Next Generation Computing

Supercomputing Challenge Kickoff
Oct 11-12, NM Tech, Socorro, NM

Bob Robey[1] and Matthew Curry[2]
[1]Los Alamos National Laboratory
[2]Sandia National Laboratories
[1,2]Supercomputing Challenge Board of Directors

LA-UR-14-27945

Topics

- Cloud Supercomputing
  - Mobile
- GPU Computing
  - IOS, Android
  - Hobbyist devices – Jetson TK1, Parallella
- MIC – Intel accelerator

Why these topics?

- To help you develop skills that will be in high demand
- Previous years we focused on parallel computing – it has been expanded to include a wider array of new technologies
- Lay out skill path needed
- How do you start?

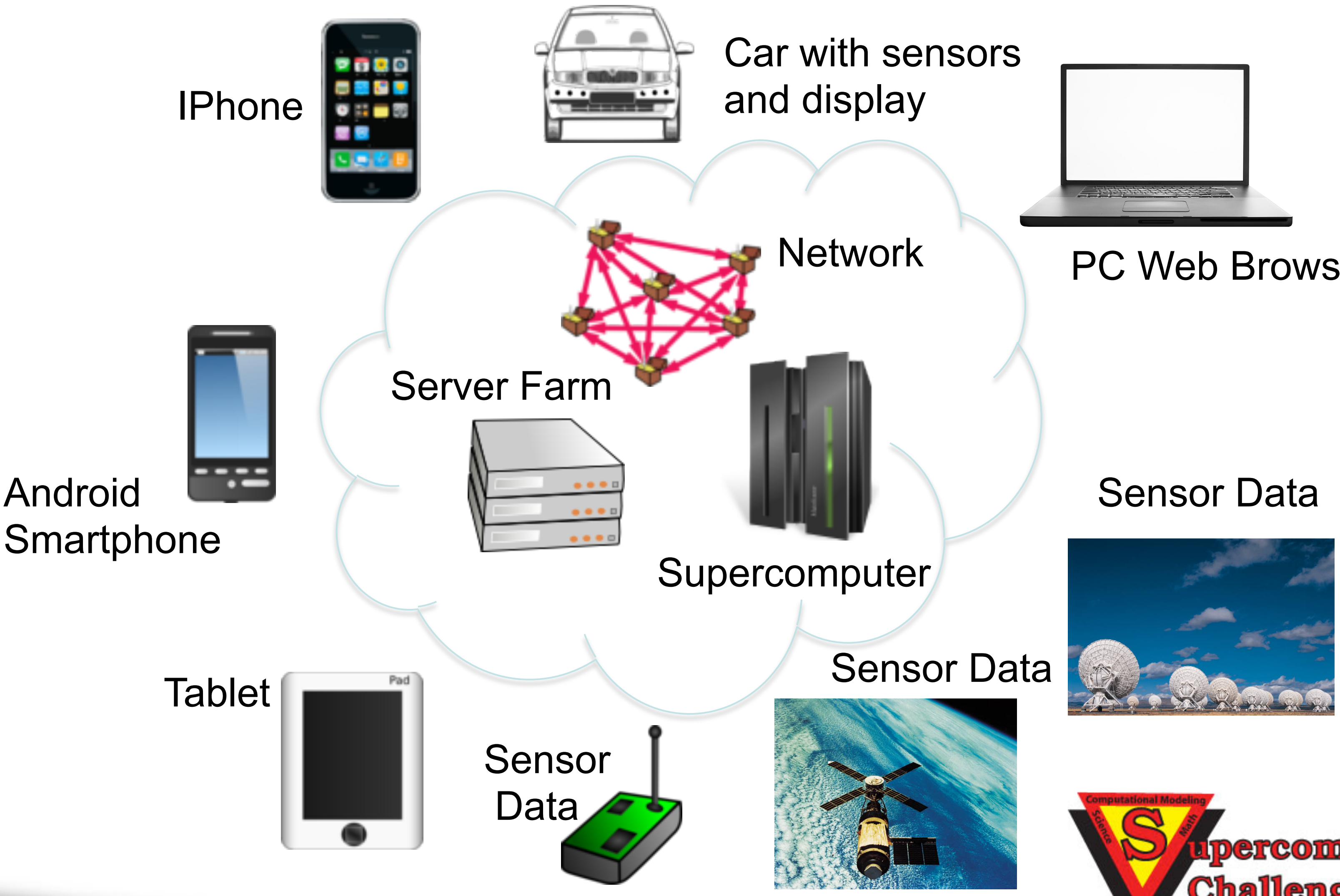
Cloud Supercomputing

Weather simulation is a prototypical example. Data is gathered from many distributed sources and fed into cloud servers. Then a simulation is run on a supercomputer. Lastly the weather is distributed via mobile and web.

