Reliable Software Systems

Week 1: Reliability? Systems?

January 10, 2018
Alyssa Pittman
University of Washington Allen School

Example outage: Maersk

NotPetya virus brought down all their computers

They couldn’t move any shipments for 2 days

It took about two weeks for operations to normalize

They estimated $300 million losses

Not to mention the impact on their customers

https://pivotts.com/maersk-attack-illustrates-enormous-cost-downtime/

reliable

adjective

re·li·able | \ri-ˈlī-ə-bəl \

Definition of *reliable*

(Entry 1 of 2)

1: suitable or fit to be relied on : DEPENDABLE

2: giving the same result on successive trials

https://www.merriam-webster.com/dictionary/reliable

Reliability from the user's perspective

Available

Fast

Consistent

Correct

Durable

Secure

Reliability from the developer's perspective

Resilient when faced with...

- Failures
- Change

Scalable when faced with...

- More usage
- More data

Why do companies care?

Money

Their customers’ money

Customer trust

Employee morale

Why should you care?

Everything is interconnected

You may be oncall

Ownership!

Why would people *not* care?

Features!

Moving fast!

Brief history of ops

1944 - Colossus, the first programmable computer, in use for WWII.

1952 - IBM’s 701, the first commercial computer, is announced.

1969 - UNIX is born, the first multi-user OS.

1980’s - The role of sysadmin develops.

1993 - Windows is born.

2000’s - The rise of the internet and web systems - “throw it over the wall”

2010’s - DevOps, SRE, oncall software engineers.

Adapted from “A Brief History of System Administration”,
from UNIX and Linux System Administration Handbook, 5th Edition

Service Oriented Architecture

A way of designing software that is oriented towards breaking the problem space into independent pieces which interact using a communication protocol over the network.

A service:

- Is a representation of a repeatable business activity that has a specified outcome
- Is self-contained
- May be composed of other services
- Is a “black box” to consumers of the service

Remote Procedure Calls (RPC)

