Over the past seven years, my research has been focused on enhancing the elasticity of cloud-based systems that serve real-time data. Real-time data from different data sources, ranging from traffic cameras to stock exchanges, can introduce different workload patterns that are difficult to predict in advance, especially for long-running, large scale services in the cloud. This makes resource provisioning for real-time data notoriously difficult. Cloud resources allocated to processing real-time data need to be constantly adjusted over time to avoid **under-provisioning** (which delays critical events, such as traffic accidents, from being processed) and **over-provisioning** (which severely hinders resource efficiency). Research [1] shows that major cloud providers, like Amazon AWS and Google Cloud, severely overprovision resources. This leads to both a high carbon footprint for the cloud providers and increased monetary costs for the cloud users, increasing the cost by up to 544% [1].

The primary objective of my research has been to continuously improve cloud-based systems, so they adapt elastically to real-time data. This is achieved by first deciphering the intricate relationships between workload, application characteristics, and user requirements. Subsequently, I propose system abstractions and architectural designs that drastically reduce the cost of re-provisioning cloud resources for individual or multiple tenant applications. My past research focused on improving elasticity of cloud-based systems from the following perspectives:

- I've developed policies, metrics, and system implementations that empower real-time data processing systems to dynamically scale out and scale in, both in single and multi-tenant environments. This approach enables applications to adeptly adjust provisioned resources by automatically recognizing workload shifts and translating these observations into real-time resource requirements.
- I introduced a programmable control plane complemented by a meticulously designed control message. This innovation ensures real-time data processing systems can execute user-defined operations with minimal interruptions, all while maintaining computational state consistency. This design further improves resource efficiency by eliminating the need to restart applications during the re-provisioning process
- I've built an event-driven framework tailored for large-scale, multi-tenant real-time data processing applications. This design facilitates resource multiplexing among various tenants or their application operators at the granularity of individual events. Such precision ensures that cloud resources are allocated to the prioritized events, optimizing resource efficiency and consistently meeting performance benchmarks.
- I've conducted an in-depth analysis correlating cloud application characteristics with their performance across diverse hardware configurations. This research also offers a comprehensive guide on the optimal cloud resources settings for various workloads, shedding light on potential redesigns of cloud engines to harness hardware resources more effectively under alternative computer architectures.

The rise of AI applications and their integration into various data workflows have introduced new complexities. My previous research delved into real-time data processing applications primarily set up in the datacenter settings, utilizing consistent hardware and well-understood processing semantics. Moving forward, my research will pivot towards two main directions: enhancing system adaptability in heterogeneous computing environments and constructing AI pipelines that prioritize semantic-driven modularity. Specifically, I’m interested in how these aspects would impact future systems that process real-time data. As the cloud is poised for significant expansion, addressing system adaptability in varied settings becomes crucial for capitalizing on this growth. Moreover, streamlined AI workflows can result in resource efficiencies, accelerated advancements, and a competitive edge in the international AI landscape. This research direction not only resonates with the tech industry's goals of fostering effective AI-human collaborations and ensuring AI system integrity but also positions it at the forefront of global AI advancements.

Past Projects

Project 1: Stream Processing Elasticity

List of publications and preprints::

- Xu, Le, Boyang Peng, and Indranil Gupta. "Stela: Enabling stream processing systems to scale-in and scale-out on-demand." In 2016 IEEE International Conference on Cloud Engineering (IC2E), pp. 22-31. IEEE, 2016.
- Ghosh, Mainak, Ashwini Raina, Le Xu, Xiaoyao Qian, Indranil Gupta, and Himanshu Gupta. "Popular is cheaper: Curtailing memory costs in interactive analytics engines." In Proceedings of the Thirteenth EuroSys Conference, pp. 1-14. 2018.
- Kalim, Faria, Le Xu, Sharanya Bathey, Richa Meherwal, and Indranil Gupta. "Henge: Intent-driven multi-tenant stream processing." In Proceedings of the ACM Symposium on Cloud Computing, pp. 249-262. 2018.

