

# **Persistência Poliglota**

Prof Andreza Leite

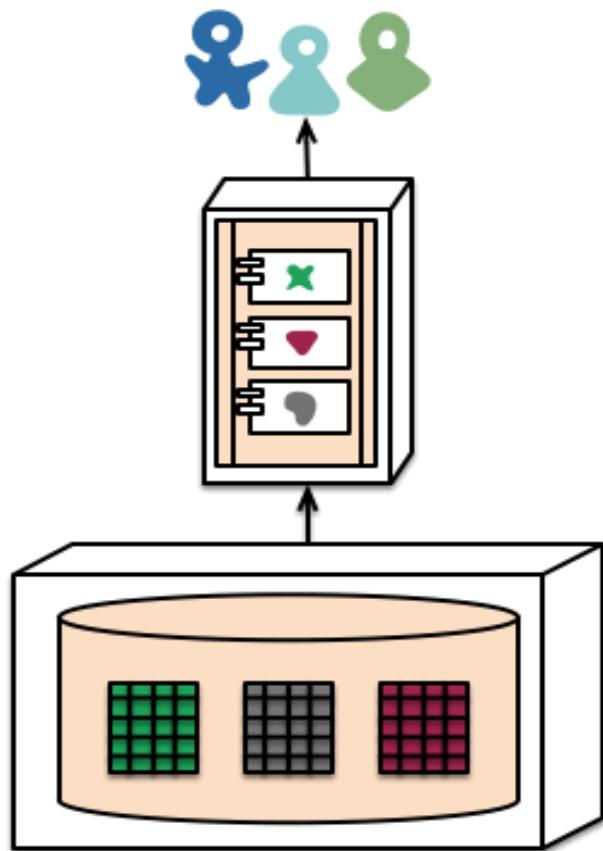
# Poliglota



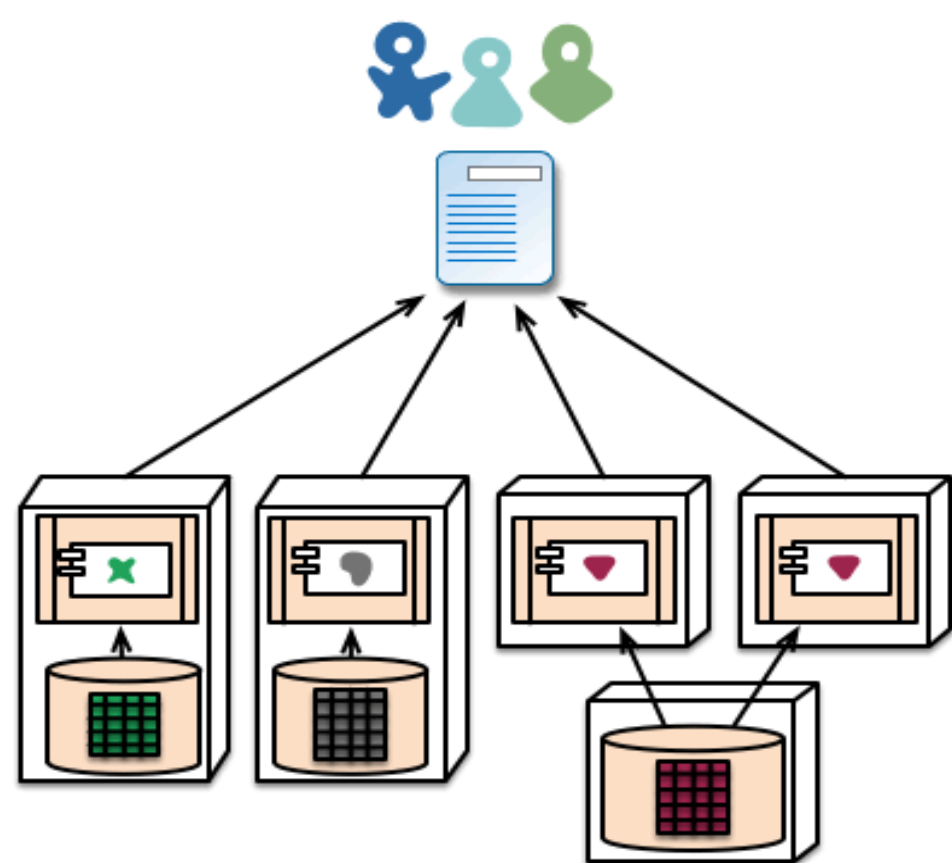
# Persistência Poliglota (PP)



# PP::monolítico X microserviços



Ambiente monolítico - único banco de dados



Microserviços - banco de dados por aplicação

# PP::aplicação eficiente

Atende bem seu objetivo e está sempre disponível quando é requisitada;

- Escalável, Segura, Nunca é o gargalo para inovações

Pode usar mais de uma solução de persistência:

- SQL onde for mais apropriado (ACID, transações, normalização);
- NoSQL onde você não precisa das features acima;

# PP::Arquiteto

Não amarrar a  
uma única  
solução

Pode usar SQL e  
NoSQL ao  
mesmo tempo.

