MTAGS11 – Seattle, Washington

Parallel High-Resolution Climate Data Analysis using Swift

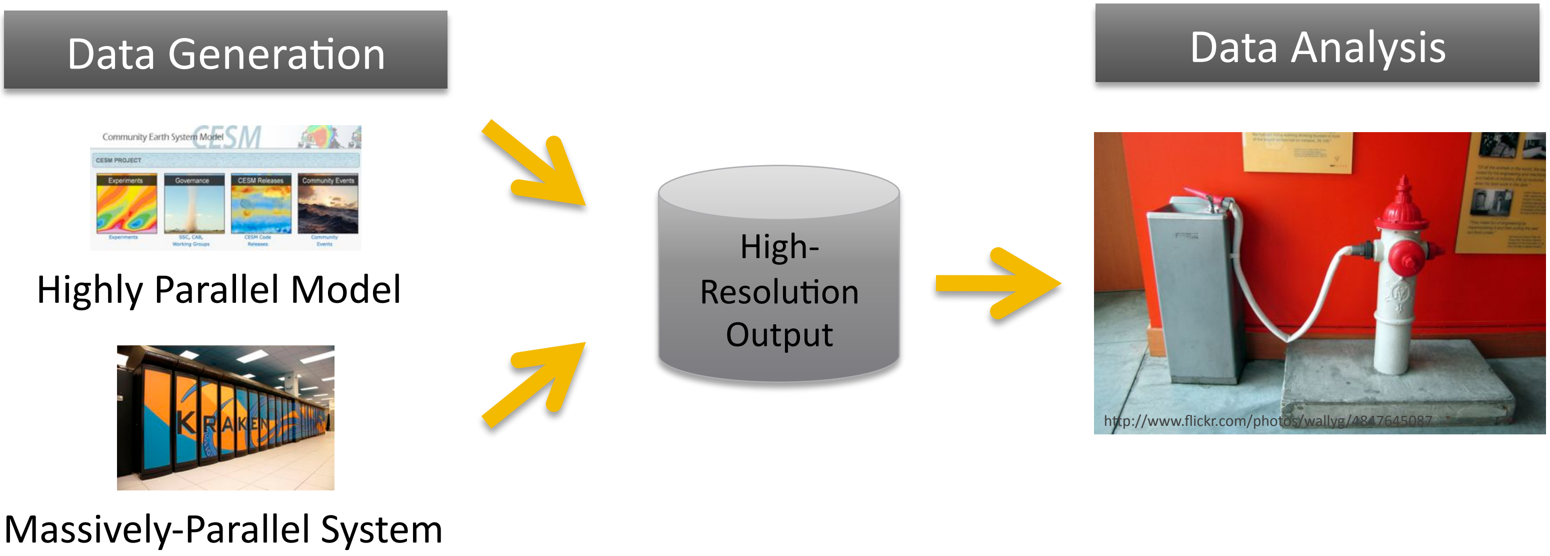
Matthew Woitaszek

mattheww@ucar.edu

Computational and Information Systems Laboratory

National Center for Atmospheric Research

Overview: Many-Task & Data-Intensive



Data Generation

Highly Parallel Model

Massively-Parallel System

High-Resolution Output

Data Analysis

http://www.flickr.com/photos/wallyg/4847645087

- Complimentary aspects
  - Workflow development: generation and analysis
  - Infrastructure support

Science: The PetaApps Project

- High-resolution climate experiment
  - Explore impact of weather noise on climate… and technical/computer science issues
  - ~18M CPU hours on NICS Kraken (np=5844)

- “Supersizing” the data: ~100TB for 155 years

0.1°	Ocean	[3600 x 2400 x 42]	100x
0.1°	Sea-ice	[3600 x 2400 x 20]	
0.5°	Atmosphere	[576 x 384 x 26]	4x
0.5°	Land	[576 x 384]	

