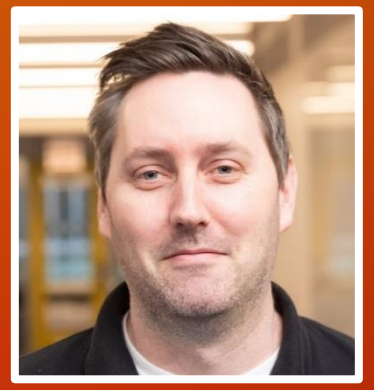
SERVERLESS HPC: CHALLENGES, OPPORTUNITIES, AND FUTURE PROSPECTS FOR ACCELERATED CLOUD COMPUTING



Kyle Chard
University of Chicago

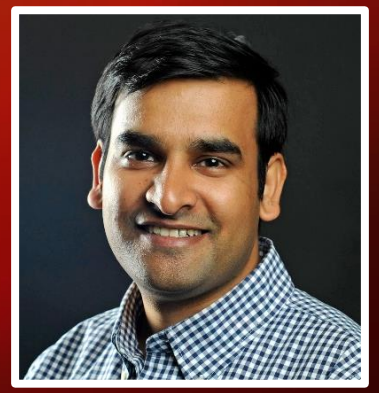
Ian Foster
Argonne National Laboratory

Moderator

Marcin Copik
ETH Zurich

Torsten Hoefler
ETH Zurich

Satoshi Matsuoka
Riken-CCS

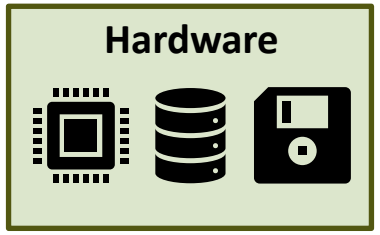


Devesh Tiwari
Northeastern University

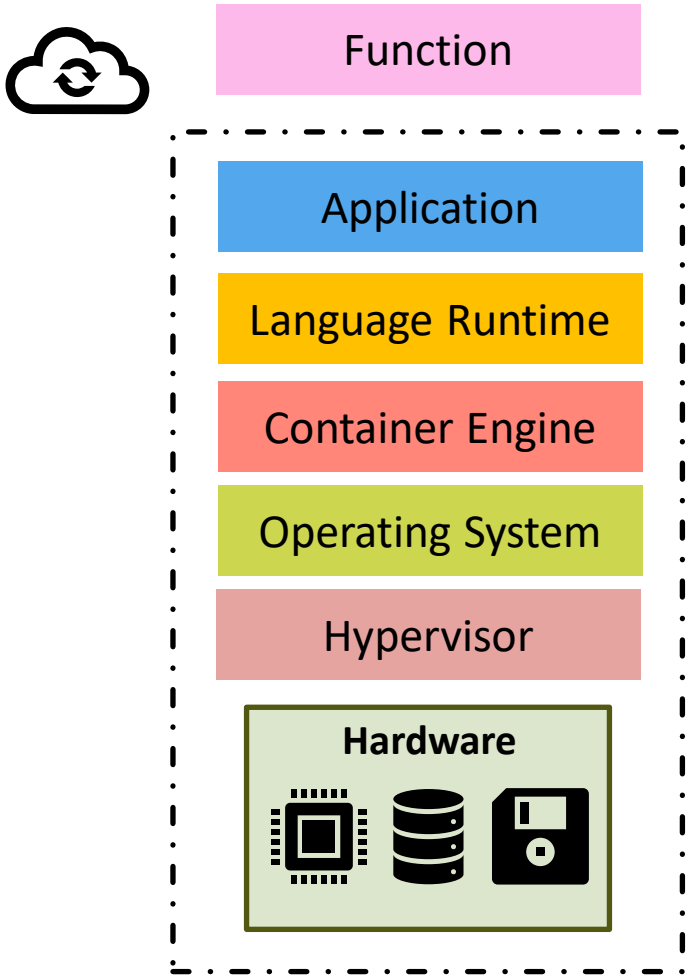
SC24
Atlanta, GA | hpc creates.

Does Serverless Have Servers?

Does Serverless Have Servers?

