

# AI编译器-系列之前端优化

# 布局转换



# ZOMI



# Talk Overview of Frontend Optimizer

## I. AI 编译器前端优化

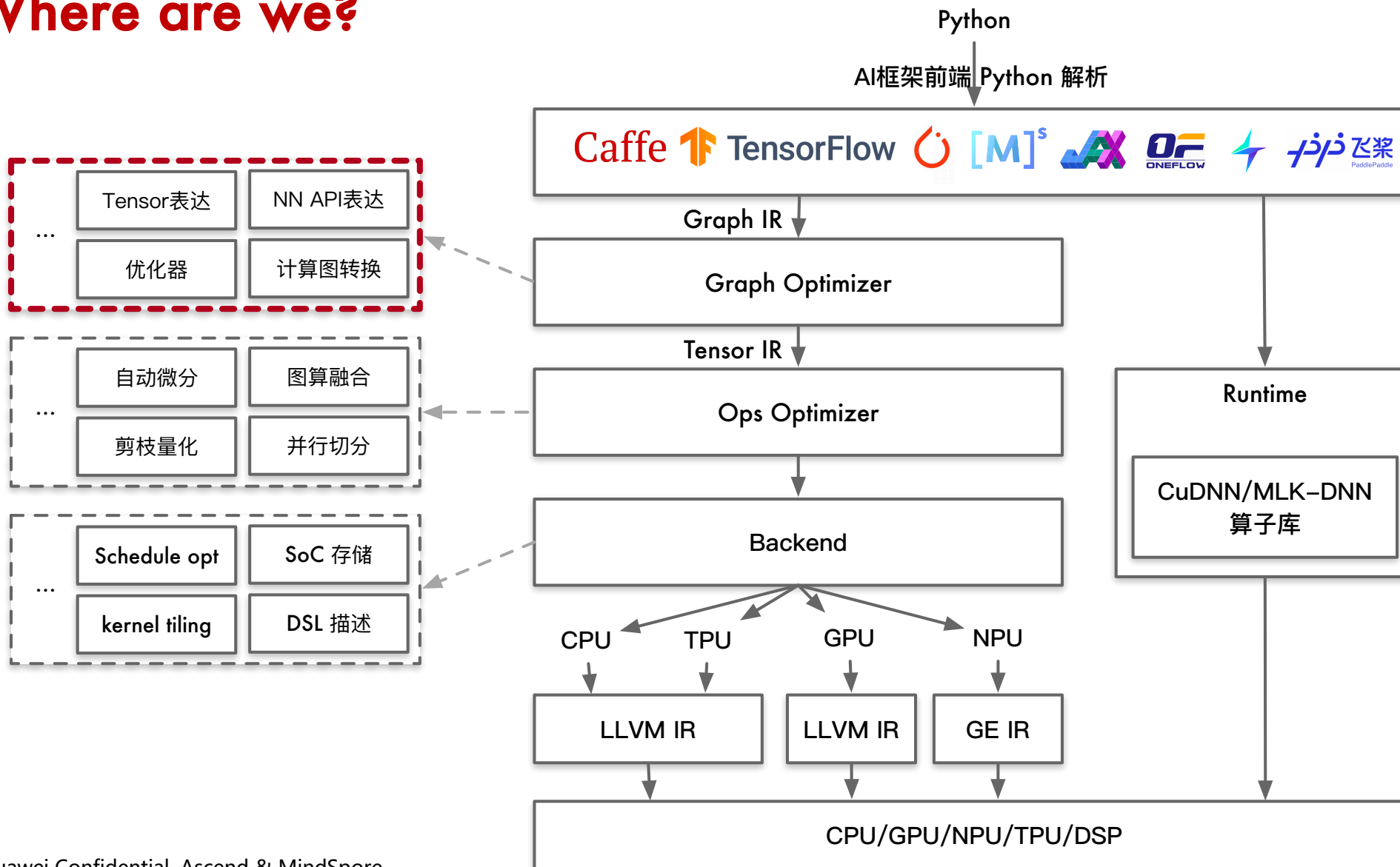
- 图层 - Graph IR
- 算子融合 - OP Fusion
- 布局转换 - Layout Transform
- 内存分配 - Memory Allocation
- 常量折叠 - Constant Fold
- 公共子表达式消除 - CSE
- 死代码消除 - DCE
- 代数简化 - ARM

