Hewlett Packard Enterprise

Powersched

A HPC System Power and Energy Management Framework

Marcel Marquardt

May 10, 2023

Introduction

Motivation

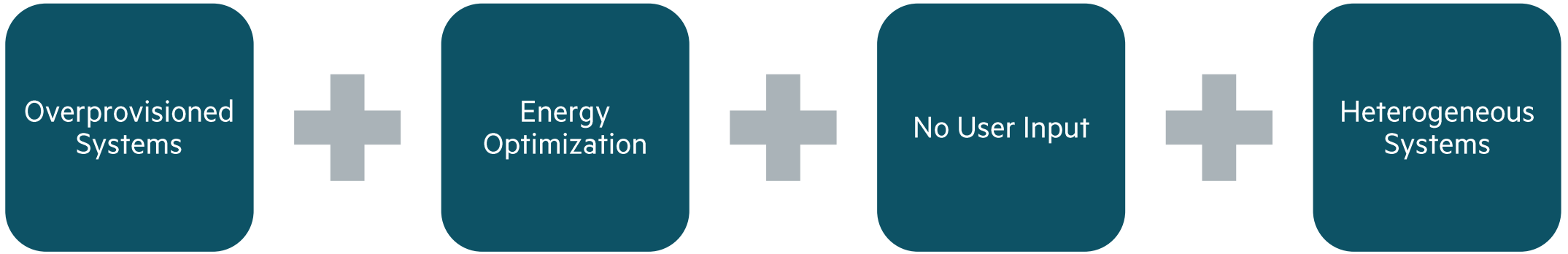
TOP500 The List.

Rank	System	Cores	Rmax (PFlop/s)	Rpeak (PFlop/s)	Power (kW)
1	**Frontier** - HPE Cray EX235a, AMD Optimized 3rd Generation EPYC 64C 2GHz, AMD Instinct MI250X, Slingshot-11, HPE DOE/SC/Oak Ridge National Laboratory United States	8,730,112	1,102.00	1,685.65	21,100
30	**Hawk** - Apollo 9000, AMD EPYC 7742 64C 2.25GHz, Mellanox HDR Infiniband, HPE HLRS - Höchstleistungsrechenzentrum Stuttgart Germany	698,880	19.33	25.16	3,906

Source: https://www.top500.org/lists/top500/2022/11/

Source: https://www.hlrs.de/solutions/systems/hpe-apollo-hawk

Problem Statement



Related Work

Existing products

Lenovo EAR

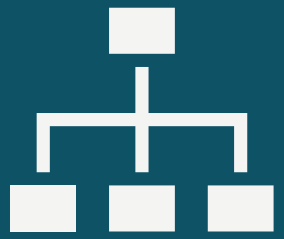
BULL DPO

Intel GEOPM

No holistic system view or management

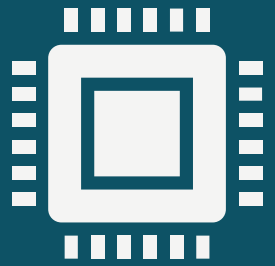
Powersched Framework

Requirements



Overprovisioned Systems

- Holistic view on cluster
- Shift budget between nodes
- Scheduler Integration
- Distribution policies



