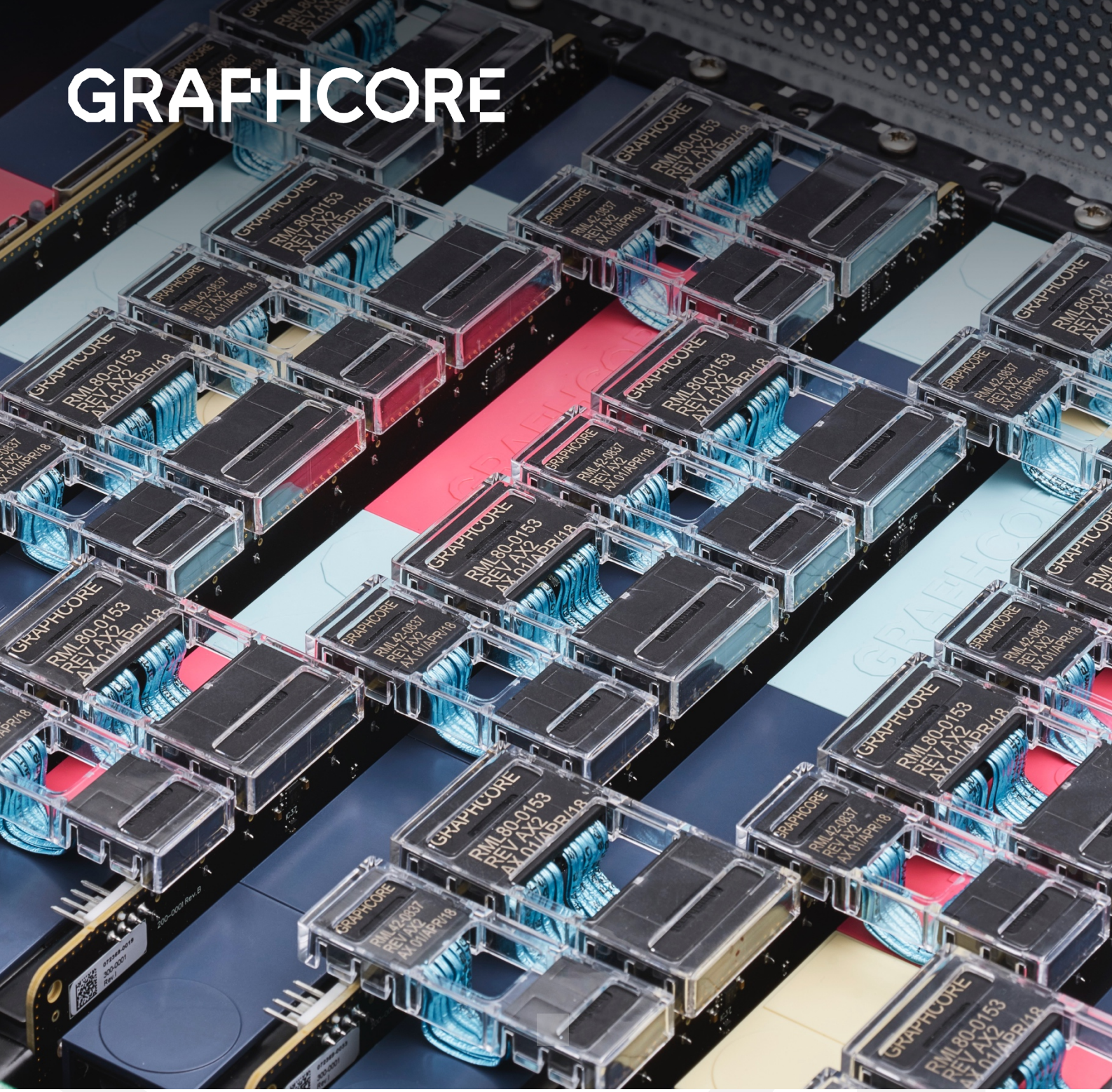


戴尔DSS8440 GRAPHCORE IPU服务器

白皮书

目录

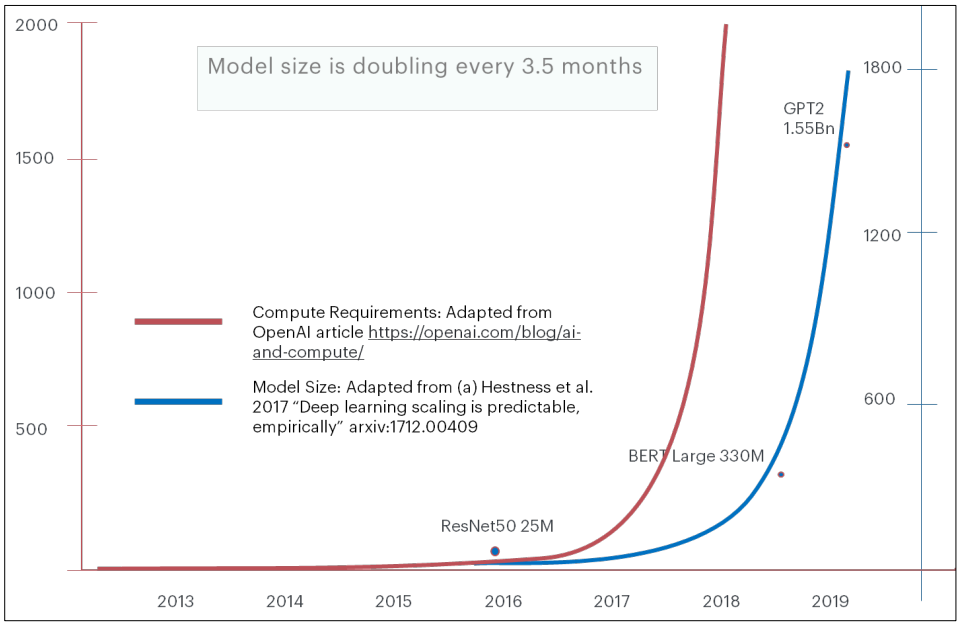
基于IPU的机器智能 3
Graphcore C2 IPU处理器卡 6
Dell DSS 8440 IPU服务器 7
Poplar SDK 9
基于Poplar SDK的灵活性 10
基于IPU服务器的机器智能 11
使用图编译域（GCD）进行模型并行 11
Pipelining 12
重新计算(Recomputation) 14
数据并行训练或多副本训练 15
用于扩展多个DSS 8440 IPU服务器的InfiniBand 16
摘要 17
开始使用IPU 18

