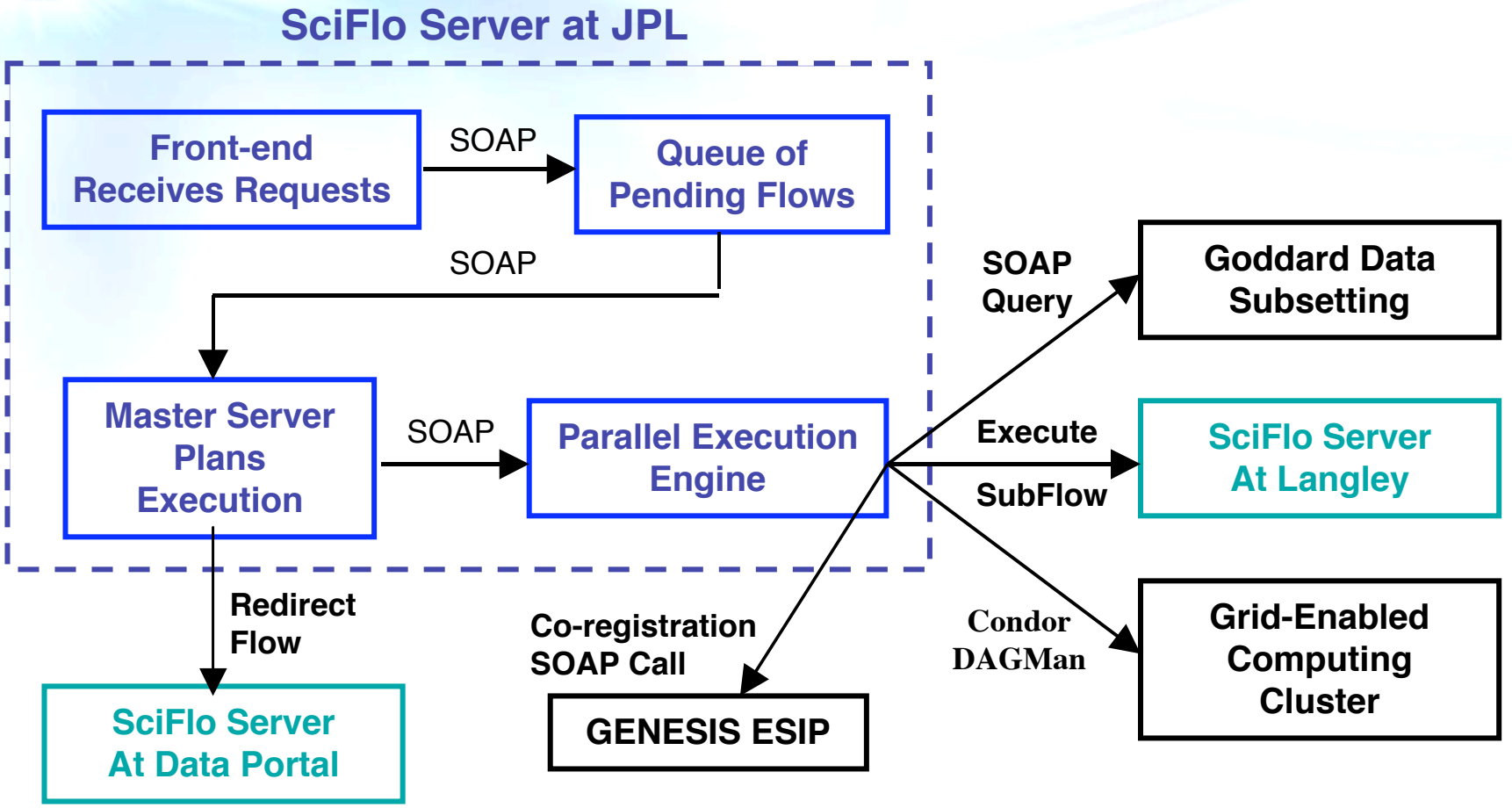
SciFlo™: Scientific Knowledge Creation on the Grid Using a Semantically-Enabled Dataflow Execution Environment

Brian Wilson, Tom Yunck, Elaine Dobinson, Benyang Tang, Gerald Manipon, Dominic Mazzoni, Amy Braverman, and Eric Fetzer
Jet Propulsion Laboratory

Do multi-instrument science by *authoring* a dataflow doc. for a *reusable* operator tree. Access scientific data by *naming* it.

Distributed Computing Using SciFlo



Inject data query or flow execution request into SciFlo network from any node.

Outline

- Enabling Technologies
  - Web Services: SOAP
  - Grid Services: OGSI & Globus v3.2
  - Parallel dataflow engines
  - Semantic Web: OWL inference using metadata
- SciFlo Distributed Dataflow System
  - Loosely-coupled distributed computing using Web (SOAP) and Grid services
  - Specifying a processing stream as an XML document
  - Dataflow engine for automated execution and load balancing
- Multi-Instrument Earth Science
  - Motivating Example: Compare the temperature & water vapor profiles retrieved from AIRS (Atmospheric Infrared Sounder) swaths and GPS limb soundings.

SciFlo: Scientific Knowledge Flow

Computing Paradigms

- Old: Big Iron mainframe, multiple users
- Current: Desktop PCs and the Internet
- New: The Grid – Computing as a utility
  - Desktops connected to computing resources worldwide
  - Petaflops of cpu, petabytes of storage
  - Bulk bandwidths: hundreds of GB/sec
  - Secure, services-based architecture
  - Vast library of analysis and modeling tools
  - Real time 3D visualizations, animations
  - Semantic understanding of service requests
  - Global-scale computing on your desktop

SciFlo: Scientific Knowledge Flow

A Conceptual Grid

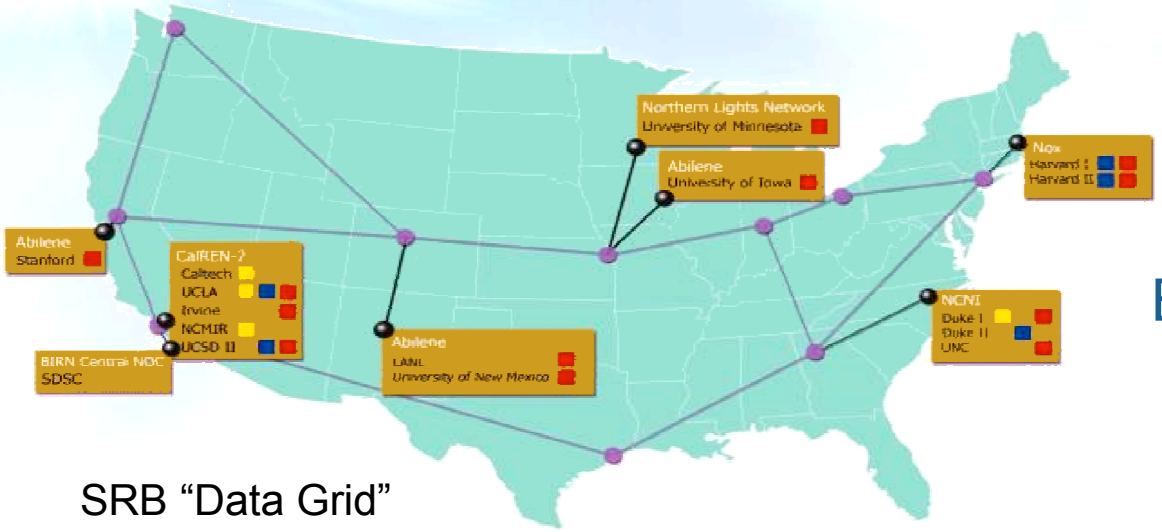
"On-Demand"
Virtualization

Scientist

Portal Data Storage CPU Algorithms Networks

SciFlo: Scientific Knowledge Flow

Some Grid Examples



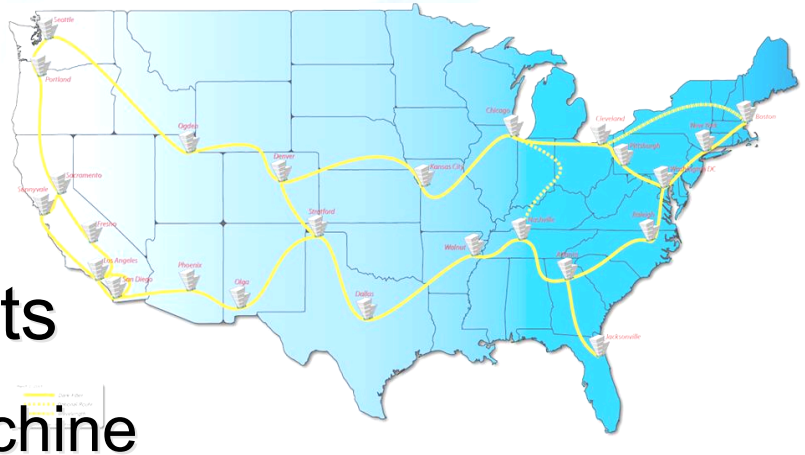
SRB "Data Grid"

The Storage Resource Broker (SRB) and Bio-Informatics Research Network (BIRN)

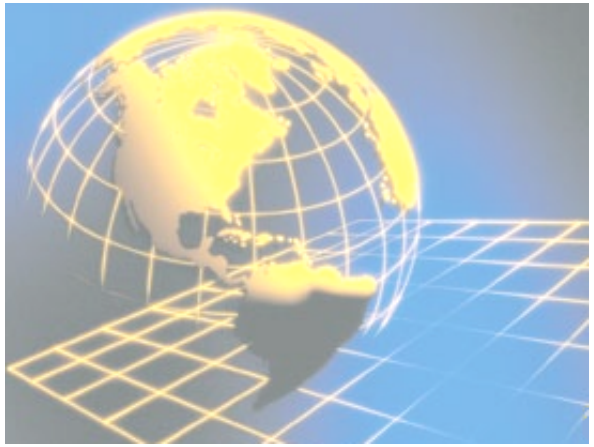
The NSF TeraGrid

SciFlo: Scientific Knowledge Flow

Buzzword Blizzard

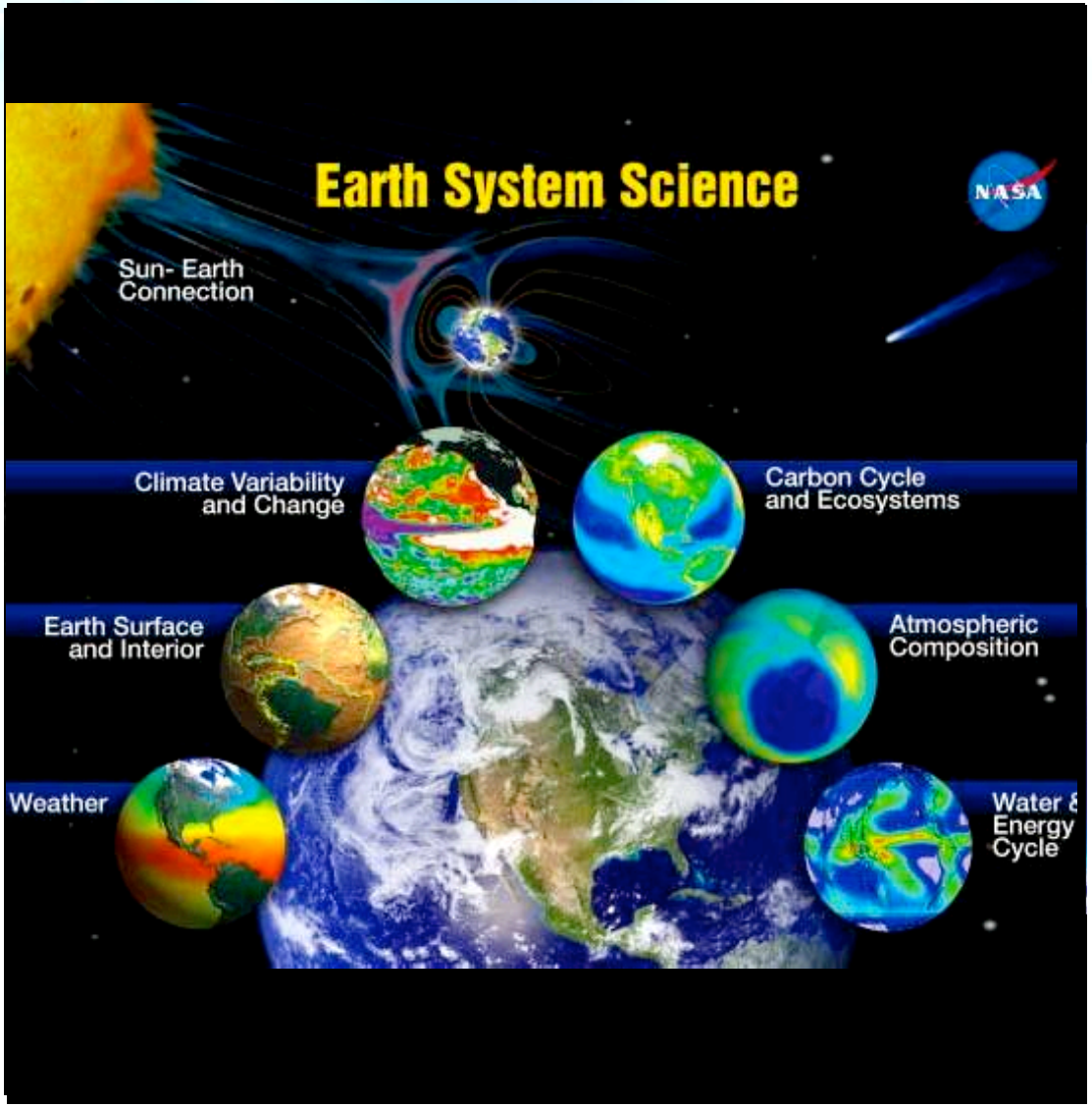


- The Global Grid
  - Decentralization
    - Peer-to-Peer nets
      - Machine-to-machine
        - Automated workflows
          - Distributed execution
            - Dynamic load balancing
              - Grid web services
                - Multi-scale integration
                  - Plug-and-play software



