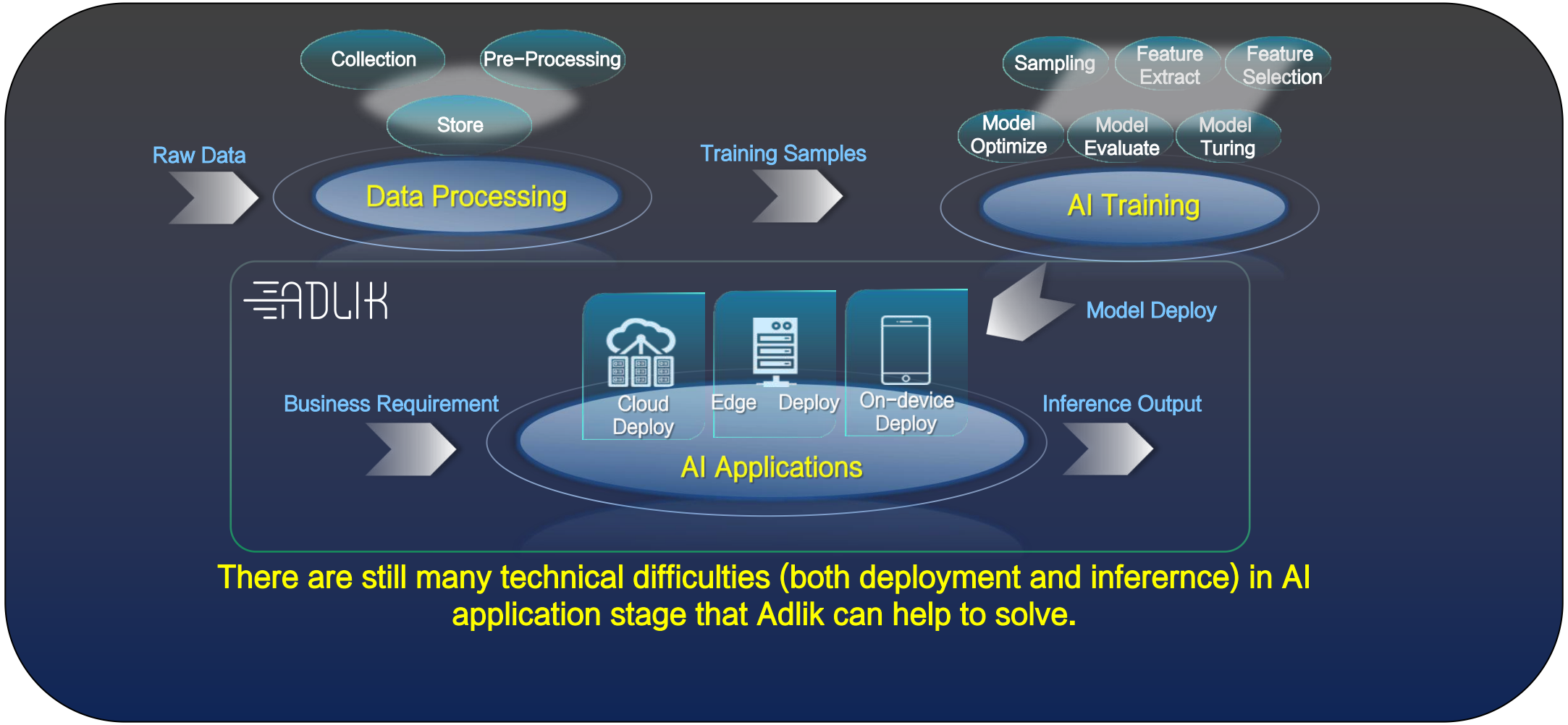
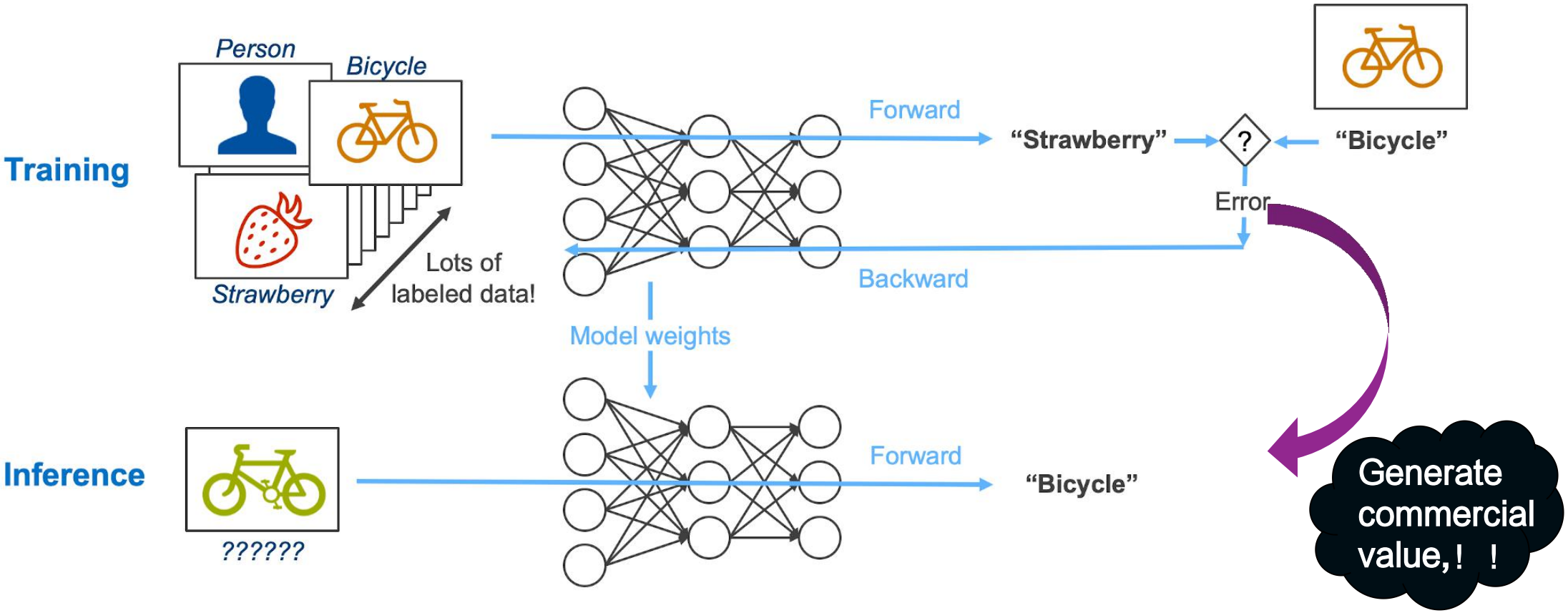
Adlik对深度学习模型推理优化的实践

背景: Three Big Stages in Machine Learning Pipeline

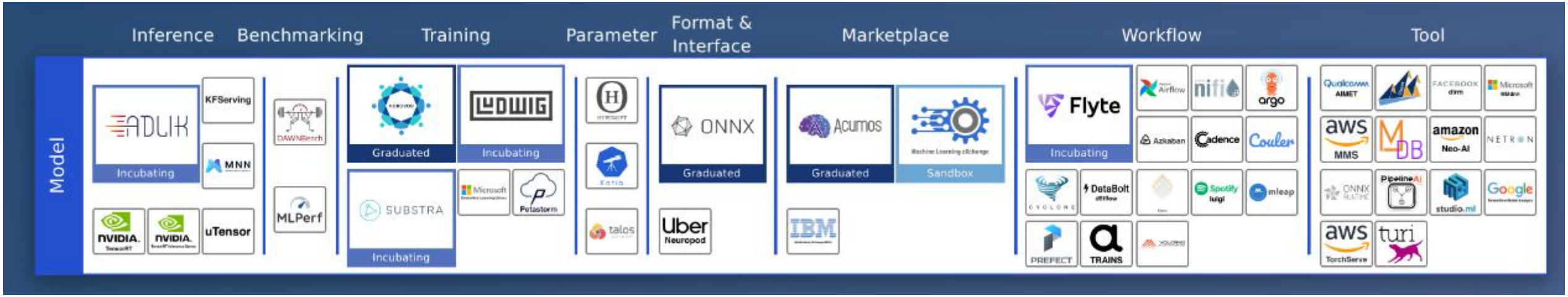


背景: Inference



Adlik项目简介

- 在Linux基金会AI和数据基金会（LF AI & Data）开源。
- 是一种可以将深度学习模型从训练完成，到部署到特定硬件并提供应用服务的端到端工具链，其应用目的是为了将模型从研发状态快速部署到生产应用环境。
- Adlik可以和多种推理引擎协作，支持多款硬件，提供统一对外推理接口，并提供多种灵活的部署方案，以及工程化的自适应参数优化方案，为用户提供快速、高性能的应用服务提供助力。



