Microsoft

Deep Learning @ Scale at Microsoft

David Ku

Stanford Scaled ML Conference 2017

Agenda

- AI at Microsoft
- Scaling deep learning
  - Learnings from two scenarios
- Driving progress in deep learning
  - Challenges and solutions

AI at Microsoft

- Machine Learning has been an integral part of Microsoft products
  - Bing and Ads – push limit of scale & agility
  - Kinect and Hololens – new UI metaphors
  - Cortana and Xiaoice – human-to-computer
  - Office 365 and LinkedIn – knowledge graphs

- Microsoft Research has and continues to be active in ML/AI research
  - Speech, vision, gestures, input
  - Natural language and conversational dialogs
  - Machine reading comprehension
  - Systems for AI, AI for systems

- As a company, we reached an inflection point around AI

Data & Intelligence at the core

Growing AI Muscles at Microsoft

- Be great in Intelligent Agents and Assistants
  - Cortana, Xiaoice, Bots, virtual customer assistants
  - Natural language, dialog, memory and context, tasks and domains

- Strengthen AI capabilities and tech foundation
  - Data, telemetry, experimentation, modeling, developer tools
  - Strengthen AI tools and infrastructure for 1st party and 3rd party

- Actively Infuse AI into Microsoft products
  - Office, Dynamics, Azure intelligence
  - Delivering AI solutions & capabilities for our customers

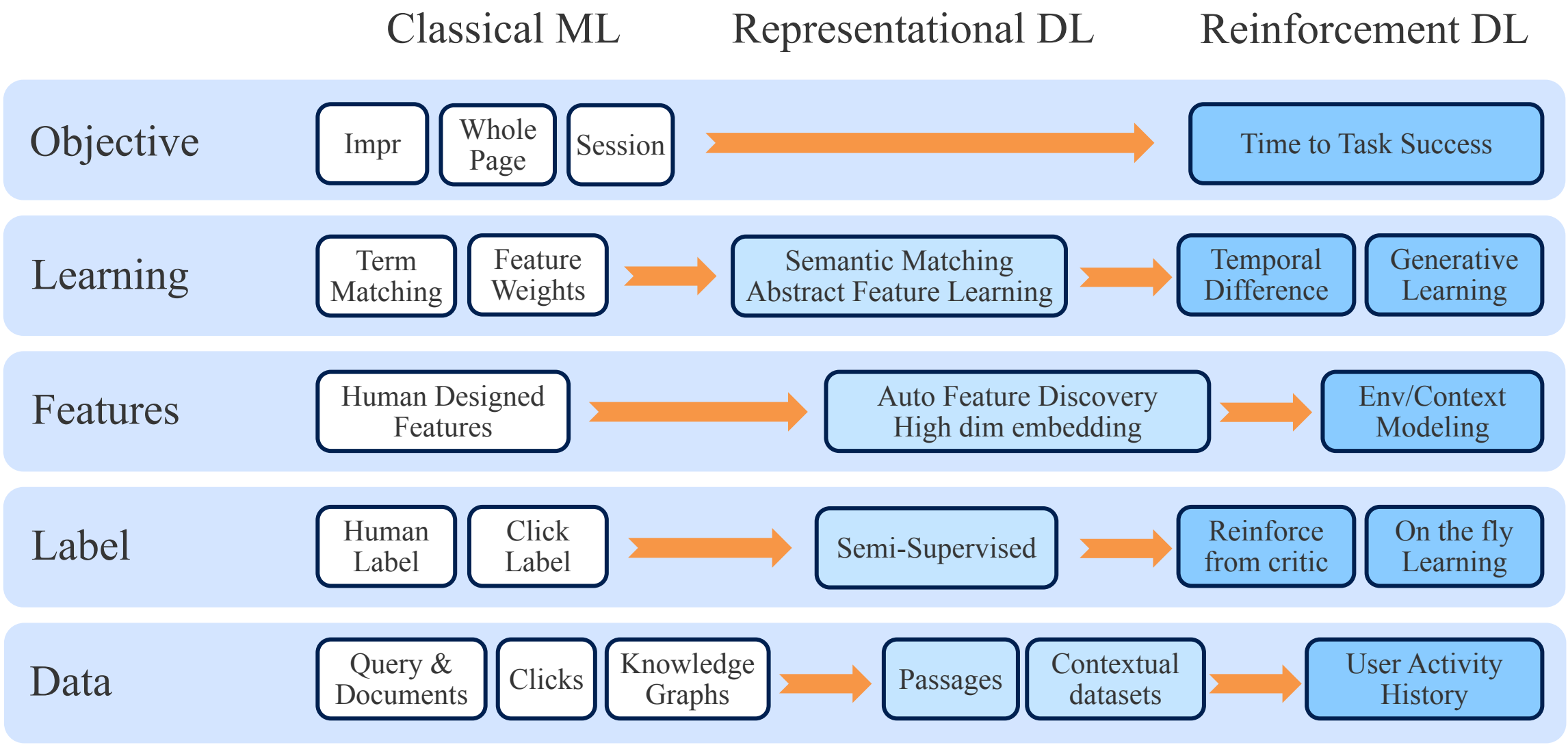
- Establish new organization to drive this: AI + Research

