能耗和单点故障的风险。随着边缘计算、雾计算和分布式计算的兴起，算力开始向网络边缘和用户侧扩散，形成了多层次、多维度的算力分布格局。这种分布式的算力布局不仅能够降低数据传输的延迟，提高数据处理的效率，还能增强系统的弹性和安全性，实现计算资源的就地可用和智能调度。

- 网络架构与算力分布的协同优化

  网络架构与算力分布的协同优化是构建高效计算生态的关键。一方面，网络架构需要根据算力分布的特点和需求进行设计和优化，如采用软件定义网络（SDN）、网络功能虚拟化（NFV）和网络切片等技术，实现网络资源的动态分配和智能管理，以适应算力分布的灵活性和多样性。另一方面，算力分布也需要充分考虑网络架构的约束和优势，如利用边缘计算节点的低延迟特性，进行实时数据处理和决策支持；利用数据中心的强大算力和海量存储，进行复杂计算和数据分析。

云计算网络通过实现了资源的池化和算力的动态分配，构建了一个高度灵活、可扩

展的算力连通基础。这一基础不仅提高了计算资源的利用率和响应速度，还确保了数据的安全性和隐私保护，是现代云计算体系架构中不可或缺的核心组成部分。

云计算网络通过于高速、低延迟的网络连接和跨域协同技术，支持跨地理区域和跨云环境的计算资源调度和数据传输，实现更高维度的算力流动。多云互联技术使得计算资源可以在不同的云提供商之间灵活调配，增强了算力的连通性和可用性。

3.3.2 云计算网关是算力开放的门户

云计算网关是云内外数据和应用交互的门户，也是优化网络性能、确保数据安全和实现资源高效调度的重要环节。

在高性能云计算场景下，数据的快速传输和处理是基本要求。云计算网关通过采用高速网络接口、智能路由算法、数据压缩技术等，极大提高了数据传输速度，减少了网络延迟，这对于需要处理大量数据的高性能计算任务如大规模数据分析、深度学习模型训练等至关重要。

云计算网关配备高速网络接口，如 100Gbps、200Gbps 甚至更高速率的以太网接口，能够提供极高的数据吞吐量，显著减少数据传输延迟。同时，网关优化传输协议，如使用零拷贝技术减少数据在操作系统内核与用户空间之间的复制，以及采用更高效的网络协议（如 QUIC、HTTP/3）来加速数据传输。

云计算网关内置智能路由算法，能够根据网络状况和数据传输需求动态选择最优路径，避免网络拥堵，减少传输延迟。通过实时监测网络流量，网关能够智能地进行流量控制，平衡负载，确保数据传输的稳定性和高效性。

云计算网关通过数据压缩技术，能够在数据传输前对其进行压缩，减小数据包的大小，从而减少传输时间和带宽消耗。这在处理大量数据或长距离传输时尤为重要，可以显著提高传输效率。此外，网关还优化传输策略，如采用预取技术预测数据需求，提前加载数据到缓存，减少等待时间。

3.3.3 高性能云计算需要网络卸载进行性能提升

高性能云计算旨在提供强大的计算能力以处理大规模数据集、执行复杂的计算任务和满足实时性要求。然而，随着数据量的激增和计算需求的多样化，网络性能成为了制约高性能云计算发展的关键因素之一。

网络卸载作为一种关键的优化技术，通过将数据包处理、加密解密等网络密集型任务从 CPU 卸载至 DPU 等专用硬件，显著减轻 CPU 负担，提升数据处理速度和网络吞

吐量，还通过硬件加速降低了延迟，增强了安全性，从而有效解决了高性能云计算中的网络性能瓶颈，助力其实现更高效、更安全、更具成本效益的网络传输与处理能力。

3.3.3.1 网络卸载的概念与原理

网络卸载（Network Offloading）是一种网络技术，其核心理念是将原本由主机的中央处理器（CPU）执行的网络数据处理任务，转移给 DPU 设备执行。这一过程的主要目的是为了减轻 CPU 的负担，提高网络数据处理的效率和速度，从而提升整个系统的性能。

在传统的网络数据处理流程中，当数据包到达主机时，网卡会将数据包传递给 CPU，CPU 随后在操作系统内核中对数据包进行解析、处理和转发。然而，随着网络流量的不断增大和网络应用的复杂化，这样的处理方式可能会导致 CPU 的资源过度消耗，影响系统的响应速度和整体性能。

网络卸载技术利用 DPU 的计算能力，将数据包的接收、解析、加密/解密、压缩/解压缩、流量控制、负载均衡等网络处理任务从 CPU 上卸载下来。这样一来，CPU 就可以专注于运行应用程序和执行更为复杂的计算任务，而不再需要频繁地处理网络数据包，从而提高了 CPU 的使用效率和系统的整体性能。网络卸载通过优化网络数据处理流程，不仅提高了系统的性能和效率，还增强了安全性，降低了成本，提升了资源分配的灵活性，对于需要处理大量网络数据、实时通信和高并发访问的系统尤为重要，是高性能云计算网络架构中优化性能和资源利用的关键技术之一。

3.3.3.2 网络卸载有助于云网络性能提升

网络卸载通过将特定的任务从 CPU 转移到专用的硬件设备上执行，以提高系统的整体性能和效率，特别是在吞吐量方面。在高性能计算、云计算、网络处理等场景中，硬件卸载能够显著提升数据处理能力和网络传输速度。

网络卸载对整体性能的影响主要表现在以下几个方面：

- 减轻 CPU 负担：

  在服务器和网络设备中，网络数据包的处理如接收、解析、转发、加解密等会占用大量 CPU 资源，导致 CPU 成为系统瓶颈。网络卸载技术将原本需要 CPU 处理的网络任务，转移到专用硬件上执行，极大地释放了 CPU 的计算资源，使 CPU 可以专注于更复杂的计算任务，提高了 CPU 的利用率和系统的整体性能。

- 降低网络延迟：

  DPU 被设计用于加速网络任务，处理速度远超软件实现。数据包在网络间传输时，通过旁路 CPU 的直通技术，减少数据包在 CPU 中的停留时间，提高传输速度。通过网络卸载，数据包在网络中的停留时间大幅减少，显著降低了网络延迟，这对于实时性要求高的应用场景相当关键。

- 增加吞吐量：

  DPU 通常采用并行处理结构，具有高度并行处理能力，能够同时处理多个数据包，对比 CPU 的串行方式有更高处理能力，极大提升包处理效率和网络吞吐量。DPU 内置多个硬件队列，可以将数据包根据其特征分类到不同的队列中，如根据 IP 地址、端口号或协议类型。这种流分类既提升了包处理的并发度，还有助于实现高级 QoS 策略，确保关键数据流获得优先处理。

3.3.3.3 IoD 技术是云计算卸载技术的主要实现方案

IoD 作为一种新方案，有效应对了传统 IaaS 架构面临的挑战。如图3.5所示，该架构将网络虚拟化的任务下沉到 DPU 内，提供了基于硬件的网络设备虚拟化，以及虚拟交换与路由的能力。

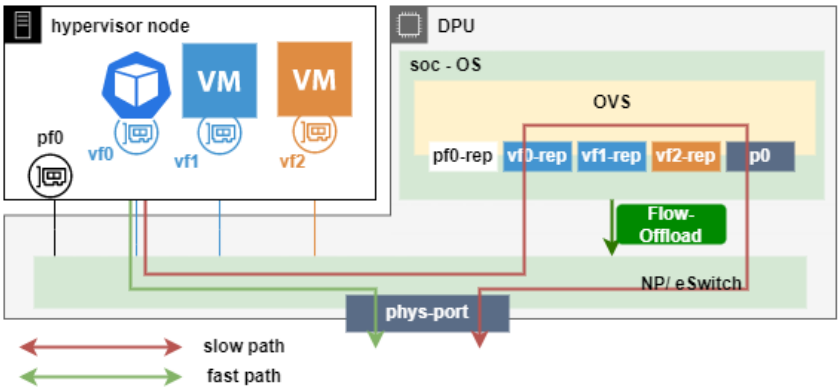

图 3.5: IoD 网络卸载加速原理

在网络设备虚拟化方面，该架构充分利用 DPU 的 SR-IOV 技术，直接在硬件层面提供高性能虚拟网络设备 (VF)，充分满足大规模虚拟机及容器环境的网络需求。另外，通过将原本宿主机承担的 VirtIO 后端处理逻辑下移至 DPU，并融合 vDPA 技术，该架构不仅为虚拟机和容器提供了一个高性能、标准化的 VirtIO 设备接口，还在确保卓越性能的同时，强化了虚拟机热迁移能力，提升了云环境的灵活性与可靠性。

