

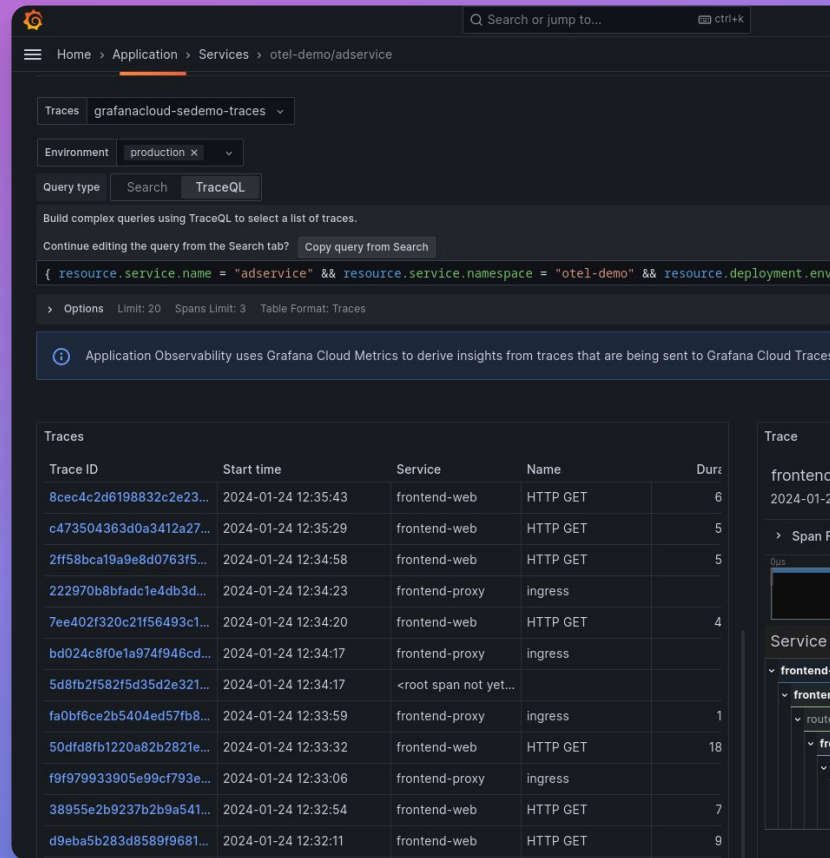
Estado da Arte na Observabilidade

As melhores ferramentas

da atualidade



Juraci Paixão Kröhling
Engenheiro de software



The screenshot displays the Grafana Cloud interface for viewing traces. The breadcrumb navigation shows the path: Home > Application > Services > otel-demo/adservice. The 'Traces' dropdown is set to 'grafanacloud-sedemo-traces'. The 'Environment' is set to 'production'. The 'Query type' is set to 'TraceQL'. A search bar contains the query: `{ resource.service.name = "adservice" && resource.service.namespace = "otel-demo" && resource.deployment.en`. Below the search bar, there are options for 'Limit: 20', 'Spans Limit: 3', and 'Table Format: Traces'. A notification states: 'Application Observability uses Grafana Cloud Metrics to derive insights from traces that are being sent to Grafana Cloud Traces'. The main table, titled 'Traces', lists individual trace entries with columns for Trace ID, Start time, Service, Name, and Duration. The table contains 10 rows of data.

