寒武纪软件栈快速入门

版本 *0.1.0*

2021 年 1 月 18 号

Cambricon 寒武纪

目录

目录 i

插图目录 1

表格目录 2

1 软件栈架构总览 3

2 软件栈用户文档说明 5

2.1 用户文档分类说明 . . . 5

2.2 模块及用户手册对照表 . . . 5

3 软件栈安装部署 10

4 AI 框架和开源生态各模块概述 11

4.1 Caffe . . . 11

4.2 PyTorch . . . 11

4.3 TensorFlow . . . 12

4.4 Horovod . . . 12

5 寒武纪加速库各模块概述 13

5.1 CNML . . . 13

5.2 CNNL . . . 13

5.3 CNPlugin . . . 13

5.4 CNCL . . . 13

6 BANG 异构计算平台各模块概述 15

6.1 CNCC . . . 15

6.2 CNAS . . . 15

6.3 BANGPy . . . 15

6.4 CNStudio . . . 15

6.5 CNGDB . . . 16

6.6 CNDrv . . . 16

6.7 CNRT . . . 16

6.8 CNCodec 16
6.9 CNDev 16
6.10 CNPerf 17
6.11 CNPAPI 17

7 MLU 多平台驱动各模块概述 18

7.1 Driver 18
7.2 CNMon 18
7.3 CNVirt 18

插图目录

1.1 Neuware 软件栈架构图 3

表格目录

2.1 模块及用户手册对照表 5

1 软件栈架构总览

寒武纪 Neuware 软件栈架构如下图所示：

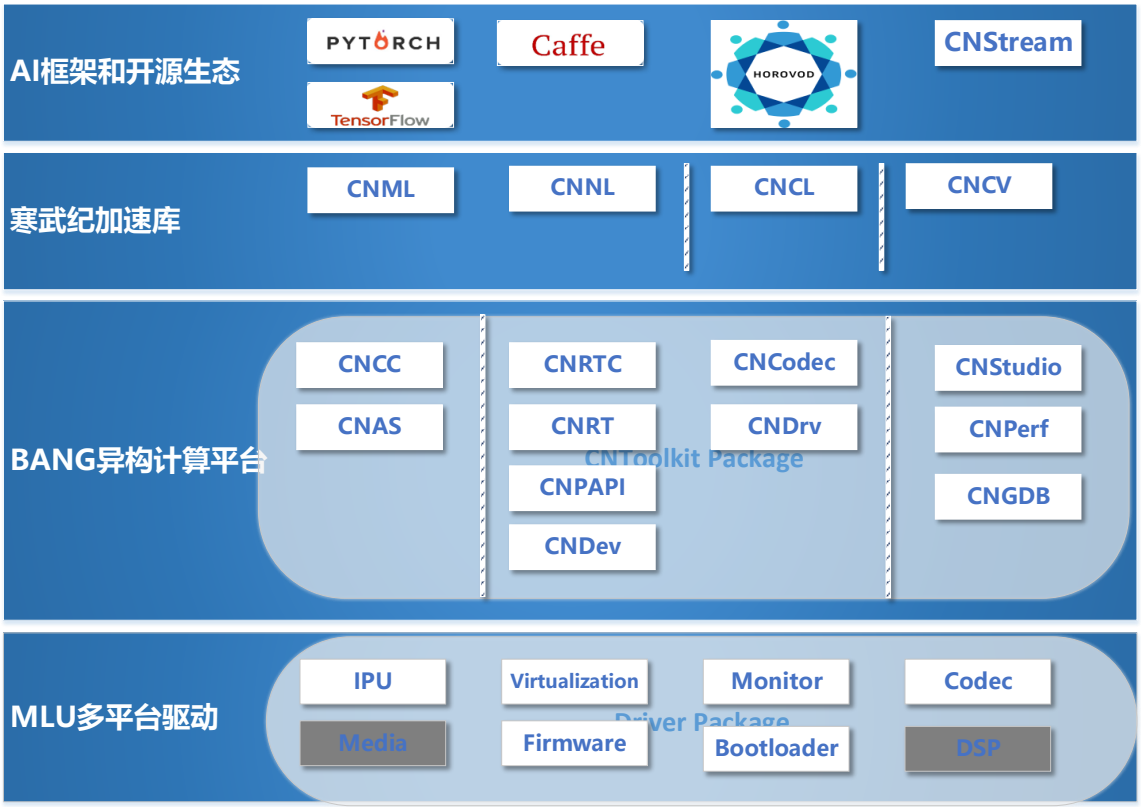

图 1.1: Neuware 软件栈架构图

- **AI 框架和开源生态**

  该层以 AI 为中心，持续利用开源软件生态支撑上层解决方案。该层在开源软件生态的基础上，使其更适合寒武纪 MLU 硬件平台。AI 框架和开源生态各模块概述参见AI 框架和开源生态各模块概述 章节。

