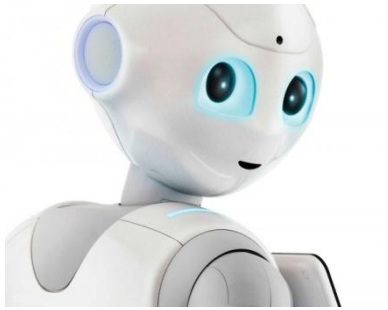
Introducing the ReQuEST competitions, platform, scoreboard and long-term vision

Open and reproducible tournaments for Pareto-efficient AI/SW/HW co-design



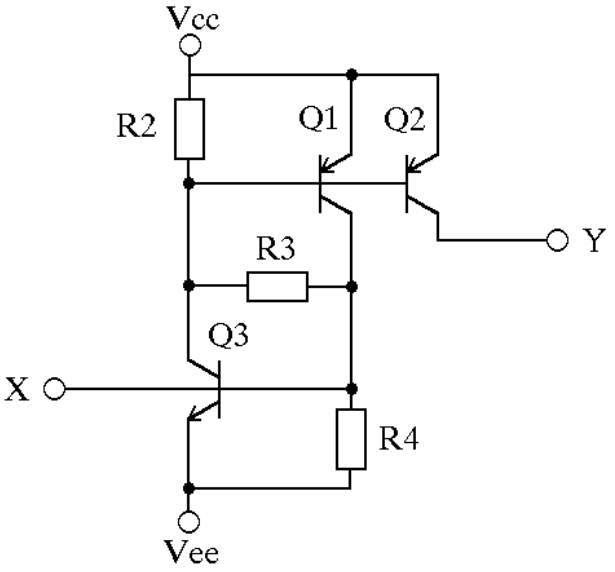



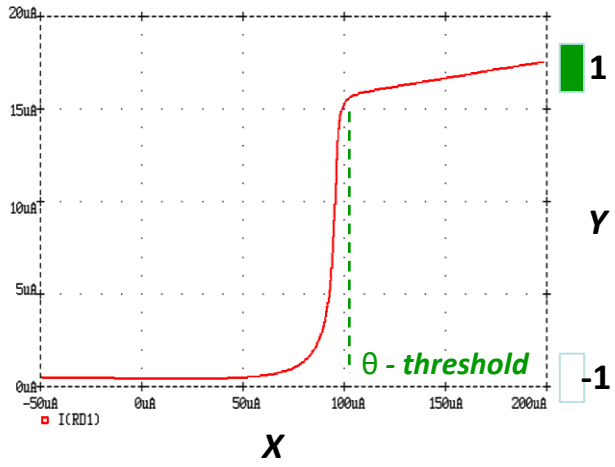

cTuning Foundation

Grigori Fursin
Chief Scientist, non-profit cTuning foundation
CTO and co-founder, dividiti

$$\frac{d\vec{v}}{dt}$$

Looking back to 1993: my first cross-disciplinary R&D project with industry





Semiconductor neuron (analog computation)

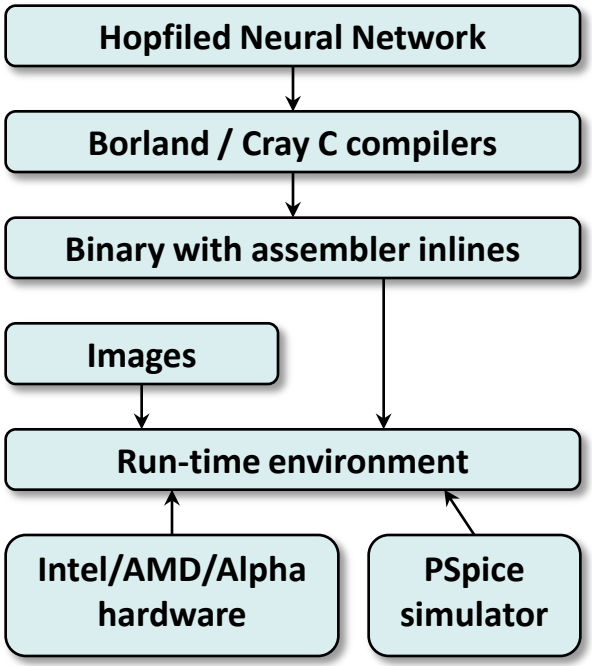
Designing brain-inspired computer

10x smaller 10x more accurate
100x faster / more energy efficient
then traditional platforms

AI / ML use cases

Looking back to 1993: my first cross-disciplinary R&D project with industry

Emulation of training/prediction on personal computers or supercomputers (Cray T3D)



Must have been solved by now with new technology?

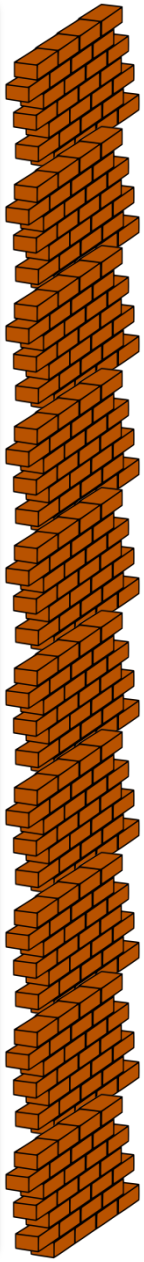
Are we there yet?

Designing brain-inspired computer

10x smaller 10x more accurate
100x faster / more energy efficient
then traditional platforms

... **failed** because modeling using available HW/SW was
- too complex
- too slow
- too unreliable and costly

and we didn't have GPGPUs and highly optimized math/NN libraries



AI / ML use cases

25 years later: AI and ML revolutionizes multiple industries

2018: many cross-disciplinary R&D groups (ML/AI/systems)

AI hardware
- All major vendors (Google, NVIDIA, IBM, Intel, ARM, Qualcomm, Apple, AMD …)

AI models
Numerous groups in academia & industry (DeepMind, IBM, OpenAI, Microsoft, Facebook …)

AI software
- AI frameworks (TensorFlow, MXNet, Caffe2, CNTK, Theano)
- AI libraries (cuDNN, libDNN, ArmCL, OpenBLAS)

AI integration/services
- Cloud services (AWS, Google, Watson, Azure ...)

Practical and successful AI system must be co-designed

Hardware | **Algorithms**

Are we there yet?

Data sets | **Libraries**

for various form factors
(IoT, mobile, data centers)




while trading off multiple constraints
(accuracy, speed, energy, size, costs)

and maximizing ROI
(faster time to market, R&D sustainability, much better than all competitors)

AI / ML use cases






Machine learning and artificial intelligence became very hot topics

2018: many cross-disciplinary R&D groups (ML/AI/systems)

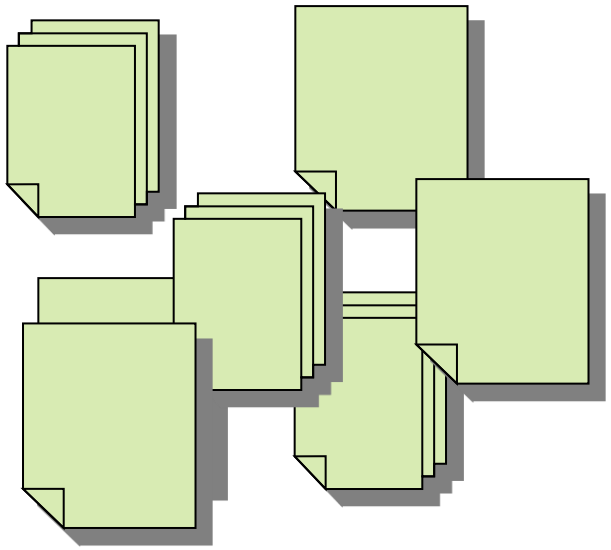
AI hardware
- All major vendors (Google, NVIDIA, IBM, Intel, ARM, Qualcomm, Apple, AMD …)

AI models
Numerous groups in academia & industry (DeepMind, IBM, OpenAI, Microsoft, Facebook …)

AI software
- AI frameworks (TensorFlow, MXNet, Caffe2, CNTK, Theano)
- AI libraries (cuDNN, libDNN, ArmCL, OpenBLAS)

AI integration/services
- Cloud services (AWS, Google, Watson, Azure ...)