ZTE中兴

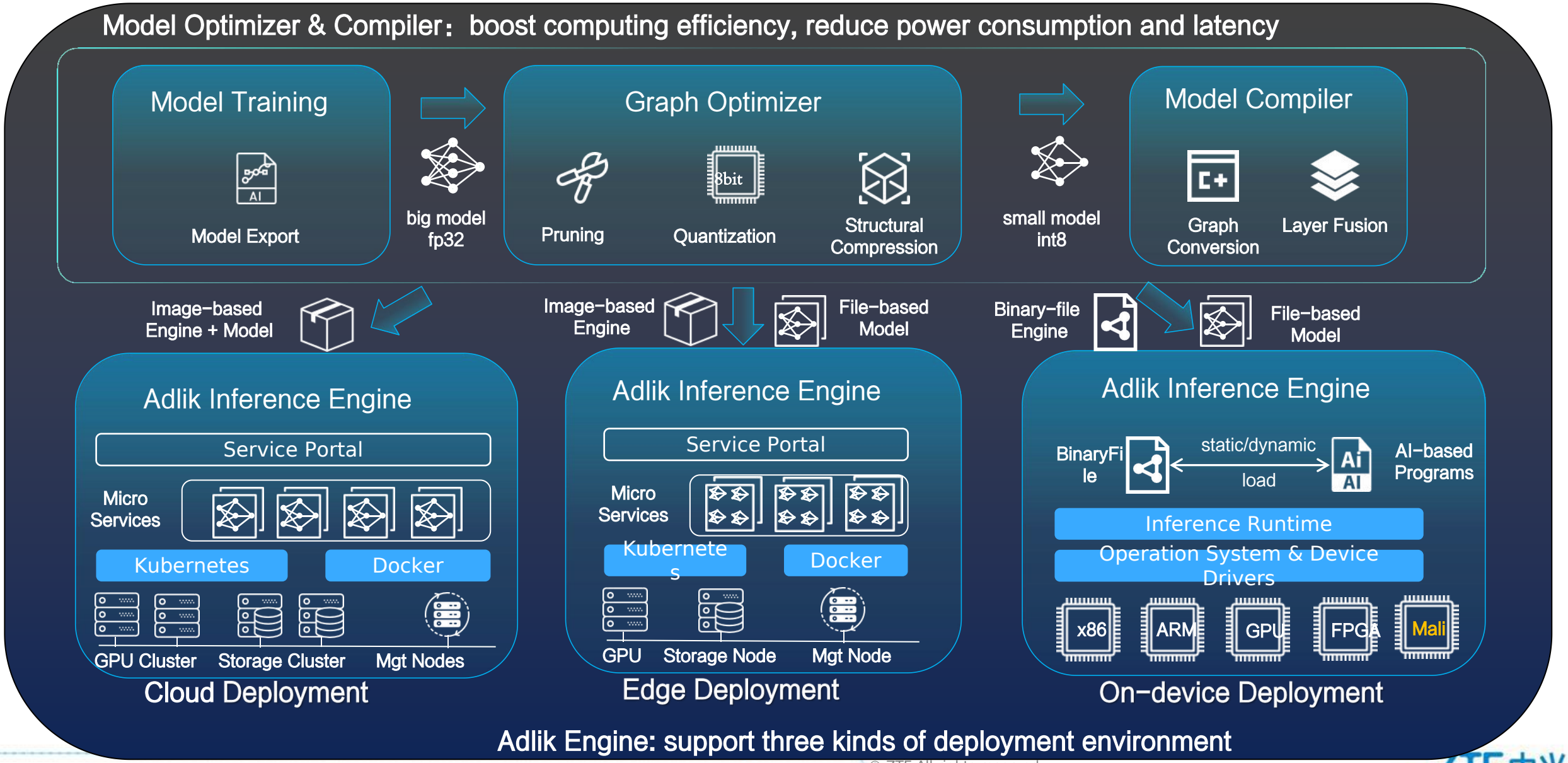
技术特点

Adlik主要解决在深度学习落地过程中的一系列问题，如：

- 针对不同设备的推理框架有很多，对用户难以选择，学习成本大
- 不用应用场景的部署条件不同，有基于容器化部署场景，也有基于嵌入式硬件部署的场景，同样的模型服务，不同部署方案要掌握不同的技术
- 根据性能需求有很多的模型调优工作
- 推理服务应用于不同硬件，需要多类异构计算引擎的支持

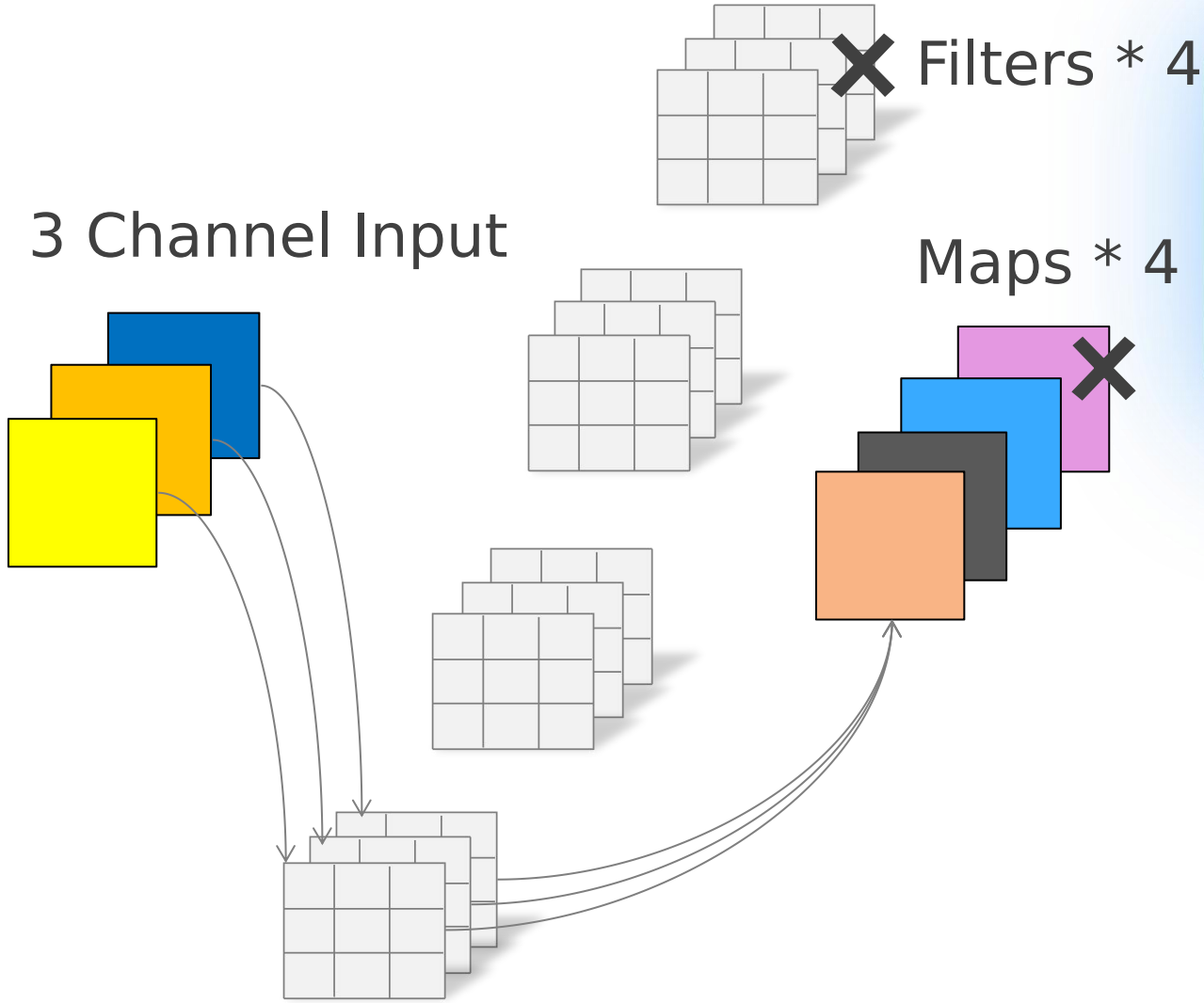
利用Adlik，开发者可以方便地通过剪枝、量化、压缩等技术来优化主流训练框架如TensorFlow、Keras、Caffe、PyTorch等训练出的模型，并针对推理侧模型部署的运行时与硬件，自动地完成最优化的模型编译工作，不仅可以提升模型的推理效率，减少时延和能耗，更具备一键式提供模型应用的部署服务的能力。

Adlik架构



6

Adlik Feature: 剪枝

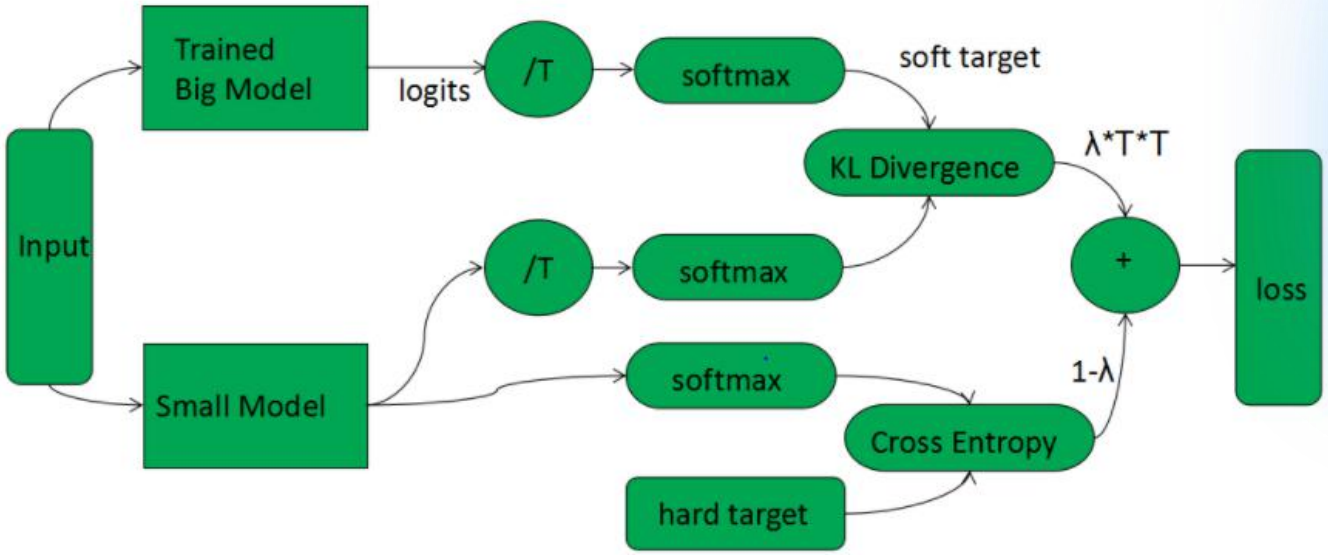


- 支持多节点多GPU剪枝和调优方案
- 支持通道剪枝，和filter剪枝，能够有效降低模型参数量和Flops。

ResNet-50	Top-1	Parameters	Size
baseline	76.19%	25.61M	99MB
pruned	75.50%	17.43M	67MB