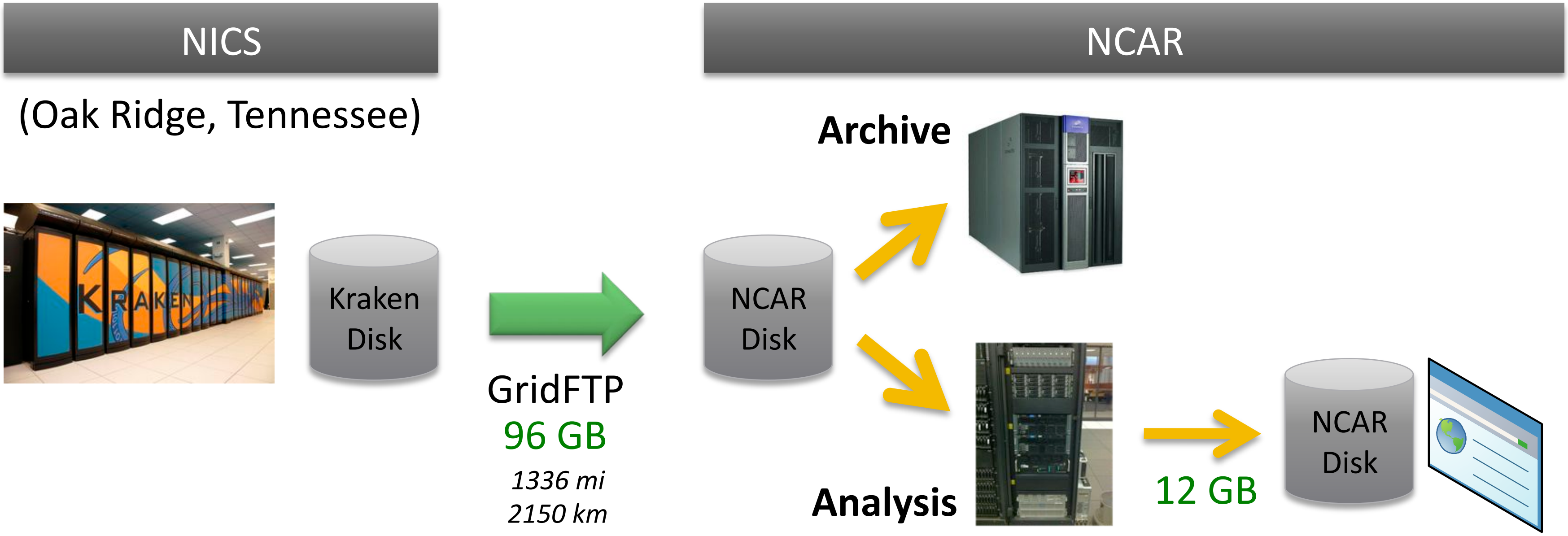
Workflow: The AMWG Diagnostic

- Analysis process for CESM atmosphere component
- Hardware over the years
  - Past: Tape as a file server
  - Now: Central shared disk
  - Emerging: Data-intensive platforms
- Model constantly changing, resolution increasing, and hardware improving… *but it's the same C-shell script!*

John (Scientist)

This hasn't really changed since 2001.

The AMWG Analysis Pipeline



NICS

(Oak Ridge, Tennessee)

Kraken Disk

GridFTP

96 GB

1336 mi

2150 km

NCAR

NCAR Disk

Archive

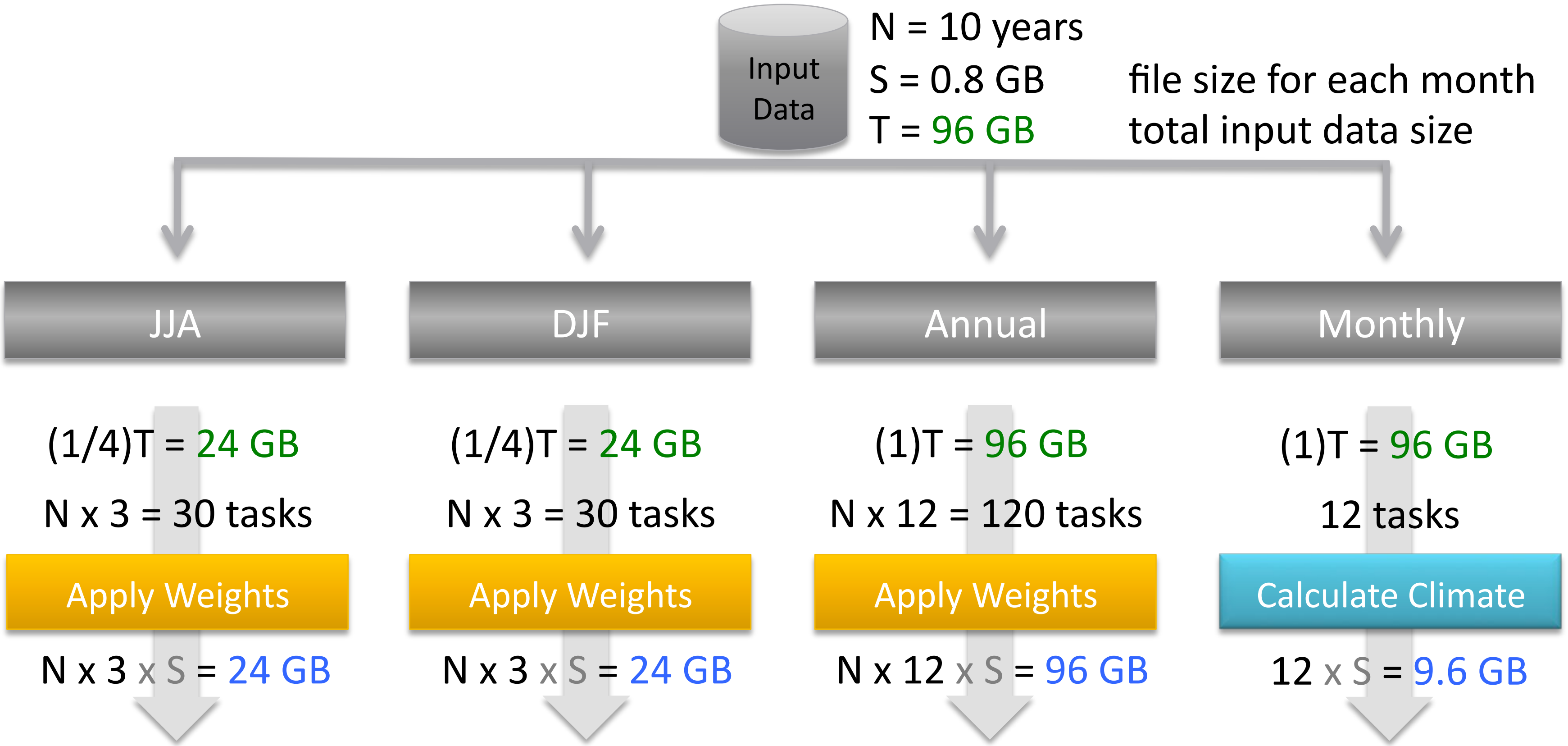
Analysis

12 GB

NCAR Disk

- Original inefficiencies
  - Analysis “diagnostic” process is a **serial** C-shell script; 2° to 0.5° ➔ *minutes to hours*
  - Major components invoked manually

