

12/27

Putting the census apis on hold for now. Deep dive back into etcd which will probably follow back to census and vulcand. However, this deep dive may possibly take a while as I really document and possibly market this knowledge and skill set.

12/20

Finally starting to understand the data apis more in detail at census.gov. These folks actually know what the heck they are doing and have done a pretty nice job.

12/17

Vulcand is now up and running and starting to look at census.gov as a nice data set to use for show casing a new api server based on etcd and vulcand.

12/04

Today is the day I told Michael about the Spn.ee Go project. Plum is up and running, needless to say.

11/21

On my 54th birthday I am close to having the new generic Spn.ee ported over to Go. It is located in a github repository called Plum. This is my first working program written in Go of any substantial amount. This is great news. Based on this log you got this thing up and running in about two weeks which is pretty good. Granted all of the functionality was there so it was just a port job, but still I tip my hat to you on focusing and getting this work done.

11/06

These are my first days of writing real code in golang. Since August 6 when I first installed Go on my computer till now I have been reviewing and understanding code. First with Etcd and then with Docker, all reading

and no writing. Although I did get things up and running and write short println statements here and there, after literally exactly THREE months I am now writing real code. Today I started getting my code talking to Redis. Yesterday I think I figured out a solution to do the JSON schema validation.

11/05

First post from Ashland. We are loving it here. Driving towards parity in go with spneg. Doing all sorts of set up stuff including working more on spneg to align with go spneg which is called plum. For example today I got up and running a suitable SecureRandom version of code in Go. Also, wrote some more code in spneg to mirror what I should see in Go. This is great practice for my first real Go programs. And it gives a clean separation of concerns on a clear project to implement.

10/31

Starting to publish data off to RabbitMQ from a Go client which means the simulators will not have to be re-written now in Go. Instead, I will move on to getting Spnee written in Go using Negroni et al. I am reviewing my options on that front now. On the Spinnakr side of the world we switched all of the JSON document to Strings from Longs as well as returning the calculated data back to the user.

10/25

Spoke to Michael once this week, and it was today, yesterday I spoke to Justin for the first time this week. start working (again) on nsq. the first time I looked at it I wasn't ready, now I am. and its good stuff. and very relevant at this time.

10/24/2014

starting to focus more and more on core go programming, and just becoming a better go programmer by looking at lots more stuff. for the first time today I looked at all of the core packages that come with the

distribution. I didn't even know that all of the core packages ship with the release, but today I found them. This was in the light of starting to look at libchan and spdystream, and then taking a look at the core bytes package. We are now up to 19 packages with over 3000 stars on Github.

10/23

Today I removed a bunch of stuff off my hard disk. I do this once or twice a year. In doing this, I wanted to recap the year in general. Going off to Ashland in about a week, the year is coming to a close. At the beginning of the year I was big into re-learning C. When my job at Spinnakr came to a close on June 10 I moved into learning elasticsearch in the context of Angular.js. I also updated my Craig app to publish data into elasticsearch so it could be viewed by angular. When my job at Spinnakr comes to a close I will get back into using this as my new UI to see the new job postings. For now, all my energy will go into Go until Spinnakr comes to a close. I used Redis, LLVM, and chrome as a model to dive back into C. We will do another recap soon, but for now suffice it to say that the turning point in the year was on August 6. That was the day that I got golang up and running on my machine. So in the end I satisfied my goal of getting into a new language in a big way. Instead of C it turned out to be Go.

10/21

begin work on a new simulator

10/20

broke out linear regression into a test, and added two ways to do standard deviation

10/13

json-schema to validate JSON data coming from the customer

10/12

Starting to look into details of linux features cgroups and namespaces in

parallel with more study of libcontainer.

10/10

Docker up and running on Ubuntu.

10/09

Rolled out first version of the Generic Event System to Amazon. We started working on this system on **August 06**, so here we are about 2 months later with a first cut at a new system. My first paycheck back to work was **August 01**, and on **July 22** I got a healthcare reimbursement for the months of June, July and August.

10/2

Wrote up the new product spec for our customers. Also, spending much more time diving into the details of all components of CoreOS.

10/1

Storm is up and running and generating out events on my local box. Also spent some time looking at systemd for the first time.

9/15

Start reading the Raft paper and dissertation in more detail. This is my first day spending a significant amount of time on the paper and dissertation.

9/12

Got the warden code integrated into Spn.ee. Its working and not breaking anything. Consistently making headway on etcd.

9/3

Communication with Blake and Xiang about etcd. At the end of the day I got it up and running with my way of pulling in gogoprotobuf based on the In magic. They then committed a whole bunch of their changes back to

the master branch of coreos. I learned a lot more today about go and package management.

8/30

Got the new etcd up and running, trick was running it with

```
go run *.go -cors='**'
```

8/29

Spent most of the week working on etcd and now the new etcd with the brand new raft implementation. the etcd team decided to re-implement raft from scratch so that is what I am now looking at.

8/21

More work on Go. Used the bloom filter as an example to learn what packages are all about.

8/20

More work on the simulator to support message types and also broke out the options code into a separate class so that it can be included now in different places for testing.

8/19

Starting to look at hash functions in the context of bloom filters.

8/13

**First week back to work with Spinnakr.** Michael woke up and decided we should do some more work (I guess). So I started up by writing a simulator that publishes messages onto RabbitMQ.

8/10

Starting to dive into the source code of Docker and its associated parts.

8/9

Coreos, VirtualBox and Docker all up and running on my Mac. This follows up to a couple of days ago when I got Golang up and running for the first time. We went to the pow wow in Siletz today.

8/7

RabbitMQ and Bunny are now working. In the previous round of work we used the amqp gem. But now bunny is the place to be. I got a pub/sub system working today along with the top layer of the JSON document on to the channel.

8/6

Golang is up and running on my computer for the first time in my life. Looks like this is the first commit to this [repository](#). So this is when we started working on the Spinnakr Generic Event System.

And it all started [here](#).

8/5

Start back into Kibana, hopefully I will get through writing up the documentation. Once that is complete then we can start to build out an application. Yet another detour as I quickly learn about Flot.

