JUNIPER NETWORKS | Engineering Simplicity

# 为 AI 集群设计数据中心

作者： Aninda Chatterjee, Vivek V

## 关于本文档

本文档是为高性能 AI 集群构建网络基础设施的通用设计文档。

## 文档反馈

我们鼓励您提供反馈，以便我们改进文档。
请将您的意见发送至 design-center-comments@juniper.net。请附上文档或主题名称、URL 或页码，以及软件版本（如适用）。design-center-comments@juniper.net.

# 目录

# 引言

人工智能（AI）在过去十年中迅速发展，对支持研究、开发和部署的 AI 集群的需求也随之增长。AI 集群可以构建为各种规模，以满足特定的需求和工作负载。在本文档中，我们探讨 AI 集群的网络基础设施需求，并深入分析其业务负载模式（Workload Paradigm）以及瞻博网络数据中心针对 AI 集群的设计方法。

## AI/ML 工作负载与架构

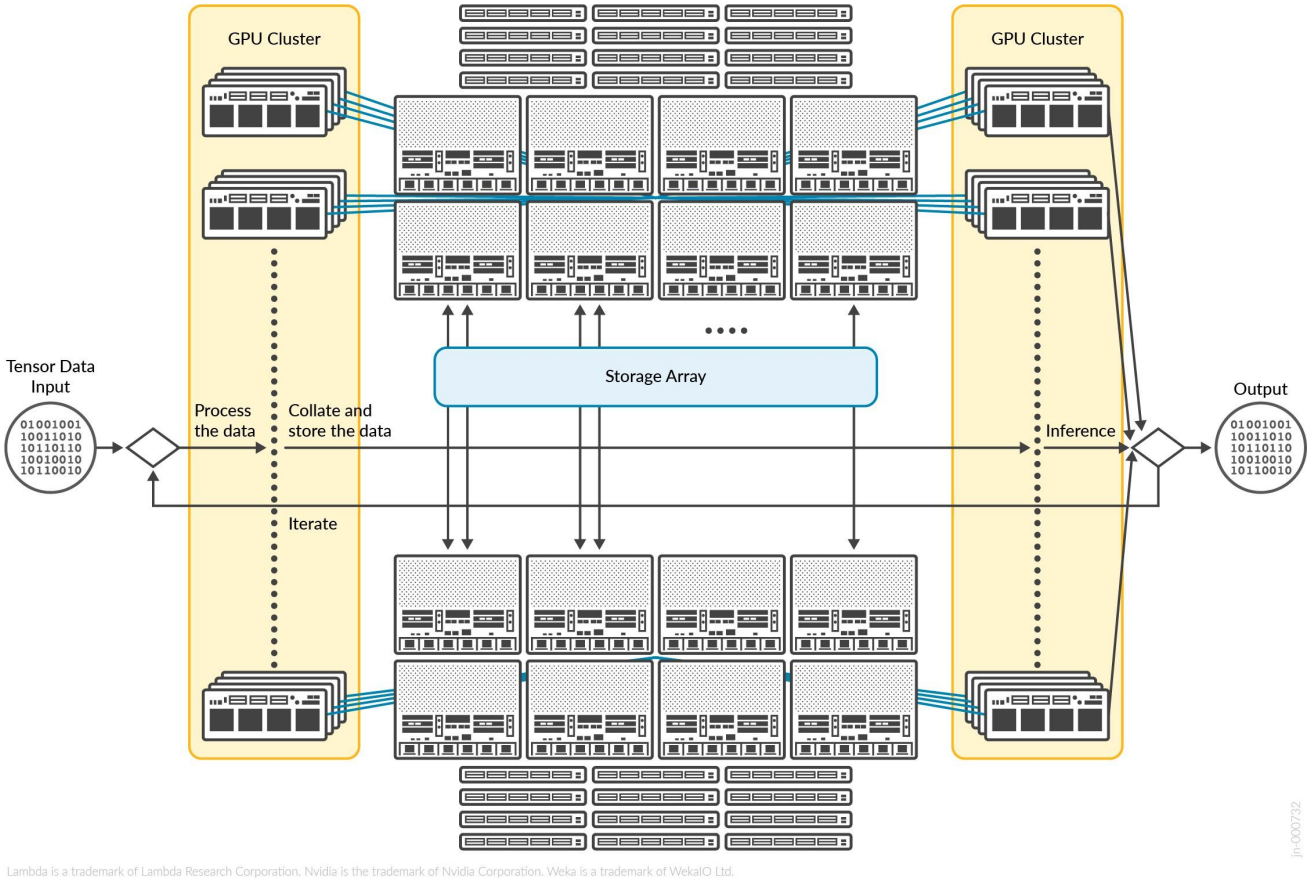
AI/ML 工作负载涵盖广泛的任务和应用，利用人工智能和机器学习技术来分析、理解和预测数据。这些工作负载是许多现代技术进步与应用的核心，通常是网络、存储和计算密集型。一些常见的工作负载类型包括：

- **监督学习(Supervised Learning)——在监督学习中，模型使用带标签的数据集进行训练，每个输入数据点都与已知的目标或标签相关联。**
- **无监督学习(Unsupervised Learning)——无监督学习涉及处理无标签数据，模型在没有明确指导的情况下学习数据中的模式和结构。**
- **强化学习(Reinforcement Learning)——强化学习涉及训练智能体在环境中做出一系列决策，以最大化奖励信号。**
- **深度学习（Deep Learning）——深度学习是机器学习的一个子集，专注于多层神经网络（深度神经网络）。**
- **自然语言处理（Natural Language Processing, NLP）——自然语言处理任务涉及对人类语言的处理与理解，使机器能够与文本或语音数据进行交互。**
- **计算机视觉（Computer Vision）——计算机视觉任务涉及对视觉数据（如图像和视频）的理解与解释，以识别对象、模式和场景。**
- **时间序列分析(Time Series Analysis)——时间序列分析关注随时间变化的数据，涉及基于历史数据建模并预测未来值。**
- **推荐系统(Recommendation Systems)——推荐系统使用AI/ML根据用户的偏好、行为或历史数据向用户推荐项目或内容。**
- **生成模型(Generative Models)——生成模型旨在生成与现有数据相似的新数据。**
- **异常检测(Anomaly Detection)——异常检测工作负载专注于识别数据中偏离预期行为的罕见或异常模式。**

与传统网络架构相比，AI/ML集群网络的架构是反直觉的。

虽然图1将整个集群显示为一个连续的网段，但实际上是三个独立的网段，每个网段仅为集群的特定方面提供服务。图1展示了AI/ML数据流。每个GPU通过GPU网络端口与后端网络通信。承载GPU的服务器有一个专用的存储网络端口，通过存储网络连接到外部存储阵列。因此，GPU/计算节点通过GPU服务器自带的存储专用接口与存储节点通信，无需跨网络跳转。图1在相应部分进一步描述了这些网段。

图1：GPU训练与推理过程的High-Level流程

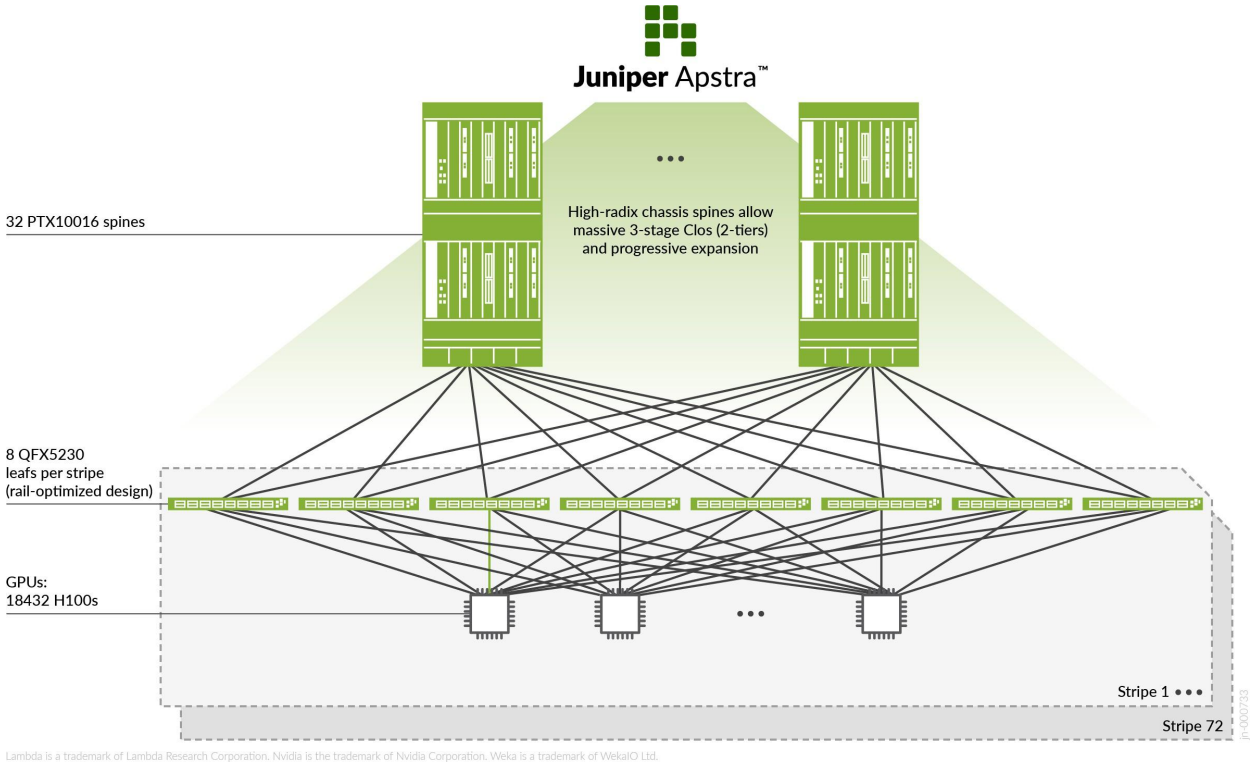



# AI/ML集群规模

AI/ML集群的规模因工作负载的具体需求以及满足这些需求所需的规模而有很大差异。AI/ML集群中的节点数量取决于机器学习模型的复杂度、数据集大小、期望的训练速度以及可用预算等因素。节点数量可能从一个小型集群到包含数千个计算、存储和网络节点的数据中心级集群不等。本文档仅涵盖规模缩小版的 AI 集群。

集群大小取决于基础设施中存在的终端节点数量。本文档介绍了针对不同集群规模的设计考虑和基础设施要素。图2展示了一个大规模集群。

图2: AI/ML集群规模

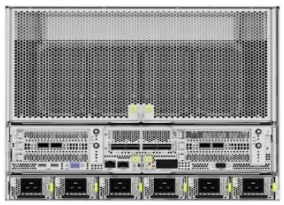



# 集群基础设施

本节概述了本设计指南中使用的节点（网络、计算和存储）。该设计指南以Juniper QFX5220-32CD、QFX5230-64CD和PTX设备作为网络基础设施，采用NVIDIA A100和H100 GPU进行计算，并使用Weka存储节点。表1给出了这些节点的High-Level概览。

表1：Juniper用于集群设计的基础设施节点

| | |
|---|---|
| QFX5220-32CD<br>  <br>IP FABRIC交换机：可作为Leaf交换机或Spine交换机，扩展能力高达 12.8 Tbps。 | 端口：32 x 400GbE、64 x 200GbE、128 x 100GbE、32 x 40GbE、128 x 25GbE、128 x 10 GbE<br><br>操作系统：Junos OS Evolved<br><br>尺寸：1U 机架安装，宽 x 高 x 深 17.26 x 1.72 x 21.1 英寸（43.8 x 4.3 x 53.59 厘米） |
| QFX5230-64CD<br>  <br>IP FABRIC交换机：可作为Leaf交换机或Spine交换机，扩展能力高达 25.6 Tbps | 端口：最多 64 × 400GbE、128 × 200GbE、256 × 100GbE、64 × 40GbE、256 × 25GbE、256 × 10GbE 端口<br><br>操作系统：Junos OS Evolved<br><br>尺寸：2U 机架安装，宽 x 高 x 深 17.4 x 3.43 x 25.6 英寸（44.2 x 57.76 x 81.28 厘米） |

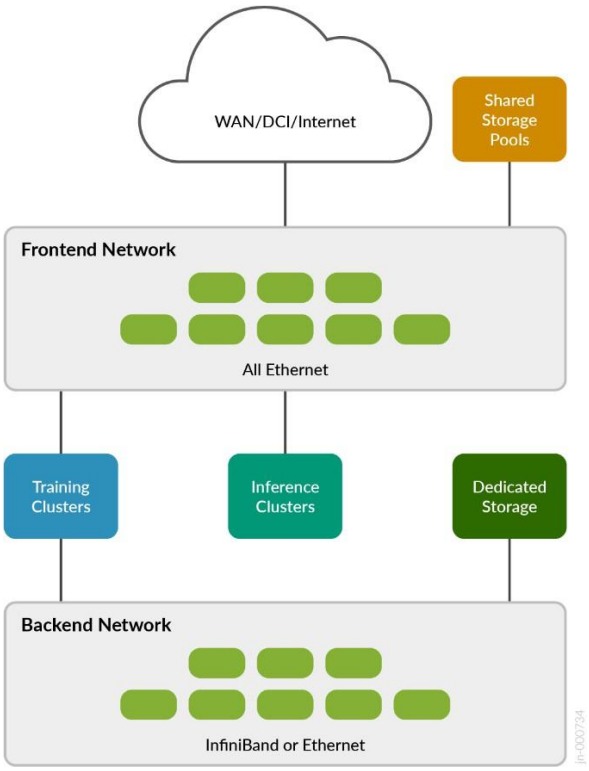
| | |
|---|---|
| PTX10008<br>  <br>IP FABRIC交换机：可作为Spine交换机，扩展能力高达 115.2 Tbps。 | 端口：最多 288 个 400GbE，1152 个 100GbE，288 个 40GbE，1152 个<br>25GbE，1152 个 10GbE<br><br>操作系统：Junos OS Evolved<br><br>尺寸：13U 机架式，17.4 x 22.55 x 32 英寸（44.2 x 57.76 x 81.28 厘米）宽 x 高 x 深 |
| Lambda Hyperplane8 HGX-A100<br>  <br>标准网络：1x NVIDIA ConnectX-6 Dx 适配卡，100GbE，双端口 QSFP28，AIOM PCIe 4.0 x16<br><br>存储网络：1x 200 Gbps NVIDIA ConnectX-6 VPI NIC：双端口 QSFP56，HDR InfiniBand/以太网<br><br>GPU Direct RDMA 网络：8x NVIDIA ConnectX-7 适配卡 200Gb/s NDR200 IB 单端口 QSFP PCIe 4.0 x16 | 操作系统：Ubuntu 22.04：包含用于管理 TensorFlow、PyTorch、CUDA、cuDNN 等的 Lambda Stack<br><br>处理器：2x AMD EPYC 7763：64 核，2.45~3.5 GHz，256MB 缓存，PCIe 4.0<br><br>GPU：8x NVIDIA A100 (80GB) SXM4：采用 NVLink 和 NVSwitch 互联结构的 HGX 平台<br><br>系统盘：2x 1.92 TB M.2 NVMe：数据中心级 SSD，1 DWPD，PCIe 4.0<br><br>数据盘：6x 3.84 TB U.2 NVMe：数据中心级 SSD，1 DWPD，PCIe 4.0<br><br>尺寸：4U 机架式，6.9 x 17.6 x 35.4 英寸（174 x 446 x 900 毫米）高 x 宽 x 深 |
| NVIDIA DGX H100<br>  <br>网络（GPU）：4 个 OSFP 端口，支持 8 个单端口 NVIDIA ConnectX-7 VPI：400 Gb/s InfiniBand/以太网<br><br>网络（存储）：2 个双端口 NVIDIA ConnectX-7 VPI：1x 400 Gb/s InfiniBand/以太网 | DGX H100 系统：DGX H100 系统，80GB，10 NIC，标准支持，3 年<br><br>GPU：8x NVIDIA H100 80GB Tensor Core GPU，配备 NVLink 和 NVSwitch 互联结构<br><br>处理器：双 Intel Xeon Platinum 8480C 处理器：共 112 核，2.00 GHz（基础频率），3.80 GHz（最大加速频率），内存：2TB DDR5<br><br>系统盘：2x 1.92TB M.2 NVMe 驱动器；内部存储：8x 3.84TB U.2 NVMe 驱动器<br>尺寸：8U 机架式，14 x 19.0 x 35.3 英寸（174 x 446 x 900 毫米）高 x 宽 x 深 |

| | |
|---|---|
| Weka 存储节点<br>  <br>前端网络：1x NVIDIA ConnectX-6 Dx 适配器卡，100GbE，双端口 QSFP28，PCIe 4.0 x16<br><br>存储网络：2x NVIDIA ConnectX-6 VPI 适配器卡，HDR IB (200Gb/s) 和 200GbE，双端口 QSFP56，OCP 3.0 | 操作系统：1x Ubuntu | 22.04<br><br>处理器：1x AMD EPYC 9454P（48 核，2.75~3.8GHz，256MB 缓存，290W）<br><br>系统盘：2x 1.92 TB M.2 NVMe：数据中心级 SSD，1 DWPD，PCIe 4.0<br><br>数据盘：7x 7.68 TB U.2 NVMe：数据中心级 SSD，1 DWPD，PCIe 4.0<br><br>软件：3 年期 Weka Flash tier 许可证，包含快照和高性能协议服务 - POSIX + NFS-W + S3 + SMB-W |

