/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

How we made data processing scalable at nugg.ad

- young and energetic
- older and a little wiser

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

nugg.ad provides targeting service for online advertising

- socio-demographics (10 in total)
- product interests (48 in total)
- Advertisement clicks
- Frequency
- Geographic

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

online service creates a lot of data

- page clicks
- advertisement clicks
- predictions
- surveys

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

Find a way to use all of this data

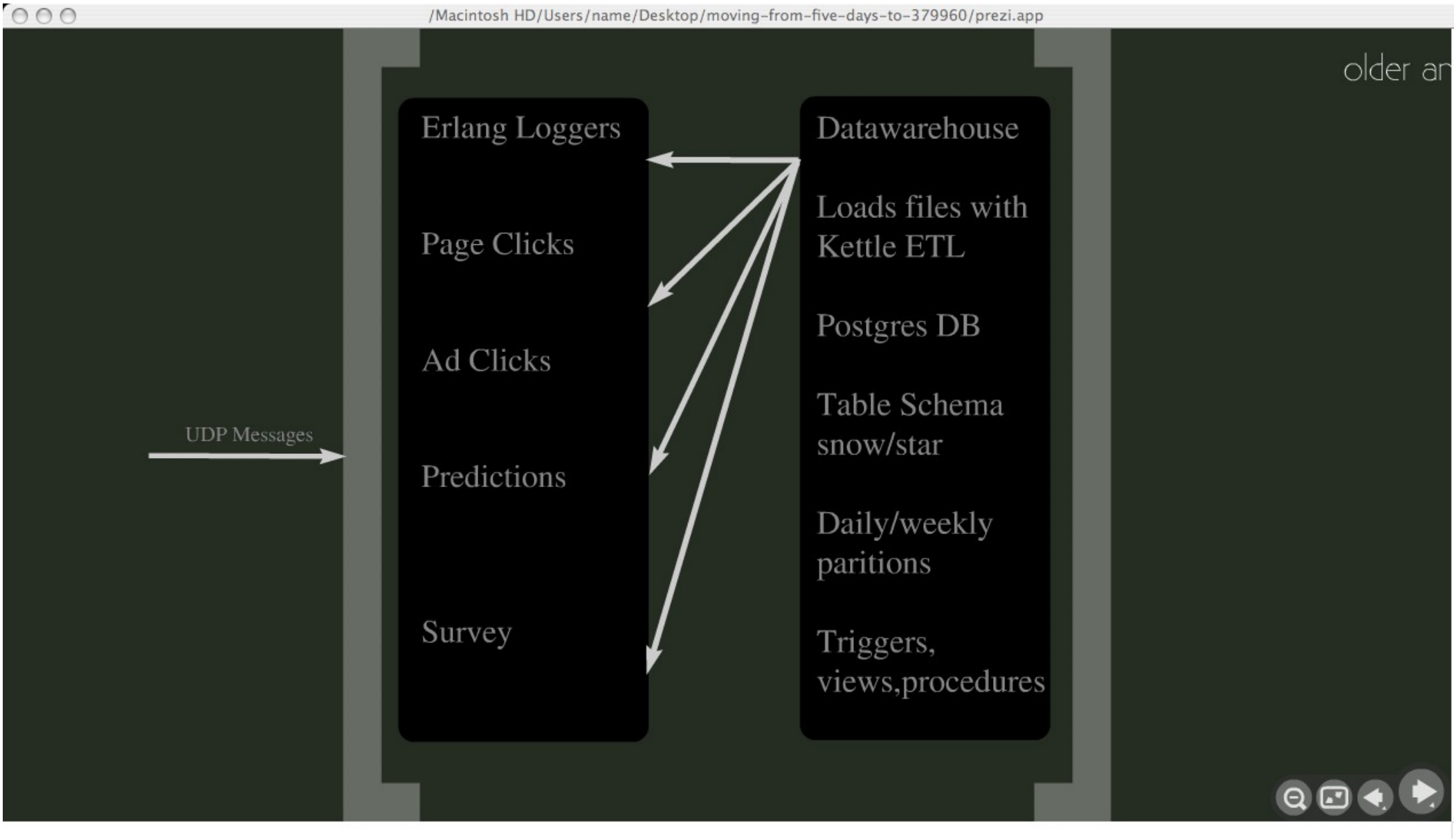
/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

what we need to do

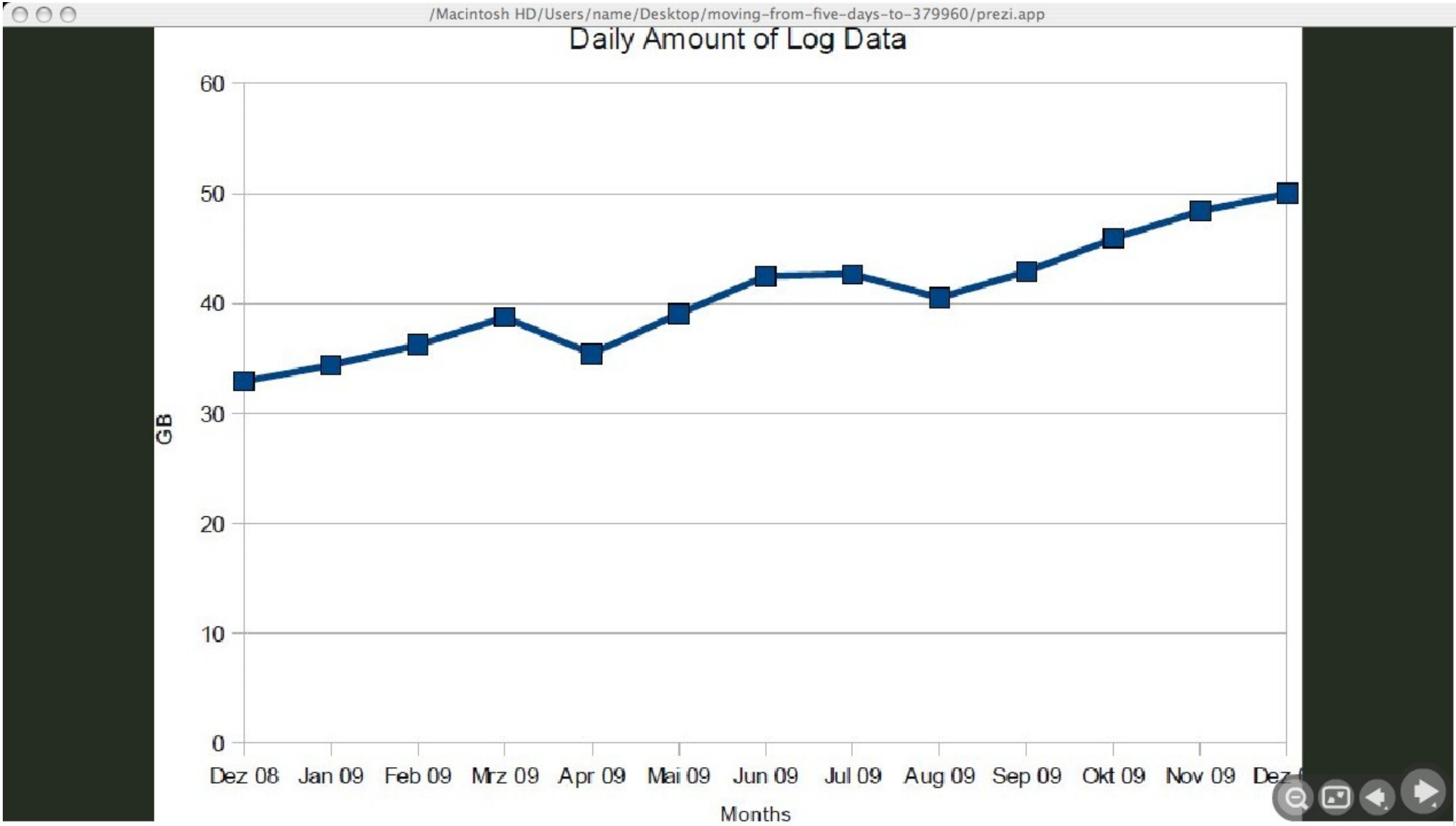
- Aggregate event based logs
- Run analytics
- create training data
- Reports
- create files for db bulk loading