在虚拟交换与路由方面，DPU 内置了高性能网络交换处理单元（NP/eSwitch），不

仅支持用户按需构建灵活多变、功能完备的 SDN 网络架构，而且在虚拟功能（VF）之间乃至 VF 与外部网络的交互中，实现了高效的数据包转发与处理。与 OVS 类似，DPU 对数据包的处理也分为“快路径”（Fast Path）与“慢路径”（Slow Path）：快路径直接经由 DPU 硬件加速完成，确保极低延迟与高吞吐量；慢路径则涉及复杂的计算与决策，由 DPU 内集成的 CPU 单元进行处理。

为确保从传统 IaaS 架构平滑高效过渡到 IoD 架构，如图3.6所示，DPU 设计上普遍兼容通过 OVS 流表卸载或集成 Linux TC Flower 规则的配置方式，以此灵活编程和优化 DPU 的数据转发逻辑，实现无缝迁移与高性能网络管理。

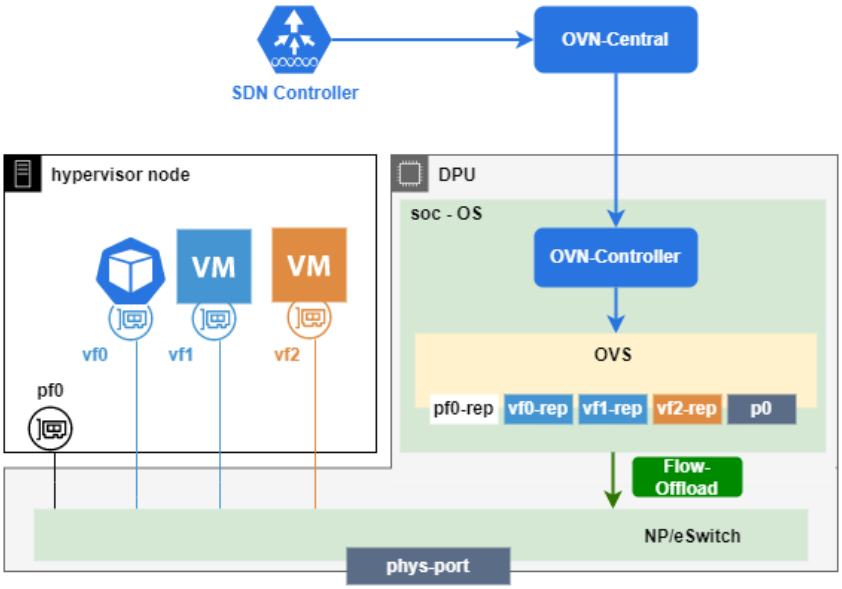

图 3.6: IoD 云计算系统对接示意图

3.4 云计算存储技术分析

3.4.1 单一存储技术方案无法满足云计算要求

在云计算中，不同业务场景对存储的需求存在显著差异，因为每种业务都有其独特的特点和数据处理方式。这种差异性导致了单一存储技术方案无法满足所有业务需求的情况。

首先，对于需要高性能和低延迟的在线交易处理业务，如金融交易系统，存储系统需要能够快速响应请求并确保数据的实时性和一致性。这种业务场景对存储系统的读写速度和数据保护要求非常高。

其次，对于大规模数据分析和处理业务，如人工智能模型训练和大数据分析，存储系统需要具备高容量、高吞吐量和良好的扩展性，以支持海量数据的存储和处理需求。在这种场景下，存储系统的数据处理能力和存储效率至关重要。

另外，对于需要长期数据存储和备份的业务，如归档数据和合规性数据存储，存储系统需要具备高可靠性、数据保护和长期保存能力，以确保数据的安全性和可靠性。这种业务场景对存储系统的数据保护和数据完整性要求较高。

综上所述，不同业务场景对存储的需求差异主要体现在性能、容量、可靠性和数据处理方式等方面。因此，为了满足云计算中多样化的业务需求，需要结合多种存储技术方案，以提供针对不同业务场景的定制化存储解决方案，从而更好地支持各类业务的存储需求。这种多元化的存储技术方案选择能够更好地适应云计算业务的多样性和复杂性，提升整体业务的效率和灵活性。

3.4.2 云存储需要引入新技术突破性能限制

在云计算业务模型中，文件存储、对象存储和块存储在不同的业务场景下通常会使用不同类型的存储协议来实现数据存储和访问。

- 文件存储：适用于需要以文件为单位进行管理和访问的场景，如共享文件、应用程序数据存储等，通常会使用网络文件系统（Network File System，NFS）或者服务器消息块（Server Message Block，SMB/CIFS）等协议来实现文件级访问和共享。
- 对象存储：适用于存储大规模非结构化数据，如图片、视频、文档等，并具有高扩展性和容量无限制的特点。通常使用基于 HTTP/HTTPS 的 RESTful API 作为存储协议，如 Amazon S3 API、OpenStack Swift API 等，用于上传、下载和管理对象数据。
- 块存储：适用于需要高性能、低延迟和数据一致性要求较高的场景，如数据库存储、虚拟机存储等。块存储通常使用协议如 iSCSI（Internet Small Computer System Interface）或者 NVMe- oF（NVMe over Fabrics）来提供块级访问，以便直接将存储设备映射到主机并提供块级数据传输。

通过选择适合不同存储类型的存储协议，可以更好地满足云计算业务模型中文件存储、对象存储和块存储的各自需求。但是随着云计算应用类型的发展，尤其是像 AIGC 这一类应用的爆发，对包含块存储、文件存储以及对象存储都提出了更高的性能要求。比如百万级别的单磁盘 IOPS 需求，微秒级的存储时延需求等。因此需要综合考虑业务场景、性能要求和数据访问方式，通过引入 E-BoF 存储后端、CAS 容器附加存储、DPU

存储加速等新技术来综合提升存储系统性能，才能够为应用类型的发展提供完备的存储能力支撑。

3.4.3 IoD 技术可以提升存算分离架构下的处理性能

在复杂的云计算场景中，DPU 在存储方向上扮演关键角色。DPU 通过存储加速、数据处理、数据安全和智能存储管理等功能，优化存储系统性能和效率，适用于不同云计算业务需求。结合云计算业务，DPU 可提供高性能存储加速，满足对速度和响应时间要求高的应用；其数据处理功能减轻主机 CPU 负担，提高整体计算效率；其数据安全功能保护云端数据免受攻击，确保数据隐私和完整性；其智能存储管理功能优化资源利用率，提高云端存储系统的可靠性和可扩展性。通过与网络存储设备集成，DPU 实现高效数据传输和存储管理，为云计算业务提供高性能、安全可靠的存储解决方案，满足多样化的存储需求。

为应对传统 IaaS 架构中存储服务效率低下的困境，IoD 架构从两方面进行策略优化。首先，它将网络存储协议处理任务从宿主机 CPU 卸载至 DPU，由 DPU 直接对接远程存储系统，并通过 PCIe 接口为宿主机呈现遵循标准驱动协议的 NVMe 或 VirtIO 块设备，这一迁移显著减轻了宿主机 CPU 的负担。其次，该架构利用 DPU 集成的 RDMA 能力，利用内核旁路与零拷贝技术，显著增强了 DPU 与存储后端间的数据交互效率，实现了数据传输的高速直通，规避了传统 TCP/IP 协议栈中的性能瓶颈。

如图3.7所示，IoD 架构下的存储应用，不仅宿主机 CPU 资源得以释放，更专注于业务处理，而且凭借 RDMA 的高效数据传输能力，确保了存储服务的高性能表现，为云环境下的数据密集型应用提供了强有力的支撑。

在设备虚拟化方面，IaaS 存储领域也沿袭 IaaS 网络虚拟化的优化思路，基于 VirtIO 后端处理逻辑的下沉至 DPU，并结合 vDPA 技术，为虚拟机和容器环境打造了一致且高效的 VirtIO 块设备接口。这一设计不仅确保了存储访问的高性能表现，还通过标准化接口强化了虚拟机的热迁移特性，提升了云平台的整体灵动性和可靠性。

