SWEN438 - DevOps

Lecture 11 - Telemetry

Dr David J. Pearce

School of Engineering and Computer Science
Victoria University of Wellington

Telemetry

"Two hours into the outage, we discovered it was caused by one of the our networking vendors who **accidentally** *made a change to our production phone system instead of the hot spare.*

The outage will impact our **quarterly revenue***, but we don't know how much yet. In order to prevent this from happening again, we're putting together a project to monitor our critical systems for unauthorized changes."*

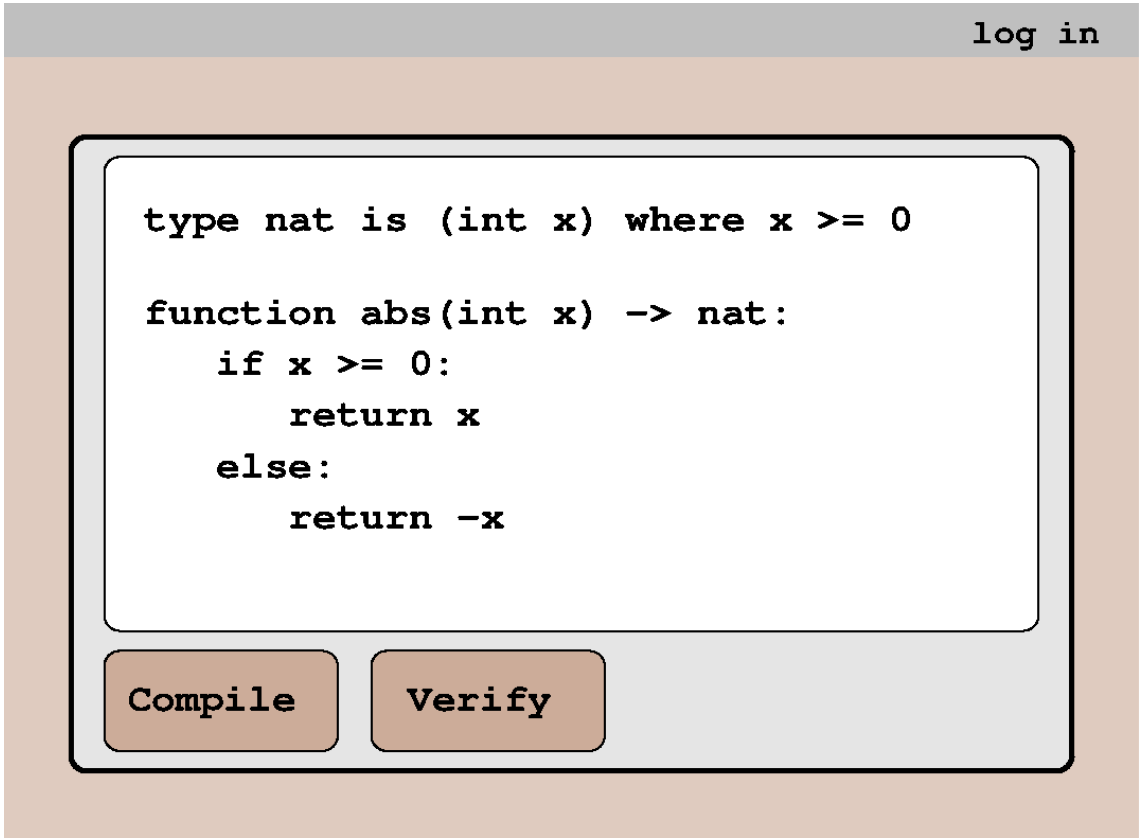
–The Phoenix Project

Telemetry

“an automated communications process by which measurements and other data are collected at remote points and are subsequently transmitted to receiving equipment for monitoring”

