# Introdução

Abaixo encontram-se as soluções para cada uma das 4 questões do exercício – Aplicação lavoura Demonstrativas, da Yara International. Nos exercícios com respostas técnicas, será enviado, por e-mail, um diretório zipado com os arquivos e também um link para acesso ao repositório de arquivos no seguinte link:

Github: <https://github.com/DennCardoso/Yara-International---Lavouras-Demonstrativas>

Qualquer dúvida sobre a execução, estou à disposição no e-mail: [dennis.cardoso@outlook.com](mailto:dennis.cardoso@outlook.com)

# Premissas do exercício

* *Cada relatório pode ter um ou mais tratamentos  (Yara e/ou Atual)*
* *Cada relatório está associado a um único cliente e a um único talhão ( item da tabela Lavoura)*
* *Cada tratamento está associado a apenas um tipo de cultivo*
* *Cada tratamento pode ter um ou mais produtos associados*
* *Cada da item da tabela Lavoura (Talhão) pode ter apenas um tipo de cultivo  (SistemaProdução ou CulturaDePousio)*
* *O campo* ***TamColhido*** *da Tabela lavoura contém a área do talhão medida em hectares.*
* *A produtividade absoluta de cada Tratamento é dada pelo campo* ***ProdutividadeDoTratamento*** *dada em número de sacas.*

# Questões e soluções

# Exercício

*Diversos problemas estão presentes na estrutura da base de dados fornecida, alguns deles são por exemplo:*

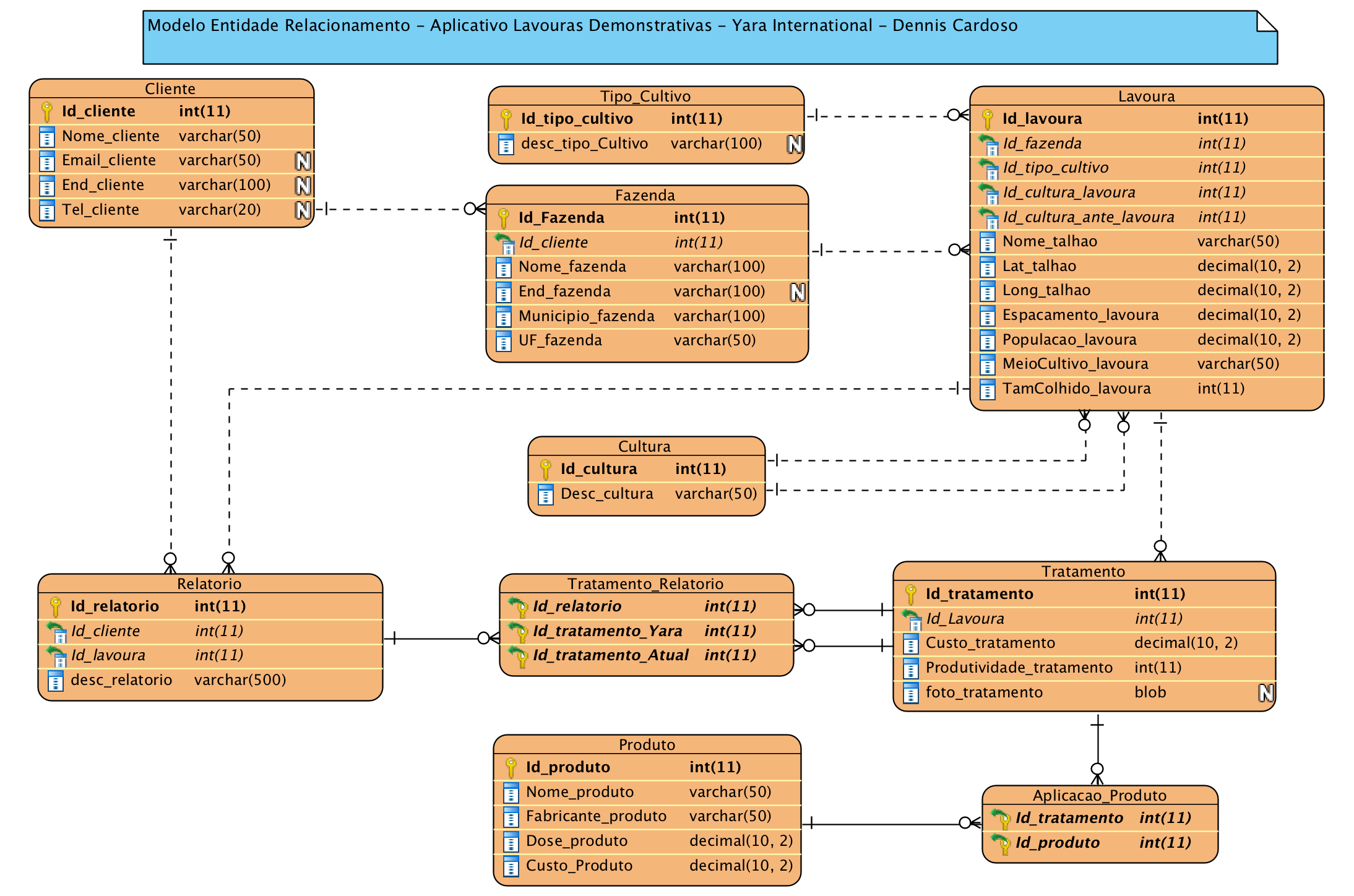
* *Dados de fazendas e clientes na mesma tabela*
* *Tabelas TratamentoYara e TratamentoAtual duplicadas*
* *Nomes de campos repetidos e sem estrutura*

