AWS re:Invent

CMP309-R

Easily deploy and scale a cloud app with Amazon Lightsail

Mike Coleman

Developer Advocate, Amazon Lightsail

Amazon Web Services

AWS re:Invent

aws

What you will learn

What is Amazon Lightsail?

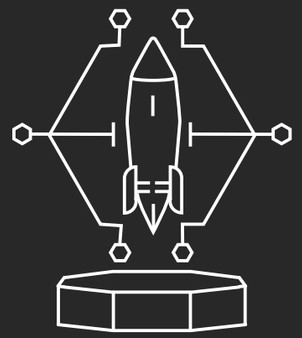
When to choose Lightsail

Using Lightsail databases

Horizontal scaling with snapshots

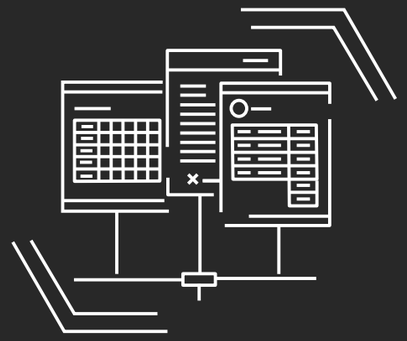
Vertical scaling with Amazon Elastic Compute Cloud (Amazon EC2)