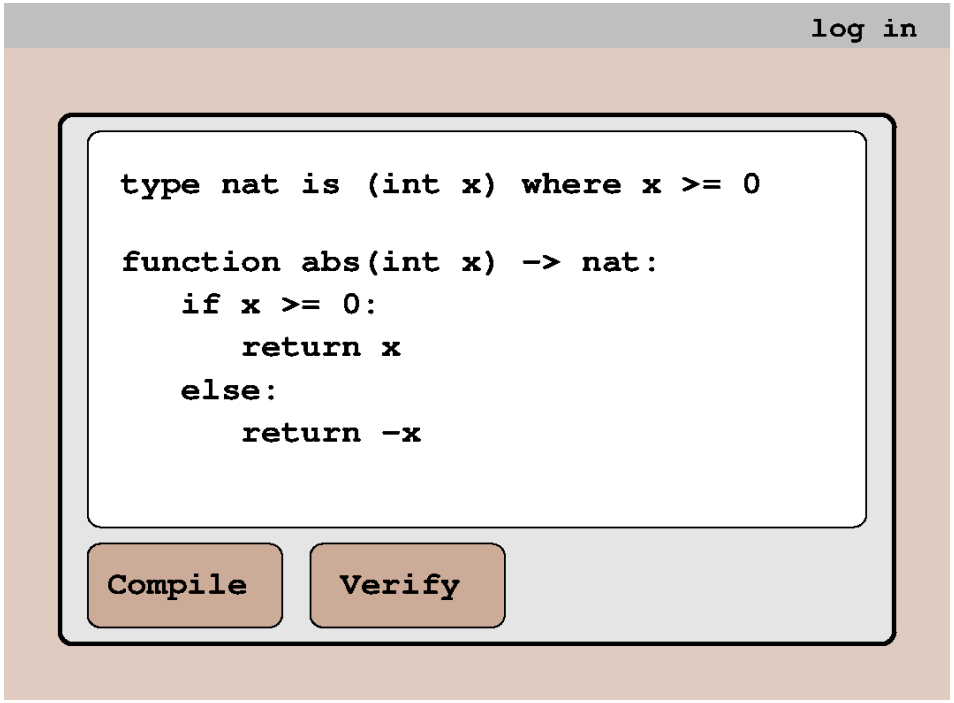
–The DevOps Handbook

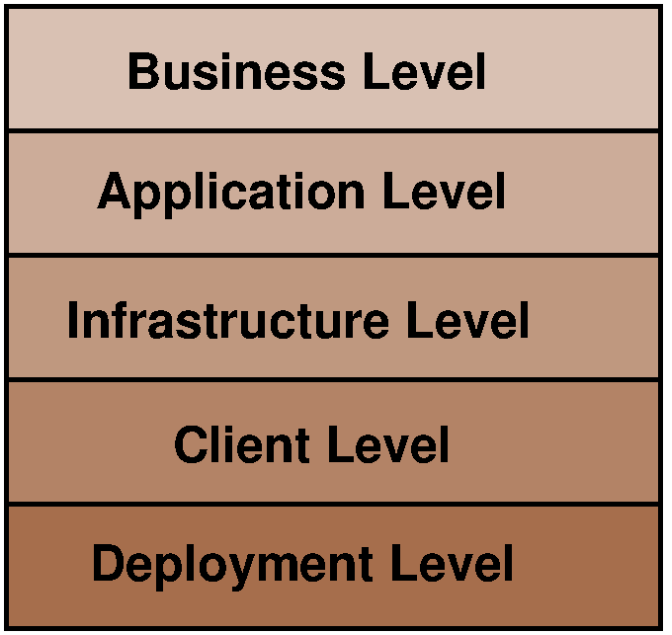
Telemetry



- What **key metrics** can we record?

Levels of Metric





- How do our metrics fit with **business organisation**?

