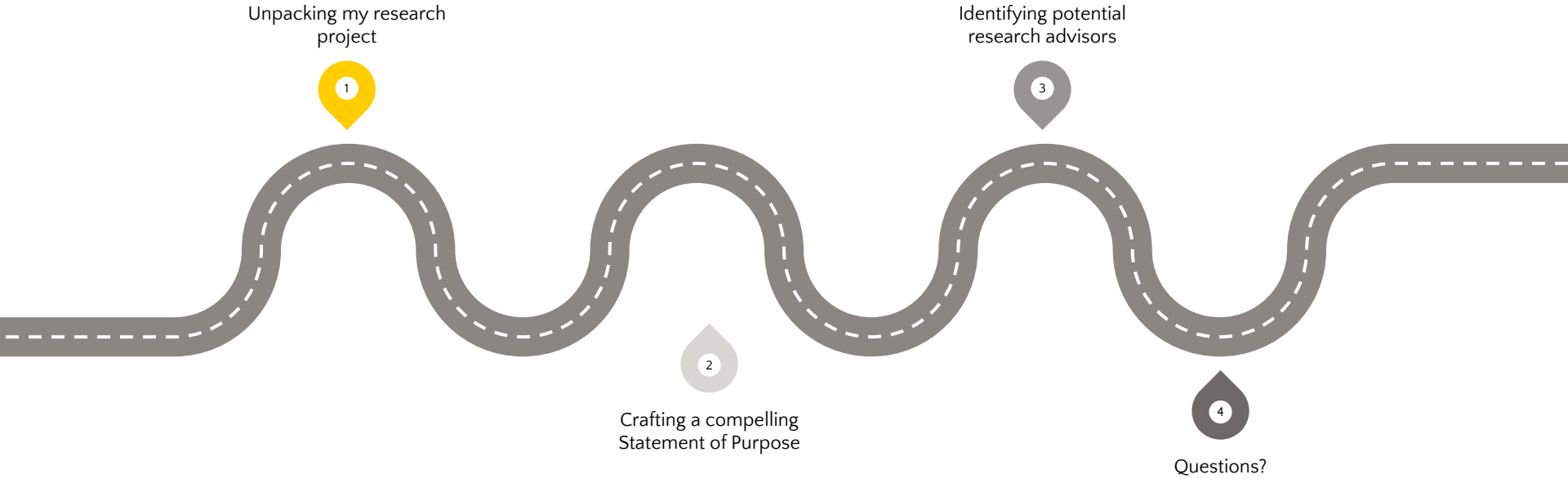
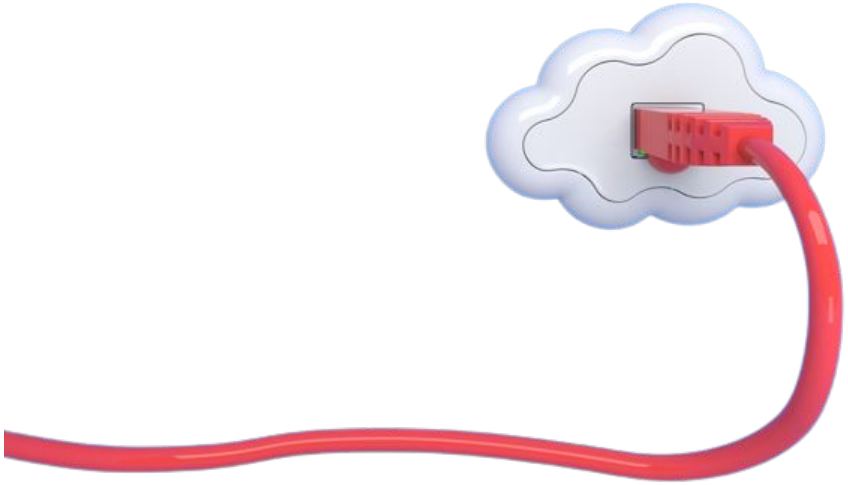
Breaking into Research: Insights from my journey