- **寒武纪加速库**

  该层为寒武纪基于 BANG 异构计算平台开发的自研加速库，包含了自研算子库、推理引擎和通信库等。寒武纪自研加速库按照行业生态设计和发布，为上层框架和用户提供了适用于寒武纪 MLU 硬件平台的加速库接口。寒武纪加速库各模块概述参见寒武纪加速库各模块概述 章节。

- **BANG 异构计算平台**

  该层为寒武纪 BANG 异构计算平台，不仅提供了异构和并行的计算能力，还提供了功能、性能相关的工

具集。该层统一了编程模型，以 CNToolkit 软件包的形式分发，简化了用户基于寒武纪 MLU（Machine Learning Unit，机器学习单元）硬件平台的开发，为加速库或者框架提供了高效的开发能力。该层各模块概述参见BANG 异构计算平台各模块概述 章节。

- **MLU 多平台驱动**

  该层屏蔽了种类繁多的硬件细节，以统一的 Drvier 软件包的形式分发，为云侧、边侧、端侧不同的硬件抽象出统一的驱动策略和部署方法，为上层软件使用。驱动层各模块概述参见MLU 多平台驱动各模块概述 章节。

2 软件栈用户文档说明

2.1 用户文档分类说明

寒武纪用户文档主要分为四类：用户手册、开发者手册、安装指导书、版本说明书。

- **用户手册**
  主要描述该模块的概念、原理、安装部署、应用场景等相关内容，让用户对该模块有一个整体的认识，以方便用户使用该模块。
- **开发者手册**
  即 API 开发指南，主要描述了使用该模块开发用到的数据类型、API 接口以及编程指南等相关内容。有对外接口的模块才有开发者手册，开发者手册会详细的描述每一个接口的含义、使用场景、使用约束等内容，以帮助用户更快的使用该模块进行开发。开发者手册可能和用户手册成对出现，比如 CNRT 用户手册和 CNRT 开发者手册，CNML 用户手册和 CNML 开发者手册。
- **安装指导书**
  寒武纪软件栈只有驱动、CNToolkit 有独立的安装指导书，其它模块的安装部署请参考各模块的用户手册。
- **版本说明书**
  即 release notes，描述了每个模块每个版本的特性变更、已修复问题、已知遗留问题等内容。用户可以通过版本说明书，了解该模块的版本变化，决定是否需要升级。

2.2 模块及用户手册对照表

表 2.1: 模块及用户手册对照表

软件包	模块缩写	中文手册名称	英文手册名称	手册说明
驱动	Driver	寒武纪 Linux 驱动安装手册-vx.x.x.pdf	Cambricon-Driver-User-Guide-CN-vx.x.x.pdf	主要描述了寒武纪驱动在不同操作系统安装卸载以及升级的步骤。

		寒武纪 Windows 驱动安装手册-vx.x.x.pdf	Cambricon-Driver-Install-Guide-Win-CN-vx.x.x.pdf	
	CNMon	寒武纪 CNMON 用户手册-vx.x.x.pdf	Cambricon-CNMON-User-Guide-CN-vx.x.x.pdf	该手册描述了 CNMon 工具的主要功能，可获得设备型号、驱动版本号、内存开销、设备温度和板卡温度等硬件信息。
	CNVirt	寒武纪虚拟化用户手册-vx.x.x.pdf	Cambricon-CNVIRT-User-Guide-CN-vx.x.x.pdf	该手册描述了寒武纪虚拟化技术的工作原理，多个虚拟机如何共享一张单卡资源，以及虚拟机的安装、卸载和迁移。
CNToolkit	BANG C	寒武纪 BANGC 开发者手册-vx.x.x.pdf	Cambricon-BANGC-Developer-Guide-EN-vx.x.x.pdf	该手册描述了 BANG C 编程模型，BANG C 编程接口，如何调试 BANG C 程序，以及 BANG C 性能优化。
		寒武纪 BANGC 性能调优指南-vx.x.x.pdf	Cambricon-BANGC-Best-Practices-Guide-EN-vx.x.x.pdf	该手册从访存、向量、编译等几个场景详细描述了如何优化 BANG C 程序，使 BANG C 程序运行效率和性能达到最优化。
	BANGPy	寒武纪 BANGPy 用户手册-vx.x.x.pdf	Cambricon-BANGPy-User-Guide-CN-vx.x.x.pdf	该手册提供了基于 Python 的编程框架，通过面向张量的编程为 MLU 系列硬件平台提供了便捷的程序开发接口。
	CNGDB	寒武纪 CNGDB 使用手册-vx.x.x.pdf	Cambricon-CNGDB-User-Guide-CN-vx.x.x.pdf	该手册讲解了如何使用 CNGDB 调试运行寒武纪硬件上的程序。
	CNStudio	寒武纪 CNStudio 使用手册-vx.x.x.pdf	Cambricon-CNStudio-User-Guide-CN-vx.x.x.pdf	该手册讲解了 CNStudio 作为 VSCode 的插件如何编辑、格式化 BANG C 程序，使 BANG C 的开发更加方便。