Another example is waze.com, a traffic app that gathers data from mobile devices and recalculates best routes based on traffic and accident data that is input from other people. What are the possible components?_

- Distributed data collection
- Large network
- "Cloud" database on large server farm
- Supercomputer
- Data dissemination and local compute

Cloud Supercomputing



IPhone

Car with sensors and display

PC Web Browser

Network

Server Farm

Android Smartphone

Supercomputer

Sensor Data

Tablet

Sensor Data

Sensor Data

Computational Modeling Science Math Supercomputing Challenge

Cloud Supercomputing Exercise

Pick a favorite application.

- What components of cloud supercomputing are used in this application?
- Could some of the other elements be used to enhance the application?
- Where is the computing done and why?

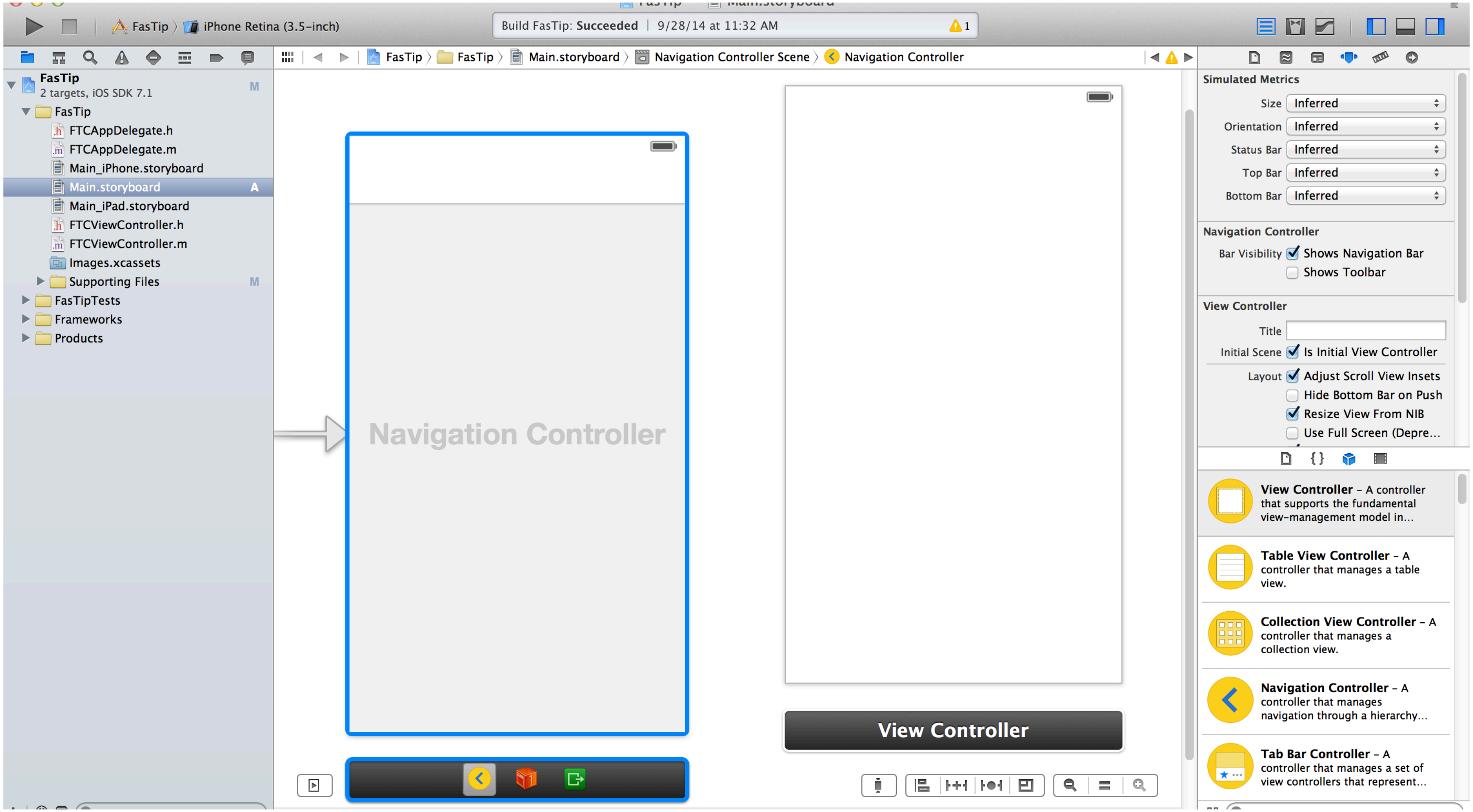
Mobile Computing

- Use an Internet-connected smartphone to provide and gather data anywhere
- Phones will often include networking, gyroscopes, GPS, two cameras, microphone, accelerometer, compass, light sensor, pressure sensor

Apple/iOS

- Free software development kit (SDK) and emulator on a Mac
  - https://developer.apple.com/technologies/ios/
- $100/yr Developer Account (necessary for deploying on a device)
- Only one programming language, Objective C (but being replaced by Swift)
  - https://developer.apple.com/library/mac/documentation/cocoa/conceptual/ProgrammingWithObjectiveC/Introduction/Introduction.html
- Uses Storyboarding to build app