ResNet-50	MACs	Inference speed
baseline	$5.10*10^7$	7.2 pcs/s
pruned	$3.47*10^7$	9.57 pcs/s

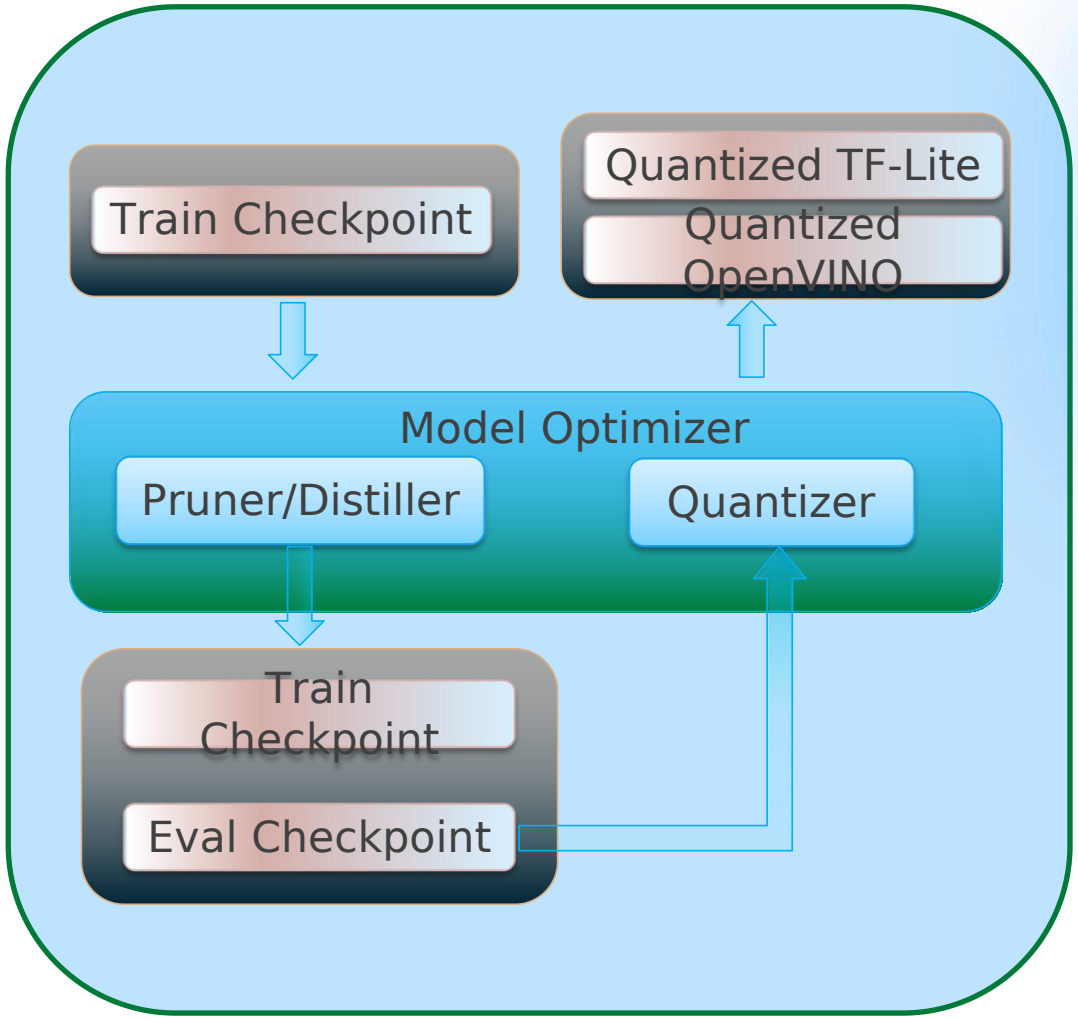
Adlik Feature: 蒸馏



- 降低模型规模，参数量大小和Flops。
- 提升模型性能

8

Adlik Feature: Model Optimizer



- 支持多教师组合蒸馏，有效提升模型性能
- 支持8bit校准量化PTQ，量化过程只需要小批量数据和很短时间就可以完成。

	Params	Flops	Accuracy	Size
ResNet-50	25610152	3899M	76.174%	99M
+ pruned(72.8%)	6954152	1075M	72.28%	27M
+ distill	6954152	1075M	76.39%	27M
+ quantize			75.938%	7.1M

Model Optimizer Result: 7.1/99 = 7.2%

Adlik Feature: Model Optimizer

Inference Benchmark Result:

- 基于 MLPerf SingleStream模式

ResNet-50	FP32	INT8	FP32_pruned	INT8_pruned
Latency(ms)	6.74	2.82	3.32	1.34

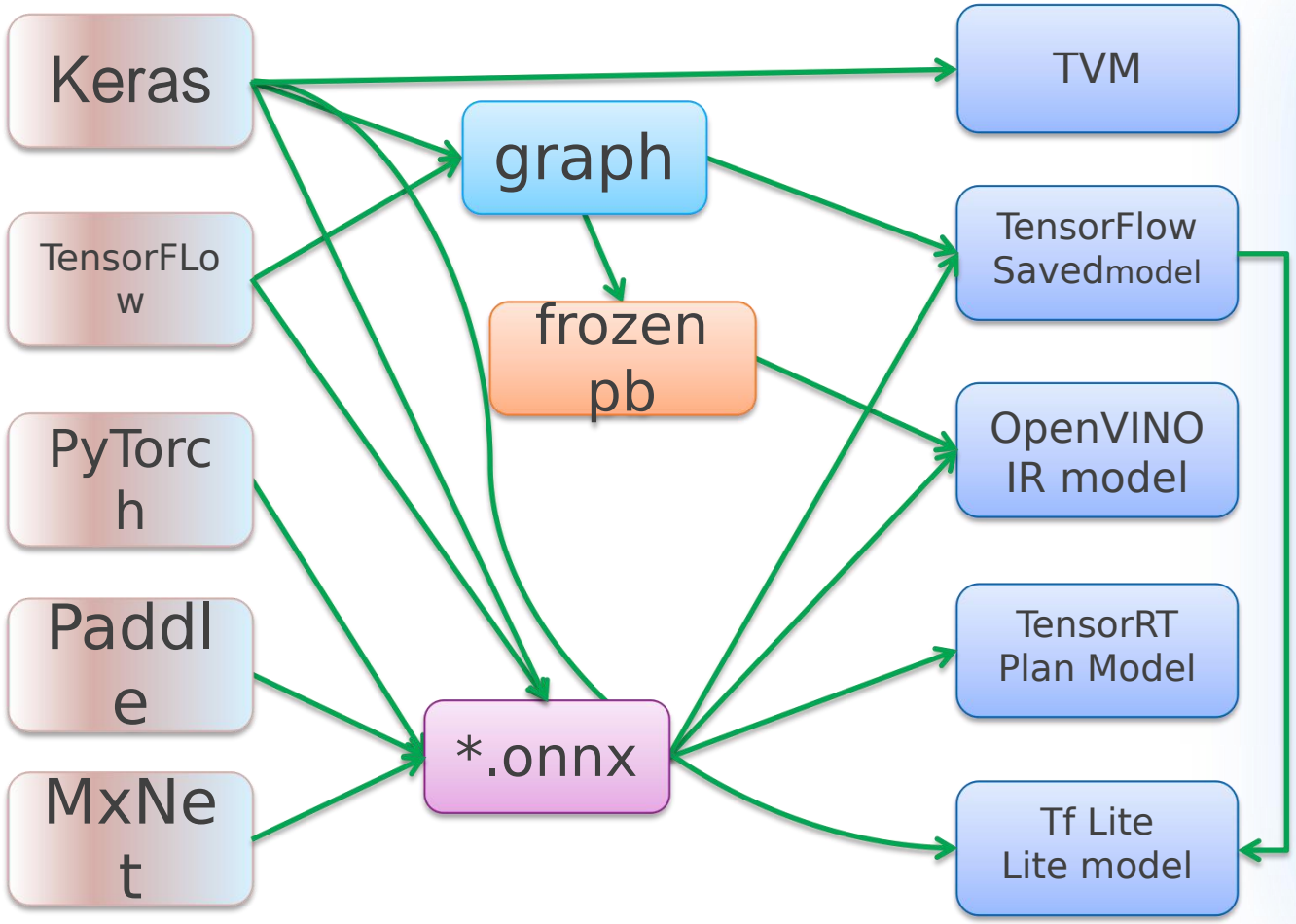
Batch size: 1, ZXCLOUD R5300 G4; Intel(R) Xeon(R) Platinum 8260 CPU @2.40GHz

