MODELAGEM DE DADOS

Análise de requisitos e modelagem de dados de um sistema de e-commerce, sob o ponto de vista informacional.

MBA Executivo em Business Analytics e Big Data

T8 – Modelagem Informacional – Professor: Thiago Araújo

Julho - 2021

****

Rafael H. Rocha de Souza

A56660250

INTRODUÇÃO

Na era do *big data*, nos deparamos com incontáveis fontes e um vasto universo de dados que precisam ser coletados e, posteriormente, armazenados para serem tratados e, só então, aproveitados como informação útil. Em geral, o gerenciamento de projetos de implantação de uma *data warehouse* ou de um *data lake* requer um planejamento cuidadoso e nesse projeto não será diferente.

Inicialmente, esse relatório pretende descrever a arquitetura dos componentes do ponto de vista de dados e logo depois irá apresentar a modelagem conceitual e relacional da base operacional e o diagrama dimensional da base analítica. Serão sugeridos mecanismos eficientes que ajudem a criar e monitorar critérios para selecionar e organizar as informações que interessam.

OBJETIVO E DEFINIÇÃO DO PROBLEMA

O objetivo desse projeto é aplicar técnicas para modelar um sistema informacional, com foco em criar uma estrutura de armazenamento que possibilite o encaixe e o resgate de informações em um determinado padrão. Ele vai permitir que usuários do banco de dados possam acessar, armazenar e operar de forma eficiente. Será levado em consideração, que qualquer erro durante o processo de modelagem poderá comprometer a usabilidade do sistema, tornando real a necessidade de um segundo trabalho de programação e possível reformulação de todo o banco de dados o que tomaria tempo e geraria custos desnecessários.

O alvo será criar e apresentar a modelagem de um sistema de e-commerce e para alcançar precisão e eficiência no resultado, vamos iniciar uma minuciosa análise dos requisitos propostos.

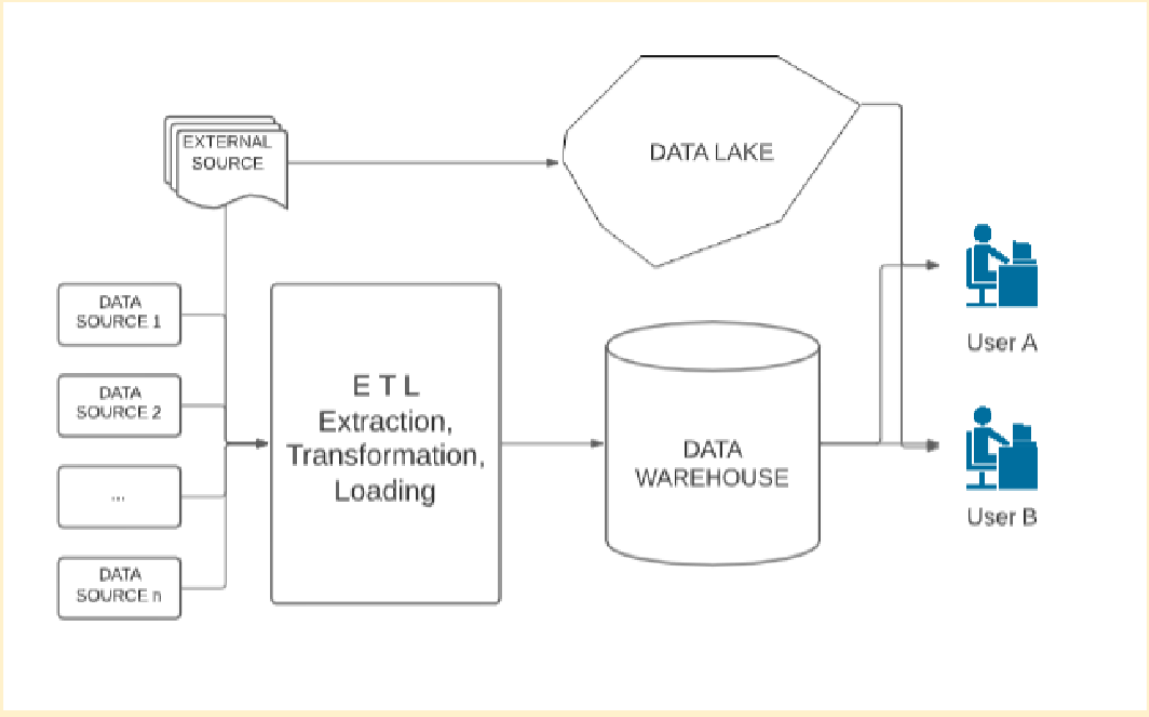
Abaixo a catalogação de todas as regras de negócio, como se darão o surgimento dos dados, como deverão ser armazenados e outras demandas:

- Os usuários desse sistema poderão visualizar produtos ofertados e realizar a compra dos mesmos, e serão identificados por e-mail, nome completo, data de nascimento, endereços (casa, escritório, etc.) e informações do seu cartão de crédito para cobrança;
- Cada usuário terá um carrinho de compras, que exibirá os produtos selecionados para compra e quantidade de cada;

