SciServer

Collaborative data-driven science

SciServer Compute Workshop

Johns Hopkins University

NSF

idies

SciServer

Collaborative data-driven science

Bring Your Code to the Data

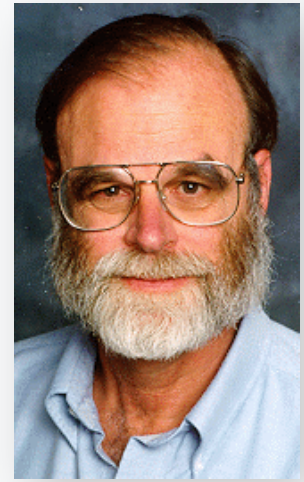
Alex Szalay

Johns Hopkins University

NSF idies

Motivation and History

- Started with the SDSS SkyServer
- Built very quickly in 2001
- Goal: **instant access** to rich content
- Idea: **bring the analysis to the data**
- Interactive access at the core
- Much of the scientific process is about data
  - Data collection, data cleaning, data archiving, data organization, data publishing, mirroring, data distribution, data analytics, data curation…
- 2012: NSF DIBBS to extend/reengineer SkyServer



Jim Gray

Where Are We Going?

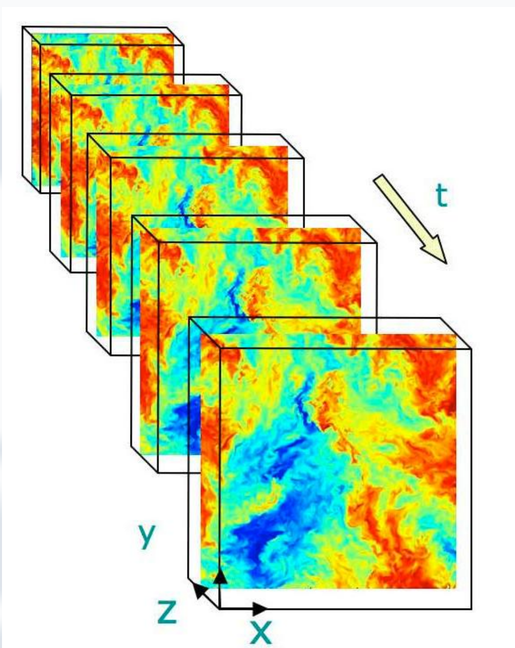
- Interactive science on petascale data
- Sustain and enhance our astronomy effort
- Grow a footprint into new disciplines
- Build scalable open numerical laboratories
- Scale system to several petabytes
- Deep integration with the "Long Tail"
- Use sharable, well-defined building blocks

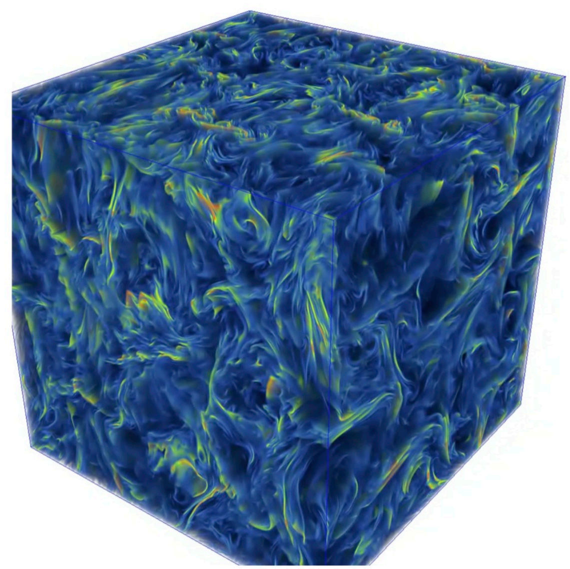
Data in HPC Simulations

- HPC is an instrument in its own right
  - Largest simulations approach/exceed petabytes
- Need public access to the best and latest
- Also need ensembles of simulations for UQ
- Creates new challenges
  - How to access the data?
  - What is the data lifecycle?
  - What are the analysis patterns?
  - What architectures can support these?
- On Exascale everything is a Big Data problem

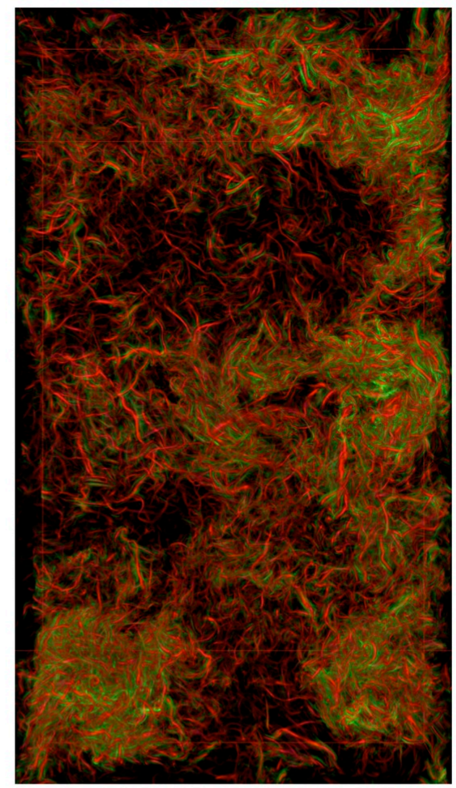
Turbulence databases (JHUTB)

