



## **Aplicações em ETL**

**Bootcamp: Desenvolvedor Business Intelligence**

**Ricardo Brito Alves**

**2020**

## **Aplicações em ETL**

Bootcamp: Desenvolvedor Business Intelligence

Ricardo Brito Alves

© Copyright do Instituto de Gestão e Tecnologia da Informação.

Todos os direitos reservados.

## Sumário

---

Capítulo 1. Processo ETL.....	7
Definição de BI.....	7
OLAP .....	8
Relatórios Ad hoc.....	9
Integração de dados .....	10
Definição de ETL .....	11
Como surgiu o ETL.....	12
Extração.....	13
Transformação.....	14
Carga .....	14
Importância do ETL.....	15
Benefícios do ETL.....	17
Principais Conceitos em ETL.....	17
Capítulo 2. Ferramentas ETL e Tendências .....	22
Quadrante Mágico de Gartner .....	22
Ferramentas ETL .....	23
Big Data .....	24
Data Analytics .....	24
Principais Fornecedores .....	26
Pentaho .....	27
Visão Geral do Pentaho.....	27
Pentaho Data Integration .....	27
Benefícios do Pentaho .....	28
Pentaho Data Integration Suite .....	29

Outras ferramentas ETL (Apache Hadoop, NoSQL).....	29
Capítulo 3. Requisitos para ETL.....	31
Etapas do Projeto de BI .....	31
Desafios da Implantação de um projeto de BI .....	33
Componentes críticos para escolha da ferramenta.....	35
Requisitos para ETL .....	36
Componentes Críticos .....	38
Definição de Ambiente.....	38
Escalabilidade.....	39
Recuperabilidade ou Rerunnability .....	40
Chaves.....	40
Performance .....	40
Processamento Paralelo.....	42
Ferramentas.....	42
Capítulo 4. Modelagem Dimensional.....	44
Modelo Star Schema .....	45
Modelo Snowflake .....	45
Modelo StarFlake.....	46
OLAP .....	48
Regras essenciais para modelagem dimensional.....	49
Tabela Dimensão.....	50
Hierarquia de Dimensões .....	51
Surrogate Key .....	51
Tipos de Dimensão .....	52
Degerate Dimension .....	52
Junk Dimension .....	53

Role-Playing Dimension.....	54
Conformed Dimension .....	54
Slowly Changing Dimension (SCD) .....	54
Tabela Fato.....	56
Fato Transacional .....	57
Fato Agregada .....	57
Fato Consolidada.....	57
Fato SnapShot Periódico .....	58
Fato SnapShot Acumulados .....	58
Fato sem Fato.....	58
Métricas .....	59
Tipos de Métricas.....	59
Métrica Aditiva .....	59
Métrica Derivada.....	60
Métrica Semi-Aditiva .....	60
Métrica Não-Aditiva.....	60
Capítulo 5. Metadados .....	61
Papel dos metadados no data warehouse .....	62
Repositório de Metadados em ETL.....	68
Como os metadados do data warehousing podem ser gerenciados? .....	69
Desafios para o gerenciamento de metadados.....	70
O que é ETL orientado a metadados? .....	70
Capítulo 6. Staging Area .....	71
ODS – Operational Data Store.....	71
Diferenças entre ODS e DW .....	72
Staging area.....	72

Tipos de Staging area.....	73
Staging 1.....	74
Staging 2.....	74
Staging 2 Aux.....	74
Tratamento de erros .....	75
Categoria de erros .....	75
Tipos de erros .....	76
Tratamento de erros .....	76
Glossário Pentaho.....	77
Referências.....	80

## Capítulo 1. Processo ETL

---

### Definição de BI

---

BI é um termo criado pelo Gartner Group, que define como "um termo genérico que inclui aplicações, infraestrutura, ferramentas e melhores práticas que permite o acesso e análise de informações para melhorar e otimizar decisões e desempenho". É um processo de coleta, transformação, análise e distribuição de dados para melhorar a decisão dos negócios.

Descreve a capacidade de a empresa ter acesso e explorar seus dados, desenvolvendo percepção e conhecimento, o que leva à melhora do processo de tomada de decisões. Sua infraestrutura tecnológica é composta de Data Warehouse ou Data Marts, data mining e ODS, além das ferramentas pertinentes. Os principais processos uma aplicação de Business Intelligence são:

- Modelagem Dimensional.
- ETL.
- OLAP.

O BI pode ser usado para adquirir insights táticos para otimizar processos de negócios, identificando tendências, anomalias e comportamentos que requerem ação de gerenciamento e também para visão estratégica, para alinhar vários processos de negócios aos principais objetivos de negócios por meio de gerenciamento e análise de desempenho integrados. Além disso, o BI proporciona uma tomada de decisão baseada em fatos e uma visão única dos dados.

**Figura 1 - Conceito de pirâmide do BI.**

Fonte: <https://www.dreamstime.com>.

## OLAP

---

OLAP é uma ferramenta de Business Inteligente utilizada para apoiar as empresas na análise de suas informações, visando obter novos conhecimentos que são empregados na tomada de decisão.

O termo OLAP refere-se a um conjunto de ferramentas voltadas para acesso e análise ad hoc de dados, com o objetivo final de transformar dados em informações capazes de dar suporte as decisões gerenciais de forma amigável e flexível ao usuário e em tempo hábil.

OLAP trouxe uma grande capacidade de efetuar cálculos complexos como previsões, percentuais de crescimento e médias diversas considerando-se a variável tempo. É uma ferramenta muito importante no contexto gerencial, ajudando a analisar de forma mais eficiente, a quantidade de dados crescente armazenada pelas organizações, transformando-os em informação útil. As ferramentas OLAP proporcionam condições de análise de dados on-line necessárias para responder as possíveis perguntas dos analistas, gerentes e executivos. São aplicações que os



usuários finais têm acesso para extraírem os dados de suas bases e construir os relatórios capazes de responder as suas questões gerenciais.

Elas surgiram juntamente com os sistemas de Apoio à Decisão para fazerem a consulta e análise dos dados contidos nos Data Warehouses e Data Mart.

## Relatórios Ad hoc

---

Bill Inmon (2002), conceitua a consulta ad-hoc como:

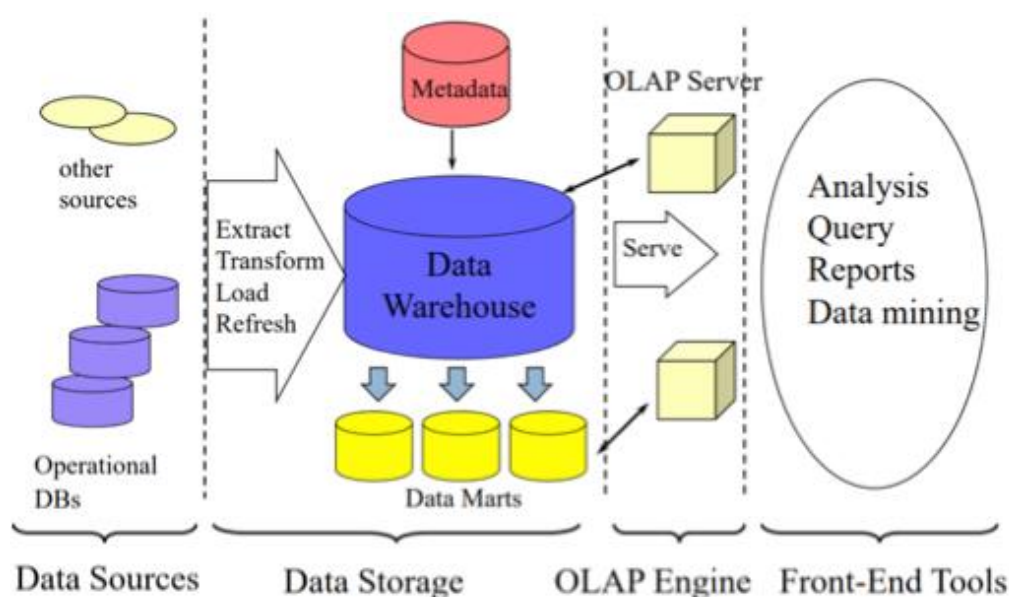
São consultas com acesso casual único e tratamento dos dados segundo parâmetros nunca antes utilizados, geralmente executado de forma iterativa e heurística. Isso tudo nada mais é do que o próprio usuário gerar consultas de acordo com suas necessidades de cruzar as informações de uma forma não vista e com métodos que o levem a descoberta daquilo que procura. (INMON, 2002)

Uma das características que devem estar presentes em ferramentas OLAP é a capacidade de efetuar algumas operações, como:

- **Drill Across:** ocorre quando o usuário pula um nível intermediário dentro de uma mesma dimensão. Por exemplo, a dimensão tempo é composta por ano, semestre, trimestre, mês e dia. A operação Drill Across é executada quando o usuário passa de ano direto para trimestre ou mês.
- **Drill Down:** ocorre quando o usuário aumenta o nível de detalhe da informação, diminuindo a granularidade. Ela influencia diretamente na velocidade do acesso às informações e no volume de dados armazenados.
- **Drill Up:** é o contrário do Drill Down, ocorre quando o usuário aumenta a granularidade, diminuindo o nível de detalhamento da informação.
- **Drill Throught:** ocorre quando o usuário passa de uma informação contida em uma dimensão para uma outra. Por exemplo: Inicia na dimensão do tempo e no próximo passo analisa a informação por região.

- **Slice and Dice:** é uma das principais características de uma ferramenta OLAP. Como a ferramenta OLAP recupera o cubo, surgiu a necessidade de criar um módulo, que se convencionou de Slice and Dice, para ficar responsável por trabalhar esta informação. Ele serve para modificar a posição de uma informação, trocar linhas por colunas de maneira a facilitar a compreensão dos usuários e girar o cubo sempre que tiver necessidade.

**Figura 2 - Componentes do BI.**



## Integração de dados

A integração de dados ou *data integration* envolve a combinação de dados residentes em diferentes fontes e fornece aos usuários uma visão unificada desses dados. Esse processo se torna significativo em uma variedade de situações, incluindo domínios comerciais (como quando duas empresas similares precisam mesclar seus bancos de dados) e científicos (combinando resultados de pesquisa de diferentes repositórios de bioinformática, por exemplo). A integração de dados aparece com frequência crescente à medida que o volume (ou seja, big data) e a necessidade de compartilhar dados existentes explodem. Tornou-se o foco de extenso trabalho

teórico, e numerosos problemas em aberto permanecem sem solução. A integração de dados incentiva a colaboração entre usuários internos e externos.

## Definição de ETL

---

Na fase de integração dos dados executamos o ETL. Essa sigla significa *Extract, Transform and Load*, ou seja, Extração, Transformação e Carga.

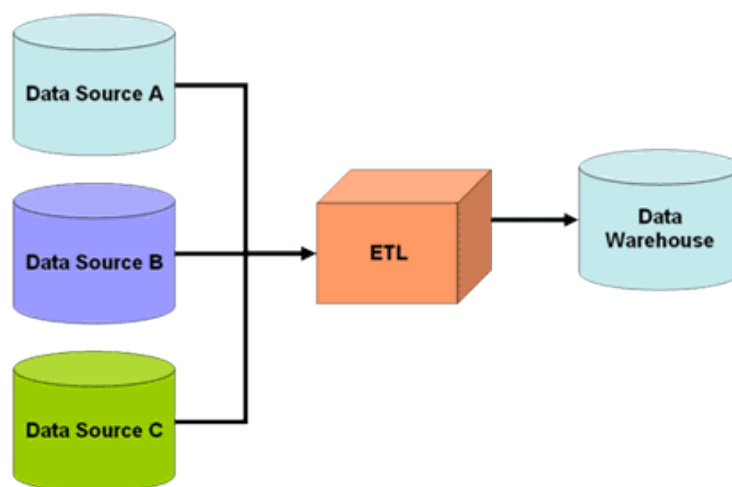
Comumente utilizado para construir um Data Warehouse. Nesse processo, os dados são retirados (extraídos) de um ou mais sistemas-fonte ou origens, convertidos (transformados) em um formato que possa ser analisado, e armazenados (carregados) em um armazém ou outro sistema.

Conforme Ralph Kimball:

Um sistema ETL projetado adequadamente extrai dados dos sistemas de origem, aplica padrões de qualidade e consistência de dados, alinha os dados para que fontes separadas possam ser usadas em conjunto e, finalmente, entrega dados em um formato pronto para apresentação, para que os desenvolvedores possam criar aplicativos e os usuários finais podem tomar decisões. (KIMBALL, 2004)

A extração e carga são obrigatórias para o processo, sendo a transformação/limpeza opcional, mas que são boas práticas, tendo em vista que os dados já foram encaminhados para o sistema de destino. É considerada uma das fases mais críticas do Data Warehouse e/ou Data Mart.

Os projetos de Data Warehouse consolidam dados de diferentes fontes. A maioria dessas fontes tende a ser bancos de dados relacionais ou arquivo de texto (texto plano), mas podem existir outras fontes. Uma aplicação ETL tem que ser capaz de se comunicar com as bases de dados e ler diversos formatos de arquivos utilizados por toda a organização. Essa pode ser uma tarefa não trivial, e muitas fontes de dados podem não ser acessadas com facilidade.

**Figura 3 – Ferramenta de ETL.**

Fonte: <http://gedxml.com.br/>.

### Como surgiu o ETL

---

ETL ganhou popularidade nos anos 1970, quando as organizações começaram a usar múltiplos repositórios ou bancos de dados para armazenar diferentes tipos de informações de negócios. A necessidade de integrar os dados que se espalhavam pelos databases cresceu rapidamente. O ETL tornou-se o método padrão para coletar dados de fontes diferentes e transformá-los antes de carregá-los no sistema-alvo ou destino.

No final dos anos 1980 e início dos 1990, os data warehouses entraram em cena. Sendo um tipo diferente de banco de dados, eles forneceram um acesso integrado a dados de múltiplos sistemas – computadores mainframes, minicomputadores, computadores pessoais e planilhas. Mas diferentes departamentos costumam usar diferentes ferramentas ETL com diferentes armazéns. Adicione isso a junções e aquisições, e muitas empresas acabam com distintas soluções ETL, que não foram integradas.

Com o tempo, o número de formatos, fontes e sistemas de dados aumentou muito. Extrair, transformar e carregar é, hoje, apenas um dos vários métodos que as

organizações utilizam para coletar, importar e processar dados. ETL e ELT são, ambos, partes importantes de uma estratégia ampla de data integration das empresas.