基于IPU的机器智能

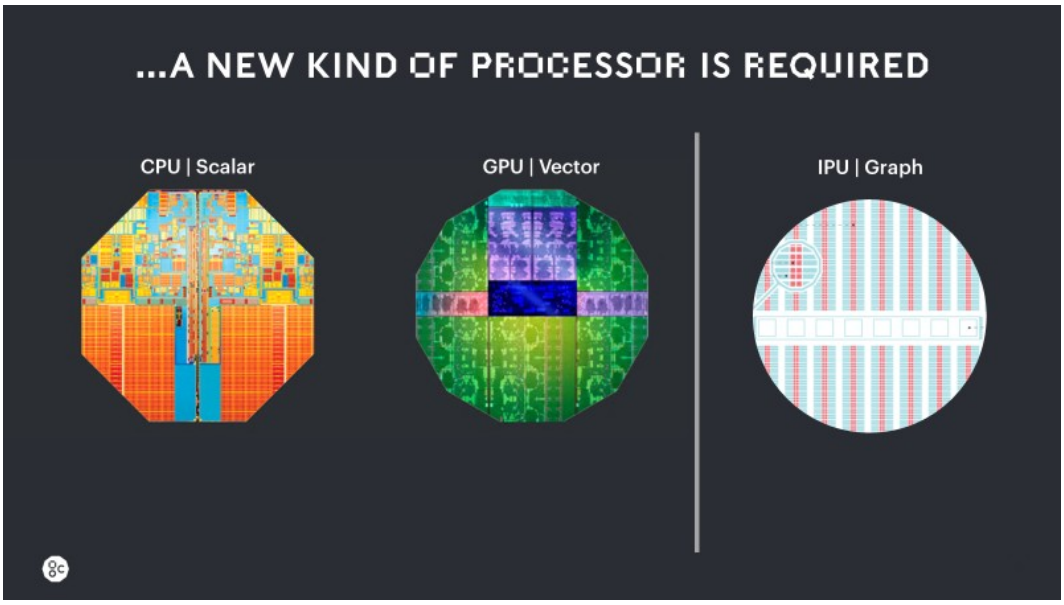
70多年来，我们一直在给计算机下达指令，告诉它在程序中执行各种操作。而如今，机器智能浮出水面，计算机开始利用数据自我学习。这是自微处理器诞生以来，或者自电子计算机问世以来，所发生的最大变化。

在过去几年中，我们看到了机器智能系统的巨大进步。通过使用先进的机器学习，我们现在可以准确地识别图像，我们正在建立可以理解语言结构的模型。这些系统已经产生了巨大的影响。尽管机器智能才刚刚起步，但在感知能力上，新型的机器智能模型正在不断提供更高的准确性和效率，正在形成各种先进的机器智能系统，这些系统在很多特定的任务上已经超过了人类。基于Attention网络的transformer，例如BERT，在未来需要更高效的学习方法，例如利用稀疏模型结构，更有效地扩展到数十亿或数万亿个参数，这对理解更丰富的语言是非常必要的。我们需要开发技术，构建可以从经验中学习的机器智能。

下图说明，机器智能模型的总体规模呈指数增长，大约每3.5个月翻一番。此外，计算需求相对于模型大小来说，其比例有时超过2的乘方。综合来看，机器智能的计算需求正在显著增长。扩大模型规模可以提高模型精度。机器智能的创新在于通过越来越少的参数获得更高的精度，因此separable convolution和sparse transformer等技术正在大行其道。

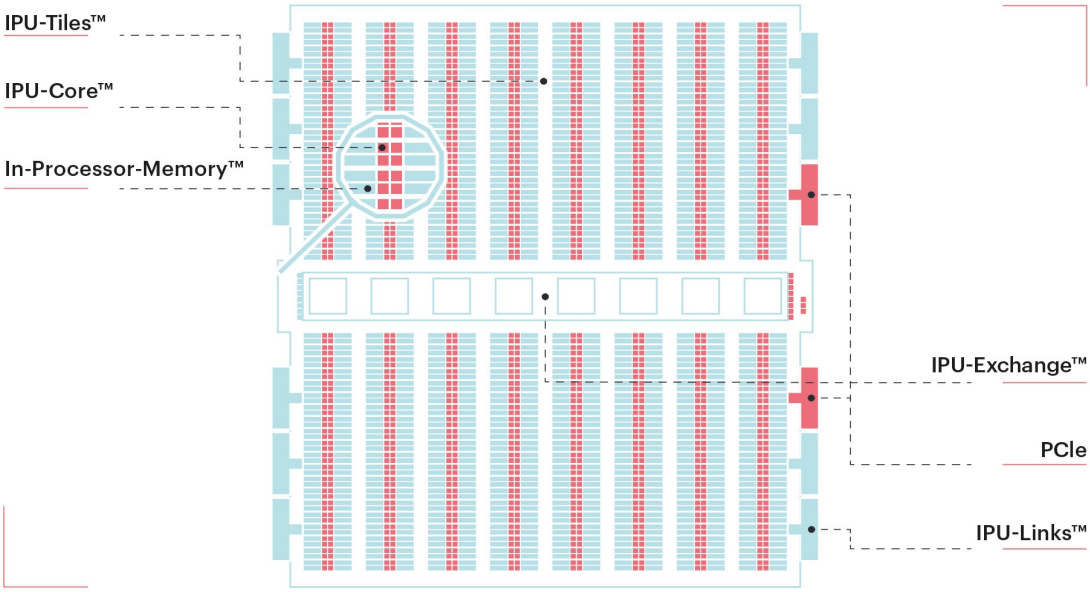


在Graphcore，我们与许多机器智能领域的领先创新者合作。他们告诉我们，当今的CPU和GPU无法为他们提供足够的支持。他们需要一种新型的处理器，以便更有效地支持更复杂的知识模型，快速进行训练和推导。这就是我们创建智能处理业务单元IPU的原因。IPU作为一种全新的处理器，可满足这些创新者的需求，帮助他们在机器智能领域取得新的突破。