- Os produtos serão cadastrados por um administrador que será um outro tipo de usuário do nosso sistema;
- Cada produto deverá ter um nome, um conjunto de imagens, uma descrição e seu preço;
- Nossos usuários poderão visualizar seu histórico de compras, com os itens e produtos inclusos, e a respectiva quantidade;
- Esse histórico deverá exibir também a data em que os pagamentos ocorreram, a quantidade de itens comprados e o endereço utilizado para entrega;
- Todos os produtos estarão relacionados por critérios de semelhança pré definidos, e o sistema deverá disponibilizar uma funcionalidade para associá-los entre si e sugerir ao usuário um novo produto assim que a sessão terminar;
- Analistas acessarão a quantidade de produtos comprados por cada usuário, o valor faturado em produtos por cada usuário em um mês, o valor faturado por produto em um mês, o número de unidades de um produto vendidas por mês, e o ranking de produtos mais vendidos;
- A aplicação estará integrada por API a uma plataforma contratada de comunicação externa para dar suporte ao time de *customer experience* e ao de vendas. Por meio de chat a plataforma irá interagir com os nossos usuários, e nossos analistas pretendem visualizar o número de chamados abertos por cada usuário e a quantidade de chamados tratados por cada atendente;
- Outro serviço externo contratado, que também deverá estar integrado e disponível para consultas do nosso time de produto, é uma ferramenta de rastreamento de cliques e textos digitados pelos nossos usuários, sua posição geográfica e quanto tempo levou em cada tela. Esse tipo de serviço tende a gerar grandes volumes de dados e os nossos analistas irão inspecioná-los para descobrir a melhor usabilidade da ferramenta.

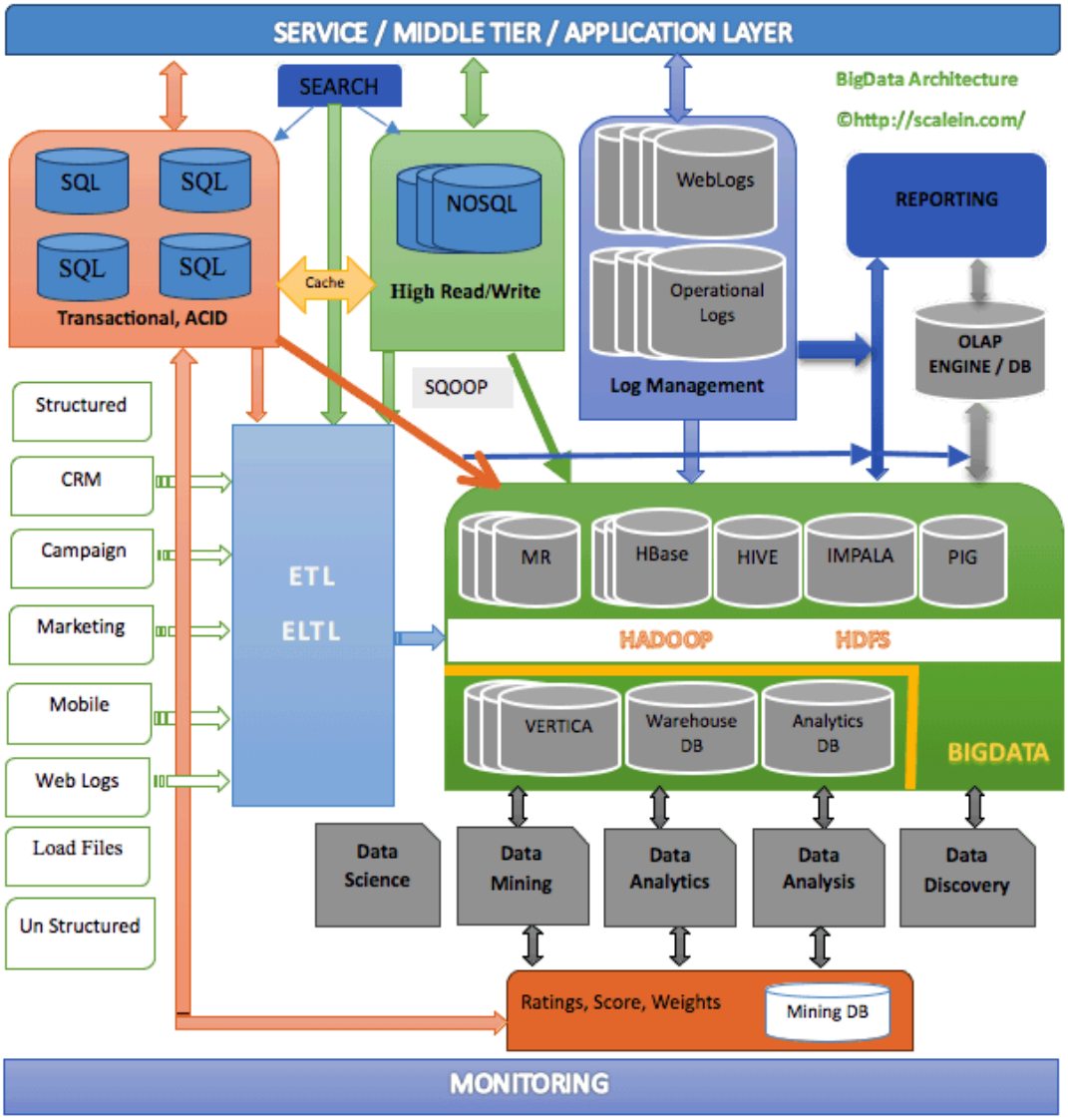
DESCRIÇÃO GERAL DO SISTEMA E SOLUÇÕES DE ARMAZENAMENTO

Antes de partir para a modelagem, vamos analisar a arquitetura dos componentes do ponto de vista de dados, suas características e minúcias, para entendermos melhor as etapas operacionais que serão apresentadas mais a frente neste projeto.