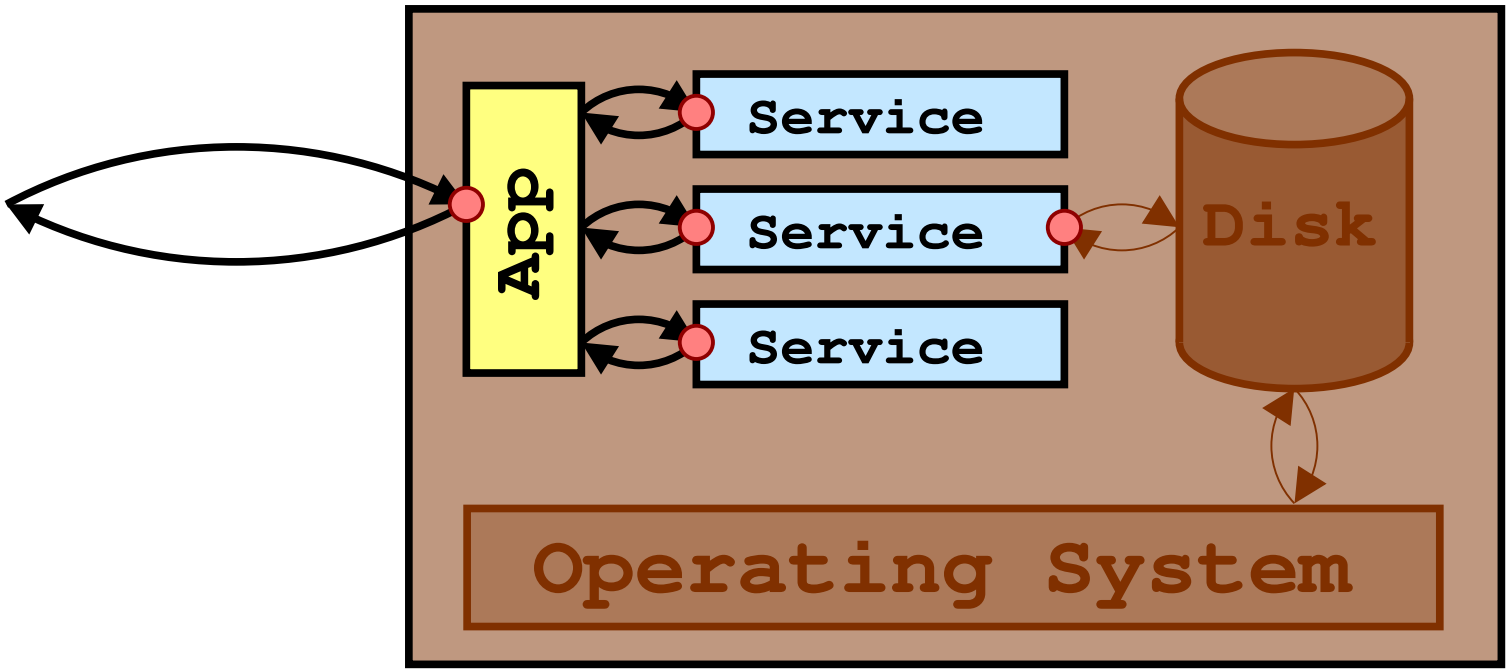
Levels of Metric (Cont'd)

- **Business Level**. E.g. account creations, sales revenue, etc.
- **Application Level**. E.g. time to process request, fault rate, average program size, rate of compile failures, rate of verification failures, etc.
- **Infrastructure Level**. E.g. server traffic, number of concurrent connections, CPU load, memory usage, etc.
- **Client Level** E.g. frequency of interactions, types of interactions, etc.
- **Deployment Level**. E.g. deployment frequencies, lead time for changes, etc.