# Talk Overview

## Layout Transformation – 布局转换

- 数据内存排布
- 张量数据布局
- NCHW与NHWC
- 华为昇腾数据排布
- 编译布局转换优化

# Where are we?

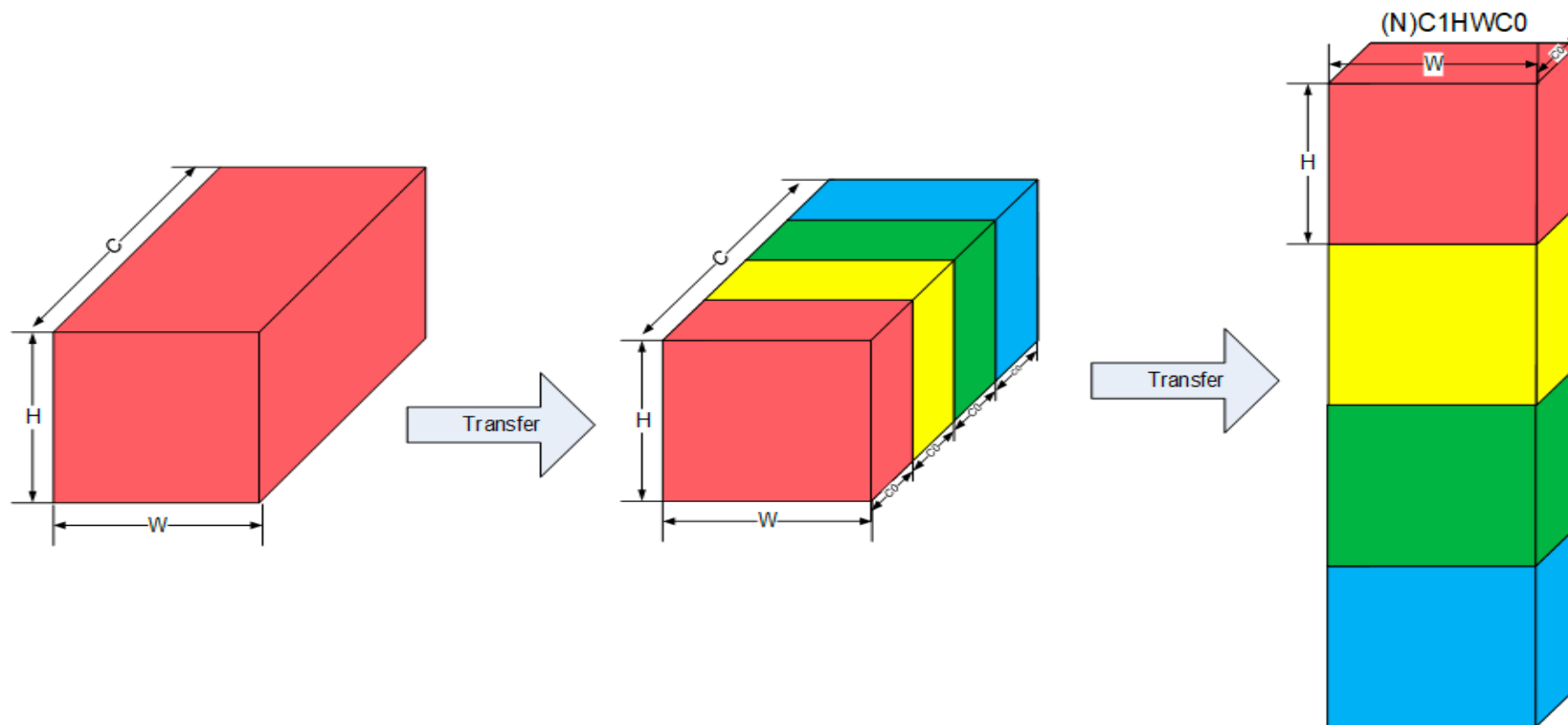




# 华为昇腾 数据排布

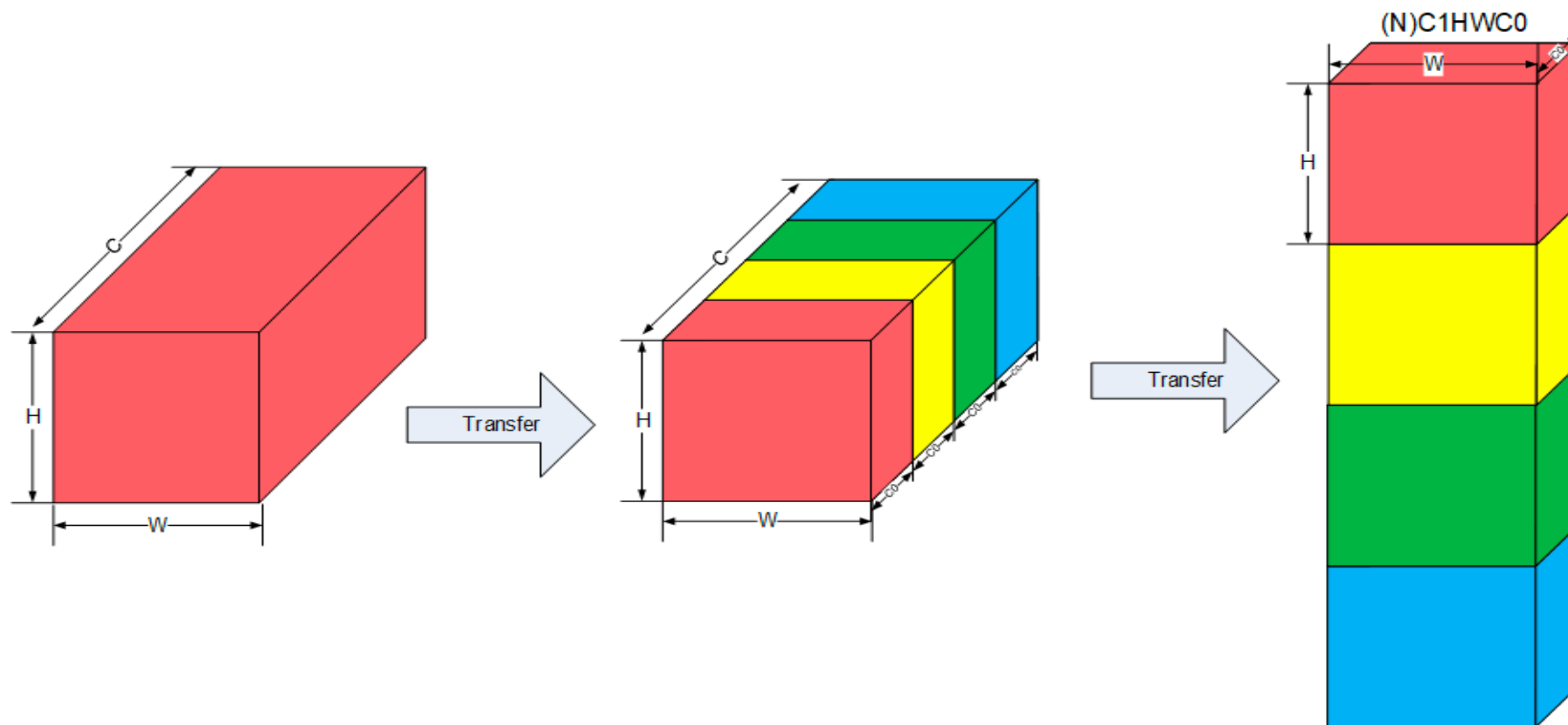
# NC1HWC0 (5HD)