表1所示的组件是整体AI/ML集群解剖结构的一部分，如图3所示。其中，前端网络连接到外部用户和数据，后端网络支持训练集群的AI模型训练功能。

## AI/ML集群组件

图3：AI/ML集群解剖结构



集群组件：

- 训练集群
- 推理集群
- 共享存储池
- 专用集群存储

集群网络（Cluster Networks）：

- 前端网络（Frontend Network）：
  - 推理集群使用此网络
  - 共享存储池（Shared storage pools）
  - 用于训练的管理网络
- 后端网络 (Backend)：
  - GPU 计算架构
  - 专用存储架构（可能与计算融合）

---

*注：整个 AI/ML 集群设有独立的智能平台管理接口（IPMI）/带外（OOB）管理网络。*

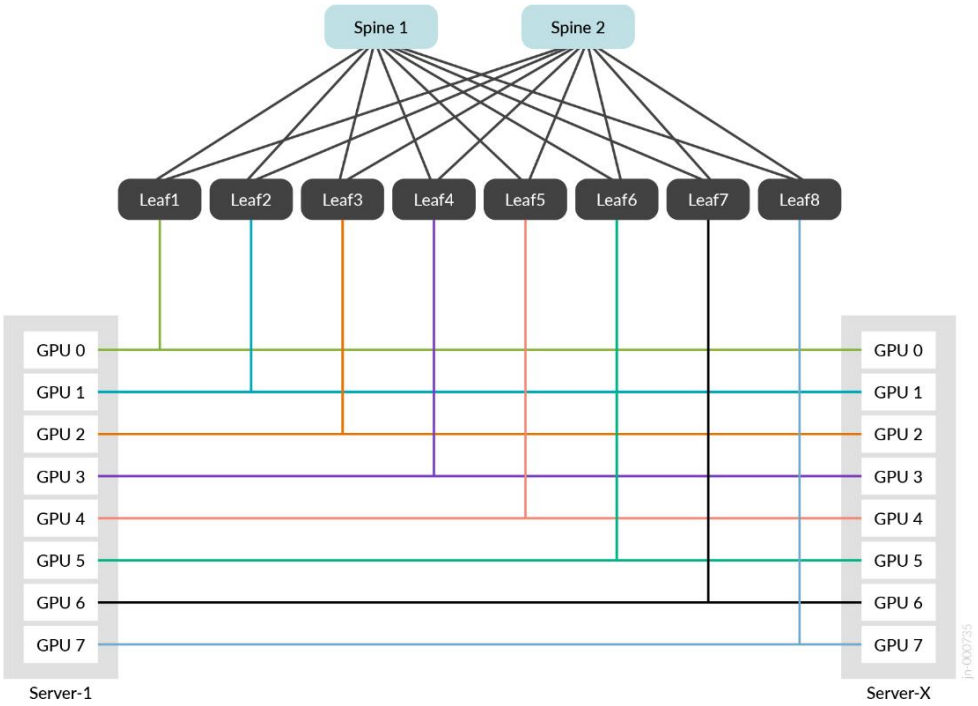
---

# High-Level 设计

以下章节介绍 AI 集群设计的考量要点，主要围绕训练集群（而非推理集群，后者在 GPU 和存储节点的总体设计上可能有所不同）。

## 轨道优化条带设计(Rail-Optimized Stripes)

图4：轨道优化条带设计




在为AI集群设计网络基础设施时，关键目标是为AI流量提供最大吞吐量、最低延迟和最小的网络干扰。NVIDIA提出的轨道优化条带设计（Rail-Optimized Stripe）是一种从计算节点延伸到架顶Leaf交换机的设计，充分考虑了AI集群的网络需求。

从网络基础设施的角度来看，这种设计类似于大多数现代数据中心部署中常见的三层 Clos 架构。例如，对于DGX H100计算服务器（服务器内有8个GPU和8个NIC），服务器内的任意到任意GPU通信通过连接到NVSwitch的高吞吐量NVLink通道实现。除此之外，服务器中的每个NIC连接到唯一的架顶Leaf交换机（NIC1连至leaf1，NIC2连至leaf2，依此类推），所有服务器都遵循相同的方法，从而形成‘轨道’设计。NVIDIA文档提供了有关此类设计的更多信息。

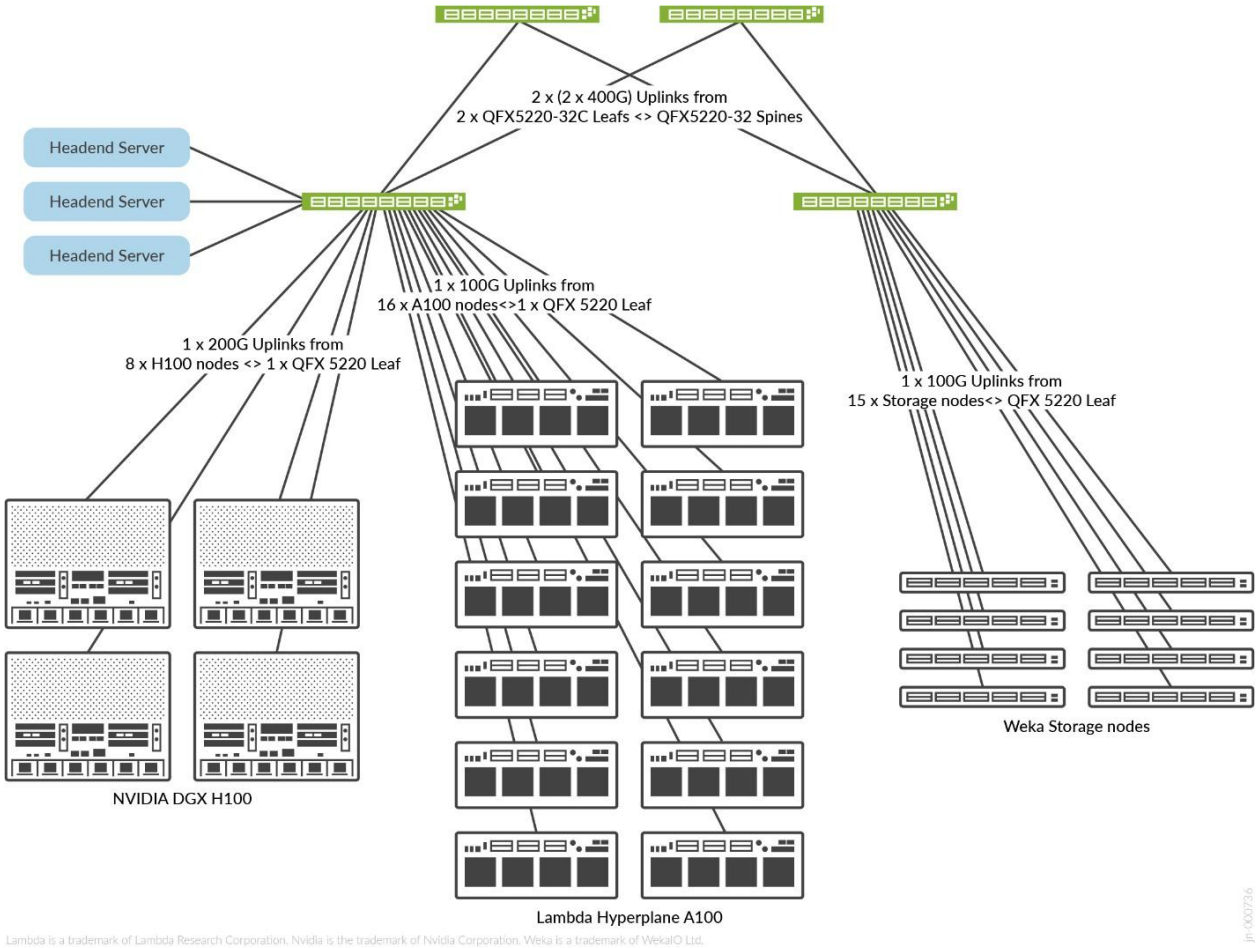
借助这种设计以及PXN（PCIe x NVLink）等优化，网络干扰得以最小化：数据被移动到与目的地位于同一轨道的GPU上，从而无需跨越轨道即可发送数据，最大限度地减少了所需的网络跳数。这种针对轨道优化条带设计的GPU/NIC连接性将在设计与实现章节中详细探讨。

后端网络的High-Level设计使用轨道优化条带设计（Rail-Optimized Stripe）作为基本构建单元，并通过复制相同设计来扩展AI集群规模。

# High-Level前端网络设计

AI集群的前端网络（Front-end Network）为集群外部的用户和数据提供连接。

图5：前端网络架构




前端网络的设计考虑因素包括：

- 前端网络负责处理多种元素，例如训练编排；如果部分 GPU 专用于推理，还可能涉及推理服务，包括处理来自终端用户的应用程序接口（API）调用来提供模型推理结果、遥测等。
- 因此，前端网络预计不会像后端网络那样接收同等数量的网络流量和数据流，后端网络处理训练方法和存储。因此，在前端网络设计中，我们选择将 QFX5220-32CD 同时用作Leaf交换机和Spine交换机，其端口映射如下所述。
- 为使超配比(oversubscription ratio)等于 1，在本设计中，共有 16 条来自 A100 服务器的 100G 链路和 8 条来自 H100 服务器的 200G 链路，满负荷时网络负载为 3200 Gbps。从 QFX5220-32CD Leaf交换机到 QFX5220-32CD Spine交换机的上行链路为每台Spine交换机配置 2 条 2 × 400G 上行链路，使得总上行带宽达到 3200 Gbps，从而维持超配比为 1。

- 所使用的存储节点（Weka 存储节点）每个节点均有 1 条 100G 链路连接到Leaf交换机。为保证设计上的对等，存储Leaf交换机到Spine交换机的上行链路也设置为 3200 Gbps。根据需求，可通过移除上行链路降至 1600 Gbps，或为额外的 15 个存储节点进行配置，以保持超配比为 1。

## High-Level 后端网络设计

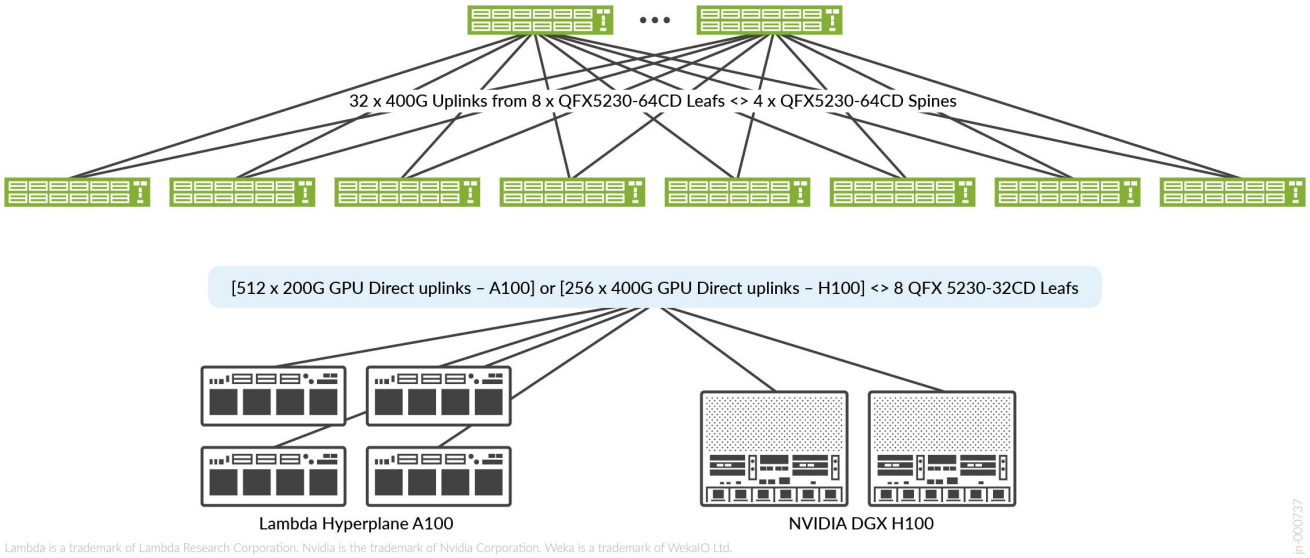
如图 3 所示，AI 集群的后端网络设计可分为两个部分：

- 计算或 GPU 网络。
- 存储网络。

本文档涵盖计算和存储网络的三种示例拓扑变体，其区别体现在所使用设备的角色、端口密度和容量（即集群规模）。这些设计遵循客户用例和 GPU 厂商 1:1 超配比的建议。我们与客户紧密合作，深入了解其训练工作负载的具体网络需求，包括集群规模和可能的超配效率。

### 计算设计 1：以 QFX5230-64CD 作为Spine交换机、QFX5230-64CD 作为Leaf交换机的轨道优化 GPU 条带设计。

图6：采用QFX5230-64CDs作为Spine交换机和Leaf交换机的轨道优化与网络优化设计






虽然设备容量和处理大规模流量的能力是重要的设计考量，但实施成本也同样重要。对于较小规模的集群，高端口密度的Spine交换机并非必需。该设计的设计规格如下：

- 在图6中，每个条带将64个A100 GPU或32个H100 GPU连接到一台Leaf交换机。它展示了基于A100和基于H100的两种GPU集群视图，有助于在选择集群类型时做出更明智的决策。
- A100 GPU的网络端口容量为200 Gbps，而H100 GPU的网络端口容量为400 Gbps。

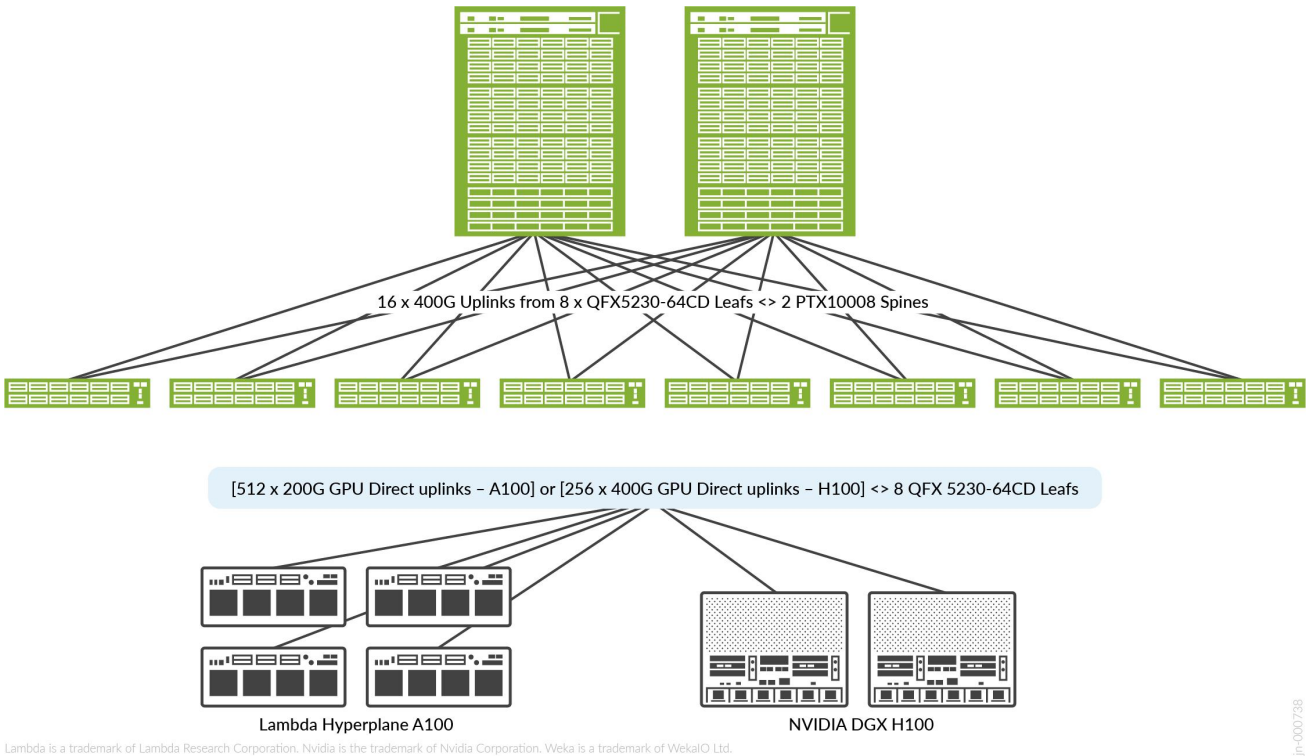
- 因此，每台Leaf交换机在最大负载下接收12.8 Tbps的入站流量。Leaf交换机通过32条400 Gbps链路分别连接至四台Spine交换机，从而提供12.8 Tbps的出站容量，使超配比保持为1。

本设计的规模计算方法如下：

- 每台 QFX5230-64CD Spine交换机具有 64 个 400G 端口。
- Leaf交换机上的 64 个端口划分如下：32 个 400G 上行链路，8 台Leaf交换机共 256 条上行链路，分布在四台Spine交换机之间；以及 32 个 400G 下行端口。这些下行端口进一步拆分为 64 个 200G 端口，可连接 64 个 A100 节点（每节点一个）或 32 个 H100 节点（每节点一个），或两者的混合。
- 因此，每个条带的集群规模为 512 个 A100 GPU（64 个 A100 GPU × 8 台Leaf交换机）或 256 个 H100 GPU。

