Como a plataforma de buscas do iFood funciona

Lucas Viecelli


PGConf.Brasil 2019
1º A 3 DE AGOSTO
SÃO PAULO - SP
PALESTRAS | TUTORIAIS | LIGHTNING TALKS
+ 1 DIA DE WORKSHOPS
PGCONF.COM.BR

Um pouco sobre o iFood

Infraestrutura

Alguns números:

+900

Instâncias no pico

+90

Instâncias de DB

99%

Automatizado

Por que não o RDS?

- Sem acesso a todos os parâmetros do servidor, OS e database.
- Sem acesso remoto do host via ssh.
- Sem acesso ao super user.
- Não se pode instalar outras aplicações no host do banco, Ex: pgbouncer, pgpool.
- Falta de visibilidade

Então como fizemos nosso próprio RDS?

- Com configuration manager.
- Com infrastructure as code.
- Com shell script.
- Com scripts em python.

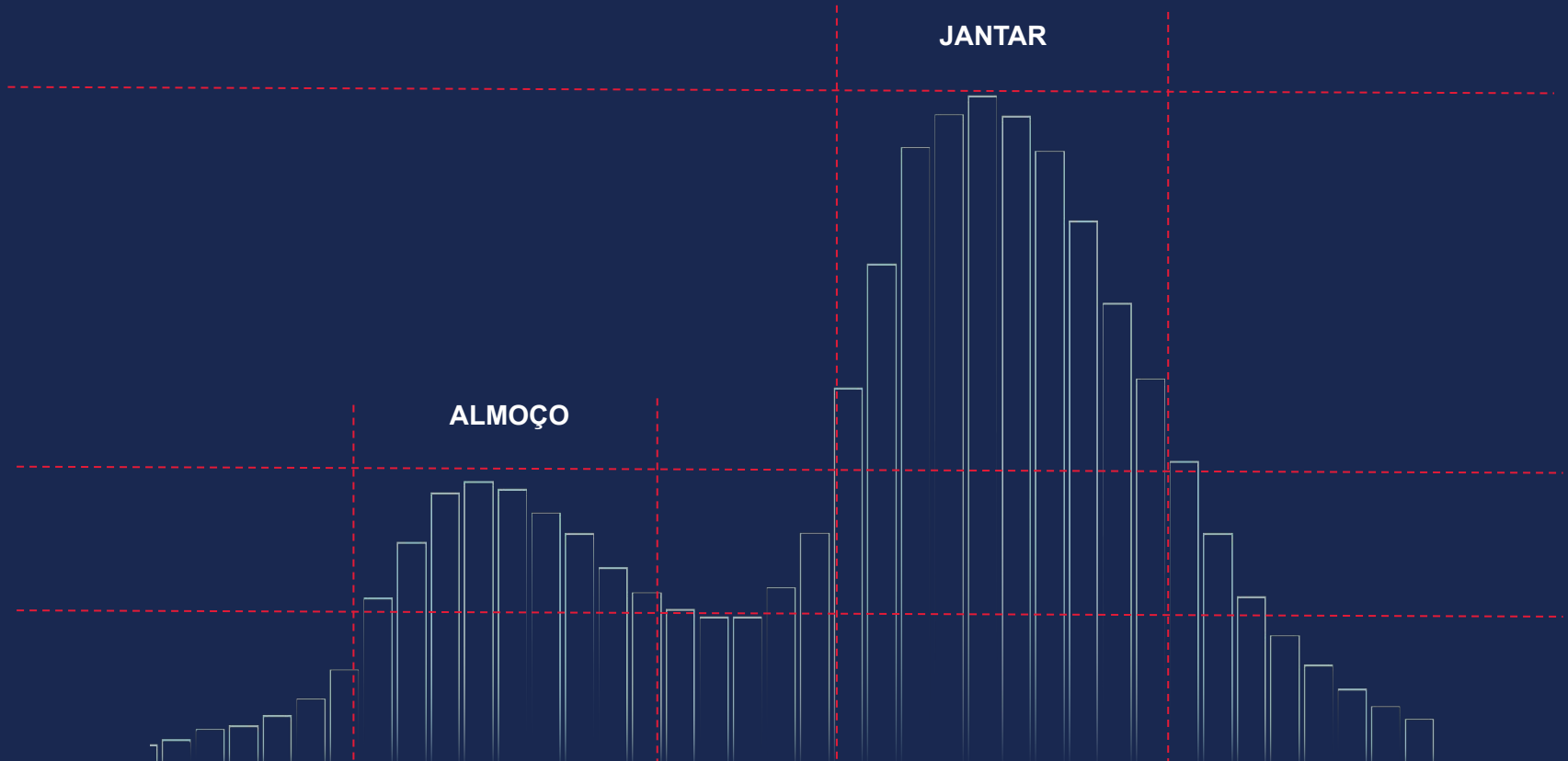
CHEF

HashiCorp Terraform

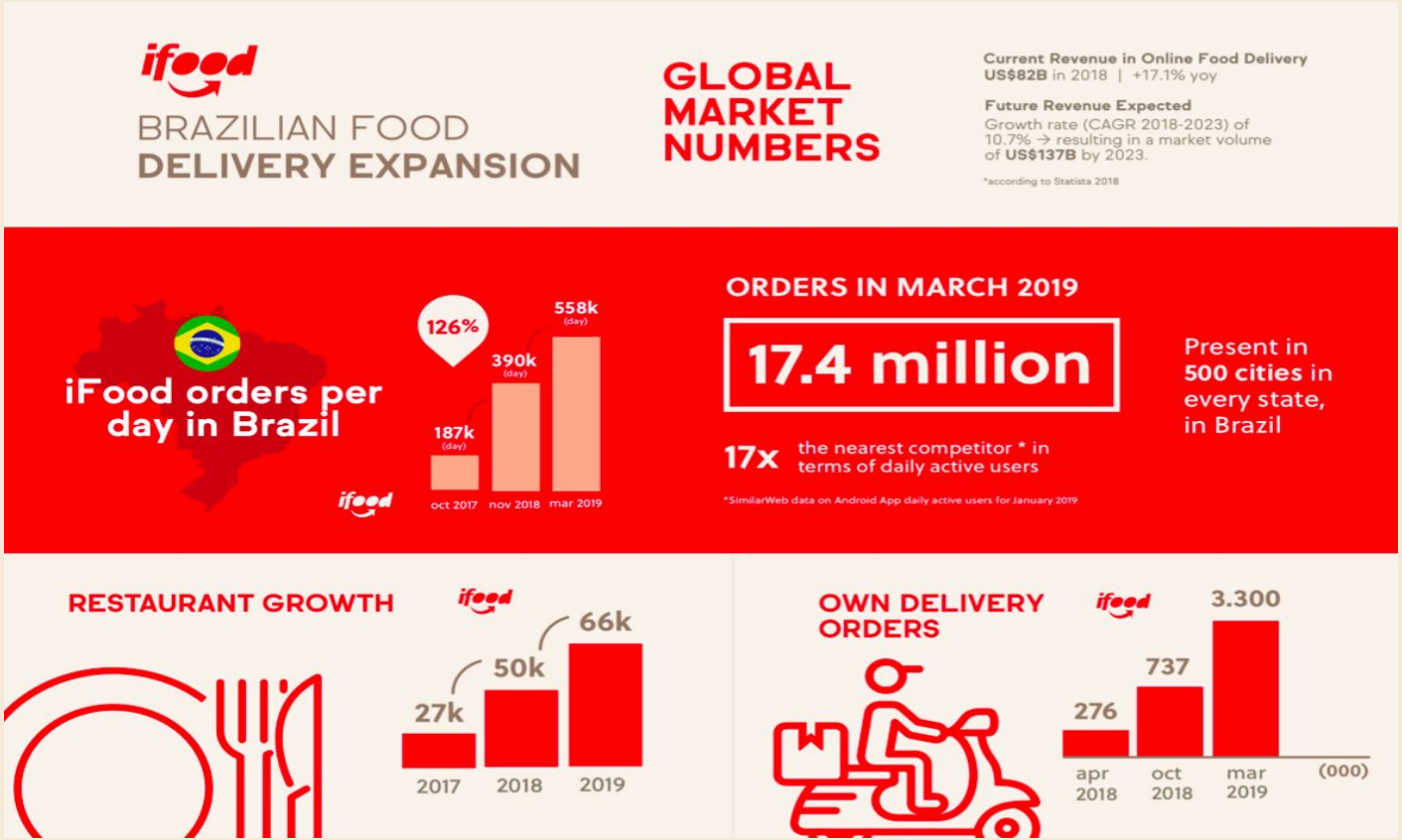
```
#!/bin/bash
```

python™

Infraestrutura

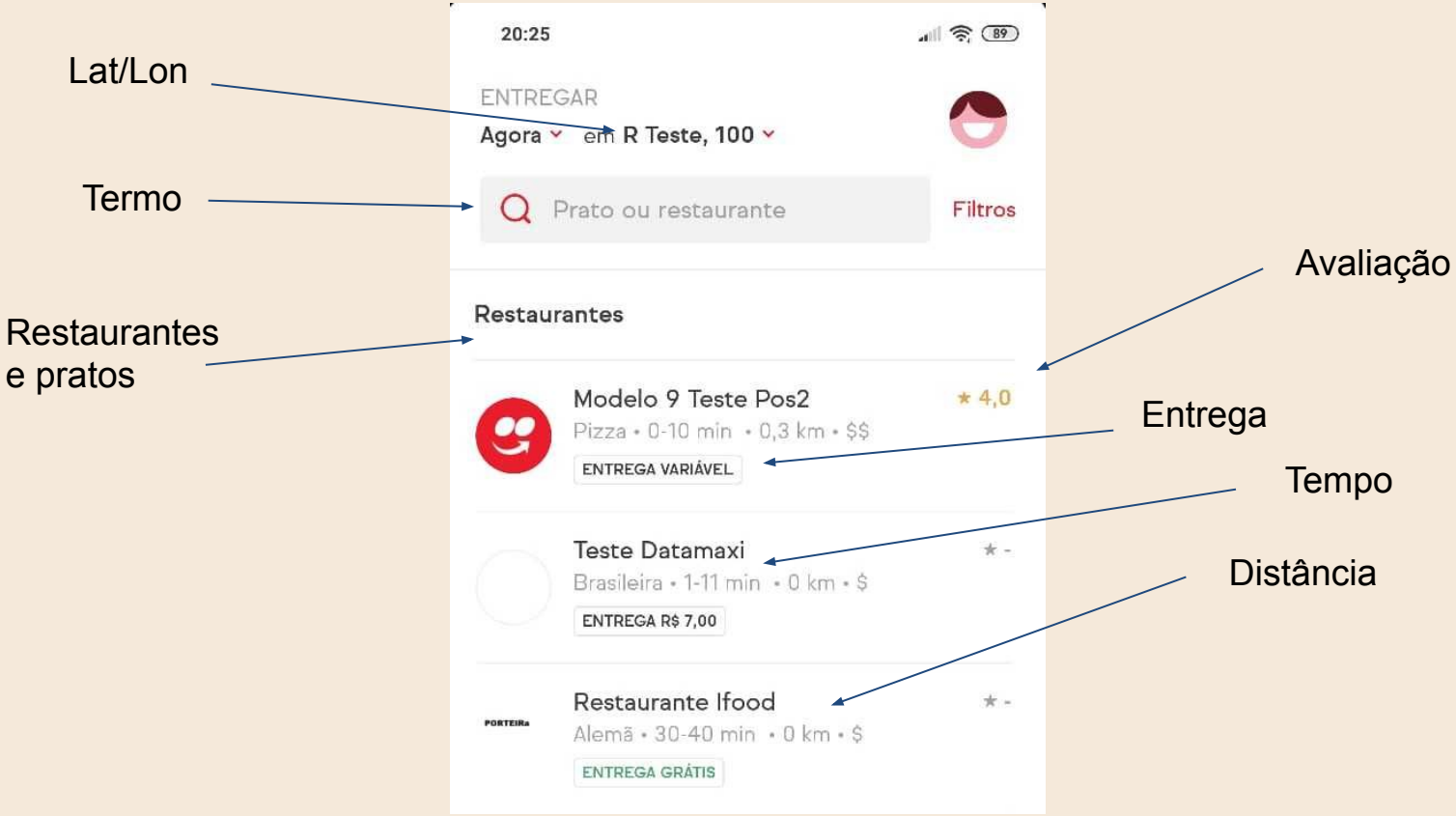


Crescimento...