SciFlo: Scientific Knowledge Flow

The Vision of Earth System Science



- **Characterize**
  Earth's varied behavior
- **Understand**
  the Earth as an integrated system
- **Predict**
  Earth's response to complex forcings

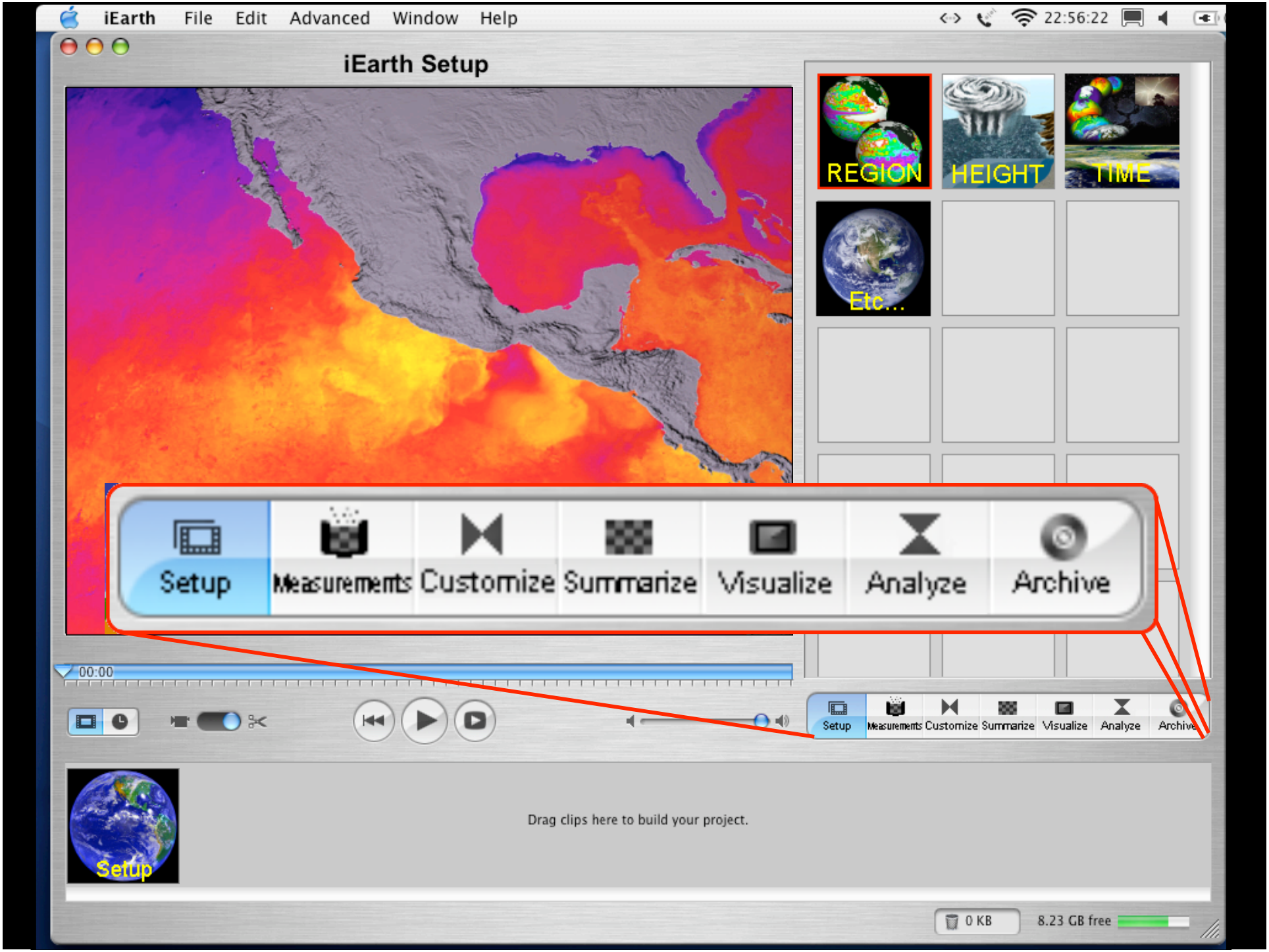
SciFlo: Scientific Knowledge Flow

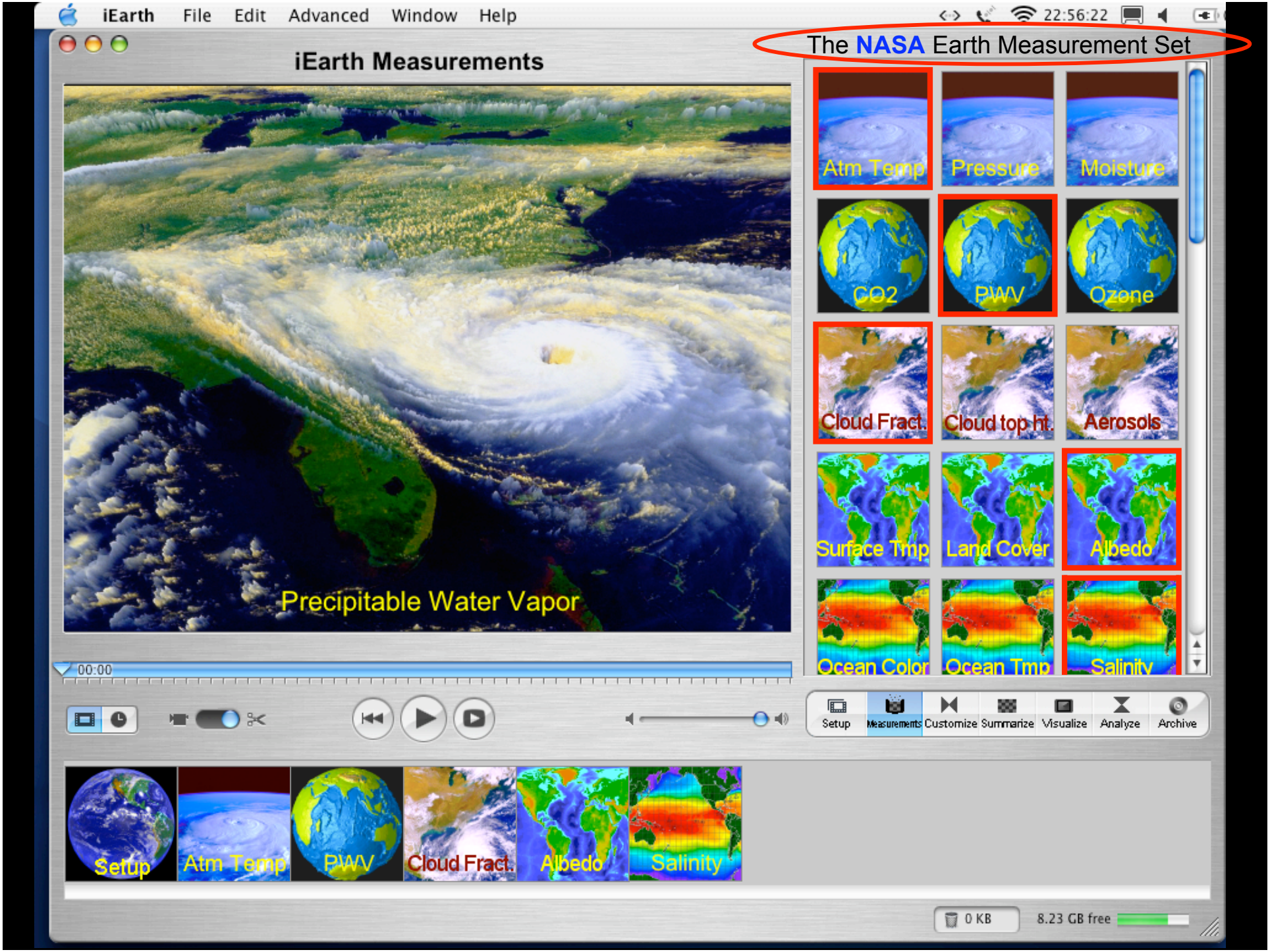
Today's Earth Science IT Challenges

- Coping with vast and diverse data sets:
  - Locating the right products (Data Discovery)
  - Retrieving large data volumes swiftly
  - Fusing diverse, incommensurate products
  - Visualizing massive multidimensional data
  - Discovering knowledge: Summarize/Analyze/Mine…
  - Predicting: Data Assimilation, Earth System Modeling: Tools / Environments / Frameworks
- Sample research scenario Today: Multi-year effort for a modest, cross-instrument study


Finder File Edit View Go Window Help
83:17:14
(Charged)
Tue 11:25 PM
iEarth


iEarth
File
Edit
Advanced
Window
Help
22:56:22
New Project... ⌘N
Open Project... ⌘O
Open Recent
Save Project ⌘S
Save Frame As... ⌘F
Import... ⇧⌘I
Export... ⇧⌘E
Show Info ⌘I
Empty Trash...
Layout
Please Begin
00:00
Setup
Measurements
Customize
Summarize
Visualize
Analyze
Archive
Drag clips here to build your project.
0 KB
8.23 GB free


iEarth File Edit Advanced Window Help
22:56:22
iEarth Setup
REGION
HEIGHT
TIME
Etc...
Setup
Measurements
Customize
Summarize
Visualize
Analyze
Archive
00:00
Drag clips here to build your project.
Setup
0 KB
8.23 GB free


iEarth File Edit Advanced Window Help
22:56:22
iEarth Measurements
The NASA Earth Measurement Set
Precipitable Water Vapor
Atm Temp
Pressure
Moisture
CO2
PWV
Ozone
Cloud Fract.
Cloud top ht.
Aerosols
Surface Tmp
Land Cover
Albedo
Ocean Color
Ocean Tmp
Salinity
00:00
Setup
Measurements
Customize
Summarize
Visualize
Analyze
Archive
Setup
Atm Temp
PWV
Cloud Fract.
Albedo
Salinity
0 KB
8.23 GB free

SciFlo Engine

- iEarth Vision will be enabled by the open-source SciFlo Engine.
- Automate large-scale, multi-instrument science processing by *authoring* a dataflow document that specifies a *tree* of *executable* operators.
  - iEarth Visual Authoring Tool
  - Distributed Dataflow Execution Engine
  - Move operators (executables) to the data.
  - Built-in reusable operators provided for many tasks such as subsetting, co-registration, regridding, data fusion, etc.
  - Custom operators easily plugged in by scientists.
  - Leverage convergence of Web Services (SOAP) with Grid Services (Globus v3.2).
- Hierarchical namespace of objects, types, & operators.
  - sciflo.data.EOS.AIRS.L2.atmosphericParameters
  - sciflo.operator.EOS.coregistration.PointToSwath

Outline

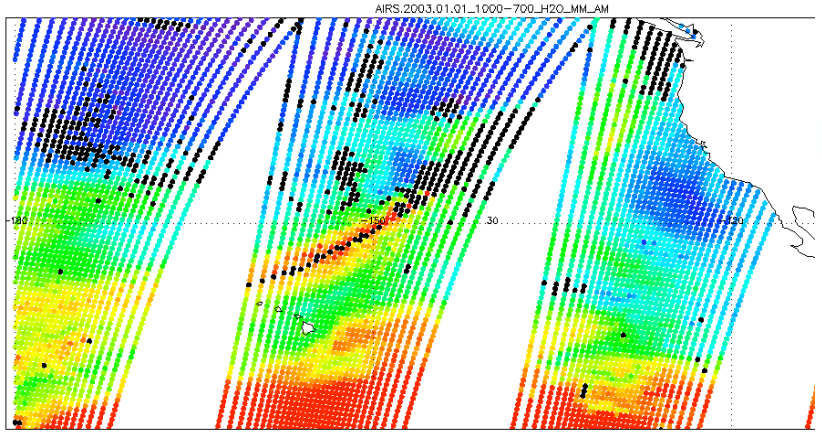
- Enabling Technologies
  - Web Services: SOAP
  - Grid Services: OGSI & Globus v3.2
  - Parallel dataflow engines
  - Semantic Web: OWL inference using metadata
- SciFlo Distributed Dataflow System
  - Loosely-coupled distributed computing using Web (SOAP) and Grid services
  - Specifying a processing stream as an XML document
  - Dataflow engine for automated execution and load balancing
- Multi-Instrument Earth Science
  - Motivating Example: Compare the temperature & water vapor profiles retrieved from AIRS (Atmospheric Infrared Sounder) swaths and GPS limb soundings.