Fonte: venublog.com

A figura acima ilustra conceitualmente a arquitetura geral do nosso sistema. De acordo com os requisitos apresentados, a opção foi utilizar uma arquitetura híbrida combinando fontes internas (dados gerados pelo próprio sistema transacional de e-commerce) e fontes externas (dados gerados por outras fontes como o serviço de chat e a ferramenta de rastreamento de ações de usuários). Adicionalmente, para armazenamento, utilizaremos também de forma híbrida uma solução que combina o DATA WAREHOUSE e o DATA LAKE de forma a atender os requisitos solicitados. Já na ilustração abaixo, podemos observar com mais detalhes, os quatro principais componentes de uma arquitetura de dados, eles serão descritos a seguir.



Fonte: scalein.com

1. **Dados provenientes de fontes heterogêneas** – Na nossa aplicação usaremos dados estruturados (SQL ou NoSQL), ou quaisquer outros dados provenientes de API's integradas, como por exemplo, os dados extraídos do serviço da plataforma contratada de comunicação externa; ou vindo de outros meios, sendo eles semiestruturados ou não estruturados. Vale mencionar mais uma vez, nossos serviços contratados e suas fontes, que após o processo de ETL (*extract, transform and load*) estarão disponíveis para consultas dos usuários finais. Logs, cliques de usuário, visitas e análise de comportamento gerarão um grande volume de dados. Os *Data Lakes* serão utilizados para departamentalizar parte dos nossos dados, separando-os por setores dentro da empresa, como por exemplo para a equipe de vendas e *customer experience* e para o time de produto. Os dados contidos neles, deverão ser gerenciados por especialistas, os quais apresentarão visões multidimensionais para ferramentas de *front end*.

2. **Transformação** – No processo de ETL podemos importar e exportar ferramentas, e através desse processo essencial podemos carregar todas as fontes de dados no *pipeline* de processamento de dados. Nossa ferramenta de gerenciamento de log e cliques, por exemplo, pode gerar muita utilidade analítica.

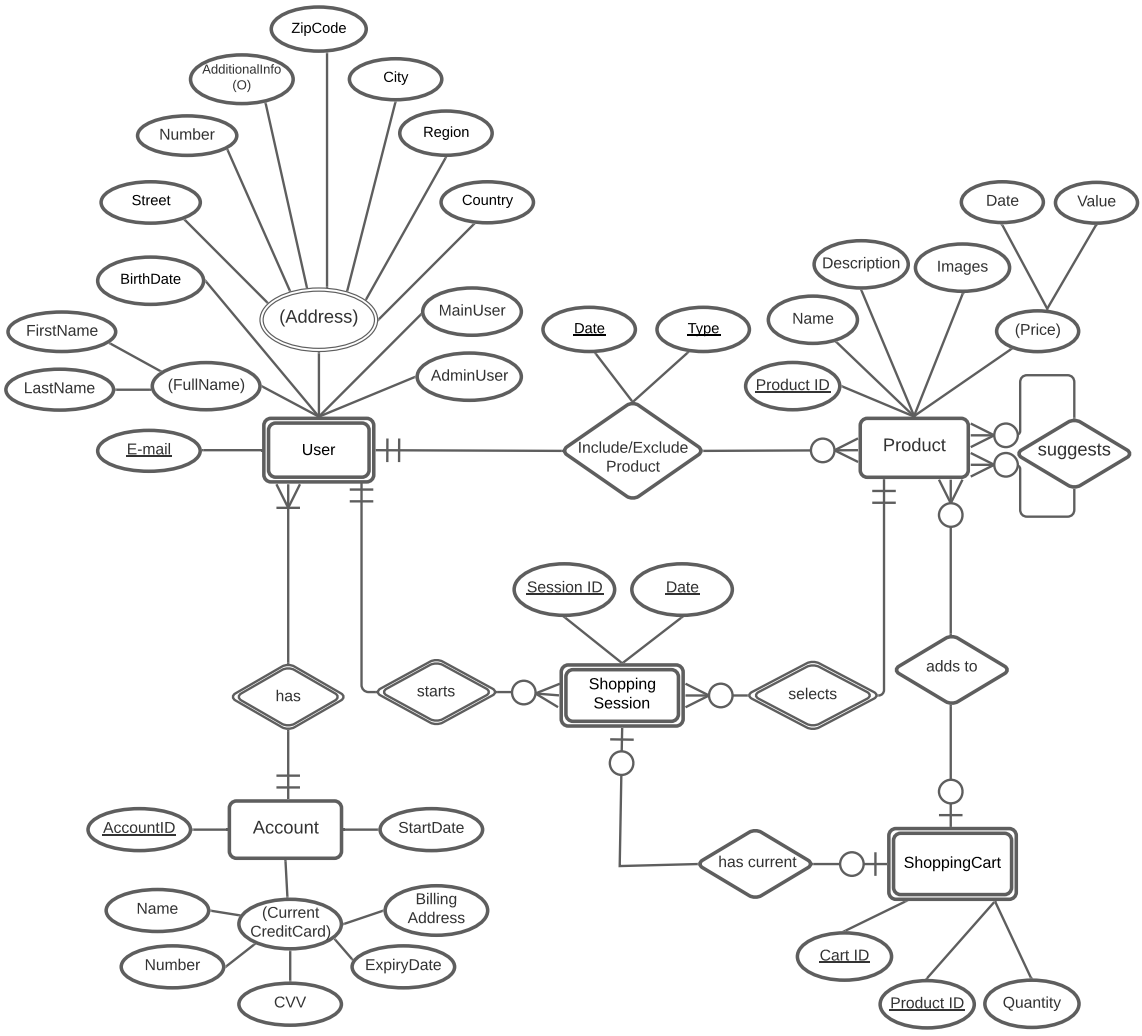
3. Processamento e Integração – Mais uma fonte de dados vastos, que combina dados estruturados e não estruturados em um só lugar, em tempo real ou com carregamento incremental. Do ponto de vista da disposição física dos dados, nosso DW terá uma estrutura centralizada, com o intuito de maximizar o poder de processamento e acelerar os processos de buscas analíticas. Consistência e capacidade de recuperação de dados são críticas e no caso do processamento analítico (OLAP – *on-line analytical processing*) deve-se dar maior importância aos dados históricos, totalizados e consolidados em detrimento dos dados detalhados e individualizados. Com objetivo principal em analisar o negócio, esse tipo de sistema de uso informativo e estrutura variável, estará disponível para a comunidade gerencial com dados condicionados a análises de granularidade detalhada e resumida. O nosso banco de dados operacional terá como característica um tipo de processamento OLTP (*on-line transaction processing),* que possibilitará operações diárias de inclusão, alteração e exclusão, com sua estrutura estática. Nessa etapa o foco estará na geração de dados utilizáveis, materializados ou agregados, que poderão ser consumidos por seus componentes.