The primary focus of this project revolves around enhancing the scalability and efficiency of real-time data processing systems within cloud environments. The initial phase of my project (*Stela*) was dedicated to developing a system capable of **on-demand** scale-in and scale-out for real-time data processing applications, aiming to optimize post-scaling throughput while minimizing computation interruption during scaling operations. A novel metric was introduced to measure the impact of an operator on the entire application, guiding the scale-out and scale-in processes. This metric was instrumental in selecting which operators receive more resources during scale-out and which machines to remove during scale-in to minimize detriment to application performance. The implementation of a scheduler for on-demand planning of user-triggered scale-out and scale-in was evaluated using various workloads and topologies, demonstrating a substantial improvement in post-scale throughput ranging from 45-120% during on-demand scale-out and 2-5 times during on-demand scale-in. The subsequent project (*Henge*) extended this system to accommodate **multi-tenant resource sharing**. In one continuing project, my work introduces a utility function and a new metric named Juice to evaluate user satisfaction and job processing efficiency respectively. Through these advancements, the system showcased a notable reduction in dollar cost by 40-60%, significantly impacting cloud providers' profit margins. In another continuing project (*Getafix*), my work focuses on reducing query makespan and multi-tenant resource consumption, **adapting to changing data popularity**, for OLAP queries arriving in real time. This work introduces a novel bin-packing-based mechanism that reduces query makespan and cloud memory consumption simultaneously. The experiments show that our system could achieve memory resource saving up to 2.15 times and reduce public cloud dollar cost by 10 million per 100 Terabyte data annually, and does so without compromising query performance.

The implications of this line of works are profound in the realm of real-time data processing reconfiguration. These applications pose common challenges caused by unpredictable input volume changes, resource requirements, and performance targets. These challenges necessitate adaptive and user-facing reconfiguration strategies for real-time data applications, designed for cloud users who manage their resources, as well as providers who allocate cloud resources to different users. My work addresses these challenges by offering a systematic approach to both on-demand and automatic resource allocation, **significantly reducing the manual effort required by users for continuous monitoring and re-deployment**. My work first provides a robust framework for users to optimize resource allocation in response to workload fluctuations, thereby **enhancing throughput and reducing cloud spending**. Then the extension of this work delves into a multi-tenant setting further augments the system's capability to balance resources among various tenants sharing cloud resources, achieving higher overall utility and substantial cost savings. This not only cuts down cloud spending and boosts energy efficiency but also alleviates the engineering effort required for constant monitoring and re-deployment, making a significant stride towards more autonomous and efficient real-time data processing in cloud environments.

Project 2: Programmable Control Plane for Stream Processing

List of publications and preprints:

- Mai, Luo, Kai Zeng, Rahul Potharaju, Le Xu, Steve Suh, Shivaram Venkataraman, Paolo Costa et al. "Chi: A scalable and programmable control plane for distributed stream processing systems." *Proceedings of the VLDB Endowment*11, no. 10 (2018): 1303-1316.

The crux of my work (*Chi*) lies in improving the adaptability of stream processing systems to ensure timeliness amidst constantly changing external factors. The primary hurdles include the absence of a dynamic, programmable online mechanism for cloud users to monitor and modify their applications without restarting their pipelines that

potentially run on hundreds or thousands of machines. To tackle these challenges, my project aimed at **proposing a novel programming abstraction enabling users to program arbitrary control operations**, and designing mechanisms to asynchronously execute these control policies, thereby **avoiding global synchronization**. This work first studies transient workload patterns and user requirements we observed from more than 200,000 machines in production clusters, which motivates for the need for efficient control operations. Then the work proposes a system layer termed the control plane, dispatching messages that flow along regular data messages along the pipeline, to manage these control mechanisms. Through this control plane and a set of API functions, users can easily implement a wide spectrum of complex control policies enabling continuous monitoring, feedback, and dynamic re-configuration operations encompassing all aspects of online tuning for streaming applications, ranging from query planning and resource allocation/scheduling to parameter tuning.

The current limitations in stream processing systems, which necessitate pausing and restarting applications for control operations, significantly hinder the efficiency and responsiveness of real-time data processing. This work advocates for a design where control operations are seamlessly embedded into the data plane, enabling a plethora of programmable control operations like reconfiguration and monitoring to be carried out online without halting running applications. This design **drastically reduces the overhead of control operations**, leading to a more elastic system design that can adapt to workload changes in real time. By reducing the latency by 61% and minimizing workload skew during runtime, this project significantly contributes to **enhancing the automatic tuning of critical system parameters**. The novel control-plane/control policy design not only significantly reduces the complexity of system design but also enables control operations to be executed at a processing rate, showcasing a promising avenue toward achieving low latency in stream processing systems. This, in turn, is pivotal for organizations and entities reliant on real-time analytics for decision-making, thereby having a broader impact on how data-driven insights are garnered and utilized in a rapidly evolving digital landscape.

