Sterling Survey System

Designing the Server, Database, and Administrative Tools in Support of a Multi-Mode, Multi-Platform Survey Authoring Framework

Chris Schlapper
University of Wisconsin Survey Center
University of Wisconsin-Madison

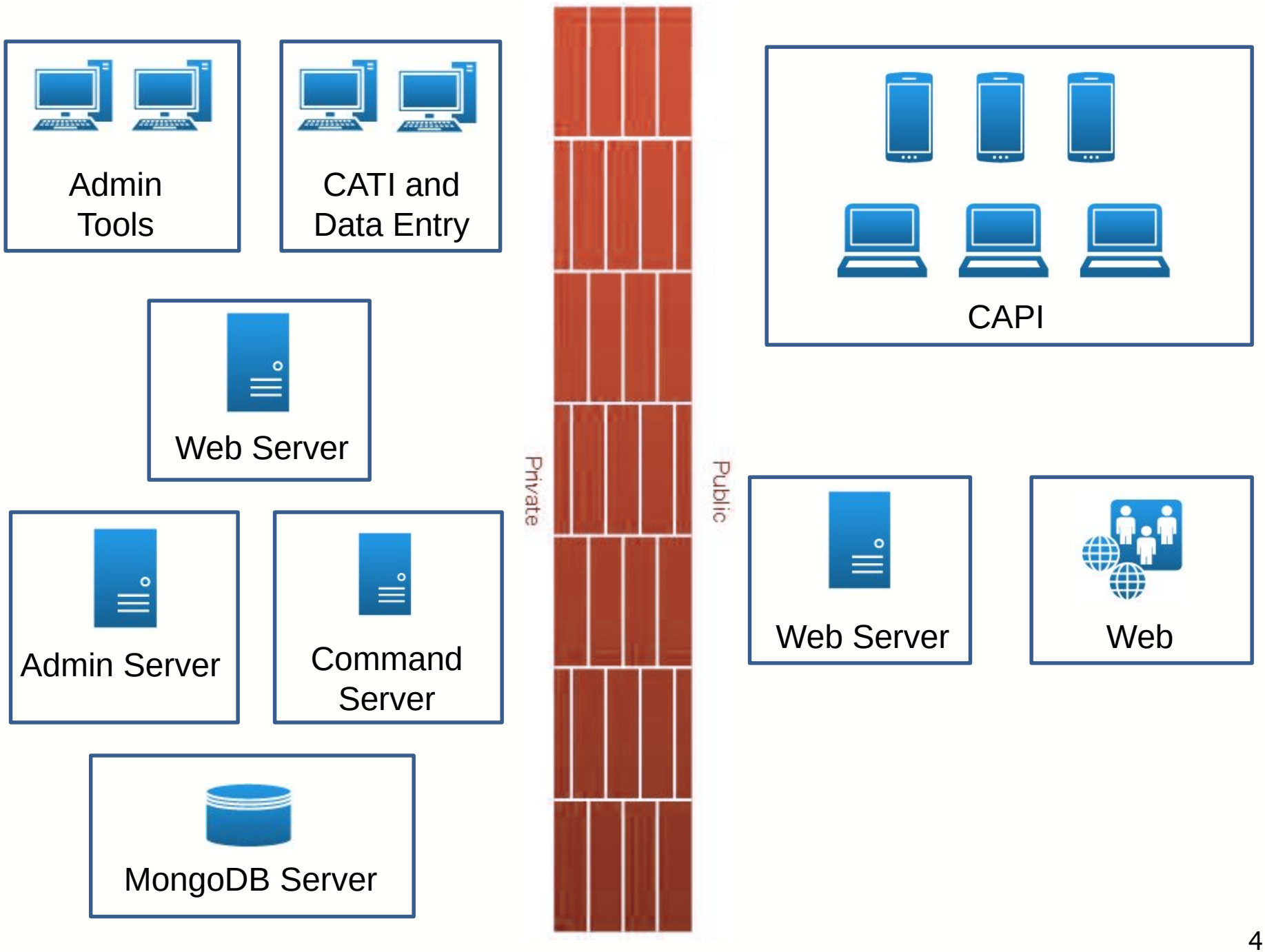
2017 International Field Directors and Technologies Conference
May 23, 2017

Sterling Survey System

- Survey Authoring Framework
  - All survey modes
  - Multi-Mode
  - Multiple Platforms and devices

