tinyML® Foundation

Enabling Ultra-low Power Machine Learning at the Edge

tinyML Summit April 22 - 24, 2024

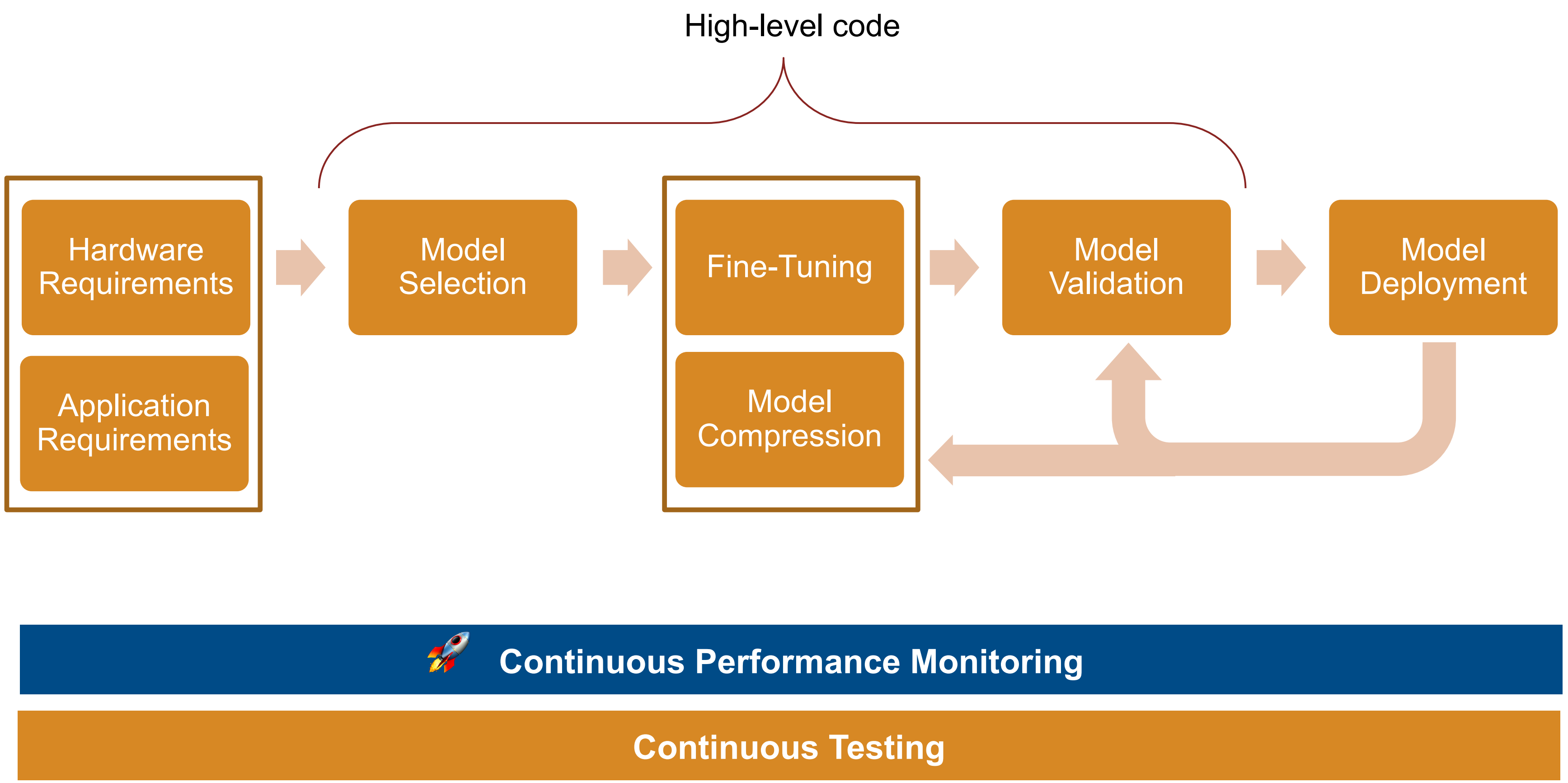
TINY ML

www.tinyML.org

Accelerating Model Optimization on the Edge Through Automated Performance Benchmarking and End-to-End Profiling

Nayara Aguiar, PhD
Performance Engineer
MathWorks

Deploying AI Models: A Tale on Performance



When does performance become a concern in your AI deployment pipeline?