Numerous publications and reports



Numerous models, data sets, benchmarks, libraries and tools

Multiple competitions focusing mostly on accuracy (Kaggle)

A few competitions focusing on optimizing other metrics besides accuracy:

LPIRC – Low-Power Image Recognition Challenge

AI / ML use cases

Industrial adoption of AI/ML is still very slow

2018: many cross-disciplinary R&D groups (ML/AI/systems)

AI hardware
- All major vendors (Google, NVIDIA, IBM, Intel, ARM, Qualcomm, Apple, AMD …)

AI models

Numerous groups in academia & industry (DeepMind, IBM, OpenAI, Microsoft, Facebook …)

AI software
- AI frameworks (TensorFlow, MXNet, Caffe2, CNTK, Theano)
- AI libraries (cuDNN, libDNN, ArmCL, OpenBLAS)

AI integration/services
- Cloud services (AWS, Google, Watson, Azure ...)

→

- Technological chaos: continuously changing algorithm/model/SW/HW stack

- Outdated/non-representative training sets
- No established methodologies and automation to benchmark and co-design efficient SW/HW/model stack
- Very little artifact sharing & reuse (optimizations, features, mispredictions, etc)
- **Growing gap between academic and industrial research (toy examples)**

Often result in over-provisioned, under-performing, inaccurate and expensive technology

↓ Will die

← Must be redesigned

AI / ML use cases

Artifact evaluation and ACM taskforce on reproducibility

In 2016 we joined special ACM taskforce on reproducibility
to develop a common methodology for artifact sharing and evaluation across all SIGS!

We co-authored "**Result and Artifact Review and Badging**" policy:

http://www.acm.org/publications/policies/artifact-review-badging

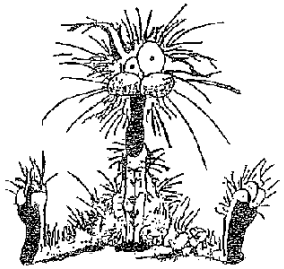
1) Define terminology

Repeatability (Same team, same experimental setup)
Replicability (Different team, same experimental setup)
Reproducibility (Different team, different experimental setup)

2) Prepare new sets of badges (covering various SIGs)

Artifacts Evaluated – Functional
Artifacts Evaluated – Reusable
Artifacts Available
Results Replicated
Results Reproduced

Artifact Evaluation did not solve reusability issues



Algorithm

CUDA 7.x LLVM 3.7 function-level pass reordering LLVM 3.4 ARM v8 MPI OpenCL OpenMP ICC 11.1 prof gprof ATLAS CUDA 4.x Intel SandyBridge LTO GCC 4.2.x TBB per phase GCC 5.2 LLVM 3.0 reconfiguration perf AVX MVS 2013 HMPP memory size GCC 4.5.x frequency polyhedral XLC execution time transformations IPA hardware GCC 4.3.x ICC 12.0 program predictive Scalasca counters SSE4 level scheduling ICC 11.0 MKL bandwidth GCC 4.4.x GCC 4.6.x LLVM 2.6 LLVM 2.9 SimpleScalar Codelet PAPI ICC 12.1 Jikes Testarossa Amplifier algorithm precision HDD size Open64 oprofile TLB loop-level GCC 4.7.x SVM VTune LLVM 2.7 GCC 4.1.x algorithm-level ICC 10.1 TAU Phoenix

Result

- everyone uses their own ad-hoc scripts to prepare and run experiments with many hardwired paths

- difficult (sometimes impossible) to reproduce empirical results across ever changing software and hardware stack (highly stochastic behavior)

- practically impossible to customize and reuse artifacts (for example, try another compiler, library, data set)

- practically impossible to run on another OS or platform

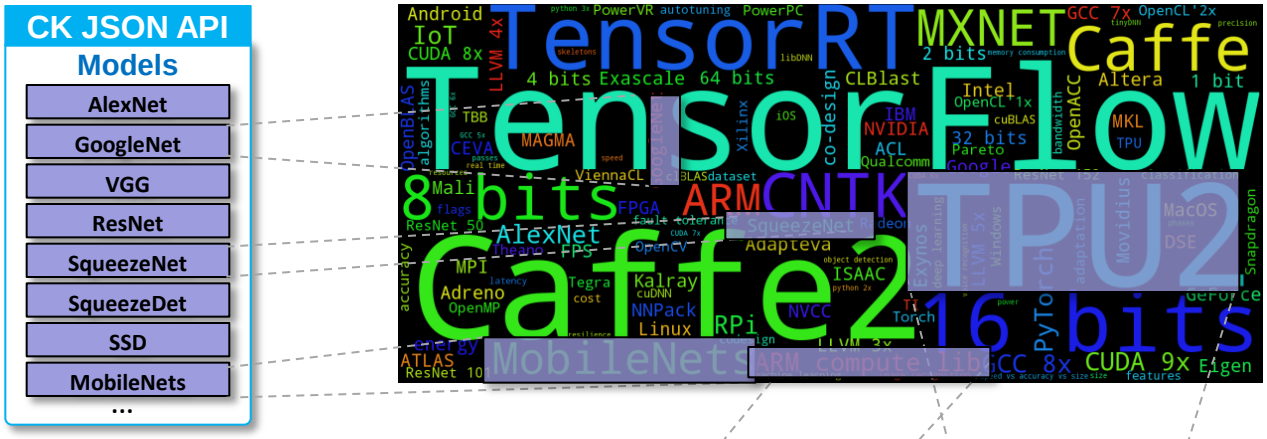
- no common API and meta information for shared artifacts and results (benchmarks, data sets, tools)

Our practical approach: common framework to share and reuse artifacts and knowledge

Open-source Collective Knowledge framework (CK)

cKnowledge.org ; github.com/ctuning/ck

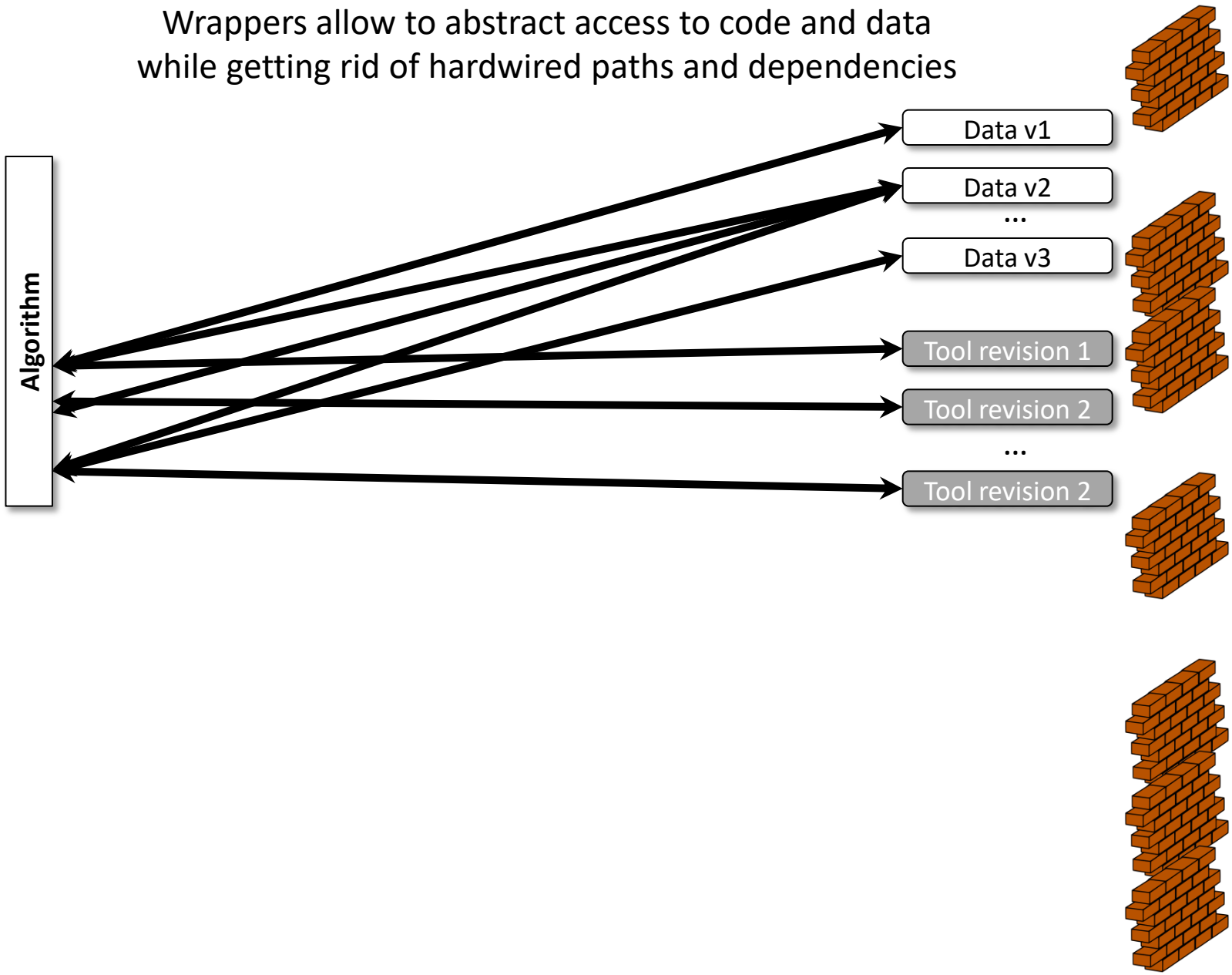
1) Implement and share Python wrappers
with a common API an unified JSON meta-information
for common groups of research artifacts
(models, data sets, libraries, frameworks, hardware, environments)