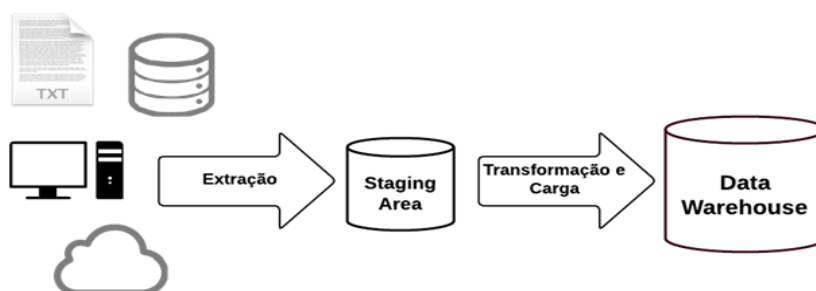
## Extração

A extração é a primeira etapa de um processo de ETL, sendo a coleta de dados dos sistemas de origem (Data Sources ou sistemas operacionais) extraindo-os e transferindo-os para o ambiente de DW, onde o sistema ETL pode operar independente dos sistemas de origem. Um processo ETL precisa ser capaz de se comunicar com bases de dados, visto que os projetos de Data Warehouse consolidam dados de diferentes fontes, e ler diversos formatos de arquivos utilizados por toda a organização.

O primeiro passo é definir as fontes de extração e os dados podem vir das mais diversas fontes, por exemplo: Sistemas de gestão (SIG, ERP, CRM etc.), SGBD's (Oracle, SQLSERVER, DB2 etc.), arquivos como planilhas do Excel e documentos de texto.

A etapa de extração é a fase onde os dados são extraídos dos OLTPs (Online Transaction Processing) e conduzidos para a staging area (área temporária), onde são convertidos para um único formato. A conversão se faz necessária devido a heterogeneidade existente nas informações oriundas desses sistemas, sendo essencial a conformação prévia para o tratamento adequado.

**Figura 4 – Etapa de Extração de Dados.**



Fonte: <https://canaltech.com.br/>.

## Transformação

---

Após a extração, teremos subsídios para iniciar a etapa de transformação e limpeza dos dados. Nessa fase são corrigidos, padronizados e tratados os desvios e inconsistências, transformando os dados de acordo com as regras do negócio.

É nesta etapa que realizamos os devidos ajustes, podendo assim, melhorar a qualidade dos dados e consolidar dados de duas ou mais fontes. O estágio de transformação aplica uma série de regras ou funções aos dados extraídos para ajustar os dados a serem carregados.

O processo de transformação de dados é composto por vários subprocessos, como por exemplo:

- Limpeza - inconsistências e valores ausentes nos dados são resolvidos.
- Padronização - regra de formatação é aplicada ao conjunto de dados.
- Eliminar duplicados - dados redundantes são excluídos ou descartados.
- Verificação - dados inutilizáveis são removidos e anomalias são sinalizadas.
- Classificação - os dados são organizados de acordo com o tipo.
- Outras tarefas - quaisquer regras adicionais/opcionais podem ser aplicadas para melhorar a qualidade dos dados.

## Carga

---

A etapa de carga ocorre em sequência com a de transformação. Assim que são efetuados os tratamentos necessários nos dados, a carga no DW é iniciada. Essa fase se resume na persistência dos dados na base consolidada. Consiste em estruturar e carregar os dados.

Dentro de um mesmo DW temos diferentes períodos de execução para cada tipo de processo de carga. Alguns são mensais, outros diários. Neste momento também é definida a latência das informações, isso pode variar para cada tabela a ser carregada. Latência é sinônimo de atraso, é uma expressão de quanto tempo leva para um pacote de dados ir de um ponto designado para o outro.

As cargas podem ser full ou incrementais. A full ou total, se trata da carga completa dos dados toda vez que há a execução de um novo processo de ETL. Nesse tipo de carga todos os dados da origem são extraídos e transformados, recarregando os dados antigos e incrementando com os novos. Já a incremental considera apenas os novos registros dos sistemas operacionais no ETL, inserindo-os ao repositório do DW. A carga total é mais custosa que a incremental, tanto em tempo de carga, quanto em processamento (CPU).

## Importância do ETL

---

Há anos, inúmeras empresas têm confiado no processo de ETL para obter uma visão consolidada dos dados que geram as melhores decisões de negócios. Hoje, esse método de integrar dados de múltiplos sistemas e fontes, ainda é um componente central do kit de ferramentas de data integration de uma organização.

- Quando utilizado com um data warehouse corporativo (dados em repouso), o ETL fornece o contexto histórico completo para a empresa;
- Ao fornecer uma visão consolidada, o ETL facilita para os usuários corporativos a análise e a criação de relatórios sobre dados relevantes às suas iniciativas;
- O ETL pode melhorar a produtividade de profissionais analíticos, porque ele codifica e reutiliza processos que movem os dados sem que esses profissionais possuam a capacidade técnica de escrever códigos ou scripts;
- O ETL evoluiu ao longo do tempo para suportar os requisitos emergentes de integração para coisas como streaming data;

- As organizações precisam tanto de ETL quanto ELT para unir dados, manter a precisão e fornecer a auditoria necessária para armazenar dados, criar relatórios e realizar análises.

O processo ETL é uma das fases mais críticas na construção de um sistema DW, visto que:

- Grandes volumes de dados são processados.
- Serão implementadas as regras e fórmulas dos indicadores que irão compor as tabelas de Fato.
- É essencial para a criação das estruturas de Dimensões e Fatos no ambiente do DW. É ele que faz a “ponte” de ligação entre o operacional e o DW.

As ferramentas, que darão suporte ao processo, devem ser bem escolhidas, pois são essenciais para a correta execução das atividades do ETL. A janela de operação do ETL deve ser analisada, pois não é em qualquer momento que ele poderá ser executado e também analisar a periodicidade de execução.

O ETL é fundamental para qualquer iniciativa de DW. Porém, deve ser planejado com cuidado para não comprometer os sistemas transacionais (OLTP) das empresas. Um bom ETL deve ter escalabilidade e ser passivo de ser mantido.

Estudos relatam que o ETL e as ferramentas de limpeza de dados consomem um terço do orçamento num projeto de DW, podendo, no que tange ao tempo de desenvolvimento de um projeto de DW, chegar a consumir 80% desse valor. Outros estudos mencionam, ainda, que o processo de ETL consome 55% do tempo total de execução do projeto de DW.

As ferramentas de ETL podem ser utilizadas para fazer todo tipo de trabalho de importação, exportação, transformação de dados para outros ambientes de banco de dados ou para outras necessidades a serem endereçadas. Exemplo: Importação de base de outra empresa.



## Benefícios do ETL

---

**Histórico dos dados:** ao reunir as informações do negócio em um repositório, o ETL fornece um histórico da empresa e auxilia a produção de relatórios e análises sobre dados relevantes para iniciativas corporativas.

**Fácil de usar:** Ferramentas de ETL eliminam a necessidade de programar o código para extrair e processar dados. Isso já é feito pela ferramenta, basta especificar as fontes dos dados e as regras para a sua transformação.

**Funções avançadas para categorizar e limpar dados:** Ferramentas ETL possuem muitas funções para auxiliar na limpeza e categorização de dados. Além disso, o uso dessas ferramentas apresenta vantagens na transferência e transformação de um grande volume de dados com regras complexas, simplificando seu cálculo, mudança e integração.

**Metainformação:** Os metadados, ou seja, informações sobre como identificar, localizar e compreender os dados, são gerados automaticamente pelas aplicações ETL. Isso reduz falhas e auxilia a administração desses dados.

## Principais Conceitos em ETL

---

**Business Intelligence (BI):** Ou inteligência de negócios, refere-se ao processo de coleta, organização, análise, compartilhamento e monitoramento de informações que oferecem suporte a gestão de negócios. É o conjunto de teorias, metodologias, processos, estruturas e tecnologias que transformam uma grande quantidade de dados brutos em informação útil para tomadas de decisões estratégicas.

Descreve a capacidade de a empresa ter acesso e explorar seus dados, desenvolvendo percepção e conhecimento, o que leva à melhora do processo de tomada de decisões. Sua infraestrutura tecnológica é composta de Data Warehouse ou Data Marts, Data Mining e ODS, Benchmarking, além das ferramentas pertinentes.

**Figura 5 – Composição do BI.**



Fonte: <https://www.oficinadanet.com.br>.

**Modelagem Dimensional:** Também chamada de modelagem multidimensional, é a técnica de modelagem de banco de dados para o auxílio às consultas do Data Warehouse nas mais diferentes perspectivas. A visão multidimensional permite o uso mais intuitivo para o processamento analítico pelas ferramentas OLAP (Online Analytical Processing). Toda modelagem dimensional possui dois elementos imprescindíveis: as tabelas Fatos e as tabelas Dimensões. Ambas são obrigatórias e possuem característica complementares dentro de um Data Warehouse.

**Data Warehouse:** É um depósito de dados digitais que serve para armazenar informações detalhadas relativamente a uma empresa, criando e organizando relatórios através de históricos que são depois usados pela empresa para ajudar a tomar decisões importantes com base nos fatos apresentados.

**Data Mart:** É uma subdivisão ou subconjunto de um DW. Os data marts são como pequenas fatias que armazenam subconjuntos de dados, normalmente organizados para um departamento ou um processo de negócio. Normalmente o Data

Mart é direcionado para uma linha de negócios ou equipe, sendo que a sua informação costuma pertencer a um único departamento.

**OLAP:** Online Analytical Processing ou Processo Analítico em Tempo Real é uma das ferramentas mais usadas para a exploração de um Data Warehouse. O OLAP possibilita alterar e analisar grandes quantidades de dados em várias perspectivas diferentes. As funções básicas do OLAP são: visualização multidimensional dos dados, exploração, rotação, vários modos de visualização.

OLAP e o Data Warehouse são destinados a trabalharem juntos, enquanto o DW armazena as informações de forma eficiente, o OLAP deve recuperá-las com a mesma eficiência, porém com muita rapidez. Um cubo OLAP é uma estrutura de dados montada de forma multidimensional, e que proporciona uma rápida análise de valores quantitativos ou medidas relacionadas com determinado assunto, sob diversas perspectivas diferentes.

**Figura 6 – Demo de Cubo com Dimensões e Medidas.**



Fonte: <https://ayltoninacio.com.br/blog/>.

**Staging Area:** É uma localização temporária onde os dados dos sistemas de origem são copiados, facilitando a integração dos dados antes de sua atualização DW. Tem como função agilizar o processo de consolidação, proporcionando um melhor desempenho na fase da atualização dos dados.

A Staging area é o único lugar para determinar os valores que vêm efetivamente dos sistemas legados. A Staging area deve ser usada para limpeza dos dados que entram no processo de extração e transformação.

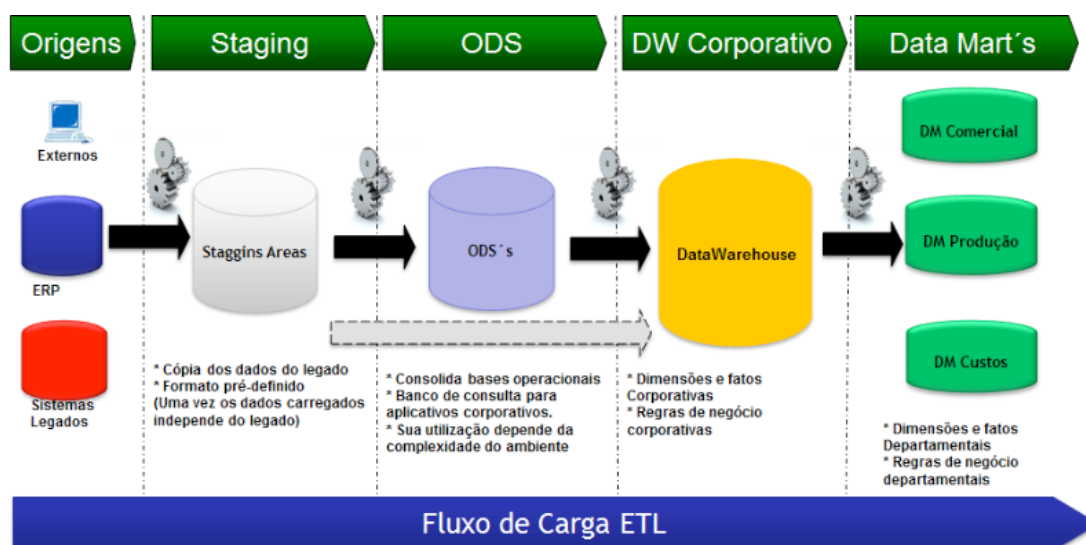
**ODS:** Operational Data Store é um repositório de dados onde são colocados os dados que a empresa trabalha no seu dia a dia, para que sejam consultados por outros sistemas, ou por áreas de inteligência. Um ODS reúne dados de várias aplicações e não é semelhante a um Data Warehouse, pois não tem o compromisso de armazenar histórico de dados e de servir para processos de auditoria sobre esses dados. Entretanto o ODS deve armazenar dados que tem “valor” para seus consumidores e de manter-se atualizado.

Uma área de preparação normal destina-se apenas ao recebimento dos dados operacionais das origens transacionais, a fim de transformar os dados e carregá-los no armazém de dados. Um ODS também oferece essa funcionalidade, mas, além disso, pode ser consultada diretamente.

Dessa forma, as ferramentas de análise que precisam de dados mais próximos do tempo real podem consultar os dados do ODS à medida que são recebidos dos respectivos sistemas de origem, antes de operações demoradas de transformação e carregamento.

**Data Lake:** É um repositório que centraliza e armazena todos os tipos de dados gerados pela e para a empresa. Eles são depositados ali ainda em estado bruto, sem o processamento e análise e até mesmo sem uma governance. A ideia é manter na organização dados que podem ser estrategicamente úteis, mesmo que eles, na realidade, não sejam requeridos em nenhum momento posterior. O data lake seria, em alguns casos já é, o local de armazenamento dessas informações.

**Figura 7 – Processos do ETL.**



Fonte: <https://www.igti.com.br/>.

## Capítulo 2. Ferramentas ETL e Tendências

---

### Quadrante Mágico de Gartner

---

**Gartner:** Gartner Group é uma empresa de consultoria criada por Gideon Gartner, em 1979. Focada na área de tecnologia, seu objetivo é criar conhecimento por meio de pesquisas, consultorias, eventos e levantamento de soluções, para que seus clientes tomem decisões mais embasadas todos os dias.

Esses clientes dividem-se em empresas e também executivos individuais, chegando a um total de 10 mil, possui uma capitalização de mercado de US\$10 bilhões ou mais e emprega uma equipe de mais de 1.900 analistas que recolhem insights de mais de 380.000 interações com clientes a cada ano.

O quadrante mágico de Gartner é o grande produto da empresa, é uma representação gráfica do mercado tecnológico por determinado período. Define forças dentro de um segmento empresarial, fazendo com que fiquem nítidas as qualidades e possíveis falhas das empresas mais significativas da área de tecnologia.

O quadrante possui dois eixos:

- No eixo X (horizontal) temos a abrangência da visão da empresa em relação à tecnologia.
- No eixo Y (vertical) temos a capacidade de executar o que se propõem.

Esses dois eixos definem quatro quadrantes:

- **Challengers:** Desafiadores – Empresas com boa capacidade de execução, mas que não agrega tanto em inovação;
- **Niche Players:** Concorrentes de Nicho – Empresas de nichos de mercado. Não possuem grande expressão no mercado geral como um todo e comumente possuem produtos específicos;
- **Leaders:** Líderes – Possuem boa inovação e entregam o que prometem;

- **Visionaries:** Visionários – Possuem extrema inovação, mas não possuem tanta capacidade para entregar o que prometem.