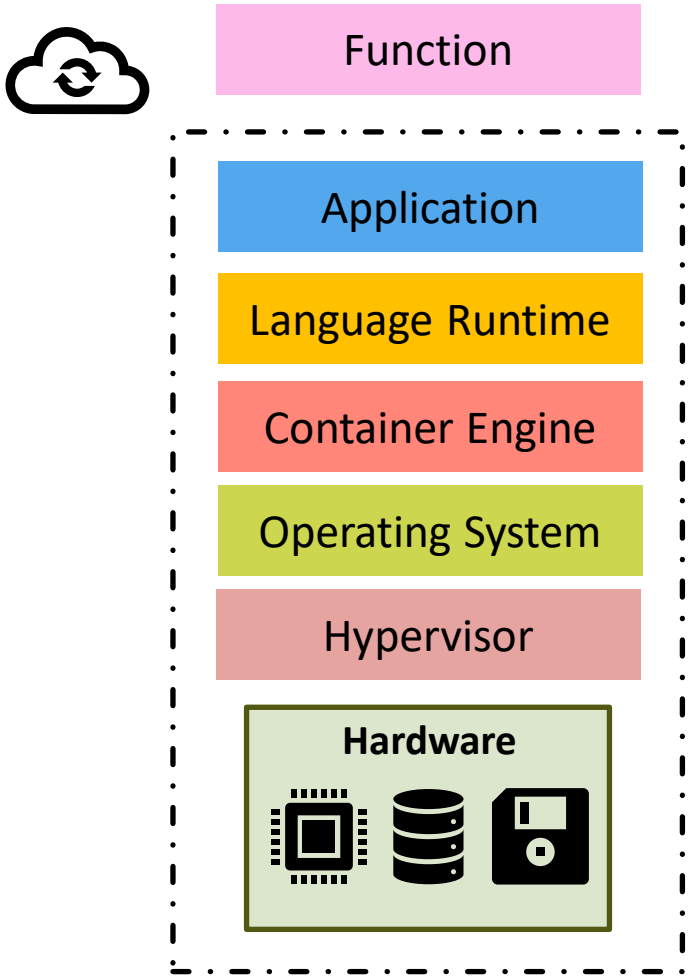


Does Serverless Have Servers?

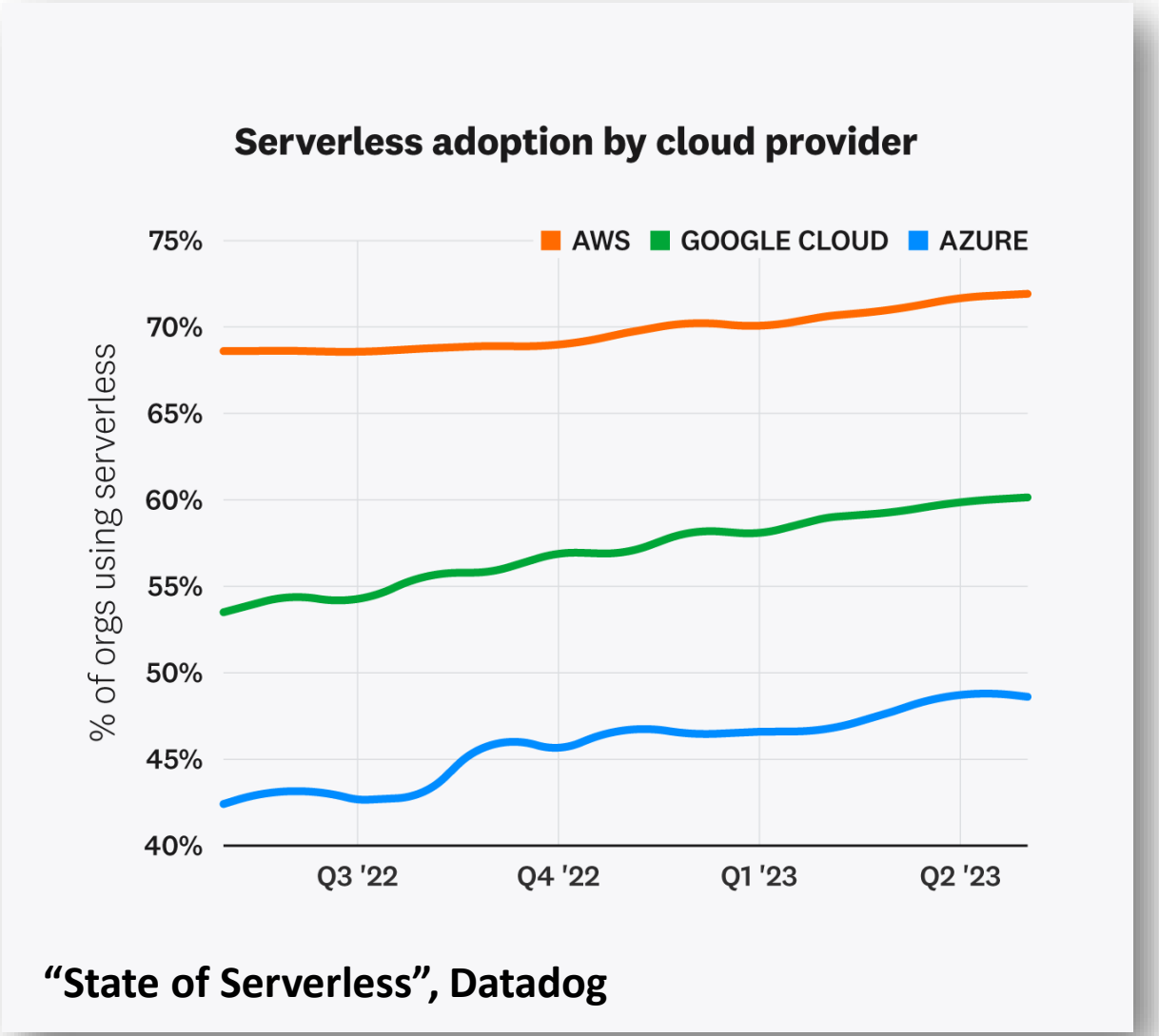


Function-as-a-Service

Does Serverless Have Servers?