Project 3: Fine-grained stream processing framework

List of publications and preprints:

- Xu, Le, Shivaram Venkataraman, Indranil Gupta, Luo Mai, and Rahul Potharaju. "Move fast and meet deadlines: Fine-grained real-time stream processing with cameo." In *18th USENIX Symposium on Networked Systems Design and Implementation (NSDI 21)*, pp. 389-405. 2021.
- Su, Li, Xiaoming Qin, Zichao Zhang, Rui Yang, Le Xu, Indranil Gupta, Wenyuan Yu, Kai Zeng, and Jingren Zhou. "Banyan: a scoped dataflow engine for graph query service." *Proceedings of the VLDB Endowment* 15, no. 10 (2022): 2045-2057.
- Xu, Le, Divyanshu Saxena, Neeraja J. Yadwadkar, Aditya Akella, and Indranil Gupta. "Dirigo: Self-scaling Stateful Actors For Serverless Real-time Data Processing." *arXiv preprint arXiv:2308.03615* (2023).

The focal point of my work (*Cameo*) is enhancing resource provisioning in multi-tenant stream processing systems to address the dual challenges of maintaining high resource utilization without over-provisioning while ensuring performance isolation. The prevalent "slot-based" approach used by today's popular stream processing engines necessitates manual configuration by users to allocate resources, often leading to severe over-provisioning as users tend to estimate resource needs based on workload peaks to avert congestion. To circumvent these issues, this project proposes a scheduler for cloud-based stream processing engines that schedules different tenant operator executions based on the data to be processed, thus **introducing a more fine-grained execution paradigm based on the actor model**. This work starts by studying a production cluster that ingests more than 10 PB per day over several 100K machines, showing the presence of provisioning needs that are difficult to predict the potential benefit of fine-grained resource provisioning. The core of this framework is translating a job's performance targets into priorities of individual messages, which in turn, helps the streaming engine to accurately identify and prioritize performance-critical messages. The developed techniques dynamically derive priorities of operators using both static input (e.g., job deadline) and dynamic stimulus (e.g., tracking stream progress, profiled message execution times). A scheduling framework is built to incorporate mechanisms like scheduling contexts, a context handling interface for pluggable scheduling strategies, and an interface for real-time scheduling policies like Earliest Deadline First (EDF) and Least Laxity First (LLF). In the continuing work (*Dirigo*), I explored how to enable controller-less auto-scaling for the fine-grained processing paradigm proposed in my previous work. This project proposes a **self-scaling actor model that dynamically parallelizes computation within a dataflow,** and a set of communication protocols that

ensure the auto scaling processes could be carried out **without compromising consistency of computational states**. Another continuing work (Banyan) explores an alternative actor-model design that exposes hierarchical parallel execution patterns. It proposes scoped dataflow, an **actor-based dataflow model that captures parallelizable execution scope in the dataflow graph.** Scoped dataflow model significantly improves efficiency of execution management for workload that involves hierarchical function invocations in nature (e.g. graph exploration) while ensuring performance isolation between concurrent queries effectively.

This line of work addresses a critical gap in resource provisioning and performance isolation through exploring the granularity of resource provisioning. The conventional slot-based approach often leads to resource over-provisioning and underutilization, which not only escalates costs for users but also diminishes the energy efficiency of data centers. By leveraging the virtual actor model and reducing granularity of management of today's cloud workload, these works facilitate fine-grained resource sharing among operators directly. The priority-based scheduling framework developed in my project enables stream processing systems to automatically adjust resources provisioned on a per-message basis with performance target awareness. The experimental results, showcasing up to 4.6 times in query latency and the ability to efficiently handle transient workload spikes, underscore the potential of this approach in significantly advancing the state of resource provisioning in cloud-based stream processing engines. This not only **alleviates the burden on users to manually configure resources** but also **substantially enhances resource utilization and energy efficiency in data centers**. Moreover, the following project keeps exploring how to **enable efficient, state-consistent parallelization that meets service level objectives in multi-tenant stream processing systems**.