7/31

Day One logstash. Prior to today I had not even looked at logstash. Its a pretty cool product as it is written in Ruby. So I thought I would take it out for a spin and see what its all about.

7/30

Got the ejs tests up and running on the browser side by setting the ulimit higher. The tests were failing because of this. I am now in communication with Spencer Alger which is great news.

7/29

I file my first pull request against elasticsearch-js on the bug in yamldoc with the match function. Spencer acknowledges at the end of the day that indeed it is a bug, this causes a huge refactor by him of the way the yamldoc match function works. He was appreciative of the fact that I pointed out the issue to him and opened up an entree for me to have his work email address.

7/24

making some really good progress now with grunt, and a better understanding of the mocha integration tests. starting to break out the tests from one big generated file into multiple smaller files so I can get a better overall handle on the API.

7/21

working on spn.ee to see what is up with our application. got it working part one.

7/17

Went to Kp today to meet for the first time Caroline King-Widdall. Also, Peter's parents came over after dinner. I got grunt up and running today and am starting to review a testing scenario that runs all of the tests without doing a download of the product. It would be great to just bring the product up and be able to run the tests.

7/16

Dad's birthday. Today was the first day working on the Angular Testing platform. Prior to today I had not even looked at testing for Angular. Looks like Jasmine from Pivotal Labs is the basis for Angular testing. Not clear that now I need to do end to end testing.

7/12/2014

Scott's birthday. Today I spent the whole day looking at and understanding asciidoc along with deploying a working copy of a book to

github's gh-pages. At the end of the day, asciidoc is the way to go. Once again, the folks at elasticsearch got it right.

7/10/2014

So here we are. Last week I got a call from Michael saying I was back on board, one day later I spoke to Adam with Michael on the line, more better. I took the rest of the week off. July 4, came and went and I went to Portland Sunday to hang out with Mike and Martin. This week no call from Michael, so what else is new.

In any event, I am becoming increasingly more IMPRESSED with **Angular**, it is the next step in UI evolution for the browser following in the steps of backbone. And as a side note, lodash is the new frontrunner to replace underscore. On April 14, 2012 John Dalton merged underscore into lodash and never looked back. I discovered that today. I found it because Kibana is using lodash and dug a little deeper. [Here](#) is his official response to the question. In the summer of 2012 underscore was the rage and after going to work for Spinnaker at the end of July I never really got the hit that lodash was the new kid on the block. Today I found out for sure.

Today I also further researched flot as Kibana is using this as its graphics package. It looks pretty rock solid and well maintained by a guy at Google whose name is David Schnur. He has been maintaining this package for several years. It looks like it is good to go for further development and I think using this as your core graphics package, at least in terms of kibana clones will probably work out pretty well. I had taken several days off from looking at Angular and Kibana and concentrated on the gmail authentication earlier in the week along with more work on understanding elasticsearch. But coming back into Angular after taking the July 4 holiday off I am even more impressed with it. And having a KILLER example like Kibana to really understand and sink my teeth into Angular is huge.



7/8/2014

Starting to work on OAuth2.0 for gmail api.

7/2/2014

I am now publishing the JSON data that used to land in Redis, into ElasticSearch via the

<https://github.com/sksamuel/elasticsearch-river-redis>

Once its in ElasticSearch I can bring up Kibana and see the data there. I can also click on one of those pieces of content which will bring up a more detailed screen from which I can click on the URL and look at the individual job.

What I have done here is used a generic Kibana / ES UI to REPLACE my custom Backbone / Marionette UI which I completed prior to working at Spinnakr.

This is great news !

I can call Part ONE of this job done.

Way cool....

Talking to Michael today about possible next steps.

No clue what they will be.

Since June 10 I have done this work and gotten to this point.

6/27/2014

Narrowing down more of what my plan is for a job search, along with beginning to write the Craig data out to Elasticsearch. It will be my NEW UI for viewing Craig data, as opposed to the old UI before based on backbone and marionette. The new UI will be based on Angular.

6/25/2014

Day 1 of full out scale looking into Kibana as a core platform from which to move forward. This seems kind of huge to me. Especially from the point of view of learning Angular and subsequently more ES.

6/22/2014

Three days after my first look at angular I made my first commit on Github to an angular project called [AngularJSFeedReader](#).

6/19/2014

Day 1 of Angular.js. It turned out to be a big day as at the end of the day I found Kibana which is an Angular app. Also, because elasticsearch is using Angular I decided to dive into it more. It appears that this may have been the right call. Plus, on Github it is in the top 5 all time repositories by stars and forks, above Backbone.

6/17/2014

Elasticsearch is turning out to be a huge win in many areas. Yesterday I uncovered more how the REST API works and also began looking at Grunt and the node client.

6/13/2014

I uninstalled jdk6 and installed jdk8 on my local macbook today. Then I went on to install and get up and running elasticsearch. Then did a detailed dive into Lucene. I now understand how to build and index and then search across that index with a search term. Its amazing how you can build a company out of such a simple concept. But even more interesting and encouraging is how Google built a business out of that concept. And their follow on businesses were even more brilliant.

6/12/2014

Spent pretty much a full day looking at crypto currency stuff especially bitcoin-ruby. I am understanding the blockchain better.

6/10/2014

Today was my last day of work at Spinnakr. We found out this afternoon at around 4PM.

4/30/2014

Spending more time on LLVM and looking at rubinius and terra

4/17/2014

Starting to dive more into learning C++ with Chrome's gn as my first program to master. Will give me more familiarity with not only the chrome build but also how base works and interacts with this code.

4/4/2014

First update from Oregon ! We are so happy to be here. We have now been here almost two weeks having arrived Monday March 24. So far, I had to deal with the chrome build issue where I was using a faulty version of Python, switching back to the default Python on the Mac fixed my problem.

2/27/2014

Haven't updated in awhile. So I have been working on Google chrome since arriving in Florida. We got DynamoDb up and running with a nice Eventmachine client that is able to quickly do Bulk writes. Yesterday was my first day working with Cassandra.

One of the strategies and goals for this year is to come back up to speed with the C Programming Language and the C++ gets to go along for the ride since there is so much nice code sitting there in the Chrome project.

1/5/2014

First day looking into cryptography via openssl. Then moved on to some other stuff.

