

## Systems and Network Projects at UW

1. Arrakis, an operating system for datacenters: A refactored operating systems that provides applications direct access to I/O devices without compromising security and isolation. <https://arrakis.cs.washington.edu/>
2. Sapphire, a distributed programming platform for mobile/cloud applications: Sapphire removes much of the complexity of managing a wide-area, multi-platform environment, yet still provides developers with the fine-grained control needed to meet critical application needs. <https://sapphire.cs.washington.edu/research/project/sapphire.html>
3. Agate – user-driver privacy for mobile/cloud applications: Agate is a new trusted distributed runtime system that gives users control over how mobile/cloud applications share sensitive user data collected on mobile devices (e.g., photos, GPS location). <http://sapphire.cs.washington.edu/research/project/agate.html>
4. Tapir, a high performance distributed storage system: Tapir provides linearizable distributed transactions using a light-weight replication mechanism with *no consistency guarantees*. By enforcing consistency *only* at the transaction layer, TAPIR eliminates coordination at the replication layer, enabling TAPIR to provide the same consistency guarantees as existing systems like Spanner with better latency and throughput. <http://irenezhang.net/research/tapir/index.html>
5. Datacenter networks:
  - a. Reliability: F10 is a novel network topology that has many of the same desirable properties as existing datacenter networks, but with much better fault recovery properties. We then create a series of failover protocols that benefit from this topology and are designed to cascade and complement each other. <http://homes.cs.washington.edu/~arvind/papers/failover.pdf>
  - b. Performance: We have developed a new approach to wiring ToR switches to reduce congestion at that layer with datacenters. Our approach, called Subways, takes advantage of the trend towards multiple network interfaces per server machine, by wiring servers to the ToR switches in adjacent racks. <ftp://ftp.cs.washington.edu/tr/2014/09/UW-CSE-14-09-01.pdf>
6. Web performance: Web page load time is a key performance metric that many companies and people are trying to optimize. We have developed a tool that extracts dependencies during the page load and use it to optimize web page loads. <http://wprof.cs.washington.edu/>
7. Internet reliability: A chronic problem with the Internet is that it is vulnerable to outages, black holes, hijacking, and denial of service. We have developed a simple, easy to implement, routing solution that is sufficient for addressing the aforementioned Internet vulnerabilities. <https://faculty.washington.edu/simpeter/sigc150-peterA.pdf>
8. Predictable low-latency systems: Interactive services often have large-scale parallel implementations. To deliver fast responses, the median and tail latencies of a service's components must be low. In this project, we explore the hardware, OS, and application-level sources of poor tail latency in high throughput servers executing on multi-core machines. <http://drkp.net/papers/latency-socc14.pdf>

## Systems & Networking @UW

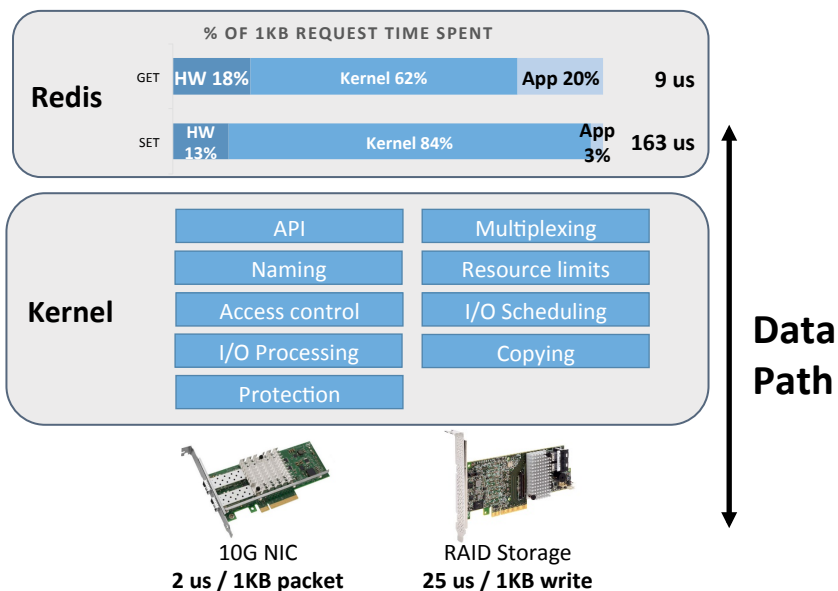
### Research Areas

- Datacenter operating systems
- Mobile/cloud programming platforms
- Datacenter networks
- High performance distributed storage
- Predictable low-latency systems
- Internet reliability and performance
- Web performance
- Secure & privacy-preserving systems  
and more...

## OS for the Data Center

- Server I/O performance matters
  - Key-value stores, web & file servers, lock managers
- Devices (network and disk) are becoming fast
  - 10G NICs, multiple NICs/node
  - 20G SSDs for datacenters
  - Need to process a packet/block every 1-2us
- OS is once again a bottleneck!