Amazon Lightsail: Cloud made easy



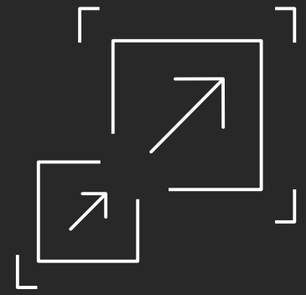
Launch

- Get started in minutes
- Low predictable pricing



Manage

- Intuitive global console
- Robust API and CLI

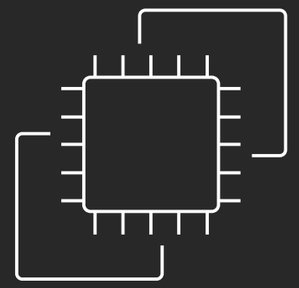


Grow

- Easily scale your ideas
- Access other AWS services

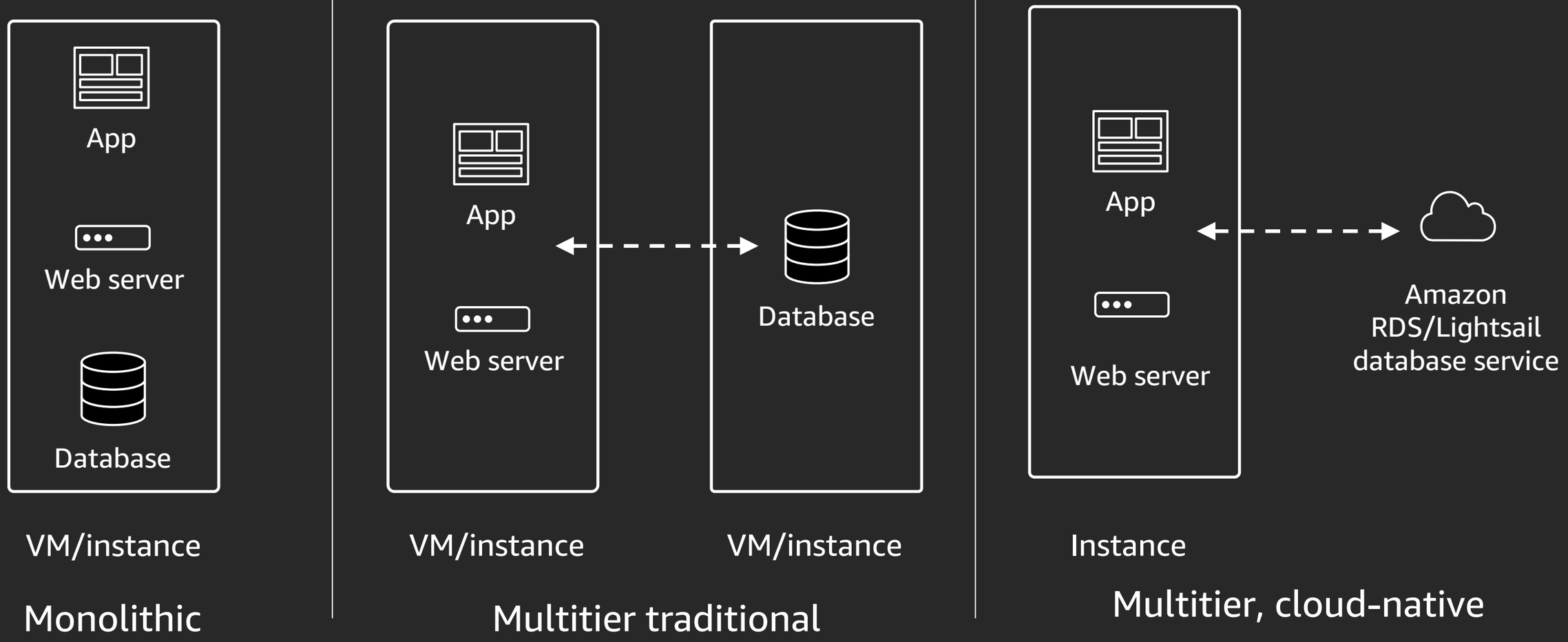
When to choose Lightsail

- Small-scale, multitier apps
- Websites/web apps
- Testing environment
- Line-of-business software



- Large, multitier applications
- Customized workloads
- Advanced networking
- Database read replicas