Two DL Scenarios

- Deep Question & Answering
- Every document in Web corpus, each having 30+ passages
- Encoding of questions and passages, joint modeling and training, reinforcement learning
- Critical: latency < 5-10ms

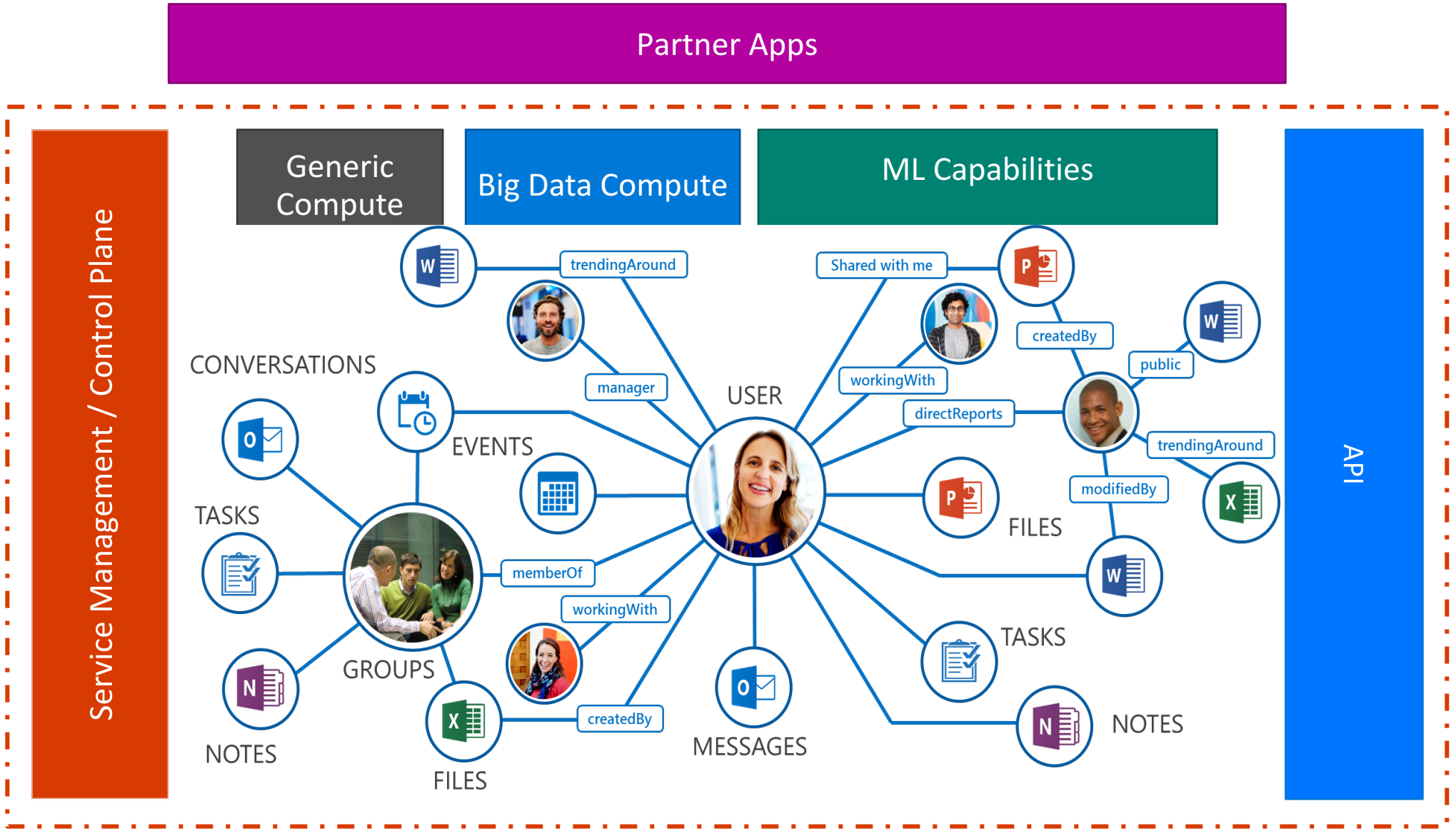
- Machine reading comprehension
- Massive: 18 Trillion emails, 250 billion updates & 5+ PB logs daily
- Representation of knowledge across neural and symbolic domains, with domain extensions
- Critical: strict data use compliance