**Figura 8 – Quadrante Mágico de Gartner.**



**Fonte: Gartner Group.**

## Ferramentas ETL

Algumas das ferramentas mais conhecidas de ETL são:

- IBM InfoSphere DataStage;
- Informática Power Center;
- SAP Business Objects Data Services;
- Microsoft SQL Server Integration Services (SSIS);
- Oracle Data Integrator (ODI);
- Pentaho Data Integration (PDI).

Temos que considerar as particularidades de cada projeto, como orçamento, tamanho, recursos, quantidade de usuários.

## Big Data

---

Razões para usar o Big Data:

- Entender padrões;
- Prever situações;
- Criar fronteiras;
- Informar coleções de dados;
- Estimar parâmetros escondidos;
- Calibrar.

O Big Data traz novos desafios na gestão de dados, como a manutenção de uma linearidade dos dados, sua integridade e qualidade, a fim de que eles possam ser transformados em informação útil.

Soluções de Inteligência Operacional podem correlacionar e analisar dados de fontes variadas em várias latências (desde o batch, até o tempo real), para revelar informações importantes.

## Data Analytics

---

O Data Analytics pode ser usado em vários segmentos de mercado. Os bancos usam essa estratégia para evitar possíveis fraudes. Na educação, você pode medir o progresso dos alunos e avaliar a eficácia do sistema. No varejo, o principal uso é rastrear as características sociais e comportamentais dos clientes, de modo a prever tendências e hábitos.

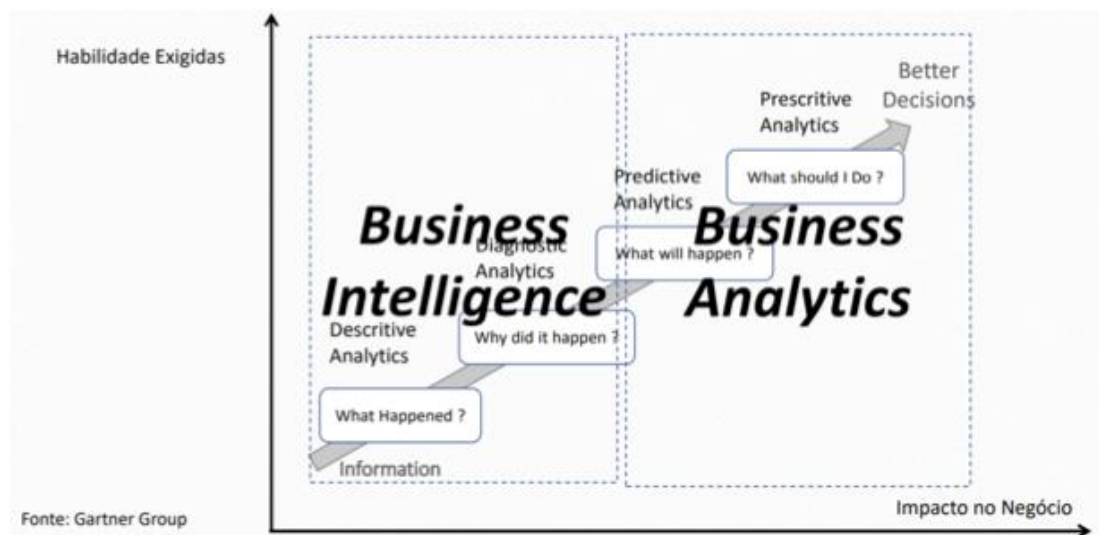


Geralmente, o processo de análise do fluxo de dados segue as seguintes etapas:

- Coleta.
- Ingestão e transformação.
- Armazenamento.
- Análise.
- Desenvolvimento de algoritmos.
- Visualização.

É importante avaliar quais são os principais insights desejados na etapa de visualização, ou até mesmo o levantamento de quais problemas de negócio você gostaria de resolver.

**Figura 9 – Business Intelligence x Business Analytics.**



Fonte: Gartner Group.

## Principais Fornecedores

---

ORACLE, SAP e IBM têm adquirido empresas que possuem ferramentas de BI. Dessa forma, eles estão integrando as ferramentas e conseguindo gerar plataformas completas.

Oracle comprou a Hyperion que já tinha comprado a Brio e como ferramenta ETL tem o ODI. A SAP comprou a Business Object que por sua vez já havia comprado a CrystalReports e que traz como ferramenta de ETL o Data Services. IBM comprou a Ascential que é o fabricante do DataStage e comprou o Cognos que tem o Data Manager.

O PowerCenter Informática é uma plataforma de integração de dados corporativos líder de mercado, dimensionável e de alto desempenho, que promove automação, reutilização e agilidade.

IBM DataStage tem exatamente as mesmas características do PowerCenter. Além disso, tem como grande vantagem a capacidade de fazer processamento paralelo durante as suas cargas.

O IBM Cognos Data Manager fornece recursos *extract, transform and load* (ETL) dimensionais para business intelligence de alto desempenho.

A ferramenta Talend Open Studio é uma solução versátil para integração de dados. Este produto pode melhorar significativamente a eficiência de projetos no trabalho com movimentação de dados. Também disponibiliza um ambiente de desenvolvimento gráfico e fácil de usar.

O Software QlikView fornece as soluções em BI. Da mesma forma como o Power BI, o QlikView trabalha com o conceito de Self-Service, ou seja, qualquer usuário avançado pode criar soluções de análises consistente, não necessitando de uma equipe de TI.

## Pentaho

---

A empresa americana Pentaho, desenvolveu uma suíte de softwares, para o desenvolvimento de uma solução em BI, desde a parte de levantamento de requisitos e DW, até a geração de dashboards para os usuários, essa suíte, chamada de Pentaho BI Suite Community Edition é bastante complexa e inclui ETL, OLAP, metadata, data mining, relatórios e dashboards.

## Visão Geral do Pentaho

---

- Pentaho é um software de código aberto comercial para negócios de Inteligência (BI). Foi desenvolvido desde 2004.
- Pentaho fornece relatórios abrangentes, OLAP análise, painéis, integração de dados, dados mineração e uma plataforma de BI.
- Benefícios:
  - Plataforma com código aberto (open source).
  - Tem uma comunidade de suporte aos usuários.
  - Funciona bem em multiplataforma - Windows, Linux, Macintosh, Solaris, Unix.
  - Tem pacote completo de relatórios, ETL para armazenamento de dados, mineração de dados do servidor OLAP e também painel de controle.

## Pentaho Data Integration

---

O Pentaho Data Integration (PDI) é uma solução de extração, transformação e carregamento (ETL) que utiliza uma abordagem inovadora orientada por metadados.

- Migração de dados entre diferentes bancos de dados e aplicativos.
- Carga de grandes conjuntos de dados em bancos de dados, aproveitando ao máximo os ambientes de processamento em nuvem, em cluster e em paralelo.
- Limpeza de dados com etapas que variam de transformações muito simples a muito complexas.
- Integração de dados, incluindo a capacidade de alavancar ETL em tempo real como fonte de dados para o Pentaho Reporting.
- Prototipagem rápida de esquemas ROLAP.
- Funções do Hadoop: execução e programação de tarefas do Hadoop, design simples do Hadoop MapReduce, integração do Amazon EMR.

### Benefícios do Pentaho

---

- Designer gráfico fácil de usar, com mais de 100 objetos de mapeamento prontos para uso, incluindo entradas, transformações e saídas.
- Arquitetura de plug-in simples para adicionar suas próprias extensões personalizadas.
- Designer integrado (Spoon) que combina ETL com modelagem de metadados e visualização de dados, fornecendo o ambiente perfeito para o desenvolvimento rápido de novas soluções de Business Intelligence.
- A arquitetura do mecanismo de streaming oferece a capacidade de trabalhar com volumes de dados extremamente grandes.
- Desempenho e escalabilidade de classe corporativa com uma ampla variedade de opções de implantação, incluindo servidores ETL dedicados, em cluster e/ou baseados em nuvem.

## Pentaho Data Integration Suite

---

Kettle é um acrônimo para “Kettle E.T.T.L. Environment”.

- Extração, transformação, transporte e carregamento de dados.
- Spoon é uma interface gráfica do usuário que permite projetar transformações e tarefas que podem ser executadas com as ferramentas Kettle - Pan e Kitchen.
- Pan é um mecanismo de transformação de dados que executa diversas funções, como ler, manipular e gravar dados em/e de várias fontes de dados.
- É um programa que executa tarefas projetadas pelo Spoon em XML ou em um repositório de banco de dados.
- As tarefas geralmente são agendadas no modo de lote para serem executadas automaticamente em intervalos regulares.
- Faz parte da suíte do Data Integration, e utiliza as técnicas de ETL (Extract-Transform-Load), para a obtenção dos dados que virão das várias fontes de dados, e que obrigatoriamente teremos de cruzá-las em algum momento dentro do ciclo de ETL.
- Ele é capaz de ler e escrever em vários formatos de SGBD, como Oracle, PostgreSQL, SQLServer, MySql, entre outros, e importar arquivos texto como CSV, planilhas Excel e bases de dados ODBC (apenas em Windows).

## Outras ferramentas ETL (Apache Hadoop, NoSQL)

---

**Apach Hadoop** é um framework para Big Data. O Hadoop é naturalmente adequado para análise de dados. O processo Hadoop de três estágios carrega dados no HDFS, processa os dados através do MapReduce e, em seguida, os resultados são recuperados do HDFS. Por exemplo, o Hadoop pode mesclar dados de duas fontes de dados, como dados de streaming on-line com dados no Data Warehouse,

e usando o MapReduce com uma linguagem de programação de código aberto como R, gerar relatórios analíticos. O processo é inerentemente uma operação em lote, adequada para tarefas de computação analítica ou não interativa.

**NoSQL** é um banco de dados não relacional (exemplo: Cassandra, MongoDB, Redis, HBase, Amazon DynamoDB, Neo4j).

- NoSQL foi projetado para aplicativos em tempo real que geralmente podem interagir com clientes externos à sua organização.
- Ele fornece a capacidade de consultar os dados, para que os usuários possam detalhar os dados à medida que eles mudam.
- Permite o processamento ágil de informações de alto desempenho em grande escala.
- Ele armazena dados não estruturados em vários nós de processamento, bem como em vários servidores.
- Como tal, a infraestrutura de banco de dados distribuída NoSQL tem sido a solução para alguns dos maiores Data Warehouses.

### **Diferença entre Hadoop x NoSQL**

À primeira vista, os bancos de dados NoSQL e o Hadoop parecem ser tecnologias semelhantes, se não competitivas. Ambos gerenciam conjuntos de dados grandes e de rápido crescimento, ambos podem lidar com uma variedade de formatos de dados e podem aproveitar o hardware comum trabalhando juntos como um cluster.

Enquanto o NoSQL é uma infraestrutura de banco de dados distribuída que pode lidar com as grandes demandas de big data, o Hadoop é um sistema de arquivos que permite computação paralela em massa. Usando o MapReduce, o Hadoop distribui um conjunto de dados entre vários servidores e opera com esses dados. Os resultados do processamento do MapReduce são recombinações e armazenados no sistema de arquivos distribuído do Hadoop, HDFS, que disponibiliza dados para outros nós de computação no cluster Hadoop.

## Capítulo 3. Requisitos para ETL

---

### Etapas do Projeto de BI

---

Esse é o passo a passo da implementação de um projeto de BI. São várias atividades necessárias para o levantamento, desenvolvimento e implantação do Business Intelligence. Essas atividades estão intrinsecamente ligadas e são imprescindíveis para o êxito do projeto

#### 1. Levantamento de Necessidades

Essa é a etapa inicial. Nela, a equipe de especialistas irá conversar com os gestores para entender quais suas necessidades e que informações eles gostariam de extrair de seus dados. Selecione as pessoas-chave, de todos os segmentos da sua empresa, que irão utilizar a ferramenta e inclua-os nessa etapa. A primeira pergunta nesse momento é: o que você quer responder através do Business Intelligence? Liste todos os objetivos que deseja alcançar, sem se preocupar se já possui os dados para tanto ou não.

#### 2. Identificação das fontes de dados e requisitos de informação

Após mapear as necessidades, os analistas irão checar se o banco de dados da empresa já possui os dados necessários para responder às perguntas dos gestores. Nessa etapa, a equipe irá conversar com os gestores dos departamentos da empresa envolvidos, buscando entender quais os sistemas computacionais usados, onde estão e qual a qualidade dos dados. Aqui há 2 cenários possíveis:

- A empresa já possui um Data Warehouse. Os dados nesse local já estão tratados, portanto são confiáveis, se correlacionam quando possível e podem ser acessados facilmente. Empresas que possuem um data warehouse já estão com os dados prontos para serem usados e prontas para implementar a solução de BI.
- A empresa ainda não possui um Data Warehouse. Nesse caso, a equipe responsável pela implementação do BI irá auxiliar também na construção do

armazém de dados. Será um pouco mais demorado, porém nada que não possa ser feito.

### 3. Planejamento

Nessa etapa ocorre o detalhamento e a documentação de toda a estrutura do projeto. Feito isso, a equipe irá validar o projeto com os gestores.

### 4. Implementação da Ferramenta

Aqui o software de BI escolhido pela empresa será de fato implementado. Os especialistas irão desenvolver as aplicações necessárias para responder às perguntas levantadas na primeira etapa. Essa etapa pode ser dividida em 3 sub-etapas:

- **Carga de Dados:** ou ETL, onde serão extraídos os dados dos locais definidos na etapa 2, transformá-los segundo certos critérios e carregá-los no banco de dados.
- **Desenvolvimento:** utilizar os dados gerados no ETL e criar as dashboards, levando sempre em consideração as necessidades e regras de negócio definidas na etapa 1.
- **Testes:** essa etapa é realizada simultaneamente com a implementação do BI para evitar grande quantidade de retrabalho. Serão realizados diversos testes para checar se tudo está saindo como o esperado.

### 5. Disponibilidade aos Usuários

Essa etapa consiste na entrega do produto ao usuário final. Essa etapa envolve também o processo de capacitação dos colaboradores que irão utilizar a ferramenta no dia a dia. Como essa capacitação irá ocorrer dependerá da equipe contratada: pode ser por meio da criação de manuais, de treinamentos ou de suporte.



## 6. Retroação

Aqui o projeto de BI já está implementado e funcionando. O processo de retroação é a melhoria contínua das funcionalidades da ferramenta. A equipe estará disponível para fazer ajustes necessários, melhorar aplicações já feitas, e personalizar cada vez os recursos de acordo com o cenário da empresa.

### Desafios da Implantação de um projeto de BI

---

Como qualquer novo recurso, inserir o Business Intelligence para pequenas empresas envolve certos desafios. Porém, estar ciente das dificuldades iniciais ajudará a estar preparado.