IPU支持跨数千个独立线程的大规模并行处理。再加上庞大的内存带宽，可以使成千上万的并行线程全部独立运行。

Graphcore Colossus GC2 IPU架构



Graphcore IPU专为机器智能新世界而设计，它有如下关键的特性：

- 旨在进行跨越数千个独立处理线程进行灵活的地址访问
- 利用In-Processor-Memory把整个知识模型存放在处理器内部，可最大限度提高内存带宽并把延迟降到最低
- IPU-CORE支持极高性能的混合精度浮点算法，该算法针对张量进行了优化，并为生成随机数提供硬件支持
- 每个IPU-Core最多可以运行6个独立的硬件程序线程，从而可使单个IPU中的知识模型上并行运算超过7,000个独立程序
- IPU支持现有的传统DNN模型，并为机器智能的创新开辟了新的途径，例如概率系统和进化策略
- 重要的是，IPU可以进行高效的训练和推导，并且为机器智能计算提供了最低的延迟

每个IPU上有1216个高性能机器学习处理器内核（IPU-Core），每个内核包含6个处理器线程。因此，在计算过程中，一共支持7296个处理器线程全速并行运行，这能让每个IPU-Core都提供了超过100 gigaflops的计算性能。

每个IPU-CORE都与快速的256kB本地处理器内内存紧密耦合，并能一起构成IPU-Tile。6个处理器线程都可以直接访问In-Processor-Memory。总的来说，IPU大约有300MB的内存，内存带宽为45TBps，并且没有片外内存，因此IPU避免了内存带宽的问题。IPU支持EEE FP32和FP16数据格式，可以处理深度神经网络中的所有层，并且能够进行250 teraflops的混合精度的16/32位算术计算。

每个内核都直接连接到IPU-Exchange，这是位于硅片中间的交叉开关，可以传输62.5Tbps的数据。IPU通过16通道PCI Express接口连接到主机处理器。IPU有10个高速IPU-Link™端口，每个端口在每个方向和8个通道上具有160Gbps的带宽。IPU-Link端口允许IPU直接交换数据，而无需通过主机处理器或主机内存交换数据，从而使多个IPU能够以约450GB/s的IPU-Link带宽高效地工作。IPU还可在硬件层面上生成用于配置熵注入的随机数。

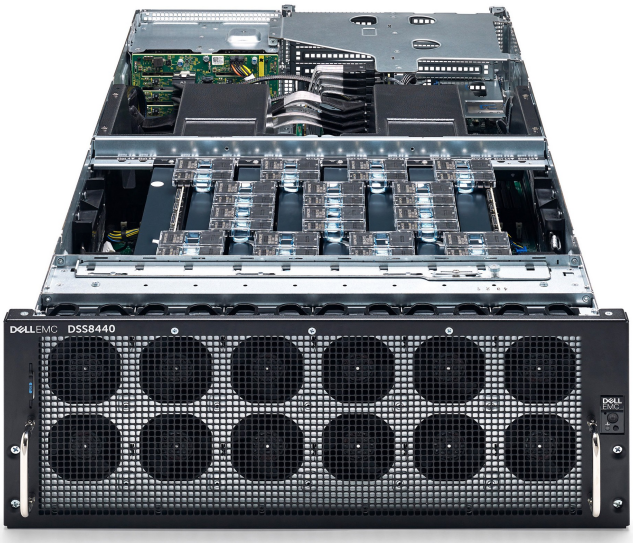
Graphcore C2 IPU处理器卡

Graphcore C2 IPU处理器PCIe卡是包含两个IPU的双插槽全高PCI Express Gen3/4卡。共有80个IPU-Link，每个Link以32Gbps的速度传输，总共提供约2.5Tbps或450GB/s的芯片到芯片带宽。在C2卡上，192GB/s的IPU-Link带宽用于连接C2卡本身上的两个IPU，而256GB/s的IPU带宽用于将C2卡连接在一起，从而使跨多个C2卡的IPU-Processor能够形成连续的处理资源，同时进行工作（请参见下面的DSS 8440系统拓扑图）。

卡上一共有两个IPU处理器和2416个单独的处理器内核，总共拥有200 TFLOPS的处理能力。这样会导致315W卡在DSS 8440 IPU服务器中会被被动制冷。

戴尔DSS 8440 IPU服务器

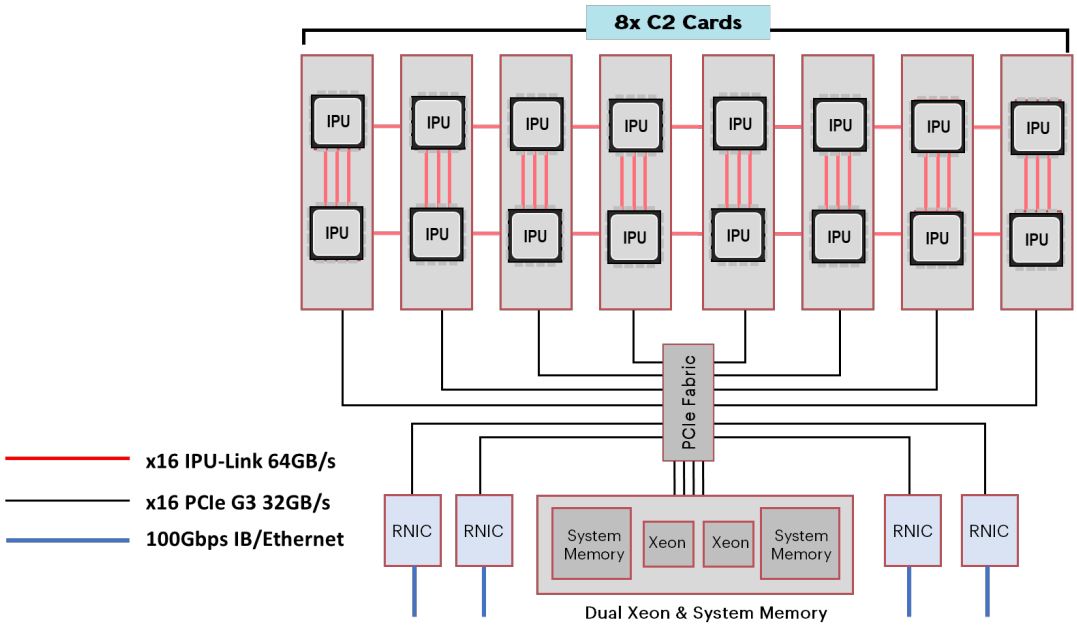
DSS 8440 IPU服务器是两插槽的4U服务器，旨在为机器智能应用提供卓越的性能，8个Graphcore C2卡（双IPU）可提供高达1.6PetaFLOP的混合精度机器智能计算能力。



