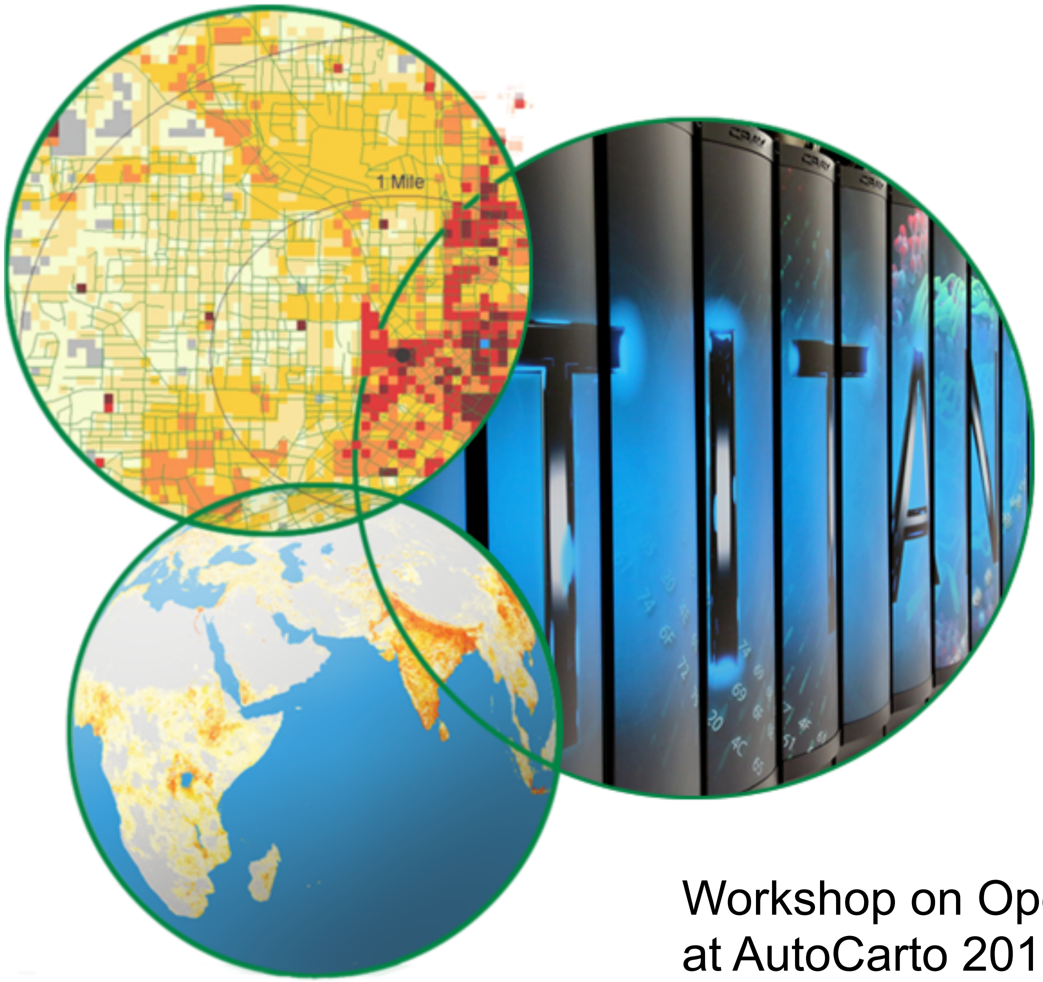
Open-Source Software in the Cutting Edge Geocomputaion Research at Oak Ridge National Laboratory



Alex Sorokine
SorokinA@ornl.gov

Workshop on Open Source Geospatial Technologies
at AutoCarto 2016, Albuquerque, New Mexico, USA
14 September 2016

Outline

- **Overview**
  - Oak Ridge National Laboratory
  - Geographic Information Science and Technology Group
  - Oak Ridge Leadership Computing
- **Projects**
  - pd-GRASS
  - From Everywhere To Everywhere (FETE)
  - Settlement Mapper
  - Web Applications
- **Summary and Conclusions**

ORNL is DOE's largest science and energy laboratory

- $1.5B budget
- 4,500 employees
- $750M modernization investment
- Nation's largest materials research portfolio
- 3,200 research guests annually
- Nation's most diverse energy portfolio
- Forefront scientific computing facilities
- World's most intense neutron source
- World-class research reactor
- 1,888 journal articles published in CY14
- Managing billion-dollar US ITER project
- 198 invention disclosures in FY15
- 66 patents issued in FY15

GIScience and Technology Group

Geographic

GIS&T Application focus areas

Geographic Information Science and Technology

Population and Critical Infrastructure Modeling

- Population distribution in space and time
- Socioeconomic characterization
- Event specific population response

Energy Assurance

- Energy and water resources assurance and reliability
- Modeling and visualizing the "Smart Grid"
- Bioenergy and renewable energy integration

Transportation M&S

- Multimodal route optimization
- Intelligent evacuation planning

Emergency Preparedness and Response

- Time critical decision support
- Disaster risk analysis

Climate Change Science

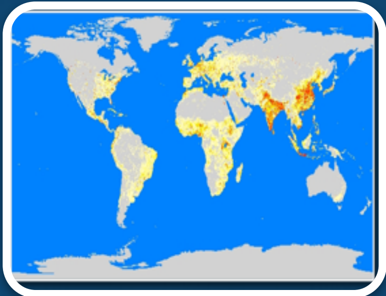
- Climate extremes and infrastructures
- Climate change and population response
- Energy and national security

Earth Science Informatics

- Real time data integration
- Data and knowledge management

Population Distribution and Dynamics

Improving Knowledge of Population Distribution and Dynamics



LandScan Global

- Spatial resolution of 30 arc seconds (~1km)
- Ambient population (average of 24 hours)
- Remote sensing based global data modeling and mapping

LandScan USA

- Spatial resolution of 3 arc seconds (~90m) coverage for the United States
- Nighttime and daytime population
- Integration of infrastructure and activity databases

LandScan HD (ongoing development)

- Spatial resolution of 3 arc seconds (~90m) global coverage
- Ambient population (average of 24 hours); Nighttime and daytime population where adequate land use data is available
- Settlement mapping from very high resolution imagery (1m or less)
- Integration of population density and activity databases

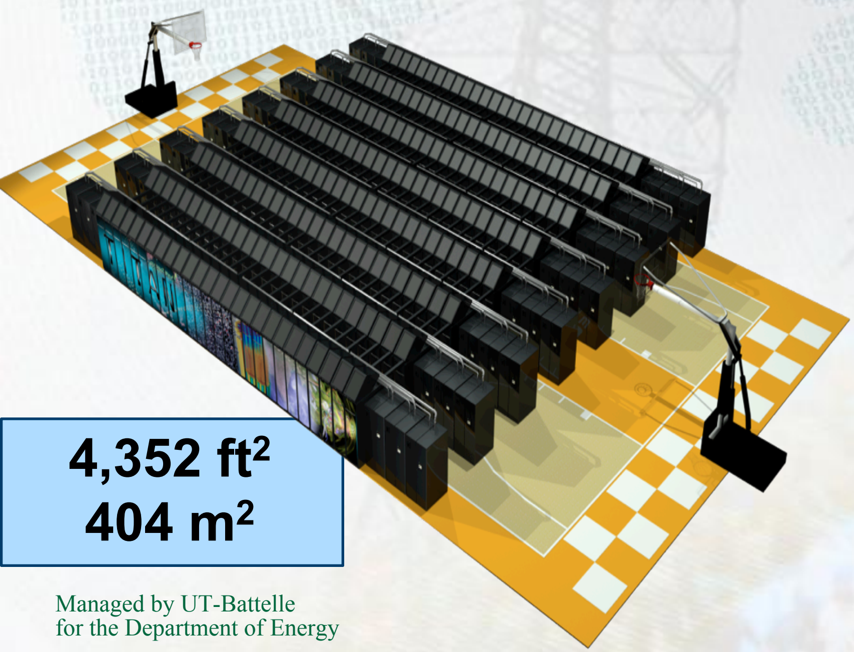
What is the Leadership Computing Facility (LCF)?

- Collaborative DOE **Office of Science user-facility program at ORNL and ANL**
- Mission: Provide the computational and data resources required to solve the most **challenging** problems.
- **2-centers/2-architectures to address diverse and growing computational needs of the scientific community**
- Highly competitive user allocation programs (INCITE, ALCC).
- Projects receive 10x to 100x more resource than at other generally available centers.
- LCF centers partner with users to enable science & engineering breakthroughs (Liaisons, Catalysts).

OAK RIDGE National Laboratory

Argonne NATIONAL LABORATORY

ORNL's "Titan" Hybrid System: Cray XK7 with AMD Opteron and NVIDIA Tesla processors



4,352 ft^2
404 m^2

SYSTEM SPECIFICATIONS:

- Peak performance of 27.1 PF (24.5 & 2.6)
- 18,688 Compute Nodes each with:
- 16-Core AMD Opteron CPU (32 GB)
- NVIDIA Tesla "K20x" GPU (6 GB)
- 512 Service and I/O nodes
- 200 Cabinets
- 710 TB total system memory
- Cray Gemini 3D Torus Interconnect

SUMMIT: coming in 2018

TITAN VS SUMMIT

Compute System Comparison

ATTRIBUTE	TITAN	SUMMIT
Compute Nodes	18,688	~3,400
Processor	(1) 16-core AMD Opteron per node	(Multiple) IBM POWER 9s per node
Accelerator	(1) NVIDIA Kepler K20x per node	(Multiple) NVIDIA Volta GPUs per node
Memory per node	32GB (DDR3)	>512GB (HBM+DDR4)
CPU-GPU Interconnect	PCI Gen2	NVLINK (5-12x PCIe3)
System Interconnect	Gemini	Dual Rail EDR-IB (23 GB/s)
Peak Power Consumption	9 MW	10 MW

pd-GRASS

PARALLEL DISPLAY FOR GRASS GIS

Parallel Visualization with GRASS

- **EVEREST -- exploratory scientific visualization facility**
  - Open in 2004
  - Upgraded in 2014
- **pd-GRASS**
  - developed 2006
  - Why OSS?

ELSEVIER

Computers & Geosciences 33 (2007) 685–695

COMPUTERS & GEOSCIENCES

www.elsevier.com/locate/cageo

Implementation of a parallel high-performance visualization technique in GRASS GIS

Alexandre Sorokine*,[1]

Geographic Information Science & Technology Group, Oak Ridge National Laboratory, P.O. Box 2008 MS–6017, Oak Ridge, TN 37831-6017, USA

Received 29 March 2006; received in revised form 24 July 2006; accepted 18 September 2006

Abstract

This paper describes an extension for GRASS geographic information systems (GIS) that enables users to perform geographic visualization tasks on tiled high-resolution displays powered by the clusters of commodity personal computers. Parallel visualization systems are becoming more common in scientific computing due to the decreasing hardware costs and availability of the open source software to support such architecture. High-resolution displays allow scientists to visualize very large data sets with minimal loss of details. Such systems have a big promise especially in the field of GIS because users can naturally combine several geographic scales on a single display.