存储的数据安全直接关系到信息资产的防护与合规问题。在实施存算分离架构的场景下，为维护数据传输的隐私和完整，加密操作需在发起端（Initiator）执行，筑起数据安全的第一道屏障。传统 IaaS 架构下，数据加解密依赖宿主机 CPU 的软件处理，尽管普适性强，但显著消耗 CPU 资源，拖累运算效能，尤其在处理大数据应用时，性能瓶颈更为突出。IoD 架构则展现出变革优势：数据加密处理下沉至 DPU 硬件层面，实现实时随路（inline）加密处理，大幅减轻了对 CPU 的依赖。借助硬件加速技术，不仅

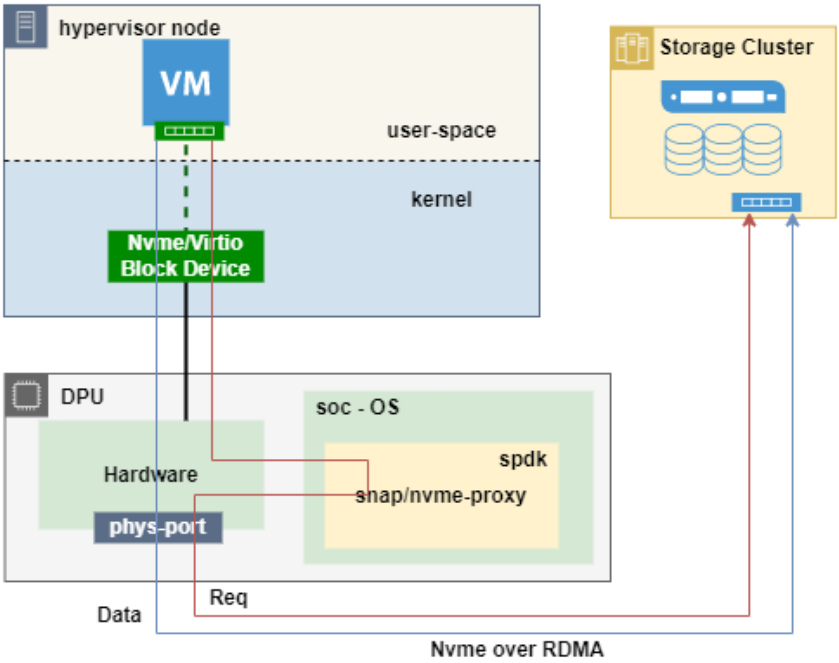

图 3.7: IoD 云计算存储卸载图

加密效率显著提升，数据安全性亦得到增强，为云存储带来更高效、稳固的解决方案。

综上所述，DPU 在文件、对象和块存储方面的改进和提升，为云计算业务提供了更高效、更安全、更可靠的存储解决方案，满足不同业务场景下对存储性能和管理的多样化需求。从存储网络协议的角度看，将 NVMe-oF 卸载到 DPU 中可以为存储底层网络带来更高性能和低延迟，更好的系统扩展性等优势。

3.5 云计算安全技术分析

3.5.1 纷繁庞杂的云计算安全体系

和传统数据中心业务环境相比，云计算环境租户网络边界模糊，共用基础网络结构，安全隔离依赖于策略管理，云服务厂商的安全管理可控度低，数据的保密性面临更多的安全风险，这些问题导致云计算安全体系更加纷繁庞杂。

业务系统的复杂度决定了网络安全体系的复杂度。通常情况下，网络安全设计会从安全物理环境、安全通信网络、安全区域边界、安全计算环境、安全管理中心等多个维度打造立体化安全防护体系，安全合规是基础底线，上层业务安全需求层出不穷。云基础设施由云厂商建设，云厂商或云服务商为云数据中心基础设施提供安全保障，确保不同租户群之间的安全隔离，云租户基于云计算平台自建业务，为自身整体安全负责。

除了传统的网络安全防护之外，云计算安全防护体系还需要关注虚拟边界安全、虚拟化安全、容器安全、镜像和快照保护、数据完整性和保密性、虚拟机之间或容器之间的安全检测，云服务商还需要对云租户的网络、客户系统、数据操作等进行审计。计算环境比网络边界安全更为复杂，安全防护设施相对薄弱，计算环境的安全问题已经成为云计算安全体系中的短木板，亟待加强。

在确保安全、性能、效率的前提下，帮助客户打造高性能的云计算环境，保障业务持久稳定的运行，是打造高性能云计算安全系统的重中之重。

3.5.2 安全处理性能提升需要异构算力加持

从需求端分析，随着云计算、大数据、视频云联网、人工智能的高速发展，网络流量负载呈指数级增长，数据已经成为资产，数据的重要性日益突出。另一方面，云计算环境的网络和数据安全风险日益严峻，网络安全攻击事件时有发生，安全防护需求不断增加。

网络安全处理机制需要对网络报文进行细粒度的处理，需要对全流量进行解析和检测分析，在此基础上执行一系列复杂操作，例如应用层协议识别、威胁检测、关联分析、字段转换、负载均衡、网络协议封装与剥离、数据加密与解密等，这些流量处理动作非常消耗服务器算力，为了满足高性能安全需求，服务器大部分甚至全部算力会被安全功能消耗掉，这将导致业务应用工作负载没有算力可用。网络接口带宽不断提升，对于 100GE、200GE 高速网络场景，即使将全部服务器算力用于安全功能负载，也无法满足高性能安全需求。通用服务器主机受限于底层软硬件架构，无法突破安全性能瓶颈，借助网络处理芯片的异构算力来提升安全性能，已成为业内共识。

DPU 成为高性能云计算异构算力的关键组件，将高算力消耗的安全组件从主机侧下沉到 DPU，实现业务应用工作负载和安全基础设施工作负载分离，是当前云计算安全体系主流趋势。让宝贵的服务器算力资源聚焦于业务应用工作负载上，由专用 DPU 完成高消耗的安全基础设施工作负载，可以为客户提供更加安全高效的业务体验。

借助强大的安全异构算力，DPU 可以对一系列安全功能进行硬件加速，而不会增加主机 CPU 算力消耗，这些功能包括防火墙访问控制、动态加解密、协议识别、威胁检测、可信计算等方面。

3.5.3 安全卸载技术在高性能云安全中至关重要

安全卸载技术可以将服务器承载的高消耗安全工作负载卸载到 DPU 中，使主机将宝贵的算力真正聚焦在高性能应用服务上，所以安全卸载技术在高性能云安全中至关重要。安全卸载技术需要具有如下功能和特点：

- 支持安全卸载技术，充分释放服务器主机算力。
- 提供面向租户的贴身防护，云内虚机/容器安全隔离，包括东西向和南北向流量可以实时检测和阻断。
- 支持分布式部署，部署在海量安全计算环境中，不额外增加机房空间。
- 具有灵活可编程的网络处理器，实现网络解析和编辑。
- 采用专用硬件加速器，具有高性能安全处理能力。
- 支持控制面、数据面的灵活卸载方式，满足客户各种应用场景的安全加速需求。
- 支持功能弹性可扩展，网络、安全可以随时获取，灵活可扩展，安全随着业务变化而变化。
- 具有通用的产品适配性，即插即用的驱动和封装接口，灵活对接主流的服务器硬件、操作系统、OVS、云平台等，具有良好的兼容性。

3.5.4 DPU 将成为可信计算服务中的重要组件

由于漏洞缺陷层出不穷，传统基于先验特征库的检测方式，无法有效应对不断升级的黑客攻击，无法有效保障业务应用的安全，数据资产被攻击、业务被中断等安全事件时有发生。国家网络安全等级保护 2.0 强化了可信计算功能要求，把可信验证列入各个级别并逐级提出各个环节的主要可信验证要求，可基于可信根对设备的系统引导程序、系统程序、重要配置参数和应用程序等进行可信验证。

硬件可信根可以实现核心密钥的安全存储，不可导出，主机受保护数据离开环境无法解密。可信计算通过逐级信任度量机制构建主机从固件到 OS 再到应用的信任链，构建主机内核级信任链，对所有加载运行的程序代码进行完整性度量，确保系统运行的所有程序代码及配置都处于可信状态，抵御已知/未知恶意代码攻击。基于可信计算技术，结合可信根确保系统运行程序可信、防止平台身份假冒、密封数据防盗窃。