Evaluating performance throughout the development process enables early detection of bottlenecks

Tracking performance makes it easier to pinpoint the source of issues

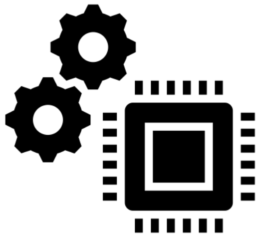
Mitigation of performance issues is less costly with early detection

Quality targets for final product can be met while enhancing development process

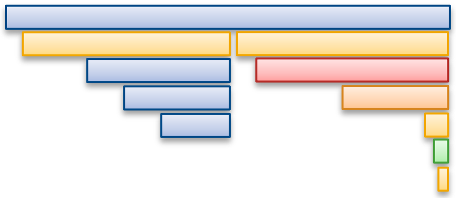
Optimizing performance of the AI deployment pipeline presents multiple challenges

Dependencies	**Multi-step process**		**Cross-team collaboration**	
Definition	**Speed**	**Energy**	**Accuracy**	**Memory**
Standardization	**Tools**	**HW/SW Stack**	**Processes**	

We will focus on our ongoing efforts to enhance the process for performance benchmarking and profiling



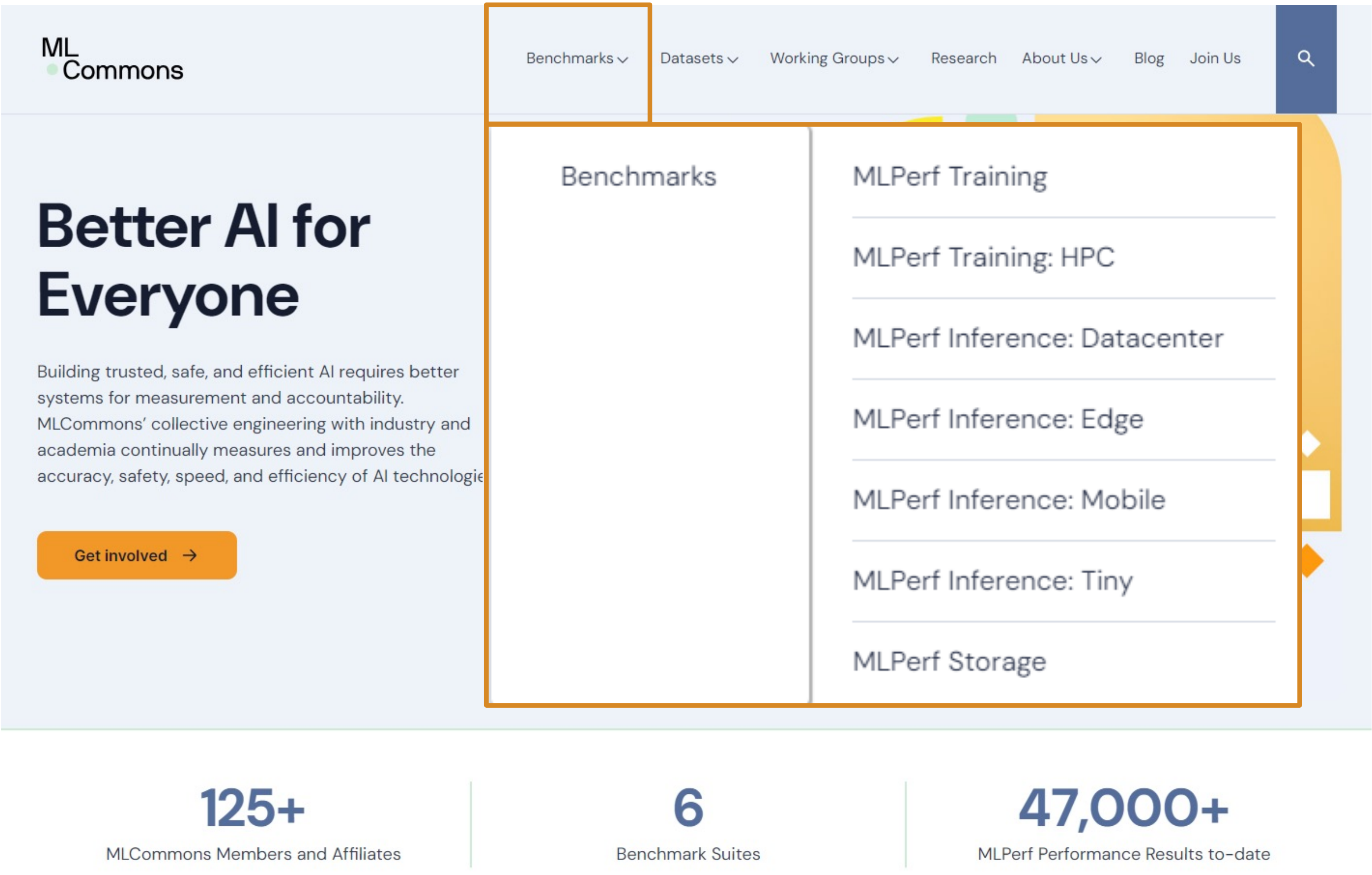
Automating Performance Benchmarking



Enabling End-to-End Performance Profiling

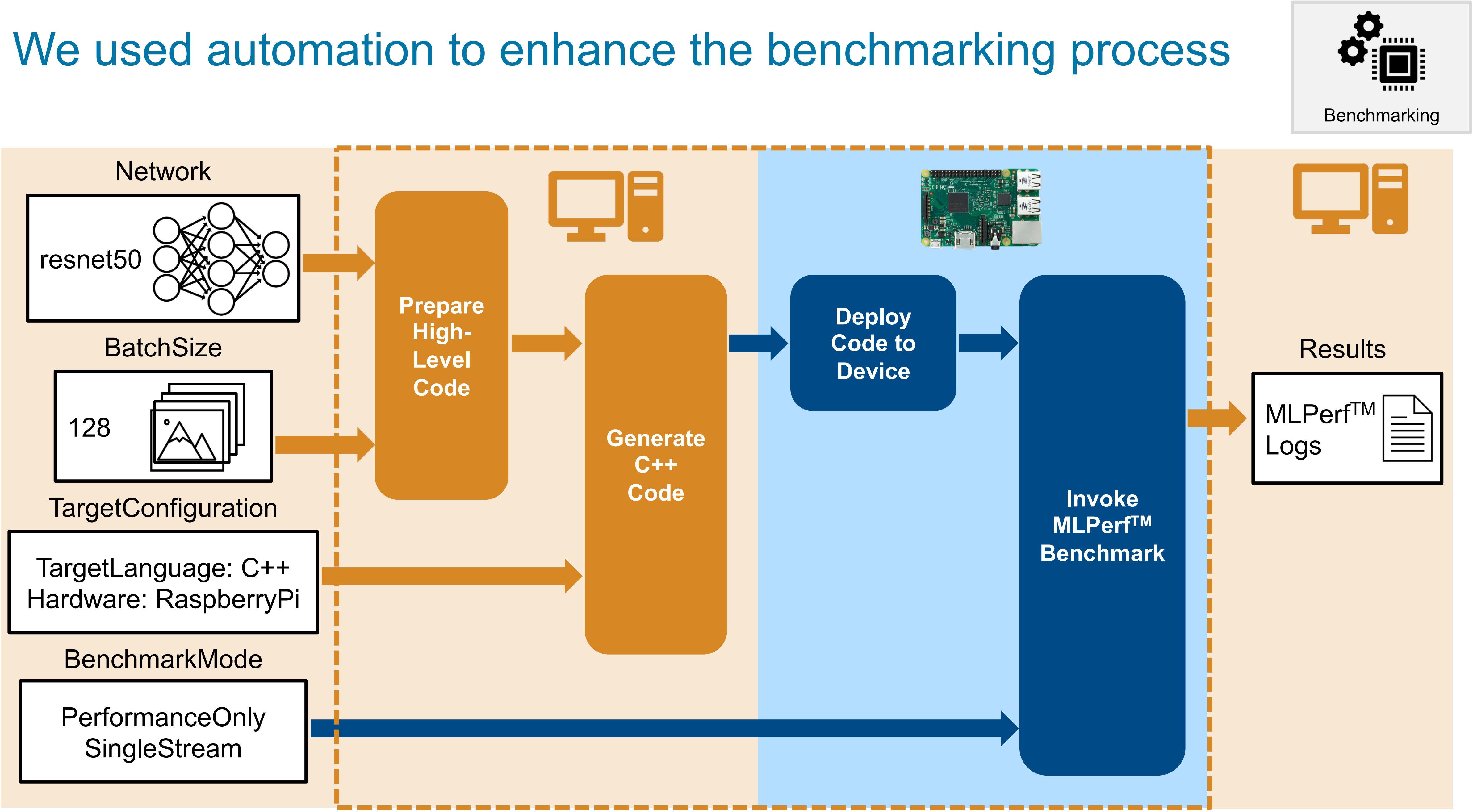
Integrating MLPerf™ to our internal performance benchmarking strategy helped us standardize tooling

