Pacific Northwest
NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

Big Data "Intro to R" Bootcamp

LANDON SEGO, TONY BLADEK, AMANDA WHITE, RYAN HAFEN

Pacific Northwest National Laboratory

24 July 2014

Schedule

Morning (9:00 a.m.-12:00 p.m.)

- Bootcamp introduction-administrivia (0.5 hrs.)
- Session 1: R Introduction/Fundamentals (2 hrs.)
- Session 2: Tessera Introduction (0.5 hrs.)

Lunch (12:00 p.m.-1:00 p.m.)

Graphs 101 (LLNL) (1:00-1:30 p.m.)

Afternoon (1:30 p.m.-4:30 p.m.)

- Session 3: Introduction to Tessera tools with data (1.0 hrs.)
- Session 4: Using Tessera with Hadoop to analyze large data (1.0 hrs.)
- Session 5: Using Trelliscope with large data (0.5 hrs)
- Session 6: Summary/Feedback (0.5 hrs.)

Bootcamp Introduction

- Bootcamp Introduction (9:00-9:30 a.m.)

Bootcamp Introduction

Goal:

- Introduce R as a tool for statistically and visually analyzing data
- Provide hands-on experience with the core R language and Tessera, a collection of primarily PNNL-developed R packages that enables scalable data analytics

Bootcamp Introduction

By the end of today's Bootcamp you should

- ❑ have an understanding for when and why you might use R for statistical analysis data
- ❑ feel comfortable with the basic functions of R
- ❑ understand how R can use functions from contributed packages, such as the Tessera R packages (DataDR and Trelliscope)
- ❑ understand the scalability that you get when you use a divide and recombine methodology as implemented in the DataDR package
- ❑ know how you might use visualization in data exploration
- ❑ understand how these tools might be applied to a dataset, such as a realistic (but simulated) large data set of network traffic data (Netflow)

Trainers

- Landon Sego
  - Statistical Scientist at PNNL since 2006
  - Ph.D. Statistics (Virginia Tech)
- Amanda White
  - Scientist at PNNL since 2002
  - M.S. Operations Research (Stanford)
- Ryan Hafen
  - Statistical Scientist at PNNL since 2010
  - Ph.D. Statistics (Purdue)
  - Lead developer of Tessera tools (DataDR and Trelliscope)

Session 1: R Introduction/Fundamentals

- Session 1:
  - R Introduction/Fundamentals 9:30 a.m.- 11:05 a.m.

R Introduction/Fundamentals

- **Goal:** Introduce the power and fundamentals of R using hands-on exercises

At the end of this session, you should

- ❑ understand the basic functionality contained in the R development environment RStudio
- ❑ understand how to create and interact with data using core R functions
- ❑ understand what an R package is and where they can be found

R Introduction/Fundamentals

- What is R?
  - An integrated suite of software facilities for data manipulation, calculation, statistical analysis, and graphical display
  - Derived from a well developed, simple and effective programming language (called ‘S’)
  - Freely distributed under GNU General Public License
  - R comes with ~25 standard and recommended packages
  - DataDR and Trelliscope are R packages (libraries) which must be downloaded separately (www.tesseradata.org)

R Introduction/Fundamentals

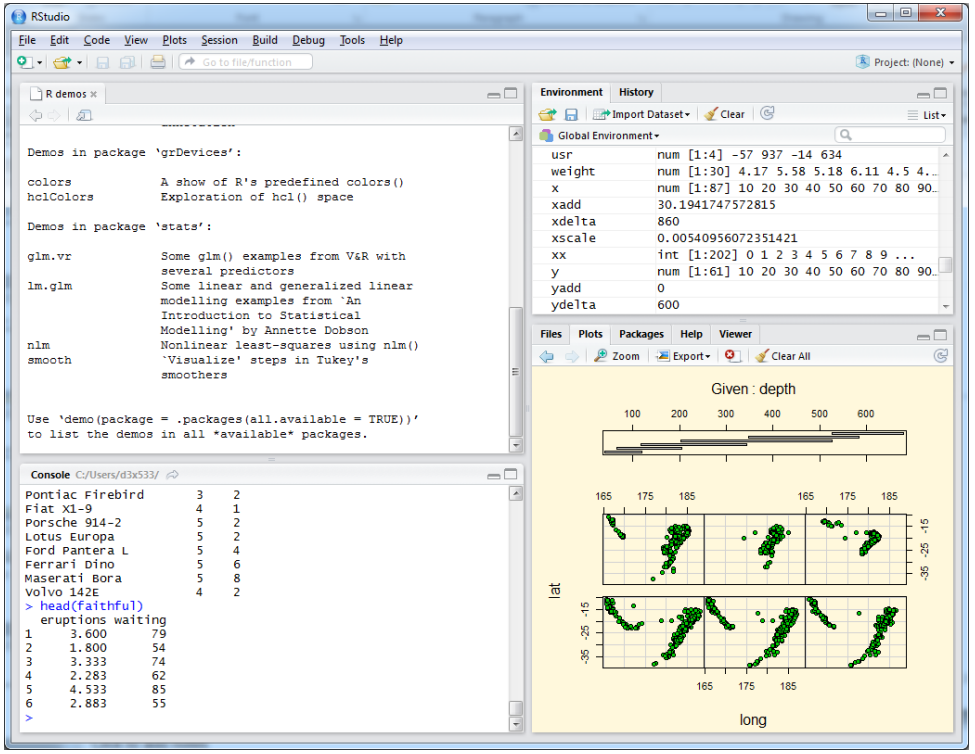
- What can R do?
- R has a wide variety of statistical and graphical methods
  - Linear and non-linear modeling
  - Descriptive statistics (min, max, median, mode, etc…)
  - Time-series analysis
  - Classification
  - Clustering
  - Quantiles (percentiles)
  - Plotting (including geospatial data)
  - Graphing
  - And much more…

R Introduction/Fundamentals

- More R facts
  - R is highly extensible
    - Over 5000 user contributed packages (libraries) on CRAN, GitHub, R-Forge and Bioconductor
    - Support for integrating other languages: C, Fortran, Java, etc.
  - It has an exceptionally effective design: programming with data
  - Thriving community of 2 million users including many commercial companies
  - Effective Core Development Group
  - Largest collection of numerical and visualization methods of any software environment for statistics and machine learning