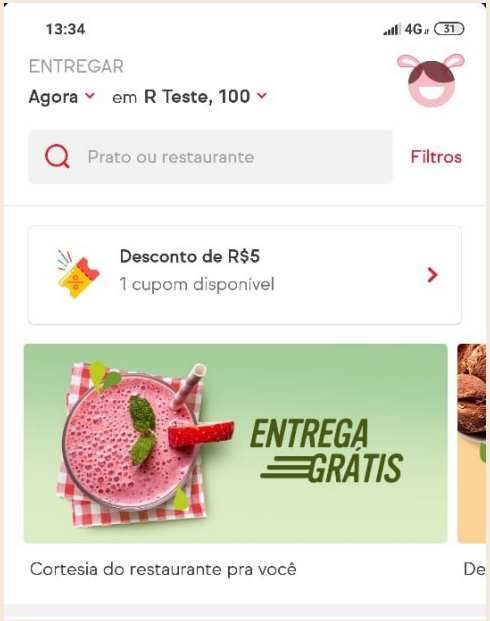


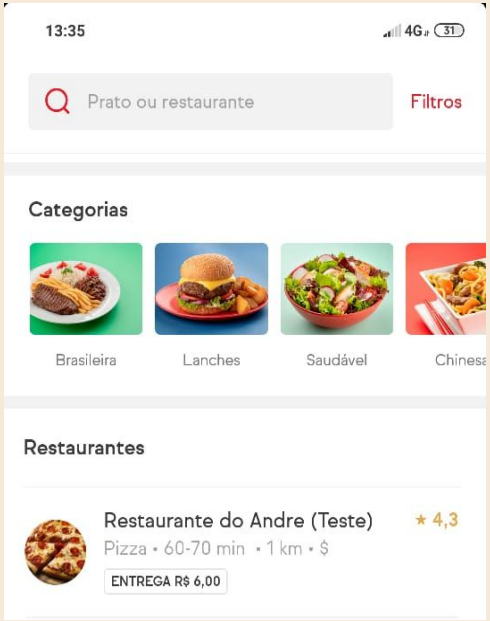
Plataforma de Busca

O problema: dados de vários serviços integrados na mesma tela

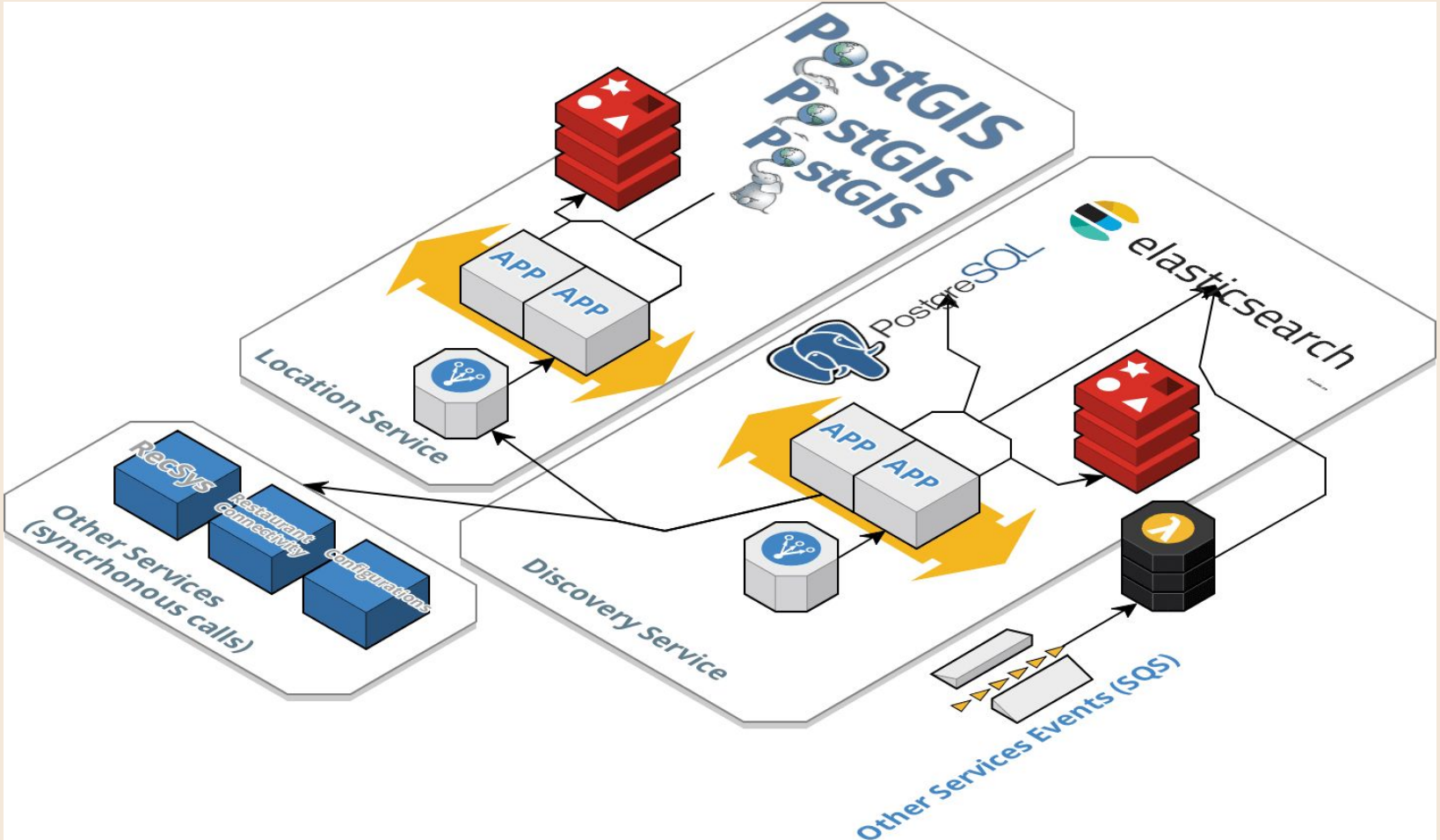


Usamos essa estrutura em várias partes da app





Solução antiga