- 昇腾AI处理器中，为了提高通用矩阵乘法（GEMM）运算数据块的访问效率，所有张量数据统一采用NC1HWC0的五维数据格式：



# NC1HWC0 (5HD)

- C0与达芬奇微架构强相关，等于AI Core中矩阵计算单元的大小，对于FP16类型为16，对于INT8类型则为32，这部分数据需要连续存储。其中 $C1 = C/C0$ ，如果结果不整除，向上取整。



## NHWC -> NC1HWC0

1. 将NHWC数据在C维度进行分割，变成C1份NHWC0。
2. 将C1份NHWC0在内存中连续排列，由此变成NC1HWC0。

```
1  
2 Tensor.reshape( [N, H, W, C1, C0] ).transpose( [0, 3, 1, 2, 4] )  
3
```

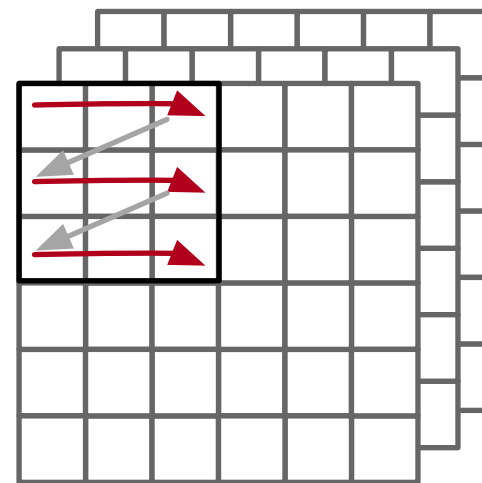
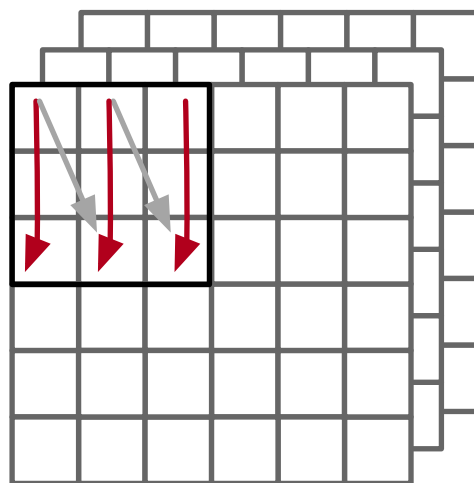
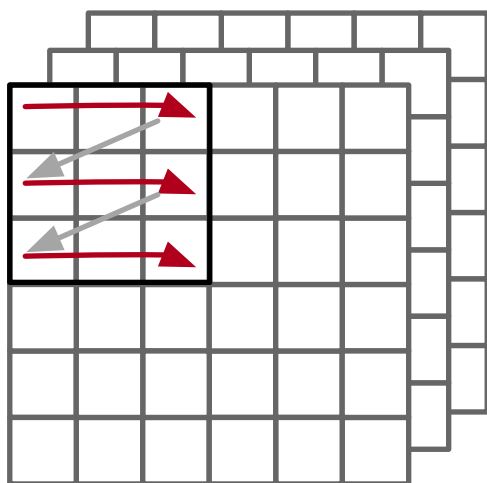
## NCHW -> NC1HWC0

```
1  
2 Tensor.reshape( [N, C1, C0, H, W] ).transpose( [0, 1, 3, 4, 2] )  
3
```