R Introduction/Fundamentals

- Several Integrated Development Environments (IDE)s are available for R:
  - Linux
    - ***RStudio***, JGR, Rattle
  - Windows
    - ***RStudio***, Tinn-R, Revolution-R
  - Mac
    - ***RStudio***



- How do you get R?
  - R is available at
    - www.R-project.org
    - Comprehensive R Archive Network (CRAN): www.cran.us.r-project.org
  - RStudio (IDE)
    - Comes in both Desktop (Windows/Mac/Linux) and Server(Linux) versions
    - URL: https://www.rstudio.com/ide/download/

Activity 1.1: RStudio Server Introduction

- We are serving a version of RStudio on an Amazon Web Services cluster.
- Odd-numbered users go to:

  http://goo.gl/jklvfU or

  https://ec2-54-88-152-228.compute-1.amazonaws.com

- Even-numbered users go to:

  http://goo.gl/ydkCC3 or

  https://ec2-54-88-86-129.compute-1.amazonaws.com

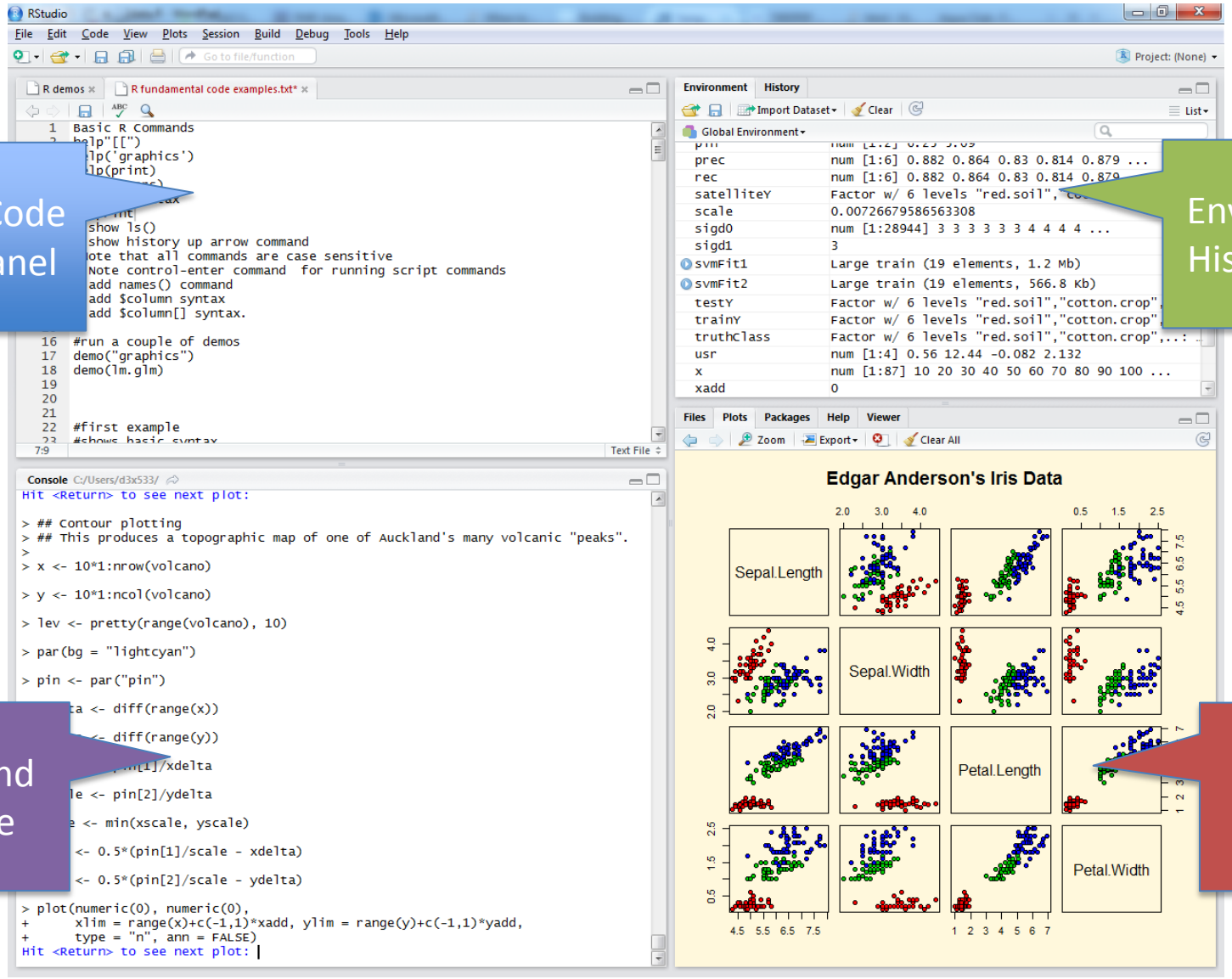
- And log in with your username and password
- See the RStudio Quick Start Guide

Activity 1.1: RStudio Server Introduction

- When you have completed this section, you should:
  - ❑ understand RStudio Panels/Interface
  - ❑ know some of the hotkeys and shortcuts

Activity 1.1: RStudio Server Introduction

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965



Demos/Code Editor Panel

Environment/ History Panel

Command Console

Output Panel

Hands-on Introduction to R

- Activity 1.2
  - Getting started – basic variables and operations in R
- Activity 1.3
  - Data structures – vectors, lists and how to interact with them
- Activity 1.4:
  - Utilities - Reading/writing to disc, packages, functions
- Activity 1.5:
  - Statistical and graphical analyses

- We'll be working with an abbreviated set of these activities today. The complete set is available at
  http://tesseradata.org/docs-r-intro-bootcamp/

Activity 1.2: Getting started

- File: “Activity_1.2.R”
- Duration: ~15 min
  - Activity 1.2.1, Help commands
  - Activity 1.2.2, Demos
  - Activity 1.2.3, R as a calculator

- When you’ve completed this section, you should
  - ❑ Be comfortable looking for help on functions
  - ❑ Know how to run the built-in demos
  - ❑ Know how to execute basic mathematical operations