- http://turbulence.pha.jhu.edu/





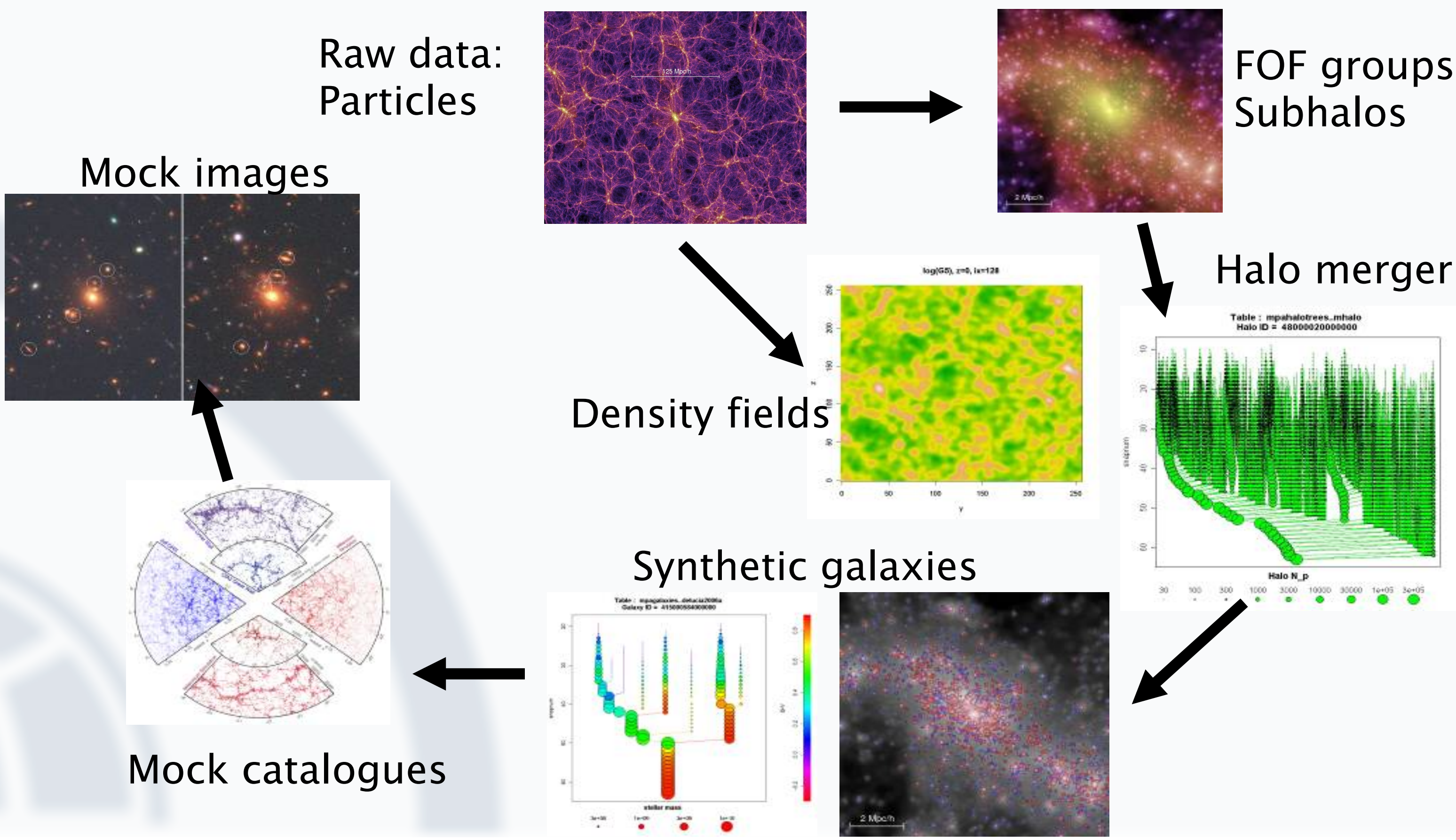
Current field $\nabla \times \mathbf{B}$