Goals of Telemetry

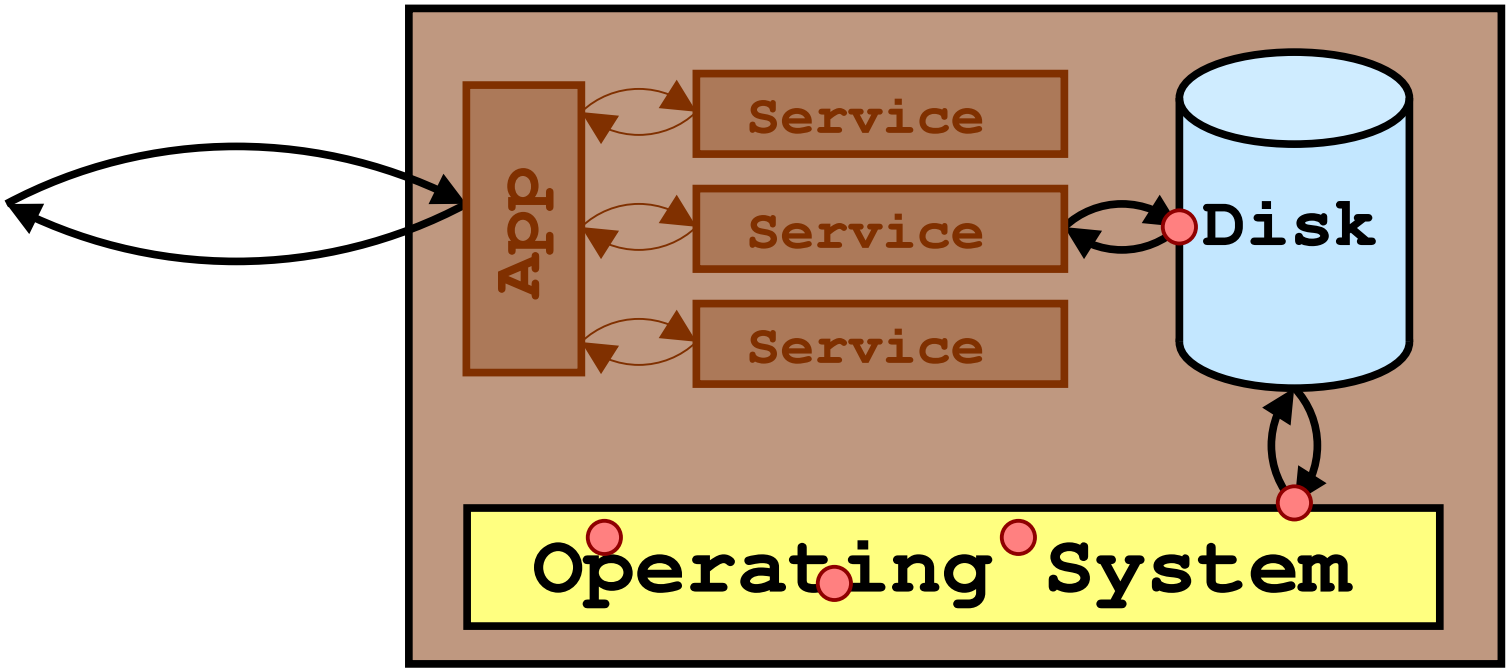
- **Identify** outage or service degradation.
- **Identify** non-functional concerns (e.g. unauthorized activity).
- **Identify** problematic deployments
- **Provide** information for diagnosis and debugging.
- **Provide** information for data analysis (e.g. A/B Testing).

White Box Monitoring



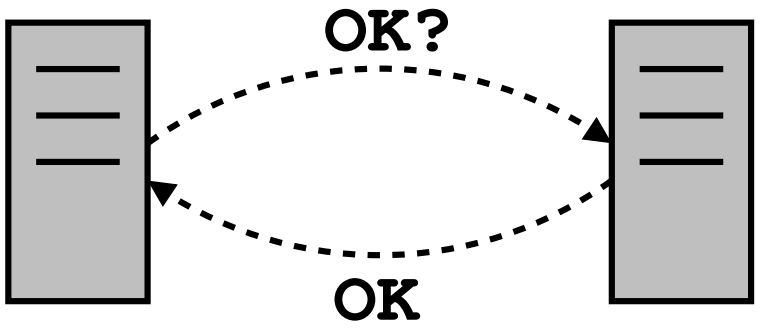
- Monitoring **with** knowledge of application(s) internals
- Can **add** instrumentation to emit telemetry as necessary