- 基于OpenVINO Benchmark

	ResNet-50	FP32	INT8	FP32_pruned	INT8_pruned
Async Mode	Latency(ms)	22.56	6.35	6.63	2.09
	FPS	526.83	1863.60	1782.49	5685.45
Sync Mode	Latency(ms)	5.24	1.82	2.45	1.28
	FPS	190.73	549.93	408.03	781.56

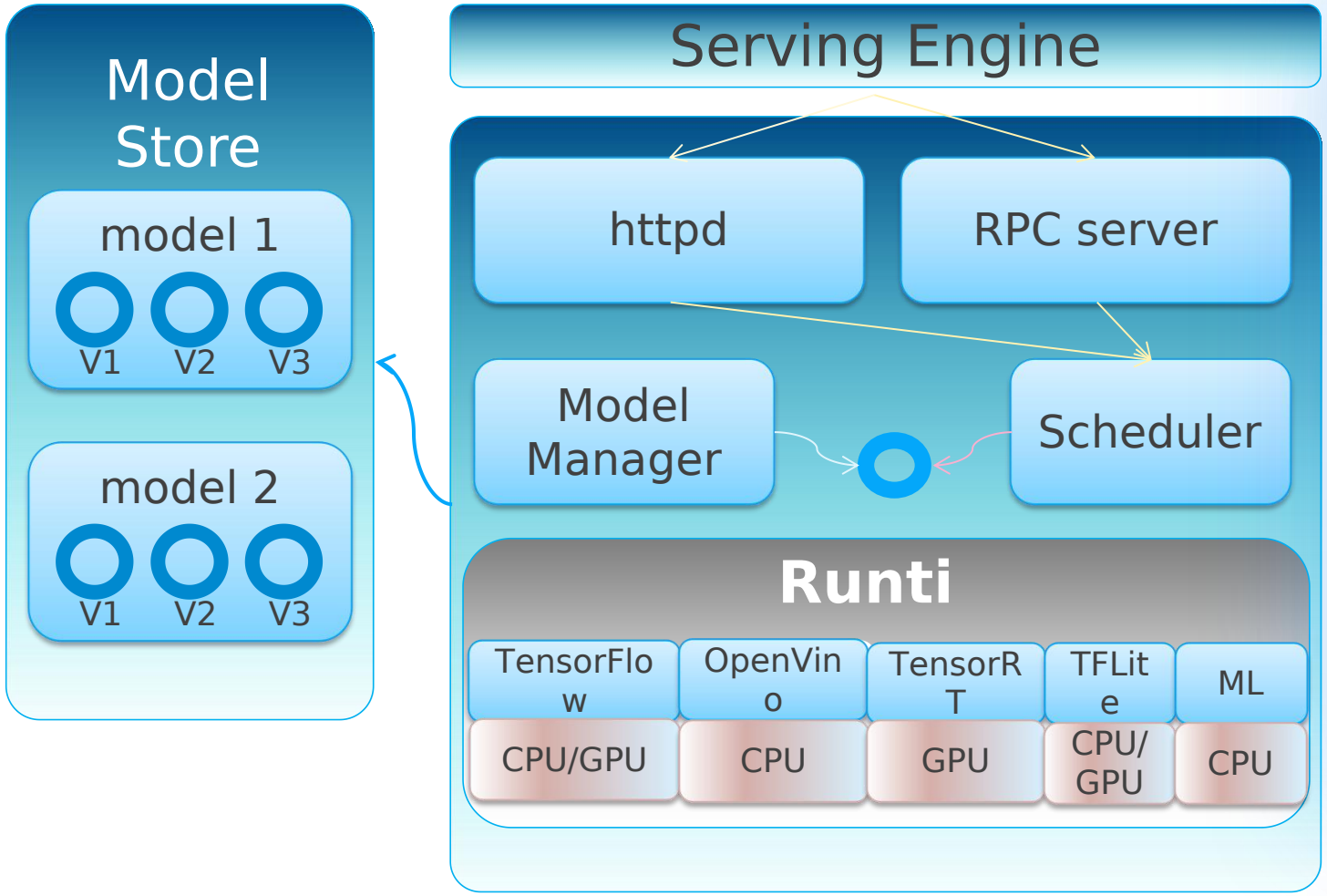
Batch size: 1, ZXCLOUD R5300 G4; Intel(R) Xeon(R) Platinum 8260 CPU @2.40GHz

Adlik Feature: Model Compiler



- 支持使用统一接口方案将多种原始训练模型格式转换到目标运行时模型格式
- 支持构建DAG 完成端到端的不同模型表达格式的转换
- 支持TfLite, TensorRT, OpenVINO模型量化