# FRACTAL Z and FRACTAL NZ

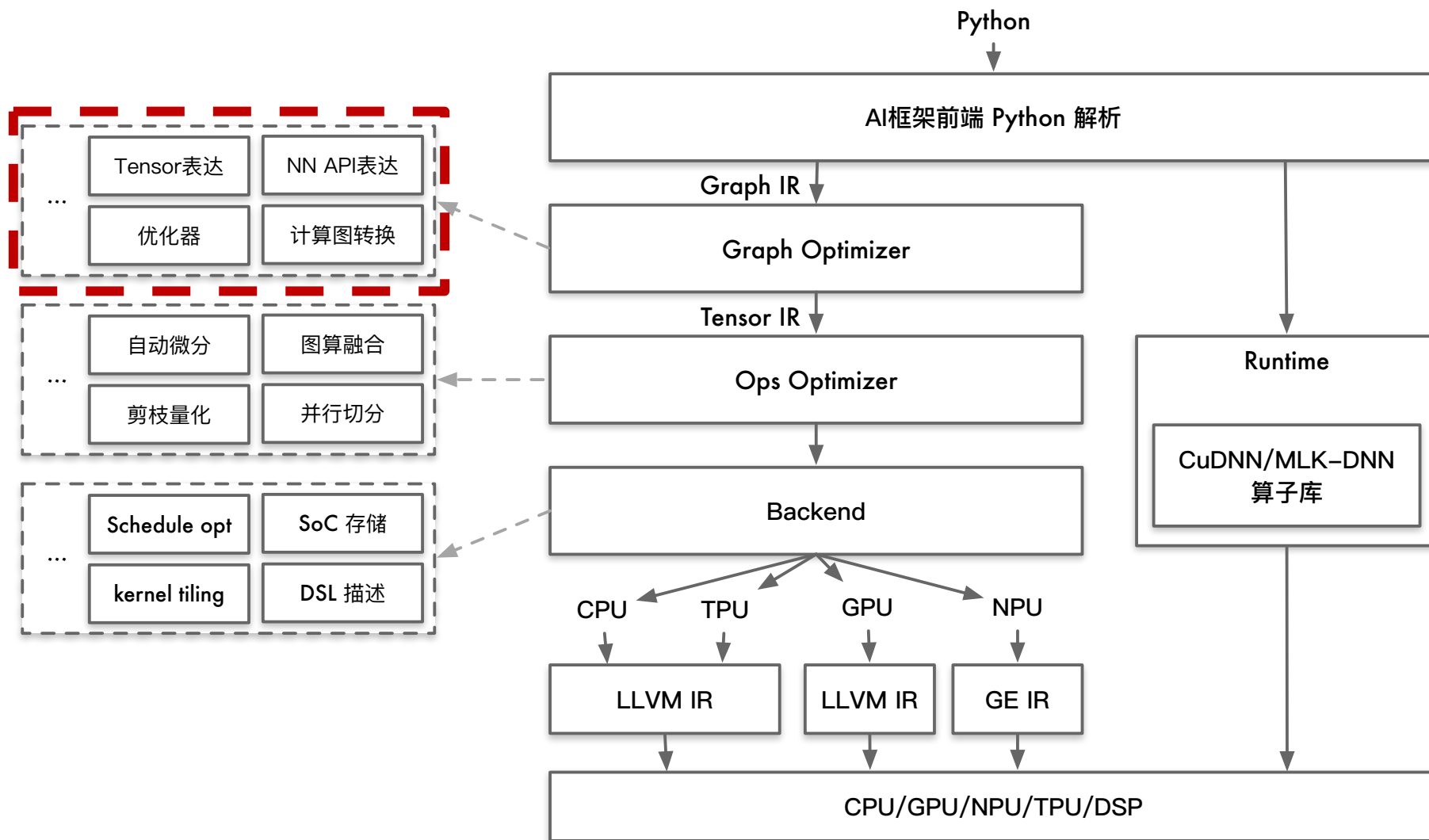
在cube单元计算时，输出矩阵的数据格式为NW1H1H0W0，不同硬件会采用不同的格式：

- 小z大Z：块内按照行排序，块间按照行排序，如Feature Map数据存储
- 小n大Z：块内按照列排序，块间按照行排序，如Weight数据存储
- 小z大N：块内按照行排序，块间按照列排序，如Conv结果输出