#### 1. Falta de profissionais especializados

Para que um sistema de BI seja corretamente instalado e mantido dentro de uma empresa, é necessário que haja uma equipe de TI bem estruturada e dedicada. Porém, nem sempre esse é o caso para pequenas e médias empresas. Mesmo que o provedor do serviço e das ferramentas possa assumir boa parte do trabalho técnico, ainda é necessário um responsável da sua própria organização para cuidar do funcionamento. Em alguns casos, pode valer a pena terceirizar esse serviço, encontrando profissionais especializados rapidamente. Porém, é interessante construir uma equipe mais estruturada dentro do seu negócio ao longo do tempo.

#### 2. Excessos e faltas na aquisição

Ao buscar soluções de Business Intelligence para pequenas empresas, muitas tendem a dar um passo maior do que a perna, ou dar passo nenhum. Os benefícios dessas ferramentas são realmente relevantes, mas eles só serão otimizados quando o serviço está de acordo com as demandas da empresa. Antes de buscar por ferramentas, recursos e outros componentes, tente, primeiro, avaliar a demanda da sua empresa. Quantas ferramentas precisam ser integradas, quais

recursos devem ser providenciados etc. Isso pode poupar muitos gastos desnecessários ao longo do tempo.

### **3. Tempo de implementação prolongado**

As soluções de BI não começam a atuar da noite para o dia. É necessário que uma equipe técnica avalie sua estrutura, veja como integrar melhor as suas ferramentas de gestão, providencie os recursos necessários etc. Isso significa que o resultado pode levar um tempo para aparecer, dependendo do tamanho da sua necessidade.

#### **Projetos de pequeno porte:**

- Projetos de pequeno budget.
- Projetos onde precisamos datamarts limitados, normalmente um.
- Extração de dados limitada.
- Quantidade de usuários limitados.

Para projetos desse porte podemos levar em conta as ferramentas de ETL oferecidas pelos fabricantes dos bancos de dados. Como, por exemplo, SQL Server Integration Services – SSIS. É uma ótima oportunidade para utilizar a suíte de BI oferecida pela Microsoft, tem um custo/benefício profissional muito bom no mercado, além de tudo, vem com o Integration Services (SSIS) que é uma ferramenta de ETL bem completa.

#### **Projetos de médio porte:**

- O projeto não envolve integração de BI com BSC ou mesmo com planejamento estratégico.
- Para esses projetos vale à pena cotar ferramentas com foco em Business Intelligence ou também verificar se seu fabricante de ERP oferece alguma ferramenta de BI.

Caso sua empresa seja um cliente SAP você pode usar o SAP BO DataServices para BI como a ferramenta de ETL ou adotar o SAP BW (suíte completa de BI com Warehouse, OLAP e Extratores). Isso dará menos trabalho para extrair os dados, já que as tabelas são todas normalizadas e já que a empresa possui o ERP, fica mais fácil para negociação de aquisição da ferramenta.

**Projetos de grande porte:**

- Projetos de alto budget.
- Integração do BI com outras soluções.
- Desenvolvimentos mais parrudos.

Para esses projetos vale à pena cotar ferramentas de grande porte. Por exemplo, Cognos – Data Stage/Data Manager, Oracle Data Integrator, PowerCenter Informática, SAS Data Management. A integração é o diferencial da Cognos e da Oracle, e ferramentas como Data Stage e Data Manager têm a capacidade de pôr um processo de carga para rodar paralelamente.

### Componentes críticos para escolha da ferramenta

---

Carregamento incremental: permite atualizar o DW com novos dados sem fazer uma recarga completa de todo o conjunto de dados.

Auditoria e registro: é necessário registro detalhado no pipeline ETL para garantir que os dados possam ser auditados após serem carregados e que os erros possam ser depurados.

Manipulação de vários formatos de origem: para obter dados de diversas fontes, como a API do Salesforce, o aplicativo financeiro de back-end e bancos de dados como MySQL e MongoDB, o processo precisa ser capaz de lidar com uma variedade de formatos de dados.

Tolerância a falhas: em qualquer sistema, ocorrem inevitavelmente problemas. Os sistemas ETL precisam ser capazes de se recuperar normalmente.

Suporte para notificação: se você deseja que sua organização confie nas análises, é necessário criar sistemas de notificação para alertá-lo quando os dados não forem precisos.

Baixa latência: algumas decisões precisam ser tomadas em tempo real, portanto a atualização dos dados é fundamental. Embora existam restrições de latência impostas por integrações de dados de origem específicas, os dados devem fluir pelo seu processo de ETL com a menor latência possível.

## Requisitos para ETL

---

Antes de iniciar um projeto de ETL é necessário que os seguintes itens estejam bem alinhados.

- Requisitos de Negócio.
  - Foco na Decisão;
  - Questões de Informação;
  - Prioridade.
- Viabilidade dos Dados.
  - Performance.
- Latência dos Dados.
  - Histórico.
- Política de Compliance e Segurança.
  - Segurança.

**Foco na Decisão:** este deve ser o objetivo inicial de um sistema de apoio à decisão. As primeiras questões direcionadas aos usuários devem ser elaboradas de maneira a identificar: Quais os objetivos do negócio que necessitam análises? Quais decisões devem ser tomadas a fim de satisfazer estes objetivos? Com estas questões respondidas, já demos um grande passo na formalização dos objetivos que devem nortear o desenvolvimento do projeto.

**Questões relacionadas a informação:** com objetivos e decisões identificadas, precisamos perguntar ao usuário: Que informações você necessita para tomar estas decisões? Quais medidas (indicadores, KPIs, etc.) melhor refletem estas informações? Com informações e medidas na mão, temos grande parte dos fatos e dimensões de negócio identificados.

**Prioridade:** Quão importante para o negócio são as decisões e as informações identificadas nos tópicos anteriores? Identificar as prioridades é essencial, principalmente se no futuro, por qualquer motivo, tivermos que ‘fatiar’ ou reduzir o escopo do projeto.

**Performance:** Quanto tempo o usuário tem para tomar determinada decisão? Com que frequência deve ser tomada esta decisão? Quanto tempo é aceitável entre o momento que um evento acontece no mundo real e o momento que este evento é observado no BI? Essas questões nos ajudam a definir as necessidades de tempo de resposta, frequência de integração e latência.

**Histórico:** Quanto do passado é necessário estar disponível para auxiliar a tomada de decisão? Requisitos de histórico para cada decisão podem explicitar oportunidades de estratégias heterogêneas de histórico.

**Segurança:** Existe restrição de quem deve acessar as informações? Muitas vezes, questionamentos associadas a políticas de segurança são feitos tardiamente, somente no momento de desenvolvimento das aplicações de acesso as informações. Não raro, os requisitos se mostram bem complexos, trazendo surpresa desagradáveis de impacto ao cronograma. Portanto, esta etapa tão importante, executada no início do projeto, deve direcionar todo o desenvolvimento.

## Componentes Críticos

---

- Carregamento incremental.
- Auditoria e registro.
- Manipulação de vários formatos de origem.
- Tolerância a falhas.
- Suporte para notificação.
- Baixa latência.
- Escalabilidade.
- Precisão.

## Definição de Ambiente

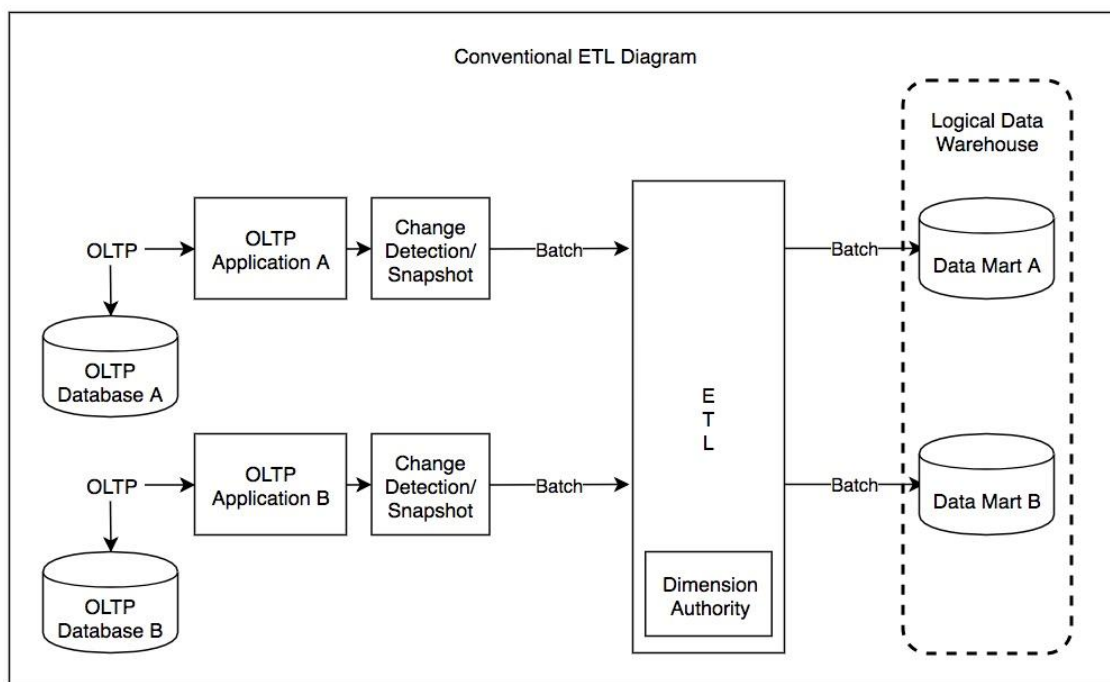
---

Os processos de ETL podem envolver complexidade considerável e problemas operacionais significativos podem ocorrer com sistemas de ETL projetados incorretamente.

O intervalo de valores ou qualidade dos dados em um sistema operacional pode exceder as expectativas dos projetistas no momento em que as regras de validação e transformação são especificadas.

O perfil de dados de uma fonte, durante a análise de dados, pode identificar as condições de dados que devem ser gerenciadas pelas especificações de regras de transformação, levando a uma alteração das regras de validação implementada de forma explícita e implícita no processo ETL.

**Figura 10 – Diagrama ETL.**



Fonte: <https://www.wikiwand.com/>.

## Escalabilidade

Os Data Warehouses são normalmente montados a partir de uma variedade de fontes de dados com diferentes formatos e finalidades. O ETL é um processo essencial para reunir todos os dados em um ambiente padrão e homogêneo.

Escalabilidade: a análise de projeto deve estabelecer a escalabilidade de um sistema ETL durante toda a vida útil de seu uso - incluindo a compreensão dos volumes de dados que devem ser processados nos contratos de nível de serviço. Alguns sistemas ETL precisam ser dimensionados para processar terabytes de dados para atualizar os Data Warehouses com dezenas de terabytes de dados.

O aumento do volume de dados pode alterar o que foi pensado como arquitetura do BI.

## Recuperabilidade ou Rerunnability

---

Os procedimentos de Data Warehousing geralmente subdividem um grande processo de ETL em partes menores, executando sequencialmente ou em paralelo. Para acompanhar os fluxos de dados, faz sentido marcar cada linha de dados com "row\_id" e marcar cada parte do processo com "run\_id". Em caso de falha, essas IDs ajudam a reverter e executar novamente a peça com falha.

A melhor prática também exige pontos de verificação, que são estados em que determinadas fases do processo são concluídas. Uma vez em um ponto de verificação, é uma boa ideia gravar tudo em disco, limpar alguns arquivos temporários, registrar o estado etc.

## Chaves

---

Chaves exclusivas desempenham um papel importante em todos os bancos de dados relacionais, pois unem tudo. Uma chave exclusiva é uma coluna que identifica uma determinada entidade, enquanto uma chave estrangeira é uma coluna em outra tabela que se refere a uma chave primária.

As chaves podem compreender várias colunas; nesse caso, são chaves compostas.

Em muitos casos, a chave primária é um número inteiro gerado automaticamente que não tem significado para a entidade de negócios que está sendo representada, mas existe apenas para o objetivo do banco de dados relacional - geralmente chamado de chave substituta.

## Performance

---

Os fornecedores de ETL avaliam seus sistemas de registro em vários TB (terabytes) por hora (ou ~ 1 GB por segundo) usando servidores poderosos com várias CPUs, vários discos rígidos, várias conexões de rede gigabit e muita memória.



A parte mais lenta de um processo ETL geralmente ocorre na fase de carregamento do banco de dados. Os bancos de dados podem ter um desempenho lento porque precisam cuidar da simultaneidade, manutenção da integridade e índices. Mesmo usando operações em massa, o acesso ao banco de dados geralmente é o gargalo no processo ETL. Alguns métodos comuns usados para aumentar o desempenho são:

- Observe as tabelas de partição (e índices): tente manter partições de tamanho semelhante.
- Observe o catálogo.
- Faça toda a validação na camada ETL antes do carregamento.
- Desativar gatilhos (triggers) nas tabelas de banco de dados de destino durante o carregamento.
- Gere IDs na camada ETL (não no banco de dados).
- Dedicar tempo à modelagem.
- Monitorar os comandos SQL durante o ETL.
- Utilizar índices em colunas muito acessadas.
- Observar as boas práticas nos comandos SQL:
  - Evite o SELECT \*.
  - Desvantagem do ORDER BY e DISTINCT. Só utilize se for realmente necessário.
  - Joins em excesso.
  - Valores calculados devem ser gravados?
  - Comando Truncate Table x Delete.

- Conversões com UPPER, TO\_CHAR e etc em cláusulas WHERE.

## Processamento Paralelo

---

Um desenvolvimento recente no software ETL é a implementação do processamento paralelo. Ele permite vários métodos para melhorar o desempenho geral do ETL ao lidar com grandes volumes de dados. Os aplicativos ETL implementam três tipos principais de paralelismo:

- **Dados:** dividindo um único arquivo sequencial em arquivos de dados menores para fornecer acesso paralelo.
- **Pipeline:** permitindo a execução simultânea de vários componentes no mesmo fluxo de dados, por exemplo procurando um valor no registro 1 ao mesmo tempo em que adiciona dois campos no registro 2.
- **Componente:** a execução simultânea de vários processos em diferentes fluxos de dados no mesmo trabalho, por exemplo classificando um arquivo de entrada enquanto remove duplicatas em outro arquivo.

Todos os três tipos de paralelismo geralmente operam combinados em um único trabalho ou tarefa.

## Ferramentas

---

Ao usar uma estrutura estabelecida de ETL, é possível aumentar as chances de terminar com melhor conectividade e escalabilidade. Uma boa ferramenta de ETL deve ser capaz de se comunicar com os diversos bancos de dados relacionais e ler os vários formatos de arquivo usados em uma organização.

As ferramentas ETL, na maioria dos casos, contêm uma GUI que ajuda os usuários a transformar convenientemente os dados, usando um mapeador de dados

visual, em vez de gravar programas grandes para analisar arquivos e modificar tipos de dados.

## Capítulo 4. Modelagem Dimensional

A modelagem dimensional é a técnica de modelagem de banco de dados para o auxílio às consultas do Data Warehouse, nas mais diferentes perspectivas, onde as informações se relacionam de maneira que podem ser representadas metaforicamente como um cubo.