Mac Xcode Storyboard



FasTip 〉 iPhone Retina (3.5-inch)
Build FasTip: **Succeeded** | 9/28/14 at 11:32 AM
⚠ 1
FasTip 〉 FasTip 〉 Main.storyboard 〉 Navigation Controller Scene 〉 Navigation Controller
Navigation Controller
View Controller

- FasTip
  - 2 targets, iOS SDK 7.1 M
- FasTip
  - FTCAppDelegate.h
  - FTCAppDelegate.m
  - Main_iPhone.storyboard
  - Main.storyboard A
  - Main_iPad.storyboard
  - FTCViewController.h
  - FTCViewController.m
  - Images.xcassets
  - Supporting Files M
- FasTipTests
- Frameworks
- Products

Simulated Metrics

Size	Inferred
Orientation	Inferred
Status Bar	Inferred
Top Bar	Inferred
Bottom Bar	Inferred

Navigation Controller

Bar Visibility ☑ Shows Navigation Bar
□ Shows Toolbar

View Controller

Title

Initial Scene ☑ Is Initial View Controller

Layout ☑ Adjust Scroll View Insets
□ Hide Bottom Bar on Push
☑ Resize View From NIB
□ Use Full Screen (Depre…

View Controller – A controller that supports the fundamental view-management model in…

Table View Controller – A controller that manages a table view.

Collection View Controller – A controller that manages a collection view.

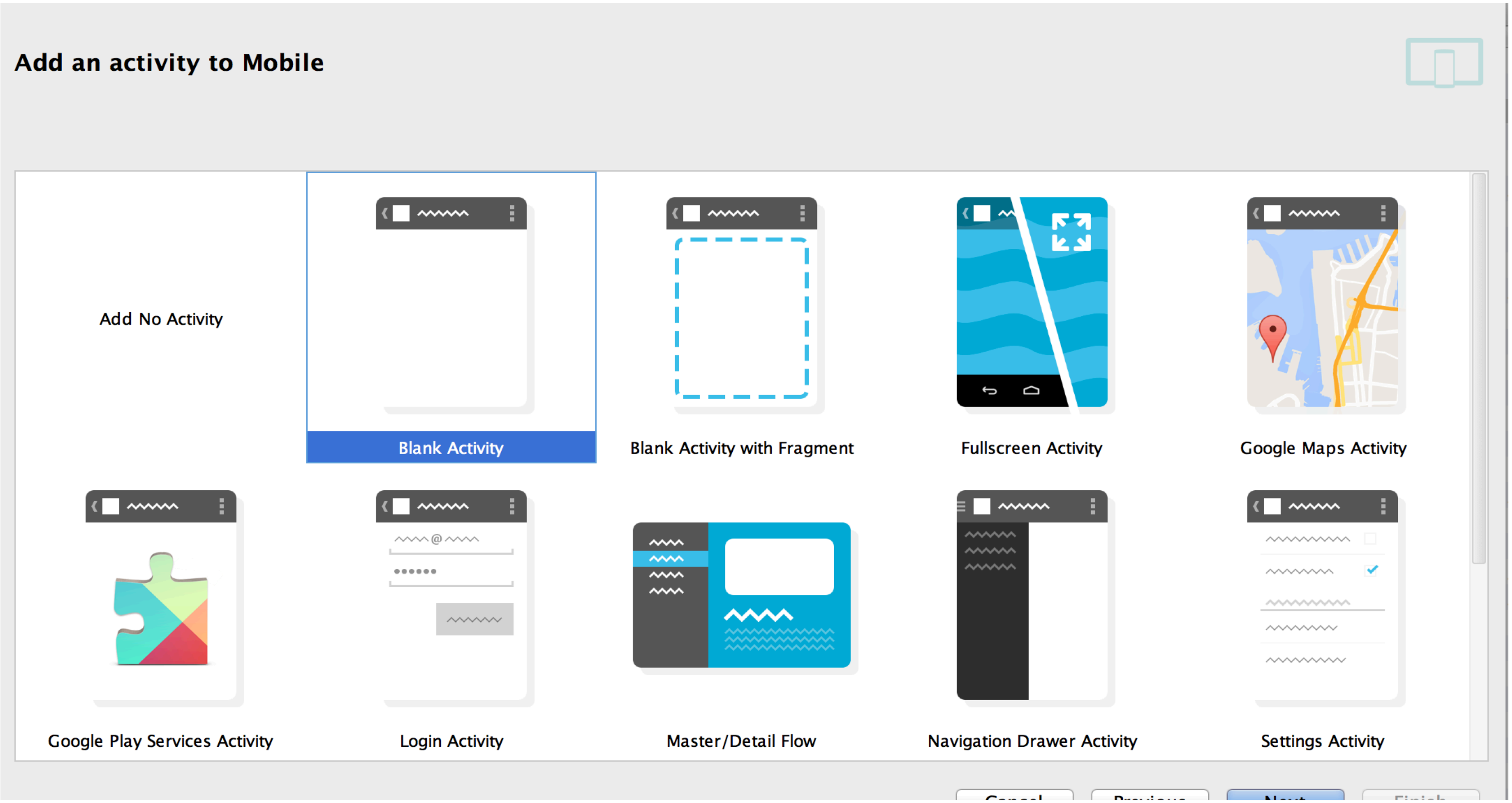
Navigation Controller – A controller that manages navigation through a hierarchy…