## 计算设计 2：以 PTX10008 为Spine交换机、QFX5230-64CD 为Leaf交换机的轨道优化 GPU 条带设计

图7：采用QFX5220-32CD作为Leaf交换机、PTX10008作为Spine交换机的轨道优化设计与网络优化设计




该设计中的规模计算如下：

- 在此情况下，一台PTX10008Spine交换机有8个槽位，每个槽位安装一块36端口400G业务板，因此整机共有288个400G端口。
- 每台Leaf交换机提供32个400G上联端口，8台Leaf交换机共计256个上联端口，分配给两台Spine交换机，即每台Spine交换机对应128个端口。
- 据此计算，Spine交换机还剩余160个空闲端口。由于PTX采用模块化机箱，可通过选择合适的槽位数来匹配所需的条带（Stripe）数量。

- 由于Leaf交换机为 QFX5230-64CDs，每个条带的 GPU 节点数为 64 个 A100 节点 × 8 台Leaf交换机 = 每个条带 512 个 A100 GPU 或 256 个 H100 GPU。
- 值得注意的是，模块化Spine交换机通过仅需增加业务板和新线缆，即可随着更多条带加入整个集群而更轻松地逐步扩展，无需重新布线。

# High-Level 存储网络设计

图 8：使用 QFX5220-32CDs 作为Leaf交换机和Spine交换机的后端存储设计

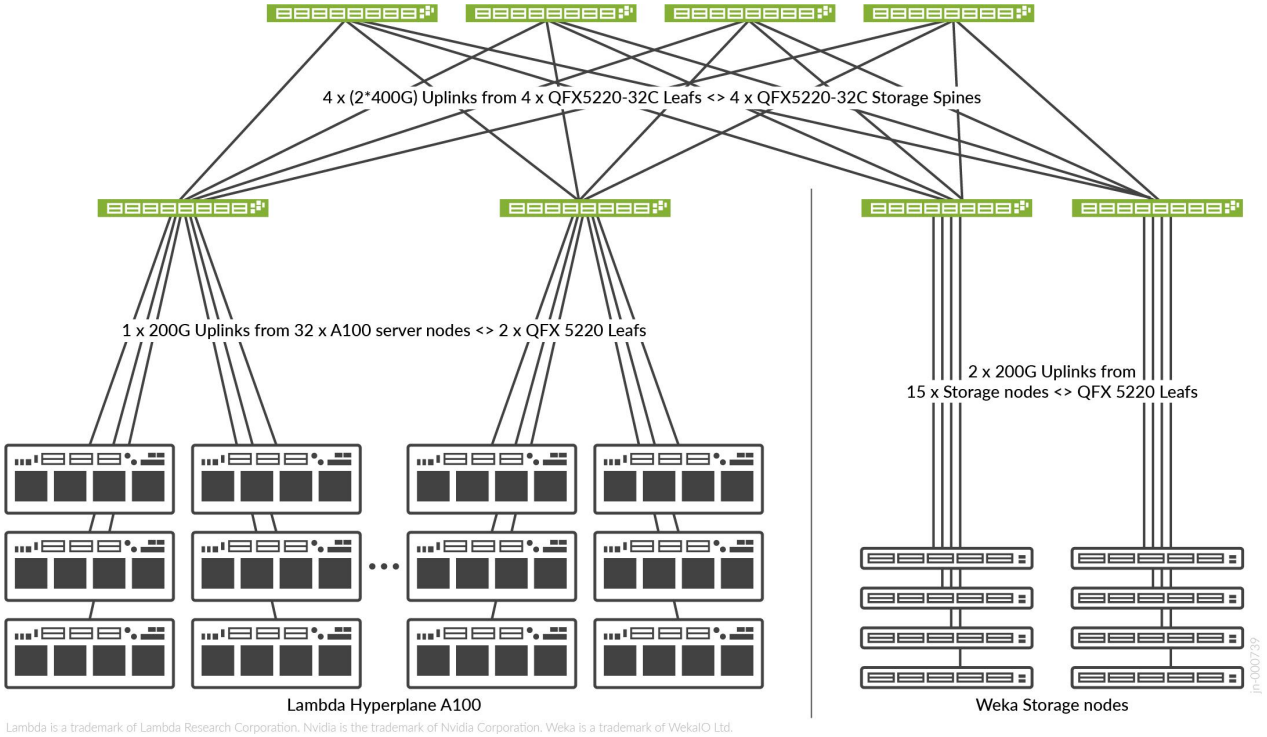



**图 8 展示了存储网络设计，包含两个主要部分：**

- GPU 存储节点段（如上例中的 Hyperplane8-A100）。
- 专用存储节点段（如上例中的 Weka AMD 存储节点）。

尽管存储集群中没有轨道优化（rail-optimization）的概念，但仍建议遵循轨道优化和网络优化的条带设计（stripe design），因为特定 GPU 数量与所需的最低存储节点数量之间存在直接关系（见表 2）。有关存储节点的更多信息，请参阅 AMD 文档。**Table 2.AMD documentation.**

表 2：GPU 数量与存储节点需求矩阵（来源：AMD）

| GPU 数量 | 所需总吞吐量 (GBytes/s) | 所需总吞吐量 (Gbits/s) | 每个存储节点的吞吐量 (GBytes/s) | 所需存储节点数 |
|---|---|---|---|---|
| 128 | 256 | 2048 | 36 | 8 |
| 256 | 512 | 4096 | 36 | 15 |

| 512 | 1024 | 8192 | 36 | 29 |
|---|---|---|---|---|
| **GPU数量** | **所需总吞吐量 (GBytes/s)** | **所需总吞吐量 (Gbits/s)** | **单存储节点吞吐量 (GBytes/s)** | **所需存储节点数** |
| 1024 | 2048 | 16,384 | 36 | 57 |
| 2048 | 4096 | 32,768 | 36 | 114 |

由于我们的 GPU 集群中 1 个条带（stripe）包含 32 台 A100 服务器，即 256 个 GPU，因此存储设计包含 15 个专用存储节点。规格和设计考虑因素如下：

- 该存储网络对应的 GPU 网络设计如图 6 和图 7 所示。
- 对于其他每个条带包含更多 GPU 的设计，需要相应修改专用存储节点的数量。
- 如图 10 所示，存储网络设计具有以下特征：
  - 计算-存储段包含 32 个 A100，每个 A100 均通过一条 200G 链路与叶节点连接，产生最高 6.4T 的向北流量，由 2 台 QFX5220-32CD 叶节点分担（每台 3.2T）。
  - 专用存储段有 15 个 Weka 存储节点，在本设计中每个节点通过 2 条 200G 链路向北连接至 QFX5220-32CD 叶节点，因此总负载为 2 × 200 × 15 = 6000 Gbps 即 6T，同样由 2 台 QFX5220-32CD 叶节点分担。
  - 本设计中所有四台QFX5220-32CDLeaf交换机均具有8条400G北向链路（每台Spine交换机两条），总出口容量达到6.4T，从而保持1:1的超配比（oversubscription ratio）。
  - 在专用存储网络（storage segment）中，Leaf交换机到Spine交换机的配置保持不变。由于存储节点数为15，Leaf交换机（leaf nodes）上的入口与出口流量容量比为6 Tbps : 6.4 Tbps，使其处于轻微欠订阅状态。
  - QFX5220-32CDSpine交换机（spines）拥有32个400G端口，而在此设计中仅使用了其中8个。这样做是为了兼顾冗余与扩展，这些Spine交换机（spines）在满容量时可支持图8所示规模的四倍，从而也能支持四倍的终端节点（end nodes）数量。
  - 该设计的另一种变体是使用两台Spine交换机（spines）而非四台，由于只利用了32个端口中的16个，链路级冗余保持不变，但节点级冗余会发生变化，并且扩展能力被限制为原设计的两倍而非四倍。

# 设计与实现

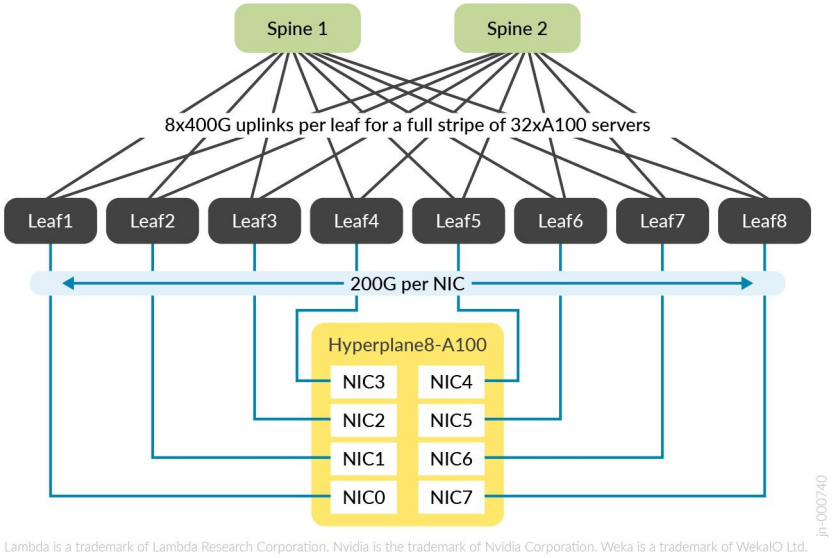
## 引言

本节将通过一个示例设计，展示如何使用 Juniper Apstra 编排网络基础设施。

从物理连接来看，这三张网络（后端网络、前端网络和存储网络）分别通过线缆连接，各自独立构建三层 Clos 架构。这意味着每台Leaf交换机都通过点对点三层链路连接到每台Spine交换机。按照轨道优化条带设计，每个 GPU 集群中相同位置的 NIC 都连接到后端网络中的同一台Leaf交换机。

图 9 展示了后端网络的一部分，假定每台Leaf交换机的端口密度为 32 x 400G。图中重点展示了一台 Hyperplane8-A100 服务器，共包含 8 个 A100 GPU 和 8 个 200G NIC。每个 NIC 均按照轨道优化条带设计连接到不同的Leaf交换机。

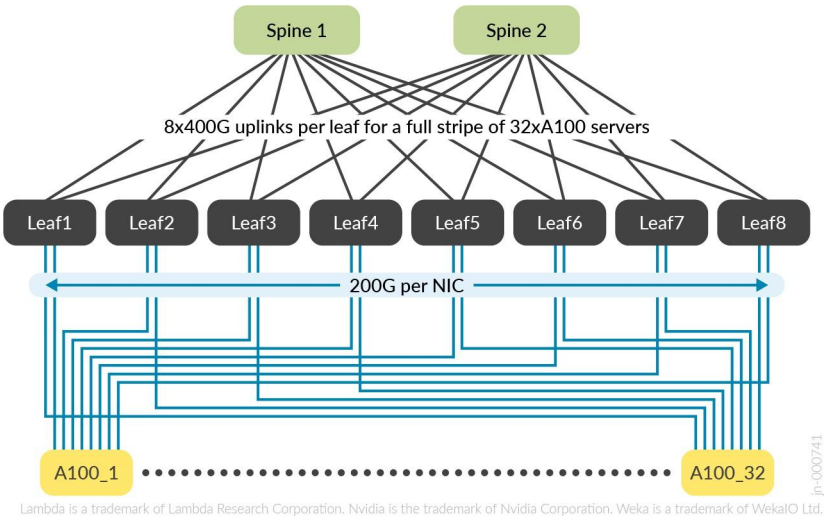
图 9：Hyperplane8-A100 与后端网络的连接




每个Leaf交换机从一台 A100 服务器接收 200G 流量，因为每台服务器仅通过一个 200G 网卡对应一块 GPU，并连接到一台Leaf交换机。将此配置扩展到 32 台 Hyperplane8-A100 组成的完整条带，意味着一个Leaf交换机将承接 32 × 200G 的流量。由于超配比必须为 1，上行链路带宽也必须完全匹配。使用两台Spine交换机时，每个Leaf交换机需为每台Spine交换机提供 8 × 400G 上行链路才能满足要求。

因此，针对我们的设计和实现，图 10 展示了后端网络拓扑，演示了 Juniper Apstra 如何构建这样的架构。

图 10：后端网络实现示例拓扑




Leaf交换机采用 QFX5220-32CD，Spine交换机采用 PTX10008，并配备 36 × 400G 业务板，可实现大规模扩展。

# 利用 IP FABRIC设计

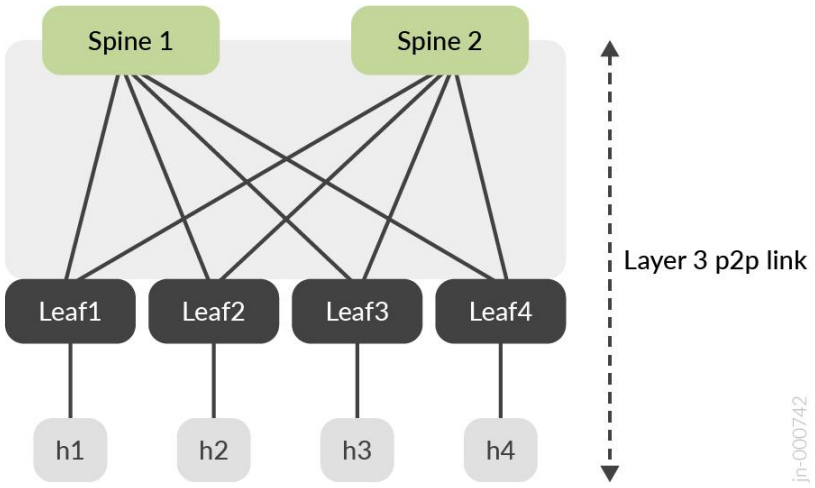
IP FABRIC是一种路由基础设施，采用胖树、Clos 类型的架构设计。这种设计在每一层提供统一的容量，仅使用三层链路，消除了对传统二层技术（如生成树协议（STP））的需求。而是通过路由协议将前缀信息注入架构中。IP FABRIC作为现代数据中心设计中的底层网络（underlay），可扩展至大型数据中心架构。

IP FABRIC通常延伸至主机（host），完全消除网络中的二层链路，并依赖三层路由协议进行收敛。如图11所示，在Leaf交换机与Spine交换机之间的三层点到点链路上启用路由协议后，每个Leaf交换机拥有到达Spine交换机的等价路径，并利用 ECMP 哈希将数据包分发到多条链路。Leaf交换机与主机之间使用路由协议或静态路由。

## 优势

- 通过等价多路径路由（ECMP）更好地利用所有可用路径。
- 更快的收敛速度。
- 消除对二层链路以及传统二层协议（如 STP）的需求。

图 11：端到端三层 Clos 架构（L3 Clos Fabric）设计




# 在面向 AI 集群的 IP FABRIC中部署的 IP 服务

典型的数据中心架构无需对流量进行精细调优；但 AI 集群对网络基础设施的要求却很独特。其流量模式为高密度、低熵，这意味着网络基础设施中常出现长流（elephant flows），且流自身变化极小。

由于这些流量特征，尽管Leaf交换机到Spine交换机之间存在等价路径，但默认哈希算法基于三层头部信息进行组合，可能导致链路利用率不均。由于流间差异不足，某条链路可能会被过度占用，致使这些长流（elephant flows）导致短流（mice flows，即低带宽流）发生丢包，并可能引发流冲突（flow collisions）。因此，建议在面向 AI 集群的 IP FABRIC 的Leaf交换机上配置动态负载均衡（DLB）。

此外，AI集群的流量需要无损网络。由于后端GPU网络采用RoCEv2（RDMA over Converged Ethernet），因此需要配置拥塞控制方法。本方案使用数据中心量化拥塞通知（DCQCN）。

## 动态负载均衡

静态负载均衡仅使用数据包头部（如三层头部信息）在存在多条等价路径时确定出接口。这种基于流的哈希导致同一流的数据包始终沿同一等价路径接口发送。然而，在处理AI集群流量时，这会造成接口利用率低下，因为AI集群流量具有高密度、长流特征且流之间差异微小，而静态负载均衡不考虑各等价链路的实际负载。

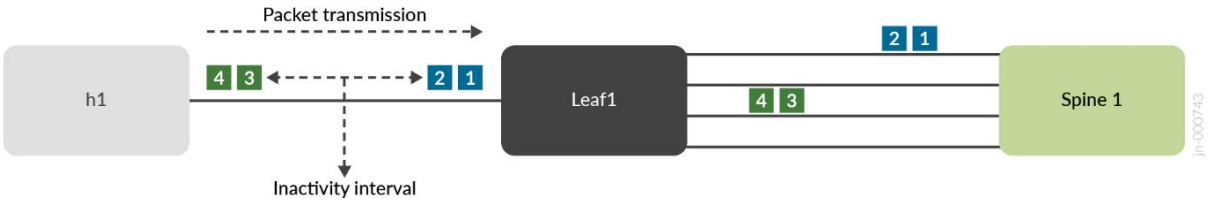
尽管常见思路是跨等价路径进行逐包负载分担，但这种设计影响很大，会导致数据包乱序。这正是动态负载均衡（DLB）的用武之地。DLB可按flowlet进行调度，flowlet是同一流内由一段空闲间隔分隔的突发数据。DLB的原理是：如果等价路径间的传输延迟小于flowlet之间的间隔（即空闲间隔），则该流的下一个数据包将通过利用率较低的链路发送，从而确保整个流的数据包即便经由不同链路也能按序到达。

