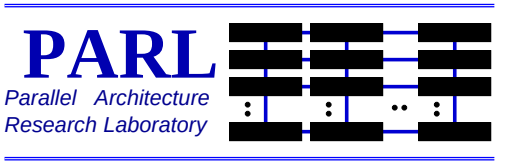
CERSe
a Tool for High Performance
Remote Sensing Application Development

Nathan DeBardeleben
Walter B. Ligon III
Sourabh Pandit
Dan C. Stanzione Jr.

presented by Nathan DeBardeleben

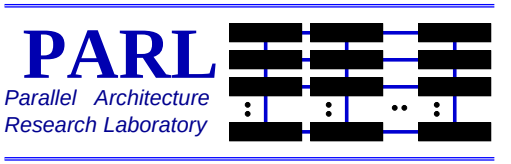
Parallel Architecture Research Laboratory (PARL)
Clemson University, Clemson, South Carolina
http://www.parl.clemson.edu/

Outline



- Problem
- CERSe
- Applications of CERSe
- Future Work

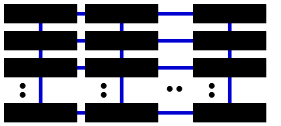
Problems Facing the RS Community



- Diversifying user base requires lower cost solutions
- Increasing temporal, spectral, and spatial resolution
- Massive data storage and processing power
- Parallel computers are difficult to program
- Rapid code development, execution, analysis, and maintenance

Proposed Solution

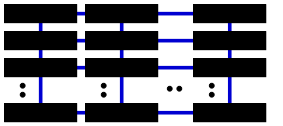
PARL
Parallel Architecture
Research Laboratory



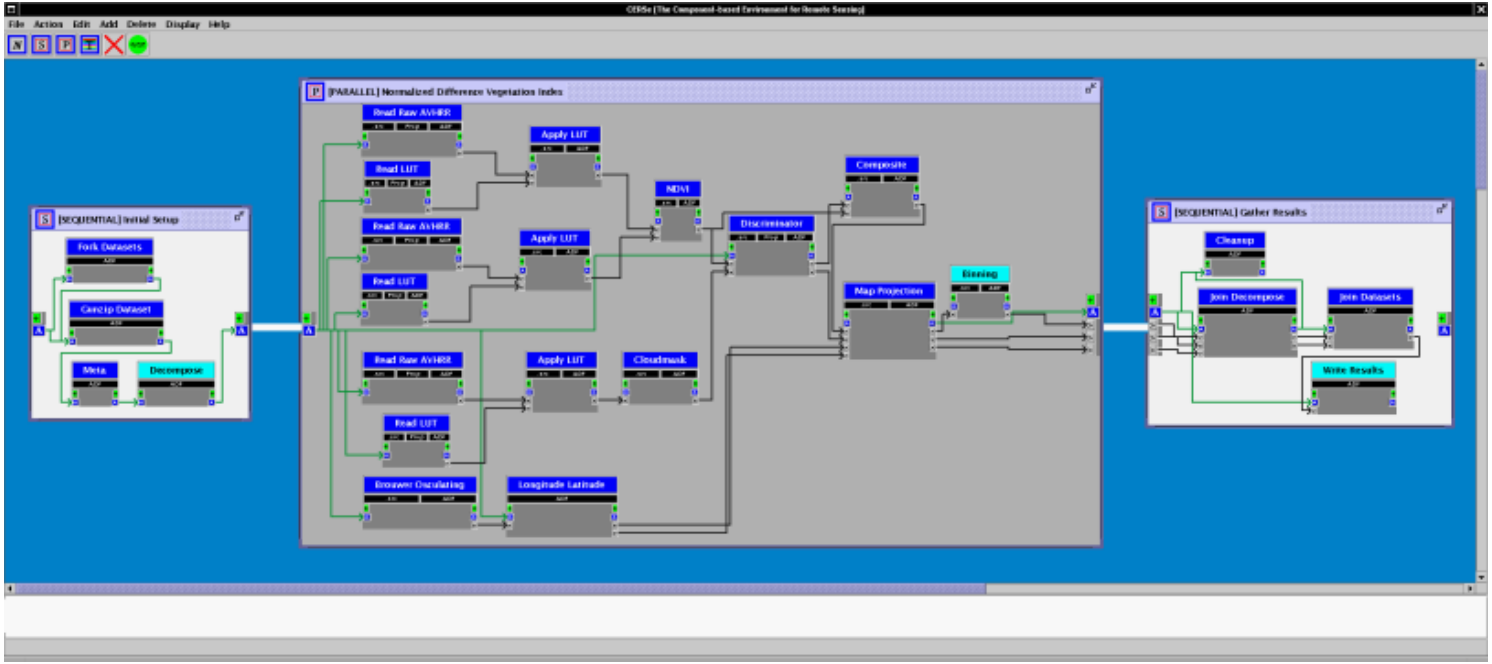
- A Problem Solving Environment (PSE)
- Provide:
  - High performance
    - Parallel computers
    - Usable without advanced parallel computing knowledge
  - Ease of use
    - Development and analysis tools
  - Extensibility
    - Code reuse