Benchmarking



Source: https://mlcommons.org/

We used automation to enhance the benchmarking process



This benchmark can also be wrapped in our testing infrastructure for further automation

Benchmarking

Testing Framework

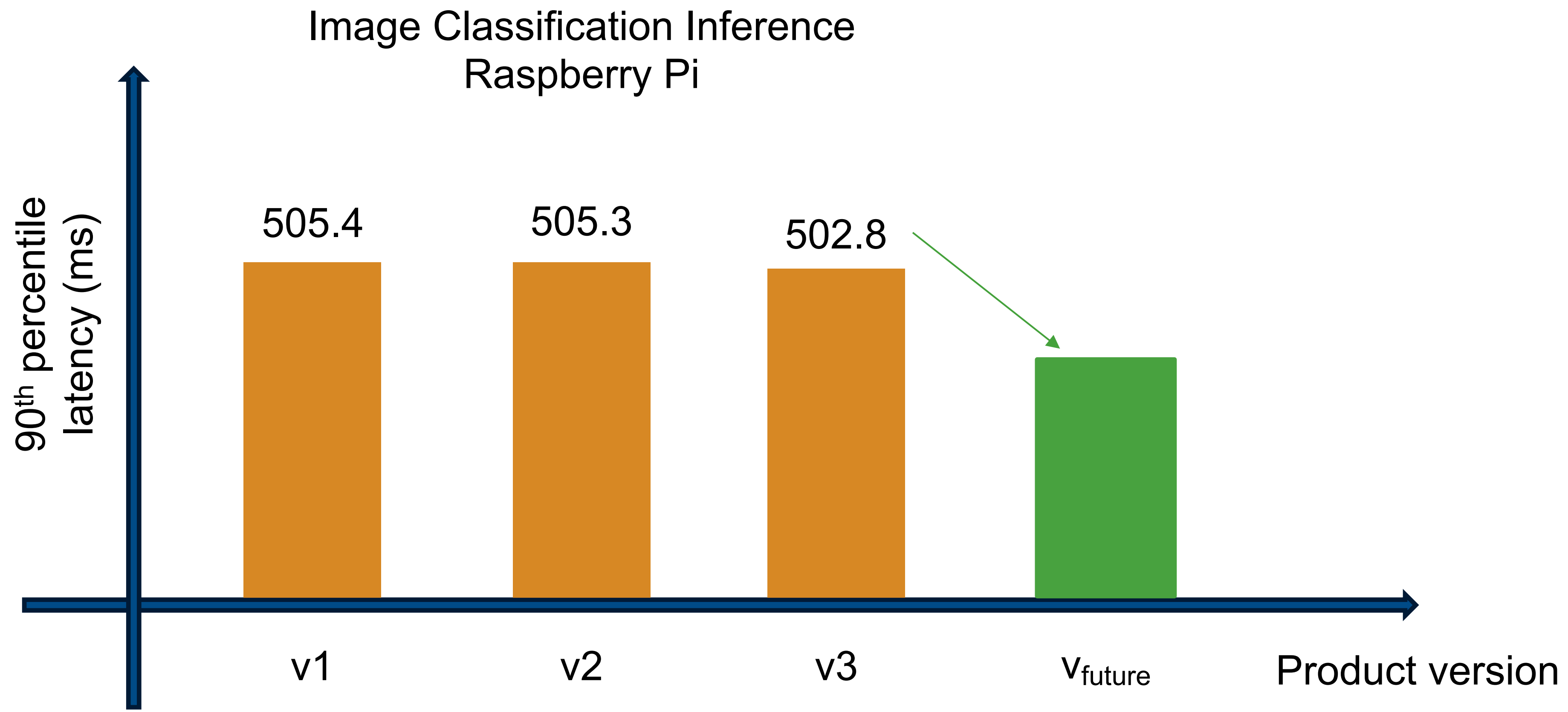
Network = **{networkA, networkB}**
BatchSize = **{64, 128}**
TargetConfiguration = **{configA, configB}**
BenchmarkMode = **{mode1, mode2}**

