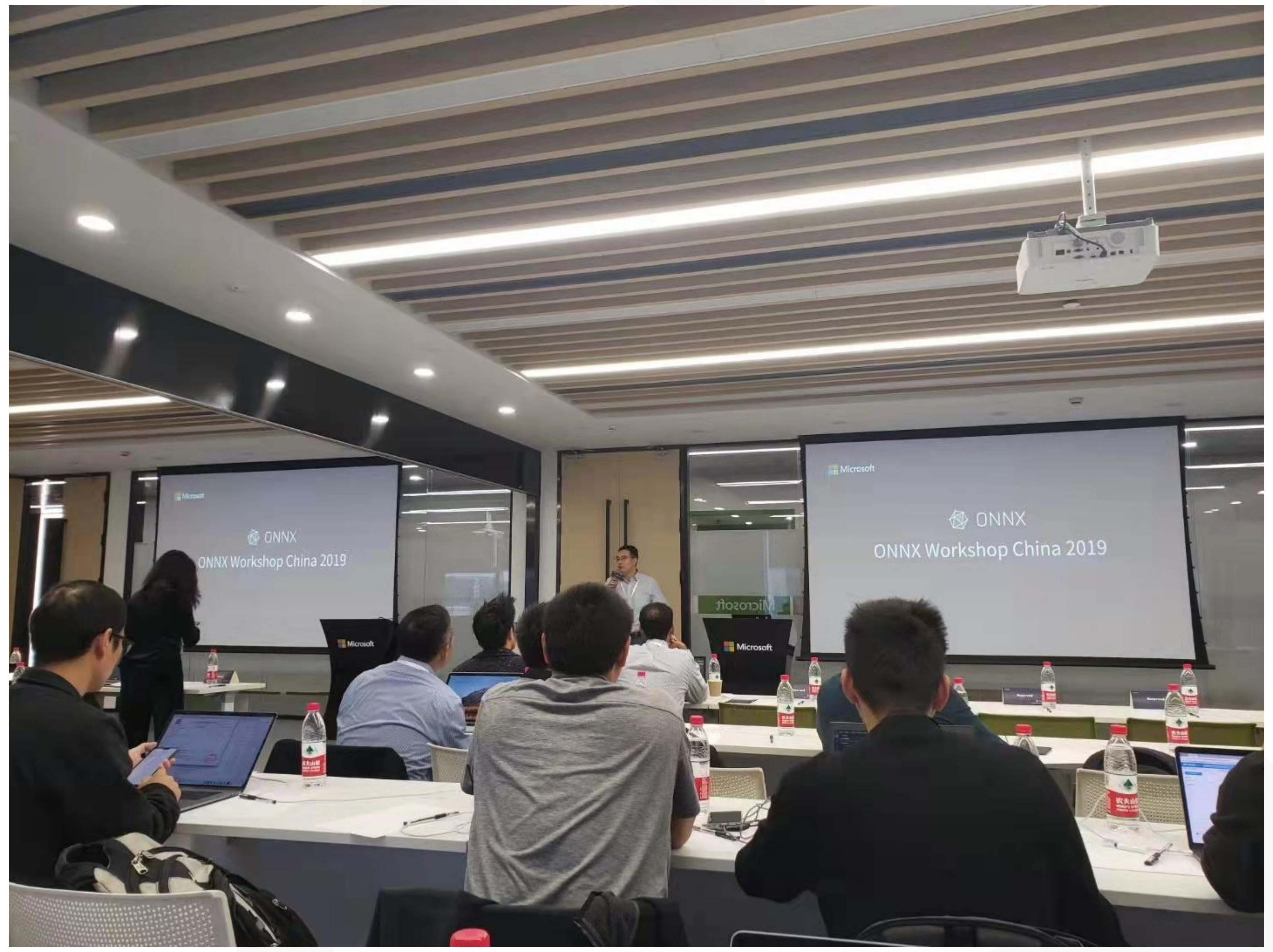
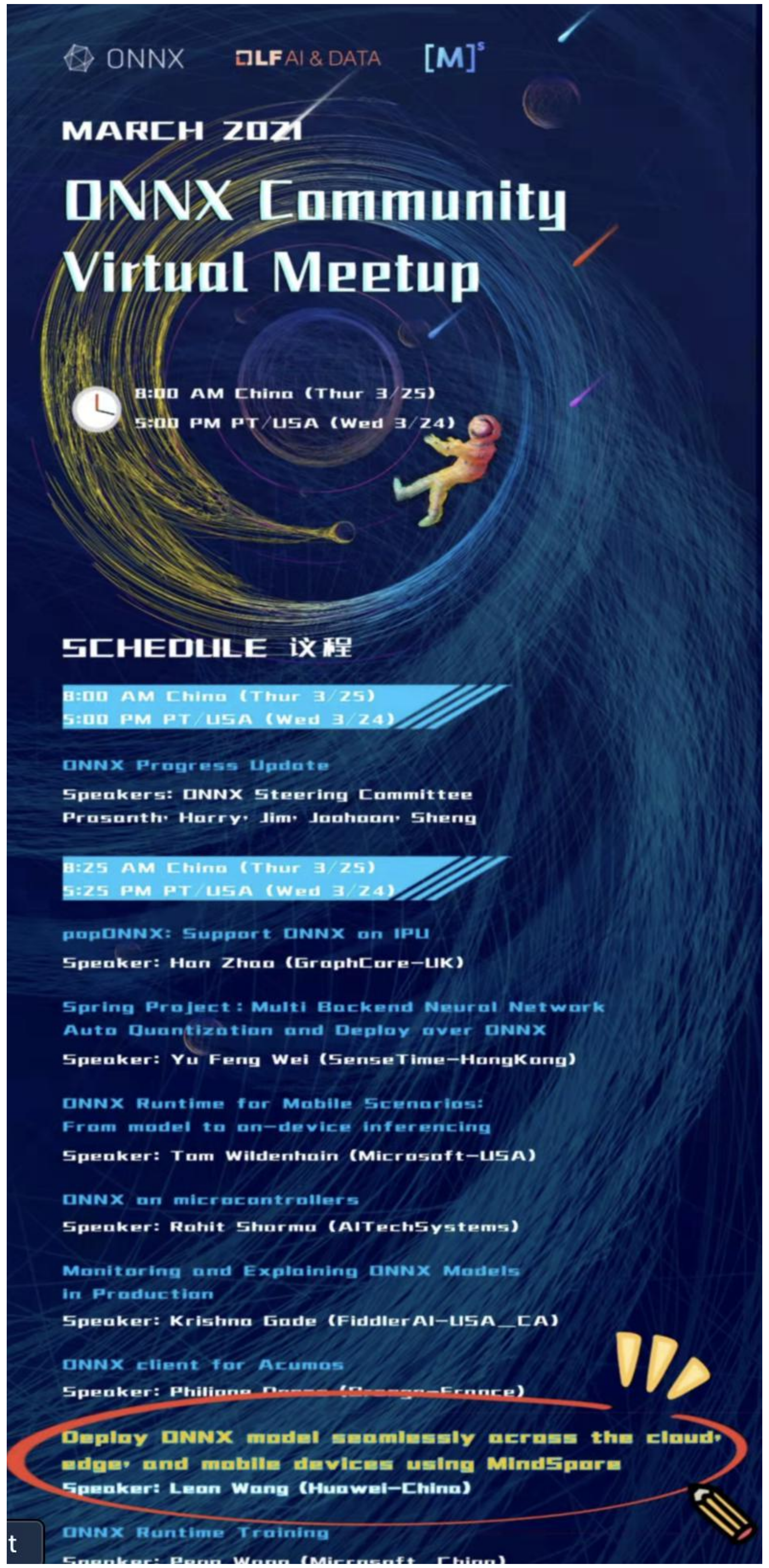
Ascend CANN and ONNX : inference interoperability for better performance

