A origem da complexidade - Dívida técnica na área de dados

André Gomes dos Santos

Diamante

AGGRANDIZE

COMMVAULT

TD SYNNEX

Platina

GOLDENGATEBR
Consultoria & Treinamentos

DISCOVER

Ouro

scansource

VERTICA
by opentext

Prata

TRACES

CAFÉ COM CLOUD

ROX
We take care of your data

Apoio

FIAP

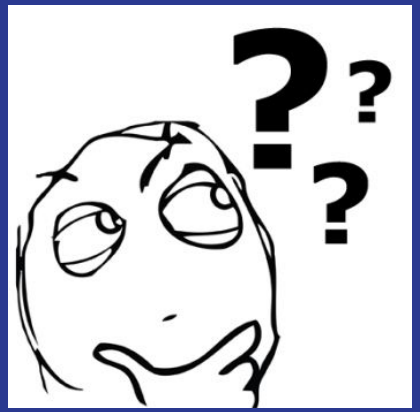
GRUPO POSEIDON
DIGITAL

Who am I ?

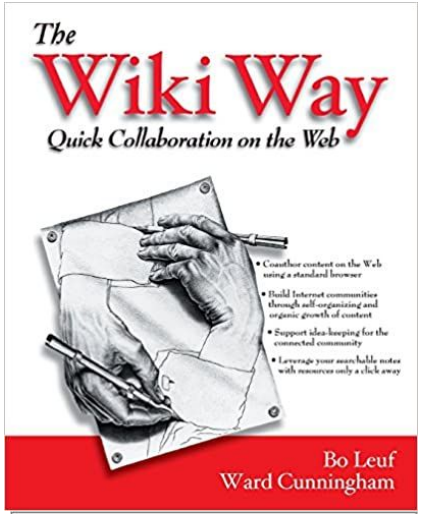
https://www.linkedin.com/in/andregoems/

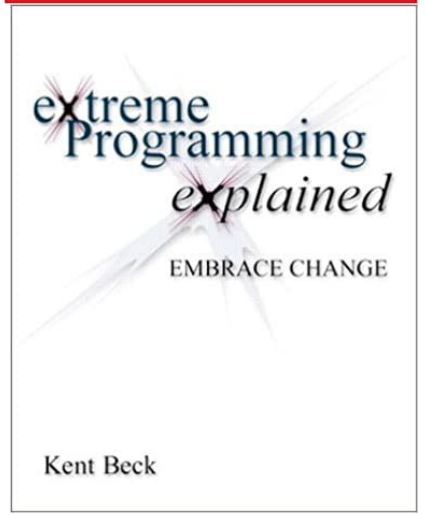
Disclaimer

Dívida (débito) técnica ?



Ward Cunningham

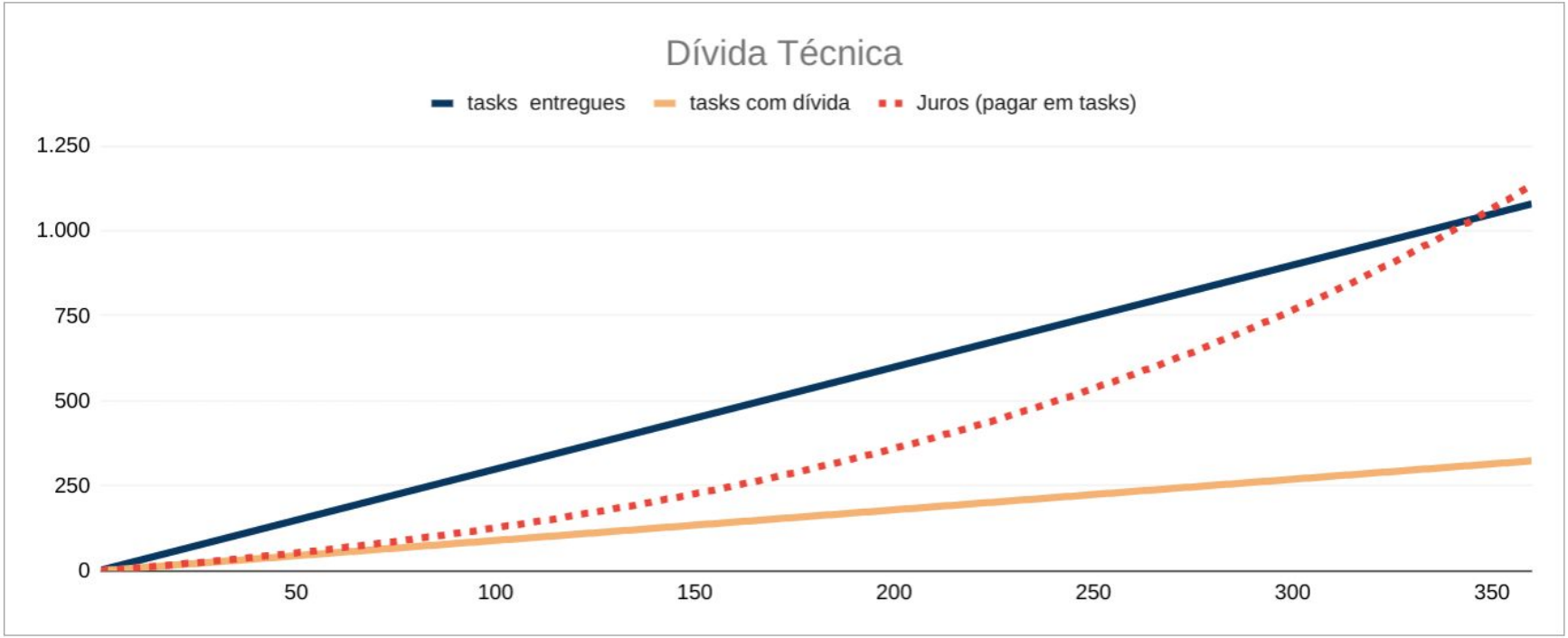




É uma **metáfora** criada para explicar a **natureza evolutiva** de um software, e também o **custo de aprendizado e refatoração** no decorrer da vida da aplicação.

Evento	Valores
Total pago no financiamento	R$ 216.797,75
Valor financiado	R$ 100.000,00
Juros devidos	R$ 116.797,75
Pagamento mensal estimado	**R$ 3.613,30**

360 dias
1.080 tasks (esforço/custo/tempo/time)



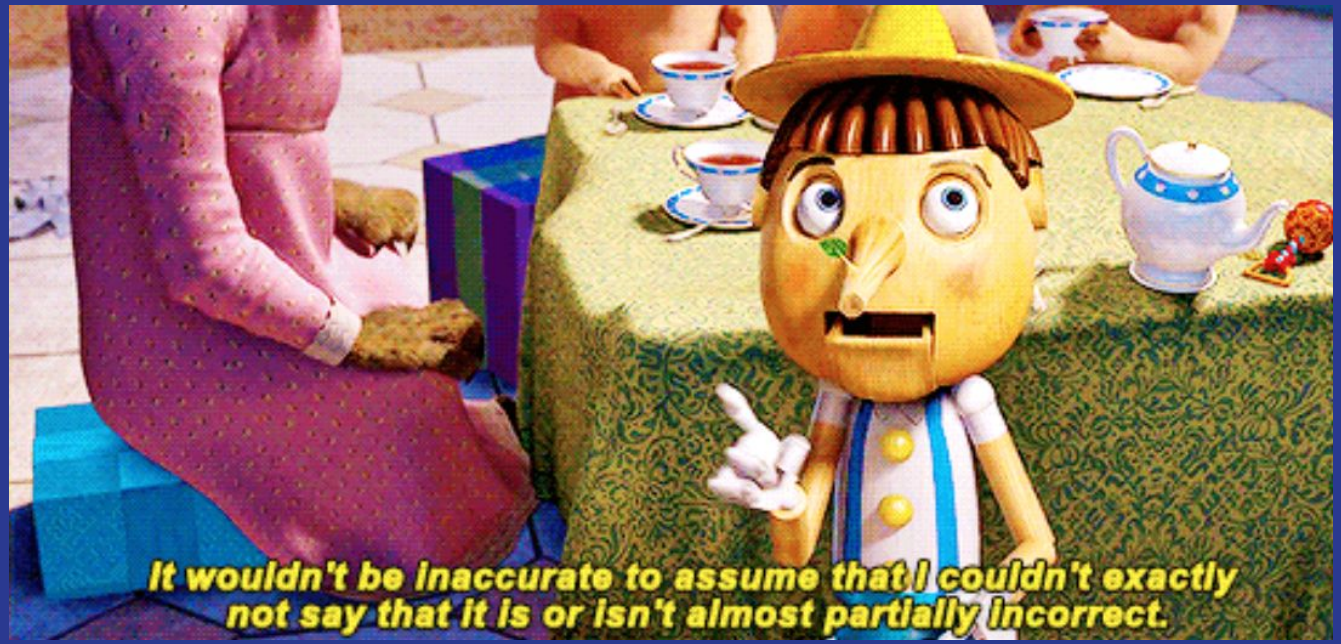
Faturamento de 50.000 R$ em dias normais