SciFlo: Scientific Knowledge Flow

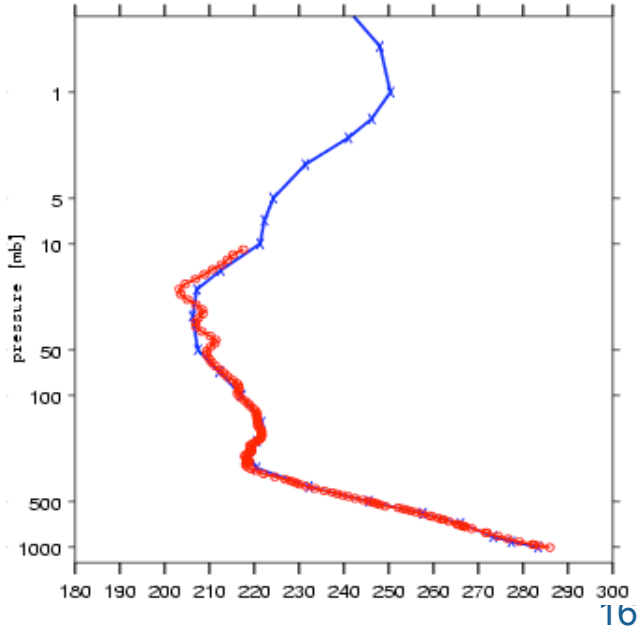
Motivating Examples

- **Data Discovery & Access**
  - What atmospheric temperature data (from all EOS instruments) is available in the tropical Pacific on Jan. 3, 2004? Retrieve it.

AIRS Swaths



- **Multi-Instrument Science Questions**
  - Compare the AIRS temperature profiles to the GPS temperature profiles and to the ECMWF model grid over the oceans.



Three Generations of the Web

- 1st: Static HTML pages with Pictures!
  - Hyperlinks: click and jump!
  - Easy authoring of text with graphics (HTML layout).
  - Killer App: Having your own home page is hip.
  - Cons: Too static, One-way communication.
- 2nd: Dynamic HTML with streaming audio & video
  - Browser as an all-purpose, ubiquitous user interface.
  - Fancy clients using embedded Java applets.
  - Killer App: Fill out your time card on-line.
  - Cons: Applets clunky, ease of authoring disappears, information is still HTML (semi-structured).
- 3rd: SOAP-based Web Computing & Semantic Web
  - Exchange structured data in XML format (no fragile HTML); semantics ("meaning") kept with data.
  - Programmatic interfaces rather than just GUI for a human.
  - Killer Apps: Grid Computing, automated data processing.

What is Simple Object Access Protocol (SOAP)?

- Distributed Computing by Exchange of XML Messages
  - Lightweight, Loosely-Coupled API
  - Programming language independent (unlike Java RMI)
  - Transport protocol independent
- Multiple Transport Protocols Possible
  - HTTP or HTTPS (HTTP using SSL encryption) POST
  - Email (SMTP), Instant Messaging (MQSeries, Jabber)
  - Store & Forward Reliable Messaging Services (Java JMS)
  - Even Peer-to-Peer (P2P) protocols
- SOAP Toolkits for all languages
  - Apache Axis for Java, modules for python/perl, Visual C# .Net.
- Web Service Description Language (WSDL)
  - Generate call to a service automatically from WSDL document.
  - Data types and formats expressed in XML schema.
  - Publish services in UDDI catalogs for automated discovery.

Why use SOAP to create scientific services?

- New paradigm of loosely-coupled distributed computing
  - SOAP messaging is exploding in the business world.
  - Used in Grid Computing: OGSI and Globus Toolkit v3.0
  - Asynchronous workflow (return results when available)
- Leverage XML standards (WSDL, UDDI, WS-*)
  - Service API is XML messages, not code.
  - Data types and formats expressed in XML schema.
- Exchanging large scientific data sets
  - Small objects in XML format
  - Even medium-size objects (2D grid slices) can be in XML.
  - Large objects as references (ftp or http URL) to HDF-EOS or netCDF binary containers (files).
- Security & authentication less important in science
  - Use HTTPS, Basic (user/password) Authentication, or WS-Security & WS-Authentication

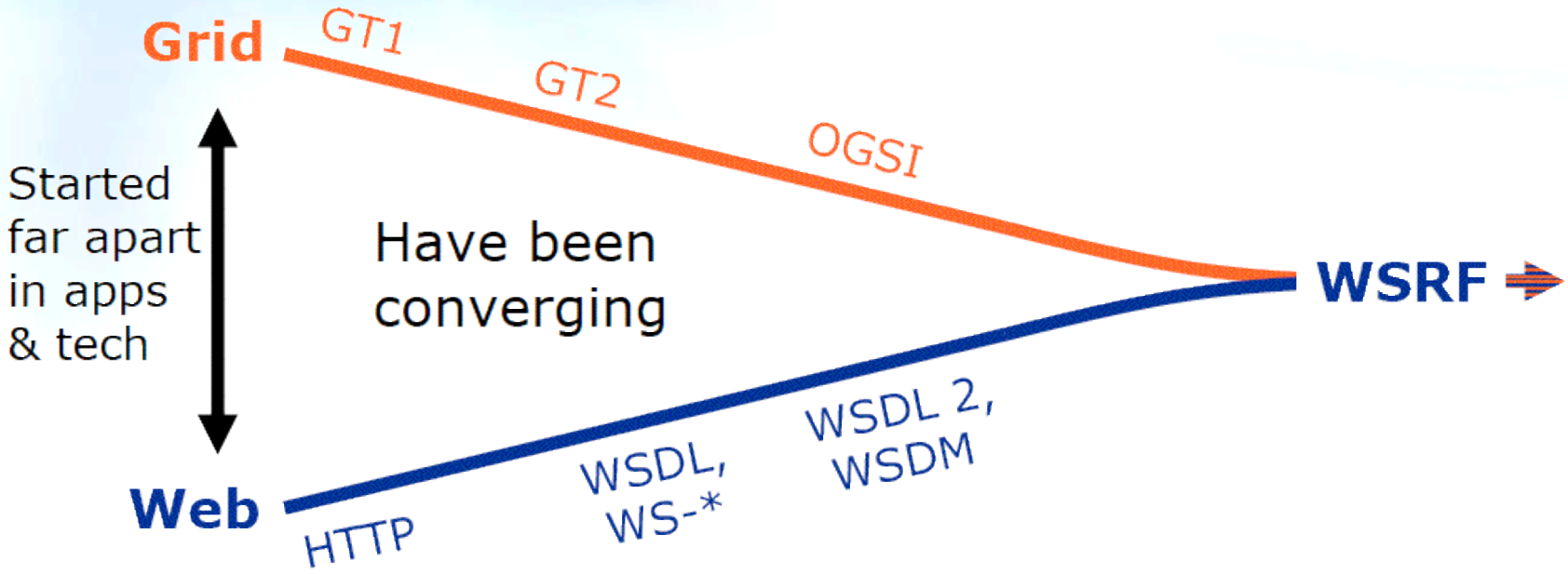
Third Generation of the Web

- SOAP-based Web Computing & Semantic Web
  - Exchange structured data in XML format (not HTML)
  - Semantics or “meaning” kept with the data
  - Emphasize programmatic interfaces
  - Web (Grid) Services
  - Leverage WS-Security and other WS-* standards
- Simple Object Access Protocol (SOAP)
  - Distributed Computing by Exchange of XML Messages
  - Lightweight, Loosely-Coupled API
  - Programming language independent
  - Multiple Transport Protocols Possible (HTTP, P2P)
  - Web Services Description Language (WSDL)
  - Publish Services in catalogs for automated discovery

Evolving Grid Computing Standards (I)

- History of Scientific Computing as a Utility
  - The Grid began as effort to tightly couple multiple super- or cluster computers together (e.g., Globus Toolkit v1 & v2).
  - Needed job scheduling, submission, monitoring, steering, etc.
  - SETI@HOME success
- OGSI: Open Grid Services Infrastructure
  - WS-Resource Framework (WSRF): Capabilities treated as storage or computing resources exposed on the web.
  - Globus v3.2 is open-source implementation using Java/C.
  - A service is Grid-enabled by inheriting from Java class.
  - Standard is complex and growing.
  - Challenge: Ease of installation & use.
- SciFlo is a lighter weight peer-to-peer (P2P) approach.

Evolving Grid Computing Standards (II)



[From Globus Toolkit "Ecosystem" presentation at GGF11 by Lee Liming]

Evolving Grid Computing Standards (I)

- History of Scientific Computing as a Utility
  - The Grid began as effort to tightly couple multiple super- or cluster computers together (e.g., Globus Toolkit v1 & v2).
  - Needed job scheduling, submission, monitoring, steering, etc.
  - SETI@HOME success
- OGSI: Open Grid Services Infrastructure
  - WS-Resource Framework (WSRF): Capabilities treated as storage or computing resources exposed on the web.
  - Globus v3.2 is open-source implementation using Java/C.
  - A service is Grid-enabled by inheriting from Java class.
  - Standard is complex and growing.
  - Challenge: Ease of installation & use.
- SciFlo is a lighter weight peer-to-peer (P2P) approach.

SciFlo: Scientific Knowledge Flow

Web Browser

Web Portal

Simulation Tool

Registration Service

Data Viewer Tool

Chat Tool

Credential Repository

Certificate authority

Data Catalog

Telepresence Monitor

Camera

Camera

A Compute Server

B Compute Server

C Database service

D Database service

E Database service

Application Developer	9
Off the Shelf	13
Globus Toolkit	0
Grid Community	0

Grid Applications

Users work with client applications

Application services organize VOs & enable access to other services

Collective services aggregate &/or virtualize resources

Resources implement standard access & management interfaces

SciFlo: Scientific Knowledge Flow

Grid Apps Using Globus Middleware

Web Browser

CHEF

Simulation Tool

Globus Index Service

Globus GRAM Compute Server

Globus GRAM Compute Server

Camera

Camera

Telepresence Monitor

Data Viewer Tool

CHEF Chat Teamlet

MyProxy

Certificate Authority

Globus MCS/RLS

Globus DAI Database service

Globus DAI Database service

Globus DAI Database service

Application Developer	2
Off the Shelf	9
Globus Toolkit	4
Grid Community	4

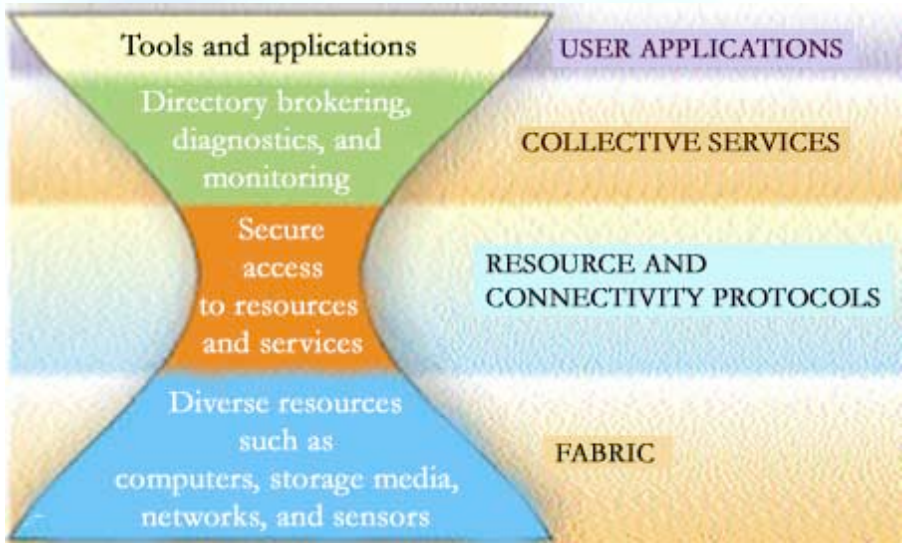
Users work with client applications

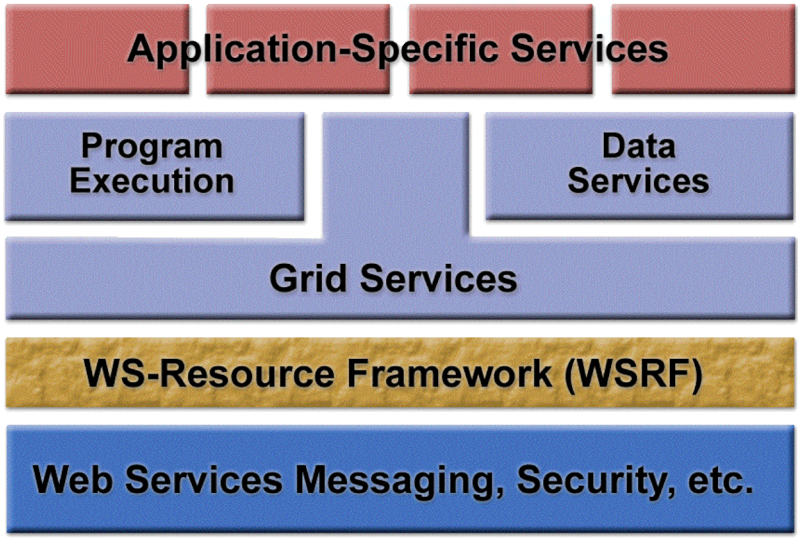
Application services organize VOs & enable access to other services