Memory Lane - Huawei's Participation In ONNX





ONNX
LF AI & DATA
[M]^S

MARCH 2021

ONNX Community Virtual Meetup

8:00 AM China (Thur 3/25)
5:00 PM PT/USA (Wed 3/24)

SCHEDULE 议程

8:00 AM China (Thur 3/25)
5:00 PM PT/USA (Wed 3/24)

ONNX Progress Update
Speakers: ONNX Steering Committee
Prasanth, Harry, Jim, Jaahoon, Sheng

8:25 AM China (Thur 3/25)
5:25 PM PT/USA (Wed 3/24)

popONNX: Support ONNX on IPU
Speaker: Han Zhao (GraphCore-UK)

Spring Project: Multi Backend Neural Network Auto Quantization and Deploy over ONNX
Speaker: Yu Feng Wei (SenseTime-HongKong)

ONNX Runtime for Mobile Scenarios: From model to on-device inferencing
Speaker: Tom Wildenhain (Microsoft-USA)

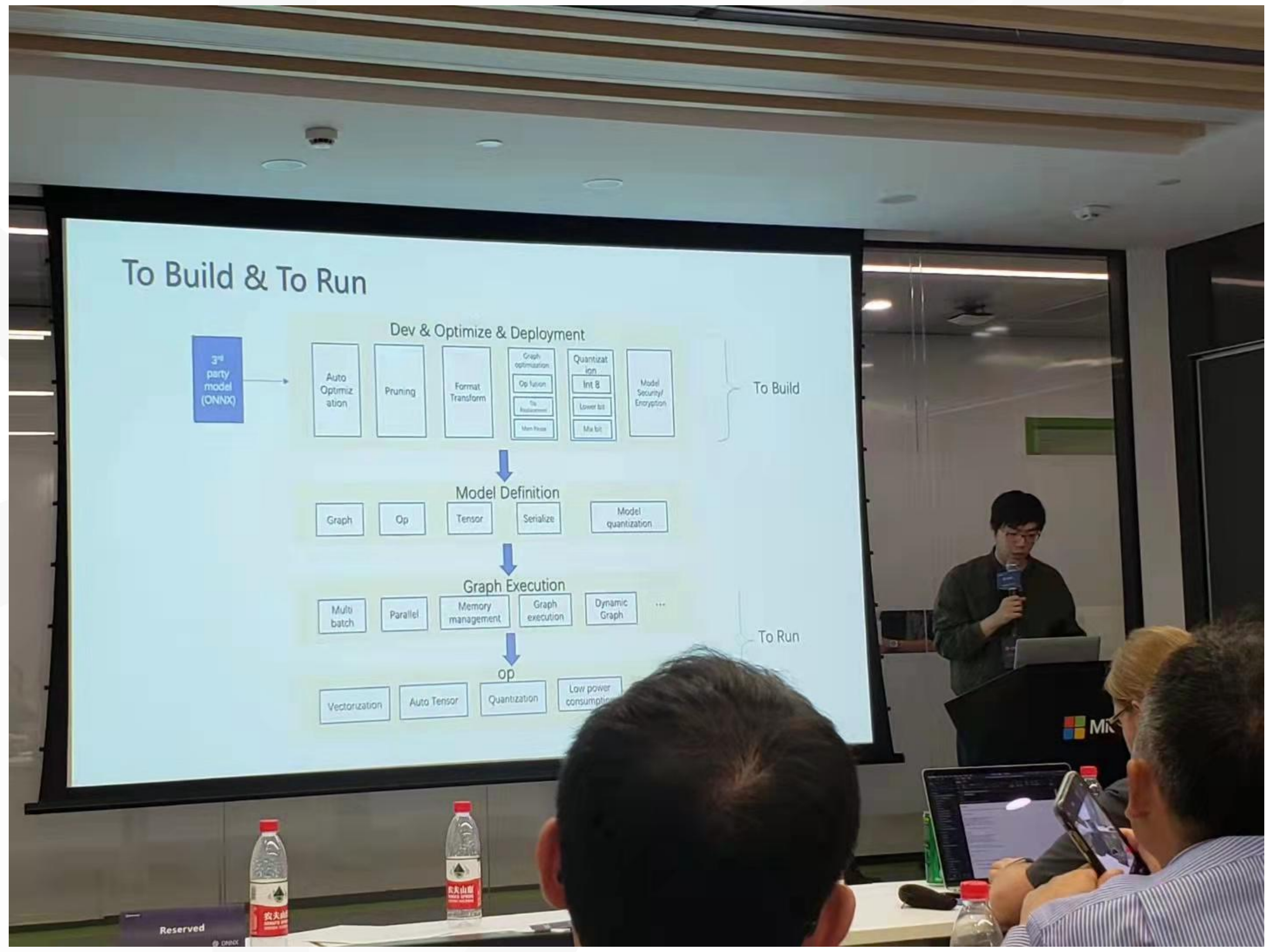
ONNX on microcontrollers
Speaker: Rohit Sharma (AITechSystems)

Monitoring and Explaining ONNX Models in Production
Speaker: Krishna Gade (FiddlerAI-USA_CA)

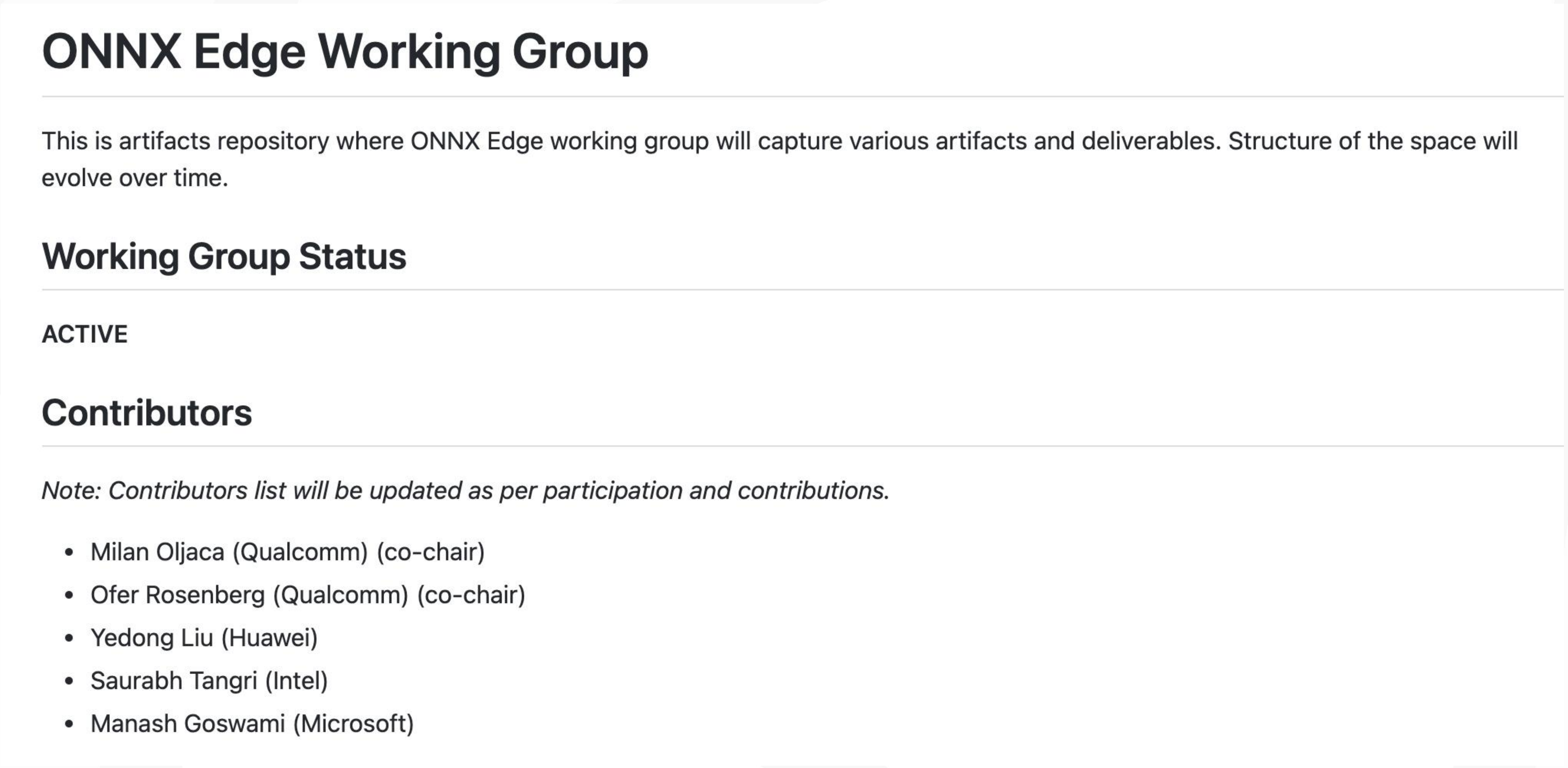
ONNX client for Acumos
Speaker: Philippe Dooze (Orange-France)