在QFX5220-32CD上，DLB通过以下配置实现：

```
*snip*
forwarding-options
    { enhanced-hash-key
    {
        ecmp-dlb {
            flowlet {
                inactivity-interval 16;
            }
            ether-type
                { ipv4;
            }
        }
    }
}
policy-options {
    policy-statement DLB
        { then {
            load-balance per-packet;
        }
    }
}
routing-options
    { forwarding-table
    {
        export DLB;
    }
}
*snip*
```

本例中，假设来自同一流的多个数据包到达数据中心网络的Leaf交换机leaf1，需通过四条可能的等价路径发往Spine交换机spine1。

图12：DLB数据包流




数据包1至4属于同一条流，但数据包2和3之间存在一段延迟（单位为微秒）。此时，DLB根据配置的空闲间隔，将数据包1和2视为同一个flowlet，而将数据包3和4视为另一个flowlet。

DLB通常作为Broadcom ASIC中的一个引擎实现，持续接收等价路径的链路使用信息，并为每条链路分配一个质量等级。该分配是动态的，会随链路利用率的变化而不断变化。DLB判定：若发送数据包2所需的时间小于数据包2与数据包3之间的空闲间隔，那么即使数据包3通过不同的链路发送，也只会在数据包2到达之后才到达。这样，即使同一流的数据包通过不同链路发送，也能避免数据包乱序问题。

## 数据中心量化拥塞通知 (DCQCN)

现代高性能计算（HPC）或AI集群的数据中心要求高吞吐量和低延迟。标准TCP/IP协议栈难以满足这些要求，因为它会带来大量CPU开销，因此改用RoCEv2（RDMA over Converged Ethernet v2）。与InfiniBand不同，RoCEv2可以轻松集成到现有基于以太网的数据中心中，从而降低基础设施成本，并提供更大的灵活性。

RoCEv2通过配置优先级流控（PFC）和显式拥塞通知（ECN）等额外特性，为HPC和AI集群提供了所需的无损网络基础设施。
DCQCN（数据中心量化拥塞通知）是PFC与ECN的结合。

PFC作用于接口的接收缓冲区，当接收缓冲区超过特定阈值时，会发送Pause帧。这些Pause帧指示接收方在特定时间间隔内停止发送数据包，通过暂停传输来避免缓冲区溢出。然而，在网络中使用PFC也有缺点——持续的Pause帧会导致入口端口拥塞和丢包。因此，PFC常因队头阻塞（HOL blocking）造成受害流，以及不公平性（又称停车场问题），导致整体应用和网络性能不佳。此外，PFC工作在接口级别（具体为队列级别），无法区分不同的流。

ECN则是在支持ECN的发送方与接收方之间的一种端到端拥塞通知方法。DCQCN的总体思路是让ECN在PFC之前提前触发，发送方收到通知后便降低发送速率，从而对带有ECN标记的流量进行流控。

数据中心量化拥塞通知（DCQCN）的正确运行需要平衡两个相互冲突的要求：

- 确保基于优先级的流控（PFC）不会过早触发，以便让显式拥塞通知（ECN）有机会发送拥塞反馈来减缓流速。
- 确保基于优先级的流控（PFC）不会过晚触发，从而因缓冲区溢出导致数据包丢失。

# 使用 Juniper Apstra 构建面向 AI 集群的数据中心

## Juniper Apstra 概述

Juniper Apstra 是一款多厂商意图驱动网络系统（Intent-Based Network System, IBNS），负责编排数据中心部署，并通过 Day-0 到 Day-2 运维管理从小型到大型数据中心。它非常适合构建面向 AI 集群的数据中心，通过监控和遥测服务提供宝贵的 Day-2 洞察。

通过 Juniper Apstra 部署数据中心网络架构是一项模块化功能，利用各种构建块来实例化架构。基本构建块如下：

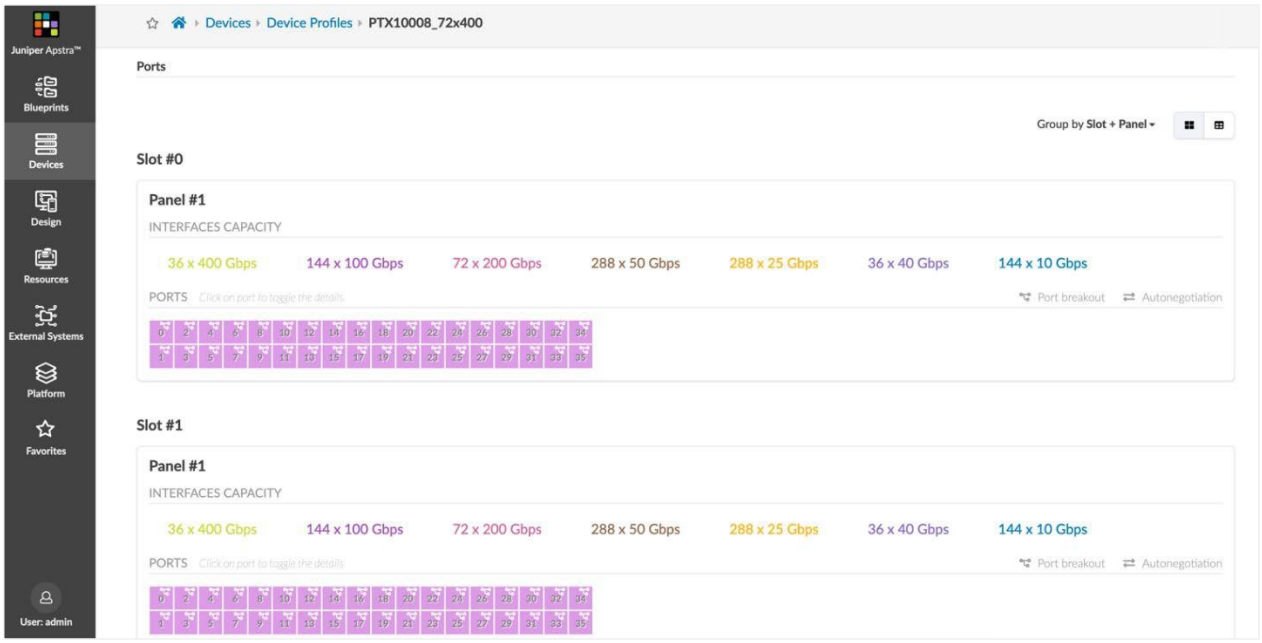
- **逻辑设备（Logical Device）是对交换机端口密度、速率及可能的拆分组合的逻辑表示。由于是逻辑表示，所有硬件细节都被抽象化。**
- **设备配置文件（Device Profile）定义了交换机的硬件规格，包括硬件（如 CPU、RAM、ASIC 类型等）和端口组织。Juniper Apstra 为不同厂商的常见数据中心交换机提供了多个预定义的设备配置文件。**
- **接口映射表（Interface Map）将逻辑设备与设备配置文件绑定，生成端口模式，并应用于由设备配置文件表示的特定硬件和网络操作系统。Juniper Apstra 默认提供了多个预定义接口映射表，也支持根据需要创建自定义接口映射表。**
- **机架类型（Rack Type）定义了 Juniper Apstra 中的逻辑机架，这与数据中心中物理机架的构造方式相同。但在 Juniper Apstra 中，这是对物理机架的抽象视图，其中包含了到用作Leaf交换机的逻辑设备（Logical Device）的链接、连接到每个Leaf交换机的系统类型与数量、任何冗余要求（如 MLAG 或 ESI LAG），以及每个Leaf交换机到每个Spine交换机的链路数量。**
- **模板（Template）将一个或多个机架类型作为输入，并定义架构的整体方案/设计，包括选择三级 Clos 架构、五级 Clos 架构或折叠式Spine交换机设计，以及选择构建一个 IP FABRIC（IP Fabric，如有需要可包含静态 VXLAN 端点）还是一个基于 BGP EVPN 的架构（以 BGP EVPN 作为控制平面）。**
- **蓝图（Blueprint）是架构的实例化，仅将模板作为输入。蓝图需要额外的用户输入来激活架构——这包括 IP 地址池、ASN 池、接口映射表（Interface Map）等资源。此外还需完成其他虚拟配置，例如定义新的虚拟网络（VLAN/VNI）、构建新的 VRF、定义到主机或 WAN 设备等系统的连接。**

## 后端网络

后端网络采用 PTX10008 Spine交换机构建，配备两块 PTX10K_LC1201_36CD 业务板，总共提供 72 个 400G 端口。在本实施场景中，使用了由 32 台 A100 服务器组成的单个条带（stripe），连接到 8 台 QFX5220-32CD Leaf交换机。

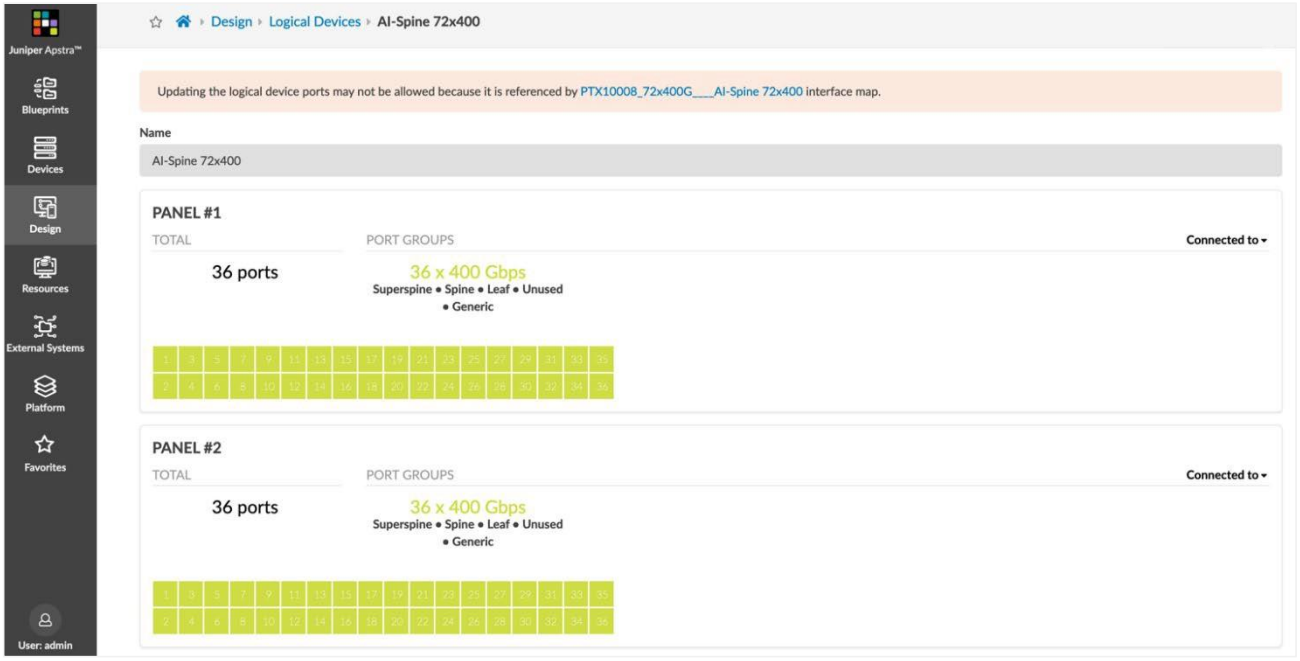
PTX Spine 交换机在 Juniper Apstra 中被定义为模块化设备配置文件（Device Profile），如下所示。它配备两块 36 端口 400G 业务板，为Leaf交换机互联提供总计 72 个 400G 接口。

图 13：Juniper Apstra 设备配置文件（Device Profile）



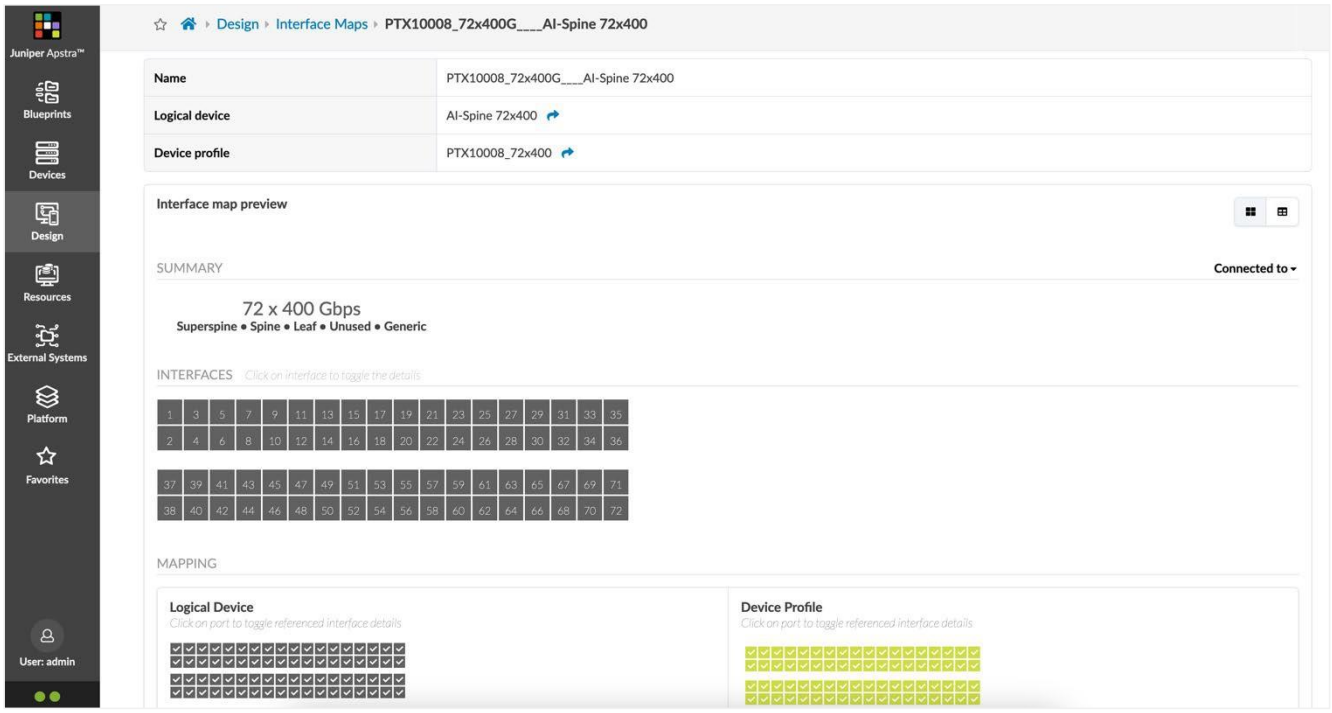

逻辑设备（Logical Device）的创建方式如图 14 所示。

图 14：Juniper Apstra 逻辑设备（Logical Device）




最后，创建接口映射表（Interface Map），将逻辑设备（Logical Device）与设备配置文件（Device Profile）关联。

图 15：Juniper Apstra 接口映射表




Leaf交换机采用 QFX5220-32CD，提供 16 个 400G 端口，其余 16 个端口每个拆分为两个 200G 端口，共获得 32 个 200G 端口。创建一个新的逻辑设备和接口映射表，如图 16 和图 17 所示。