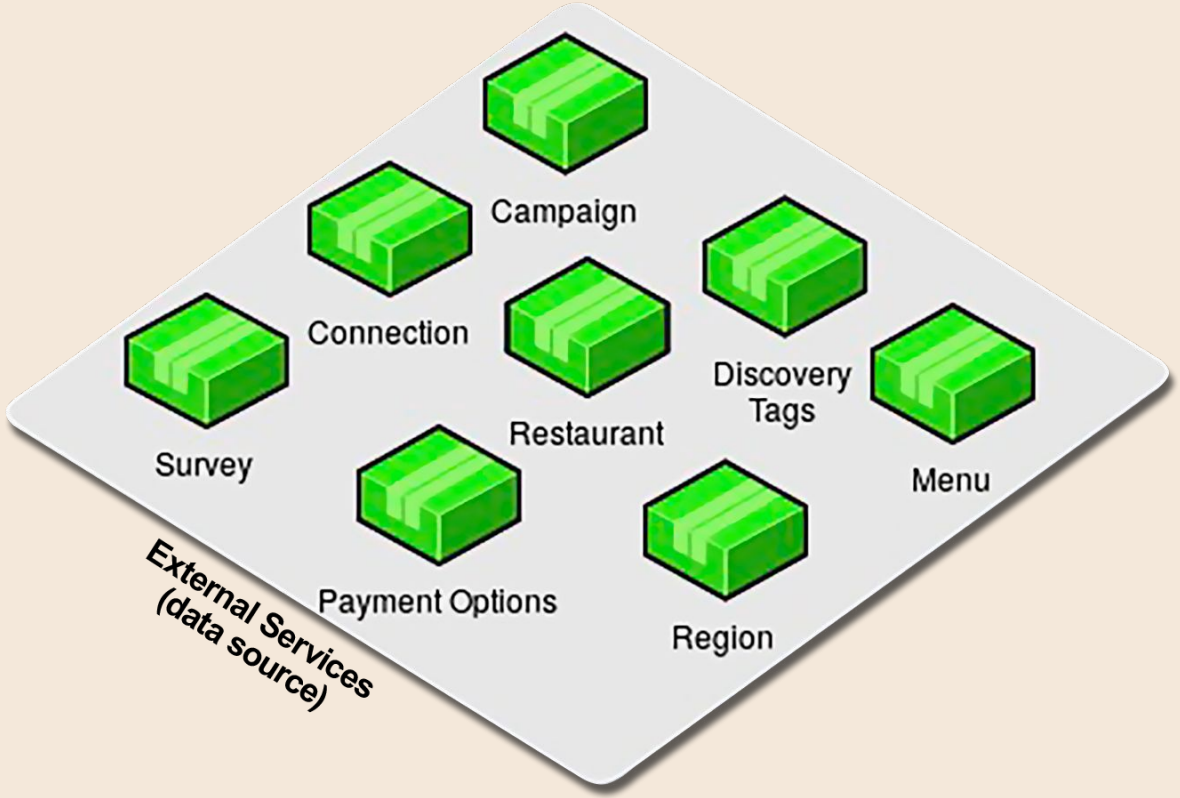


Problemas com essa solução:

- Sincronização dos dados complexa
- Difícil garantir consistência
- Tempo de sincronia alto
- Dificuldade em gerenciar caches entre as várias camadas
- Dependência de outros serviços (que poderiam não ser críticos, como o "Location Service"), dificulta manutenção
