

Documentação do Data Warehouse - Fórmula 1

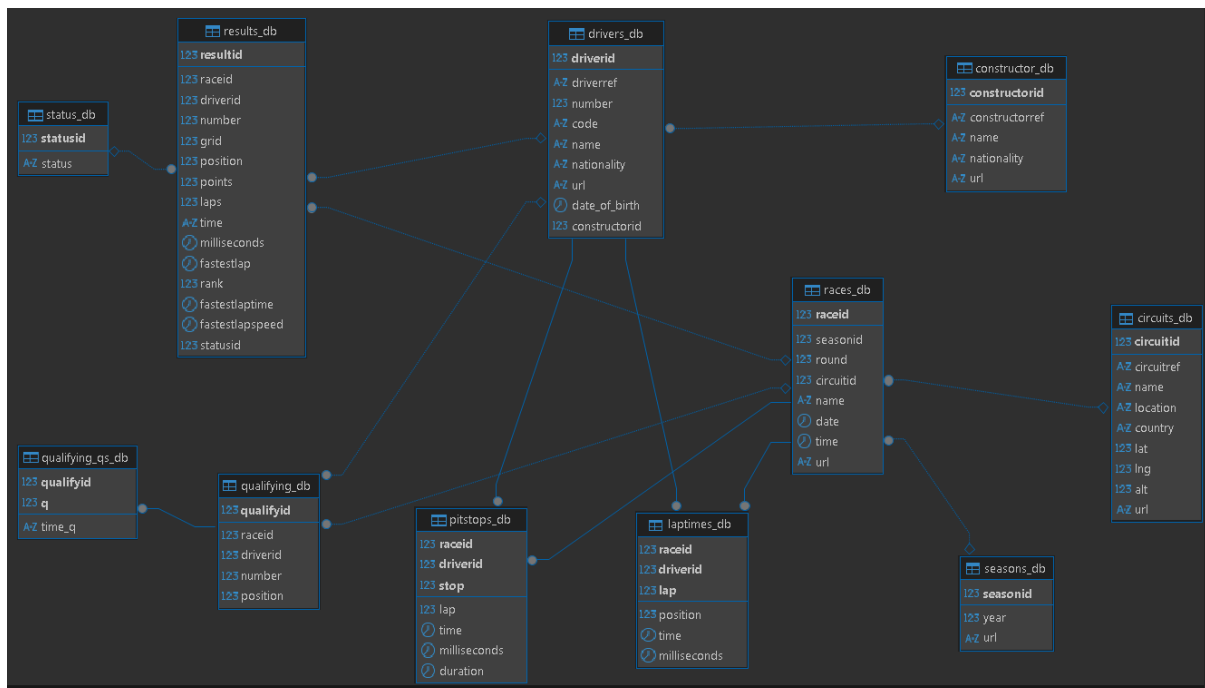
1. Objetivos do Projeto e Perguntas de Negócio:

Visão Geral: O objetivo deste projeto foi migrar uma base de dados transacional de Fórmula 1 para um ambiente analítico (Data Warehouse), permitindo a geração de relatórios de inteligência sobre segurança, desempenho de pilotos e confiabilidade das equipes.