# 编译布局 转换优化

# Where are we ?

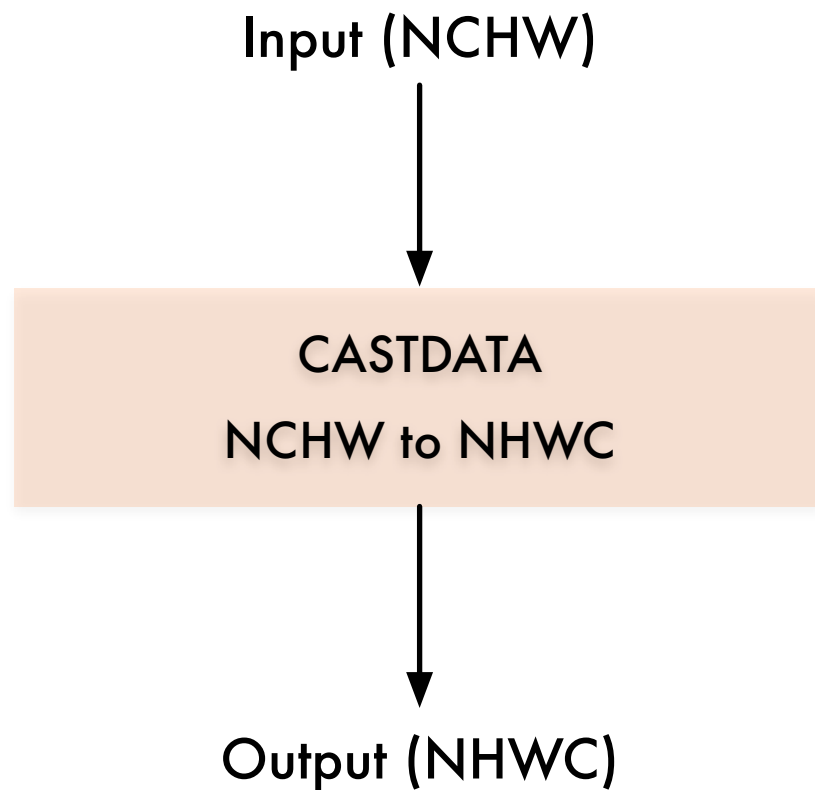
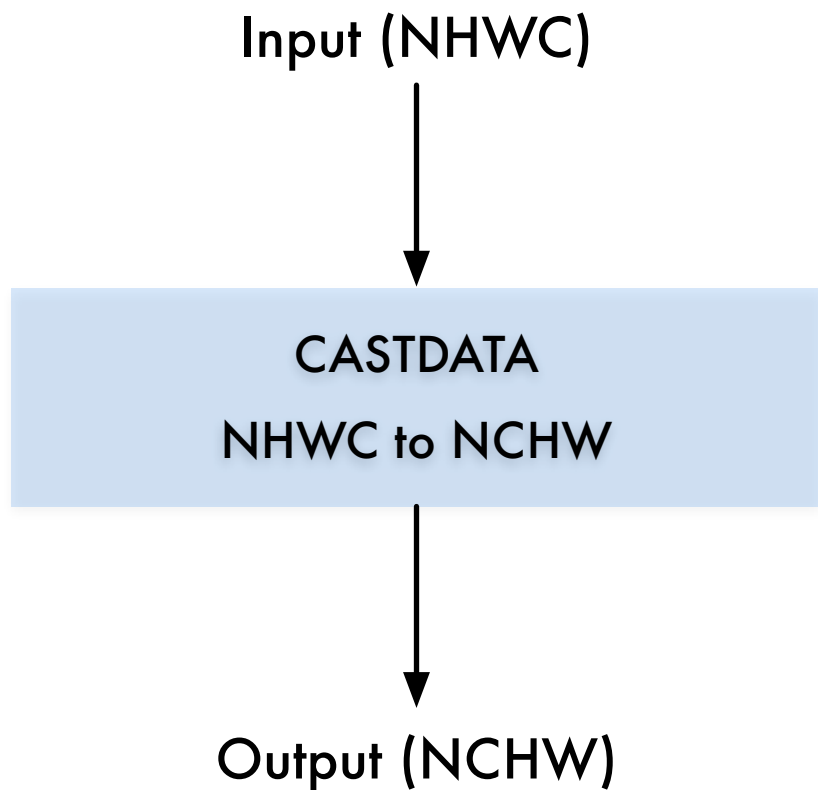