The AMWG Diagnostic Workflow



Input Data

N = 10 years

S = 0.8 GB file size for each month

T = 96 GB total input data size

JJA	DJF	Annual	Monthly
(1/4)T = 24 GB	(1/4)T = 24 GB	(1)T = 96 GB	(1)T = 96 GB
N x 3 = 30 tasks	N x 3 = 30 tasks	N x 12 = 120 tasks	12 tasks
Apply Weights	Apply Weights	Apply Weights	Calculate Climate
N x 3 x S = 24 GB	N x 3 x S = 24 GB	N x 12 x S = 96 GB	12 x S = 9.6 GB

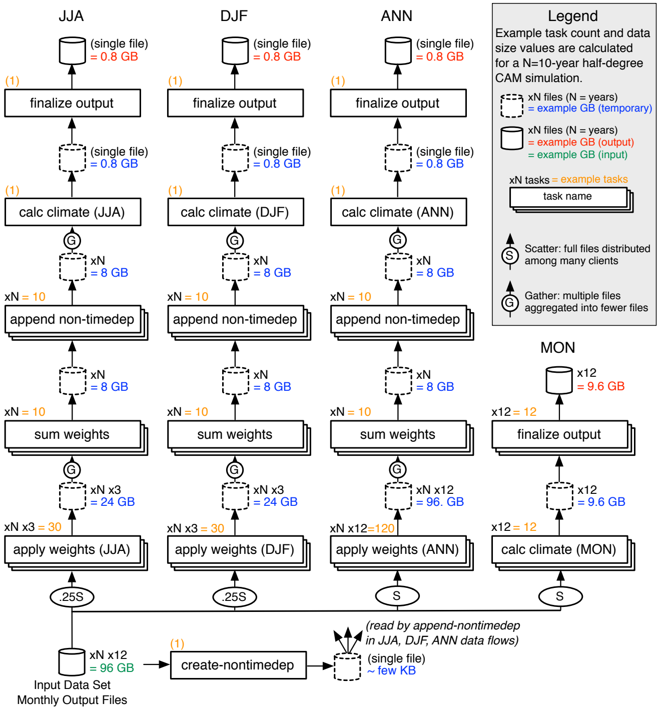
- First step: Given 96 GB of input data, read 240 GB and write 153.6 GB

The AMWG Diagnostic Workflow

- A data-intensive, many-task workflow

Input Data

10 year simulation	
120 files	(one per month)
0.8 GB	file size
96 GB	total input data size