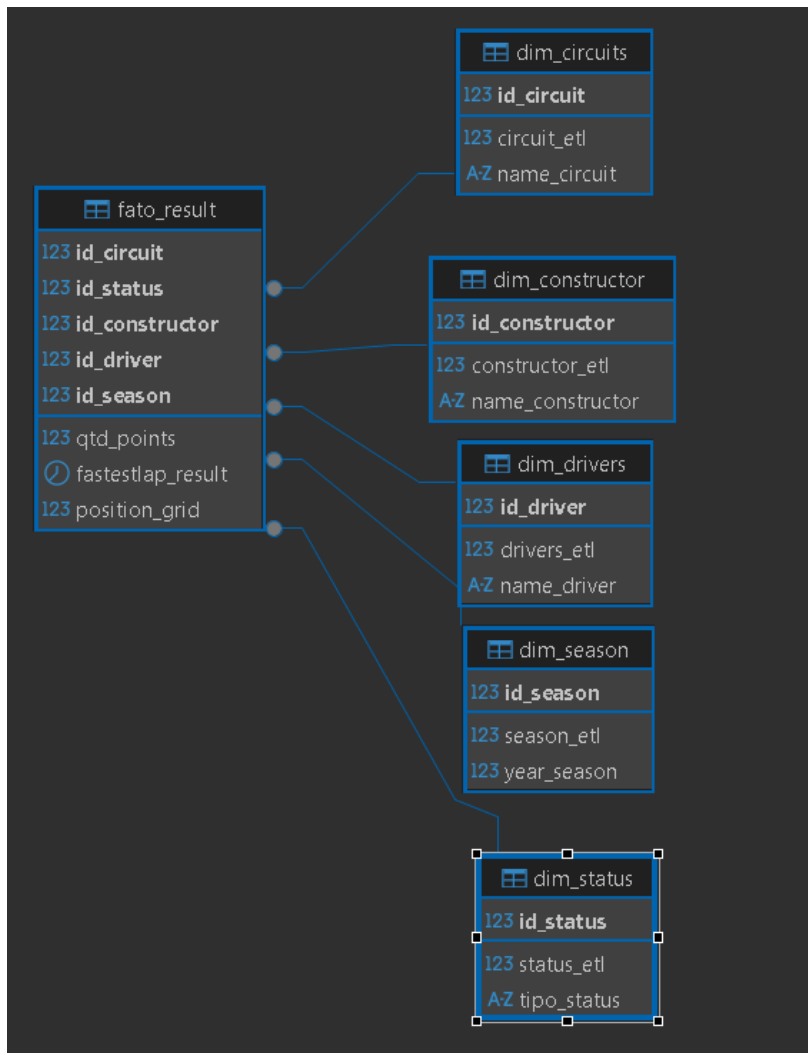
Requisitos Analíticos (As Perguntas): Para guiar a modelagem do DW, foram levantadas as seguintes perguntas estratégicas que o sistema precisa responder:

- **Segurança:** Qual é a pista com o maior histórico de acidentes?
- **Performance das Equipes:** Quais são as construtoras mais vitoriosas em cada circuito específico?
- **Confiabilidade:** Quais equipes apresentaram o maior índice de falhas técnicas (motor, câmbio etc.)?
- **Performance dos Pilotos:**
 - Quais pilotos acumulam o maior número de Pole Positions?
 - Quem são os recordistas de volta mais rápida em cada pista?
- **Regulamento:** Quais temporadas tiveram mais infrações e desclassificações de pilotos?

2. Modelagem de Dados



Modelo Transacional (Origem) - Alta complexidade de Join



Modelo Dimensional (Destino) - Star Schema otimizado para análise.

3.1: Visão Geral do Fluxo ETL

Esta etapa do processo de ETL (*Extract, Transform, Load*) é responsável por popular as tabelas de dimensão do Data Warehouse. O processo foi desenvolvido no **Pentaho Data Integration (PDI)** e consiste na extração direta das tabelas normalizadas do banco de origem para as tabelas desnormalizadas (ou de lookup) do DW.

Ferramenta utilizada: Pentaho Data Integration (Kettle) **Banco de Dados:** PostgreSQL

3.2: Diagrama de Carga (Pentaho)

O fluxo de carga segue um mapeamento direto (1:1) para as principais entidades do negócio.

