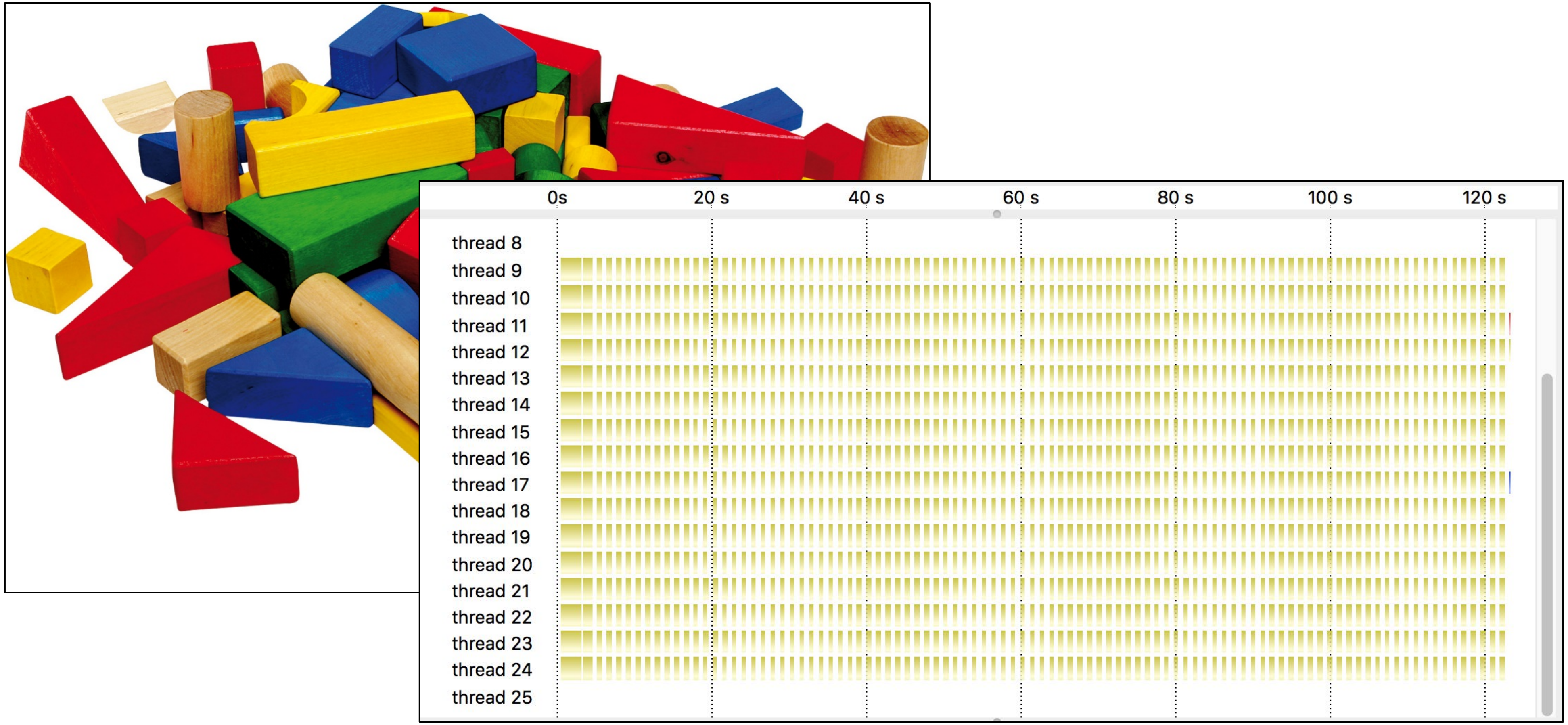
Asynchronous C++

Hartmut Kaiser (hkaiser@cct.lsu.edu)

Today’s Parallel Applications



HPX – The C++ Standards Library for Concurrency and Parallelism

- Exposes a coherent and uniform, standards-oriented API for ease of programming parallel, distributed, and heterogeneous applications.
  - Enables to write fully asynchronous code using hundreds of millions of threads.
  - Provides unified syntax and semantics for local and remote operations.
- Enables using the Asynchronous C++ Standard Programming Model
  - Emergent auto-parallelization, intrinsic hiding of latencies,