DSS 8440 IPU服务器利用IPU-Link的“阶梯式”拓扑，在C2卡上的两个IPU之间具有192GB/s的“梯级”，在C2卡之间的总体双向带宽为256GB/s。此配置允许在IPU之间高速共享模型参数、activation或数据

C2卡通过高速PCIe Gen3交换矩阵连接到DSS 8440 IPU-Server 的主机Xeon CPU。DSS 8440 IPU服务器中实际上有四个独立的PCIe交换机，但在拓扑上显示为“黑匣子” PCIe结构。正如将在Poplar SDK一节中更深入地解释的那样，主机IP不直接参与机器智能的处理任务，因为IPU能够完全独立地运行ML程序。

DSS 8440 IPU服务器还配备了大量的本地存储，并且IPU能够直接使用此主机系统的内存，这对于涉及大型（数十亿参数）模型的应用特别有用。



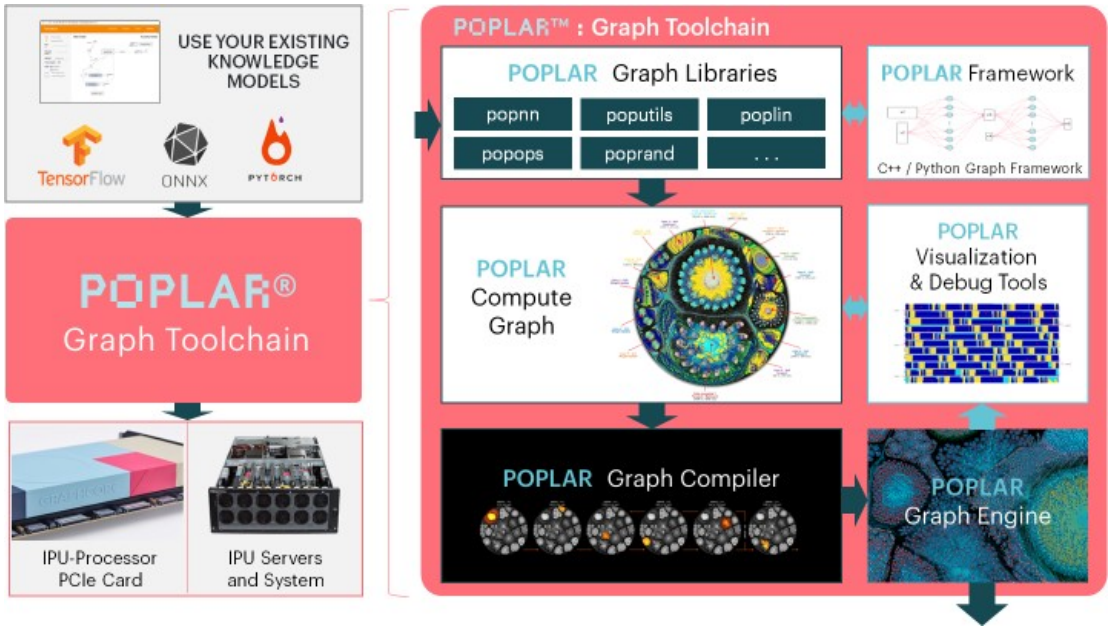
对于分布式训练应用，DSS 8440 IPU服务器还提供多个100Gbps网络链接，以实现服务器到服务器的可扩展性，默认的配置是双100Gpbs InfiniBand适配器。在讨论机器智能模型的实现时，或设计训练或推导数据和模型的并行配置时，经常会使用这个DSS 8440 IPU-Server拓扑。

由于具备上述这些特性，毫无疑问，DSS 8440 IPU服务器是机器智能训练和推导应用的理想之选。

Poplar® SDK

DSS 8440 IPU服务器使用Graphcore的Poplar SDK构建机器智能应用，可为当今的领先模型提供开箱即用的先进性能。Poplar SDK可与TensorFlow、Pytorch和ONNX等广泛使用的框架一起使用。它支持高阶的机器智能图描述，可编译加载到IPU上的优化的Poplar图和相关的控制程序。大规模的处理器内内存意味着可以充分利用庞大的处理器内带宽，可将整个模型加载到IPU上。Poplar SDK还为模型并行和pipelining提供广泛的支持，例如支持具有数十亿参数的下一代NLP模型。Poplar SDK的具体功能将在下面进行详细介绍。

Poplar SDK支持一系列标准框架。通过TensorFlow，Poplar SDK可直接接受XLA图，并将XLA编译输出为Poplar图和控制程序。其他ML框架的输出可作为中间表征被PopART（Poplar Advanced Runtime）接受。PopART除了本身就是一个ML框架外，还提供了与其他ML框架连接的广泛灵活性。



PopLibs是Poplar的标准库，具有各种运算符和功能，可供开发人员定制。其源代码已经提供给客户使用，PopLibs将在2020年开源，以支持IPU社区分享在IPU上的各种创新。PopLibs的特性包括：

- popnn-神经网络函数（activation function、pooling、loss）
- poplin-矩阵乘法和卷积函数
- popops-Pointwise 和 reduction operator