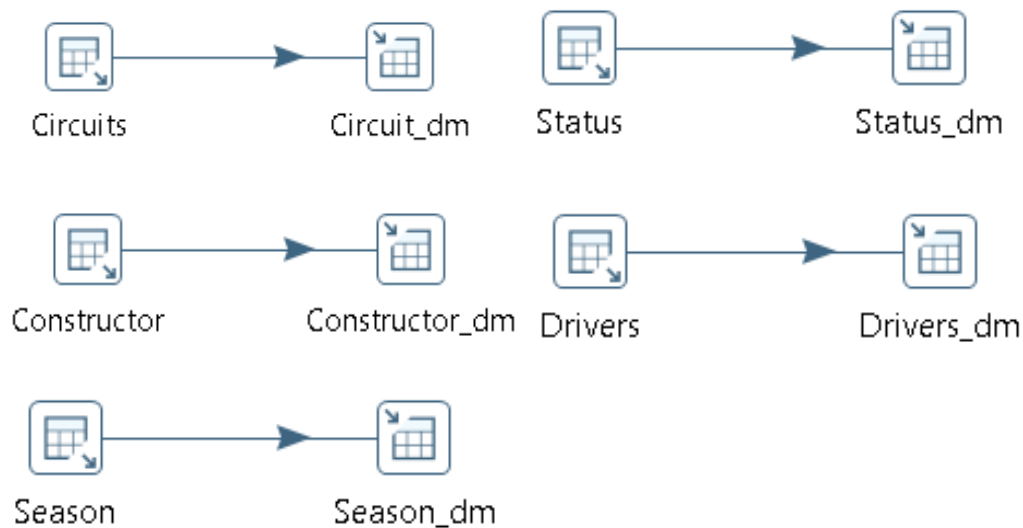


Tabela Origem (Source)	Tabela Destino (Dimension)	Descrição do Conteúdo
Circuits	Circuit_dm	Dados geográficos e nomes das pistas/circuitos.
Status	Status_dm	Dicionário de situações de corrida (ex: Acidente, Desclassificado, Motor).
Constructor	Constructor_dm	Dados das equipes (scuderias) construtoras.
Drivers	Drivers_dm	Informações pessoais e profissionais dos pilotos.
Season	Season_dm	Identificação das temporadas (anos) do campeonato.

3.3: Fluxo de ETL da Tabela Fato

Esta é a etapa central do processo, em que os dados transacionais de corridas são transformados em métricas de negócio e conectados às dimensões criadas na Fase 1.



Dicionário do Processo ETL (Pentaho):

Passo (Step)	Tipo	Descrição Técnica
bd_fato	<i>Table Input</i>	Extrai os dados brutos da tabela de origem results_db, fazendo <i>joins</i> iniciais com races_db para obter datas e IDs originais.
circuits	<i>Database Lookup</i>	Consulta a dimensão dim_circuits para substituir o ID original da pista pela nova <i>Surrogate Key</i> do DW (id_circuit).
construct	<i>Database Lookup</i>	Busca a chave primária da equipe na tabela dim_constructor.
driver	<i>Database Lookup</i>	Busca a chave primária do piloto na tabela dim_drivers.
season	<i>Database Lookup</i>	Busca a chave da temporada (ano) na tabela dim_season.
status	<i>Database Lookup</i>	Busca a chave do status da corrida (ex: 'Finished', 'Collision') na tabela dim_status.
Remover Duplicatas	<i>Unique Rows</i>	Garante a integridade dos dados removendo linhas duplicadas acidentais geradas na extração.
Tabela_fato	<i>Table Output</i>	Carrega os dados tratados e validados na tabela final fato_result no PostgreSQL.

Dicionário de Dados do Data Warehouse

Abaixo está o detalhamento técnico da tabela central (fato_result) e suas conexões.

Tabela: fato_result

Esta tabela armazena o desempenho de cada piloto, em cada corrida, de cada temporada.

Coluna	Tipo de Dado	Tipo de Chave	Descrição / Regra de Negócio
id_circuit	Integer	FK (Foreign Key)	Referência para a tabela dim_circuits (Local da corrida).
id_status	Integer	FK (Foreign Key)	Referência para a tabela dim_status (Situação final: acabou, bateu, etc.).
id_constructor	Integer	FK (Foreign Key)	Referência para a tabela dim_constructor (Equipe do carro).
id_driver	Integer	FK (Foreign Key)	Referência para a tabela dim_drivers (Piloto).
id_season	Integer	FK (Foreign Key)	Referência para a tabela dim_season (Ano do campeonato).
qtd_points	Numeric/Int	Métrica	Pontuação obtida pelo piloto na corrida. Usado para somatórios.
fastestlap	Time/Varchar	Métrica	O tempo da volta mais rápida registrada pelo piloto naquela prova.
position_grid	Integer	Métrica	A posição de largada do piloto (ex: 1 = Pole Position).