Bing and Deep Learning Evolution



Office and Substrate Intelligence

Compliance Boundary



Scaling Deep Learning

5 Ingredients for Deep Learning[1]

- Data, lots of data
- Flexible models
- Enough computing power
- Computationally efficient inferences
- Priors that defeat curse of dimensionality

5 Challenges for Deep Learning

- Data policy and compliance
- Agility to experiment
- Cost-effective computing
- Low-latency runtime
- Vibrant community for collaboration and research

[1]Yoshua Bengio talk 2017

(1) Data policy and compliance

Challenge:

Office365 makes strong promise to users and business customers:

- Country privacy laws
- Contractual commitments
- Microsoft privacy policy

No eyes on access to content without explicit consent

Implication:

- Limited data access for human inspection and annotation
- Data processing must stay within compliance boundary
- Models need to be compliant, cannot reveal data between users and tenants
- Infrastructure must respect country data sovereignty

(1) Data policy and compliance

Solutions:

Data access through public datasets, employee/user donation programs, or approved anonymization strategies

Compliant foundation, where we apply strict engineering standards for building compliant software & services, leveraging O365 infrastructure services

Semi-supervised model training to augment and scale training

Actively exploring:

- Design user experiences that elicit and encourage feedback
- Hierarchical modeling of cross-tenant knowledge and patterns
- New approaches to privacy, e.g., differential or stochastic privacy
- Unsupervised model training

(2) Agility to Experiment

Challenge:

Model development often requires working with models and algorithms that are created on different DL frameworks

Lots of work required to manage E2E agility, including model development, offline experimentation, online flighting, and collaboration/sharing across groups

Significant time lag between small early pilots to production-ready deployments

Implication:

- Must support multiple DL frameworks, with tools to ease interoperability across them
- Must support full lifecycle of ML model development to deployment
- Must ensure there is automated path to evolve from small pilots to large-scale production runs

(2) Agility to Experiment

Solutions:

DL training cluster supports CNTK and Tensorflow models on GPUs, enabling agility to experiment and extend

