Observabilidade e Monitoramento I

Prof. Helder Prado Santos

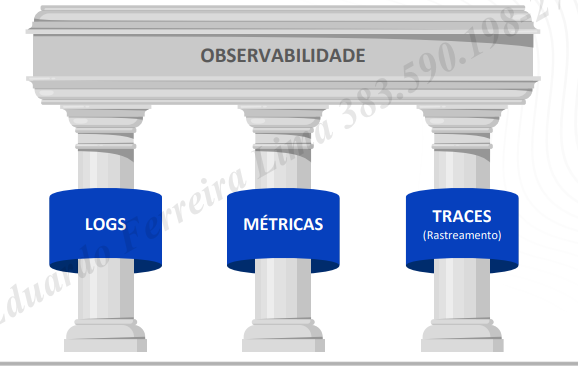
* Do desenvolvimento para produção
  + Paralelo com o avião
  + Quando o avião esta pronto para voar colocamos o software em produção
    - Quando o avião esta voando podemos pegar problemas não vistos no projeto ou seja problemas na produção
    - Se preparar para os riscos em produção
    - Conceito de monitoramento e observabilidade são diferentes
    - Não botem uma aplicação em produção sem se preocupar com monitoramento e observabilidade

Tempo 12:10

* Monitoramento
  + Monitoramento é o processo de coletar, analisar e usar dados sobre o desempenho de um sistema, infraestrutura ou aplicação com base em métricas predefinidas.
  + O objetivo é detectar anomalias, falhas e problemas de desempenho. Ele se baseia em alertas configurados antecipadamente, sendo eficaz para monitorar eventos e condições específicas.
  + Coletar analisar e usar os dados
  + Normalmente dados de monitoramento são:
    - dados de sistema,
    - infra-estrutura
    - aplicação
    - também pode ser analisados e quantificados alguns dados de negocio
    - Se baseam em alertas configurados
  + Características do Monitoramento
    - Proativo:
      * O monitoramento é configurado para observar algumas métricas-chave, como:
        + Uso de CPU
        + Memória
        + Tempo de resposta
        + Disponibilidade (muito importante para big techs conta do uptime/dowtime+uptime Notacao de 9 ex 9.99999% de disponibilidade).
        + Quantidade de requisições por segundo
    - Métricas Definidas:
      * Baseia-se em métricas pré-configuradas, como:
        + uso de recursos
        + contagem de erros
        + latência – tempo que demora para a requisição sair do cliente para chegar ao servidor
      * métricas são medidas quantitativas para monitorar dados que são gerados pelo monitoramento (qualquer coisa quantificável pode ser uma métrica)
    - • Alertas:
      * Envia notificações automáticas quando essas métricas ultrapassam limites estabelecidos
        + por exemplo, CPU acima de 80%
      * Super importante tanto no monitoramento quanto na observabilidade
      * Tem que enviar o aviso de forma automática quando o limite for estabelecido
    - Visão Superficial:
      * Ele oferece uma visão clara sobre a "saúde" do sistema, mas é limitado àquilo que foi previamente configurado para monitorar.
      * Se um problema não foi antecipado, ele pode passar despercebido.
      * Limitado ao que foi antecipado.
* Observabilidade
  + OBSERVABILIDADE é um conceito mais amplo e profundo, focado na capacidade de entender o comportamento interno de um sistema complexo apenas examinando suas saídas (logs, métricas, traces).
    - Temos que analisar apenas três parâmetros
      * Logs
      * Métricas
      * Traces
  + Enquanto o monitoramento mede coisas predefinidas, a observabilidade permite que você descubra problemas não antecipados, explorando dados mais detalhados
  + Muito importante para sistemas Muito complexos pois e impossível monitorar todos os parâmetros de sistemas muitos complexos
* Observabilidade vs Monitoramento

|  |  |  |
| --- | --- | --- |
| **Caracteristica** | **Monitoramento** | **Observabilidade** |
| Foco | Métricas e eventos pré definidos | Diagnóstico e entendimento do sistema |
| Objetivo | Detectar anomalias e falhas conhecidas | Explorar e diagnosticar comportamentos desconhecidos |
| Coleta de Dados | Métricas simples e limites | Logs métricas e tracing (dados ricos e detalhados) |
| Alertas | Baseados em limiares predefinidos | Suporte a exploração e investigação de falhas |
| Visão | Limitada ao que foi configurado | Ampla cobrindo vários tipos de dados |
| Abordagem | Proativa (monitora e alerta) | Reativa e exploratória (permite investigação profunda) |
| Adequado Para | Problemas conhecidos e limiares previsíveis | Problemas complexos e emergentes |

