面向泛在计算场景的算力原生技术白皮书

（2022 年）

算网融合产业及标准推进委员会

2022年6月

版权声明

参与编写单位

中国移动通信有限公司研究院、浪潮通信有限公司

主要撰稿人

王羽琪、解子岩、罗馨玥、赵奇慧、黄蕾、王升、王紫程

前　言

随着智能社会的不断进化，人工智能、物联网、区块链、AR/VR等 核心技术成为改善生产水平、民生福祉、政务效率的重要手段，也是算力发展的核心驱动力，带动了云计算中心、边缘计算设备、高算力终端的巨大算力需求，算力也正在从集中云部署往多极化方向发展，边缘算力与端算力的出现与规模化的算力形成互补形成算力，形成“云、边、端”泛在协同发展趋势。

算力日益呈现泛在互联的特征，为打破生态竖井，提高算力利用率，中国移动提出算力原生技术，目标是能够屏蔽底层多样算力资源，使能应用在底层多样计算架构间平滑迁移。

为凝聚产业共识，进一步推动算力原生技术成熟，本文分析了当前算力发展的趋势，系统性介绍了算力原生的概念和当前产业现状、阐述了算力原生总体技术架构、演进路线和关键技术，对算力原生的技术挑战进行了分析并提出推进建议，最后呼吁业界在关键技术、加快构建算力原生标准和繁荣开源生态三方面展开紧密合作。

目　录

图 目 录

一、面向泛在计算场景的算力原生

（一）算力原生的背景与概念

算力作为信息社会的基础，已成为数字经济基础性、核心的发展动力，直接影响数字经济发展的速度，决定数智社会的发展高度。受到网络技术发展和网络带宽成本限制，算力资源从集中的部署方式，正在往多级化的方向发展，增设边缘侧设备成为必然趋势，未来将形成云端侧负责大体量复杂计算、边缘侧负责简单计算执行、终端侧负责感知交互的泛在部署形式。未来算力形态将不仅限于人脑，同时将在以大型数据中心、数据中心为代表的云端，以智慧路桩、边缘计算单元为代表的边缘端，以人脑、无人驾驶汽车、IoT 设备、个人电脑、AR/VR 一体机为代表的终端，形成算力“云、边、端”泛在算力部署架构。