1/1/2014

Start off the year on a good note working on khmer and redis and spending more time learning C and C++.

## 2013

12/20/2013

The key point is to learn C and C++ really well. This is important. This is the root language of the computer, and probably will be for the rest of my life. All the other languages are derived from it, and all operating systems are written in it. I have three (3) current projects related to learning C and C++. They are slowly coming up to speed on Linux, which will continue to be in the background. Come up to speed on the source code of Redis which is dramatically in the foreground and take a look at the Khmer project in reference to Hash functions, bloom filters, and peripherally the DNA sequencing problem.

11/16/2013

Got a Makefile to build some redis files. Also, back into Backbone with our Spinnakr app focused on Visitor Loyalty.

11/14/2013

Starting to explore Redis source code in a big way, for the first time. As I told Scott in a recent email I have not looked at or written any C code since prior to doing Java at SFI. Meaning sometime before March 1995 when Java was released. The motivation to get back into C came from my initial wanderings through Khmer and the need to better understand hash tables, bloom filters, and now hyperloglog. All of this is related to storing and retrieving very large sets of data.

11/13/2013

Nearing completion of rolling out the visitor loyalty recommender in concert with tweaking Marek's UI. I had not written a lick of Javascript probably all year until today when I did a small piece of code based on underscore.js. Very cool stuff.

10/25/2013

Backup problem turned out to be a pilot error on my part because I was tarring together only file names that ended in -a\* instead of -a\* AND -b\*. However, I did change the compression method to native postgresql which will avoid one extra step of decompressing the files on the recover side without having to uncompress with bzip2 or gzip. Also, checked in V0 of the Loyalty Recommender. I am now able to talk to the Database remotely using a HTTP Post and Update. This is great news, and will allow us to create Recs in the Db without actually having to be on the machine. This is fairly clean way to talk to the Db.

10/17/2013

Round one of funnel src dst complete, spent the day reviewing how to bring up Spinnakr. It has been months since I did this, along with Spn.ee. Also, back into looking at Marek's Feed / Backbone code.

9/10/2013

On 8/12, my last update was the day we sent the offer off on the house, mas o menos, and now here I am finally updating all of the work I have done since then. It has been a major whirlwind, and finally I am able to say that now things are working but we need lots of optimization as well as a hybrid regex word / phrase combo.

8/12

Start working on the redis interface to nltk.

8/9

Work on manually building out login and signup feature sets and getting them to run and train/test.

8/6

Code reorg of the ml piece. Michael and I had a good talk privately on

driving forward with picking sign in or log in pages. So its similar to spam email.

8/5

start back into machine learning

8/4

Finished up the Visitor Loyalty and Visit Redis Db clean up and modified the API sheet accordingly. Back to machine learning manana.

8/3

Day 1, Category Theory and first introduction to David I. Spivak.

7/29

Taking a break from Machine Learning to start expiring keys from Visitor Loyalty.

7/23

FreqDist working with live data along with filtering out unwanted cruft.

7/18

Start grabbing URL's from Redis and writing HTML pages via Nokogiri using the MIFunnel interface

7/17

Writing URL's from the action table to redis

7/11

Start looking into getting NaiveBayes and the NLTK algorithms to get data from Redis instead of the filesystem.

7/10

Start persisting web page data from Spn.ee to Redis. After Marek's help

yesterday I am up and running and successfully grabbing data.

July 8

Arrived in Boise last Wednesday July 3 and its Monday morning. Spent the past week on the road and in Boise working out how the NaiveBayes algo works. Ready to start looking at grabbing live data.

6/26

Just starting to be able to read data from S3 after getting the write stuff to work yesterday. All in the context of the backup gem.

6/21

Begin work on Fog and writing customer website data to S3. We nixed Dynamodb because it was too expensive and limited to 64K boundaries.

6/14

in the morning I briefly took a look at Java as Michael requested that as a possible language implementation. the rest of the day I reviewed Python. This was my first day getting back into Python.

6/13

visitor loyalty appears to be done for now, and deployed it off this morning. in the afternoon I started thinking about using **Python** as the language of choice for machine learning. I was comparing it to ml as a service. Also on this day I spoke to Scott who said Java is a dead language.

6/12

**machine learning begin.** this is the start of the machine learning project at Spinnakr.

6/6

For the first time in my life I understand how the Amazon VPC works with

the two subnets. The private subnet must use a NAT instance to talk to the outside world.

5/31

Start working on weekly visitor loyalty four categories

5/29/2013

Rspec tests are up and running for the first time ever.

5/25

On Saturday morning I finished Visitor Loyalty and checked it into the code base and Marek took over and made some minor tweaks.

5/23

Working on the new Redis Loyalty database in coordination with the matching algorithm.

5/21

Started working on the new loyalty database with just the visitorid as the key.

5/17

The UI is up and running, along with displaying data from both the recommendation feed and the sped feeds. Total code review of both the latest Backbone infrastructure along with the breakout of Marek's charting package.

5/14

Got everything working. So I am now matching search terms and displaying the proper message.

5/10

On May 8 Michael assigned me a new task having to deal with Spnee.rb.



So yesterday I worked all day on getting my whole infrastructure setup to start testing it today (Friday).

5/4

On Saturday morning after arriving home from Denver last evening I switched the IP addresses on both the action and visit code queues. Search is remaining the same. The action is now production on the VPC reading the production database on a 12 hour cycle and the visit is now live with the new algorithm substitution for zunionstore.

4/29

After a long day, the zunionstore algorithm was resolved with a new algorithm that gave the exact same results. Sometimes you just have to relook at a problem --- instead of banging it over the head.

4/23

The RedisVisitMove data migration is up and running. I am now able to transfer data from the larger Redis database, create a compressed format of just visitor IDs and dates and then map that onto the VisitorId Trackable Id mapping.

4/19

Start working on creating one key that has all of the visitor IDs and associated daily visit numbers.

4/18

Redis backup deployed across all servers is now up and running.

4/17

Continue work on a Redis Backup script based on the awesome backup gem. In the latter part of the day I finally got crontab up and running. Getting the magic formula for that was a bit of a pain.