This paper discusses architecture, implementation, and operation of PD-GRASS—a GRASS GIS extension for high-

Visualization System

- **EVEREST**
  - Commodity visualization cluster
  - 27 DLP
  - 11,520 x 3,072 pixels
  - 14 visualization nodes + 1 headnode
  - 30 x 8 ft (9.14 x 2.43 m)

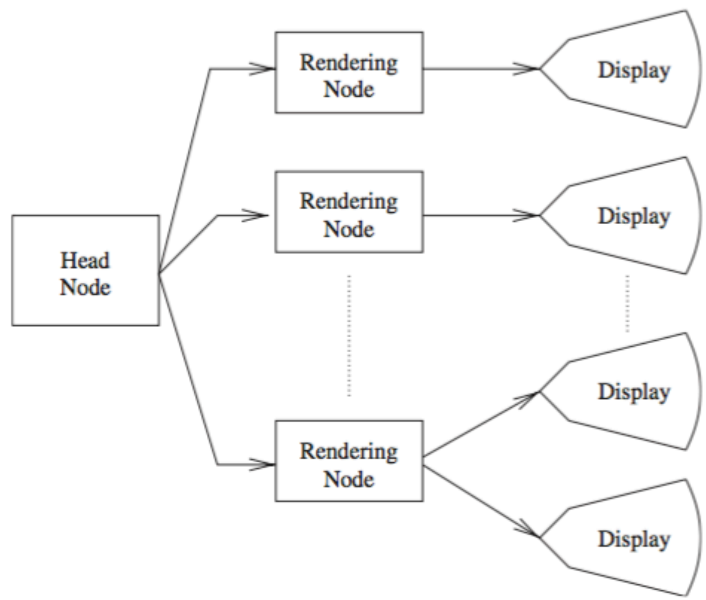


Fig. 1. Visualization cluster architecture.

- **Visulization nodes**
  - Commodity PCs with Linux
- **Comparison to conventional systems**
  - Desktop monitor 1-2 megapixel
  - IMAX theater ~80 megapixel (less in reality)
  - 4K monitor ~9 megapixel

Goals and Challenges

Goals

- Generate 30 megapixel image from GIS
- Goals
  - Visualization of large datasets
  - Group work
- Advantages of display walls for geovisualization
  - Ability to portray multiple scale at once
  - Inherit parallelism for faster rendering

Technical Challenges

- VNC, Xdmx, rdp, Chromium (OpenGL renderer)
  - Inability to generate large enough image
  - Bandwidth limitation
  - Disk I/O bottleneck
- ArcGIS
  - License
- Google Earth
  - Not able to download backdrop large enough

Why GRASS?

- No licensing fees
- Supports Linux
- Full command-line control
- Flexible installation options
  - Customized version of Red Hat Linux
  - No root access
- Customizable compilation process
  - Ability to disable parts that you are not using

Design and Implementation

- **Design**
  - a copy of GRASS runs on each node
  - Control from the head node via command line
  - Copy of data on each node (avoids files system limits)
- **Implementation**
  - Customized installation script
    - Unnecessary dependencies removed
    - Synchronized installation on multiple nodes
  - Modified GRASS shell to be launched via ssh
  - Shell scripts that mimic standard GRASS com
    - d.* -- display commands
    - Execute GRASS commands in parallel on multiple nodes
    - Synchronize database
    - cleanup

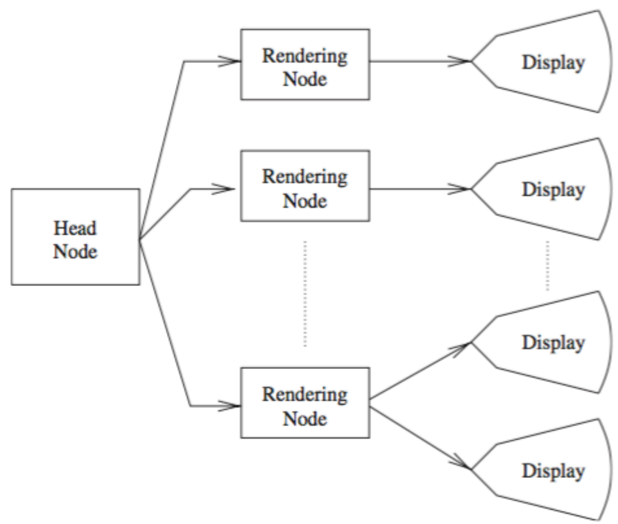


Fig. 1. Visualization cluster architecture.

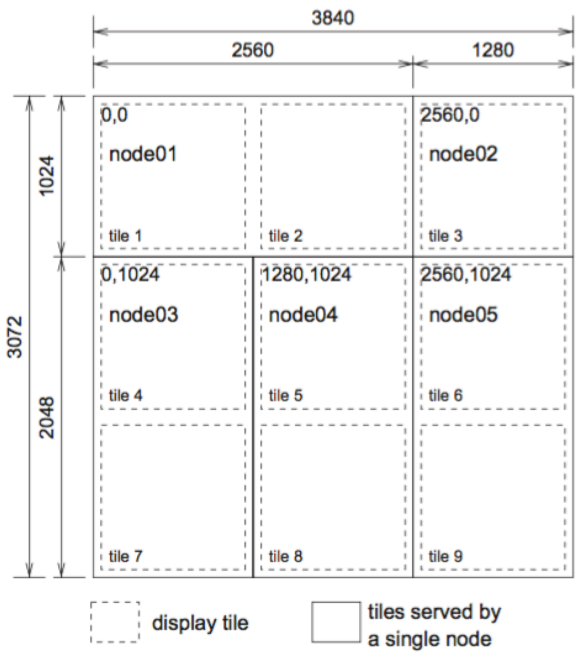
Operation

Configuration

a

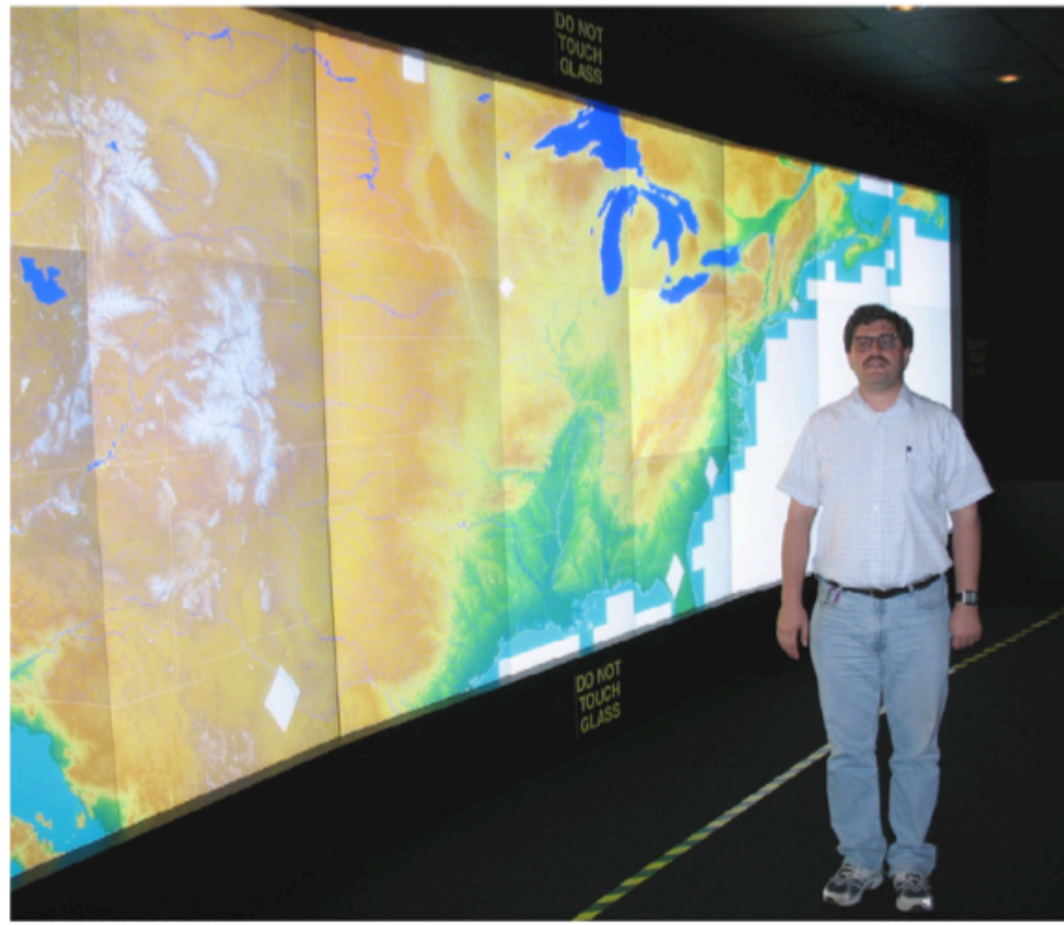
#node	x0	y0	width	height
node01	0	0	2560	1024
node02	2560	0	1280	1024
node03	0	1024	1280	2048
node04	1280	1024	1280	2048
node05	2560	1024	1280	2048

b



SRTM

```
pd.rast srtm3sec
pd.vect streams color = blue
pd.vect lakes fcolor = blue
pd.vect states color = white
```



Lessons of pd-GRASS

- **Advantages**
  - Deeply customizable for non-standard systems
    - Ability to fix the bugs
  - Install on any number of nodes you need
  - Developer community
- **Challenges**
  - Compilation for non-standard systems

FETE

FROM EVERYWHERE TO EVERYWHERE

FETE: From Everywhere To Everywhere

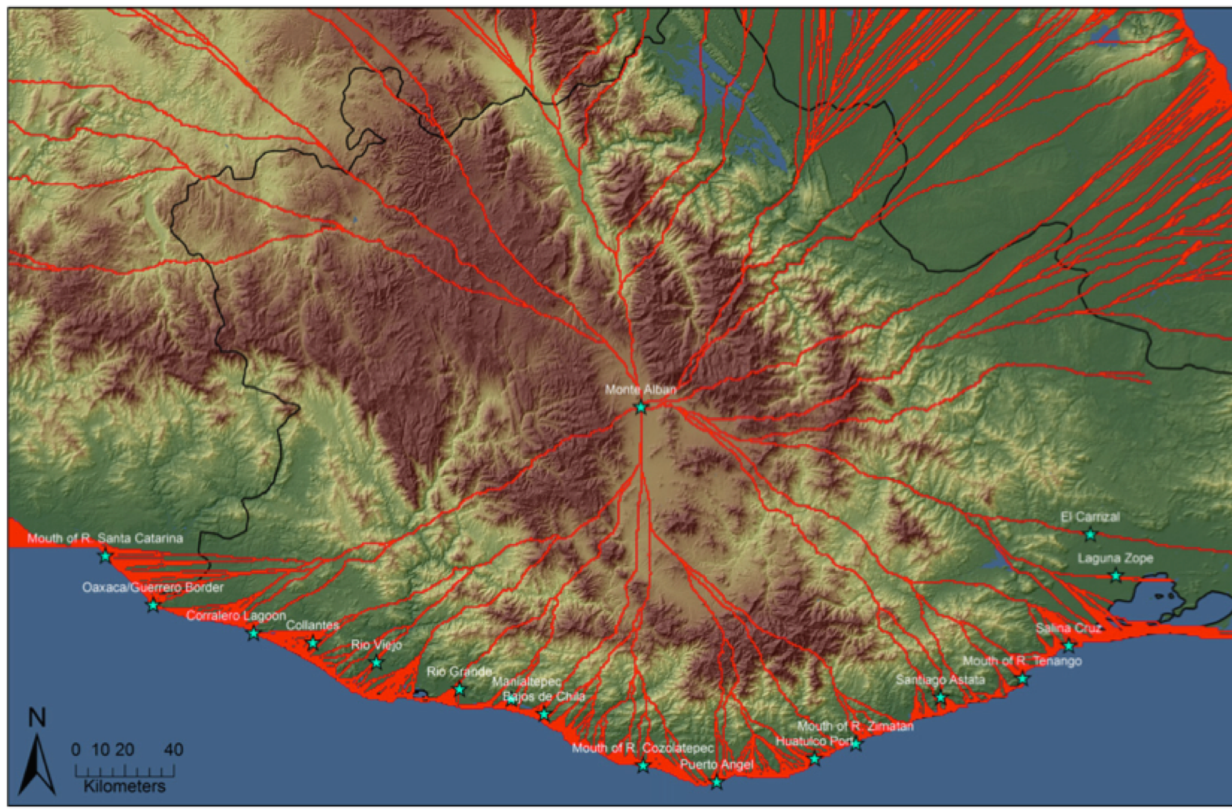
Geographic Information Science and Technology

