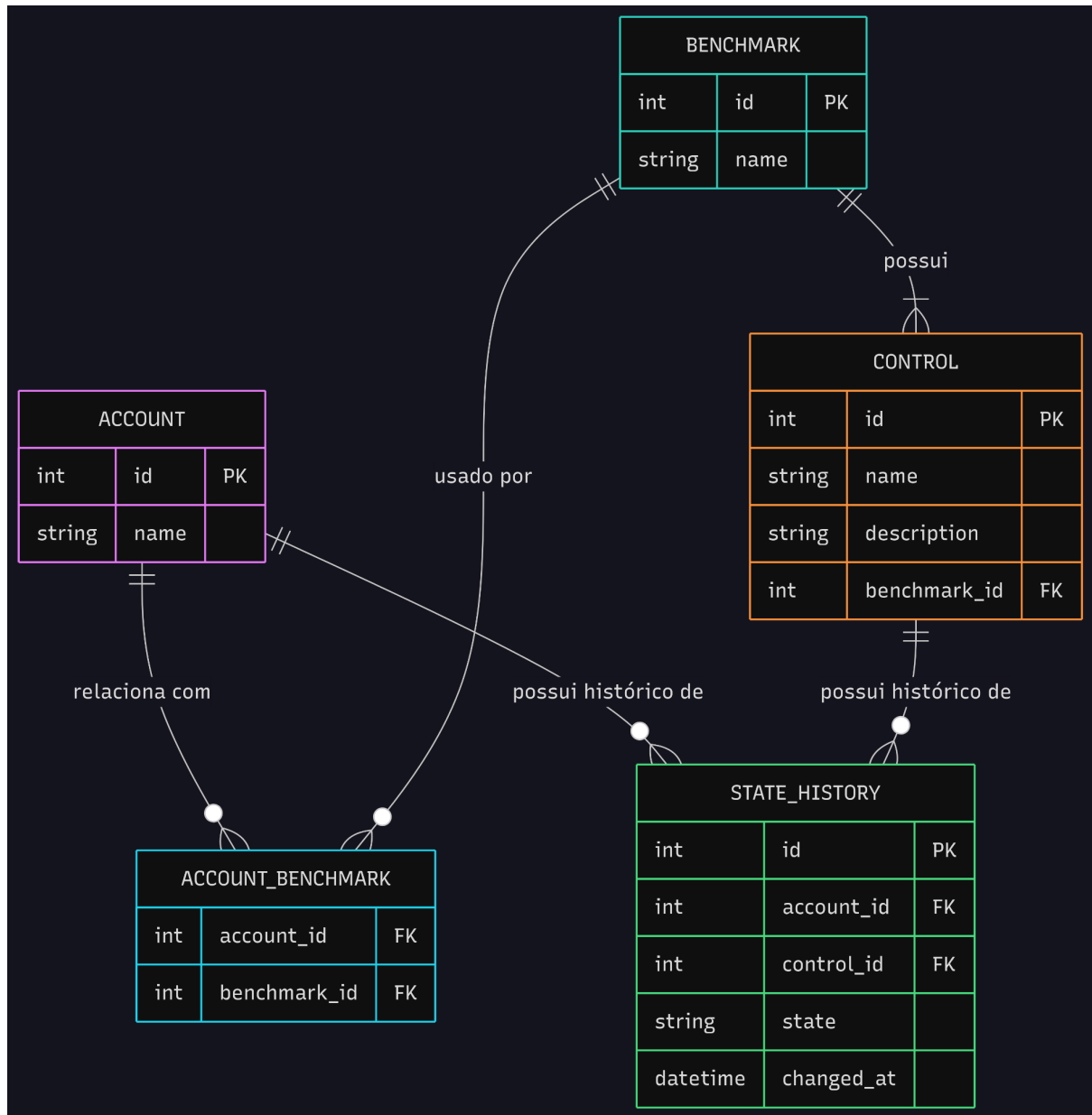


Primeira Questão: Benchmarks por Account — estado atual e histórico

## 1. Modelo Entidade-Relacionamento (MER)



Este modelo é composto por cinco entidades principais para representar o mini-mundo de forma eficiente e sem redundância de dados.

### Entidades e Atributos

- **Account**
  - id (Chave Primária)
  - name
- **Benchmark**
  - id (Chave Primária)
  - name

- **Control**
  - id (Chave Primária)
  - name
  - description
  - benchmark\_id (Chave Estrangeira para Benchmark)
- **Account\_Benchmark** (Tabela de Junção)
  - account\_id (Chave Estrangeira para Account)
  - benchmark\_id (Chave Estrangeira para Benchmark)
- **State\_History**
  - id (Chave Primária)
  - account\_id (Chave Estrangeira para Account)
  - control\_id (Chave Estrangeira para Control)
  - state (ex: 'ok', 'alarm')
  - changed\_at (Timestamp da mudança de estado)

### Cardinalidades

- Benchmark 1 -- N Control: Um Benchmark possui muitos Controles.
- Account N -- M Benchmark: Uma Account pode se relacionar a muitos Benchmarks (e vice-versa), o que é resolvido pela tabela de junção Account\_Benchmark.
- A tabela State\_History armazena o histórico de múltiplos estados (N) para a combinação de uma Account e um Control.

## 2. Índices Recomendados por Cenário

A estratégia de indexação foca em otimizar as operações de filtro (*WHERE*) e ordenação (*ORDER BY*) para cada cenário.

### Cenário (Q1): Obter estado Atual para uma Account.

- **Índice Recomendado:**
  - **Tabela:** State\_History
  - **Colunas:** (account\_id, control\_id, changed\_at)
- **Justificativa:**  
Este índice composto é ideal para encontrar o "último estado". A consulta filtra primeiro por account\_id e depois por control\_id. Como os resultados já estão pré-ordenados pela data (changed\_at), o banco de dados pode encontrar o estado mais recente de cada controle, sem precisar varrer e ordenar os dados, o que seria muito custoso.

### Cenário (Q2): Obter mudanças de estado em um INTERVALO para uma Account.

- **Índice Recomendado:**
  - **Tabela:** State\_History
  - **Colunas:** (account\_id, changed\_at)
- **Justificativa:**  
A consulta filtra primariamente por account\_id e depois aplica um filtro de intervalo na coluna changed\_at. Este índice permite ao banco de dados localizar rapidamente

todos os registros da conta especificada e, em seguida, fazer uma varredura muito eficiente apenas no intervalo de tempo desejado, evitando uma leitura completa da tabela.

**Cenário (Q3): Obter estado em uma DATA/HORA X para uma Account.**

- **Índice Recomendado:**
  - **Tabela:** State\_History
  - **Colunas:** (account\_id, control\_id, changed\_at)
- **Justificativa:**

O mesmo índice do Cenário 1 resolve para esta consulta com máxima eficiência. A lógica é quase idêntica: para um account\_id e control\_id, o banco de dados busca o primeiro registro cuja data (changed\_at) seja menor ou igual à data observada. A ordenação do índice torna esta busca extremamente rápida.

**Índices de Suporte (para junções)**

- **Tabela:** Control -> **Índice em:** (benchmark\_id)
- **Tabela:** Account\_Benchmark -> **Índice em:** (account\_id)
- **Justificativa:** Índices em colunas de chave estrangeira são fundamentais para acelerar as operações de JOIN entre as tabelas, que serão necessárias em todos os cenários para ligar os Benchmarks aos seus respectivos Controles e Accounts.