4/16

Action is up and running on the VPC. The Sped server is up and running and serving up the API on port 4567.

4/11

Wrap up probably the final static version of action.

4/10

Pgbouncer is up and running. The key point was having Michael give me root access to Amazon and then we jumped forward. Raghav topped off the finish line after I changed the security groups.

4/9

Spent the past two days working on a better understanding of Postgresql and Pgbouncer.

4/6

Daily Search Term API up and running reading data from the queue instead of the local database.

4/3

Back to work on Monday and by Wednesday in the evening after taking Lalo for a walk and having some dinner around 9PM I finished the new Sped *dailyvisits* API call.

3/22

Had a call with Michael at Noon.

3/21

Chef up and running on Amazon

3/20

Start looking at Erlang for the first time.

3/19

AMQP ruby gem is up and running and the code is staying up and receiving messages off the queue each time RabbitMQ is brought up and down.

3/17

Read the 0.9.1 AMQP spec versus the 1.0 spec and saw the difference between the two. The 1.0 spec is now over a 100 pages and basically just a wire protocol. Whereas the earlier spec not part of Oasis was a simple wire protocol plus functional spec on top of it.

3/16

Its Saturday morning and I am starting to make real good progress on amqp error handling and recovery. For the past 2 or 3 days I have been struggling to figure it out. The light finally came on this glorious Saturday morning.

3/13/2013

Almost a complete understanding of Marek's Feed and how he melds Erb and Backbone.

3/11

Made lots of progress on better understanding RabbitMQ including getting the web ui up and running on my local machine.

3/7

up at 3am and worked till morning. there was a bug in my redis code and I ended up deploying a medium amazon instance for the first time. total revamp of spne code. everything is now broken out into their own separate directories. also some class renames to delineate better the visit code from the less used visitor code. at the end of the day after a nice long walk with John and after a nice chicken dinner I figured out how

to strip out a forked git repository and knock it down to just a README.

3/6

zunionstore in Redis is now working for Visitor Loyalty across X days.

2/27

Snapshot push to Spinnakr, everything is working.

2/26

The word cloud is up and running in spinmar.

2/25

Got a working version of all of the trackableids on one page so that you can check out the Sped API.

2/21

Jason Davies Word Cloud is up and running.

2/19

First day working on search recommendations based on Michael's sample code.

2/15

Visitors and Aggregate up and running live writing to Redis from RabbitMQ.

02/13

I am now able to control the number of messages I receive on an AMQP subscriber and count the number of messages I receive as well. All in all, I spent the whole day better understanding how AMQP works.

02/09

Found a bug where the Welcoming and WelcomeBars were not being set

properly and passed back.

02/08

The Actions refactor code is starting to take shape and work.

02/07

Start working on RedisAggregate.

02/06

Got the Redis Pub/Sub up and running to fire off the Visitor postprocess.

02/03

Watch more videos on Storm. Michael notes that we are “doubling down” on storm in one of his replies to me.

02/02

First day looking at Storm in concert with RabbitMQ. At 5AM on Saturday morning I listened to Nathan Marz give his talk on Storm.

01/31

Roll updated Backbone code to Spinnakr site. Start working on rolling out Redis to Amazon with RabbitMQ.

01/30

Upgraded spinmar to Backbone 0.9.10 and Marionette v1.0.0-rc4

01/29

I am now showing a no data message when there is no data. also, the menu bars are showing the active tab. this was a bit of a pain, as I had to write the active entry at the end instead of at the beginning because of handlebars re-writing everything over on each request.

01/27

I have not worked on the UI side of the code called spinmar since Jan 07. I am now starting to work on dealing with error processing when no data is available.

01/22

Remove old Spinnakr code prior to backbone deployment in early December. Also --- got the site id up and running in show.html

01/21

Start to come up to speed on the details of So.js and Spnee.rb in the context of understanding Justin's new code with visitor\_id and SecureRandom

01/20

Blow away all of the old amazon instances I was no longer using. Michael fixed the database so that Rake db:seed works. Find Faraday as a testing platform and Justin starts to take a look at it.

01/19

Drive up to South Lake Tahoe with Adam and Marek.

01/18

We are now writing data to Redis via Rabbit MQ's pub/sub. This is very cool. I got a nice test publisher up and running that exercises the different scenarios.

01/15

Postgresql database up and running on Amazon. Also, all new Spn.ee data models rolled out to redis code base.

01/14

Start working on RabbitMQ round two.

01/07

It is Monday and I am done with work ! No more work until I arrive in Ca. next Monday. I finished up the code today and integrated it into the Spinnakr web site.

01/06

Wrote the bubble simulator and tidied up a bunch of code in anticipation for tomorrow's (hopefully) final push.

01/04

Things are pretty much wrapped up for Ca. I have a couple of minor things left to do but for the most part I am good to go. Today I got the bar, detail, and calendar code all up and running with the new Redis API which I also implemented today.

01/03

Handlebars.js menu bar up and running in Spinmar. Final dev talk with Michael prior to going out to Ca. Michael is on vacation next week.

01/02

Rickshaw simulators and Calendar simulator completed with the new Redis API next.

01/01/2013

Lots of progress has been made. We decided NOT to go with Miso for now with rickshaw graphs since the data model is REALLY not just rows in a table and instead we are going with a pure **Moment.js** solution to filter out the week starting dates. Also, the Richshaw data simulator is now working and generating out the same JSON data structure as the server side redisaggregate. This was actually a minor big breakthrough just to get this correct on the Javascript side.

## 2012

12/27/2012

At 3AM I started working on getting jsoncache into the product. Also, spent most of the day working on getting namespacing correct. Also, I now have a framework in place for running the same file in both node.js and inside the browser modeled after the way backbone does it.

12/25

First day working on moment.js. Its going to be a nice package in the arsenal.

12/20

Write the editor / graphics spec for Marek early in the morning.

12/19

The first cut of Miso Dataset is up and running with bubbles.

12/14

Early Friday morning we now have datatables talking to sped/redis. And by early Friday evening it was checked into the Spinnakr code base.

12/13

Spne is completed and now we are moving on to Datatables integration

12/12

Start exploring using Miso once again, I think this is going to be a good solution. Also back into Datatables after a 3 year hiatus from Caltech. This was used in my Python based CRData version.