Devin White   Andrew Hardin

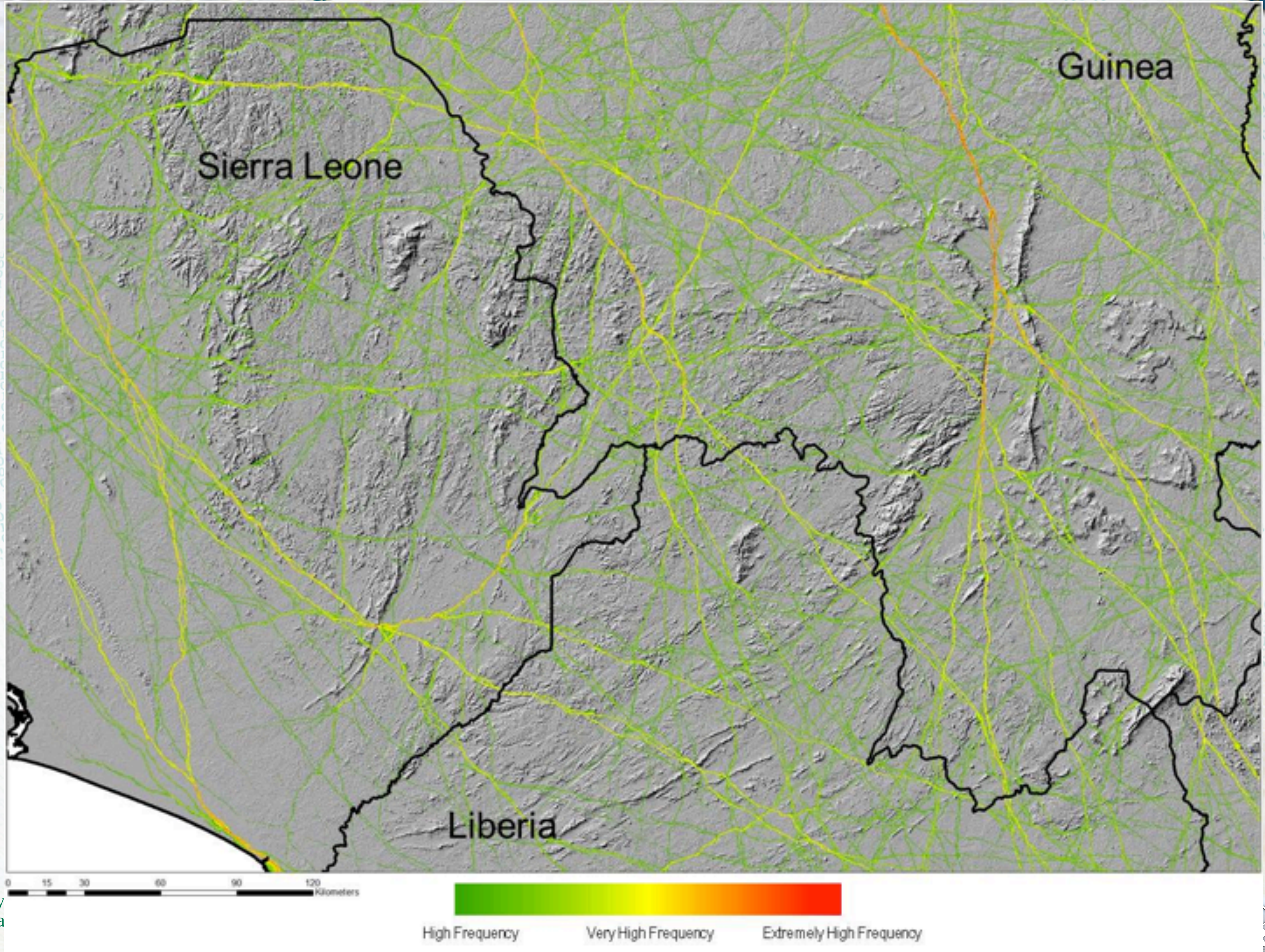
- Difficult to parallelize problem on a massively-parallel system

- White and Barbe
  pedestrian trans
  precolumbian O
  Archaeological

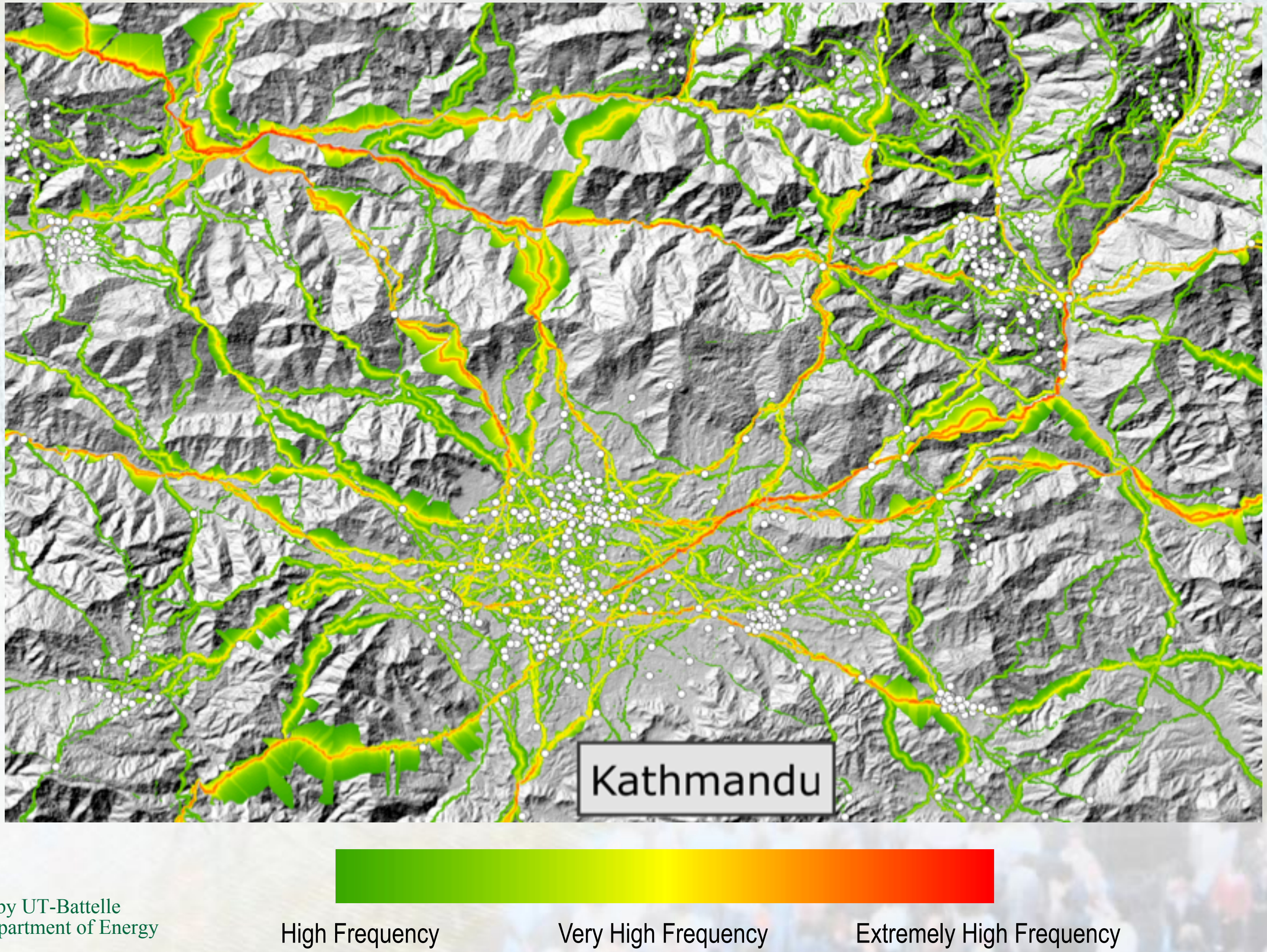
- Finding likely pl
  calculating netw
  land use



Pedestrian Network Example

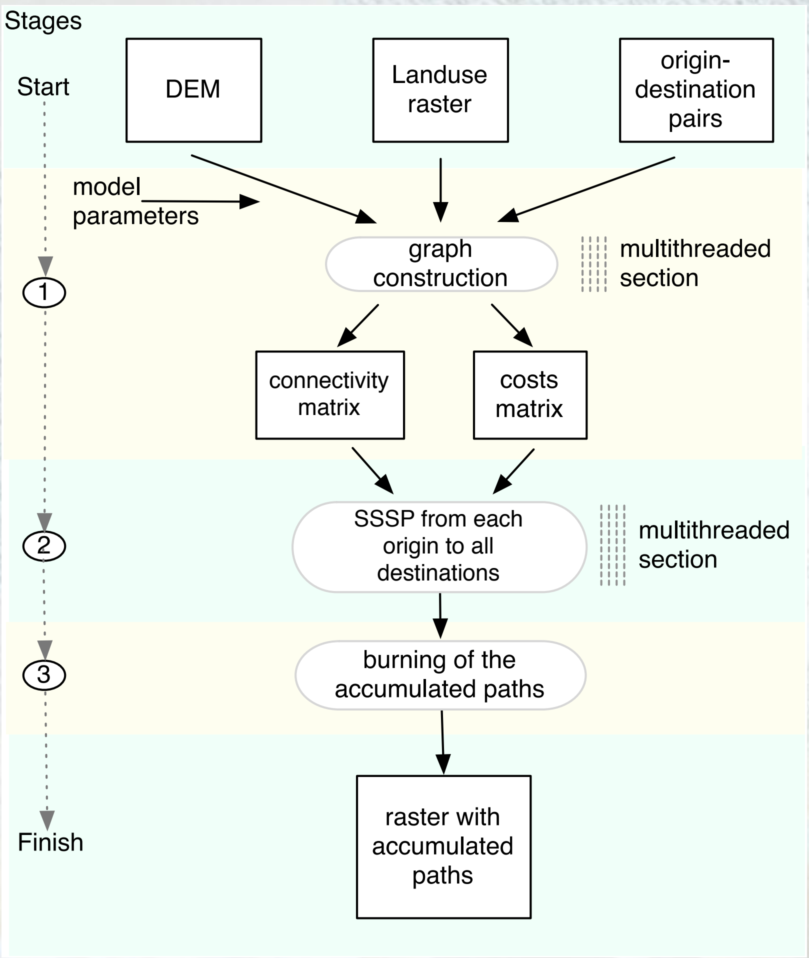


Kathmandu, Nepal



Algorithm

- DEM and land use rasters as input
- Create graph using 8 neighbors and costs based on Tobler equation
- Find shortest paths for large number of origins and destinations
- Calculate density of the routes (accumulated paths)