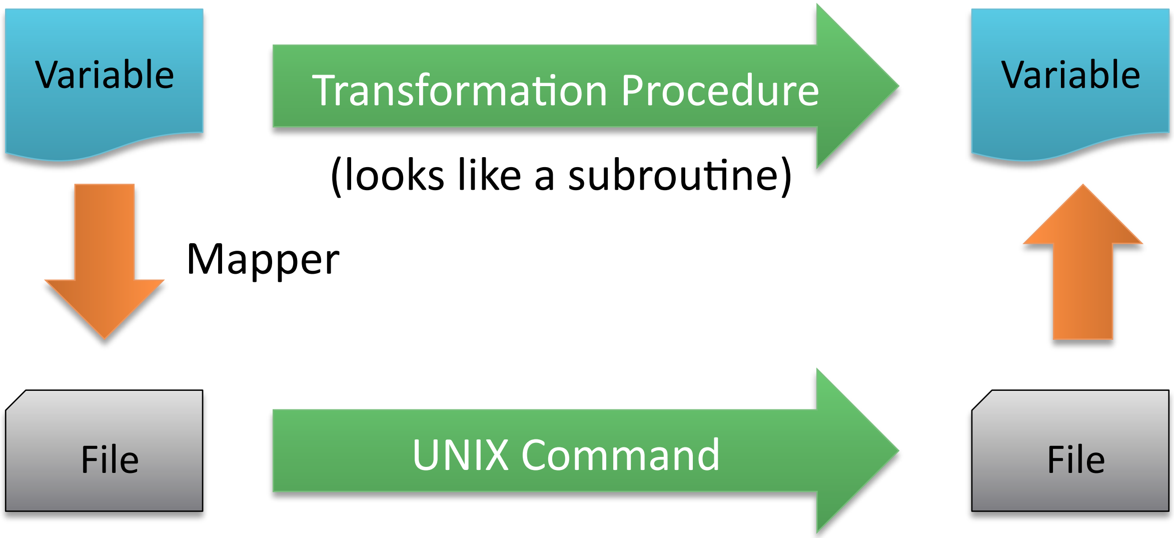
- Workflow data handling volume

Input data	96 GB
Read from disk	444 GB

Output data	12 GB
Write to disk	194 GB

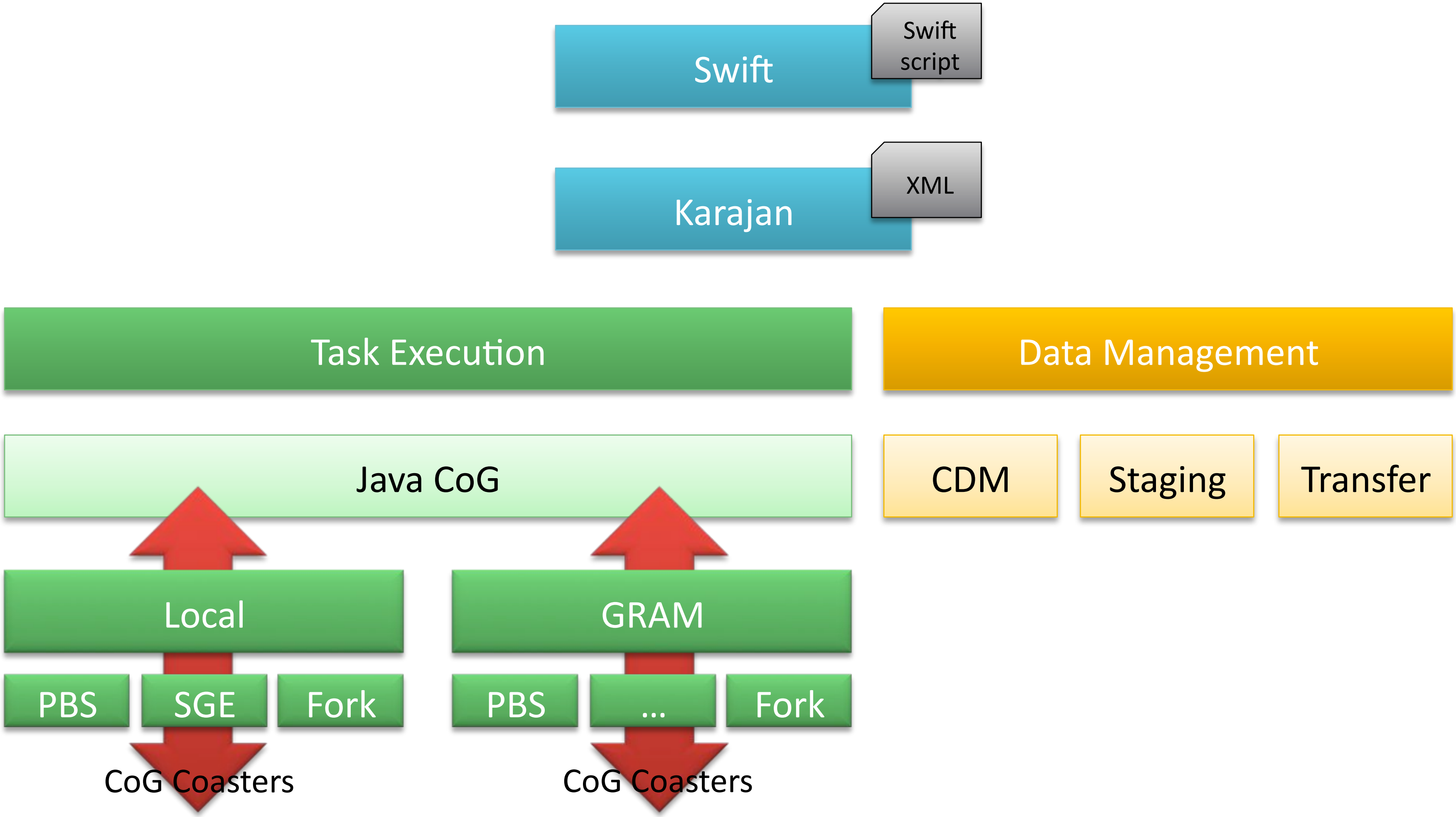
Swift Parallel Scripting

- A scripting language designed for **transforming data stored in files**



- **Trigger-based execution** exposes parallelism

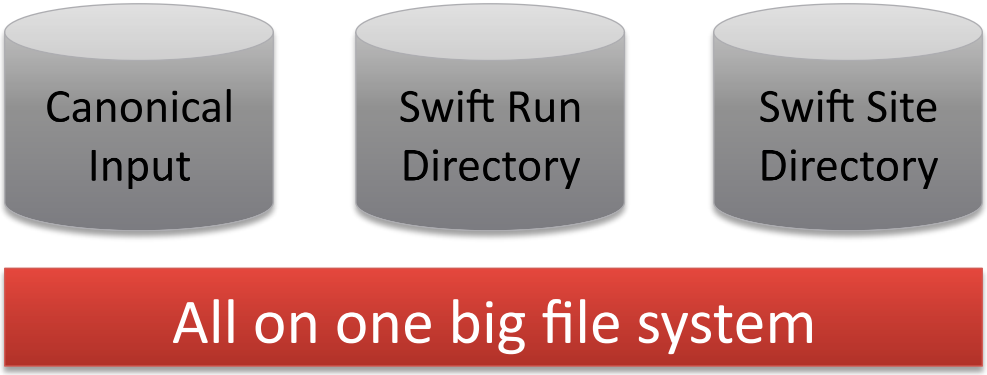
Swift: Coordinating Tasks on Sites



Data Management

- Swift manages the runtime data environment
  - Distribute tasks across multiple sites and Grids
  - Ensure re-startability and task independence