Common application architectures



The application: Simple to-do list

PHP front end

MySQL back end

Phase 1

- Deploy app into a single Lightsail instance

Phase 2

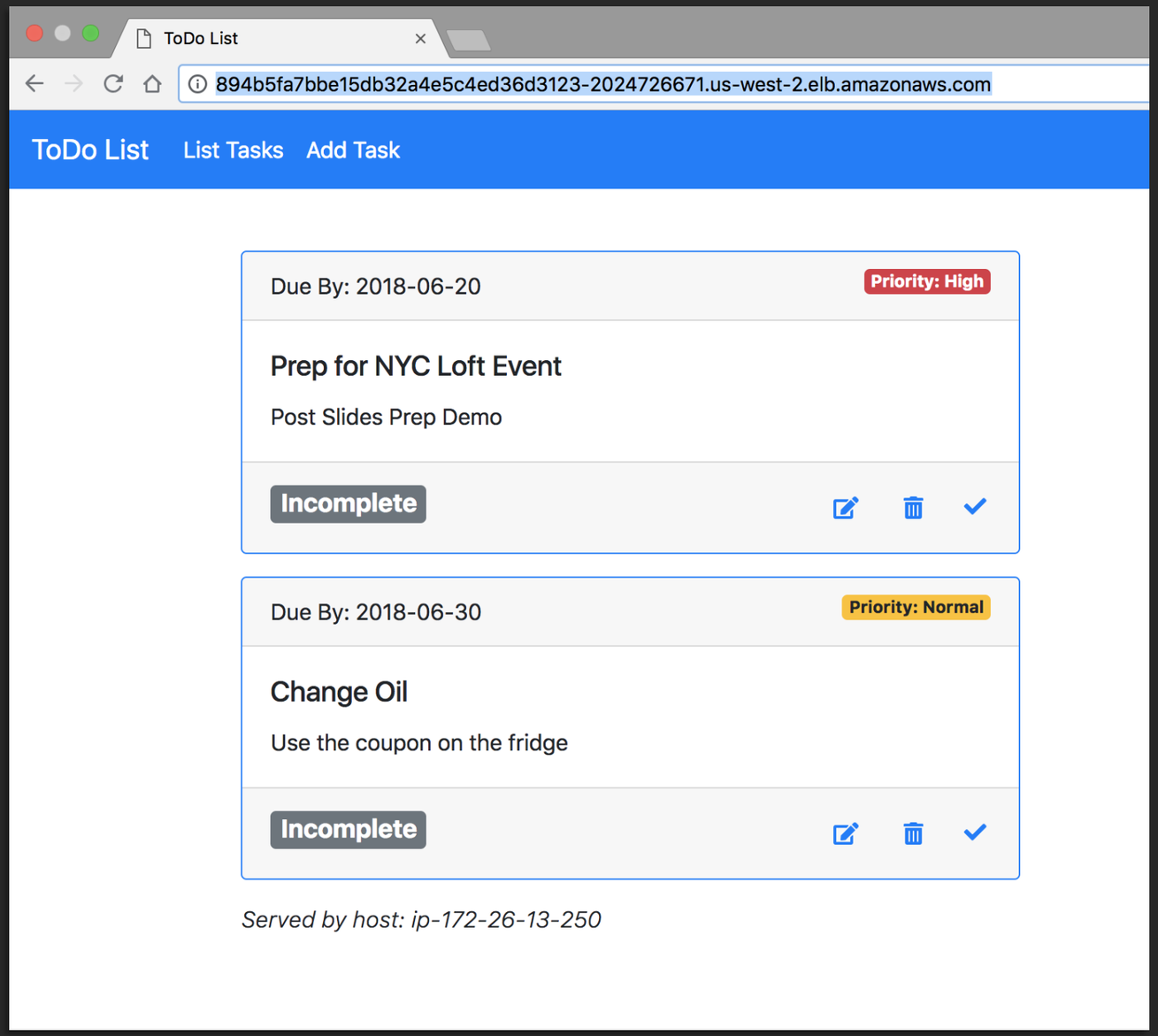
- Implement a Lightsail database

Phase 3

- Scale the web front-end

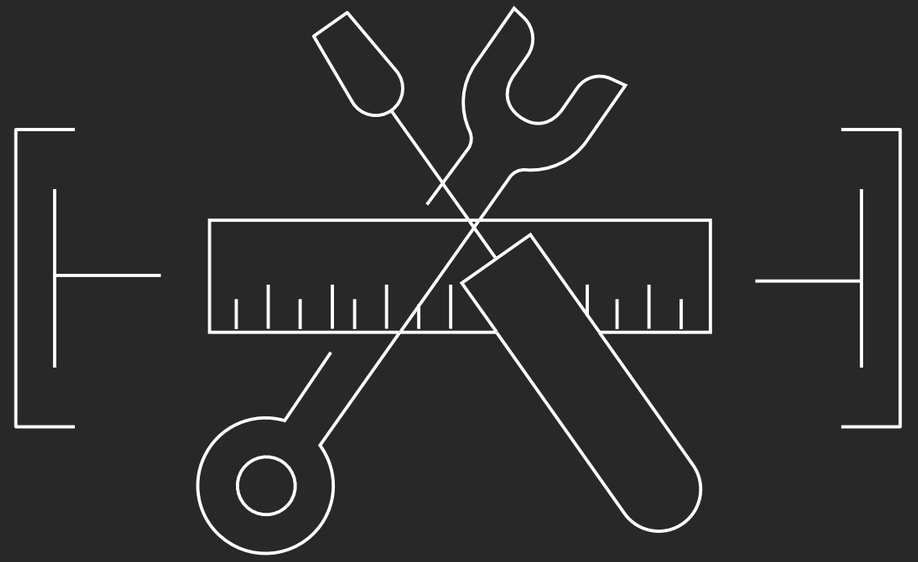
Phase 4

- Move the application to Amazon RDS/EC2

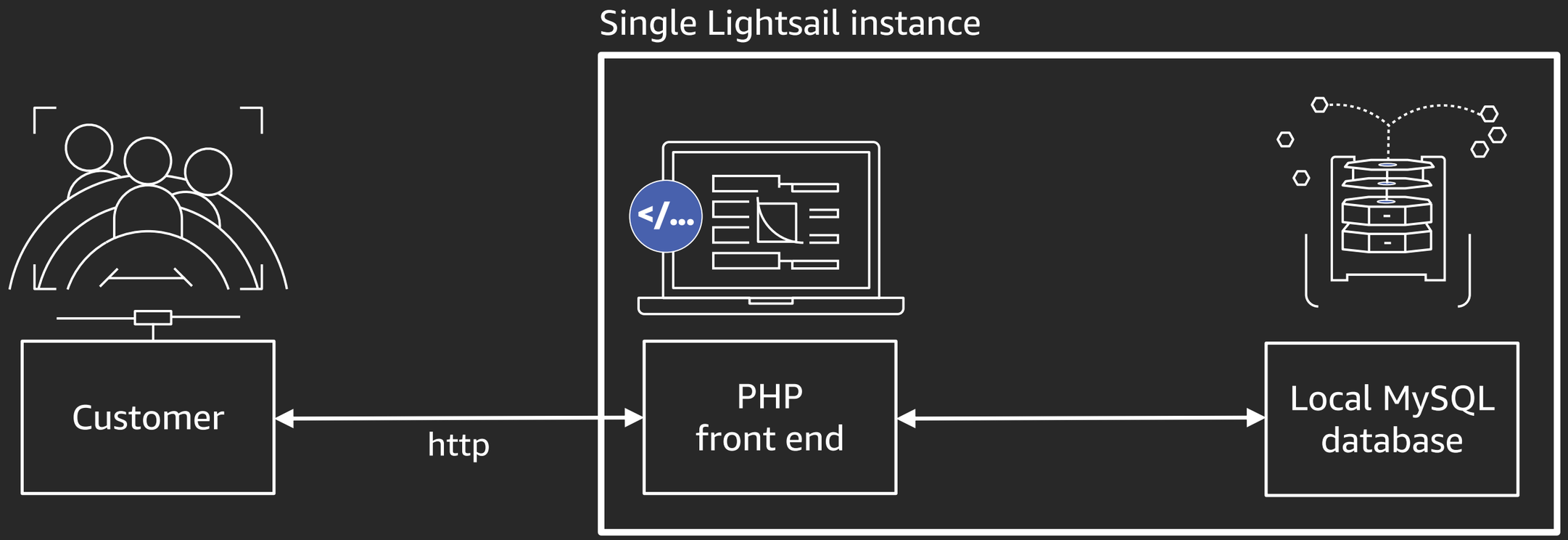


Getting our phase 1 application running

1. Create Lightsail instance
2. Clone GitHub repo
3. Configure database connectivity



Application architecture: Phase 1



Refining our application

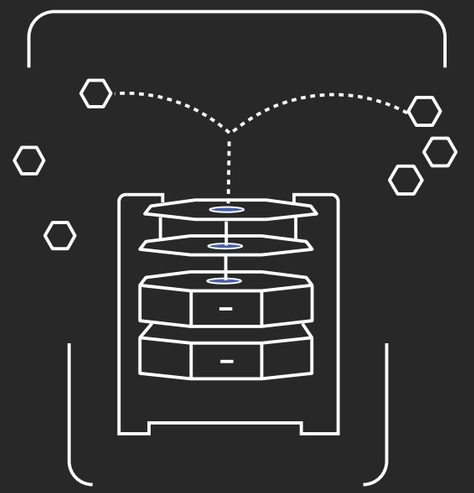
Potential issues

- Single point of failure
- Responding to increasing demand
- Database/web front end tied closely together

Solution

- Separate database/web front end
- Created multiple front-end instances
- Add a Lightsail load balancer