Faturamento de 300.000 R$ em feriados.

Evento	Valores
Total pago no financiamento	R$ 216.797,75
Valor financiado	R$ 100.000,00
Juros devidos	R$ 116.797,75
Pagamento mensal estimado	**R$ 3.613,30**

- **task/job/feature/tabela/relatório entregue/feito/desenvolvido sem valor percebido é custo.**
- **Dívida deve ser controlada.**
  - Dívida deve ser adquirida conscientemente.
  - Dívida deve ser paga.

Falácias


It wouldn't be inaccurate to assume that I couldn't exactly
not say that it is or isn't almost partially incorrect.

As 8 falácias da computação distribuída.

1. A rede é de confiança.
2. A latência é zero.
3. Largura de banda é infinita.
4. A rede é segura.
5. A topologia não muda.
6. Há somente um administrador.
7. O custo de transporte é zero.
8. A rede é homogênea.

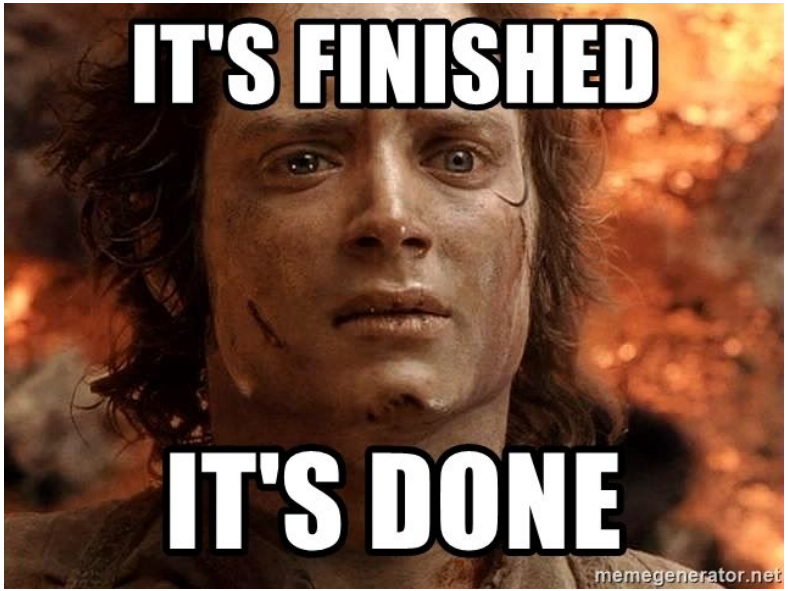
https://cybernetus.com/post/as-oito-falacias-da-computacao-distribuida/

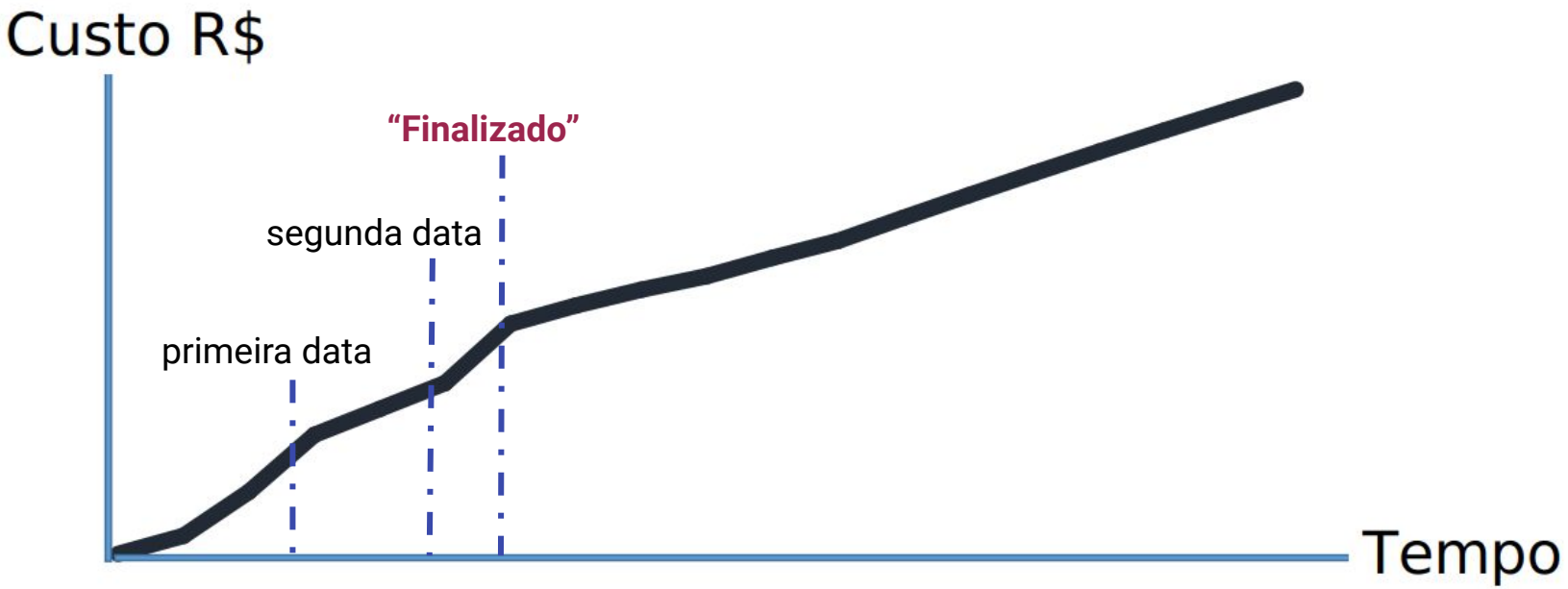
Falácias da computação empresarial. (Grandes projetos/empresas)