AI / ML use cases

Our practical approach: common framework to share and reuse artifacts and knowledge

Wrappers allow to abstract access to code and data
while getting rid of hardwired paths and dependencies



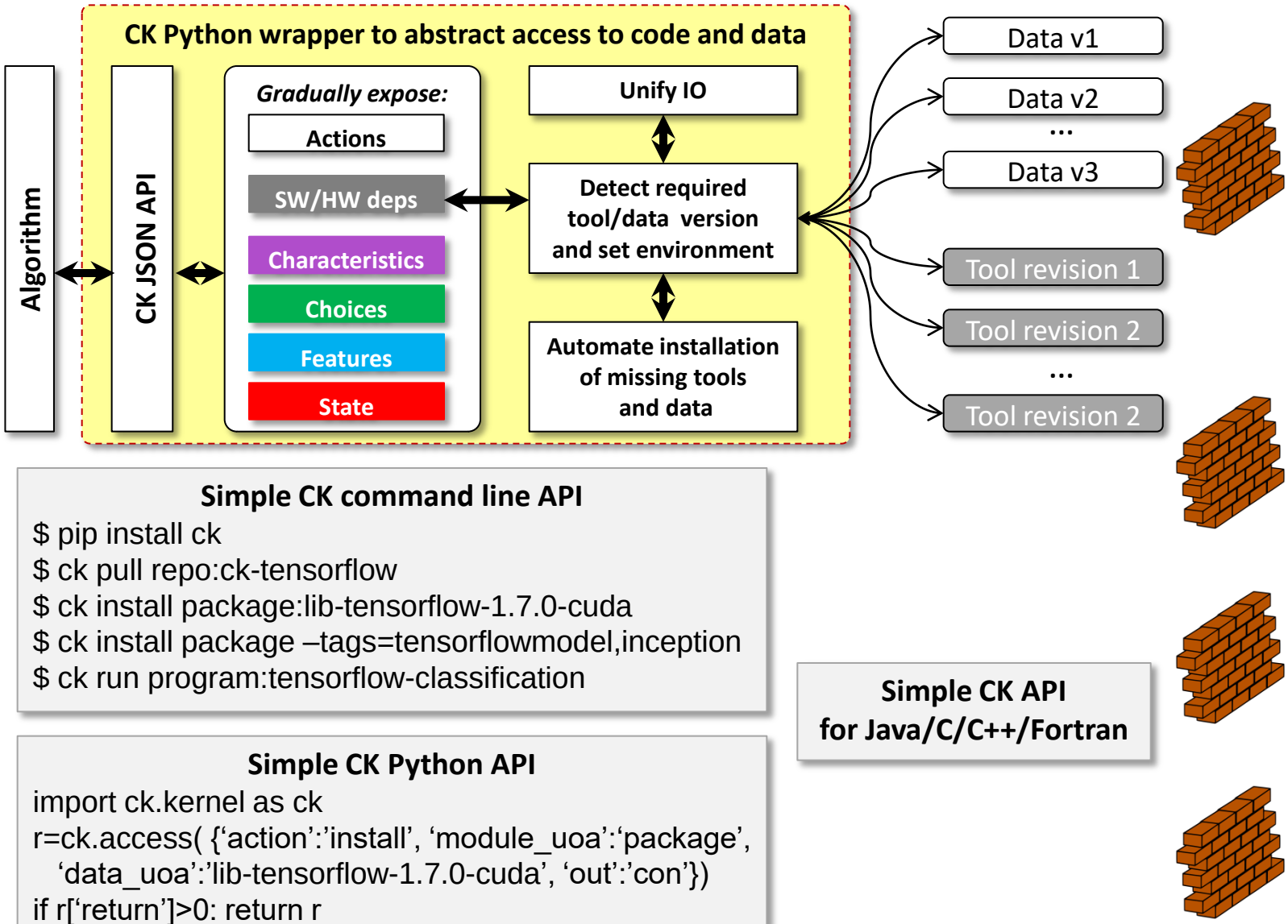
AI / ML
use cases






Our practical approach: common framework to share and reuse artifacts and knowledge

2) Gradually expose and unify information required for SW/HW co-design via JSON. Connect with CK cross-platform software and package manager.



Simple CK command line API

```
$ pip install ck
$ ck pull repo:ck-tensorflow
$ ck install package:lib-tensorflow-1.7.0-cuda
$ ck install package –tags=tensorflowmodel,inception
$ ck run program:tensorflow-classification
```

Simple CK Python API

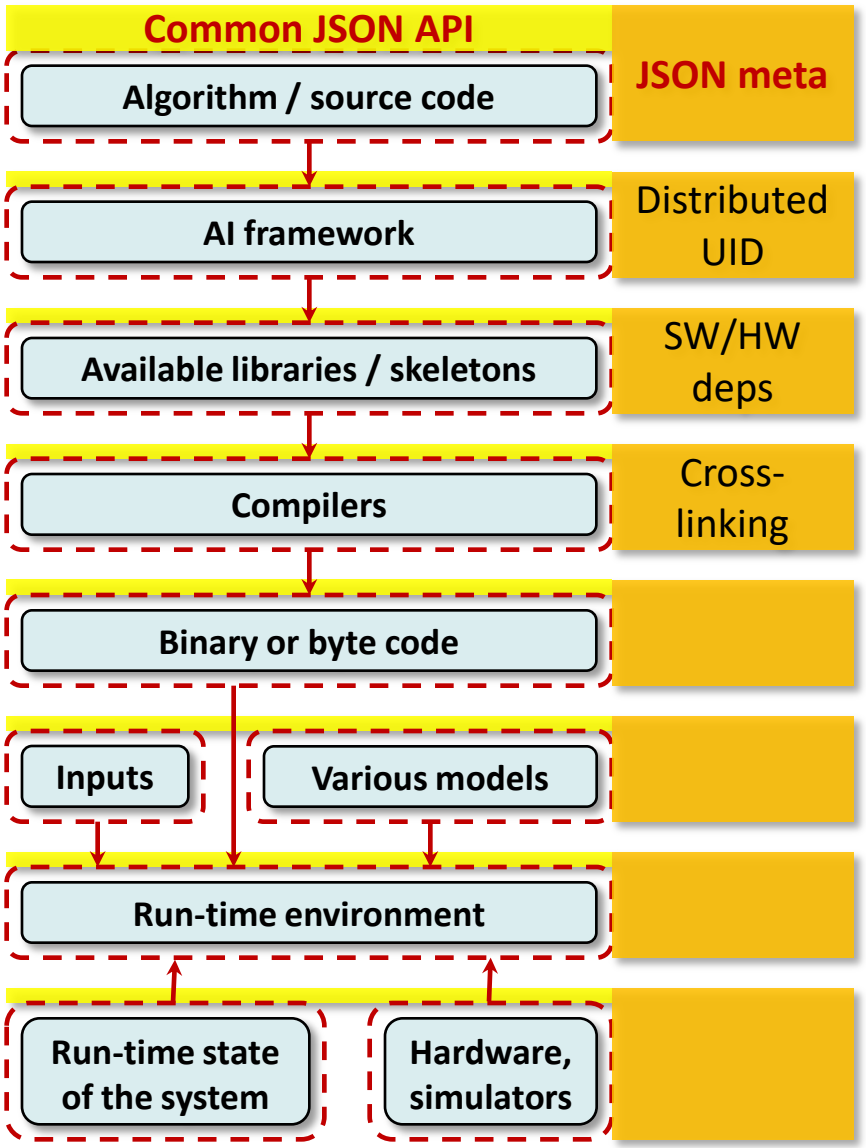
```
import ck.kernel as ck
r=ck.access( {‘action’:’install’, ‘module_uoa’:‘package’,
   ‘data_uoa’:’lib-tensorflow-1.7.0-cuda’, ‘out’:’con’})
if r[‘return’]>0: return r
```