Function-as-a-Service



"State of Serverless", Datadog

Humble Beginnings

Introducing AWS Lambda

Posted On: Nov 13, 2014

AWS Lambda is a compute service that runs your code in response to events and automatically manages the compute resources for you, making it easy to build applications that respond quickly to new information. AWS Lambda starts running your code within milliseconds of an event such as an image upload, in-app activity, website click, or output from a connected device. You can also use AWS Lambda to create new back-end services where compute resources are automatically triggered based on custom requests. With AWS Lambda you pay only for the requests served and the compute time required to run your code. Billing is metered in increments of 100 milliseconds, making it cost-effective and easy to scale automatically from a few requests per day to thousands per second.

AWS Lambda is available in Preview. Learn more at http://aws.amazon.com/lambda.

Humble Beginnings

Introducing AWS Lambda

Posted On: Nov 13, 2014

AWS Lambda is a compute service that runs your code in response to events and automatically manages the compute resources for you, making it easy to build applications that respond quickly to new information. AWS Lambda starts running your code within milliseconds of an event such as an image upload, in-app activity, website click, or output from a connected device. You can also use AWS Lambda to create new back-end services where compute resources are automatically triggered based on custom requests. With AWS Lambda you pay only for the requests served and the compute time required to run your code. Billing is metered in increments of 100 milliseconds, making it cost-effective and easy to scale automatically from a few requests per day to thousands per second.

AWS Lambda is available in Preview. Learn more at http://aws.amazon.com/lambda.

Humble Beginnings

“Toy”
Serverless

Humble Beginnings

“Toy”
Serverless

Functions run in single-tenant VMs

Humble Beginnings

“Toy”
Serverless


Functions run in single-tenant VMs


One language: Node

Humble Beginnings

“Toy”
Serverless


Functions run in single-tenant VMs


One language: Node


Deployment as zip files <= 250 MB

Humble Beginnings

“Toy”
Serverless


Functions run in single-tenant VMs


One language: Node


Deployment as zip files <= 250 MB


1 vCPU, up to 1 GB memory

Humble Beginnings

“Toy”
Serverless


Functions run in single-tenant VMs


One language: Node

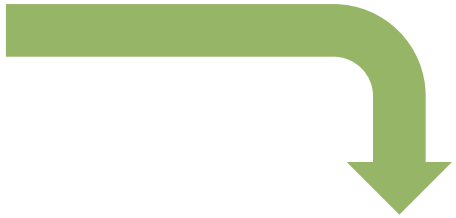

Deployment as zip files <= 250 MB


1 vCPU, up to 1 GB memory


Simple HTTP triggers

Serverless Functions Grew Larger and More Powerful

"Toy" Serverless

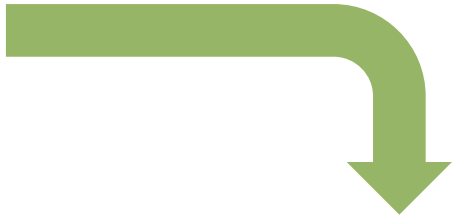


General-Purpose Serverless

- Functions run in single-tenant VMs
- One language: Node
- Deployment as zip files <= 250 MB
- 1 vCPU, up to 1 GB memory
- Simple HTTP triggers

Serverless Functions Grew Larger and More Powerful

"Toy" Serverless



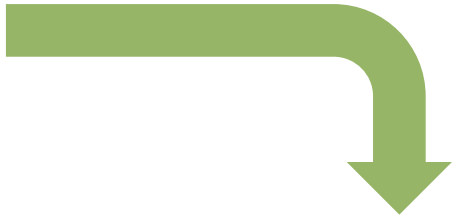
General-Purpose Serverless

- Functions run in single-tenant VMs
- One language: Node
- Deployment as zip files <= 250 MB
- 1 vCPU, up to 1 GB memory
- Simple HTTP triggers

- Dedicated sandboxes and microVMs

Serverless Functions Grew Larger and More Powerful

"Toy" Serverless



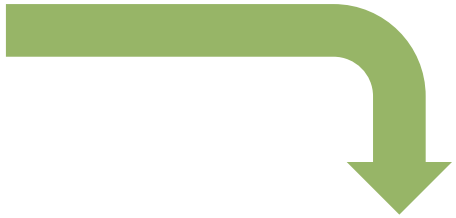
General-Purpose Serverless

- ⚠ Functions run in single-tenant VMs
- ⚠ One language: Node
- ⚠ Deployment as zip files <= 250 MB
- ⚠ 1 vCPU, up to 1 GB memory
- ⚠ Simple HTTP triggers

- Dedicated sandboxes and microVMs
- Many languages, including compiled

Serverless Functions Grew Larger and More Powerful

"Toy" Serverless



General-Purpose Serverless

- ⚠ Functions run in single-tenant VMs
- ⚠ One language: Node
- ⚠ Deployment as zip files <= 250 MB
- ⚠ 1 vCPU, up to 1 GB memory
- ⚠ Simple HTTP triggers