CERSe

PARL
Parallel Architecture
Research Laboratory

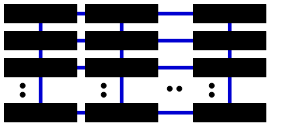


- The Component-based Environment for Remote Sensing
- Modular approach
  - Modules are subroutines in C or Fortran
  - Modules are placed onto a canvas to create a dataflow graph
  - Data passes between modules with an easy-to-use interface
- Programs are created on a workstation and then run on a parallel computer

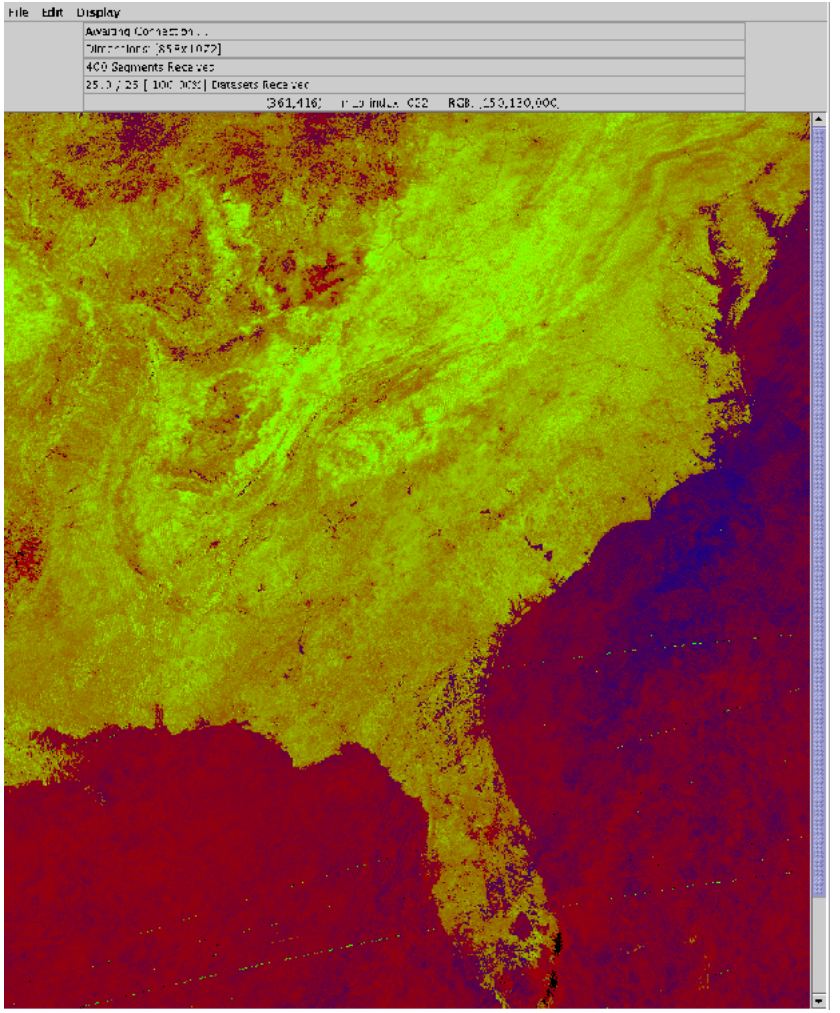

CEESe [The Component-based Environment for Remote Sensing]
File Actions Edit Add Delete Display Help
[PARALLEL] Normalized Difference Vegetation Index
[SEQUENTIAL] Initial Setup
Fork Datasets
Gunzip Dataset
Meta
Decompose
Read Raw AVHRR
Read LUT
Apply LUT
NDVI
Discriminator
Composite
Read Raw AVHRR
Read LUT
Apply LUT
Map Projection
Binning
Read Raw AVHRR
Read LUT
Apply LUT
Cloudmask
Brouwer Osculating
Longitude Latitude
[SEQUENTIAL] Gather Results
Cleanup
Join Decompose
Join Datasets
Write Results