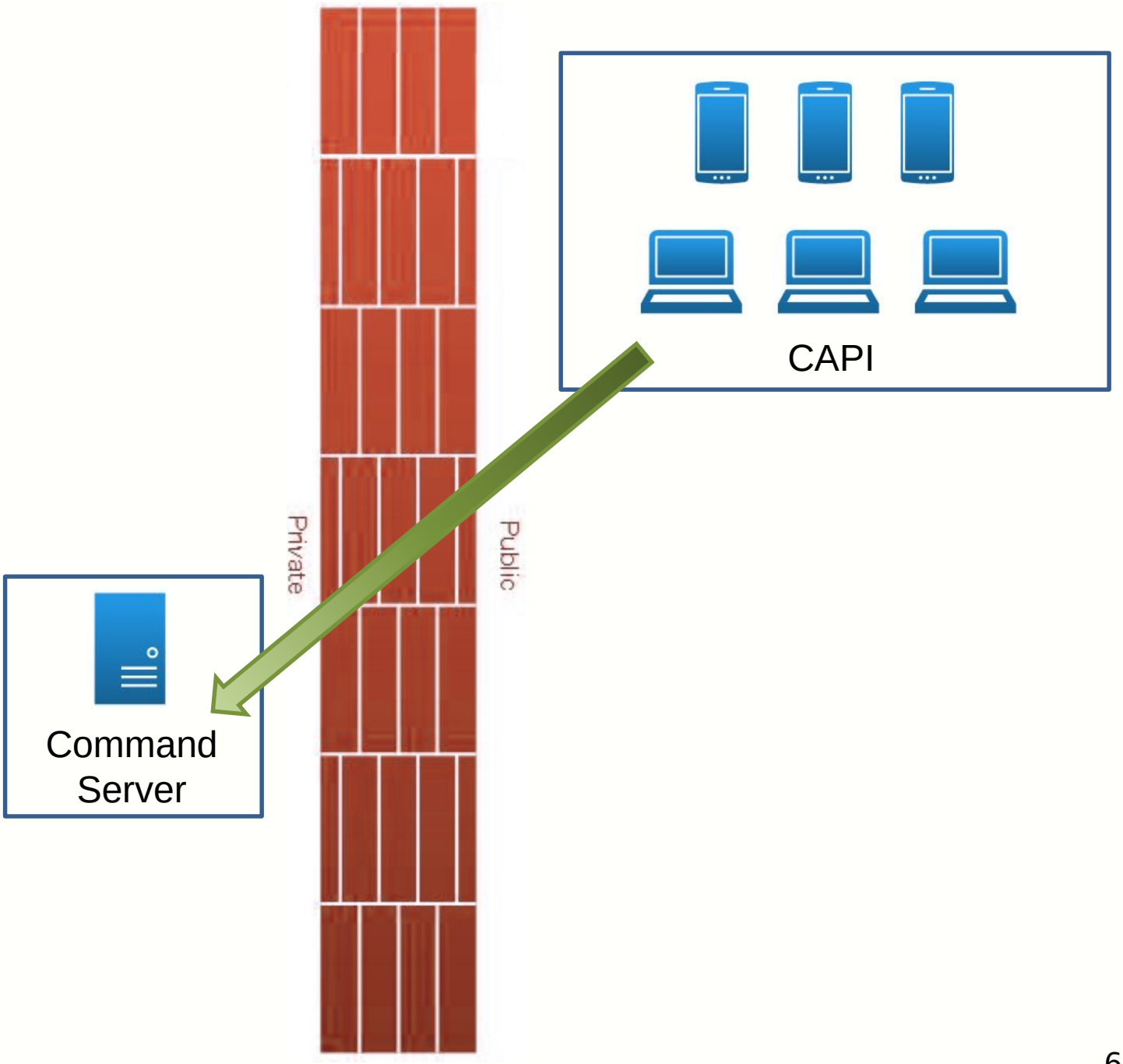
Server-Side Implementation

- Server configuration
- Server Framework
- Database
- Survey administrative tools


Admin Tools
CATI and Data Entry
CAPI
Web Server
Private
Public
Admin Server
Command Server
Web Server
Web
MongoDB Server

Server-Side Implementation

- Server configuration
- Server Framework
- Database
- Survey administrative tools


CAPI
Private
Public
Command
Server

Server Framework Technology – Node.js

- Open-Source JavaScript runtime environment
  - Execute JavaScript code outside of a browser
- Open-Source libraries
    - Extend capabilities of standard JavaScript
    - Examples
      - https
      - Jquery
      - Bluebird
      - JsonToCsv

Why Node.js?

- JavaScript, same software language used for instrument authoring
- Large open-source libraries
- Ability to build a targeted web communication protocol
  - Full web-server functionality not desired
  - Only small set of valid commands needed

Command Server Functions

- Programmed using Node.js
- Validates client-issued commands
  - JSON data structure containing parameters
  - Invalid commands or parameters are not processed
- Executes allowed commands
- Communicates with database
  - User account with limited access
  - Updates
  - Queries data to return to client device

Command Server – Three Commands

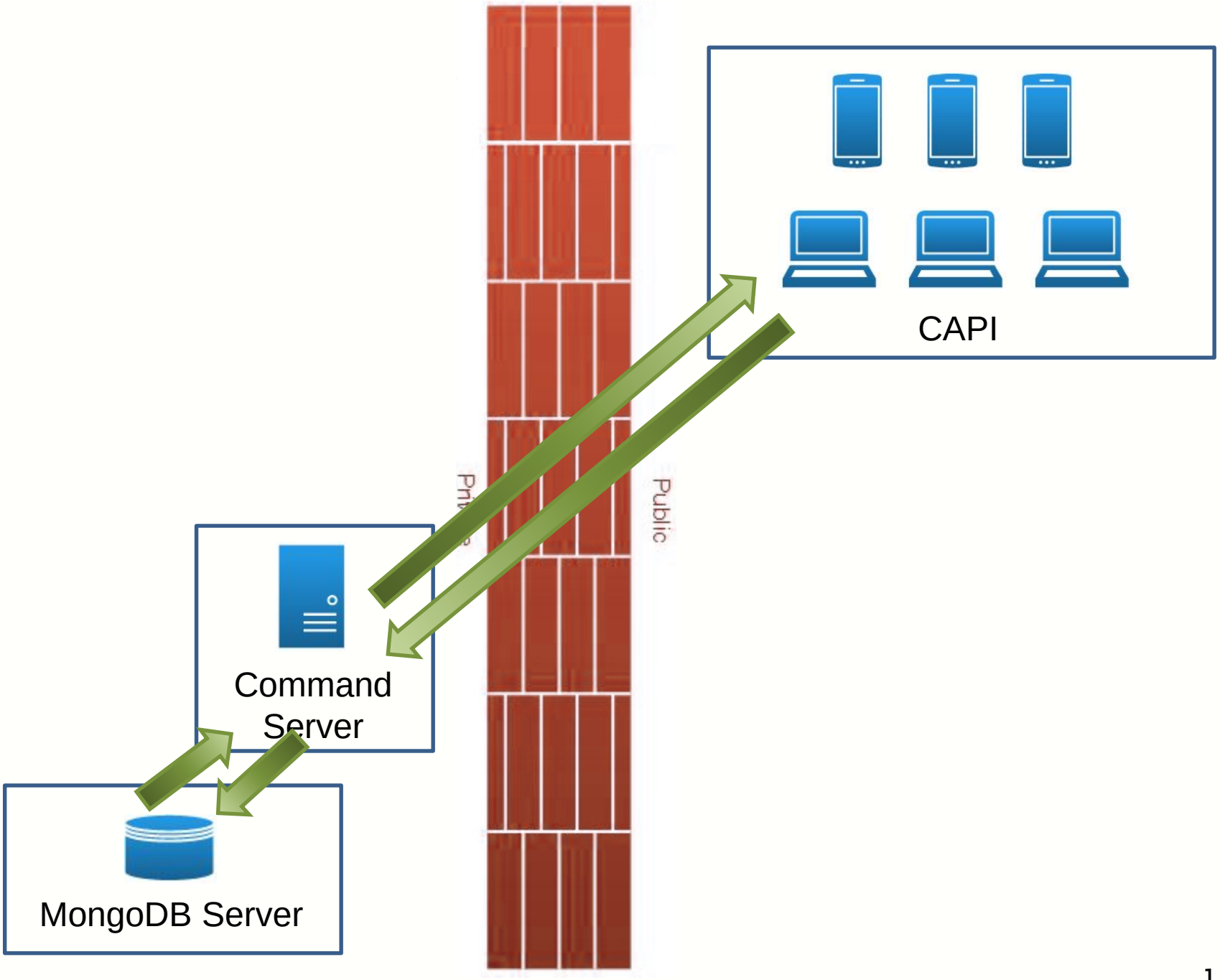
- Save cases to database
  - Collected by the instrument
  - All data not previously saved