Project 4: Large Memory Analytics in Scale-up Setting

List of publications and preprints:

- Wang, Wenting, Le Xu, and Indranil Gupta. "Scale Up vs. scale out in cloud storage and graph processing systems." In *2015 IEEE International Conference on Cloud Engineering*, pp. 428-433. IEEE, 2015.
- Kim, Mijung, Jun Li, Haris Volos, Manish Marwah, Alexander Ulanov, Kimberly Keeton, Joseph Tucek, Lucy Cherkasova, Le Xu, and Pradeep Fernando. "Sparkle: optimizing spark for large memory machines and analytics." In *Proceedings of the 2017 Symposium on Cloud Computing*, pp. 656-656. 2017.

My research primarily focuses on optimizing resource provisioning in distributed systems deployed in the cloud. Given the challenges faced by deployers, such as budget constraints or throughput requirements, I delved into understanding the cost-efficiency of deploying cloud applications either in a scale-out cluster or a single scale-up machine. By studying the performance of a key-value store in both settings, I applied a linear regression model to map virtual machine configurations to costs provided by major public cloud providers. This allowed me to approximate the cost-efficiency of running specific workloads on various VM configurations. My findings highlighted scenarios where one option was preferable over the other. For instance, users with resource-intensive workloads could achieve better cost-efficiency using a scale-up setting. Furthermore, I explored the performance of the widely adopted memory-centric data analytics framework, Spark, in both scale-out and scale-up settings. By introducing a memory mapping optimization that leverages large, shared memory, I found that the scale-up configuration, in most cases, outperformed the scale-out configuration due to faster in-memory data access. This research also revealed that for certain operations, a larger cluster doesn't necessarily equate to better performance, emphasizing the need for a more tailored approach to resource provisioning.

The implications of my work are profound for the realm of cloud computing and distributed systems. Currently, the myriad of machine choices offered by cloud providers makes it daunting for end-users to select the most cost-effective resources. This often results in suboptimal cloud deployments, leading to either unmet performance targets or inflated deployment costs. My research offers structured guidance, enabling users to make informed decisions when choosing between scale-up and scale-out configurations. Moreover, the revelation that many cloud-based frameworks today aren't optimized for scale-up settings underscores a significant gap in the field. By redesigning cloud engines to better utilize scale-up memory-centric architecture, my work paves the way for significant improvements in cloud resource efficiency. This not only has the potential to **reduce costs for end-users** but also

promotes **more sustainable and efficient use of cloud resources**, driving innovation and sustainability in the ever-evolving landscape of cloud computing.

Current and Future Projects

List of publications and preprints:

- Peter Schafhalter, Sukrit Kalra, Le Xu, Joseph E. Gonzalez, and Ion Stoica. "Leveraging Cloud Computing to Make Autonomous Vehicles Safer." 2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2023).
- Bodun Hu, Le Xu, Jeongyoon Moon, Neeraja J. Yadwadkar, Aditya Akella. "MOSEL: Inference Serving Using Dynamic Modality Selection."

My past research focused on real-time data processing applications that are predominantly deployed in pure cloud environments, operating on homogeneous hardware where the processing semantics of operators are either well-defined or thoroughly studied. However, with the surge in artificial intelligence (AI) applications and their integration into numerous data processing pipelines, new challenges have emerged. In response, my future research will address two primary areas: improving system elasticity with environmental heterogeneity and building end-to-end AI pipelines with semantic-guided modularity. The cloud computing industry, projected to witness exponential growth, will be better positioned to harness this growth efficiently by addressing system elasticity in diverse environments. Furthermore, efficient AI pipelines can lead to cost savings, faster innovations, and enhanced competitiveness in the global AI market. This research aligns with the technology industry's aspirations, which aims to **develop effective methods for AI-human interactions,** as well as. **ensure the safety and security of AI systems**.

