Scaling our own website with FoundationDB

MAR 20, 2014 – BY DAVID BROWNING

We obviously like to brag about our products here at FoundationDB, but we put our money where our mouth is–so to speak–and "dogfood" them whenever we can. So it's no surprise that our website runs on top of our SQL Layer and core storage substrate. While the site doesn't currently generate a crazy database load, I thought it was worth showing how FoundationDB gives you the confidence to scale your SQL application on top of a fault-tolerant database.

We deploy everything on Amazon EC2 as two logical layers: the app tier and the storage tier.

The app tier

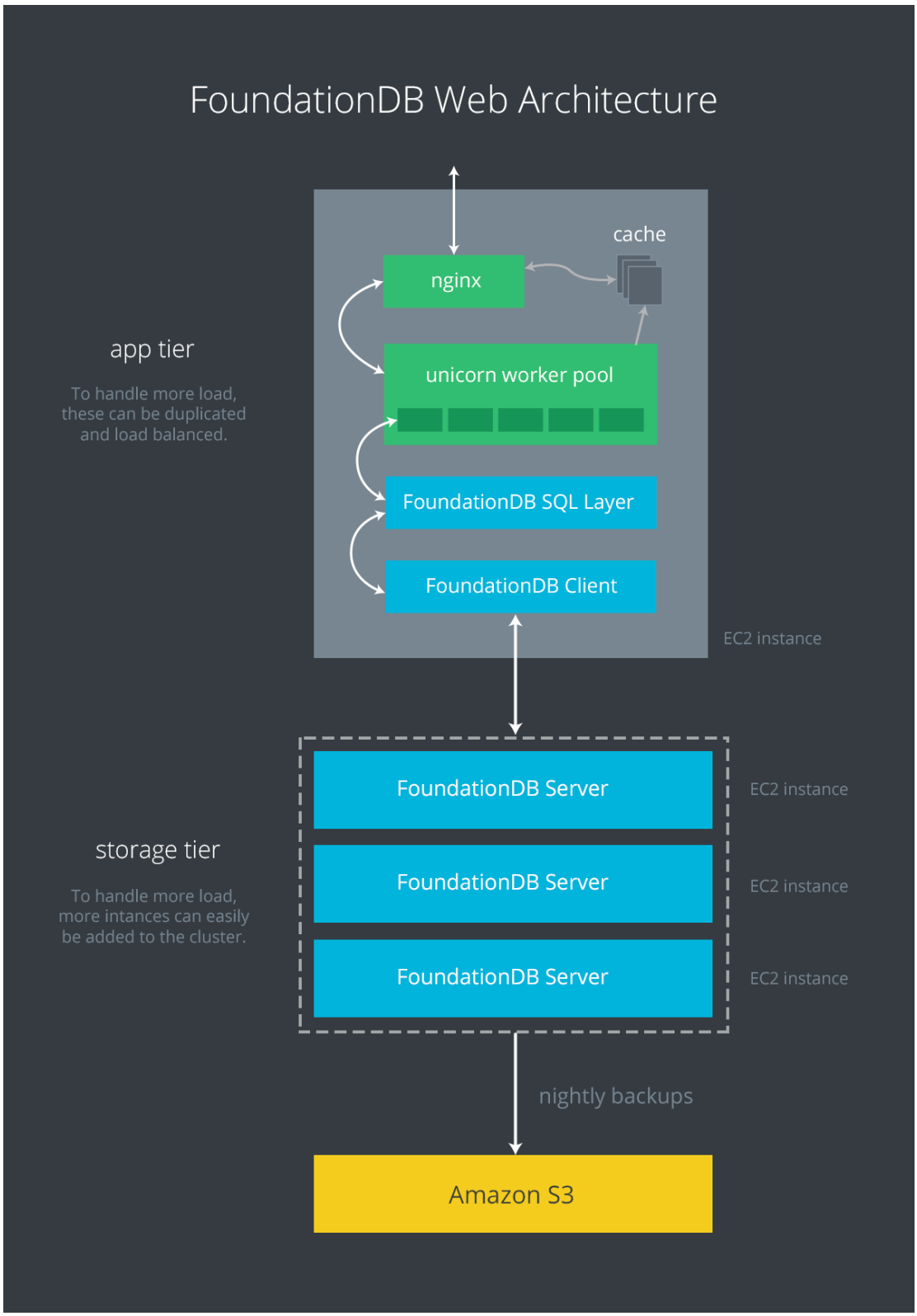
The app tier is currently made up of a single instance. Nginx serves the web request, first checking to see if it can be served by a cached file. The vast majority of our website content is static and can therefore be served very quickly as a cached file rather than traversing a web framework. If the content isn't (or can't be) cached, then it is passed to a pool of unicorn workers that run our Ruby on Rails application.

Ruby on Rails (using ActiveRecord) speaks SQL under the covers, so we're using the FoundationDB SQL Layer to handle SQL queries. The SQL Layer then uses the FoundationDB client to read from and write to the cluster of FoundationDB server machines.