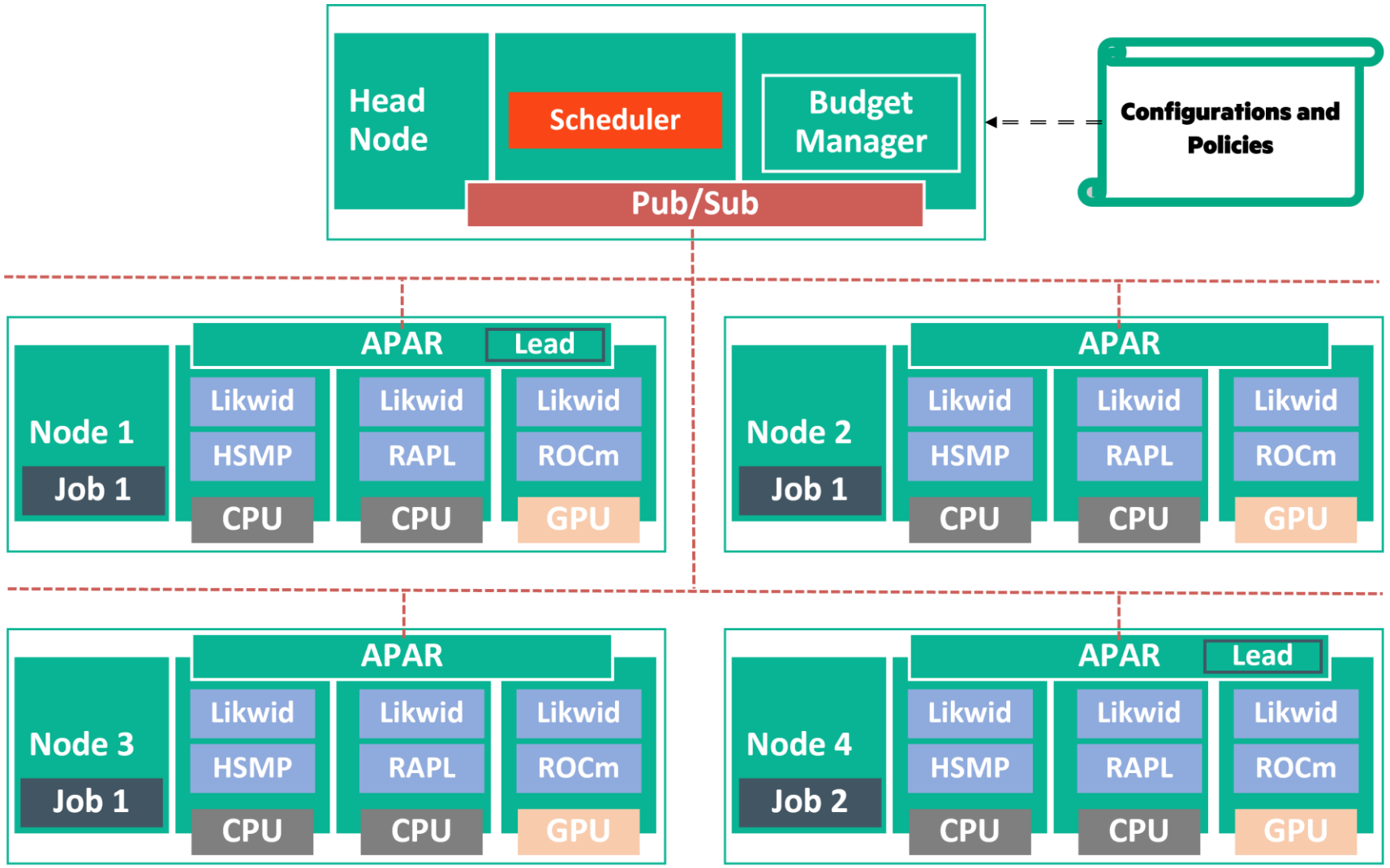
Heterogeneous Systems

- Multiple components in modern systems
- Support diverse landscape

Energy Optimization

- Use system metrics only
- Assign power level with best energy efficiency
- React to different app phases