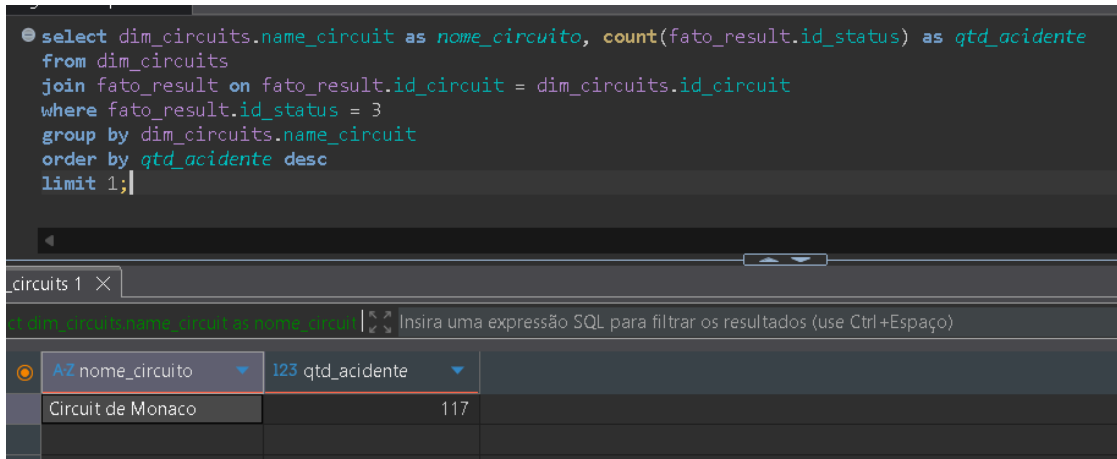
5. Glossário de Consultas Analíticas

Esta seção documenta as consultas SQL padrão utilizadas para validar os requisitos de negócio definidos no início do projeto. Cada consulta representa um indicador-chave de desempenho (KPI) extraído do Data Warehouse.

5.1. Análise de Segurança: Pistas com Maior Risco

- **Descrição:** Identifica o circuito com o maior histórico acumulado de acidentes. Essencial para análises de segurança e gestão de riscos.
- **Lógica Técnica:** Filtra a tabela fato pelo status de "Acidente" (`id_status = 3`) e realiza uma contagem agrupada por nome do circuito.
- **Script SQL:**

```
select dim_circuits.name_circuit as nome_circuito, count(fato_result.id_status) as qtd_acidente
from dim_circuits
join fato_result on fato_result.id_circuit = dim_circuits.id_circuit
where fato_result.id_status = 3
group by dim_circuits.name_circuit
order by qtd_acidente desc
limit 1;
```



nome_circuito	qtd_acidente
Circuit de Monaco	117

5.2. Dominância de Equipes por Circuito

- **Descrição:** Determina qual construtora obteve a maior pontuação acumulada em cada uma das pistas, revelando tendências históricas de desempenho.
- **Lógica Técnica:** Utiliza `DISTINCT ON` para retornar apenas a equipe líder de cada pista, ordenando pela soma total de pontos.
- **Script SQL:**

```

select * from (select distinct on (dim_circuits.name_circuit)
dim_circuits.name_circuit as nome_circuito,
dim_constructor.name_constructor as nome_equipe, sum(fato_result.qtd_points) qtd_pontos
from dim_constructor
join fato_result on fato_result.id_constructor = dim_constructor.id_constructor
join dim_circuits on dim_circuits.id_circuit = fato_result.id_circuit
group by dim_circuits.name_circuit, dim_constructor.name_constructor
order by dim_circuits.name_circuit, qtd_pontos desc) as equipes_pistas_campeas
order by qtd_pontos desc

```

circuits(+) 1 X

st * from (select distinct on (dim_circuits.name_circuit) Insira uma expressão SQL para filtrar os resultados (use Ctrl+Espaço)