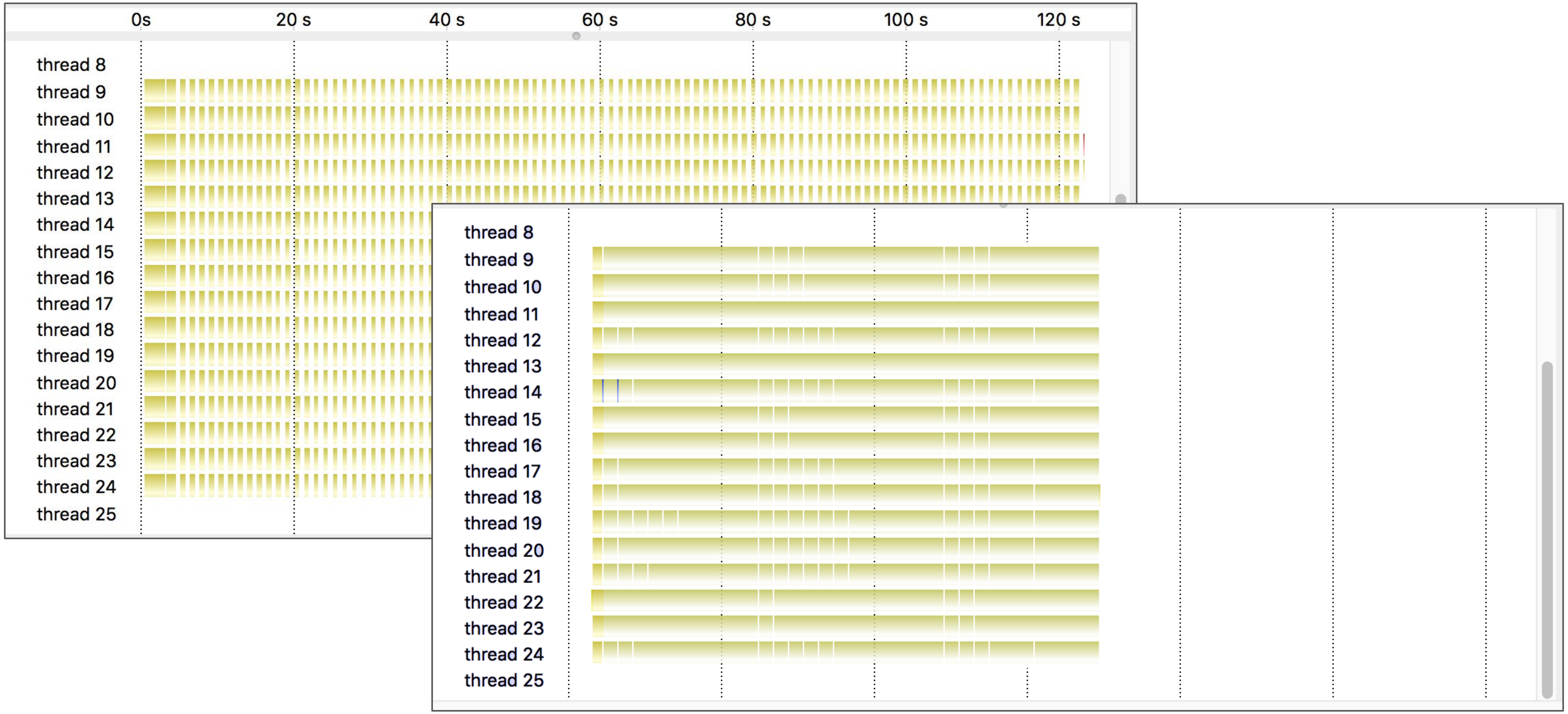
STE||AR GROUP

HPX – The API

- As close as possible to C++ standard library, where appropriate, for instance

std::thread, std::jthread	hpx::thread (C++11), hpx::jthread (C++20)
std::mutex	hpx::mutex
std::future	hpx::future (including N4538, ‘Concurrency TS’)
std::async	hpx::async (including N3632)
std::for_each(par, …), etc.	hpx::for_each (C++17)
std::latch, std::barrier	hpx::latch, hpx::barrier
std::experimental::task_block	hpx::experimental::task_block (TS V2)
std::experimental::for_loop	hpx::experimental::for_loop (TS V2)
std::bind	hpx::bind
std::function	hpx::function
std::any	hpx::any (C++20)
std::cout	hpx::cout

The Solution to the Application Problem



The Solution to the Application Problems

LSU CENTER FOR COMPUTATION & TECHNOLOGY

STE||AR GROUP