# AI编译器布局优化

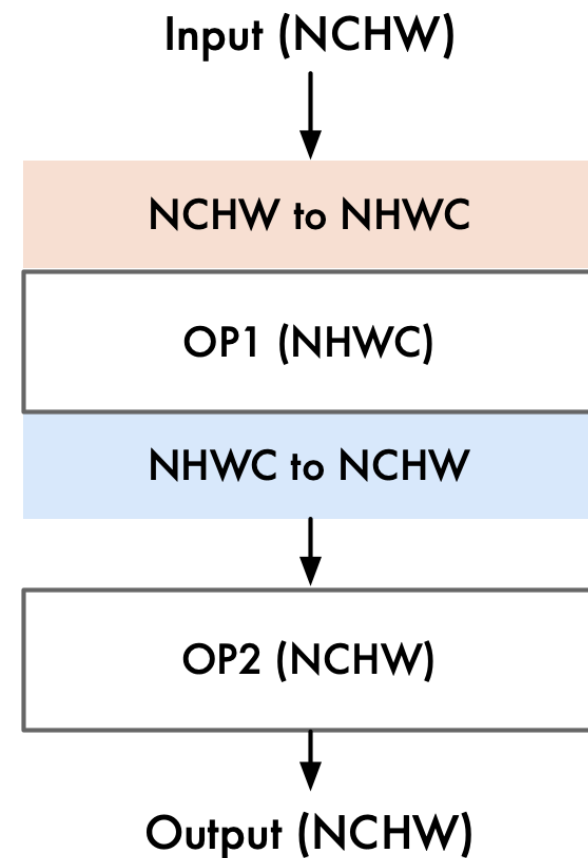
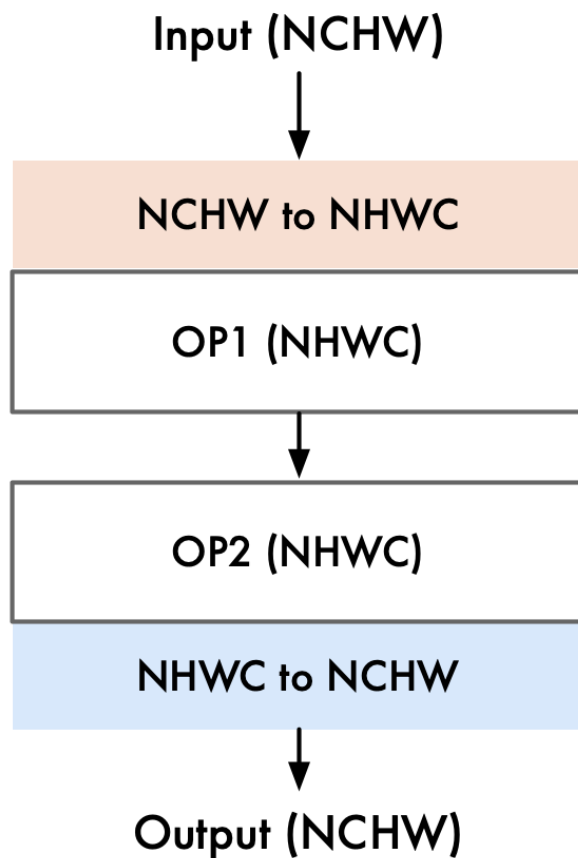
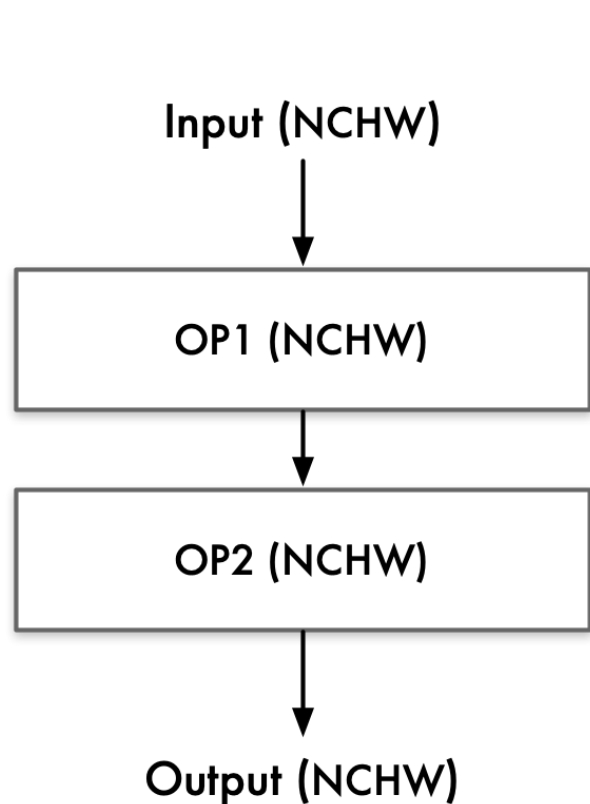
- **目的**：将内部数据布局转化为后端设备友好的形式
  - **方式**：试图找到在计算图中存储张量的最佳数据布局，然后将布局转换节点插入到图中
  - **注意**：张量数据布局对最终性能有很大的影响，而且转换操作也有很大的开销
- NCHW 格式操作在 GPU 上通常运行得更快，所以在 GPU 上转换为 NCHW 格式是非常有效。一些 AI 编译器依赖于特定于硬件的库来实现更高的性能，而这些库可能需要特定的布局。此外，一些 AI 加速器更倾向于复杂的布局。边缘设备通常配备异构计算单元，不同的单元可能需要不同的数据布局以更好地利用数据，因此布局转换需要仔细考虑。AI 编译器需要提供一种跨各种硬件执行布局转换的方法。