- ⏩ Dedicated sandboxes and microVMs
- ⏩ Many languages, including compiled
- ⏩ Containers & snapshots

Serverless Functions Grew Larger and More Powerful

"Toy" Serverless

General-Purpose Serverless

AWS Lambda: up to 60,000 vCPUs

- ⚠ Functions run in single-tenant VMs
- ⚠ One language: Node
- ⚠ Deployment as zip files <= 250 MB
- ⚠ 1 vCPU, up to 1 GB memory
- ⚠ Simple HTTP triggers

- Dedicated sandboxes and microVMs
- Many languages, including compiled
- Containers & snapshots
- Multi-core functions

Serverless Functions Grew Larger and More Powerful

"Toy" Serverless

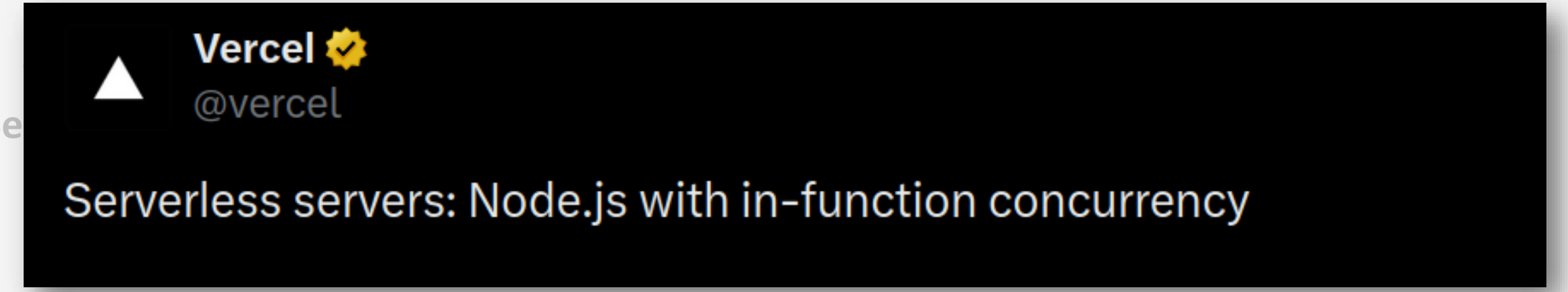
General-Purpose Serverless

AWS Lambda: up to 60,000 vCPUs

- ⚠ Functions run in single-tenant VMs
- ⚠ One language: Node
- ⚠ Deployment as zip files <= 250 MB
- ⚠ 1 vCPU, up to 1 GB memory
- ⚠ Simple HTTP triggers

- Dedicated sandboxes and microVMs
- Many languages, including compiled
- Containers & snapshots
- Multi-core functions
- Serverless workflows

Serverless Functions Grew Larger

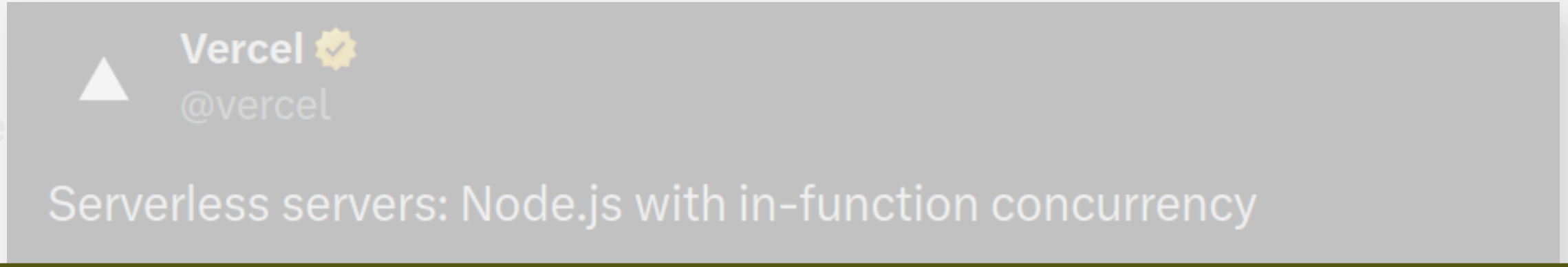


Serverless

- Functions run in single-tenant VMs
- One language: Node
- Functions deployed as zip files <= 250 MB
- 1 vCPU, up to 1 GB memory
- Simple HTTP triggers

- Dedicated sandboxes and microVMs
- Many languages, including compiled
- Container deployment
- Multi-core functions
- Serverless workflows

Serverless Functions Grew Larger



Functions with multi-threading and concurrency are a novelty in 2024.

Cloud still has a lot to learn from HPC!