4. Consumo dos dados – Etapa onde expomos os dados de forma utilizável para os nossos usuários finais ou para camadas internamente (ad-hoc) ou externamente (usando APIs).

Além dos quatro componentes lógicos, o monitoramento desempenha um papel crucial na detecção de qualquer falha na nossa *pipeline* de dados, juntamente com alterações de limite para identificar quaisquer gargalos em termos de desempenho, escalabilidade, eficiência e rendimento.

MODELAGEM CONCEITUAL DA BASE DE DADOS OPERACIONAL

Essa parte do projeto se caracteriza na criação dos primeiros desenhos das tabelas que representarão de maneira gráfica todo o sistema de armazenagem do nosso banco de dados, levando em consideração todos os requisitos e regras de negócio sugeridas nas etapas anteriores. Foram analisados todos os elementos e fenômenos relevantes em um sistema de e-commerce e a partir daí formado o seguinte modelo abstrato do corpo de conhecimento adquirido: o Modelo Entidade-Relacionamento ou MER.



Essa modelagem conceitual nos possibilita ter uma discussão mais aberta e a troca de ideias com pessoas que possuem conhecimento técnico.

Os conceitos abaixo são utilizados para descrever os papéis de cada componente do nosso sistema de e-commerce:

- **Instâncias**: definem ocorrências ou registros que passam a existir de uma entidade;

- **Entidades**: denominam algo identificável, singular e tenha existência bem delimitada, podendo ser classificada como “fraca” ou “forte”;
- **Atributos**: se referem à cada característica possuída pelas instâncias de uma entidade ou de um relacionamento;
- **Relacionamentos**: são a associação entre as entidades e são representados por linhas no diagrama. Capturam as interações entre as entidades;
- **Cardinalidades**: elas são representadas pelos símbolos que aparecem nas extremidades de uma linha de relacionamento;
- **Normalização**: é um conjunto de regras que visa minimizar as anomalias, redundâncias e inconsistências, visa facilitar a manipulação dos dados e a manutenção do sistema de informações.

Entidades e atributos identificados:

ENTIDADE	ATRIBUTOS	DESCRIÇÃO
Account		Conta do usuário
	AccountID	Número da conta, chave primária
	StartDate	Data de criação da conta
	CurrentCreditCard	Cartão de crédito, atributo composto: • Number: 16 dígitos do cartão; • Name: Nome do titular; • ExpiryDate: Data de validade; • CVV: Código de verificação; • BillingAddress: Endereço de cobrança do cartão.

ENTIDADE	ATRIBUTOS	DESCRIÇÃO
User		Usuário
	Email	E-mail do usuário. Chave primária da entidade
	FullName	Nome completo composto com: • FirstName: Primeiro nome;

		• LastName: Último nome.
	BirthDate	Data de nascimento
	Address	Endereços multivalorados (casa, trabalho, etc.): • Street: Rua; • Number: Número; • AdditionalInfo: Informações adicionais; • Region: Região; • City: Cidade; • Country: País; • ZipCode: Código Zip.
	AdminUser	Atributo com dados booleanos: TRUE: Indica o usuário adm; FALSE: Indica o usuário como não adm. OBS: A função do administrador foi propositalmente colocado dentro da entidade User, ele além de ter a mesma visão do usuário, ele poderá utilizar o relacionamento Include/Exclude.
	MainUser	Atributo com dados booleanos (usuário titular da conta): TRUE: Indica o usuário titular; FALSE: Indicaria um possível usuário dependente.