Simple CK API for Java/C/C++/Fortran

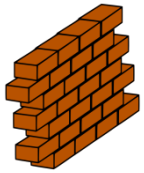
AI / ML use cases

Our practical approach: common framework to share and reuse artifacts and knowledge

3) Assemble and share portable experimental workflows
from customizable and reusable "plug&play" CK blocks as LEGO™



Implement universal, multi-objective and multi-dimensional auto-tuning, modeling and co-design



$$\vec{b} = B(\vec{c}, \vec{f}, \vec{s})$$

Flattened JSON vectors

Optimize behavior **b** of any object in the CK (program, library function, kernel, …) as a function of design and optimization choices **c**, features **f** and run-time state **s**

AI / ML use cases

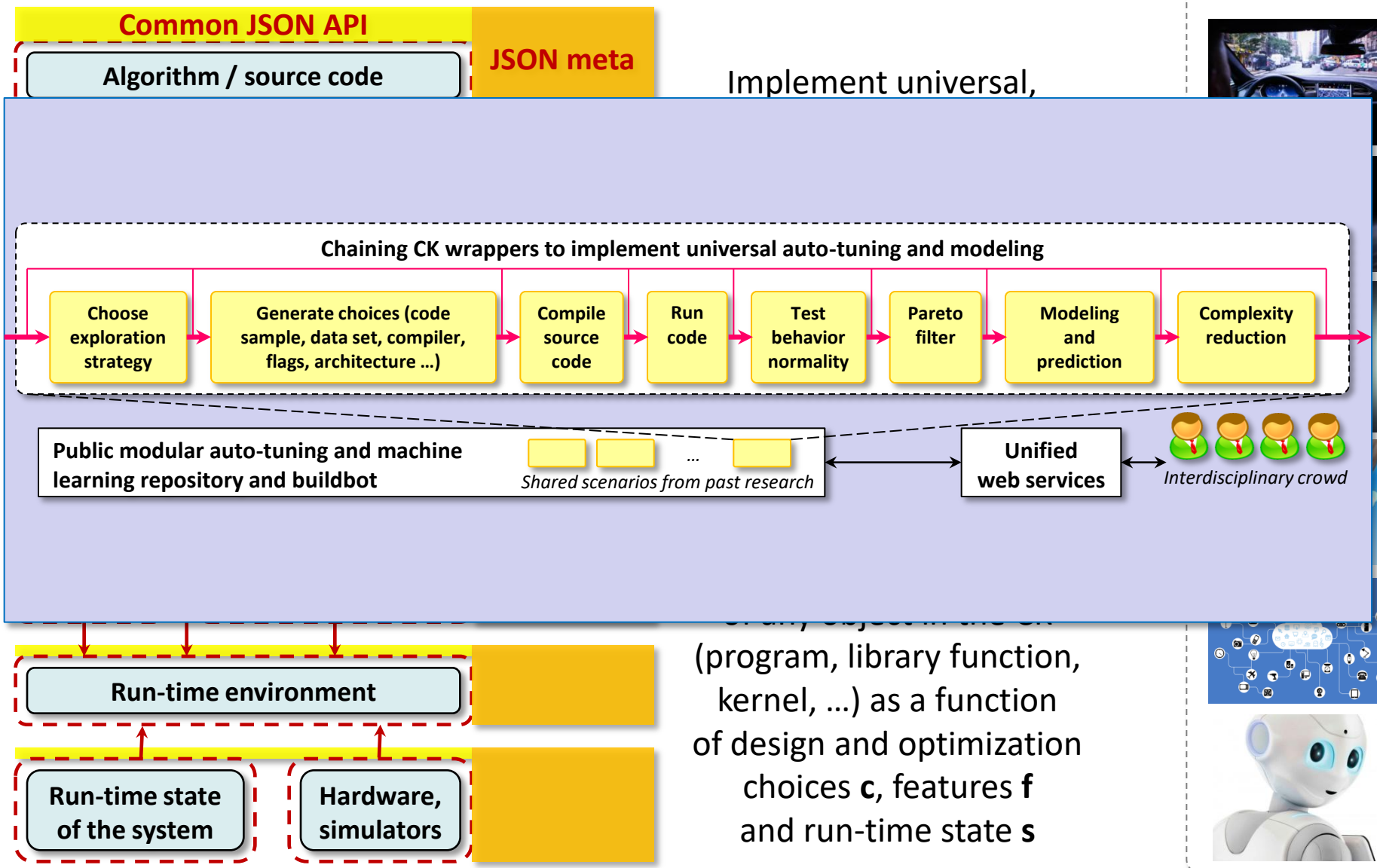





See **cKnowledge.org/shared-repos**

Our practical approach: common framework to share and reuse artifacts and knowledge

3) Assemble and share portable experimental workflows from customizable and reusable "plug&play" CK blocks as LEGO™



Implement universal, (program, library function, kernel, ...) as a function of design and optimization choices **c**, features **f** and run-time state **s**

AI / ML use cases

See **cKnowledge.org/shared-repos**

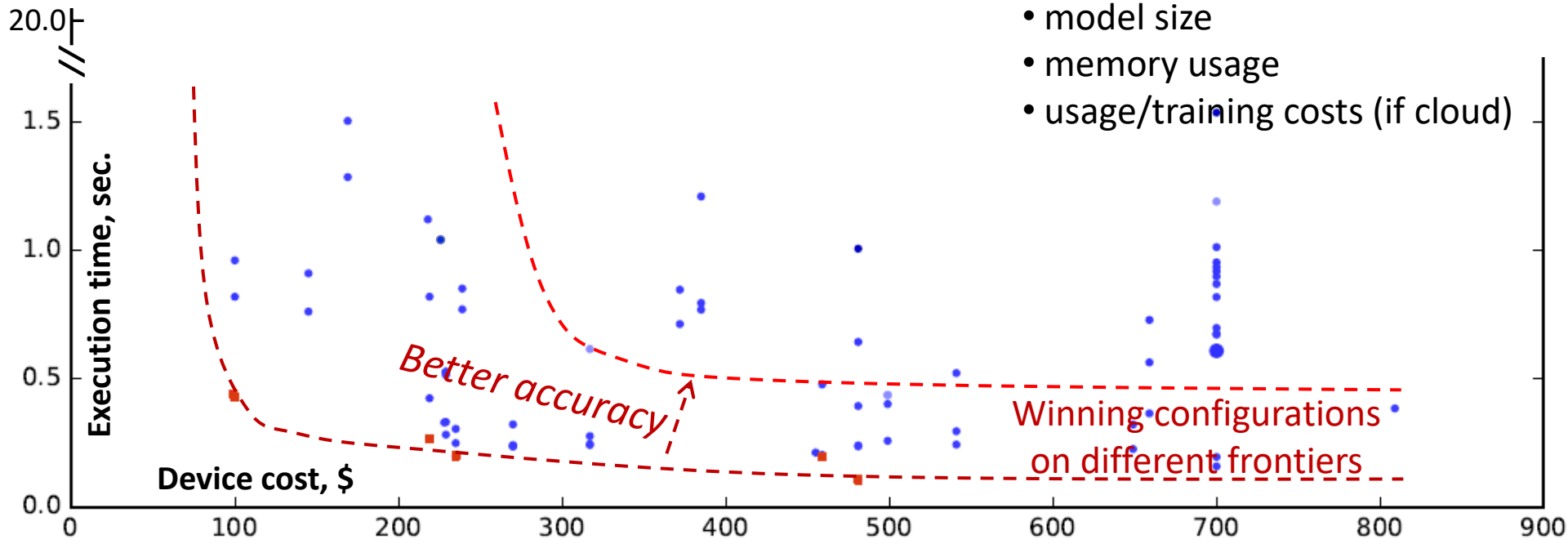
Simple CK-based Android app to crowdsource AI benchmarking