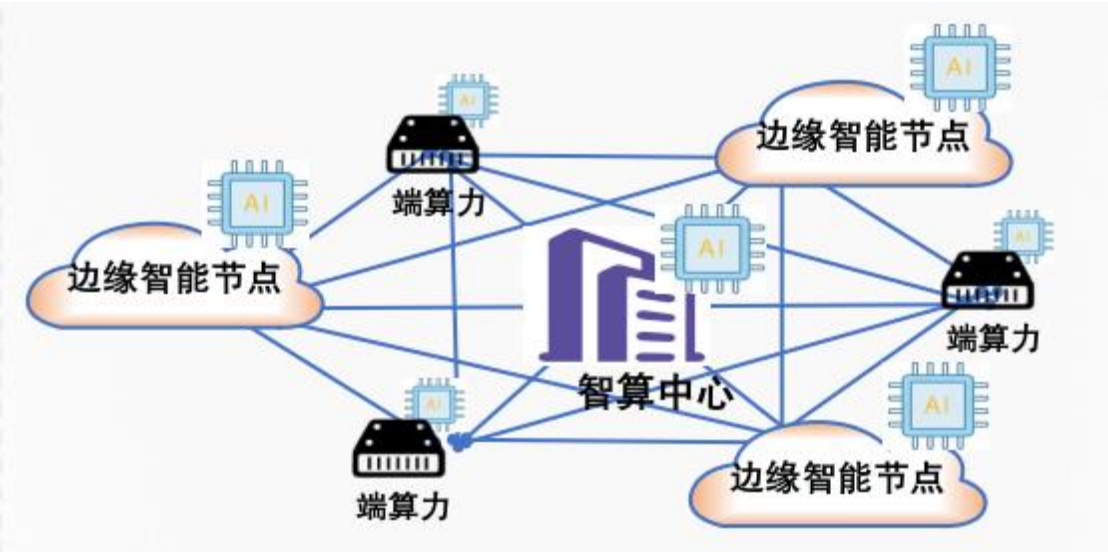

图 1 算力形态呈现泛在互联特征

当前算力泛在化部署，节点间高效互联，算力的形态呈现泛在互联的特征，与此同时，随着人工智能、大数据等产业的迅猛发展，算力芯片厂商和种类日益丰富，算力内核呈现异构化的特征。算力发展当下面临最大的问题是生态竖井问题，各个厂商芯片架构与软件栈碎片化发展，生态相互隔离，导致应用跨架构迁移困难，对于应用开发者，同一应用需要针对不同的软硬件多次开发，带来开发成本挑战；对于算力服务商，应用适配复杂，跨架构迁移难，带来资源利用率挑战；对于芯片制造商，生态壁垒问题日益加剧，带来生态构建挑战。