- poprand-随机数函数
- poputil-构建图的实用函数
- popsolver –模型求解函数

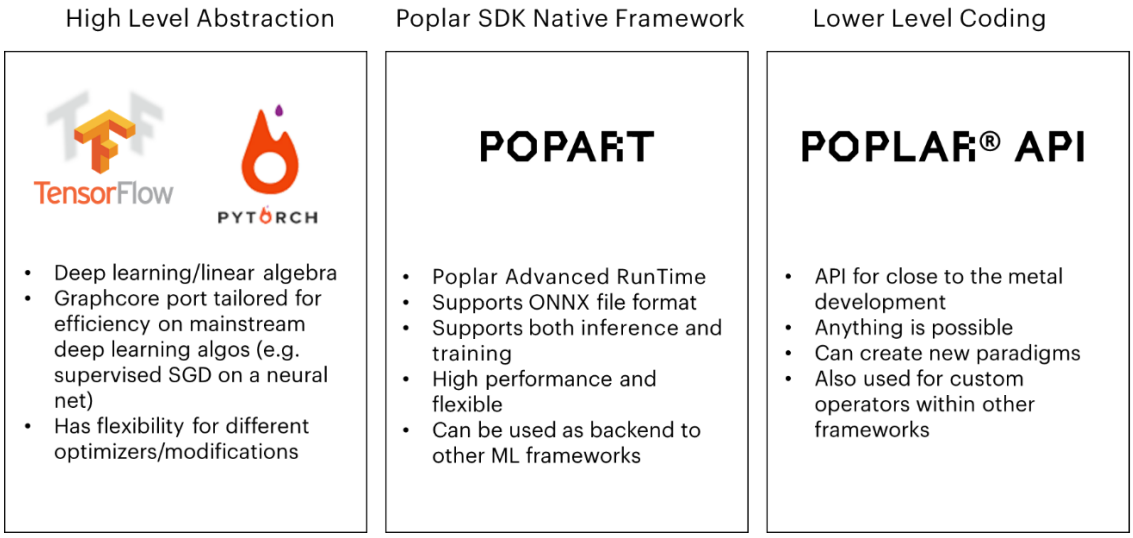
基于Poplar SDK的灵活性

Poplar SDK提供了三种不同级别的开发体验，具体取决于开发人员的背景和目标。下图描述了Poplar SDK三种不同的使用方式：

- 高级抽象-这为开发人员提供了一条简单的路径，使他们可以使用TensorFlow、Pytorch或ONNX等标准ML框架工具中描述的现有图。无需更改，即可通过Poplar SDK为IPU实施和优化现有的ML框架模型。

- PopART-Poplar Advanced RunTime是Poplar SDK本机图的runtime，可处理用于推导和训练的高级计算图。PopART为IPU特定的图优化提供支持，并提供其他支持，使开发人员更接近目标硬件。

- 较低级别的编码底层开发-Poplar API提供直接的C++编码体验，允许开发人员直接针对IPU为图进行编码，并通过完全控制明确指定tensors、vertices和edges。这可用于创建自定义的operator、自定义的网络层或用于其他的优化目的。开发人员可以访问PopLibs源代码以获取示例和灵感，以加快开发速度。



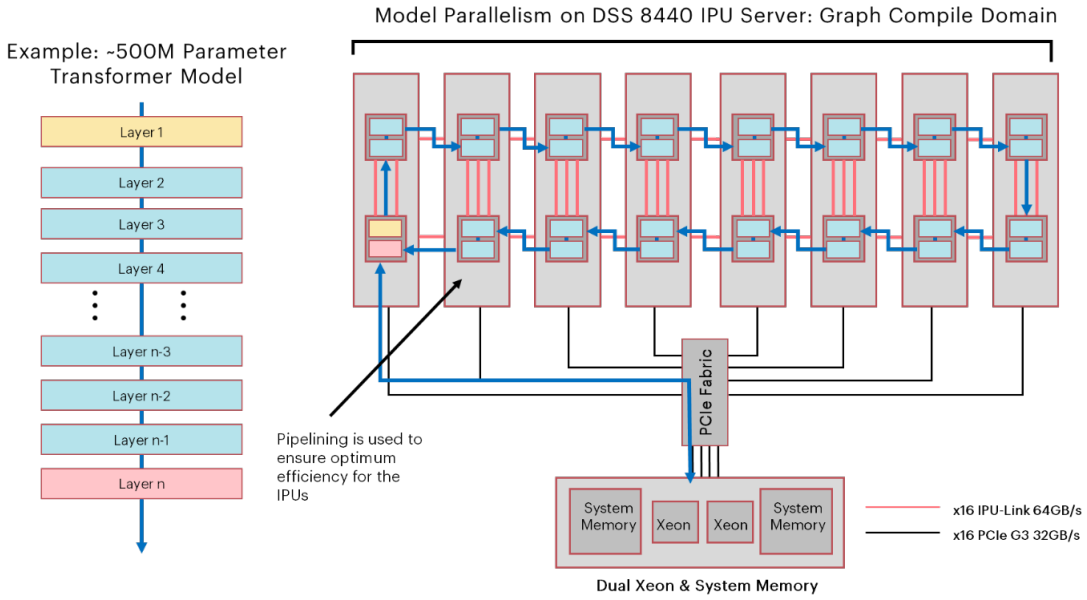
基于IPU服务器的机器智能

DSS 8440 IPU服务器为用于训练和推理推导的机器智能知识模型提供了新的可能性，所支持的知识模型涵盖了通常的数据并行体系结构，以及高效的pipelining模型并行体系结构。当然，最佳实施方法取决于模型本身，但是由于IPU及其Poplar SDK为机器智能处理提供了一个全新的平台，用户很快就会发现这是一个可以更好优化模型性能的机会。市场在某种程度上已经适应了传统的处理体系结构，但同时也渴望在机器智能方面取得突破。在某种程度上，通过IPU以全新的方式支持模型架构，创新将会变得更加容易。