Vorticity $\nabla \times \mathbf{u}$

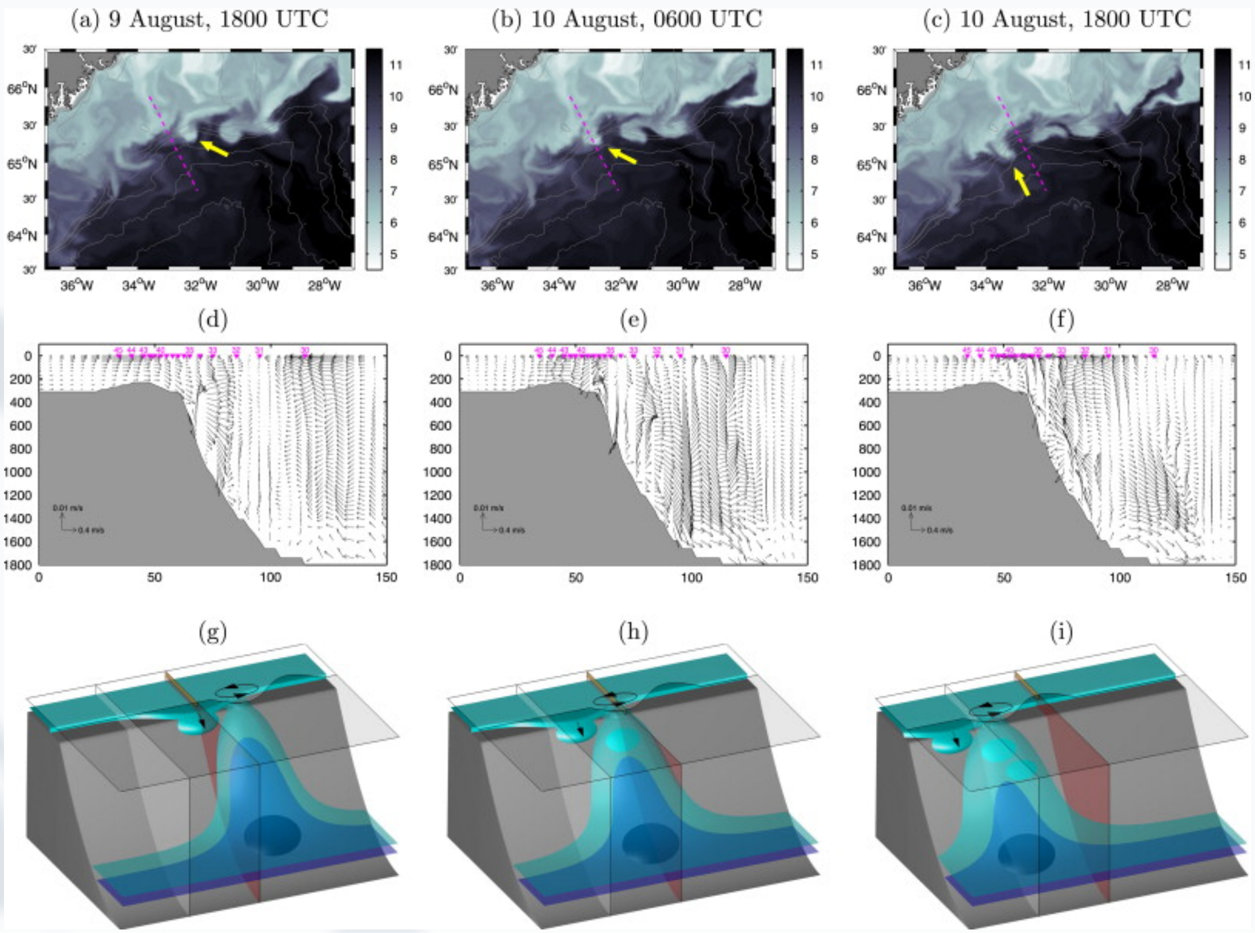
Cosmological Simulations

- Mirror of Millennium Database

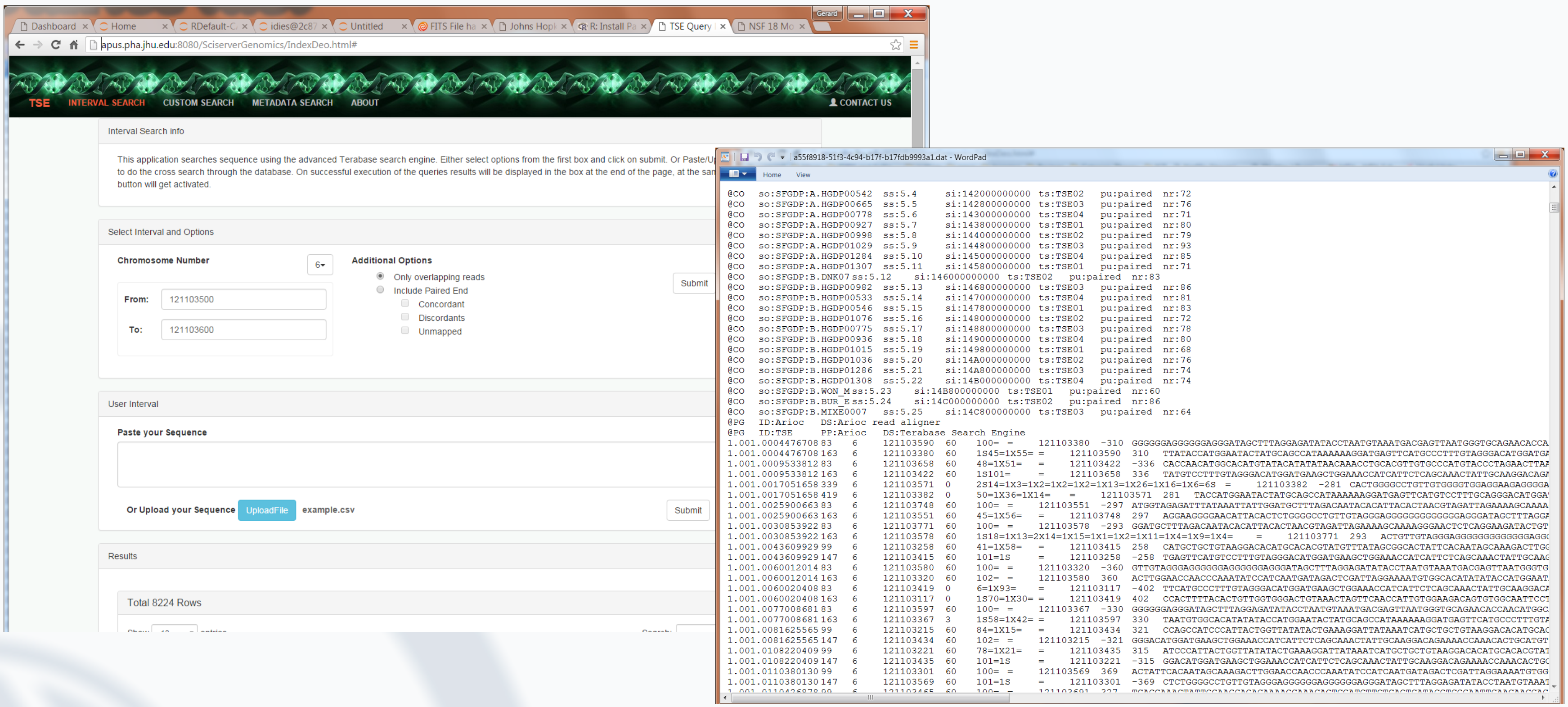


Oceanography

Hydrostatic and non-hydrostatic simulations of dense waters cascading off a shelf: The East Greenland case Marcello G. Magaldi, Thomas W.N. Haine

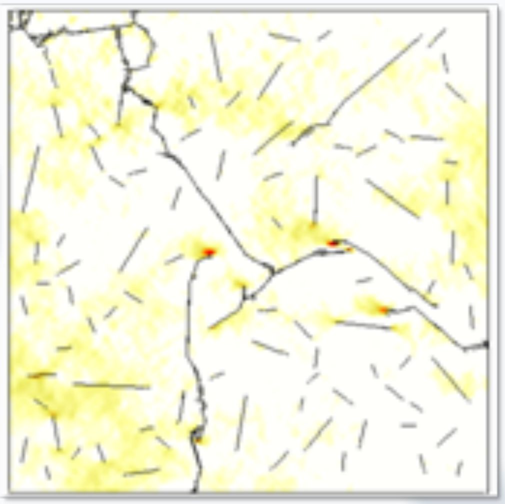


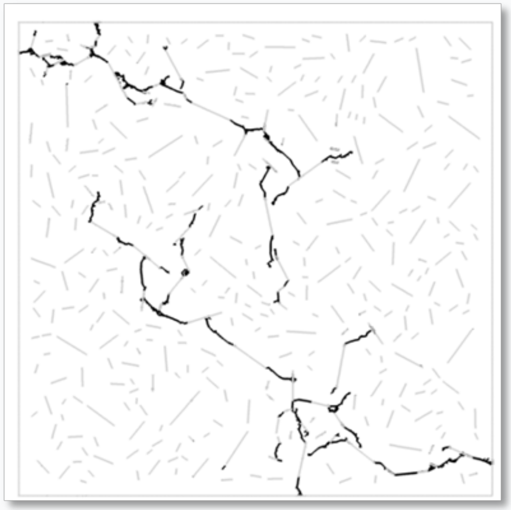
Genomics



Building a DB of a trillion short reads from Next Gen Sequencing

Materials Science





Daphalapurkar, Brady, Ramesh, Molinari. JMPS (2011)

Open Numerical Laboratories

- Create interactive Numerical Laboratories
- Analysis server-side through web service
- Use virtual sensor metaphor
- Many access patterns are local
- No need to download whole data sets
- Concept very successful in turbulence and cosmological N-body
- turbulence.pha.jhu.edu:
  19 trillion points delivered!
- Total science data in SciServer currently ~2.5PB

New Analysis Patterns Emerging

- User written crawlers, inefficient
- Cutouts delivered to users, slow
- Scalability challenge (over 100TB scales)
- Requests for scripting access
- Need for easy joins with long-tail data
- Still expecting interactive response