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

young and energetic


/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app
older ar
Erlang Loggers
Page Clicks
Ad Clicks
Predictions
Survey
Datawarehouse
Loads files with Kettle ETL
Postgres DB
Table Schema snow/star
Daily/weekly paritions
Triggers, views,procedures
UDP Messages

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app



/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

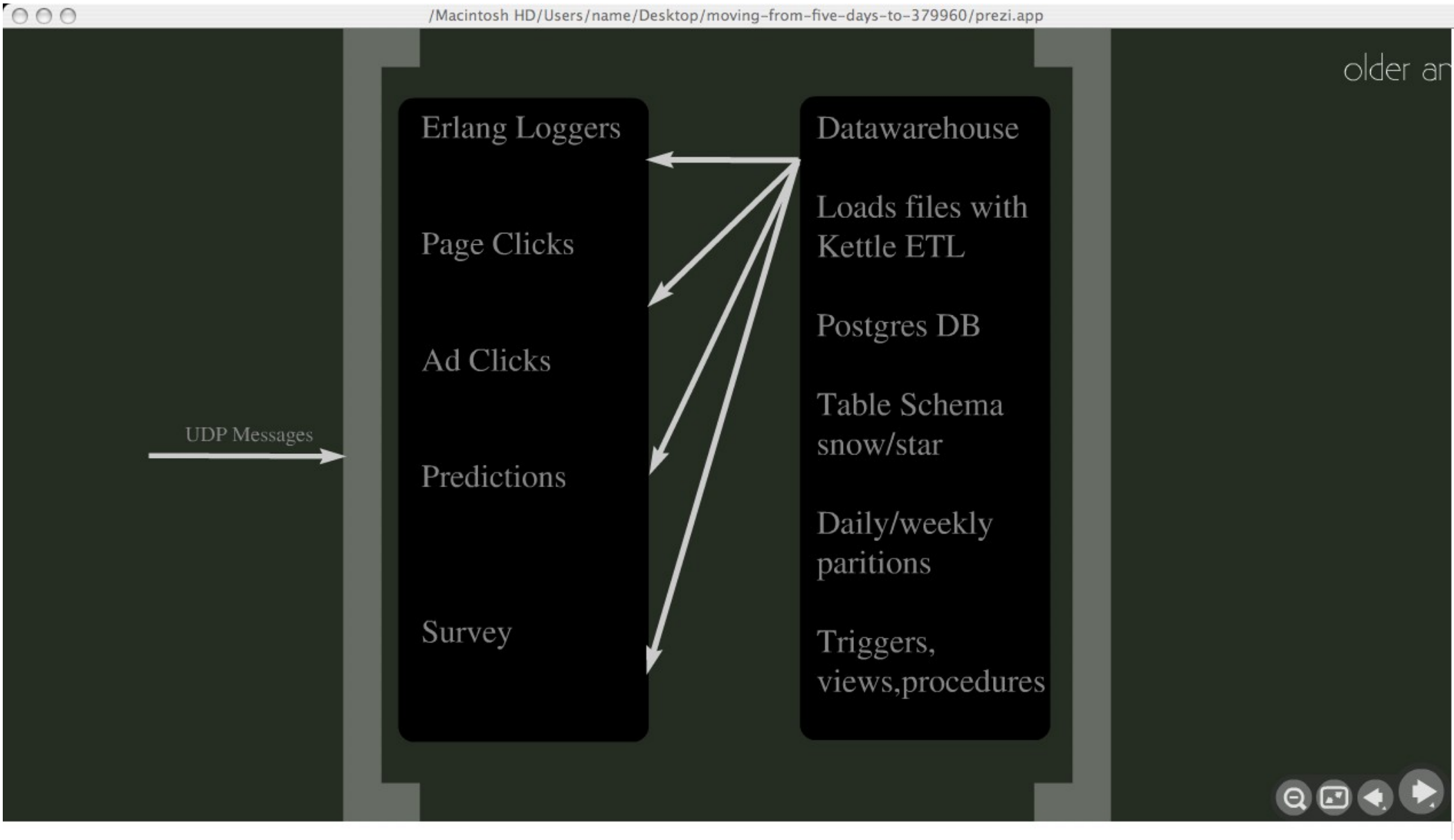
Headache now a migraine

January 2008

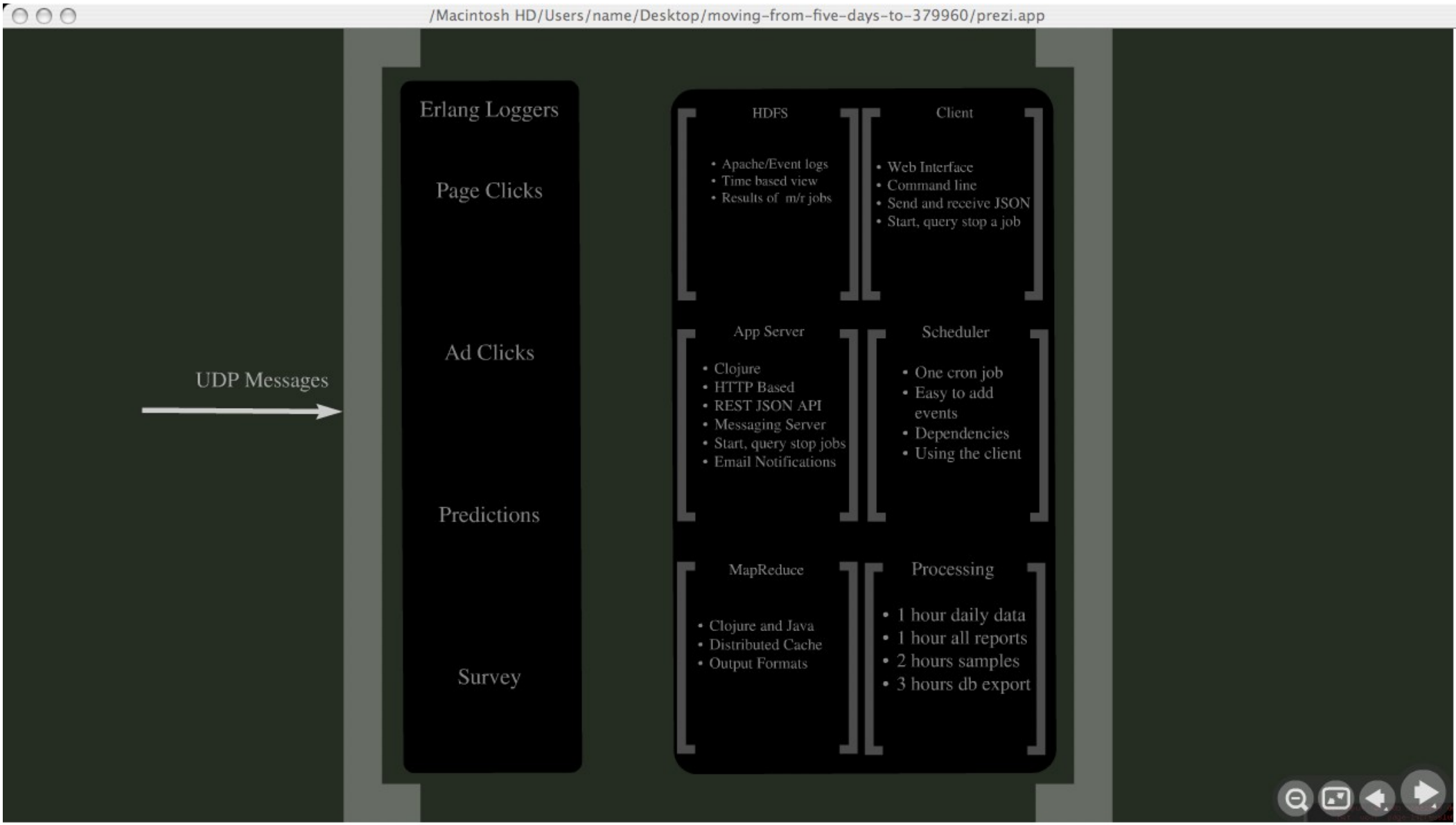
12 hours daily data processing
2 days to create weekly reports
2 days to create training data
2 days to create db export files

January 2009

- 23 hours daily data processing
- 5 days to create weekly reports
- 4 days to create training data
- 5 days to create db export files


/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app
older ar
Erlang Loggers
Page Clicks
Ad Clicks
Predictions
Survey
Datawarehouse
Loads files with Kettle ETL
Postgres DB
Table Schema snow/star
Daily/weekly paritions
Triggers, views,procedures