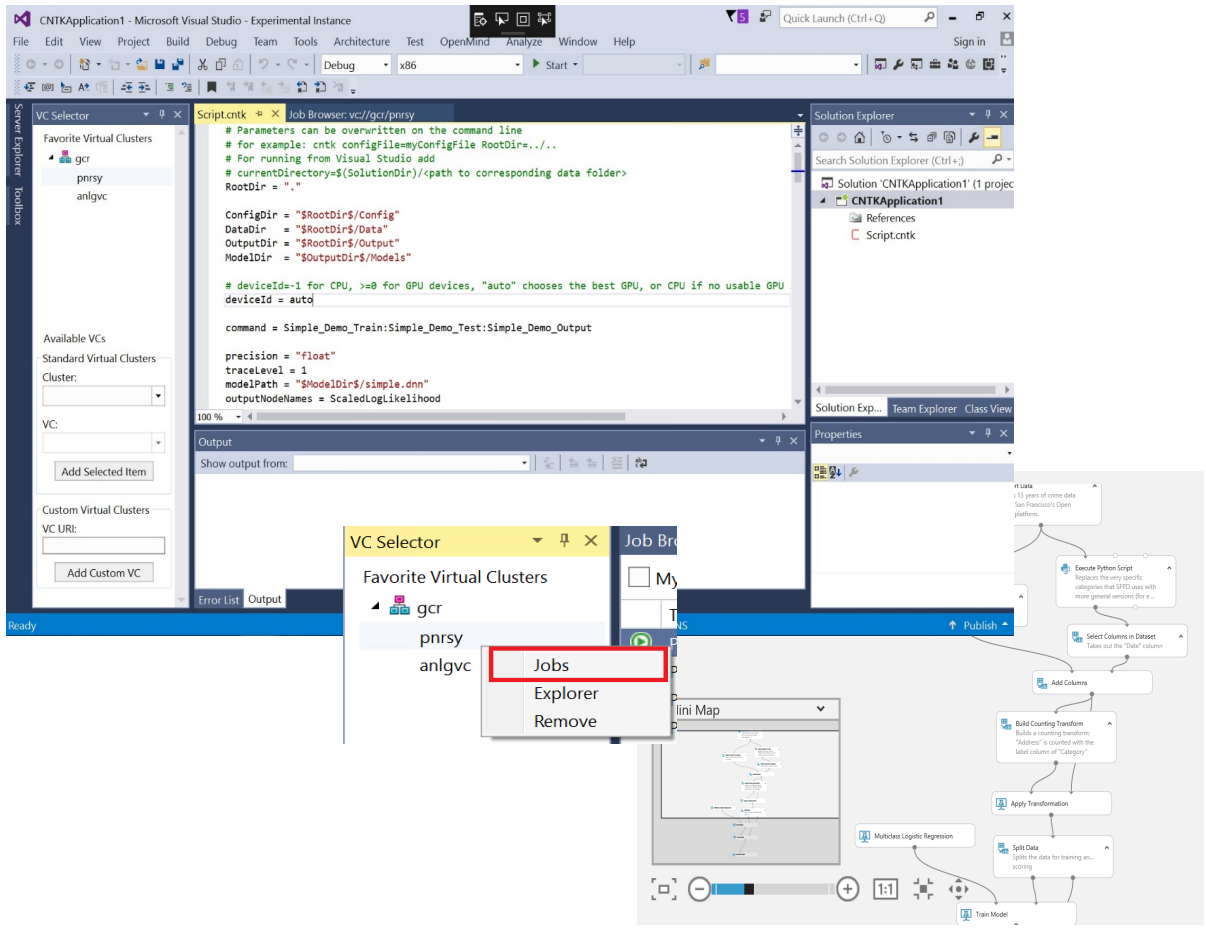
DL inference as a service to automate and streamline deep model runtime

OpenMind extended Visual Studio and VS Code to support DL frameworks (TF, CNTK, …) and Git to accelerate inner loop

Actively exploring:

- Model and dataset sharing for ML collaboration, full lifecycle
- Continue to embrace and integrate open source community of tools, code, and models into our environments
- Improve debugging, visualization, and exploration of DL models

OpenMind Studio



- IDE for Deep Learning, supports multiple DL frameworks & platforms (Visual Studio + VS Code)
- Rich editing and debugging, with Intellisense and symbolic execution
- Local execution and analysis, integrate with Tensorboard & tools
- Cloud deployment against multiple fabrics and computing backends
- Shared gallery on Git for better collaboration, reuse, and sharing

(3) Cost-effective Computing

Challenge:

Deep learning model training is expensive and time-consuming, and there is never enough computing resources

Software and hardware acceleration (GPU, FPGA, ASIC) is necessary but complicated to do at scale

Cost management is necessary and hard

Implication:

- We must dramatically (10x-100x) improve the performance/cost of compute to support DL
- We must hide the complexity of hardware acceleration from the model developers
- We should enable easy tradeoff between performance and costs

(3) Cost-effective Computing

Solutions:

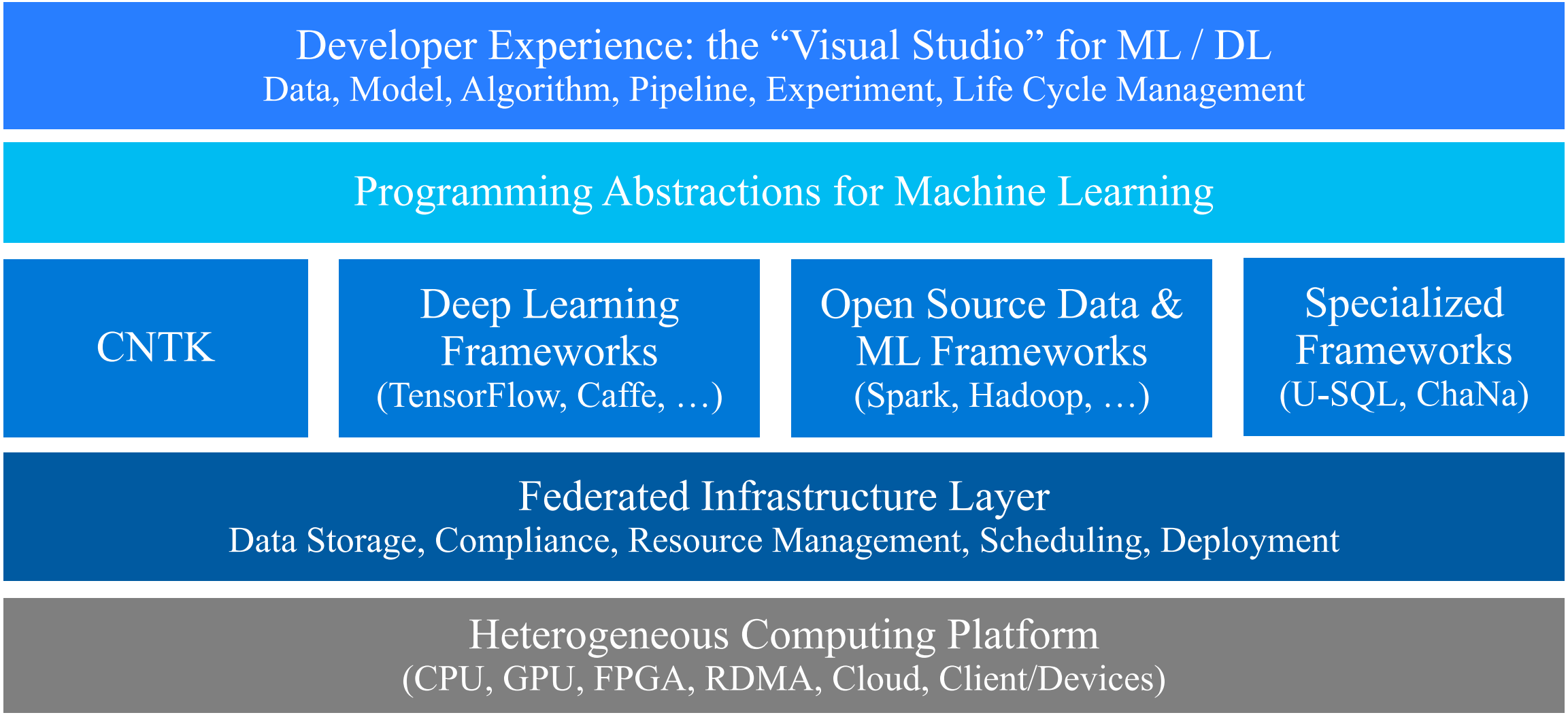
ML cloud infrastructure to dynamically optimize across heterogeneous compute resources (CPU, GPU, FPGA, and ASIC)