- Synchronize
  - Adds or removes assigned cases on client device
  - Updates case ownership in database

- Save LocalStorage
  - Copies client device's LocalStorage to database
  - Backup and debugging resource

Server-Side Implementation

- Server configuration
- Server Framework
- Database
- Survey administrative tools


CAPI
Public
Command
Server
MongoDB Server

Server Database Technology – NoSQL Database

- MongoDB
  - Stores documents, not records
  - Associates data in complex ways
  - JSON data structure
    - Key/Value pairs
    - Nested sub-documents
      - As many nested sublevels as you need
      - Rosters
  - Flexibility in database design
  - Data mining

Why Not SQL Database?

- Relational model
  - MySQL
    - Familiarity
    - Wide-adoption
  - Strict schema definitions
    - Changes during field period
    - Schema changes over time
      - Querying across multiple projects may fail

Why Not SQL Database?

- Entity-Attribute-Value (EAV) model

csid	q1	q2	q3	q4	q5	q6
113489	Y	15	10/31/2016	N	DK	14

entity	attribute	value
113489	q1	Y
113489	q2	15
113489	q3	10/31/2016
113489	q4	N
113489	q5	DK
113489	q6	14

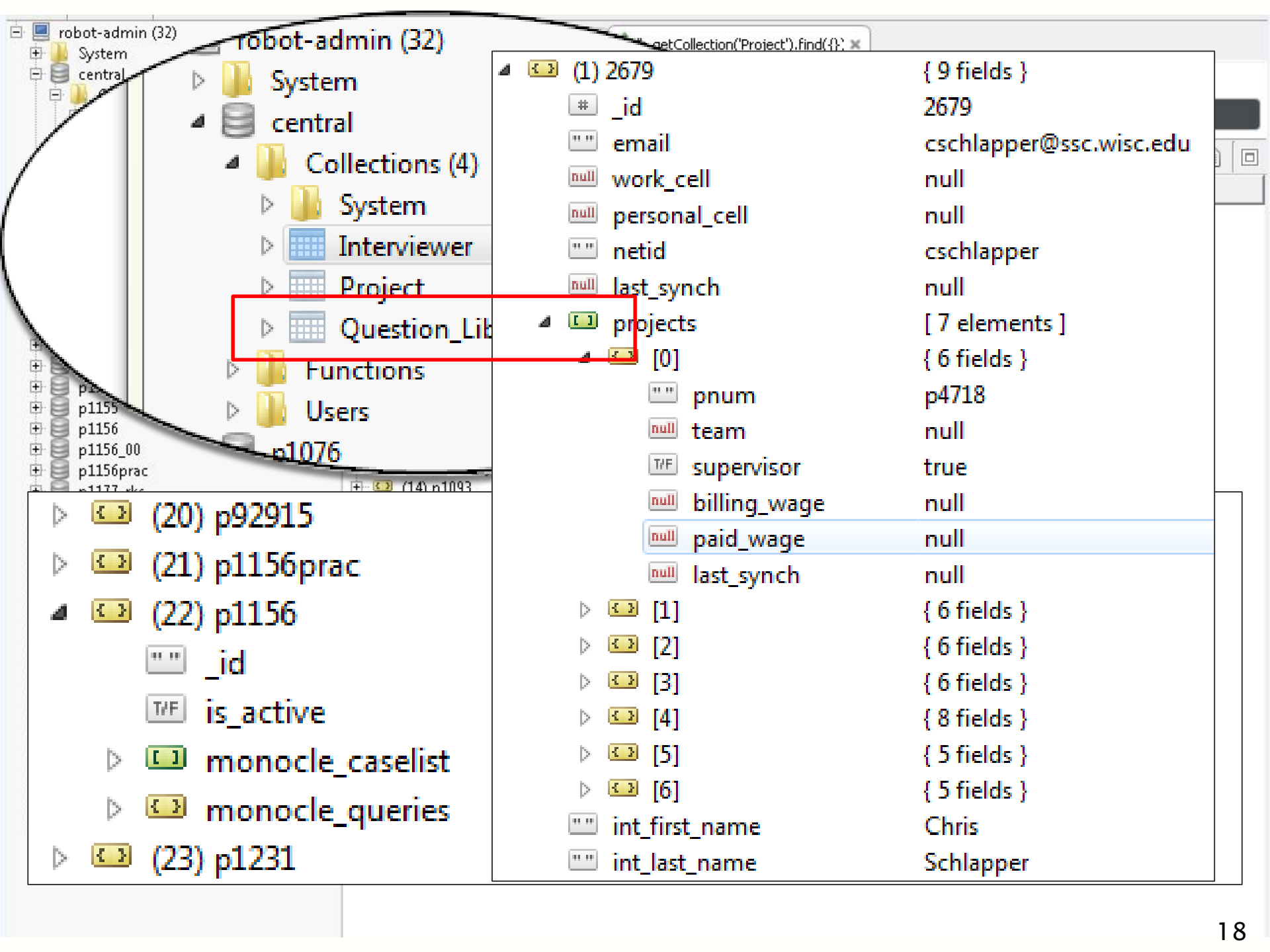
- Provides flexibility
- Writing queries more challenging
  - Search the attributes to understand the data structure
- Slow query response time
  - Pivot tables, GROUP_CONCAT