We evaluated ~3000 AI/SW/HW co-design configurations for efficiency

- **Hardware:** 800+ distinct platforms provided by volunteers (mainly low-power CPUs and GPUs)
- **Algorithms:** image classification, object detection
- **AI frameworks:** TensorFlow, Caffe
- **Math libraries:** OpenBLAS, CLBlast, ViennaCL, Eigen
- **Models:** AlexNet, GoogleNet, SqueezeNet, MobileNets
- **Data sets:** ImageNet, KITTI and user images

Characteristics:

- speed (execution time,sec.)
- device cost ($)
- energy (if available)
- model accuracy
- model size
- memory usage
- usage/training costs (if cloud)



Out of 3000 AI/SW/HW configurations only 3 were selected on various Pareto frontiers.

All AI/SW/HW configurations above Pareto frontiers lose competition (not suitable for AI)!

*Most data is now publicly available in an interactive and reproducible form at **cKnowledge.org/repo**

ReQuEST: reproducible tournaments on Pareto-efficient SW/HW co-desiging of AI workloads

2018: many cross-disciplinary R&D groups (ML/AI/systems)

AI hardware
- All major vendors (Google, NVIDIA, ARM, Intel, IBM, Qualcomm, Apple, AMD …)

AI models
Numerous groups in academia & industry (DeepMind, OpenAI, Microsoft, Facebook …)

AI software
- AI frameworks (TensorFlow, MXNet, Caffe2, CNTK, Theano)
- AI libraries (cuDNN, libDNN, ArmCL, OpenBLAS)

AI integration/services
- Cloud services (AWS, Google, Azure ...)

cKnowledge.org/request

Finding the most efficient AI/SW/HW stacks across diverse models, data sets and platforms via open competitions, share them as reusable CK components and visualize on a public scoreboard

Organizers (A-Z)

Luis Ceze, University of Washington
Natalie Enright Jerger, University of Toronto
Babak Falsafi, EPFL
Grigori Fursin, cTuning foundation/dividiti
Anton Lokhmotov, dividiti
Thierry Moreau, University of Washington
Adrian Sampson, Cornell University
Phillip Stanley Marbell, University of Cambridge

Collective Knowledge Platform

Interdisciplinary community

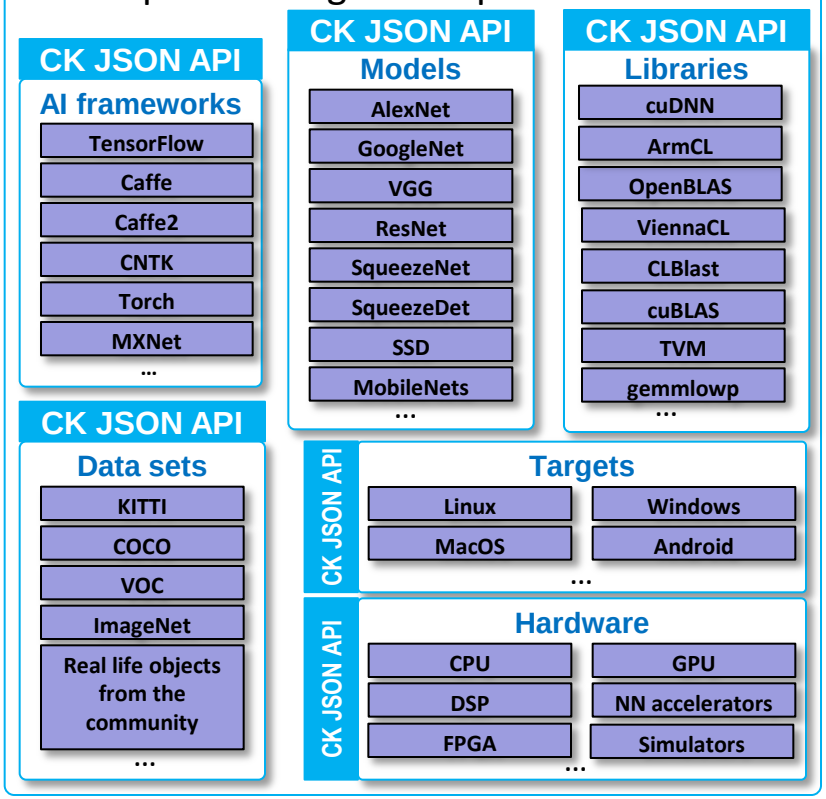
AI / ML use cases





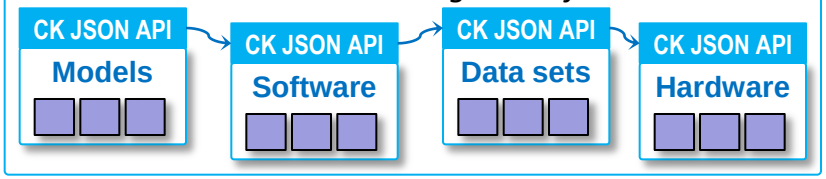

ReQuEST vision: common SW/HW co-design platform and repository

1) Repositories of customizable, portable and plug&play AI/SW/HW CK components with exposed design and optimization choices



2) Customizable CK workflow framework for automatic AI/SW/HW co-design

Assemble scenarios such as image classification as LEGO™



3) Regular ReQuEST tournaments sponsored by ACM

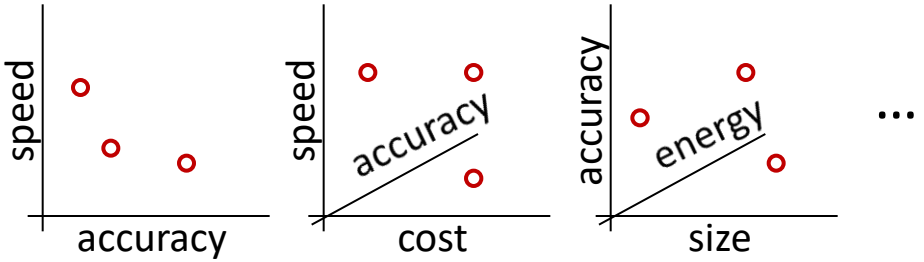
cKnowledge.org/request

and organized by leading universities (Cornell, EPFL, Washington, Toronto, Cambridge) and the growing industrial consortium to find the most efficient AI/SW/HW stacks across diverse models, data sets and platforms and share them as CK components

4) Winning AI/SW/HW stacks and workflows are presented on a live scoreboard and become available for further customization, optimization and reuse via CK

cKnowledge.org/repo

different co-design categories



Advisory Board

Advisory/industrial board (A-Z)

- **Michaela Blott**, Xilinx
- **Unmesh Bordoloi**, General Motors
- **Ofer Dekel**, Microsoft
- **Maria Girone**, CERN openlab
- **Wayne Graves**, ACM
- **Vinod Grover**, NVIDIA
- **Sumit Gupta**, IBM
- **James Hetherington**, Alan Turing Institute
- **Steve Keckler**, NVIDIA
- **Wei Li**, Intel
- **Colin Osborne**, ARM
- **Andrew Putnam**, Microsoft
- **Boris Shulkin**, Magna
- **Greg Stoner**, AMD
- **Alex Wade**, Chan Zuckerberg Initiative
- **Peng Wu**, Huawei
- **Cliff Young**, Google