ENTIDADE	ATRIBUTOS	DESCRIÇÃO
Product		Produto
	ProductID	Identificação do produto, chave primária da entidade.
	Name	Nome do produto
	Description	Descrição do produto

	Images	Imagens do produto
	Price	Preço. Atributo composto: • Value: Valor do preço; • Date: Data do preço. Obs: Este atributo muda no tempo.

ENTIDADE	ATRIBUTOS	DESCRIÇÃO
ShoppingSession		Entidade fraca que representa a relação do usuário com o produto e o carrinho de compras.
	SessionID	Identificação da sessão de compra, chave primária da entidade.
	Date	Data da sessão

ENTIDADE	ATRIBUTOS	DESCRIÇÃO
ShoppingCart		Carrinho de compras, entidade fraca.
	CartID	Identificação do carrinho de compras, chave primária da entidade.
	ProductID	Identificação do produto.
	Quantity	Quantidade.

Relacionamentos identificados:

RELACIONAMENTO	DESCRIÇÃO
Has	Representa o relacionamento do usuário e sua conta.
Starts	Representa o relacionamento do usuário com a *shopping session*, tendo em vista que quando um usuário começa navegar pelos produtos se inicia uma sessão.
Selects	Representa o relacionamento da sessão com o produto.
Adds to	Representa o relacionamento do produto (sendo adicionado) ao carrinho de compras.
Has Current	Representa o relacionamento da *shopping session* com o carrinho de compras.

Include/Exclude Product (Admin only)	Representa o relacionamento do usuário administrador com a lista de produtos disponíveis na plataforma. O usuário “adm” pode incluir/excluir nenhum ou mais produtos e o produto pode ser incluído ou excluído por um administrador. Possui como atributos a data de inclusão e exclusão e o tipo (TRUE: Inclusão, FALSE: Exclusão)

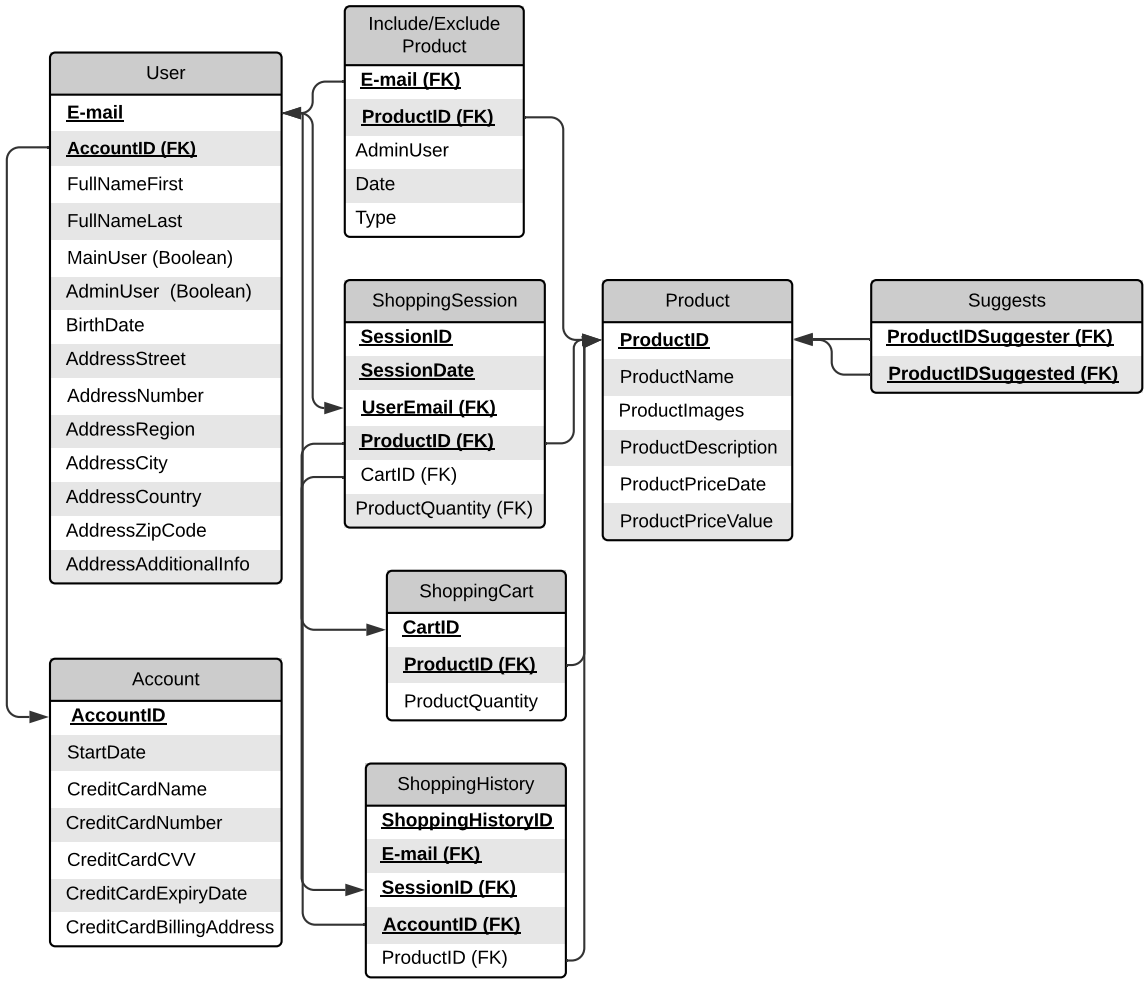
A modelagem dos dados partirá para a segunda etapa, onde apresentaremos a modelagem relacional.