* + Exemplo de monitoramento:
    - Você configura um monitoramento para alertá-lo se o uso da CPU de um servidor ultrapassar 90%. Quando esse limite é atingido, você recebe um alerta para investigar.
  + Exemplo de Observabilidade:
    - Ao notar que o tempo de resposta de uma aplicação está aumentando, você usa ferramentas de observabilidade para investigar os logs, traces e métricas de múltiplos serviços e descobre que o problema está em uma interação lenta com o banco de dados.
* É como se…
  + Monitoramento é como um sistema de alarme
    - ele é configurado para soar quando algo específico dá errado (como portas abertas ou movimento detectado). Ele avisa se algo conhecido falha, mas não pode prever o inesperado.
  + Observabilidade é como uma câmera de segurança inteligente:
    - ela grava todos os eventos e você pode investigar o que aconteceu mesmo que o alarme não tenha soado. Se houver um comportamento estranho que você não antecipou, você tem os dados para analisá-lo e entender melhor o que ocorreu.
* O que preciso monitorar?
  + Os 4 sinais de SRE (Site Reliability Engineers) - 4 sinais de ouro (livro grátis no google)
    - 1. Latência
      * Mede o atraso inicial da comunicação
      * o tempo de viagem dos dados
    - 2. Tráfego
      * A quantidade de dados transmitido e recebido
      * Requisições por segundo
      * Quantidade de dados ativos
      * Volume de dados sendo trocados
    - 3. Erros
      * Quantidade de erros
        + Erros de servidor
        + Pagina não encontrada
    - 4. Saturação.
      * Algo que esta chegando no limite
        + Limite de custo de uso de banco de dados
  + Segundo a google esses são os 4 pontos que devemos monitorar inicialmente
  + Link do livro <https://sre.google/sre-book/table-of-contents/>
* Os 3 Pilares da Observabilidade
  + Logs
  + Métricas
  + Traces (rastreamento)



* Pilar: Logs ( O mais importante de todos a base para a observabilidade)
  + Logs são registros textuais gerados por aplicações, sistemas operacionais, servidores ou outros componentes de software para descrever eventos, ações e estados ocorridos em determinado momento.
  + Um registro uma fotografia um momento no tempo
  + Log tem sempre o time Stamp a data e hora que ocorreu
  + Logs tem uma forma fixa dentro de um processo podendo variar a cada processo
  + Quem pode emitir logs
    - A aplicação
    - O servidor
    - O cliente
  + Os logs geralmente incluem informações como:
    - Timestamp (Data e Hora): Quando o evento ocorreu.
    - Nível de Log: Indica a severidade do evento, como INFO, DEBUG, ERROR, WARNING, etc.
    - Mensagem: Uma descrição do evento, o que aconteceu ou o status do sistema.
    - Contexto: Informações adicionais como ID de usuário, nome do serviço, ou qualquer dado que ajude a identificar a causa ou impacto do evento.
  + Pontos Importantes do Pilar Logs
    - Coleta Eficiente de Logs:
      * Centralização de Logs;
        + Repositório de Logs (correlação de dados)
      * Correlação de Logs.
    - Estruturação dos Logs
      * Logs Estruturados;
      * Níveis de Severidade.
    - Retenção e Rotação de Logs:
      * Política de Retenção de Logs;
        + Log é caro
        + Armazenar log é caro
        + Processar log é caro
        + Quanto tempo eu vou armazenar o log
      * Rotação de Logs.
    - NÍVEIS DE LOG:
      * DEBUG:
        + Informações detalhadas usadas para depuração.
      * INFO:
        + Informações sobre operações normais.
        + Eventos coisas que já aconteceram
      * WARNING:
        + Potenciais problemas que não afetam o sistema, mas podem necessitar de atenção.
      * ERROR:
        + Erros que impactam partes do sistema, mas não o desativam completamente.
      * CRITICAL/FATAL:
        + Erros críticos que podem resultar em falha total ou interrupção significativa do serviço.
* Pilar: Métricas
  + Métricas são medições ***quantificáveis*** de um sistema. Elas fornecem uma visão em tempo real de como os recursos estão sendo utilizados e como as aplicações estão se comportando, facilitando a detecção de anomalias e o diagnóstico de problemas.
  + As métricas são ***simples, agregadas e leves*** em termos de armazenamento, sendo ideais para monitoramento contínuo.
  + As métricas vem dos logs
  + Categorias de métricas
    - Métricas de Sistema (Infraestrutura):
      * CPU: Percentual de uso de CPU de uma máquina ou serviço
      * Memória: Quantidade de memória sendo utilizada;
      * Rede: Quantidade de dados sendo enviados e recebidos
      * Disco: Espaço em disco disponível e taxa de leitura/gravação
      * … entre muitas outras.

Fim do bloco 1