UDP Messages

older and a little wiser


/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app
UDP Messages
Erlang Loggers
Page Clicks
Ad Clicks
Predictions
Survey
HDFS
• Apache/Event logs
• Time based view
• Results of m/r jobs
Client
• Web Interface
• Command line
• Send and receive JSON
• Start, query stop a job
App Server
• Clojure
• HTTP Based
• REST JSON API
• Messaging Server
• Start, query stop jobs
• Email Notifications
Scheduler
• One cron job
• Easy to add events
• Dependencies
• Using the client
MapReduce
• Clojure and Java
• Distributed Cache
• Output Formats
Processing
• 1 hour daily data
• 1 hour all reports
• 2 hours samples
• 3 hours db export

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

App Server

- Clojure
- HTTP Based
- REST JSON API
- Messaging Server
- Start, query stop jobs
- Email Notifications

So

- On
- Eas
  eve
- De
- Us

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

HDFS

- Apache/Event logs
- Time based view
- Results of m/r jobs

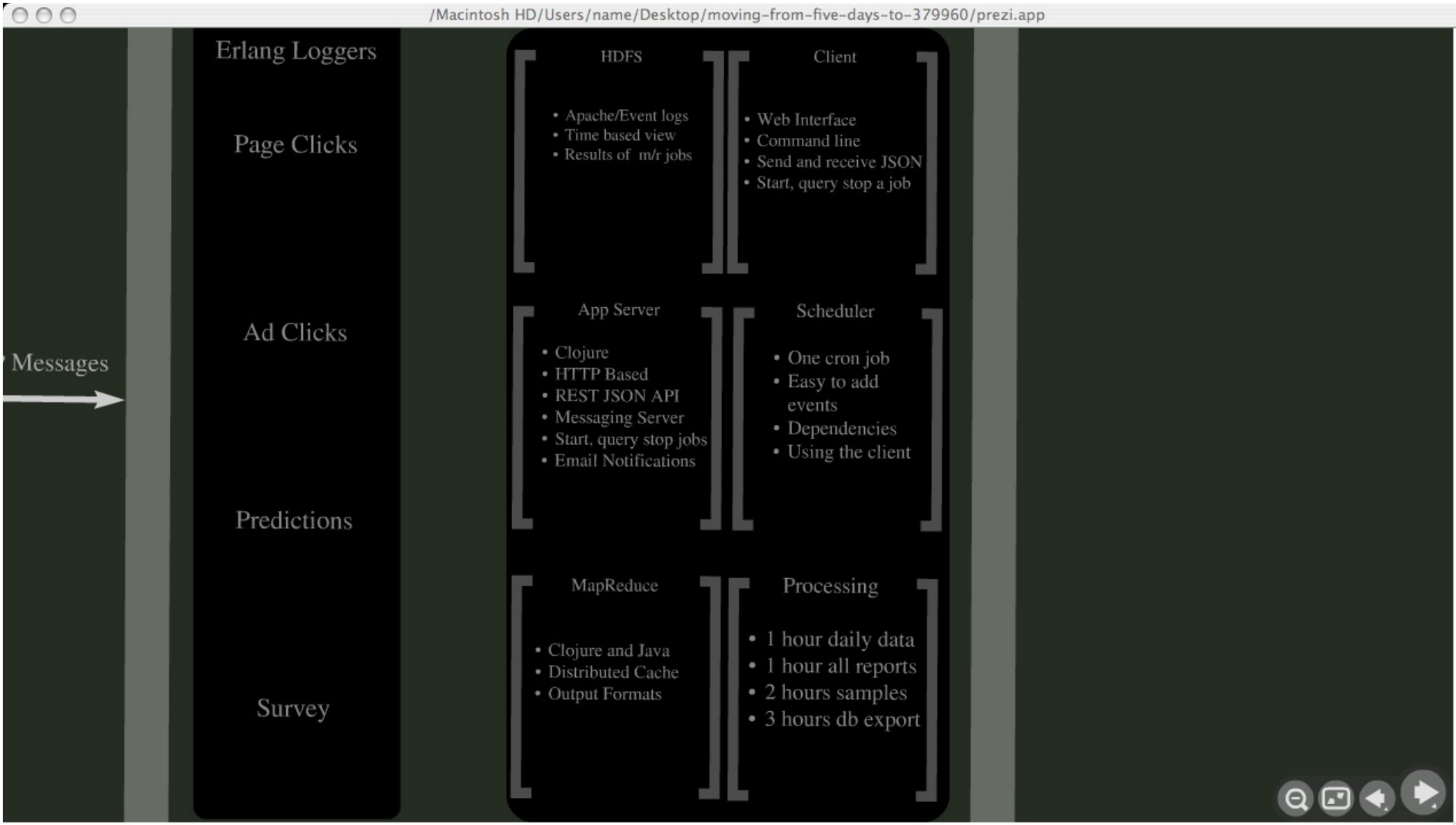
/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