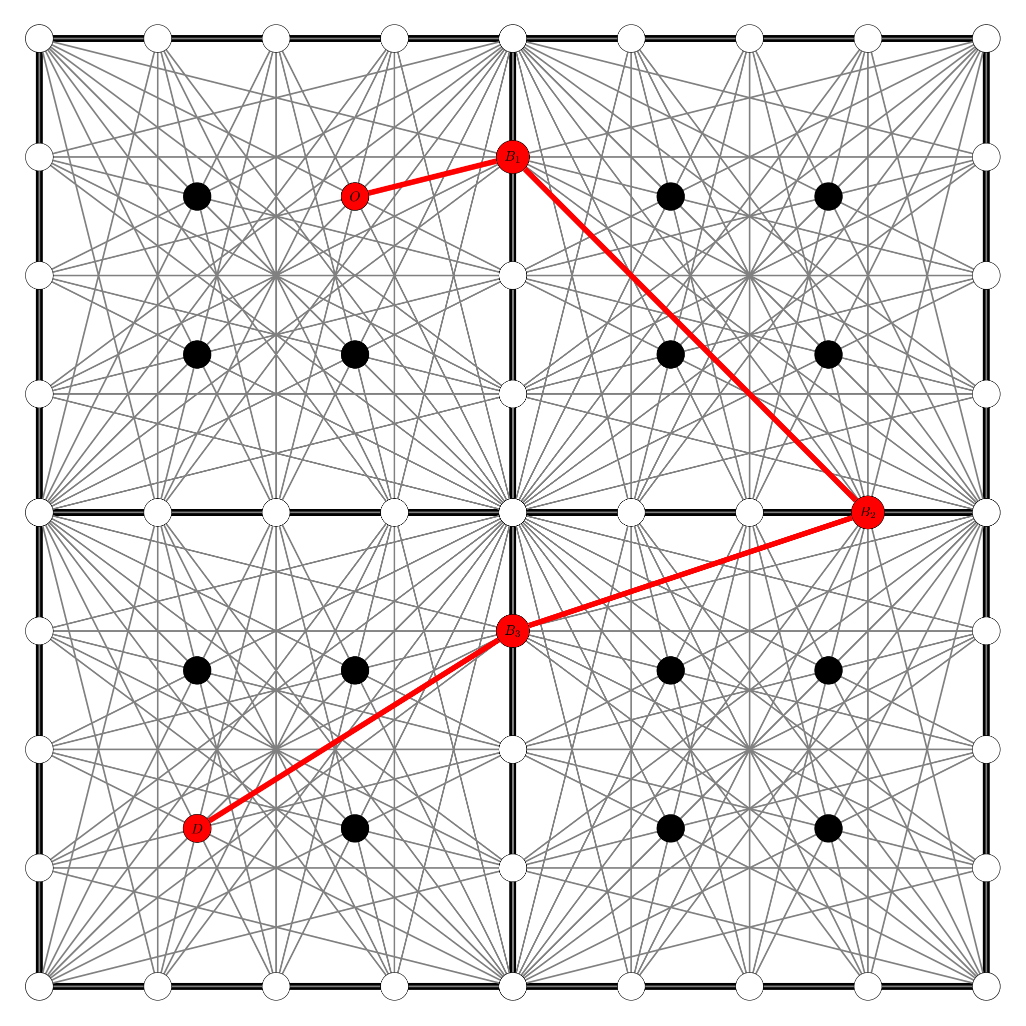


Approaches to Parallelization

- Distributed priority queue
  - requires very large number of small communications between the nodes
- Approximate methods
  - different line of research
- grid-SSSP [Arge et al. 2001]
  - originally developed for out-of-RAM computations
  - subgraph with complete sets of paths
  - applicable to geographic data

gridSSSP on a raster

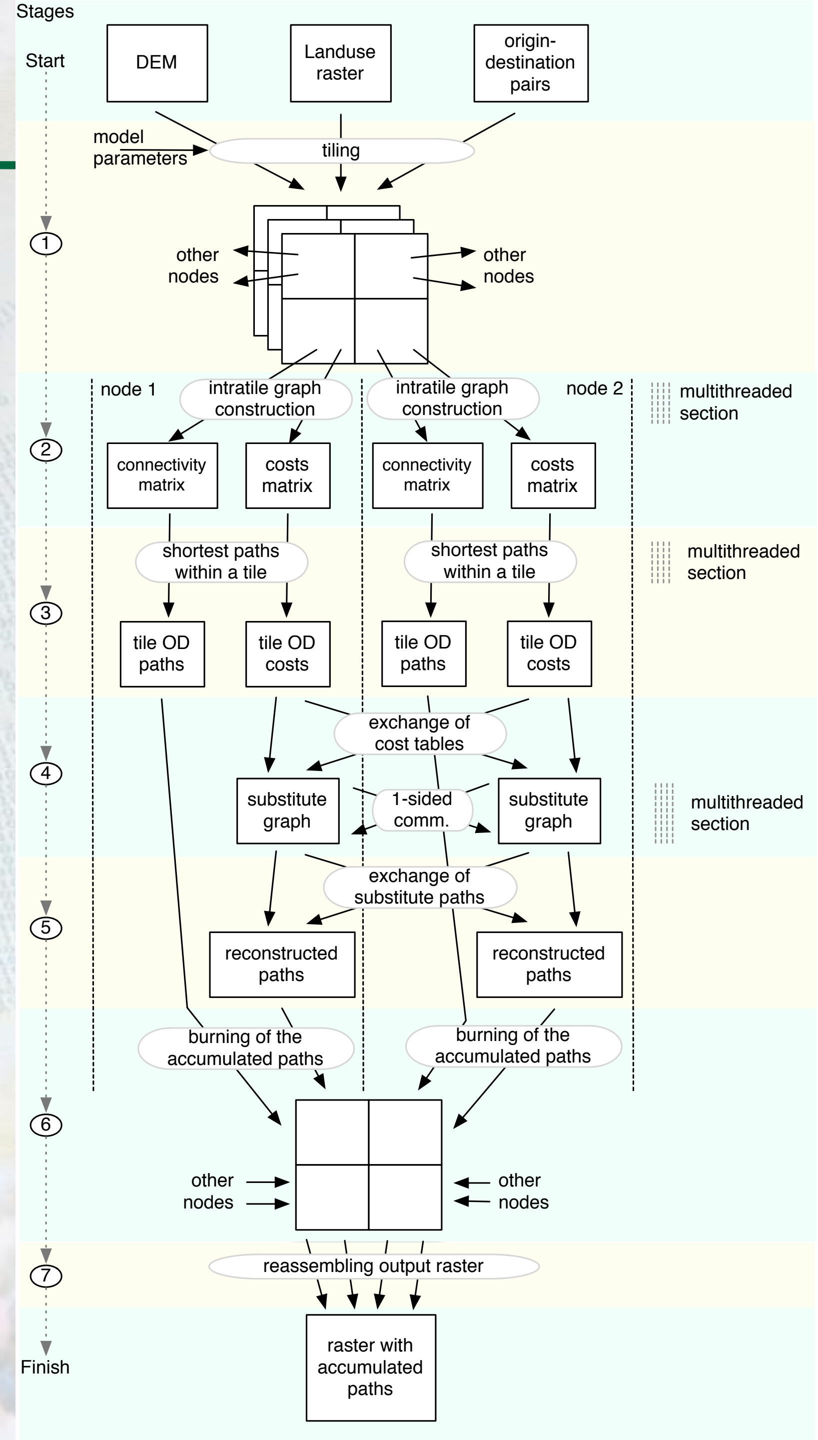
- Create tiles
- Precompute and store all paths within each tile for
  - edge nodes
  - origins and destinations
- Build substitute graph
  - use connectivity between origin-destinations and edge nodes
  - costs from previous step
- Find shortest paths in the substitute graph
- Reconstruct complete paths



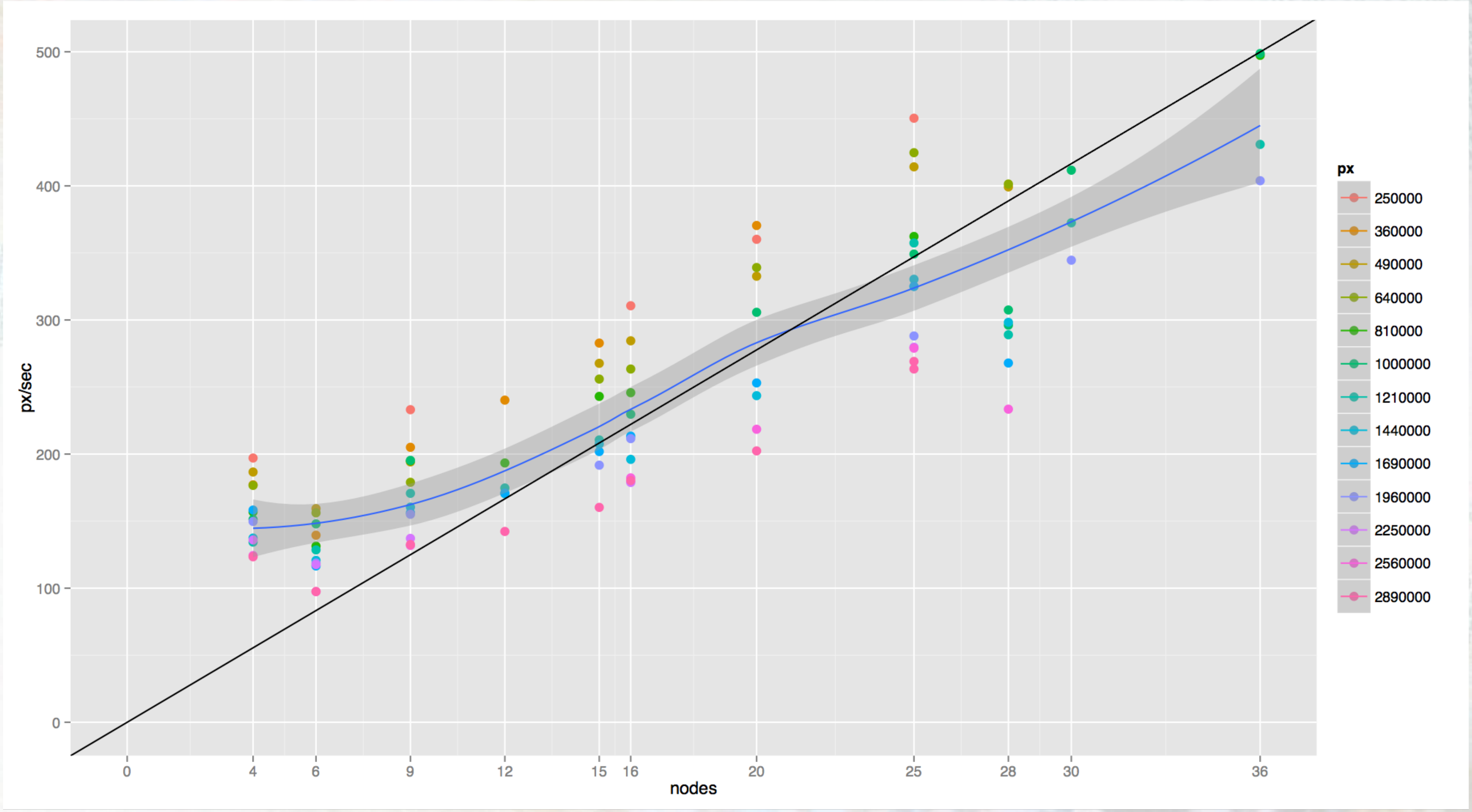
Parallel Algorithm

Geographic Information Science and Technology

- Create tiles, load data
- Build graphs inside tiles
- Find shortest paths in tiles
- Exchange path costs
- Build substitute graph and find shortest paths
- Reconstruct paths
- Create accumulated paths
- Reconstruct output raster