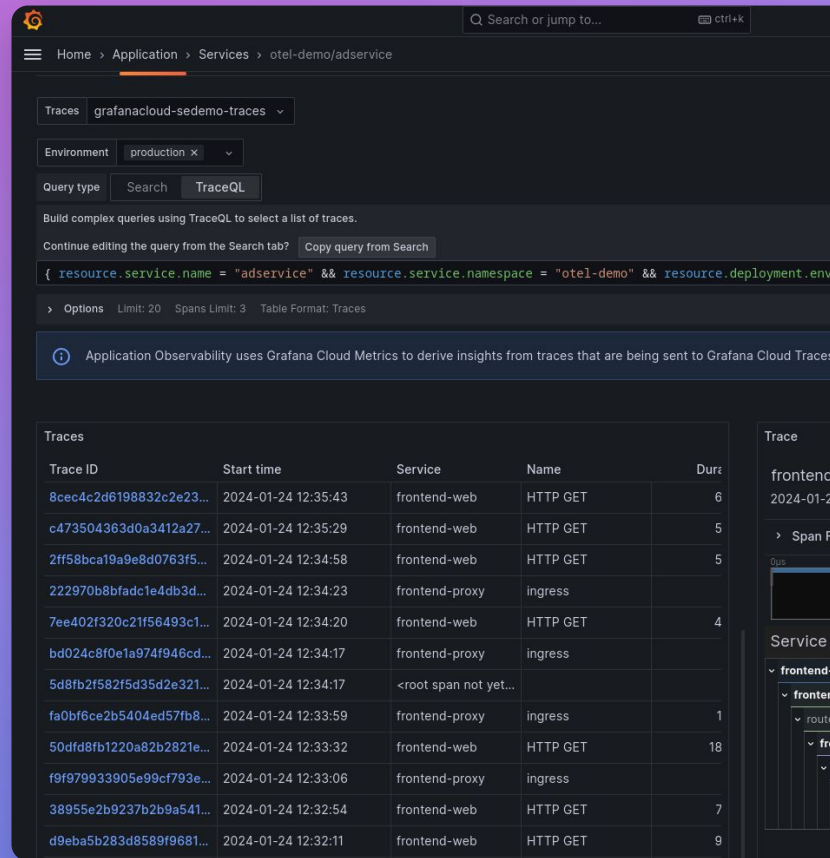
Trace ID	Start time	Service	Name	Duration
8cec4c2d6198832c2e23...	2024-01-24 12:35:43	frontend-web	HTTP GET	6
c473504363d0a3412a27...	2024-01-24 12:35:29	frontend-web	HTTP GET	5
2ff58bca19a9e8d0763f5...	2024-01-24 12:34:58	frontend-web	HTTP GET	5
222970b8bfadc1e4db3d...	2024-01-24 12:34:23	frontend-proxy	ingress	
7ee402f320c21f56493c1...	2024-01-24 12:34:20	frontend-web	HTTP GET	4
bd024c8f0e1a974f946cd...	2024-01-24 12:34:17	frontend-proxy	ingress	
5d8fb2f582f5d35d2e321...	2024-01-24 12:34:17	<root span not yet...		
fa0bf6ce2b5404ed57fb8...	2024-01-24 12:33:59	frontend-proxy	ingress	1
50dfd8fb1220a82b2821e...	2024-01-24 12:33:32	frontend-web	HTTP GET	18
f9f979933905e99cf793e...	2024-01-24 12:33:06	frontend-proxy	ingress	
38955e2b9237b2b9a541...	2024-01-24 12:32:54	frontend-web	HTTP GET	7
d9eba5b283d8589f9681...	2024-01-24 12:32:11	frontend-web	HTTP GET	9

Estado da Arte na Observabilidade

As melhores ferramentas
DE ARMAZENAMENTO
da atualidade



Juraci Paixão Kröhling
Engenheiro de software



The screenshot displays the Grafana Cloud interface for viewing traces. The breadcrumb navigation shows the path: Home > Application > Services > otel-demo/adservice. The 'Traces' dropdown is set to 'grafanacloud-sedemo-traces'. The 'Environment' is set to 'production'. The 'Query type' is set to 'TraceQL'. A search query is entered: `{ resource.service.name = "adservice" && resource.service.namespace = "otel-demo" && resource.deployment.en`. The 'Options' section shows 'Limit: 20', 'Spans Limit: 3', and 'Table Format: Traces'. A notification states: 'Application Observability uses Grafana Cloud Metrics to derive insights from traces that are being sent to Grafana Cloud Traces'. The 'Traces' table lists the following data:

Trace ID	Start time	Service	Name	Duration
8cec4c2d6198832c2e23...	2024-01-24 12:35:43	frontend-web	HTTP GET	6
c473504363d0a3412a27...	2024-01-24 12:35:29	frontend-web	HTTP GET	5
2ff58bca19a9e8d0763f5...	2024-01-24 12:34:58	frontend-web	HTTP GET	5
222970b8bfadc1e4db3d...	2024-01-24 12:34:23	frontend-proxy	ingress	
7ee402f320c21f56493c1...	2024-01-24 12:34:20	frontend-web	HTTP GET	4
bd024c8f0e1a974f946cd...	2024-01-24 12:34:17	frontend-proxy	ingress	
5d8fb2f582f5d35d2e321...	2024-01-24 12:34:17	<root span not yet...		
fa0bf6ce2b5404ed57fb8...	2024-01-24 12:33:59	frontend-proxy	ingress	1
50dfd8fb1220a82b2821e...	2024-01-24 12:33:32	frontend-web	HTTP GET	18
f9f979933905e99cf793e...	2024-01-24 12:33:06	frontend-proxy	ingress	
38955e2b9237b2b9a541...	2024-01-24 12:32:54	frontend-web	HTTP GET	7
d9eba5b283d8589f9681...	2024-01-24 12:32:11	frontend-web	HTTP GET	9