*Com base no contexto fornecido no modelo json (slides anteriores) e campos relevantes, crie um diagrama EER modelando uma nova estrutura de banco de dados com dados normalizados (reduzir a redundância de dados, aumentar a integridade de dados e o desempenho) facilitando a construção de* ***relatórios de produtividade para diversos  agregadores.***

## Solução:

Baseado no retorno dos arquivos json e, principalmente, no entendimento da regra de negócio, temos abaixo o modelo de dados para o aplicativo Lavouras Demonstrativas.

A aplicação utilizada para o desenvolvimento do modelo ER é Visual Paradigm. O arquivo pode ser acessado no Github como também no arquivo zip (***Yara - Sistema de Lavouras Demonstrativas.vpp).***



Resposta do modelo em relação às premissas:

* Cada relatório pode ter um ou mais tratamentos  (Yara e/ou Atual)
  + Cada relatório possui N tratamentos, onde estão relacionadas na tabela “Tratamento\_Relatório”. Além disso, considerei que um tratamento pode ter N relatórios, por isso existe uma tabela de relacionamento entre as entidades Relatorio e tratamento (many-to-many).
* Cada relatório está associado a um único cliente e a um único talhão (item da tabela Lavoura)
  + Seguindo a lógica do relacionamento entre as entidades, a tabela *relatório* contém as chaves estrangeiras: *Id\_cliente* (tabela cliente) e *Id\_lavoura* (Tabela Lavoura). A chave primária da tabela é *id\_relatório,* o que garante a unicidade da tabela relatório e o registro de apenas um talhão e um cliente por relatório cadastrado.

* Cada tratamento está associado a apenas um tipo de cultivo
  + Um tratamento contem apenas um cultivo por meio da chave da lavoura, uma foreign key na entidade *tratamento*.
* Cada tratamento pode ter um ou mais produtos associados
  + Para atender essa premissa, considerei que para um tratamento pode existir N produtos e para cada Produto pode existir N tratamentos. Portanto, a tabela de relacionamento *Aplicacao\_Produto* foi criada para compor o relacionamento N para N entre as duas entidades: *tratamento* e *Produto* e garantir a premissa que um tratamento pode ter diversos produtos aplicados.
* Cada item da tabela Lavoura (Talhão) pode ter apenas um tipo de cultivo  (SistemaProdução ou CulturaDePousio)
  + Considerei que ‘Sistema Produção’ e ‘Cultura de Pousio’, pela sua definição de negócio, tipos de Cultivo diferentes. Desta maneira, foi criada a tabela domínio *Tipo\_Cultivo* para registrar o valor do cultivo e a descrição. Na tabela *Lavoura* foi criada a coluna *id\_tipo\_cultivo* para armazenar qual o tipo de cultivo utilizado, garantindo, portanto, a premissa acima.
* O campo **TamColhido** da Tabela lavoura contém a área do talhão medida em hectares.
  + TamColhido foi definido como a coluna *TamColhido\_lavoura* da tabela *Lavoura*, Decimal, com 10 bytes e 2 casas após ponto flutuante.
* A produtividade absoluta de cada Tratamento é dada pelo campo **ProdutividadeDoTratamento** dada em número de sacas.
  + ProdutividadeDoTratamento foi definido como a coluna *Produtividade\_tratamento* na tabela *Tratamento*, Decimal, com 10 bytes e 2 casas após ponto flutuante.

