

## **Instagram Technology Stack**

Instagram is a fast growing, relatively new social media company that offers real-time photo and message sharing services to its users. The company started as a local server operation but switched their stack within days of launching due to the unpredicted influx of people creating accounts and uploading photos. Now almost two years old, Instagram has settled on a reliable stack to power its services using the following technologies.

### **Application Servers**

Instagram uses an ever growing number of Amazon Elastic Compute Cloud (EC2) High-CPU Extra Large machines running Django. A Python-based web framework. Each instance has 7 GB of memory, 20 EC2 compute units on 8 virtual cores and 1690 GB of storage on a 64-bit platform. This platform was chosen because Instagram found that their processes were largely CPU-dependent and not as much memory-dependent, and these instances are the right balance between the two. The WSGI server is managed with Gunicorn, and parallel code deployments are executed with Fabric.

### **Data Storage**

The massive amounts of data shared on Instagram are managed by PostgreSQL on 12 EC2 Quadruple Extra-Large memory instances, with twelve replicas at another site. The instances have (per machine) 68.4 GB of memory, 26 EC2 Compute units over 8 virtual cores, 1690 GB of storage and are EBS-optimized at 1000Mbps. The data is arranged in a “shard” configuration where each machine gets a fraction of each data piece. The EBS drives are arranged in a RAID configuration using mdadm. The data in memory is managed with vmtouch to keep it consistent across the cluster. The PostgreSQL instances are in a master-replica setup that are made with Streaming Replication. EBS snapshots are frequently made to backup the XFS file system, which allows for freezing and unfreezing of the RAID arrays simultaneously to make accurate and reliable snaps. Pgouncer connects the data storage servers to the application servers to combine all the connections to PostgreSQL. The photos uploaded to Instagram go directly to Amazon S3. Amazon CloudFront is used as the CDN. The main feed, activity feed, and the sessions system are powered by Redis with additional Quadruple Extra-Large memory instances. Redis mirrors the Data storage system in terms of master-replica setup and snapshotting. The geo-search API is Apache Solr. Caching is done with 6 instances of Memcached.

### **Task Queue and Push Notifications**

Gearman is used as the task queue system to process notifications and sharing of photos to other

websites, and perform fan feed-out. Approximately 200 workers are processing the task queue at all times. Push notifications are handled through the open-source Twisted service pyapns.

### **Load Balancing**

Load balancing is done through Amazon Elastic Load Balancer, with 3 NGINX machines. To lessen the CPU load on nginx, SSL termination is performed at the ELB level. DNS services are provided with Amazon Route53.

### **Operating System and Hosting**

Instagram runs Ubuntu Linux 11.04 “Natty Narwhal” on Amazon EC2, because they found other versions of Linux to be too unstable for their needs.

### **Monitoring**

Munin is used to graph metrics and monitor normalcy across all the server deployments. Custom plugins specific for Instagram are written to monitor other statistics that aren't included in the system level functions. This service is monitored externally by Pingdom; notifications and incidents are handled with PagerDuty software. Error reporting is managed with the open-source Django application Sentry, which allows for real-time error observation.