**Figura 11 – Cubo OLAP.**



**Fonte: Nardi (2007).**

Podemos fatiar este cubo e aprofundar em cada dimensão ou eixo para extrair mais detalhes sobre os processos internos que ocorrem na empresa, que em um modelo relacional torna-se muito complicados de serem extraídos e muitas vezes até impossíveis de serem analisadas.

O modelo dimensional permite visualizar dados abstratos de forma simples e relacionar informações de diferentes setores da empresa de forma muito eficaz.

Toda modelagem dimensional possui dois elementos imprescindíveis: as tabelas Fatos e as tabelas Dimensões. Ambas são obrigatórias e possuem característica complementares dentro de um Data Warehouse.

## Modelo Star Schema

---

O Star Schema, idealizado por Ralph Kimball, é o modelo mais utilizado na modelagem dimensional para dar suporte à tomada de decisão e melhorar a performance de sistemas voltados para consulta.

O modelo é composto no centro por uma tabela Fato, rodeada de dimensões, ficando parecido com a forma de uma estrela.

Dessa forma, as tabelas dimensionais devem conter todas as descrições que são necessárias para definir uma classe como Produto, Tempo ou Loja nela mesma, ou seja, as tabelas de dimensões não são normalizadas no modelo estrela, então campos como Categoria, Departamento, Marca contém suas descrições repetidas em cada registro, assim aumentando o tamanho das tabelas de dimensão por repetirem estas descrições de forma textual em todos os registros.

## Modelo Snowflake

---

No modelo Snowflake, defendido por Bill Inmon, as tabelas dimensionais relacionam-se com a tabela de fatos, mas algumas dimensões relacionam-se apenas entre elas. Isto ocorre para fins de normalização das tabelas dimensionais, visando diminuir o espaço ocupado por estas tabelas, então informações como Categoria, Departamento e Marca tornaram-se tabelas de dimensões auxiliares.

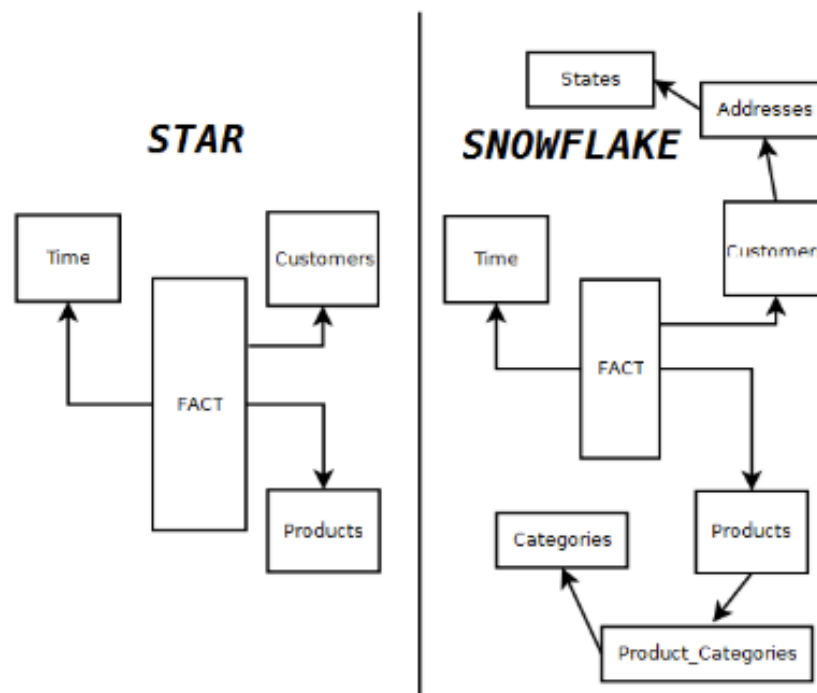
Construindo a base de dados desta forma, passamos a utilizar mais tabelas para representar as mesmas dimensões, mas ocupando um espaço em disco menor do que o modelo estrela.

Este modelo chama-se Snow Flake, pois cada dimensão se divide em várias outras tabelas, onde organizadas de certa forma lembra um floco de neve.

## Modelo StarFlake

É um híbrido entre os modelos Star Schema e Snowflake, aproveitando o melhor de cada um.

**Figura 12 – Star Schema x SnowFlake.**



Fonte: <http://rpblogbi.blogspot.com/2018>.

**Figura 13 – Considerações sobre os modelos dimensionais.**

	Resumo	
	Star Schema	Snowflake
Clareza	mais fácil	mais difícil
Número de tabelas	Menor <	Maior >
Complexidade de consultas	Consultas mais simples	Consultas mais complexas
Desempenho	Menor numero de chaves estrangeiras consequentemente mais rápido a consulta	Maior número de chaves estrangeiras, consequentemente mais lento
Forma Normal	2FN Desnormalizada	A 3FN normalização das dimensões também tem efeito negativo
Joins	Poucos	Muitos
Manutenção	Possui redundância	Sem redundância, pois está na 3FN
Tipo DW	Pequenos	Grandes
Tabelas Dimensão	Somente uma para cada dimensão	Mais que uma para cada dimensão
Normalização	Tanto as dimensões quanto as fatos DESNORMALIZADAS	Tabelas dimensão normalizadas, porém a fato desnormalizada
Modelo	Top Down	Bottom-Up
Quando usar	Quando as tabelas dimensões forem menores	Quando as tabelas dimensões forem maiores
Idealizador	Ralph Kimball	Bill Inmon

Fonte: <http://rpblogbi.blogspot.com/2018>.

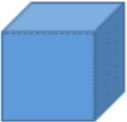
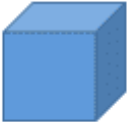
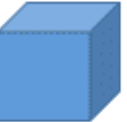
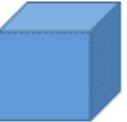
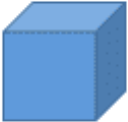




## OLAP



A visão multidimensional permite o uso mais intuitivo para o processamento analítico pelas ferramentas OLAP (Online Analytical Processing).

O OLAP possui um conjunto de técnicas para o tratamento dos dados contidos na visão multidimensional do Data Warehouse. As ferramentas OLAP podem suportar os diferentes tipos de arquitetura: MOLAP, ROLAP ou HOLAP.

O MOLAP é um tipo de arquitetura que utiliza estrutura de banco de dados multidimensional. O ROLAP utiliza a arquitetura relacional dos dados, onde o banco de dados possui a estrutura tradicional. Já o HOLAP (Híbrido) é a junção das duas anteriores, utilizando os melhores aspectos e recursos de cada um dos dois tipos.

**Figura 14 – Considerações sobre os modelos dimensionais.**

	MOLAP	HOLAP	ROLAP
Cube Structure			
Preprocessed Aggregates			
Detail-Level Values			

 Multidimensional Storage
  Relational Storage

Fonte: <https://fard-solutions.com/2016/>.



**MOLAP**

- Mais rápido (quase sempre).
- Mais armazenamento total em disco.
- Instantâneo.

**ROLAP**

- Mais lento (quase sempre).
- Suporta tempo real (com algumas ressalvas).

**HOLAP**

- Tão rápido quanto MOLAP para agregações.
- Tão lento quanto ROLAP para dados no nível da “folha”.

### Regras essenciais para modelagem dimensional

---

As 10 regras essenciais para a modelagem de dados dimensional, segundo Kimball:

Regra #1: Carregue dados detalhados para as estruturas dimensionais.

Regra #2: Estruture os modelos dimensionais em torno dos processos de negócios.

Regra #3: Tenha certeza de que cada tabela fato tenha uma dimensão de data associada.

Regra #4: Certifique-se que todos os dados em uma única tabela Fato estão na mesma granularidade ou nível de detalhe.

Regra #5: Resolva relacionamentos muitos-para-muitos em tabelas Fato.

Regra #6: Resolva os relacionamentos muitos-para-um nas tabelas de dimensões.

Regra #7: Gravar nomes de relatórios e valores de domínios de filtros em tabelas dimensão.

Regra #8: Tenha certeza de que as tabelas dimensão usam uma chave artificial (SK).

Regra #9: Crie dimensões padronizadas para integrar os dados na empresa.

Regra #10: Avalie requisitos e realidade continuamente para desenvolver uma solução de DW/BI que seja aceita pelos usuários de negócios e suporte seu processo de tomada de decisões.

## Tabela Dimensão

---

As dimensões identificam um indicador de análise sobre um empreendimento, negócio ou ação feita. Através das dimensões é possível identificar quando (mês, ano), onde (estado, região) e com quem (segurado, produto) ocorreu um indicador de análise (prêmio emitido).

A tabela dimensão tem como finalidade armazenar informações como tempo, geografia, produto, cliente. É comum uma tabela dimensão possuir várias colunas de informação com o objetivo de representar sua hierarquia. Sua interação com as tabelas fato é feita através da relação 1:N. Possuem uma chave primária para garantir a unicidade de seus registros e está presente na tabela fato, consequentemente como parte de sua chave primária.

As dimensões armazenam 3 coisas:

- A Surrogate Key.
- A Natural Key.
- Os atributos.

Além de tornar a tabela de Fato mais reduzida, mover informações sobre dimensão para uma tabela separada, tem uma vantagem adicional: você pode adicionar novas informações sobre cada membro da dimensão.

## Hierarquia de Dimensões

---

É o conjunto de atributos que possuem uma ordem lógica do maior ao menor nível, é uma forma hierárquica de organizar os dados nas dimensões, por exemplo:

- Ano, mês e dia.
- Categoria, subcategoria e produto.
- Capitão, sargento, cabo e soldado.

As dimensões podem relacionar-se através de **hierarquias**. É comum existir apenas uma hierarquia por dimensão, mas podem existir múltiplas, se necessário. A hierarquia existe apenas nas dimensões, porque a métrica é só um valor, e quem vai dizer se esse valor está correspondendo a um determinado nível da hierarquia é o atributo.

O grão, ou detalhe, é o menor nível da hierarquia da dimensão. É a informação base, o menor detalhe da informação. É muito comum o cliente querer esse tipo de hierarquia para poder ir descendo ou subindo o nível da informação, é o que chamamos de drill down e drill up. Assim podemos ter uma visualização por ano e fazer um drill down, tendo a visualização por mês e depois por dia (que seria o grão).

## Surrogate Key

---

Em um banco de dados, as chaves são usadas para identificar as linhas de uma tabela e fazer as conexões entre elas. No Data Warehouse, temos a Surrogate Key nas dimensões, que é a chave artificial utilizada para conectar a tabela na Fato. A Surrogate Key nada mais é que a Primary Key da dimensão.

A Surrogate Key é uma chave artificial e auto incremental. A palavra artificial vem do tipo, porque ela não existe em lugar nenhum, não está lá no transacional como a Natural Key (PK que vem do legado), ela é criada no Data Warehouse. E é auto incremental porque toda vez que é chamada, troca de número, então ela começa

com 1 e vai indo para 2, 3, 4 e assim por diante. Ela é gerada automaticamente na hora da carga, quando você carrega a dimensão no ETL.

Na tabela Fato, essa Surrogate Key vai ser uma Foreign Key, a chave que serve para relacionar os dados entre duas tabelas, sempre apontando para uma Primary Key em outra tabela, que no caso da dimensão, vai ser a Surrogate Key. Assim, a tabela Fato receberá apenas o código da Surrogate Key da linha que ela está referenciando e não os atributos.

## Tipos de Dimensão

---

Existem 5 tipos fundamentais de dimensões:

- Degenerate Dimension.
- Junk Dimension.
- Role-playing Dimension.
- Conformed Dimension.
- Slowly Changing Dimension.

## Degenerate Dimension

---

Degenerate Dimension é uma chave de dimensão na tabela Fato que não possui sua própria tabela de Dimensão, ou seja, é a Dimensão que não mereceu ser uma Dimensão e foi inserida como coluna na tabela Fato, pois todos os atributos interessantes foram colocados em Dimensões analíticas. Elas são usadas frequentemente quando a granularidade de uma tabela Fato representa o menor grão do relacional. Cardinalidade 1 para 1, por exemplo:

- Número de ordem de compra.

- Número de fatura.
- Número de autorização.

A decisão de usar uma Degenerate Dimension é muitas vezes baseada no desejo de fornecer uma referência direta a um sistema transacional, sem a sobrecarga da manutenção de uma tabela Dimensão separada. A inclusão desses campos nas tabelas Fato é feita para reduzir a duplicação e simplificar procedimentos. Poderíamos simplesmente incluir esses campos em uma tabela de Dimensão, mas neste caso teríamos uma linha desta dimensão para cada linha na tabela Fato, dessa forma teríamos duplicação de informação e complexidade.

### Junk Dimension

---

A Junk Dimension é simplesmente uma estrutura que fornece um local para armazenar os atributos ou uma coleção de códigos transacionais aleatórios que não estão relacionados a nenhuma dimensão específica.

Esses tipos de atributos não se integram facilmente às dimensões convencionais, como Cliente, Fornecedor e Produto, no entanto, alguns dos atributos diversos contêm dados que têm um valor comercial significativo, dessa forma são armazenados em uma Junk Dimension.

O primeiro conceito da Junk Dimension está associado a dimensões de baixa cardinalidade (domínios de valor como sexo, estado civil).

O segundo conceito de Junk Dimension está associado a dimensões degeneradas, que são representadas por textos, como observações, avaliações por extenso.

## Role-Playing Dimension

---

Uma única dimensão pode ser referenciada várias vezes em uma tabela Fato, com cada referência vinculada a uma função logicamente distinta para a dimensão. Exemplo: uma tabela Fato pode ter várias datas, cada uma delas representada por uma chave estrangeira para a dimensão da data. É essencial que cada chave estrangeira se refira a uma visão separada da dimensão da data, para que as referências sejam independentes.

## Conformed Dimension

---

As tabelas de Dimensões estão em conformidade quando os atributos em tabelas de Dimensões separadas têm os mesmos nomes de coluna. As informações de tabelas Fato separadas podem ser combinadas em um único relatório, usando atributos de Dimensão conformes que estão associados a cada tabela Fato.

No Data Warehouse, uma dimensão conformada é uma dimensão que tem o mesmo significado para todas as tabelas Fatos da estrutura. As dimensões conformes permitem que as tabelas Fatos e suas medidas sejam acessados da mesma maneira em várias tabelas Fatos, garantindo relatórios consistentes em toda a empresa.

Em termos simples, uma Dimensão conformada é qualquer Dimensão compartilhada em várias tabelas Fato ou áreas de assunto, por exemplo: as dimensões de Data e Localização são chamadas de Conformed Dimension, pois são compartilhadas em várias tabelas de fatos com o mesmo significado.