# Exercício

*Dada a estrutura que você criou no exercício 1, construa scripts DDL para gerar as tabelas necessárias e crie arquivos com as consultas para produzir um relatório que indique:*

* + *A média  da* ***diferença  de produtividade******por  área cultivada*** *entre tratamentos  para cada cultura*
  + *A média da* ***diferença  relativa de produtividade******por  área cultivada*** *entre tratamentos por* ***UF***

*Em todos itens analise a* ***performance*** *das consultas.*

## Solução:

O DLL para criação da estrutura da base de dados pode ser acessado no arquivo **Criacao\_Database\_yara.sql**, bem como no código fonte abaixo:

#Criacao **do** **Database** yara

#**create** **database** yara;

#Usar **database** yara

use yara;

#Criacao da Tabela Cliente

**CREATE** **TABLE** Cliente(

Id\_cliente int **NOT** **NULL** AUTO\_INCREMENT,

Nome\_cliente varchar(50) **NOT** **NULL**,

Email\_cliente varchar(50),

End\_cliente varchar(100),

Tel\_cliente varchar(20),

**CONSTRAINT** PK\_Cliente **PRIMARY** **KEY** (Id\_Cliente)

);

#Criacao da Tabela Fazenda

**CREATE** **TABLE** Fazenda(

Id\_fazenda int **NOT** **NULL** AUTO\_INCREMENT,

Id\_cliente int **NOT** **NULL**,

Nome\_fazenda varchar(100) **NOT** **NULL**,

End\_fazenda varchar(100) **NOT** **NULL**,

Municipio\_fazenda varchar(100) **NOT** **NULL**,

UF\_fazenda varchar(50) **NOT** **NULL**,

**CONSTRAINT** PK\_Fazenda **PRIMARY** **KEY**(Id\_fazenda),

**FOREIGN** **KEY** (Id\_cliente) **REFERENCES** Cliente(Id\_cliente)

);

#Criacao da Tabela Tipo Cultivo

**CREATE** **TABLE** Tipo\_Cultivo(

Id\_tipo\_Cultivo int **NOT** **NULL** AUTO\_INCREMENT,

Desc\_tipo\_cultivo varchar(100),

**CONSTRAINT** PK\_Tipo\_Cultivo **PRIMARY** **KEY**(Id\_tipo\_Cultivo)

);

#Criacao da Tabela Cultura

**CREATE** **TABLE** Cultura(

Id\_cultura int **NOT** **NULL** AUTO\_INCREMENT,

Desc\_cultura varchar(50),

**CONSTRAINT** PK\_Cultura **PRIMARY** **KEY** (Id\_cultura)

);