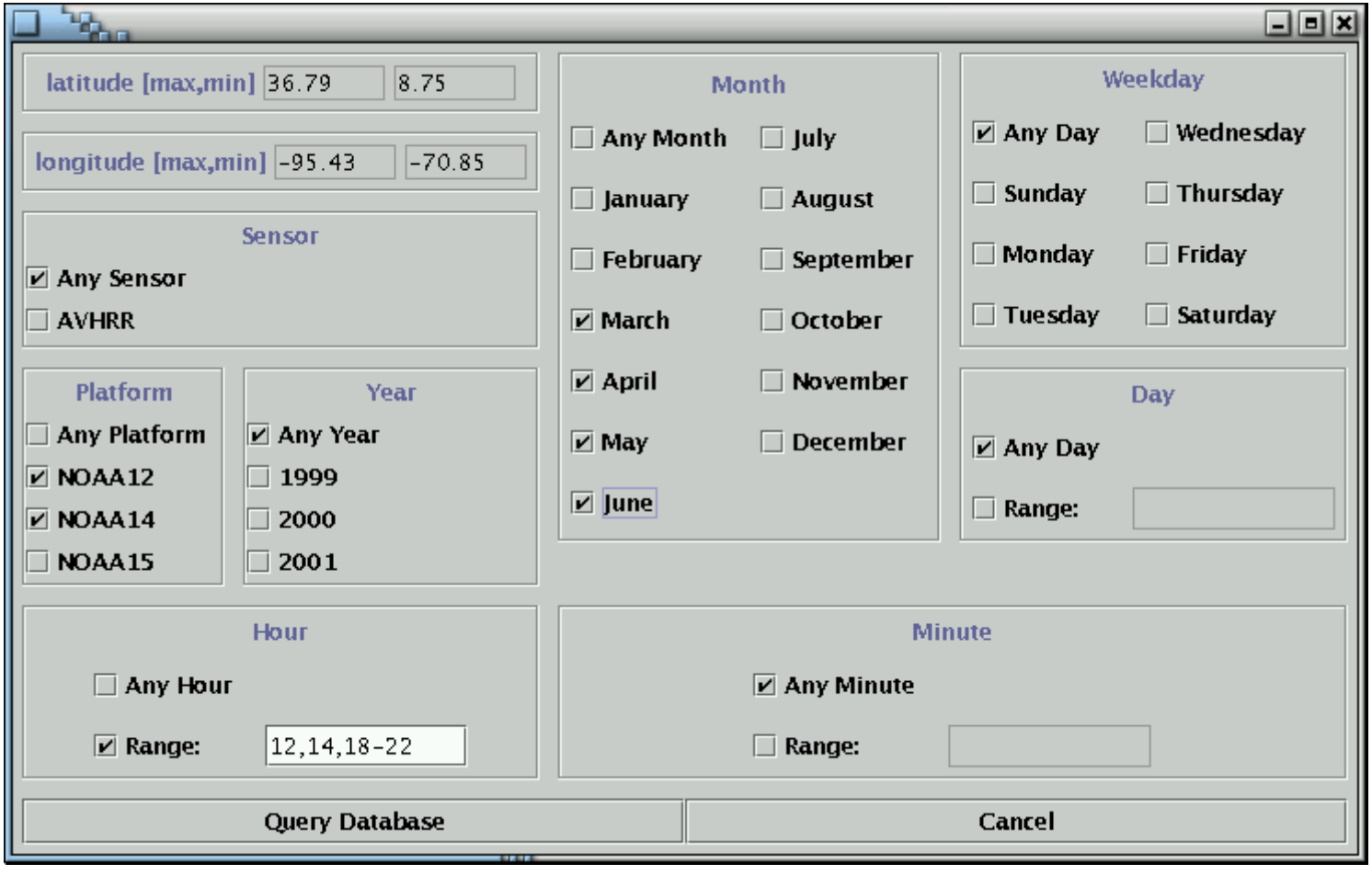
Graphical User Interface

PARL
Parallel Architecture
Research Laboratory



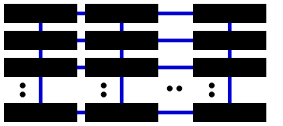
- Editor allows creation of the dataflow graph
- Results can be visualized in real-time
- Tools for introspection
- Performance analysis


File Edit Display
Awaiting Connection...
Dimensions: [858x1072]
400 Segments Received
25.0 / 25 [100.00%] Datasets Received


latitude [max,min]
36.79
8.75
longitude [max,min]
-95.43
-70.85
Sensor
Any Sensor
AVHRR
Platform
Any Platform
NOAA12
NOAA14
NOAA15
Year
Any Year
1999
2000
2001
Month
Any Month
January
February
March
April
May
June
July
August
September
October
November
December
Weekday
Any Day
Sunday
Monday
Tuesday
Wednesday
Thursday
Friday
Saturday
Day
Any Day
Range:
Hour
Any Hour
Range:
12,14,18-22
Minute
Any Minute
Range:
Query Database
Cancel

Achieving Parallelism

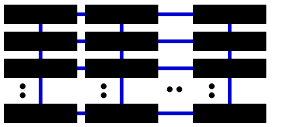
PARL
Parallel Architecture
Research Laboratory



- Distributor modules partitions satellite datasets
- Many drop-in parallel modules available
- Results reassembled by combining modules
- Module designers can use MPI for parallel computation
- Threads, queues, and other internal components increase performance (see paper)

Applications of CERSe

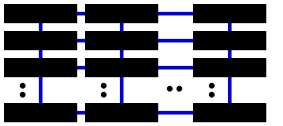
PARL
Parallel Architecture
Research Laboratory