- One language: Node
- Functions deployed as zip files <= 250 MB
- 1 vCPU, up to 1 GB memory
- Simple HTTP triggers

- Many languages, including compiled
- Container deployment
- Multi-core functions
- Serverless workflows

Serverless was never designed for HPC

AWS Lambda turns 10: A rare look at the doc that started it

November 14, 2024 • 5460 words

Serverless was never designed for HPC

31. How does Lambda support parallel processing?

Developers can run multiple applications and/or ***multiple copies of the same application simultaneously****. They can also access Lambda APIs programmatically from within applications, using the AWS client SDK, which allows them to delegate and orchestrate work by running other applications.*

Serverless was never designed for HPC

31. How does Lambda support parallel processing?

Developers can run multiple applications and/or ***multiple copies of the same application simultaneously****. They can also access Lambda APIs programmatically from within applications, using the AWS client SDK, which allows them to delegate and orchestrate work by running other applications.*

Burst Launches, Colocation Policies
Bulk Synchronous Parallel Model
Communicators, Message Passing, Collectives
…

Serverless was never designed for HPC

31. How does Lambda support parallel processing?

*Developers can run multiple applications and/or **multiple copies of the same application simultaneously**. They can also access Lambda APIs programmatically from within applications, using the AWS client SDK, which allows them to delegate and orchestrate work by running other applications.*

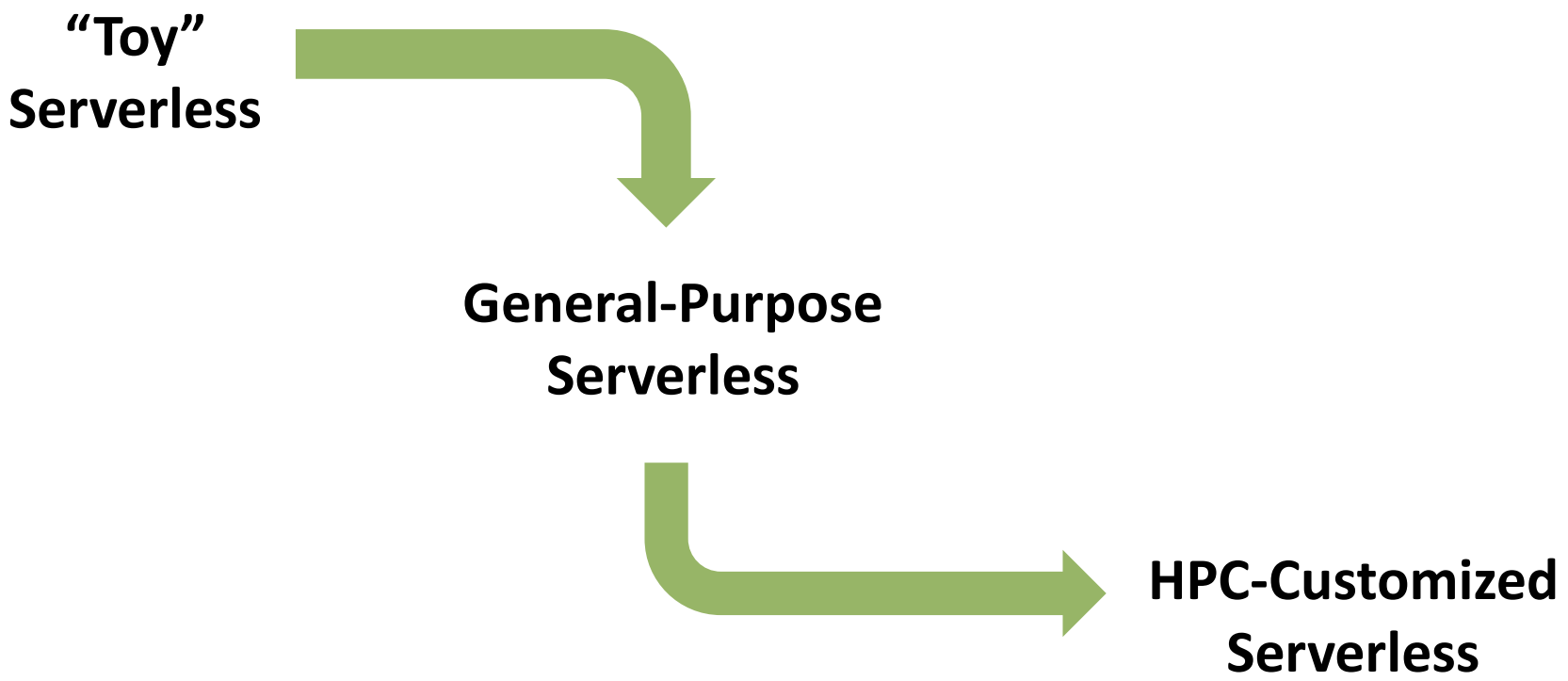
HPC

Burst Launches, Colocation Policies
Bulk Synchronous Parallel Model
Communicators, Message Passing, Collectives
…

Cloud

Embarrassingly Parallel

But we can make it closer to HPC!



But we can make it closer to HPC!

But we can make it closer to HPC!

Fast and Lightweight Sandboxes
(gVisor, Catalyzer, SEUSS, Photon)

But we can make it closer to HPC!

Fast and Lightweight Sandboxes
(gVisor, Catalyzer, SEUSS, Photon)

Stateful Functions
(Cloudburst, Faasm, Crucial, PraaS)

But we can make it closer to HPC!