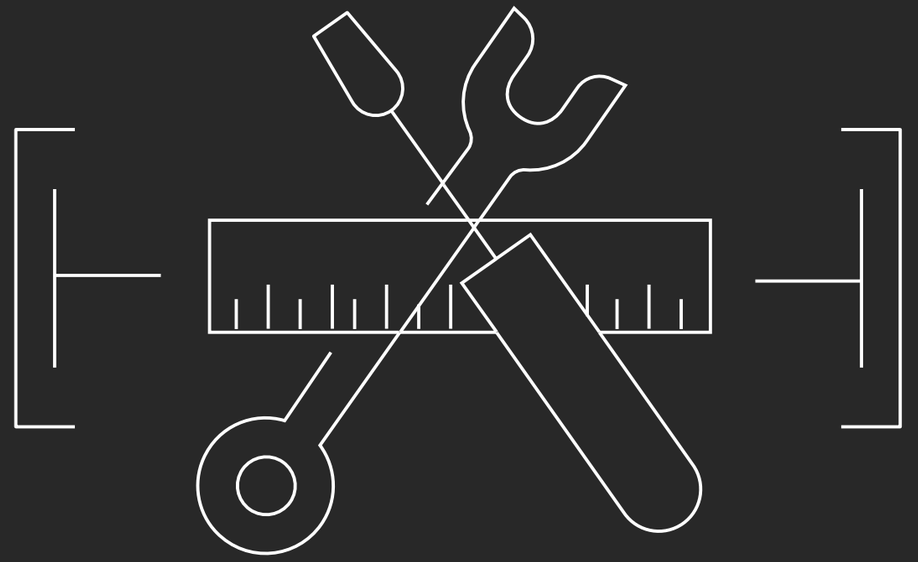
Lightsail databases



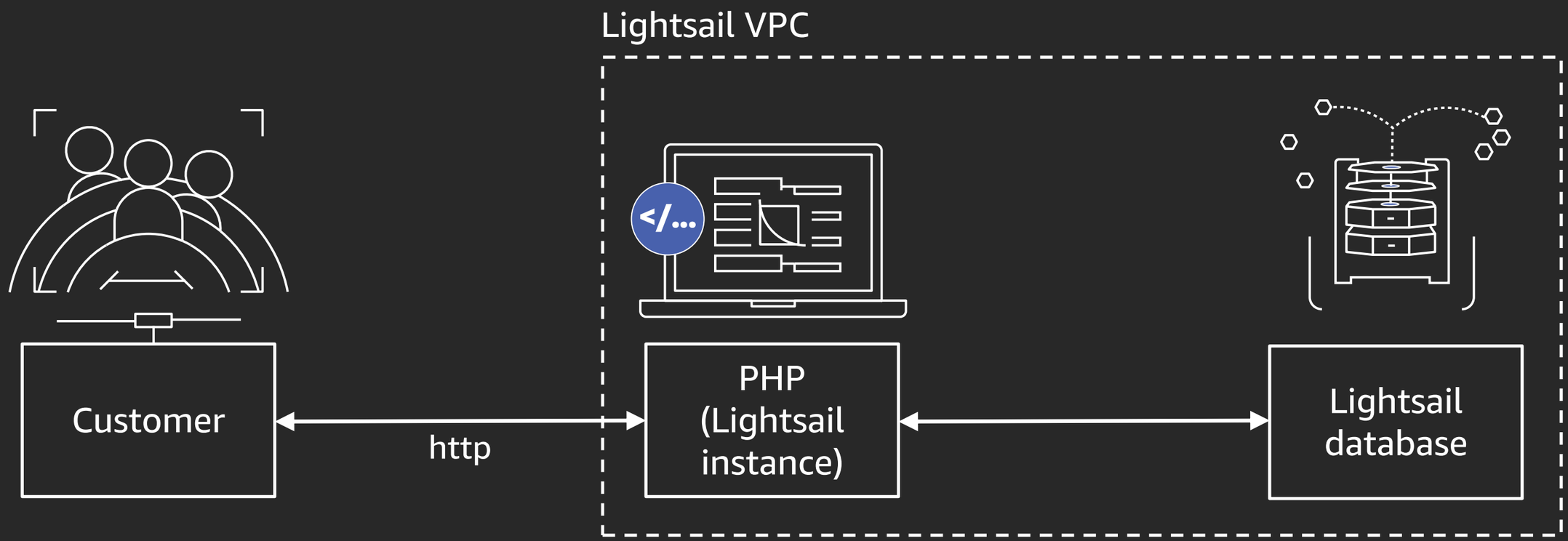
- Easy to create and manage MySQL databases
- Standard and high availability options
- Four instances sizes to choose from
- Public and private access
- Automated backups
- Fully managed—no need to manage/patch underlying system

Getting our phase 2 application running

1. Create Lightsail database
2. Reconfigure the front end to point at the new Lightsail database



Application architecture: Phase 2



Horizontal scaling with snapshots

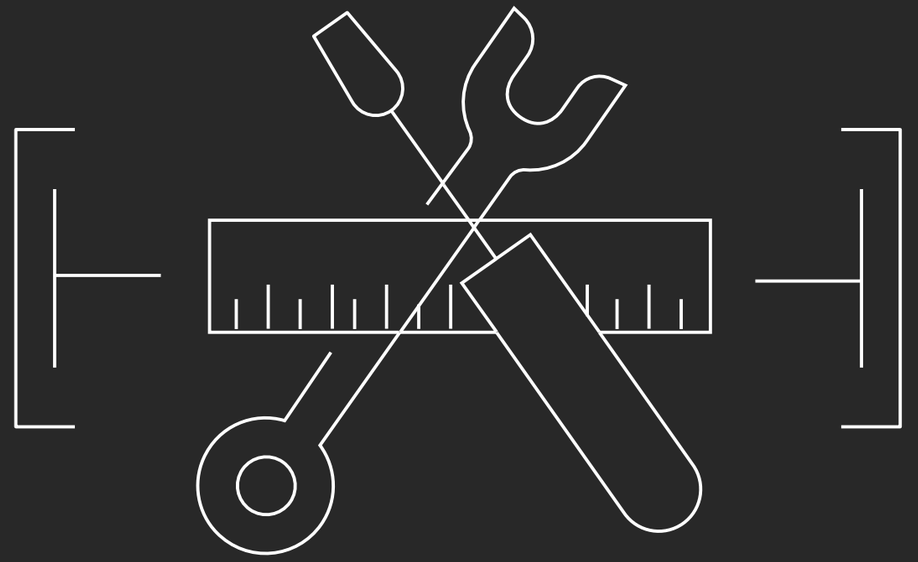
- Snapshots are point-in-time backups
- Can be used to
  - Move a server to a new region
  - Move a server to Amazon EC2
  - Upsize an existing server
  - *Replicate an existing server*