Architectural Challenges

- Need to define sharp tradeoffs
  - Data Analytics system is different from supercomputer
  - What is the right balance between I/O and compute?
- Need high bandwidth to large data
  - Computations/visualizations must be on top of the data
  - Must support at least few 100TB per server
  - Petascale: 3 copies for production (or erasure code?)
  - Wide area data movement/backbone is hard
- Lessons from the database world:
  - It is nontrivial to schedule complex I/O patterns
  - For subsets we must use indexing, cache resilient storage

Directions

- Offer more computing resources server side
- Enhanced visualization tools (ParaView)
- **Augment and combine** SQL queries with easy-to-use scripting tools
- **Heavy use of virtual machines/ Docker**
- **Interactive portal via iPython/Matlab/R**

SciServer

Collaborative data-driven science

Workshop Overview

Mike Rippin

April 27, 2016

Johns Hopkins University

NSF

idies

Introductions and Logistics

The Team

- ✧Alex Szalay (PI)
- ✧Mike Rippin (PM)
- ✧Ani Thakar
- ✧Jordan Raddick
- ✧Bonnie Souter
- ✧Gerard Lemson
- ✧Jaiwon Kim
- ✧Dmitry Medvedev
- ✧Deoyani Heinis
- ✧Manu Popp
- ✧Victor Paul
- ✧Sue Werner
- ✧Jan Vandenberg

Agenda

8:30 AM	Continental Breakfast & Coffee	
9:00 AM	**Welcome**	Alex Szalay
9:05 AM	**SciServer Overview**	Mike Rippin
9:25 AM	**Getting Started with SciServer**	Jordan Raddick
9:40 AM	**Technical Overview**	Dmitry Medvedev
10:30 AM	Coffee	
10:45 AM	**Demo Notebook #1**	Gerard Lemson
12:00 PM	Lunch	
1:00 PM	**Astronomy & Cosmology Examples**	Gerard Lemson
2:30 PM	Break	
2:45 PM	**Explore & Customize**	Participants
3:30 PM	**Q&A**	
3:50 PM	**Closing Remarks**	Mike Rippin
4:00 PM	Adjourn	

Logistics

- Stay in this room all day
- Restrooms
- Coffee and Breaks – morning and afternoon
- Lunch – ‘Working Lunch’ if preferred
- Wrap up – 4pm

Structure of the Day

Test Environment

- Technology
  - Everyone should be able to connect to WIFI
  - Everyone will create an account
  - Everyone will create a Docker Container
- Workshop running in TEST Environment
- MyDB etc is temporary
- Jupyter Notebooks can be saved and taken away

Objectives of the Workshop

Participants	SciServer Team
• Set up a SciServer Notebook • Authenticate with the SciServer Login Portal • Import and query SDSS with CasJobs • Save your data and graphics locally • Save your data and graphics on SciDrive • Save & Retrieve your data in MyDB • Learn the SciServer API	• Test the Compute feature set • Test out the Architecture • Gain early feedback from participants • Implement this feedback before live release

We want this to be Interactive…

Structure of the Workshop

- Agenda sets the scene
- To start: *Structured*
  - First example workbooks cover the 'building blocks' and will be done in a structured way
- Subsequently: *Flexible*
  - Notebooks delve deeper into specifics
  - Timing and deviations are fine, Q&A, examples etc
  - Tune to the experiences and needs of participants

Emphasis on PRACTICAL exercises…

SciServer Project Background

SciServer Project Award

Award

- NSF DIBBs (Data Infrastructure Building Blocks)
- 5 years: 2013 – 2018
- Approx $10M
- Cooperative Agreement

SciServer Project Objectives

Objectives

- Extend infrastructure for SDSS to support additional Science Domains
- Host and serve petabyte datasets
- Support custom user datasets
- Provide access and query services
- Provide scalable compute services
- Support analyses and data sets too large to handle locally
- Provide collaborative tools for shared analysis

Computations stay CLOSE to the DATA…

SciServer Project Components

Major Components	Supporting Technologies
Core	Microsoft SQL Server
• Login Portal	Open Stack
• CASJobs	Docker
• SciServer Compute	Jupyter
• SciDrive	
Applications	
• SkyQuery	
• SkyServer	
• GLUSEEN	
• Turbulence	

SciServer Project Timeline

Timelines

Year 1 (2013-2014)	Project Setup, Scoping, Planning, Begin Refactoring, SDSS Unification
Year 2 (2013-2014)	Architectural Refactoring – API, Single Sign-on, prototype Compute
Year 3 (2013-2014)	**SciServer System Release, Interactive Compute, Scalable Job Management, Basic Dashboard, Initial Collaborative capabilities**
Year 4 (2013-2014)	Implementation in Science Domains, Educational workbooks
Year 5 (2013-2014)	System Scale out, Data Analytics, Advanced Deployment Scenarios

NOW

SciServer Project Current Plans

Timelines – Year 3

Apr 2016	• SciServer System Release
May 2016	• Interactive Compute • SkyQuery • Gluseen
August 2016	• Prototype Scalable Job Management • Basic Dashboard • Initial Collaborative capabilities
October 2016	• Scalable Job Management • Turbulence • Cosmology
November 2016	• Project 3 year Review