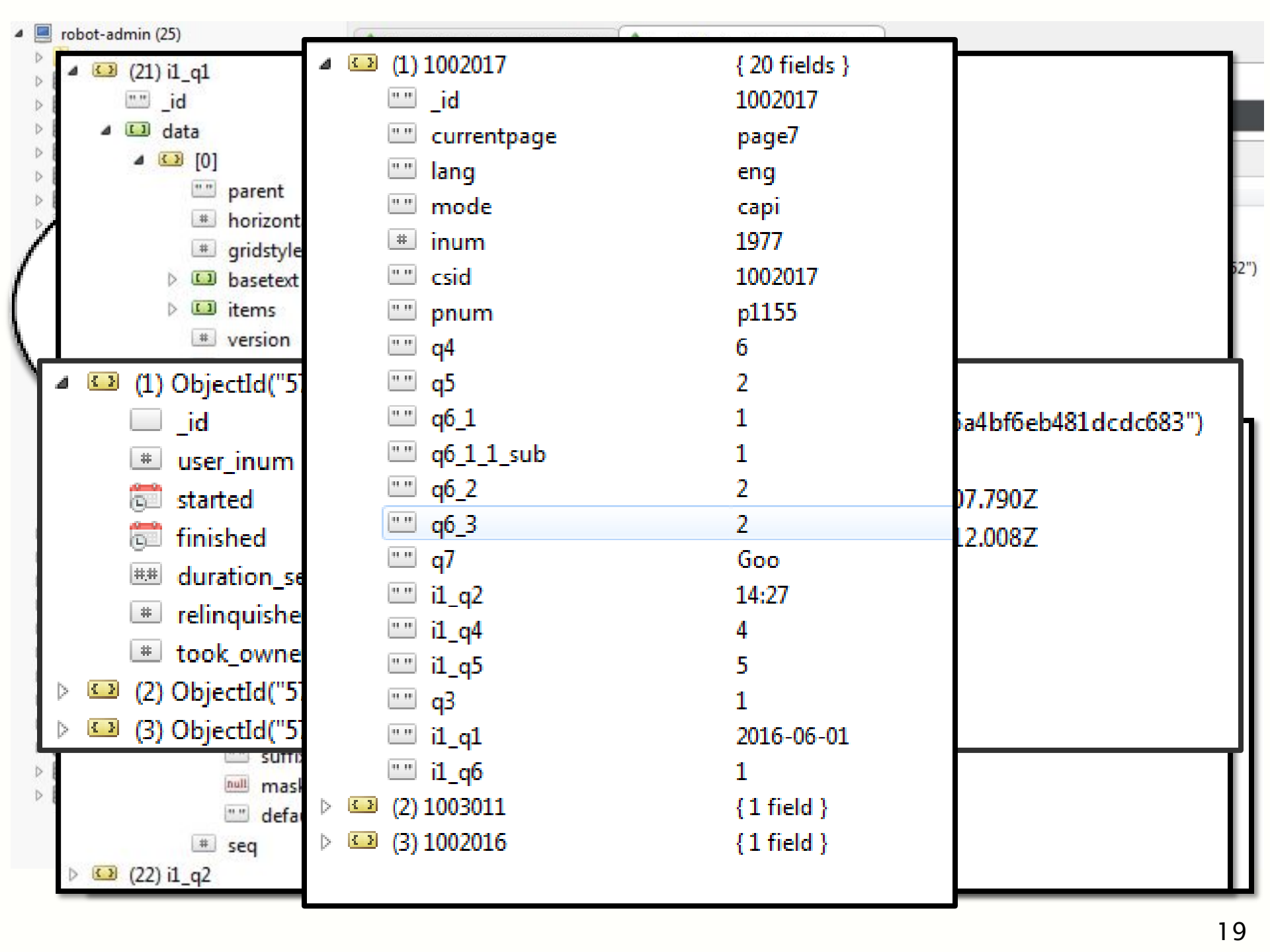
Why MongoDB?

- Non-Relational database
  - No mandatory schema definitions
    - Allow for changes during field period
  - Implement standard variables
    - Allow for querying across multiple projects without breaking for older projects
  - Nested sub-documents

Why MongoDB?

- Big Data
  - Indices
  - Fast query-response time
  - Scale horizontally
    - Distribute data across multiple machines
- JSON data structure
    - Native to JavaScript/node.js
    - Client-side and server-side


robot-admin (32)
System
central
Collections (4)
System
Interviewer
Project
Question_Lib
Functions
Users
p1076
(1) 2679
{ 9 fields }
_id
2679
email
cschlapper@ssc.wisc.edu
work_cell
null
personal_cell
null
netid
cschlapper
last_synch
null
projects
[7 elements]
[0]
{ 6 fields }
pnum
p4718
team
null
supervisor
true
billing_wage
null
paid_wage
null
last_synch
null
[1]
{ 6 fields }
[2]
{ 6 fields }
[3]
{ 6 fields }
[4]
{ 8 fields }
[5]
{ 5 fields }
[6]
{ 5 fields }
int_first_name
Chris
int_last_name
Schlapper
(20) p92915
(21) p1156prac
(22) p1156
_id
is_active
monocle_caselist
monocle_queries
(23) p1231


robot-admin (25)
(21) i1_q1
_id
data
[0]
parent
horizont
gridstyle
basetext
items
version
(1) ObjectId("5
_id
user_inum
started
finished
duration_se
relinquishe
took_owne
(2) ObjectId("5
(3) ObjectId("5
null
mask
defa
seq
(22) i1_q2
(1) 1002017
{ 20 fields }
_id
1002017
currentpage
page7
lang
eng
mode
capi
inum
1977
csid
1002017
pnum
p1155
q4
6
q5
2
q6_1
1
q6_1_1_sub
1
q6_2
2
q6_3
2
q7
Goo
i1_q2
14:27
i1_q4
4
i1_q5
5
q3
1
i1_q1
2016-06-01
i1_q6
1
(2) 1003011
{ 1 field }
(3) 1002016
{ 1 field }
a4bf6eb481dcdc683")
07.790Z
12.008Z

Node.js Implementation Notes

- Node Package Manager (npm)
  - Install libraries
    - Some libraries depend on other libraries
    - Global or local installation
- Asynchronous code execution
  - Ideal for client-side website
  - Challenges on server-side
  - Promises
    - Bluebird module

MongoDB Notes

- No security “out-of-the-box”
- NoSQL query language
  - Different from SQL
  - Retrieving data from sub-documents
- RoboMongo
  - Administrative console program