## Slowly Changing Dimension (SCD)

---

Slowly Changing Dimensions (SCD ou Dimensões que Mudam Lentamente) retrata as dimensões que sofrem atualizações em seus campos e os classifica pelo tipo de mudança existente em cada uma delas. Todas as dimensões são SCD, porque elas vão precisar atualizar para se manterem sincronizadas com o transacional, a

única exceção é a dimensão de tempo, que a gente chama de tipo 0, porque depois que os dados foram inseridos, não precisa mais atualizar. Apesar do SCD tipo 2 ser predominante e normalmente utilizado, podem haver situações onde outros tipos possuam melhores aplicabilidades. Cabe analisar cada um e verificar a melhor estratégia para o versionamento dos dados, mantendo, assim, a base histórica do DW com alto grau de precisão e confiabilidade.

Vários tipos de SCD podem ser identificados no DW, variando de acordo com as características de atualizações das Dimensões. As alternativas mais comuns de SCD são:

- SCD Tipo 1.
- SCD Tipo 2.
- SCD Tipo 3.
- SCD Híbrido.

**SCD Tipo 1:** É a alteração que não armazena histórico na Dimensão, ou seja, não é feito o versionamento do registro modificado. Trata-se do tipo mais simples, pois não há nenhum controle específico para a atualização dos dados, havendo apenas a sobreposição.

**SCD Tipo 2:** É a técnica mais utilizada para atualizações de dimensões. Nesse tipo de SCD é adicionado um novo registro com as mudanças, preservando sempre os dados anteriores. Dessa forma, os registros da tabela fato vão apontar para a versão correspondente nas dimensões de acordo com a data de referência.

**SCD Tipo 3:** Permite manter as modificações no mesmo registro. Essa técnica funciona com a adição de uma nova coluna na tabela de Dimensão, onde é armazenada a atualização, mantendo na antiga coluna o valor anterior.

**SCD Híbrido:** Conhecido também como SCD Tipo 6, combina todas os SCD anteriores. Isso o torna bastante flexível para as atualizações das dimensões, porém com um grande custo de complexidade. Na solução híbrida é combinado os SCD de

acordo com a estratégia e conveniência, sendo mais completo que os demais SCD. Dessa forma é flexibilizado as atualizações, de maneira que melhor se adeque às modificações dos dados nas dimensões.

## Tabela Fato

---

A tabela Fato é a principal tabela do Data Warehouse, ela vai se conectar nas dimensões, podem existir uma ou mais tabelas fato. Elas armazenam principalmente:

- Métricas – que são os fatos propriamente ditos (tudo que a empresa for mensurar é uma métrica).
- Foreign key – chave estrangeira que serve para relacionar os dados das Dimensões com a Fato.

As tabelas Fatos são classificadas pelas suas granularidades, mas independente de sua granularidade, cada métrica em uma tabela Fato deve estar exatamente no mesmo nível de detalhe. Existem 6 tipos de fatos:

- Fato transacional.
- Fato agregada.
- Fato consolidada.
- Fato snapshot periódico.
- Fato de snapshot acumulado.
- Fato sem Fato.



## Fato Transacional

---

As tabelas fatos transacionais são as mais comuns, geralmente utilizam métricas aditivas, aquelas que somam por todas as Dimensões. Há 2 formas de armazenar os dados em uma tabela Fato transacional.

- Em uma forma de transação por linha.
- Em uma forma de linha por transação – é a mais utilizada.

## Fato Agregada

---

A tabela fato agregada serve para juntar uma grande quantidade de dados quando não precisar analisar no nível do grão, com a função de acelerar o desempenho das consultas. Assim, cria uma fato agregada com os dados que precisa e poderia permanecer com uma fato normal e outra agregada. Exemplo: Uma tabela Fato visitantes e Dimensões de data e hora de um determinado site. A Fato vai monitorar sempre que alguém acessa o site, mas não os visitantes por minuto, porém posso precisar ver de forma consolidada.

## Fato Consolidada

---

A tabela fato consolidada é bem parecida com a agregada, mas serve para combinar 2 tipos de processos (área de negócio, área de assunto, processo de negócio). A Fato consolidada serve para consolidar duas tabelas Fato. No processamento do ETL, na hora de carregar a tabela Fato, carrega uma, carrega a outra, e mistura as duas. O grão precisa ser o mesmo entre as duas. Exemplo: tem uma Fato Venda e depois precisa juntar ela com uma Fato Venda Orçada. Depois disso, cria uma Fato Consolidada e coloca nela o valor real e o orçado, que antes estavam em fatos diferentes.

## Fato SnapShot Periódico

---

As tabelas snapshots periódico registram dados que é instantâneo em um período de tempo predefinido, podendo ser diariamente, semanalmente ou mensalmente. Como o nome indica, tira-se uma "imagem do momento" em que o fato ocorreu, os dados de origem da tabela snapshot periódico são dados de uma tabela de fato transacional em que se escolhe um período a ser capturado.

## Fato SnapShot Acumulados

---

As tabelas de snapshots acumulado descreve a atividade de um processo de negócios que possui início e fim. Esse tipo de tabela de fato possui várias colunas de data para representar marcos no processo. A medida em que as etapas do processo forem sendo concluídas, o registro correspondente na tabela de fato é atualizado.

## Fato sem Fato

---

As tabelas de fato sem fato são encontradas na modelagem de data warehouse, esta tabela não possui nenhuma medida, ela contém apenas chaves estrangeiras para tabelas dimensionais, sendo suficiente para responder a questões relevantes, ou seja, seria uma tabela Fato sem métricas. Ela também é chamada de Fato de Associação ou de Intersecção, mas o termo técnico é Fato sem Fato ou Factless Fact Table.

Serve para fazer uma intersecção de Dimensões. Às vezes a gente quer comparar ou cruzar algo somente entre duas ou mais Dimensões e não tem uma métrica para fazer essas comparações. Essa tabela Fato é exceção, só é usada quando se precisa fazer uma intersecção entre as Dimensões.

## Métricas

---

Métricas são medidas brutas que servem de subsídios aos indicadores. São compostas por vários tipos, como valor, quantidade, peso, volume ou outro formato quantitativo. Tudo que a empresa quer mensurar é métrica, geralmente sendo o que o usuário quer medir, por exemplo:

- Número de vendas.
- Seguidores em determinada rede social.
- Quantidade de propostas implantadas.
- Quantidade de beneficiários.

## Tipos de Métricas

---

Por mais que todas as métricas sejam numéricas, elas têm significados e interpretações diferentes, e é para identificar isso e saber que tipo de cálculos e cruzamentos pode fazer com cada uma, que servem os tipos de métricas. Os 4 tipos de métricas são:

- Aditivas.
- Derivadas.
- Semi-aditivas.
- Não-aditivas.

## Métrica Aditiva

---

São as métricas que permitem operações matemáticas como soma e subtração por todas as dimensões. Dentro da Fato há diversas linhas, e as métricas aditivas devem poder somar todas elas. Alguns exemplos de métricas aditivas:

- Quantidade de vendas.
- Valor da venda (se não for calculado).
- Quantidade de colaboradores.
- Quantidade de demissões.
- Quantidade de admissões.

### Métrica Derivada

---

São métricas que já estavam na Fato e que são calculadas, criando uma nova métrica, que chamamos de derivada. Por exemplo, uma métrica derivada pode ser a multiplicação da quantidade da venda com o preço unitário da Dimensão de produto. Ela pode ser armazenada diretamente na Fato ou calculada em tempo de execução nos cubos na ferramenta OLAP.

### Métrica Semi-Aditiva

---

Métrica semi-aditiva pode ser somada por todas as dimensões menos de tempo, exceto se colocar um filtro que diga que ele só pegue o último registro. Saldo de estoque e saldo bancário, quando representado de forma monetária, são métricas semi-aditivas bem comuns, porque são aditivas em todas as dimensões, exceto na tempo.

### Métrica Não-Aditiva

---

São métricas tipo percentual, algum cálculo feito em tempo de execução, que não podem ser somadas por nenhuma Dimensão.

## Capítulo 5. Metadados

---

Os usuários da equipe ou do data warehouse podem usar metadados em várias situações para criar, manter e gerenciar o sistema. A definição básica de metadados no data warehouse é "são dados sobre dados".

Os metadados podem conter todos os tipos de informações sobre dados DW, como:

- Origem de quaisquer dados extraídos.
- Uso desses dados DW.
- Qualquer tipo de dados e seus valores.
- Recursos de dados.
- Lógica de transformação para dados extraídos.
- Tabelas DW e seus atributos.
- Objetos DW.

Os metadados atuam como um sumário dos dados no sistema DW, que mostra a técnica com mais detalhes sobre esses dados. Pode pensar em um índice em qualquer livro que atue como metadado, para o conteúdo desse livro.

Da mesma forma, os metadados funcionam como um índice para o conteúdo DW. Todos esses metadados são armazenados em um repositório. Ao analisar os metadados, os usuários finais sabem de onde podem começar a analisar o sistema DW. Além disso, é difícil para os usuários finais saber por onde começar a análise de dados em um sistema DW tão grande.

## Papel dos metadados no data warehouse

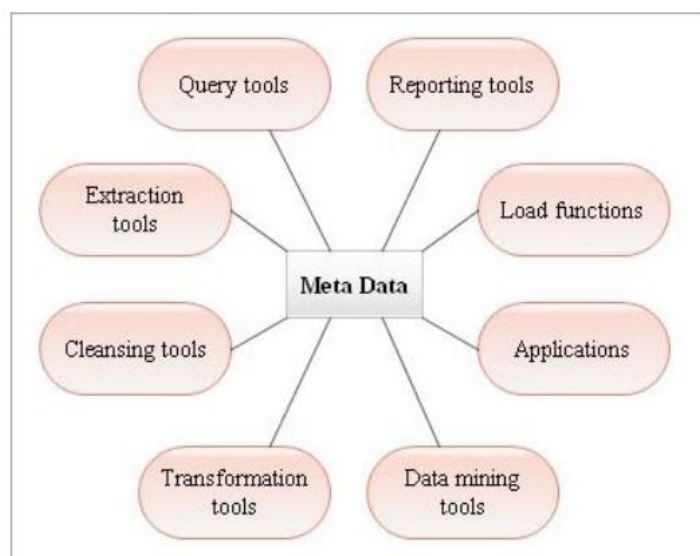
No passado, os metadados eram criados e mantidos como documentos. Porém, no mundo digital de hoje, várias ferramentas facilitaram esse trabalho, gravando metadados em cada nível do processo DW.

Os metadados criados por uma ferramenta podem ser padronizados (ou seja, os dados podem ser trazidos para um formato exclusivo) e podem ser reutilizados nas outras ferramentas em qualquer lugar do sistema DW. Os sistemas operacionais mantêm dados atuais, os sistemas DW mantêm dados históricos e atuais.

Os metadados devem acompanhar todas as alterações que ocorrem nos sistemas de origem, nos métodos de extração/transformação de dados e no conteúdo da estrutura ou dos dados que surgirão nesse processo. Os metadados manterão várias versões para acompanhar todas essas alterações ao longo de vários anos.

Os metadados suficientes fornecidos no repositório, ajudarão qualquer usuário a analisar o sistema de maneira mais eficiente e independente. Ao entender os metadados, você pode executar qualquer tipo de consulta nos dados DW para obter os melhores resultados.

**Figura 15 – Papel do Metadados.**



Fonte: <https://www.softwaretestinghelp.com/>.

### Exemplos de metadados:

- Os metadados de uma página da web podem conter o idioma em que estão codificados, as ferramentas usadas para construí-lo, os navegadores de suporte etc.
- Os metadados para uma imagem digital podem conter o tamanho da imagem, resolução, intensidade da cor, data de criação da imagem etc.
- Os metadados de um documento podem conter a data de criação do documento, data da última modificação, tamanho, autor, descrição etc.

### Comparação entre dados e Metadados

	Dados	Metadados
1	Dados são um conjunto de informações.	Metadados são informações sobre dados.
2	Os dados podem ou não podem ser processados.	Os metadados são sempre dados processados.

### Tipos de metadados

A classificação dos metadados em vários tipos nos ajudará a entendê-los melhor. Essa classificação pode ser baseada em seu uso ou nos usuários, conforme os tipos abaixo:

- 1) Backroom Metadados: direciona os DBAs ou os usuários finais para os processos de extração, limpeza e carregamento.
- 2) Front room Metadados: instrui os usuários finais a trabalhar com ferramentas e relatórios de BI.
- 3) Metadados do processo: armazena os metadados do processo ETL, como o número de linhas carregadas, rejeitadas, processadas e o tempo necessário para carregar em um sistema DW etc. Essas informações também podem ser acessíveis aos usuários finais.

Ao mesmo tempo, as estatísticas das tabelas temporárias também são importantes para a equipe ETL. Esses metadados armazenam os dados do processo das tabelas de preparação, como o número de linhas carregadas, rejeitadas, processadas e o tempo necessário para carregar em cada tabela de preparação.

- 4) Linhagem de dados: armazena a transformação lógica de cada elemento do sistema de origem no elemento de destino DW.
- 5) Definições de negócios: O contexto das tabelas DW foi derivado das definições de negócios. Cada atributo em uma tabela está associado a uma definição de negócios. Portanto, eles devem ser armazenados como metadados ou em qualquer outro documento para referência futura. Os usuários finais e a equipe ETL dependem dessas definições de negócios.
- 6) Definições técnicas: As definições técnicas são usadas exclusivamente na área de armazenamento temporário de dados mais do que nas definições de negócios. O principal objetivo é reduzir a ambiguidade ao criar tabelas intermediárias (Staging tables) e reutilizar quaisquer tabelas existentes. As definições técnicas armazenam os detalhes de cada tabela intermediária, como localização e estrutura. Cada tabela intermediária (Staging) é tecnicamente documentada aqui, se não documentada, significa que a tabela intermediária não existe. Isso evita a recriação da mesma tabela de preparação.
- 7) Metadados comerciais: os dados serão armazenados em termos comerciais, para benefício dos usuários finais/analistas/gerentes/quaisquer usuários. Os metadados comerciais são proxy para os dados do sistema de origem, ou seja, nenhuma manipulação de dados será feita nele. Pode ser derivado de quaisquer documentos e regras de negócios.
- 8) Metadados técnicos: armazenará dados técnicos, como atributos de tabelas, seus tipos de dados, tamanho, atributos de chave primária, atributos de chave estrangeira e quaisquer índices. Isso é mais estruturado quando comparado aos metadados de negócios. Os metadados técnicos



destinam-se principalmente à equipe de DW, como desenvolvedores/testadores/analistas/DBAs, para criar ou manter o sistema. Isso também é usado significativamente pelos administradores para monitorar as cargas de banco de dados e backups de dados etc.

9) Metadados operacionais: como sabemos, os dados no sistema DW são provenientes de muitos sistemas operacionais com diversos tipos e campos de dados. As extrações DW transformam esses dados em um tipo exclusivo e carregam todos esses dados no sistema. Ao mesmo tempo, ele deve poder vincular novamente os dados aos dados do sistema de origem. Os metadados que armazenam todas essas informações de fontes de dados operacionais são conhecidos como metadados operacionais.