Collective services aggregate &/or virtualize resources

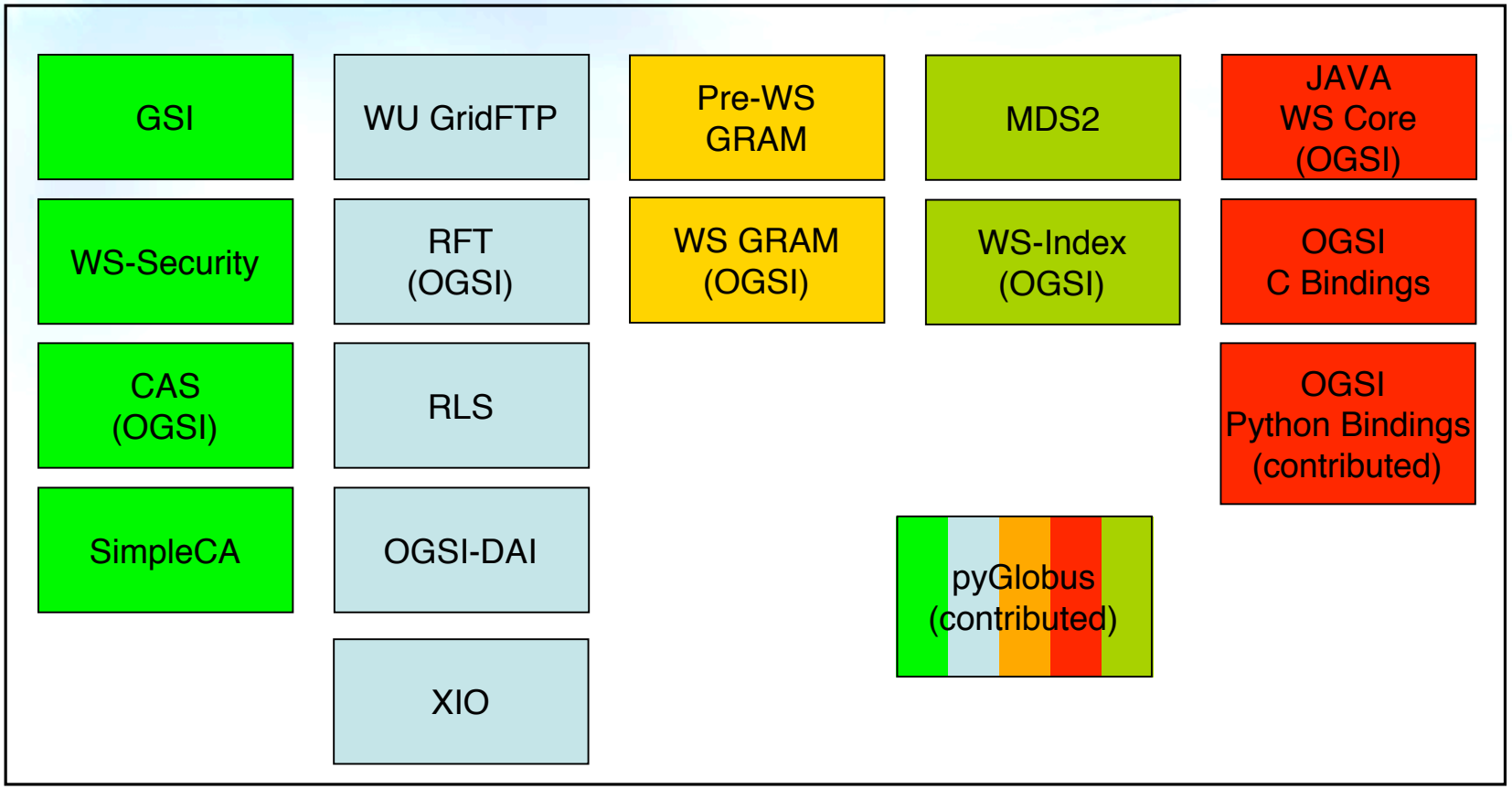
Resources implement standard access & management interfaces

Theory -> Practice

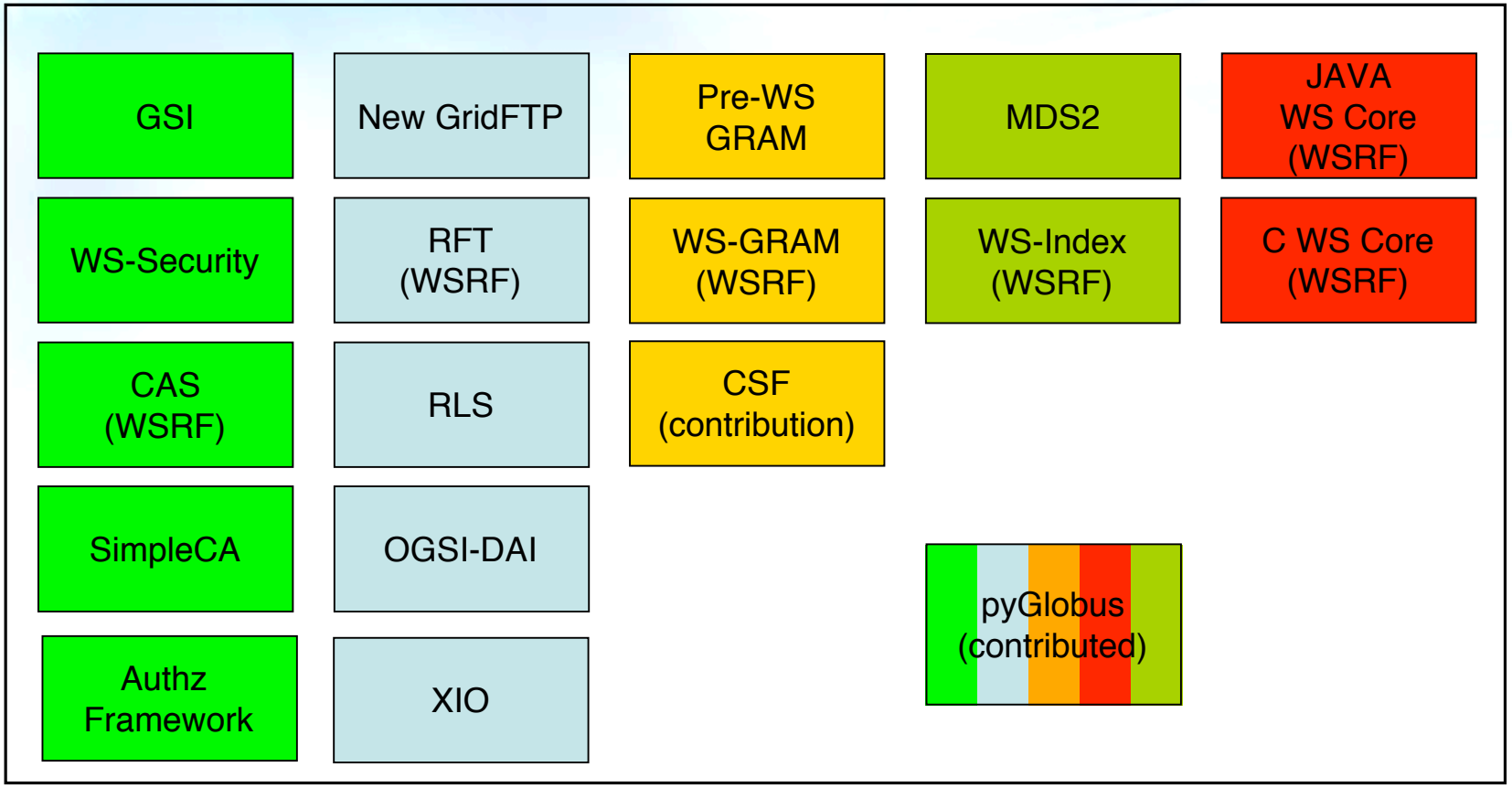




Components in Globus Toolkit 3.2



Planned Components in GT 4.0



Dataflow / Workflow Engines

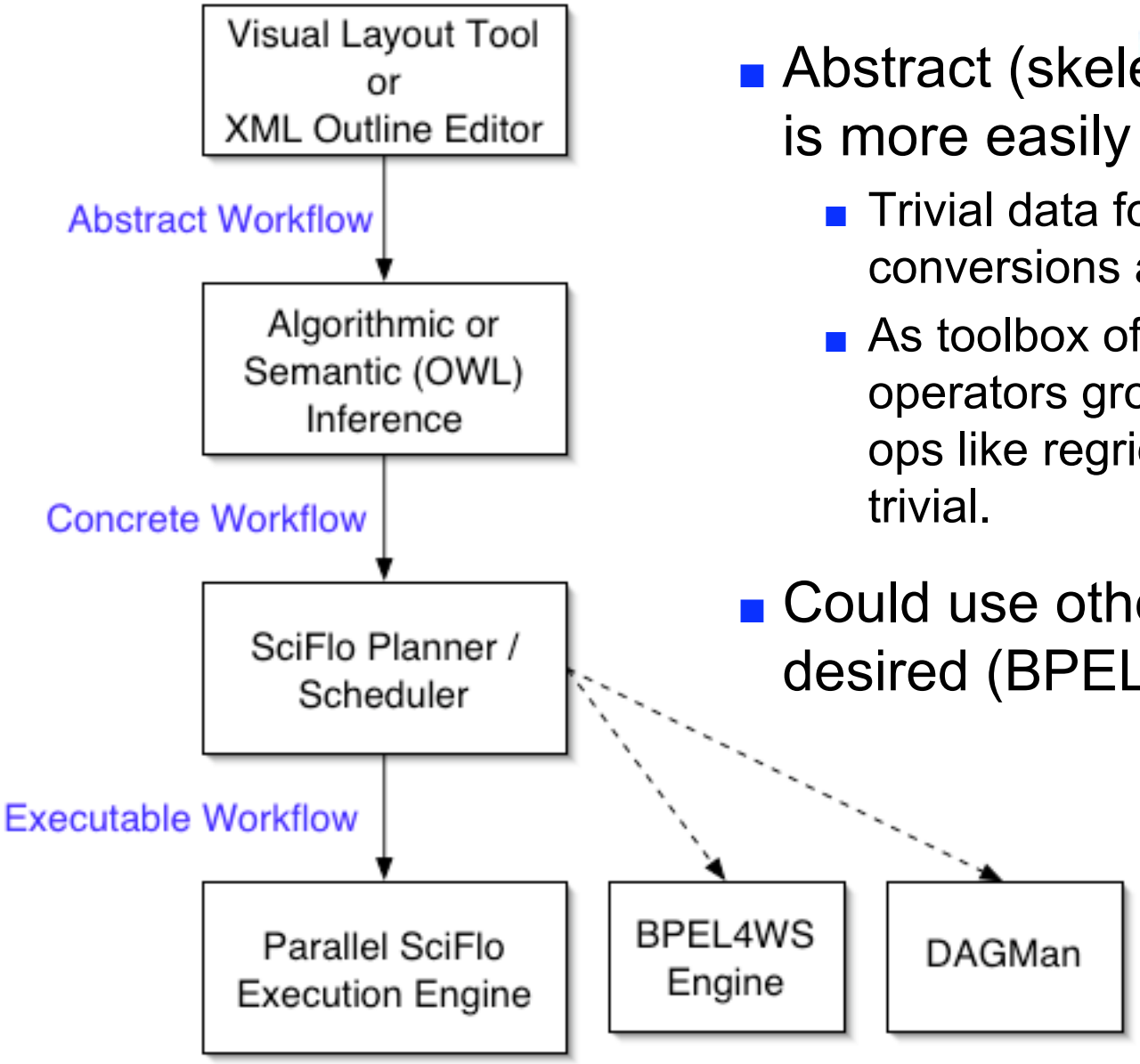
- Grid:
  - Schedule & submit cluster computing jobs
  - Operator tree is a Directed Acyclic Graph (DAG)
  - CONDOR, CONDOR-G, DAGMan
  - Globus Alliance Standards: GSI, GRAM, MDS, RLS, XIO, etc.
  - Chimera -> Pegasus -> DAGMan -> Executing Grid Job
- Web:
  - Several web choreography standards
  - IBM's Business Process Execution Language (BPEL4WS)
- Less convergence here than in OGSI/WSRF
  - Marketplace winners?
  - 10 workflow groups spoke at Global Grid Forum (GGF) meeting
  - Sciflo will use some Globus capabilities via python bindings (pyGlobus).