Lightsail load balancers

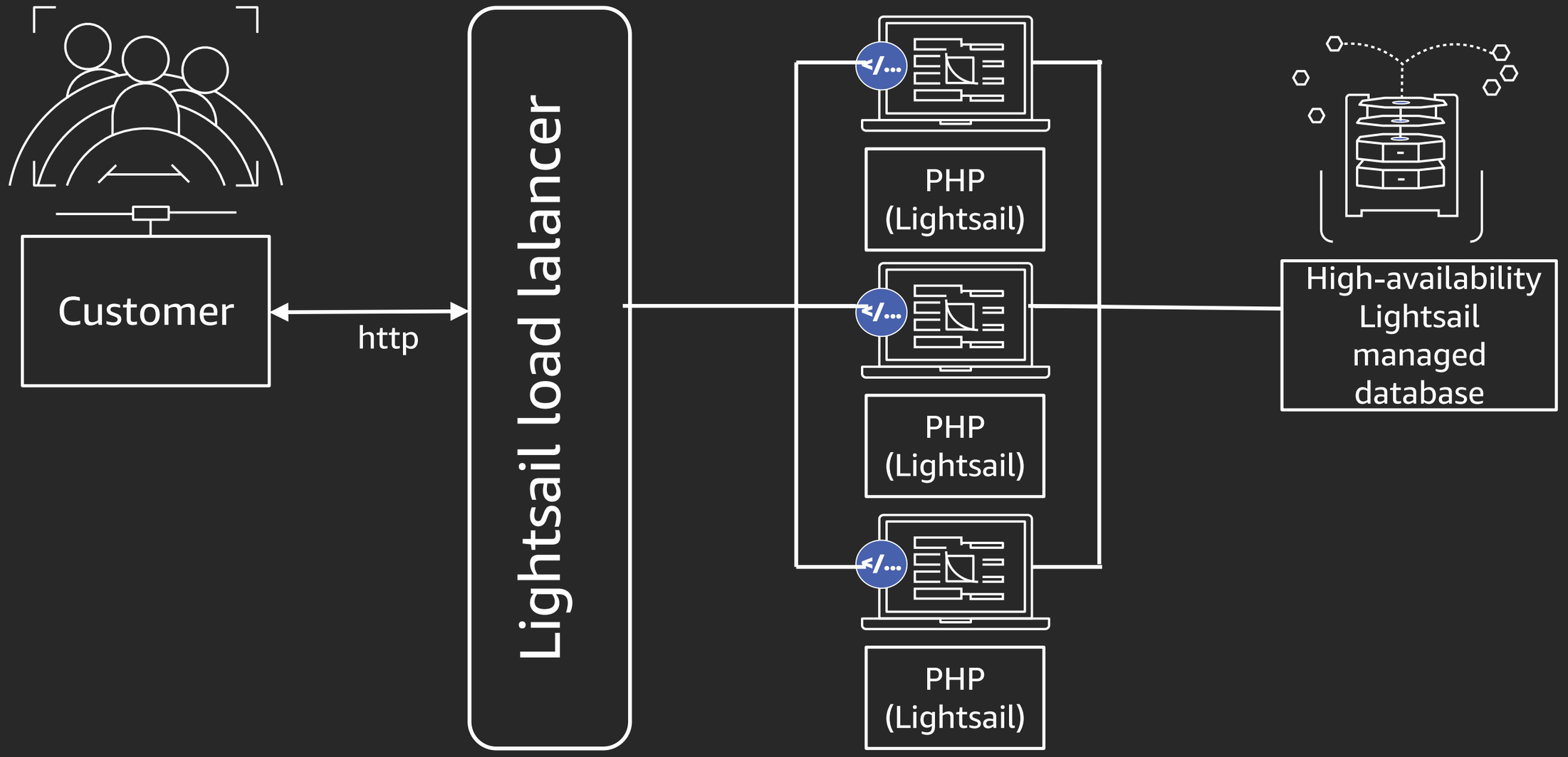
- Simplified Amazon load balancer implementation
- Set up in a few clicks
- Easy SSL certs
- Handles http/https traffic
- Balances across ports 80 and 443

Getting our phase 3 application running

1. Create a Lightsail load balancer
2. Snapshot the web front end
3. Deploy 2 new instances from the snapshot
4. Place all 3 front-end instances behind the load balancer
5. Optional: If you manage a domain, create a cname record for the load balancer