DPU 可以通过内置硬件可信根芯片，支持 TRNG 真随机数发生器、SM2/SM3/SM4 密码算法，支持安全固件更新、加密存储、启动度量可信，增强 DPU 卡自身的可信安全。

DPU 通常以 PCIe 接口与服务器主板相连，通过集成硬件可信根芯片，以国产密码为基础，在提供安全功能卸载的同时，对服务器主机的业务系统进行可信验证，包括启动度量、OS 可信、应用程序全信任链可信，提升整体安全防护能力。

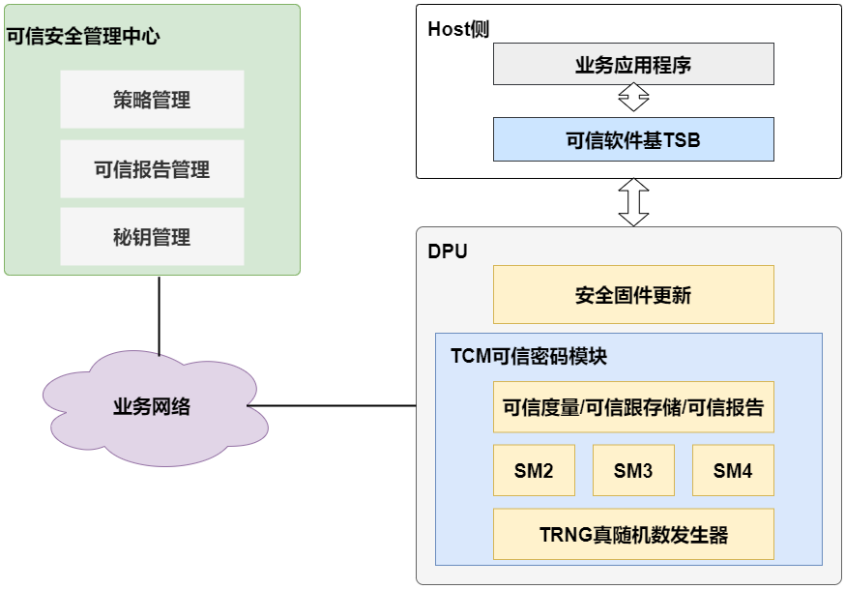

图 3.8: IoD 云计算可信计算服务

如图3.8所示，DPU 集成可信计算能力，有利于积极推进可信计算安全体系在高性能云计算关键信息基础设施中的应用，成为可信计算一种创新性的新实践新方式，筑牢网络安全的坚实屏障。

3.5.5 IoD 技术助力构建“零信任”网络

传统网络安全模型围绕网络边界建立安全防御体系，内部往往被视为受信任区域（Trust Zone），这种模型无法满足日益复杂多变的安全环境。基于零信任安全思想和安全模型由此诞生。零信任是一种以安全性为核心的模型，核心思想是网络边界内外的任何访问主体包括用户、设备、应用等，在未经过验证前都不予信任，需要基于持续的验证和授权建立动态访问信任，始终保持“永不信任、始终验证”的原则。

在云计算环境中，尤其是面向分布式、容器化应用程序的多租户，对于东西向的网络流量，租户的网络安全存在空白。DPU 的软硬件架构非常适合云安全零信任网络，为传统网络安全架构提供新思路、新实践，将成为下一代云安全重要的基础构成。DPU 零信任安全架构优势在于其分布式的、内嵌在计算节点的“关口”物理位置，可以承担零信任安全网关的角色，将可信任网络边界下沉到计算节点，为计算节点提供 inline 模式的网络流量检测和阻断能力，为租户提供东西向和南北向网络流量安全防护；DPU 支

持强大的网络编程芯片、密码算法芯片等，充分卸载主机 CPU 算力，可以为用户进行多层次的验证和授权，对访问主体授于最小的访问权限，并支持全链路加密和持续监控，有效防止内外部的网络威胁。

基于 DPU 的零信任架构，核心能力为业务应用工作负载和基础设施工作负载的分离，基于 DPU 架构对访问主体进行持续验证和授信，DPU 为上层安全应用负载提供强大的安全基础设施加速能力，用户在设备、应用程序、数据的每个接触点上构建可信验证，对访问主体授予最小访问控制权限，并对全链路访问进行数据加密，网络流量解析与监控保护。在如图3.9所示的 IoD 架构下，通过将加解密操作转移到 DPU 的专用硬件模块中执行，不仅释放了宿主机 CPU，还借助 DPU 内置的 CPU 来高效处理控制层面的逻辑，实现了数据处理与安全管理的高效协同。这一设计不仅优化了资源利用，还通过硬件加速保障了数据安全处理的高性能，为云环境下的零信任安全实践提供了坚实的基础。

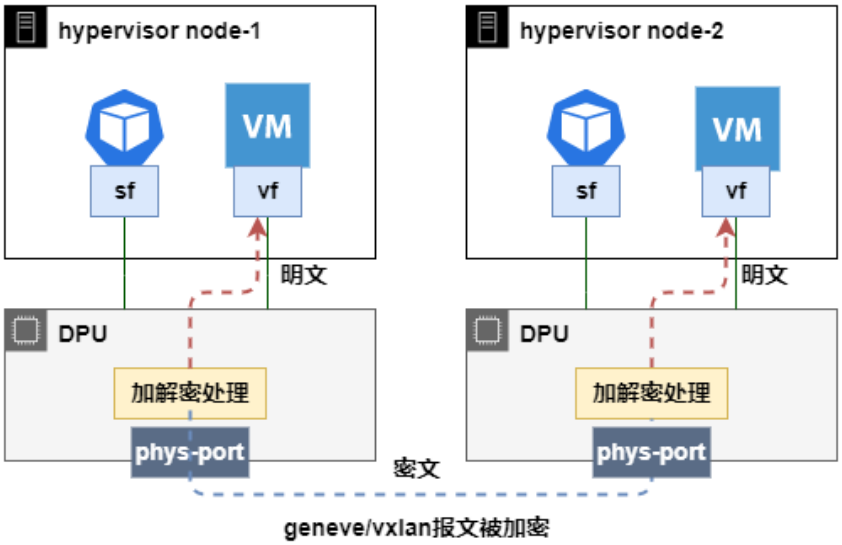

图 3.9: IoD 云计算安全卸载

综上，作为下一代云安全的基础构成，零信任安全的应用离不开 DPU 基础设施，借助 DPU 的各种硬件加速引擎和网络可编程引擎，从底层硬件信任根开始构建逐层的安全应用功能，凭借与业务和安全应用的深度融合，与云控制平台的分布式安全策略联动，最终实现面向云计算场景的零信任网络安全体系。DPU 零信任安全架构，可以促进零信任安全技术和应用的快速发展。

3.6 云计算服务治理技术分析

3.6.1 服务治理技术是云原生时代的重要基础

微服务架构已经成为现代软件开发的重要趋势。在微服务架构中，大规模应用被拆分成多个微服务，每个微服务独立运行在不同的容器中。微服务架构具有模块化、独立部署、松耦合、高可用的特点。

和传统的单体式应用相比，微服务架构具有一系列优势，但是也面临一些挑战。微服务架构中的每个微服务都需要注册和监控，微服务自身需要支持重试和重传、服务发现等功能，微服务组件之间需要通信。如果每个微服务都自己去处理这些类似的工作，代价较大。假如微服务使用了不同的编程语言（如：C、Java），也难以共享这些功能组件。同时微服务架构属于分布式系统，存在分布式系统所面临的一系列问题，如：运维工作量大，运维成本高。因此，微服务架构需要进行高效的服务治理。

服务网格能够解决微服务架构的痛点，是微服务架构的新方向。在此之前，一些微服务框架（例如：Spring Cloud、Dubbo）曾被用来进行服务治理。和前期的微服务框架相比，服务网格具有明显的优势。服务网格可以同时支持多种技术栈，比如部分微服务模块用 Java 来开发，其余则可以使用 Golang 开发。业务开发人员只需关注自己的业务逻辑部分，其余服务注册、发现以及追踪治理都交给服务网格来完成。最终实现“无感、无侵入”的效果。