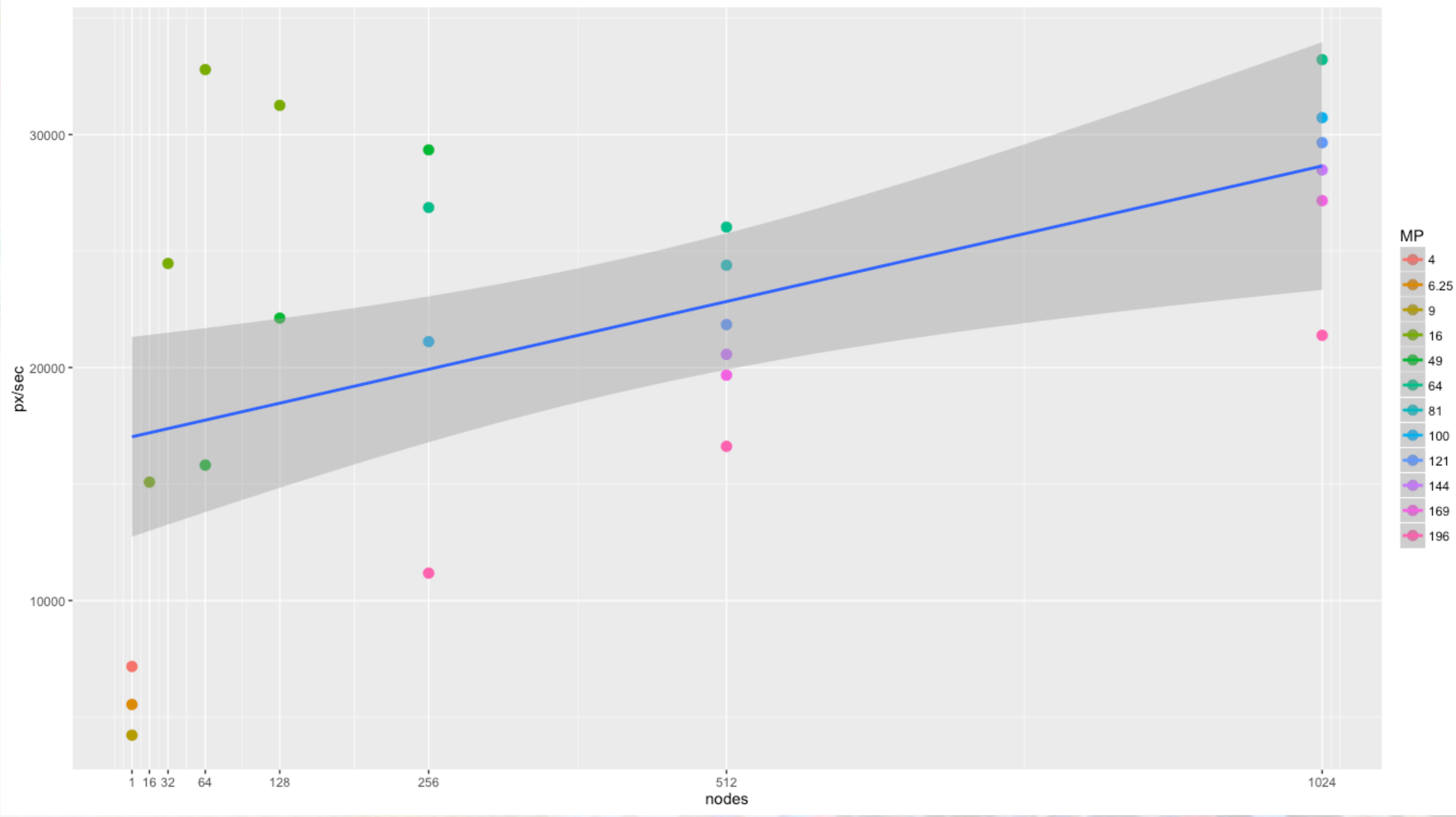


Speedup on Commodity Cluster



for ~6 million paths

Speedup on TITAN



up to ~4 billion paths

FETE Software

- **Compilers**
  - C++11 and C
  - GCC, Intel compiler, Portland Group
- **Libraries**
  - OpenMP
  - MPI: OpenMPI, mpich
- **Geospatial libraries: customized cross-compilation**
  - gdal
  - proj4

SETTLEMENT MAPPER

Settlement Mapper

Geographic Information Science and Technology

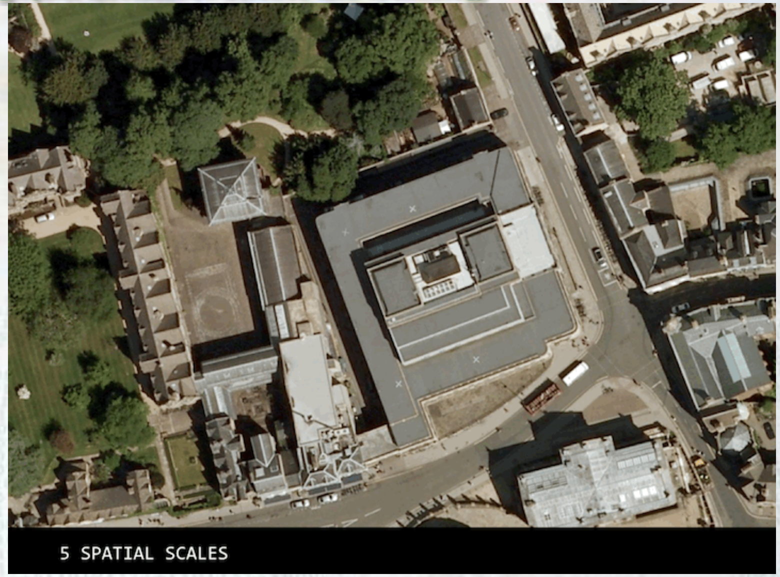
Dilip Patola

- **It's important to know where people live**
- **Monitoring from space**
  - Cost effective and efficient

  Supporting Emergency Response and Planning

Computer Vision Challenges

Dilip Patola

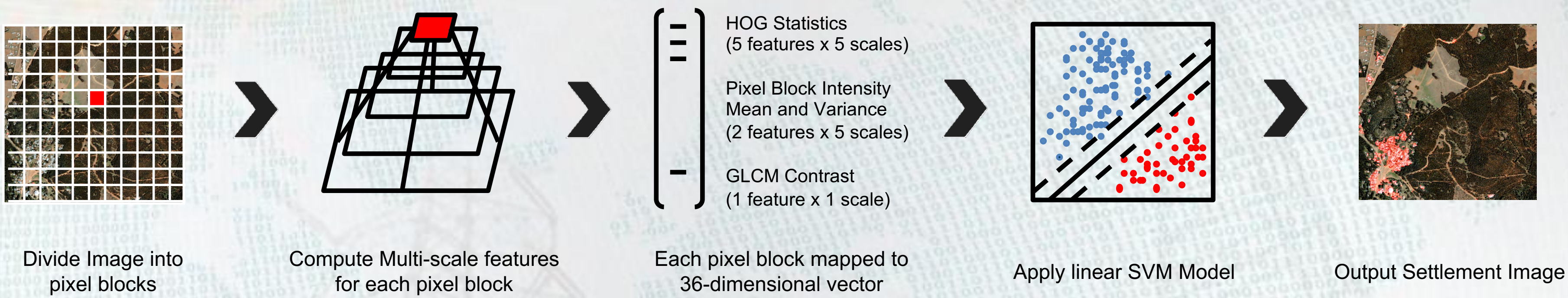


With high-resolution imagery, pixel features are not discriminatory.

Multi-scale features provide accurate settlement characterization

- We aggregate low level features from 5 spatial scales.
  - Histogram of Gradient Orientations(HOG) & Gray Gray-Level Co-Occurrence Matrix (GLCM)
  - Texture extraction (TEXTONS) using Leung Malik (LM) filter bank.
  - Normalized Difference Vegetation Index (NDVI)
  - Band Ratios(BANDRT)
  - Dense Scale Invariant Feature Transform (DSIFT)

Settlement Mapping Process



- Settlement mapping unit is pixel block
- We rely on structural patterns more than spectral components
- Pixel patterns examined at different scales
- A linear support vector machine (SVM) classifier is used to map the feature vector (z) to one of the binary classes (settlement and non-settlement).