	CNDrv	寒武纪驱动 API 开发者手册-vx.x.x.pdf	Cambricon-Driver-API-Developer-Guide-CN-vx.x.x.pdf	该手册讲解了驱动对外提供的设备管理和内存管理等接口，比如驱动版本号、设备功耗、内存拷贝等接口。
	CNRT	寒武纪 CNRT 用户手册-vx.x.x.pdf	Cambricon-CNRT-User-Guide-CN-vx.x.x.pdf	该手册讲解了寒武纪运行时库的工作原理，重点讲解了如何编写及加载、运行离线模型，并配有示例说明。
		寒武纪 CNRT 开发者手册-vx.x.x.pdf	Cambricon-CNRT-Developer-Guide-EN-vx.x.x.pdf	该手册描述了 CNRT 提供的对外接口。
	CNCodec	寒武纪 CNCodec 开发者手册-vx.x.x.pdf	Cambricon-CNCodec-Developer-Guide-CN-vx.x.x.pdf	该手册描述了 CNCodec 的基本功能，支持的视频以及图片的格式和规格约束，同时描述了提供的编解码接口以及示例说明。
	CNDev	寒武纪 CNDEV 开发者手册-vx.x.x.pdf	Cambricon-CNDEV-Developer-Guide-CN-vx.x.x.pdf	该手册描述了寒武纪提供的设备接口，通过这些接口能够得到板卡的健康状态，支持的编解码信息，设备的拓扑结构等更详细的硬件信息。
	CNPAPI	寒武纪 CNPAPI 开发者手册-vx.x.x.pdf	Cambricon-CNPAPI-Developer-Guide-CN-vx.x.x.pdf	该手册描述了寒武纪对外提供的性能分析接口。
	CNPerf	寒武纪 CNPERF 开发者手册-vx.x.x.pdf	Cambricon-CNPERF-User-Guide-CN-vx.x.x.pdf	该手册介绍了工具的使用以及提供的详细功能，比如 CNRT 函数运行性能信息，网络运行的性能瓶颈等。
寒武纪加速库	CNML	寒武纪 CNML 用户手册-vx.x.x.pdf	Cambricon-CNML-User-Guide-CN-vx.x.x.pdf	该手册描述了 CNML 支持的算子，支持的运行模型：在线和离线、逐层和融合，以及如何利用 CNML 进行编程。