通过Poplar SDK中的创新功能，用户可以充分利用IPU的独特架构优势。

使用图编译域（GCD）进行模型并行

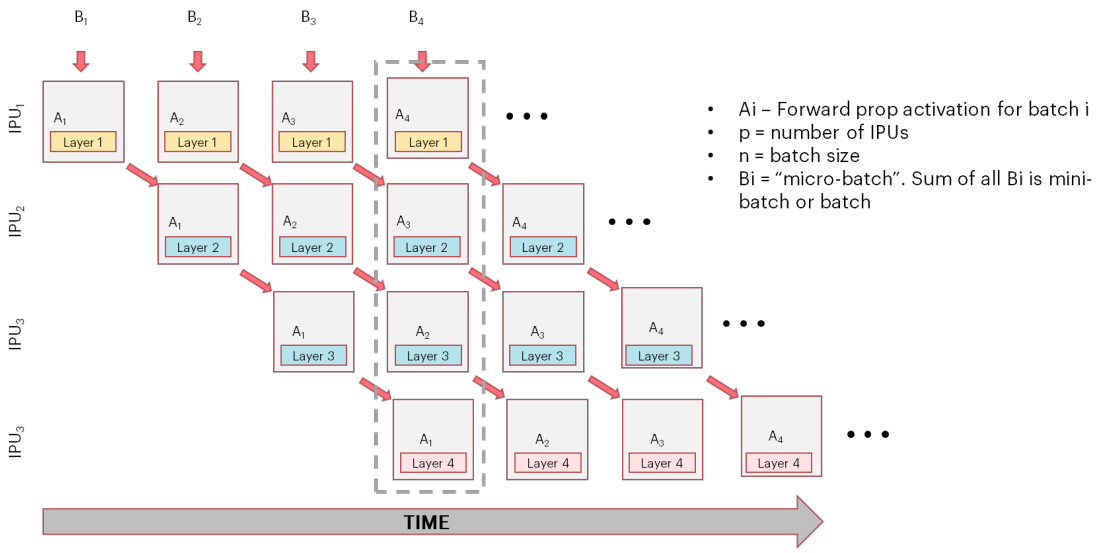
图形编译域（GCD）定义了能够定义Poplar SDK能够编译模型并行的IPU数量，能使较大的模型充分利用高带宽的IPU处理器内内存。GCD可以覆盖DSS 8440 IPU服务器中的16个IPU，这意味着可以通过服务器的16个IPU对大型模型进行分区。中小型模型可以完全容纳在单个IPU的处理器内内存中，从而可以跨越16个IPU实现数据的并行处理。



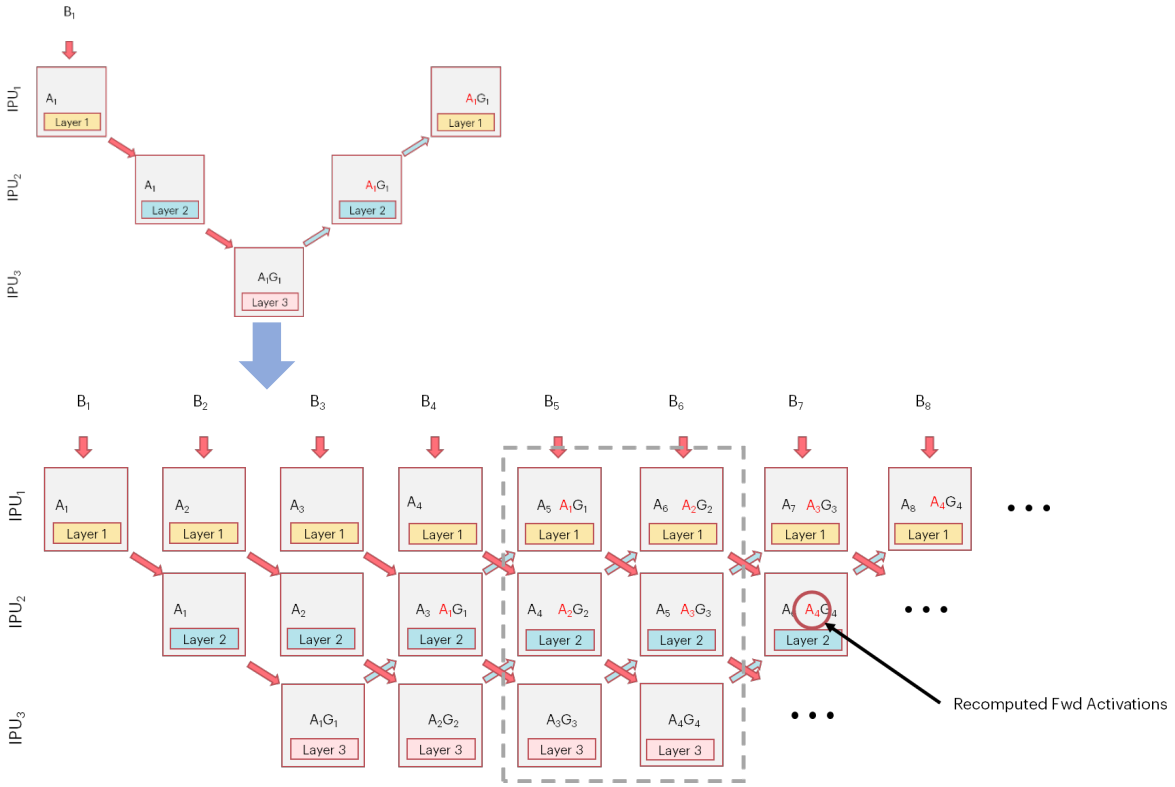
结合pipelining（见下文），GCD对于最大程度减少等待时间并最大化超大图的计算特别有用。上图显示了一个基于transformer的大型模型（通常是NLP用例），该模型已在DSS 8440 IPU服务器的所有16个IPU上进行了“分片（sharded）”。

Pipelining

Pipelining与模型并行实施（通过GCD）相结合，可最大限度地利用模型并行处理中涉及的所有IPU，可以提高处理器效率，并提高吞吐量和延迟性能。下图着重说明了在模型并行中如何使用pipelining进行推导（虚线框表示pipeline主体中最大程度利用所有IPU的点）。Pipeline可以跨任何深度（DSS 8440 IPU服务器提供高达16个IPU）。当activation传递到后续IPU时，模型并行系列（IPU1）中的第一个IPU接受第一个batch（B1）。IPU1继续接受新batch，依次为这些batch传递activation。第二个IPU接受activation并处理下一层，依此类推。pipelining使所有IPU都处于活动激活状态，并且通过batch4（B4），所有IPU都可以并行进行主动处理。该处理pipeline对推导推理来说相对简单，不过对于基于back propagation的训练来说将会比较复杂。