12/11

The date code for Redis and Visitor Loyalty is in place on the Ruby side of



the world.

12/07

Late on Friday evening I checked the bubble code into Spinnakr. Michael prompted me to check it in, and I took the initiative to complete. We have lots more work to do.

12/06

Working on putting a number in a box that allows me to draw N bubbles.

12/05

Spent all day working on cleaning up and refining redisvisitor.

11/30/2012

Justin sends me the first round of his code in a tarball. I complete a first cut of date category putting dates into buckets that the circle packing drop down will use.

11/28

Long talk with Justin at 8:30AM about RabbitMQ and Storm. Day 1 of working on circle packing.

11/27

Start looking at Justin's RabbitMQ code.

11/26

It took a couple of hours to figure out how to install RabbitMQ on my macbook. Key point was telling it to use localhost as the name, see notes on environment variables. Tomorrow start working on Visitor Loyalty schema understanding.

11/25

Michael makes changes and fully gets up and running on backbone code

base

11/24

Backbone is up and running on Spinnakr

11/22

Jsoncache is up and running.

11/20

Started working on Backbone all day.

11/19

We had our Monday dev call today and I helped Justin figure out how to write data to the queue and then take data off the queue and write it to the database.

11/18

Created my first pull request for Michael. This will be my first check into the Spinnakr code base. Late in the day after returning from a bunch of food shopping for Thanksgiving in town, I got an email back from Michael that he had accepted my first pull request. I am now officially a committer in the Spinnakr code base.

11/17

Saturday morning realization that to test my code all I have to do is change the welcome bar ID in the javascript files. Initially thought I would have to mess with resetting user passwords via Devise. This was a huge breakthrough from a testing perspective point of view. Once I figured this out, I was able to test the other users data quickly and then put my Redis Server and Sped server out there on Amazon.

11/14

I now have a production database up and running under postgresql for the

first time. The data import took over 5 hours on an m1.small. Also, figured out how to restore the ruby backup gem database backups. Also, I got my redisaggregate program to read a database on a different machine.

11/13

First introduction to Chef from Raghav. I am getting my database and spinnakr instance up and running on different machines. By the way, the AMI and snapshot EBS of the postgresql database worked great today when I fired it up.

11/10

Completed first round of understanding how to integrate Marionette into Spinnakr graphics. Now must move on to the integration of loading templates into Rails. This is the area of concentration. Exactly how does rails load templates in concert with Backbone and more importantly Marionette. The reason I need to solve this is because Rails is not loading my Handlebars templates properly.

11/06

First round of Backbone integration working with gridlayout.

11/04

Figured out the bug in the new Marionette v1.0 version. There was an extra "prototype" needed to compile the Handlebars.js template. My stormabq/jobs is up and running with the new version of Marionette.

11/02

Crontab is now working on Amazon and Resque is running with their demo programs.

11/01

Started working on Resque upon arrival in Pittsburgh. Making progress

throughout the week.

10/25

In wrap up phase prior to heading out to Pittsburgh. I got all of the database stuff up and running on the new instance on Amazon that is built via the database snapshot. I also got the pgAdmin3 talking to the second database as well which is cool. Final thing left to do is starting to launch of some simple cron job tests generated via whenever that does simple stuff like maybe some logging stuff on when each data transfer gets fired off.

10/24/2012

Another fun day of Amazon. Learned how to create an AMI from the console along with creating an EBS snapshot from the console as well. The way to create a snapshot is to simply umount and that is it. I tried creating the snapshot by umounting and then detaching. That is clearly a no no. Also, got postgresql up and running by itself on its own drive with Redis. Then when I come and mount a new drive with postgres on it I am able to start and stop it.

10/23

The postgresql port of all my code is up and running on Amazon. Amazon work always takes a full day even though you think it should take an hour. Having stuff up and running on the mounted /spinvar drive makes things a bit easier. And Ruby now appears to be building faster, maybe RVM was upgraded to make this task simpler. Lots more work to do on Amazon and also configuration of moving my code around between my local box and Amazon and how to configure the IP address stuff. There are 2 spots that need addressing. One is the Javascript world and two is the Ruby world.

10/22

The postgresql port of all of my code is up and running on my local box.

I had to blow away all of my code on Github and start over due to my lack of understanding of git merging. But after the blow away I got everything in place. Having the core configuration files in a different repository which I set up as well should make this a bit easier going forward.

10/19

The drop in CSS integration framework is up and running. See my weekly update from today for more details. It is very cool and is based on Bootstrap.

10/18

Refactored all of my Rails views and controllers to get in line with Redisaggregate. All javascript entry points for the graphics are one line simply passing in the data and the welcome ID to name mapping.

10/16

Back from Taos. I got the calendar integrated into the Spinnakr product. Also, starting to work on getting PostgreSQL up and running with Spinnakr.

10/11

D3 calendar up and running prior to going to Taos.

10/11

Skipped CSS work on hover and dove head first into the D3 calendar.

10/10

Hover is up and running in the Spinnakr branch, but CSS work needs to be done

10/09

Starting to integrate the rickshaw-hover code into my Spinnakr branch.

10/08

I am starting to now understand the prototype class architecture in rickshaw based on prototype.node.js. Postgresql is up and running on my local box and amazon as well.

10/06

spent the week working on rickshaw-hover along with a new API call dealing with Javascript dates. Looked at both D3 and Moment.js dates.

9/27

Talk to michael in the morning. Searched most of the day for a Javascript calendar widget only to discover a very nice D3 example which should pretty much do (I think) what we want.

9/26

Shipped off my first round of Rickshaw graphics code to Michael via Amazon. This is great news and a feather in my personal cap.

9/20

Start working on a D3 package called Rickshaw. Wrote my first Node program since starting work at Spinnakr way back on July 30. Also installed a new version of node V0.8.9.

9/15

Finished the week on a good note. We now have a working version of the first API call that returns a JSON data structure of all of the analytics information needed for the /manage/messaging page. Next week we will move on to wiring this call up into the Spinnakr UI side of the world.

9/11

Justin started work yesterday. I had a talk with Michael on the phone as my internet went down. Started working on the Rails side of the Spinnakr website.