10) Informações do sistema de origem: podem ser coletados os seguintes metadados de vários sistemas de origem:

- Banco de dados ou sistema de arquivos: Isso armazenará os nomes dos bancos de dados ou dos sistemas de origem.
- Especificações da tabela: Isso armazenará todos os detalhes sobre as tabelas, como nome da tabela, finalidade, tamanho, atributos, chaves primárias e chaves estrangeiras.
- Regras de tratamento de exceções: Isso armazenará diferentes métodos de recuperação do sistema em caso de falhas no sistema.
- Definições de negócios: Isso armazenará as definições de negócios para uma breve compreensão dos dados.
- Regras de negócios: Isso armazenará um conjunto de regras para cada tabela para entender seus dados e evitar inconsistências.

11) ETL Job Metadados: são muito importantes, pois armazenam os detalhes de todos os jobs a serem processados no planejamento, para carregar o sistema ETL. Esses metadados armazenam as seguintes informações:

- Nome do Job;
- Objetivo do Job;
- Tabelas/arquivos de origem: fornece os nomes e o local de todas as tabelas e arquivos dos quais os dados estão sendo originados por este Job ETL. Pode ter mais de uma tabela ou nome de arquivo.
- Tabelas/arquivos de destino: fornece os nomes e o local de todas as tabelas e arquivos nos quais os dados estão sendo transformados por esse Job ETL. Pode ter mais de uma tabela ou nome de arquivo.
- Dados rejeitados: fornece os nomes e o local de todas as tabelas e arquivos dos quais os dados de origem pretendidos não foram carregados no destino.
- Pré-processamentos: fornece os nomes dos scripts dos Jobs ou dos quais o job atual depende. Isso significa que eles precisam ser executados com sucesso antes de executar o Job atual.
- Pós-processamentos: fornece os nomes dos scripts dos Jobs que devem ser executados imediatamente após o Job atual para concluir o processo.
- Frequência: fornece informações sobre a frequência com que o Job deve ser executado, ou seja, diariamente, semanalmente ou mensalmente.

12) Metadados da transformação: armazenam todas as informações de construção relacionadas ao processo ETL. Toda manipulação de dados no processo ETL é conhecida como transformação de dados.

Qualquer conjunto de funções, procedimentos armazenados, cursores, variáveis e loops, no processo ETL, podem ser considerados como transformações. Mas essas transformações não podem ser documentadas separadamente como metadados.

Todo o processo ETL é construído com transformações de dados. Poucas transformações no ETL podem ser predefinidas e usadas no sistema DW. Os desenvolvedores de ETL gastam seu tempo construindo ou reprocessando todas as transformações de dados. Reutilizar as transformações predefinidas durante o desenvolvimento do processo ETL acelerará o trabalho.

Veja as transformações de dados abaixo para encontrar no ETL:

- Extrações de dados de origem: isso envolve transformações de dados para ler dados do sistema de origem, como uma consulta SQL Select ou FTP ou dados XML/mainframe.
- Geradores de chaves substitutas: Surrogate Keys, o novo número de sequência que deve ser gerado para cada linha da tabela de banco de dados é armazenado como metadados.
- Lookups: podem ser formados com todas as instruções IN, inner joins, e outer joins. Eles são usados principalmente para reter as chaves substitutas de todas as respectivas tabelas de dimensão ao carregar um fato.
- Filtros: são recomendados para classificar os dados que devem ser extraídos, carregados e rejeitados no processo ETL. Filtrar os dados nos estágios iniciais do sistema ETL é uma boa prática. Os filtros são aplicados dependendo das regras de negócios ou restrições.
- Agregados: Dependendo do nível de granularidade dos dados, os metadados relacionados às funções agregadas podem ser usados como soma, contagem, média etc.
- Estratégias de atualização: essas são as regras aplicadas a um registro durante a atualização dos dados. Se houver alguma modificação nos dados existentes, isso indicará se um registro deve ser adicionado, excluído ou atualizado.

- Carregador de destino: armazenará os detalhes do banco de dados, nomes de tabelas e nomes de colunas nos quais os dados devem ser carregados através do processo ETL. Além disso, também armazenará os detalhes do utilitário de carregamento em massa, se houver, que é executado durante o carregamento de dados no sistema ETL.

Toda transformação pode ser nomeada de maneira distinta com uma breve nota sobre sua finalidade.

Alguns exemplos de convenções de nomenclatura estão listados aqui para a lista acima de transformações.

```
SRC_<name of the table>
SEQ_<surrogate key column name>
LKP_<Name of the table referred>
FIL_<Purpose>
AGG_<Purpose>
UPD_<Update type>_<Name of table>
TRG_<Name of table>
```

## Repositório de Metadados em ETL

---

Um repositório de metadados é um local em que qualquer tipo de metadado é armazenado em um banco de dados local ou em um banco de dados virtual. Cada tipo de metadado, como metadados comerciais ou metadados técnicos, pode ser separado logicamente em um repositório.

Além desses dois tipos de metadados, o repositório também possui mais um componente chamado navegador de informações.

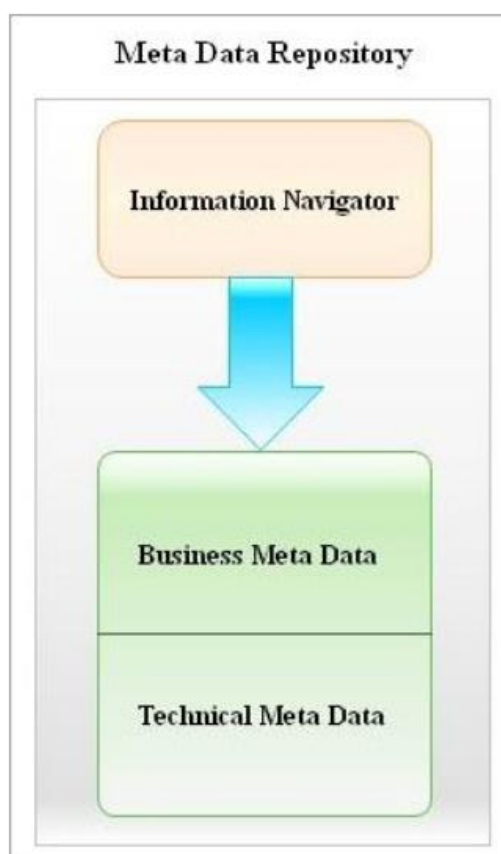
O navegador de informações pode ser usado para executar as tarefas abaixo:

- Interface da ferramenta de consulta (query tool): fornece uma interface para as ferramentas de consulta para acessar os metadados do DW.
- Drill Down para obter detalhes: permite que o usuário faça busca detalhada dos metadados para obter informações mais detalhadas. Por exemplo, no primeiro nível, o usuário pode obter uma definição de tabela de dados. Ao

detalhar, ele pode obter os atributos da tabela no próximo nível. Ao detalhar mais os dados, ele pode obter os detalhes de cada atributo.

- Revisar consultas e relatórios predefinidos: permite ao usuário revisar consultas e relatórios predefinidos. Isso funciona como uma referência para estruturar consultas por conta própria com parâmetros adequados.

**Figura 16 – Repositório do Metadados.**



Fonte: <https://www.softwaretestinghelp.com/>.

### Como os metadados do data warehousing podem ser gerenciados?

Pessoas, processos e ferramentas são as principais fontes para gerenciar metadados.

- As pessoas devem entender os metadados para o uso apropriado.
- O processo incorporará metadados no repositório de ferramentas ou com o progresso do ciclo de vida do DW para uso futuro.
- Posteriormente, os metadados podem ser gerenciados por ferramentas.

## Desafios para o gerenciamento de metadados

---

Depois que os metadados são criados, podemos enfrentar os desafios abaixo ao integrar e gerenciar os metadados no sistema.

- Trazer vários formatos de metadados para um formato padrão pode exigir mais esforço se várias ferramentas estiverem sendo usadas no sistema DW, pois os metadados podem ser armazenados em planilhas, aplicativos ou bancos de dados.
- Os formatos de metadados não possuem padrões estabelecidos em todo o setor. Com essa falta de processo padronizado, é difícil passar metadados por vários níveis do sistema e ferramentas DW.
- A manutenção consistente de várias versões de metadados históricos é uma tarefa complexa.

## O que é ETL orientado a metadados?

---

O ETL orientado por metadados estabelece uma camada para simplificar o processo de carregamento de dados em um sistema DW. Você pode decidir se processa os dados no sistema ou não, dependendo dos metadados. Portanto, você pode chamá-lo como ETL controlado por metadados.

## Capítulo 6. Staging Area

---

### ODS – Operational Data Store

---

Operational Data Store (armazenamento de dados operacional) é um repositório de dados, onde são colocados os dados que a empresa trabalha no seu dia a dia, para que sejam consultados por outros sistemas. Geralmente os dados são armazenados em um ODS através de ferramentas de softwares ETL, que extraem dados de banco de dados de diversas origens e coloca, de forma integrada, no ODS. Além disso, há estratégias que utilizam o ODS como base origem para os Data Warehouse.

Um ODS difere de uma base de dados de uma aplicação, pois reúne dados de várias aplicações e não é semelhante a um Data Warehouse, pois não tem o compromisso de armazenar histórico de dados e de servir para processos de auditoria sobre esses dados. Entretanto o ODS deve armazenar dados que tem “valor” para seus consumidores e de manter-se atualizado.

ODS é uma base de dados integrada, volátil, de valores correntes e que contém somente dados detalhados. Também pode ser entendido como uma visão integrada do mundo operacional. Normalmente, sua construção adota bases de dados relacionais. Algumas características do ODS:

- Possibilitar a integração de dados de várias aplicações.
- Ter alto desempenho na hora de armazenar seus dados e, principalmente, na hora de consultas sobre esses dados.
- Ter dados de negócio atualizados e ao mesmo tempo servir para processos decisórios.

Um dos benefícios do ODS é poder otimizar a criação do DW e possibilitar a realização de consultas relacionais sobre dados históricos.

## Diferenças entre ODS e DW

---

As principais diferenças entre ODS e DW são que ODS é voltado para consultas com granularidades maiores, enquanto um DW normalmente é utilizado para consultas complexas e em dados agregados.

Outra diferença é que o ODS é para relatórios operacionais que exigem informações próximas ou em tempo real, enquanto um DW é destinado a análise de tendência histórica e, geralmente, trabalha com grande volume de dados. Além do que, ODS contém apenas uma pequena janela de dados, enquanto um DW contém todo o histórico de dados.

Um ODS fornece informações para decisões operacionais e táticas em dados em tempo real atuais ou próximo, enquanto um DW fornece um feedback para as decisões estratégicas que levam a melhorias gerais do sistema.

A frequência de carga de dados da ODS poderia ser a cada poucos minutos ou por hora, enquanto que em um DW a frequência das cargas de dados pode ser diária, semanal, mensal ou trimestral.

## Staging area

---

Staging Area é uma localização temporária onde os dados dos sistemas de origem são copiados. Desta forma, ao invés de acessar os dados diretamente da fonte, o processo de “transformação” do ETL pega os dados da Staging Area para tratar e entregar os dados.

Em alguns casos, ao invés da “Staging” ser uma tabela temporária, pode ser uma view materializada que pode ser executada (manualmente ou mediante programação de carga) para ter os dados sempre atualizados.

Como alguns projetos precisam ter várias fontes de dados, a necessidade de termos uma “STAGING” acaba sendo obrigatório para que possa reunir o máximo de



dados possível e, com isso, poder selecionar os dados, transformando-os em informação.

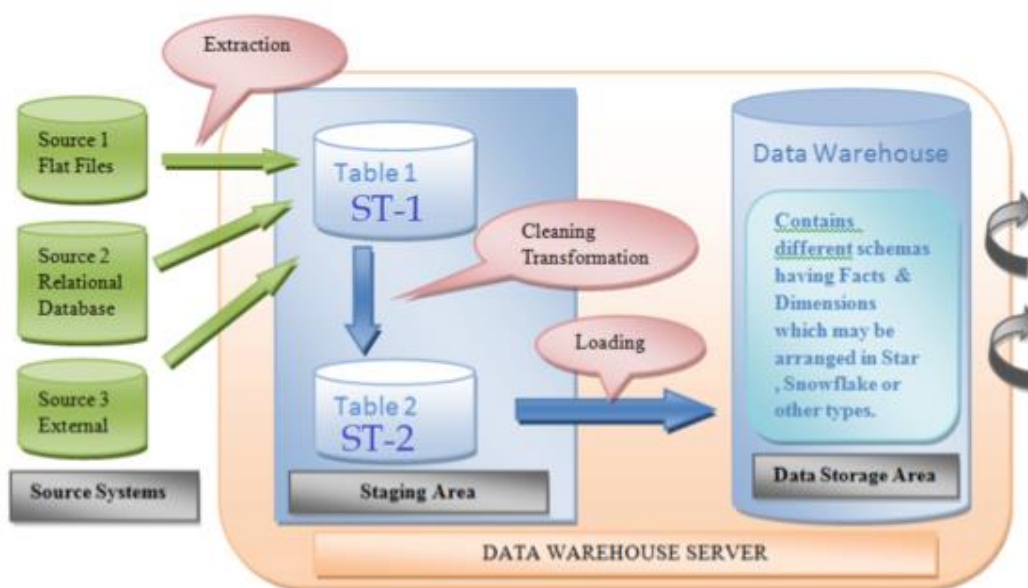
O benefício deste recurso é poder armazenar os dados em sua origem “bruta” para poder trabalhar em cima deles, ao invés de ficar sempre tendo que acessar a Fonte de dados. Assim, conseguimos evitar uma série de problemas, como baixa performance, por exemplo. Além disso, você tem a segurança que os dados estarão a sua disposição até que você execute algum processo que limpe a STAGING. A Staging area é uma área de armazenamento intermediário situada dentro do processo de ETL. Auxilia a transição dos dados das origens para o destino final no DW.

### Tipos de Staging area

A Staging area pode possuir três tipos de stagings:

- Staging 1: Dados sem nenhuma transformação. Cópia exata da origem.
- Staging 2 Aux: Query com transformações. Visão do dia.
- Staging 2: Query com transformações. Visão histórica.

**Figura 17 – Tipos de Staging Area.**



## Staging 1

A Staging 1 serve para executar a extração, busca os dados sem nenhuma transformação, cópia exata da origem. A extração, basicamente, seria buscar as informações dos sistemas legados e fontes externas da empresa, e colocá-las na Staging Área para validação, transformação e carga. Existem várias técnicas para fazer isso. O importante é termos as informações novas ou atualizadas, tendo assim um retrato dia a dia do que foi incluído, excluído e alterado. A partir disso não precisamos mais do banco de dados de produção, ou seja, não corremos o risco de concorrer consumindo assim recursos dos sistemas legados.

## Staging 2

A Staging 2 refere-se à transformação. Com os dados na Staging area podemos fazer as transformações necessárias. Essas transformações vão variar dependendo da modelagem e das fontes de dados. Ela utiliza os dados da Staging 1 e utiliza os métodos da carga Insert/Update.

## Staging 2 Aux

A Staging 2 Aux também se refere à transformação. A diferença que utiliza o método de carga Truncate.