Tab Bar Controller – A controller that manages a set of view controllers that represent…

Android

- Free SDK and emulator for Mac, PC, or Linux: http://developer.android.com/sdk/index.html
- Free deployment on devices (sideloading)
- Plethora of programming languages
  - Java via official SDKs: http://developer.android.com/training/basics/firstapp/
  - Google/MIT App Inventor: http://appinventor.mit.edu/explore/
  - Python: http://kivy.org/
  - BASIC (http://laughton.com/basic/)
  - Processing (http://processing.org, https://code.google.com/p/ketai/)

Android Storyboarding



Add an activity to Mobile

Add No Activity

Blank Activity

Blank Activity with Fragment

Fullscreen Activity

Google Maps Activity

Google Play Services Activity

Login Activity

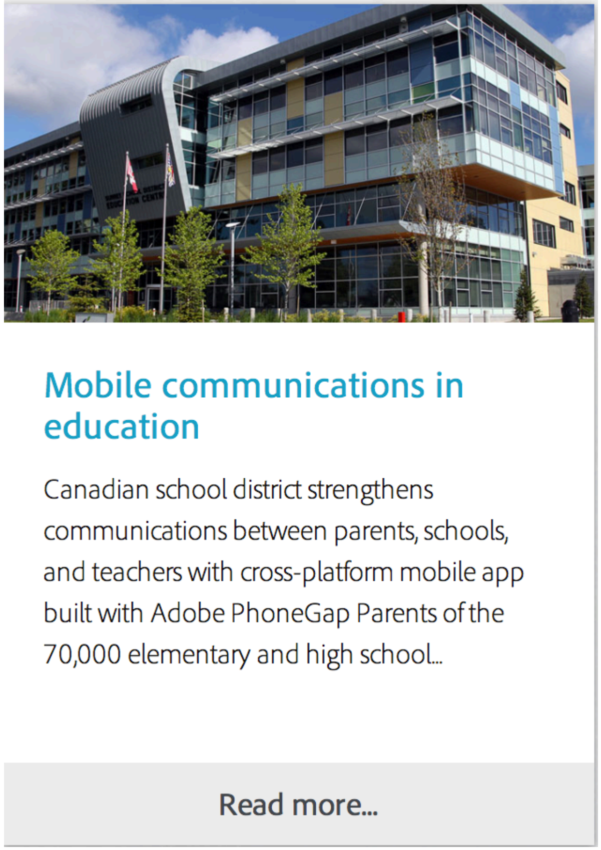
Master/Detail Flow

Navigation Drawer Activity

Settings Activity

Supercomputing Challenge

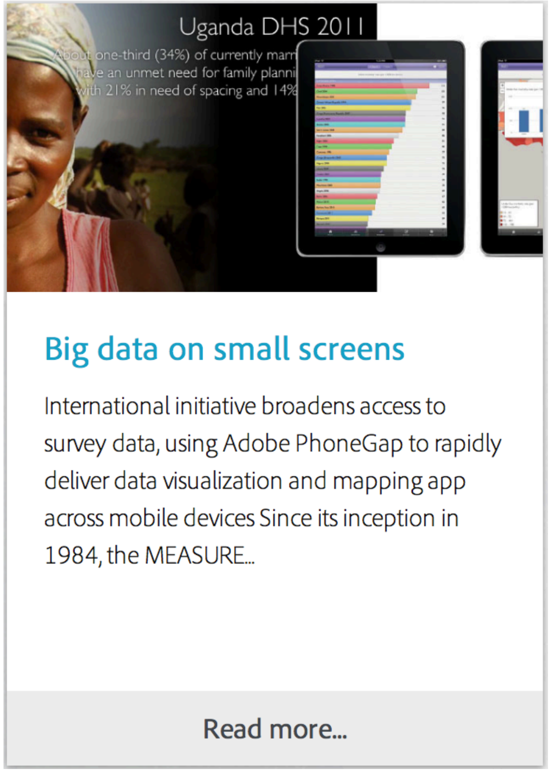
Cross-platform -- PhoneGap



Mobile communications in education

Canadian school district strengthens communications between parents, schools, and teachers with cross-platform mobile app built with Adobe PhoneGap Parents of the 70,000 elementary and high school...