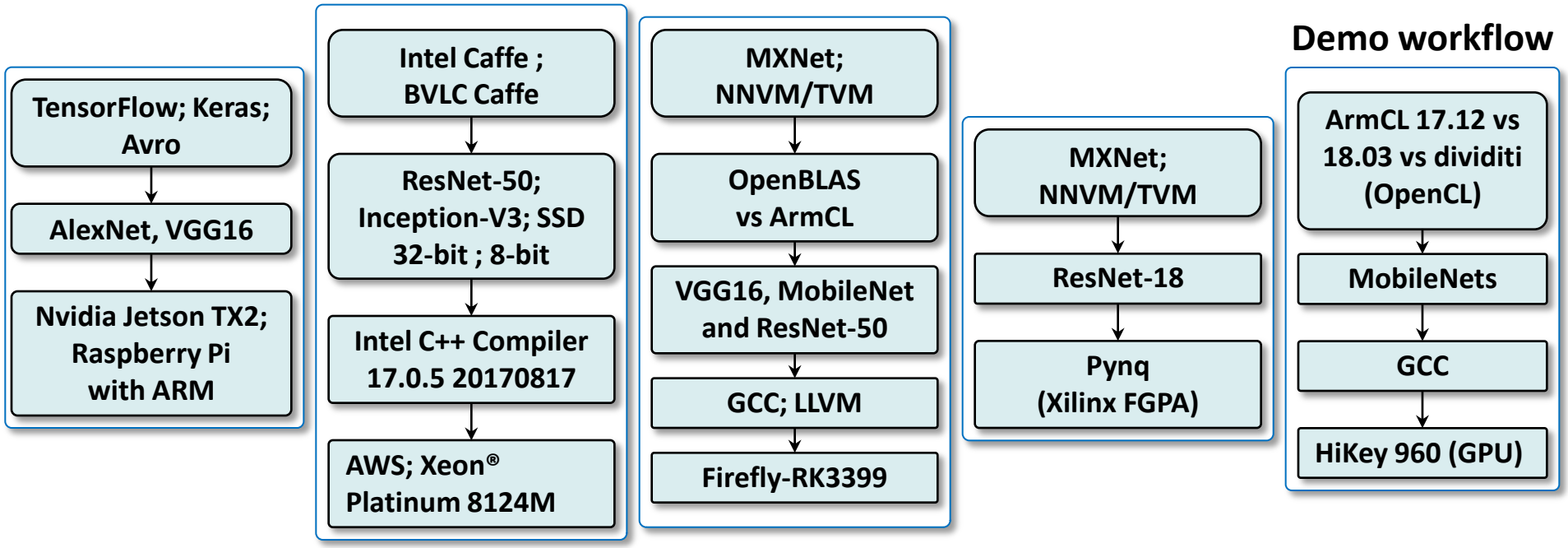
Advisory board suggests algorithms, data sets, models and platforms for competitions.

For a proof-of-concept our advisory board suggested to build a public repository of the most efficient, portable, customizable and reusable **image classification** algorithms in the CK format optimized across diverse models, data sets and devices from IoT to HPC in terms of accuracy, speed, energy, size, complexity and costs.

Long term goal of such repository with reusable artifacts is to help accelerate AI/ML innovation and speed up its adoption by industry!

1st reproducible ReQuEST tournament and workshop at ASPLOS'18

8 intentions to submit and 5 submitted image classification workflows with unified Artifact Appendices

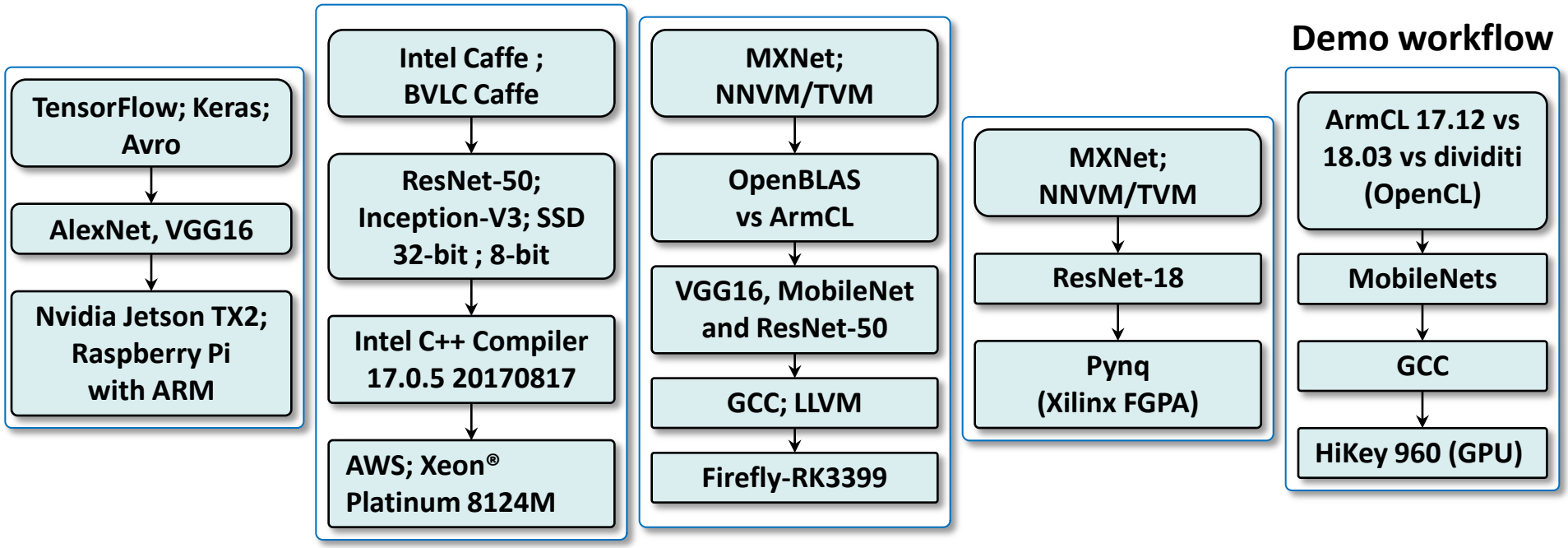


Open evaluation: https://github.com/ctuning/ck-request-asplos18-results via tickets

Functional?					
CK unification					
CK experiments					
CK dashboard					

1st reproducible ReQuEST tournament and workshop at ASPLOS'18

8 intentions to submit and 5 submitted image classification workflows with unified Artifact Appendices



Open evaluation: https://github.com/ctuning/ck-request-asplos18-results via tickets

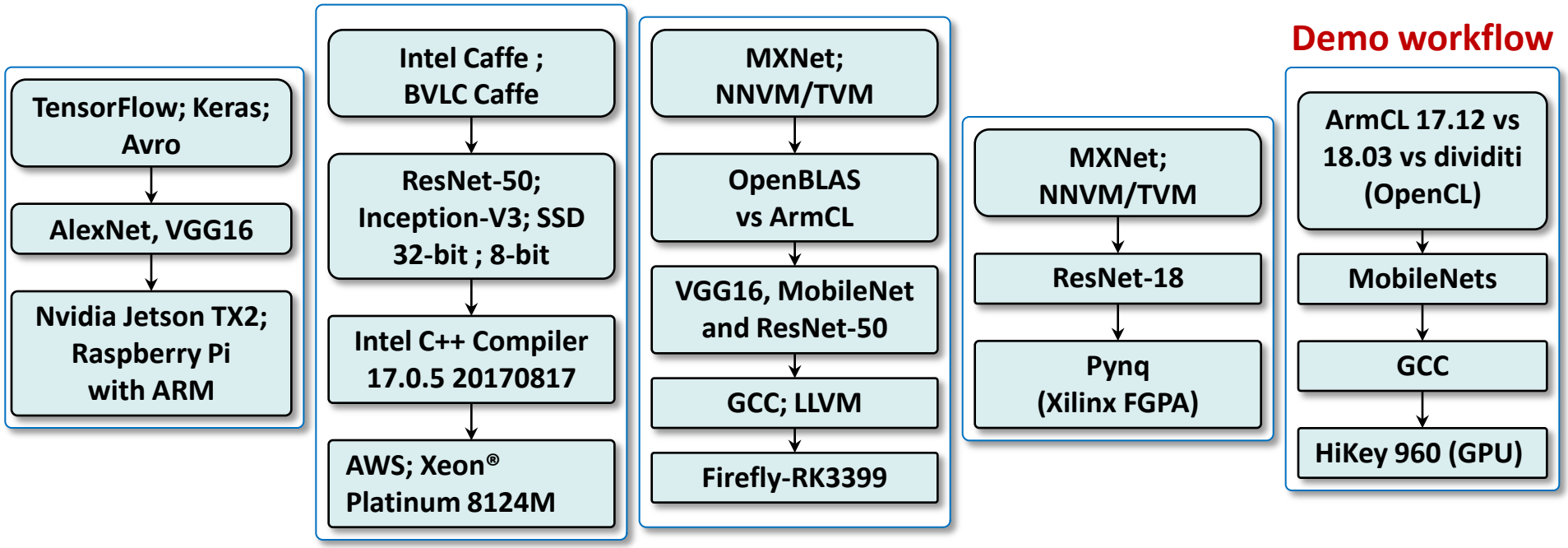
Functional?					
CK unification					
CK experiments					
CK dashboard					

Very time consuming!