Fast and Lightweight Sandboxes
(gVisor, Catalyzer, SEUSS, Photon)

Stateful Functions
(Cloudburst, Faasm, Crucial, PraaS)

Networking and Communication
(Boxer, FMI, rFaaS)

But we can make it closer to HPC!

Fast and Lightweight Sandboxes
(gVisor, Catalyzer, SEUSS, Photon)

Stateful Functions
(Cloudburst, Faasm, Crucial, PraaS)

Networking and Communication
(Boxer, FMI, rFaaS)

HPC FaaS
(Globus Compute, rFaaS, Lithops)

But we can make it closer to HPC!

Fast and Lightweight Sandboxes
(gVisor, Catalyzer, SEUSS, Photon)

Stateful Functions
(Cloudburst, Faasm, Crucial, PraaS)

Networking and Communication
(Boxer, FMI, rFaaS)

HPC FaaS
(Globus Compute, rFaaS, Lithops)

New HPC Workloads
(Cirrus, LambdaML, Mashup, DayDream)

But we can make it closer to HPC!

Fast and Lightweight Sandboxes
(gVisor, Catalyzer, SEUSS, Photon)

Stateful Functions
(Cloudburst, Faasm, Crucial, PraaS)

Networking and Communication
(Boxer, FMI, rFaaS)

HPC FaaS
(Globus Compute, rFaaS, Lithops)

New HPC Workloads
(Cirrus, LambdaML, Mashup, DayDream)

Improved Cold Starts
(RainbowCake, IceBreaker, Medes)

But we can make it closer to HPC!

Fast and Lightweight Sandboxes
(gVisor, Catalyzer, SEUSS, Photon)

Stateful Functions
(Cloudburst, Faasm, Crucial, PraaS)

Networking and Communication
(Boxer, FMI, rFaaS)

HPC FaaS
(Globus Compute, rFaaS, Lithops)

New HPC Workloads
(Cirrus, LambdaML, Mashup, DayDream)

Improved Cold Starts
(RainbowCake, IceBreaker, Medes)

Improved Scheduling (Wukong, Palette, PraaS, ProPack, Pheromone)

But we can make it closer to HPC!

Fast and Lightweight Sandboxes
(gVisor, Catalyzer, SEUSS, Photon)

Stateful Functions
(Cloudburst, Faasm, Crucial, PraaS)

Networking and Communication
(Boxer, FMI, rFaaS)

HPC FaaS
(Globus Compute, rFaaS, Lithops)

New HPC Workloads
(Cirrus, LambdaML, Mashup, DayDream)

Improved Cold Starts
(RainbowCake, IceBreaker, Medes)

Improved Scheduling (Wukong, Palette, PraaS, ProPack, Pheromone)

HPC Utilization
(HPC-Whisk, Serverless Disaggregation)

But we can make it closer to HPC!

Fast and Lightweight Sandboxes
(gVisor, Catalyzer, SEUSS, Photon)

Stateful Functions
(Cloudburst, Faasm, Crucial, PraaS)

Networking and Communication
(Boxer, FMI, rFaaS)

HPC FaaS
(Globus Compute, rFaaS, Lithops)

New HPC Workloads
(Cirrus, LambdaML, Mashup, DayDream)

Improved Cold Starts
(RainbowCake, IceBreaker, Medes)

Improved Scheduling (Wukong, Palette, PraaS, ProPack, Pheromone)

HPC Utilization
(HPC-Whisk, Serverless Disaggregation)

Benchmark Suites
(SeBS, Serverlessbench, FaaSDom)

But we can make it closer to HPC!

Fast and Lightweight Sandboxes
(gVisor, Catalyzer, SEUSS, Photon)

Stateful Functions
(Cloudburst, Faasm, Crucial, PraaS)

Networking and Communication
(Boxer, FMI, rFaaS)

HPC FaaS
(Globus Compute, rFaaS, Lithops)

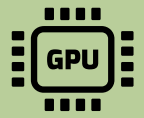
New HPC Workloads
(Cirrus, LambdaML, Mashup, DayDream)

Improved Cold Starts
(RainbowCake, IceBreaker, Medes)

Improved Scheduling (Wukong, Palette, PraaS, ProPack, Pheromone)