Activity 1.3: Data structures

- File: “Activity_1.3.R”
- Duration: ~40 min
  - Activity 1.3.1, Numeric vectors
  - Activity 1.3.2, Character vectors
  - Activity 1.3.3, Logical vectors
  - Activity 1.3.4, Integer and complex vectors
  - Activity 1.3.5, Named vectors
  - Activity 1.3.6, Data frames
  - Activity 1.3.7, Matrices
  - Activity 1.3.8, Lists
  - Activity 1.3.9, Factors

Activity 1.3: Data structures

- When you've completed this section, you will have
  - Learned how to create data of various types including: numeric, character, vectors, lists and data frames
  - Learned how to inspect variables
  - Learned how basic mathematical functions work on those types

Activity 1.4: Utilities: Reading/writing to disc, packages, functions

- File: “Activity_1.4.R”
- Duration: ~20 min
  - Activity 1.4.1, Working directory and sourcing files
  - Activity 1.4.2, Read and write data to/from disc
  - Activity 1.4.3, Installing packages
  - Activity 1.4.4, Making your own functions

- When you’ve completed this section, you will
  - ❑ understand the RStudio working directory
  - ❑ know how to run code from a file on-disk
  - ❑ know how to interact with files created in R
  - ❑ have learned how to create a function

Activity 1.5: Statistical and graphical analyses

- File: “Activity_1.5.R”
- Duration: ~30 min
  - Activity 1.5.1, A simple linear regression model
  - Activity 1.5.2, Trellis plots
  - Activity 1.5.3, Time series analysis

- When you’ve completed this section, you should know
  - ❑ How to apply a simple linear regression model to a built-in dataset
  - ❑ How to use the basic built-in Trellis plots
  - ❑ How to apply a basic time series analysis

R-isms: Some general things to remember…

- R stores every object it uses in memory, unless you explicitly ask it to read or write from/to disc.
- Types (int, float, double, char, etc.) do not have to be explicitly declared for new R objects (like C or Java)
- Vectors are indexed from 1 to n, not from 0 to (n-1)
- Filename paths in R are written using forward slashes, regardless of operating system:
  - Linux/Mac ~/myData/someData.csv
  - Windows C:/Users/Me/myData/someData.csv
- R is CaSe SeNsiTivE!
- R will accept double or single quotes for all character strings. Use both in the same statement if you need nesting of quoted strings.

R Fundamentals: CRAN

- Important packages from CRAN
  - **ggplot2**—A graphing library for R based upon the book "The Grammar of Graphics' For our purposes, it allows for the easy creation of Trellis plots
  - **lattice**– A library for R specifically designed to help in the display of trellis graphs
  - **plyr**--plyr is a set of tools that solves a common set of problems: It supports the *Divide and Recombine* philosophy of Tessera
  - **parallel**—an R package that supports 'course-grained parallelization'. It supports the *Divide and Recombine* philosophy used by Tessera by allowing all of the operations on the 'divided' data to run in parallel

R Fundamentals:

- **Goal:** Introduce the power and fundamentals of R using hands-on exercises

At this point, you should

- ✓ understand the basic functionality contained in the R development environment RStudio
- ✓ understand how to create and interact with data using core R functions
- ✓ understand what an R package is and where they can be found

Break: 11:05-11:15 a.m.

- Break: 11:05 a.m.-11:15 a.m.

- Next: Tessera Introduction

Session 2: Tessera Introduction

Pacific Northwest NATIONAL LABORATORY
Proudly Operated by Battelle Since 1965

- Tessera Introduction (11:15 a.m. - Noon)

Tessera Introduction

Our goal in creating Tessera was to

- Enable users to
  - visually explore large datasets,
  - develop sophisticated algorithms, and
  - derive mission-critical insight
  - **with minimal lines of code**
- While providing:
  - a familiar, interactive, desktop programming environment (R)
  - automatic management of the complicated tasks of distributed storage and computation required for big data

Tessera Introduction

Tessera Introduction

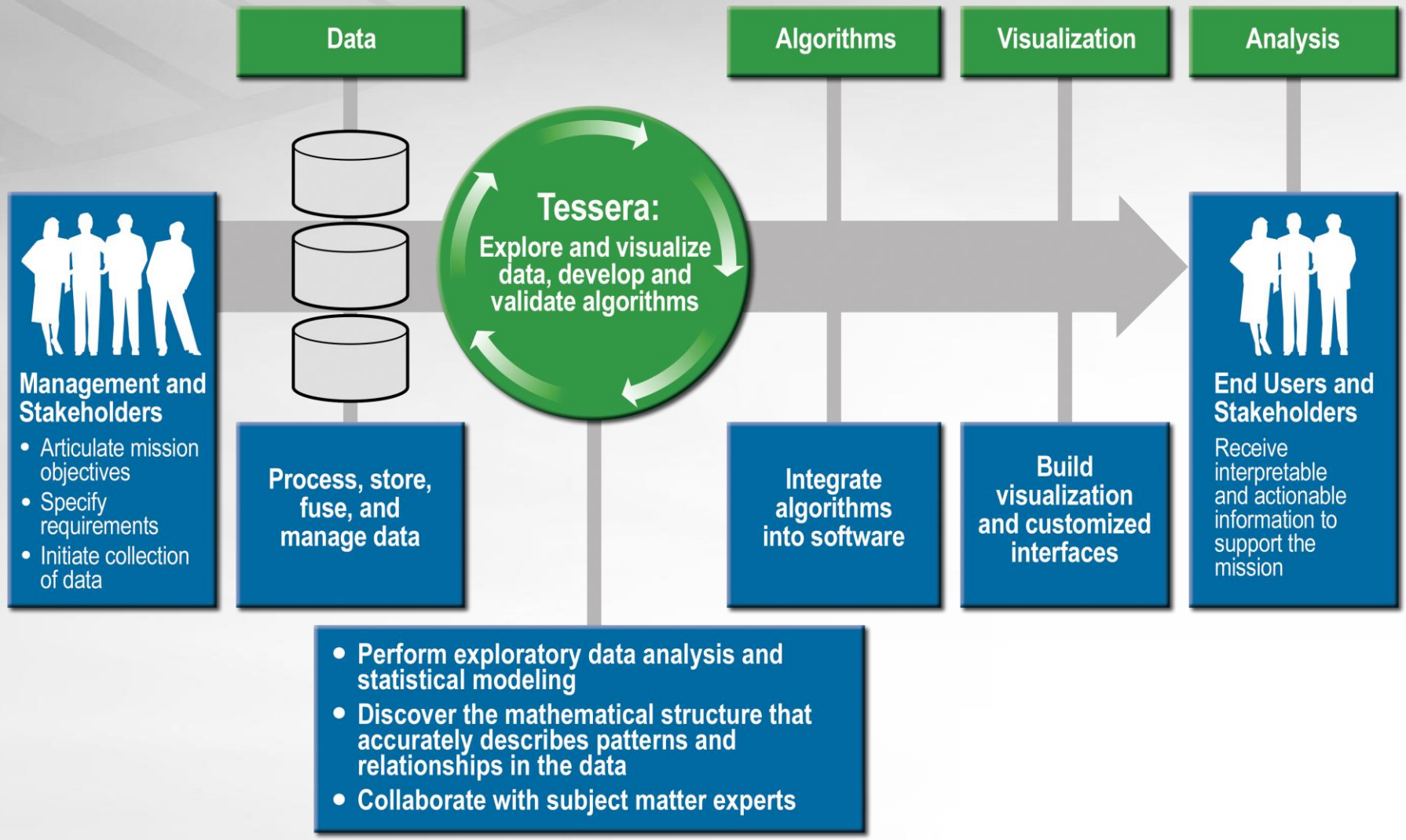
- Data analysts are overwhelmed with large, complex data
- Complex data requires deep analysis
  - Beyond summaries, tabulations, or simple interactive tools
  - Seek models that explain complex systematic behavior
  - Process is iterative: explore, hypothesize models, fit & validate models, refine
  - Final process can then be integrated into a user-friendly tool for use by domain experts
- Tessera enables deep analysis of large data for greater insight
  - Leads to better information and more precise algorithms
  - Fewer false positives
  - Greater sensitivity
  - More accurate conclusions