# 数据转换节点

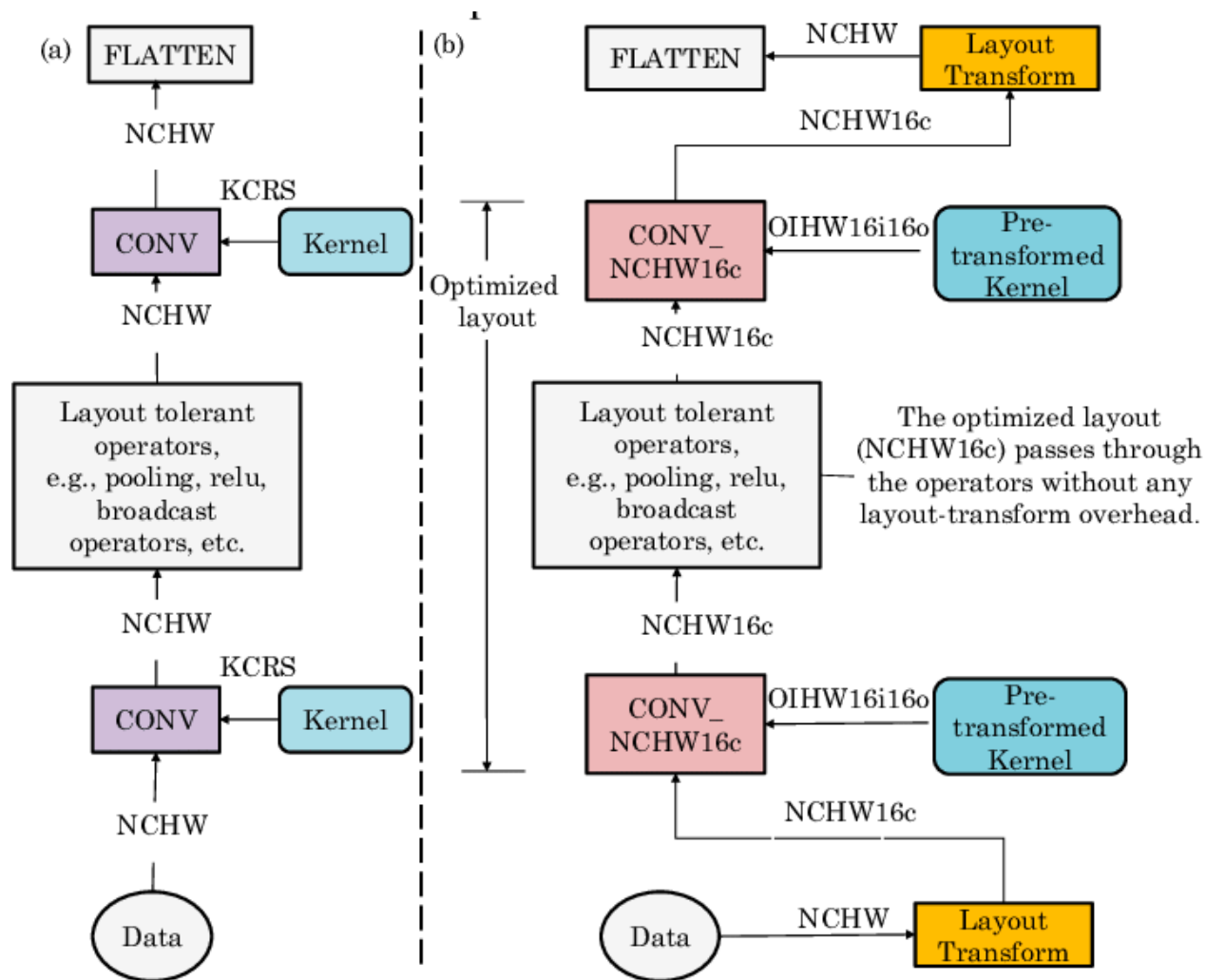


# 常见数据转换节点



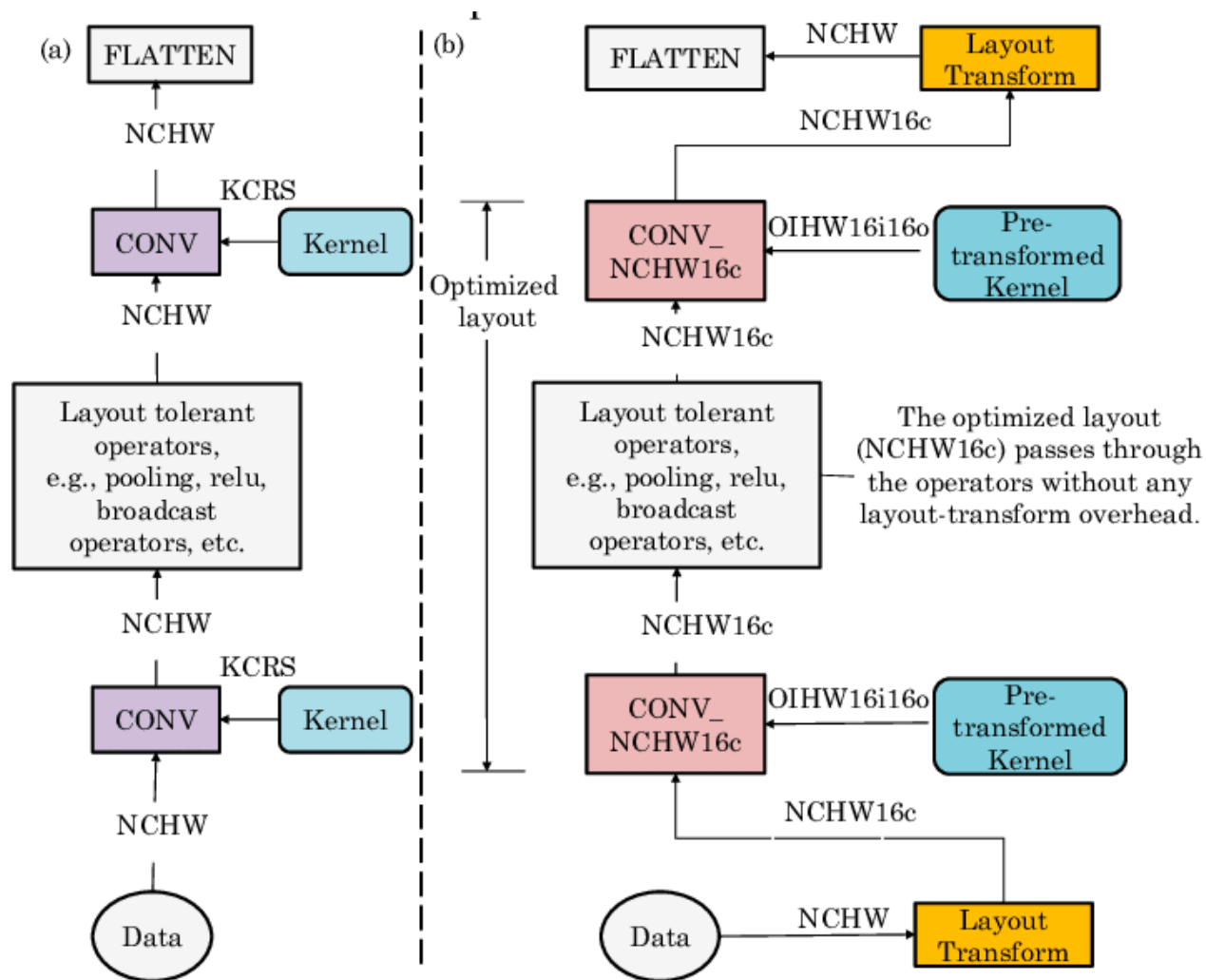
# 训练场景

- 不改变原计算图
- 插入转换算子
- 取消转换算子



# 推理场景

- 不改变原计算图
- 插入转换算子
- 取消转换算子
- 权重布局转换
- 算子替换



# Reference

1. CANN V100R020C20 TBE自定义算子开发指南 <https://support.huawei.com/enterprise/zh/doc/EDOC1100180762/f96da97d>
2. CANN V100R020C20 TBE自定义算子开发指南 (推理) <https://support.huawei.com/enterprise/zh/doc/EDOC1100180762/8e6a99eb>
3. 深度学习NCHW和NHWC数据格式 <https://blog.csdn.net/Dontla/article/details/123141775>
4. [DLComplier] The Deep Learning Compiler: A Comprehensive Survey – 3 <https://zhuanlan.zhihu.com/p/543187086>
5. Pytorch NCHW/NHWC 理解 <https://zhuanlan.zhihu.com/p/556222920>
6. <https://docs.nvidia.com/deeplearning/cudnn/developer-guide/index.html>
7. 深度学习框架zf\_谈谈深度学习框架的数据排布 [https://blog.csdn.net/weixin\\_26854555/article/details/112360638](https://blog.csdn.net/weixin_26854555/article/details/112360638)
8. 华为Ascend昇腾CANN详细教程（一） [https://blog.csdn.net/m0\\_37605642/article/details/125691134](https://blog.csdn.net/m0_37605642/article/details/125691134)
9. Tensor中数据摆放顺序NC4HW4是什么意思，只知道NCHW格式，能解释以下NC4HW4格式吗？  
<https://www.zhihu.com/question/337513515/answer/768632471>
10. 谈谈深度学习框架的数据排布 <https://zhuanlan.zhihu.com/p/149464086>
11. 数据排布格式 [https://support.huaweicloud.com/TIKopdevgd\\_beta/tik1.5\\_10\\_0005.html](https://support.huaweicloud.com/TIKopdevgd_beta/tik1.5_10_0005.html)
12. 数据布局与内存对齐 <https://books.innohub.top/rustinfo/info/alignment>
13. <https://hughiehao.github.io/2021/10/28/%E5%BC%A0%E9%87%8F%E6%95%B0%E6%8D%AE%E5%AD%98%E5%82%A8%E6%96%B9%E5%BC%8F.html>



BUILDING A BETTER CONNECTED WORLD

THANK YOU

Copyright©2014 Huawei Technologies Co., Ltd. All Rights Reserved.

The information in this document may contain predictive statements including, without limitation, statements regarding the future financial and operating results, future product portfolio, new technology, etc. There are a number of factors that could cause actual results and developments to differ materially from those expressed or implied in the predictive statements. Therefore, such information is provided for reference purpose only and constitutes neither an offer nor an acceptance. Huawei may change the information at any time without notice.