Deploy ONNX model seamlessly across the cloud, edge, and mobile devices using MindSpore
Speaker: Leon Wang (Huawei-China)

ONNX Runtime Training
Speaker: Peng Wang (Microsoft_China)



Memory Lane - Huawei's Participation In ONNX



ONNX Edge Working Group

This is artifacts repository where ONNX Edge working group will capture various artifacts and deliverables. Structure of the space will evolve over time.

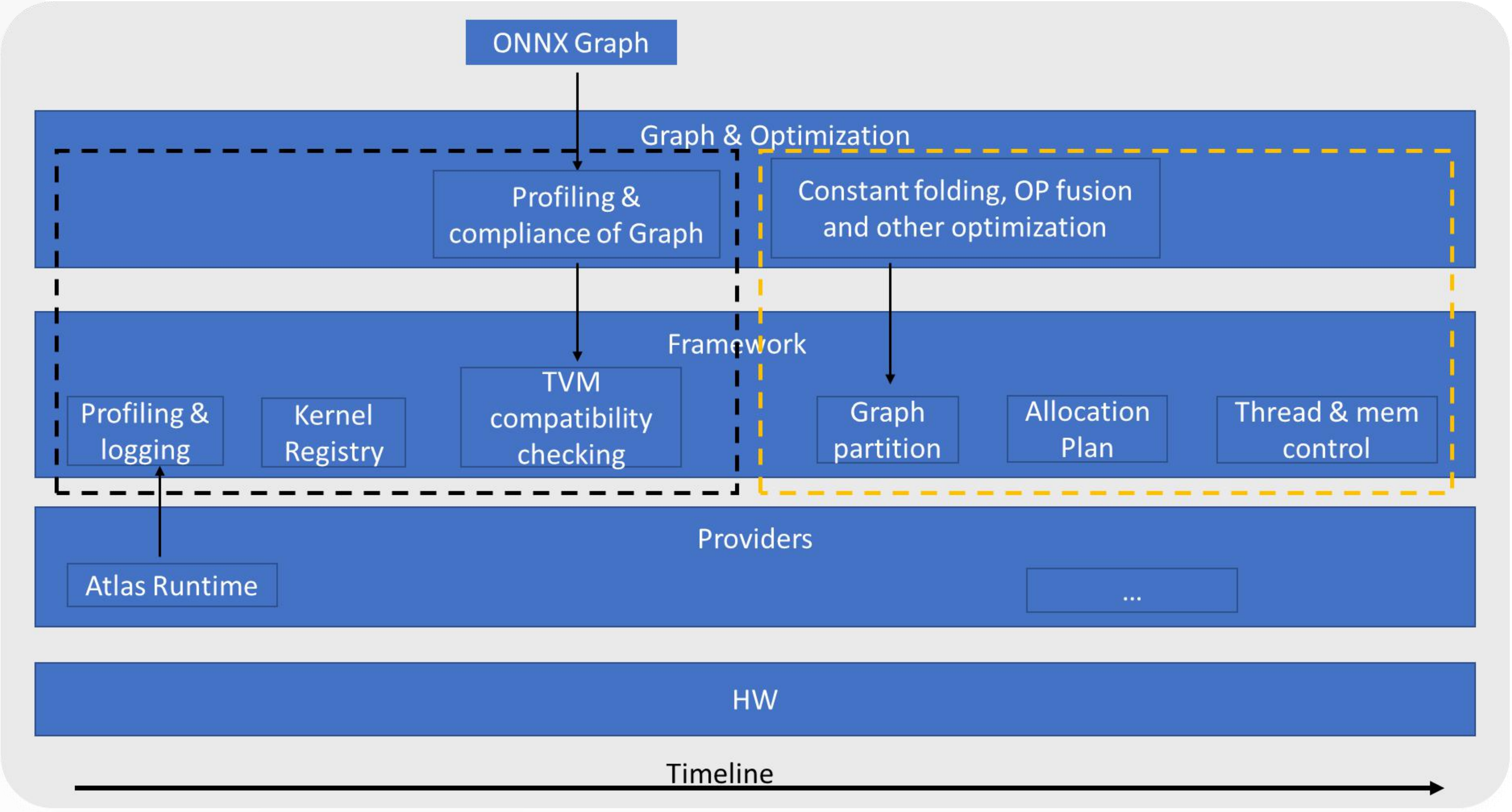
Working Group Status

ACTIVE

Contributors

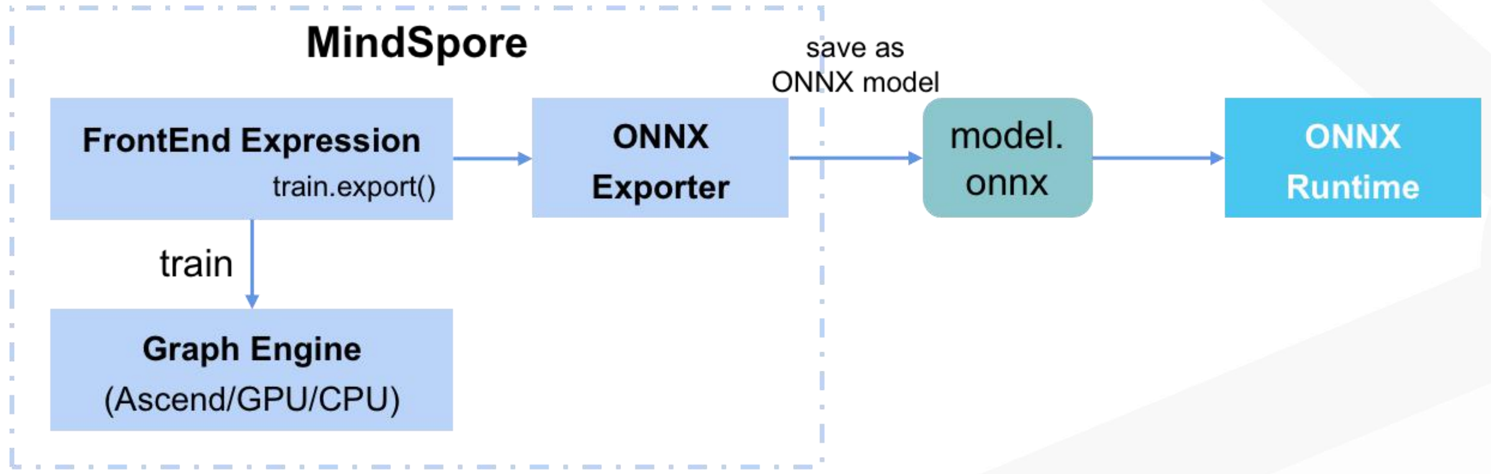
Note: Contributors list will be updated as per participation and contributions.

- Milan Oljaca (Qualcomm) (co-chair)
- Ofer Rosenberg (Qualcomm) (co-chair)
- Yedong Liu (Huawei)
- Saurabh Tangri (Intel)
- Manash Goswami (Microsoft)