Tessera Introduction

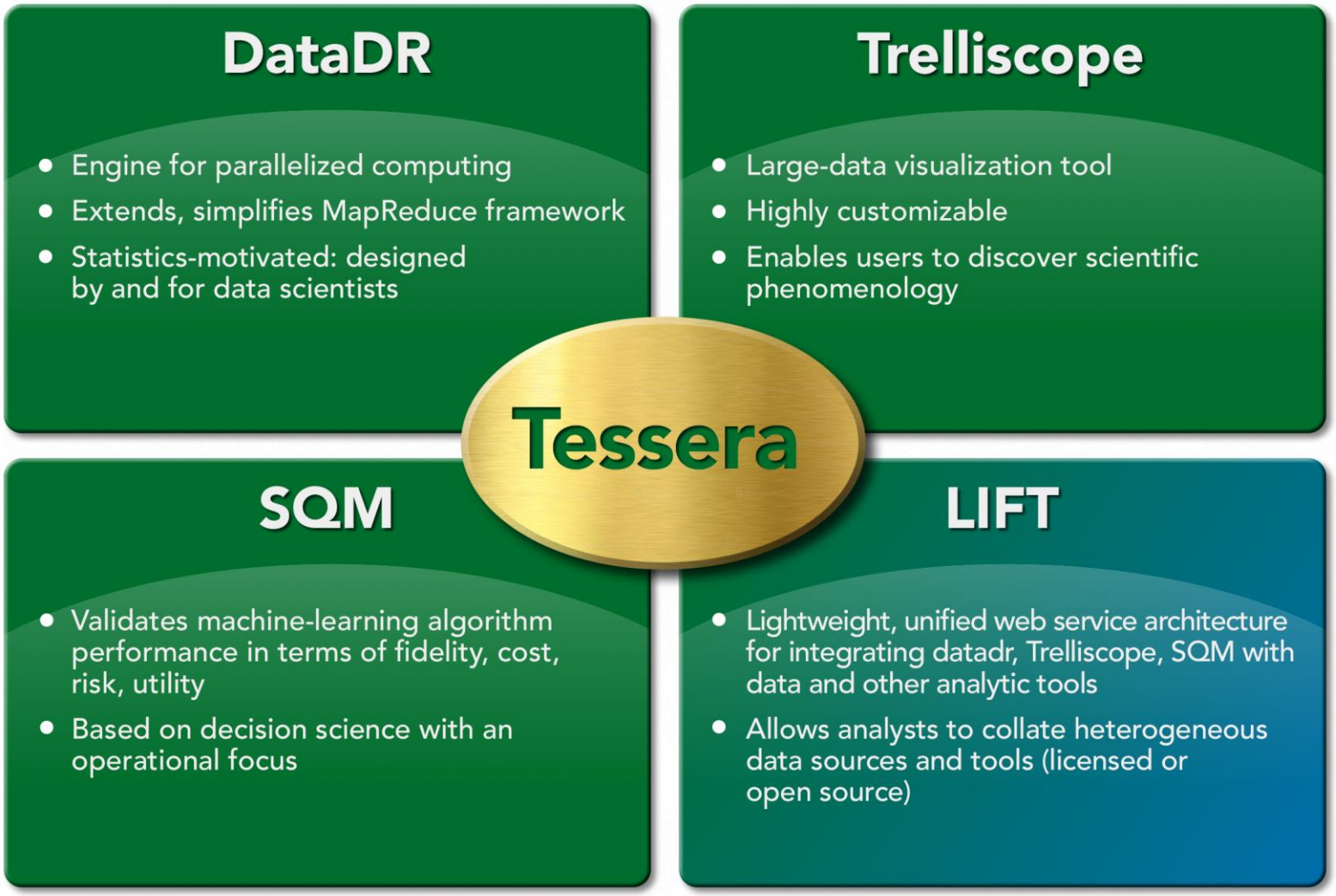
- Multiple sponsors and funding sources

- Application areas
  - Electric Power Industry- Electric Power Grid
  - High Energy Physics
  - NetFlow (cyber data)
  - Threat detection

The Role of Tessera



Tessera Fundamentals



Tessera Fundamentals

- Tessera facts
  - Developed in R
  - Open Source with multiple users
  - All the power of R is available to you on a variety of hardware types and sizes
  - Developed by data scientists