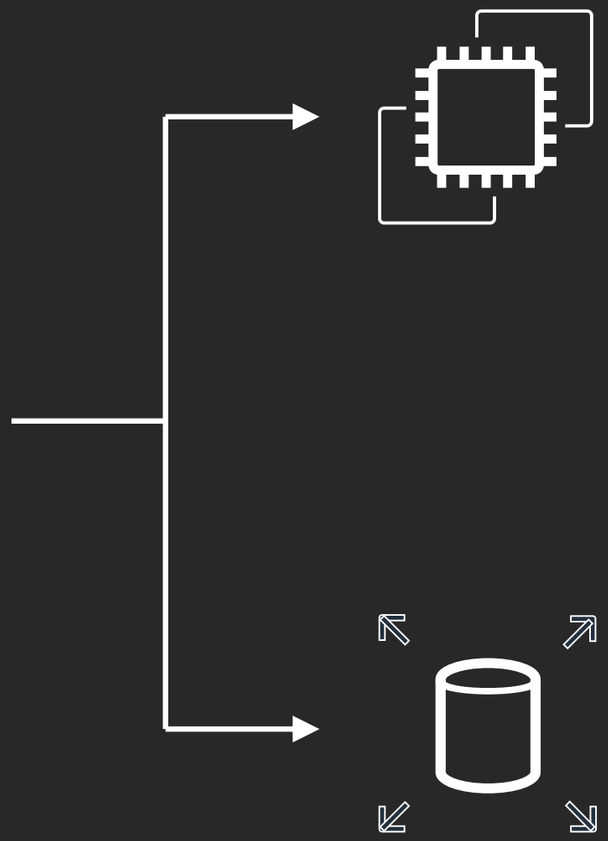
Application architecture: Phase 3



Upgrade to EC2

Move your data to EC2

- Snapshot instance or disk
- Choose export to EC2 from snapshot



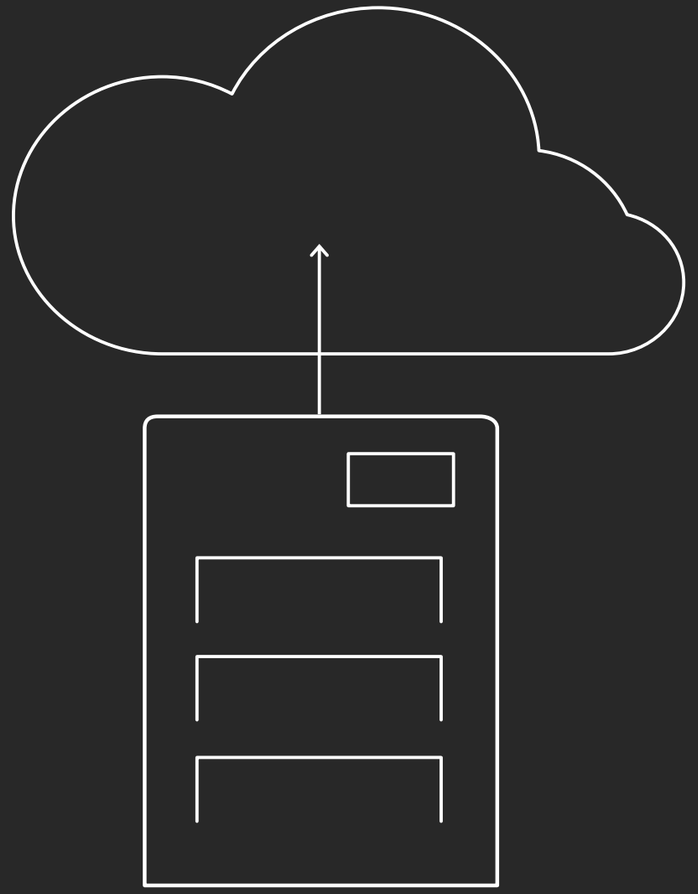
Create a new instance in EC2

- Or use EC2 export wizard (simplified setup with preconfigured VPC, etc.)
- Use EC2 creation wizard (all configuration options available)