另外，云原生时代的众多技术，如 Serverless 等均是以服务治理技术为基础发展出来的，因此服务治理能力也是高性能云计算的重要衡量标准。

3.6.2 传统服务治理技术的局限性

在云原生场景中，引入服务网格解决了微服务架构的一些痛点，但也带来了新的挑战：

- 服务网格的资源开销大

  每个微服务 POD 都需要运行一个 Sidecar 容器，默认情况下每个 Sidecar 容器占用 2 个 CPU 核。假设一台服务器运行了 60 个 POD，那么 Sidecar 容器将额外占用 12 个 CPU 核。

- 业务转发时延高

  应用程序的每个数据包都必须通过 Sidecar 容器，数据包需要在应用程序和内核之

间往返多次。

3.6.3 IoD 技术带来新的服务治理模式

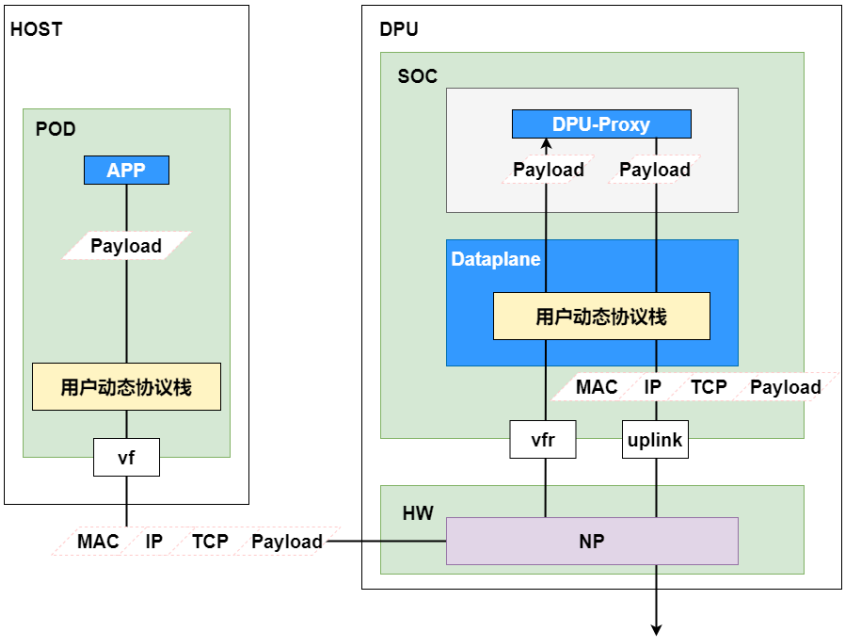

图 3.10: IoD 服务治理卸载

如图3.10所示，在 IoD 技术体系下，可以将原有体系中用来做服务治理的 Sidecar 容器下沉到 DPU，同时采用“集中式”网关的模式来完成服务，这一思想也契合了当前服务治理的技术发展方向，如 Cilium Service Mesh 与 Istio Ambient 等都采用了类似的方案。同时结合主机侧协议栈 PRELOAD 技术与 DPU 优化的 Data Plane 设计，可以有效降低服务治理业务的处理时延。

3.7 IaaS on DPU(IoD) 高性能云计算全景

总的来说，IoD 技术是从云计算架构视角出发，结合 DPU 的实际能力，尝试将云计算的网络、存储、安全、管控、运维等尽可能多的能力卸载下沉到 DPU，在尽量保证现有技术体系能够平滑演进的同时，又能够为云计算带来巨大的性能提升。基于 IaaS on DPU (IoD) 技术的高性能云计算的全景图如图3.11所示：

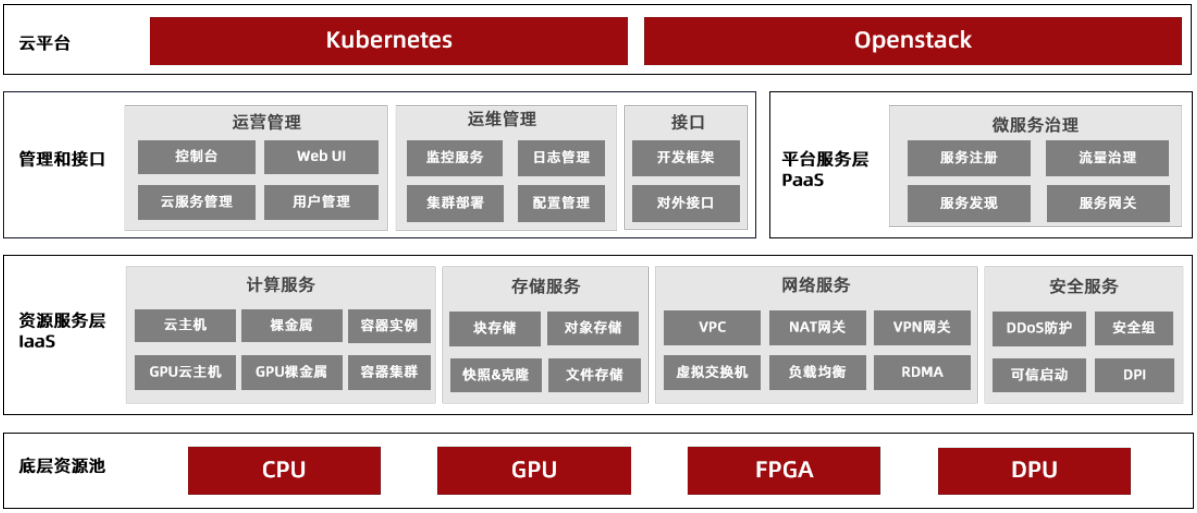

图 3.11: IoD 技术全景图

第 4 章 高性能云计算系统架构持续演进

4.1 高性能云计算可观测性建设

4.1.1 可观测建设是云计算运维体系的关键环节

随着云计算多租户、虚拟化、云原生等应用的普及，传统的从交换机等网络设备侧插入独立导流监控系统的方式已经无法完整地、有区分度地监控那些部署在云中的业务系统的健康程度。上云的业务越来越多，云计算所能提供的业务相关的可观测性也逐渐被重视起来——CNCF 将可观测性称为云原生时代的必备能力，而 Gartner 也将可观测性列为“2023 年十大技术趋势”之一。

可观测性不是某种具体的工具或技术，而是更偏向于是一种能力或理念，是指运行中的系统可被调试的能力，这种可调试能力的核心就是能够在系统运行时，有足够的工具、方法，能够从多个角度对其进行观察、理解、询问、探查和调度。

早期可观测性建设仅仅只是获取系统相关的数据并用之进行系统级的运维监控，更多的是用于发现问题；而随着业务系统的多样性、复杂度的增加，尤其是基于“云原生化”微服务趋势，业务系统的部署更加灵活和抽象，此时，仅凭某一个角度、某一组数据或者纯人工操作，无法满足系统运维的需求。这种情况下，不仅是业务系统本身的软件和成千上万的进程，其所依赖的云计算基础设施资源的状态，如网络负载、存储效率、安全漏洞等等，以及分布式部署于多云环境中的子系统，都需要被观测到。

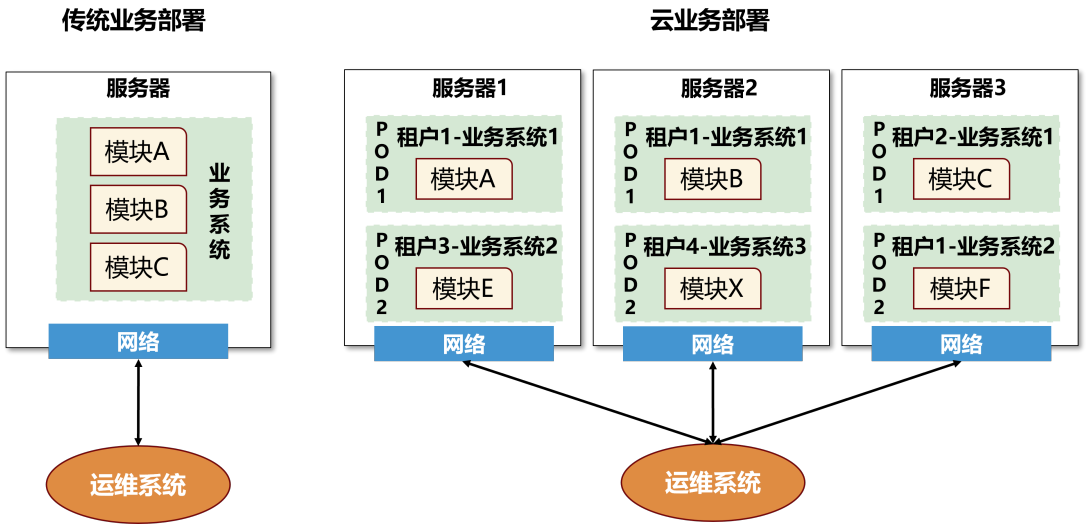

图 4.1: 云计算可观测体系

如图4.1所示，可观测的部分从最早的服务器级别、个别指标的观测，到现在需要进行数据流、数据包级别的观测，还要区分不同服务器、不同租户、不同 POD，粒度更细，复杂度更大。同时，开发运维人员还要通过这些观测到的数据，再进一步地进行人工或自动化的判断、调整、优化等工作，以便让业务系统更好、更安全地工作。