SciServer

Collaborative data-driven science

Getting Started with SciServer

Jordan Raddick

April 27, 2016

Johns Hopkins University

NSF

idies

Resources

- Agenda:
www.sciserver.org/outreach/spring-workshop/detailed-agenda

- Documentation and Support (go here now!):
www.sciserver.org/outreach/spring-workshop/documentation

SciServer

Collaborative data-driven science

Inside *SciServer Compute*

Dmitry Medvedev

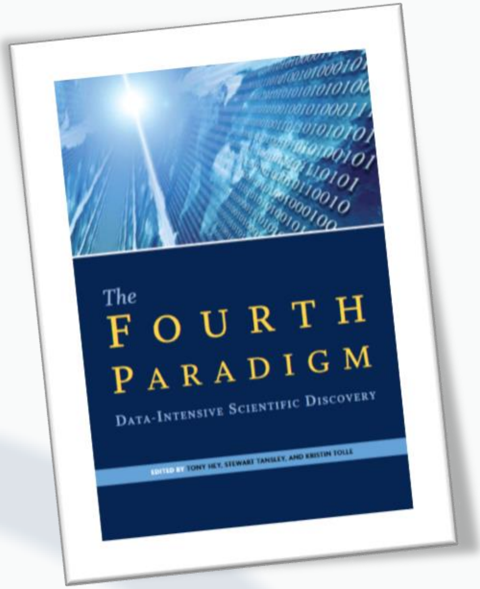
Johns Hopkins University

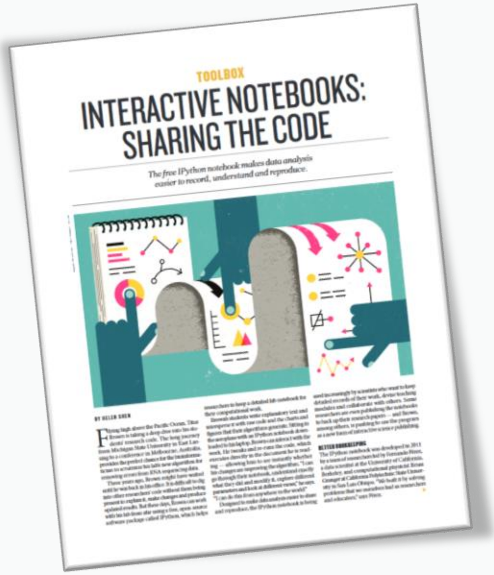
Johns Hopkins University

NSF

idies

Inspired by…





"For data analysis, one possibility is to move the data to you, but the other possibility is to move your query to the data… Often it turns out to be more efficient to move the questions than to move the data."