Black Box Monitoring



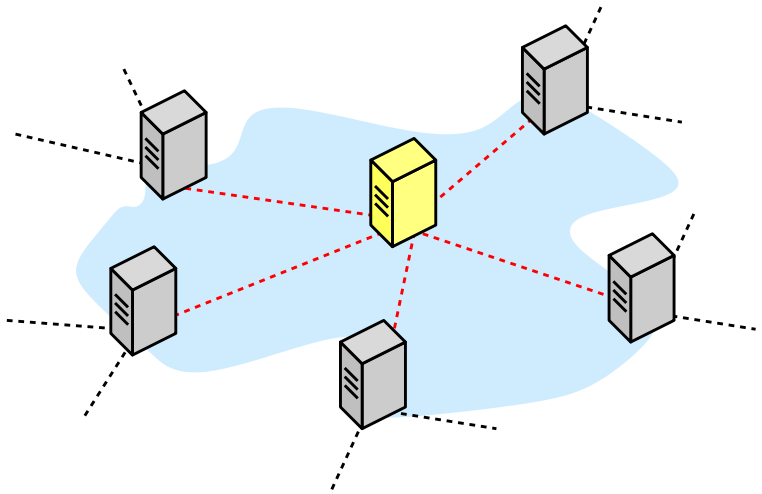
- Monitoring **without** knowledge of application(s).
- May have access to information provided through API
- May have access to server information (e.g. CPU Load)

Health Checks



- Periodically **probe** to check system alive
- Can use **non-specific** probes (e.g. ping or HTTP)
- Can use **application-specific** probes

Content Delivery Networks (CDN)