Roadmap

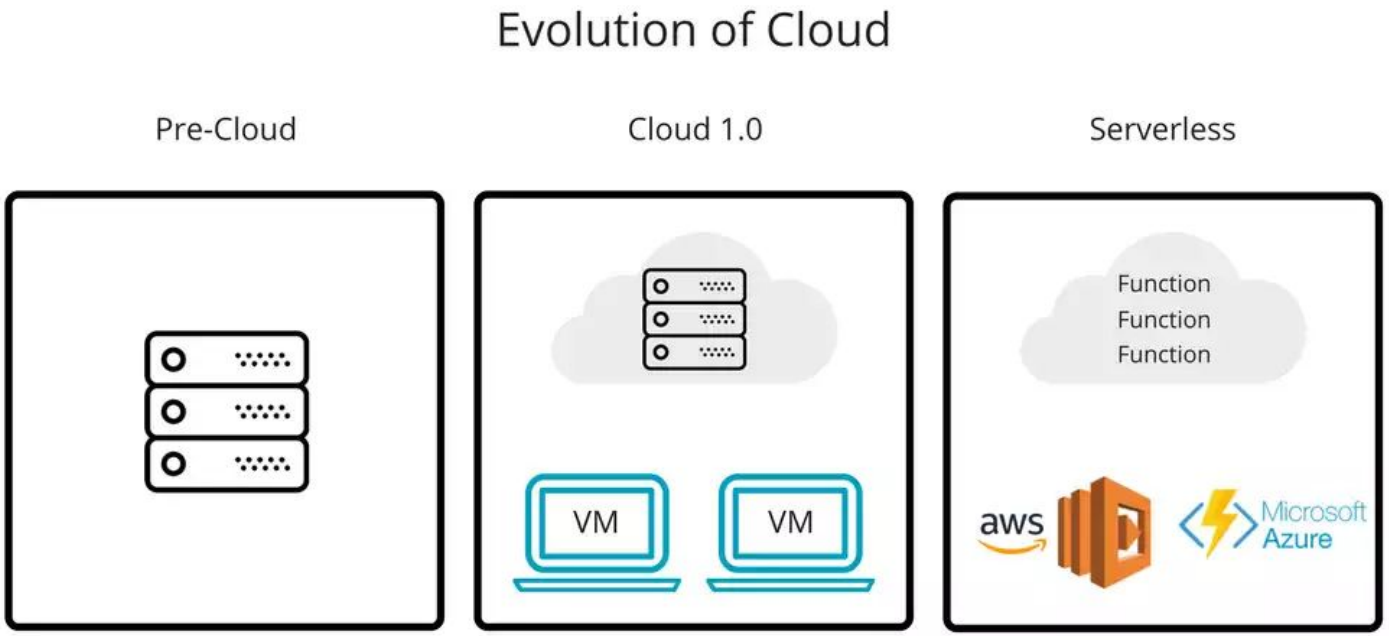

Unpacking my research project
1
Identifying potential research advisors
3
2
Crafting a compelling Statement of Purpose
4
Questions?

My Research

- Designing systems for emerging cloud workloads.
- Typical workloads today are highly distributed (e.g., machine learning).
- My research focuses on building the cloud system that power the next generation of workloads.
- I currently focus on improving state-of-the-art in serverless computing.