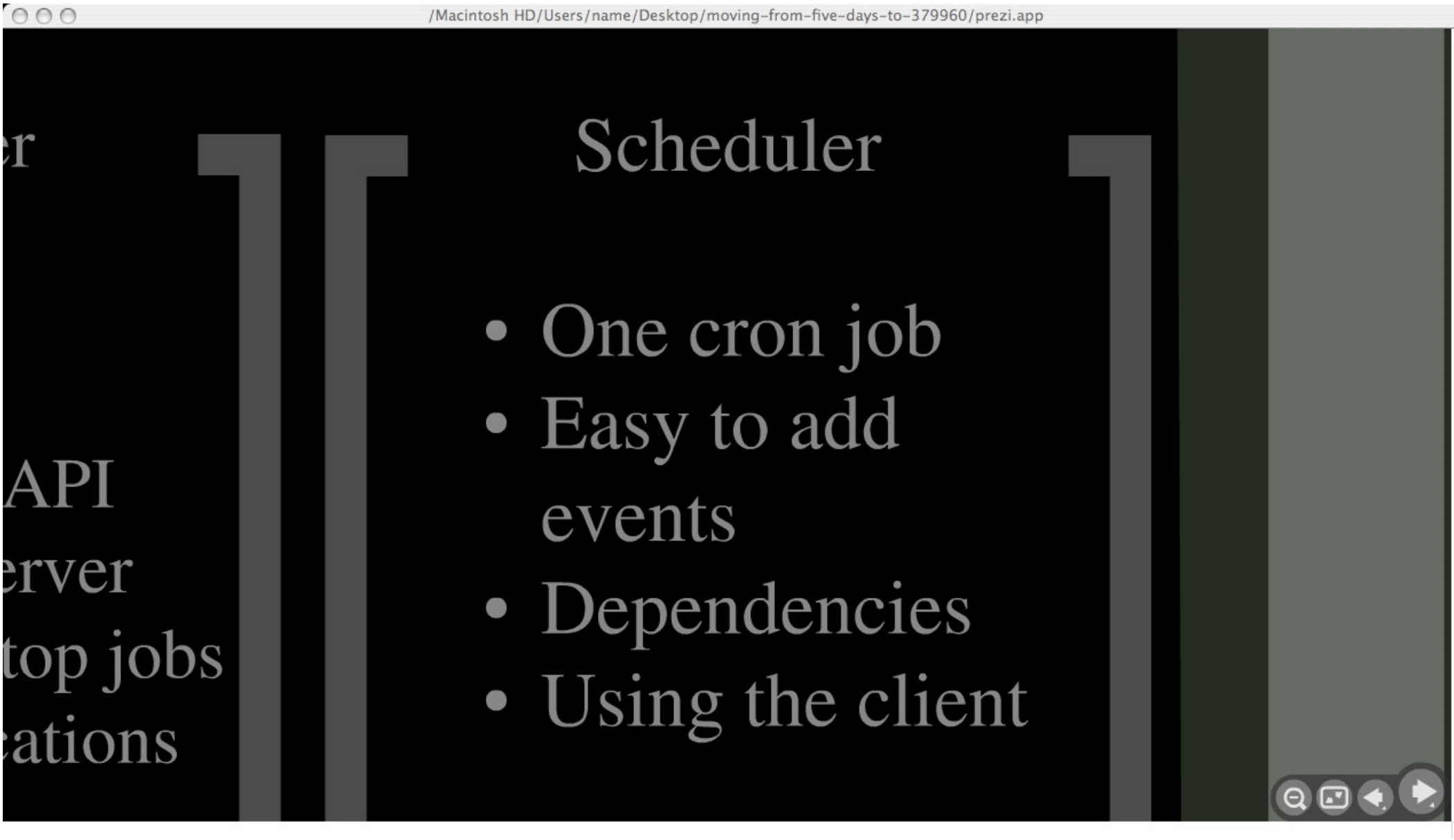
Client

- Web Interface
- Command line
- Send and receive JSON
- Start, query stop a job

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

```
{ "request" :
  {
   "action"   : "days-click-report",
   "input"    : "/aggregate/click/month",
   "output"   : "/report/click/month"
  }
}
```


/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app
Messages
Erlang Loggers
Page Clicks
Ad Clicks
Predictions
Survey
HDFS
• Apache/Event logs
• Time based view
• Results of m/r jobs
Client
• Web Interface
• Command line
• Send and receive JSON
• Start, query stop a job
App Server
• Clojure
• HTTP Based
• REST JSON API
• Messaging Server
• Start, query stop jobs
• Email Notifications
Scheduler
• One cron job
• Easy to add events
• Dependencies
• Using the client
MapReduce
• Clojure and Java
• Distributed Cache
• Output Formats
Processing
• 1 hour daily data
• 1 hour all reports
• 2 hours samples
• 3 hours db export