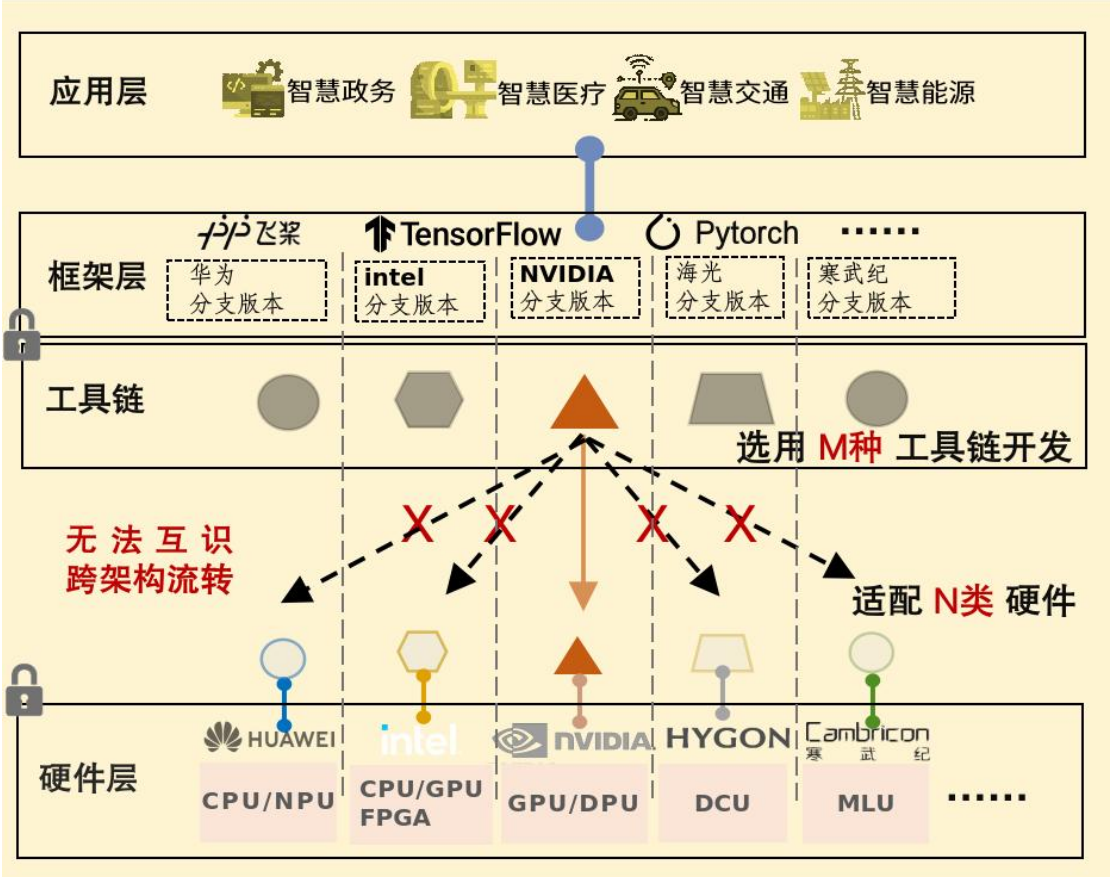

图 2 当前算力存在生态竖井

面对泛在算力发展目前的困境和挑战，本文提出了算力原生技术，算力原生技术是解决大规模、多样泛在的异构计算资源生态隔离，实

现应用跨架构无感迁移执行的重要技术。算力原生技术能为用户提供一个统一的软件平台抽象，使其在编程时既可以充分利用丰富的异构资源、又不必考虑复杂的系统拓扑细节，形成全系统的编程及迁移执行环境。

算力原生是以统一的算力资源抽象模型和标准的编程范式接口为基础，以跨架构编译优化技术和原生运行时技术为依托，使能多样泛在算力环境下屏蔽复杂软硬件差异的技术，实现同一应用一套代码、动态重构一体部署、灵活迁移高效执行的技术。

算力原生通过构建标准统一的算力抽象模型及编程范式接口，打造开放灵活的开发及适配平台，实现各类异构硬件资源与计算任务有效对接、异构算力与业务应用按需适配、灵活迁移，充分释放各类异构算力协同处理效力、加速智算应用业务创新，实现异构算力资源一体池化、应用跨架构无感迁移、产业生态融通发展的目标愿景。

（二）算力原生产业现状

英特尔公司在异构编译开发平台和打造开源生态方面进行了探索。英特尔推出了一个跨架构编译规范 OneAPI，它是英特尔为自身的 CPU、GPU、FPGA、以及各种硬件加速器提供的一个异构编译开发平台，使用统一的编程语言 DPC++及硬件加速库，对下封装一层异构 runtime 插件接口，可实现一套代码的跨平台执行。它提供了通用、开放的编程体验， 面向开发者可以自由选择架构，无需在性能上作出妥协，大大降低了使用不同的代码库、编程语言、编

程工具和工作流程带来的复杂性。同时 OneAPI 规范还基于现有标准创建了 OneAPI 组件规范的开源实现，以使 OneAPI 能够快速用于新的硬件架构或软件语言。

华为公司在统一 AI 开发框架方面做出了探索。为融通智算生态，降低 AI 应用开发门槛，提高芯片算力利用率，华为公司推出了一款 AI 计算框架 MindSpore。它提供了全场景统一 API，为全场景 AI 的模型开发、模型运行、模型部署提供端到端能力，可以适应所有的 AI 应用场景,降低了不同场景适配不同框架的技术难度，可在按需协同的基础上通过实现 AI 算法显著减少模型开发时间，实现一次性算子开发、一致开发和调试的体验。

阿里云在 AI 场景下针对统一异构硬件接口和编译平台方面进行了探索。针对业界缺少统一的异构硬件接口标准以及 AI 芯片厂商需对不同框架进行适配的问题，阿里云提出了基于 HALO/ODLA 的 AI 部署解决方案，异构计算编译框架 HALO（Heterogeneity AwareLowering Optimization）可以基于面向深度学习的异构硬件统一接口规范 ODLA（Open Deep Learning API），将模型编译为与 AI 框架、设备无关的代码，应用经 HALO 与厂商提供的 ODLA 运行库链接后，就可运行在相应的平台上。这种方式，无需厂商提供底层的指令集或编译后端，只需提供符合接口的运行库。同时厂商基本都会提供算子级别的运行库，与 ODLA 对接容易，对新硬件的支持比较友好。