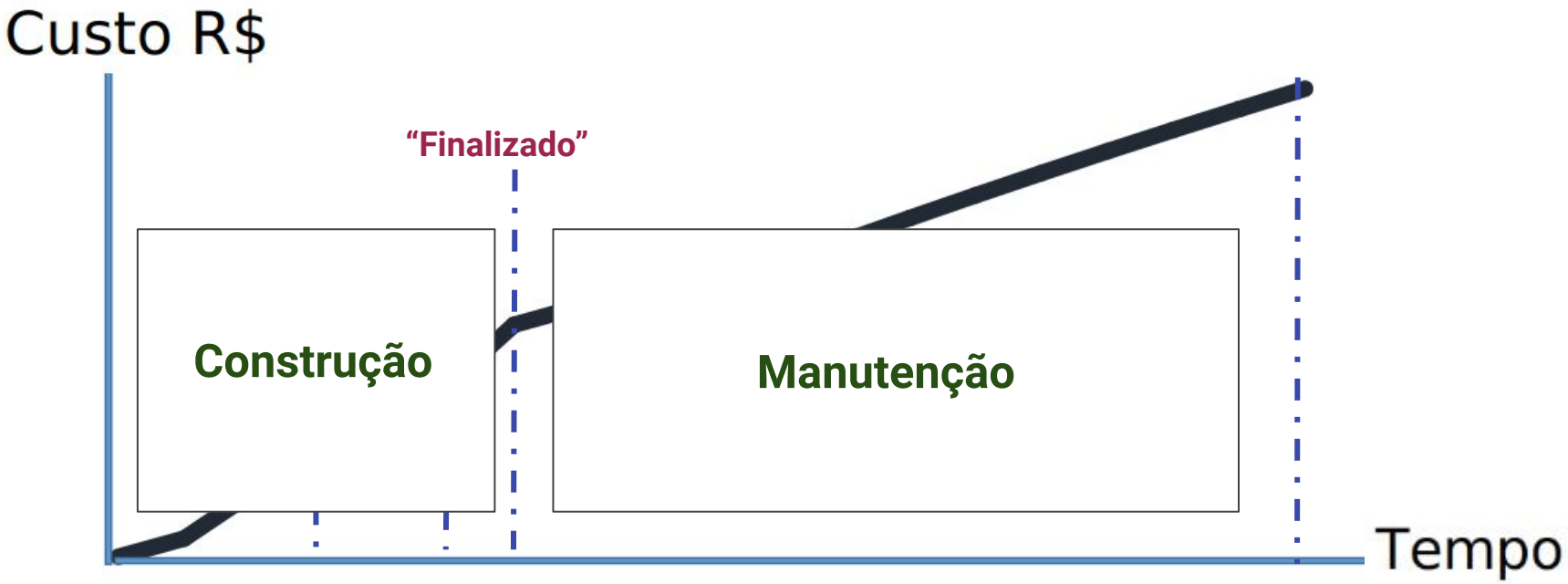
1. **A nova tecnologia é sempre melhor do que a tecnologia antiga.**
2. Sistemas não são “sistemas distribuídos”.
3. **A lógica de negócios pode e deve ser centralizada.**
4. **Dados, objetos ou qualquer outro tipo de modelo podem ser centralizados.**
5. O sistema é monolítico.
6. **O sistema está finalizado.**
7. **Os fornecedores podem fazer com que os problemas desapareçam.**
8. **A arquitetura é a mesma em todos os lugares.**
9. **Os desenvolvedores só precisam se preocupar com problemas de desenvolvimento.**

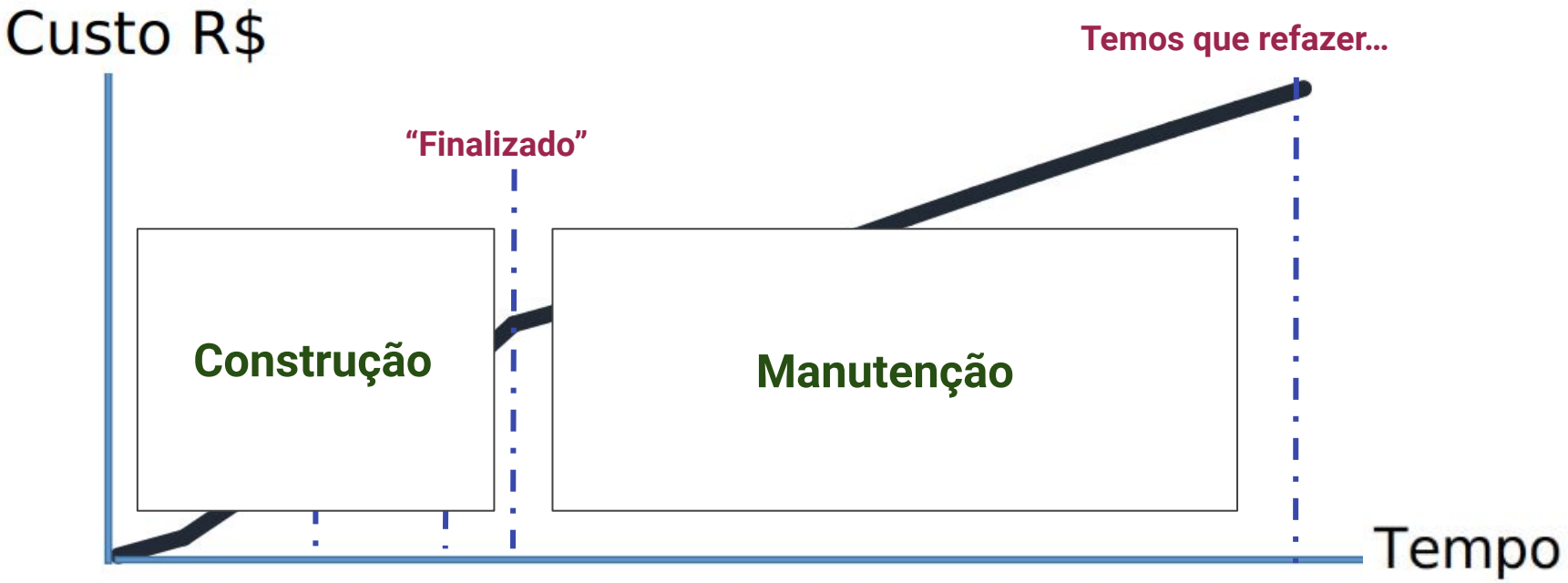
http://blogs.tedneward.com/post/enterprise-computing-fallacies/

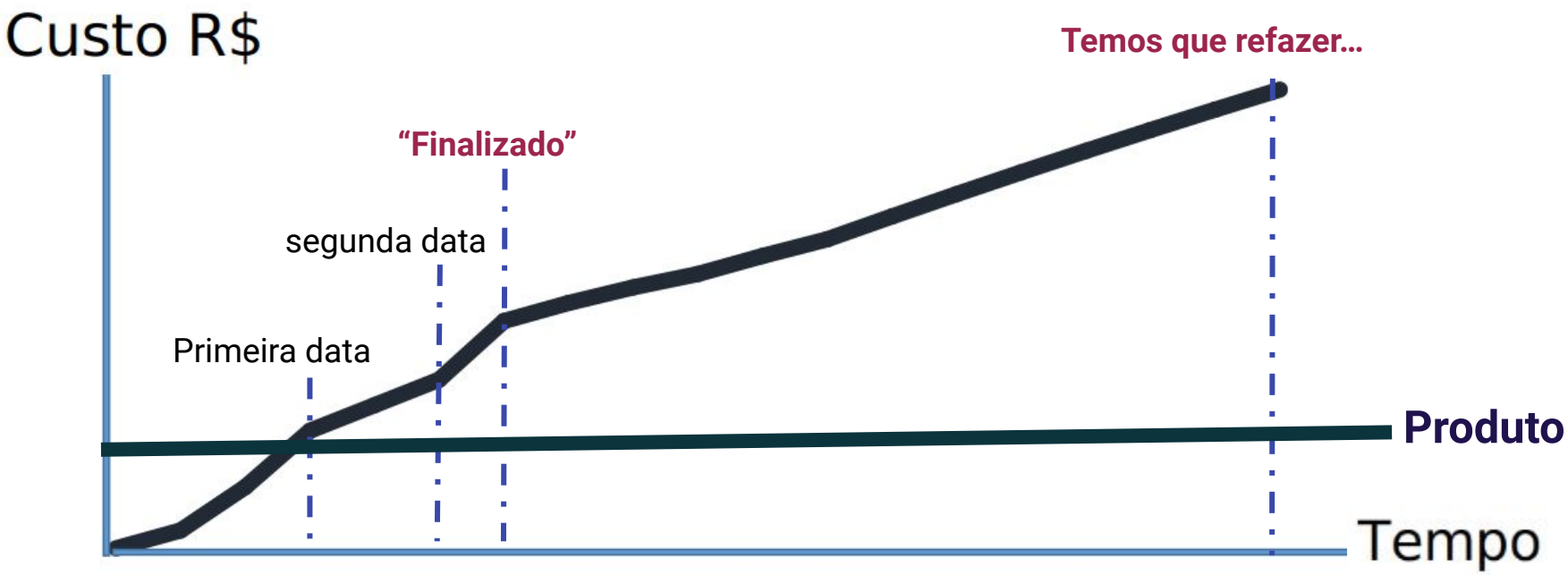
Falácia: O sistema está finalizado.