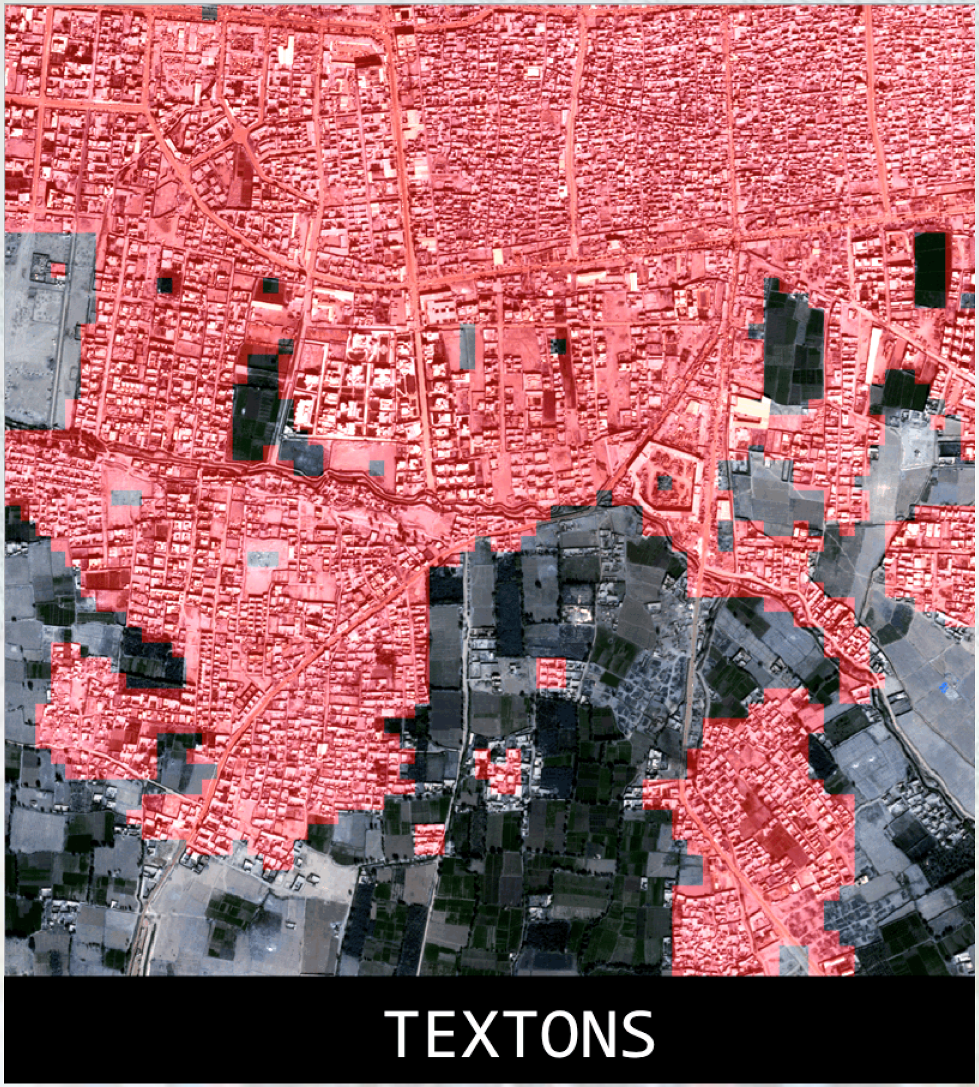
Performance

Geographic Information Science and Technology

33 images of 0.5 m spatial resolution, each covering an area of 2.6 km2, collected from various parts of Kandahar, Afghanistan

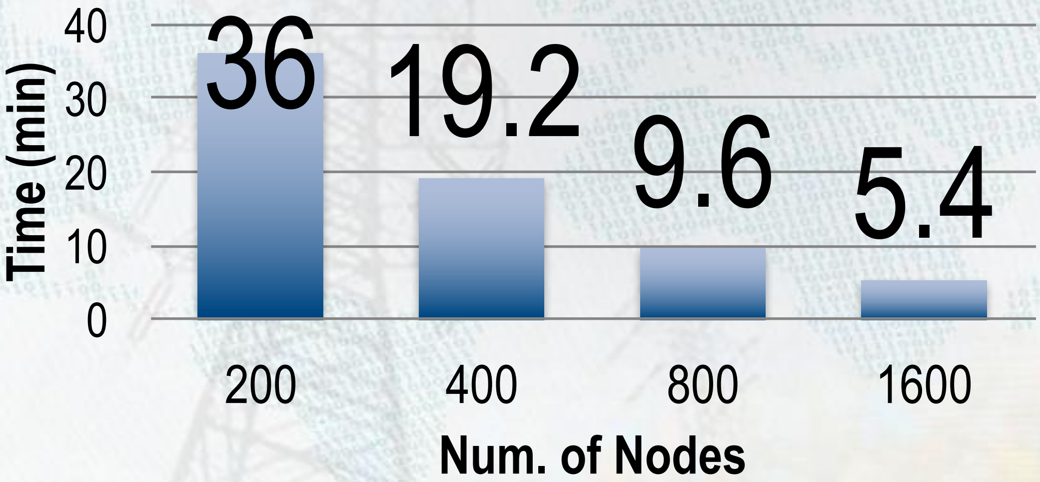
Processing times are based on a 4 C2075 GPU workstation

Feature	Accuracy(%)	Runtime(s)
HOG	93.5	1.6
TEXTONS	92.7	4.7
VEGIND	91.4	1.77
BANDRT	86.1	1.93
DSIFT	90.8	11.33



SMTool on TITAN SUPERCOMPUTER

- Processed California with HOG feature
- NAIP 1 m spatial resolution imagery
- 400 GB compressed = 4 TB uncompressed
- RGB Bands processed



From Hours to Seconds

Web Applications

Bioenergy KDF

Aaron Myers

Goal

- Facilitate Informed Decision Making
- Ability to synthesize, analyze, and visualize bioenergy research and data

Approach

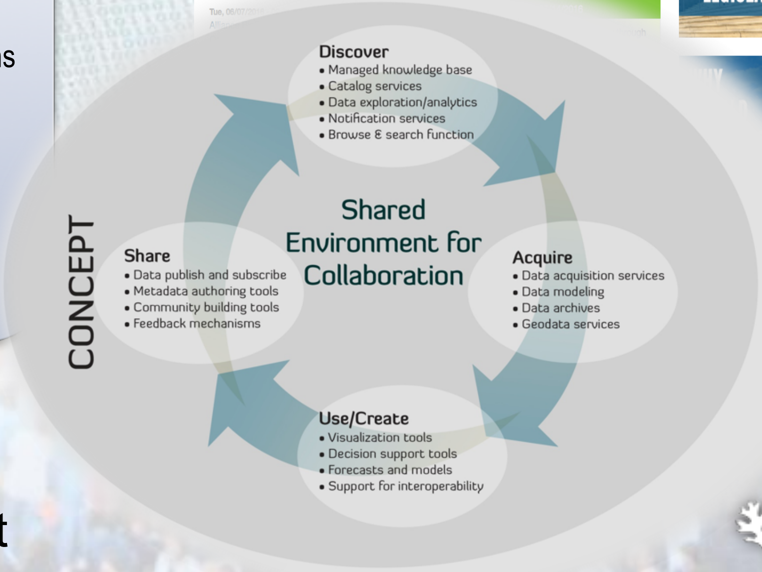
- Web-enabled Platform for Collaboration and Knowledge Exchange
- Built with Open-Source Tools and Software
- Bring researchers and information closer together

Capabilities

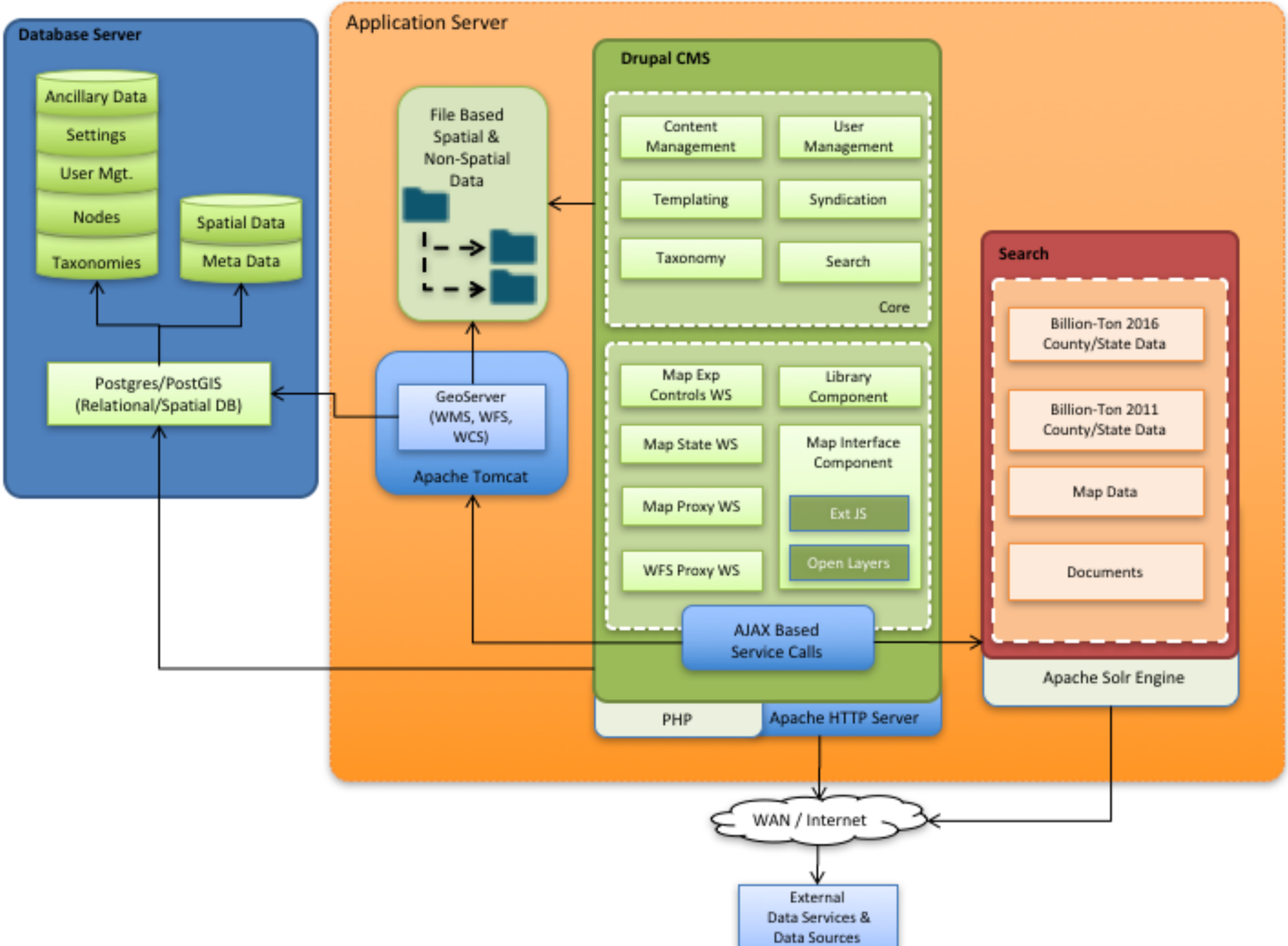
- Search
- Contribute
- Collaborate
- Visualize
- Analyze