图 16：支持 16 x 400G 和 32 x 200G 端口的 QFX5220-32CD Leaf交换机逻辑设备

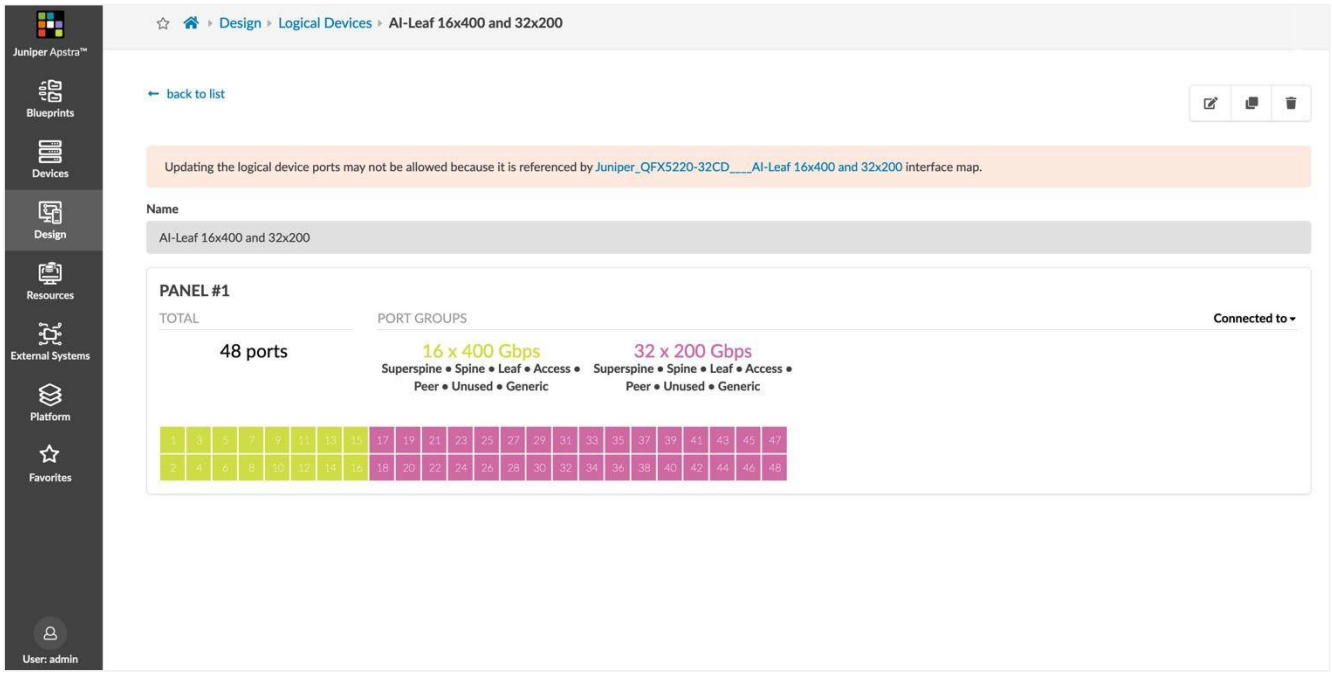
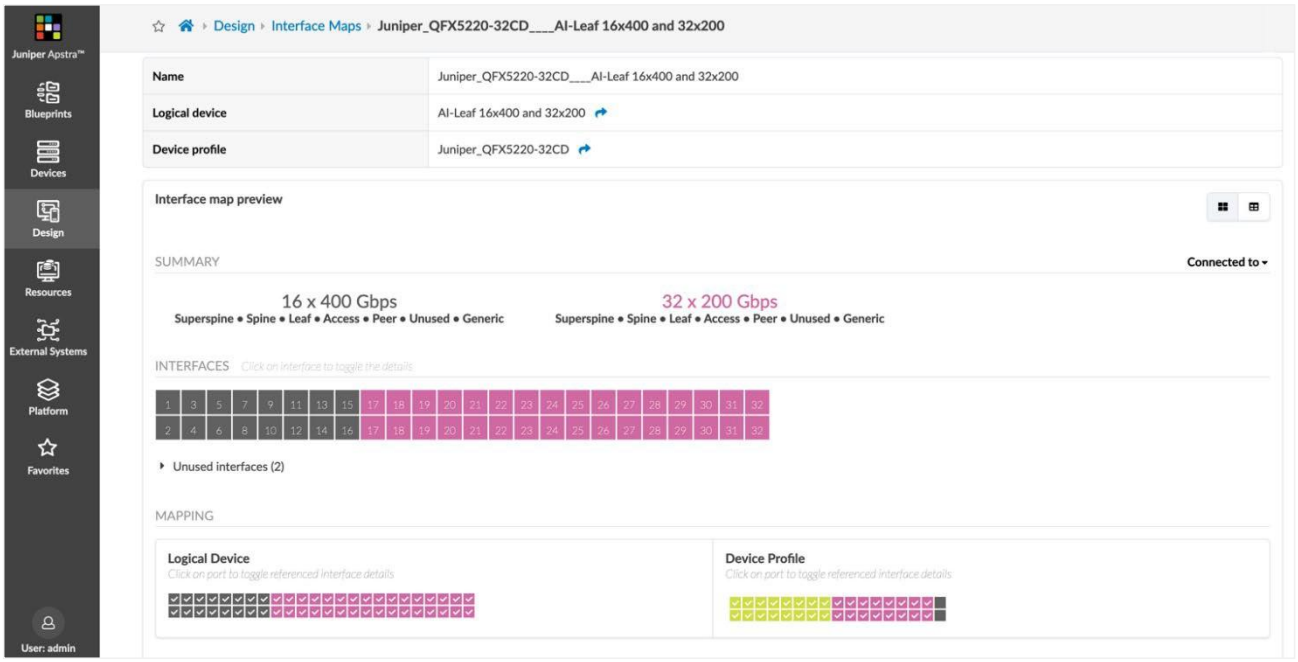


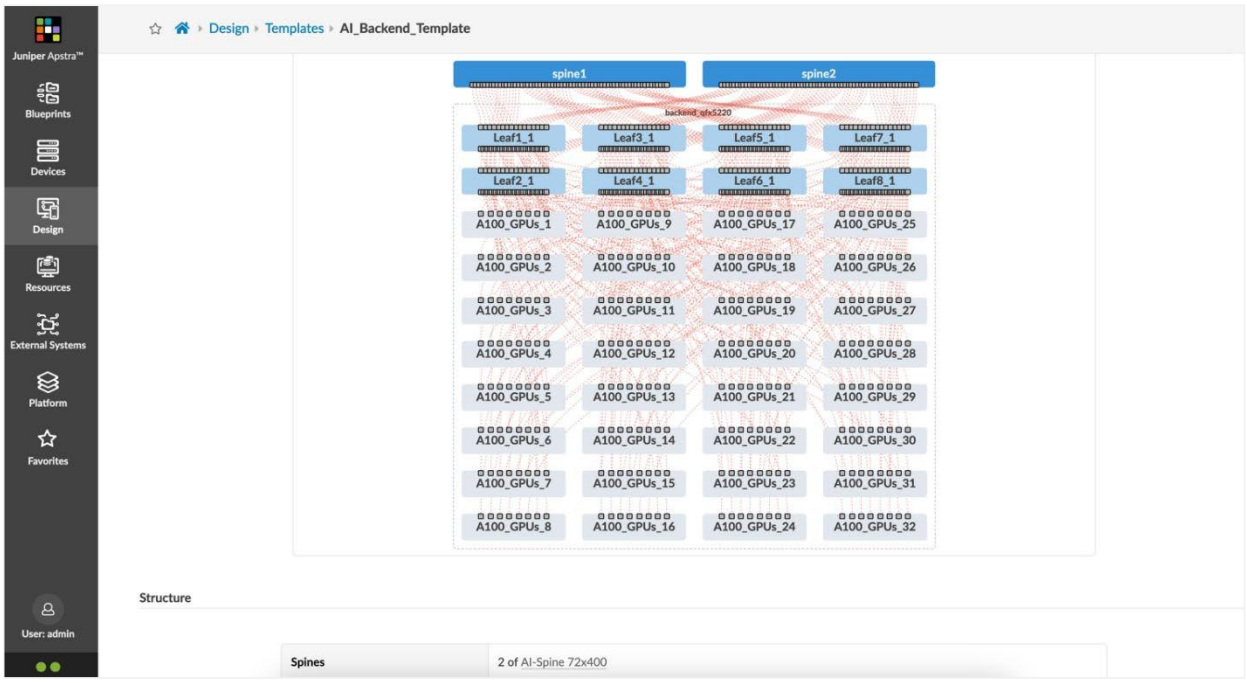
图 17：支持 16 x 400G 和 32 x 200G 端口的 QFX5220-32CD 接口映射表




要构建后端网络，首先创建一个机柜类型，并将上述逻辑设备作为Leaf交换机。其中包含 8 台Leaf交换机和 32 套通用系统（32 个 Hyperplane8-A100），这些系统按照轨道优化条带设计连接到这些Leaf交换机。每个Leaf交换机与每个Spine交换机之间均有 8 条 400G 链路。该机柜类型将作为后端网络模板的输入。该模板配置为三层 Clos 设计，并使用静态 VXLAN（若用户未定义任何静态 VXLAN 端点，则实际上构建了一个 IP FABRIC）。

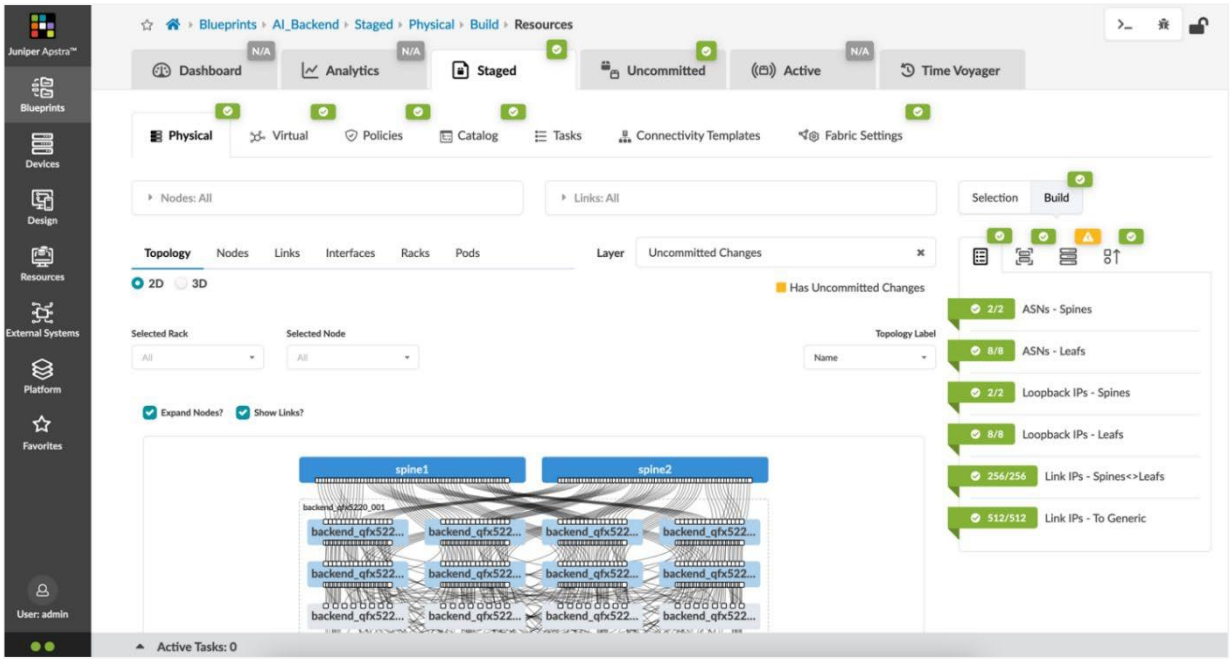
图18展示了该模板，其中的Spine交换机的定义使用了为PTX10008设备配置文件（Device Profile）创建的逻辑设备（Logical Device）。

图18：用于后端网络（Backend Network）的Juniper Apstra模板



最后，使用该模板实例化了一个蓝图（Blueprint）。

图19：用于后端网络（Backend Network）的Juniper Apstra蓝图




为了开始构建网络架构（fabric），蓝图（Blueprint）中添加了多项资源，包括Leaf交换机和Spine交换机的ASN池、Spine交换机和Leaf交换机的环回地址，以及Leaf交换机与Spine交换机之间的点对点地址。

为 ASN 和环回接口定义了一个公共 ASN 池和 IP 池，如图 20 和图 21 所示。

图 20：Juniper Apstra ASN 池

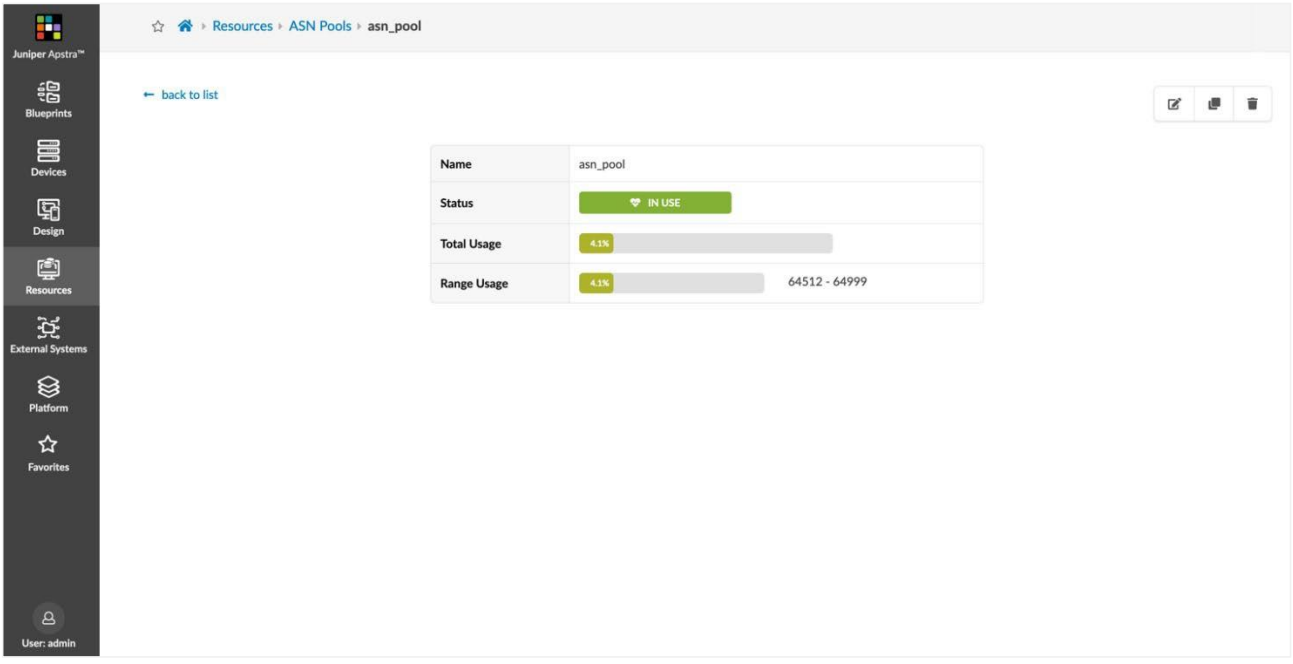
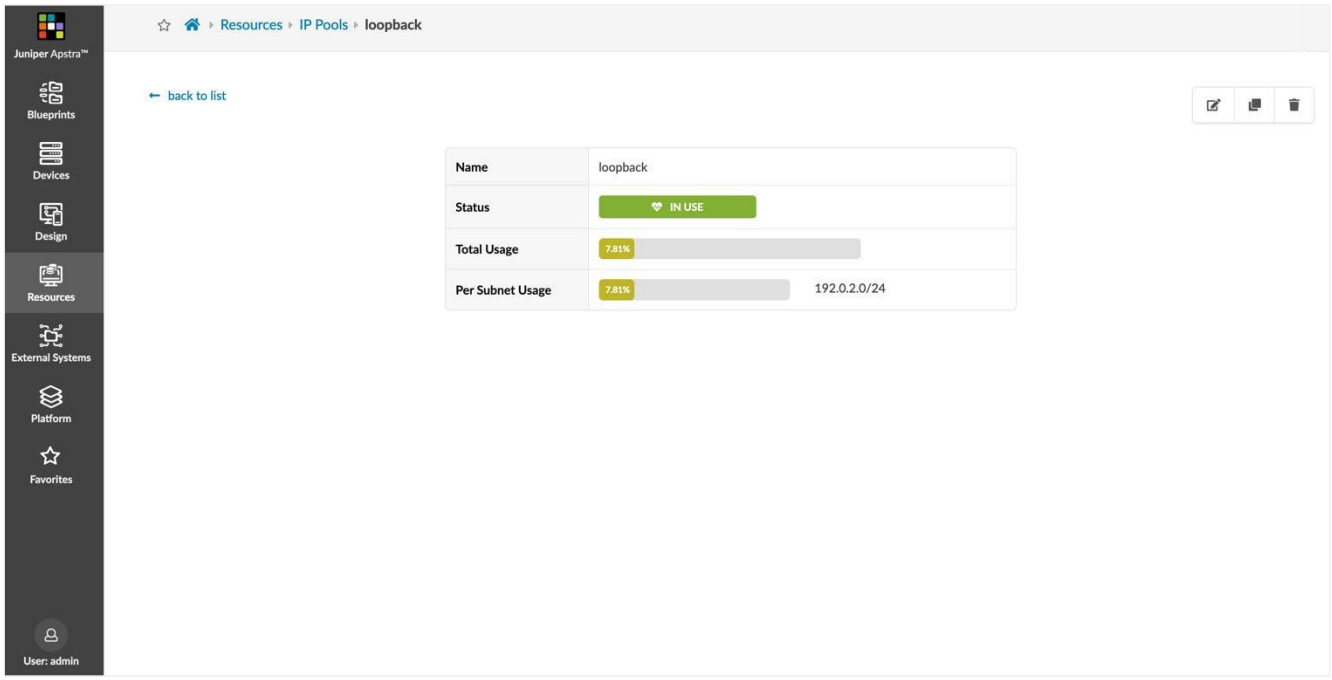
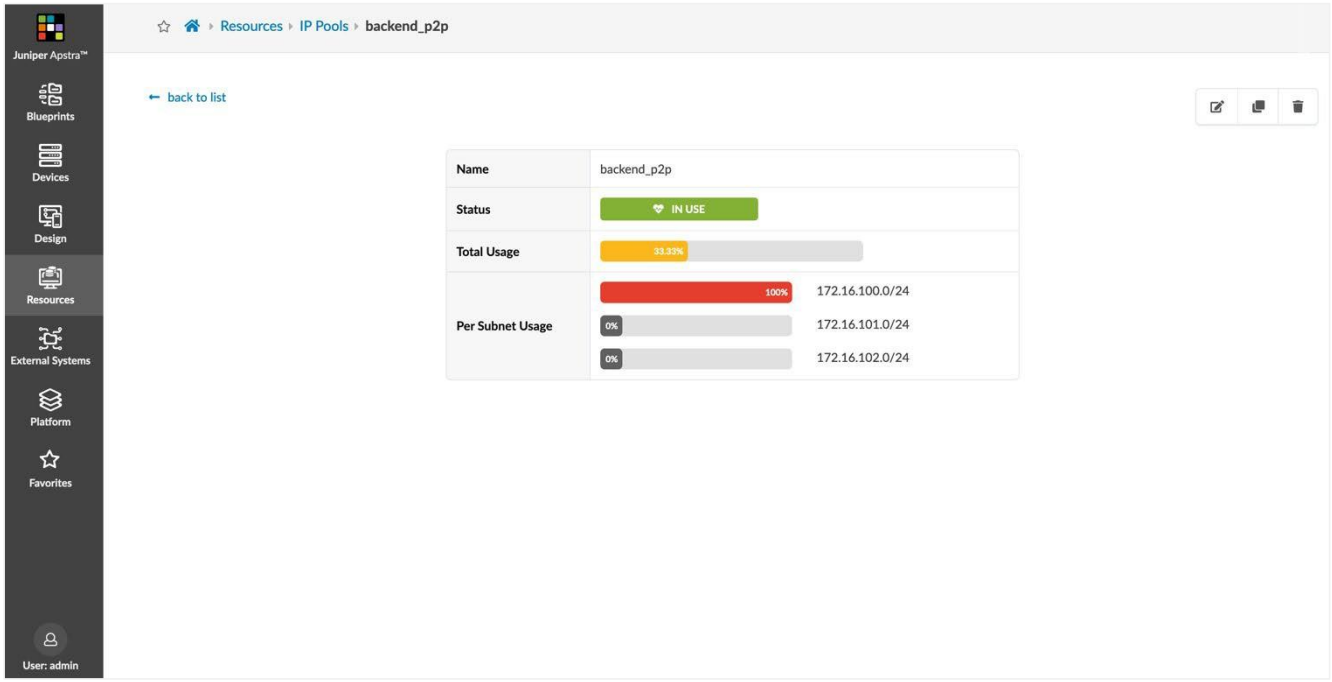