MongoDB Notes – Database Design

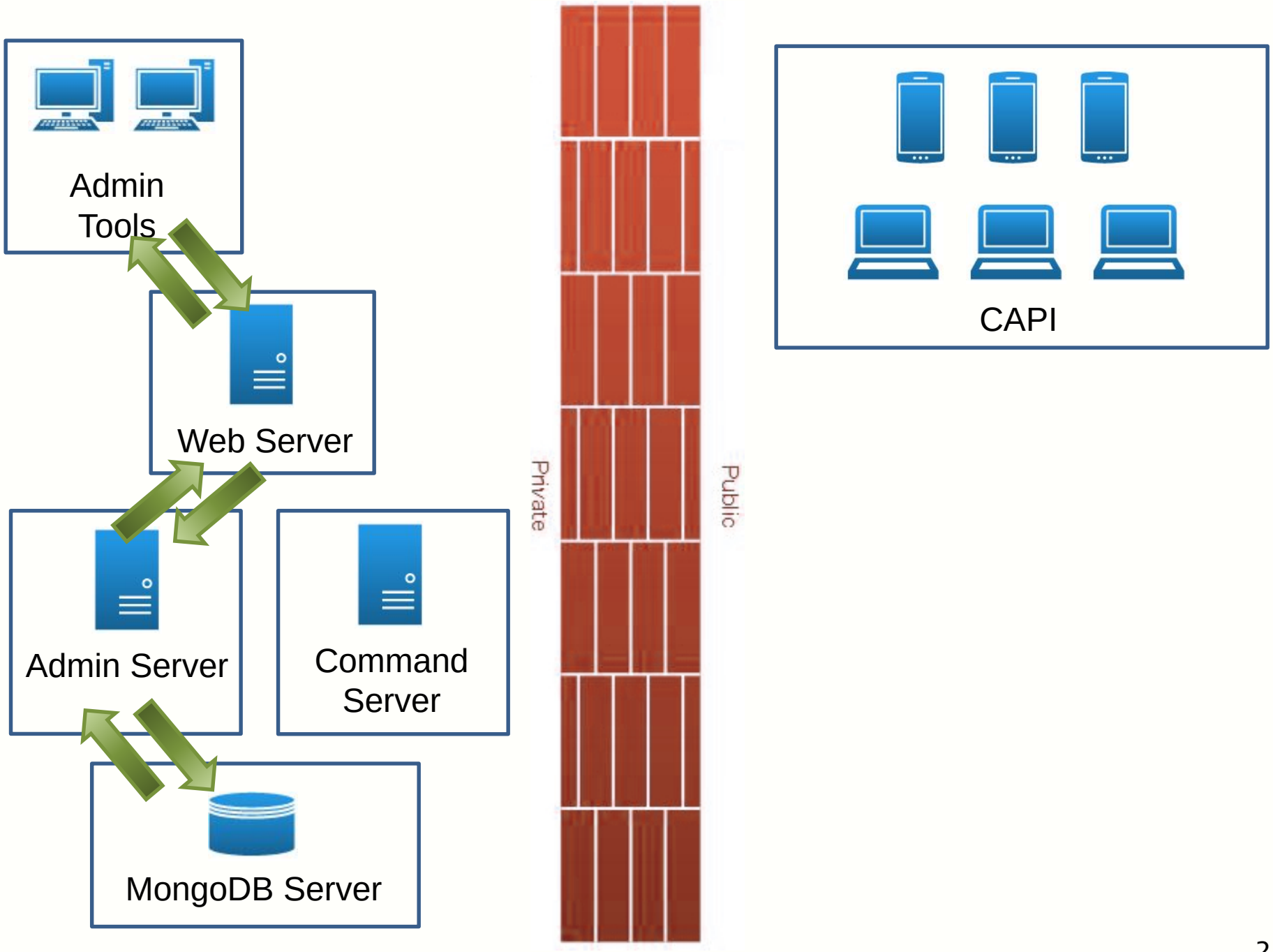
- Optimize for NoSQL
- No table joins
  - Run multiple queries and programmatically join the data
  - Data-duplication recommended
    - Store “join” data as sub-document
- JSON structure
  - Associative arrays
  - Avoid using a value for the key
    - Optimized for client-side
    - Difficult to query on database
      - Key analogous to a database column name

Server-Side Implementation

- Server configuration
- Framework
- Database
- Survey administrative tools

Administrative Requirements

- Used by Project Directors, Programmers, and Field Supervisors
- Save Question data
  - During survey authoring process
- Instrument debugging
- Load sample
- Manage project during field period
  - Data dashboard and reports
  - Case assignment
  - Roll-back case to a prior version
- Data delivery


Admin Tools
Web Server
Admin Server
Command Server
MongoDB Server
Private
Public
CAPI