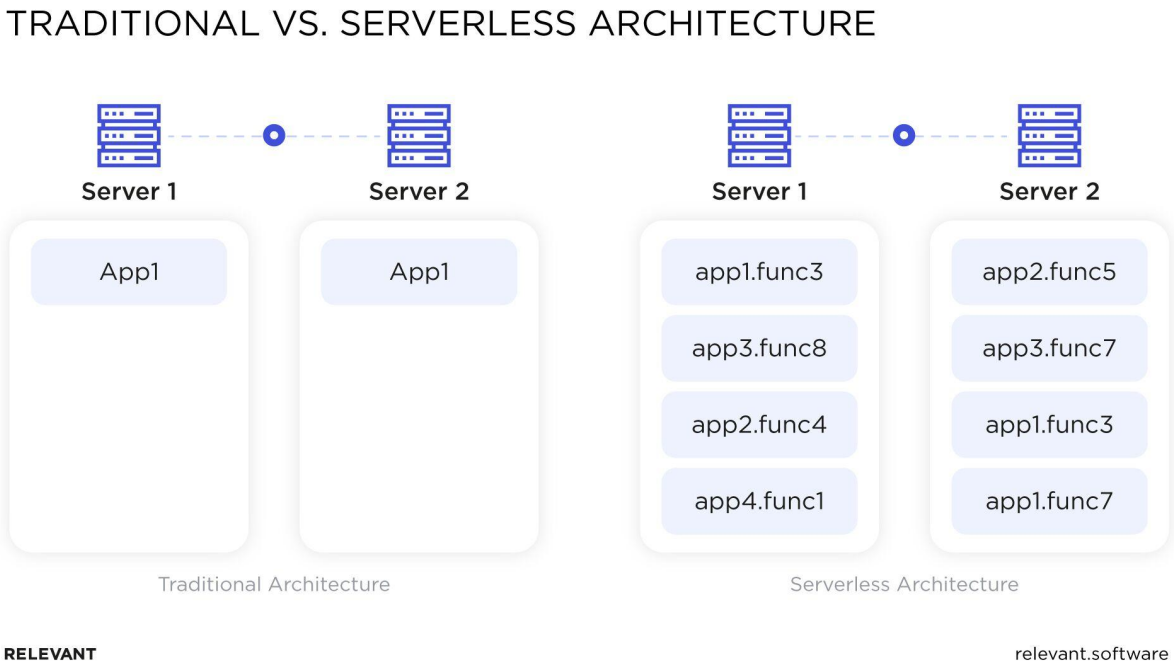


Background: Cloud Computing



https://www.altexsoft.com/blog/cloud/pros-and-cons-of-serverless-architecture/

Background: Cloud Computing



https://relevant.software/blog/serverless-architecture/

What is Serverless Computing?

- Event-driven paradigm where cloud-providers handle the administration of infrastructure.
- A serverless function consists of small, stateless logical unit of code executed in a sandboxed environment (e.g., container).
- Reduces burden on developers and enables fine-grained billing and highly elastic resource provisioning for cloud-providers.

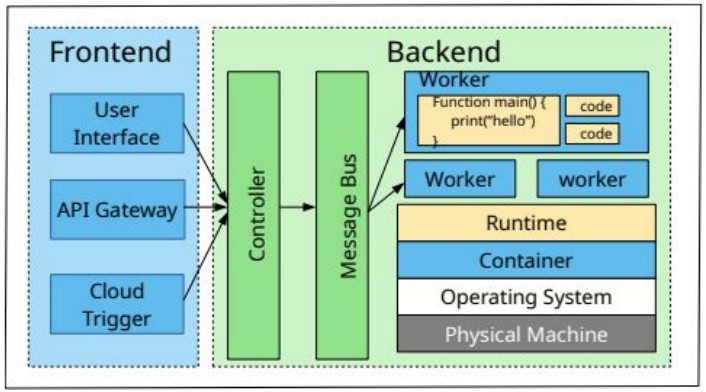
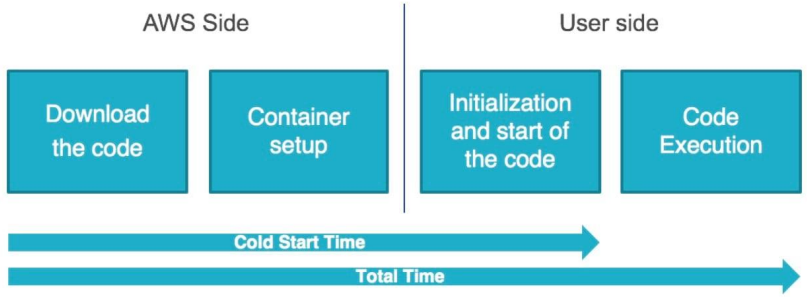


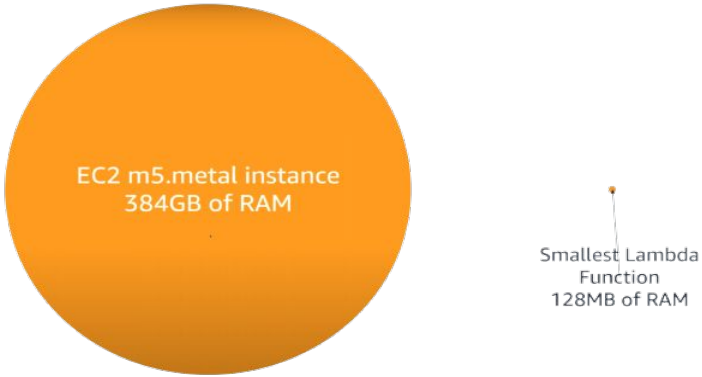
Figure 1. Overall architecture of a serverless platform.

Characteristics and Challenges

- Function executions are short-lived. 50% execute in < 1 second.
- Allow consolidation of a large number of serverless functions with low hardware costs.
- To maximize resource utilization efficiency, cloud providers routinely “evict” functions.