#Criacao da Tabela Lavoura

**CREATE** **TABLE** Lavoura(

Id\_Lavoura int **NOT** **NULL** AUTO\_INCREMENT,

Id\_fazenda int **NOT** **NULL**,

Id\_tipo\_cultivo int **NOT** **NULL**,

Id\_cultura\_lavoura int **NOT** **NULL**,

Id\_cultura\_ante\_lavoura int,

Nome\_talhao varchar(50) **NOT** **NULL**,

Lat\_talhao decimal(10,2) **NOT** **NULL**,

Long\_talhao decimal(10,2) **NOT** **NULL**,

Espacamento\_lavoura decimal(10,2) **NOT** **NULL**,

Populacao\_lavoura decimal(10,2) **NOT** **NULL**,

MeioCultivo\_lavoura varchar(50) **NOT** **NULL**,

TamColhido\_lavoura int **NOT** **NULL**,

**CONSTRAINT** PK\_Lavoura **PRIMARY** **KEY**(Id\_Lavoura),

**FOREIGN** **KEY** (Id\_fazenda) **REFERENCES** Fazenda(Id\_fazenda),

**FOREIGN** **KEY** (Id\_tipo\_cultivo) **REFERENCES** Tipo\_Cultivo(Id\_tipo\_cultivo),

**FOREIGN** **KEY** (Id\_cultura\_lavoura) **REFERENCES** Cultura (Id\_Cultura),

**FOREIGN** **KEY** (Id\_cultura\_ante\_lavoura) **REFERENCES** Cultura (Id\_cultura)

);

#Criacao da Tabela Relatório

**CREATE** **TABLE** Relatorio(

Id\_relatorio int **NOT** **NULL** AUTO\_INCREMENT,

Id\_Cliente int **NOT** **NULL**,

Id\_Lavoura int **NOT** **NULL**,

Desc\_relatorio varchar(500) **NOT** **NULL**,

**CONSTRAINT** PK\_relatorio **PRIMARY** **KEY**(Id\_relatorio),

**FOREIGN** **KEY** (Id\_cliente) **REFERENCES** Cliente(Id\_Cliente),

**FOREIGN** **KEY** (Id\_Lavoura) **REFERENCES** Lavoura(Id\_Lavoura)

);

#Criacao da Tabela Tratamento

**CREATE** **TABLE** Tratamento(

Id\_tratamento int **NOT** **NULL** AUTO\_INCREMENT,

Id\_lavoura int **NOT** **NULL**,

Custo\_tratamento decimal(10,2) **NOT** **NULL**,

Produtividade\_tratamento int **NOT** **NULL**,

foto\_tratamento blob,

**CONSTRAINT** PK\_tratamento **PRIMARY** **KEY** (Id\_tratamento),

**FOREIGN** **KEY** (Id\_lavoura) **REFERENCES** Lavoura(Id\_lavoura)

);

#Criacao da Tabela Tratamento\_relatorio

**CREATE** **TABLE** Tratamento\_Relatorio(

Id\_relatorio int **NOT** **NULL**,

Id\_tratamento\_yara int **NOT** **NULL**,

Id\_tratamento\_atual int **NOT** **NULL**,

**CONSTRAINT** PK\_tratamento\_relatorio **PRIMARY** **KEY** (Id\_relatorio, Id\_tratamento\_yara, Id\_tratamento\_atual),

**FOREIGN** **KEY** (Id\_relatorio) **REFERENCES** Relatorio(Id\_relatorio),

**FOREIGN** **KEY** (Id\_tratamento\_yara) **REFERENCES** Tratamento(Id\_tratamento),

**FOREIGN** **KEY** (Id\_tratamento\_atual) **REFERENCES** Tratamento(Id\_Tratamento)

);

#Criacao da Tabela Produto

**CREATE** **TABLE** Produto(

Id\_produto int **NOT** **NULL** AUTO\_INCREMENT,

Nome\_produto varchar(50) **NOT** **NULL**,

Fabricante\_produto varchar(50) **NOT** **NULL**,

Dose\_produto decimal(10,2) **NOT** **NULL**,

Custo\_produto decimal(10,2) **NOT** **NULL**,

**CONSTRAINT** PK\_Produto **PRIMARY** **KEY** (Id\_produto)

);

#Criacao da Tabela Aplicacao\_Produto

**CREATE** **TABLE** Aplicacao\_Produto(

Id\_tratamento int **NOT** **NULL**,

Id\_produto int **NOT** **NULL**,

**CONSTRAINT** PK\_Aplicacao\_Produto **PRIMARY** **KEY** (Id\_tratamento, Id\_produto),

**FOREIGN** **KEY** (Id\_tratamento) **REFERENCES** Tratamento(Id\_tratamento),

**FOREIGN** **KEY** (Id\_produto) **REFERENCES** Produto(Id\_produto)

);