MODELAGEM RELACIONAL DA BASE DE DADOS OPERACIONAL

O modelo relacional representa o banco de dados como uma coleção de relações. Uma relação nada mais é do que uma tabela de valores. Cada linha da tabela de valores proposta na nossa modelagem representará uma coleção de valores de dados relacionados, e cada linha denotará uma entidade ou um relacionamento. No entanto, o armazenamento físico dos dados será independente da maneira como os dados serão organizados logicamente. As restrições de integridade relacional se referem as condições que devem estar presentes para uma relação válida.

A restrições no sistema de gerenciamento do nosso banco de dados foram principalmente divididos em três categorias principais: restrições de domínio, de chave e de integridade referencial.

A modelagem relacional da base de dados operacional do nosso sistema de e-commerce será apresentada no diagrama abaixo:



Após analisar e refletir com cautela os requisitos propostos, os tópicos abaixo se tornaram relevantes e nortearam a modelagem:

ENTIDADE	ATRIBUTOS	DESCRIÇÃO
User (Usuário)		Entidade fraca da entidade Account (Conta)
	E-mail	E-mail do usuário, chave primária do tipo STRING.
	AccountID (FK)	Código da conta, chave estrangeira com a entidade User.
	BirthDate	Data de nascimento. Tipo: DATE.
	Address	Endereço (atributo composto multivalorado), podendo ser da casa, trabalho e etc: • Street: Rua (STRING) • Number: Número (INT)

		• City: Cidade (STRING) • Region: Região (STRING) • Country: País (STRING) • AdditionalInfo: Complemento do endereço, atributo opcional (STRING) • ZipCode: Código ZIP (INT)
	AdminUser	Atributo BOOLEANO que identifica o usuário como adm (TRUE) ou não adm (FALSE).
	MainUser	Atributo BOOLEANO que identifica o usuário como titular da conta (TRUE) ou dependente (FALSE).
	FullName	Nome completo (atributo composto), com: • FullNameFirst: Primeiro nome (STRING) • FullNameLast: Último nome (STRING)

ENTIDADE	ATRIBUTOS	DESCRIÇÃO
Account (Conta)		
	AccountID	Número da conta, chave primária (INT).
	StartDate	Data de criação da conta (DATE).
	CurrentCreditCard	Dados do cartão de crédito corrente (atributo composto): • Number: 16 dígitos do cartão (INT); • Name: Nome do titular (STRING); • ExpiryDate: Data de validade (DATE); • CVV: Código de verificação (INT); • BillingAddress: Endereço de cobrança do cartão (STRING).

ENTIDADE	ATRIBUTOS	DESCRIÇÃO
Product (Produto)		
	ProductID	ID do produto, chave primária da entidade (INT).
	Name	Nome do produto (STRING)
	Description	Descrição do produto (STRING)

	Images	Imagens do produto (Blobs do tipo: jpg, png, jpeg e bitmap).
	Price	Preço. Atributo composto: • Value: Valor do preço (STRING); • Date: Data do preço (DATE). Obs: Este atributo muda no tempo.

ENTIDADE	ATRIBUTOS	DESCRIÇÃO
ShoppingCart		Carrinho de compras, entidade fraca.
	CartID	Identificação do carrinho de compras, chave primária da entidade (STRING) .
	ProductID (FK)	Identificação do produto, chave estrangeira (INT).
	Quantity (FK)	Quantidade, chave estrangeira (INT).

ENTIDADE	ATRIBUTOS	DESCRIÇÃO
ShoppingSession		Entidade fraca que representa a relação do usuário com o produto e o carrinho de compras.
	SessionID	Identificação da sessão de compra, chave primária da entidade (STRING).
	SessionDate	Data da sessão (DATE).
	UserEmail (FK)	Email do usuário, chave estrangeira (STRING).
	ProductID (FK)	Identificação do produto, chave estrangeira (INT).
	CartID (FK)	Identificação do carrinho de compras, chave estrangeira (INT).
	Quantity (FK)	Quantidade, chave estrangeira (INT).

ENTIDADE	ATRIBUTOS	DESCRIÇÃO
ShoppingHistory		Representa o histórico da relação do usuário com: produto, quantidade e carrinho de compras.
	ShoppingHistoryID	Identificação do histórico de compras, chave primária da entidade (STRING).
	UserEmail (FK)	Email do usuário, chave estrangeira (STRING).

