Design and Structure of Amazon's E-commerce Database β-ν-μ

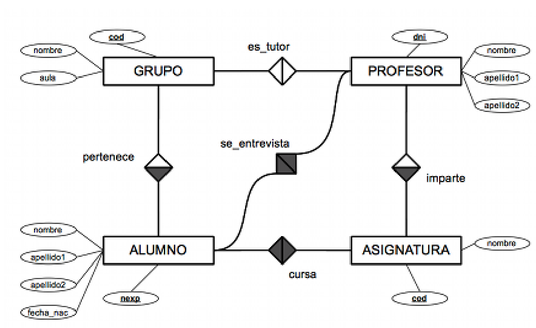
Juan David Quiroga[1], Luis Alejandro Morales[2]
Universidad Distrital Francisco Jose de Caldas
jdquirogag@udistrital.edu.co[1], lamoralesn@udistrital.edu.co[2]

Introduction

The e-commerce landscape has experienced unprecedented growth in recent years, with Amazon emerging as a leader in transforming the online shopping experience. As one of the largest global retailers, Amazon manages an enormous volume of transactions, product information, and user data daily, necessitating a robust and scalable database system capable of efficiently handling this complexity. Traditional relational databases often face challenges such as performance bottlenecks and scalability issues when processing high transaction volumes and extensive product catalogs. Consequently, it is crucial to design a database that not only stores customer and product data but also ensures rapid access and data integrity. This poster explores the challenges inherent in Amazon's database architecture and presents a solution based on an Entity-Relationship Diagram (DER). By clearly modeling the key entities—such as Customers, Products, Orders, and Shopping Carts—and defining their relationships, we aim to create an efficient database structure that enhances query performance and supports the seamless operations of the e-commerce platform.

Goal

The primary goal of this research is to design an efficient and scalable database structure for Amazon's e-commerce operations. The research question focuses on how to effectively model the relationships between key entities such as customers, products, and orders, while the expected final product is a comprehensive DER that illustrates this structure.



Proposed Solution

The proposed solution utilizes an Entity-Relationship Diagram (DER) as the foundation for designing Amazon's database. The DER provides a visual representation of the key entities involved in the e-commerce ecosystem and illustrates their attributes and relationships. This structured approach follows a systematic methodology encompassing ten steps, which include:

1. **Define components:** Identifying the main entities such as Customer, Product, Order, etc.
2. **Define entities:** Specifying each entity's role and importance in the database.
3. **Define attributes for each entity:** Listing the necessary attributes for each entity.
4. **Define relationships:** Establishing how these entities interact with each other.
5. **Define types of relationships:** Clarifying the nature of relationships (e.g., one-to-many).
6. **First view of the diagram:** Creating an initial DER to visualize the entities and their relationships.
7. **Divide many-to-many relationships:** Simplifying complex relationships into manageable structures.
8. **Second view of the diagram:** Updating the DER based on optimizations and feedback.
9. **Obtain the data structure:** Determining the tables and columns based on the DER.
10. **Define data and component properties:** Specifying data types and constraints for each component.