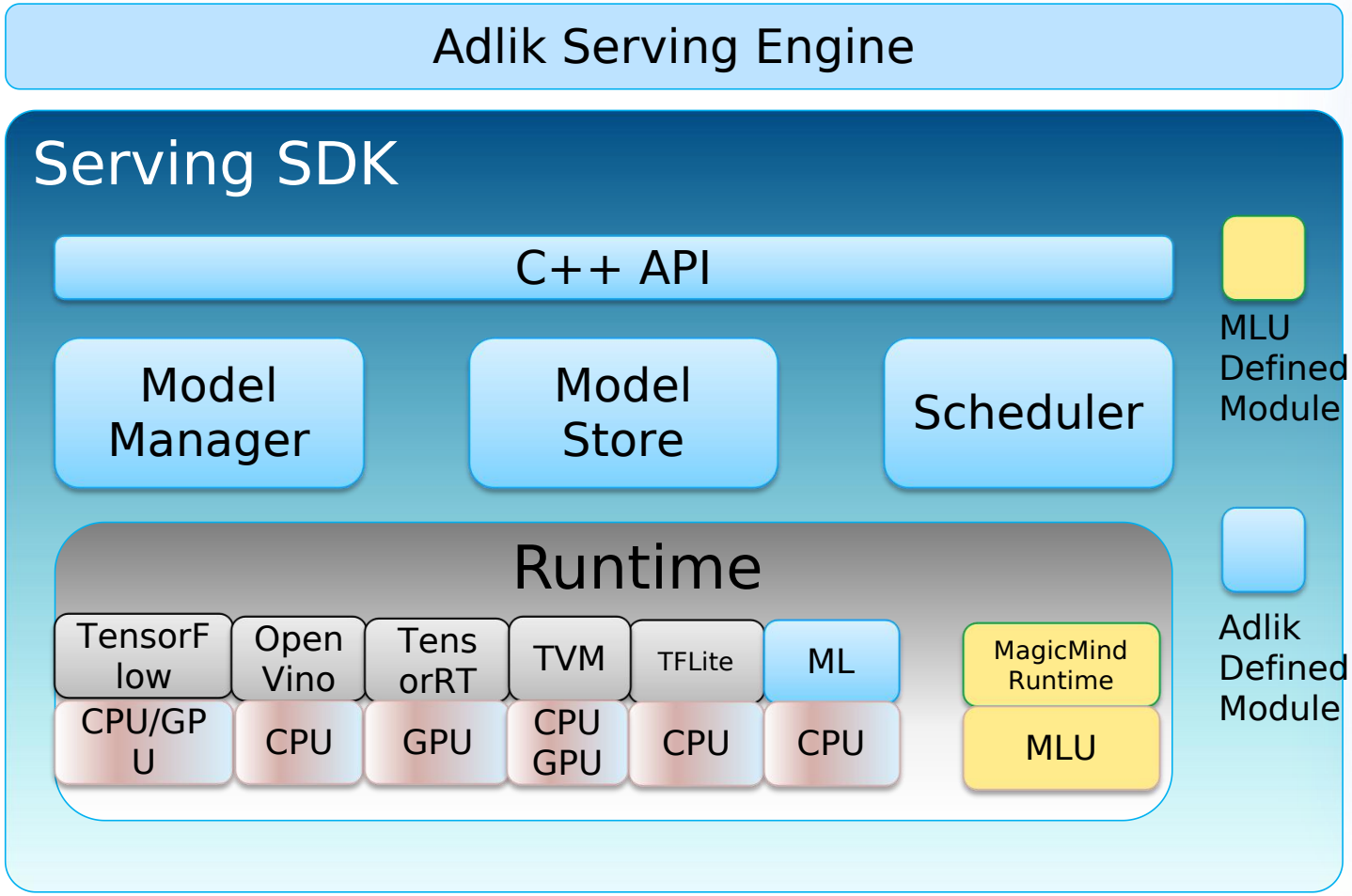
Adlik Feature: Adlik Inference Engine



- 支持模型上载，升级，版本管理，推理执行和相关监控
- 统一的推理接口
- 多运行时，多模型实例的统一管理和调度
- 支持用户自定义运行时
- 支持机器学习

12

Adlik Feature: Adlik Serving SDK



- C++ API
- 支持用户自定义运行时
- 支持用户自动Op
- 支持模型编排
- 用户易于扩展自己运行时

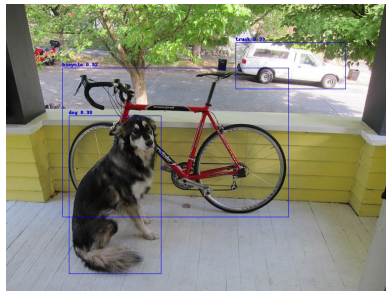
Usecase: 云原生应用

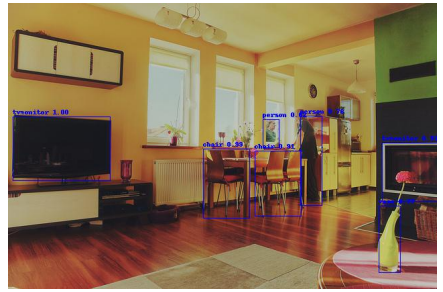
Docker Environment

```
docker run -it --rm -v /media/B/work/keras:/model 10.233.170.2:5000/adlik/model-compiler:7.0_10.0 bash
root@ecaf2fd16421:/# cd model/
root@ecaf2fd16421:/model# python3 compile_model.py
Source type: ONNXModelFile.
Target type: OpenvinoModel.
Compile_path: ONNXModelFile -> OpenvinoModel.
{'status': 'success', 'path': 'model_tf_yolov3_608_128/yolov3_1.zip'}
docker run -it --rm -v /home/t630/zkl:/model -p 31000:8500 10.233.170.2:31000/00253486/adlik_serving-openvino:latest bash
/# adlik-serving --model_base_path=/model/yolov3_repos/ --grpc_port=8500 --http_port=8501
I adlik_serving/server/core/server_core.cc:54] Adlik serving is running...
I adlik_serving/server/grpc/grpc_options.cc:88] grpc server port: 8500
I adlik_serving/server/grpc/grpc_server.cc:24] grpc server is serving...
I adlik_serving/server/http/http_options.cc:35] http server port: 8501
python3 yolov3_client.py -n yolo416 -b 1 dog.jpg
```

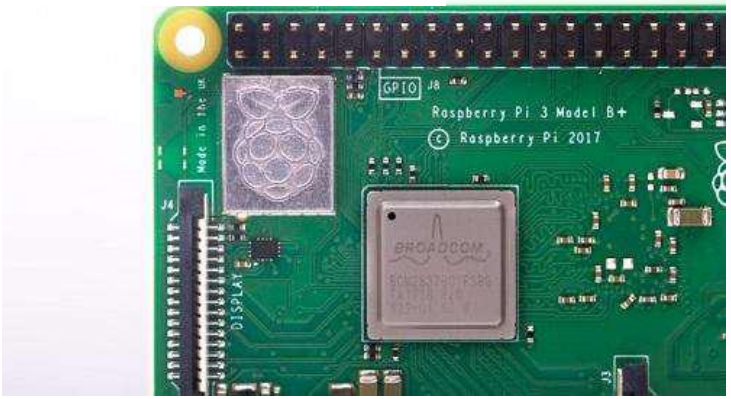
Kubernetes Environment

```
kubectl create -f compiler.yaml
pod/model-compiler created
kubectl get pod | grep compiler
model-compiler                          1/1     Running           0          24s
ls
yolov3  yolov3_1.zip
kubectl create -f openvino-serving.yaml
kubectl get pod | grep openvino-serving
openvino-serving                        1/1     Running           0          24s
kubectl create -f openvino-svc.yaml
kubectl get pod | grep openvino-serving
openvino-service          NodePort    10.254.255.197   <none>          8500:31501/TCP                79s
python3 yolov3_client.py -b 1 dog.jpg
```





Usecase: 嵌入式设备上的应用



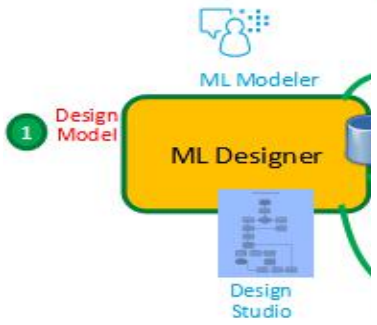
- 在Jetson Nano 和Raspberrt Pi.设备上部署Adlik
- 使用Adlik optimizer量化Resnet-50, Inception V3, 然后将其编译到TfLite运行时模型格式
- 设备侧，直接本地读取突破，启动推理进程，调研Adlik推理接口进行推理运行

Usecase: O-RAN提案

O-RAN ALLIANCE

O-RAN.WG2.AIML-v01.02

ML Model life



* ML Training host can be part of non-R
offline

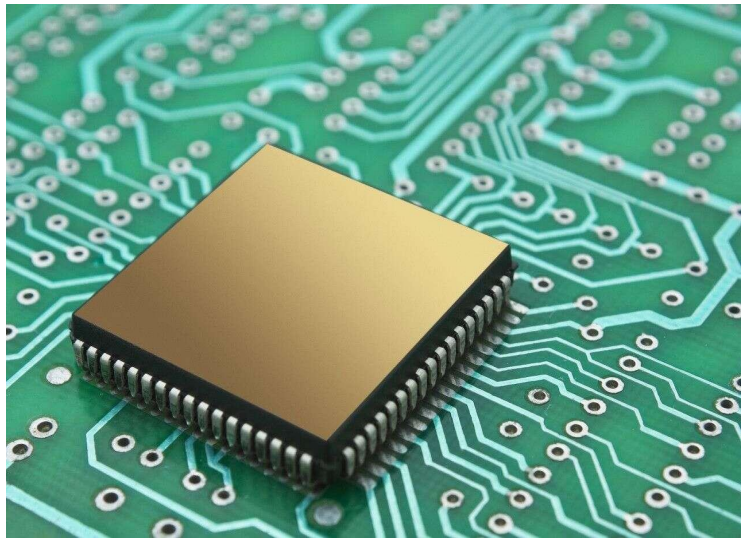
Revision History

Date	Revision	Author	Description
2020.08.31	01.02.00	R. Jana	Clean Baseline doc
2020.08.31	01.02.01	Intel, ATT, CMCC, Altran, Samsung	Adding approved CR INT.AO-2020.07.06-WG2-CR-0001-AIML model termination procedure-v03.docx
2020.10.09	01.02.02	IBM, ZTE, CMCC	IBM.AO-2020.06.05-WG2-CR-0001-AIML-v05.docx
2020.11.29	01.02.03	Intel, Samsung, Amdocs	INT.AO-2020.10.19-WG2-CR-0006-Reinforcement Learning-v02.docx
2020.11.29	01.02.03	Intel, Samsung, Amdocs	INT.AO-2020.10.19-WG2-CR-0007-DS for RL-v04.docx
2021.01.16	01.02.04	IBM	IBM-2020.06.05-WG2-CR-0002-AIML-v11.docx
2021.01.16	01.02.04	IBM	IBM-2020.06.05-WG2-CR-0003-AIML-v08.docx
2021.01.16	01.02.04	NOK	NOK-2020.11.26-WG2-CR-0001-ModelLifecycle-v03.docx
2021.02.23	01.02.04	ZTE, CMCC	ZTE.AO-2020.06.03-WG2-CR-001-AIML-v8.doc
2021.03.11	01.02		Editorial updates for publication