**Figura 18 – Comparativo dos Tipos de Staging Area.**

Staging 1	Staging 2 Aux	Staging 2
Cópia dos dados	Query com as transformações	
Origem: dados transacionais e arquivos	Origem: ST1	
Toda a informação pertinente	Apenas campos que serão utilizados na dimensão	
Não possui chave primária	Possui chave primária	
Metodo Truncate	Metodo Truncate	Metodo Update / Insert
Pouco espaço de armazenamento	Pouco espaço de armazenamento	Maior espaço de armazenamento
Latência pequena	Latência pequena	Latência grande
ST1 Aux tem como objetivo copiar os dados da origem para não sobrecarregá-los nas operações necessárias ao ETL	ST2 Aux tem como objetivo otimizar o processo de carga diário	ST2 Aux tem como objetivo servir de base histórica para possíveis reprocessamentos

## Tratamento de erros

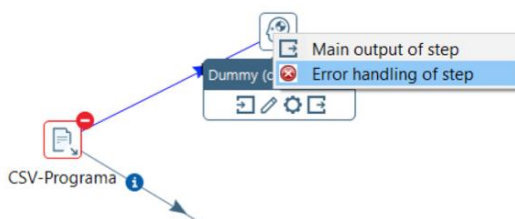
Quando um componente de fluxo de dados aplica uma transformação aos dados da coluna, extrai dados de fontes ou carrega dados nos destinos, podem ocorrer erros. Frequentemente, os erros ocorrem por causa de valores de dados inesperados. Por exemplo, uma conversão de dados falha porque uma coluna contém uma cadeia de caracteres em vez de um número, uma inserção em uma coluna de banco de dados falha porque os dados são uma data e a coluna tem um tipo de dados numéricos, ou uma expressão não é avaliada porque o valor de uma coluna é zero, resultando em uma operação matemática que não é válida.

## Categoria de erros

Em geral, os erros se enquadram nas categorias a seguir:

- **Erros de conversão de dados:** ocorrem se uma conversão resulta em perda de dígitos significativos, perda de dígitos insignificantes e truncamento de cadeias de caracteres. Os erros de conversão de dados também ocorrem se não houver suporte para a conversão solicitada.
- **Erros de avaliação de expressão:** ocorrem se expressões avaliadas em tempo de execução, realizarem operações inválidas ou se tornarem sintaticamente incorretas devido a valores de dados ausentes ou incorretos.
- **Erros de pesquisa:** ocorrem se uma operação de pesquisa não conseguir localizar, por exemplo, uma tabela de pesquisa correspondente.

**Figura 19 – Saídas de Erro.**



**Fonte: Pentaho.**

## Tipos de erros

---

Os erros fazem parte de uma de duas categorias: erros ou truncamentos.

- **Erros:** um erro indica uma falha inequívoca e gera um resultado NULL. Esses erros podem incluir erros de conversão de dados ou erros de avaliação de expressão.
- **Truncamentos:** um truncamento é menos grave que um erro. Um truncamento gera resultados que podem ser utilizáveis ou até mesmo desejáveis. Pode-se optar por tratar truncamentos como erros ou como condições aceitáveis.

## Tratamento de erros

---

Opção	Descrição
Falha no Componente	A tarefa Fluxo de Dados falha quando ocorre um erro ou um truncamento. Falha é a opção padrão para um erro e um truncamento.
Ignorar Falha	O erro ou truncamento é ignorado e a linha de dados é direcionada para a saída da transformação ou fonte.
Redirecionar Linha	O erro ou truncamento da linha de dados é direcionado para a saída de erro da fonte, transformação ou destino.

## Glossário Pentaho

---

**CE - *Community Edition*** - Do inglês Community Edition, traduzido como Edição da Comunidade, referência usada para a versão desenvolvida em parceria com a comunidade open source. Sigla muito usada nas extensões dos arquivos de instalação (ex.: pentaho-server-ce-8.3.0.0-371.zip).

**BI - *Business Intelligence*** - Do termo inglês BI (Business Intelligence), pode ser traduzido como Inteligência de negócios, refere-se ao processo de coleta, organização, análise, compartilhamento e monitoramento de informações que oferecem suporte a gestão de negócios.

**PDI - *Pentaho Data Integration*** - Ferramenta contida na plataforma [Pentaho](#), originalmente chamada de [Kettle](#), responsável por integração de diversas fontes de dados, utilizado no [Pentaho](#) para construção do [Data warehouse](#) de uma aplicação de Business Intelligence. O seu principal executável é o software spoon.

**PRD - *Pentaho Report Designer*** - Ferramenta que permite profissionais de BI criarem relatórios detalhados e de qualidade, além de ser altamente flexível com a manipulação de dados para a apresentação nos relatórios.

**PME - *Pentaho Metadata Editor*** - A ferramenta simplifica a criação de relatórios de metadados, que são dados sobre dados. O PME mapeia a estrutura de dados físicos para um modelo de negócios mais lógicos.

**PSW - *Pentaho Schema Workbench*** - Ferramenta que permite editar e criar modelos multidimensionais (Mondrian). Os profissionais de BI poderão criar modelos Mondrian graficamente ou defini-los codificando manualmente arquivos XML.

**PUC - *Pentaho User Console*** - Interface do servidor [Pentaho](#) destinado a uso pelos usuários do sistema, com disponibilização de relatórios, dashboards e cubos divididos por soluções.

**Kettle** - Ferramenta para integração de dados, faz parte da solução [Pentaho](#), onde é responsável pelo processo de Extração, Transformação e Carga (ETL), renomeado pela [Pentaho](#) para [Pentaho Data Integration](#) ou simplesmente PDI.

**ETL** - Do inglês Extract Transform Load (Extração Transformação Carga), são ferramentas de software cuja função é a extração de dados de diversos sistemas, transformação desses dados conforme regras de negócios e, por fim, a carga dos dados em um Data Mart ou um [Data Warehouse](#). É considerada uma das fases mais críticas do [Data Warehouse](#) e/ou Data Mart;

**PAT - Pentaho Analysis Tool** - Um visualizador de cubos OLAP, seu nome foi renomeado para Saiku.

**Saiku** - É um sistema modular de código aberto, oferecendo suíte de análise OLAP leve que permanece facilmente incorporável, extensível e configurável. O Saiku foi disponibilizado também como um plugin [Pentaho](#) para visualização de cubos Mondrian, anteriormente o Saiku se chamava PAT (Pentaho Analysis Tool).

**JPivot** - JPivot é uma biblioteca de tags JSP personalizadas que apresenta uma tabela OLAP e permitir que os usuários executem navegações OLAP típicas como slice and dice, drill-down e roll-up. Ele usa Mondrian como servidor OLAP. JPivot também suporta XMLA datasource acesso. Porém a mesma não está sofrendo mais atualizações, então recomenda-se migrar para Pivot4J ou Saiku.

**Pivot4J** - Pivot4J é uma biblioteca de tags JSP personalizadas que apresenta uma tabela OLAP, fork do jPivot, e permitir que os usuários executem navegações OLAP típicas como slice and dice, drill-down e roll-up. Ele usa Mondrian como servidor OLAP. Pivot4J também suporta XMLA datasource acesso.

**Spoon** - Ferramenta gráfica que se desenha e valida todo processo do [Pentaho Data Integration](#) (transformações e jobs).

**PAN** - É a ferramenta de linha de comando [Pentaho Data Integration](#) para executar transformações a partir de um repositório [Pentaho Data Integration](#) (banco de dados ou empresa) ou de um arquivo local.

**Kitchen** - É a ferramenta que executa jobs a partir de um repositório [Pentaho Data Integration](#) (banco de dados ou empresa) ou de um arquivo local.

**CTools** - É um conjunto de ferramentas para construção de dashboards para a [Suite Pentaho BI Open Source](#), utilizando tecnologias da web como JavaScript, CSS e HTML.

**CDF** - Community Dashboard Framework - Principal componente das CTools.

**Step** - Representa os passos dentro de uma transformação, onde podem ser agrupadas por Inputs, outputs etc.

**Hop** - Define o fluxo de dados entre dois passos chamados de steps (conexão).

**Data Lake** - Termo criado pelo CTO (Chief Technical Officer) do [Pentaho](#), James Dixon, para descrever um componente importante no universo da análise de dados. O data lake é um repositório que centraliza e armazena todos os tipos de dados gerados pela e para a empresa. Os dados são depositados ainda em estado bruto, sem o processamento, análise e até mesmo uma governança adequada.

## Referências

---

BIX TECNOLOGIA. *Como funciona a implementação de um projeto BI?* 2018. Disponível em: <<https://www.bixtecnologia.com.br/home/index.php/4886/como-implementar-um-projeto-de-bi/>>. Acesso em: 06 ago. 2020.

DANIEL, Eduardo Jose. *Glossário Pentaho Business Intelligence*. Ambiente Livre, 2020. Disponível em: <<https://www.ambientelivre.com.br/tutoriais-pentaho-bi/glossario-pentaho-business-intelligence.html>>. Acesso em: 06 ago. 2020.

DATA INTEGRATION. In: *Wikipédia, a enciclopédia livre*. Flórida: Wikimedia Foudation, 2020. Disponível em: <[https://en.wikipedia.org/wiki/Data\\_integration#cite\\_note-refone-1](https://en.wikipedia.org/wiki/Data_integration#cite_note-refone-1)>. Acesso em: 06 ago. 2020.

DEVMEDIA. Disponível em: <<http://www.devmedia.com.br/>>. Acesso em: 06 ago. 2020.

EASIER. *Ferramenta de ETL*. Disponível em: <<http://gedxml.com.br/asa2/index.php/ferramenta-de-etl?view=featured>>. Acesso em: 06 ago. 2020.

ECKERSON, Wayne; WHITE, Colin. *Evaluating ETL and Data Integration Platforms*. USA, 2003.

ELIAS, Diego. *Entendendo o processo de ETL*. Canaltech, 2014. Disponível em: <<https://canaltech.com.br/business-intelligence/entendendo-o-processo-de-etl-22850/>>. Acesso em: 06 ago. 2020.

ELIAS, Diego. *O que é Business Intelligence? BI na prática*. Disponível em: <<https://www.binap pratica.com.br/o-que-e-bi/>>. Acesso em: 06 ago. 2020.

ERIKA. *Um estudo sobre as ferramentas OLAP*. DevMedia, 2007. Disponível em: <<https://www.devmedia.com.br/um-estudo-sobre-as-ferramentas-olap/6691>>. Acesso em: 06 ago. 2020.



E-SETORIAL. *Levantamento de requisitos para BI: uma questão de seguir o roteiro*. 2012. Disponível em: <<https://www.e-setorial.com.br/blog/88-levantamento-de-requisitos-para-bi-uma-questao-de-seguir-o-roteiro>>. Acesso

HITACHI. *Pentaho*. Disponível em: <<https://community.hitachivantara.com/s/pentaho>>. Acesso em: 06 ago. 2020.

INACIO, Aylton. *Criando e alimentando um cubo OLAP físico no MySQL*. AyltonInacio Blog, 2018. Disponível em: <<https://ayltoninacio.com.br/blog/criando-e-alimentando-um-cubo-olap-fisico-no-mysql>>. Acesso em: 06 ago. 2020.

KIMBALL GROUP. Disponível em: <<http://www.kimballgroup.com/>>. Acesso em: 06 ago. 2020.

KIMBALL, Ralf. *The Data Warehouse Toolkit: The Complete Guide to Dimensional Modeling*. USA, 2002.

MACHADO, Felipe Nery Rodrigues. *Tecnologia e Projeto de Data Warehouse: Uma visão multidimensional*. São Paulo, Brasil, 2008

MUSARDO, Igor. *Modelos de consultas e relatórios AD-HOC para Sistemas de BI*. Musardos, 2008. Disponível em: <<https://musardos.com/modelo-de-consultas-e-relatorios-ad-hoc-para-sistemas-de-bi/>>. Acesso em: 06 ago. 2020.

NOVATO, Douglas. *O que é Business Intelligence?* Oficina da Net, 2017. Disponível em: <<https://www.oficinadanet.com.br/post/13153-o-que-e-business-intelligence>>. Acesso em: 06 ago. 2020.

PITON, Rafael. *Tabela Dimensão: os 5 tipo que você deve conhecer*. Piton, 2017. Disponível em: <<https://rafaelpiton.com.br/blog/data-warehouse-tipos-dimensoes/>>. Acesso em: 06 ago. 2020.

POSITIVO. *O que é data lake*. 2018. Disponível em: <<https://www.meupositivo.com.br/panoramapositivo/o-que-e-data-lake/>>. Acesso em: 06 ago. 2020.

PRIMAK, Fábio Vinicius. *Decisões com B.I.: Business Intelligence*. Rio de Janeiro: Editora Ciência Moderna, 2008.

REDAÇÃO FLEXA. *Como aplicar o business intelligence para pequenas e médias empresas*. Flexa, 2018. Disponível em: <<https://flexa.cloud/como-aplicar-o-business-intelligence-para-pequenas-e-medias-empresas/>>. Acesso em: 06 ago. 2020.

RODRIGO. *Modelagem Dimensional – Snowflake*. RP Blog BI, 2018. Disponível em: <<http://rpblogbi.blogspot.com/2018/10/modelagem-dimencional-snowflake.html>>. Acesso em: 06 ago. 2020.

SANTANA, Eduardo. *Você já ouviu falar em Staging Area? Veja a contribuição dela no ETL*. Bufallos, 2017. Disponível em: <[http://bufallos.com.br/bg\\_br/staging-area/](http://bufallos.com.br/bg_br/staging-area/)>. Acesso em: 06 ago. 2020.

SAS. *ETL: O que é e qual sua importância?* Disponível em: <[https://www.sas.com/pt\\_br/insights/data-management/o-que-e-etl.html](https://www.sas.com/pt_br/insights/data-management/o-que-e-etl.html)>. Acesso em: 06 ago. 2020.

SOFTWARE TESTING HELP. *Metadata In Data Warehouse (ETL) Explained With Examples*. 2020. Disponível em: <<https://www.softwaretestinghelp.com/metadata-in-data-warehouse-etl/>>. Acesso em: 06 ago. 2020.

SOLVIMM. *O que é ETL*. 2018. Disponível em: <<https://solvimm.com/blog/o-que-e-etl/>>. Acesso em: 06 ago. 2020.

VIEIRA, Marcio Junior. Kimball University: As 10 Regras Essenciais para Modelagem de Dados Dimencional. In: *Tutoriais Pentaho Business Intelligence e Analytics*, 2020. Disponível em: <<https://www.ambientelivre.com.br/tutoriais-pentaho-bi/kimball-university-as-10-regras-essenciais-para-a-modelagem-de-dados-dimencional.html>>. Acesso em: 06 ago. 2020.

WIKIWAND. *Extract, transform, load*. Disponível em: <[https://www.wikiwand.com/en/Extract, transform, load](https://www.wikiwand.com/en/Extract,_transform,_load)>. Acesso em: 06 ago. 2020.