1. **System elasticity with environmental heterogeneity**

In the rapidly evolving landscape of cloud computing and real-time data processing, the need to address system elasticity in heterogeneous environments has become paramount. My previous research predominantly focused on enhancing system elasticity within homogeneous hardware settings, such as CPU and memory resources, in high-speed networked data centers. However, the real-world application of cloud-based real-time data processing, especially when intertwined with AI components like ML pipelines, demands a more intricate execution environment. My future research endeavors will pivot toward two pivotal areas:

- **Elastic Systems for Heterogeneous Hardware**: Machine learning (ML) pipelines are the computational foundations for almost all AI applications nowadays. Modern ML pipelines often incorporate elements like neural networks that necessitate specialized hardware, such as GPUs and TPUs, for timely results. This introduces complexities in designing elastic systems, as it's crucial to understand the performance characteristics of executing pipeline components across varied hardware. A significant portion of my research will be dedicated to ML inference on multi-modal data, where different data types (e.g., image, video, audio) are processed by a unified model for predictions. Each modality presents unique resource consumption profiles and contributes differently to prediction accuracy. My goal is to optimize configurations for processing these modalities on GPUs in real-time, adhering to latency targets and maintaining minimum prediction accuracies. This research can be expanded to encompass ML models with diverse accuracy profiles operating on assorted hardware.

- **Elastic Systems for Heterogeneous Environments**: The proliferation of edge devices and IoT infrastructures has transformed the data collection landscape. These devices, now more powerful than ever,

are increasingly processing data locally, reducing the data influx to data centers. This shift prompts the question: How should stream processing systems be reimagined to scale applications seamlessly across the modern cloud infrastructure hierarchy, from edge devices to data centers? My research will delve into intricate real-time data processing pipelines, such as those used in autonomous driving, to harness the potential of elastic scaling and optimize cloud resource utilization. By enhancing elasticity across the cloud-edge continuum, devices like autonomous vehicles can access cloud resources more efficiently, ensuring faster ML inferences with updated models and delivering higher accuracy results. This not only elevates driving safety but also enriches the overall driving experience. Furthermore, the real-time data from these devices offers a dynamic, real-time perspective of the world, paving the way for innovative applications and scenarios.

Optimizing cloud resources and enhancing real-time data processing capabilities can lead to significant economic benefits, improved safety standards in industries like autonomous driving, and the creation of new market opportunities beyond data centers such as edge intelligence. Efficiently harnessing this growth by addressing system elasticity in heterogeneous environments can significantly reduce time and complexity running AI-centric applications on end devices, bringing the benefit of advanced AI algorithms to everyday life.

2. **End-to-end AI pipeline with semantic-guided modularity**

My future research endeavors will focus on enhancing the modularity and efficiency of end-to-end AI pipelines. Drawing from my previous findings, I've discovered that the optimal granularity of provisioning is pivotal for resource efficiency in data processing pipelines. While this principle has been applied to many existing frameworks, especially those handling real-time workload changes, it hasn't been fully explored in the realm of ML-driven applications. These applications, predominantly driven by inferences from ML models, often operate as black boxes, making their semantics elusive to users. This obscurity hinders granular provisioning, thereby bypassing potential optimizations like operator sharing, replacement, and parallelization. The number of AI-related publications grows exponentially [2], resulting in a large number of publicly accessible ML models: HuggingFace [3], one of the major providers hosting open-sourced ML models, supports over 120,000 models alone [4]. As a result, selecting, optimizing, and deploying the best combination of models from these model pools to perform a task becomes a daunting task for both human users and autonomous agents driven by AI. This means there is an imperative need to delve deeper into model semantics by supporting a method that formalizes the scope of problems addressed by different models, gauging overlaps in functionalities between models, and assessing the implications of using alternative models for specific tasks. Efficient, modular AI pipelines can lead to cost savings and faster innovations in future research driven by AI-based applications. Moreover, with the burgeoning interest in using large language models (LLMs) as autonomous agents based on natural language requests, as highlighted by [5][6], integrating model semantics into LLMs could revolutionize the way we approach AI tasks.

References:

[1] Wang, Yuanli, Baiqing Lyu, and Vasiliki Kalavri. "The non-expert tax: quantifying the cost of auto-scaling in cloud-based data stream analytics." *Proceedings of The International Workshop on Big Data in Emergent Distributed Environments*. 2022.

[2] Artificial Intelligence Index Report 2023

[3] HuggingFace. https://huggingface.co.

[4] Hugging Face Hub documentation. https://huggingface.co/docs/hub/index.

[5] Wang, Lei, et al. "A survey on large language model based autonomous agents." arXiv preprint arXiv:2308.11432 (2023).

[6] Autonomous AI and Autonomous Agents Market Share & Size, Global Trends, Statistics, Growth Forecast. MarketsandMarkets.