		寒武纪 CNML 开发者手册-vx.x.x.pdf	Cambricon-CNML-Developer-Guide-EN-vx.x.x.pdf	该手册描述了 CNML 提供的对外接口。
	CNPlugin	寒武纪 CNPlugin 用户手册-vx.x.x.pdf	Cambricon-CNPlugin-User-Guide-CN-vx.x.x.pdf	该手册描述了 CNPlugin 支持的算子，以及用如何开发 CNPlugin 算子，并注册到 CNML。
		寒武纪 CNPlugin 开发者手册-vx.x.x.pdf	Cambricon-CNPlugin-Developer-Guide-EN-vx.x.x.pdf	该手册描述了 CNPlugin 提供的对外接口。
	CNNL	寒武纪 CNNL 用户手册-vx.x.x.pdf	Cambricon-CNNL-User-Guide-CN-vx.x.x.pdf	该手册描述了 CNNL 提供的算子，以及如何用 CNNL 进行编程。
		寒武纪 CNNL 开发者手册-vx.x.x.pdf	Cambricon-CNNL-Developer-Guide-EN-vx.x.x.pdf	该手册描述了 CNNL 提供的对外接口。
	CNCL	寒武纪 CNCL 用户手册-vx.x.x.pdf	Cambricon-CNCL-User-Guide-CN-vx.x.x.pdf	该手册描述了 CNCL 多机多卡通讯原理以及对外接口。
AI 框架和开源生态	Caffe	寒武纪 Caffe 用户手册-vx.x.x.pdf	Cambricon-Caffe-User-Guide-CN-vx.x.x.pdf	该手册描述了寒武纪 Caffe 支持的算子，通过快速入门介绍了运行网络模型的步骤。
	PyTorch	寒武纪 Pytorch 用户手册-vx.x.x.pdf	Cambricon-PyTorch-User-Guide-CN-vx.x.x.pdf	该手册描述了寒武纪 Pythorch 支持的算子，通过快速入门介绍了运行网络模型的步骤。
	TensorFlow	寒武纪 TensorFlow 用户手册-vx.x.x.pdf	Cambricon-TensorFlow-User-Guide-CN-vx.x.x.pdf	该手册描述了寒武纪 Pythorch 支持的算子，通过快速入门介绍了运行网络模型的步骤。

	Horovod	寒武纪 Horovod 用户手册-vx.x.x.pdf	Cambricon-Horovod-User-Guide-CN-vx.x.x.pdf	该手册描述了 Horovod 的概念，编译安装以及如何在其它框架中使用 Horovod。

3 软件栈安装部署

根据寒武纪 Neuware 软件栈架构图，驱动和 CNToolkit 为上层软件依赖库，需要优先安装，因此软件栈安装部署步骤如下所示。

注解：

文件名中的 x.x.x 指当前版本号。

- **步骤一、安装寒武纪驱动**

  寒武纪驱动兼容 Linux 系统和 Windows 系统，安装寒武纪驱动前请先获得寒武纪驱动安装包。

  – Linux 系统驱动安装步骤请参考《Cambricon-Driver-User-Guide-CN-vx.x.x.pdf》手册，中文名称为《寒武纪 Linux 驱动安装手册-vx.x.x.pdf》。

  – Widnows 系统安装步骤请参考《Cambricon-Driver-Install-Guide-Win-CN-vx.x.x.pdf》，中文名称为《寒武纪 Windows 驱动安装手册-vx.x.x.pdf》。

- **步骤二、安装 CNToolkit 软件包**

  安装 CNToolkit 软件包前请先获得 CNToolkit 安装包。

  安装 CNToolkit 软件包请参考《Cambricon-CNToolkit-Installation-And-Updation-Guide-CN-vx.x.x.pdf》，中文名称为《寒武纪 CNToolkit 安装升级使用手册-vx.x.x.pdf》。

- **步骤三、安装寒武纪加速库**

  寒武纪加速库包含的各个模块为独立的安装包，因此安装加速库请参考各自的用户手册，比如安装 CNML，请参考 CNML 用户手册，各个加速库的手册名称参见模块及用户手册对照表 的描述。

- **步骤四、安装 AI 框架和开源生态库**

  寒武纪 AI 框架和开源生态库包含的各个模块为独立的安装包，因此安装 AI 框架和开源生态库请参考各自的用户手册，比如安装 TensorFlow，请参考 TensorFlow 用户手册，各个库的手册名称参见模块及用户手册对照表 的描述。

4 AI 框架和开源生态各模块概述

以下内容为 AI 框架和开源生态涉及的各模块概述。

4.1 Caffe

Caffe 的全称是 Convolutional Architecture for Fast Feature Embedding，是一款开源深度学习编程框架，用以实现处理器并行架构下的深度卷积神经网络以及深度学习应用。

为支持寒武纪 MLU，寒武纪定制了开源深度学习编程框架 Caffe（以下简称 Cambricon Caffe）。Cambricon Caffe 兼容 BVLC（Berkeley Vision and Learning Center）Caffe 的 Python/C++ 编程接口和 BVLC Caffe 网络模型，增加了离线、多核正向推理功能。应用程序开发人员可以使用 Python 接口或 C/C++ 接口直接调用 Caffe 数据结构和函数来完成各种正向推理任务。Cambricon Caffe 不仅支持 float32 和 float16 网络模型，而且在寒武纪 MLU 上高效支持 int8 和 int16 网络模型。