2..4 weeks per workflow!

1st reproducible ReQuEST tournament and workshop at ASPLOS'18

8 intentions to submit and 5 submitted image classification workflows with unified Artifact Appendices



Open evaluation: https://github.com/ctuning/ck-request-asplos18-results via tickets

Functional?	✔	✔	✔	✔	✔
CK unification	✔	✔	✔		✔
CK experiments		✔	✔		✔
CK dashboard		✔			✔

Our example: exploring MobileNets design using Arm Compute Library

"**MobileNets**: Efficient Convolutional Neural Networks for Mobile Vision Applications" (Andrew G. Howard et al., 2017, https://arxiv.org/abs/1704.04861):

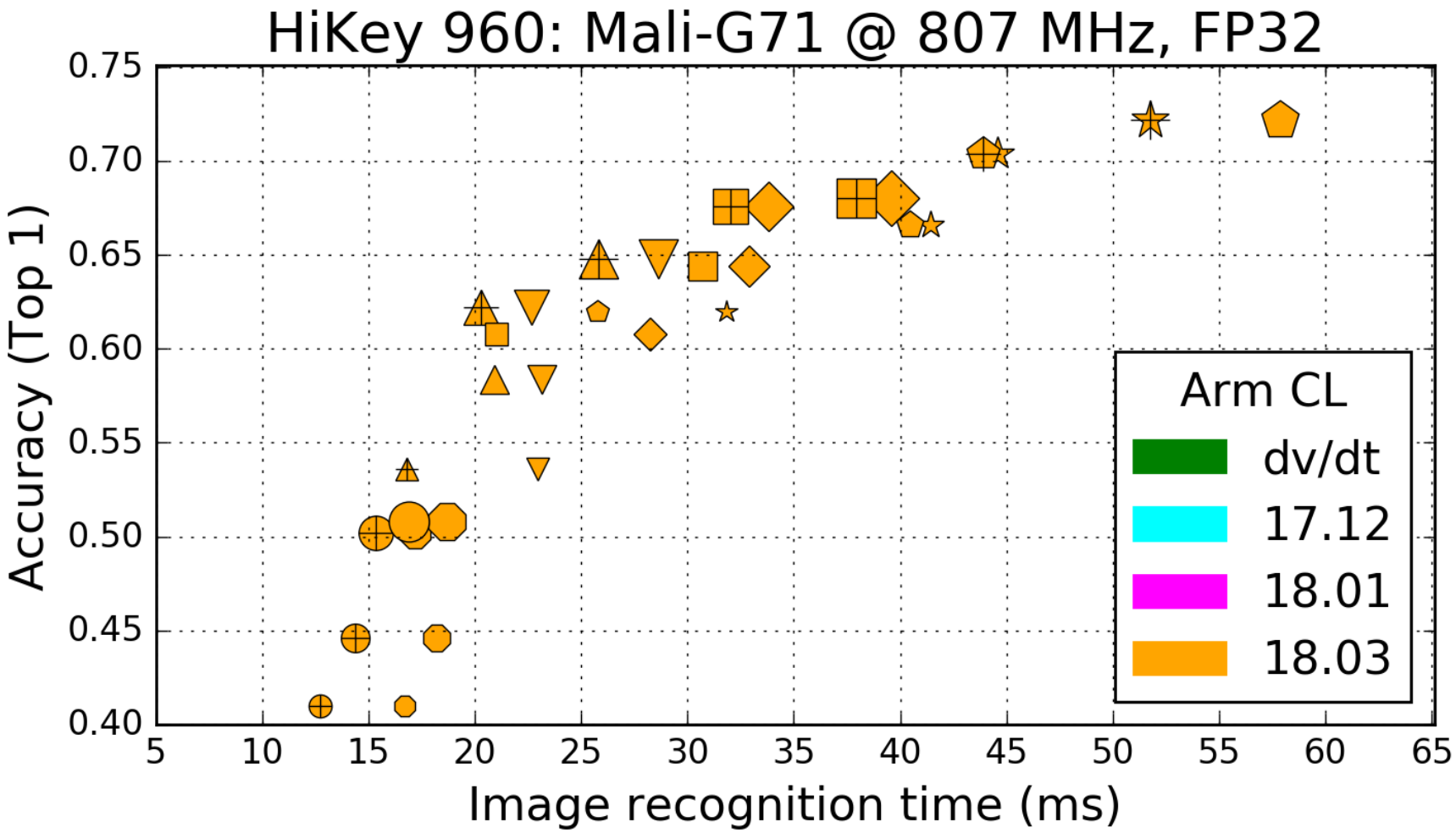
- Parameterised CNN family using depthwise separable convolutions.
- Channel multiplier: 1.00, 0.75, 0.50, 0.25 - marker shape (see below).
- Input image resolution: 224, 192, 160, 128 - marker size.

Arm Compute Library: open-source, optimised for Neon CPUs and Mali GPUs.

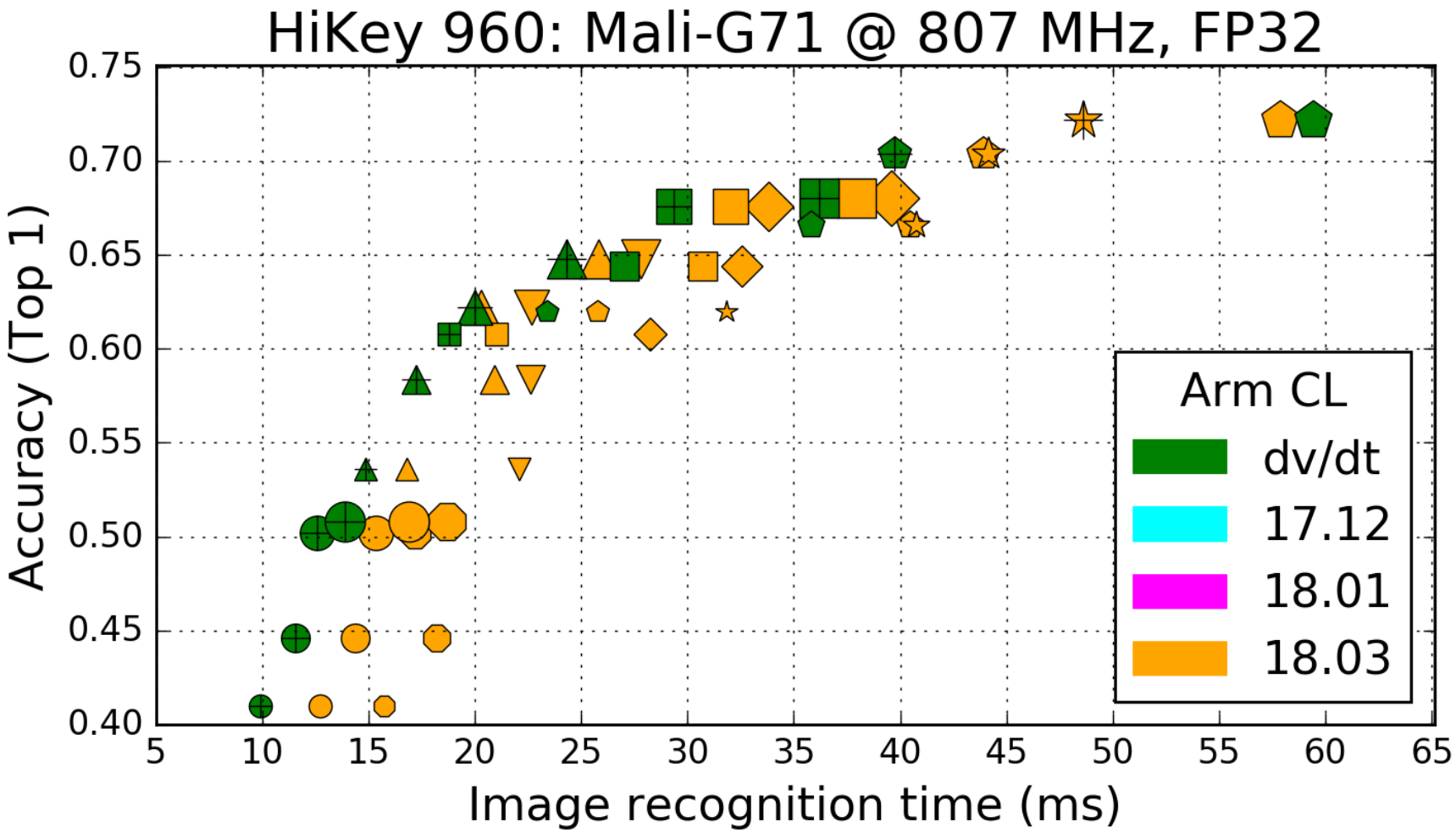
- 2 convolution approaches - marker shape depends on channel multiplier:
  - "Direct": 1.00 - pentagon, 0.75 - square, 0.50 - triangle-up, 0.25 - circle.
  - "Matrix-multiplication" (MM):
    1.00 - star, 0.75 - diamond, 0.50 - triangle-down, 0.25 - octagon.
- 4 library versions - marker colour:
  - "17.12": no opts; "18.01": dividiti's direct+MM opts;
    "18.03": Arm's MM opts; "dv/dt": dividiti's new direct opts.