Flexível, adiciona  
novas features  
rapidamente



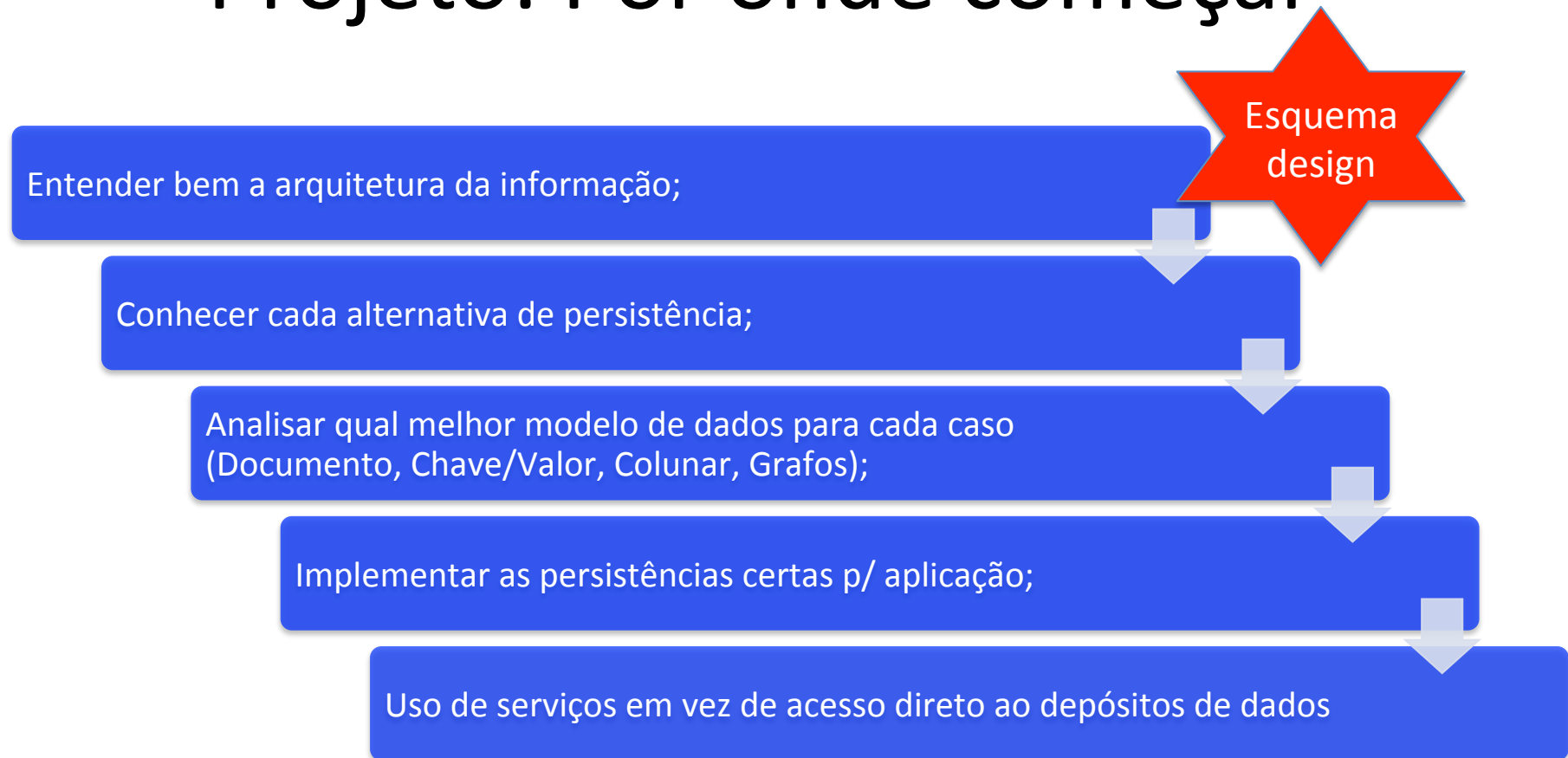
+caro?

Sim! +máquinas,  
+pessoas qualificadas,  
+infra...