浪潮公司在 5G+AI 视频的行业应用场景下，边缘计算及端计算设备的算力异构化、碎片化，AI 算法应用分发及迁移进行了探索。针对行业用户的产线柔性化及业务敏捷化需求，构建了异构算力协同平台，利用全栈编译计算引擎对接包括 Pytorch、TensorFlow、Caffe、MXNet 等多种主流 AI 计算框架，并支持包括瑞芯微、高通、NVIDIA 在内的多款边缘 AI 计算硬件后端，实现 AI 算法模型到异构边缘 AI 算力的快速适配和移植。通过异构算力协同平台，还可以基于云原生技术自定义 AI 视觉应用业务流程，完成 AI 算法与常用音视频处理 SDK、机器视觉 SDK 的集成、打包和封装，并分发到纳管的边缘 AI 计算设备，实现 AI 视觉算法应用在这些设备上的动态部署、更新及迁移，加速垂直行业 AI 应用的落地和迭代。

趋动科技在计算资源池化方面进行了探索。为解决独占式 AI 算力资源，降低 AI 算力使用成本，趋动科技推出了猎户座（OrionX）AI 算力资源池化解决方案。该方案颠覆了原有的 AI 应用直接调用物理 GPU 算力的方法，通过在 AI 应用与物理 GPU 之间增加软件层实现两者解耦，并通过应用调用 AI 算力的 API 重定向技术，实现 AI 算力资源的池化和细粒度分配。

二、算力原生技术架构

（一）算力原生总体架构

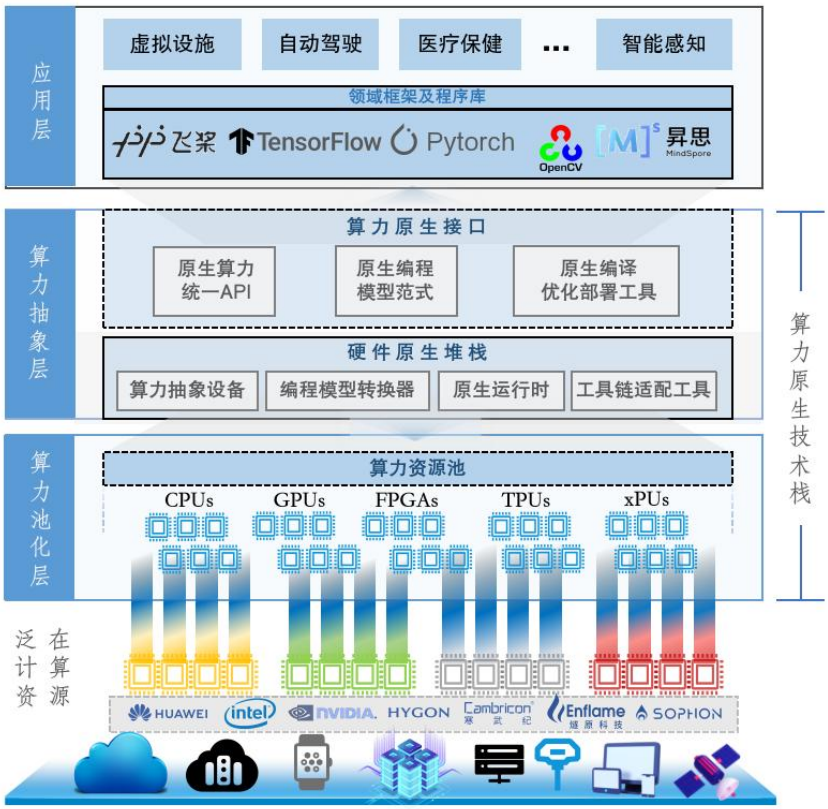

图 3 算力原生总体架构

(1)算力抽象层

算力抽象层实现异构算力资源的统一抽象，屏蔽底层硬件架构，具体包括编译模型转换器、原生运行时和算力原生接口。其中编译模型转换器将基于特定芯片编程的应用程序转译成与底层硬件架构无关的原生程序；原生运行时动态感知底层硬件并实现原生程度与本地计算资源的即时映射；算力原生接口提供原生算力统一API及原生编程模型范式，辅助用户生成原生程序。