Palestrante



Juraci Paixão Kröhling
Engenheiro de software



Quem sou eu

Engenheiro de software na Grafana Labs

Membro do Comitê de Governança do projeto OpenTelemetry

Embaixador da Cloud Native Computing Foundation (CNCF)

Mantenedor de módulos do OpenTelemetry Collector

Ex-mantenedor do Jaeger

Ex-mantenedor do OpenTracing

Criador do Dose de Telemetria

Palestrante



Juraci Paixão Kröhling
Engenheiro de software



Agenda

Introdução

- Aviso
- Infraestrutura de observabilidade é complexa

Ferramentas de armazenamento

- Critérios
- Opções principais
- Outras opções a se considerar

Perguntas e respostas

Aviso de conflito
de interesse!



“

Eu trabalho na Grafana Labs.





“

*Mas aqui são minhas opiniões sinceras
como engenheiro de software.*



Por que não falou do...

- Porque eu esqueci
- Porque não é código aberto (definido pela OSI)
- Porque não é mais tão relevante assim
- Porque senão ficaríamos aqui até o final do trimestre



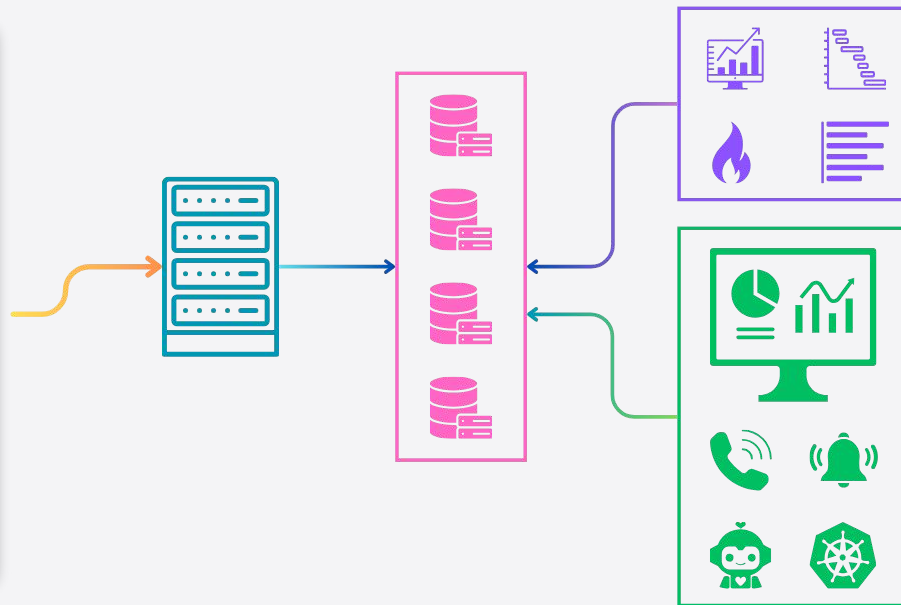

Componentes de uma solução de observabilidade

```
121
122 func (r *dnsResolver) resolve(ctx context.Context) ([]string, error) {
123     r.shutdownWg.Add(1)
124     defer r.shutdownWg.Done()
125
126     addrs, err := r.resolver.LookupIPAddr(ctx, r.hostname)
127     if err != nil {
128         _ = stats.RecordWithTags(ctx, resolverSuccessFalseMutators, mNumResolutions.M(1))
129         return nil, err
130     }
131
132     _ = stats.RecordWithTags(ctx, resolverSuccessTrueMutators, mNumResolutions.M(1))
133
134     backends := make([]string, len(addrs))
135     for i, ip := range addrs {
136         var backend string
137         if ip.IP.To4() != nil {
138             backend = ip.String()
139         } else {
140             // it's an IPv6 address
141             backend = fmt.Sprintf("[%s]", ip.String())
142         }
143
144         // if a port is specified in the configuration, add it
145         if r.port != "" {
146             backend = fmt.Sprintf("%s:%s", backend, r.port)
147         }
148
149         backends[i] = backend
150     }
```



Componentes de uma solução de observabilidade

```
121
122 func (r *dnsResolver) resolve(ctx context.Context) ([]string, error) {
123     r.shutdownWg.Add(1)
124     defer r.shutdownWg.Done()
125
126     addrs, err := r.resolver.LookupIPAddr(ctx, r.hostname)
127     if err != nil {
128         _ = stats.RecordWithTags(ctx, resolverSuccessFalseMutators, mNumResolutions.M(1))
129         return nil, err
130     }
131
132     _ = stats.RecordWithTags(ctx, resolverSuccessTrueMutators, mNumResolutions.M(1))
133
134     backends := make([]string, len(addrs))
135     for i, ip := range addrs {
136         var backend string
137         if ip.IP.To4() != nil {
138             backend = ip.String()
139         } else {
140             // it's an IPv6 address
141             backend = fmt.Sprintf("[%s]", ip.String())
142         }
143
144         // if a port is specified in the configuration, add it
145         if r.port != "" {
146             backend = fmt.Sprintf("%s:%s", backend, r.port)
147         }
148
149         backends[i] = backend
150     }
151 }
```