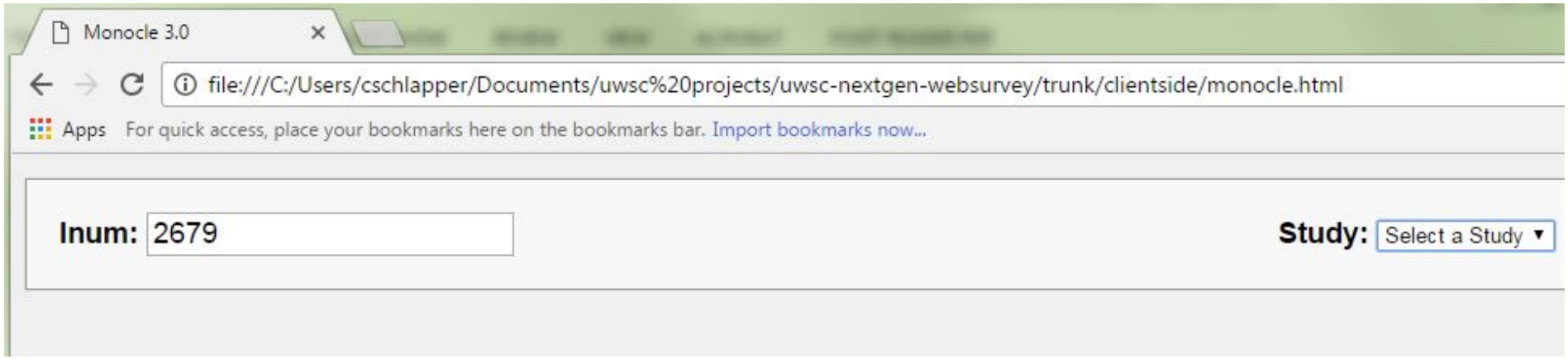
Admin Server Functions

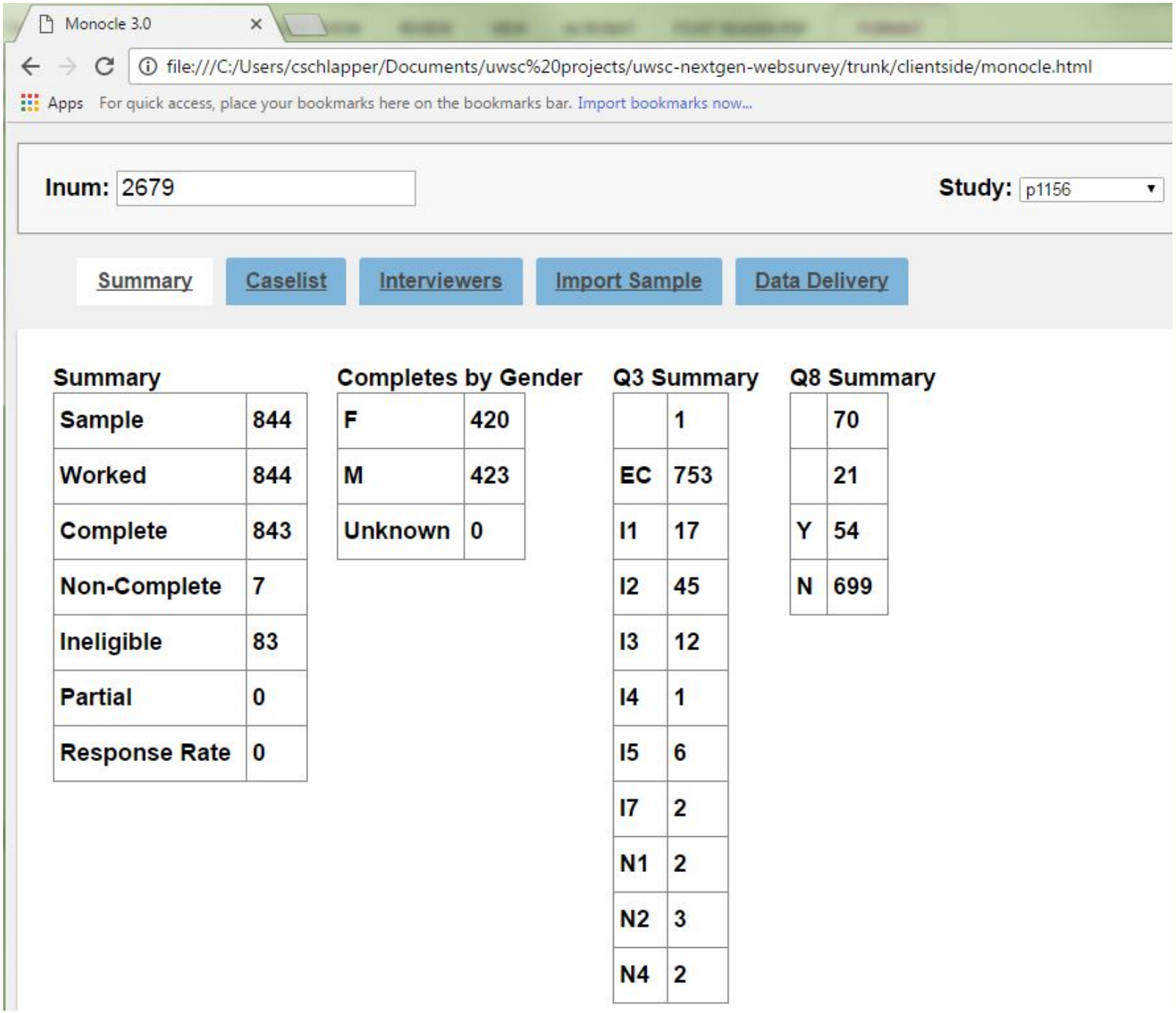
- Programmed using Node.js
- Separate path from client-issued commands
  - Behind firewall
  - More functionality
  - JSON data structure containing parameters
  - Invalid commands or parameters are not processed
  - Requires greater security permissions
- Executes allowed commands
- Communicates with database
  - User account with elevated access
    - Create new schema
    - Access to administrative data collections

Admin Server – Commands

- Save Question data during authoring process
- Load sample data
- Create data delivery files
- Fulfill queries for Project Management software

Monocle 3.0


Monocle 3.0
file:///C:/Users/cschlapper/Documents/uwsc%20projects/uwsc-nextgen-websurvey/trunk/clientside/monocle.html
Apps
For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Inum: 2679
Study: Select a Study


Monocle 3.0
file:///C:/Users/cschlapper/Documents/uwsc%20projects/uwsc-nextgen-websurvey/trunk/clientside/monocle.html
Apps For quick access, place your bookmarks here on the bookmarks bar. Import bookmarks now...
Inum: 2679
Study: p1156
Summary
Caselist
Interviewers
Import Sample
Data Delivery
Summary
Sample 844
Worked 844
Complete 843
Non-Complete 7
Ineligible 83
Partial 0
Response Rate 0
Completes by Gender
F 420
M 423
Unknown 0
Q3 Summary
1
EC 753
I1 17
I2 45
I3 12
I4 1
I5 6
I7 2
N1 2
N2 3
N4 2
Q8 Summary
70
21
Y 54
N 699