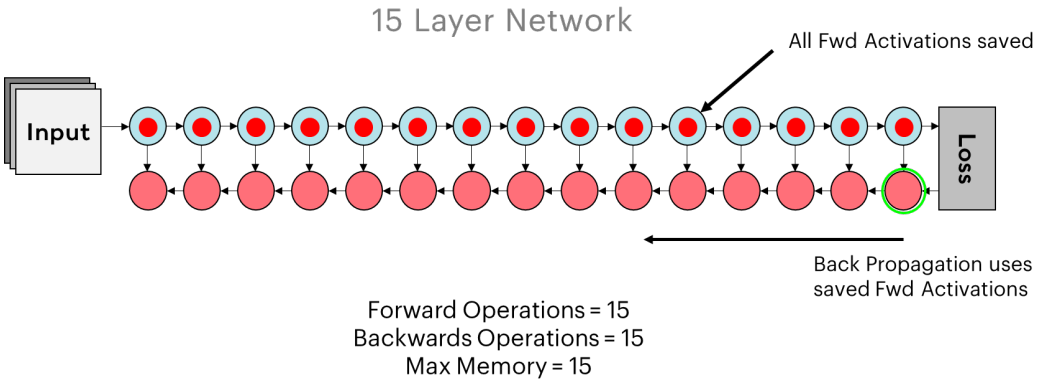


对于训练，pipelining需要既适应forward pass，又适应back propagation和权重更新。下图显示了单个forward pass和back propagation，然后显示了具有并行重叠批处理的完整pipelining。虚线框显示了pipeline的主体（可以是任何深度，更大的深度可以增加批处理的规模），这里最大程度地利用了所涉及的IPU处理前向activation，从存储的activation输入中重新计算先前的activation，并计算梯度更新。通过允许开发人员充分利用高带宽的处理器内内存，重新计算（参见下方）是此方法的关键实现因素。

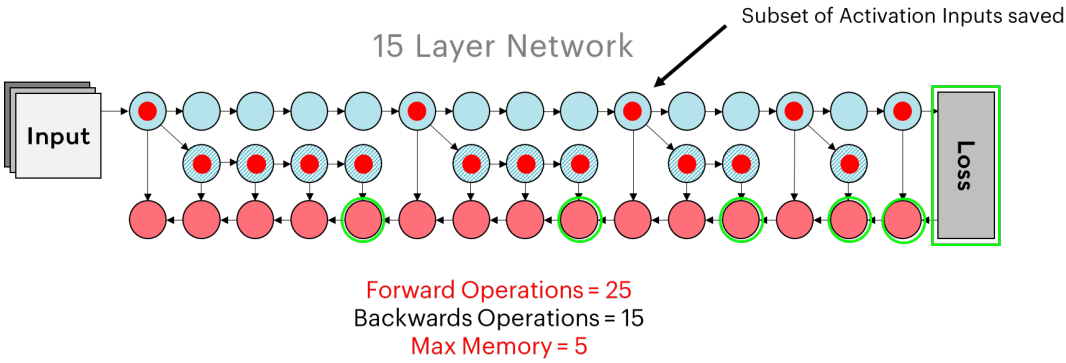


重新计算（Recomputation）

IPU体系结构的主要优势是高带宽/低延迟的InProcessor-Memory。重新计算是Poplar SDK中的功能之一，它可以最大化IPU内存资源和IPU的高性能计算能力。在多层模型中，activation是逐层计算的，通常保存为中间结果。通过重新计算，Poplar SDK会选择特定的activation输入，并根据backward pass的要求使用这些输入重新计算activation。下面第一个图说明了重新计算的常用方式，在图中所有前向activation被保存，以支持backward pass中的权重更新计算。



对于重新计算中的算力TFLOP支出，Poplar SDK通过保存选定的activation输入并在内存的保存中进行优化，来有效地利用宝贵的处理器内内存。下图显示为了进行权重更新的backward pass计算，所保存的activation输入子集如何被用于重新计算所有必需的activation历史记录。