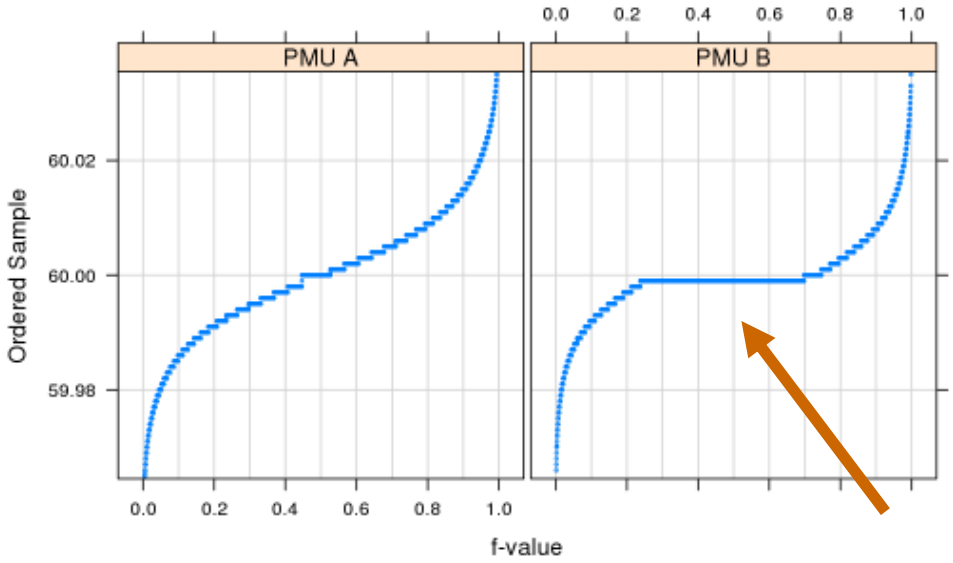
Tessera Example: Power Grid

- Comprehensive exploratory analysis to
  - characterize normal behavior
  - search for interesting events
- 1.5 years of phasor measurement unit (PMU) locations
- Data measured at 30 times per second
  - ~1.4 billion records for each variable
- 555 variables, 2 TB of data

Detailed visualization with Trelliscope discovered large amounts of bad data that had gone undiscovered

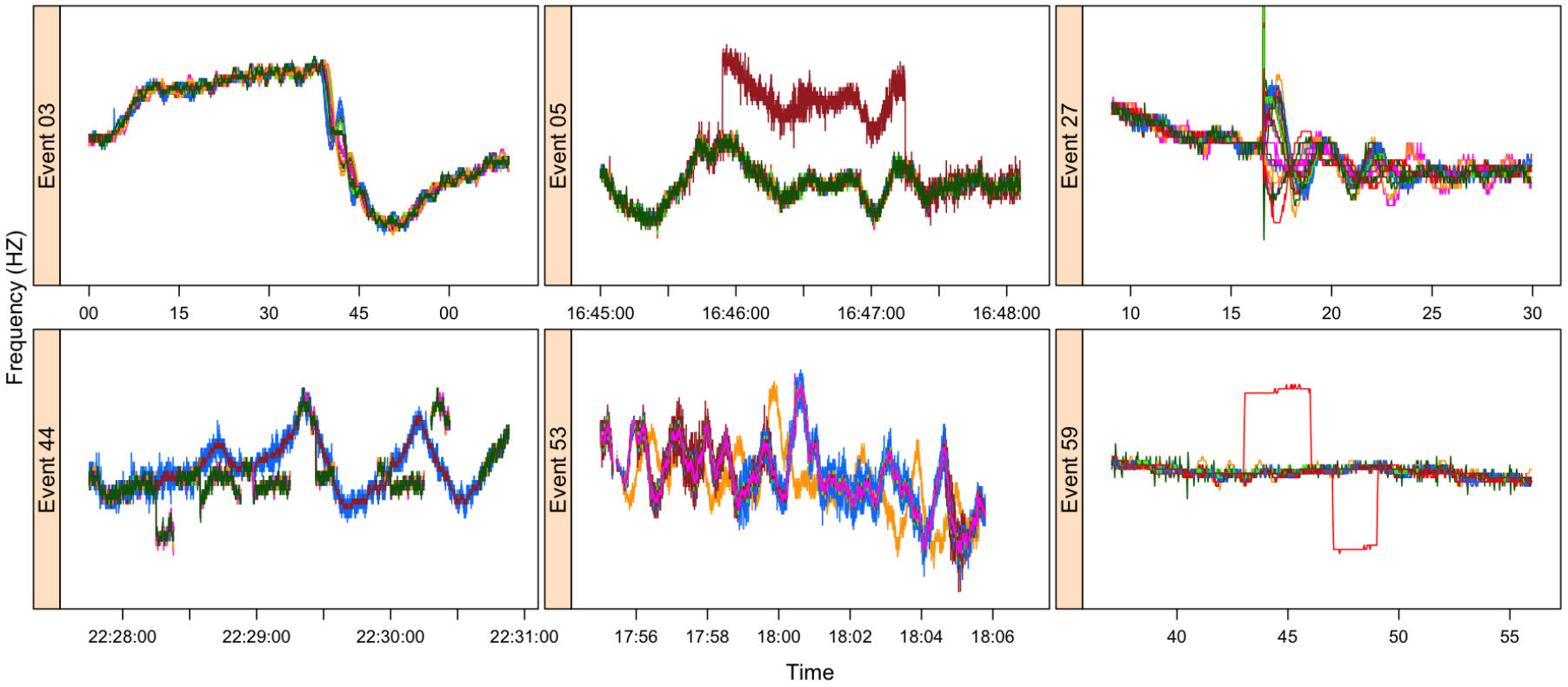
Trelliscope Example: Bad Data - Repeated Values

- Quantile plots for each of the 38 frequency series across all 1.4 billion time points were calculated and viewed with Trelliscope
- Several PMUs exhibited repeated values of 59.999
- To find the cause, we investigated the detailed data for PMUs like PMU B