- Escalabilidade

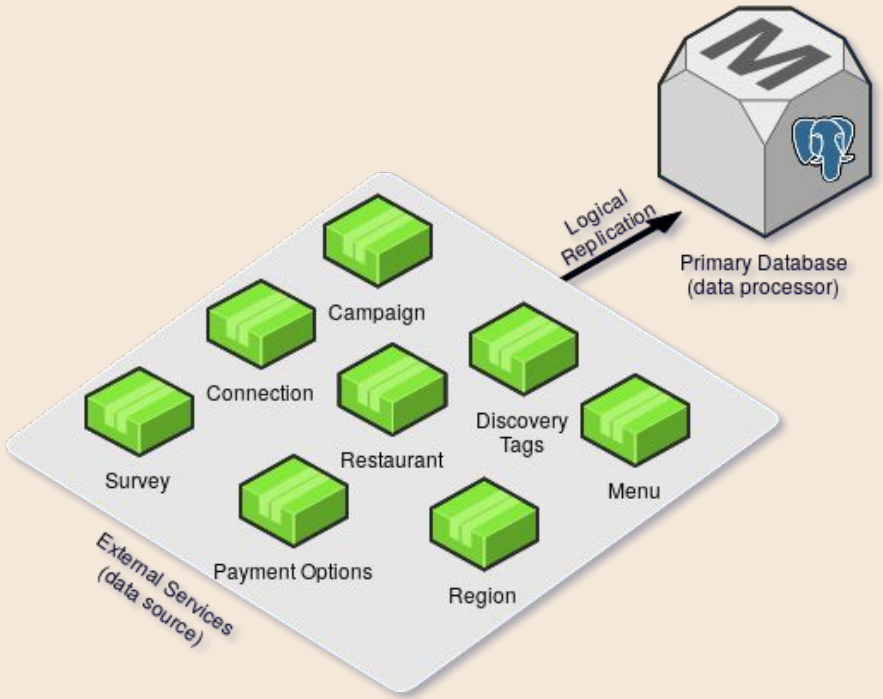
Solução

Origem dos dados



Extração dos dados

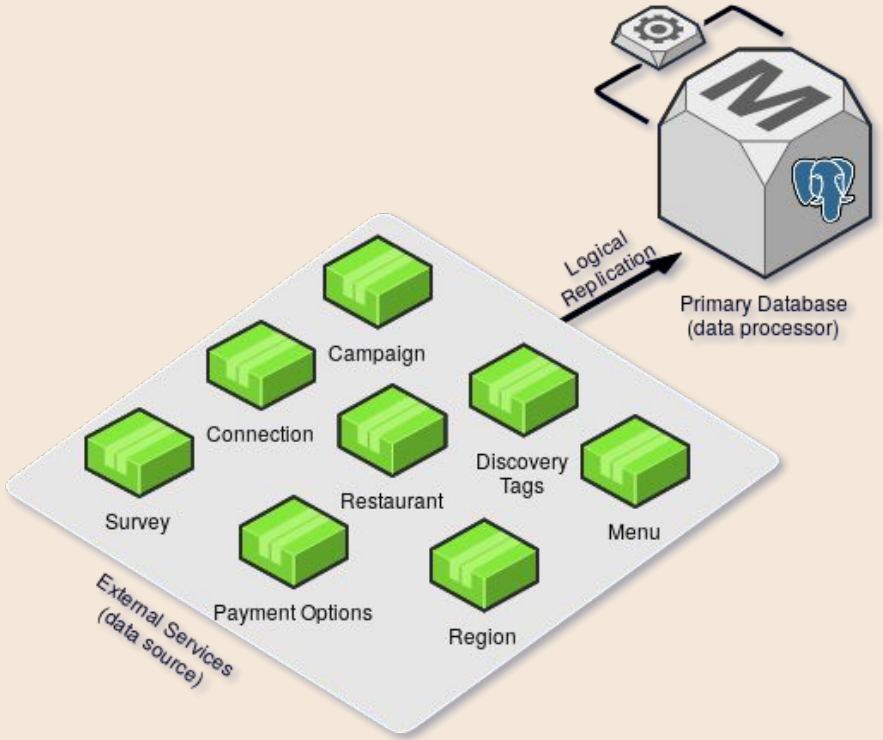
Replicamos cada uma das bases PostgreSQL para uma banco central. Cada um usando um schema, usando replicação lógica.