Custo R$
“Finalizado”
segunda data
primeira data
Tempo


Custo R$
“Finalizado”
Construção
Manutenção
Tempo

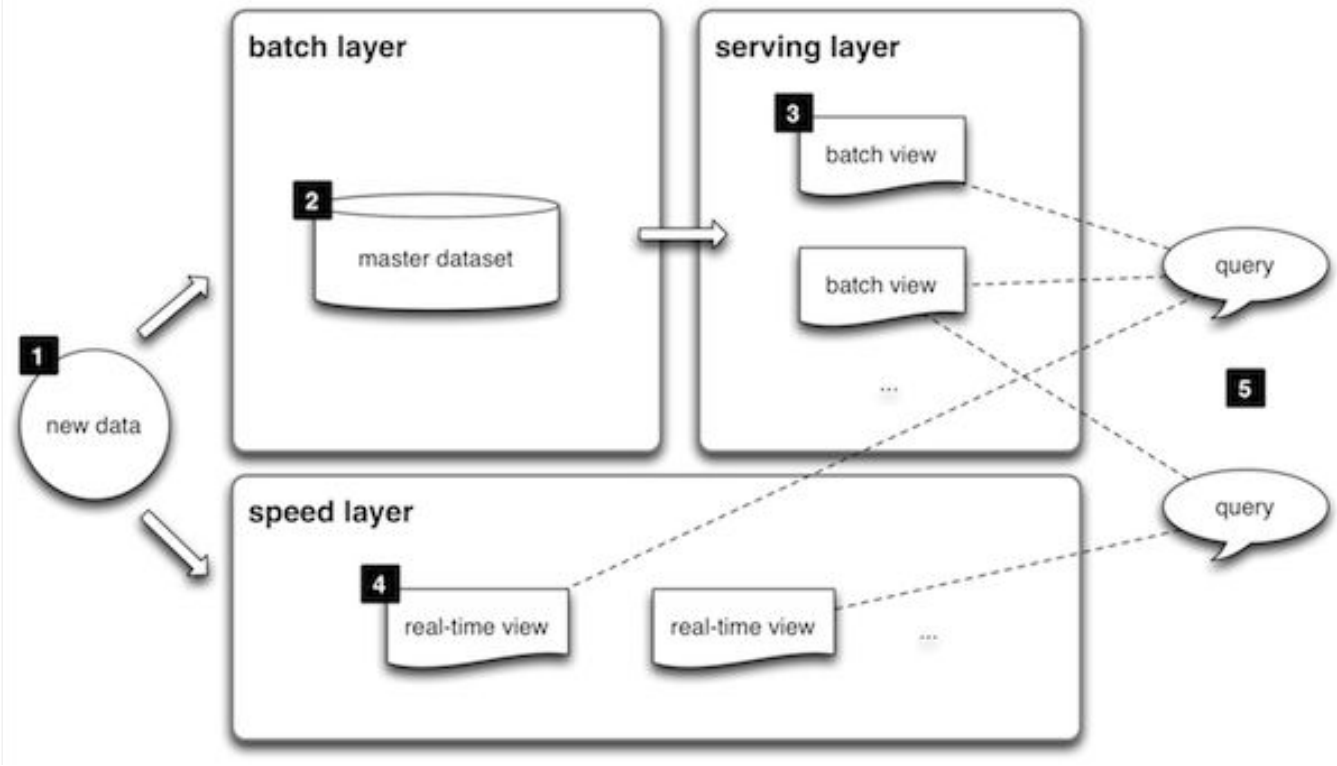

Custo R$
Temos que refazer...
“Finalizado”
Construção
Manutenção
Tempo

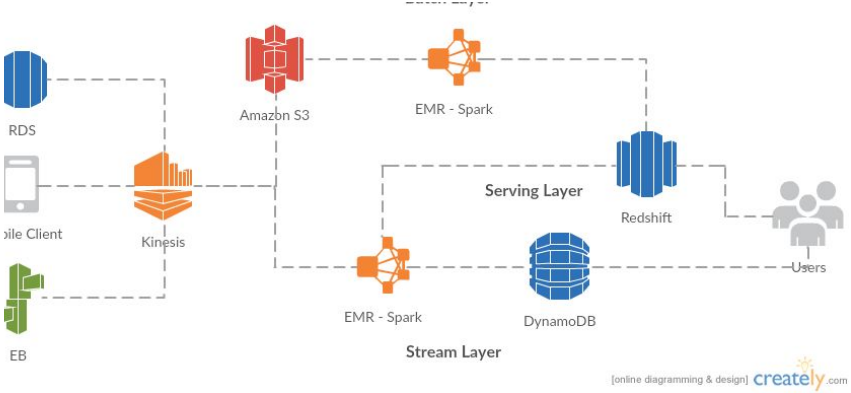

Custo R$
Temos que refazer...
“Finalizado”
segunda data
Primeira data
Produto
Tempo

- **Desenvolvimento (dados) não é um projeto de construção civil, a metáfora induz a “erros”.**
- Não existe um “momento de manutenção”.
- Refazer tudo não é a solução (eventualmente é necessário).
- Iniciativas em **TI só terminam quando o “código”** é removido/descartado.
- Iniciativas em **Dados só terminam quando o “dado”** é removido/descartado.
- **Projetos/Iniciativa = Produtos:**
  - **Precisa de feedback do usuário.**
  - **Precisa gerar/demonstrar valor/utilidade.**
  - **é uma iniciativa de longo período.**
  - **https://martinfowler.com/articles/products-over-projects.html**

Falácia: A arquitetura é a mesma em todos os lugares.

Arquitetura Lambda





Na AWS

Data Lake

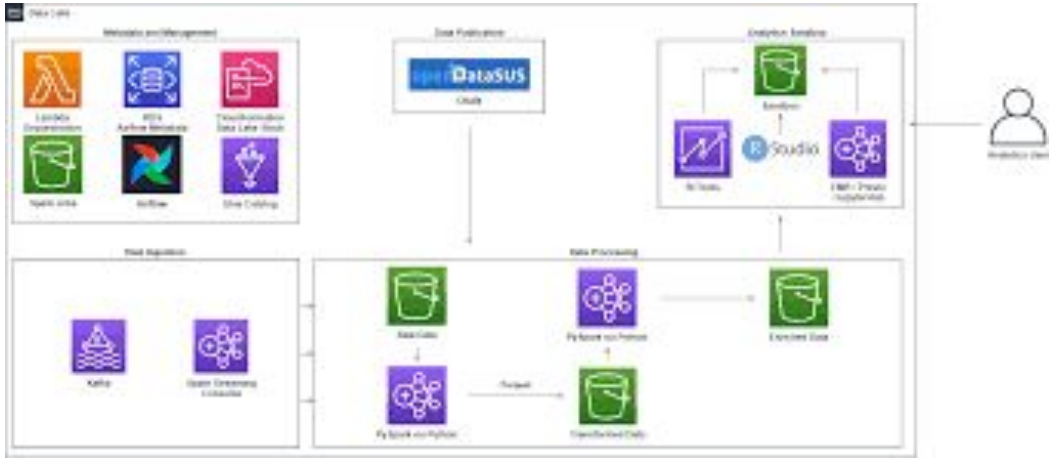
- Armazenamento: **S3**
- Metadata: **Glue**