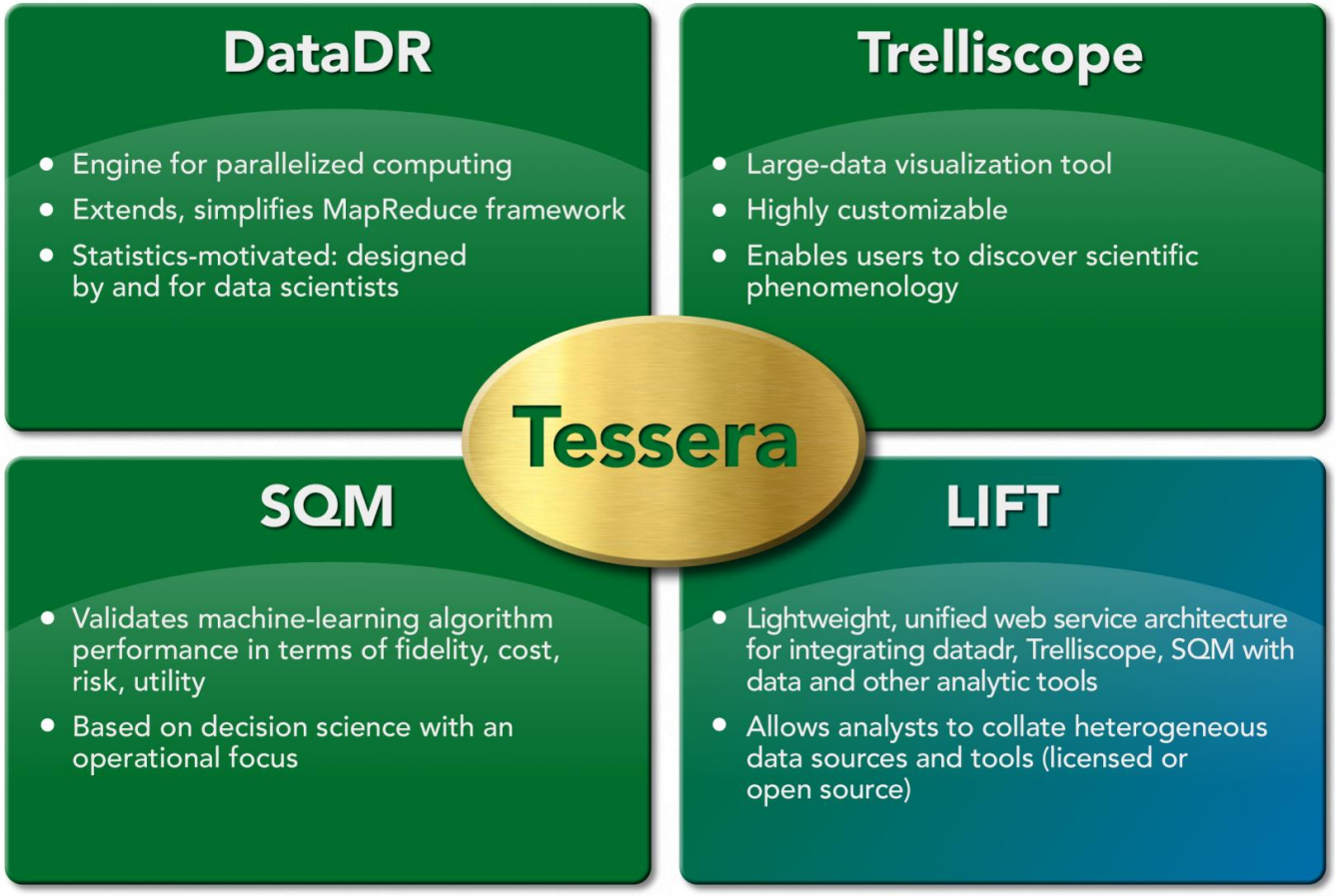
Trelliscope: Impact

- Prior to identifying bad data with Trelliscope, an event detection algorithm returned **tens of thousands of events**
- After applying automated methods of data cleaning, the algorithm returned **73 events**



Tessera Fundamentals (revisited)

- In preparation for the afternoon session, we'll review some of the fundamental concepts about Tessera



Tessera Fundamentals

Flexibility

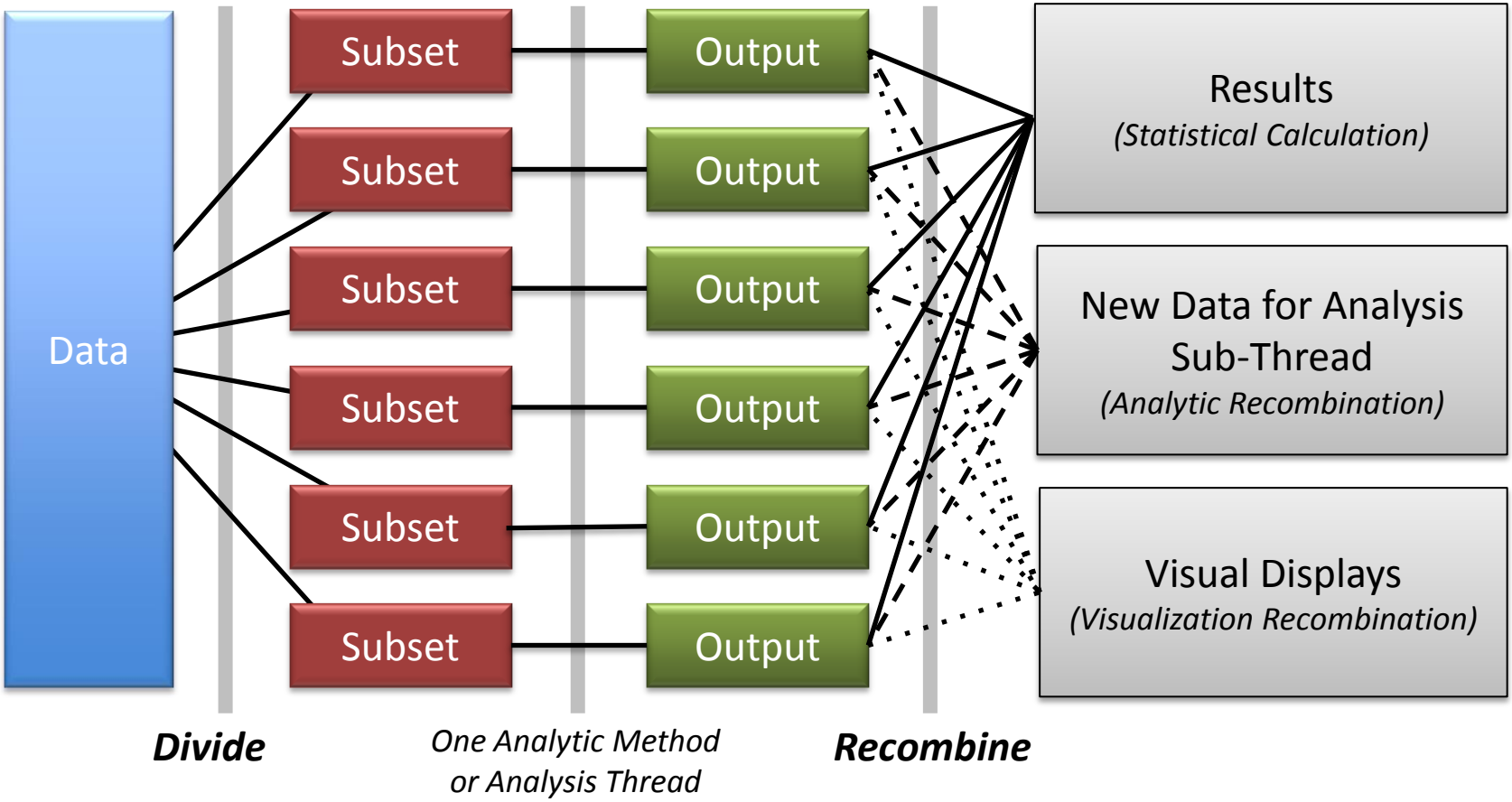
- Rapid development of code and models
- Excellent flexible statistical visualization capabilities
- Immense collection of statistical routines