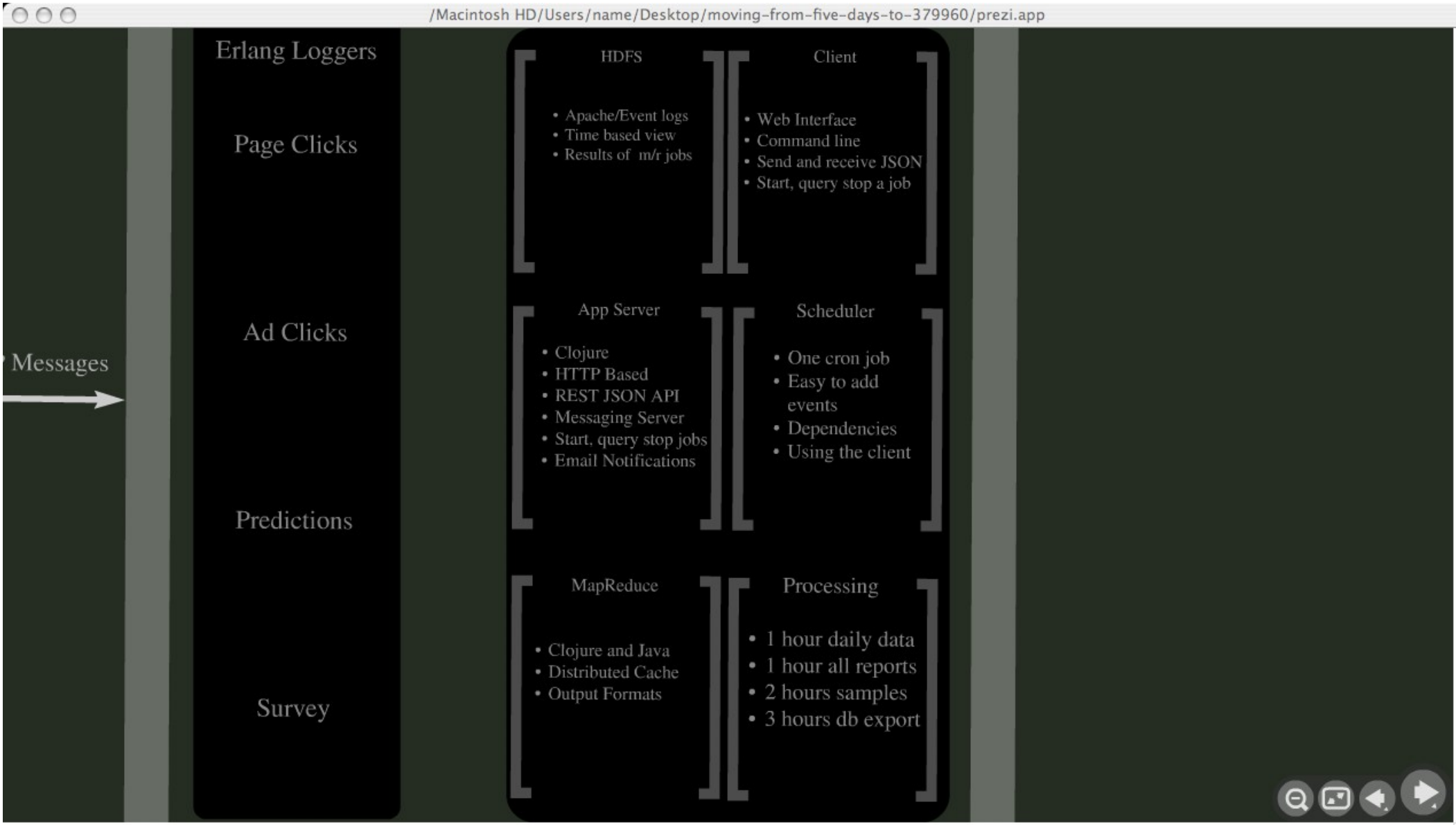

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app
r
API
erver
top jobs
cations
Scheduler
• One cron job
• Easy to add events
• Dependencies
• Using the client

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

```
{"events":
  [
    {
      "name"      :  "click-report",
      "position"  :  1,
      "schedules" :  [{"name": "days", "input": "/aggregate/click/month", "output": "/report/click/month"}]
    }
  ]
}
```

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

```
{ "request" :
  {
   "action"   : "days-click-report",
   "input"    : "/aggregate/click/month",
   "output"   : "/report/click/month"
  }
}
```

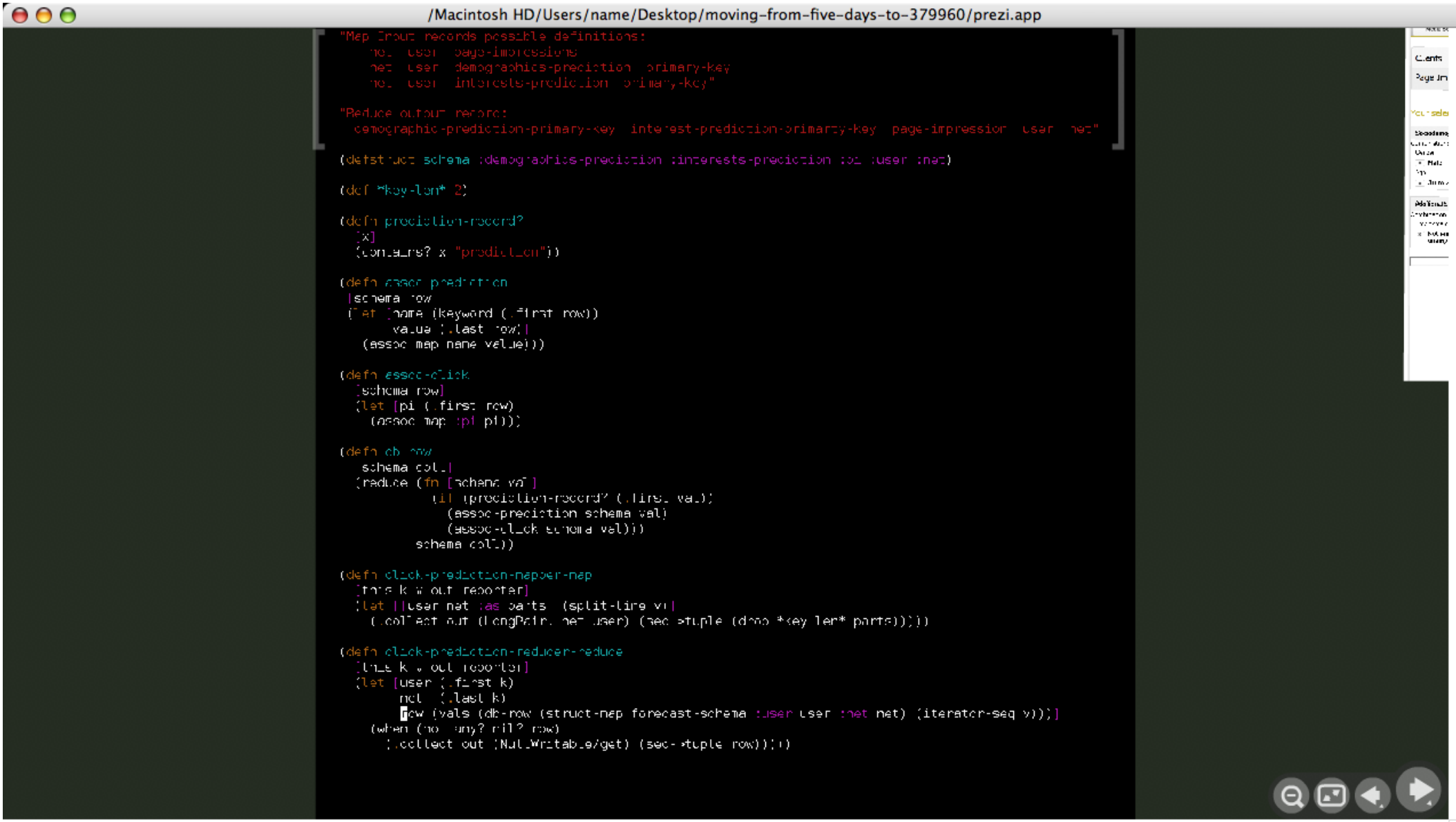

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app
Messages
Erlang Loggers
Page Clicks
Ad Clicks
Predictions
Survey
HDFS
• Apache/Event logs
• Time based view
• Results of m/r jobs
Client
• Web Interface
• Command line
• Send and receive JSON
• Start, query stop a job
App Server
• Clojure
• HTTP Based
• REST JSON API
• Messaging Server
• Start, query stop jobs
• Email Notifications
Scheduler
• One cron job
• Easy to add events
• Dependencies
• Using the client
MapReduce
• Clojure and Java
• Distributed Cache
• Output Formats
Processing
• 1 hour daily data
• 1 hour all reports
• 2 hours samples
• 3 hours db export