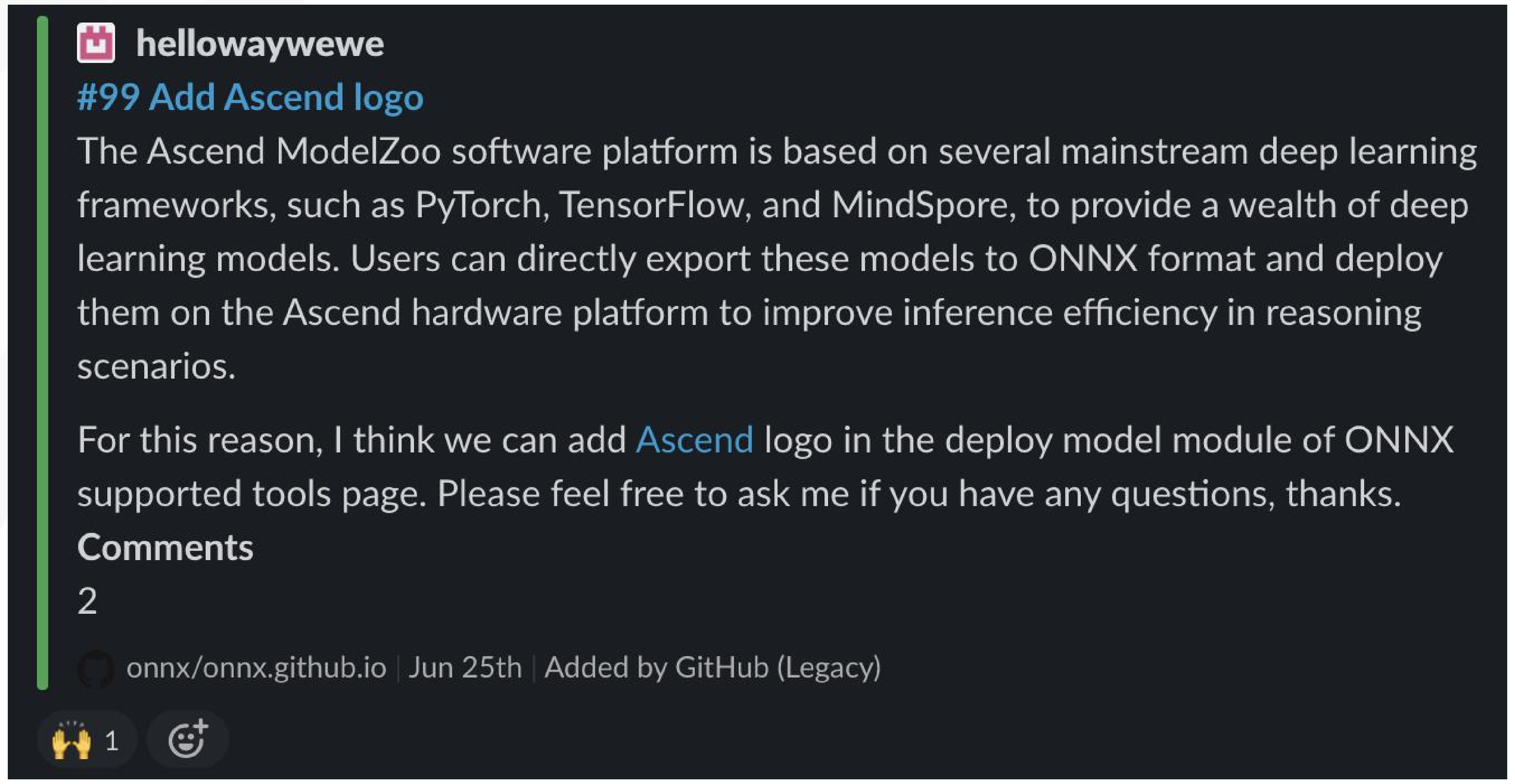


The black box is the “profiling phase” and the orange box is the “execution phase”

MindSpore ONNX Exporter Introduction



1. Use MindSpore model train API to perform model training with saving checkpoint parameters
2. Load model parameters into the network to be exported (such like LeNet)
3. Call *train.export()* to convert MindSpore model to ONNX model
4. Perform model inference on ONNX Runtime



hellowaywewe

#99 Add Ascend logo

The Ascend ModelZoo software platform is based on several mainstream deep learning frameworks, such as PyTorch, TensorFlow, and MindSpore, to provide a wealth of deep learning models. Users can directly export these models to ONNX format and deploy them on the Ascend hardware platform to improve inference efficiency in reasoning scenarios.

For this reason, I think we can add Ascend logo in the deploy model module of ONNX supported tools page. Please feel free to ask me if you have any questions, thanks.

