

# Concurrency

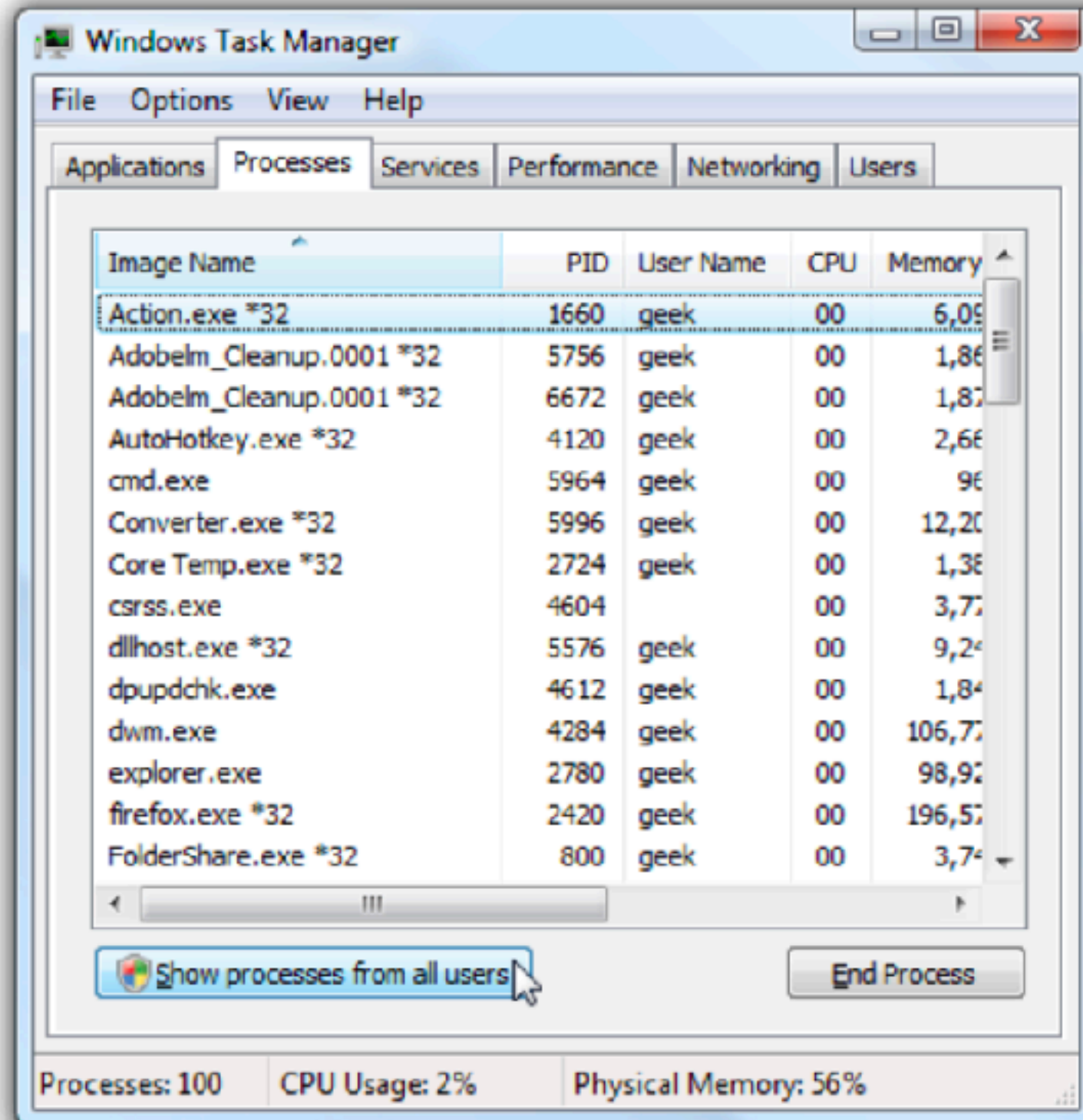
CS 242

November 6, 2017

# Today's goals

- **Concurrency basics**
- **Problems with existing abstractions**
- **The Rust Solution™**