- Esse banco central tem uma carga maior de dados I/O
- Não é acessado por nenhuma aplicação

Processamento dos dados

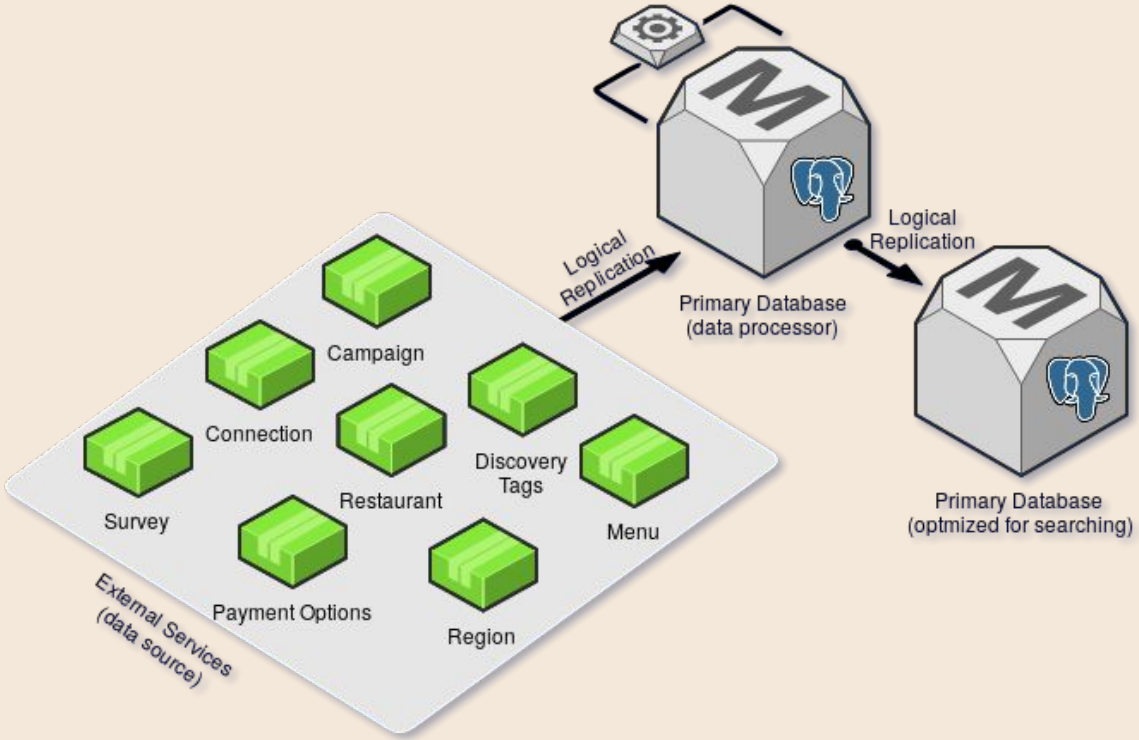
Cada tabela replicada tem um trigger, que processa os dados alimentando um novo schema otimizado para buscas