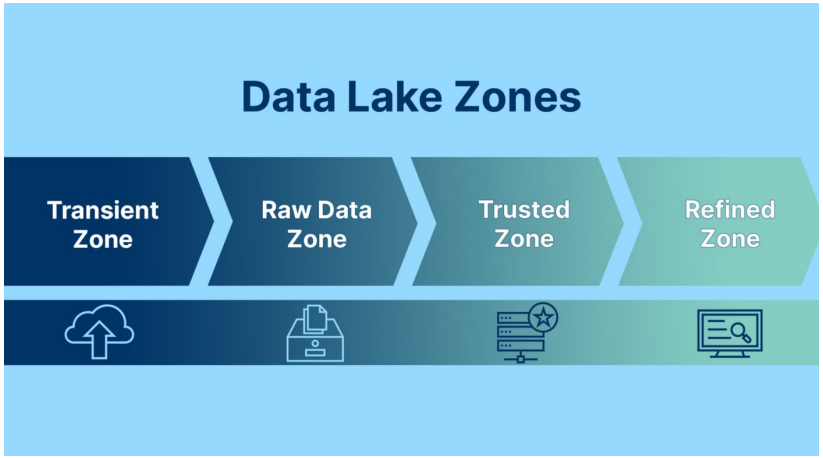
Lambda

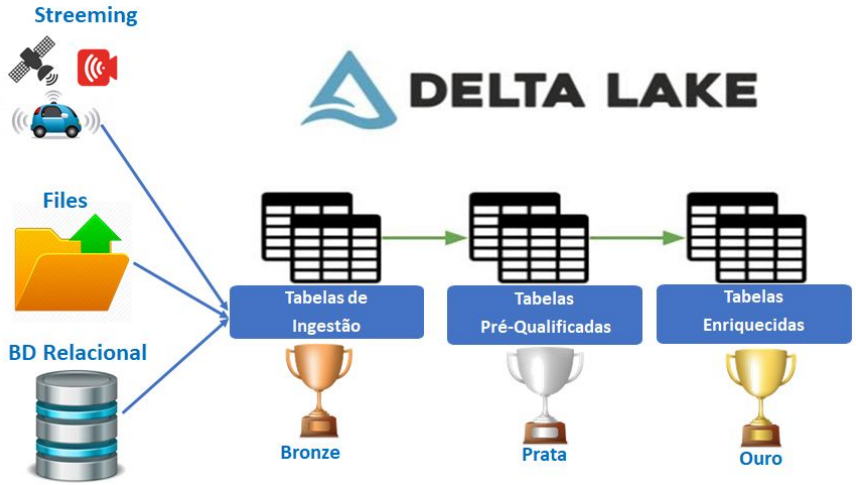
- Processamento: **Kinesis**, Kafka, Spark, etc

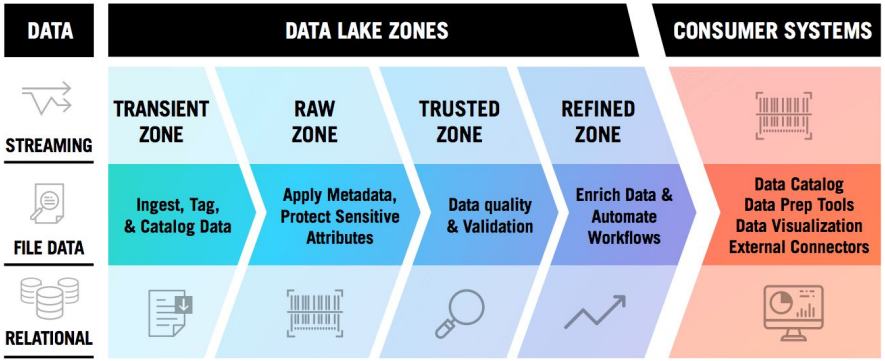
Serving

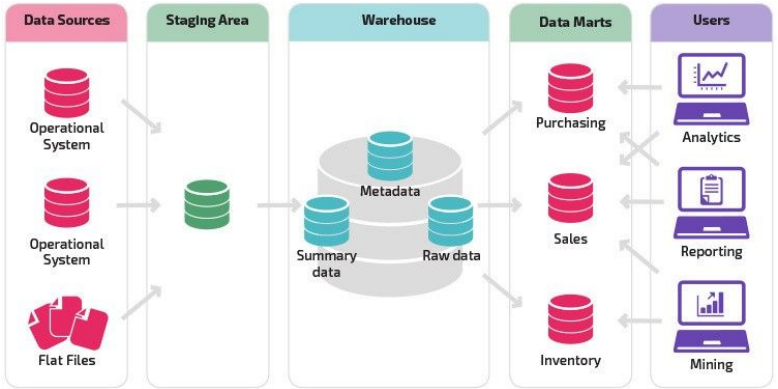
- Data Warehouse: Redshift
- Machine Learning: Sagemaker
- Analytics: Quicksight, Elastic
- Ad-Hoc: **Athena**, Redshift Spectrum, Presto, etc