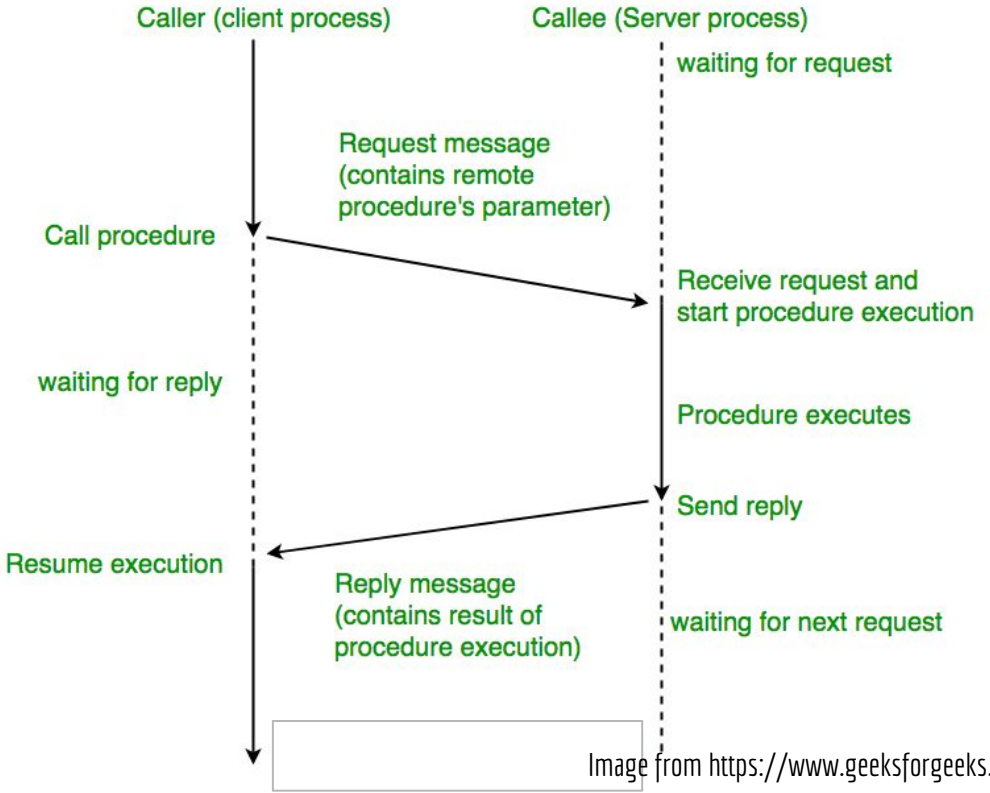


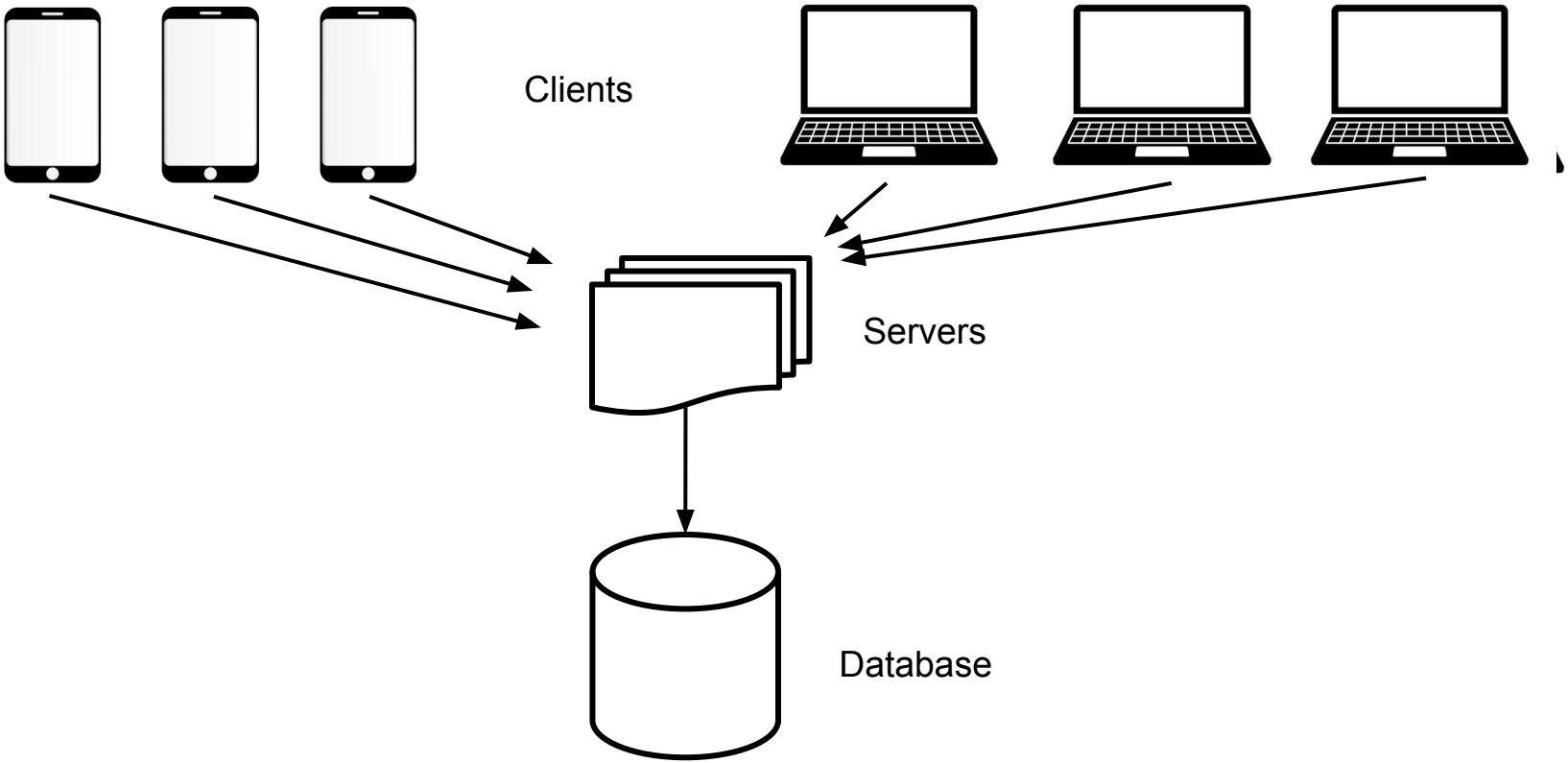
Image from https://www.geeksforgeeks.org/operating-system-remote-procedure-call-rpc/

Remote Procedure Calls (RPC)

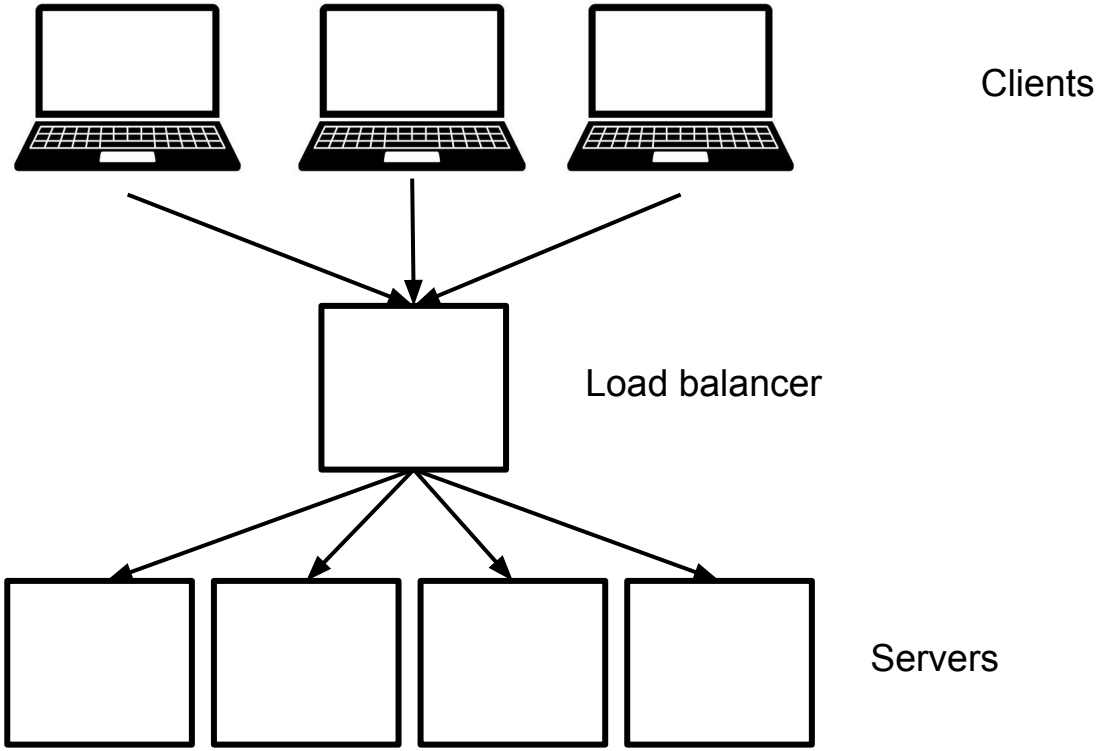
Client requests to execute a function on the server