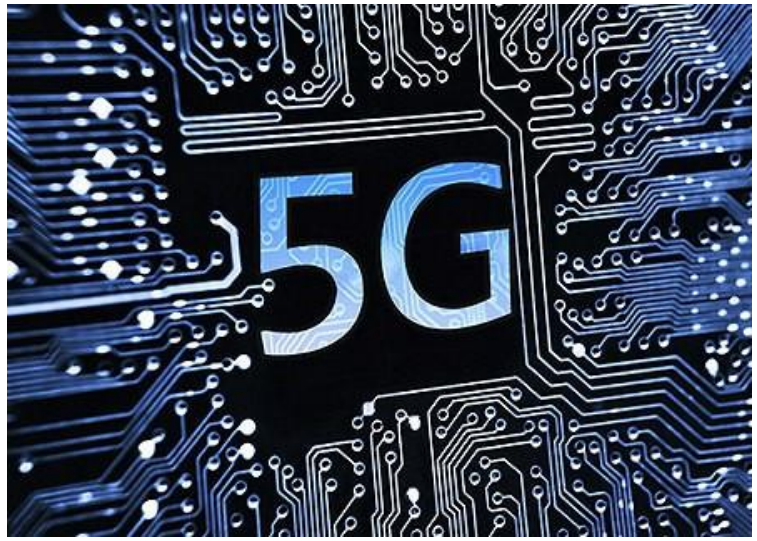
ZTE中兴

AI推理方向未来的挑战

更快的推理速度



更轻量级部署



更快+更轻

Adlik Practice: 模型图优化

BN Fold

before

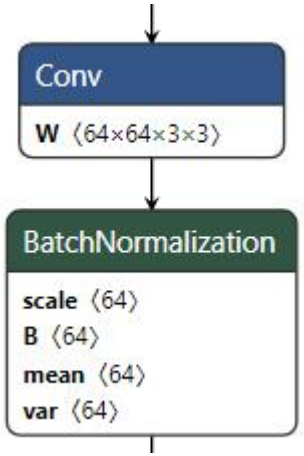
$$z = W * x + b$$

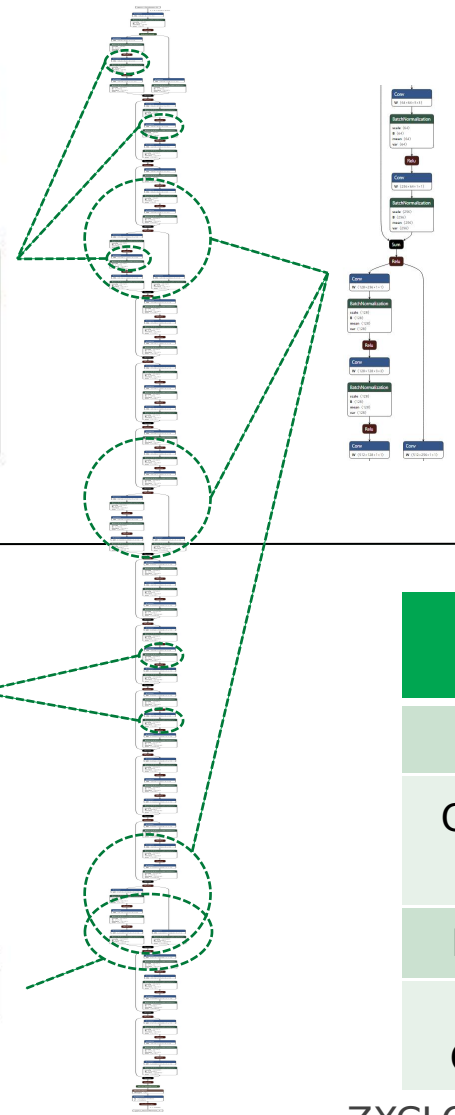
$$\text{out} = \gamma \cdot \frac{z - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta$$

after

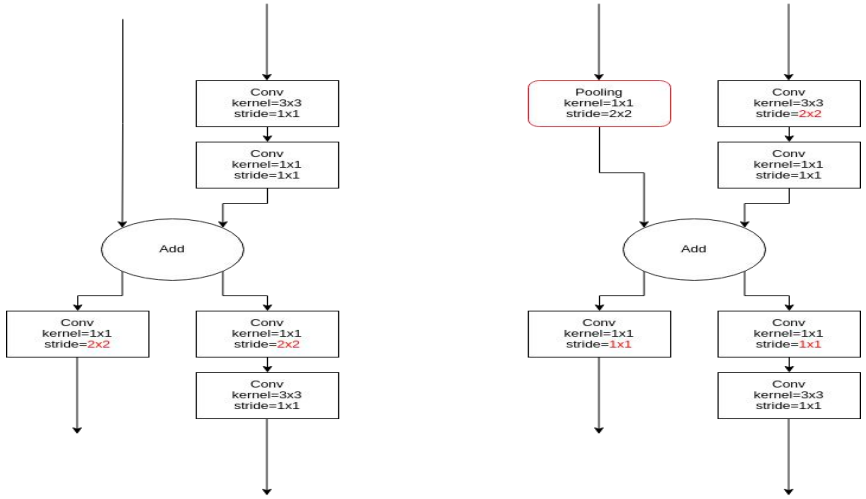
$$w_{\text{fold}} = \gamma \cdot \frac{W}{\sqrt{\sigma^2 + \epsilon}}$$

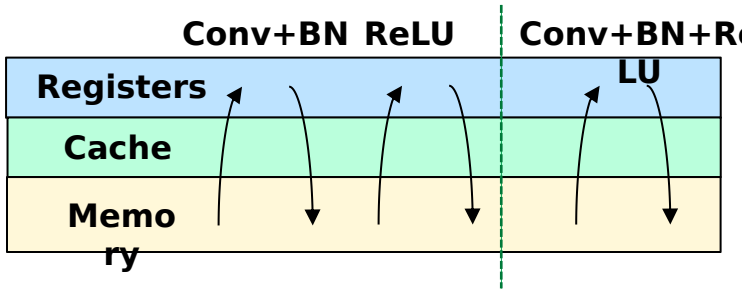
$$b_{\text{fold}} = \gamma \cdot \frac{b - \mu}{\sqrt{\sigma^2 + \epsilon}} + \beta$$





Stride Optimization (Resnet-specific)





Layer Fusion

	Inference Latency(ms)	Improvment
Benchmark	12.09	-
Constant Fold (Conv+BN)	9.87	18.39%
Layer Fusion	7.81	20.85%
Stride Optimization	6.7	14.24%

ZXCLOUD R5300 G4; Intel(R) Xeon(R) Platinum 8260 CPU @2.40GHz

Adlik Practice: OP调度优化

Step1: Schedule parameter optimization for single op

Step 2：Schedule parameter optimization in graph view

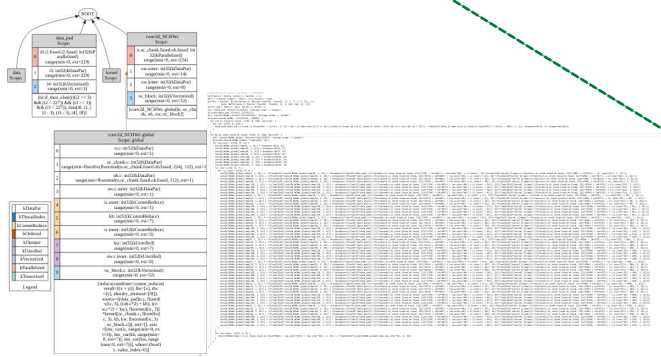
Schedule Parameters Optimizatio n Algorithm

Schedule Template

Optimized schedule parameter

Schedule Parameter s Search

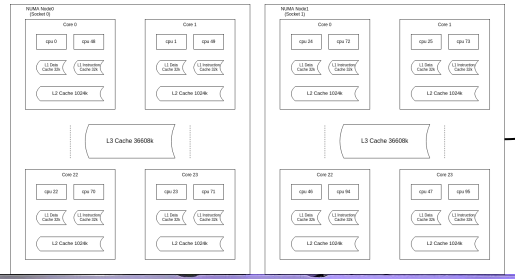
- Generate schedule based on schedule parameters and template
- Execute the program and collect performance information using VTune (Hostpot,/Memory Consumption/Memory Access)