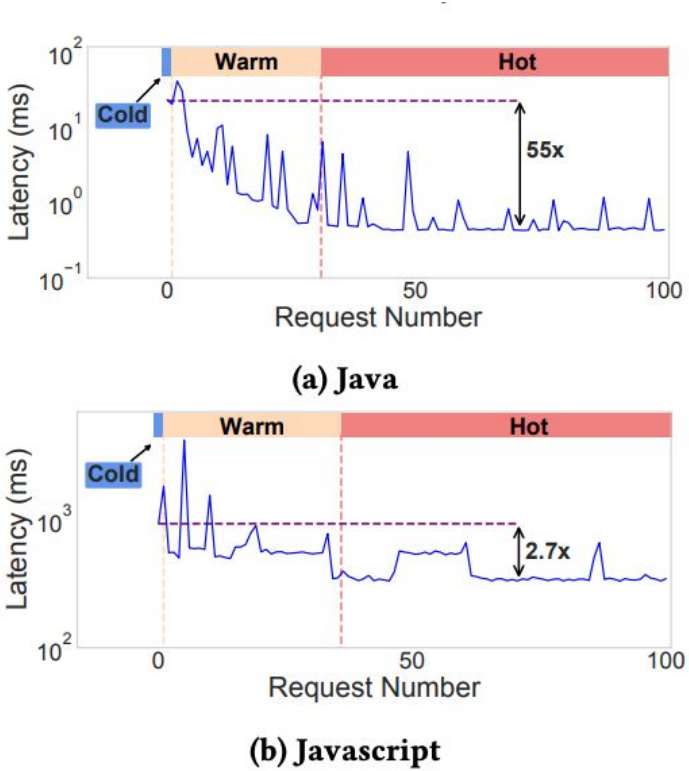


Source : AWS re:Invent 2017 – Become a serverless Black Belt



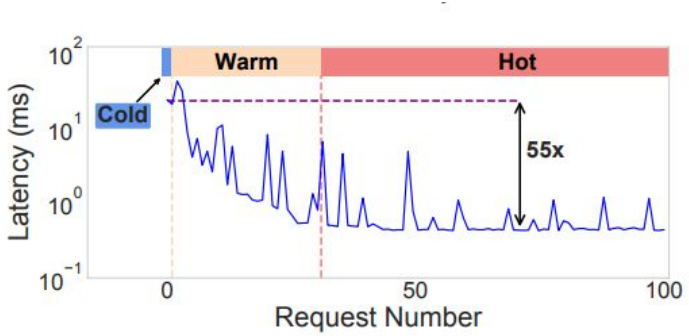
Existing Approaches & Limitations

- One existing approach aim to solve the *cold-start* is checkpoint/restore.
- Majority of the functions are written in high-level languages (e.g., JavaScript) with optimizing runtimes that gain knowledge over the lifetime of executions.
- However, everytime a container is evicted, all the runtime optimizations are lost leading to wasteful computations and missed performance gains.

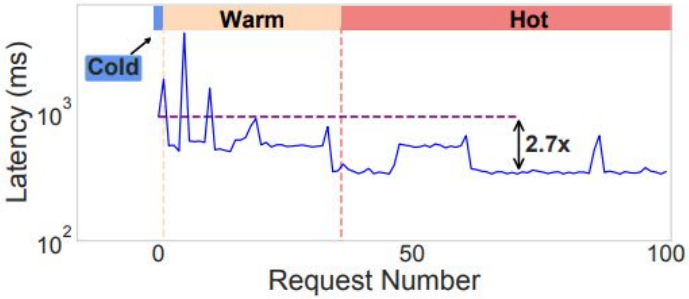


Existing Approaches & Limitations

- One existing approach aim to solve the *cold-start* is checkpoint/restore.
- Majority of the functions are written in high-level languages (e.g., JavaScript) with optimizing runtimes that gain knowledge over the lifetime of executions.
- However, everytime a container is evicted, all the runtime optimizations are lost leading to wasteful computations and missed performance gains. What to do?



(a) Java



(b) Javascript

Background: JIT Compilation

- JIT compilers speculatively optimize code by profiling its runtime execution.
- Typically, JIT compilers are suited for long-running computations.
- JIT compilers can take hundreds of invocations to fully optimize code. (can't wait till JIT fully optimizes!)
- JIT compilers can make mistakes performance degradations. (fixed requests are sub-optimal!)

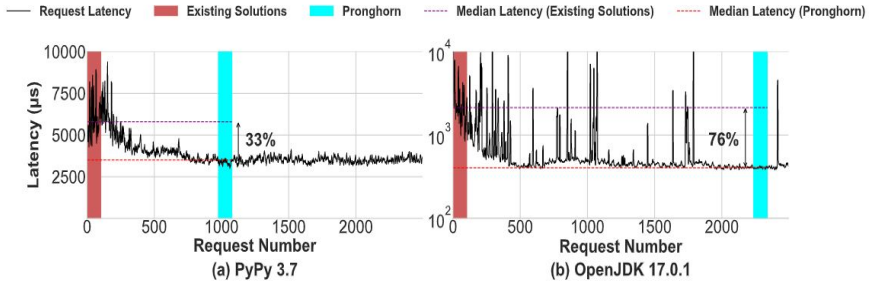


Figure 1: Dynamic HTML generation workload latency using two optimizing runtimes. Experiment shows the performance difference produced by premature checkpoints (used in existing solutions) and an ideal checkpoint. For PyPy, the performance difference is about 33%, while for JVM it is about 76%. The periodic spikes seen in the JVM sub-figure are due to garbage collection.