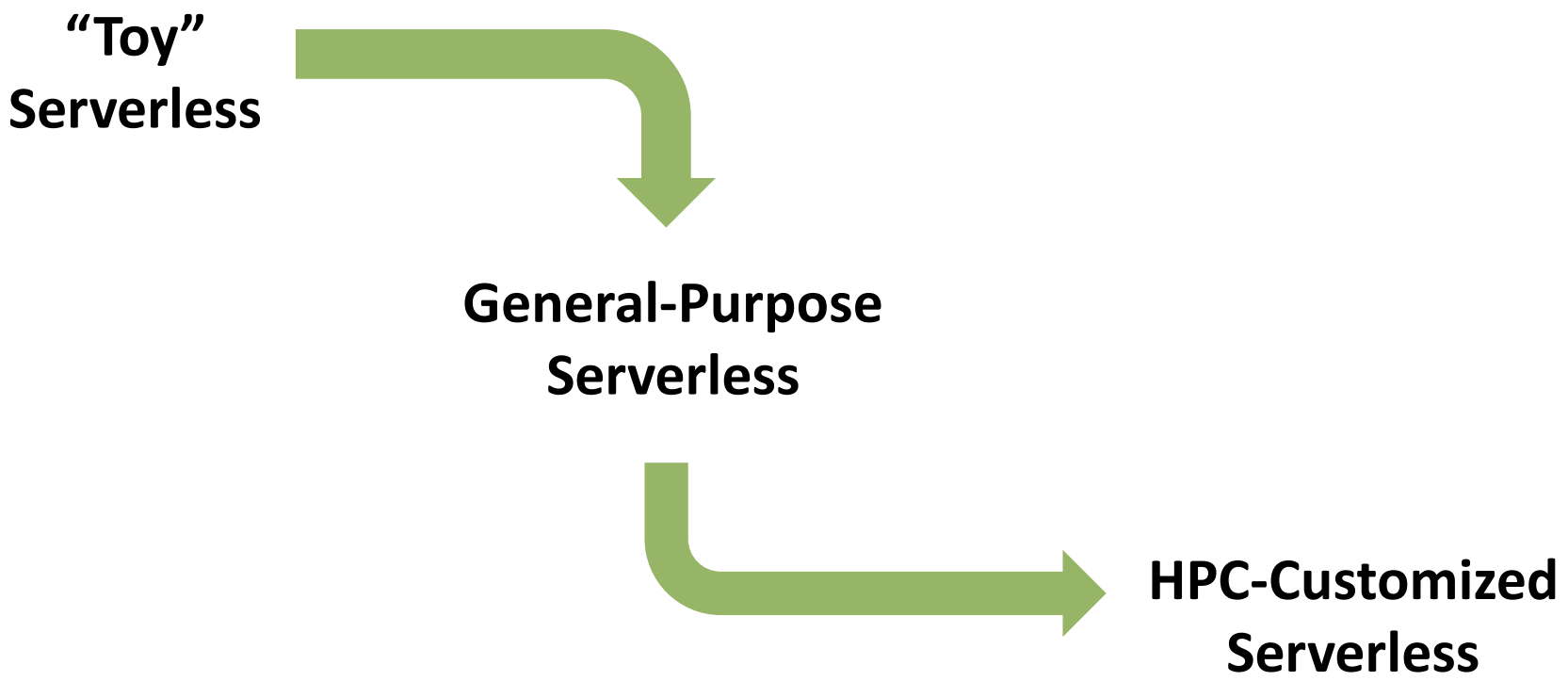
HPC Utilization
(HPC-Whisk, Serverless Disaggregation)

Benchmark Suites
(SeBS, Serverlessbench, FaaSDom)