Read more...



Uganda DHS 2011

About one-third (34%) of currently marr
have an unmet need for family planni
with 21% in need of spacing and 14%

Big data on small screens

International initiative broadens access to survey data, using Adobe PhoneGap to rapidly deliver data visualization and mapping app across mobile devices Since its inception in 1984, the MEASURE...

Read more...

Software Developer Wunderkind Opts for PhoneGap, All the Cool Kids are Doing It!

Should we be worried that a nine-year-old can explain how to interface with the 31-pin sync port, or that he's got an opinion on Objective-C? Nah. Especially when the software development...

Read more...

Computational Modeling Science Math
Supercomputing Challenge

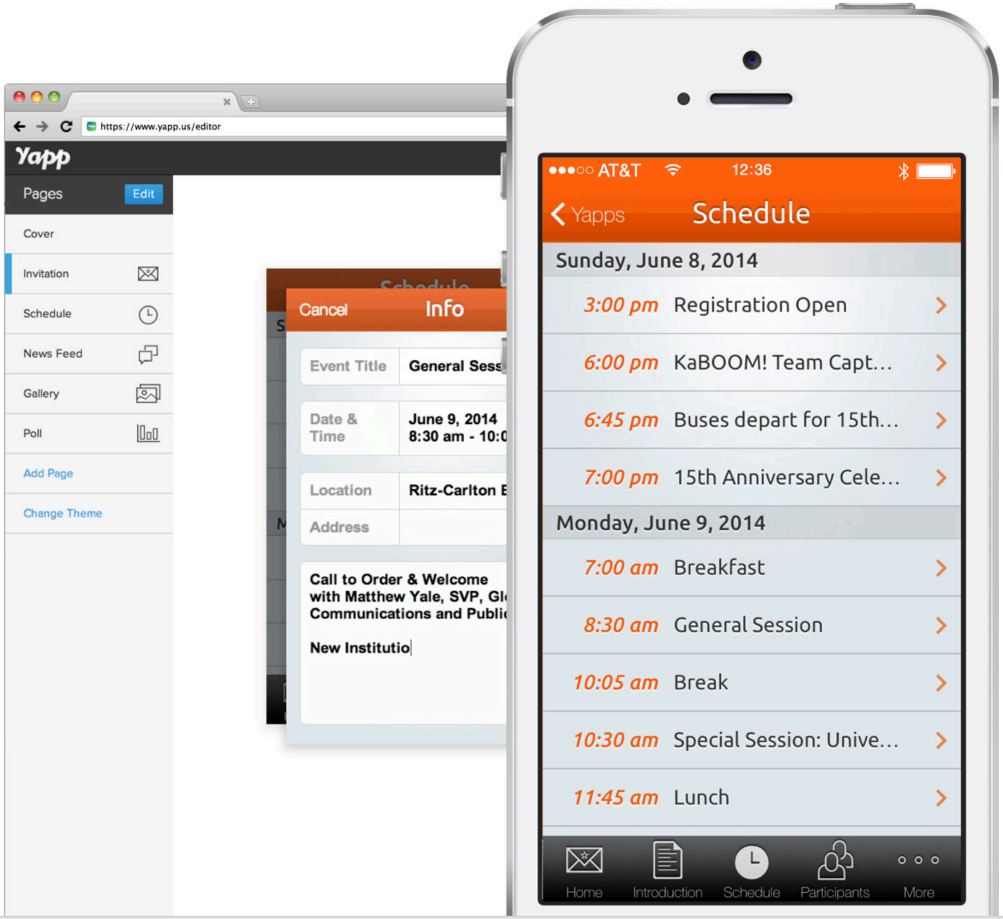
Cross-platform -- Titanium

- Produces native looking apps on each device
  - http://www.smashingmagazine.com/2014/03/10/4-ways-build-mobile-application-part4-appcelerator-titanium/
- Many others are being announced

Computational Modeling Science Math Supercomputing Challenge

Non-programming Mobile Apps

- SC Kickoff Conference App was created using Yapp (yapp.us)
- Many others exist or are being announced



Event Title: General Sess

Date & Time: June 9, 2014 8:30 am - 10:0

Location: Ritz-Carlton E

Address:

Call to Order & Welcome with Matthew Yale, SVP, Gl Communications and Publi

New Institutio

Sunday, June 8, 2014

3:00 pm	Registration Open
6:00 pm	KaBOOM! Team Capt...
6:45 pm	Buses depart for 15th...
7:00 pm	15th Anniversary Cele...