Successes

- Billion-Ton Update Data Exchange and Visualizations
- Biomass Scenario Model Tool
- Over 2,000 Bioenergy Research Related Content



bioenergykdf.net

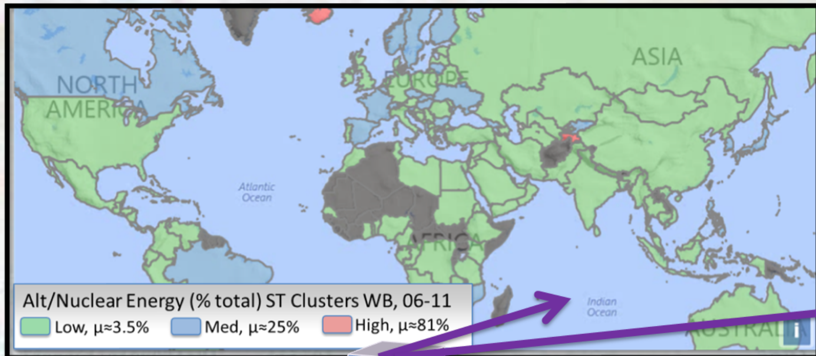


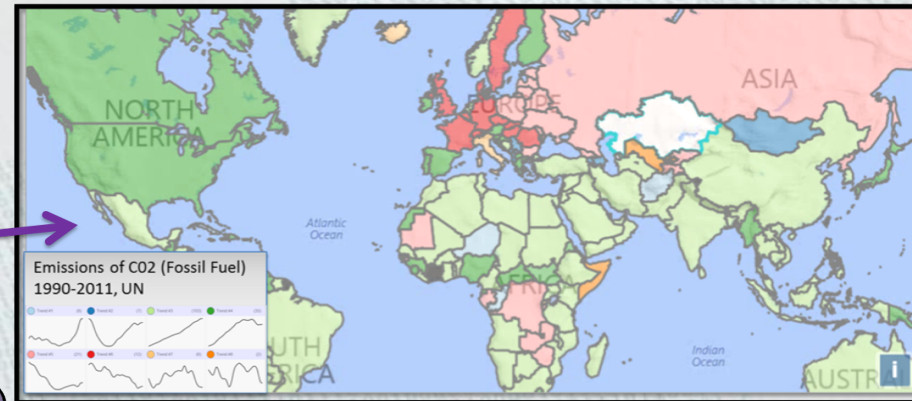
Database Server
Ancillary Data
Settings
User Mgt.
Nodes
Taxonomies
Spatial Data
Meta Data
Postgres/PostGIS
(Relational/Spatial DB)
Application Server
File Based
Spatial &
Non-Spatial
Data
GeoServer
(WMS, WFS,
WCS)
Apache Tomcat
Drupal CMS
Content
Management
User
Management
Templating
Syndication
Taxonomy
Search
Core
Map Exp
Controls WS
Library
Component
Map State WS
Map Interface
Component
Map Proxy WS
Ext JS
WFS Proxy WS
Open Layers
AJAX Based
Service Calls
PHP
Apache HTTP Server
Search
Billion-Ton 2016
County/State Data
Billion-Ton 2011
County/State Data
Map Data
Documents
Apache Solr Engine
WAN / Internet
External
Data Services &
Data Sources

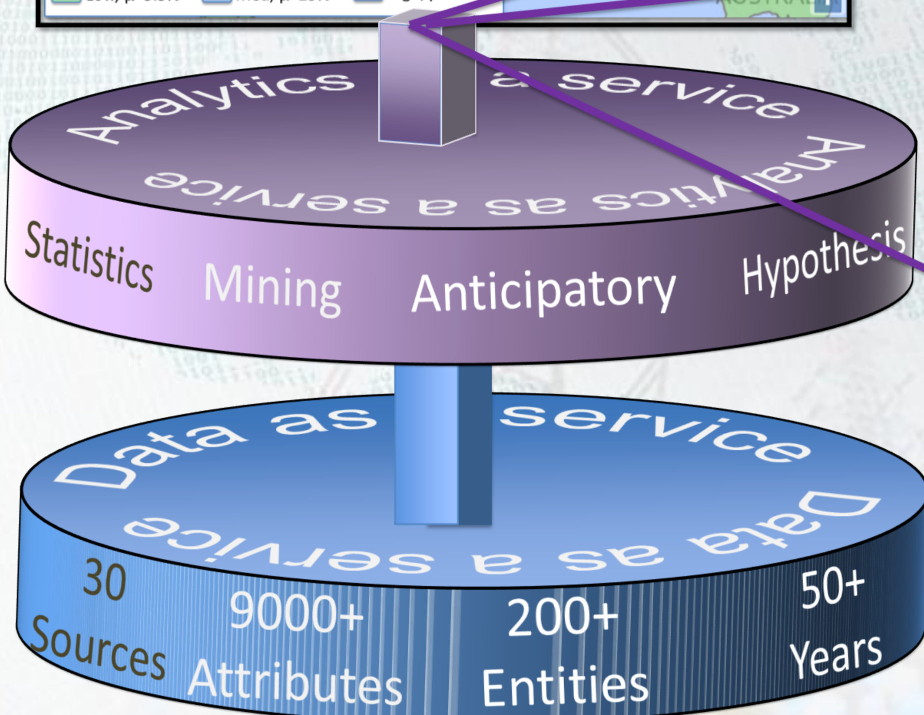
World SpatioTemporal Analytics and Mapping Project (World STAMP)

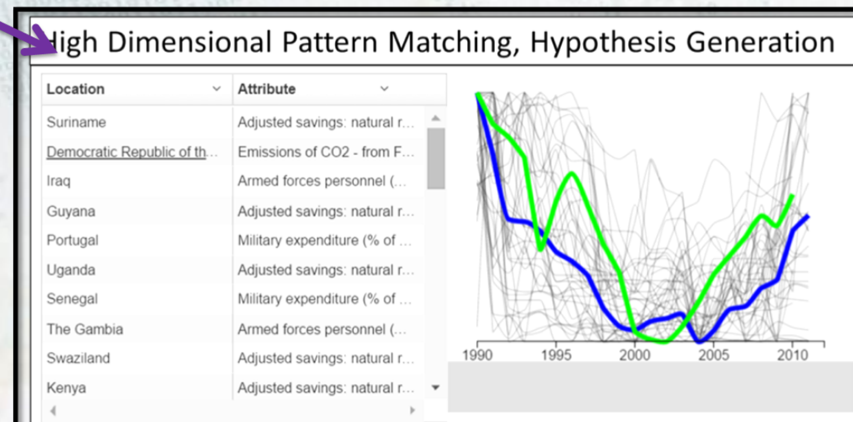
Geographic Information Science and Technology