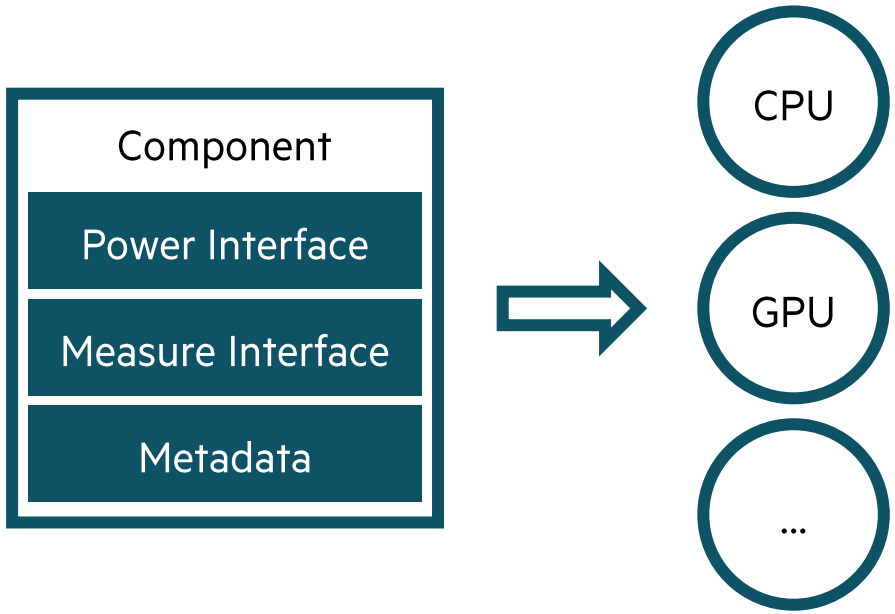
Architecture



Components

Too many components to support individually

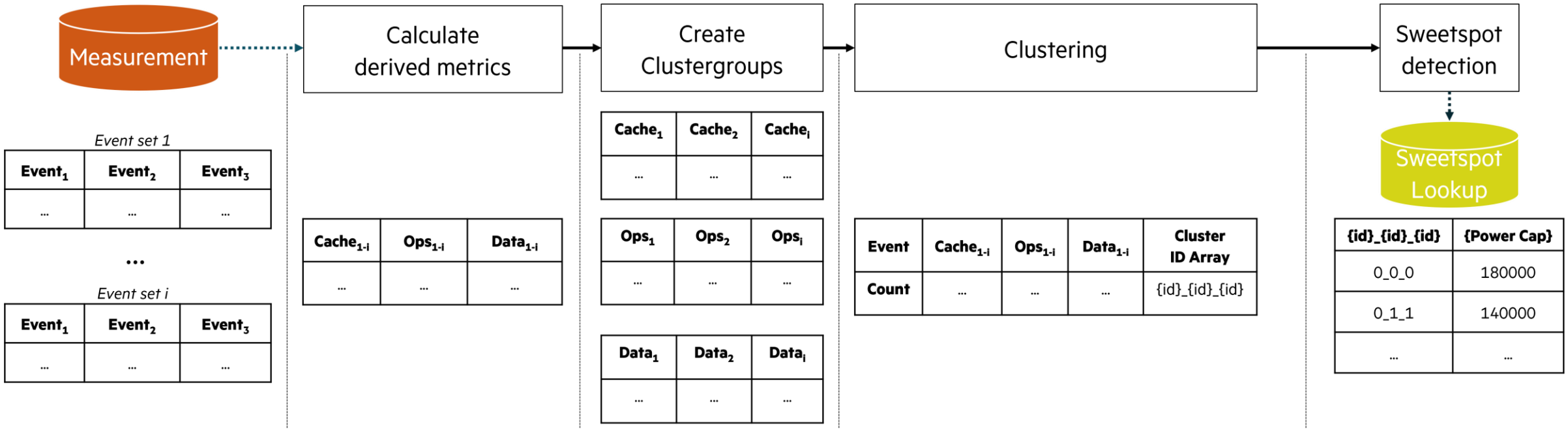
→ *can be configured by admin*



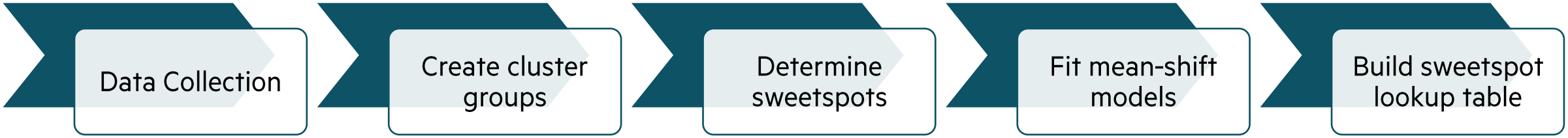

```
#----------------------------------------
#  AMD Zen2
#----------------------------------------
parameters:
  socket: int

power_limits:
  min: 100 W
  max: 200 W
  step: 20 W

power_interface:
  type: hsmp
  socket: !param socket

measure_interface:
  type: likwid
  cpu_ids: 0-255
  duration: 120s
  groups:
    - RETIRED_INSTRUCTIONS: PMC1
      RETIRED_SSE_AVX_FLOPS_ALL: PMC2
      RETIRED_BRANCH_INSTR: PMC3
      RETIRED_MISP_BRANCH_INSTR: PMC4
      RAPL_PKG_ENERGY: PWR1

    ...
```