New AI stack that enables rapid ML/DL development and deployment, open architecture to support OSS

Actively exploring:

- Automated DL serving optimization of systems and algorithms to find the right operating point
- New hardware and networking configurations, including ASICs
- Optimization for on-device computing support (IoT, mobile)

Building the new AI stack



(4) Low-latency Runtime

Challenge:

Deep model evaluation is very expensive in terms of latency, often exceed several hundred milliseconds per evaluation using default runtimes on CPU

Leveraging hardware acceleration (e.g,. FPGA or ASIC) requires significant effort, and will need to evolve to new DNN structures and precisions

Implication:

- Reconfigurability of FPGA provides flexibility, but mapping DNN to FPGA in optimized way is not easy to do at scale
- ASIC can be very cost-effective with low latency, but not always available

(4) Low-latency Runtime

Solutions:

FPGA for DNN supports virtualized FPGA configured in a Hardware-as-a-Service (HaaS) pool, with automated compiler to map from CNTK/TF specification automatically to FPGA deployment over HaaS

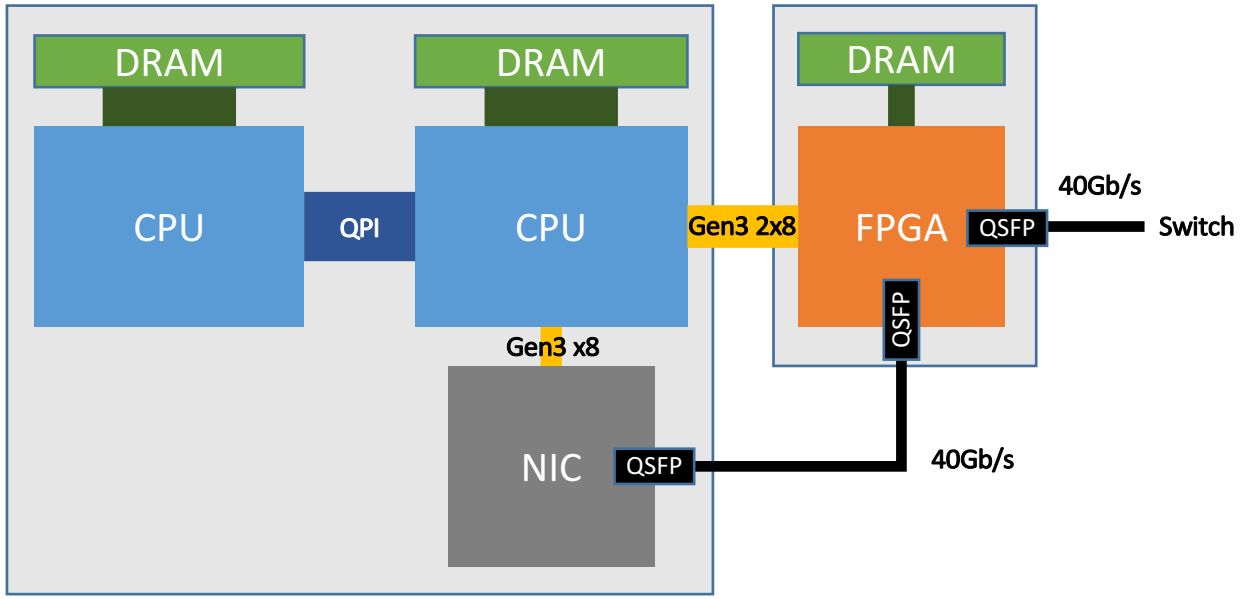
Azure Inference service to enable FPGA-based inference service for 3rd party Azure customers