Access: wstamp.ornl.gov
Contact: Robert Stewart, stewartrn@ornl.gov





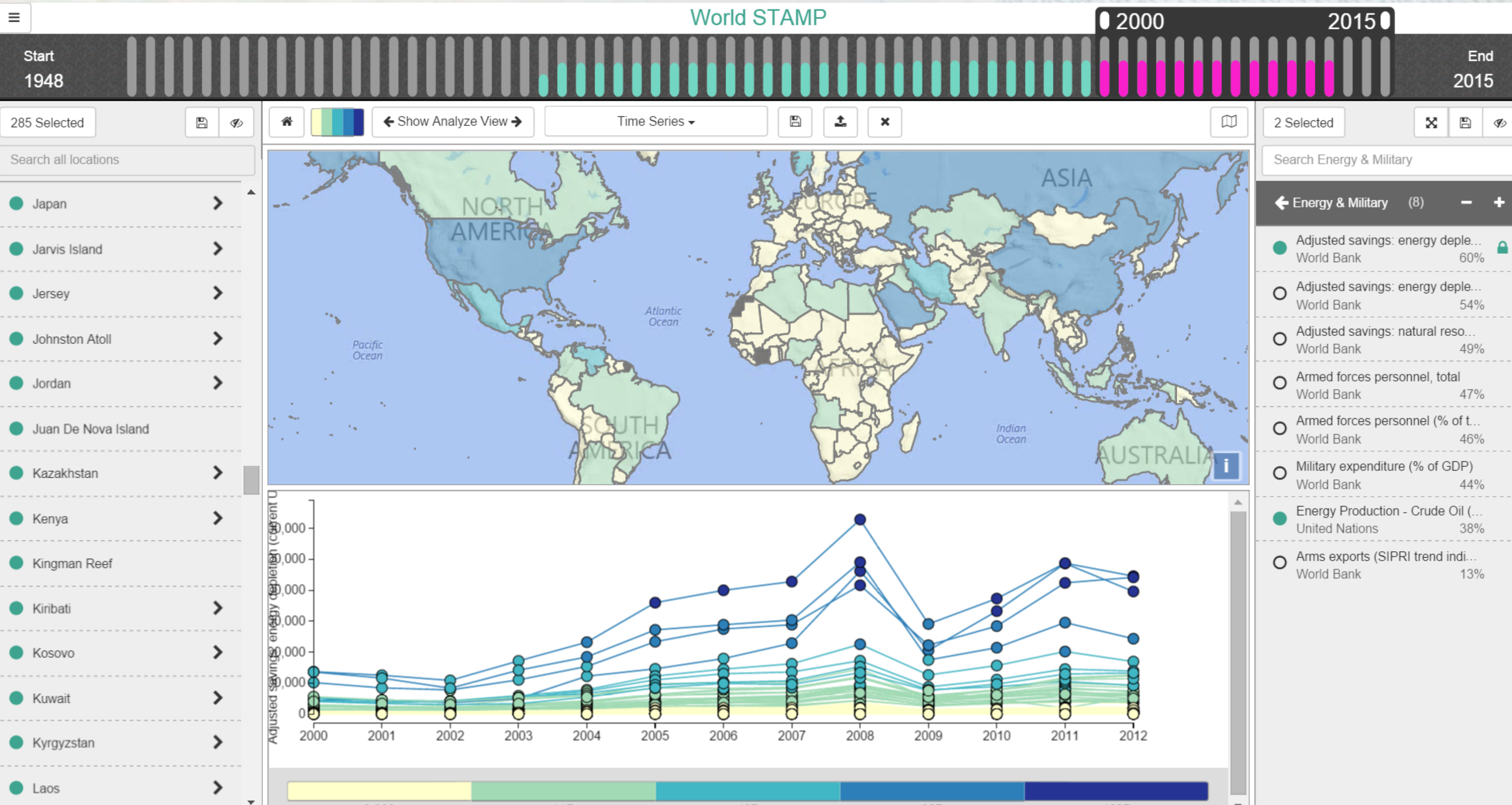


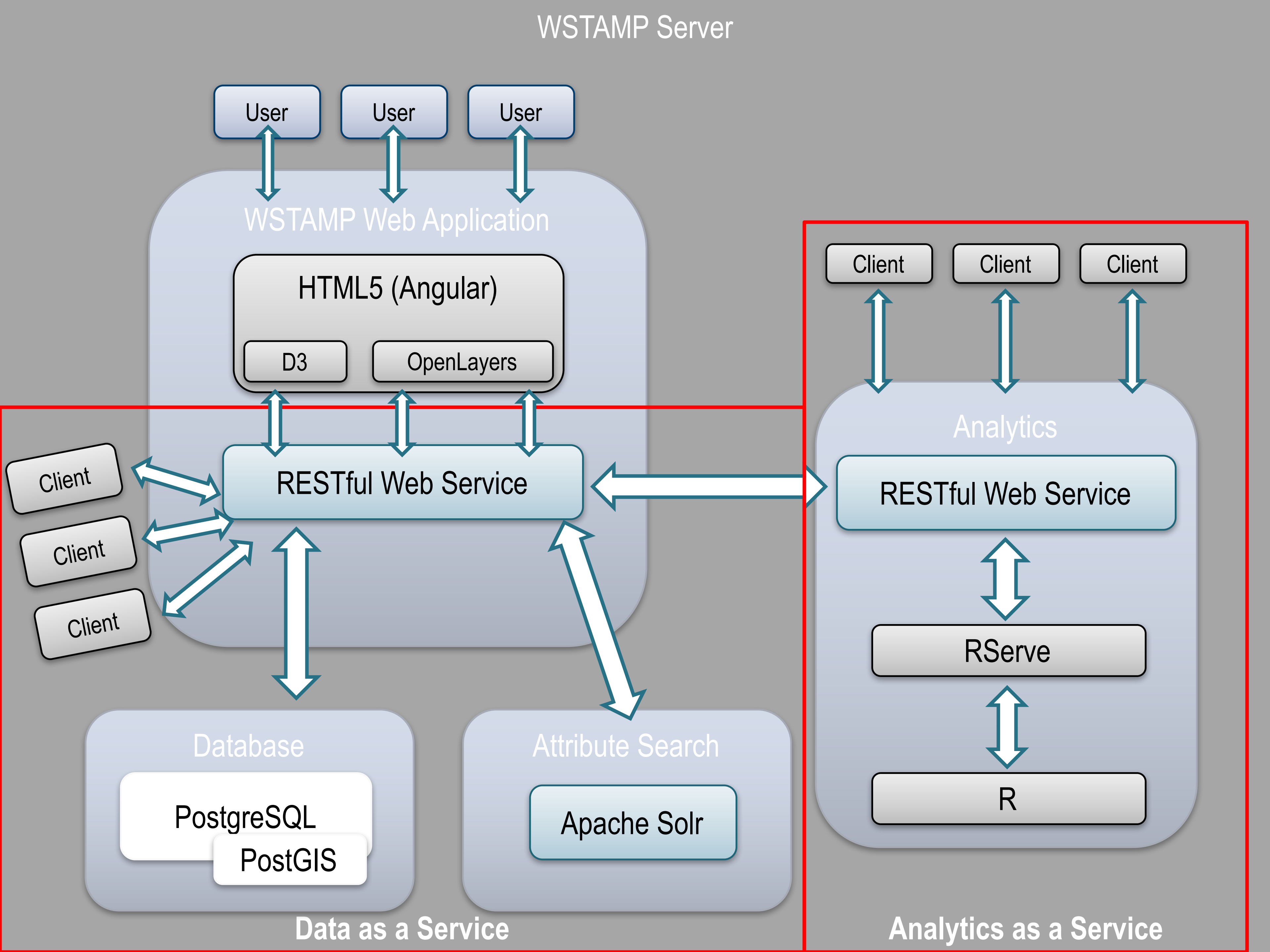


World SpatioTemporal Analytics and Mapping Project (World STAMP)

Geographic Information Science and Technology

Access: wstamp.ornl.gov
Contact: Robert Stewart, stewartrn@ornl.gov

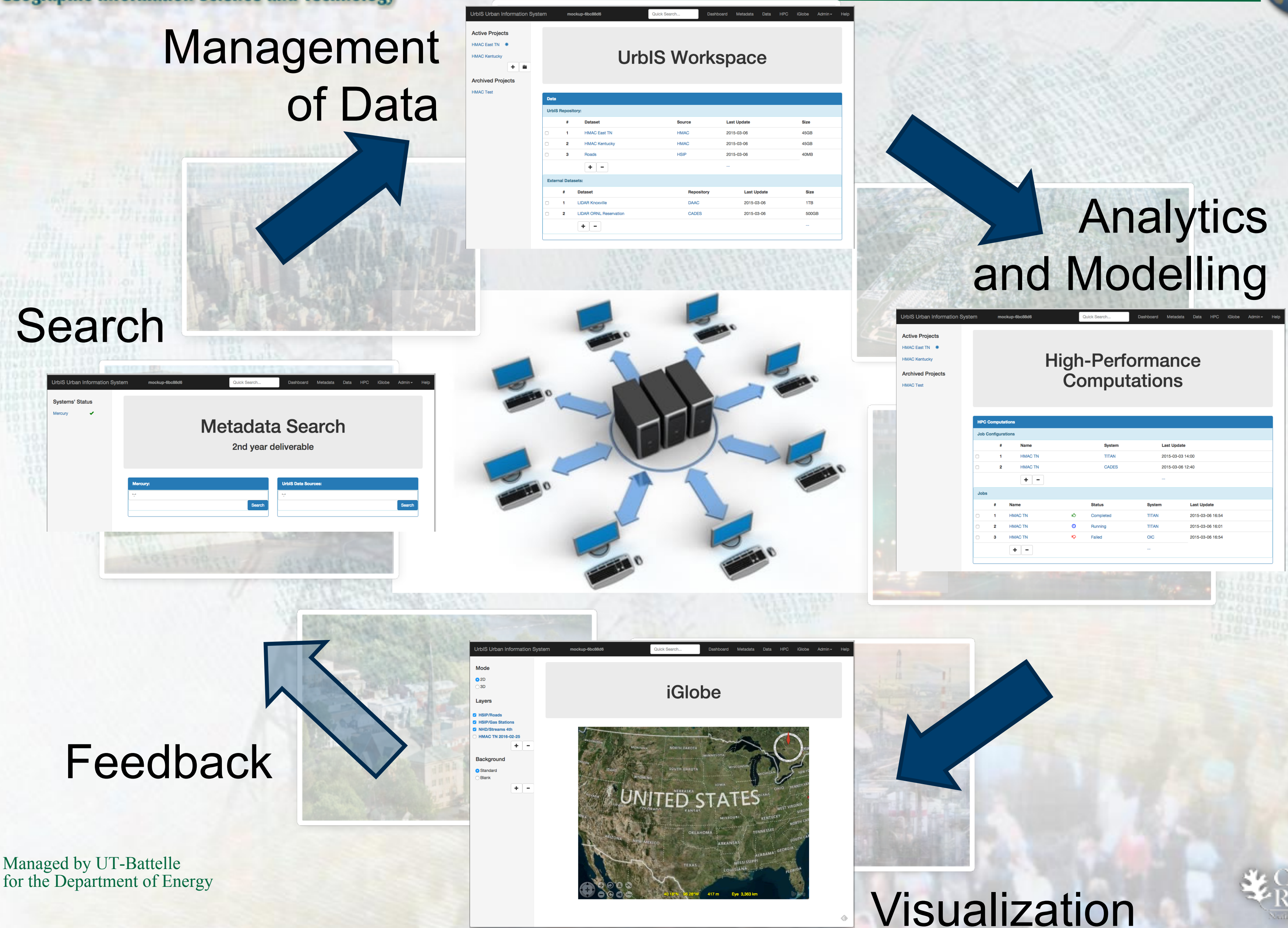




WSTAMP Server
User
User
User
WSTAMP Web Application
HTML5 (Angular)
D3
OpenLayers
Client
Client
Client
RESTful Web Service
Client
Client
Client
Analytics
RESTful Web Service
RServe
R
Database
PostgreSQL
PostGIS
Attribute Search
Apache Solr
Data as a Service
Analytics as a Service

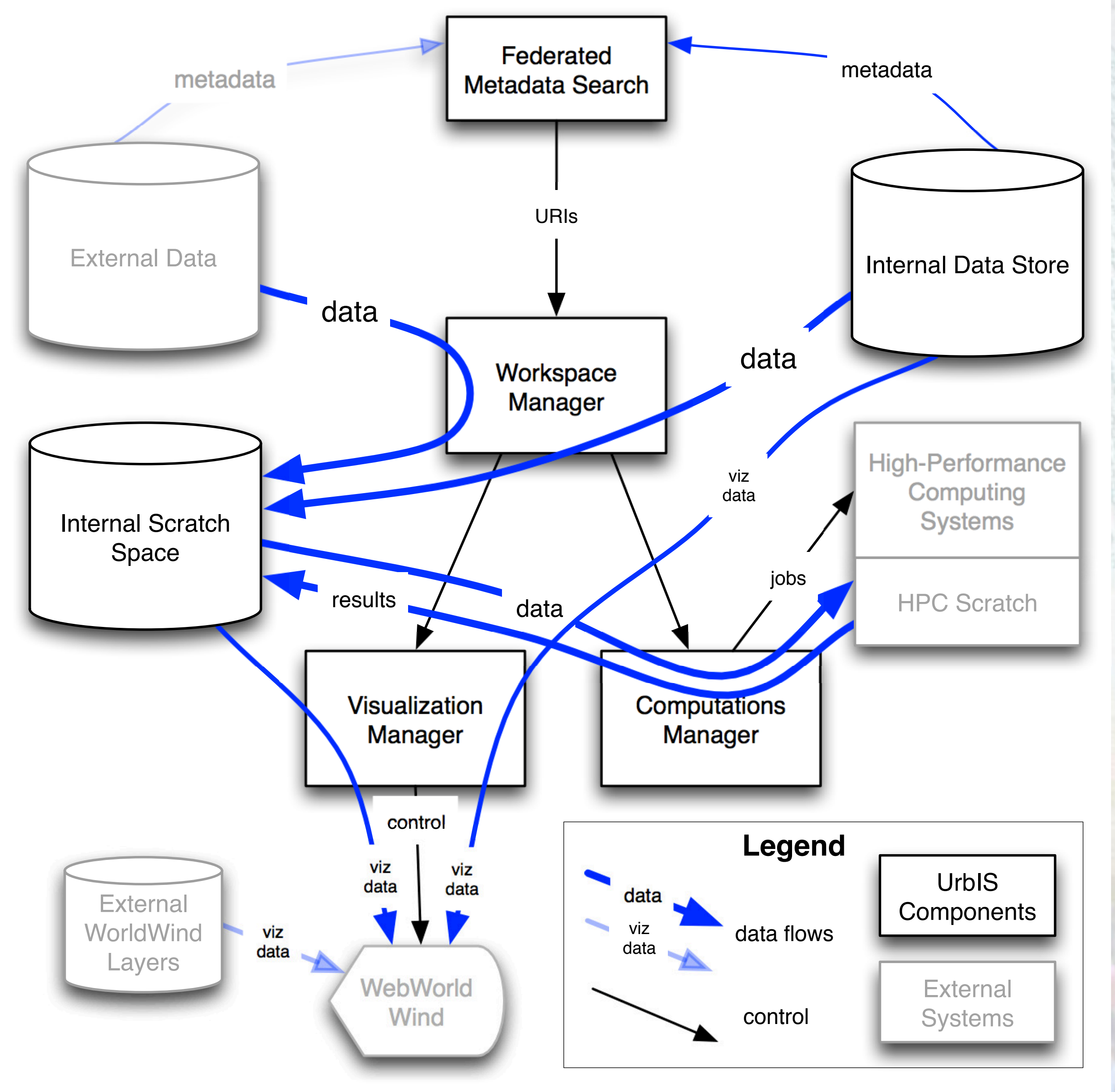
ORNL Urban Information System

Geographic Information Science and Technology



Managed by UT-Battelle
for the Department of Energy

UrbIS Architecture



Summary and Conclusions

Summary of the OSS Packages

- gdal
- Python
- R
  Rstudio
  Rserv
- QGIS
- Compiler suites
- Drupal
- NodeJS
- Bootstrap
- Sails
- CartoDB
- PostgreSQL
  PostGIS
- MongoDB
- SolR
- Geoserver
- Docker
- GDAL
- Proj.4
- GEOS
- GeoTrellis
- OpenCV
- NASA WorldWind
- GRASS
- …

Conclusions

- **OSS is everywhere**
- **OSS first!**
- **Advantages for cutting-edge research**
  - Deeply modifiable
  - Bugs are fixable
  - No usage restrictions
  - Developer community
- **Challenges**
  - Stability of licenses
  - Compilation on specialized hardware

QUESTIONS?

SOROKINA@ORNL.GOV

Acknowledgment and Copyright

- Acknowledgment
Prepared by Oak Ridge National Laboratory, P.O. Box 2008, Oak Ridge, Tennessee 37831-6285, managed by UT-Battelle, LLC for the U. S. Department of Energy under contract no. DEAC05-00OR22725.
- Copyright