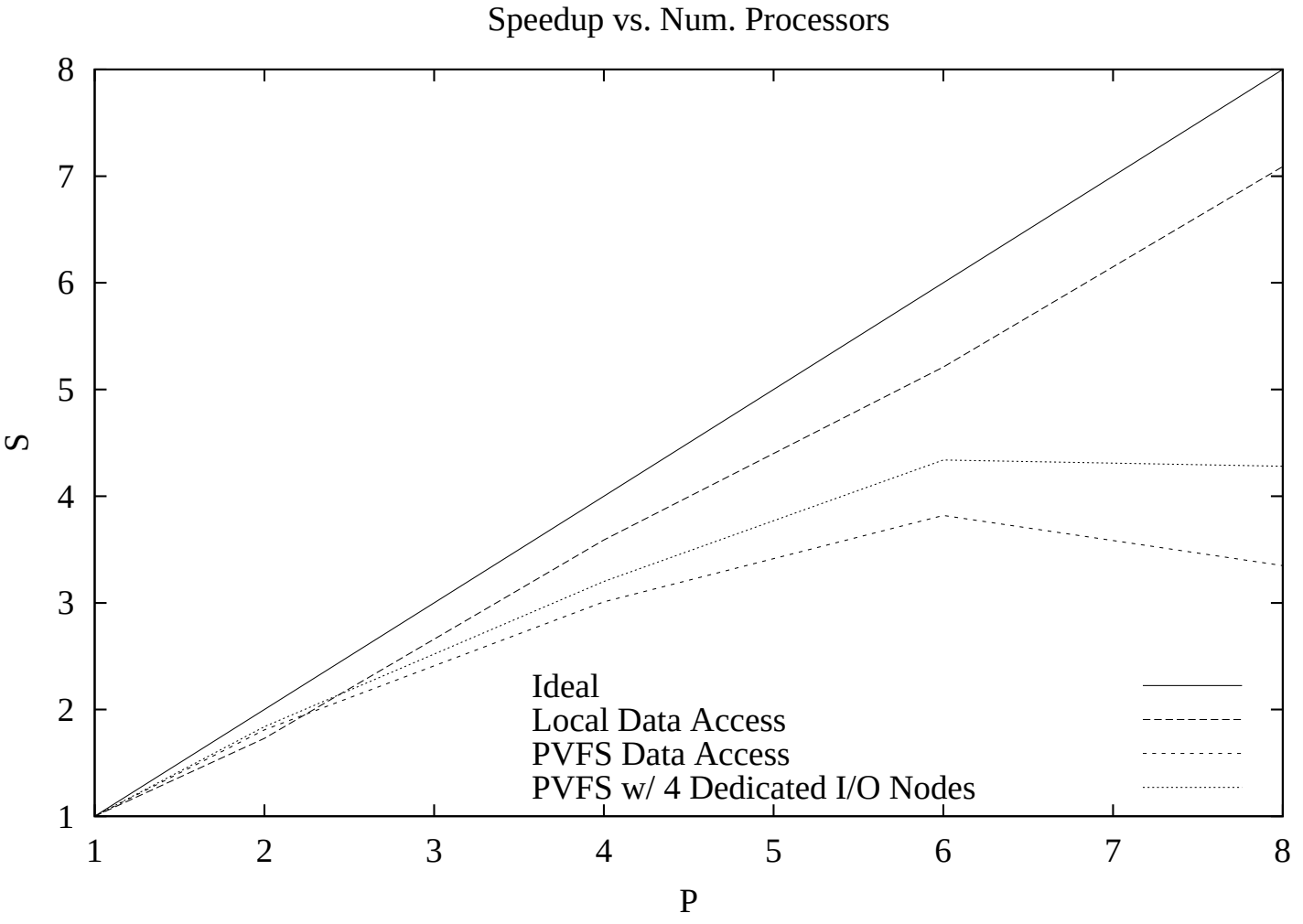
- Existing NASA code has been placed into CERSe
- Very little augmentation of original code required to work with CERSe
- Examples:
  - AVHRR
  - MODIS

Additional Technologies

PARL
Parallel Architecture
Research Laboratory

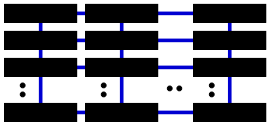


- CERSe is built upon Coven
- Coven is a framework for developing PSEs for parallel computers
- Coven provides a runtime driver and GUI can be extended for specific PSEs
- Coven is built upon The Algorithm Description Format (ADF)
- CERSe customizes Coven framework to the RS domain by changing the terminology presented in the interface


Speedup vs. Num. Processors
S
P
1
2
3
4
5
6
7
8
Ideal
Local Data Access
PVFS Data Access
PVFS w/ 4 Dedicated I/O Nodes

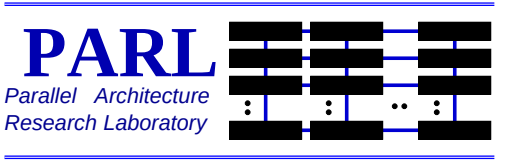
Conclusions

PARL
Parallel Architecture
Research Laboratory



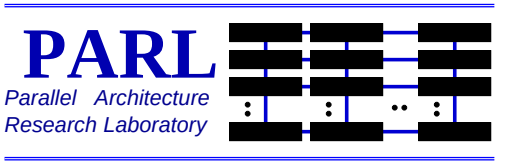
- The RS community faces a number of challenges which CERSe addresses
- Good scalability is achieved and explained in paper
- The GUI assists in usability and extensibility
- Advanced problems still require someone who understands parallel computers

Future Work



- Advanced profiling of jobs
- Advanced use of parallelism between components
- Applications in other problem domains

Acknowledgements



- This work was supported in part by:
  - NASA grant NAG5-8605 and by the
  - ERC Program of the National Science Foundation under Award Number EEC-9731680