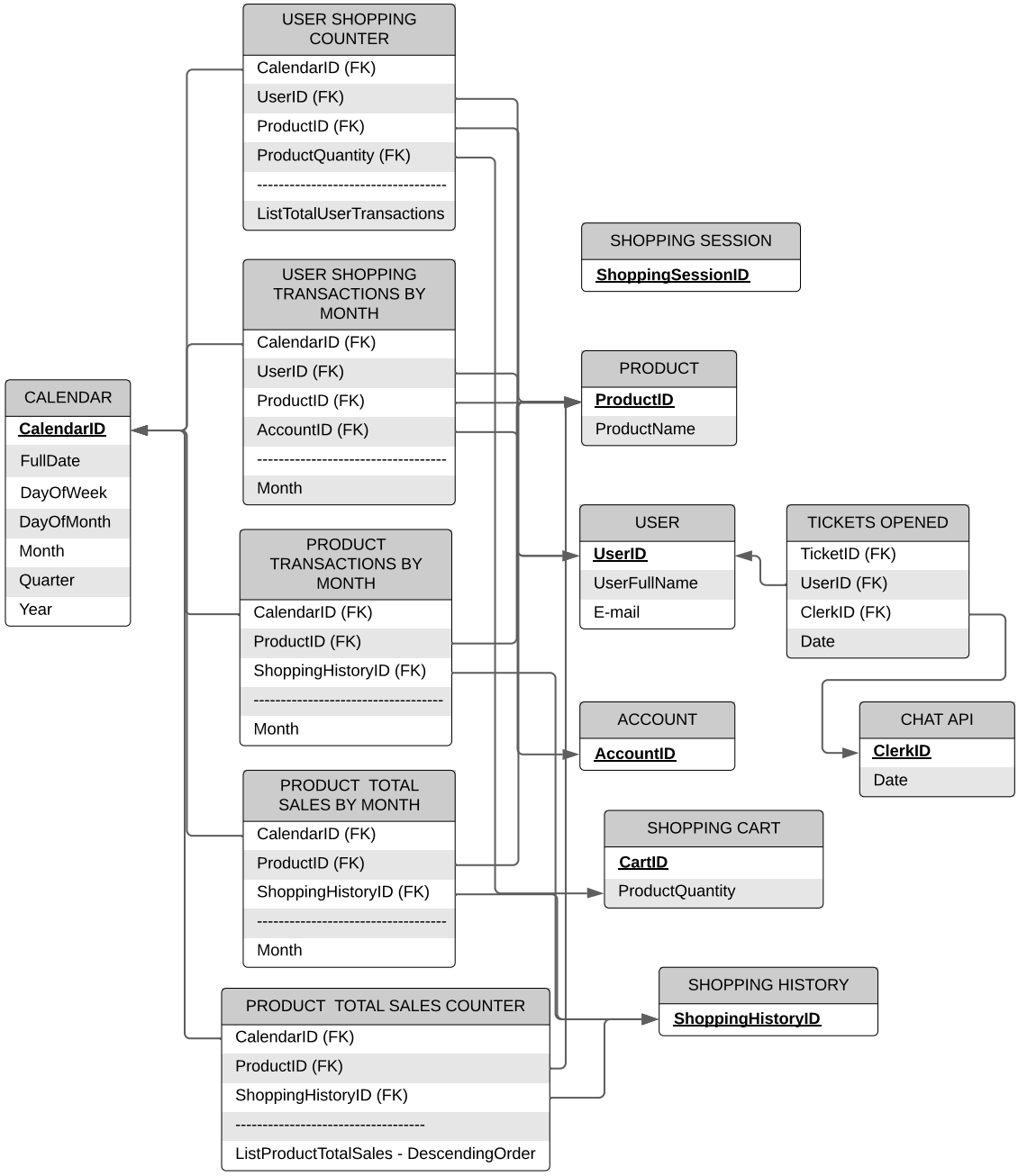
	ProductID (FK)	Identificação do produto, chave estrangeira (INT).
	AccountID (FK)	Identificação da conta do usuário, chave estrangeira (INT).
	Quantity (FK)	Quantidade, chave estrangeira (INT).

RELACIONAMENTOS
Todo usuário está associado a uma conta.
Os produtos serão cadastrados por um administrador, que será outro tipo de usuário do sistema.
Os produtos serão adicionados ao carrinho de compras de acordo com a preferência do usuário.
Include/Exclude Product (Admin only) - Ilustra o relacionamento do usuário administrador com a lista de produtos disponíveis na plataforma. O usuário “adm” pode incluir/excluir nenhum ou mais produtos e o produto pode ser incluído ou excluído por um administrador. Possui como atributos a data de inclusão e exclusão e o tipo (TRUE: Inclusão, FALSE: Exclusão)
O histórico de compras tem relacionamento com usuário, conta, produto e sessão.

Na próxima etapa vamos partir para modelagem dimensional que é uma técnica de estrutura de dados otimizada para armazenamento de dados em uma *data warehouse*.