Inum: 2679 Study: p1156

Summary | Caselist | Interviewers | Import Sample | Data Delivery

Assign Selected Cases to: Schlapper, Chris (2679) Assign Cases

Show 10 entries

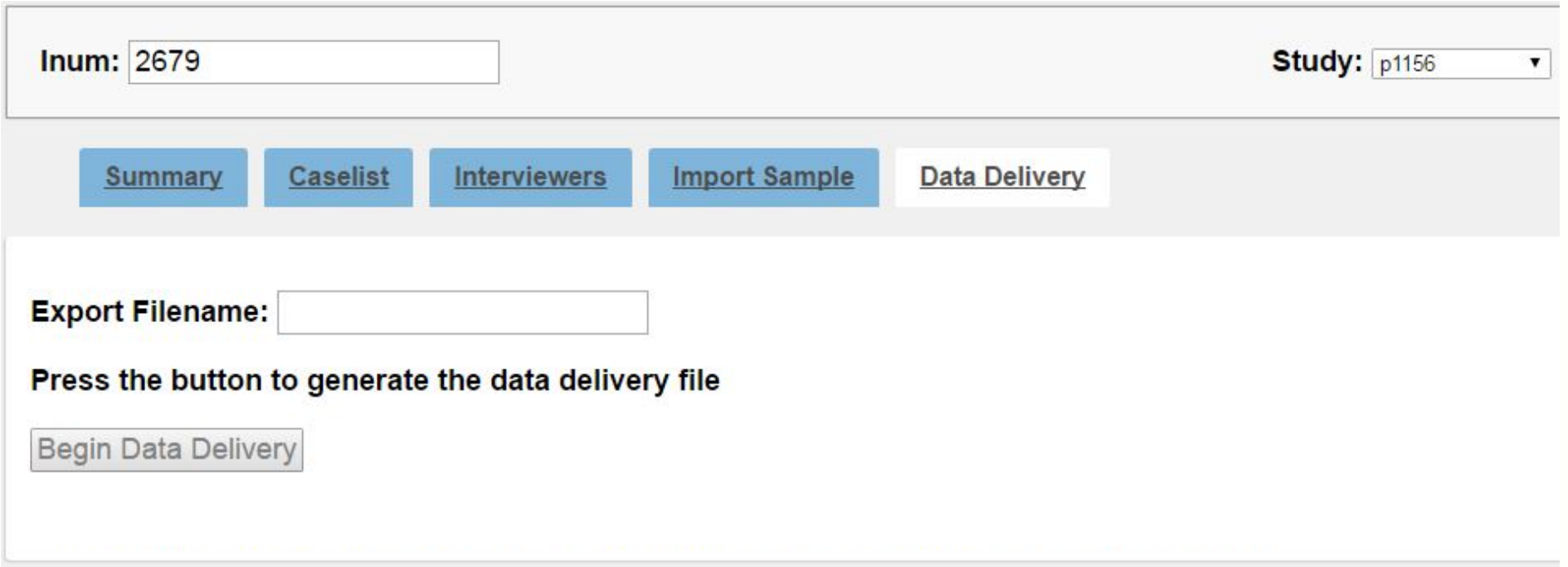
Case ID	Last ID	Accessed	Version	Code	Attempt Success	Type	Stratum	Date/Time
1002011	16	5	1	1100	12	Cig	3	2016-07-03T15:13:03.068Z
1002016	16	6	1	1100	1	Cig	3	2016-06-23T16:02:00.794Z
1002027	16	1	1	1100	1	E-Cig	3	2016-06-23T17:24:44.412Z
1002049	16	1	1	1100	1	Cig	3	2016-06-23T18:04:21.248Z
1002063	16	2	1	1100	1	Cig	3	2016-06-23T16:22:26.594Z

Inum: 2679 Study: p1156

Summary | Caselist | Interviewers | Import Sample | Data Delivery

1. Create a sample.csv file for your project.
2. Copy that file into the c:\Robot\sample\ directory
3. Press the button.

Begin importing sample for p1156



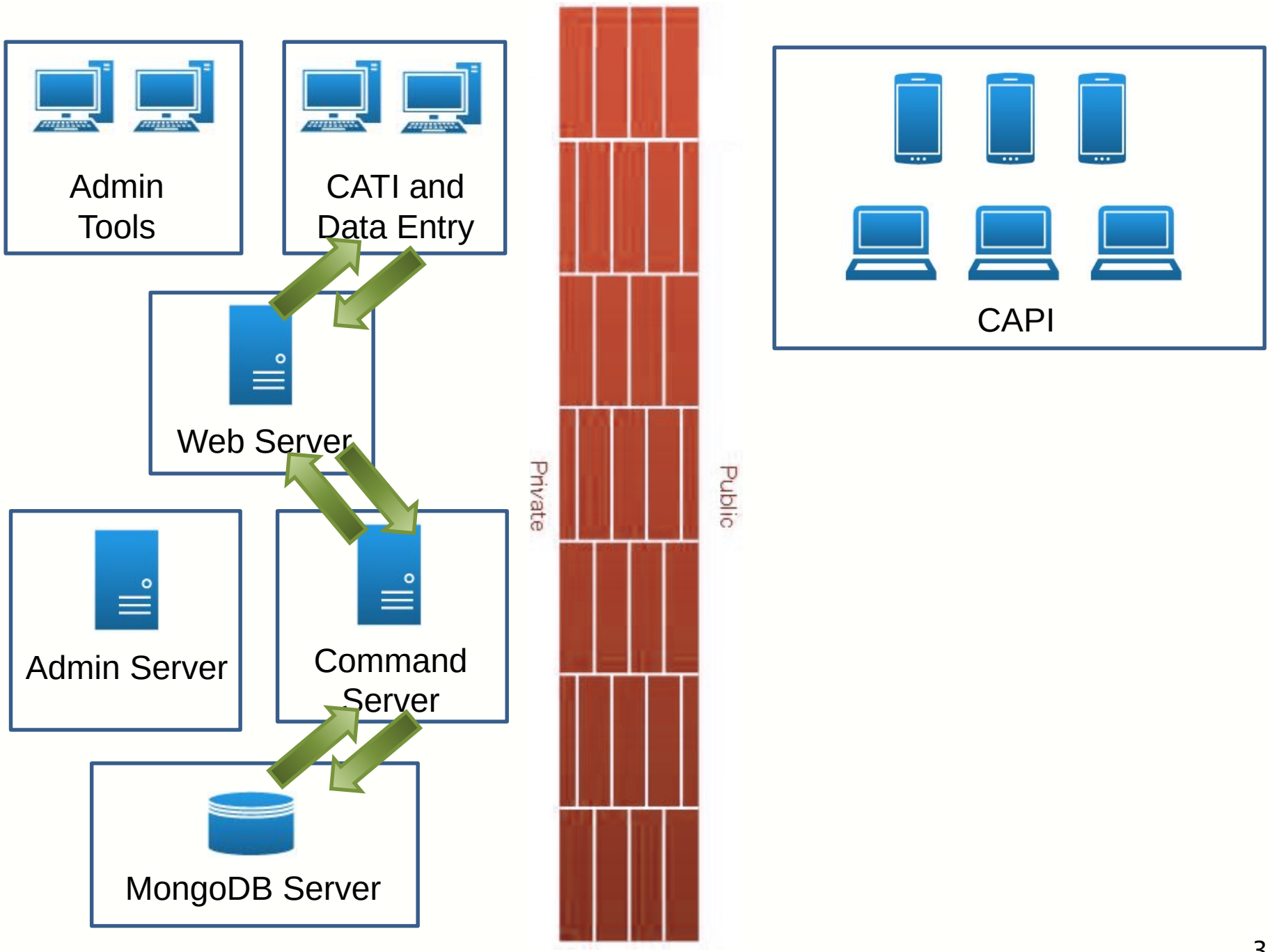
- Current implementation produces
  - CSV file
  - SPSS syntax file