=

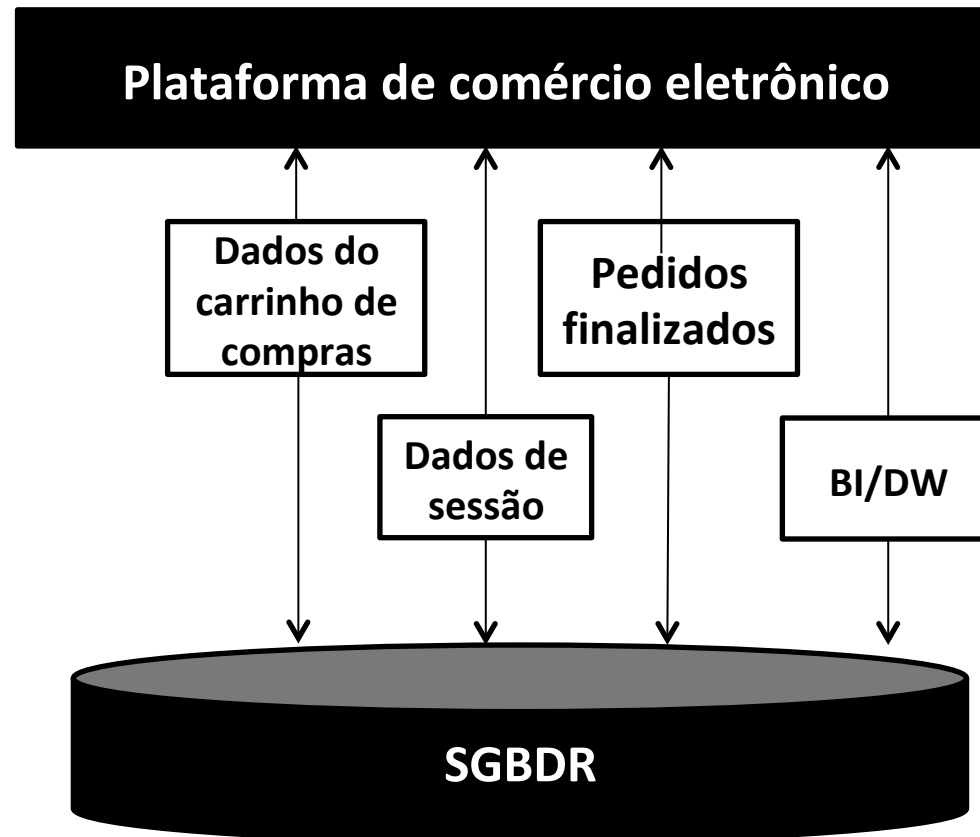
Com uma App +rápida,  
estável, segura e  
escalável vai vender +...