4.2 PyTorch

PyTorch 是一款 Facebook 开源的深度学习编程框架，适用于 Python、C++ 等编程语言，用以实现高效的 GPU 并行计算及深度学习网络搭建，具有轻松扩展、快速实现、生产部署稳定性强等优点。

为支持寒武纪 MLU，寒武纪定制了开源深度学习编程框架 PyTorch（以下简称 Cambricon PyTorch）。Cambricon PyTorch 兼容原生 PyTorch 的 Python 编程接口和原生 PyTorch 网络模型，增加了离线、多核正向推理功能。应用程序开发人员可以使用 Python 接口来完成各种正向推理任务。正向推理时，网络的权重可以从 pth 格式文件中读取，已支持的分类和检测网络结构由 torchvision 管理，可以从 torchvision 中读取。Cambricon PyTorch 不仅支持 float16、float32 等网络模型，而且在寒武纪机器学习处理器上能高效地支持 int8 和 int16 网络模型。

4.3 TensorFlow

TensorFlow 是一款基于数据流的编程框架，广泛应用于各类机器学习、深度神经网络，具有很高的可移植性及灵活性。

为支持寒武纪 MLU，寒武纪定制了开源深度学习编程框架 TensorFlow（以下简称 Cambricon TensorFlow），在原生 TensorFlow 中增加对 MLU 的支持，实现已有 TensorFlow 模型在 MLU 上快速部署、模型推理和模型训练。Cambricon TensorFlow 支持基于原生 TensorFlow v1.14 进行开发。Cambricon TensorFlow 兼容原生 TensorFlow 的编程接口（Python/C++）和模型文件格式 protobuf（pb），屏蔽了 MLU 硬件的细节，在使用方法上与原生 TensorFlow 基本相同。

4.4 Horovod

Cambricon Horovod 是为了适配寒武纪 MLU 定制的开源深度学习编程，适配了寒武纪定制框架 Cambricon TensorFlow 和 Cambricon PyTorch。Cambricon Horovod 使用寒武纪通信库 CNCL 用于多卡或多机之间的通讯。

5 寒武纪加速库各模块概述

以下内容为寒武纪加速库各模块概述。

5.1 CNML

CNML（Cambricon Neuware Machine Learning Library，寒武纪机器学习库）是一个针对机器学习以及深度学习的编程库，为用户提供简洁、高效、通用、灵活并且可扩展的编程接口，用于在 MLU 上对机器学习算法和深度学习算法进行加速。CNML 不仅提供了丰富的基本算子，用户还可以通过组合基本算子实现多样的机器学习算法和深度学习算法。

5.2 CNNL

CNNL（Cambricon Neuware Neural Network Library，寒武纪神经网络计算库）是一个基于寒武纪 MLU 并针对 DNN（Deep Neural Network，深度神经网络）开发的计算库。CNNL 针对 DNN 应用场景，提供了高度优化的常用算子，同时也为用户提供简洁、高效、通用、灵活并且可扩展的编程接口。

5.3 CNPlugin

CNPlugin 在 CNML 层提供一个接口，将 BANG C 语言生成的算子与 CNML 的执行逻辑统一起来，因此实现了 BANG C 语言对 CNML 的操作进行扩展。用 BANG C 写的算子，通过 CNPlugin 接口注册到 CNML 后，还可以支持 CNML 的特性及多种运行模式如在线、离线，逐层、融合等。

5.4 CNCL

CNCL（Cambricon Neuware Communication Library，寒武纪通信库）是面向寒武纪 MLU 设计的高性能通信库，主要包含以下功能：

- 帮助应用开发者优化了基于 MLU 进行多机多卡的集合通信（Collective）操作。
- 支持多种 MLU 处理芯片的互联技术，包括 PCIe、Interlaken、RoCE、Infiniband Verbs 以及 Sockets。

- 能够根据芯片的互联拓扑关系，自动的选择最优的通信算法和数据传输路径，从而最大化利用传输带宽完成不同的通信操作。

6 BANG 异构计算平台各模块概述

以下内容为 CNToolkit 安装包所包含的模块，以及各模块概述。

6.1 CNCC

CNCC (Cambricon Neuware Compiler Collection) 是寒武纪科技为 MLU 架构设计开发的专用编译器工具链集合，BANG C 语言是寒武纪科技为 MLU 架构设计的类 C/C++ 的编程语言，CNCC 的主要功能是将 BANG C 编程语言的源码编译为 MLU 架构的可执行程序并提供辅助的调试和优化的工具集。