- Essas triggers não mudam o comportamento das aplicações

Replicando o schema otimizado

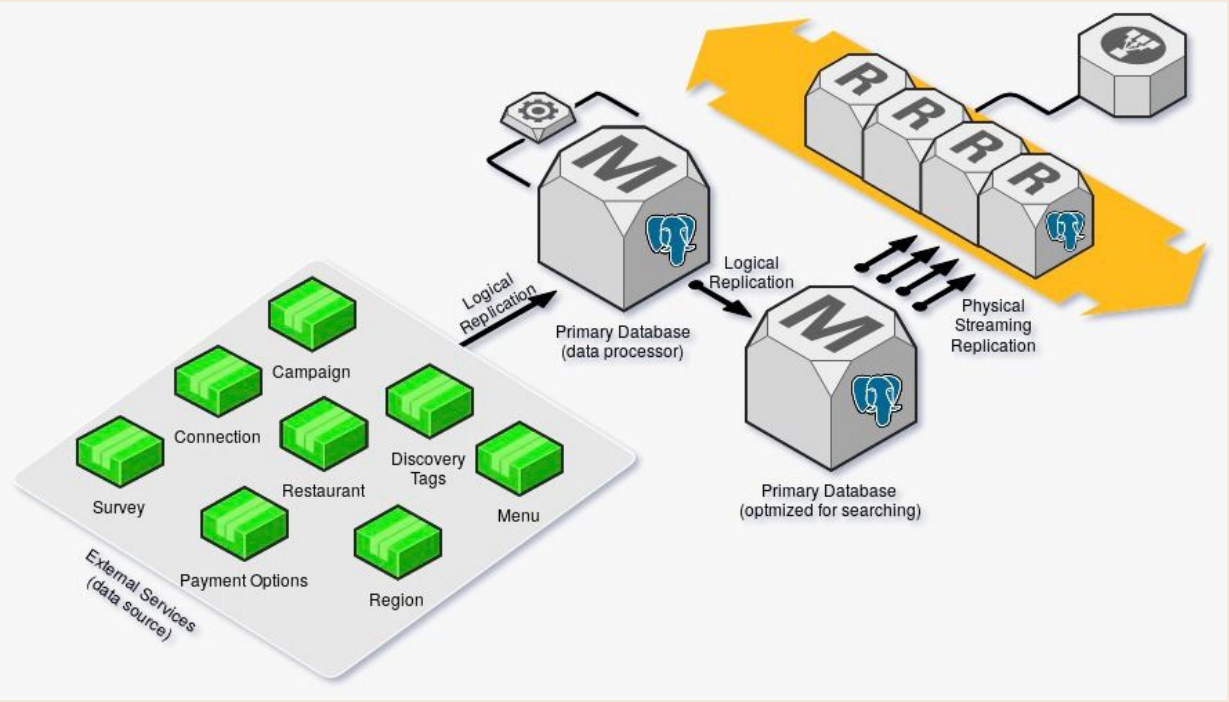
Usando replicação lógica, replicamos apenas schema otimizado para uma nova instância do PostgreSQL