The storage tier

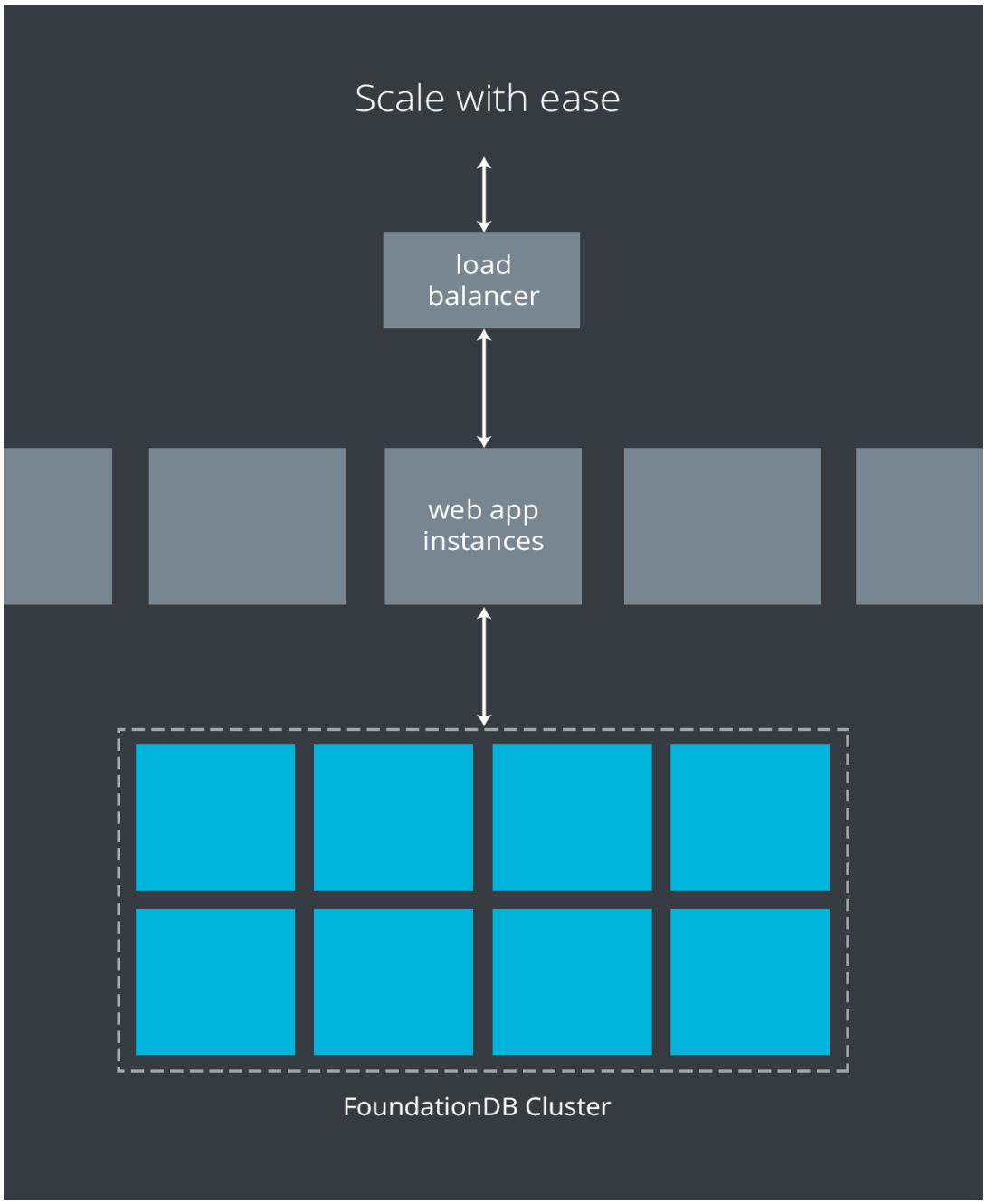
The FoundationDB cluster is made up of three EC2 instances that are each running a FoundationDB Server process. FoundationDB's excellent fault-tolerance allows us to run with double redundancy, meaning a cluster machine can go down and the app will continue humming along with zero down time.

Just to be safe, we do back up our data every night using the built-in backup mechanism and then store it in Amazon S3.


FoundationDB Web Architecture
app tier
To handle more load, these can be duplicated and load balanced.
cache
nginx
unicorn worker pool
FoundationDB SQL Layer
FoundationDB Client
EC2 instance
storage tier
To handle more load, more intances can easily be added to the cluster.
FoundationDB Server
FoundationDB Server
FoundationDB Server
EC2 instance
EC2 instance
EC2 instance
nightly backups
Amazon S3

Scaling with ease

If we end up needing to support a heavier load than our current single app instance can handle, we can very easily spin up more instances in the app tier and place a load balancer in front of them. They all communicate with the same FoundationDB cluster whose ACID transactions and fault-tolerance ensure our data is safe.



If the storage tier starts seeing heavy loads, FoundationDB makes it incredibly easy to add new instances to our data cluster.

Hopefully this explanation was helpful. Download FoundationDB today and see how it helps you architect your system in a way that greatly eases scaling when you need it!

— David Browning