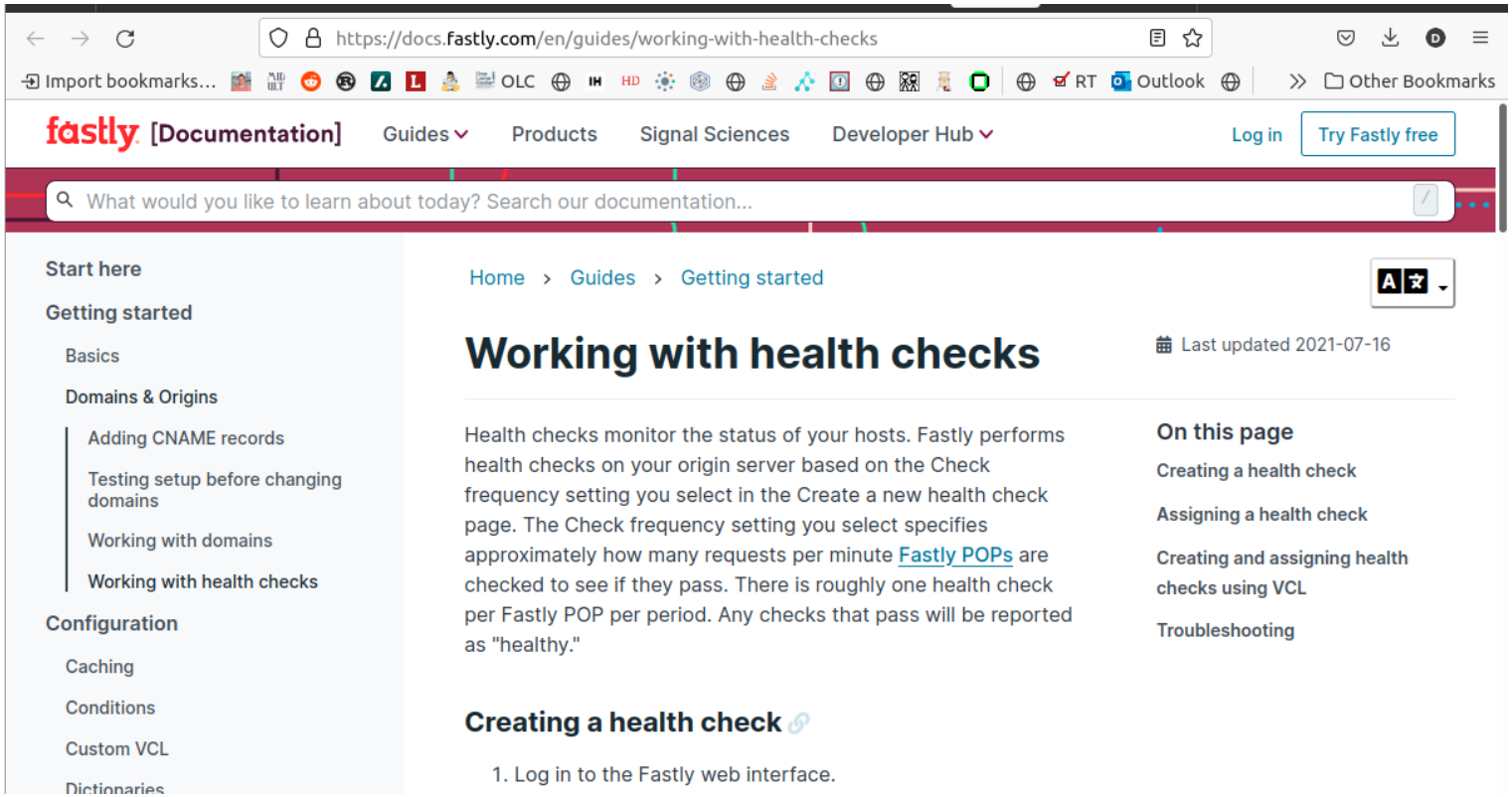


"A **content delivery network**, *or content distribution network (CDN), is a geographically distributed network of proxy servers and their data centers. The goal is to provide high availability and performance by distributing the service spatially relative to end users."*

–Wikipedia

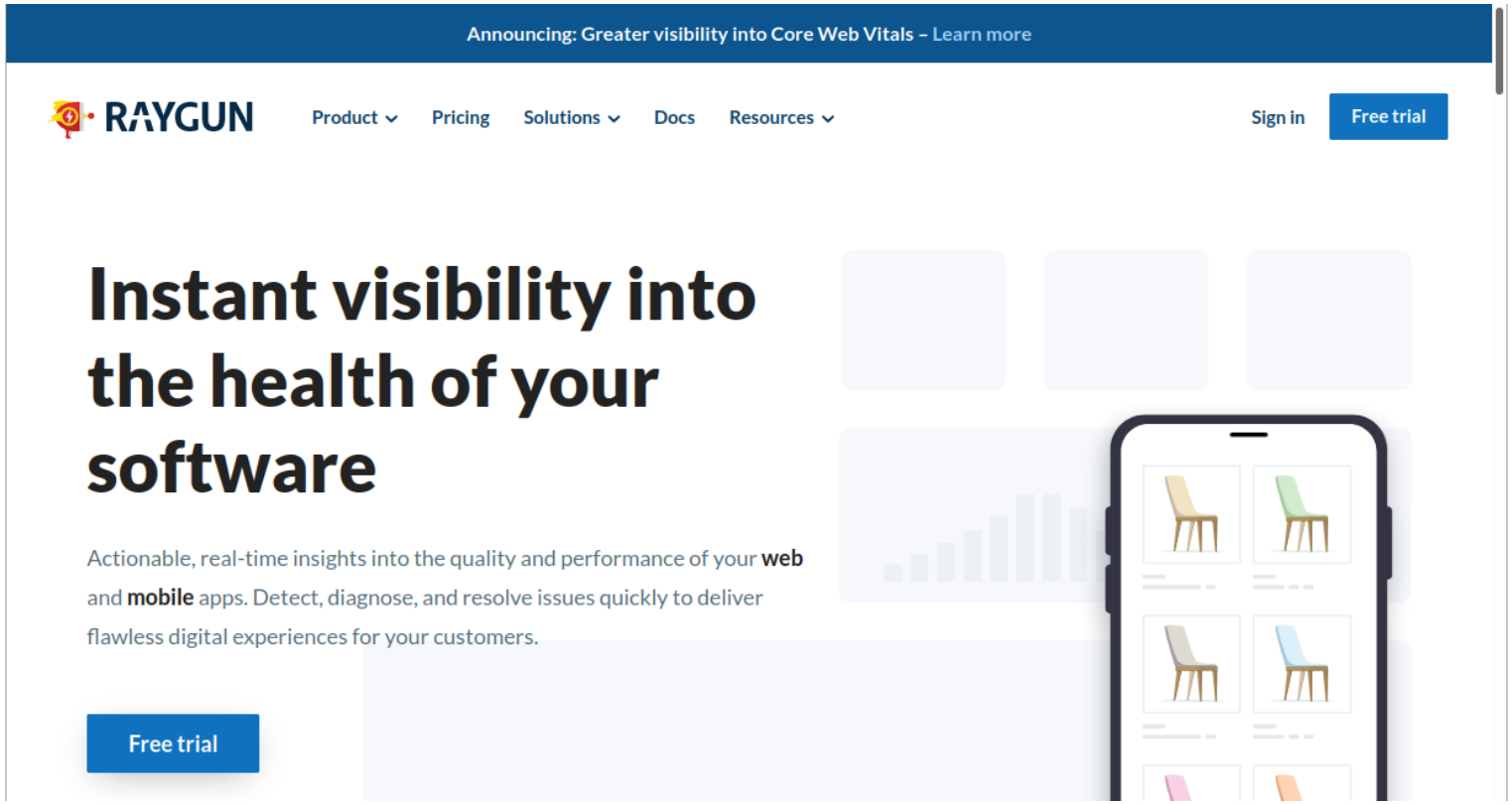
- **Examples:** *Fastly*, *CloudFlare*, etc.