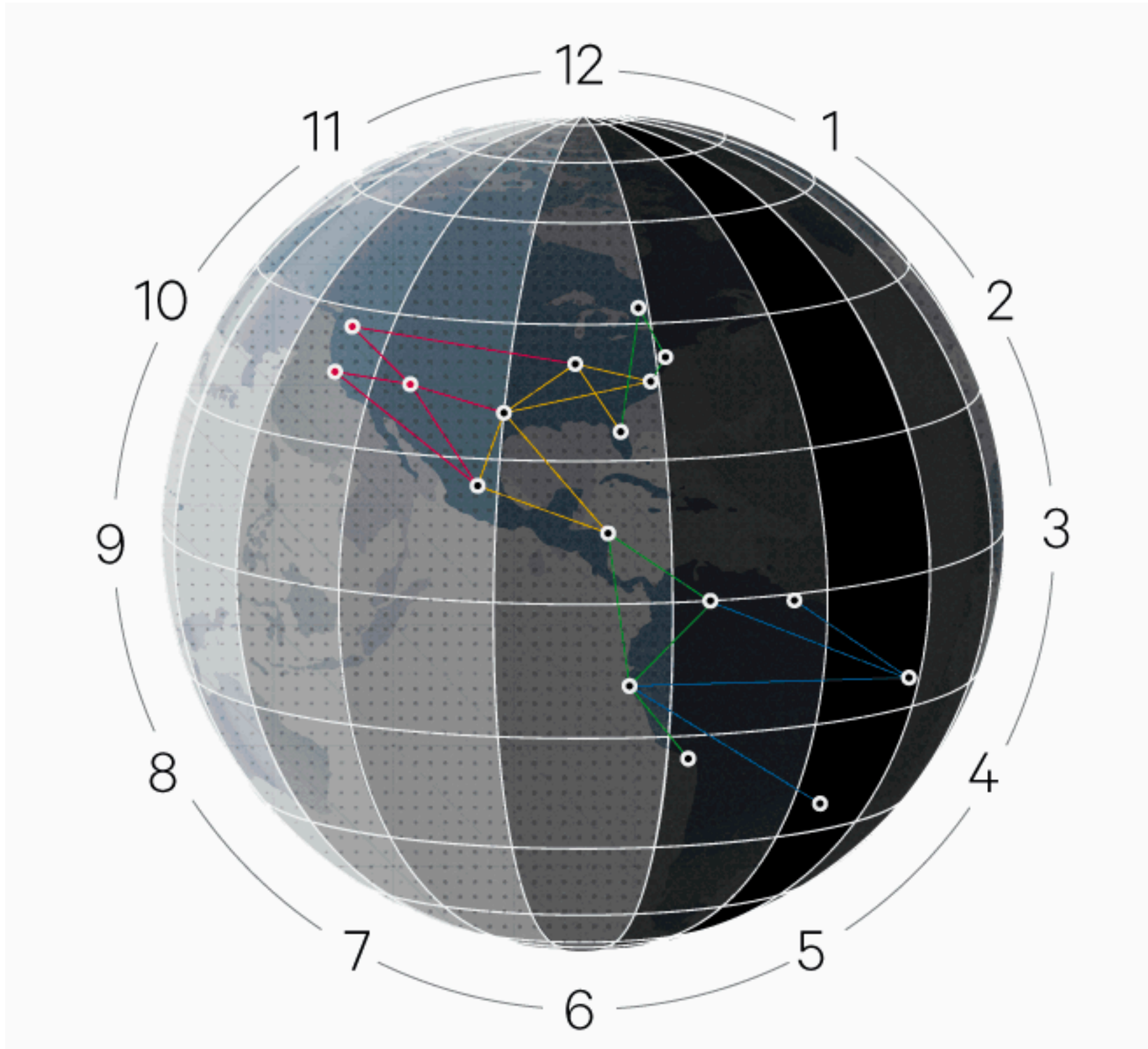
# Doing many things simultaneously



# Doing many things simultaneously



# Doing many things simultaneously





# Concurrency

*Competitive*

**vs.**

# Parallelism

*Cooperative*

# Concurrency model

- Thread: logically independent sequential program, shared access to resources (e.g. memory)
- Process: group of threads, limited access to other processes
- Scheduler: allocates execution resources to threads (e.g. CPU)

# Preemptive vs. cooperative scheduling

- Preemptive: scheduler decides when to change allocation
- Cooperative: threads decide when to change allocation (coroutines!)



# Synchronization is the problem

- **Communication: sharing data between threads**
  - Shared memory
  - Message passing
- **Coordination: refereeing access to data**
  - Locks
  - Condition variables
  - ...many more!