可观测性目前已成为能否成功管理部署在云计算中的复杂分布式系统的重要组成部分，也是 DevOps 的关键。

4.1.2 当前观测方法所面临的难题

在云计算中，当前常用的观测方法有两种：

1. 自行搭建运维监测系统

   通过软件、监测度量工具包和配套的 agent 代理，实现对构建在云计算中的业务系统全方位的、自动化、可定向调试的观测。

   这种方式通常在大型互联网云计算厂商的全自研云计算平台中被采用——原厂有足够的研发能力，按照监测系统的设计实现对云计算平台进行大规模的改造，并与监测系统对接。例如：需要针对可观测设计所定义的各项业务系统健康指标（建链数、吞吐、CPU 占用率、带宽、接收报文特征、拓扑及连通状态等等），定义不同的反馈消息格式；而这些指标的来源，也需要通过在虚拟机、裸金属或者容器的操作系统中，插入自研的与之配合的 agent 代理插件来反馈给监测系统，再在监测系统中实现数据的整合、分析和判断结论。

   由此可见，这种方式并不是一种通用的可实现方式，研发投入成本高昂，且对云租户的操作系统有较为明显的侵入性（非纯净操作系统）。另外，由于云计算尤其是虚拟化、容器化的复杂性，例如各业务系统不同模块所在的虚拟化计算节点之间互通，需要经过多次的地址转换，网络信息需要与云平台的租户、网络等信息相对比、结合，才能呈现出完整的（针对云平台运维）和细粒度的（针对租户业务运维）可观测性图谱。当云计算中的业务数量、数据量越来越多时，整个系统尤其是是 agent 代理，对服务器算力资源的占用将十分巨大，增加了云厂商、云用户的资源开销，造成成本上升。

2. 引入第三方运维监测系统

   云平台对接第三方的定制化设备和配套软件，实现对构建在云计算中的业务系统全量数据、可进行业务网络分析和调试的观测。

   这种方式通常将第三方的软硬件设备旁挂在交换机侧，最早只能监测到经过交换

机的数据，比如整个服务器的出入流量，在监测系统软件中再进行深度的解析，得到局部的分析数据。

但随着虚拟化、云原生的大量使用，虚拟化网络的粒度细化到服务器内部，导致很多业务系统的流量并不会上送到交换机，而是在服务器内部转发和处理，此时，这个监测系统就无法看到完整的数据，得到的分析结论的价值也很有限。于是，监测系统逐渐与云平台进行简单的配合，通过服务器上流表的配置实现全流量镜像，并引流到监测系统旁挂的交换机，进入到监测系统中。

这种方式虽然对云厂商的人力投入要求较少，而且没有深层的系统侵入性，但是引入第三方监测系统本身也是一笔不小的支出。而且随着云计算复杂度的增加，对安全性要求的提高，云厂商、监测系统厂商需要持续的配合、调整，以适应最新的要求。同时，流量镜像实际是对云平台中所有流量的一份或多份复制，随着业务系统数量和流量的增长，不仅会消耗大量的系统资源、网络资源，第三方监测系统的所需要的资源也将不断升级扩容，造成成本的增加。

可见，如何以最有效的方式和最优性价比，将云计算基础设施与租户业务系统、分布式微服务应用等相结合，实现有价值的对于租户业务系统状态的可见性以及自动化分析、调度能力，仍然是当前云计算可观测性的难题。

4.1.3 高性能云可观测性建设建议

CNCF 明确定义了可观测性三大支柱：度量 (Metrics)、日志 (Logging) 和链路追踪 (Tracing)。在新一代高性能云计算基础设施中，可以理解为：通过多维度、全方位的参数实现完全的度量，通过可回溯、完整记录的日志，通过多样化、可定义的链路追踪工具和方法，共同融合交互，从而实现对部署在云上的业务系统的可观测性。

云计算的本质之一就是多租户共享资源。随着租户和业务数据量增多，业务系统复杂度也线性增长，导致需要进行观测的维度更复杂、更多元，对追踪探测的手段的要求也越来越高，这势必会增加观测行为所需要的算力资源、网络带宽、系统侵入深度等，需要探索一种新的观测方案。

一种高性能且极具性价比的方案是，在所有数据必经之路的 DPU 上，顺便实现对数据的捕获、分析等。

这种方案有如下优势：

- 不会增加云网络的额外开支，无需进行报文复制和流量镜像，DPU 原生的就可以区分租户网络、租户业务网络并形成拓扑信息。

- 通过 DPU 本身的数据处理能力，可以配合监测系统完成对租户业务系统的部分状态的分析工作，降低监测任务对服务器的消耗，简化监测系统的设计复杂度。
- 结合 DPU 对网络的管理、探测能力，可天然支持多种对于业务系统的链路追踪手段、工具，用原生的网络工具实现原生的链路追踪。
- 对租户操作系统零侵入，所有对流量、报文的操作和辨识都由 DPU 完成，租户可以放心使用纯净操作系统，免除 OS 额外插件带来的担忧。

eBPF 技术和灵活可编程 DPU 的成熟，使这种方案成为可能。eBPF 本身就是原生的内核技术，Linux 内核天然支持，所以也无需额外中断来进行状态观测。而且 eBPF 可以将对数据包的处理从内核空间改为用户空间，即外部实现按需多样的沙箱式编程，通过回调函数触发以字节码注入内核的原生操作。eBPF 及相关的监控程序可以卸载到 DPU 硬件上，从而实现更高的性能。

高性能云可观测性方案可达到的效果如图4.2所示：

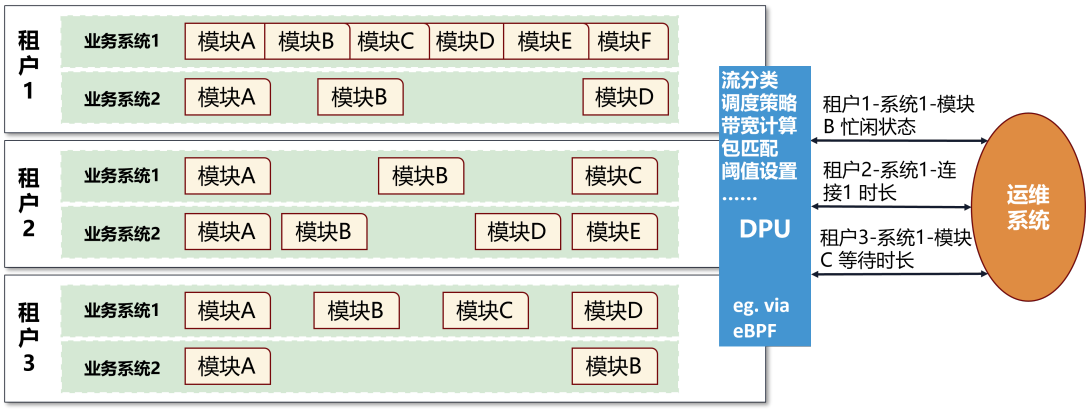

图 4.2: 基于 IoD 技术构建的可观测体系

当然，业界对云计算可观测性的探索一直没有停息。新技术新产品的出现，也给云计算可观测性提供了更多的实现方案和路径，有助于为云用户提供更加经济实用的业务可观测性的能力。