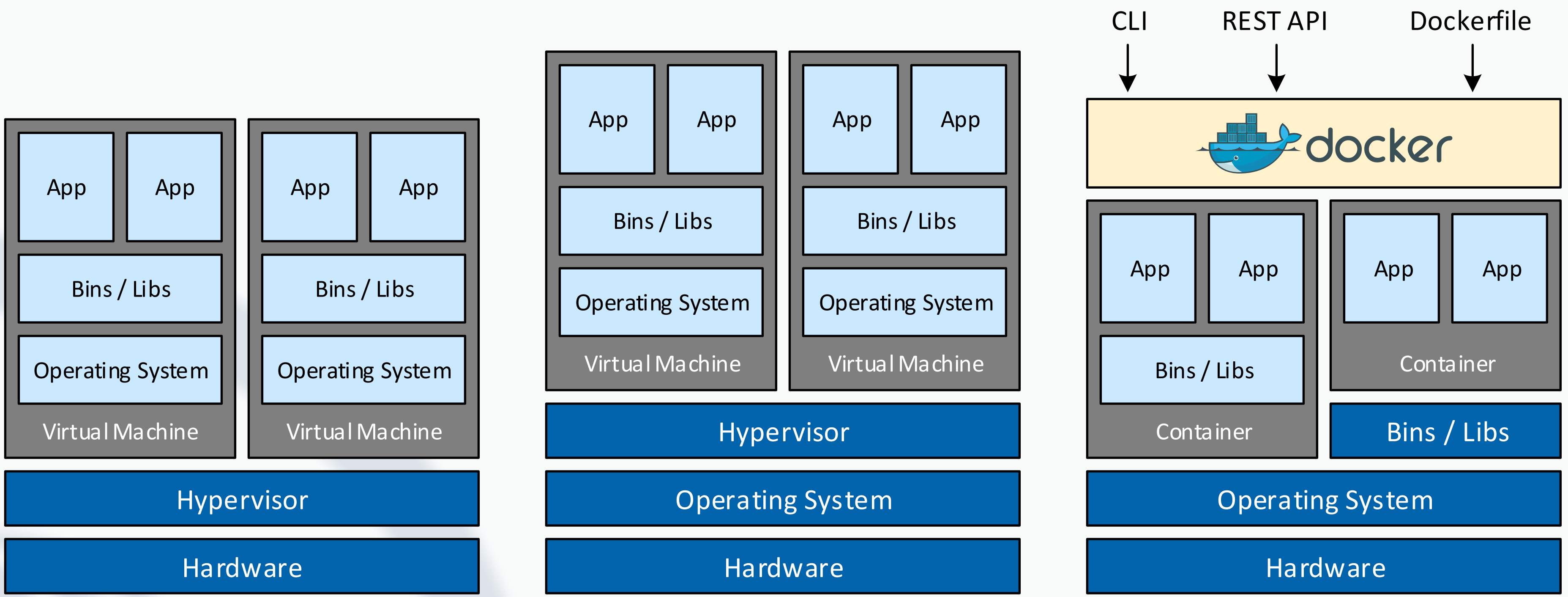
-- Jim Gray

Helen Shen's article for *Nature* – **Interactive notebooks: Sharing the code** – featured a live demo of IPython notebooks created on-demand using Docker containers, and made a strong case for using IPython notebooks in scientific data analysis.

What is SciServer Compute?

- Interactive Jupyter notebooks hosted inside Docker containers.
- Pre-configured images to create new containers from (R, Python, MATLAB, …).
- High-bandwidth, low-latency access to other SciServer services and data sources through the notebooks.
- Users manage their own containers.

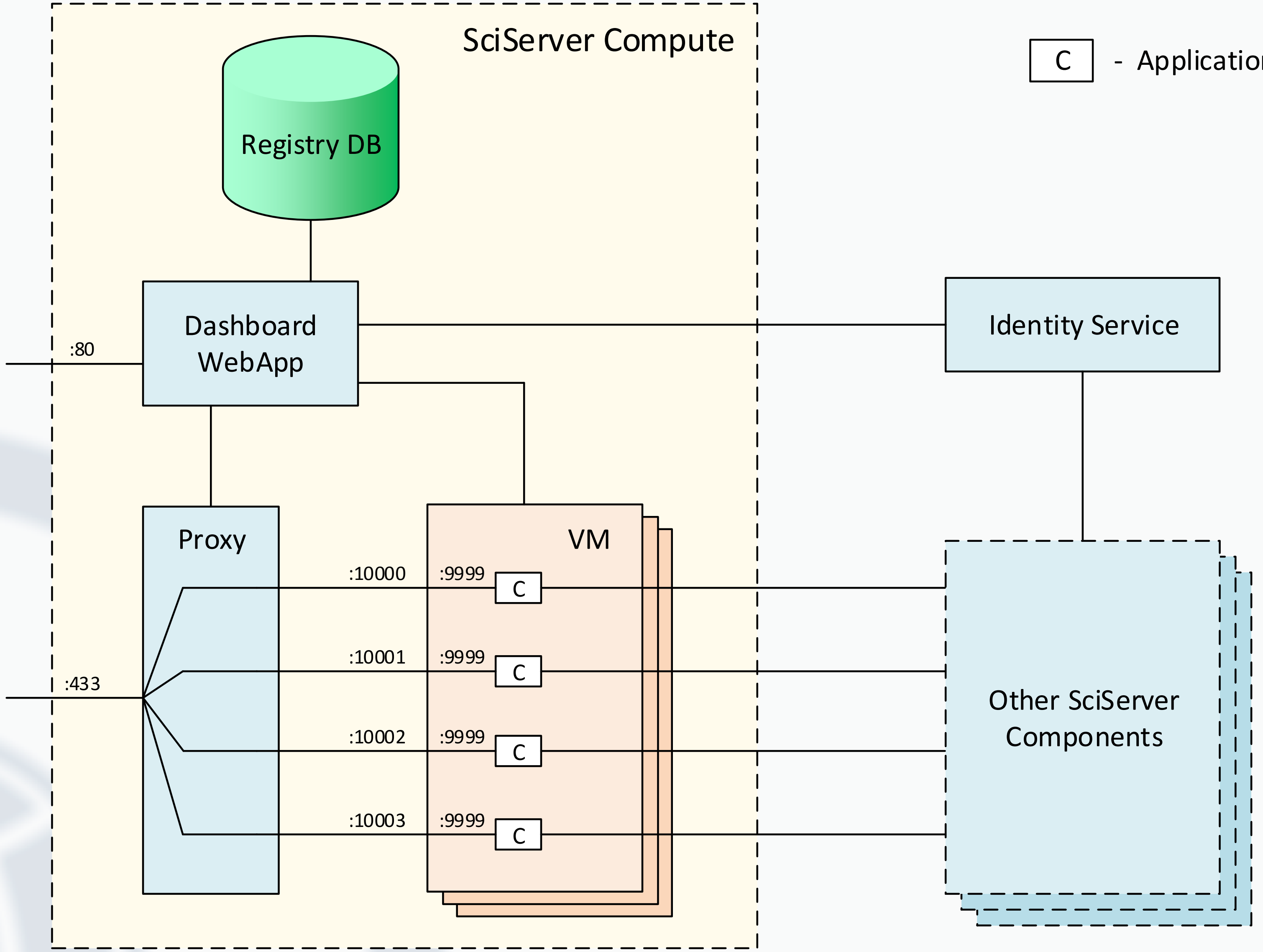
What are Docker Containers?