图 21：用于环回接口的 Juniper Apstra IP 池



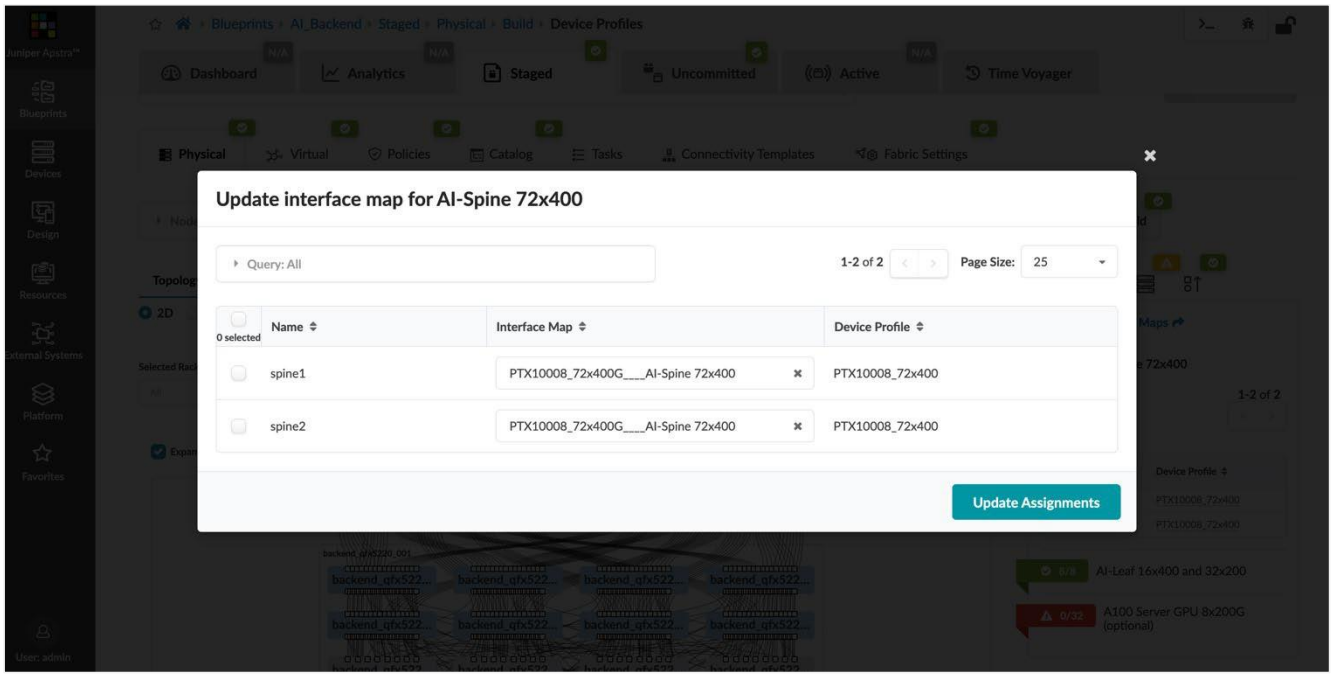

由于有八个Leaf交换机，且每个Leaf交换机到每个Spine交换机有八条链路，Leaf交换机和Spine交换机之间的点对点链路需要 256 个地址。为此，为后端网络定义了一组单独的池，如图 22 所示。

图 22：用于Leaf交换机与Spine交换机之间点对点链路的 Juniper Apstra IP 池



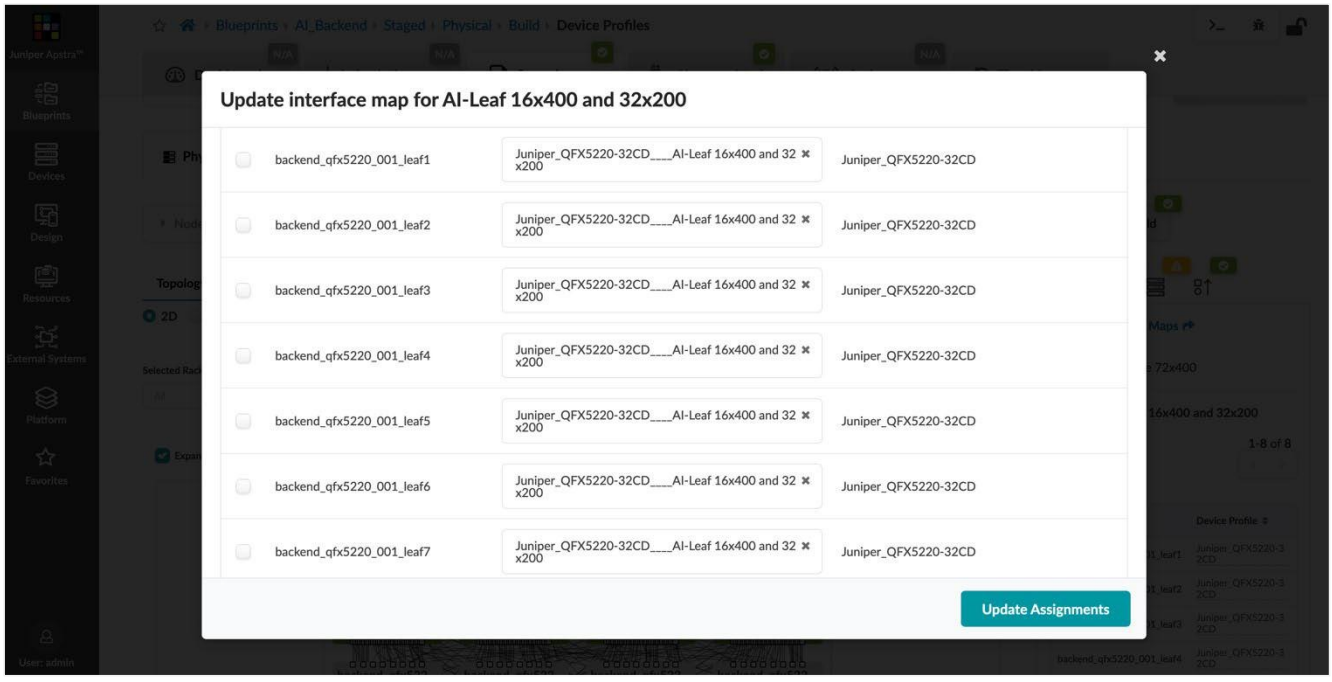

若要开始生成设备专用配置，必须先将接口映射表（Interface Maps）映射到 IP FABRIC中的每台设备。对于Spine交换机，使用与 72 x 400G PTX10008 设备相对应的接口映射表。

图 23：Spine交换机的接口映射表到设备的映射




对于Leaf交换机，使用与 QFX5220-32CD 相对应的 16 x 400G + 32 x 200G 接口映射表，如图 24 所示。

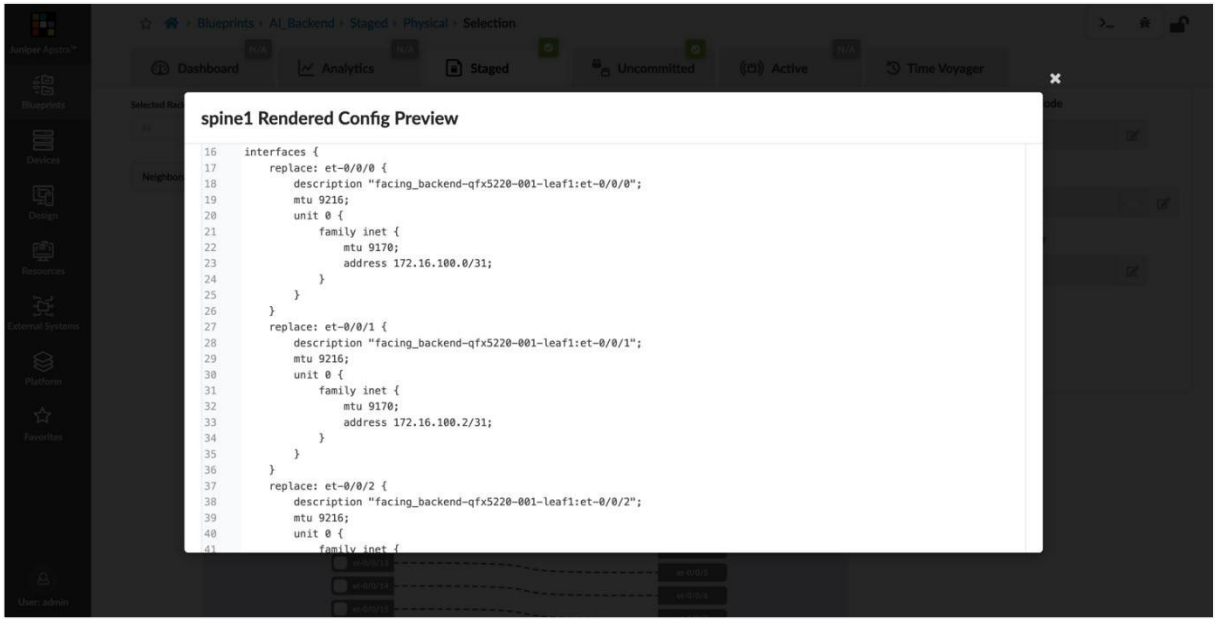
图 24：Leaf交换机的接口映射表到设备的映射




完成这些映射后，Juniper Apstra 会生成用于构建 IP FABRIC的配置。要查看 IP FABRIC中某台设备的渲染后信息，请在拓扑中相应的设备视图内，点击右侧“Config”下的“Rendered”。

例如，我们可以确认，渲染的配置中包含用于连接各Leaf交换机的点对点/31接口地址，如图25所示。

图25：Juniper Apstra为Spine1渲染的配置




在后端网络中，至GPU的连接同样是三层连接。为此，使用了Juniper Apstra中的连接模板，并将其附加到Leaf交换机上所有连接通用系统（逻辑上代表Hyperplane8-A100服务器）的链路。因此，应用点为接口本身。

此连接模板是一个IP链路原语，使用默认路由区域和带编号的IPv4寻址类型。

图26：Juniper Apstra用于连接GPU的IP链路连接模板

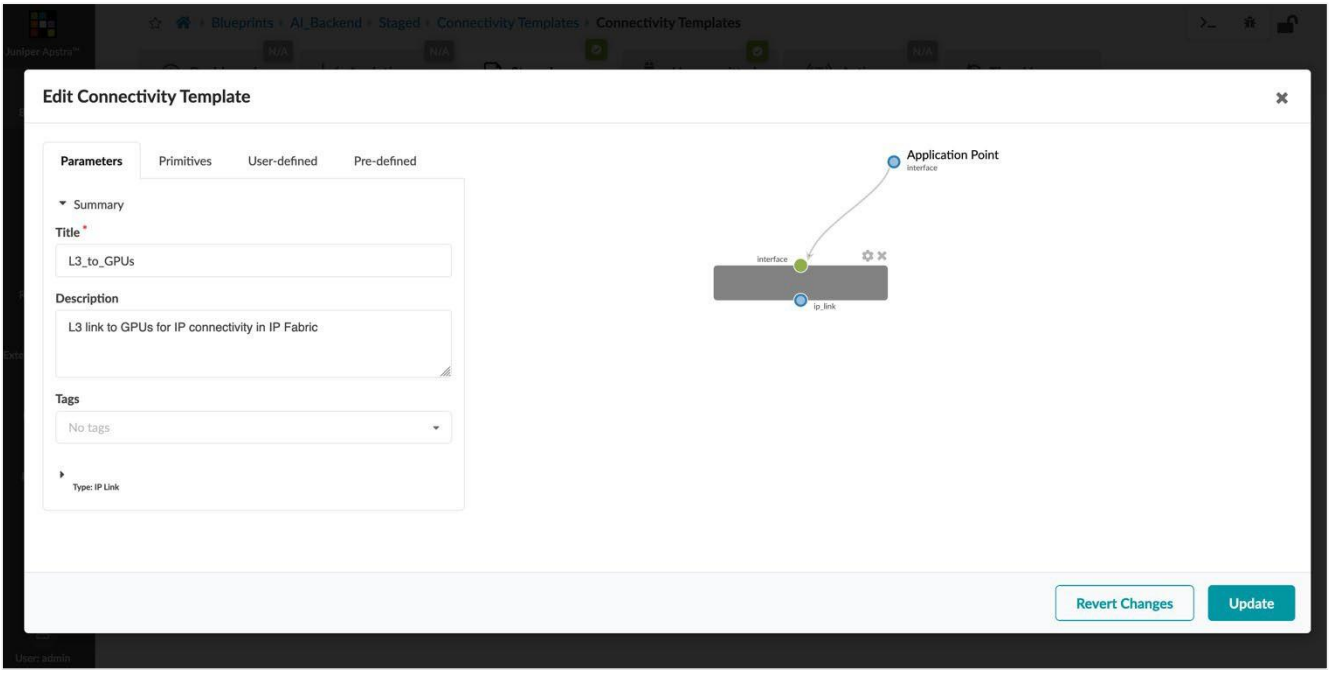
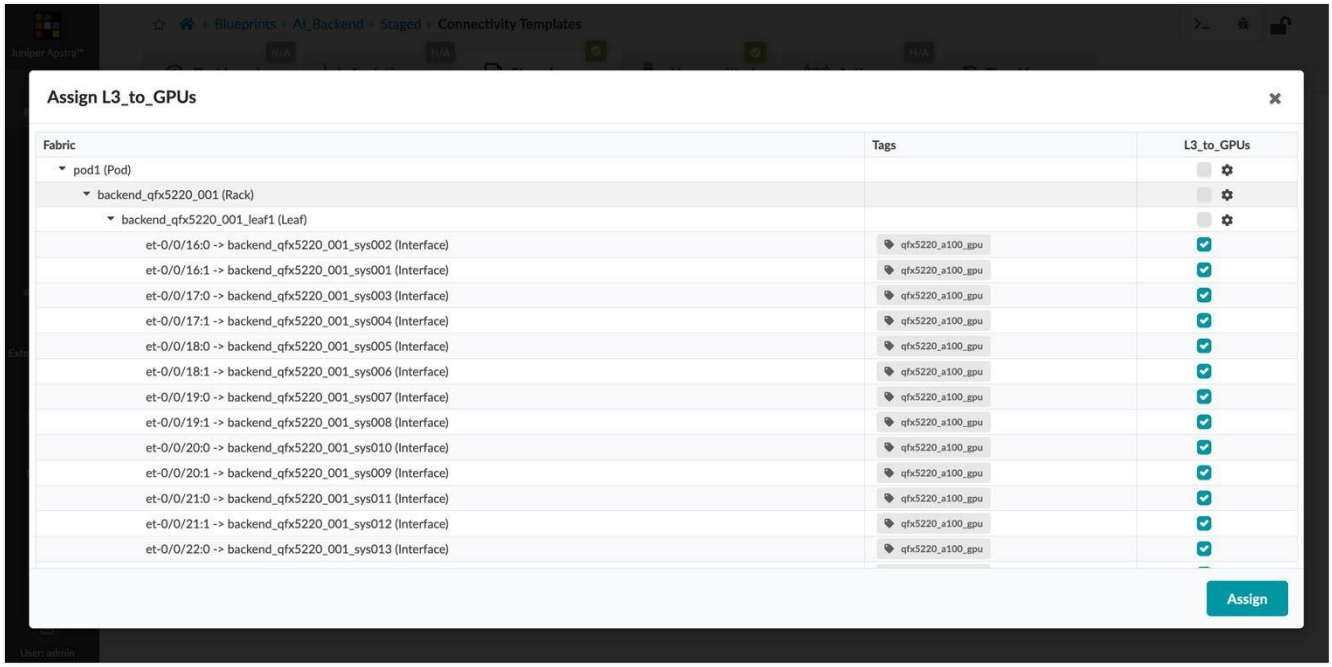