Semantic Web

- History
  - DAML: DARPA Agent Markup Language
  - OWL: Ontology Web Language (from DAML+OIL)
  - Numerous inference engines & ontologies being developed
- Semantics for Web Services
  - OWL-S: OWL-based Web service ontology
  - Describe properties & capabilities in computer-interpretable form (beyond WSDL).
- SciFlo Semantics & Inference
  - Use WSDL+ & OWL-S to describe local operators (executables), remote services, & grid computing jobs.
  - Discover & select operators to fill in missing steps in a dataflow.

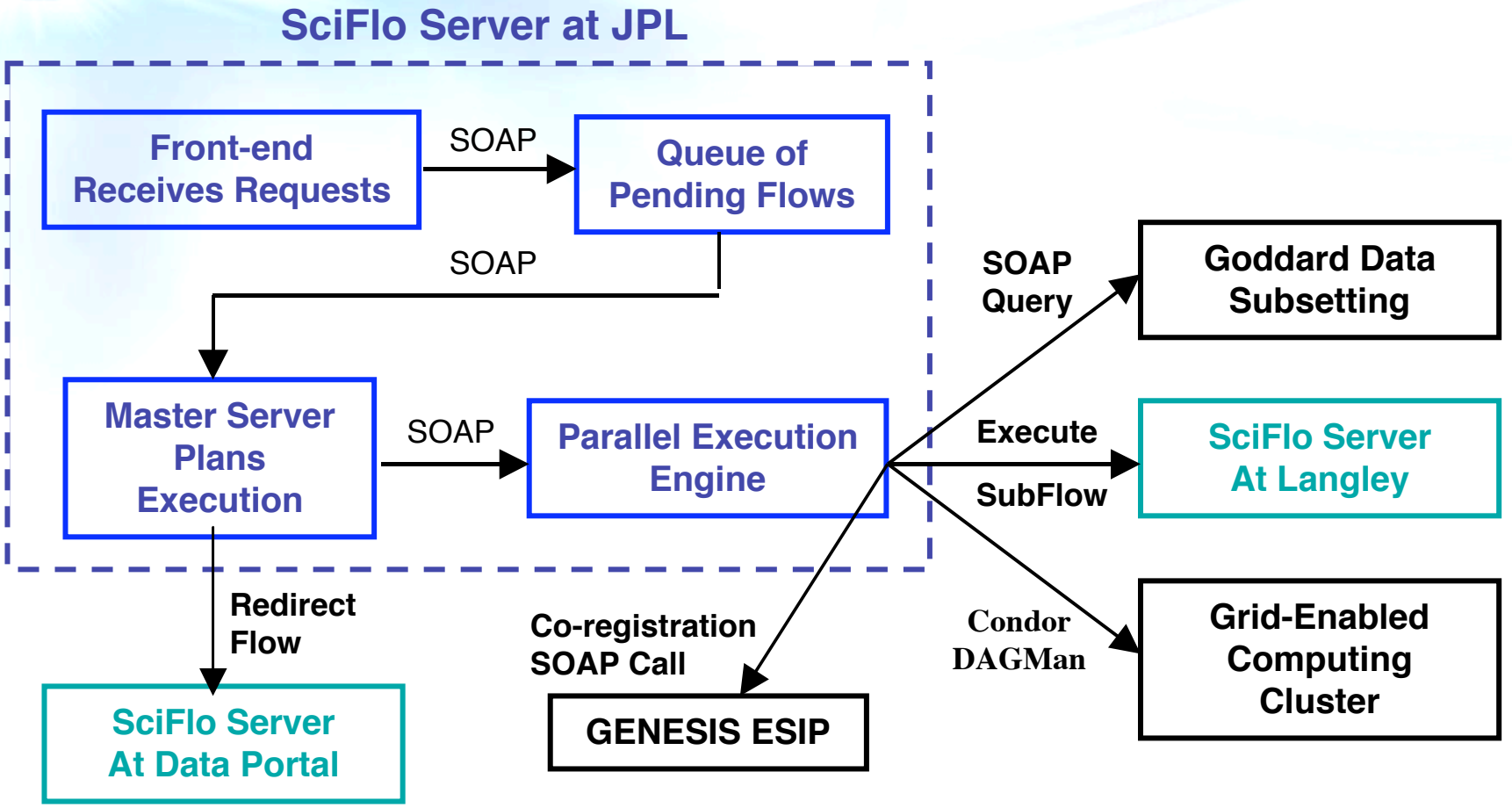
SciFlo: Scientific Knowledge Flow

Elaborating Workflow Documents



- Abstract (skeleton) Workflow is more easily authored.
  - Trivial data format & unit conversions auto-inserted.
  - As toolbox of known & reliable operators grows, even complex ops like regridding become trivial.
- Could use other backends if desired (BPEL, DAGMan).

Distributed Computing Using SciFlo



Inject data query or flow execution request into SciFlo network from any node.

SciFlo: Scientific Knowledge Flow

Enabling Multi-Instrument Earth Science

- Data Volume & Access
  - EOS retrieval products are huge binary files.
  - Need time, geolocation, & parameter subsetting at the source (Move operators to the data).
- Data Formats
  - Read/write HDFEOS & netCDF files as primary containers.
  - Define XML schemas for key datatypes (4DGeoParameterGrid).
  - Use XML metadata standards (ESML, CF climate conventions).
- Geolocation querying & co-registration
  - Reusable operators for point, swath, & grid overlaps.
  - Use standard & custom regridding toolkits.
- Software Reuse
  - Generic operators provided for dataflow assembly.
  - Any custom executable can automatically be a SciFlo operator.
  - Author XML dataflow documents, not code.

Elements of SciFlo (I)

- Hardware Paradigm – Clusters of Linux & Windows PC's.
- Stream data through an operator tree using SOAP calls.
- Move operators to servers local to large data sources.
  - Digitally-signed executables, python scripts, etc.
- One-click installation from web page
  - Installed by user (not admin/root) on Windows & Linux.
  - SciFlo client is really a "small" server (P2P possible).
- SciFlo Client for Ease of Use
  - Receives & validates XML dataflow document.
  - Submits to best or nearest server (may be itself).
  - Keeps track of queue of submitted jobs.
- "Smart" Visual Programming Tool
  - Lay out operator flow using a visual programming tool (Viper).
  - Hierarchical palettes of bundled operators & data sources.
  - Discover and suggest operators to fill in missing steps using semantic inference (OWL-S).

Elements of SciFlo (II)

- P2P Server Infrastructure
  - Distributed catalogs of data sources and operator bundles.
  - Permanent hierarchical names for data & operators.
  - Data & operator movement handled automatically by server.
- SciFlo Execution Steps:
  - Validate: Parse & validate XML doc. against schema.
  - Queue: Push flow into an execution queue if slaves busy.
  - Embellish: Infer concrete workflow from abstract workflow; insert simple unit or format conversions, or more complex operators.
  - Plan/Schedule: Construct execution plan & annotate flow document accordingly.
  - Start Execution: Parallel execution using master & slave servers.
  - Forward: Forward partially evaluated flow to another server for load balancing & locality optimization.
  - Freeze/Thaw: Store execution state while waiting; wake up when long-running operators complete.
  - Deliver: Return URLs pointing to results (or fault) to SciFlo client, which then pulls output files (via http, sftp, or GridFTP).

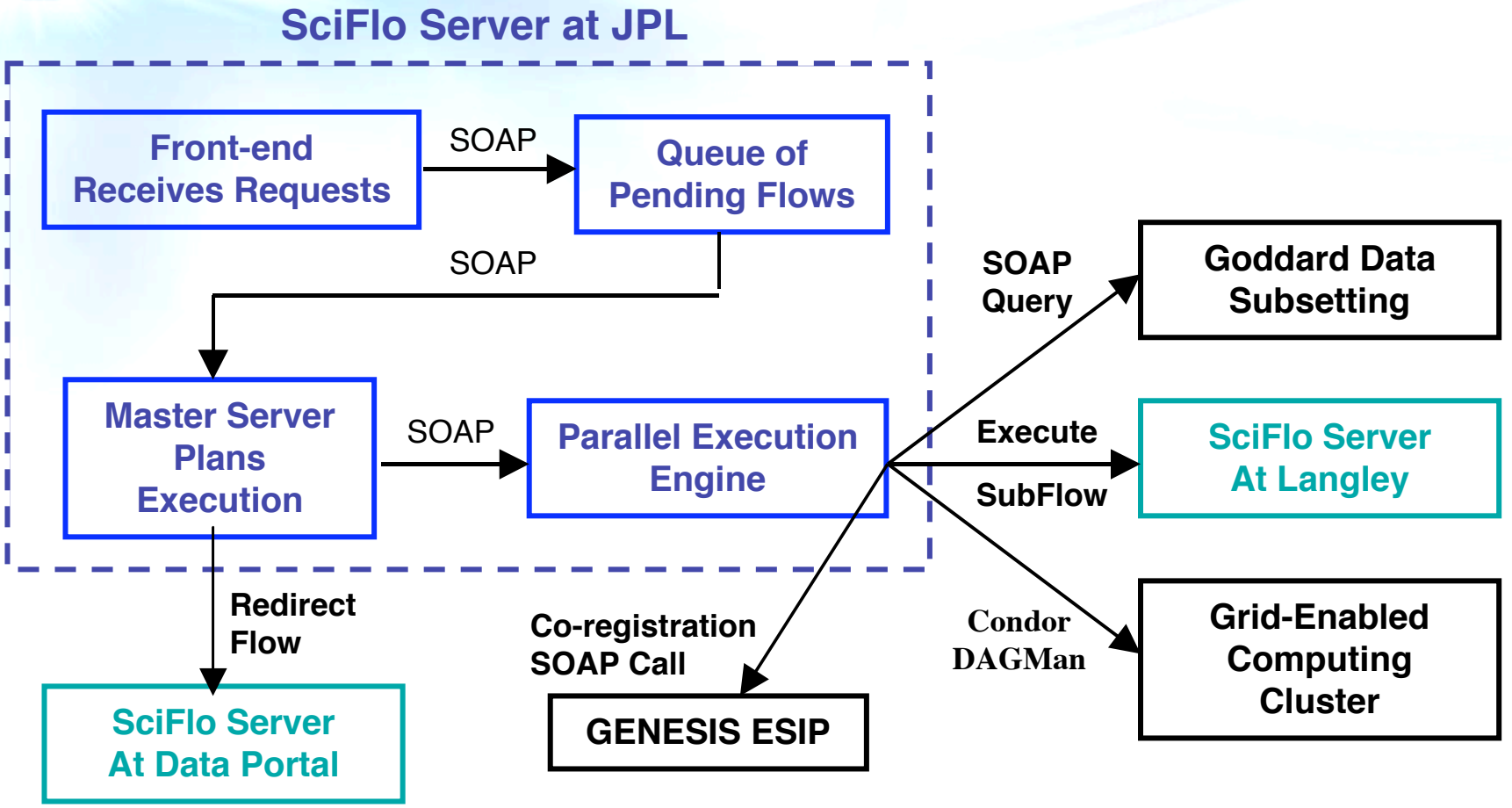
Server Architecture & Implementation

- Server Architecture
  - Group of interacting SOAP services (modules)
  - Modules can be written in different languages (C++, python).
  - Scheduler & Embellisher modules can be replaced by another SOAP service satisfying interface.
  - Can expose any flow as a new SOAP service or web form (reuse); create new SOAP/web services by authoring flow documents.
- Implementation Decisions
  - Architect using loosely-coupled SOAP calls/services.
  - Do not write everything in Java!
  - Use C/C++ & python instead (wrap read/write HDF & netCDF libraries into python, prototype server modules using python).
  - Leverage Climate Data Analysis Tools (CDAT), written in python.
  - Move intermediate datasets around in netCDF files (as binary containers), with XML metadata generated on demand.
  - Custom operators can be written as Matlab or IDL programs.