+

Scalability

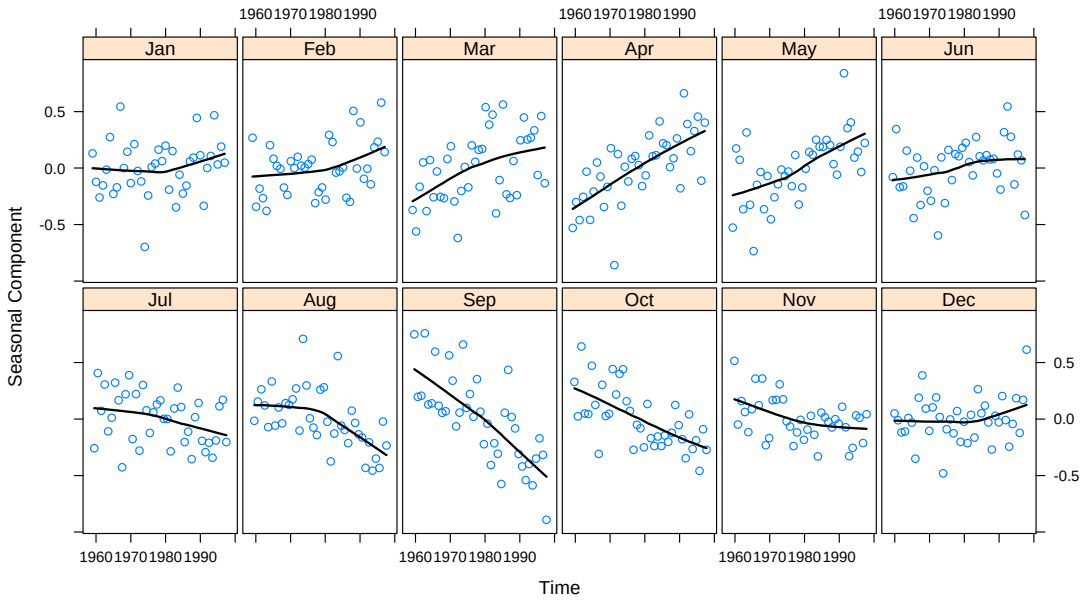
- Hides messy details of parallelization
- Takes care of partitioning, scheduling, fault tolerance, data management, and execution
- Parallel programming paradigm (MapReduce) makes sense for many statistical algorithms

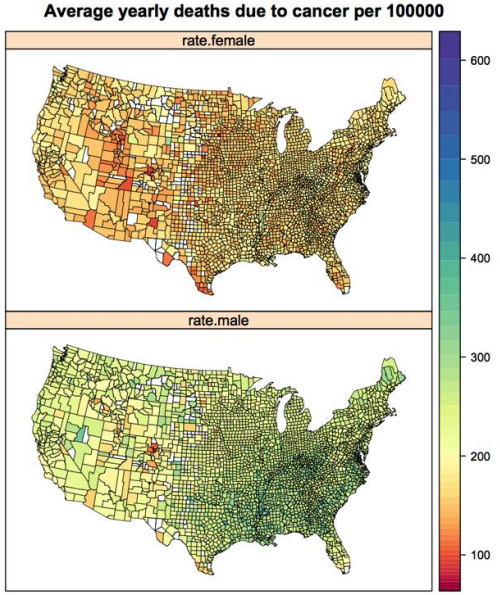
Tessera Fundamentals: DataDR



Tessera Fundamentals: Trelliscope

- Based on Trellis Display
  - Data is split into meaningful subsets
  - A visualization method is applied to each subset
  - The image for each subset is called a “panel”
  - Panels are arranged in an array of rows, columns, and pages, resembling a garden trellis