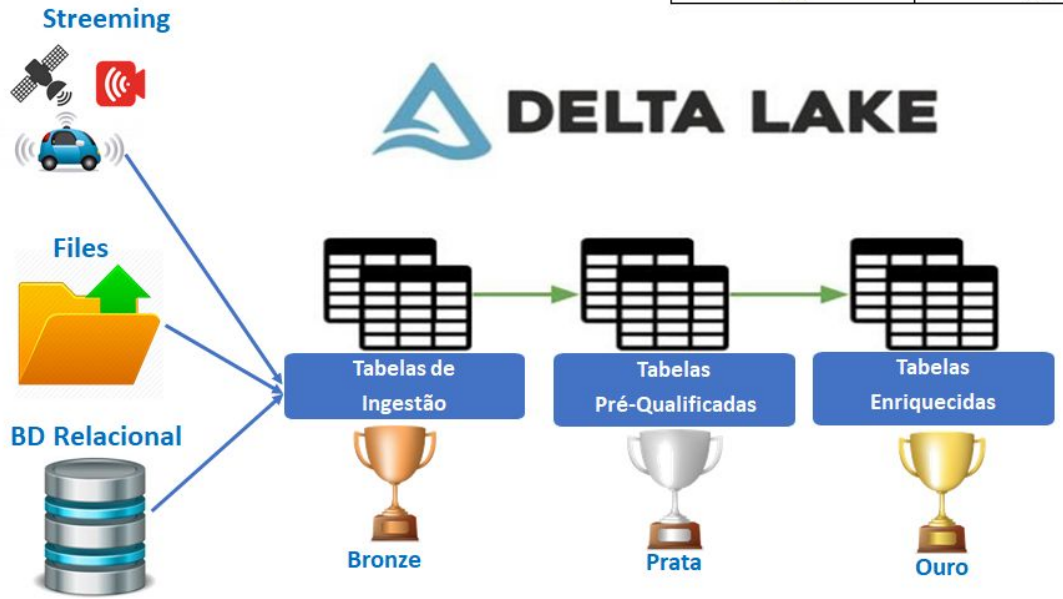

Data Lake Zones
Transient Zone
Raw Data Zone
Trusted Zone
Refined Zone

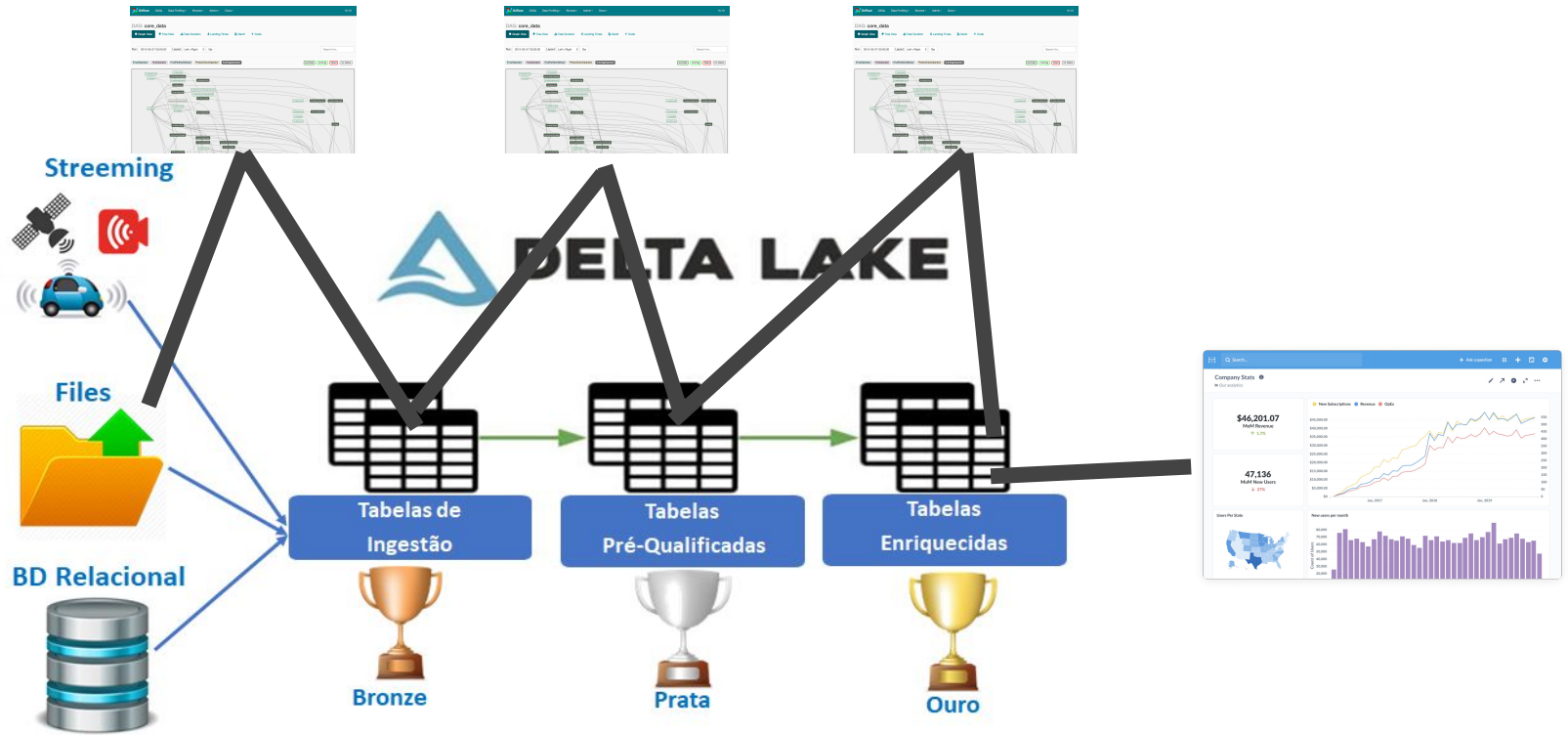

Streeming
DELTA LAKE
Files
BD Relacional
Tabelas de Ingestão
Tabelas Pré-Qualificadas
Tabelas Enriquecidas
Bronze
Prata
Ouro

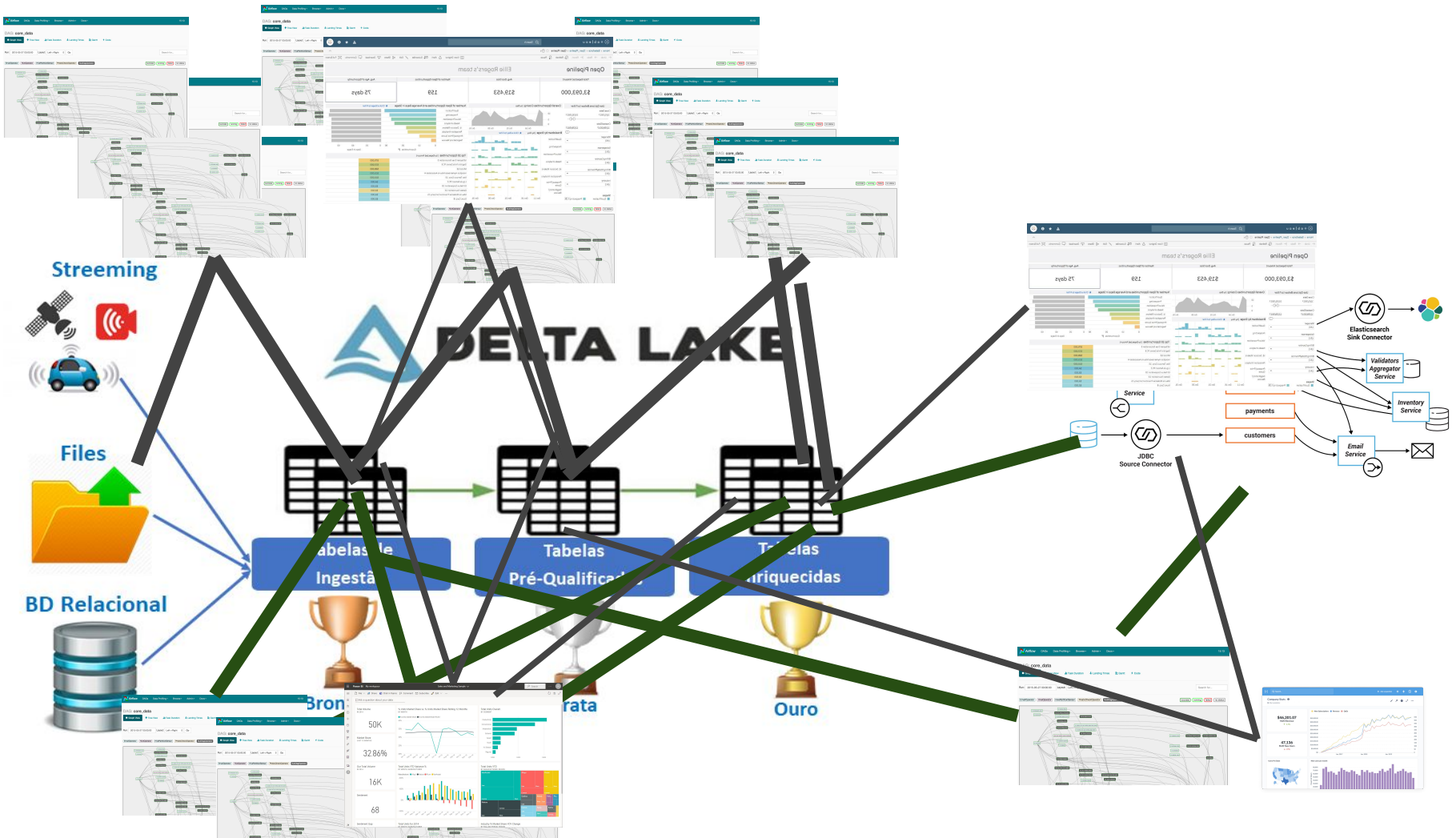

DATA
DATA LAKE ZONES
CONSUMER SYSTEMS
STREAMING
FILE DATA
RELATIONAL
TRANSIENT ZONE
RAW ZONE
TRUSTED ZONE
REFINED ZONE
Ingest, Tag, & Catalog Data
Apply Metadata, Protect Sensitive Attributes
Data quality & Validation
Enrich Data & Automate Workflows
Data Catalog
Data Prep Tools
Data Visualization
External Connectors

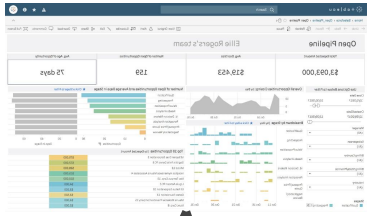

Data Sources
Staging Area
Warehouse
Data Marts
Users
Operational System
Operational System
Flat Files
Metadata
Summary data
Raw data
Purchasing
Sales
Inventory
Analytics
Reporting
Mining

Tier1	Tier2	Tier3
raw	trusted	consumption
raw	trusted	refined
Bronze	Silver	gold
Landing	Curated	production
Raw	Conformed	Modeled
land	Prepare	publish
....	...	.....