9/6

Big day at work. I am now able to build the database from scratch with seeds.rb. Michael prompted me when earlier in the day one of analytics programs was failing. By 9AM I had it working. So I no longer have to import a big development.sql file that is continually out of sync. I have been wanting to do this task since I started working at Spinnakr way back on Monday July 30.

9/5

First round of aggregate code complete, starting working on testing in sinatra with rspec.

8/29

Now that the application is up and running I wanted to get back and review 2 things...

1) the javascript so.js code

2) the spn.ee code

Now I will move on to

1) Michael's **Generating Sample Data** code

2) Michael's New Aggregates model

8/28

First time getting the whole application up and running. mhartl is now working with spn.ee and spinnakr. this is great news ! I am seeing welcome bars showing up on mhartl along with a better understanding of the spinnakr side of things.

8/17

Final wrap up before heading out to Colorado.

8/15

Wrote my first class for Spinnakr in the context of converting Redis data

structures to JSON.

8/14

started out the day with a talk with Michael. Spent the rest of the day writing Redis stuff to grab data.

8/13

drew up the entire data model on a piece of paper. for the first time I see how everything is related.

8/11

On Saturday morning at about 8AM I got the nginx.conf stuff working for both spin and spne. I am proxy\_passing on the spne side which means I am bringing up the shotgun for spne manually and not using passenger. On the spin side its all working with passenger and setting the development flag in nginx.conf was critical. Reason things were not working last night until I set the mysql password for production is because "production" is the default. This was another big breakthrough.

8/8

Start working on writing data out to Redis from MySql using DataMapper.

8/7

First day taking a look at DataMapper.

8/6

We now have the MySql data files on an EBS so that instances can come up and down. Then when the instance is up we can mount the database partition.

8/3

More work on Amazon in the context of Spinnaker



8/2

Starting to understand the Spn.ee side of the data model. At the end of the day Spinnakr was up and running on Amazon

7/31/2012

Hipchat is up and running on my local system.

7/30/2012

First day of work at Spinnakr.

7/27

Deal with paperwork stuff from Spinnakr. Also, review all of the Ruby on Rails guides including work on ActiveRecord and the way that Spree generates out the sandbox based on Rails generators and Thor.

7/26

I spoke to Michael at 6PM and got the job. Spent all day working on Active Record. It is my first time taking a detailed look at how Active Record works. It was two weeks ago today that I spoke with Michael for the first time. It was also exactly two weeks ago on that Thursday the 12th and Scott's birthday that we got our first huge rain. The second huge rain was today.

7/24

Change of plans from D3. Work will be aligned in a more parallel fashion. Honestly, the past couple of days have been challenging for me as I went through a reference check for Spinnakr with Michael and I did not get the job. Bummer... So I am back to looking again. This means I need to spend more time back on Ruby and so I spent quite a bit of time looking around for another good Ruby model package and I believe it is going to be Spree.

7/23

First full day of D3. This is going to be the major thread and trend for hopefully the next couple of weeks.

7/21

I spent a little more than one week coming up to speed on the Twitter Ruby API and I am feeling pretty good where I am at. Now it time to turn your attention to D3 and matching the visualization concept with the data that is out there. Once I have the Twitter data displayed, then I can move on to visualizing the Biology data as well. You are making good progress.

7/20

By the end of the week Twitter was in good shape for round 1 completion. Today I got streaming figured out and working. It does not do historical tweets. It only gets data from now on. There is no way to go back in time and get historical tweets like the Twitter client in your web browser does. So the good news is I understand pretty fully how things work in the Twitter API.

7/16

Did the call on Monday with Adam and then got the references together for them. I am also making big progress on the Twitter project as well.

7/14

Sent off a second letter to Zeke in Santa Fe declining his invitation to come up to Santa Fe on Tuesday for job training. I had to tell him definitively that I was NOT interested in the job.

7/13

Lots more progress on the Twitter and Redis project called wedis. I heard back from Spinnakr and I have my next call Monday with Adam.

7/12

Twitter and Redis are now up and running together. First time talking with

Michael at Spinnakr. This will be easy to remember as it happened on Scott's birthday.

7/11

In parallel starting working on D3 again. This came to me while walking around Santa Fe after my interview with Zeke at JackRabbit. So we now are onto visualizing Twitter data which has 3 parts.

1. The Ruby Twitter API
2. The Ruby Redis API
3. D3

7/10

Twitter is starting to work, now move on to storing the data in Redis. Went up to Santa Fe today to talk to Zeke at JackRabbit Systems. Got a job offer for \$75K but declined it graciously.

7/9

Massive work on Ruby and understanding modules. Discovered Yard and the Ruby Documentation system. I am making more progress on the Ruby language and understanding how things are related to each other.

My first response from Blaire at Spinnakr. We set up the first call to talk to Michael on Thursday July 12, which is Scott's birthday.

7/6

This is the day that I sent out the magical lucky resume to Spinnakr.

7/3

Here we are Tuesday morning and I am finally starting to understand rspec testing in Ruby. At the very end of last week I realized I needed to understand the testing system in Diaspora since I had a basic handle on the flow of things and the basics of the Backbone code. The testing infrastructure is VERY interesting. This is day 2 of my real indepth look

into testing and I like it ALOT. The testing system is really a whole different world and completely independent from the actual runtime system except it uses the same classes and database system.

6/28

I am ending my first week of a model Ruby on Rails project called Diaspora. Today I learned as day turned to night how exactly the rendering process happens from the controller. It is in the docs, but I had to get the connection between sometimes the controller explicitly stating the render method and specifying the view and sometimes not. It is the sometimes not that was eluding me at the moment. I am having fun, keep going. This is the platform in which to embed your Backbone application. Prior to this foray into Ruby on Rails. Backbone did not have a place to live and now it does.

06/26

Day Two of Diaspora. This is a great model system just like OPower for Node and Bookmarkly for Backbone. The HUGE big win I discovered last night is that Diaspora also uses Backbone. BINGO ! We are happy.

