1. My approach: bring photonics into the computing socket **5 minutes**
   1. What is the approach and how is it different from existing?
      1. My approach reduces distance between electronic and photonic interface
      2. In today’s solution this distance is long
   2. Why is this long distance bad? Look at some numbers:
      1. Link energy goes up for longer distances
      2. Link bandwidth goes down in larger systems
   3. What improvement does my approach achieve? 100x over state-of-the-art
2. Grand challenge: communication bottleneck (others have referred to it as … wall) **5 minutes**
   1. What is the bottleneck?
      1. High energy low bandwidth 🡪 computation slow-down
      2. Example applications affected
   2. How severe is the problem?
      1. Energy cost of a single computation task > the entire NYC electricity usage in a hot summer day
      2. And it takes >3 months to finish the task
   3. What can be achieved if the problem is solved?
      1. Computation continues to scale, but energy brought down by ~100x
3. My work using this approach: co-packaged optical data input/output **15 minutes**
   1. The package, the performance numbers
   2. What’s inside the package: chips
      1. My chip and its design
      2. Detailed link design on my chip: supporting >64 channels vs. today’s <16
         1. Comb source
         2. Transmitter
         3. Receiver
         4. Scalable link architecture that achieved >64 channels
         5. Measurement results
      3. How to control the link
         1. Automatic interleaver control
         2. Microresonator control
   3. How are the chips integrated: 3D integration and advanced packaging
   4. What’s the energy consumption and how is it calculated
4. My other work extending this approach: use photonics for computing **10 minutes**
   1. What does the system look like
   2. Why do I want to do computing within data movement
   3. How does the system work
5. Future directions **5 minutes**

**40 minutes**