Parallel, Asynchronous Dataflow Execution

- Parallelism at many levels during execution:
  - Create multiple processes on a single server node.
  - Create multiple threads within each process to execute, monitor, & respond to status queries.
  - Launch a sub-flow or operator on a slave node within the local computer cluster.
  - Invoke a remote operator via a SOAP or http CGI call and wait for results.
  - Redirect a flow or sub-flow to a server that can access a large data source locally.
  - Partially evaluate a flow and then redirect.
  - Launch a large, CPU-intensive operator on a Grid-enabled cluster or supercomputer (Sciflo and Grid interoperability).
  - Invoke the same flow multiple times if the source operators(s) yield multiple input objects (implicit file parallelism).
  - Switch execution between active flows while waiting for results from “frozen” flows.

Distributed Computing Using SciFlo



Inject data query or flow execution request into SciFlo network from any node.

Data Access by Naming

- Permanent Hierarchical Names ("Holy Grail")
  - Naming Authority assigned at each namespace level
  - Distributed P2P namespace (P2P catalog lookup)
- Proper Names
  - AIRS Level2 Parameter Retrieval Dataset (granules):
    sciflo.data.EOS.AIRS.L2.atmosphericParameters (or metadata)
  - Generic Point-To-Swath Co-registration Operator:
    sciflo.operator.EOS.coregistration.PointToSwath
- Generic Names
  - Atmospheric Temperature Data:
    sciflo.data.atmosphere.temperature.profile (or .grid)
  - Name resolves to list of EOS datasets
  - Semantics attached (3DGeoParameterGrid of temperature)

SciFlo: Scientific Knowledge Flow

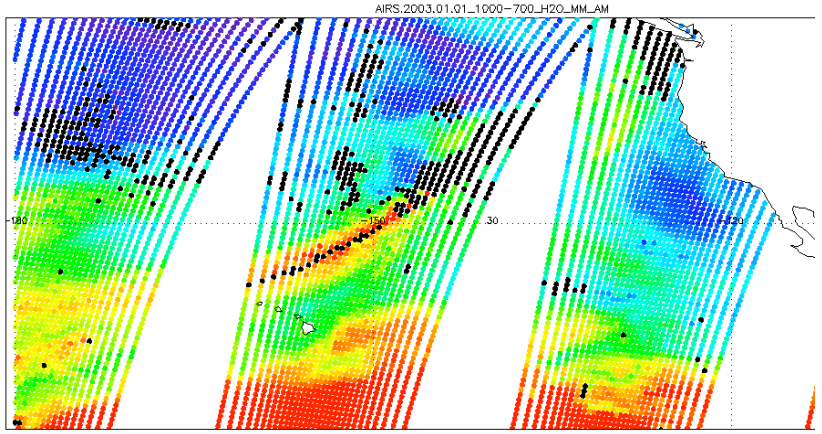
- SciFlo's Strength Lies in Combining Many Elements into a Single Open-Source System
  - Abstract XML dataflow documents translated to concrete flows.
  - Parallel dataflow execution engine
  - Semantic inference using XML metadata
  - Move operators to the data.
  - SOAP architecture, but also P2P functionality.
  - Every node is both client & server; easy node replication.
  - One-click installation onto server or desktop nodes.
  - Initiate grid computations from your desktop.
- Access data objects by naming them!
  - P2P Distributed Namespace of data sources & operators
- Server architecture
  - Group of interacting SOAP services (replaceable modules)
  - Implementation in XML, python, & C/C++ (not Java)
- Strength in Numbers: Let a million nodes bloom!

SciFlo: Scientific Knowledge Flow

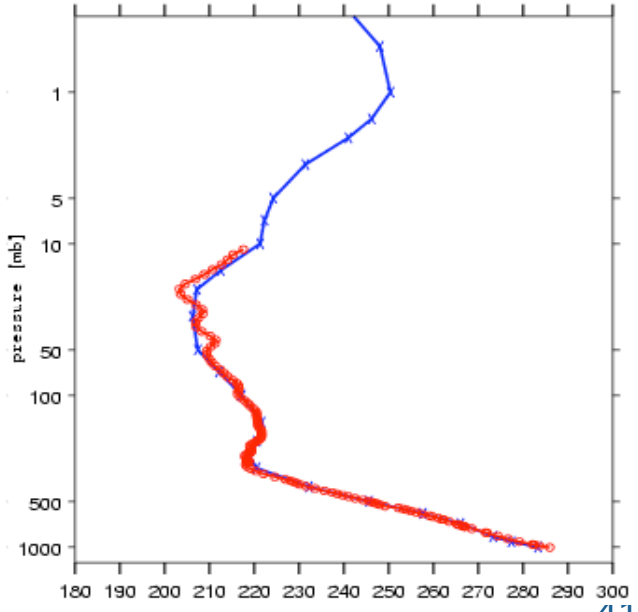
Motivating Examples

- Data Discovery & Access
  - What atmospheric temperature data (from all EOS instruments) is available in the tropical Pacific on Jan. 3, 2004? Retrieve it.

AIRS Swaths

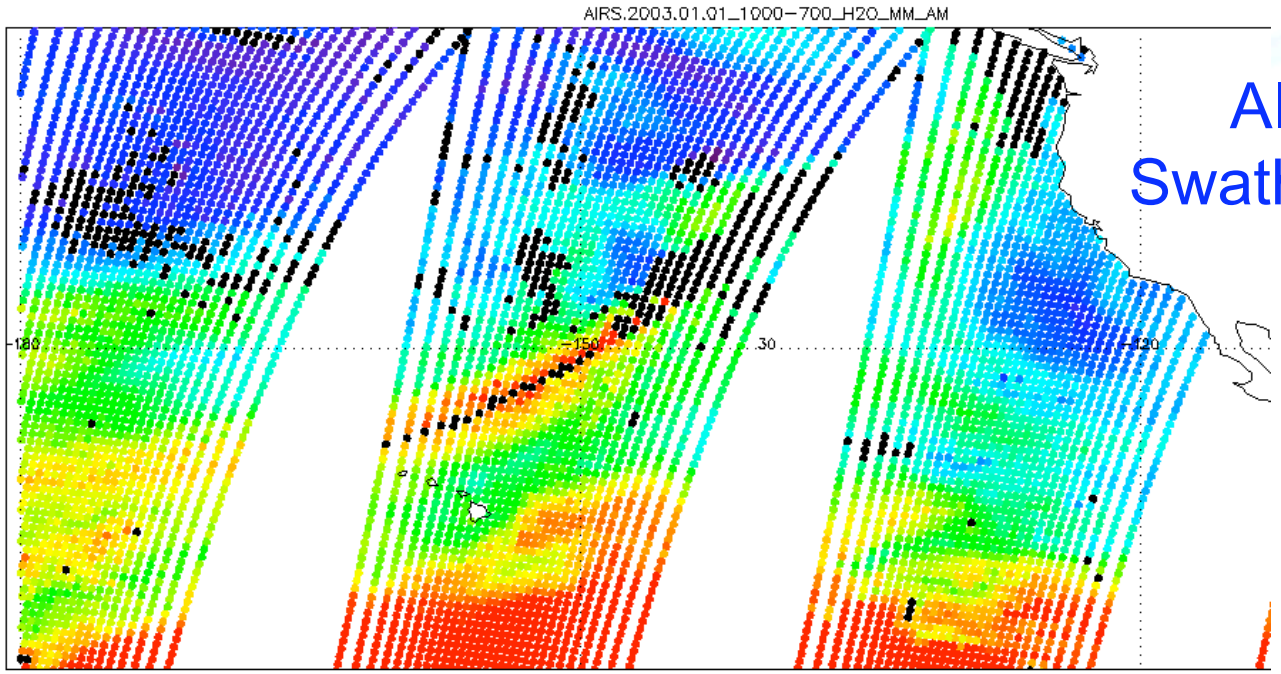


- Multi-Instrument Science Questions
  - Compare the AIRS temperature profiles to the GPS temperature profiles and to the ECMWF model grid over the oceans.