Componentes de uma solução de observabilidade



Instrumentação



Instrumentação

- Rastreamento distribuído
 - OpenTelemetry API
- Métricas
 - Micrometer
 - OpenTelemetry API
 - Prometheus
- Logs
 - Cada linguagem tem (diversas) opções próprias!



Coleta



Coleta

- Fluentbit
- Grafana Agent
- OpenTelemetry Collector
- Vector



Armazenamento



Armazenamento

- Grafana Loki - <https://grafana.com/oss/loki>
- Grafana Tempo - <https://grafana.com/oss/tempo>
- Jaeger - <https://www.jaegertracing.io>
- OpenSearch - <https://opensearch.org>
- Prometheus - <https://prometheus.io>



Armazenamento - Outras opções

- Apache SkyWalking (BanyanDB) - <https://skywalking.apache.org>
- ClickHouse - <https://clickhouse.com>
- Grafana Mimir - <https://grafana.com/oss/mimir>
- InfluxDB - <https://github.com/influxdata/influxdb>
- SigNoz - <https://signoz.io>
- VictoriaMetrics - <https://github.com/VictoriaMetrics/VictoriaMetrics>
- Zipkin - <https://zipkin.io>



Armazenamento - Critérios

- Sinais
- Tipo de banco de dados
- Compatibilidade
 - Protocolo OpenTelemetry (OTLP), Jaeger, Prometheus
- Governança
 - Licença, tipo de comunidade (fundação, empresa, ...)



Armazenamento - Grafana Loki

- Sinais
 - ☒ Métricas
 - ☐ Logs
 - ☒ Rastros
- Detalhes
 - Indexação seletiva
 - Object storage
 - Armazenamento de longa duração
 - Consulta com LogQL
 - Visualização via Grafana





Armazenamento - Grafana Loki - Compatibilidade

- ☐ OTLP
- ☐ Fluentbit
- ☐ Logstash



Armazenamento - Grafana Loki - Governança

-  Licença: AGPL-3.0
-  Grafana Labs



Armazenamento - Grafana Tempo

- Sinais
 - ☒ Métricas
 - ☒ Logs
 - ☐ Rastros
- Detalhes
 - Indexação seletiva
 - Object storage
 - Armazenamento de longa duração
 - Consulta com TraceQL
 - Visualização via Grafana





Armazenamento - Grafana Tempo - Compatibilidade

- ☐ OTLP
- ☐ Jaeger
- ☐ Zipkin



Armazenamento - Grafana Tempo - Governança

-  Licença: AGPL-3.0
-  Grafana Labs



Armazenamento - Jaeger

- Sinais
 - ☒ Métricas
 - ☒ Logs
 - ☐ Rastros
- Detalhes
 - Elasticsearch ou Cassandra como banco de dados, extensível
 - Visualização com interface própria



Armazenamento - Jaeger - Compatibilidade

- ☐ OTLP
- ☐ Jaeger
- ☐ Zipkin



Armazenamento - Jaeger - Governança

- ☐ Licença: Apache-2.0
- ☐ CNCF



Armazenamento - OpenSearch

- Sinais
 - ☐ Métricas
 - ☐ Logs
 - ☐ Rastros
- Detalhes
 - Armazenamento de objetos JSON
 - Indexação completa dos dados
 - Visualização com interface própria





Armazenamento - OpenSearch - Compatibilidade

- ☐ OTLP
- ☐ Jaeger
- ☒ Zipkin
- ☒ Fluentbit
- ☒ Logstash



Armazenamento - OpenSearch - Governança

-  Licença: Apache-2.0
-  OpenSearch é marca registrada da Amazon Web Services (AWS)



Armazenamento

- Um sinal, ou todos?
 - Rastreamento distribuído
 - Logs (ou eventos)
 - Métricas
 - Perfis
 - Stacktraces
 - ...
- Governança
- Comprar ou construir?



Resumo



Resumo

- Todas as ferramentas mencionadas aqui são boas
- Defina seus critérios e tenha experiência em primeira mão
- Construa sua arquitetura com base nas necessidades, um passo de cada vez



Perguntas e respostas



Obrigado!