Background: JIT Compilation

- JIT compilers speculatively optimize code by profiling its runtime execution.
- Typically, JIT compilers are suited for long-running computations.
- JIT compilers can take hundreds of invocations to fully optimize code. (can't wait till JIT fully optimizes!)
- JIT compilers can make mistakes performance degradations. (fixed requests are sub-optimal!)

Needs a dynamic approach!!

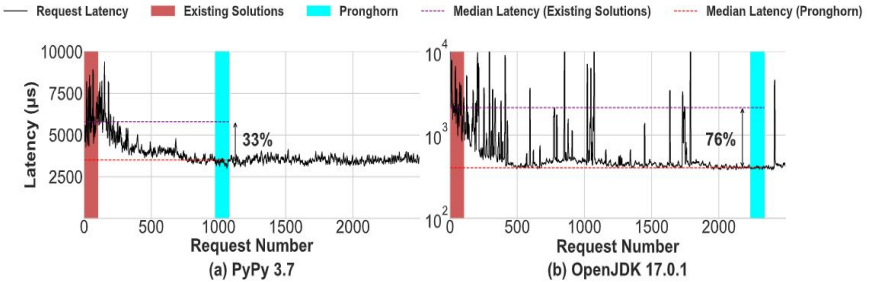


Figure 1: Dynamic HTML generation workload latency using two optimizing runtimes. Experiment shows the performance difference produced by premature checkpoints (used in existing solutions) and an ideal checkpoint. For PyPy, the performance difference is about 33%, while for JVM it is about 76%. The periodic spikes seen in the JVM sub-figure are due to garbage collection.

Solution: Checkpoint Orchestration

- A novel request-centric (dynamic) strategy that aggregates the knowledge gained by the runtimes.
- Uses latency profiles to decide (a) when to checkpoint and (b) which snapshot (artifact of a checkpoint) to restore from in the pool.
- Over time, converges to a snapshot which enables other instances to start with a hot-start.