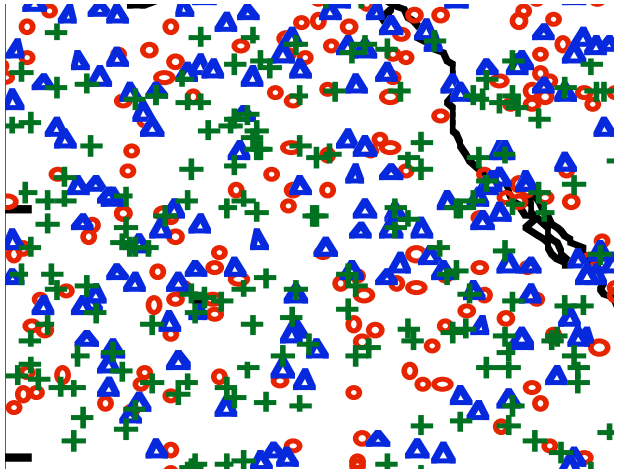


SciFlo: Scientific Knowledge Flow

AIRS/GPS Co-registration: Point to Swath



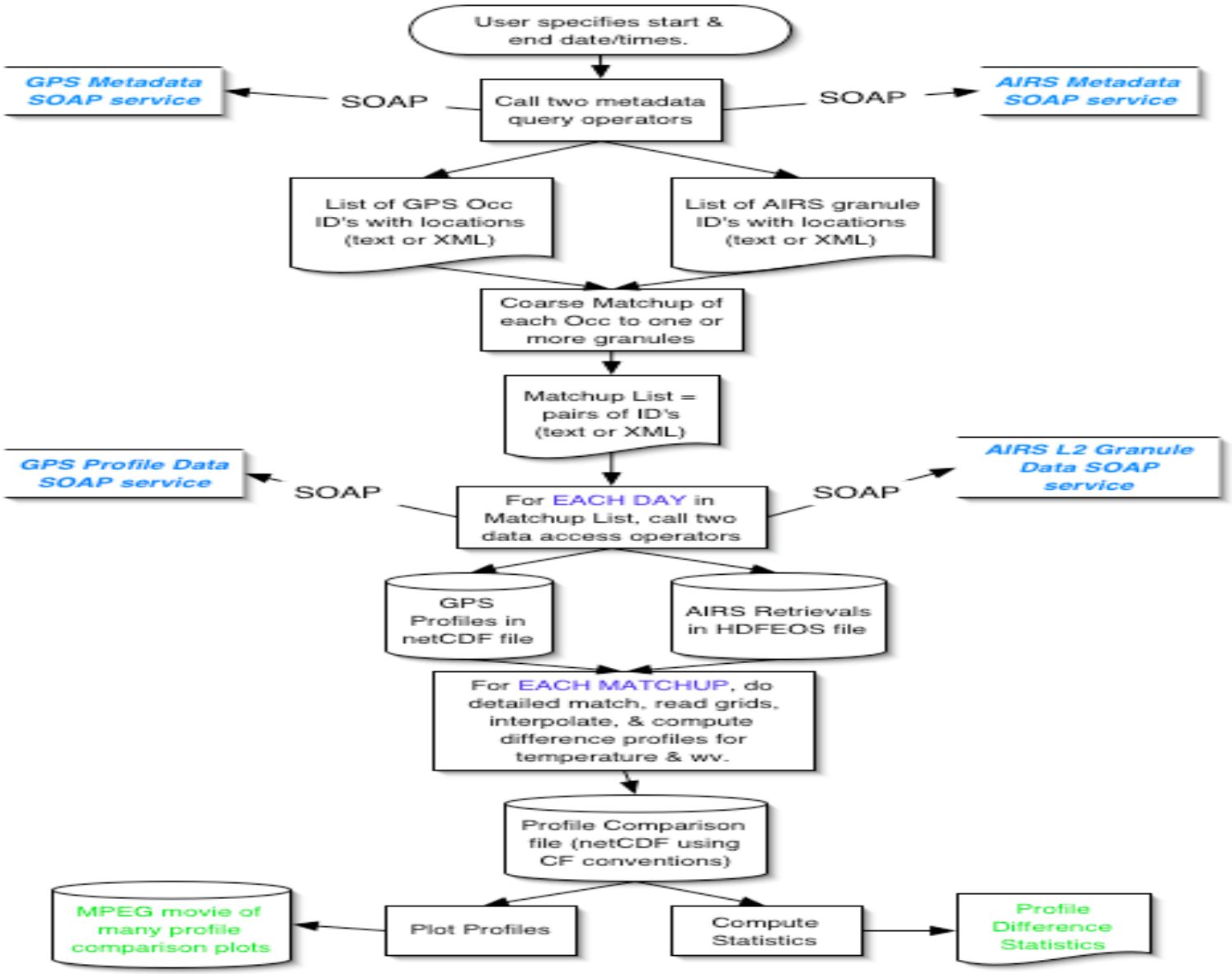
AIRS Level2
Swaths over Pacific



GPS Level2
Profile Locations

SciFlo: Scientific Knowledge Flow

AIRS versus GPS Flowchart

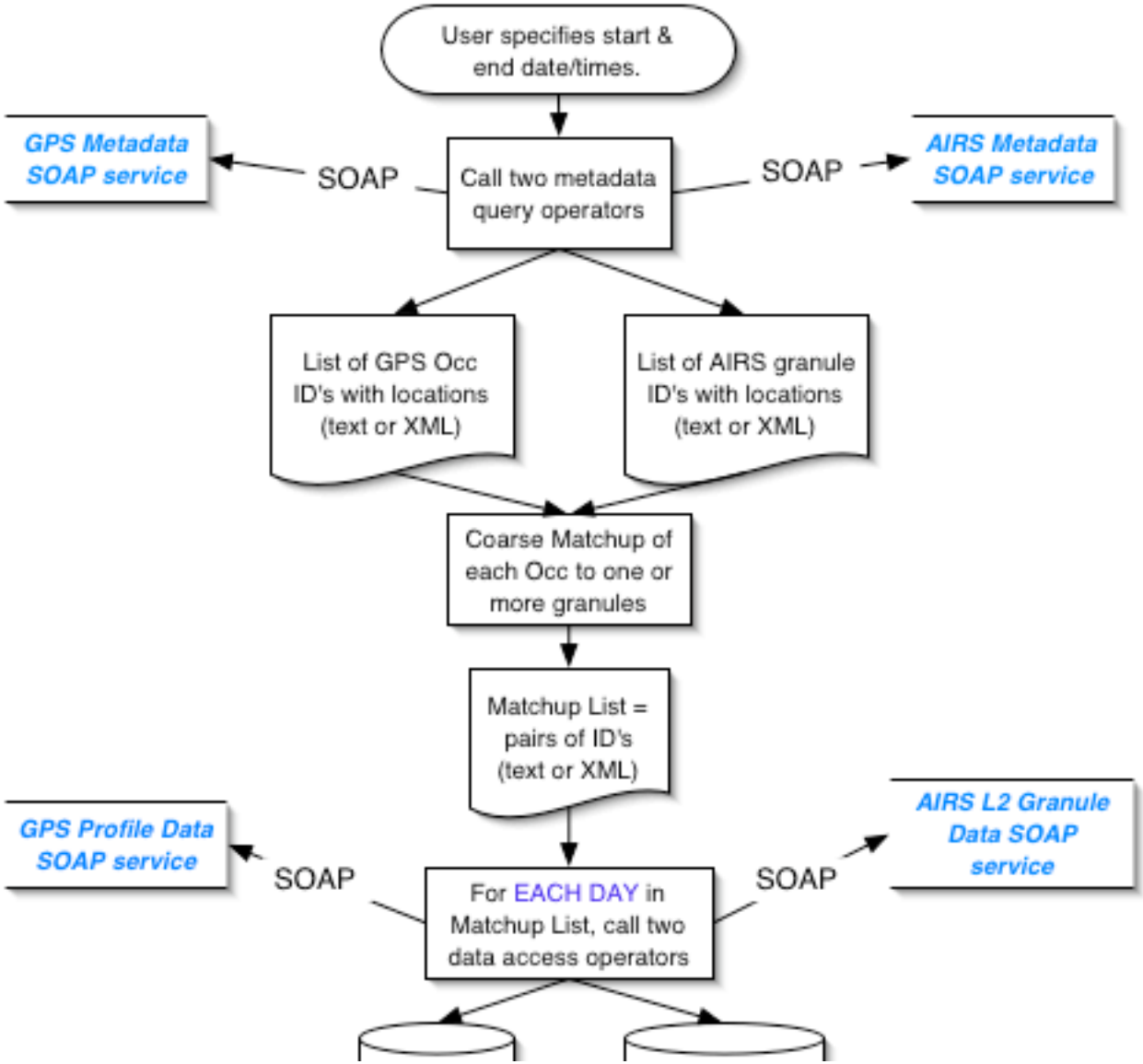


SciFlo Data Access (SOAP) Services

- getAirsMetaData(startTime, endTime, lat, lon)
  - Returns granule ID's and geolocation info. for AIRS swaths that intersect lat/lon point.
- getAirsUrlsFromGranuleIds(idList or xmlString)
  - Redirection server – Returns URLs pointing to AIRS granules in local cache or DataPool.
- getGpsMetaData(startTime, endTime, NW, SW, SE, NE)
  - Returns occultation ID's and geolocation info. for GPS limb scans that are inside a lat/lon region.
- getGpsProfiles(idList or xmlString)
  - Returns URL pointing to netCDF file containing GPS profiles for all occultation ID's in the list.
- getGpsProfile(occID)
  - Returns a single GPS profile in XML format.

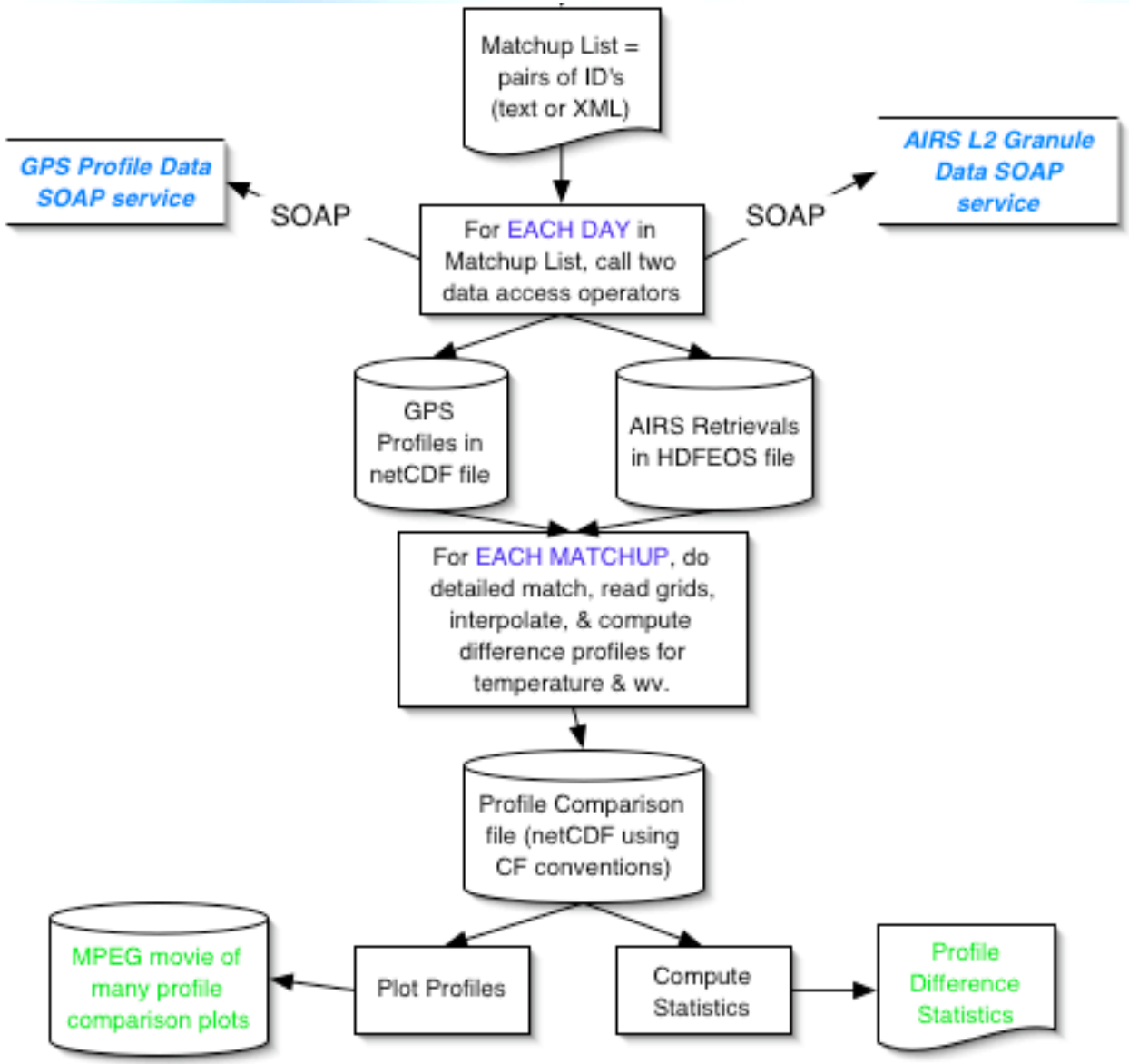
SciFlo: Scientific Knowledge Flow

AIRS/GPS Comparison Flowchart (I)



SciFlo: Scientific Knowledge Flow

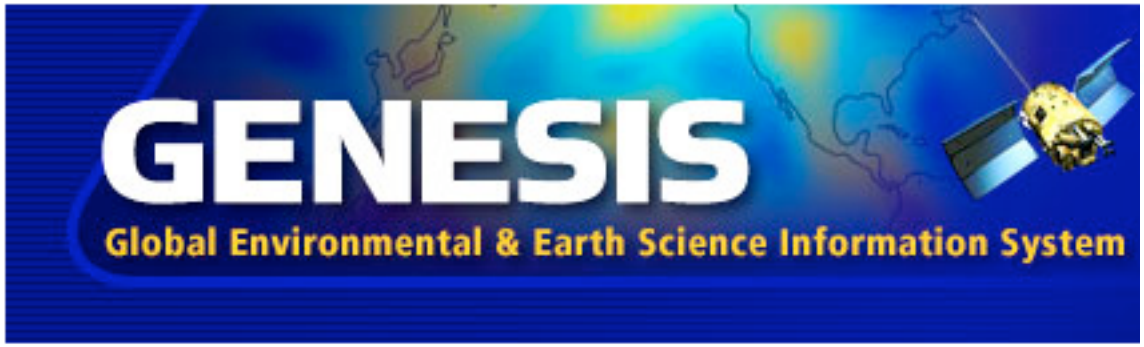
AIRS/GPS Comparison Flowchart (II)



Wilson et al., 07/[illegible]

SciFlo: Scientific Knowledge Flow

AIRS & GPS Temperature Matchup Demo



you are at: home » Registered User's Area » GPS-AIRS Matchup Demo

Starting Date/Time	2003 / 01 / 03 00 : 00
Ending Date/Time	2003 / 01 / 03 23 : 59
Time Tolerance (seconds)	60
Location Tolerance (km)	1000
Priority	1
Retrieval Type Max (between 0 and 100)	10
Land Fraction Min	0
Land Fraction Max	.1
Output whole swath?	☐

OK

- **Interface:** HTML web form auto-generated from XML dataflow doc.
- **Input:** User enters start/end time & other co-registration criteria.
- **Flow Execution:** Calls 2 SOAP data query services & total of 8 operators on 4 computers.

SciFlo: Scientific Knowledge Flow

AIRS & GPS Temperature Matchup Demo

you are at: home » Registered User's Area » GPS-AIRS Matchup Demo

Start Date/Time:	2003-01-03 00:00
End Date/Time:	2003-01-03 23:59
Time Tolerance (seconds):	60
Location Tolerance (km):	1000.0
Priority:	1
Retrieval Type Max (between 0 and 100):	10
Land Fraction Min:	0.0
Land Fraction Max:	0.1
Output whole swaths:	False

Checking GPS-AIRS matchup session...done (matchup file already exists and is current).
Number of matches found: 4
Getting matchup plots...done.
Getting swf movie of plots...done.

Finished processing.

Click here to download the NetCDF file.

Click here to download tgz of postscript plots.