Operator Schedule Template

Schedule Template Design

Schedule Performan ce Analysis



CPU Micro Architecture

1×3×224×224

Conv

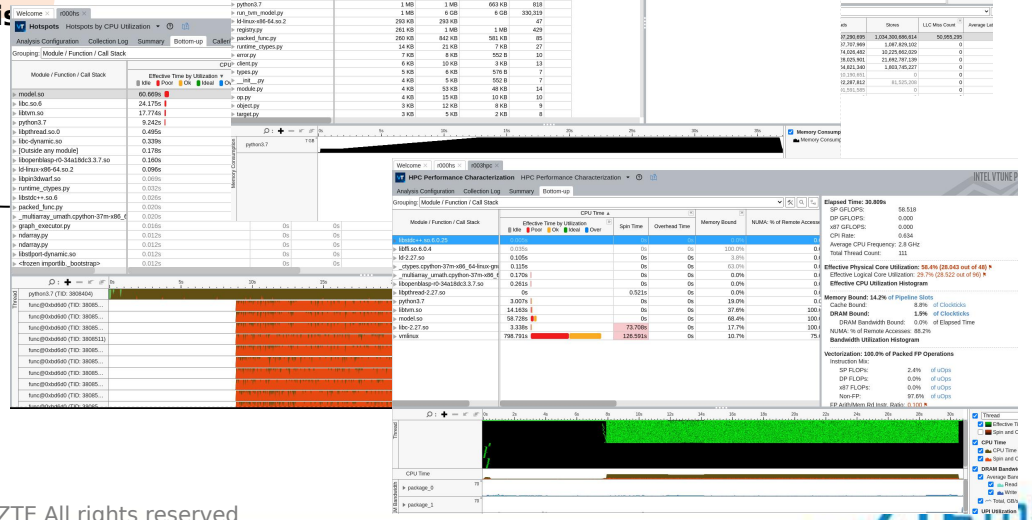
W ⟨64×3×7×7⟩

kernel_shape = 7, 7

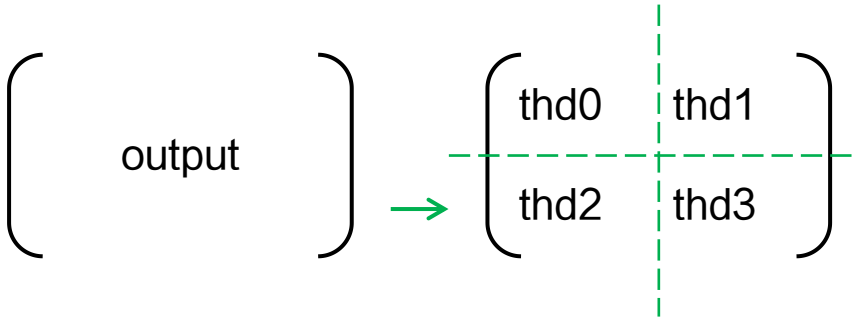
pads = 3, 3, 3, 3

strides = 2, 2

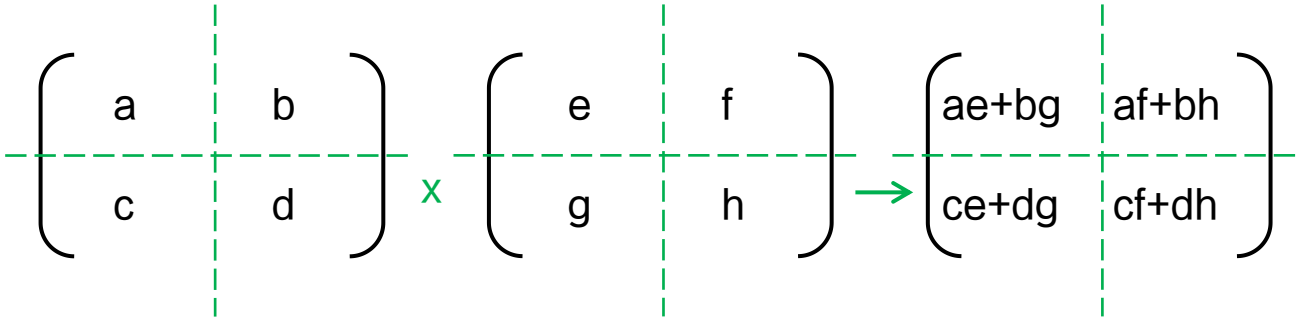
operator workload



Adlik Practice: Dense OP设计

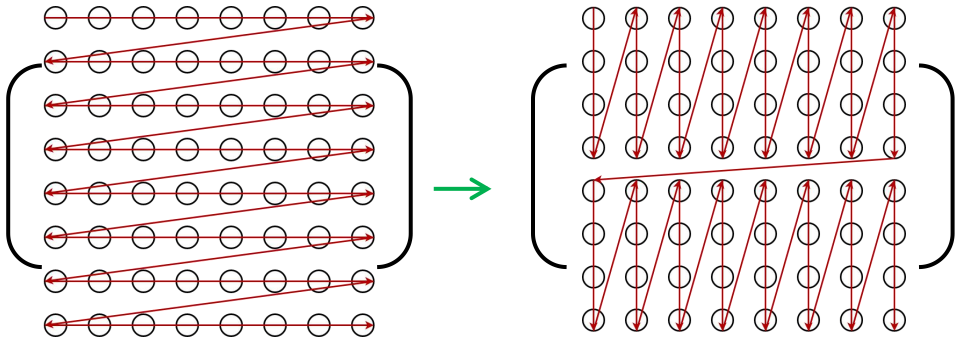


Parallelization



Blocking

Layout Reorder



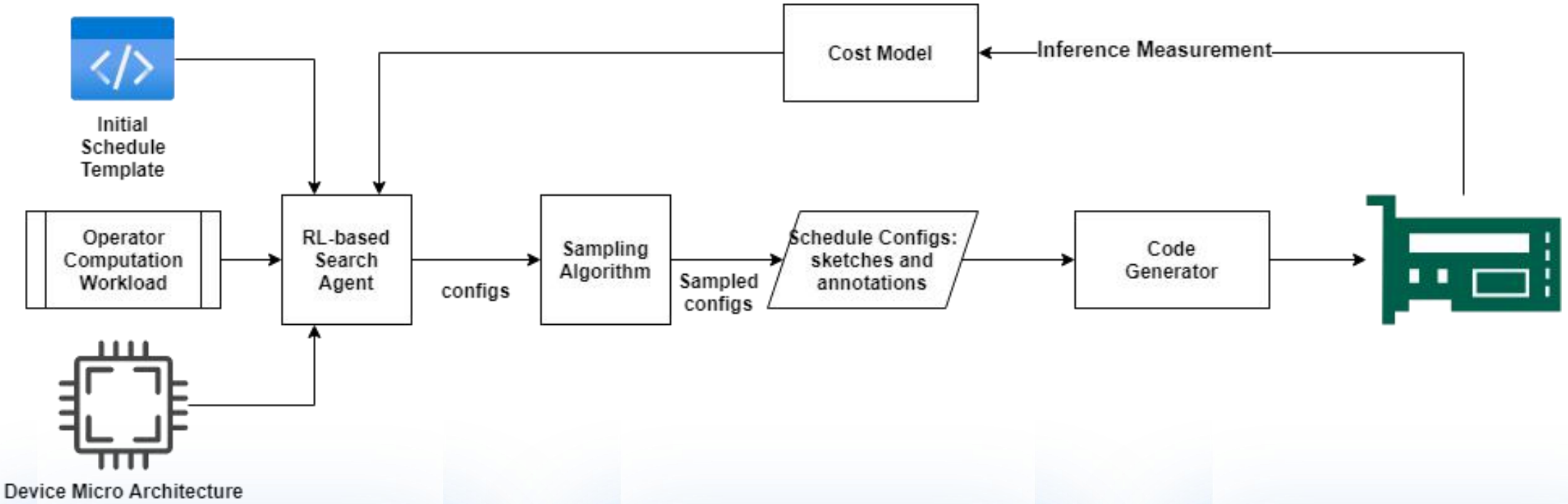
SIMD（Computing block）

m：4X/8Y/16Z的倍数；n：由X/Y/Zmm个数和实际算法设计决定。

Benchmark test by google/benchmark

Thread Number	1	2	4	8
Improvement（vs oneDNN）	6.5%	5%	7%	5%

Adlik Practice: 基于RL的调度自动搜索

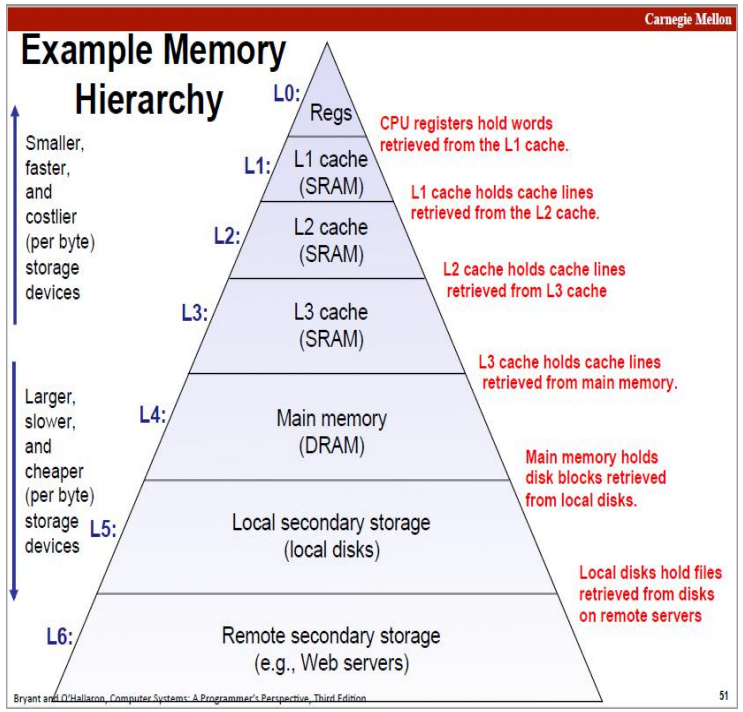


基于 Ansor
(a.k.a TVM auto scheduler)

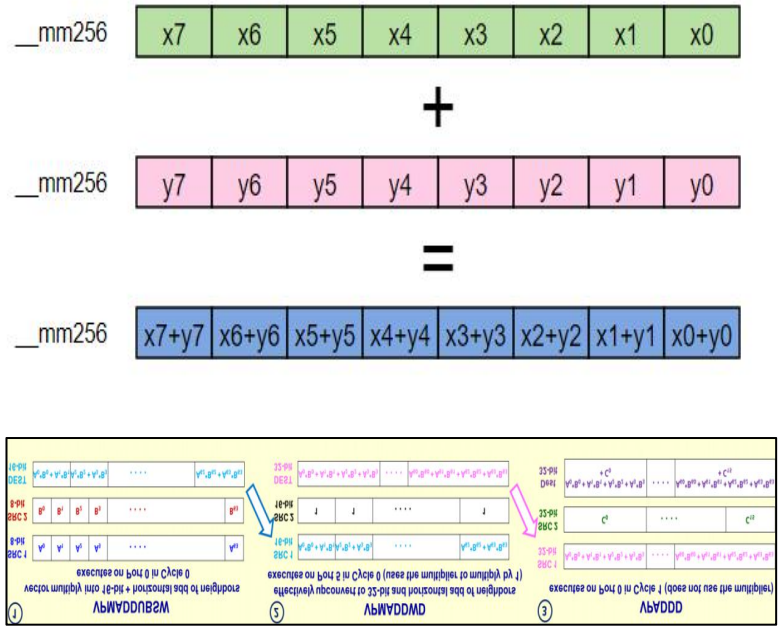
Agent 产生新配置
1. 设备微架构
2. 初始或上一次的配置
3. 算子 workload