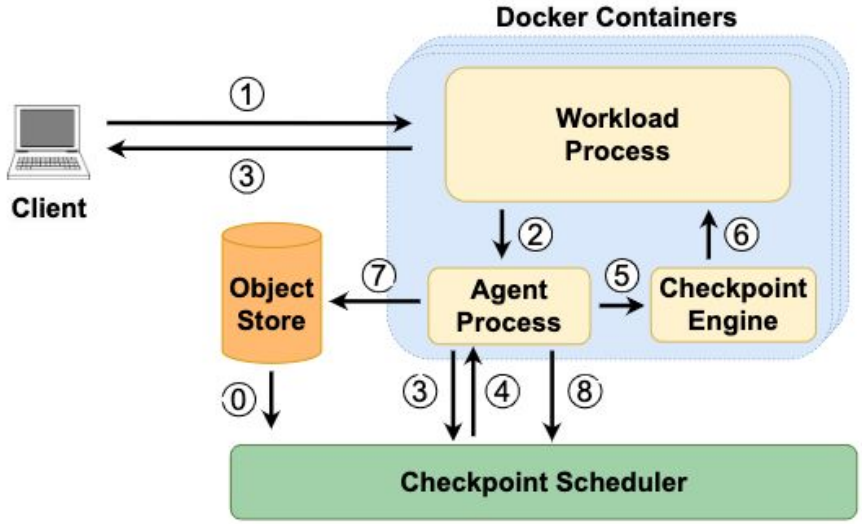


Figure 2: Pronghorn System Design

My Contributions

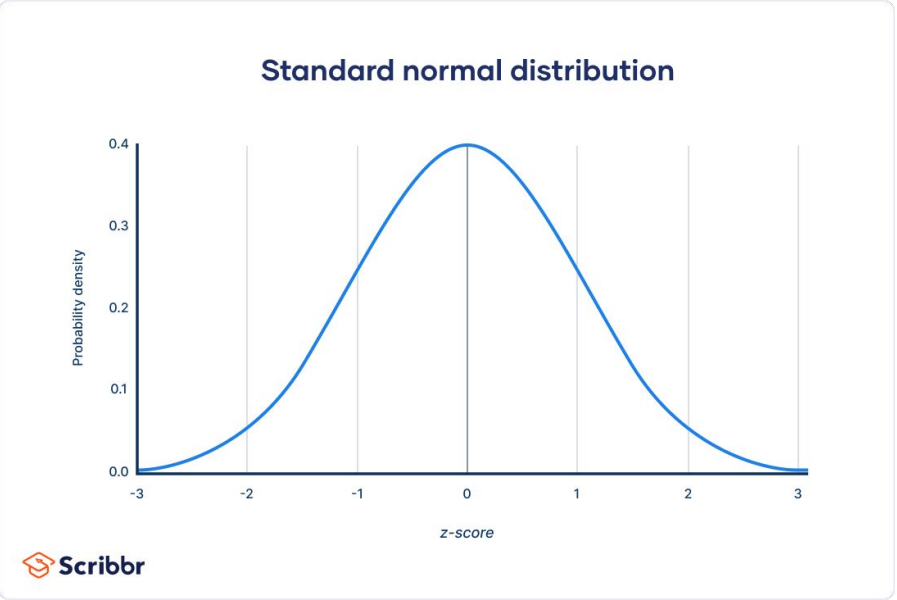
- I primarily dealt with system design and implementation.

My Contributions

- I primarily dealt with system design and implementation.
- Built the automated system evaluation infrastructure. (Lots of Python!)

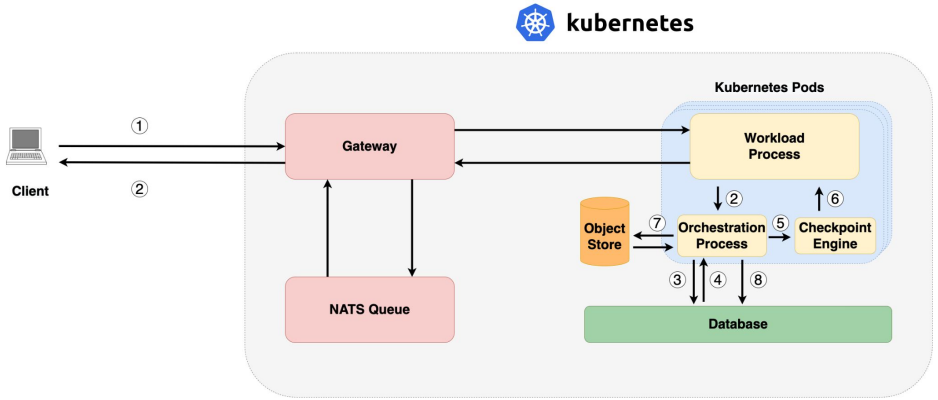
My Contributions

- I primarily dealt with system design and implementation.
- Built the automated system evaluation infrastructure. (Lots of Python!)
- Extended our evaluation infrastructure's capabilities to simulate production conditions (e.g., traffic patterns and input size variations)



My Contributions

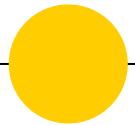
- I primarily dealt with system design and implementation.
- Built the automated system evaluation infrastructure. (Lots of Python!)
- Extended our evaluation infrastructure's capabilities to simulate production conditions (e.g., traffic patterns and input size variations)
- Designed and implemented our end-to-end, cloud-native system.



Key Insights

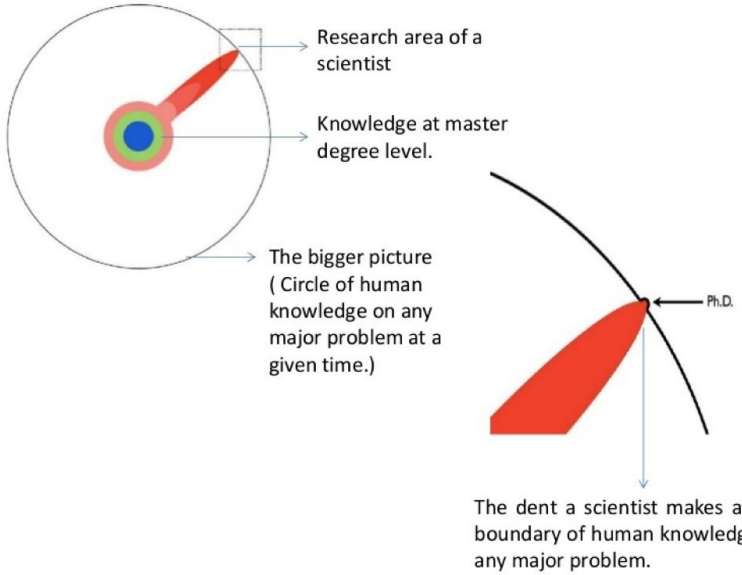
- In any endeavour, you're inevitably going to run into particularly difficult tasks – stick with the ones which are worth the suffering.
- Build exciting projects!
- Uncertainty is OK.
- In computing, there is enough space for everyone to do something they deeply care about.