4.2 轻量级虚拟化系统演进架构革新

4.2.1 轻量级虚拟化技术演进路线

虚拟化技术目前呈现如图4.3所示的态势：

容器是最轻量的虚拟化技术，但是由于其技术实现限制，隔离性、安全性无法满足所有的业务场景。虚机是隔离性安全性最好的，但是在资源损耗、启动速度等方面落

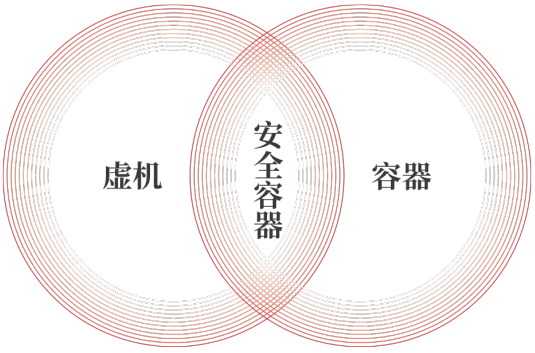

图 4.3: 安全容器技术

后于容器。因此，出现了安全容器这个折衷方案，安全容器在容器中运行一个轻量级 Hypervisor，使得兼顾隔离性安全性和效率。安全容器技术是目前发展势头最好的轻量级虚拟化技术。但是轻量级虚拟化技术并不单指安全容器，目前，在虚机和安全容器之外，还有众多的轻量级虚拟化技术都还在快速演进。

轻量级虚拟化的“轻量”主要体现在以下方面：

- 虚拟化本身的资源损耗低。
- 创建销毁快，能够接近容器的弹性。

目前有两条主要的技术路线：

- 轻量 Hypervisor：抛弃 Xen、QEMU 这种大而全的 Hypervisor，针对具体场景，优化 Hypervisor 的实现，降低虚拟化损耗，加快启动销毁速度。典型的有 AWS 开源的 Firecracker，专用于 FaaS 场景，可在 100ms 内启动，单台服务器上可启动数千个轻量虚机。Intel 主导的 Cloud Hypervisor 也能够在接近 100ms 的时间启动虚机。
- 硬件卸载：通过 DPU 或硬件加速卡等专用硬件，把复杂的设备模拟从 Hypervisor 中解放出来，使得 Hypervisor 进一步轻量，同时虚拟化效率和性能获得提升。典型的有 AWS 的 Nitro System 和阿里云的神龙系统。

4.2.2 轻量级虚拟化技术为云计算带来新气象

安全容器结合了容器的弹性和虚机的隔离及安全性优势，可满足 FaaS 等容器方式无法满足的业务场景。

而硬件卸载的轻量虚拟化技术则彻底颠覆了 IaaS 的基础架构，使得传统的 Hypervisor 变得越来越轻薄，最终将演变成硬件的一个代理。即所有的设备模拟都会由以 DPU 为代表的硬件加速卡来完成，而 Hypervisor 只是需要和相应的硬件交互，完成资源的申

请释放即可。虚拟化的效率和隔离性也将在这个过程中达到极致，最终，Hypervisor 几乎不消耗任何主机侧的资源，使得所有的资源都被业务系统所使用。通过硬件队列，硬件 QoS 等硬件隔离技术，使得多租户之间达到真正的完全隔离，互不影响。

4.2.3 DPU+ 轻量级虚拟化 = 新一代技术革命

引入 DPU 后，形成以 DPU 为核心的 IoD 架构。一方面，将主机侧所有的控制面组件下沉到 DPU，进一步释放主机侧的资源；另一方面，简化 Hypervisor 的实现，将可被硬件模拟的设备通过 DPU 硬件来模拟。最终呈现如图4.4所示形态：

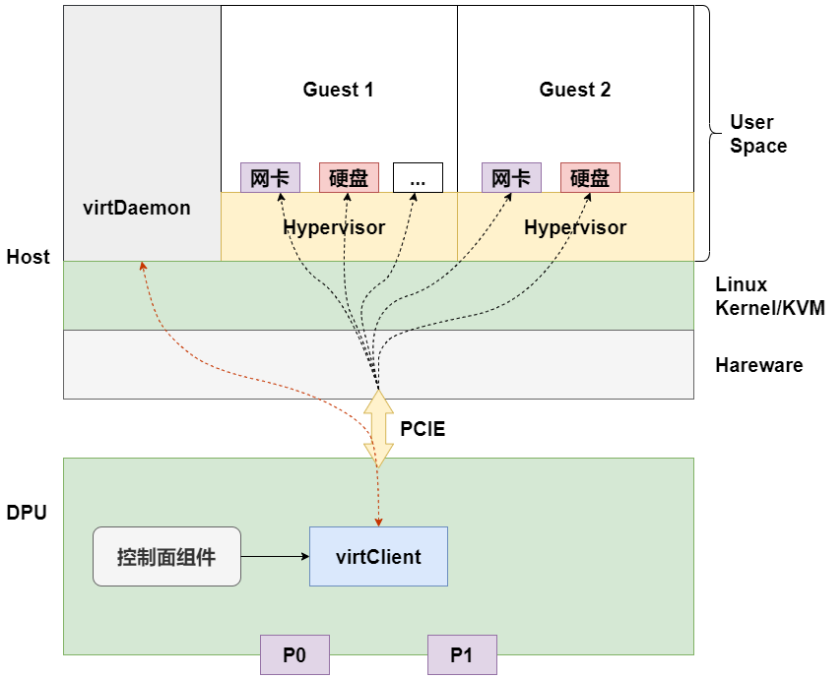

图 4.4: 轻量级虚拟化的 IoD 方案

控制面下沉到 DPU，主机侧只需保留一个 virtDaemon 进程，响应 DPU 侧的调用，调用 Hypervisor 完成虚机的管理。Hypervisor 是轻量优化的，几乎没有虚拟化损耗，而 virtDaemon 是一个被动管理组件，资源消耗非常低。主机侧和 DPU 的通信，以及硬件设备的模拟，都通过 PCIe 通道完成。

当然，对轻量级虚拟化的技术探索还在持续进行，各种新思想与新技术如雨后春笋般涌现，IoD 技术在轻量级虚拟化领域也需要不断推陈出新，保持技术竞争优势。

4.3 “一云多芯”系统融合

4.3.1 “一云多芯”的应用困境

当前云计算系统建设中，尤其是在国内环境，“一云多芯”是大家普遍关注的焦点。

通常来讲，对“一云多芯”有两种解读方法：

广义理解为由一种主处理器与多种协处理器共同组成的基础云计算环境，在这种语境中，CPU+GPU+DPU 的“3U 一体”架构就是最典型的“一云多芯”应用，此应用场景下 IoD 技术便是一种最佳实践。

狭义理解为多种不同架构与指令集的处理器需要在一个云环境内进行统一调度与管理，例如同一个 AZ 内同时存在 X86 与 Arm 处理器的情况，真实应用场景往往涉及到多种不同体系架构的 CPU、GPU 混合部署，环境更加复杂。

在狭义理解的语境下，如何实现“多芯云”在同一资源池内的统一调度与管理，仍然面临着巨大的挑战：

- 软硬件兼容性：随着引入“芯”的类型增多，需要关注的兼容性问题将以笛卡尔积模式增长。
- 任务调度：由于指令集不同，上层业务无法顺畅的在不同服务器之间迁移，尤其是虚拟机热迁移功能将无法使用。
- 服务评估：各种不同处理器难以简单按照相同方法进行业务的预先与事后评估，对云业务的可观测性建设提出了更高的挑战。

4.3.2 IoD 技术有助于完善“一云多芯”的服务评估体系

在“一云多芯”环境中，DPU 除了通过卸载加速能力为集群提升能效比之外，还能够从以下两个方面优化“一云多芯”的服务评估体系。

1. 采用 IoD 技术之后，将整个基础设施层下沉到 DPU 设备，将基础设施组件的运行环境保持统一，不再受不同 CPU 架构影响，可以简化基础设施层相关的评估与可观测难度。
2. DPU 作为服务器的核心数据通道，通过 DPU 构筑服务评估体系，实现用“灰盒”方式对云计算服务效率进行评估，能够最大程度弱化不同 CPU/GPU 架构与指令集对服务评估结果的影响。