Energy Optimization

Algorithm


Measurement
Calculate derived metrics
Create Clustergroups
Clustering
Sweetspot detection
Sweetspot Lookup
Event set 1
Event set i
Event1 Event2 Event3
Cache1-i Ops1-i Data1-i
Cache1 Cache2 Cachei
Ops1 Ops2 Opsi
Data1 Data2 Datai
Event Cache1-i Ops1-i Data1-i Cluster ID Array
Count
{id}_{id}_{id}
{Power Cap}
0_0_0 180000
0_1_1 140000

Training



Determine sweetspots

Compare instruction per watt for each application phase across all power levels

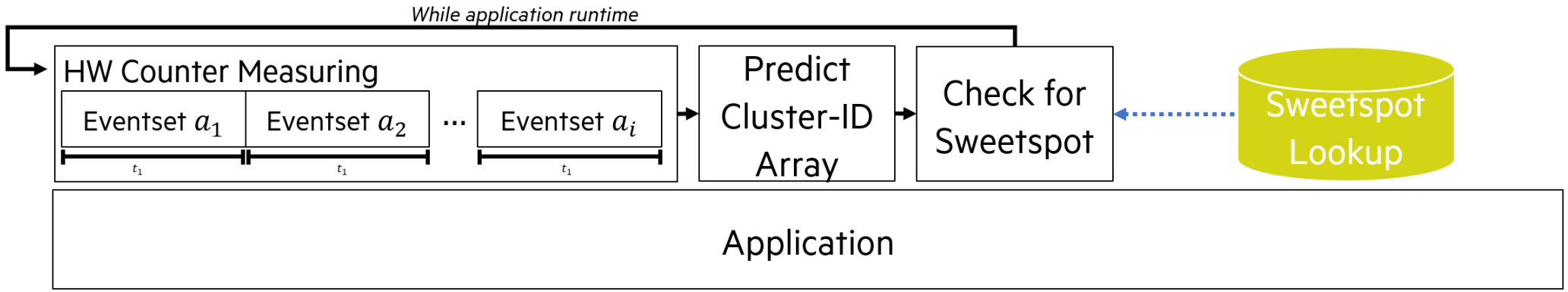
Fit mean-shift models

For each power level and cluster group separately

Build sweetspot table

Find most common sweetspot across measurements with same fingerprint

Inference


While application runtime
HW Counter Measuring
Eventset a_1
Eventset a_2
...
Eventset a_i
t_1
t_1
t_1
Predict Cluster-ID Array
Check for Sweetspot
Sweetspot Lookup
Application

Results


Introduction
Related Work
Powersched Framework
Energy Optimization
Results
Conclusion

Evaluation

Benchmark	with DPC		without DPC			
	kWh	runtime in Sec.	kWh	runtime in Sec.	energy savings in %	runtime extension in %
CG	0,8149	4043	0,8875	3982	8,18	1,53
IS	0,5415	2943	0,6392	2906	15,28	1,27
Halo3d	0,7558	4189	0,9269	4177	18,46	0,28
Halo3d-26	0,8785	4831	1,0504	4733	16,36	2,07
Swfft	0,4193	2267	0,4855	2199	13,63	3,09