1. Select pre-trained models:
   - mobilenetv2
   - resnet50
2. Create configurations for Raspberry Pi runs:
   - Using original network (FP32)
   - Using quantized network (INT8)
   - Using network equalized before quantization (INT8)
3. Batch size:
   - 1024
4. Select benchmark mode:
   - AccuracyOnly, SingleStream

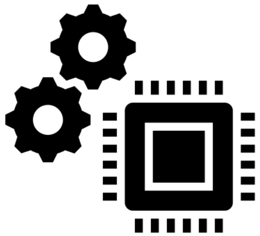
	Original (FP32)	Quantized (INT8)	Equalized (INT8)
mobilenetv2	70.3%	0.2%	60.3%
resnet50	72.2%	69.3%	68.9%

Automated benchmarks allow for performance monitoring … and profiling helps investigating performance bottlenecks

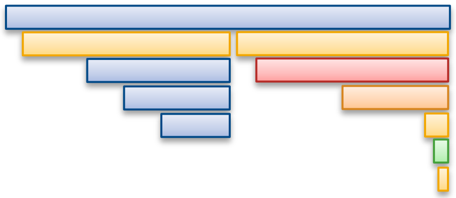
Benchmarking



We will focus on our ongoing efforts to enhance the process for performance benchmarking and profiling

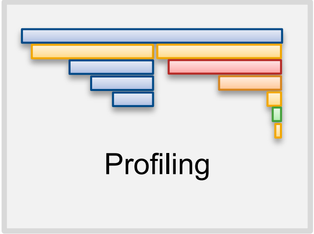


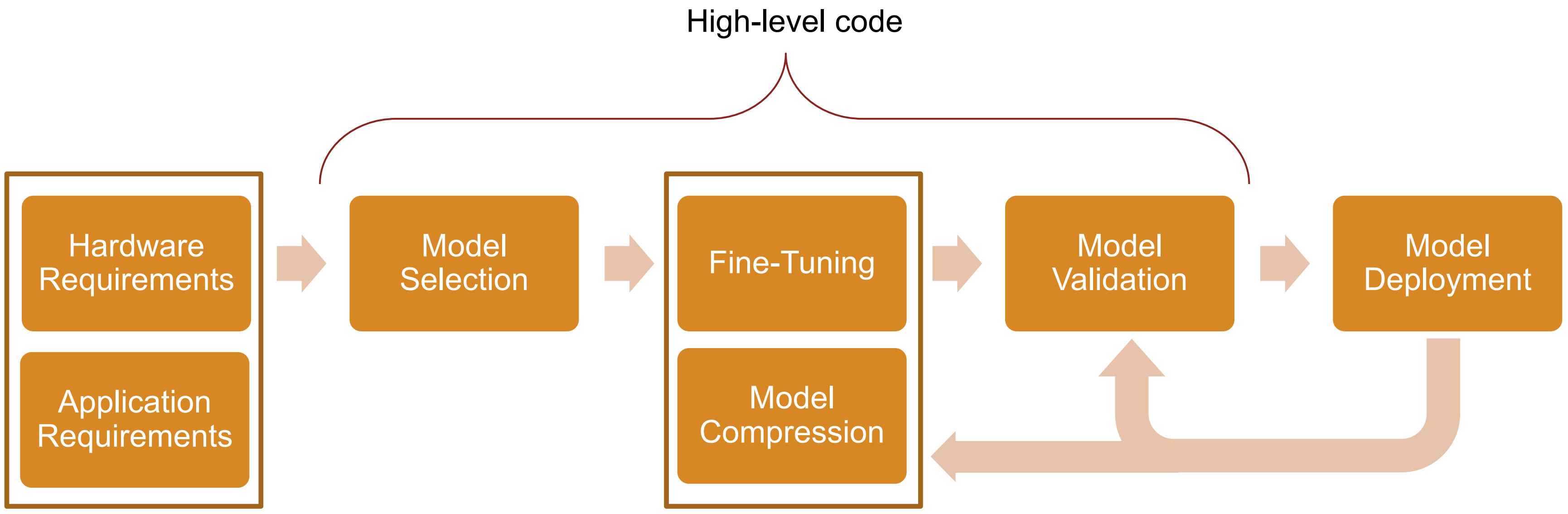
Automating Performance Benchmarking



Enabling End-to-End Performance Profiling

We develop tools that cover the entire AI deployment pipeline



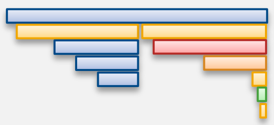


Continuous Performance Monitoring

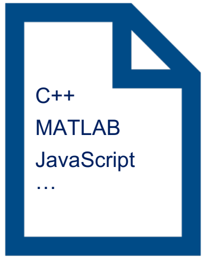
Continuous Testing

To facilitate end-to-end performance investigations, we developed the Unified Timeline