Results

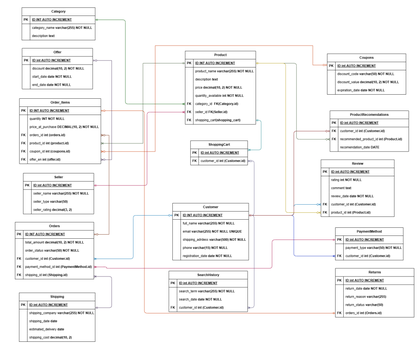
1. Define components
   - **Users:**
     - **Customers:** Interact with the site to search, evaluate, and purchase products; can leave reviews.
     - **Sellers:** Offer products on the platform.
   - **Shopping Cart System:** Enables users to add products and view selections before checkout.
   - **Payment System:** Interfaces with payment gateways (e.g., credit cards, PayPal) for transaction processing.
   - **Order Management:** Allows users to track order status (processing, shipped, delivered).
   - **Recommendation System:** Provides personalized product suggestions based on user behavior.
   - **Review and Rating System:** Facilitates users in leaving comments and ratings on products.
   - **Logistics and Shipping:** Manages inventory and coordinates product shipping.
   - **Promotions:** Implements discounts and coupons to encourage customer loyalty.
2. **Define entities and atributtes:** below are some of the entities used in this diagram
   - Customer (Usuario)
   - Product (Producto)
   - Category (Categoría de Producto)
   - Order (Pedido)
   - ShoppingCart (Carrito de Compras)
   - ShoppingCartProduct (Producto en Carrito)
   - PaymentMethod (Método de Pago)
   - Review (Reseña)
   - Shipping (Envío)
   - Offer (Oferta)
   - Seller (Vendedor)
   - SearchHistory (Historial de Búsquedas)
   - ProductRecommendations (Recomendaciones de Productos)
   - Returns (Devoluciones)
   - Coupons (Cupones)
   - OrderItems (Detalles del Pedido)
3. **Define relationships:**

	Customer	Product	Category	Order	ShoppingCart	ShoppingCartProduct	PaymentMethod	Review	Shipping	Offer	Seller	SearchHistory	ProductRecommendations	Returns	Coupons	OrderItems
Customer				X	X		X	X				X	X			
Product			X			X		X		X	X		X		X	X
Category		X														
Order	X						X		X							X
ShoppingCart	X					X										
ShoppingCartProduct		X			X											
PaymentMethod	X			X												
Review	X	X														
Shipping				X												
Offer		X														
Seller		X														
SearchHistory	X															
ProductRecommendations	X	X														
Returns																X
Coupons		X														X
OrderItems		X		X										X	X	