https://github.com/dividiti/ck-request-asplos18-mobilenets-armcl-opencl

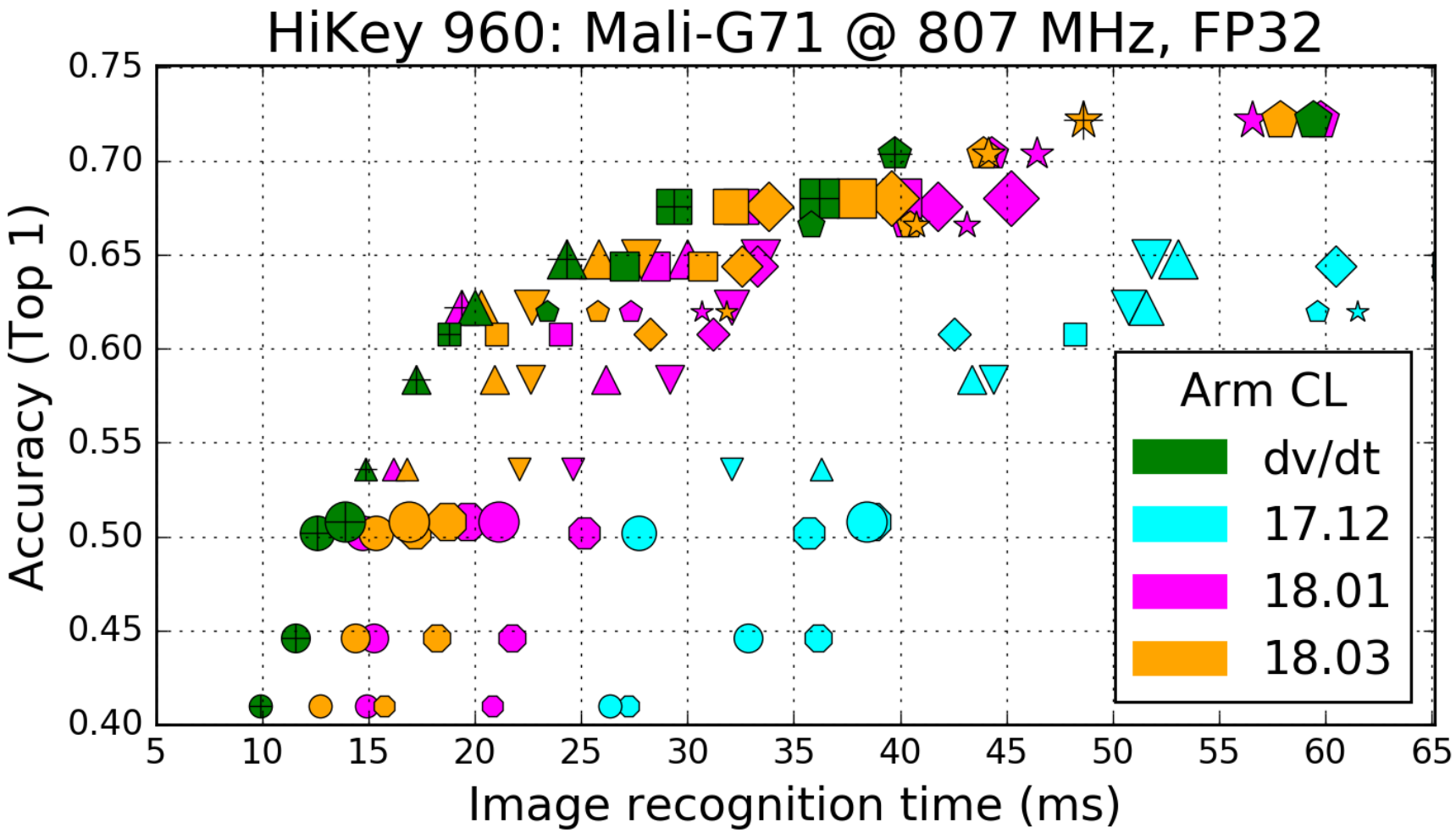
Our example: exploring MobileNets design using Arm Compute Library



Our example: exploring MobileNets design using Arm Compute Library



Our example: exploring MobileNets design using Arm Compute Library



Live scoreboard – continuously updated!

http://cKnowledge.org/request-results

https://github.com/ctuning/ck-request-asplos18-results

Only at the beginning of a long journey - next steps

- Finalize and share all artifacts, workflows and results as “plug&play” CK components (common JSON API and meta description)
- Integrate with ACM Digital Library; **provide open report to the ReQuEST advisory board**
- Continue improving framework and scoreboard (still a long way to go!)
- Gradually expose more design and optimization knobs at all AI/SW/HW levels
- Collaboratively improve models and find missing features
- Enable distributed autotuning and learning
- **Validate results in real systems while sharing more data sets and mispredictions!**
- Prepare next tournaments (likely on distributed training)
- Support validation of experimental results at other events (EMC2, WAX, LPIRC, ASPLOS)

ResCuE-HPC: 1st Workshop
on Reproducible, Customizable
and Portable Workows for HPC

SuperComputing’18

Todd Gamblin, LLNL **Michela Taufer**, U.Delaware
Milos Puzovic, Hartree **Grigori Fursin**, cTuning/dividiti

Participate, collaborate, sponsor …

Building an open repository of "plug&play" AI blocks continuously optimized across diverse data sets, models and platforms from the cloud to edge…

Advisory/industrial board (A-Z)

- **Michaela Blott**, Xilinx
- **Unmesh Bordoloi**, General Motors
- **Ofer Dekel**, Microsoft
- **Maria Girone**, CERN openlab
- **Wayne Graves**, ACM
- **Vinod Grover**, NVIDIA
- **Sumit Gupta**, IBM
- **James Hetherington**, Alan Turing Institute
- **Steve Keckler**, NVIDIA
- **Wei Li**, Intel
- **Colin Osborne**, ARM
- **Andrew Putnam**, Microsoft
- **Boris Shulkin**, Magna
- **Greg Stoner**, AMD
- **Alex Wade**, Chan Zuckerberg Initiative
- **Peng Wu**, Huawei
- **Cliff Young**, Google