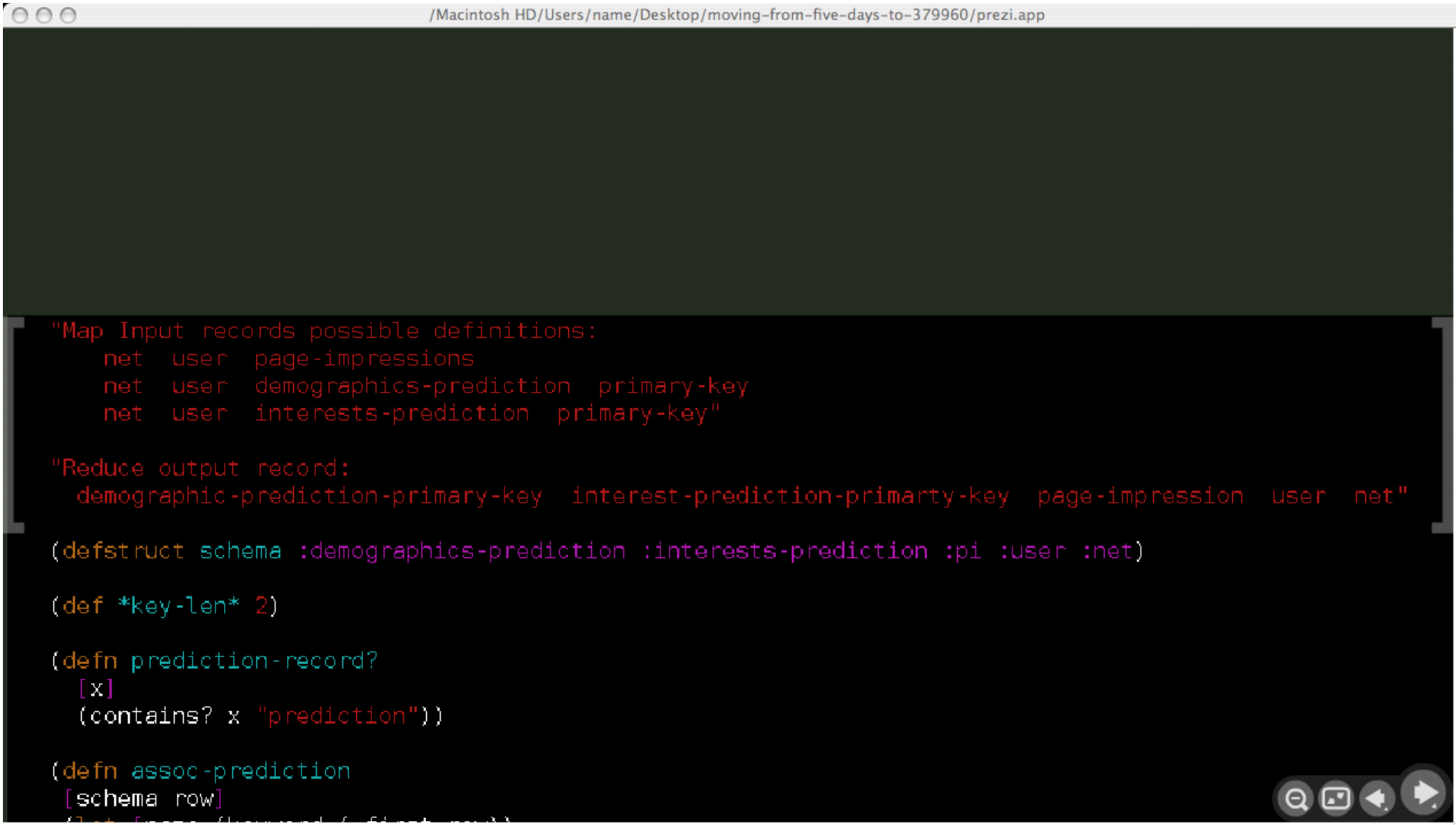
/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