- Esse banco fica “pequeno”
- Ele foi pensando apenas para leitura

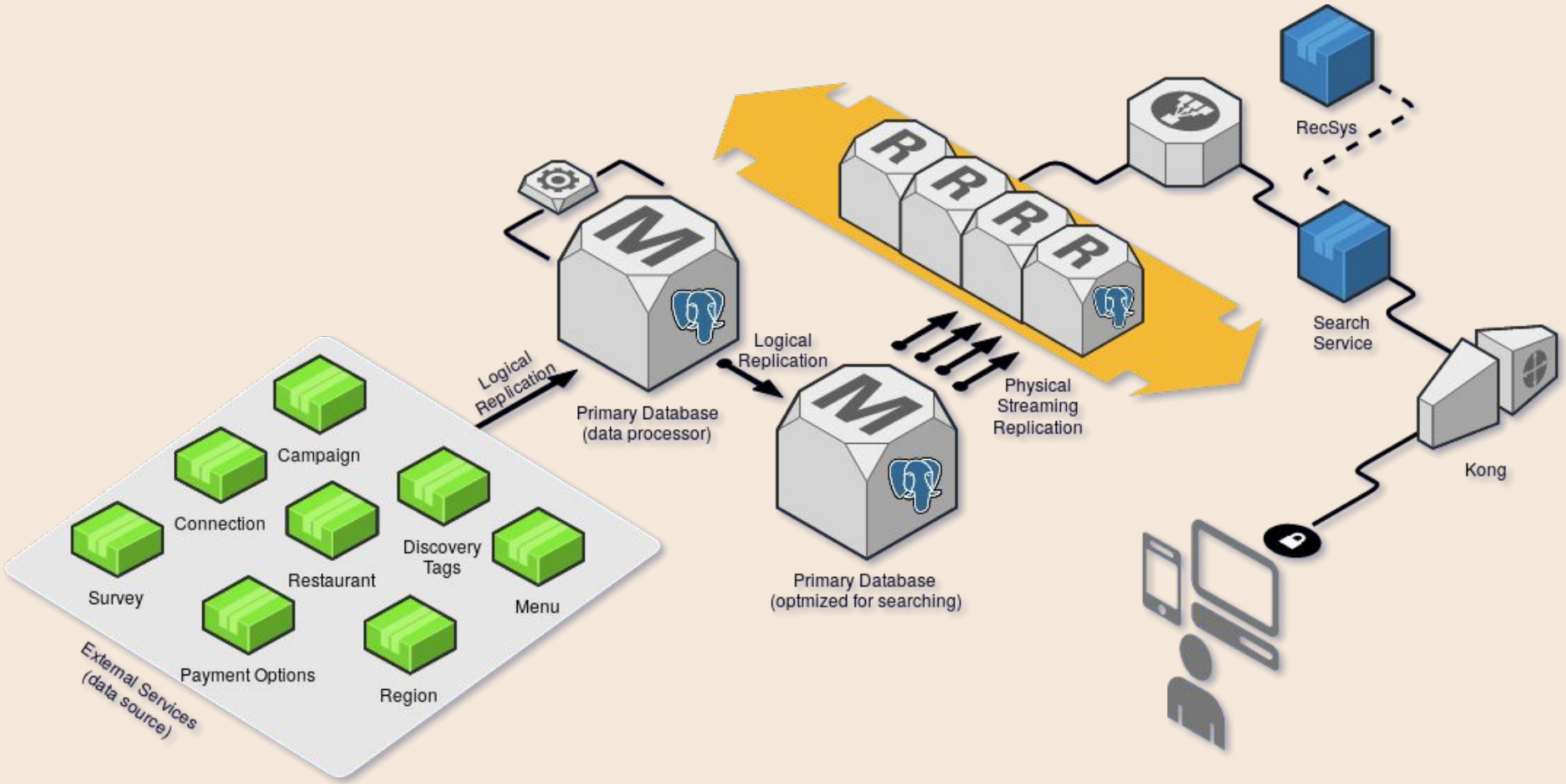
Escalando leitura

Essa instância fica num grupo de auto scalling.

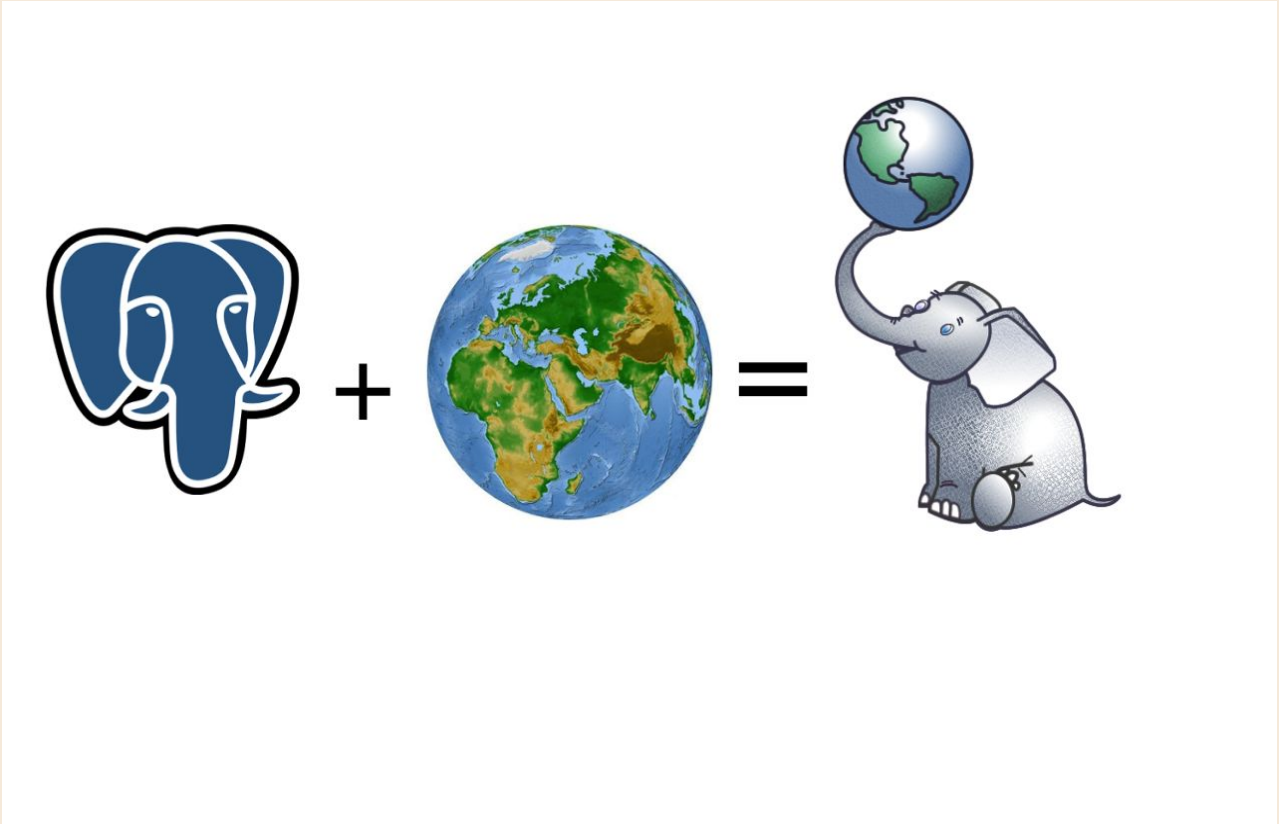


- A quantidade de instâncias depende do uso, é **elástico**
- Os dados são otimizados para leitura e pequenos para escalar rápido

Arquitetura atual



Toda busca, listagem usa lat/lon



Em nosso teste de estresse

No pico de nosso horário de teste de estresse, essa estrutura de ASG chega a **escalar ~50 máquinas c5.9xlarge (72GB de RAM cada)**

X 50

Cada réplica executa mais de 600 consultas por segundo

O que melhorou?

- Zero problemas de consistência
- Tempo de sincronia
- Mais extensível
- Fácil achar problemas
- PG é mais fácil de gerenciar que um cluster de ElasticSearch
- **Escalabilidade**
- Zero dependência dos serviços. Podem fazer deploy a vontade

Obrigado!

Lucas Viecelli
lucas.viecelli@ifood.com.br
telefone 49-99194-4888