图 27：用于 GPU IP 链路的 Juniper Apstra 连接模板配置


然后将此连接模板附加到所有Leaf交换机上所有面向 GPU 的接口。配置片段见图28

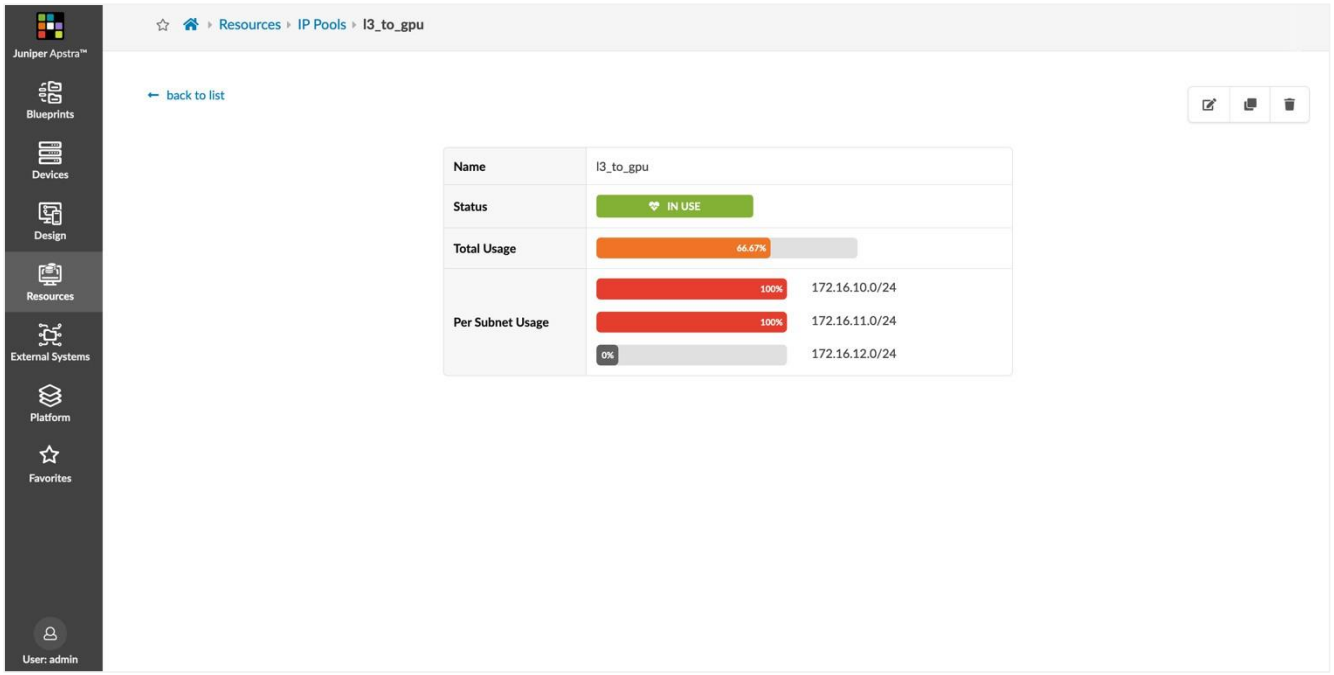
图 28：将连接模板附加到接口




连接模板附加到接口后，Juniper Apstra 还要求为这些新的点对点链路配置 IP 地址池。由于一个条带包含 32 个 Hyperplane8-A100，每台Leaf交换机就有 32×8 条通往 GPU 的链路，因此需要 512 个地址。

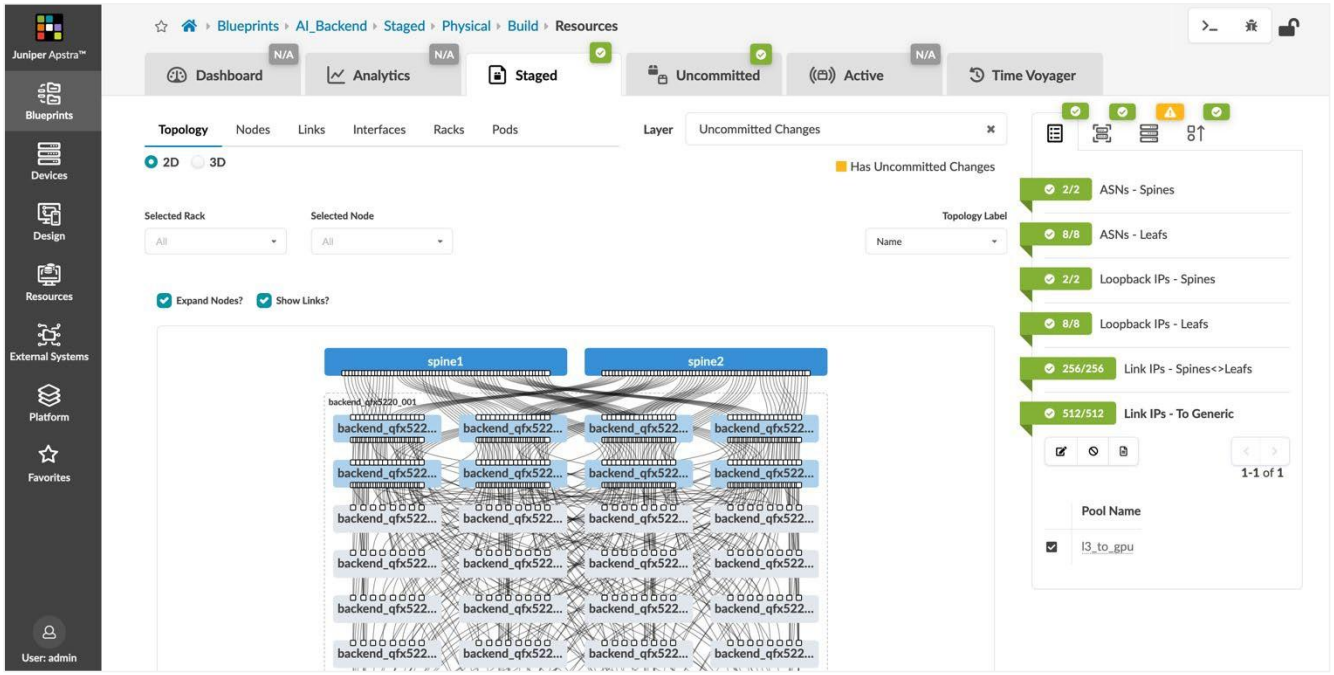
如图 29 所示，创建并使用了一组专用的 IP 地址池。

图 29：为连接 GPU 的 IP 链路而创建的 Juniper Apstra IP 地址池




该地址池已关联到 Juniper Apstra 中的资源需求。

图 30：Juniper Apstra 三层 IP FABRIC的 IP 地址池分配

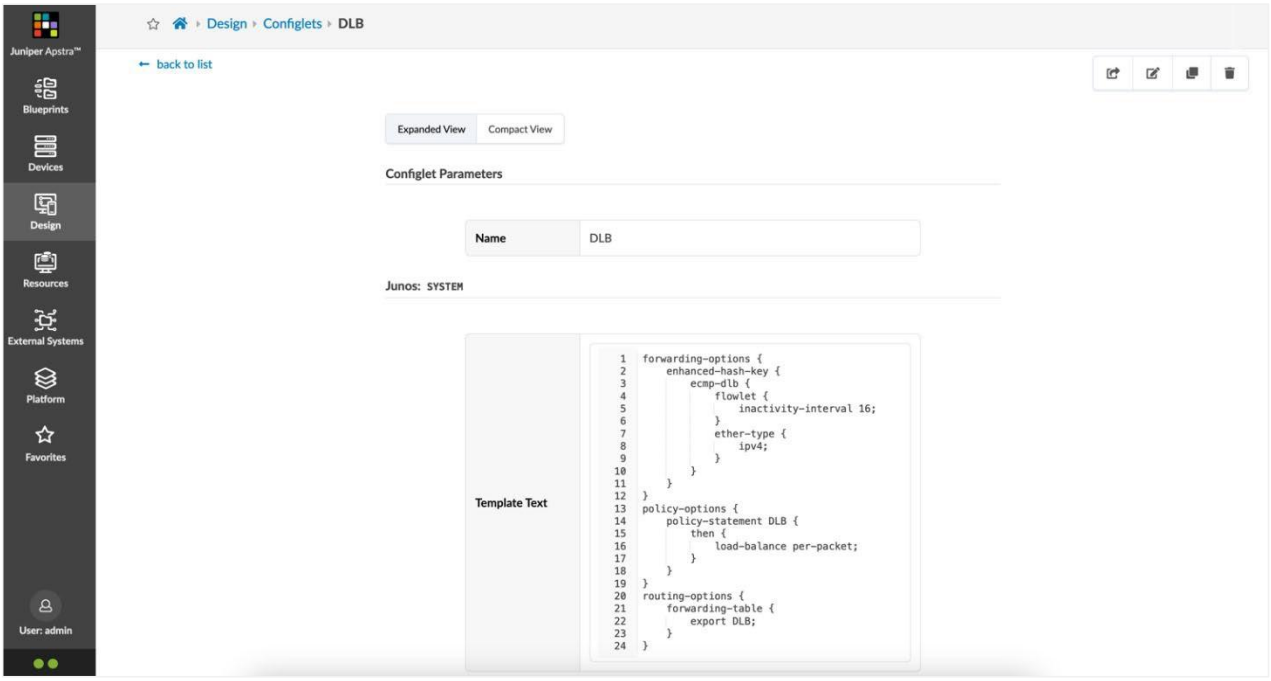



Juniper Apstra 现在为每台Leaf交换机生成这些面向 GPU 的点对点接口配置，并通过 BGP 将这些地址重新分发到 IP FABRIC中，以实现所有 GPU 服务器之间的可达性。

主架构配置完成后，必须使用 Juniper Apstra 中的配置小工具（Configlet）添加两项 IP 服务（DLB 和数据中心量化拥塞通知（DCQCN））。

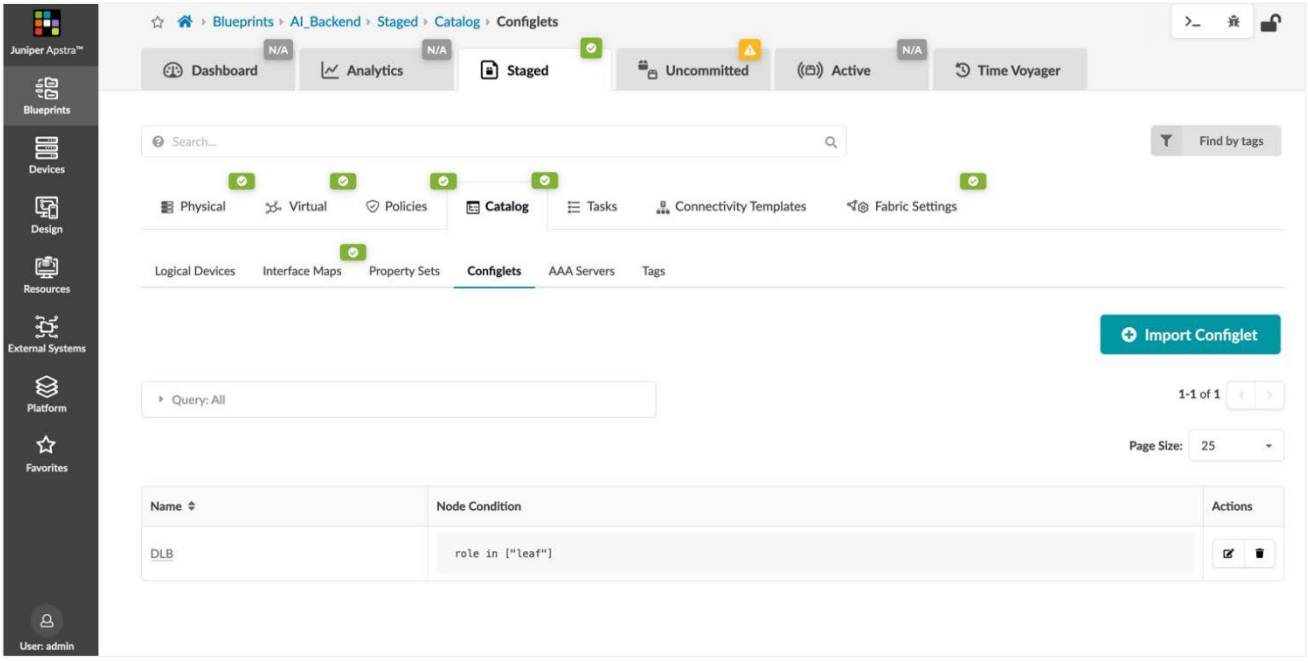
首先，如图 31 所示，为 DLB 创建一个配置小工具（Configlet）。

图 31：用于 DLB 的配置小工具（Configlet）



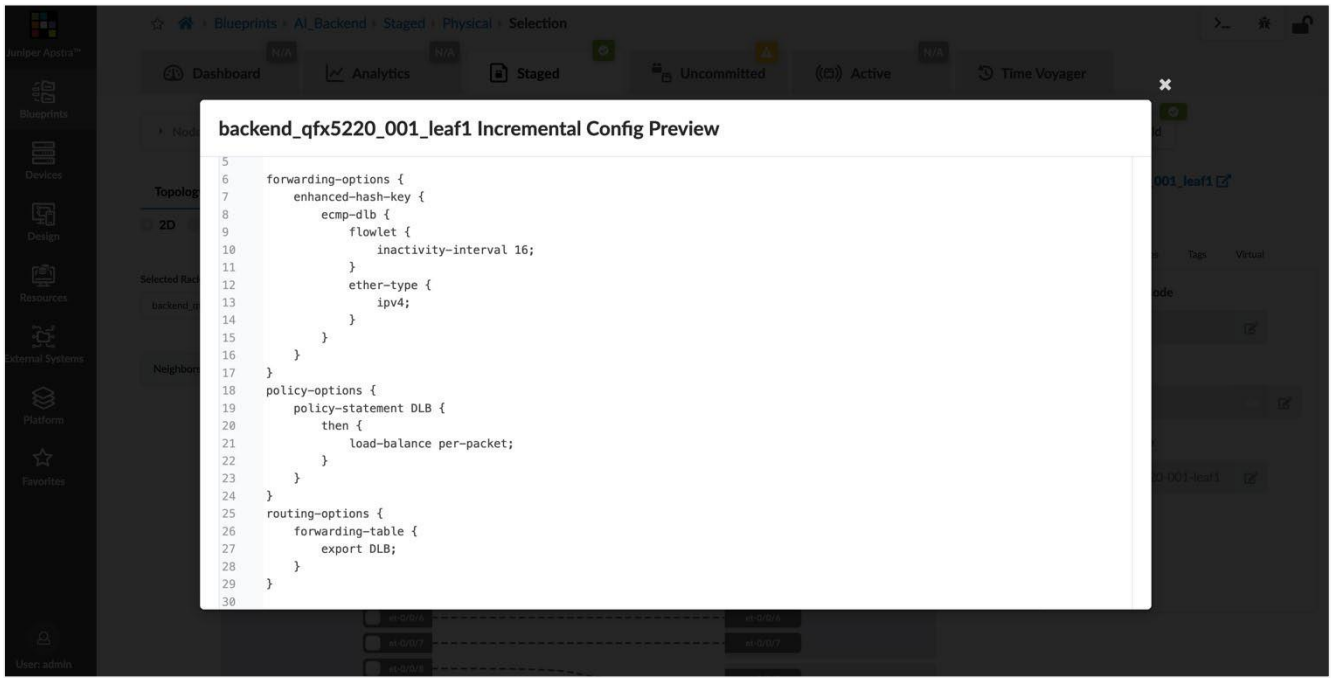

要在 Blueprint 中使用配置小工具（Configlet），必须将其导入 Blueprint 目录。通过特定条件进行分配，以确定哪些设备会接收该配置小工具所定义的额外配置。对于 DLB，我们需要所有Leaf交换机都应用此配置，因此条件需匹配 “leaf” 角色。

图 32：DLB 配置小工具（Configlet）的条件匹配



一旦将其添加到蓝图中，即可确认Leaf交换机已通过此配置小工具添加了增量配置。以 leaf1 为例，如图 33 所示。

图 33：成功导入 DLB 配置小工具后添加到Leaf交换机的增量配置




同样，数据中心量化拥塞通知（DCQCN）的相关配置也会作为配置小工具添加。由于 DCQCN 配置可能变得复杂，通过 Terraform 等工具实现自动化，可以提供可靠性和灵活性，以便在所有必要接口上准确应用该配置。