- Client invokes a “stub”
- Client needs to know how to connect to the server
- Data gets marshalled to the underlying transport protocol
- Packets get sent to the server
- Server unmarshalls data and processes request
- …and reverse to get the response back to the client

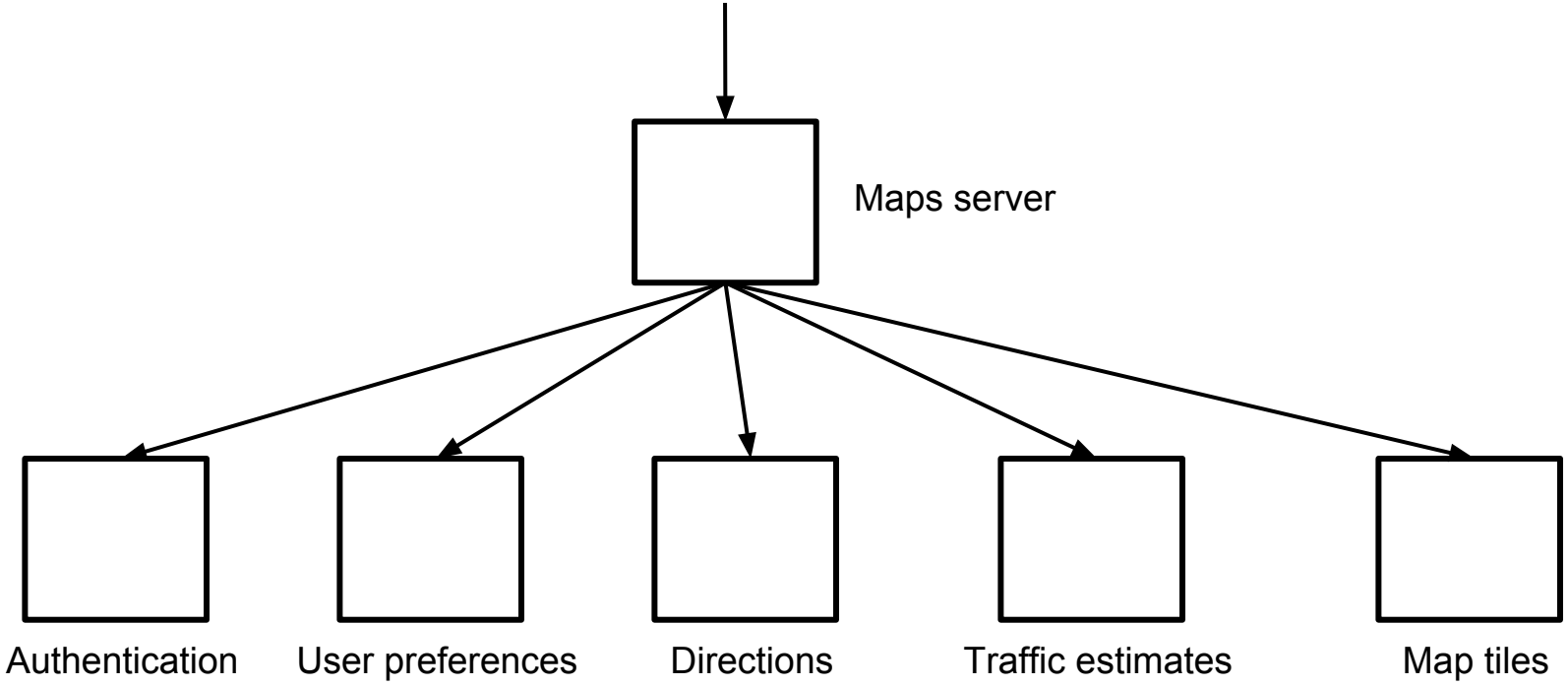
Systems today



Systems today



Systems today



Systems today

Google Compute Engine Datacenter locations

END