* *A média  da* ***diferença  de produtividade******por  área cultivada*** *entre tratamentos  para cada cultura*

O desenvolvimento dessa query pode ser encontrado no arquivo **Exercicio2\_relatorio\_DiffCultura.sql,** também como no código abaixo:

*/\**

*O select busca a média da diferença de Produtividade por área entre os tratamentos Yara e Atual, agrupados por Cultura*

*Para isso buscamos os seguintes dados: Código da Cultura, Descrição da cultura.*

*Realizamos os seguinte calculo.:*

*- Para cada comparação no relatório, temos a produtividade do tratamento Yara e do Tratamento Atual.*

*- É realizado a subtração entre Prod Yara e Prod atual e esse resultado é dividido pelo TamColhido(Ha)\*10000 (transformação e Hectares para Metros).*

*- Desta maneira temos a quantidade da diferença entre produtividade por metro quadrado (m2).*

*- Por fim, agrupamos a média desse valor por Cultura para finalmente obtermos o valor.*

*\*/*

**select**

l.Id\_cultura\_lavoura 'Código Cultura',

**c**.desc\_cultura 'cultura',

**avg**((ty.Produtividade\_tratamento - ta.Produtividade\_tratamento)/l.TamColhido\_lavoura\*10000) 'Media da Diferença de Produtividade por áreas entre tratamento'

**from** Tratamento\_Relatorio tr

**inner** **join** Tratamento ty

**on** tr.id\_tratamento\_yara = ty.Id\_tratamento

**inner** **join** Tratamento ta

**on** tr.id\_tratamento\_atual = ta.Id\_tratamento

**inner** **join** relatorio r

**on** tr.id\_relatorio = r.Id\_relatorio

**inner** **join** lavoura l

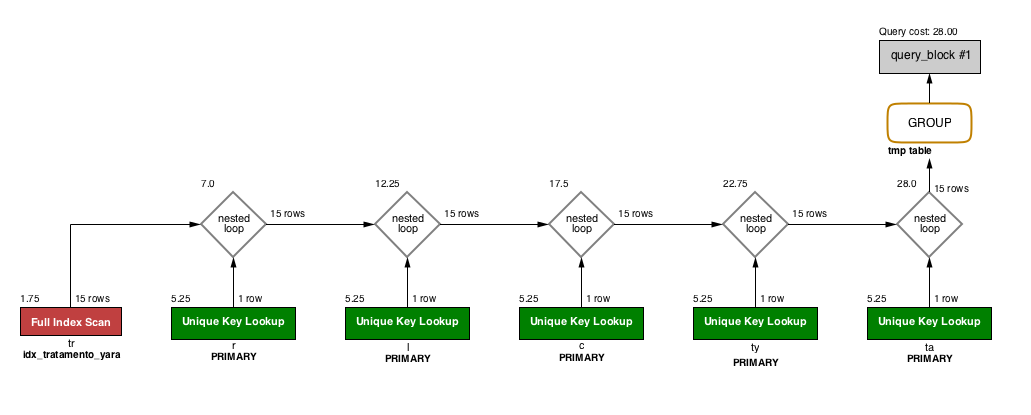
**on** r.id\_lavoura = l.id\_lavoura

**inner** **join** cultura **c**

**on** l.id\_cultura\_lavoura = **c**.Id\_cultura

**group** **by** l.Id\_cultura\_lavoura, **c**.desc\_cultura;

A performance da query acima pode ser compreendida da seguinte maneira:



|  |
| --- |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |
|  |

* Foi realizado um Full Index Scan na tabela *Tratamento\_Relatorio* para acesso a todos os itens do tratamento.
* Nas demais tabelas, foram realizadas Unique Key Lookup. Isso significa que, durante a realização do comando inner join, as queries tiveram custo baixo devido ao uso da primary key no relacionamento (e consequentemente, a unique key constraint).
* *A média  da diferença  de produtividade   por  área cultivada  entre tratamentos  para cada cultura*