MODELAGEM DIMENSIONAL DA BASE DE DADOS ANALÍTICA

O objetivo da modelagem dimensional é otimizar o banco para uma recuperação mais rápida dos dados. Desenvolvido por Ralph Kimball, esse conceito consiste em tabelas de “fatos” e “dimensões”. Em contraste, os modelos de relação são otimizados para adição, atualização e exclusão de dados no sistema. Na figura abaixo nossa proposta de modelagem dimensional:



Foram levadas em consideração diversas regras e princípios da modelagem dimensional, como por exemplo: era necessário garantir que a nossa tabela fato tivesse uma tabela de dimensão de data associada, foi preciso ter certeza que todos os fatos estivessem com a mesma granulação, foram armazenados rótulos de relatório para facilitar a filtragem de valores de domínio nas tabelas de dimensões e garantir que elas tivessem uma chave substituta.

TABELAS (FATO E DIMENSÃO)
Dimensão: User, Account, Product, Calendar, ShoppingCart, ShoppingSession.
Fato: UserShoppingCounter, UserShoppingTransactionsByMonth, ProductTransactionsByMonth, ProductTotalSalesByMonth e ProductTotalSalesCounter.

SOLUÇÕES DE ARMAZENAMENTO UTILIZADAS

Será adotado um repositório com dados históricos para análise futura, como mencionado no requisito abaixo:

"Outra ferramenta que contratamos foi um serviço de rastreamento das ações dos usuários para auxiliar a equipe de produto. Essa ferramenta captura cada clique do mouse e texto digitado pelo cliente, a sua posição geográfica estimada, quanto tempo levou em cada tela, etc; em suma, gera um volume muito grande de dados, que ainda não sabemos como serão utilizados. Provavelmente analistas irão inspecioná-los para descobrir se serão úteis de alguma forma."

O propósito abrangente desta atividade e características como Volume, Variabilidade e Velocidade com que os dados foram gerados, nos sugere um problema de Big Data. O ideal seria criar um repositório fora do banco em que as informações coletadas:

- Captura cada clique do mouse;
- Texto digitado pelo cliente;
- Posição geográfica estimada;
- Quanto tempo levou em cada tela;
- "Etc": Caracteriza a abrangência da coleta.

Para esta especificação será elaborado um Data Lake fora do escopo do banco. A transferência de dados pode ser feita por meio de JSONs com as informações segmentadas por fonte de telemetria, de maneira que os dados sejam armazenados de maneira semiestruturada.

Para isso podem ser utilizadas:

Um banco de dados distribuído *in-house*: Neste cenário os data lakes são armazenados num sistema de arquivos *Hadoop*. Neste sistema os dados são distribuídos e replicados em diversas máquinas permitindo durabilidade e disponibilidade (pela redundância). Para utilizar estes dados normalmente é necessário utilizar estratégias de *MapReduce*, para reduzir para explorar o paralelismo do sistema distribuído em *Cluster,* tipicamente com processamentos com uso do SPARK.

Repositórios na nuvem de provedores comerciais: Podem ser utilizados repositórios em nuvens como *Azure Data Lake, Amazon S3 e IBM Cloud Pak.* Estas soluções permitem não só armazenar os dados, mas realizar serviços de gestão inteligente dos dados armazenados. Uma outra vantagem das arquiteturas em nuvem é a escalabilidade do *Data Lake*. Do ponto de vista analítico em algumas soluções em nuvem as estratégias *MapReduce* ficam transparentes ao usuário, e ferramentas analíticas do próprio banco são disponibilizadas ao usuário.

A escolha da solução passa não só pelo aspecto econômico, quanto pela sensibilidade e importância estratégica e implicações jurídicas do vazamento de dados (destaca-se aqui a Lei de Proteção de Dados).

CONCLUSÃO

O sucesso da estratégia de transformação digital proposta nesse relatório, está totalmente relacionada à capacidade que nossos operadores terão em lidar com os dados: a geração, mineração, o processamento, a análise e a modelagem. Boas práticas vão gerar conhecimento e ganho de competitividade.

Nossos modelos relacional e dimensional propuseram sua forma exclusiva de armazenamento de dados com vantagens específicas, como por exemplo no modo relacional, a normalização e os modelos ER reduzem a redundância de dados. Pelo contrário, o modelo dimensional em DW organiza os dados de forma que seja mais fácil recuperar informações e gerar relatórios.

Sem dúvidas o nosso sistema de *e-commerce* será beneficiado pela modularidade existente em um banco de dados devidamente modelado. Registros são organizados de modo a viabilizar a manutenção e adição de recursos ao sistema e gerar significado relevante para apoiar nas tomadas de decisões do nosso time ao refletirem a realidade da qual foram extraídos.

BIBLIOGRAFIA

KIMBALL, Ralph; ROSS, Margy. **The Data Warehouse Toolkit: The Definitive Guide to Dimensional Modeling.** 3. ed. Indianapolis: John Wiley & Sons, Inc. 601 p.

SHARDA, Ramesh; DURSUN, Delen; EFRAIN, Turban. **Business Intelligence e Análise de Dados Para Gestão de Negócio**. 4. ed. Porto Alegre: Bookman. 2019. 584 p.

ELMASRI, Ramez; NAVATHE, Shamkant B. **Sistemas de Banco de Dados.** 6. ed. São Paulo: Pearson. 2005. 502 p.

GOLERIK, Alex. **The Enterprise Big Data Lake.** 1. ed. Califórnia: O'Reilly. 2019.