(2)算力池化层

算力池化层通过构建底层异构硬件的统一抽象模型，并对应用调用底层算力资源的请求进行重定向和再调度，从而实现各类硬件资源的一体池化，从传统的以硬件资源为单位、静态分配和调度的方式，

变为以计算能力为单位对算力资源进行动态、灵活地配给。同时为应对智算业务的潮汐效应，算力池化层可根据业务需求及算力负载情况提供算力资源弹性扩缩容的能力。

（二）算力原生演进路线

阶段一：核心目标是实现异构算力资源池化。通过构建分厂家分类型的异构资源池，引入对应用调用底层算力资源 API 的重定向技术，实现同厂家算力资源细粒度的定制化分配和灵活调度。在本阶段可实现应用对底层算力资源位置和数量的无感使用，支持业务可根据动态负载情况进行算力资源的弹性使用，充分应对潮汐效应和业务量突发等场景。

阶段二：核心目标是实现应用的跨架构迁移。在实现分厂商、分类型资源池化的基础上，构建虚拟资源模型、屏蔽底层硬件差异，实现跨厂商、跨架构的硬件资源池化供给。同时通过引入编程模型转换器，可实现基于特定芯片厂家开发的应用，在用户无需重构代码的基础上，转换成算力原生中间元语，结合算力原生运行时和各厂商工具链，生成可跨架构互识、流转的算力原生程序，最终实现在多厂家芯片上无感迁移和映射执行。

阶段三：核心目标是实现全局泛在融通。为支撑算力网络中远期实现算力解耦、泛在调度的任务式服务目标，算力原生基于一二阶段实现池化和跨架构无感迁移的基础上，通过打造算力原生统一接口和开发平台，形成统一编程模型及开发环境，使能应用开发者实现跨架构的混合并行编程，加速新应用、新服务创新。

图 4 算力原生三阶段演路线

（三）核心技术

3.1跨架构编译及优化技术

跨架构编译技术是实现应用屏蔽底层硬件差异在多样算力间迁移的基础。应用基于统一编程模型范式编写，经算力原生跨架构编译及优化技术，生成可跨架构流转的原生统一格式，提高算力原生程序的可移植性。跨架构编译技术基于异构编译研究并行编译在串行程序中的并行性、自动循环和向量化数据优化，将异构与并行程序操作下放在编译器中，降低开发者人力资源成本。

跨架构编译技术包括程序自动编译与生成和数据自动管理技术。程序自动编译与生成技术基于统一的编程模型及范式，利用数据对齐、数据分布等技术将程序中的数据自动划分到不同的处理器核中，并根据不同数据分布机制为应用程序插入通讯原语，生成可多模重构的算力原生程序；数据自动管理技术可通过分级数据分布、通讯生成和循环分块等方法对程序中的数据和计算进行分解，使得分解后的数据能够满足局部存储容量的约束。

编译优化技术包括向量优化、存储访问优化、Kernel合并优化等。

向量化优化：支持work-item内和work-item间向量化，以充分提高程序并行性和执行效率。

存储访问优化：支持片上存储的自动分配和DMA带宽优化。通过自动分配片上存储，可以减少内存访问时间，提高程序性能。而DMA带宽优化则可以更好的利用DMA资源，进一步提高程序执行效率。

Kernel合并优化：支持合并不同kernel和合并同一kernel的不同次迭代。通过合并不同Kernel，可以减少数据传输和Kernel调用开销，从而提高程序性能。而合并同一Kernel的不同次迭代，则可以极大地减少Kernel调用次数，从而进一步提高程序性能。