- But it didn’t match our single-site paradigm…



Canonical Input

Swift Run Directory

Swift Site Directory

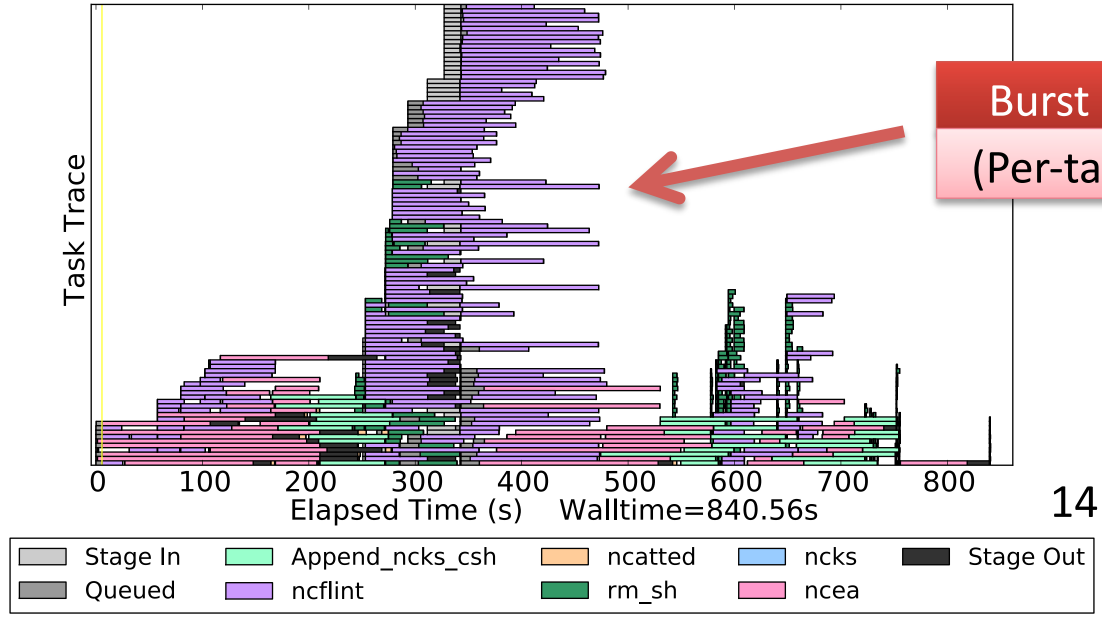
All on one big file system

Data Management Controls

- Staging
  - Problem: Copy overhead, doubled capacity
  - Solution: Swift Collective Data Management (CDM)

- Retention
  - Problem: Capacity constraints and *intermediate* files
  - Our solution (about a year ago):
    - Artificial parallelism constraints on stages
    - Manual file management
  - Or: Swift data management using variable scope

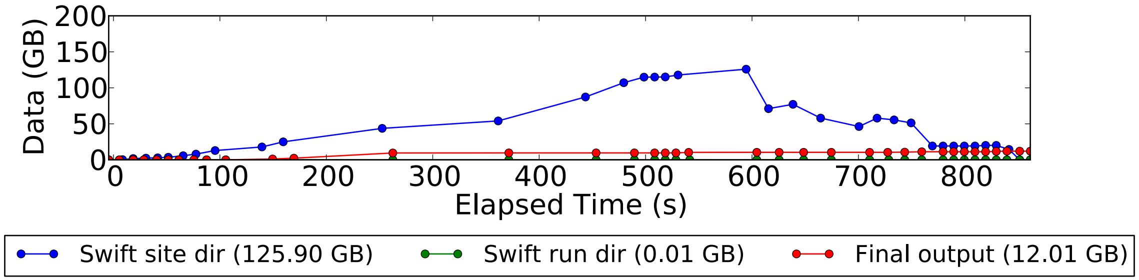
Parallelism in the AMWG Diagnostic



Burst of I/O activity

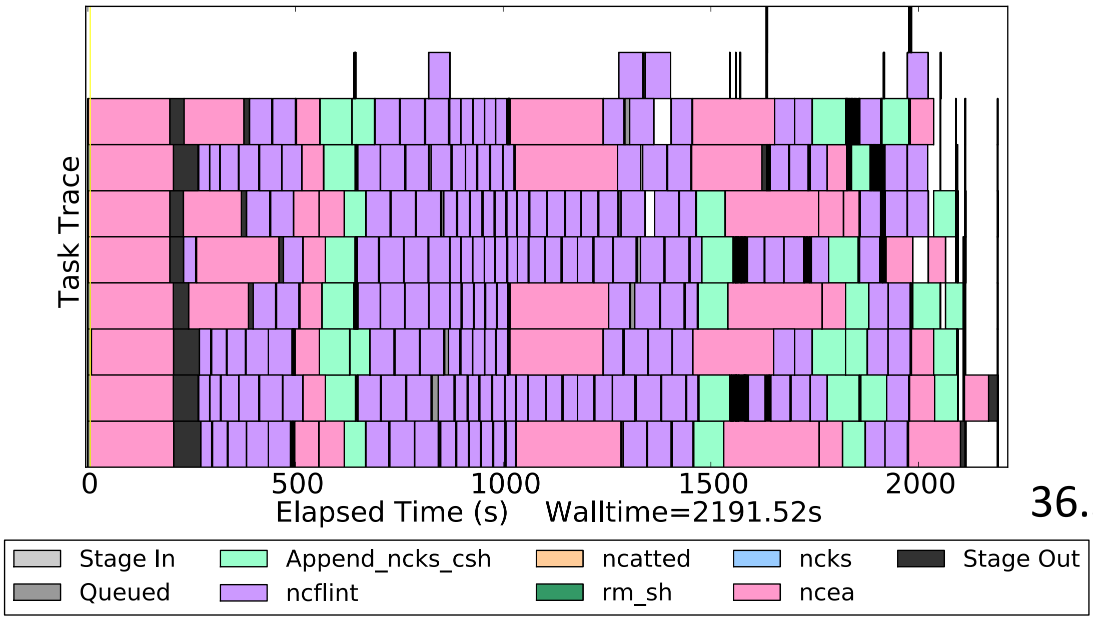
(Per-task slowdown!)

14 minutes

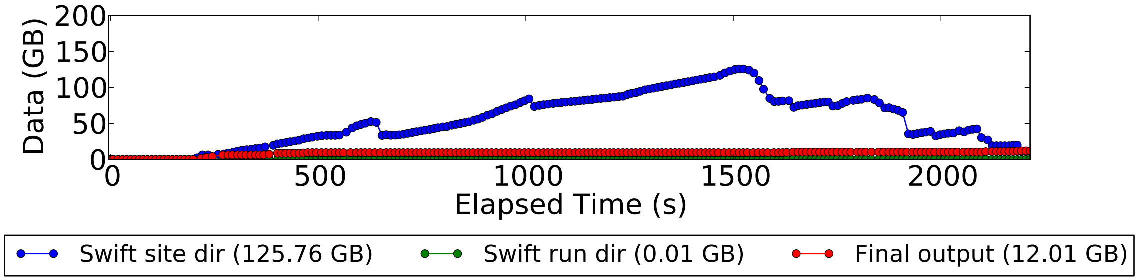


This version constrains temporary file generation by forcing serialization between the four analysis chains and removing earlier temporary files.