Summary

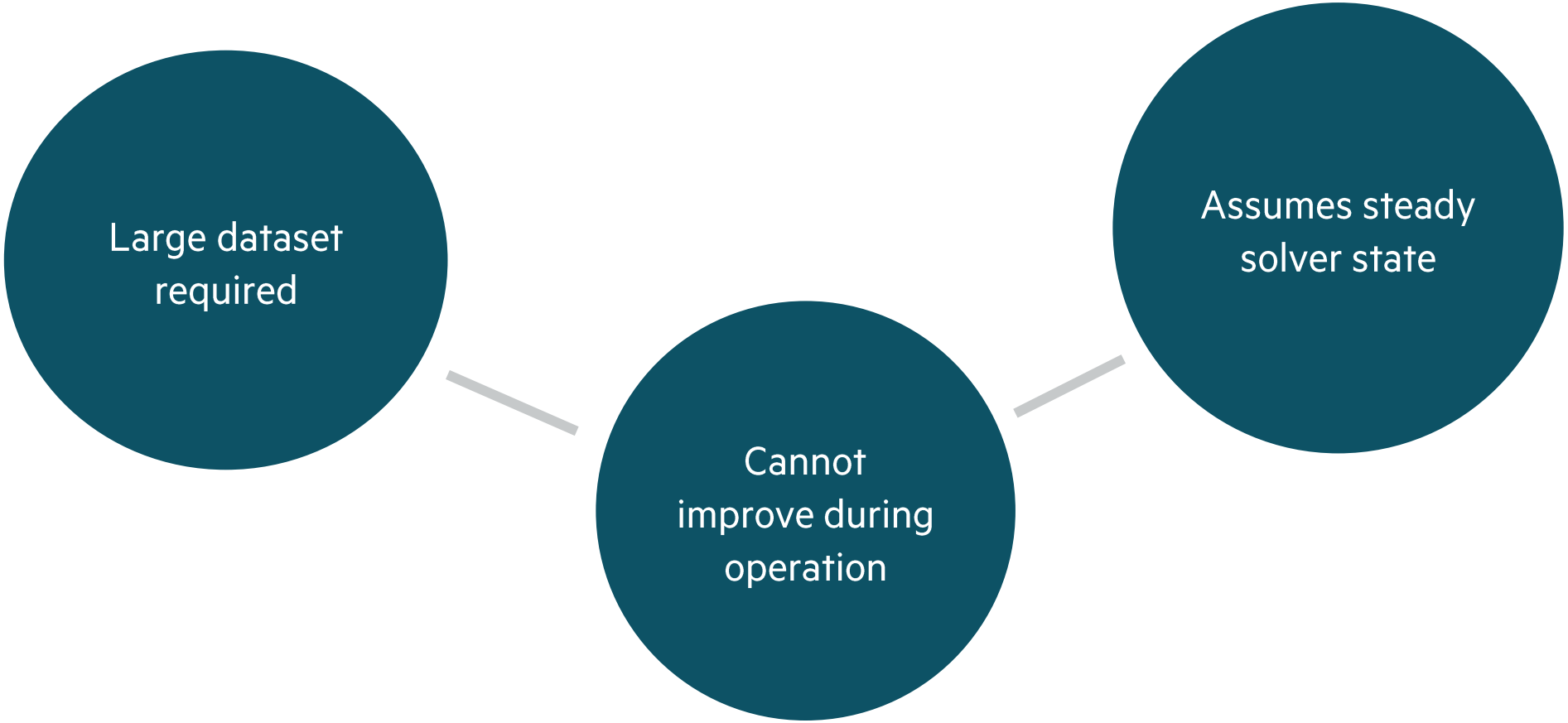
Average Energy Savings

14%

Average Runtime Extension

2%

Bottlenecks



Conclusion


Introduction
Related Work
Powersched Framework
Energy Optimization
Results
Conclusion

Summary & Outlook

Current

Extensible framework that supports multiple user-defined components

Clustering approach that can **save up to 14% energy** with only 2% runtime increase

Research

Other optimization approaches

Proper heterogeneous optimization

Reduce measurement events

Thank you!

Marcel Marquardt
marcel.marquardt@hpe.com

Christian Simmendinger
christian.simmendinger@hpe.com

Tobias Schiffmann
tobias.schiffmann@hpe.com

Jan Mäder
jan.maeder@hpe.com

Torsten Wilde
wilde@hpe.com