6.2 CNAS

CNAS 负责将 MLISA 语言编译成 MLU 架构可执行程序。MLISA (Machine Learning Instruction Set Architecture) 是寒武纪针对 MLU 硬件提出的一套指令集架构，兼顾通用计算和机器学习高性能计算。

6.3 BANGPy

BANGPy 是一种基于 Python 的编程框架，用于神经网络算子开发与网络搭建，为寒武纪 MLU 系列硬件平台提供便捷的软件接口。BANGPy 可以直接调用 MLU 硬件资源，进行算子开发验证。BANGPy 对 BANG 架构进行了更高层次的抽象与封装，相比于 BANG C 极大地减轻了用户编写算子的工作量，从而提高了算子开发效率。

6.4 CNStudio

CNStudio 是基于 VSCode（Visual Studio Code）的 BANG C 编程插件，利用 VSCode 强大的编辑和可视化操作，使编写 BANGC 更加方便。CNStudio 目前提供的主要功能包含：语法高亮、自动补全、程序调试。

6.5 CNGDB

CNGDB 是寒武纪软件调试工具，基于 GNU 开源项目 GDB 二次开发。该工具能够控制运行在寒武纪硬件上的程序，提供监视程序运行状态及获取、修改程序中间运行结果的功能，可以减轻用户开发过程中的调试负担，提高开发效率。

6.6 CNDrv

CNDrv（Cambricon Neuware Driver Library 寒武纪驱动库）提供了一套面向 MLU 设备的接口，用于主机与 MLU 设备之间的交互。CNDrv 作为寒武纪软件系统的底层支撑，通过对底层硬件的抽象封装，向上提供了用户操作底层硬件的接口，通过该接口实现了上层对底层驱动的操作。

6.7 CNRT

CNRT（Cambricon Neuware Runtime Library，寒武纪运行时库）提供了一套面向 MLU 设备的高级别的接口，用于主机与 MLU 设备之间的交互。CNRT 作为寒武纪软件系统的底层支撑，为上层软件提供了运行时库，支持了离线模型和在线模型的运行。CNRT 在 CNDrv 的上层，通过调用 CNDrv 的接口实现对底层驱动的操作。

6.8 CNCodec

CNCodec（Cambricon Neuware Codec，寒武纪硬件编解码）是一套封装了视频编解码和图片编解码的 SDK 接口。CNCodec 提供了一套 C 语言的 API，支持多路并发的编解码通道。

6.9 CNDev

CNDev（Cambricon Neuware Device Library，寒武纪设备库）是一套支持主机端应用程序获取寒武纪芯片硬件信息的软件接口。通过 CNDev 接口，用户可以对设备实现定制化管理需求，比如云服务平台的设备管理、自动化测试平台的设备管理等。

6.10 CNPerf

CNPerf（Cambricon Neuware Performance 寒武纪性能剖析工具）是一款用户层的性能剖析工具，用于分析网络的性能瓶颈；定位热点函数运行状态，分析程序的性能瓶颈。

6.11 CNPAPI

CNPAPI（Cambricon Neuware Profiling API，寒武纪性能分析接口）是一套面向用户开发定制化性能分析工具的软件接口。使用 CNPAPI 接口，用户可以精确分析 CPU 和 MLU 的行为，对程序进行性能调优。CNPerf 就是调用该接口实现的性能剖析工具。

7 MLU 多平台驱动各模块概述

以下内容为 Driver 安装包所包含的模块，以及各模块概述。

7.1 Driver

在使用寒武纪 MLU 之前需要安装相应的驱动，寒武纪发布的驱动安装包有两种：deb 包和 rpm 包。寒武纪驱动安装包使用 dkms 框架来管理驱动的安装、升级和卸载。dkms 可以针对系统内安装的每个内核版本，通过编译、安装相应的内核模块来实现驱动的加载。用户只需安装、更新相应的 dkms 软件包（deb 包或 rpm 包）即可实现驱动的安装和升级。

7.2 CNMon

CNMon（Cambricon Neuware Monitor，寒武纪硬件监测和控制工具）通过调用 CNDev 接口获取底层硬件信息。CNMon 不仅可以采集底层硬件信息，还可以实时获取上层软件对硬件资源的开销，为用户实时显示当前底层硬件的详细信息和状态。

7.3 CNVirt

CNVirt 是寒武纪 MLU 硬件虚拟化技术的简称。CNVirt 支持多个虚拟机共享一张单卡资源，每个虚拟机拥有各自单独、隔离的物理资源，可以互不影响的并行执行任务。