Streeming
Files
BD Relacional
DELTA LAKE
Tabelas de Ingestão
Tabelas Pré-Qualificadas
Tabelas Enriquecidas
Bronze
Prata
Ouro


Streeming
Files
BD Relacional
DELTA LAKE
Tabelas
Pré-Qualificadas
Ouro
Service
payments
customers
JDBC
Source Connector
Elasticsearch
Sink Connector
Validators
Aggregator
Service
Inventory
Service
Email
Service



Streeming

Files

BD Relacional

DELTA LAKE

Tabelas de Ingestão

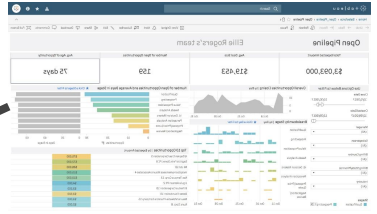
Bronze

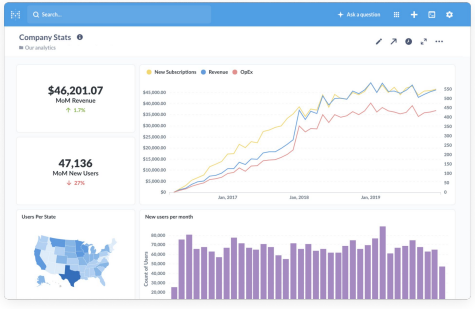
Tabelas Pré Qualificadas

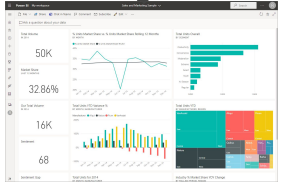
Prata

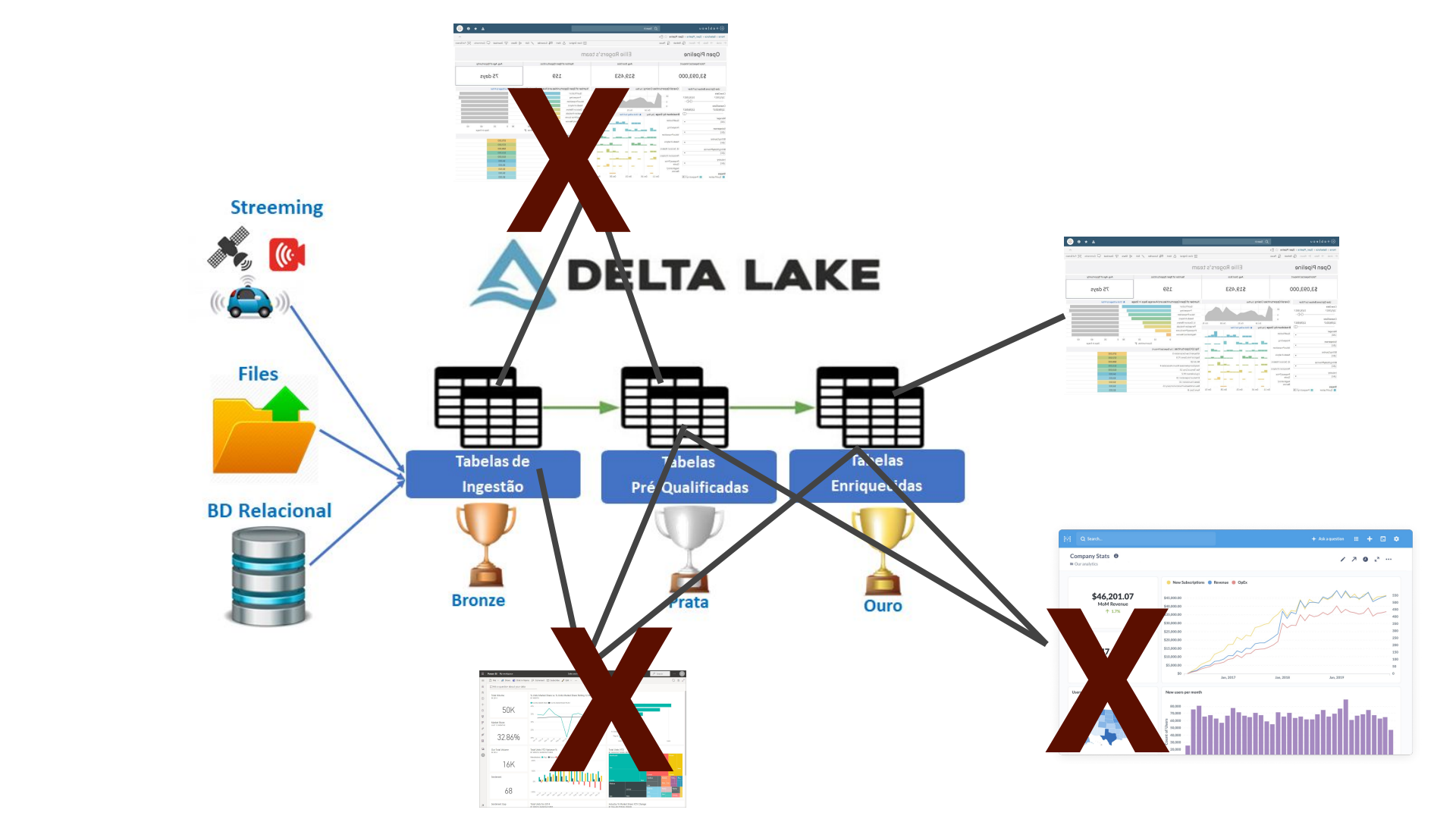
Tabelas Enriquecidas

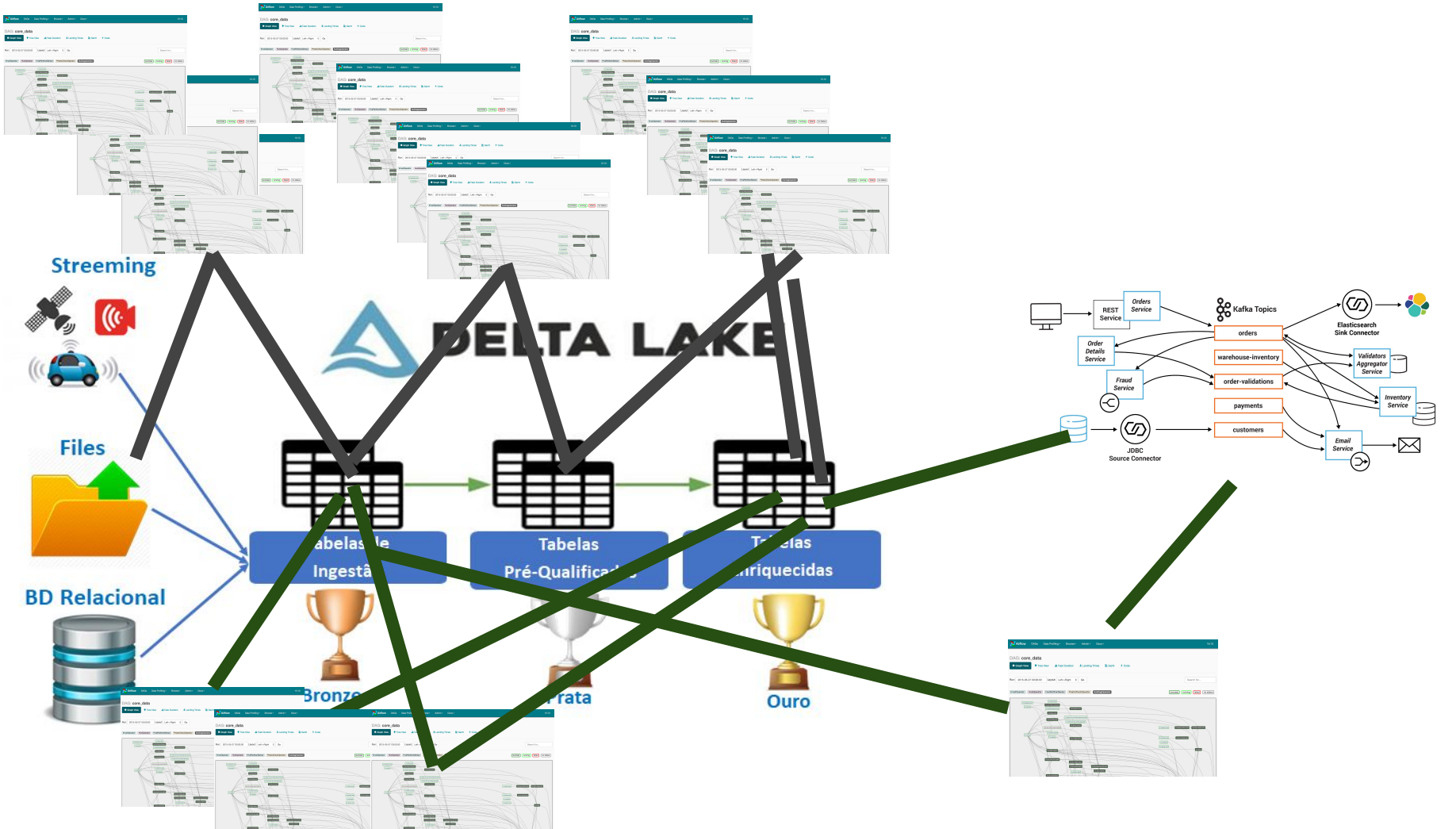
Ouro

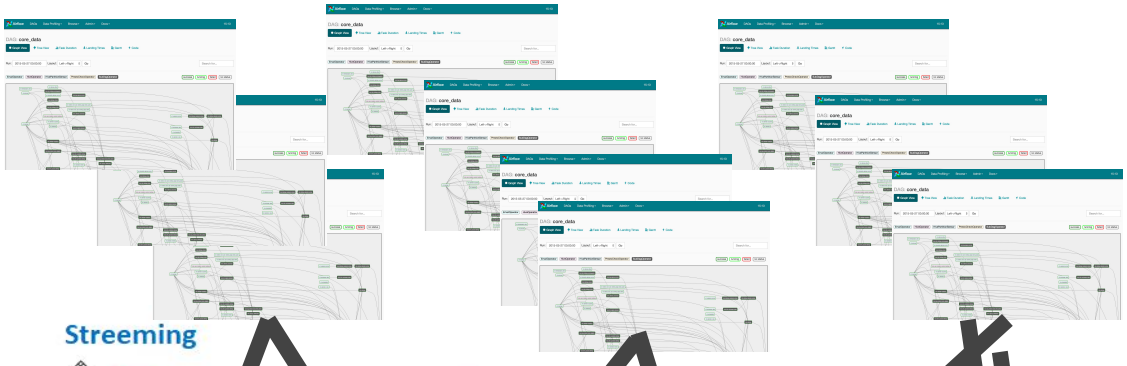







Streeming
Files
BD Relacional
DELTA LAKE
Tabelas de Ingestão
Tabelas Pré Qualificadas
Tabelas Enriquecidas
Bronze
Prata
Ouro
$46,201.07
MoM Revenue


Streeming
Files
BD Relacional