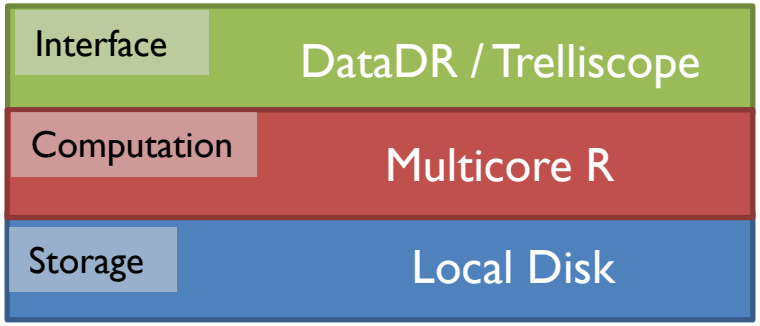




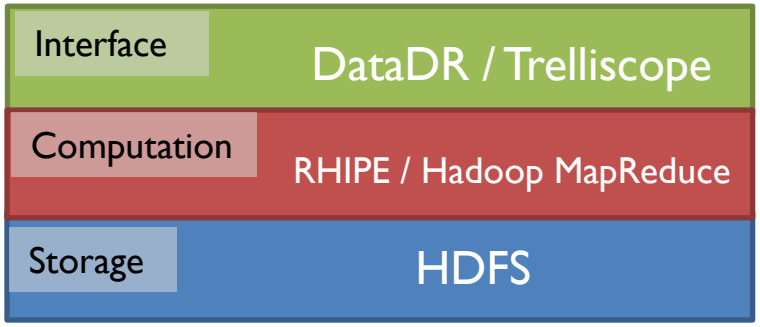
Tessera Fundamentals: Connection Types

- Regardless of what's underneath, the interface does not change

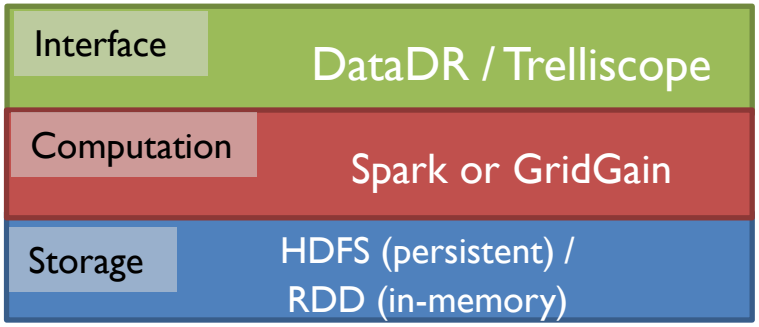
Single local node



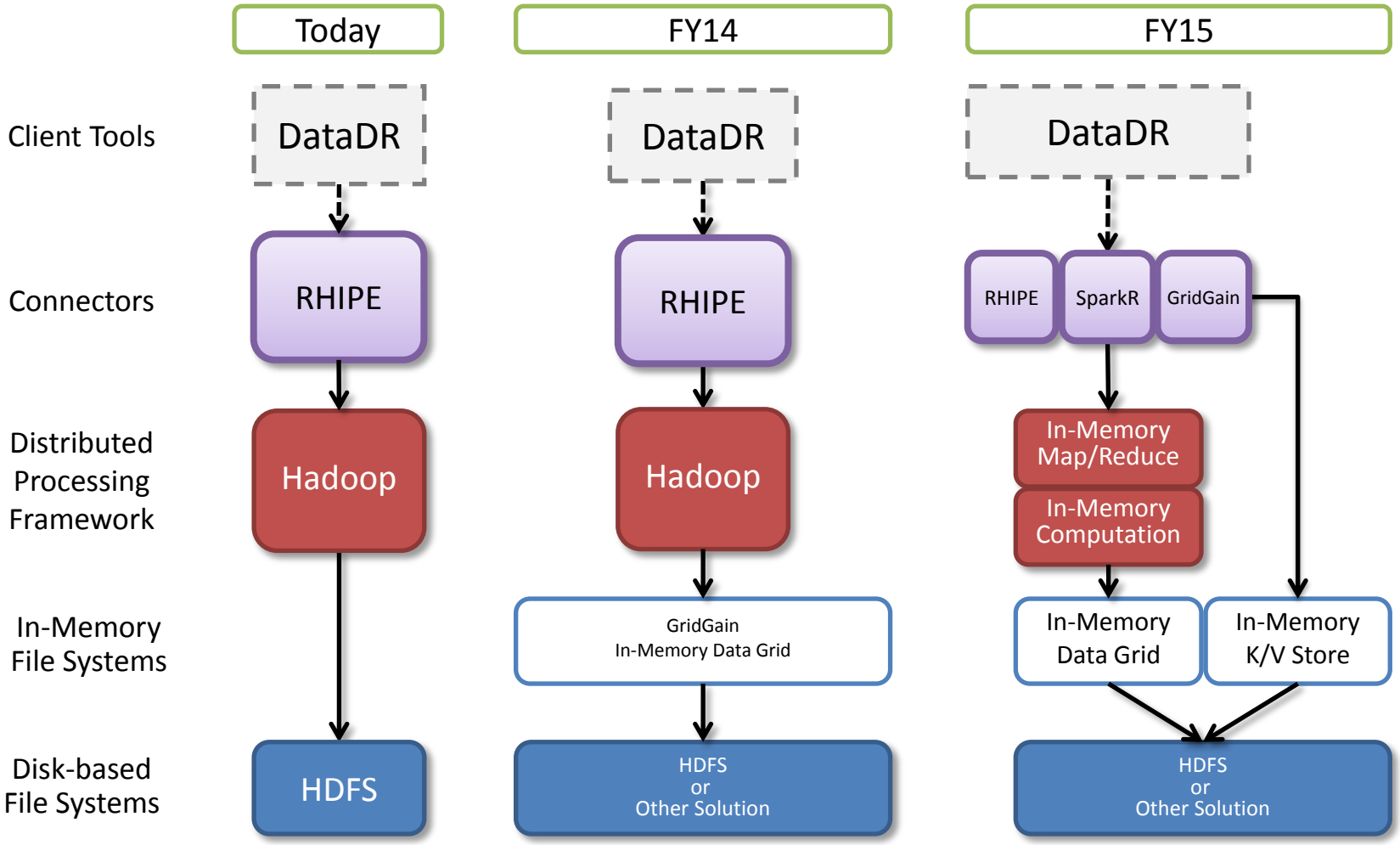
Scaling with RHIPE / Hadoop



Scaling with distributed memory



Tessera Architecture



- **Summary:**
  - Goals and motivations for Tessera
  - Example to show power of R and Tessera
  - Tessera architecture overview
    - Local computing and Hadoop/Map-Reduce computing
  - Future plans for Tessera

What's Next

- Afternoon sessions
  - Introduction to DataDR and other Tessera 'R'-based tools
  - Using Tessera tools with Hadoop to analyze large data
  - Using Trelliscope visualizations to explore large data sets

Lunch Break: 12:00 p.m.-1:00 p.m.

- Lunch Break (12:00 p.m.-1:00 p.m.)
- Graphs 101 Lightning Talk (1:00-1:30 p.m.)
- Afternoon: Intermediate Session
  - DataDR and other Tessera ‘R’-based tools