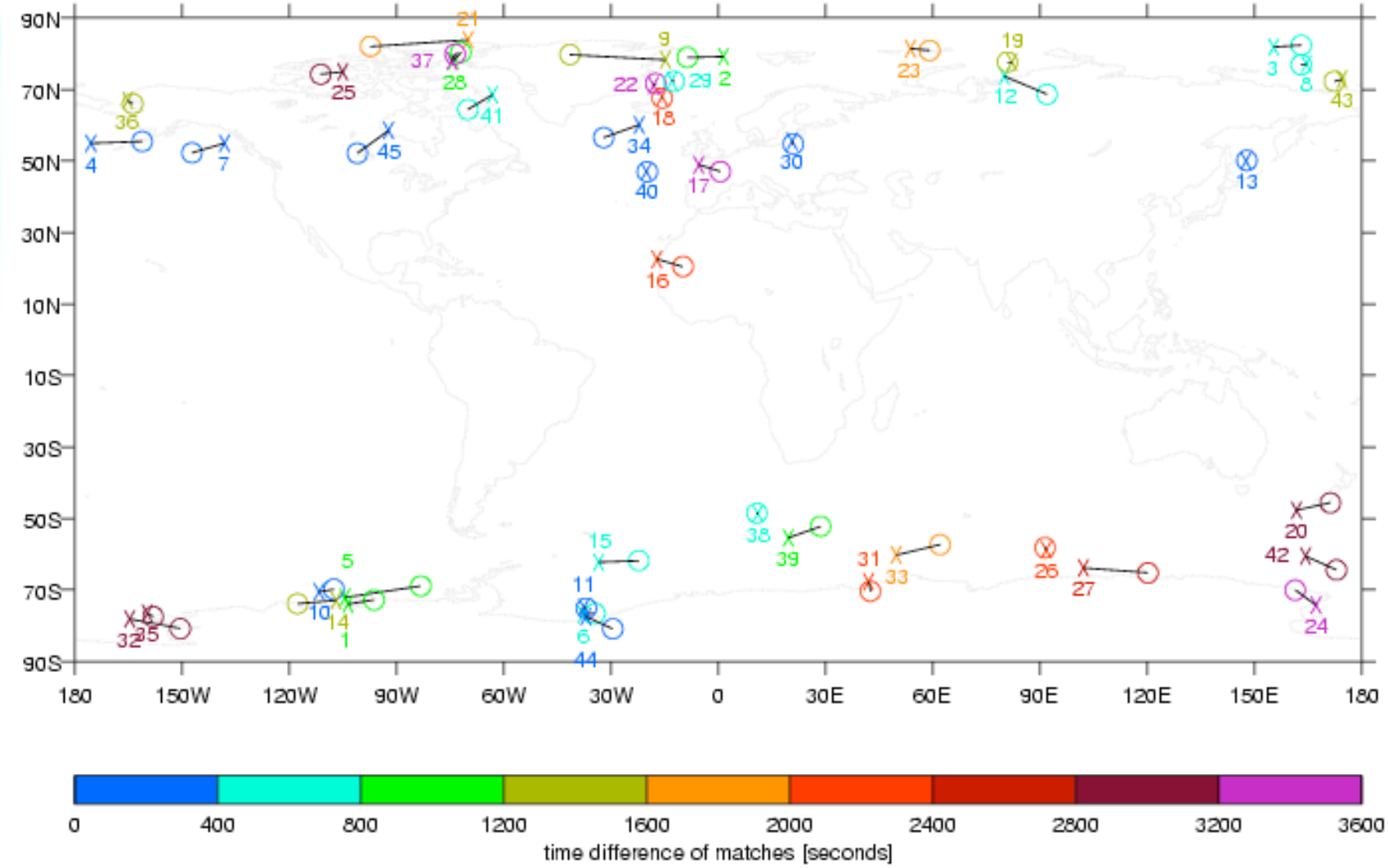
Click here to create and download mpeg movie (may take a while).

Flash movie:

- **Results Page:** Shows status updates during execution and then final results.
- **Caching:** Reuse intermediate data products or force recompute.
- **Results:** Merged data in netCDF file & plots as Flash movie.

SciFlo: Scientific Knowledge Flow

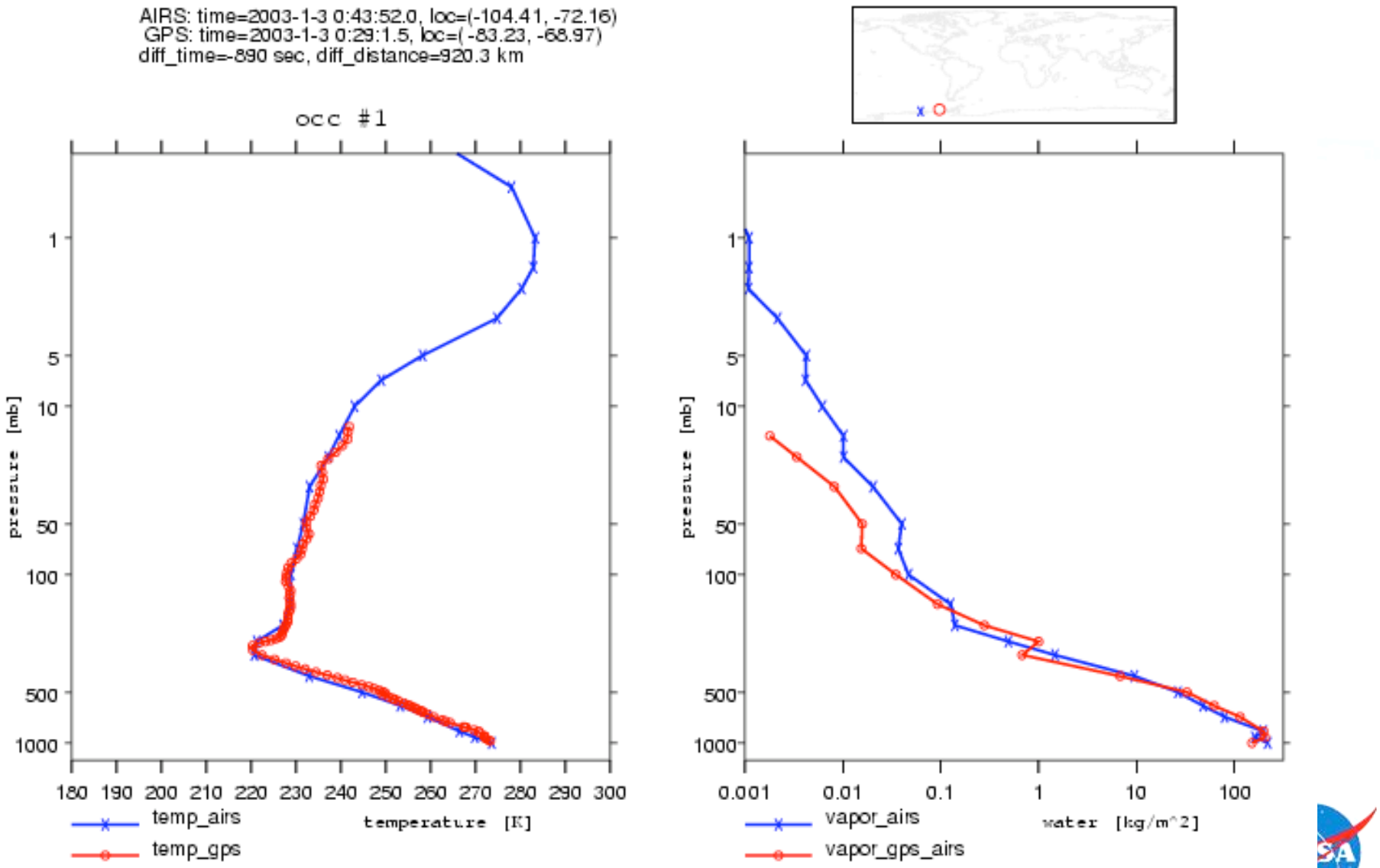
AIRS/GPS Matchups



Wilson et al., 07/20/04

SciFlo: Scientific Knowledge Flow

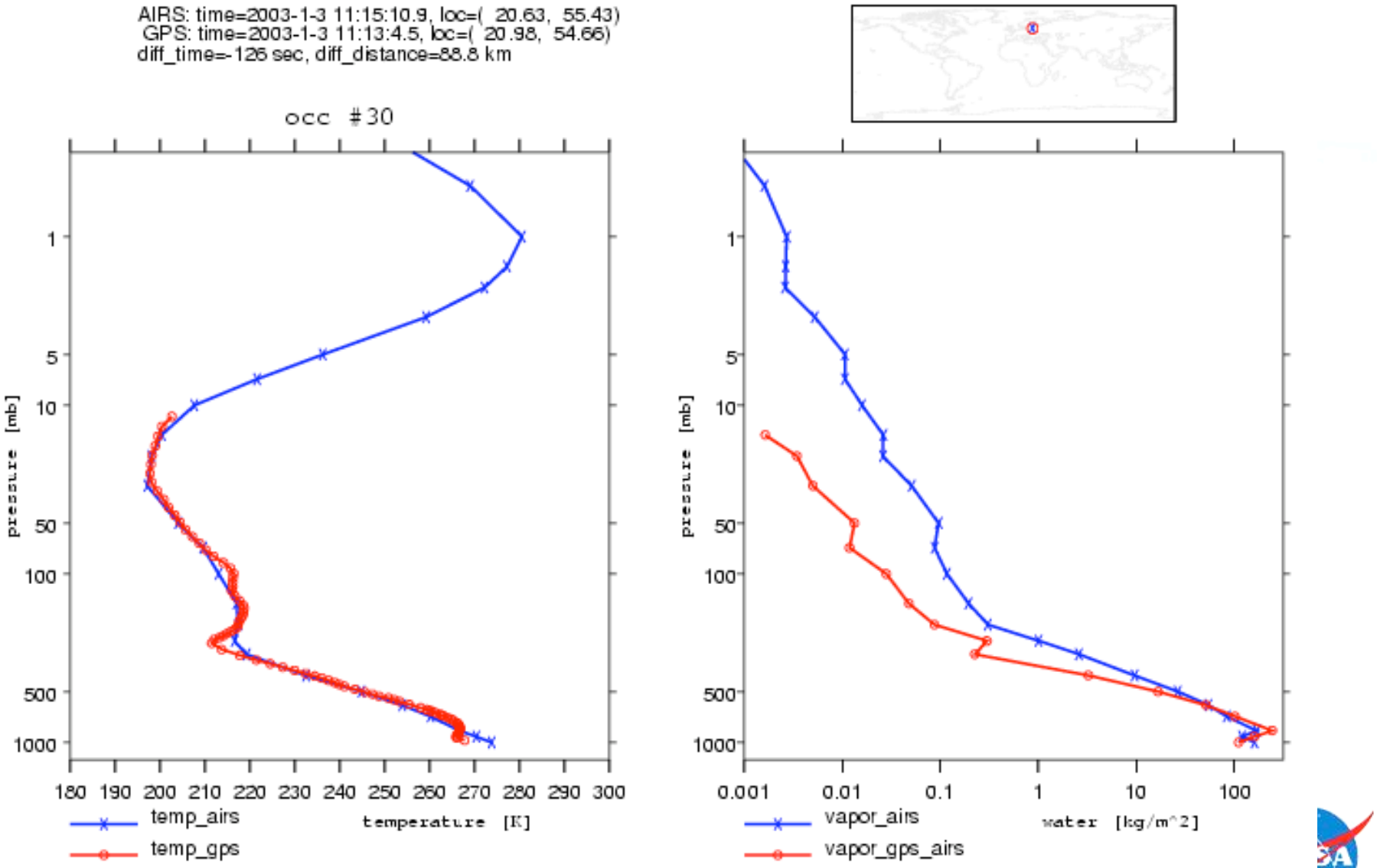
AIRS/GPS Temperature & Water Vapor Comparison Plots



Wilson et al., 07/20/04

SciFlo: Scientific Knowledge Flow

AIRS/GPS Temperature & Water Vapor Comparison Plots



Wilson et al., 07/20/04

Summary

- SciFlo's Innovation Lies in Combining Many Elements into a Single Open-Source System
  - Abstract XML dataflow documents
  - Semantic inference using XML metadata
  - Parallel dataflow execution engine
  - Move operators to the data.
  - Every node is both client & server; easy node replication.
  - SOAP architecture, but also P2P functionality.
  - Initiate grid computations from your desktop.
- Goal: SciFlo nodes inside all Science Data Centers
- Multi-Instrument Earth Science
  - Instrument Cross-Comparisons
  - Multi-Instrument Science Portals
  - Large-scale multivariate statistical studies and verification of weather/climate models.