CLOUDFAIL
SCALING TO INFINITY – BUT NOT BEYOND
Kunal Johar
What would you do?

- You take your senior design project to the next level
- You have some traction – 10-15 people a week using it
- A game-changing opportunity hits you in the face
- You need to scale to tens of thousands of users per week
Act as If

• Scaling is no big deal right?

• Amazon’s Elastic Cloud; Rackspace’s Infinite Capacity

• 50,000 is a small number even in O(N^2)

• I’m sure I can figure it out
“We are counting on you”

• Our organization depends on this software for our annual operating budget

• This year was a total disaster.  Multi-week outages.

• We need you to tell us that this will work, that the system won’t go down, no matter how much traffic we send to it
No Problem

• “The old vendor was amateur hour”

• We’ll distribute the load across multiple servers

• We’ll load test

• We’ll scale up

• DON’T WORRY
MAY 20, 2013

Paperwork Signed – Now the Challenge Begins
Our Software Does it all (soon)

• It was a Brutal Summer

  • We had 12 weeks to learn, architect, and build what ended up being 1800 hours worth of features

  • The margin for error was Zero

• We also had to make sure our system would scale to meet the super-surge of traffic in January
Full Team Buy-In

• The stakes were known to everyone.

• If we succeeded, we’d pivot ourselves to the top of the market.

• If we failed, half the team would be out of work

• Our client called failure “Mutually Assured Destruction”
Full Team Buy-In

• The stakes were known to everyone.

• If we succeeded, we’d pivot ourselves to the top of the market.

• If we failed, half the team would be out of work

• Our client called failure “Mutually Assured Destruction”
SEPTEMBER 2, 2013
Lot’s of Overtime, Heat, Stress, Anxiety. But we did it.
Memo to Developers

Hey guys,

Just wanted to thank you for the tremendous efforts you put in this summer. We would not have made the deadline if you did not only meet, but exceed the goals I had set out for you.

The launch has gone smoother than any other release, and considering the level of change - this is some great news.

While [redacted], I'm sure will still come up with new requirements that will surprise us, I believe we should a strong effort to give them confidence in us. With their last vendor, their "Sept 1 deadline was missed, actually they launched "Oct 1" - but the system wasn't totally ready, so they had hell to pay for nearly 7 months of system usage. Because of that mess-up, we were able to come in and steal [redacted] who had a 15+ year relationship with our competitor.

We got news that our competitor is hurting badly without [redacted]. They fired 2 developers and 1 support person, but we are looking past them onto the #1 Awards Software in the market:

[redacted].

Over the next set of versions we'll really solidify our base (Faster Performance, Azure) and place the features that are needed to make ourselves #1.

In some regards, it is hard to believe we made it this far. Features like Bulk Assign of Winners (which went live yesterday) seem like necessities that we lived with out.

Behind the scenes, we "faked" a lot of these features to clients on demos. When the need came, we would have our support staff, and often myself + Zack doing massive amounts of data entry.

Now we are probably the #2 maker of awards software.

I don't think that's good enough for any of us.

So thanks again, and I look forward to the journey to #1,

Kunal
Load Test or Beta Test?

• From the September 1 Launch date; until even today we have been hit with new feature requests

• “Oh! I forgot about that – but it’s really important”

• How do you balance engineering priorities vs feature priorities?
How to Construct a Load Test

- Write custom scripts that simulate real users using your app
  - Selenium Web Driver + Sauce Labs
  - Browser Mob (Neustar)
  - Load Impact

- Write a custom handler that simulates the user payload
  - Loader.io
Our Loader.io Script PayLoad

- POST 100 KB of data
- Simulate Save to Database
- GET 100 KB of data from Database
The Actual Load Test
300+ Users Per Second!

- Whoo hoo!

- 300 users per second must mean what? Thousands of users per minute!

- I report to client a very successful load test and put the matter towards some wishful thinking
SURVIVORSHIP BIAS

http://youarenotsosmart.com/2013/05/23/survivorship-bias/
Survivorship Bias

The misconception
You should focus on the successful if you wish to be successful

The truth
When failure becomes invisible, the difference between failure and success may also become invisible
Survivorship Bias

• “A Cabal of Geniuses” assembled at the request of the White House

• Top women mathematicians (human computers), Nobel Prize Winners, researchers formed the Statistical Research Group
Keeping Airlines in the Sky

• At its lowest; survivability of a WWII bomber was 50% on a mission

• “Ghosts already” is how airmen were known

• “How, the Army Air Force asked, could they improve the odds of a bomber making it home”
Armor

• Military commanders inspected the planes that made it back

• Ideally they could put armor on the whole plane, but then it wouldn’t fly

• Tons of bullet holes in key areas of the fuselage, wings, near the gunners

• The army was about to add plating to these parts of the bombers
Armor

• The scientists successfully argued “Survivorship Bias”

• Stop looking at the survivors – it is the other parts of the plane that need more armor!
WHAT IS “CLOUDSCALE”
WE OPERATE CLOUD SCALE

(well...soon)
Clouds

- Amazon Cloud Player
- iCloud
- Gmail
- Salesforce
- Dropbox
Types Clouds

- **SaaS** – Software as a Service
  - **awardsCMS, conferenceCMS, Gmail, Drop Box**