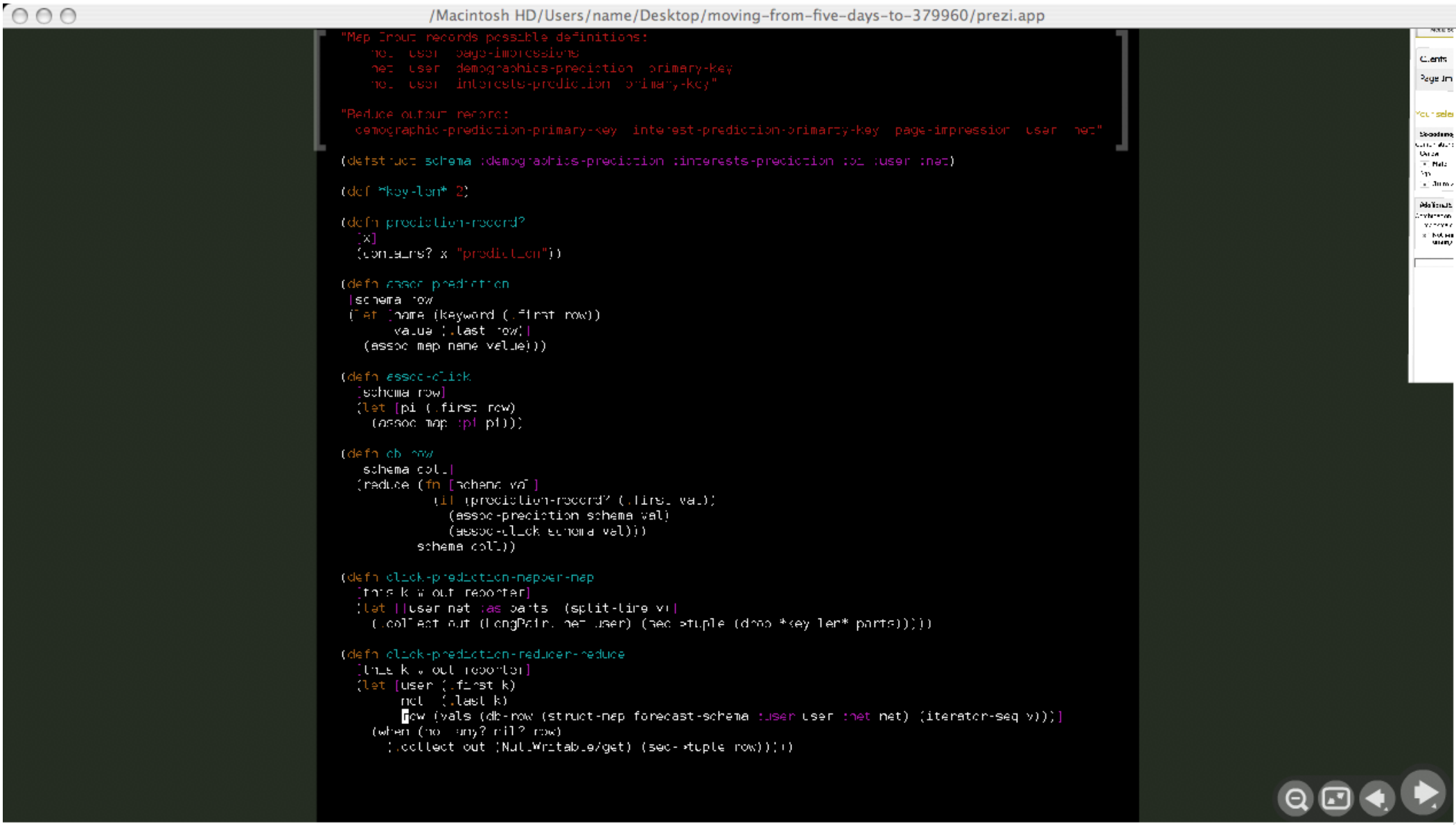
MapReduce

- Clojure and Java
- Distributed Cache
- Output Formats

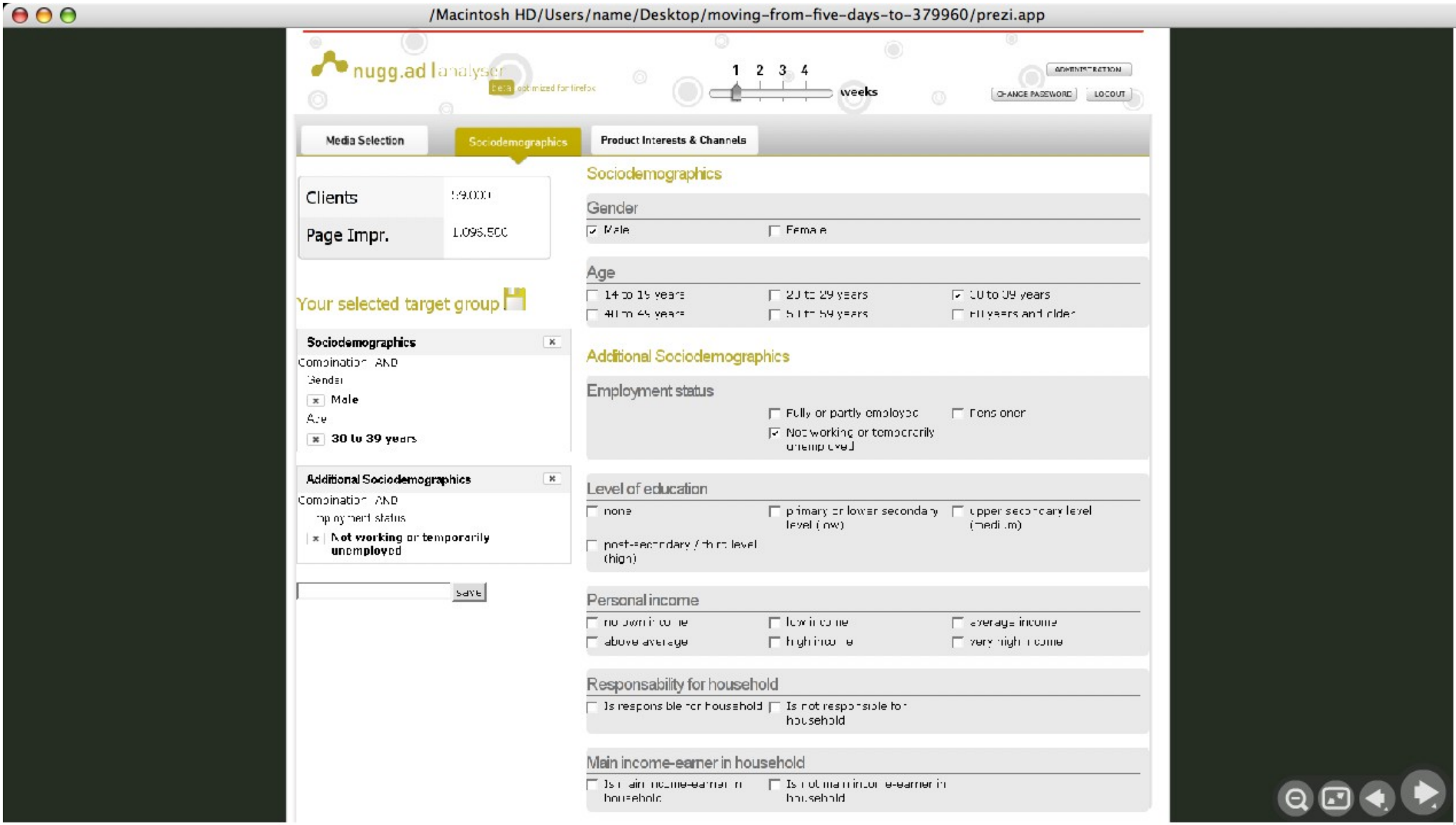
- 1
- 1
- 2
- 3

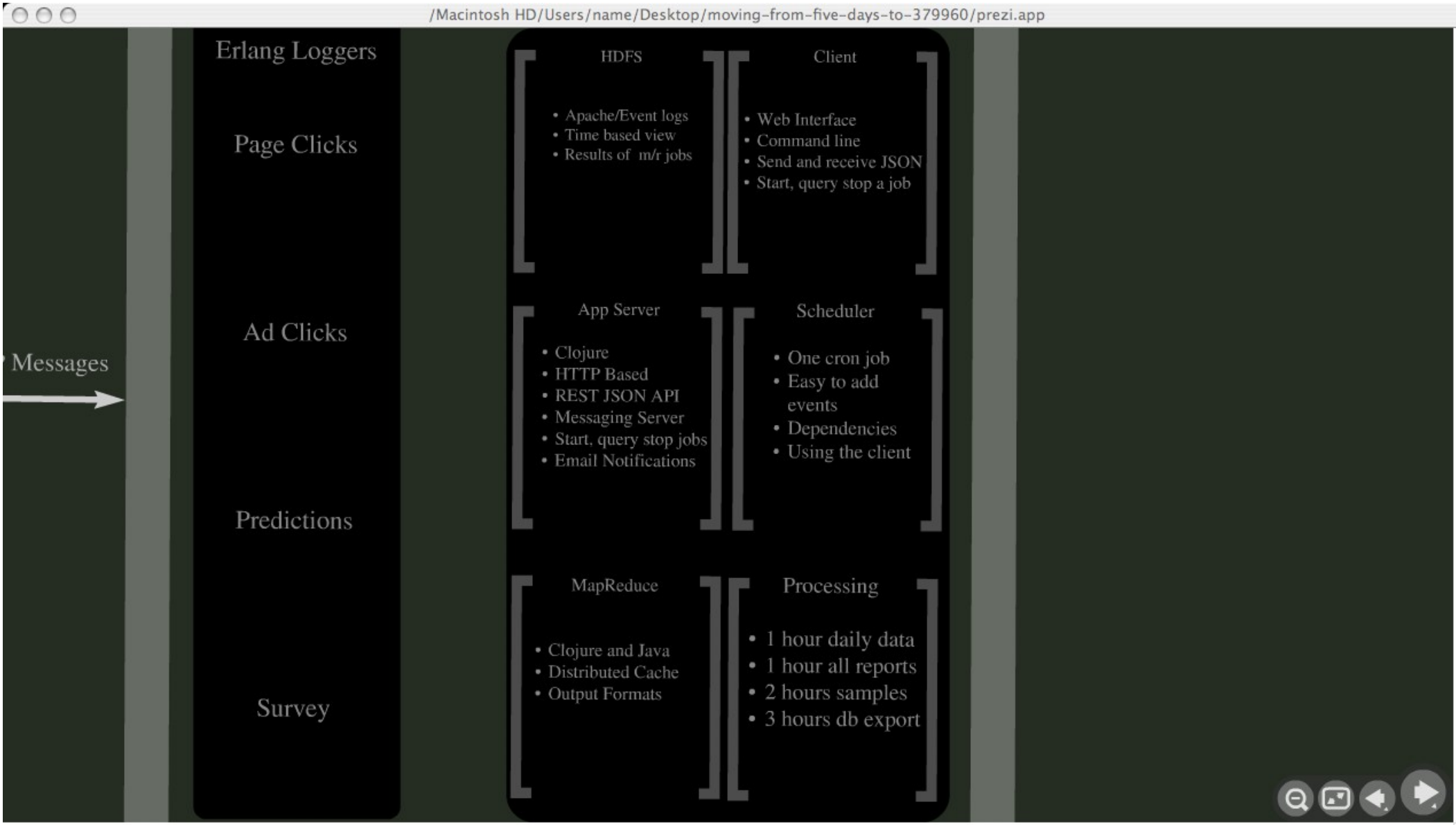

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app
"Map input records possible definitions:
net user demographics-prediction primary-key
net user interests-prediction primary-key"
"Reduce output record:
demographic-prediction-primary-key interest-prediction-primary-key page-impression user net"
(defstruct schema :demographics-prediction :interests-prediction :pi :user :net)
(def *key-len* 2)
(defn prediction-record?
[x]
(.contains? x "prediction"))
(defn assoc-prediction
[schema row
(let [name (keyword (.first row))
value (.last row)]
(assoc map name value)))
(defn assoc-click
[schema row]
(let [pi (.first row)
(assoc map :pi pi)))
(defn db-row
schema coll]
(reduce (fn [schema val]
(if (prediction-record? (.first val))
(assoc-prediction schema val)
(assoc-click schema val)))
schema coll))
(defn click-prediction-mapper-map
[_ k v out reporter]
(let [[user net :as parts (split-line v)]
(.collect out (LongPair. net user) (sec-tuple (drop *key-len* parts)))))
(defn click-prediction-reducer-reduce
[_ k v out reporter]
(let [user (.first k)
net (.last k)
row (vals (db-row (struct-map forecast-schema :user user :net net) (iterator-seq v)))]
(when (not-any? nil? row)
(.collect out (Null_Writable/get) (sec-tuple row)))))


/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app
"Map Input records possible definitions:
net user page-impressions
net user demographics-prediction primary-key
net user interests-prediction primary-key"
"Reduce output record:
demographic-prediction-primary-key interest-prediction-primarty-key page-impression user net"
(defstruct schema :demographics-prediction :interests-prediction :pi :user :net)
(def *key-len* 2)
(defn prediction-record?
[x]
(contains? x "prediction"))
(defn assoc-prediction
[schema row]


/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app
"Map input records possible definitions:
no. user page-impressions
net user demographics-prediction primary-key