Comments

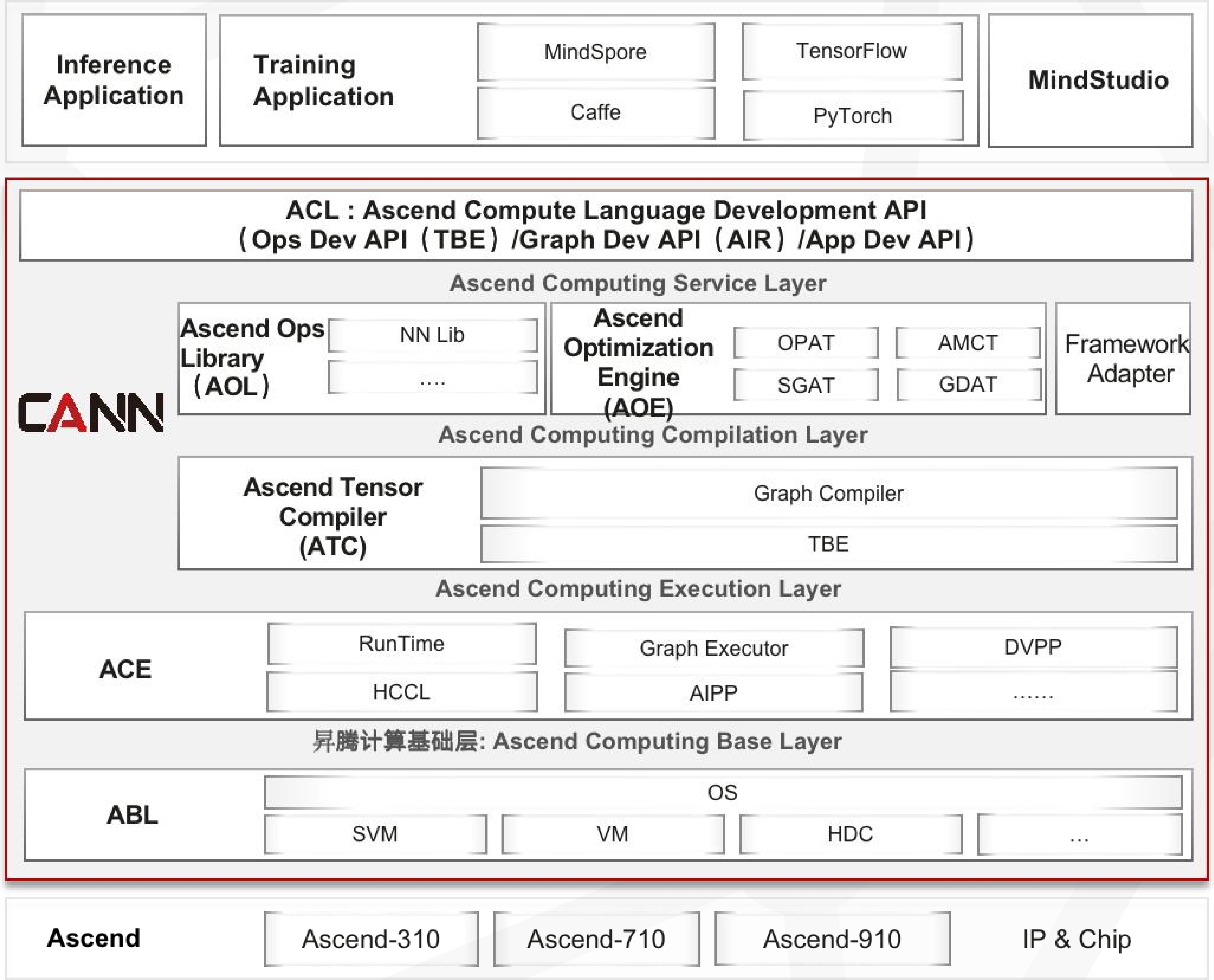
2

onnx/onnx.github.io | Jun 25th | Added by GitHub (Legacy)

🙌 1

AI Heterogeneous Computing Architecture: CANN 5.0

(Compute Architecture for Neural Networks)



Inference Application

Training Application: MindSpore, TensorFlow, Caffe, PyTorch

MindStudio

CANN

ACL：Ascend Compute Language Development API
（Ops Dev API（TBE）/Graph Dev API（AIR）/App Dev API）

Ascend Computing Service Layer

Ascend Ops Library（AOL）: NN Lib, ...

Ascend Optimization Engine (AOE): OPAT, AMCT, SGAT, GDAT

Framework Adapter

Ascend Computing Compilation Layer

Ascend Tensor Compiler (ATC): Graph Compiler, TBE

Ascend Computing Execution Layer

ACE: RunTime, Graph Executor, DVPP, HCCL, AIPP, ...

昇腾计算基础层: Ascend Computing Base Layer

ABL: OS, SVM, VM, HDC, ...

Ascend: Ascend-310, Ascend-710, Ascend-910, IP & Chip

What Is CANN

CANN is an AI Heterogeneous Computing Architecture which supports users to quickly develop AI applications on Ascend hardware platform via providing multiple layer of programming interfaces

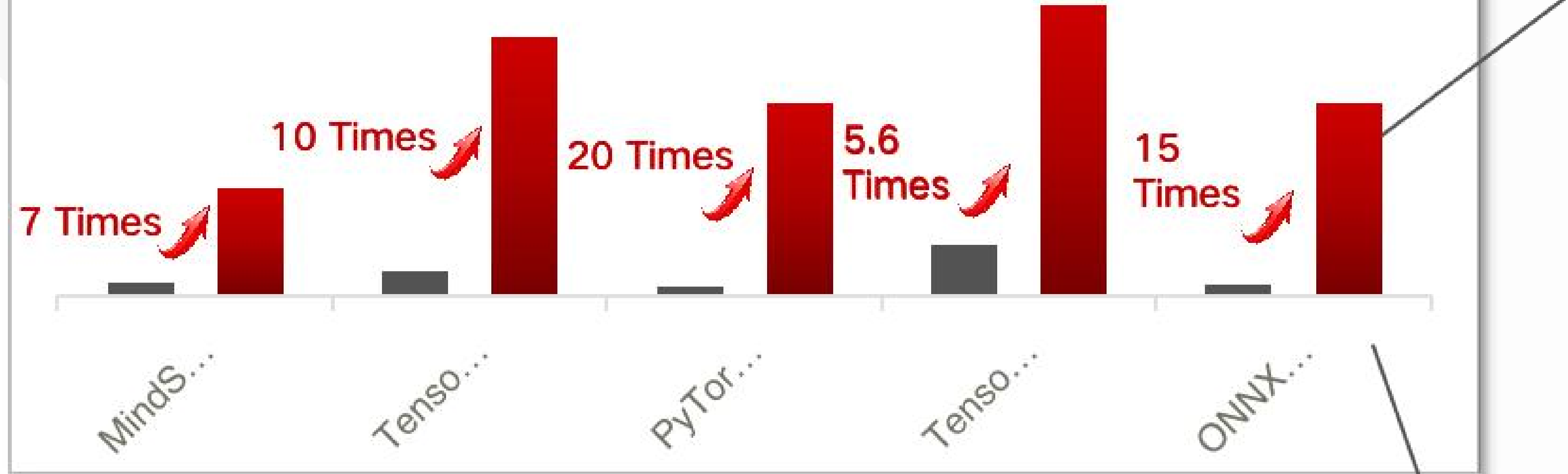
Key Features

- **Unified Appication Programming Interface：** ACL as the standardized programming interface which abstract underlying hardwares.
- **Unified Neural Network Graph Construction Interface：** AIR as the stadardized graph construction interface which supports multiple frameworks
- **High Performance Compute Engine and Operater Library**
- **Basic Service：** capabilites include drivers, virtualization, media, communications, etc.