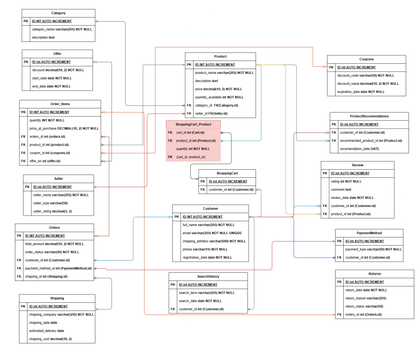
4. **Define types of relationships:**

Entidad 1	Entidad 2	Tipo de Relación	Descripción
Customer	Order	1:N	Un cliente puede hacer múltiples pedidos, pero un pedido pertenece a un solo cliente.
Customer	Shopping Cart	1:1	Cada cliente tiene un único carrito de compras.
Customer	Payment Method	1:N	Un cliente puede tener múltiples métodos de pago, pero un método de pago pertenece a un solo cliente.
Customer	Review	1:N	Un cliente puede escribir múltiples reseñas, pero cada reseña pertenece a un solo cliente.
Customer	Search History	1:N	Un cliente puede tener múltiples búsquedas registradas, pero cada búsqueda pertenece a un solo cliente.
Customer	Product Recommendations	1:N	Un cliente puede recibir múltiples recomendaciones de productos, pero cada recomendación pertenece a un solo cliente.
Product	Category	N:1	Un producto pertenece a una sola categoría, pero una categoría puede tener múltiples productos.
Product	Review	1:N	Un producto puede tener múltiples reseñas, pero cada reseña pertenece a un solo producto.
Product	Shopping Cart Product	N:M	Un producto puede estar en múltiples carritos y un carrito puede contener múltiples productos.
Product	Order Items	N:M	Un producto puede estar en múltiples pedidos y un pedido puede contener múltiples productos.
Product	Coupons	N:1	Un cupón puede aplicarse a un solo producto, pero un producto puede tener múltiples cupones disponibles.
Category	Product	1:N	Una categoría puede tener múltiples productos, pero cada producto pertenece a una sola categoría.
Order	Customer	N:1	Un pedido pertenece a un solo cliente, pero un cliente puede hacer múltiples pedidos.
Order	Payment Method	N:1	Un pedido utiliza un solo método de pago, pero un método de pago puede usarse en múltiples pedidos.
Order	Shipping	1:1	Un pedido tiene un único envío asociado.
Order	Order Items	1:N	Un pedido puede contener múltiples productos.
Shopping Cart	Customer	1:1	Cada cliente tiene un único carrito de compras.
Shopping Cart	Shopping Cart Product	1:N	Un carrito de compras puede contener múltiples productos.
Shopping Cart Product	Product	N:M	Un producto puede estar en múltiples carritos, y un carrito puede contener múltiples productos.
Payment Method	Customer	N:1	Un cliente puede tener múltiples métodos de pago.
Payment Method	Order	1:N	Un pedido utiliza un solo método de pago, pero un método de pago puede usarse en múltiples pedidos.
Review	Customer	N:1	Una reseña pertenece a un solo cliente, pero un cliente puede hacer múltiples reseñas.
Review	Product	N:1	Una reseña pertenece a un solo producto, pero un producto puede tener múltiples reseñas.
Shipping	Order	1:1	Cada pedido tiene un único envío.
Offer	Product	1:N	Un producto puede tener múltiples ofertas, pero cada oferta está asociada a un solo producto.
Seller	Product	1:N	Un vendedor puede vender múltiples productos, pero un producto pertenece a un solo vendedor.
Search History	Customer	N:1	Una búsqueda pertenece a un solo cliente, pero un cliente puede tener múltiples búsquedas registradas.
Product Recommendations	Customer	N:1	Una recomendación pertenece a un solo cliente, pero un cliente puede recibir múltiples recomendaciones.
Returns	Order Items	1:1	Cada devolución está asociada a un único producto dentro de un pedido.
Coupons	Order Items	N:1	Un cupón puede aplicarse a un solo producto dentro de un pedido.
Order Items	Order	N:1	Un pedido puede contener múltiples productos, pero cada producto dentro del pedido pertenece a un solo pedido.
Order Items	Product	N:1	Un producto dentro de un pedido pertenece a un solo producto, pero un producto puede estar en múltiples pedidos.

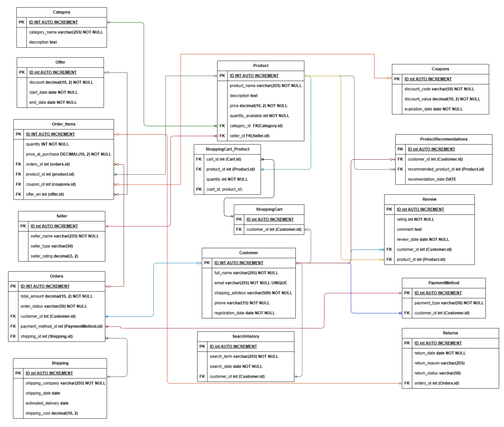
5. **First view of the diagram:**



6. **Second view of the diagram with many-to-many relationships divided:**



7. **Define data and component properties and structure:**



Conclusions

The use of entity-relationship diagrams (DERs) is crucial for effective database design, particularly for e-commerce platforms like Amazon. DERs offer a clear visual depiction of the entities within the system, along with their attributes and relationships. This visualization aids in comprehending the database structure and ensures that all components are logically and coherently interconnected.

For Amazon, the complexity of its data ecosystem requires a meticulously planned and structured approach. Proposing a database that incorporates essential entities such as "Customers," "Products," "Orders," and "Shopping Carts" enables efficient management of daily operations and enhances the user experience. By defining relationships, especially between products and shopping carts, common redundancy issues can be addressed, and referential integrity can be preserved.

Bibliography

Edraw Software. (n.d.). *ER Diagram.* Recuperado de https://www.edrawsoft.com/es/er-diagram/

Universidad de Buenos Aires. (n.d.). *Diagrama Entidad Relación.* Recuperado de https://repositorio.ub.edu.ar/handle/123456789/5155