Monday, June 9, 2014

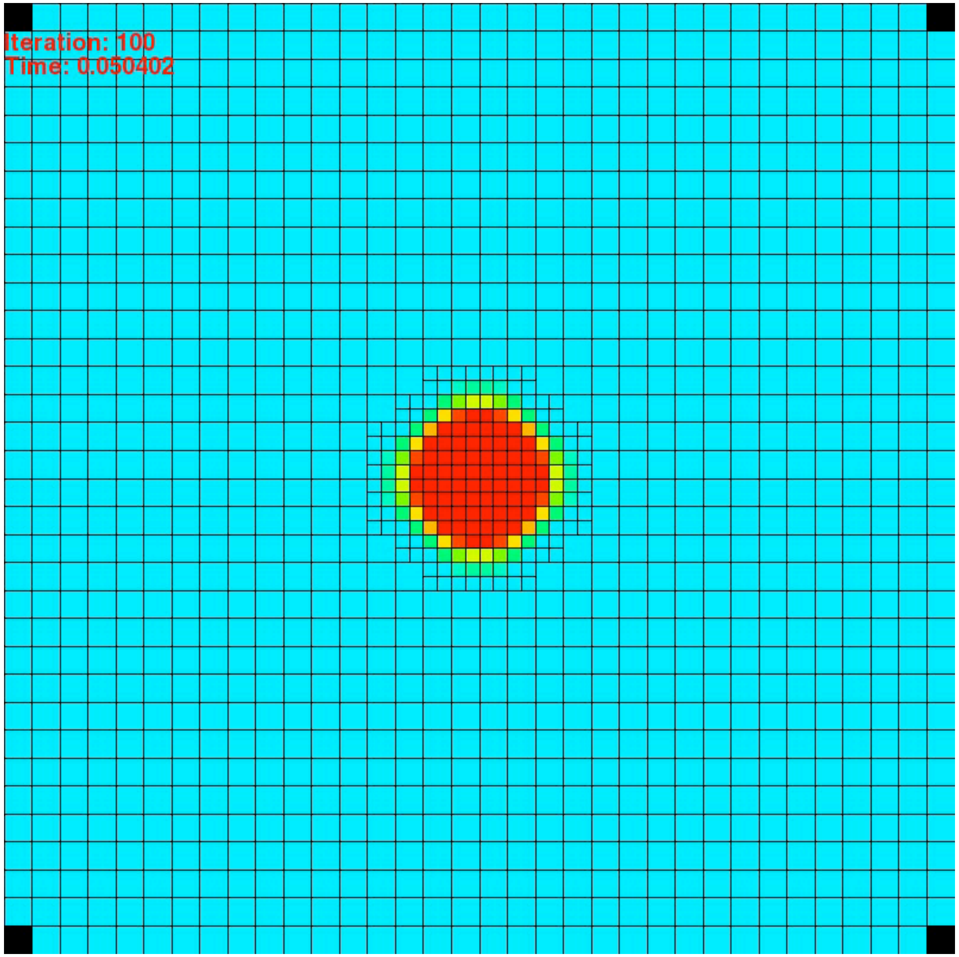
7:00 am	Breakfast
8:30 am	General Session
10:05 am	Break
10:30 am	Special Session: Unive...
11:45 am	Lunch

Computational Modeling Science Math Supercomputing Challenge

Parallel Computing

- Shallow-water simulation demo from ~2008 kickoff originated McCurdy High School project (Joseph Koby, Sarah Armstrong, Juan-Antonio Vigil, Vanessa Trujillo)
  - Morphed into adaptive mesh refinement co-design application CLAMR at http://www.github.com/losalamos, contributions by Dennis Trujillo, McCurdy
  - CLAMR runs on CPUs, GPUs, MICs

CLAMR Demo



Iteration: 100
Time: 0.050402

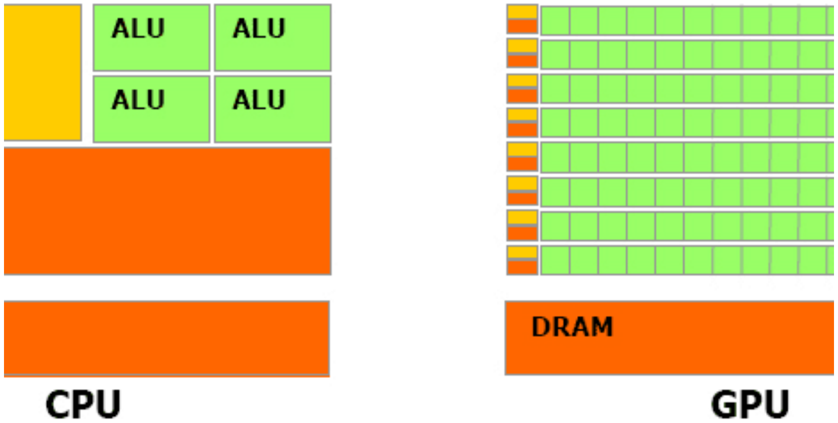
Parallel Computing Global Sums

- Dealt with different sum results depending on number of processors
- Finite precision arithmetic not associative
- Enhanced Precision Sums
  - Used Sapient, a Supercomputing Challenge project, by Jon Robey and published in Parallel Computing journal