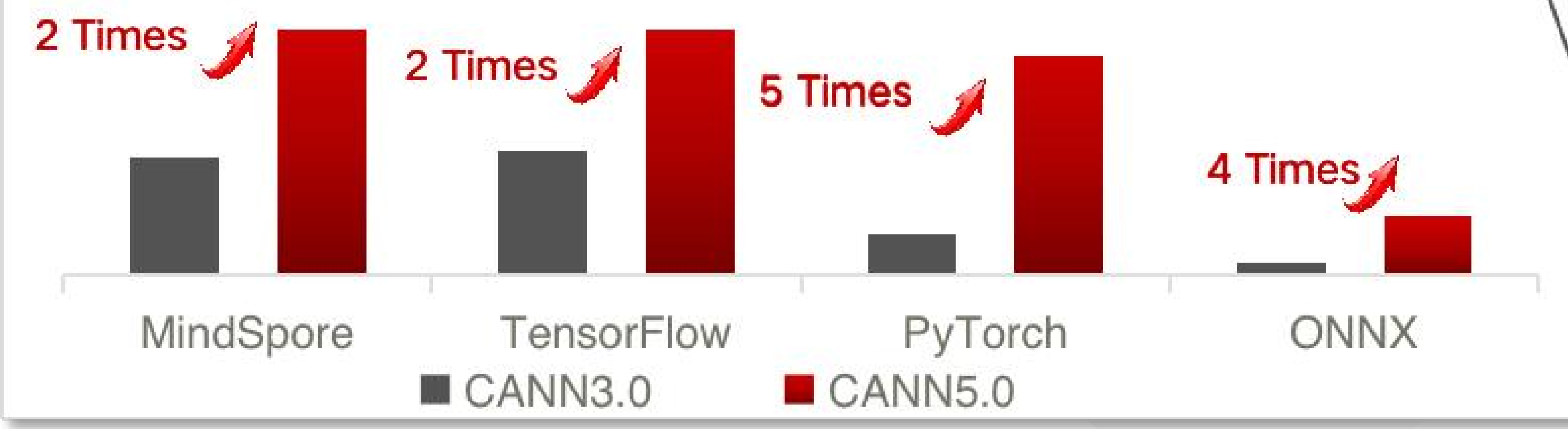
CANN 5.0 and onnx: accelerating inference model on Ascend

- Currently support 140+ onnx inference models, will reach to 200+ by the end of the year
- Support opset 8~13 with opset 11 as the key set, 90+% of the Ops will be supported on CANN by the end of the year

CANN5.0 Modelzoo Increase Dramatically



CANN5.0 Ops Support Enhanced



Ascend Modelzoo
（www.hiascend.com/software/modelzoo）

3D-Resnet	InceptionV4	RetinaNet+FPN
AlexNet	LSTM	RetinaNet-detectron2
BERT_BASE_UNCASED	MaskRCNN-NPU	SENet
Cascade_RCNN-detectron2	MGN	seresnext-50_32x4d
CascadeRCNN	MnasNet1_0	ShuffleNetV1
CRNN	MobileNetV1	ShuffleNetV2
CSPResNeXt50	MobileNetV2	ShuffleNetV2+
DeeplabV3+	MobileNetV3	SKNet50
deepmar	OSNet	SPNASNet100
Deit	PCB	SqueezeNet1_1
DenseNet121	PSENet	SSD-VGG16
DnCNN	RegNetX-1.6GF	Transformer
DPN	RegNetY-1.6GF	TransformerXL
EfficentNetB5	ReID-strong-baseline	UNet
EfficientNetB0	Res2Net101-v1b	UNet++
EfficientNetB3	Resnet101	VGG16
FasterRCNN	ResNet101	VGG19
FCN8S	Resnet152	Vilbert
GhostNet1.0x	ResNet152	VoVNet39
Googlenet	ResNet18	Wide_ResNet101_2
HRNet	Resnet34	wide_resnet50_2
I3D	ResNet34	Xception
ICNet	ResNet50	YoloV3
Inception-ResNet-V2	ResNeXt101_32x8d	YoloV4
InceptionV3	ResNeXt50	YoloV5

Future Thoughts

Pain points need to be addressed in the community：

1. PyTorch NLP and Audio models' export to ONNX is still very difficult，traning model's export to ONNX is also difficult for developers:

■Trace doesn't support loop and if；
■torchscript is underutlized in the process of exporting to ONNX
■There is a 35%~40% failure rate when exporting pytorch model to onnx

2. The iteration of Opset is very fast which creates difficulties for hardware engineers to do the adaption work