Questions?



Why PhD in Computer Science?

- Autonomy to set your own research agenda, and pursue projects **you** enjoy.
- Work on big, challenging problems with amazing people.
- Create new knowledge that will push the collective knowledge of the field forward.
- Challenge yourself → A PhD is hard but equally rewarding.



https://www.rug.nl/aletta/blog/to-expand-the-limits-of-knowledge-is-not-enough-04-09-2018

Crafting a compelling Statement of Purpose

- [Introduction] Get to the point – Why, What and How? + Overarching Goal
- [Relevant Work] Motivation + Solution + Your Contributions. Typically, 2-3 projects with similar document real estate.
- [Research Fit] Identify 2-3 potential advisors whom you'd be interested to work with.
- [Conclusion] Template paragraph reiterating key points and declaring your love for the university (don't overdo it!)
- Keep a tight feedback loop with (ideally) 3-4 trusted reviewers.

[Introduction] Statement of Purpose

Why?

Scientific advancements have led to a shift in the nature of today's computational workloads. Typical workloads in cloud environments today are increasingly distributed, requiring low latency, near-constant availability, and the ability to rapidly scale with need. How do we design efficient systems that power this new generation of workloads? How do we construct effective measures to secure these systems? Inspired by these questions, my research focuses on enhancing the efficiency, usability, and security of computer systems. In particular, I design (a) transparent orchestration layers that aggregate knowledge from individual distributed system nodes to reduce resource use inefficiency and improve usability and (b) adaptation layers to help developers execute their distributed workloads on secure cloud platforms. By enhancing the cost-efficiency and usability of secure distributed systems, I aim to enable more equitable access to cloud computing. With this broad goal in mind, I am applying to the Ph.D. program

[Introduction] Statement of Purpose

What?

Scientific advancements have led to a shift in the nature of today's computational workloads. Typical workloads in cloud environments today are increasingly distributed, requiring low latency, near-constant availability, and the ability to rapidly scale with need. How do we design efficient systems that power this new generation of workloads? How do we construct effective measures to secure these systems? Inspired by these questions, my research focuses on enhancing the efficiency, usability, and security of computer systems. In particular, I design (a) transparent orchestration layers that aggregate knowledge from individual distributed system nodes to reduce resource use inefficiency and improve usability and (b) adaptation layers to help developers execute their distributed workloads on secure cloud platforms. By enhancing the cost-efficiency and usability of secure distributed systems, I aim to enable more equitable access to cloud computing. With this broad goal in mind, I am applying to the Ph.D. program

[Introduction] Statement of Purpose

How?

Scientific advancements have led to a shift in the nature of today's computational workloads. Typical workloads in cloud environments today are increasingly distributed, requiring low latency, near-constant availability, and the ability to rapidly scale with need. How do we design efficient systems that power this new generation of workloads? How do we construct effective measures to secure these systems? Inspired by these questions, my research focuses on enhancing the efficiency, usability, and security of computer systems. In particular, I design (a) transparent orchestration layers that aggregate knowledge from individual distributed system nodes to reduce resource use inefficiency and improve usability and (b) adaptation layers to help developers execute their distributed workloads on secure cloud platforms. By enhancing the cost-efficiency and usability of secure distributed systems, I aim to enable more equitable access to cloud computing. With this broad goal in mind, I am applying to the Ph.D. program

[Introduction] Statement of Purpose

Overarching Goal?

Scientific advancements have led to a shift in the nature of today's computational workloads. Typical workloads in cloud environments today are increasingly distributed, requiring low latency, near-constant availability, and the ability to rapidly scale with need. How do we design efficient systems that power this new generation of workloads? How do we construct effective measures to secure these systems? Inspired by these questions, my research focuses on enhancing the efficiency, usability, and security of computer systems. In particular, I design (a) transparent orchestration layers that aggregate knowledge from individual distributed system nodes to reduce resource use inefficiency and improve usability and (b) adaptation layers to help developers execute their distributed workloads on secure cloud platforms. By enhancing the cost-efficiency and usability of secure distributed systems, I aim to enable more equitable access to cloud computing. With this broad goal in mind, I am applying to the Ph.D. program

Identifying Potential Research Advisors

- Evaluate fit by discipline. For example, if you're deeply interested in cloud systems, look for faculty members working in systems.
- Using your excitement in their previous and/or ongoing projects is the best way to identify a potential fit.
- Make sure that the faculty member is hiring (Keep an eye out for new appointments at the school → Academia Twitter is a great resource).
- Reach out to a faculty member **if and only if** there's something meaningful to be said. Avoid general introductory emails.