Actively exploring:

- Model compression and optimizing with reduced model precision can deliver even greater efficiency and tradeoff, active exploration underway
- Extend HaaS to support both FPGA and ASIC, in a transparent way to the model developers

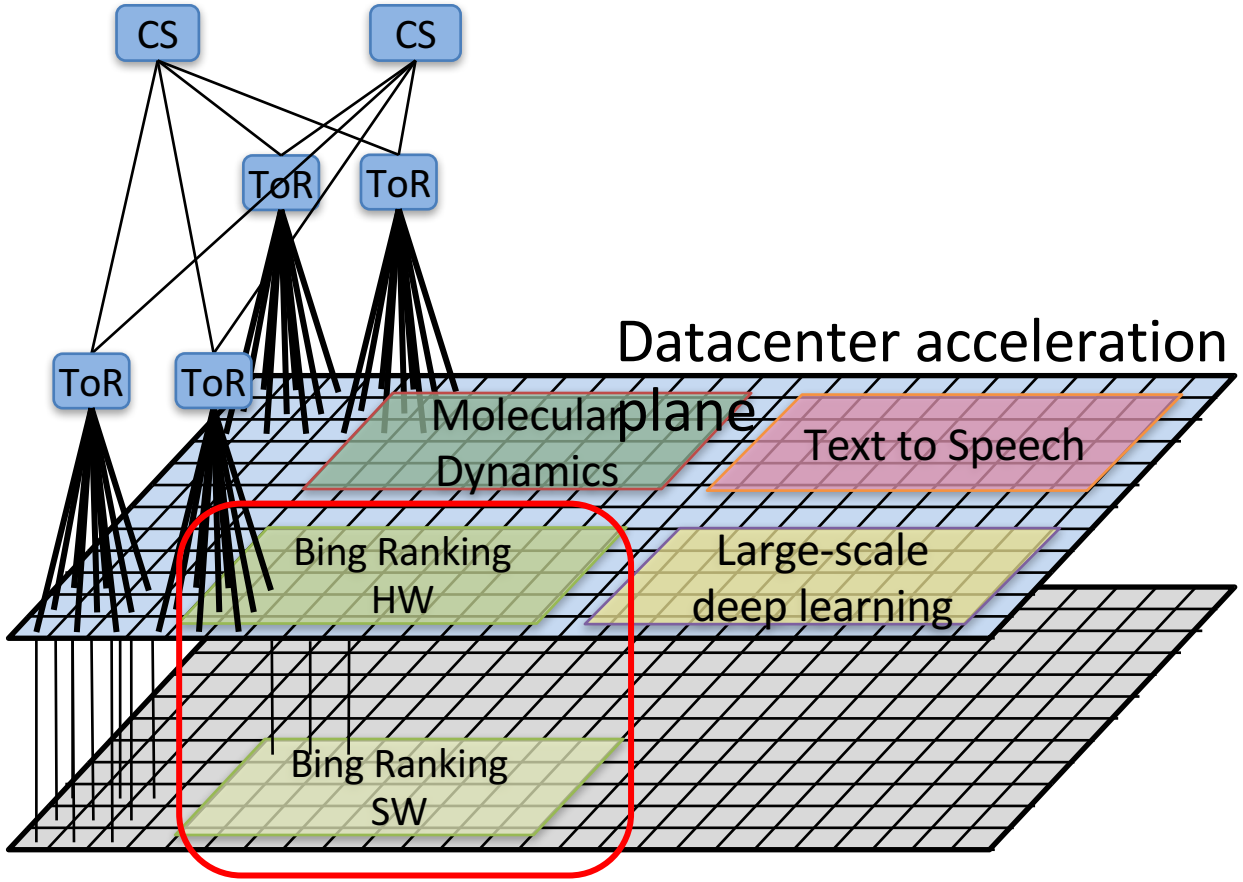
DNN Inference Acceleration using FPGA

Catapult FPGA architecture



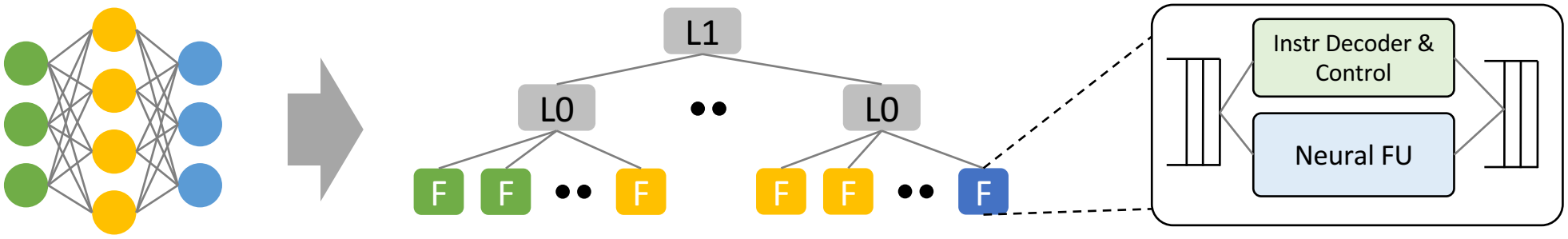
- Catapult is our cloud-scale FPGA architecture, deployed across ALL Microsoft servers
- For ultra-low latency DNN inference, we need to fit model parameters in FPGA on-chip memory, and extend to multiple FPGAs for large models
- Automated DNN compiler to map from CNTK/TF to FPGA runtime

Hardware-as-a-Service (HaaS)



- Enables solving bigger problems than possible on single FPGA
- Enables harvesting FPGA resources across network
- Services communicate with no SW intervention (via Lightweight Transport Layer, in microseconds)
- Line-rate service are local (Crypto)

Brainwave Compiler: Scalable DNN over HaaS



NN Model

FPGAs over HaaS and LTL

BrainSlice Engine

BrainWave Goals

- Ultra low latency, high throughput evaluation
- Achieve high ops/$ and ops/W vs. CPUs
- Long term, enable scalable in-situ training for model freshness & online learning

BrainSlice: SW-Programmable DNN Engine

- 1.2 TOPs of 16b fixed point ➔ inference **in hundreds of us or few ms**
- No Verilog expertise required
- BrainSlices can be composed to support large scale models

(5) Vibrant Community for DL Research

Challenge:

Deep learning is still a nascent field of study, requires commitment to research and open collaboration

Need to foster and support collaboration between academia and industry, and across companies and organizations, to advance the technology and policies around deep learning

Implication:

- Be part of the community, actively contribute
- Be proactive around the policy and principles of AI as a positive force on people and society
- Engage in joint research and collaboration, shared datasets and models

(5) Vibrant Community for DL Research

Solutions:

Embrace and Contribute to OSS such as Spark, YARN, HDFS, Dockers, HDX-1, OpenMind studio and AI tools

CNTK and TF contribute as part of open source, to learn and share with the DL community

Research and Datasets such as MSMARCO for reading comprehension dataset, and ResNet for advanced DL

Actively exploring:

- Establish more and deeper collaboration with academia to advance research topics, actively support their research efforts
- Open up more of Microsoft services and datasets to the DL community to help advance the field

Q&A

- Look forward to working with everyone
- Thank You!