- User worries about data
  - 10 emails or 10 million emails, gmail will work fine
  - 1 file or 1 million files, drop box will work fine
  - Service scales as user demand scales
Types Clouds

• IaaS – Infrastructure as a Service
  – Generic Compute Power
  – “Virtual Machine”
    • 4 processors
    • 12 GB of Ram
    • $500/month

  – Can add / remove resources as needed
IaaS

• The Good
  – Not much different from using local machines or servers
  – Very little learning curve
  – Easy to add / remove resources

• The Bad
  – Generic compute power means there it is up to the developer to do the scaling
IaaS

- Load Balancer
  - Sends transactions to one server, if server fails tries another
  - Up to developer to sync code and files among
PaaS

• Platform as a Service
  – Provides a technical function
  – Requires developers to follow a pattern, but allows scaling
  – Examples
    • Convert this video
    • Store this file
• Traffic of our system the last 2 months
• How do we deal with spikes?
• What if we get a lot of video uploads?
INFINITE SCALING

• Infinite is a myth
• The idea is to ensure that the user demand is met at all times
INFINITE SCALING

- Determine threshold of a transaction
  - Say 20 seconds
  - Know how many transactions you can do at once
  - Determine how much IaaS compute power you need
INFINITE SCALING

• Figure out things that take > 20 seconds
  – Converting a Video
  – Running a Report
  – Sending an Email Blast
  – Saving a Large Document
INFINITE SCALING

• Background tasks wait their turn to get done
  – Set a maximum wait time, say 60 seconds
  – If wait time > 60 seconds, automatically spin up more resources until threshold is met
  – When queue size shrinks, say < 30 seconds wait time, spin down resources
INFINITE SCALING

• Why bother with 20 second timer and background tasks?

• Avoid crashing
  – If a process takes > 20 seconds it has the potential to consume more resources than we planned for
  – So we should kill the process and make a note so developers can move it to a background process or optimize
LOL
WE DON’T DO THAT

Zack’s first comment as I concluded that presentation
Our Architecture

- Rackspace Cloud Sites
- Report Server
- Amazon Transcoder
- FilePicker.IO / Amazon S3
- Database Server
- SendGrid
- PDF Generator
- Credit Card Gateway
WEEK OF JANUARY 6

Everyday is a Record Traffic Day
Scale up on IaaS

• Someone trying to generate a 150 page PDF

• The norm is 10-15 pages...

• “OutOfMemoryException”
Thursday, January 9, 2014

Overview

Right now

152

active visitors on site

Pageviews

Per minute

- 20
- 19
- 18
- 17
- 16
- 15
- 14
- 13
- 12
- 11
- 10
- 9
- 8
- 7
- 6
- 5
- 4
- 3
- 2
- 1

Per second
Whoo Hoo!

• No Issues on our highest traffic day ever!

• “Can’t wait till that number hits 250 per minute!”

• “Tomorrow will be our biggest day yet!”
Friday, January 10, 2014

- Approximately 12:00 Noon
  - Site traffic is around 185 people, 50 less than the previous day’s high
  - 1 out of every 12 hits times out
  - According to Rackspace, a node is failing on cloudsites and will be taken out of rotation
  - About 10 complaints so far, but I email “Everything is under control”

- Approximately 12:30 PM
  - Traffic falls to about 150 people per minute
  - Things are fine
  - Phew
Friday, January 10, 2014

• At 1:00 PM we have a job interview for a new support person

• I have live chat open with Rackspace and am hopping back and forth between the interview --- not the best way to hire someone

• 1:45 PM interview over, and I learn traffic is at 220+ people.

• The site is pretty much dead

• While I work on the issue, my phone is ringing with an frightened customer. Our help desk is filling up with complaints non-stop

• With a stone-cold face, I walk to my teammates. “This is bad. I need help”
Backup Plan

• I knew CloudSites had some limit, but I had a plan to shift traffic at a moment’s notice in a worst case situation
Backup Plan Now in Play

• Using CloudFlare, a service that lets us rapidly change DNS records; traffic was redirected to the super server

• 1 second later
Backup Plan Part II (Scale Up)

- OK – I’ll spin up the most powerful server I can buy.
- 64 GB RAM
- 32 vCPU
Backup Plan Part II

• 19 seconds later
3:25 PM

- Rackspace gives me a one time “boost” to capacity

- Let’s me know about “HTE” for the future....
  - “If you are having a high traffic event, let us know in advance”

- I kiss the floor. My company is saved by the whim of my hosting company
9:00 PM

- Zack and I finish responding to customer complaints
- It would be weeks before I could sleep normally again
What the heck happened?

- The initial load test was testing people submitting one application at a time

- The PDF issue was actually a harbinger of things to come

- Thursday had record traffic, but Friday had people doing “Finalization” (commits)

- Our commit code was very slow, and used a lot of RAM. As a server would get overloaded, the app pool would restart – this would add load to other servers

- Demand > Supply caused a chain reaction making servers continually failing until more supply was added
Our Future Plans

- I’m too scared of PaaS for a complex use case!
- Not enough data to know when things fail.

<table>
<thead>
<tr>
<th>IaaS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Amazon Transcoder</td>
</tr>
<tr>
<td>SendGrid</td>
</tr>
</tbody>
</table>
Thanks!

Kunal Johar

kjohar@alumni.gwu.edu