3.1算力抽象技术

算力抽象旨在屏蔽应用对底层多样硬件架构的差异感知，通过抽象化的设计建立一套支持多种异构算力系统的统一算力抽象模型及相匹配的编程模型与接口，是异构算力资源对上层应用及开发者之间的桥梁。一方面使开发者在编程时既可以充分利用丰富的异构资源、又不必考虑复杂的系统细节，同时，使基于抽象模型所开发出的应用程序在运行时可以根据系统架构的变化，即时转换映射，实现快速迁移移植。算力抽象既能够包容多样算力异构，提供一个统一编程模型，又能够保持异构算力自身的并行效率，将是异构混合计算编程的重要发展趋势。算力抽象同时基于统一编程模型构建算力原生统一接口，

使应用开发者不必考虑底层硬件特性，实现一次开发，多处运行的能力，提高开发效率。

统一编程模型范式是算力抽象层面向程序开发者的开发接口和编程范式，是算力抽象层对外提供服务的最直观表达，包括：应用编程规范和算力描述规范。其中应用编程规范为应用开发者提供表达计算和数据的接口，算力描述规范用于约束多样加速硬件平台抽象方式，主要包括指令模型（至少包含SIMD、SIMT等）、内存模型（至少包含共享内存、分立内存等）、执行模型（至少包含同步、异步等）的能力。

指令模型：描述了的指令集架构，以及支持的指令类型。常见的指令模型包括SIMD和SIMT等。这些指令模型帮助我们在IR层面进行优化，从而充分利用平台的硬件资源，提高代码的执行效率。

内存模型：主要描述了平台的内存架构，包括内存类型、内存分布等。常见的内存模型包括共享内存和分离内存等。根据不同的内存模型，可以对统一中间表达IR进行相应的内存优化，从而减少内存的访问次数，提高程序性能。

执行模型：主要描述了平台的执行模式，包括同步和异步等。不同的执行模对于程序的运行效率有很大的影响。在层统一中间表达层面，可以根据执行模型来进行代码重组和优化，从而使程序在不同执行模式下都能得到高效的执行。

3.2原生运行时技术

算力原生运行时可实现对底层算力资源的感知和控制，完成原生程序的加载、解析，保障计算任务与本地计算资源的即时互映射，按

需执行。具体来说是完成算力原生平台与各类接入算力网络的平台进行灵活插件式适配，对本地的系统环境、算力内核、内存等计算资源以及供电、温度等环境资源进行注册及管控，保障算力原生程序的部署执行。原生运行时技术向上需解析原生统一程序，向下需动态感知底层硬件，实现原生程序与本地计算资源的即时映射。 统一编程模型通过算力原生接口呈现给开发者，开发者编写的源程序经跨架构编译及优化技术生成原生统一程序，运行时对原生统一程序完成加载、解析后，可将程序代码与本地计算资源完成映射后，实现算力原生程序的执行与结果输出。

原生运行时映射技术分为直接映射与间接映射，直接映射可独立完成并行任务向异构平台的映射，间接映射需借助其他异构系统协助完成任务映射。

原生运行时支持任务的机制包括：执行位置和执行顺序，同时负责对映射机制的优化。包含资源的注册及管控、计算任务再映射和自动分配机制、细粒度任务分配、存储与数据之间的通信、共享和同步等。

三、算力原生技术挑战和推进建议

（一）技术挑战

一是在高动态异构统一算力抽象方面存在挑战，在构建资源池时，为利用现有异构算力资源形成大算力供应，就需针对多厂家多类型的算力架构，抽取共性需求，在运行时或驱动或指令集层面构建一套可兼容不同异构算力的抽象模型和统一描述异构硬件能力特征，亟需拉

通业界合作伙伴达成共识，形成业界公认的跨异构硬件操作及编程范式。

二是在统一编程模型转译及综合优化方面存在挑战，多样异构算力架构间存在生态隔离的情况，不同厂家不同类型异构资源的工具链后端是不一致的，需提取共性实现将应用代码编译和转换形成统一的屏蔽底层硬件差异的可跨架构互识和流转的中间原生程序，提高应用程序的可移植性，其中还涉及到编译综合优化技术，在提高应用性能方面尤为重要。