新建cost model来加调度配置的评估

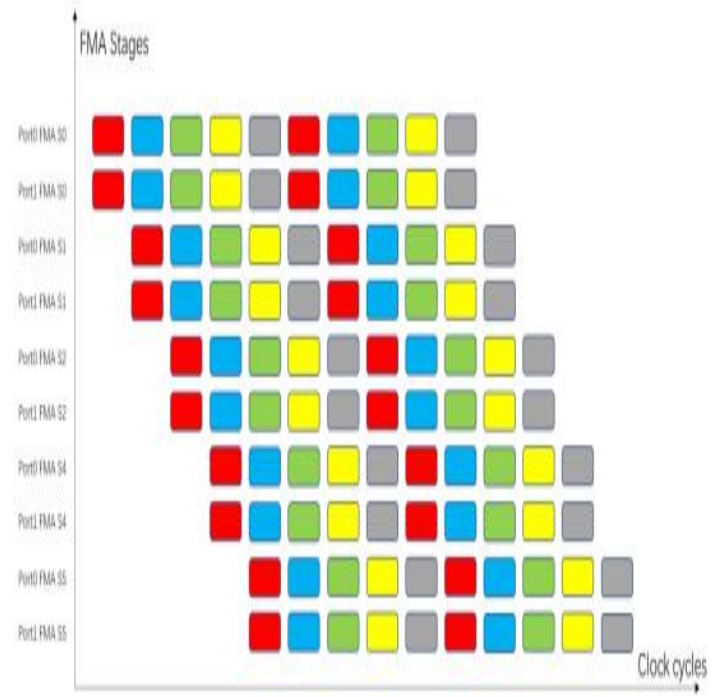
Adlik Practice: CPU的高性能计算优化



基于硬件存储结构分析，进行算法设计，充分利用缓存，提高缓存命中率

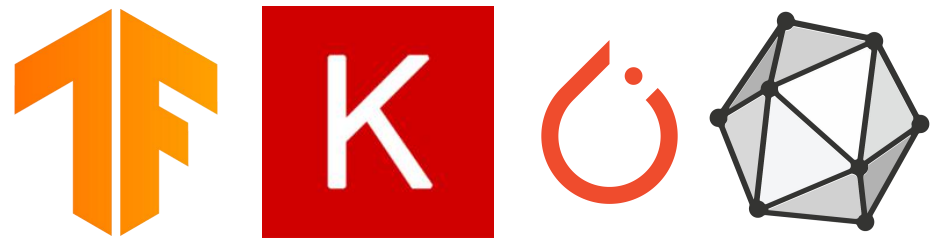


基于硬件向量化指令集进行高性能计算，使用Hardware-intrinsic编码完成微算子实现



CPU指令执行时延分析，设计高效计算流水线， 充分利用FMA计算能力

Adlik Practice: DL编译器



Compiler Frontend

Graph

Graph Optimizer

Optimized IR

CodeGen

Runtime IR

Device

High Level Code

- 模型代码生成
- 线程调度
- 线程管理
- 数据派发
- 内存分配

x.cc

x.h

Low Level Code

- 汇编级OP实现
- SIMD指令集相关实现
- 设备指令集相关

x.asm

Model

项目活跃度

项目代码

- 持续近两年稳定的代码贡献；
- 来自中兴、百度、英特尔、中国联通，Oneflow等公司的20个贡献者；
- 每半年发布一次新版本，已发布四个版本，每次版本更新都有亮眼新特性合入。

技术合作

- 积极参与产学研合作，如与北大、中科院计算所等学校的科研团队及百度、ARM、英特尔等公司的研发团队等保持密切合作；
- 积极推广项目，目前已有电信、联通、移动、宁德时代等多家企业用户；
- 在ITU AI in 5G国际挑战赛中，作为一个赛点，发布了深度学习模型推理优化的赛题，为Adlik项目的技术规划集思广益。

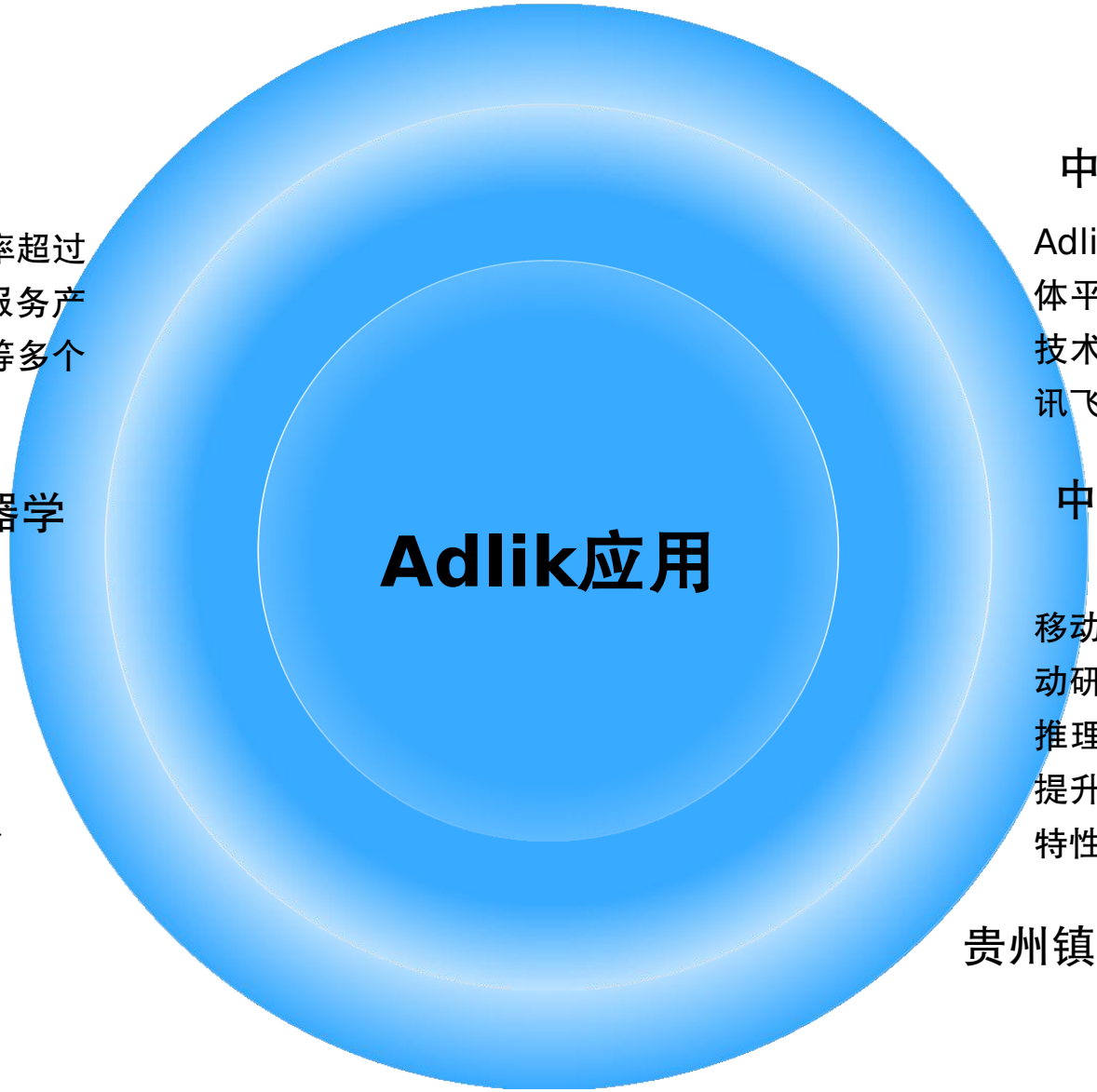
社区运营

- 每两周一次社区会议；
- Adlik项目的公众号（Adlik_AI）、视频号（Adlik）、知乎专栏、CSDN等自媒体平台都更新活跃。

技术布道

- 在多个峰会、论坛上做技术分享，如COPU开源中国开源世界峰会、GOTC全球开源技术峰会、NGMN下一代移动网络峰会、CSDN ProCon、百度Wave summit峰会等；
- 21年10月，Adlik联手LF AI & Data以及国际电信联盟（ITU）的AI for Good峰会，发起AI open day活动。

应用成熟度



中兴通讯AI平台ZAIP

服务中兴内部的AI平台，GPU利用率超过90%，提供模型模板30个以上，服务产品数十个，部署包括中兴翻译平台等多个应用。

中国电信集团AIoT机器学习服务平台项目

完成一期建设，并进入二期需求讨论中

中国铁塔AI训练平台项目

中国电信AI平台项目

Adlik作为平台推理端组件，参加整体平台的技术招标测试，最终平台技术测试综合排名第一，超过科大讯飞，浩鲸等多家业内知名公司

中国移动合作加密流量识别项目

移动同事基于Adlik发布镜像，对移动研究院的加密流量识别模型进行推理部署，取得了3倍以上的性能提升，并准备进一步应用模型优化特性

贵州镇远智慧城市VAP项目

自媒体运营

- 项目Wiki：
  https://wiki.lfaidata.foundation/display/ADLIK

- 微信公众号：Adlik_AI
- 微信视频号：Adlik

- 知乎专栏：
https://www.zhihu.com/column/adlik

- CSDN社区号：
https://blog.csdn.net/lf_ai

收益：参加的开源活动

WAVE SUMMIT 2021深度学习开发者峰会
圆桌对话：如何走顺深度学习的最后一公里

2021WAIC 世界人工智能大会
《Adlik对深度学习模型推理优化的实践》

中国人工智能产业发展联盟 2021年第一次全体大会主题发言
《Adlik深度学习模型推理加速实践》

GOTC2021全球开源技术峰会深圳站
《深度学习助力5G通讯应用示例》

ITU AI for Good峰会“开源，加速人工智能创新”论坛
《Adlik，让人工智能更触手可及》

Kubernetes on AI & Edge Days
主题发言《基于Kubernetes的AI平台资源高效调度方案》

启智开发者大会
《Adlik，让人工智能更触手可及》

THANKS

谢谢