The *Unified Timeline* facilitates performance analysis across the AI deployment pipeline

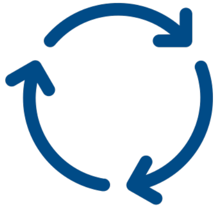


Profiling



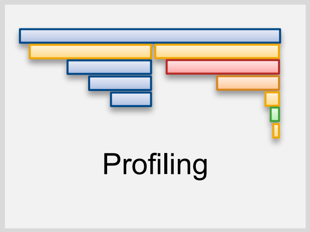
Generic

Easy to use

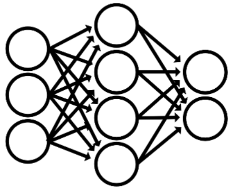


Continuously enhanced

We can use our in-house profiling tool to investigate performance bottlenecks



Network



BatchSize

Prepare High-Level Code

Inference for image classification

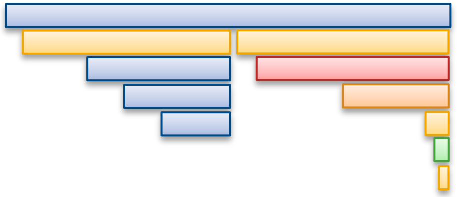
```
tic
output = net.predict(img);
toc
```

Total time

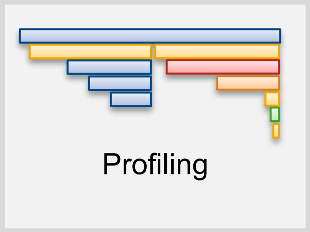
~ 700 ms

```
profile on
output = net.predict(img);
profile off
```

Time breakdown

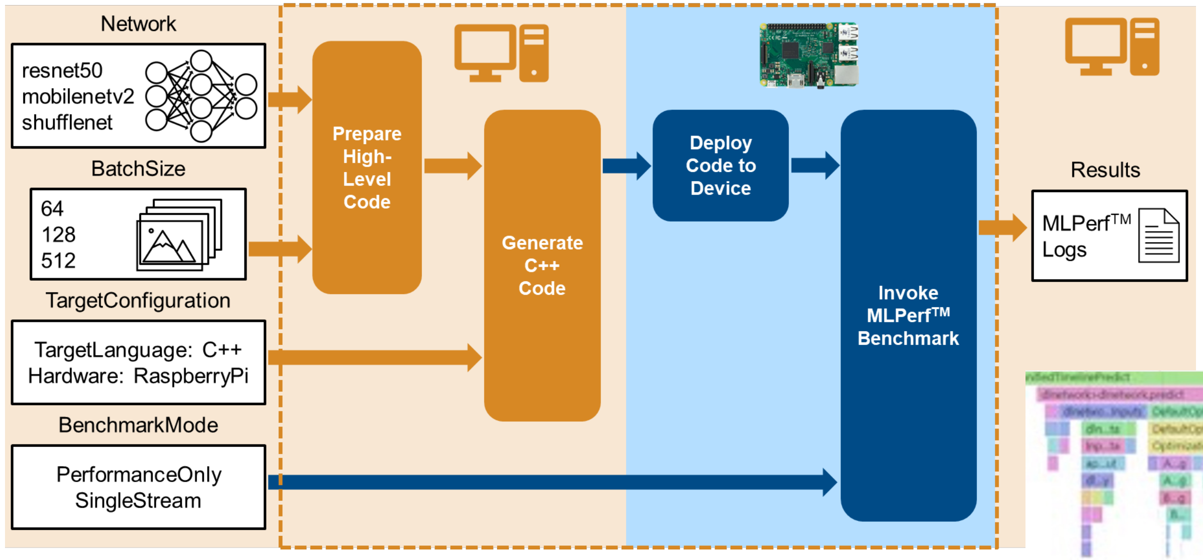


We can visualize the timeline for this code in a browser and zoom in stacks of interest



Inference for image classification

Make performance a core aspect for AI applications on the edge



- Automate performance benchmarking
- Explore performance profiling tools

… accelerate model optimization on the edge

Copyright Notice