Conveying Fit

Why School? (Generic is fine)

These experiences have been instrumental in shaping my research interests and have motivated me to pursue graduate studies. Cornell's commitment to academic freedom and impactful doctoral research compels me. I am incredibly excited about the opportunity to work with Professors Rachit Agarwal, Lorenzo Alvisi, and Robbert van Renesse. Prof. Agarwal's research on shared-nothing architecture addresses fundamental bottlenecks in cost-efficient resource usage. Working under his guidance, I believe I can significantly contribute to designing efficient and secure distributed programming frameworks on disaggregated architectures. I am also excited by Prof. Alvisi's research on combining distributed systems primitives to build scalable and fault-tolerant shared logs, an aspect I am excited to further explore in serverless computing. Finally, I am very interested in working with Prof. Renesse on transformations to enhance the scalability and reliability of modern large-scale systems to support emerging workloads.

Conveying Fit

Who are you excited to work with?

These experiences have been instrumental in shaping my research interests and have motivated me to pursue graduate studies. Cornell's commitment to academic freedom and impactful doctoral research compels me. I am incredibly excited about the opportunity to work with Professors Rachit Agarwal, Lorenzo Alvisi, and Robbert van Renesse. Prof. Agarwal's research on shared-nothing architecture addresses fundamental bottlenecks in cost-efficient resource usage. Working under his guidance, I believe I can significantly contribute to designing efficient and secure distributed programming frameworks on disaggregated architectures. I am also excited by Prof. Alvisi's research on combining distributed systems primitives to build scalable and fault-tolerant shared logs, an aspect I am excited to further explore in serverless computing. Finally, I am very interested in working with Prof. Renesse on transformations to enhance the scalability and reliability of modern large-scale systems to support emerging workloads.

Conveying Fit

Your experience + their projects!

These experiences have been instrumental in shaping my research interests and have motivated me to pursue graduate studies. Cornell's commitment to academic freedom and impactful doctoral research compels me. I am incredibly excited about the opportunity to work with Professors Rachit Agarwal, Lorenzo Alvisi, and Robbert van Renesse. Prof. Agarwal's research on shared-nothing architecture addresses fundamental bottlenecks in cost-efficient resource usage. Working under his guidance, I believe I can significantly contribute to designing efficient and secure distributed programming frameworks on disaggregated architectures. I am also excited by Prof. Alvisi's research on combining distributed systems primitives to build scalable and fault-tolerant shared logs, an aspect I am excited to further explore in serverless computing. Finally, I am very interested in working with Prof. Renesse on transformations to enhance the scalability and reliability of modern large-scale systems to support emerging workloads.

Conveying Fit

New direction!

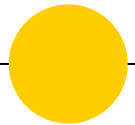
These experiences have been instrumental in shaping my research interests and have motivated me to pursue graduate studies. Cornell's commitment to academic freedom and impactful doctoral research compels me. I am incredibly excited about the opportunity to work with Professors Rachit Agarwal, Lorenzo Alvisi, and Robbert van Renesse. Prof. Agarwal's research on shared-nothing architecture addresses fundamental bottlenecks in cost-efficient resource usage. Working under his guidance, I believe I can significantly contribute to designing efficient and secure distributed programming frameworks on disaggregated architectures. I am also excited by Prof. Alvisi's research on combining distributed systems primitives to build scalable and fault-tolerant shared logs, an aspect I am excited to further explore in serverless computing. Finally, I am very interested in working with Prof. Renesse on transformations to enhance the scalability and reliability of modern large-scale systems to support emerging workloads.

Conveying Fit

Direct match :)

These experiences have been instrumental in shaping my research interests and have motivated me to pursue graduate studies. Cornell's commitment to academic freedom and impactful doctoral research compels me. I am incredibly excited about the opportunity to work with Professors Rachit Agarwal, Lorenzo Alvisi, and Robbert van Renesse. Prof. Agarwal's research on shared-nothing architecture addresses fundamental bottlenecks in cost-efficient resource usage. Working under his guidance, I believe I can significantly contribute to designing efficient and secure distributed programming frameworks on disaggregated architectures. I am also excited by Prof. Alvisi's research on combining distributed systems primitives to build scalable and fault-tolerant shared logs, an aspect I am excited to further explore in serverless computing. Finally, I am very interested in working with Prof. Renesse on transformations to enhance the scalability and reliability of modern large-scale systems to support emerging workloads.

Questions?



Contact

Email: skharban@purdue.edu
Twitter: 0xkharban