06/25

First day working on Diaspora. I got it up and running by lunchtime, installed MySql and the MySql workbench. This is great news and will be an awesome way to take my Ruby and Rails skills to the next level. Way better than hacking on CRdata. But then again once I get passed this step then maybe CRdata will be a snap if I so choose to go there.

06/24

Back to work after a nice week off in Silver City. The next couple of weeks will be focused on Ruby, Rails, and other things associated with these two topics. Got the src code for rails up and running. Rvm is the key to getting all of this stuff going.

06/14

Early in the morning, just as the first light hit. I discovered, rvm, ruby version manager, and ruby, rails, et al was up and running on my computer. From there, I started playing with Rails. It is the first time ever I have run a rails app on a macbook pro. This is really super fantastic news !

06/13

Now that Twitter is up and running the next step is to realize that the whole idea behind what I am doing is to understand the content better. So whether it be Craigslist descriptions or a set of Tweets from a group of people it would be great to do some simple NLP and so that is where the Natural package comes into play. I did some work today on their package in natural called tf\*idf [term frequency -- inverse document frequency]. This brought me to the realization about WORK. If I am going to get a job I need to broaden my scope to include Ruby on Rails jobs and Django jobs in the Python world. So here we go --- off to just review enough of this stuff to possibly get a job in this area.

06/12

Twitter is up and running in a very basic way. The node.js client is pretty simple to understand although it appears to need some work. Also, I got the twitter console called twurl up and running on my machine.

06/11

We are moving on to a new chapter in my work. We are done basically with Craig as a dataset and moving on to Twitter. I will be maintaining the Craig data set going forward but not much new work on the topic is on the horizon for now.

06/07

jobs is done. I finished up the tutorial this morning and shipped off a copy to derick bailey, david koblas, and the folks at craigslist.

06/06

html filters is now working in craig. upload final data set for now. need to spend more time working on the tutorial.

06/03

A HUGE weekend of productive work. We are now up and running on gh-pages on the first cut. Also, the Tutorial has been written and deployed to Github. Congrats on a job very well done !

06/02

Big Saturday. Not a whole lot of work but some breakthroughs. The code is now sitting up at GitHub and I am beginning to write up the markdown. So we are moving on GitHub for the first time. Today is the FIRST day of GitHub.

05/30

We are now up and running with the redis only version of craig. This is very cool as it vastly simplifies the data generation side of my marionette tutorial. I came up with the idea of doing a redis only version this morning and by 7PM we were good to go with the first version.

05/29

Going to Rand's tonight for dinner with Ken and Julie as Sima is heading out to Europe tomorrow. Now that things with Marionette and Jobase are starting to fall into place I am going to go back and generate out a larger craig dataset. While doing this it occurs to me as well that it would be nice to be able to do things with the LARGE craig dataset like pull all of the items from 5 cities a,b,c,d,e etc... Also, it would be great to randomly pull 1000 entries or 100 entries from the larger json structure. So it is back to Craig and data to wrap this up prior to the release of jobase on github.

05/28 jobasec00ah, must00alb

At the end of Monday on Memorial Day I got things in pretty good shape. I am starting to better understand Marionette. I worked all weekend, the whole time that the kids were here. The must00alb code base is the code that takes in the craigdataset and generates out the final data set that jobase uses.

05/23

Making progress on getting Marionette integrated with Handlebars.

05/19

Last day of work was Friday May 11 prior to going camping in Pagosa Springs. We are now back from the San Juan National Forest and today, Saturday I got back to work. Thanks to David Koblas, see email on this day, I got Handlebars up and running with both cats and groups. This is great news ! I have wanted to replace underscores template engine with handlebars now for quite some time and its working ! From now on forward we will be using handlebars as our default template engine.

05/03

Knockout is knocked out, I am sticking with backbone core frameworks. The past couple of days I have been working on d3 and the miso cabinet example and treemaps. I think I have this down along with the miso dataset. We now need to go back to marionette to get the multiple view thing working with the cats example.

04/27

Data.js looks to be very interesting. Also, Dance.js is based on CSS but uses a D3.js model and finally another big win for the day was the miso project using backbone for a very nice example. I believe at the moment we are well on our way to defining some very cool demos and pulling data.

04/26

So I spent the past two days looking at Knockout.js and it is nice. There are some drawbacks as pointed out by Kevin Malakoff the author of knockback. They are namely no real persistence or routing or history like backbone.js. However, there are some nice features so understanding and implementing a fairly simple D3.js example might be the way to go.

04/24

I had a nice break of about one week mas o menos. Peter left this morning around 11AM and I got right back into work. Luckily, I did a search for backbone.js and d3.js and found a nice election example implemented in knockout. So this is very cool as well. If I look at one more it will be ember.js I think. Key point is understanding multiple frameworks gives you a handle on the whole concept mo better. Yes, spent more time looking at ember as well. So, I think we are good with these three, but maybe angular.js as well. I will post here.

04/17

Day 1 of Marionette and my amazing introduction to this incredible programmer and writer named Derick Bailey. It was a Monday morning and I was cruising through a blog post about Backbone.js from a guy in Halifax who had written a short demo app about Donuts. And in one of those posts there was a comment by someone who said to take a look at Derick's project. Life works this way, sometimes you just get lucky or turn over the right rock. When you turn over that rock you know you have found something good. If it hadn't been for the past couple of weeks of hard work on Backbone I would not have realized that I "needed" Marionette, but now I know I do and it will make my backbone coding adventure much more pleasant and rewarding.

04/15

I worked all weekend on trying to get Tab Views to work only to discover that it was a harder problem than I originally thought.



04/14

Based on code I wrote in Argentina and using Backbone.js as a model I got name spacing working for the first time in my backbone code. Also, beginning to work on the “Tabs” problem and being able to run multiple views in the same application. This is NOT a trivial problem. My initial hit on it did NOT work which was to try and use of the shelf code. So tomorrow I will spend more time looking at the problem and report back to you.

04/13

OK. So the CSS is working and the tabs are working as well. Now I need to throw all of my content into templates so that there is no static HTML hanging around. Everything needs to be dynamic. Only div’s with ID’s can be static. But any other content that hangs around can not be gotten rid of quickly. So I will go to the next step. A product called Meteor was released this week along with a product called Firebase.