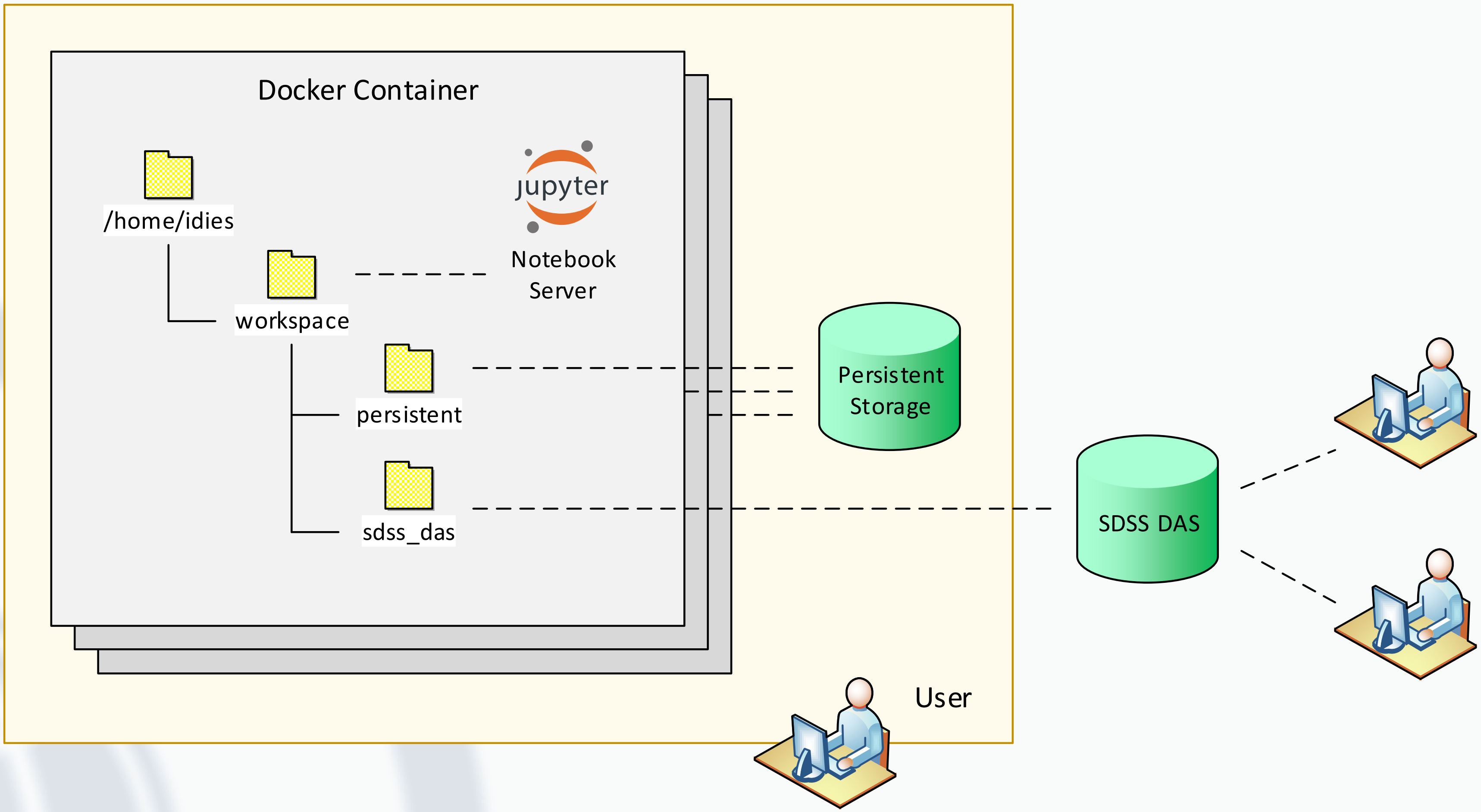
Type 1 Hypervisor

Type 2 Hypervisor

Linux Containers


SciServer Compute
Registry DB
Dashboard WebApp
:80
Proxy
:433
VM
:10000
:9999
C
:10001
:9999
C
:10002
:9999
C
:10003
:9999
C
C - Application Container
Identity Service
Other SciServer Components

Data Storage Configuration



Work in Progress

- Run asynchronous non-interactive jobs in separate Docker containers. It's meant to be more than just Jupyter notebooks!
- Create new VM nodes on-demand to accommodate growing number of users.
- Provide scratch (temporary) storage space for working with large amounts of data.
- Improve resource management.

Questions?