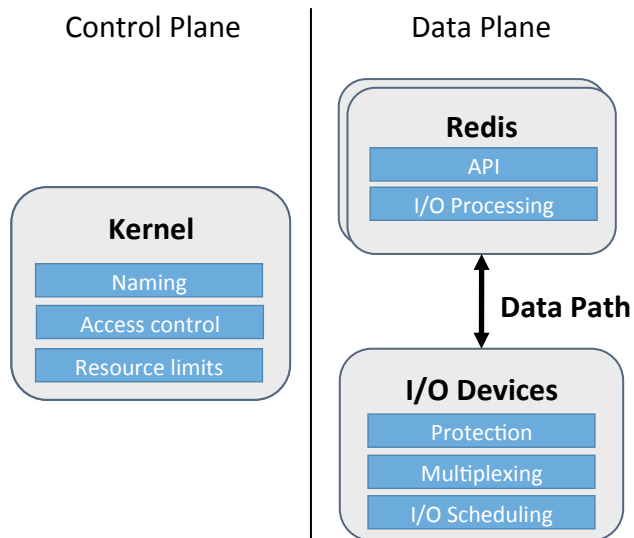
## Linux I/O Performance



## Arrakis: Datacenter OS

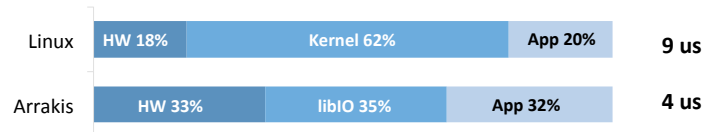
- Expect even more from hardware!
- Leverage five hardware innovations
  - Device virtualization
  - Intelligent devices
  - Programmable packet processing
  - Tighter integration between devices and CPUs
  - Hardware support for isolation
- Re-architect the OS to take advantage of these HW features
  - Split OS into control/data plane

## Arrakis I/O Architecture

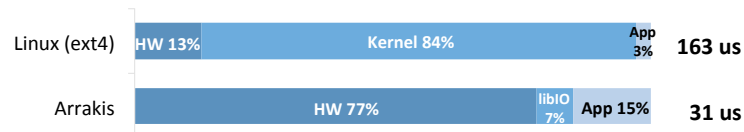


## Redis Latency

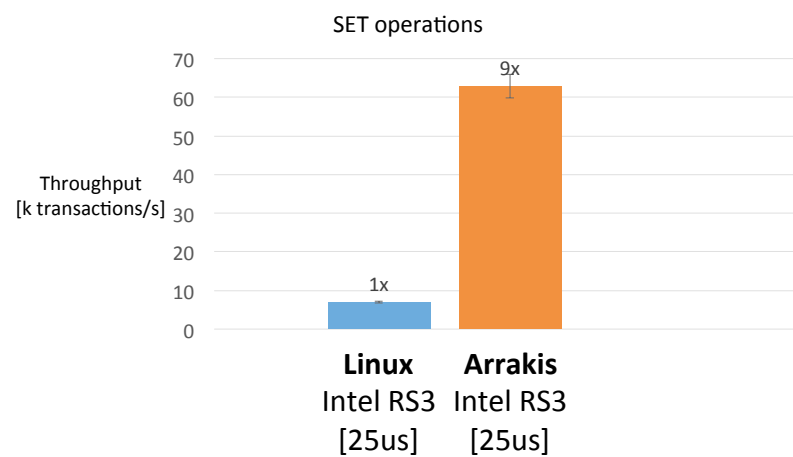
- Reduced (in-memory) GET latency by **65%**



- Reduced (persistent) SET latency by **81%**



## Redis Throughput



## Future Directions

- Simultaneously constrict and expand the OS
  - More tasks performed by HW, more control to the application
  - OS is a **datacenter-wide** management plane
- Allow applications to exert fine-grained control over HW
  - Steer packets to memory/cores, perform processing in HW
  - Application-specific network protocol
  - Develop an alternative to the POSIX API
  - Rethink distributed systems when networking and persistent memory are both very fast
- OS does global resource allocation; rethink congestion control, scheduling, etc.

## Sapphire: Programming System for Mobile/Cloud Applications

- Mobile/cloud applications implement difficult distributed deployment tasks
  - What state should be cached? Should objects be replicated?
  - What tasks should be offloaded and when? How to hide performance problems and failures? ...
- Sapphire: a new system architecture that supports pluggable and extensible deployment managers
  - Programmer partitions application into Sapphire objects, which are units of distributed deployment decisions
  - Deployment managers are useful for both:
    - Application programmers to choose deployment
    - Systems programmers to build new deployment patterns

## **Agate: User-driven Privacy for Mobile/Cloud Applications**

- Privacy guarantees for today's mobile/cloud applications depend on the programmers
  - Bugs result in accidental privacy leaks
  - Malicious applications exfiltrate sensitive data
- Can we do better? Leverage:
  - Increasing number of OS protected resources
  - Global identity management services
  - Devices are typically single-user
- Agate: trusted runtime system for mobile/cloud apps
  - Gives users the power to define privacy policies
  - Enforces these policies without needing to trust apps