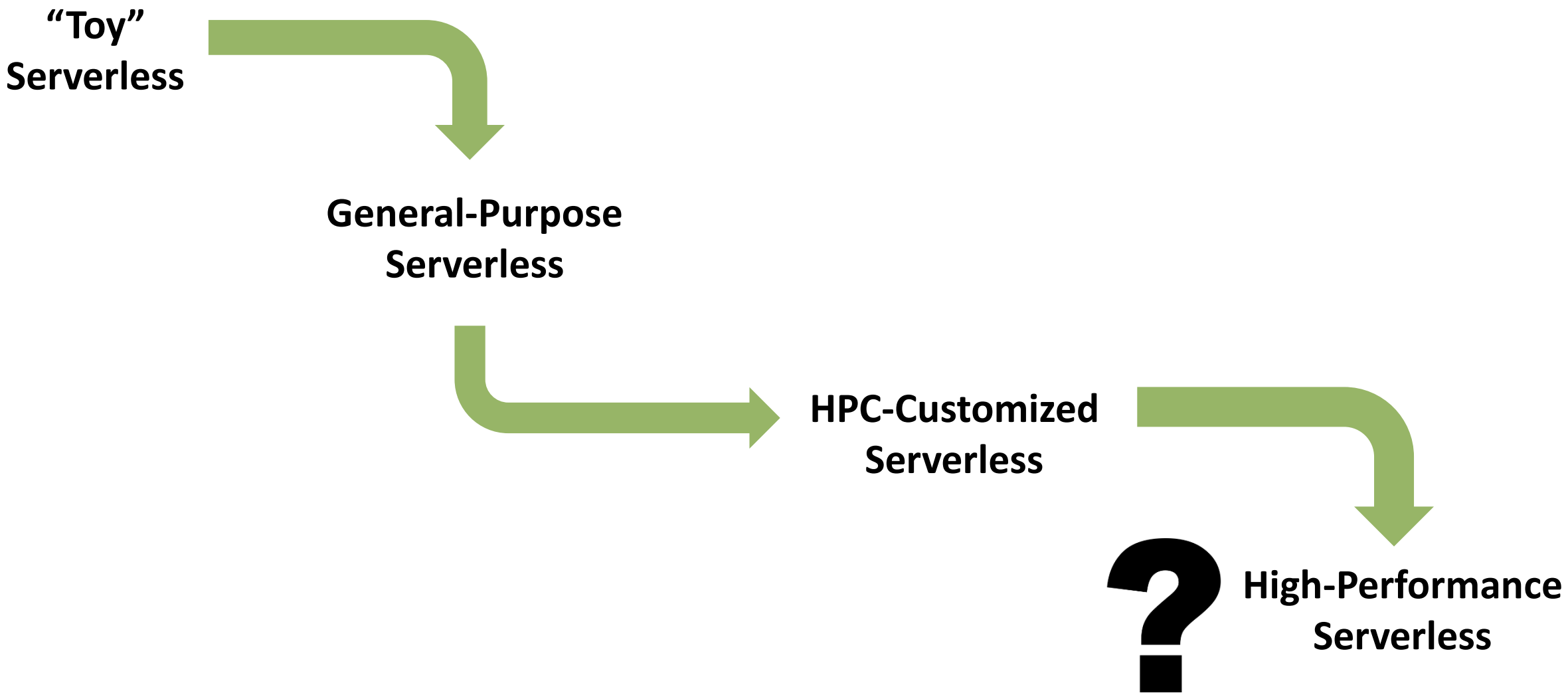


Accelerated Functions
(DGSF, KaaS, MIGnificient)

What is still missing?



What is still missing?



What is still missing?

"Toy" Serverless

General-Purpose Serverless

HPC-Customized Serverless

?

High-Performance Serverless

Serverless: One Step Toward HPC – Cloud Convergence

Infrastructure

Serverless: One Step Toward HPC – Cloud Convergence

Infrastructure

Cloud

Virtual Machines



Object Storage



Virtual Networks

HPC

Serverless: One Step Toward HPC – Cloud Convergence

Infrastructure

Cloud


Virtual Machines


Object Storage


Virtual Networks

HPC

HPCaaS


HPC in the cloud

Serverless: One Step Toward HPC – Cloud Convergence

Infrastructure

Cloud


Virtual Machines


Object Storage


Virtual Networks

HPC

HPCaaS


HPC in the cloud

Cloud in HPC

Serverless: One Step Toward HPC – Cloud Convergence

Infrastructure

Deployment

Cloud

Virtual Machines


Object Storage


Virtual Networks

Containers


(Managed) Kubernetes

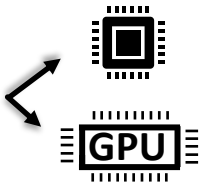
Services

HPC

HPCaaS


HPC in the cloud


IaaS, SaaS, PaaS



Cloud in HPC

Serverless: One Step Toward HPC – Cloud Convergence

Infrastructure

Deployment

Cloud

- Virtual Machines
- Object Storage
- Virtual Networks

- Containers
- (Managed) Kubernetes
- Services

HPC

HPCaaS

- HPC in the cloud
- Cloud in HPC

- HPC
- Containers
- Flux
- XaaS

IaaS, SaaS, PaaS

Serverless: One Step Toward HPC – Cloud Convergence

Infrastructure

Deployment

Compute

Cloud

Virtual Machines

Object Storage

Virtual Networks

Containers

(Managed) Kubernetes

Services

Serverless Functions

Serverless Containers

HPC

HPCaaS

HPC in the cloud

HPC

Containers

Flux

XaaS

IaaS, SaaS, PaaS

Cloud in HPC

Serverless: One Step Toward HPC – Cloud Convergence

	Infrastructure	Deployment	Compute
Cloud	Virtual Machines	Containers	Serverless Functions
	Object Storage	(Managed) Kubernetes	Serverless Containers
	Virtual Networks	Services	
HPC	HPCaaS	HPC Containers	Globus Compute (funcX)
	HPC in the cloud	Flux	RDMA rFaaS
	Cloud in HPC	XaaS	Lithops

IaaS, SaaS, PaaS

Serverless: One Step Toward HPC – Cloud Convergence

	Infrastructure	Deployment	Compute	Applications
Cloud	Virtual Machines	Containers	Serverless Functions	Dask, Spark Ray
	Object Storage	(Managed) Kubernetes	Serverless Containers	Serverless Workflows
	Virtual Networks	Services		
HPC	HPCaaS	HPC Containers	Globus Compute (funcX)	
	HPC in the cloud	Flux	RDMA rFaaS	
	Cloud in HPC	XaaS	Lithops	

IaaS, SaaS, PaaS

Serverless: One Step Toward HPC – Cloud Convergence

	Infrastructure	Deployment	Compute	Applications
Cloud	Virtual Machines	Containers	Serverless Functions	Dask, Spark, Ray
	Object Storage	(Managed) Kubernetes	Serverless Containers	Serverless Workflows
	Virtual Networks	Services		
HPC	HPC in the cloud	HPCaaS	HPC Containers	Globus Compute (funcX)
	Cloud in HPC (IaaS, SaaS, PaaS)	Flux	RDMA	rFaaS
		XaaS	LITHOPS	Lithops

Q&A

Kyle Chard
University of Chicago

Ian Foster
Argonne National Laboratory

Torsten Hoefler
ETH Zurich

Satoshi Matsuoka
Riken-CCS

Devesh Tiwari
Northeastern University

Moderator

Marcin Copik
ETH Zurich