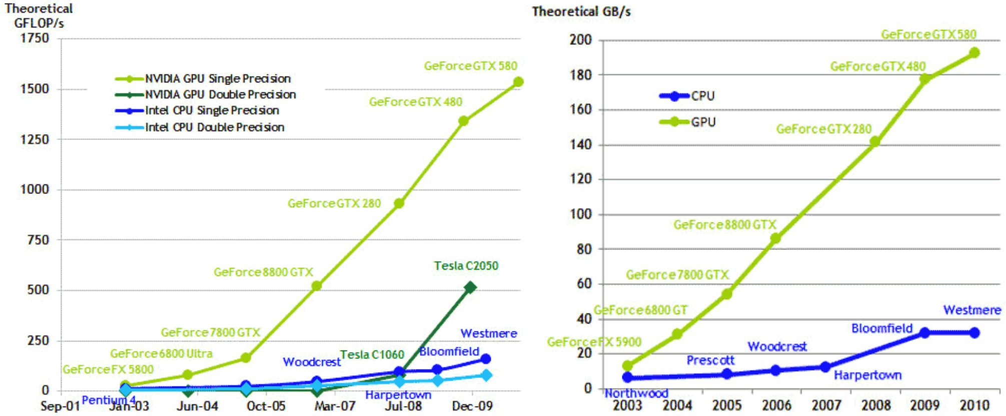
Parallel Computing Hashing

- Hashing
  - Sorting exercise – pair off and everyone compare last names. Sort alphabetically and if < move left; >move right. Repeat with next person until room is sorted. Compare time it takes versus setting out letters alphabetically across room and everyone go to their letter
- Perfect Hashing
  - Developed in part by Rachel Robey and published in SIAM Journal of Scientific Computing
  - Done instead of a Supercomputing Challenge Project
- Compact Hashing
  - Developed in part by Peter Ahrens, Los Alamos and Sara Hartse, Desert Academy
  - Near publication in SIAM Journal of Scientific Computing

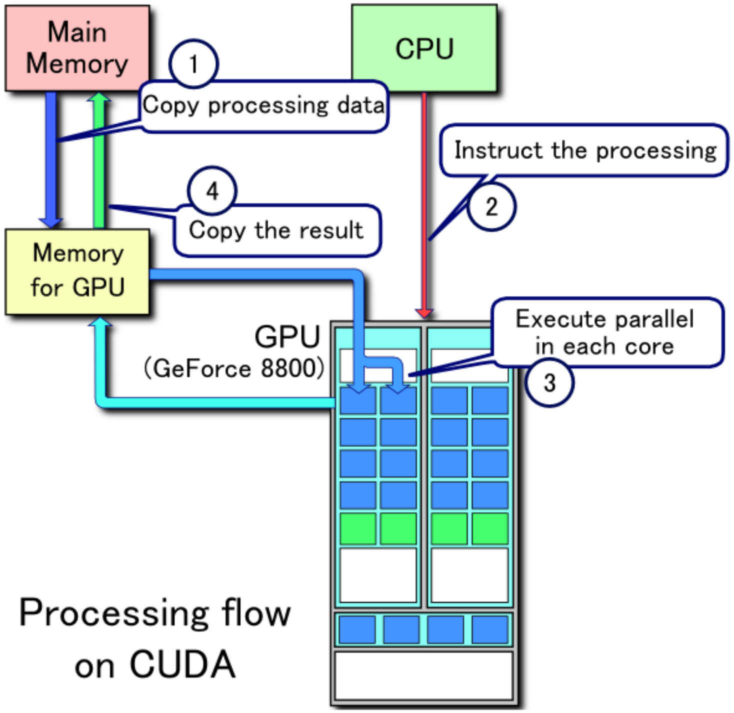
GPU Computing: Why GPUs



GPU allocates more chip space to arithmetic operations (green) than the CPU.



Floating point operations and bandwidth are increasing faster on GPUs than CPUs.



Processing flow on CUDA

“Processors” in blue are thread engines execute 32 threads each and there are many of these thread engines. (Tosaka, Wikimedia)

GPU Computing

- Massively parallel processing (speeding up computations), closer to your display (better/ faster graphics)
- Best with NVIDIA graphics cards and the CUDA programming language
- SDK, emulator, and drivers: https://developer.nvidia.com/cuda-downloads
- Samples are included!
- Designed for use with C, C++, or FORTRAN

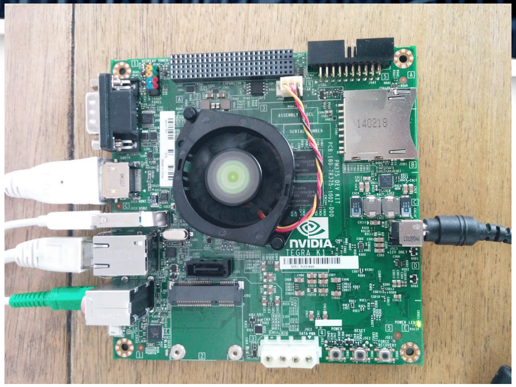
GPU Computing (cont.)