AZ nome_circuito	AZ nome_equipe	123 qtd_pontos
Circuit Gilles Villeneuve	Mercedes	394
Autodromo Nazionale di Monza	Ferrari	372,5
Silverstone Circuit	Mercedes	353
Circuit de Spa-Francorchamps	Mercedes	342
Suzuka Circuit	Mercedes	324
Circuit de Barcelona-Catalunya	Mercedes	321
Autódromo José Carlos Pace	Mercedes	319
Shanghai International Circuit	Mercedes	316
Circuit de Monaco	Mercedes	313
Yas Marina Circuit	Mercedes	299
Albert Park Grand Prix Circuit	Mercedes	298
Sepang International Circuit	Mercedes	276,5
Bahrain International Circuit	Mercedes	268
Hungaroring	Mercedes	258
Hockenheimring	McLaren	209
Marina Bay Street Circuit	Ferrari	209
Red Bull Ring	Mercedes	187
Sochi Autodrom	Mercedes	150
Nürburgring	Williams	149
Woking GP	Team Lotus	121

5.3. Integridade Esportiva: Pilotos com Mais Desclassificações

- **Descrição:** Lista os 5 pilotos com maior incidência de desclassificações na carreira, indicando comportamento de risco ou irregularidades técnicas frequentes.
- **Lógica Técnica:** Filtra pelo status "Desclassificado" (`id_status = 2`) e ordena o ranking de forma decrescente.
- **Script SQL:**

```

select dim_drivers.name_driver as nome_piloto, count(fato_result.id_status) as qtd_desclassificacoes
from dim_drivers
join fato_result on fato_result.id_driver = dim_drivers.id_driver
where fato_result.id_status = 2
group by nome_piloto
order by qtd_desclassificacoes desc
limit 5;

```

A-Z nome_piloto	123 qtd_desclassificacoes
Bellof Stefan	11
Brundle Martin	8
Johansson Stefan	4
de Angelis Elio	3
Lauda Niki	3

5.4. Performance em Classificação (Qualifying)

- **Descrição:** Identifica os 3 pilotos com maior número de *Pole Positions* (largada na primeira posição), um indicador direto de velocidade pura.
- **Lógica Técnica:** Contabiliza os registros onde a coluna *position_grid* é igual a 1.
- **Script SQL:**

```

select dim_drivers.name_driver as nome_piloto, count(fato_result.position_grid) as qtd_pole_positions
from dim_drivers
join fato_result on fato_result.id_driver = dim_drivers.id_driver
where fato_result.position_grid = 1
group by nome_piloto
order by qtd_pole_positions desc
limit 3;

```

A-Z nome_piloto	123 qtd_pole_positions
Hamilton Lewis	71
Schumacher Michael	68
Senna Ayrton	65

5.5. Histórico de Infrações por Temporada

- **Descrição:** Apresenta os 10 anos (temporadas) com maior volume de penalidades severas (desclassificações e quebras de regras).
- **Lógica Técnica:** Filtra uma lista específica de IDs de status (2, 54, 81, 96, 97) que correspondem a diferentes tipos de infrações regulamentares.
- **Script SQL:**


```

select dim_season.year_season as ano_season, count(fato_result.id_status) as qtd_regras_quebradas
from dim_season
join fato_result on fato_result.id_season = dim_season.id_season
where fato_result.id_status in(2, 54, 81, 96, 97)
group by ano_season
order by qtd_regras_quebradas desc
limit 10;

```

Season 1 X

dim_season,year_season as ano_season, c| Insira uma expressão SQL para filtrar os resultados (use Ctrl+Espaço)

123 ano_season	123 qtd_regras_quebradas
1.984	27
1.982	13
1.988	11
1.983	10
1.989	10
1.978	9
1.990	7
1.974	6
1.971	5
1.964	5