# Projeto: Por onde começar





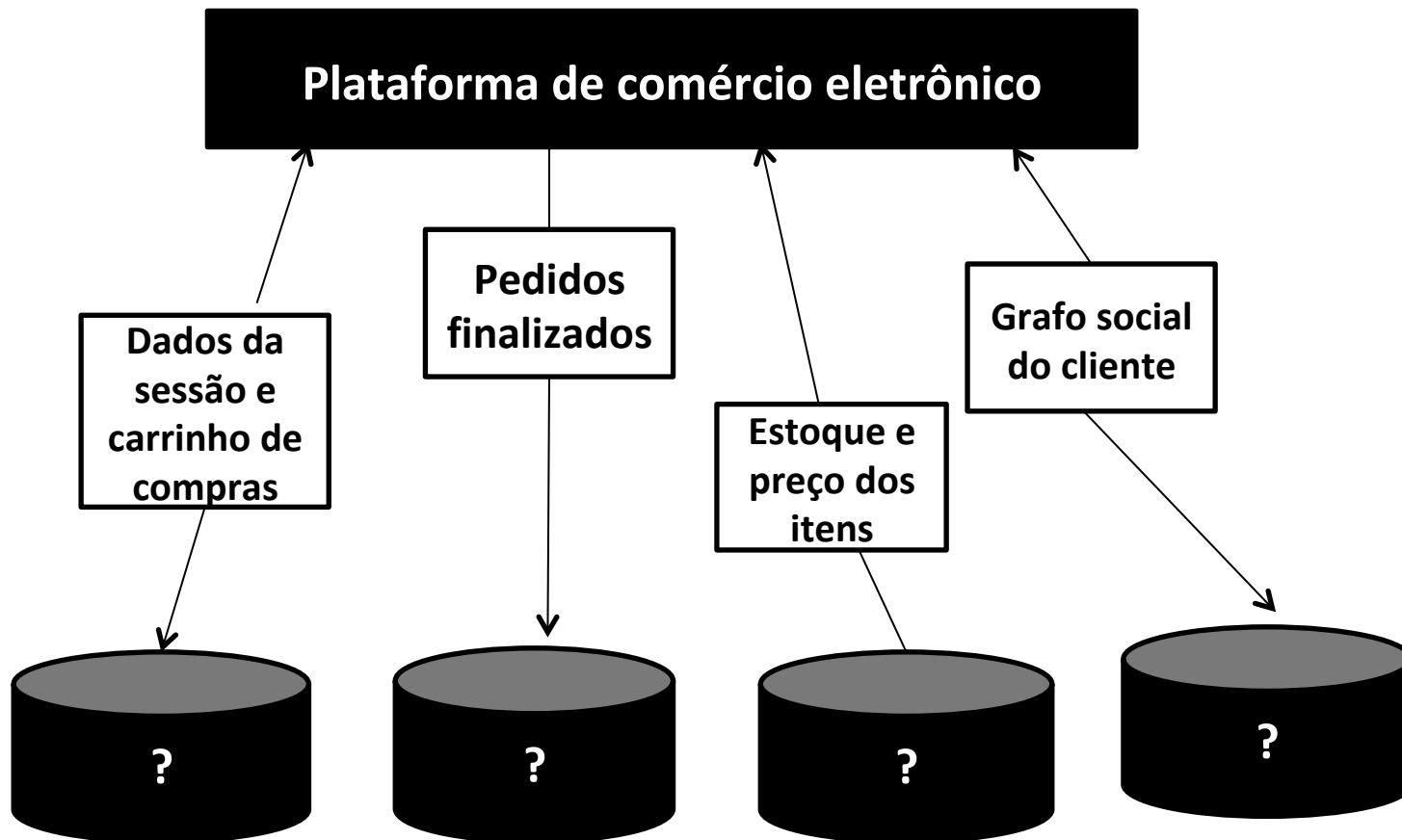
# e-commerce



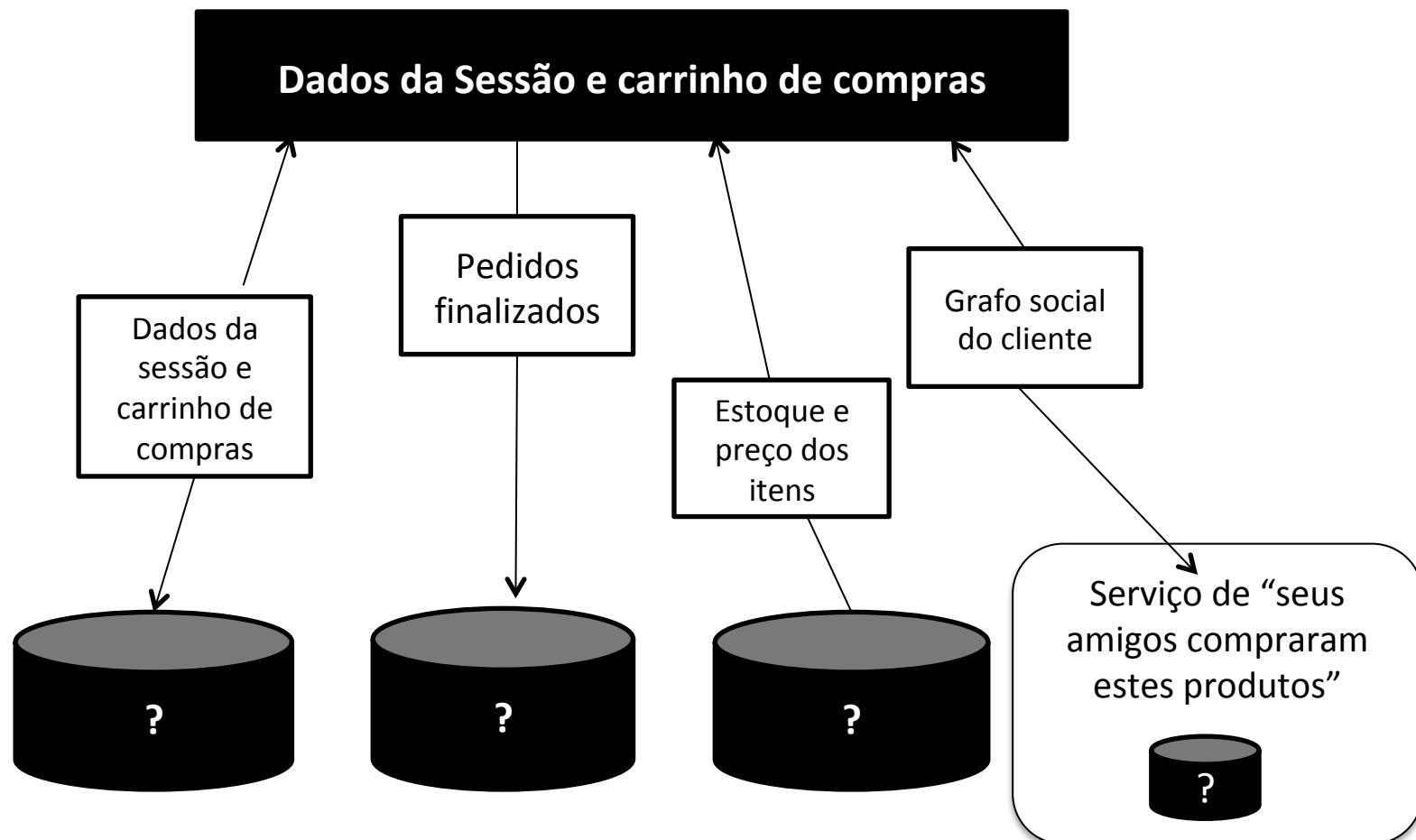
# Exemplo e-commerce

- Catálogo de produtos é a parte mais acessada de uma loja virtual, o volume de acessos é alto (blackfriday, natal, dia das mães).
- Nem todo mundo que acessa vai comprar naquele momento.
  - Colocar o catálogo de produtos em um banco separado pode destravar o e-commerce?
  - Os dados dos usuários, informações financeiras e tudo aquilo que exige transação ACID pode ficar em outro banco?
    - Assim o SQL só será acessado quando o usuário for concluir a compra.