Use EBS to rehydrate disk snapshots

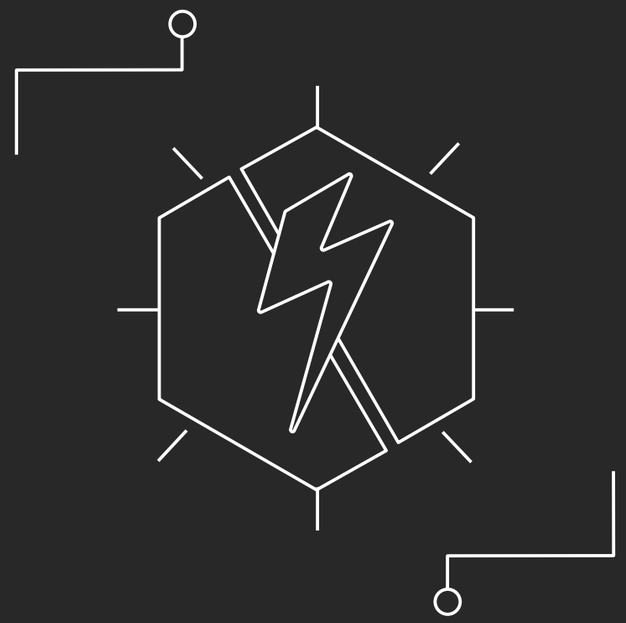
Migrating the database to RDS

1. Launch an RDS instance
2. Set security group (172.26.0.0/16)
3. Enable VPC peering
4. Migrate and verify data
5. Connect front end to RDS

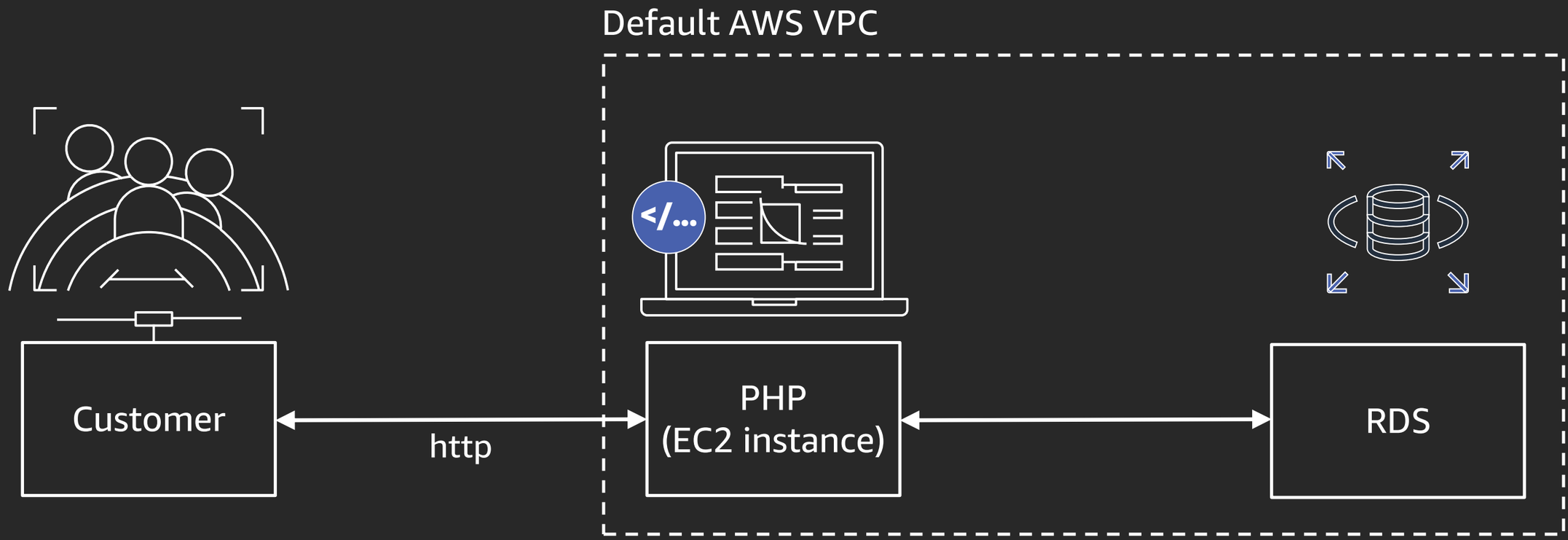


Moving your front end to EC2

1. Snapshot the Lightsail instance
2. Create new EC2 instance with export to EC2 wizard
3. Connect EC2 front end to previously created RDS database



Application architecture: Phase 4



Do-it-yourself suggestions

Deploy all your resources first

- Lightsail LAMP (PHP 7) instance
- Lightsail database
- Lightsail load balancer
- RDS service

Use the same username/password for both the Lightsail database and the RDS instance

https://github.com/mikegcoleman/todo-php ← The application

https://reinvnet.lightsailworkshop.com ← Step-by-step walkthrough

Thank you!

Mike Coleman

@mikegcoleman

AWS re:Invent

aws

Please complete the session survey in the mobile app.