04/12

Day TWO of CSS. I have messed with CSS for years, and every time it bites me in the butt. So I here I go again trying to figure this stuff out. The ONLY difference is this time I have a structure called BOOTSTRAP in which to work with. So I have something to base all of my work on and a nice model to be able to use as my basis of learning.

04/11

Now that Backbone is in pretty good shape a deep dive into Bootstrap and less in order to be able to drive everything else forward. So I would call this Day ONE of CSS.

04/10

I have been working on Backbone now for about two weeks full time and I believe it is starting to sink in. Today, after much thinking and research I

moved to a one model, multiple view system. The first insight into this was an application called Funky Calendar. From there, I decided that going with multiple views made sense and finally got circles up on the screen being driven by the check boxes. We are making HUGE progress on Backbone.

04/08

On this beautiful Sunday morning I finally got D3 and Backbone working. I was able to put three circles on the screen with colors that were read from localStorage.

04/05

I discovered that Mustache templating looks to be the way to go while hanging out at the library across from Academy waiting to eat lunch with Dan at the Thai restaurant near JSI. Embers.js uses Handlebars as their templating engine and Handlebars is a superset of Mustache. I believe this will be a slightly more robust solution than what I am currently using which is Underscore micro-templating.

04/04

Spent the whole weekend while Hb was in Denver at Ashley's baby shower for Tanner working on Backbone.js. This is my first foray into this product and the concept is very cool. I will report further progress here.

3/29

When I initially started out with Node.js and Express, "OPower" was my ticket to understanding. The same is now true for understanding Backbone. I need something that I can wrap my head around to understand this concept and it turns out that Bookmarkly is it. Very, very cool.

3/28

OK --- I just realized I do not have to put the year down on each entry, this

is good news. It will just show up every once in a while. I am starting to make huge progress in understanding Backbone. Bookmarkly and a couple of other examples are serving as great leads for understanding this stuff. Also, last evening I started diving back into the D3 examples. It will be a combination of these two things that drives the web site moving forward.

Tuesday 3/27/2012

**First day of Front end work. Looking into D3 and Backbone at the moment.**

Monday 3/26/2012

Figured out (finally) how github web pages work. Turns out the magic sauce is a branch called "gh-pages". See my email for more examples. Finished up the first round of the Craigslist project. Tomorrow I am moving on to the front end which is D3 and Backbone and Bootstrap.

3/24/2012

Total re-write of the way craig data is retrieved. I am now doing it correctly and no longer using a phony counter to drive the event system. Now it is all call back based. I also switched over to request from wwwdude. Also, I am storing all of the data I retrieve in Redis. I have also generated out a report so I know exactly how much data is associated with each city. We are making good progress here.

3/23/2012

The way connections work in Mongo and Craig overall is now working and I am generating out HTML content which looks great ! Now I will move on to getting the events from reading the content to storing the content.

3/21/2012

Streaming is now working using Clarinet. This is great news ! We now

have a very nice way to stream data off the internet and into a file and eventually Mongo as well. Also, I am able to stream data out of Mongo now, as of today, using CursorStream which is very cool. I am pretty sure it does not work in Mongode, however. But streaming works in Mongoose. My initial cut was using the native Mongo driver. We are making lots of progress on streaming which is going to be mission critical going forward.

3/19/2012

Spent the weekend looking at streams and in the end I found a buried treasure called Clarinet. This piece of technology along with many others WILL take me to the next level.

3/17/2012

Streams are HUGE and I am just starting to understand WHY they are so important in my application and future applications.

3/16/2012

Struggled all day Friday trying to get Promises to work when in the end I realized all I really needed to do was do a callback which would have solved the problem. You only really need promises if you are waiting on multiple returns of different pieces of information. For example if you go out to Mongo and Redis at the same time and need both pieces of information to move forward. Or in the case of MileWise they needed results from all of the different airlines and then they could move forward.

3/12/2012

Redid schema to further align with future data feeds. Pulling generic Mongo collection and creating a JSON object.

3/11/2012

Big weekend of work, sending out resumes. Looking at the Twitter Feed and determining that this is probably not the call for "now". Also, got

Haraka up and running on Amazon. I am able to send outbound emails to both sfi and gmail. This is pretty darn cool.

3/9/2012

Back from a lovely trip to Sedona and Flagstaff. This week was all CSS, mainly integrating Bootstrap into the Backbone application todo and the D3 word cloud.

2/29/2012

First day of working on Bootstrap as I extend my reach into the Front End Ui World. This is rather exciting as I will be bridging both front and back end work.

2/28/2012

Finish up first round of word cloud. It is ready to go for RedFork. This turned out to be done extremely quickly. Wow, I am happy.

2/27/2012

Major hacking on Jason Davies word cloud as I begin to work on integrating it into RedFork as a core piece of technology.

2/23/2012

We are now starting our voyage into front end work with the start of spending more time looking into backbone.js. It looks like this will become another piece of core technology.

2/22/2012

Today I had a huge day of work. The xml2js solution is working for me now that I installed a new version of it and set the Option 'explicitArray'. Also, I am now persisting my data model to Redis and the daoJobRedis is now driving the UI as well. So we are able to store data in both Mongoose and Redis and then use the dao so the UI's work.

2/20/2012

It has come to my attention that I need to have a separate work log from my c-series diary. Moving forward, my work schedule is going to be more sporadic and so I need to be able to chart a path forward by looking back. So it is only appropriate on Eve Tolpa's birthday 2012 that I begin this new parallel series of work and play. Although, at times I will outline my work and play together, it probably makes more sense to divide the two worlds as they should.

2/10/2012

This is my final day in Florida. During the first week I concentrated on writing data out to Mongo. I would parse the BioGrid files using the CSV parser and then write the data out to BioGrid using Mongode the tool written by Vinay at MileWise. The second week in Florida was all about D3.

1/25/2012

I left for Florida on this day.

1/24/2012

Released rf00abq.tar.gz which is the Redfork code base. While in Florida I did no work on RedFork. Focus of this work entailed a bit more than two weeks of coding including the following concepts. First round of testing with Mocha. I had not done any work with Mocha prior to this work. Also more work on code reorganization.

1/10/2012

Final Day of work at MileWise.