- Accessible from almost any programming language with some effort, but “bindings” (software interfaces) are available
  - Python
    - http://mathema.tician.de/software/pycuda/
    - https://developer.nvidia.com/how-to-cuda-python
  - Java (http://www.jcuda.org/jcuda/JCuda.html)
- Sample application execution
  - N-body simulation: https://www.youtube.com/watch?v=WVd6DT2IQ1o
  - Fluid motion: https://www.youtube.com/watch?v=wwy7bcmDjGQ
  - Mandelbrot: https://www.youtube.com/watch?v=RaCPrXYOiRE

Other GPU languages

- OpenACC – being merged into OpenMP
- OpenCL – actually any device from multi-core to GPU to handheld

Jetson TK1



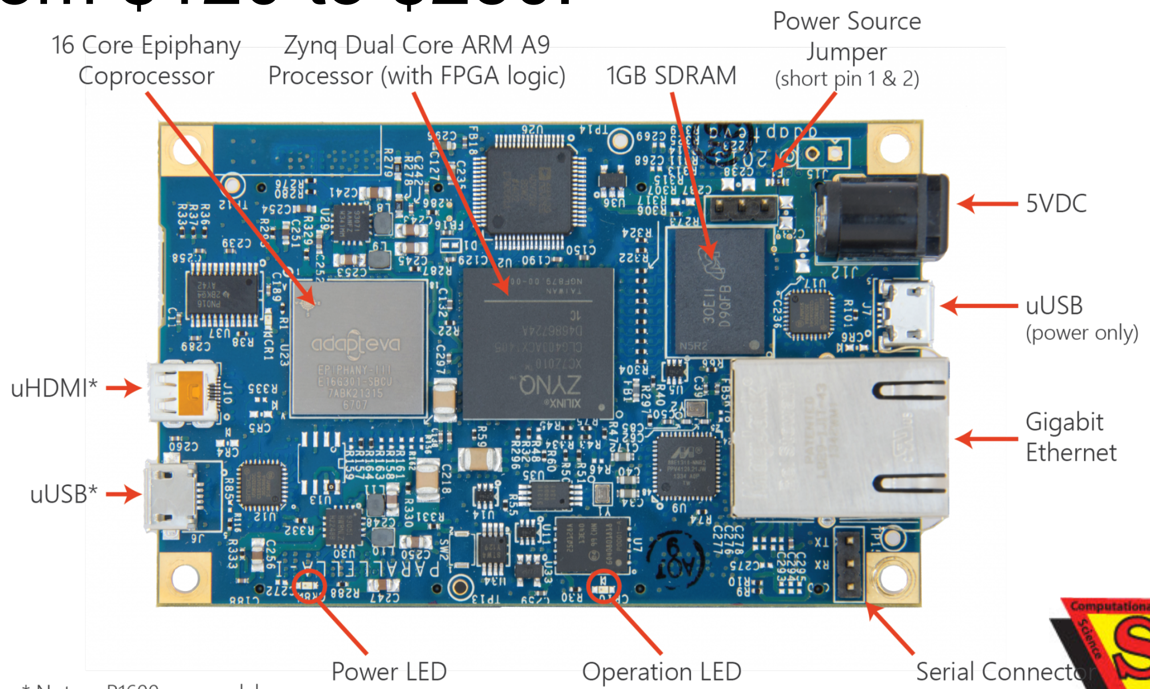
- An Nvidia Kepler GPU with192 “cores” and a 4 core ARM CPU for less than $200.
- Modified Ubuntu OS and runs Cuda Applications
  - – http://devblogs.nvidia.com/parallelforall/jetson-tk1-mobile-embedded-supercomputer-cuda-everywhere/
  - – http://www.drdobbs.com/embedded-systems/nvidia-jetson-tk1-reviewed/240168804

Jetson TK1 Demo

- Lets see some apps running on the Jetson TK1

Parellella

Another hobbyist system is the parallella board with 16-64 processors. It is a credit card size system (http://www.parallella.org/board). Prices run from $120 to $250.


16 Core Epiphany Coprocessor
Zynq Dual Core ARM A9 Processor (with FPGA logic)
1GB SDRAM
Power Source Jumper (short pin 1 & 2)
5VDC
uUSB (power only)
Gigabit Ethernet
uHDMI*
uUSB*
Power LED
Operation LED
Serial Connector

* Not on P1600-xxx models

MIC (Intel Accelerator Board)

- Massively parallel processing (speeding up computations) in a format we are used to (compatibility with existing libraries and applications)
- Entry development can be done with multicore using MPI and/or OpenMP
- Knights Corner – on a expansion card on PCI bus
- Knights Landing – main chip will have 60+ cores
- New LANL/Sandia Supercomputer, Trinity, will use Knights Landing architecture