no. user interests-prediction primary-key"
"Reduce output record:
demographic-prediction-primary-key interest-prediction-primary-key page-impression user net"
(defstruct schema :demographics-prediction :interests-prediction :pi :user :net)
(def *key-len* 2)
(defn prediction-record?
[x]
(.contains? x "prediction"))
(defn assoc-prediction
[schema row
(let [name (keyword (.first row))
value (.last row)]
(assoc map name value)))
(defn assoc-click
[schema row]
(let [pi (.first row)
(assoc map :pi pi)))
(defn db-row
[schema coll]
(reduce (fn [schema val]
(if (prediction-record? (.first val))
(assoc-prediction schema val)
(assoc-click schema val)))
schema coll))
(defn click-prediction-mapper-map
[_ k v out reporter]
(let [[user net :as parts (split-line v)]
(.collect out (LongPair. net user) (sec-tuple (drop *key-len* parts)))))
(defn click-prediction-reducer-reduce
[_ k v out reporter]
(let [user (.first k)
net (.last k)
row (vals (db-row (struct-map forecast-schema :user user :net net) (iterator-seq v)))]
(when (not-any? nil? row)
(.collect out (Null_Writable/get) (sec-tuple row)))))

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app




/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app
Messages
Erlang Loggers
Page Clicks
Ad Clicks
Predictions
Survey
HDFS
• Apache/Event logs
• Time based view
• Results of m/r jobs
Client
• Web Interface
• Command line
• Send and receive JSON
• Start, query stop a job
App Server
• Clojure
• HTTP Based
• REST JSON API
• Messaging Server
• Start, query stop jobs
• Email Notifications
Scheduler
• One cron job
• Easy to add events
• Dependencies
• Using the client
MapReduce
• Clojure and Java
• Distributed Cache
• Output Formats
Processing
• 1 hour daily data
• 1 hour all reports
• 2 hours samples
• 3 hours db export

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

e

va

che

ts

Processing

- 1 hour daily data
- 1 hour all reports
- 2 hours samples
- 3 hours db export

/Macintosh HD/Users/name/Desktop/moving-from-five-days-to-379960/prezi.app

How we made data processing scalable at nugg.ad

- young and energetic
- older and a little wiser