Parallelism in the AMWG Diagnostic

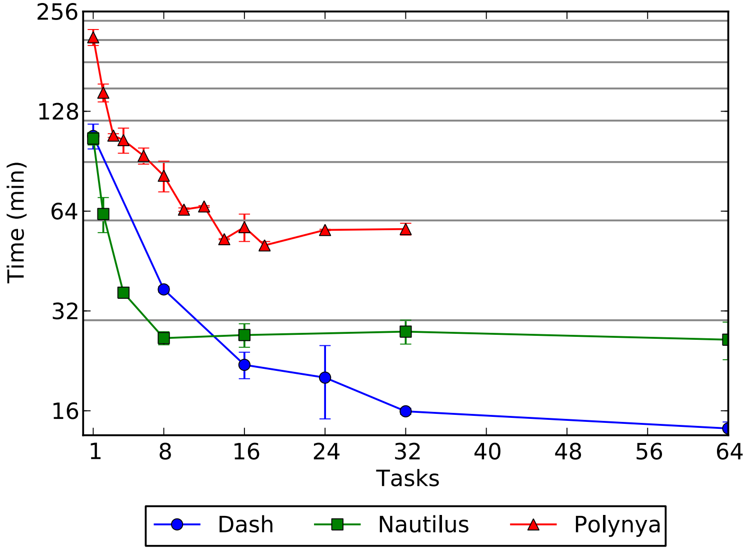


36.5 minutes



This run allows only 8 threads to be in an execution stage. Some operations, such as task clean-up, might overlap.

AMWG/Swift Prototype Performance



Comparing data analysis architectures

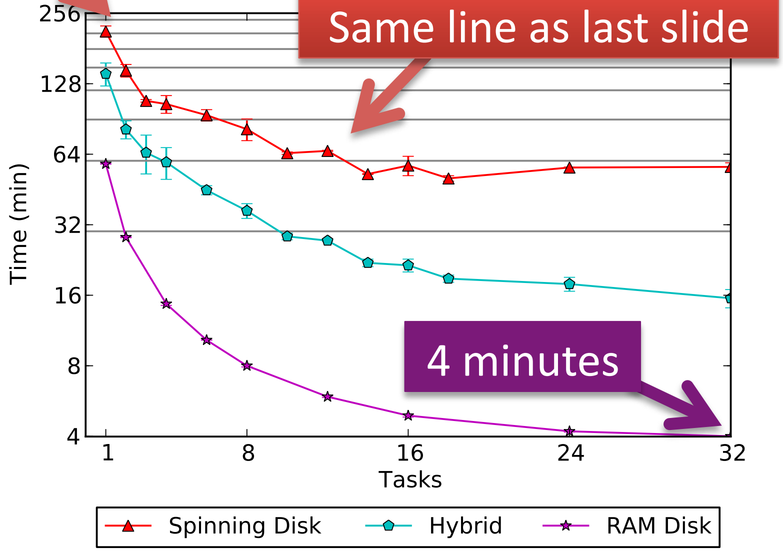
Dash - Linux cluster at SDSC with 32 nodes, 16 cores per/node, and 48 GB/node; GPFS-WAN storage (without ScaleMP)

Nautilus - SGI Altrix UV 1000 at NICS with 4 GB/core (SSI); GPFS medusa Storage

Polynya - server at NCAR with 32 cores and 1 TB RAM, 2007-era GPFS from Frost

- Performance factors:
  - Limited workflow parallelism
  - Platform characteristics (I/O throughput)

AMWG/Swift Prototype Performance



214 minutes serial

Same line as last slide

4 minutes

Comparing storage technologies on Polynya

Spinning Disk – input and temporary data on parallel file system disk (**56 min**)

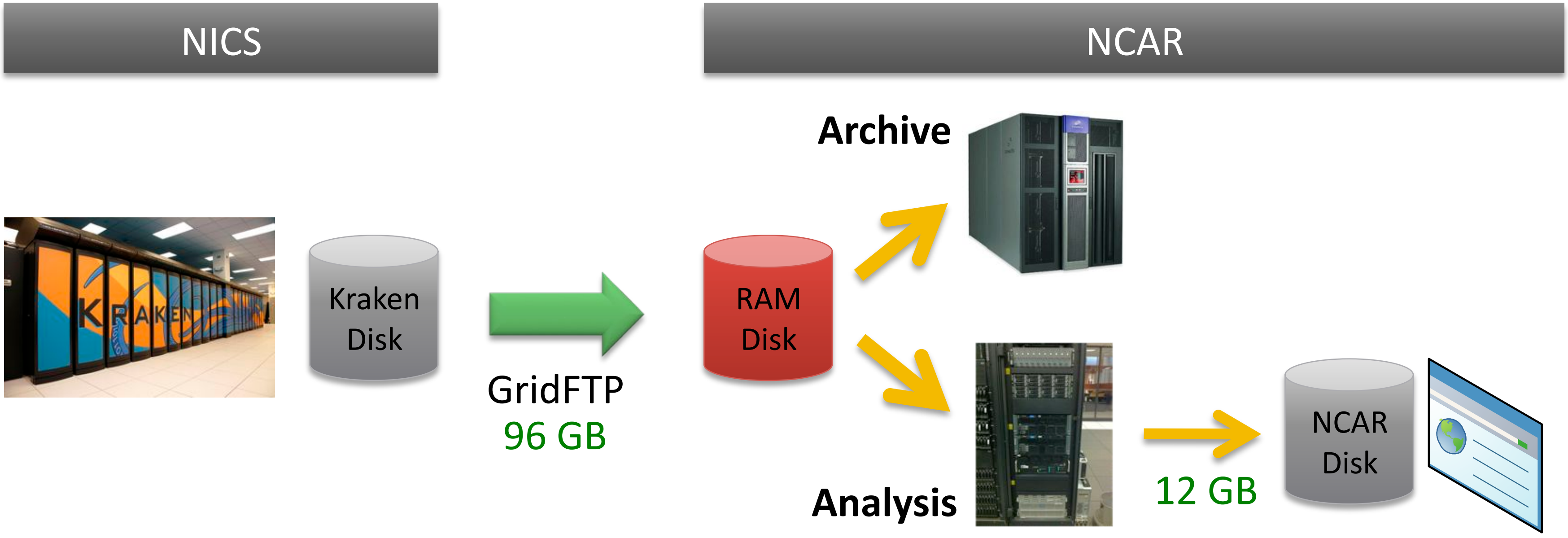
Hybrid – input on disk, temporary data on RAM disk (realistic scenario, **16 min**)