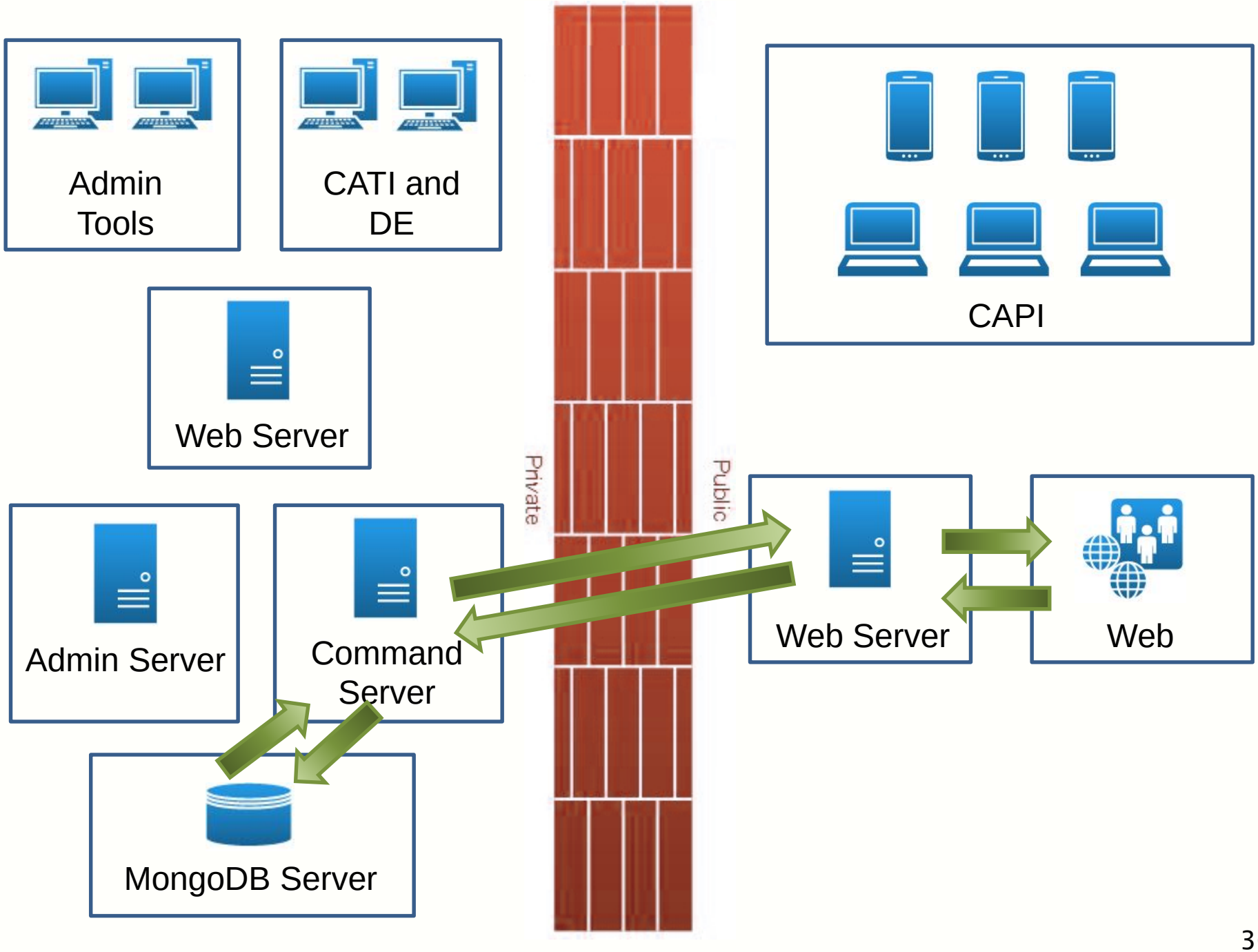
Monocle 3.0 - Future Features

- Login authentication
  - Menu of authorized projects
- Setup screens
  - Interviewers
  - Summary screen
  - Caselist
- Expand data delivery features
  - User-select subset of records and variables
  - Partial data delivery
- Implement current reports

Server-Side Implementation

- Server configuration
  - Data Entry
  - CATI
  - Web
- Framework
- Database
- Survey administrative tools


Admin Tools
CATI and Data Entry
Web Server
Admin Server
Command Server
MongoDB Server
Private
Public
CAPI


Admin Tools
CATI and DE
CAPI
Web Server
Private
Public
Admin Server
Command Server
Web Server
Web
MongoDB Server

Server-Side - Future Features

- Existing processes
  - Integration
    - Rapid implementation
  - Redesign
    - Optimize use of NoSQL
      - Data structure
      - Shard data on multiple machines
    - Examine and implement new efficiencies

Acknowledgements

- Steve Bochte
- Brendan Day
- Jaime Faus
- Kate Krueger
- Marie Nitschke
- Augie Salick
- Eric White

Tech Showcase – Session 10F

Join us for a hands-on demonstration!

Tech Showcase
3:15 pm – 4:30 pm

Thank You!

For copies of this presentation or more information, contact:

Chris Schlapper
cschlapper@ssc.wisc.edu

Please visit us at:
www.uwsc.wisc.edu