O desenvolvimento dessa query pode ser encontrado no arquivo **Exercicio2\_Relatorio\_DiffUF.sql,** também como no código abaixo:

*/\**

*O select busca a média da diferença de Produtividade por área entre os tratamentos Yara e Atual, agrupados por UF*

*Para isso buscamos os seguintes dado: UF.*

*Realizamos os seguinte calculo.:*

*- Para cada comparação no relatório, temos a produtividade do tratamento Yara e do Tratamento Atual.*

*- É realizado a subtração entre Prod Yara e Prod atual e esse resultado é dividido pelo TamColhido(Ha)\*10000 (transformação e Hectares para Metros quadrados).*

*- Desta maneira temos a quantidade da diferença entre produtividade por metro quadrado (m2).*

*- Por fim, agrupamos a média desse valor por UF para finalmente obtermos o valor.*

*\*/*

**select**

f.UF\_fazenda 'UF',

**avg**((ty.Produtividade\_tratamento - ta.Produtividade\_tratamento)/l.TamColhido\_lavoura\*10000) 'Media da Diferença de Produtividade por áreas entre tratamento'

**from** Tratamento\_Relatorio tr

**inner** **join** Tratamento ty

**on** tr.id\_tratamento\_yara = ty.Id\_tratamento

**inner** **join** Tratamento ta

**on** tr.id\_tratamento\_atual = ta.Id\_tratamento

**inner** **join** relatorio r

**on** tr.id\_relatorio = r.Id\_relatorio

**inner** **join** lavoura l

**on** r.id\_lavoura = l.id\_lavoura

**inner** **join** cultura **c**

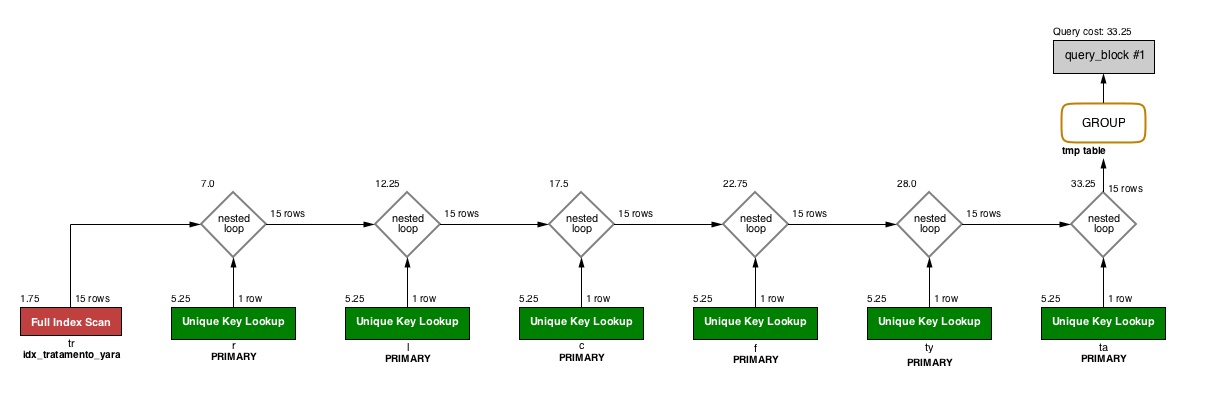
**on** l.id\_cultura\_lavoura = **c**.Id\_cultura

**inner** **join** fazenda f

**on** l.id\_fazenda = f.id\_fazenda

**group** **by** f.UF\_Fazenda

A performance da query acima pode ser compreendida da seguinte maneira:



Igualmente a query anterior, temos os mesmos dados de execução e performance:

* + Foi realizado um Full Index Scan na tabela *Tratamento\_Relatorio* para acesso a todos os itens do tratamento.
  + Nas demais tabelas, foram realizadas Unique Key Lookup. Isso significa que, durante a realização do comando inner join, as queries tiveram custo baixo devido ao uso da primary key no relacionamento (e consequentemente, a unique key constraint).