RAM disk – input and temporary data on RAM disk (pre-staged input, **4 min**)

- RAM disk provides substantial speedup
  - But isn’t pre-staging the data *cheating*?
  - We also use the data multiple times…

Current Work: Skipping Spinning Disk

What if we skipped the NCAR disk?



NICS

NCAR

Archive

Kraken Disk

GridFTP

96 GB

RAM Disk

Analysis

12 GB

NCAR Disk

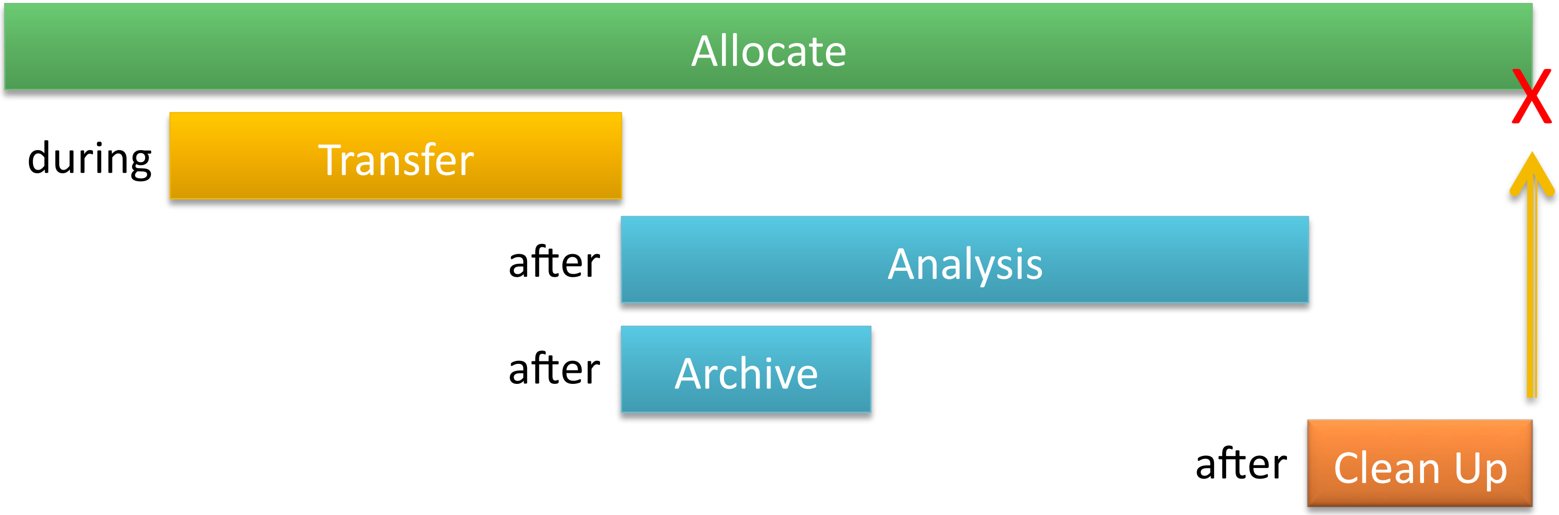
"Pre-staging" can be production mode!

Vision: Automated Analysis Pipeline

- Simulation finishes on Kraken
- Kraken job triggers pipeline on Polynya using secure ssh key
- Automated pipeline:
  - Allocates storage space
  - Retrieves data from Kraken
  - Parallel jobs for:
    - Archive to tape (from RAM disk!)
    - Analysis scripts (using RAM disk!)
  - Cleans up and notifies human

Implementation: Scheduler Support

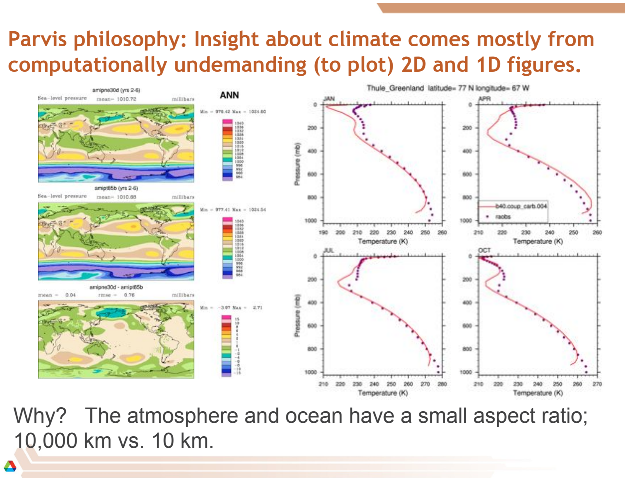
- **Co-scheduling** data and computation
  - Scheduling *data storage* on shared RAM disk
  - Scheduling *data transfers* with scheduled storage



  - **Departure from traditional quota paradigm**

Future Work: First, the Bigger Fish

- This exploratory project:
  - Motivated by NSF PetaApps
  - Swift (2010), systems (2011)



Parvis philosophy: Insight about climate comes mostly from computationally undemanding (to plot) 2D and 1D figures.

