

Scaling Lightsail to Handle More Traffic



Thomas Compagno

VP OF TECHNOLOGY

@tomcompagno www.linkedin.com/in/tomcompagno



Overview



Setup a new IAM User with Lightsail Permissions

Install the AWS CLI tools

Create a Lightsail database

Prepare our Wordpress server

Copy and Load Balance

Discuss additional aspects of scalability



Setup IAM User Access



Demo



Use AWS console to create a user

Apply appropriate Lightsail permissions

Save credentials to local machine



Demo of setting up IAM user with lightsail access



Install AWS CLI Tools



Prerequisites



Python

Version 2.7+ or 3.4+

<https://www.python.org/downloads/>



Pip

Python package installer

<https://pip.pypa.io/en/stable/installing/>



Demo



Validate prerequisites are installed

Install tools and verify



Demo of installing aws cli on a mac



Lightsail Database



Demo



Use Lightsail console to create a database

Load the database with our Wordpress backup

Validate Wordpress tables exist



Demo of creating a new LS db



Prep Wordpress Configuration



Demo



Access our Cloud9 environment

Update our config file to point to new central database

Test and take a snapshot of our Wordpress server



Demo of using cloud 9 to change wordpress,
manual snapshot to use with cli



Copy and Load Balance



Demo



Use command line and API to create a second Wordpress server

Use command line and API to create a load balancer and configure

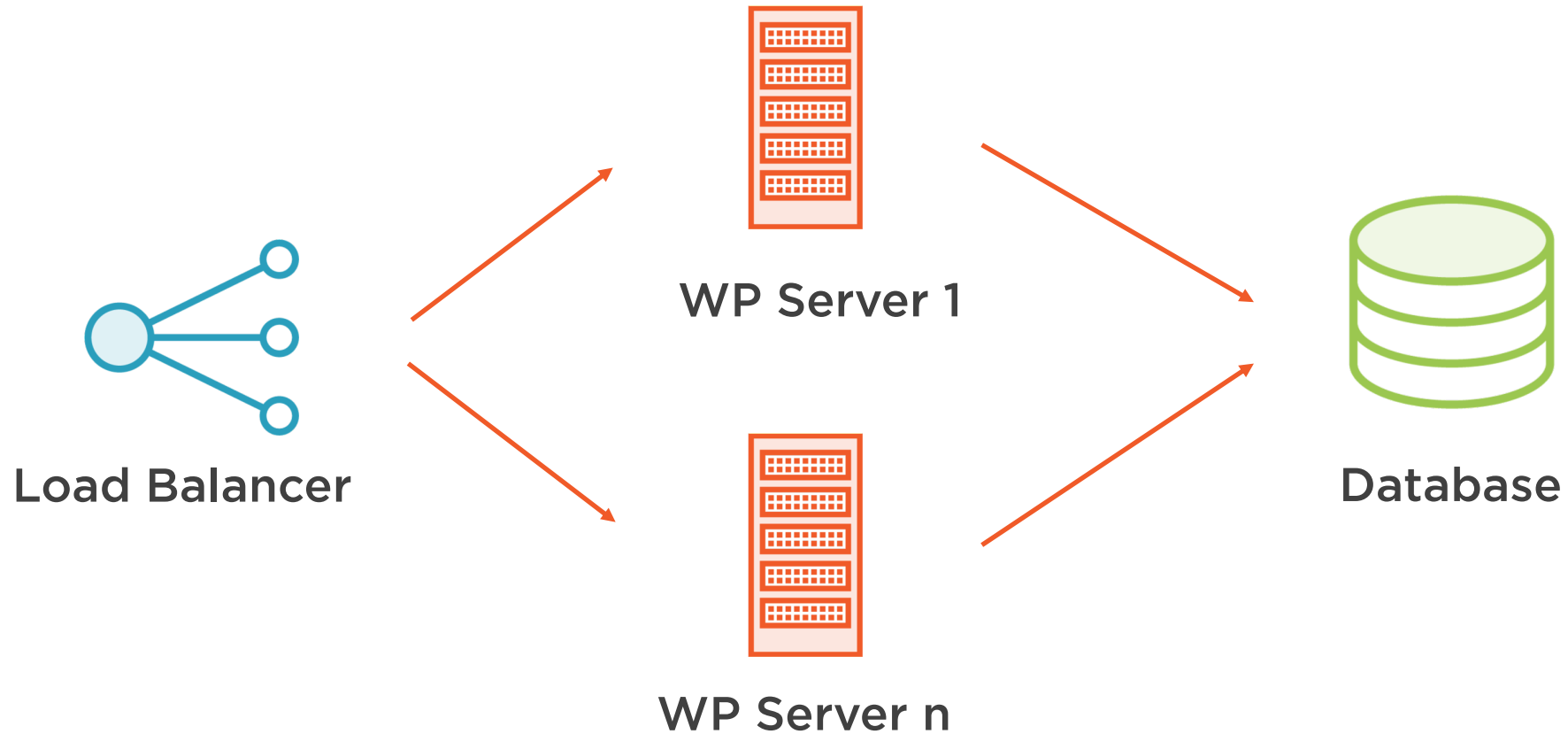
Test our scalable configuration



Demo of using cli to create new instance and load balancer and attach two instances



Implemented Architecture



Discuss Scalability



What Is Left to Do with WordPress?