总的来说，当前“一云多芯”架构仍在快速发展，IoD 技术可以预见将在未来“一

云多芯”的应用场景中发挥重要作用。

第 5 章 高性能云计算为 PaaS 服务赋能

随着云计算技术的发展，PaaS 能力的构建经历了从无到有，从单一到多样，从公有云扩展到私有云的发展历程。PaaS 服务的引入可以极大地提升上层业务的研发、部署和运维体验，已经成为云计算服务中的重要组成部分。PaaS 服务的性能往往与云计算集群的基础网络、存储等能力密切相关，因此基于 IoD 技术构建的高性能云计算系统，依托于其高吞吐、低时延的高效 IO 能力也能够给 PaaS 服务带来诸多好处。

5.1 高性能大数据计算服务

传统大数据系统通常基于 Hadoop 的生态体系，计算和存储一体的整体架构模式，但随着互联网等各个行业数据量的快速增长，企业对大数据算力、存储和网络资源需求也进一步提升，且在当前企业“降本增效”的大背景下，传统 Hadoop 生态下的大数据处理集群面临了越来越多的问题。

- 资源使用：传统 Hadoop 集群存在资源利用率低，且资源管理粒度和弹性差的问题，对于计算、网络、存储等多维度资源难以细粒度高效管理。
- 存算一体：存算一体的计算架构，虽然一定程度简化了集群的部署，但却放大了资源管理弹性差的问题，计算和存储资源难以单独管理和扩展，也增加了硬件成本。

由于传统大数据体系架构的局限性较强，随着 IoD 技术所代表软硬件融合技术的不断迭代和快速发展，现代化大数据技术体系架构逐渐呈现出异构加速、存算分离等重要的特点。相较于传统 Hadoop 大数据架构，能带来如下优点：

- 高性能：使用 DPU 加速 IaaS 基础设施能较大提升基础网络和存储性能，从而能较大幅度改善网络密集型和 IO 密集型作业的性能瓶颈。
- 降本增效：使用 DPU 加速的云计算构架能较大幅度降低网络、存储相关的 CPU、内存等计算资源的消耗，节约了资源的同时，使其能有更多的业务可用资源；云原生架构下，传统 Hadoop 集群存在的资源利用率低，且资源管理粒度和弹性差等问题，也能得到有效的解决。
- 存算分离：相较于传统“存算一体”计算架构，使用 DPU 加速的云计算场景下的“存算分离”架构能有效解决计算和存储弹性管理差的问题，能较大服务节约相关

资源；同时使用 DPU 加速的云盘也能较大幅度提升存储性能，进而提升存 IO 集型作业的整体性能。

5.2 高性能中间件服务

中间件按功能种类，可分为网络中间件、消息中间件、缓存中间件、数据库中间件等，一般网络和存储 IO 的开销较高，且对延迟敏感。传统方案通常部署于物理机或者虚拟机之上，网络传输协议一般基于 TCP。随着行业相关业务的不断发展，对其性能的需求越来越高的同时，其高成本的问题也随之被放大，传统方案在很多方面越来越难以满足需求。

- 资源方面：基于虚拟机或物理机的部署方案，资源弹性差，且对各维度资源的管理粒度低，导致整体的资源利用率低。
- 延迟高：数据传输基于 TCP 网络，且经过内核协议栈处理，会带来网络延迟高、且不稳定的问题。
- 性能一般：受限于网络和存储性能，难以提高性能，水平扩展虽能带来更高的性能，但也会导致更高的成本开销。
- 可伸缩性一般：基于物理机或虚拟机的部署方式，难以较快的进行中间件大规模快速伸缩，影响业务的实时性。

如今，中间件服务也向容器化方式转型，基于 IoD 技术的基础云计算环境能够较大提升服务性能的同时，改善业务延迟，同时提高资源的利用率，实现“降本增效”。

- 提高资源利用率：基于容器部署的中间件服务，能有效提升计算、网络、存储等多维度资源进行统一弹性管理。
- 降低延迟：基于 DPU 的 RoCE 相较于传统 TCP，能较大提升网络性能，提高网络带宽，且大幅度优化网络延迟。
- 提高性能：利用 IoD 技术带来的网络与存储加速技术，从而较大幅度改善中间件的网络和 IO 性能瓶颈，并提高处理性能。

5.3 高性能数据库服务

相较于大数据计算，数据库一般对高并发性能、低延迟、高 TP 性能提出了更高的要求。传统数据库通常基于物理机/虚拟机的分布式/主从等部署方式，为了保证性能需

求，通常会对各维度资源进行冗余配置，且对数据库的实例管理也存在启动慢、弹性差等问题。

- 资源利用率低：传统数据库，由于基于物理机/虚拟机的部署方式，存在大量的资源冗余，资源利用率低，且物理机平台难以对计算、网络、存储等多维度资源进行统一弹性管理。
- 可扩展性差：由于物理机/虚拟机平台的局限性，传统数据库的实例扩容通常需要较高的准备周期，数据库实例启动慢、弹性差。
- 延迟高：传统数据库网络数据传输都是基于 TCP 网络，且经过内核协议栈处理，会带来网络延迟高、且不稳定的问题。
- 性能差：传统数据库方案一般受限于网络和存储性能，会影响到数据库整体性能。

在基于 IoD 技术底座的高性能云平台上运行的数据库方案，能较大幅度提升性能的同时，也能更好的进行资源的统一弹性管理，并提高扩展性。

- 提高性能：使用 IoD 技术能较大提升基础网络和存储性能，从而能较大幅度改善数据库的网络和 IO 性能瓶颈，并提高性能。
- 资源管理：IoD 技术提供的裸金属、虚拟机、容器共池管理方式，能有效提升计算、网络、存储等多维度资源进行统一弹性管理。
- 提高可扩展性：借助如容器的特性，通过容器部署数据库服务，能够实现数据库实例快速启动与横向扩展，有效提升了数据库服务的扩展性。

第 6 章 未来展望

当前，云计算产业正从单纯的软件主导向着软硬件融合的新模式演进，传统云服务在依赖 DPU、GPU 等高性能硬件重构技术体系的同时，也将对产业内各个角色的职责和交互模式进行重新定义：

其一，硬件制造和芯片设计厂商将成为云基础资源的重要提供者。除了传统通用服务器供应商外，GPU 和智算服务器厂商将为 MaaS 等新型云计算服务提供高性能算力基础，而 DPU 厂商则将围绕异构算力资源和高性能网络充分释放资源潜力、打造 3U 一体的云计算基础设施。

其二，云服务和软件提供商将重构云计算软件以适应新型基础架构。云计算操作系统和应用将根据全新的基础架构进行设计，以充分利用 GPU 的并行处理和 DPU 的任务卸载能力。与此同时，针对新型基础架构的开发框架和服务也将融入云平台当中，成为云操作系统不可或缺的一部分。

其三，芯片、服务器、云服务商等多方联合方案将成为主流。多芯片、多架构组成的云计算基础设施将使单一厂商打造软硬件融合解决方案的难度呈指数性增长，而这将加速产业内各方走向各抒所长、联合打造方案的道路。IoD 技术正是多方联合打造的新型技术方案的典型代表。

高性能云底座是云计算发展的重要方向，它的实现依赖产业各方的共同努力：

一是主管部门和行业组织应制定明确的政策和标准，鼓励云计算基础设施的升级和创新。通过税收优惠、研发补助等政策激励，支持本土企业进行高性能云底座相关技术的研发和应用。与此同时，还可以通过组织行业论坛和技术交流会，促进跨行业的合作和技术共享。此外，政府部门还需要推动建立高性能云底座的标准化体系，确保不同厂商的产品和技术能够互联互通，为市场的健康发展提供保障。

二是云服务商和硬件厂商需要紧密合作，共同开发高性能云计算服务和解决方案。通过协作进行技术研发和产品适配，确保 DPU 等技术能够无缝集成到现有的云计算基础设施中。此外，云服务商还可以通过开展推广活动和技术培训，向用户展示高性能云底座方案在提升计算性能和降低能耗方面的显著优势，并帮助用户更好地理解和应用相关技术。

三是用户企业应积极响应上云用云政策文件，了解和评估高性能云底座方案在其业务中的潜在应用价值，通过主动参与产业链各方组织的试用和测试，提供实际使用中

的意见和建议，在帮助完善高性能云底座相关产品和方案的同时，实现基础设施的全面升级和业务的降本增效。

FREE THE POWER OF DATA