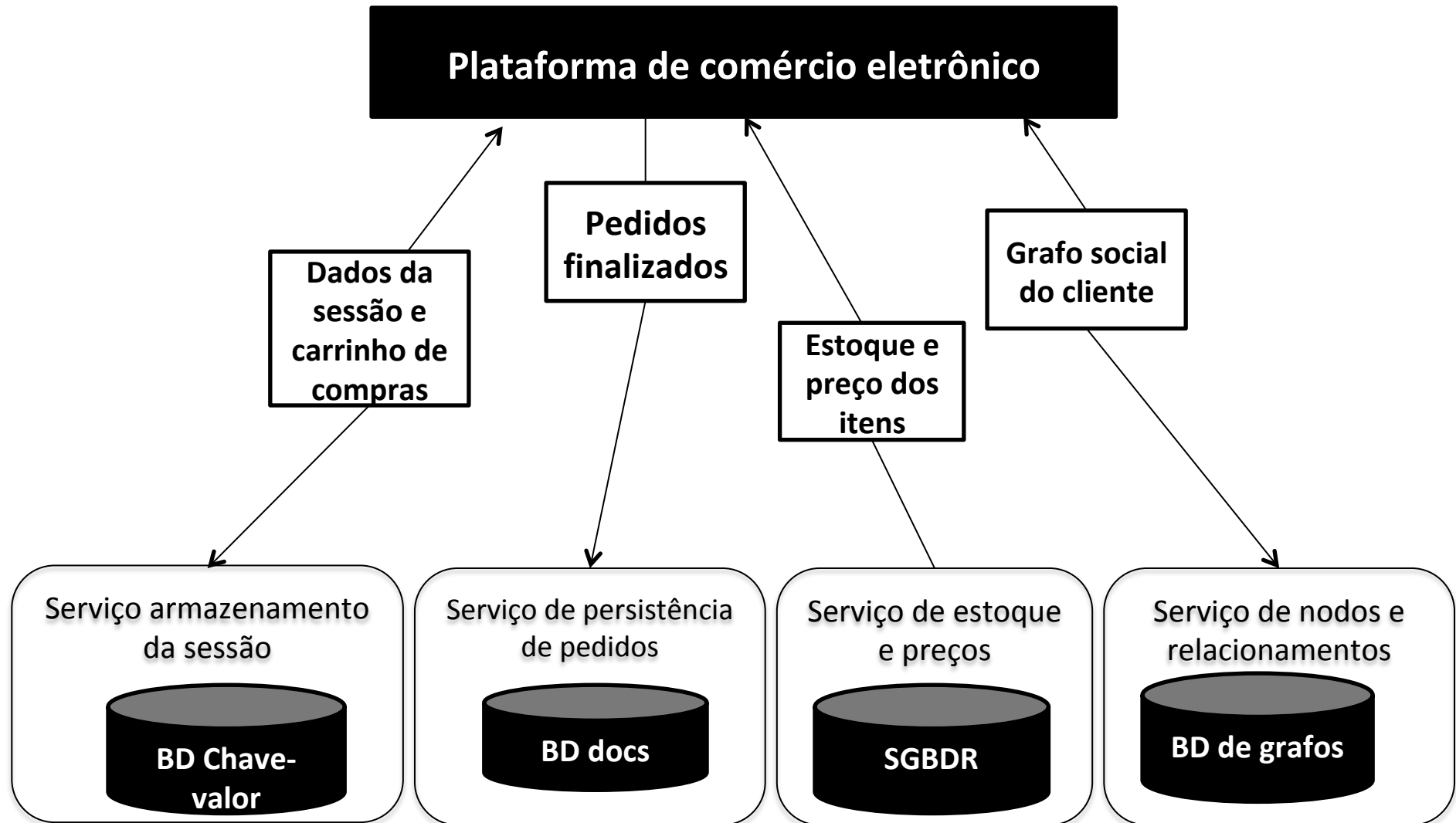
# e-commerce



# Exemplo e-commerce



# e-commerce

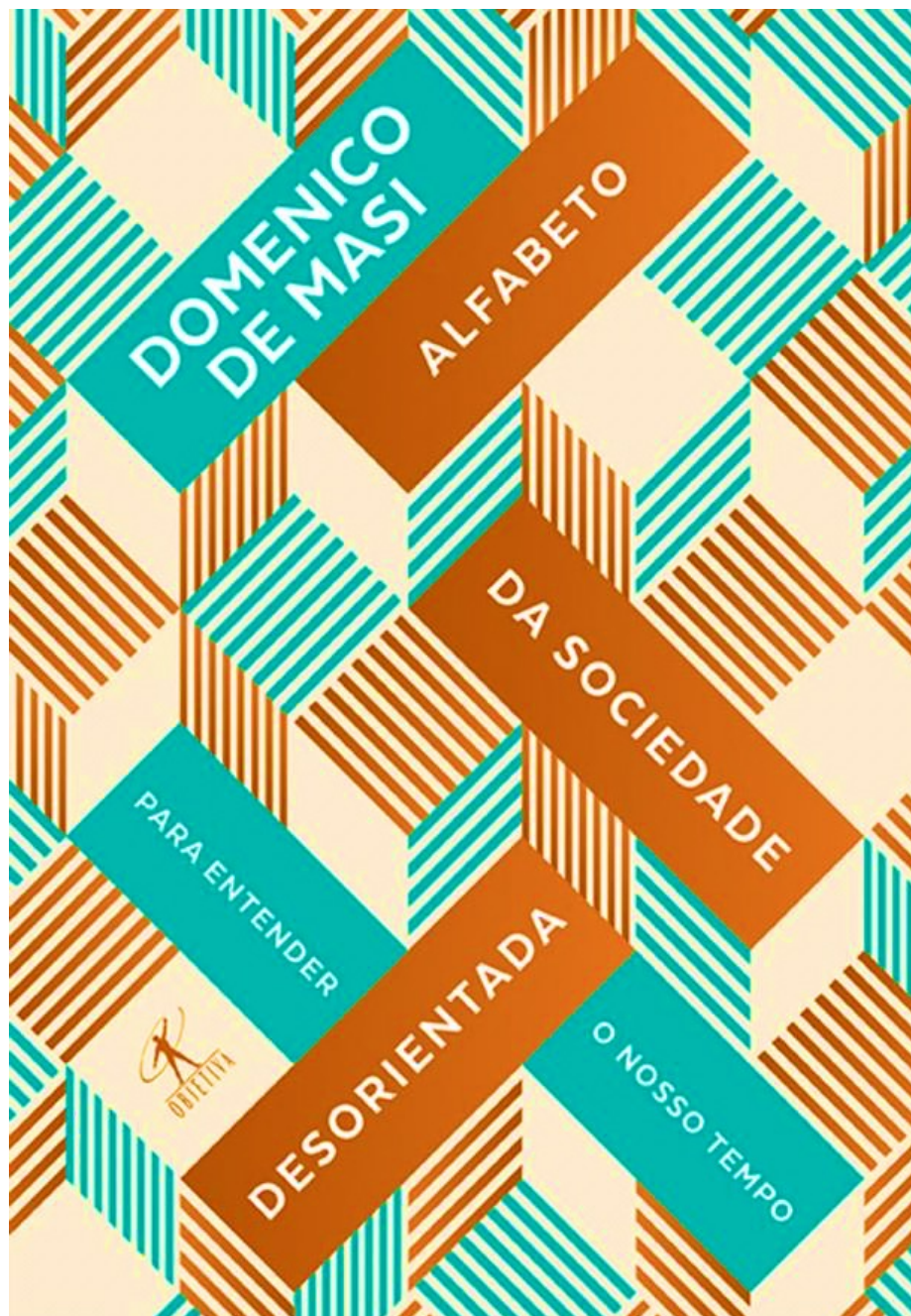


Muitos !  
Modelos !