Central file storage plugin



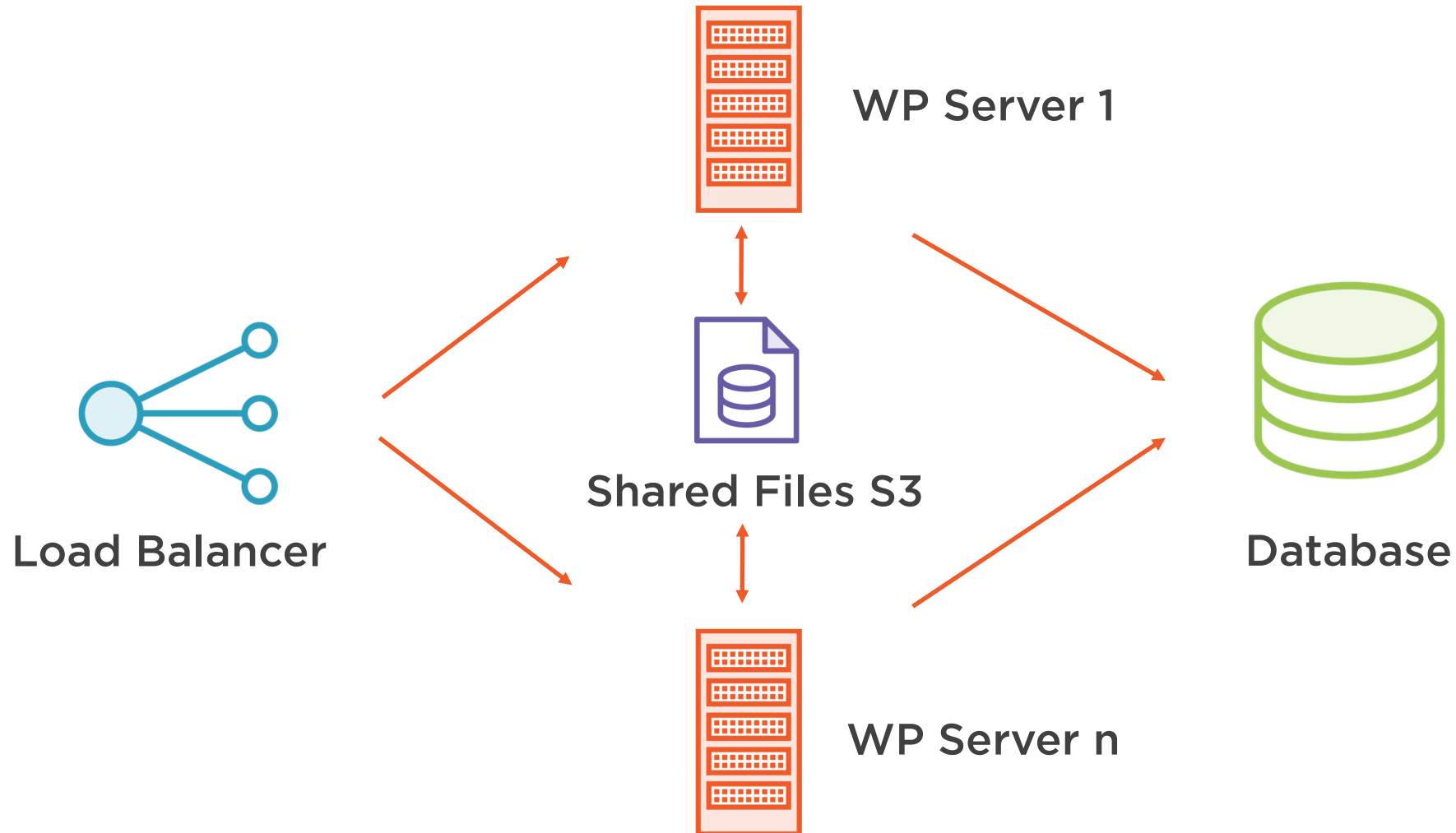
Session persistence



Workflow for updating Wordpress dependencies



Production-ready Architecture



Summary



Built a simple yet scalable architecture

Lightsail is easy in the console as well as on the command line

Globomantics team has all the tools they need