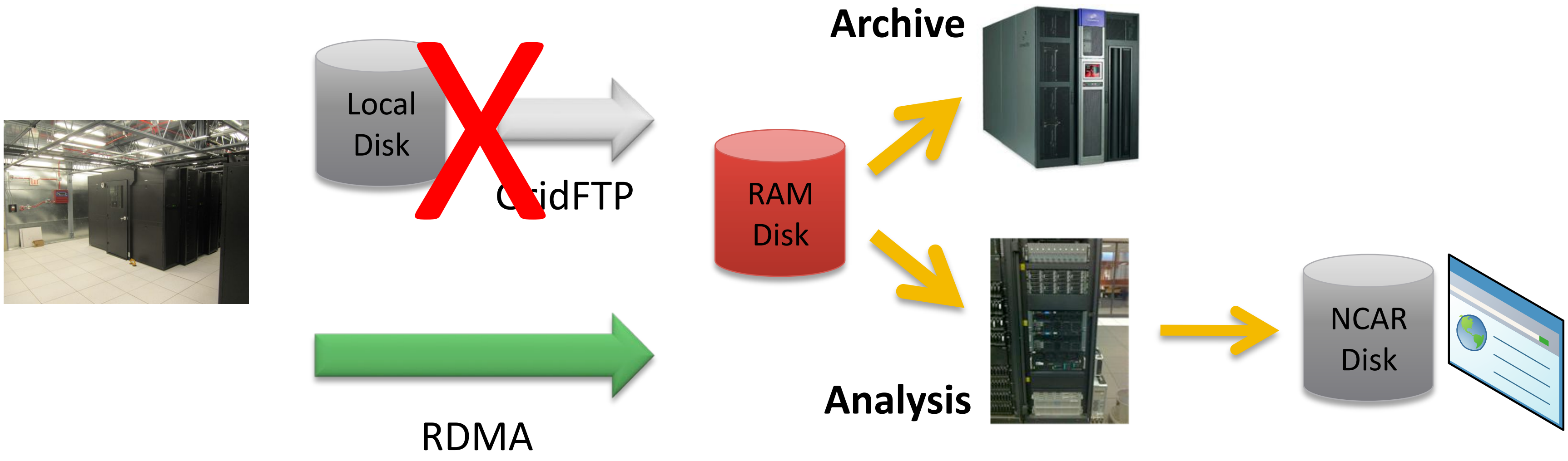
Why? The atmosphere and ocean have a small aspect ratio; 10,000 km vs. 10 km.

- **ParVis** collaborative project:
  - PI: Robert Jacob, ANL
  - Argonne, Sandia, PNNL, NCAR, UC-Davis
  - Broad-spectrum approach: PnetCDF, Swift, cloud computing, compression, NCL, data transfer

http://www.cesm.ucar.edu/events/ws.2011/Presentations/Software/jacob.pdf

Future Work: Data Direct Deposit

What if we skipped all of the disk?



Local Disk

GridFTP

RDMA

RAM Disk

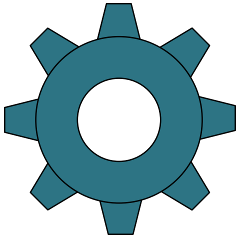
Archive

Analysis

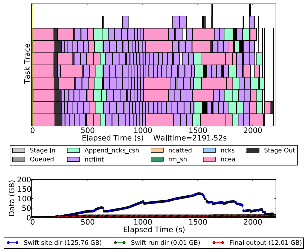
NCAR Disk

- New networking CI grant (NSF SDCI)
- Systems, networks, security… **at scale**

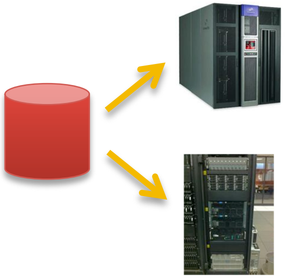
Summary and Conclusions



- AMWG workflow
  - Many-task and data-intensive



- Swift workflow management
  - Easily applied to this workflow
  - Applies to a wide range of platforms
  - Performance and reasonable scalability



- Future: a systems-centric solution including hardware and software

Acknowledgements

- Application and Workflow
  - Taleena Sines, Michael Arndt, John Dennis
- Systems
  - Dmitry Duplakin, Allan Espinosa
  - Michael Oberg, Haiying Xu
- Material and assistance from my colleagues
  - Guy Cobb, Paul Marshall, Theron Voran, Rich Loft, Henry Tufo
- Other Centers and the PetaApps Project
  - Chicago CI: Michael Wilde, Justin Wozniak, Mihael Hategan
  - SDSC Dash: Allan Snavely, Shawn Strande, Adam Jundt, Jeffrey Bennett, Eva Hocks
  - PetaApps: Kinter (COLA), Kirtman (Miami), Yelick (Berkeley), Dennis et. al. (NCAR), Bitz (Washington), and many more

I need more storage!

John
(Scientist)

I need better quota enforcement!

Oberg
(Engineer)

MTAGS11 – Seattle, Washington

Parallel High-Resolution Climate Data Analysis using Swift

Matthew Woitaszek

matthew@ucar.edu

Computational and Information Systems Laboratory

National Center for Atmospheric Research