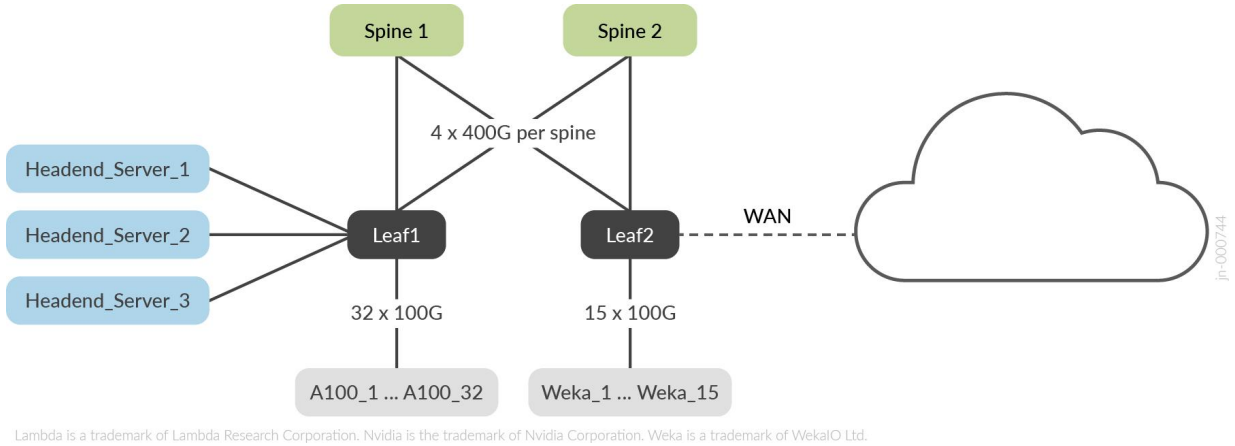
## 前端网络

前端网络的Spine交换机和Leaf交换机均使用 QFX5220-32CD。就超配比而言，其网络要求不如后端网络严格。为保持一致，超配比仍尽可能保持在接近 1 的水平。

Hyperplane8-A100s 的标准网络端口用于连接前端网络。每台服务器仅有一个此类端口，这意味着需要将 32 个 100G 前端接口接入该网络。所有 GPU 服务器端口均连接至 leaf1，存储前端接口则连接至 leaf2。

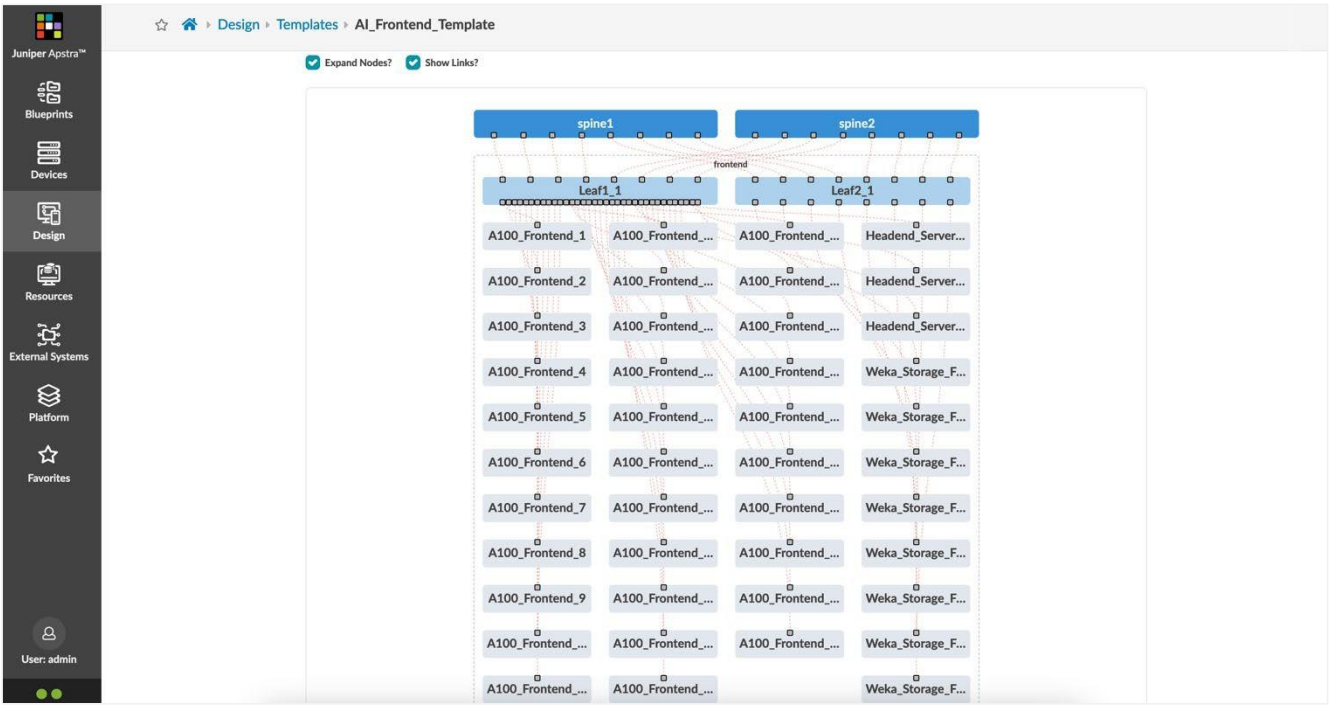
图 34 展示了一个拓扑，为 Juniper Apstra 实现的前端网络提供了 High-Level 概览。

图 34：Juniper Apstra 中前端网络实现的拓扑



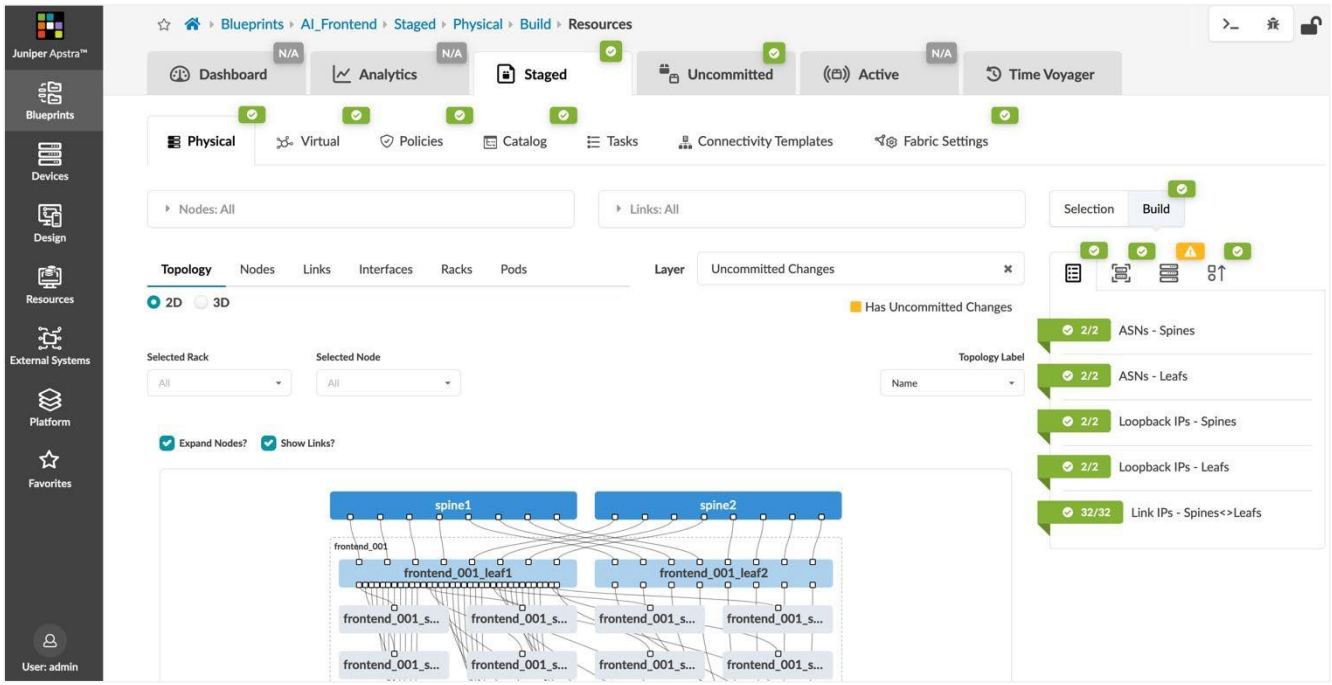

在 Juniper Apstra 中，此机架类型被添加至模板，如下所示。该模板与后端网络所用模板类似，但替换为前端专用机架类型。

图 35：前端网络的 Juniper Apstra 模板



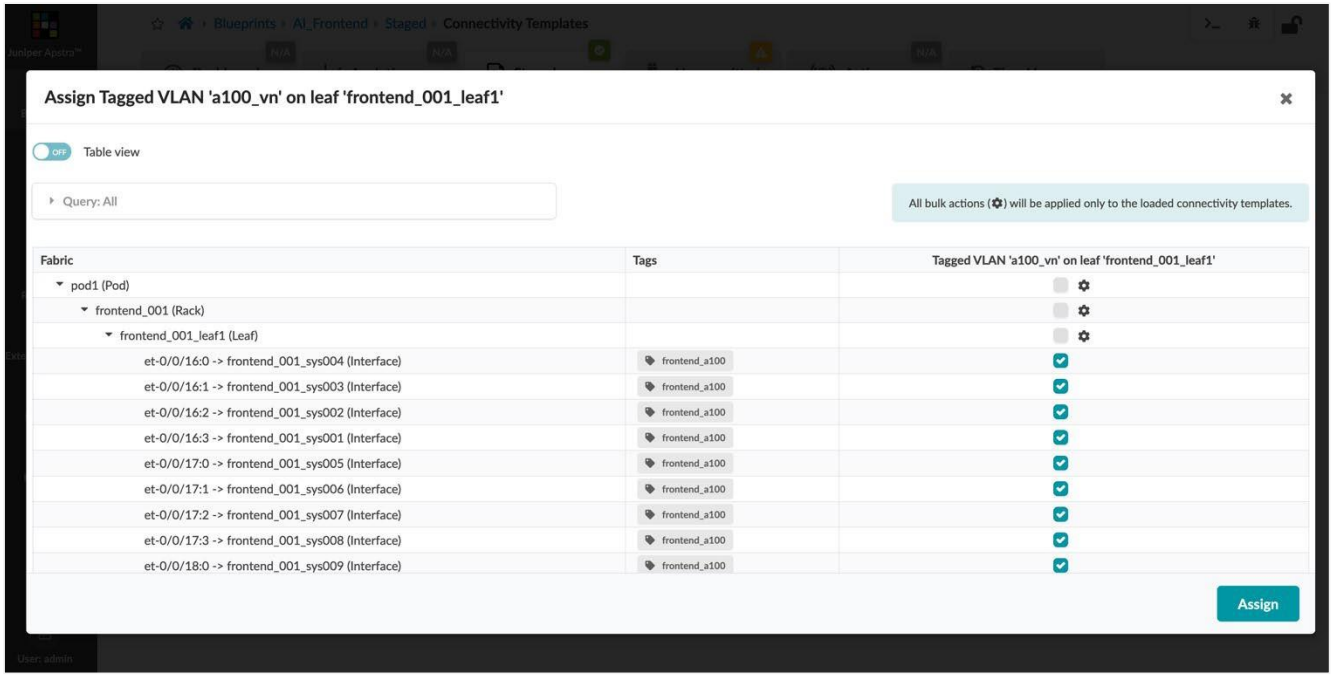

您可以使用此模板实例化一个专用于前端网络的蓝图。

图36：用于前端网络的Juniper Apstra蓝图




与任何其他架构一样，蓝图包含 ASN 和 IP 地址池等资源，以及架构中每台设备的接口映射表。对于前端网络，无需与服务器建立三层点对点链路，而是使用无标记或带标记的接口提供通用的二层连接。然后，将此连接模板分配给 leaf1 上所有面向 A100 服务器的接口。

图37：附加到前端网络Leaf交换机接口的连接模板



同样，为头端服务器使用另一个二层连接模板，为面向 WAN 的接口使用一个三层连接模板，从而可以从 WAN 边缘接收默认路由。

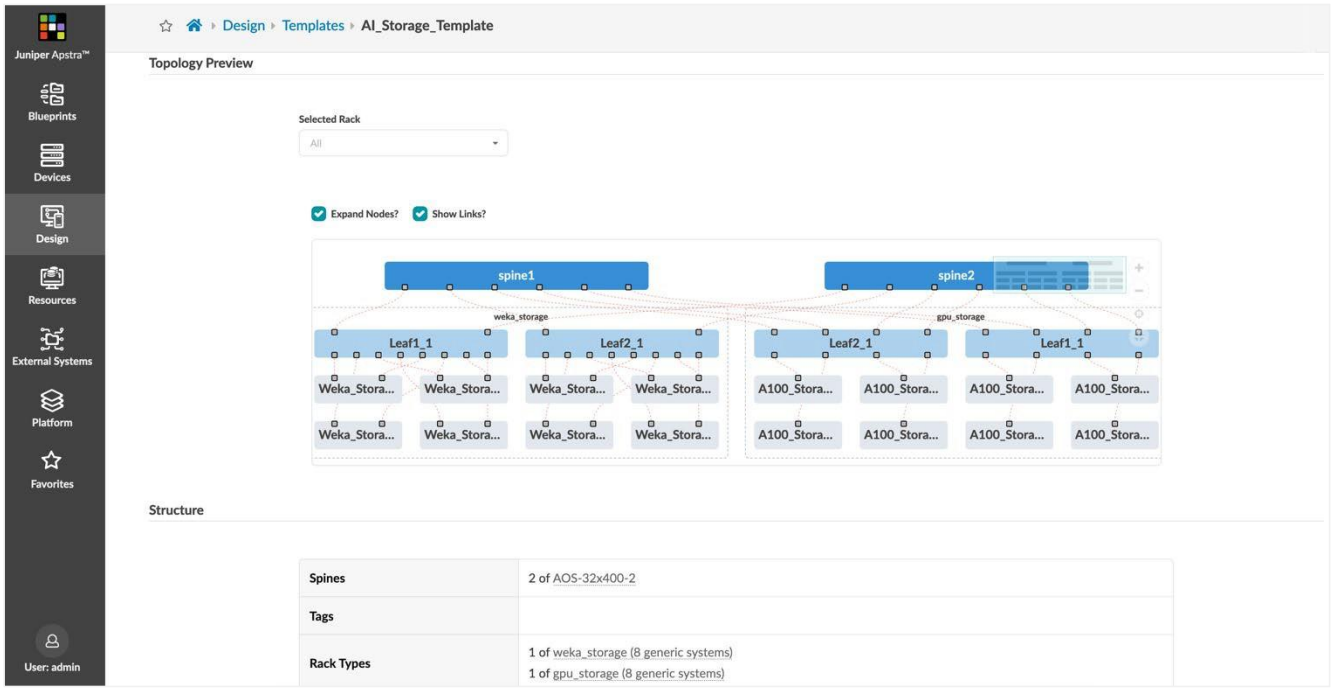
## 存储网络

存储网络用于连接 Hyperplane8-A100 服务器的存储端口以及包含 Weka 存储节点的专用存储集群。为此构建了两种机架类型，每种机架均配备 2 台 QFX5220-32CD Leaf交换机。一种机架类型连接 GPU 存储端口，另一种则连接专用存储节点。

每台 GPU 服务器均通过一个专用的 200G 存储端口连接至一台Leaf交换机。由于共有 32 台服务器，集群被划分为两组，其中 16 台连接至 leaf1，另外 16 台连接至 leaf2。因此，对于 GPU 存储机架而言，每台Leaf交换机将接收 16 路 200G 入站流量，这意味着每台Leaf交换机必须提供 8 个 400G 端口。

这些机架类型均添加至 Juniper Apstra 的存储专用模板中。

图 38：Juniper Apstra 存储网络模板




可使用此模板为存储网络实例化蓝图，而各种资源又会绑定到该蓝图。对于存储网络，我们建议服务器和存储节点使用三层链路。为此，采用了一个三层连接模板，将其附加到这些节点的所有链路上，与后端网络类似。

在此阶段，后端网络、前端网络和存储网络这三个网络均已由 Juniper Apstra 完整构建，所有必要配置已可推送至该架构的网络设备。

# 总结

本文档展示了使用 Juniper 网络设备的各种 AI/ML 集群设计，以及通过 Juniper Apstra 的相应实现。文档中的设计考量均从网络角度出发，GPU 集群和模型的性能可能会因诸多外部因素而变化，例如 GPU 硬件、网卡、拥塞控制机制、所采用的算法以及工作负载本身等。如需咨询具体实施方案，请联系 Juniper Networks 代表。

**Corporate and Sales Headquarters**
Juniper Networks, Inc.
1133 Innovation Way
Sunnyvale, CA 94089 USA
Phone: 888.JUNIPER (888.586.4737)
or +1.408.745.2000
Fax: +1.408.745.2100
www.juniper.net

**APAC and EMEA Headquarters**
Juniper Networks International B.V.
Boeing Avenue 240
1119 PZ Schiphol-Rijk
Amsterdam, The Netherlands
Phone: +31.207.125.700
Fax: +31.207.125.701



请发送反馈至：design-center-comments@juniper.net V1.0/231110/ai-clusters-data-center-designdesign-center-comments@juniper.net