# Exercício

Qual seria o pipeline que você construiria para migração desses dados para um datalake? Descreva em linhas gerais e justifique suas escolhas.

## Solução:

Para criação de um fluxo de dados entre uma base de dados estruturada para um Data Lake, podemos imaginar em uma infraestrutura de DataLake já desenvolvida, muito similar a essa que podemos ver no exemplo abaixo (imagem 1).

Basicamente, é possível transferir os dados criando um fluxo ETL (Extract, Transform and and Load). Para isso, podemos utilizar aplicações especificas para migração de dados como, por exemplo, o Apache Scoop.

O apache Scoop pode ser utilizado como fluxo ETL para transmitir dados de um banco de dados externo (fora do ambiente big data) para o ambiente interno do lake (que pode ser um RDBMS. Existem diversas distribuições do apache Scoop, como por exemplo, a distribuída gratuitamente, bem como de empresas como Hortonworks e Cloudera.

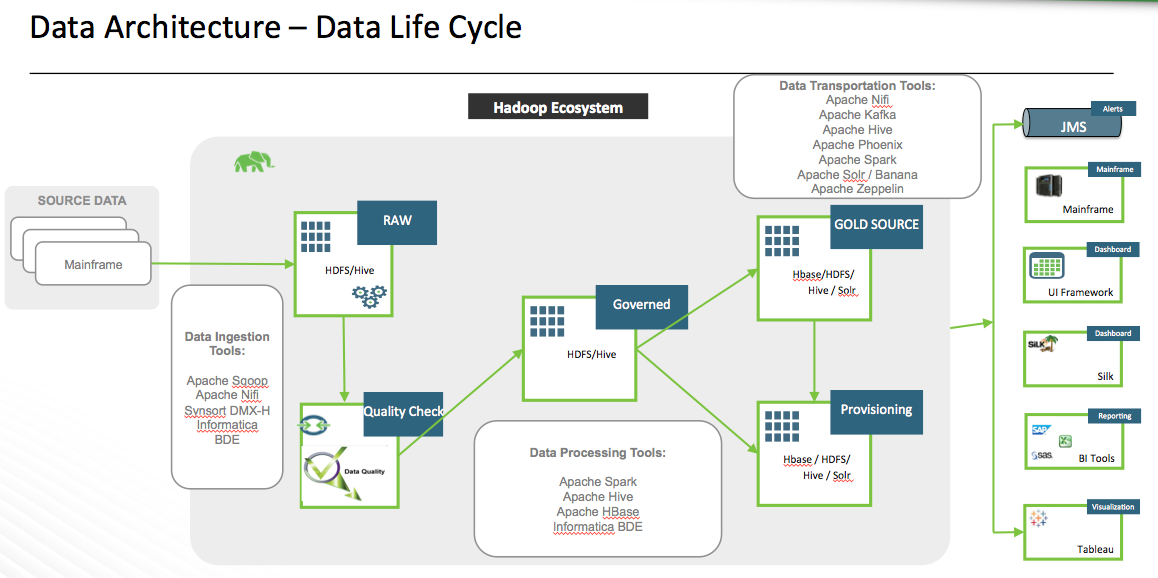


Figura 1 - Arquitetura de dados de um Data Lake - Fonte da imagem: https://hortonworks.com/apache/spark/

# Exercício

Caso a aplicação evolua e inclua dados de sensores, como você lidaria com a presença de estruturas de dados heterogêneas (dados relacionais, dados contínuos no tempo)?

Definições de negócio:

* Pousio (ou poisio entre outro milhões de espec.), em [agricultura](https://pt.wikipedia.org/wiki/Agricultura), é nome que se dá ao descanso ou repouso proporcionado às terras cultiváveis, interrompendo lhe as culturas para tornar o solo mais fértil.
* Talhão nada mais é do que a unidade mínima de cultivo de uma propriedade que é construído com base em relevo e planejamento de mecanização.

Biografia:

http://inteliagro.com.br/o-que-e-talhao/

https://pt.wikipedia.org/wiki/Pousio