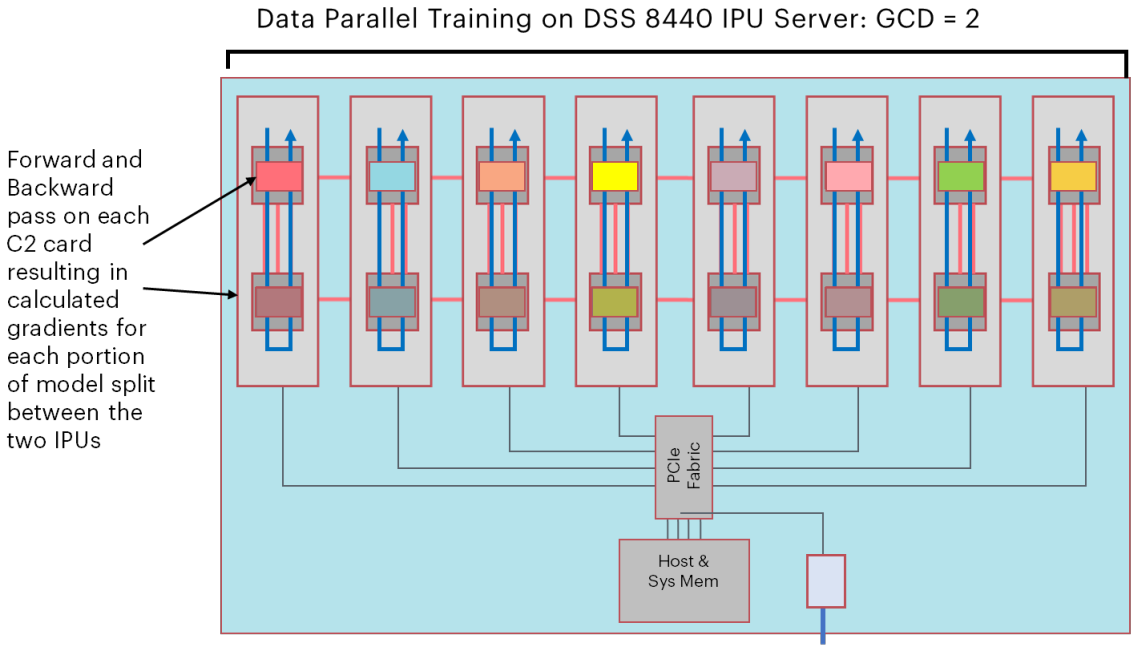


结合IPU架构的独特优势，重新计算可带来多种好处：

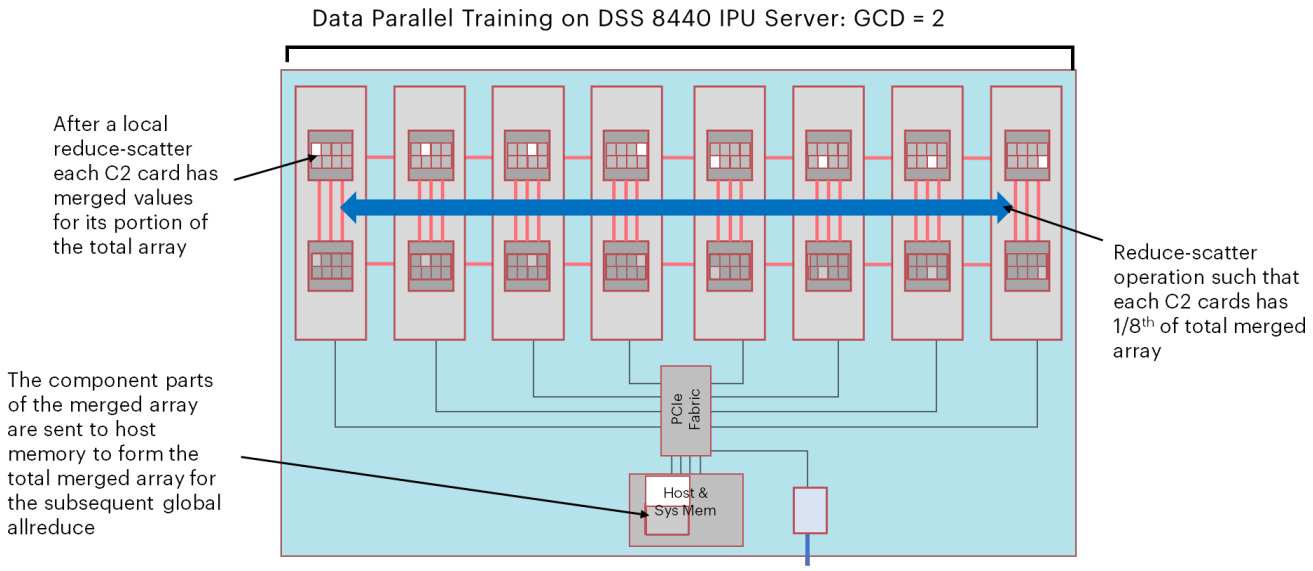
- 重新计算可以将用于ResNet和DenseNet的训练NN所需的内存减少多达5倍，能够充分利用紧密耦合的IPU InProcessor-Memory
- Poplar SDK支持受约束的方法，可最大程度地减少FLOP的目标内存开销
- 计算额外的FLOP所需的额外时间可以通过使用较大的batchebatch来抵消

数据并行训练或多副本训练

训练DNN的第一步是对输入数据进行前向推导。跨所有模型副本的输入数据量由minibatch的规模确定。通过将结果与正确答案进行比较，可以计算出forward pass之后的误差（loss）。该误差在back propagation步骤中用于计算权重梯度并更新模型参数。在DSS 8440 IPU服务器中使用模型的多个副本进行训练（并通过使用服务器上可用的100Gbps端口扩展到多个DSS 8440 IPU服务器上），是数据的并行训练，并且可以与模型并行进行组合，其中每个模型的副本也跨越多个IPU（GCD> 1）被分发。下图显示了一个数据并行训练的示例，其中GCD = 2（在1个C2卡上每个模型副本跨越2个IPU）。由于每个模型副本都并行处理不同的输入数据，因此从back propagation计算出的梯度阵列在每种情况下都将有所不同（由图中每个“顶部” IPU中的不同颜色框表示）



目标是使用一个allreduce流程将所有结果组合到跨越所有模型副本的单个模型参数更新中（由下图中的白色方框指示）。Poplar SDK支持DSS 8440 IPU服务器中的各种allreduce实施。



一旦完成本地的allreduce，则在Infiniband支持下可以完成全局的allreduce。

用于扩展多个DSS 8440 IPU服务器的InfiniBand

除了使用IPU-Link进行高带宽和低延迟的服务器内通信外，DSS 8440 IPU服务器还可以使用单个100Gbps InfiniBand适配器进行灵活的横向扩展，构建多个具有200Gbps双向带宽的服务器。

摘要

在过去的两年中，Graphcore与大型公司以及许多世界领先的创新者合作，开发IPU处理器和Poplar软件堆栈，以支持培训和推导。Poplar允许创新者使用TensorFlow、ONNX和PyTorch等现有的高级机器学习框架轻松地在IPU上运行现有的模型。

IPU开箱即用，能够为深度学习模型提供最先进的性能。我们邀请您在https://github.com/graphcore上探索我们公共的benchmark，并探索机器智能工具箱中Graphcore IPU的强大功能。

Graphcore相信截至发布之日，此处提供的信息是准确的。不过随着时间的推移和技术的发展，这些信息可能变得过时、不完整或不准确。请经常返回查看以确保您能获取最新的信息。

开始使用IPU

获取更多Graphcore资讯，敬请关注我们的官方微信公众号以及新闻发布平台。

Graphcore 官方微信号

Graphcore 官方新闻发布平台

或者访问Graphcore官方网站 graphcore.ai