三是在算力任务高动态互映射方面存在挑战，需要通过运行时实现对底层多样算力资源的感知和控制，将算力原生编译器生成的中间元语程序，原生运行时实现中间元语程序的加载、解析，保障计算任务与多样本地计算资源的即时互映射、按需执行，实现和保障应用在任意异构硬件的运行。

（二）推进建议

在标准化方面的建议：后续分别对算力原生的总体技术要求、跨架构编译及优化、算力原生抽象、算力原生运行时等方向持续推进研究与标准化制定工作。可能涉及的算力原生标准化领域包含但不限于以下内容：总体技术要求类：《算力原生总体技术要求》、《算力原生总体测试规范》等；跨架构编译及优化类：《算力原生跨架构编译及优化技术要求》等；算力原生抽象类：《算力原生统一抽象API规范标准》、《算力原生统一编程规范》等；算力原生运行时类：《算力原生运行时接口规范》等。

在技术落地方面的建议：需联合产学研各界进一步拓展算力原生技术的应用场景，推广应用案例，从需求角度推动算力原生技术的落地应用，并在应用中不断发掘新需求特性。

四、展望和产业诉求

中国移动诚邀产学研各界合作伙伴精诚合作，凝聚产业力量，共同推进算力原生技术架构发展，体系化推进算力原生关键技术成熟，通过技术攻关和试点验证、技术标准、开源生态、等多种途径，与产业伙伴一道，共筑算力原生生态创新发展。

攻关算力原生关键技术，展开试点验证工作。联合攻关跨架构编译及优化技术、算力抽象技术和算力原生运行时技术，并基于中国移动算力网络试验示范网项目，开展算力原生原型的试验试点验证工作，实现产业化应用。

联合产学研各界伙伴，推动算力原生标准化建设。联合推进算力原生在 ITU、CCSA 等组织的标准化工作，与合作伙伴共同制定标准化的统一编程范式和算力原生中间元语程序规范，融通泛在算力生态。

秉持开放共享的理念，打造算力原生开源社区。已在算力网络开源社区成立算力原生子工作组，希望联合产学研各界合作伙伴，定期在统一算力抽象模型、算力原生中间元语、算力原生编译平台和算力原生运行时开展技术交流与分享会，共同输出全球首个跨架构算力原生开源平台代码实现，打造算力原生开源开放产业生态。

参考文献

[1]算力网络白皮书[R]，中国移动，2021

[2]中国算力发展指数白皮书[R]，中国信息通信研究院，2021

[3]新型数据中心发展三年行动计划（2021-2023 年），中国工业和信息化部，2021

[4]算力网络技术白皮书[R]，中国移动，2022

[5]面向智算的算力原生技术白皮书[R]，中国移动研究院，2022

缩略语

CPU	Central Processing Unit	中央处理器
GPU	Graphics Processing Unit	图形处理器
AR/VR	Augmented Reality/Virtual Reality	虚拟现实/增强现实
FPGA	Field Programmable Gat Array	现场可编程逻辑门阵列
DMA	Direct Memory Access	直接存储器访问
AI	Artificial Intelligence	人工智能
HALO	Heterogeneity AwareLowering Optimization	面向深度学习通用加速硬件 的开放接口
ODLA	Open Deep Learning API	异构硬件统一接口规范
SDK	Software Development Kit	软件开发工具包
SIMD	Single Instruction Multiple Data	单指令多数据流
SIMT	Single Instruction Multiple Threads	单指令多线程
CUDA	Compute Unified Device Architecture	显卡厂商NVIDIA推出 的运算平台

算网融合产业及标准推进委员会（TC621）

地址：北京市海淀区花园北路 52 号

邮编：100191

电话：15727368875

传真：010-62304980

网址：www.ccnis.org.cn