*'Nossa sensação de  
crise deriva da falta de  
um modelo condutor',*

Domenico de Masi



O verbete que  
mais parece se  
aproximar de  
nossa realidade é  
o da letra D,  
**‘Desorientação’**





Calma com o andor



# Apache MetaModel

Relational  
databases



PostgreSQL



MySQL



Oracle DB



SQL Server



Embedded DBs

NoSQL  
databases



CouchDB



MonqoDB



HBase



Cassandra



ElasticSearch

Business  
apps



Salesforce.com



SugarCRM

Data files  
+ formats



CSV files



Spreadsheets

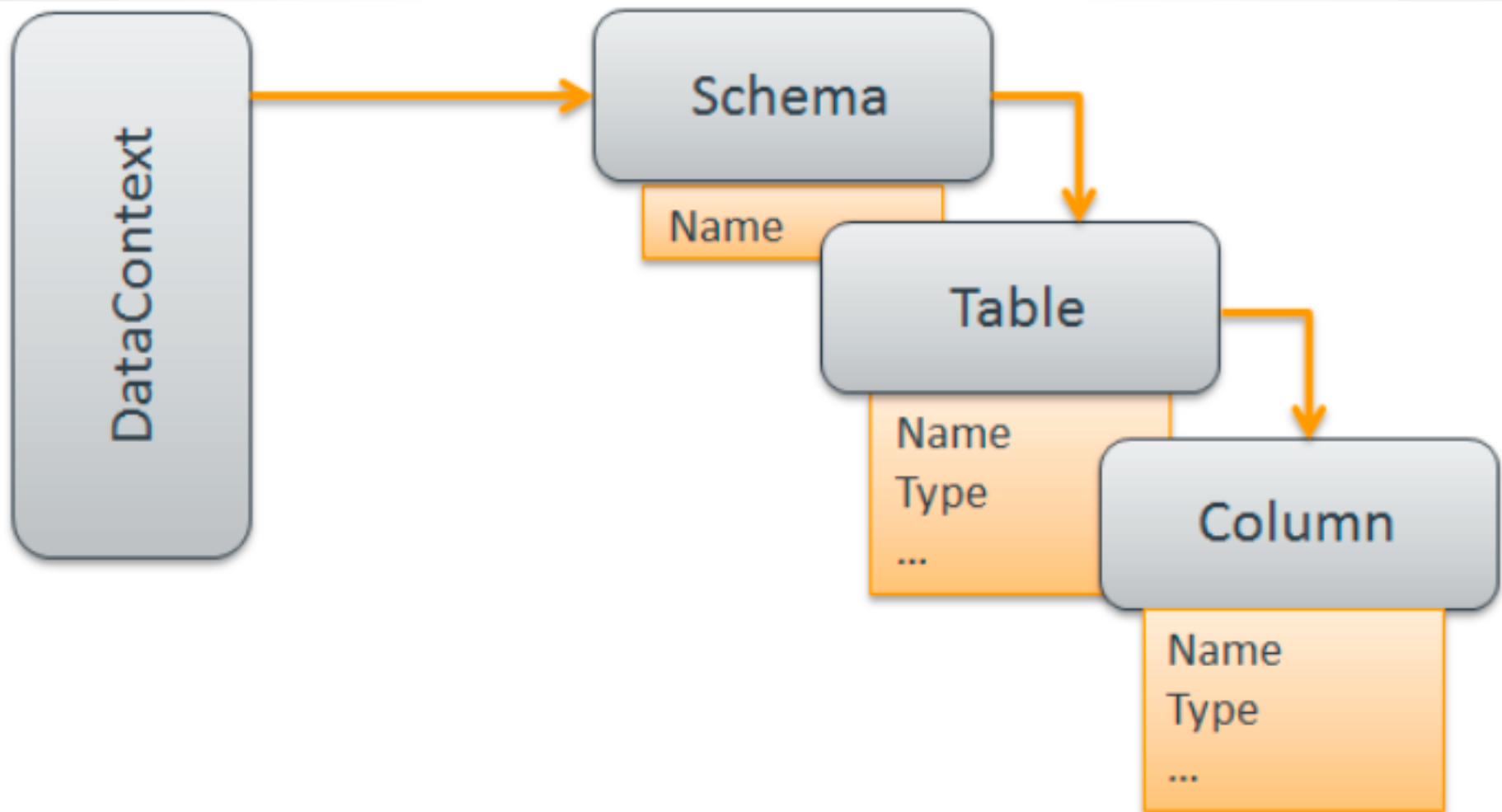


XML files

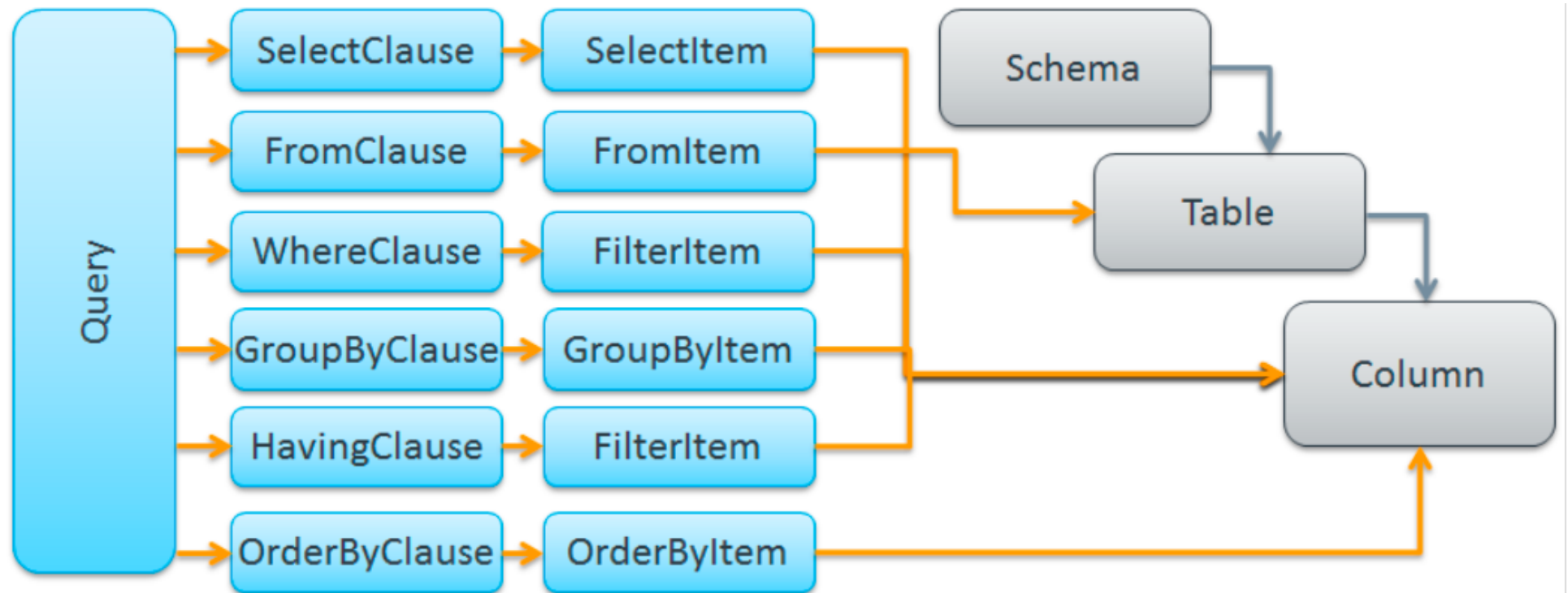


JSON files

# MetaModel *Schema* Model



# MetaModel *query* and *schema* model



# Query representation for developers

```
"SELECT * FROM foo"
```

```
Query q = new Query();  
q.from(table).selectAll();
```

## Query como uma **string**

- Fácil de escrever e ler
- Propenso a erros de análise/parsing
- Propenso a erros de digitação
- Falha em tempo de execução

## Query como um **objeto**

- Código mais envolvente para ler e escrever
- Presta-se a inspeção e mutação
- falha em tempo de compilação



**Projeto!!!**



KEEP  
CALM  
AND  
ESTUDE



# VIDA DE PROGRAMADOR

.COM.BR

/\* SUGESTÃO  
ENVIADA POR  
@OSLISSA \*/



#98

CARA, ESTOU FAZENDO  
UM CV PARA UMA VAGA...  
PODE ME AJUDAR?

O QUE VOCÊ JÁ  
COLOCOU?

SEI MEXER NO COMPUTADOR,  
DIGITAR, VER E-MAIL,  
MEXO NO FRONTPAGE...

MANJA DE SQL?

NÃO...

ENTÃO BOTA AÍ:  
"ESPECIALISTA EM  
NOSQL"

AH, LEGAL...