5.6. Recordistas de Velocidade (Volta Mais Rápida)

- **Descrição:** Informa qual piloto detém o registro de volta mais rápida em cada circuito presente no banco de dados.
- **Lógica Técnica:** Seleciona o registro com o menor tempo de fastestlap_result para cada circuito distinto, ignorando valores nulos.
- **Script SQL:**

```

select * from(select distinct on (dim_circuits.name_circuit)
dim_circuits.name_circuit as nome_circuito, dim_drivers.name_driver as nome_piloto,
fato_result.fastestlap_result as volta_rapida
from dim_drivers
join fato_result on fato_result.id_driver = dim_drivers.id_driver
join dim_circuits on dim_circuits.id_circuit = fato_result.id_circuit
where fato_result.fastestlap_result is not null
order by nome_circuito, volta_rapida) as voltas_rapidas_pistas
order by volta_rapida;

```

circuits(+) 1 X

from(select distinct on (dim_circuits.name_circuit) as nome_circuito, dim_drivers.name_driver as nome_piloto, fato_result.fastestlap_result as volta_rapida) as voltas_rapidas_pistas order by volta_rapida;

AZ nome_circuito	AZ nome_piloto	⌚ volta_rapida
Albert Park Grand Prix Circuit	Hamilton Lewis	00:00:02
Circuit de Spa-Francorchamps	Ricciardo Daniel	00:00:02
Circuit Gilles Villeneuve	Montagny Franck	00:00:02
Circuit of the Americas	Hülkenberg Nico	00:00:02
Hockenheimring	Trulli Jarno	00:00:02
Indianapolis Motor Speedway	Albers Christijan	00:00:02
Marina Bay Street Circuit	Grosjean Romain	00:00:02
Nürburgring	Räikkönen Kimi	00:00:02
Red Bull Ring	Kvyat Daniil	00:00:02
Sepang International Circuit	Grosjean Romain	00:00:02
Shanghai International Circuit	Trulli Jarno	00:00:02
Silverstone Circuit	Kovalainen Heikki	00:00:02
Suzuka Circuit	Schumacher Ralf	00:00:02
Autódromo José Carlos Pace	Yamamoto Sakon	00:00:02
Bahrain International Circuit	di Grassi Lucas	00:00:02
Baku City Circuit	Kvyat Daniil	00:00:02
Buddh International Circuit	Glock Timo	00:00:02
Circuit de Monaco	Sato Takuma	00:00:02
Korean International Circuit	Kobayashi Kamui	00:00:03
Circuit de Barcelona-Catalunya	Bruni Gianmaria	00:00:03

5.7. Confiabilidade Mecânica das Equipes

- **Descrição:** Analisa as equipes com maior índice de falhas técnicas (motor, câmbio, hidráulica, etc.), impactando o resultado final do campeonato.
- **Lógica Técnica:** Utiliza uma cláusula IN com IDs específicos de falhas mecânicas (36, 59, 35...) para isolar erros operacionais de acidentes de pista.
- **Script SQL:**

```

select dim_constructor.name_constructor as nome_equipe,
count(fato_result.id_status) as erros_operacionais_equipe
from dim_constructor
join fato_result on fato_result.id_constructor = dim_constructor.id_constructor
where fato_result.id_status in (36, 59, 35, 69, 91, 134, 28, 58, 90, 75)
group by dim_constructor.name_constructor
order by erros_operacionais_equipe desc
limit 7;

```

AZ nome_equipe	123 erros_operacionais_equipe
Minardi	11
Toleman	10
Tyrrell	9
Alfa Romeo	9
Hesketh	9
Ferrari	8
Ensign	8

6. Conclusão

O projeto de migração de dados da Fórmula 1 para o Data Warehouse foi concluído com sucesso. A nova arquitetura em *Star Schema* simplificou drasticamente a complexidade das consultas, reduzindo a necessidade de múltiplos *joins* e facilitando a leitura dos dados.

Como demonstrado na seção de Validação, o ambiente analítico é capaz de responder a todas as perguntas de negócio propostas inicialmente, fornecendo métricas precisas sobre segurança, performance de pilotos e confiabilidade das construtoras. O Data Warehouse encontra-se agora pronto para ser conectado a ferramentas de visualização (Power BI) para a criação de dashboards gerenciais.