Content Delivery Networks (CDN)



- CDN's are well positioned for **health checks**!

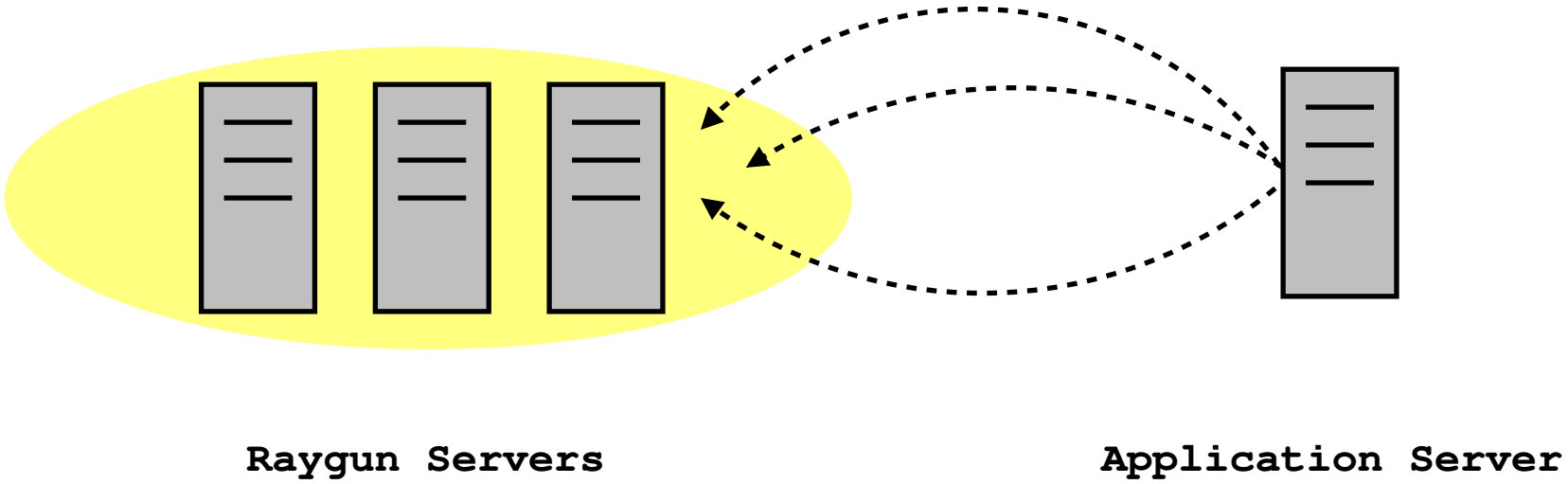
Case Study: Ray Gun



- **Wellington-based** Company!

- Offers **error monitoring & crash** reporting tools

Case Study: Ray Gun



- Code (e.g. Java) **embedded** within application
- Embedded code transmits exceptions in **real time**

Case Study: Ray Gun

```
RaygunClient client = factory.newClient();
client.setUser(user);
client.tag("blue").withData("userseg", "dev");
client.recordBreadcrumb("hello␣world");
...
client.send(anException);
```

- Straightforward client available for Java applications
- Developers add this to their code base