DELTA LAKE
Tabelas
Pré-Qualificadas
Bronze
Prata
Ouro
REST Service
Orders Service
Order Details Service
Fraud Service
Kafka Topics
orders
warehouse-inventory
order-validations
payments
customers
Elasticsearch Sink Connector
Validators Aggregator Service
Inventory Service
Email Service
JDBC Source Connector



Streeming

DELTA LAKE

Files

BD Relacional

Tabelas Ingestã

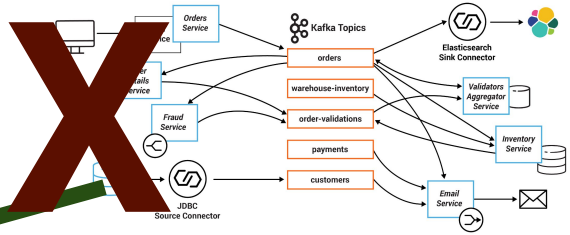
Tabelas Pré-Qualificad

Tabelas Enriquecidas

Bronze

Prata

Ouro



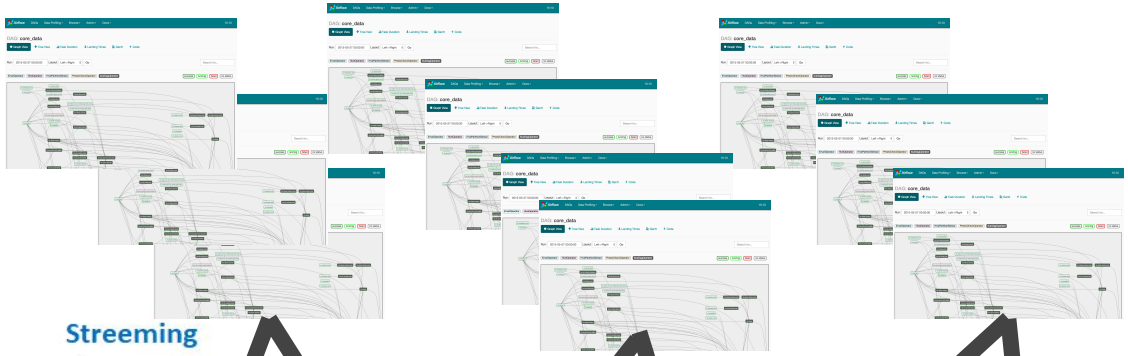
Disponibilidade(escala):

Backend = nanosegundo/milisegundo/segundo

Datalake = minuto/hora/dia

Acomplamento:

depende do Datalake, mas logicamente depende de outro backend



Streeming

Files

BD Relacional

DELTA LAKE

Tabelas de Ingestão

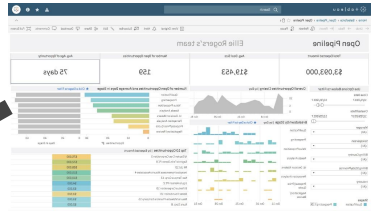
Tabelas Pré-Qualificadas

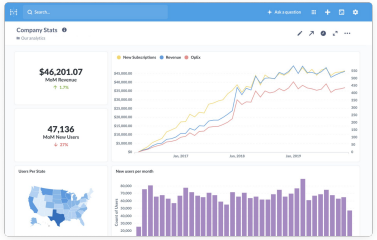
Tabelas Enriquecidas

Bronze

Prata

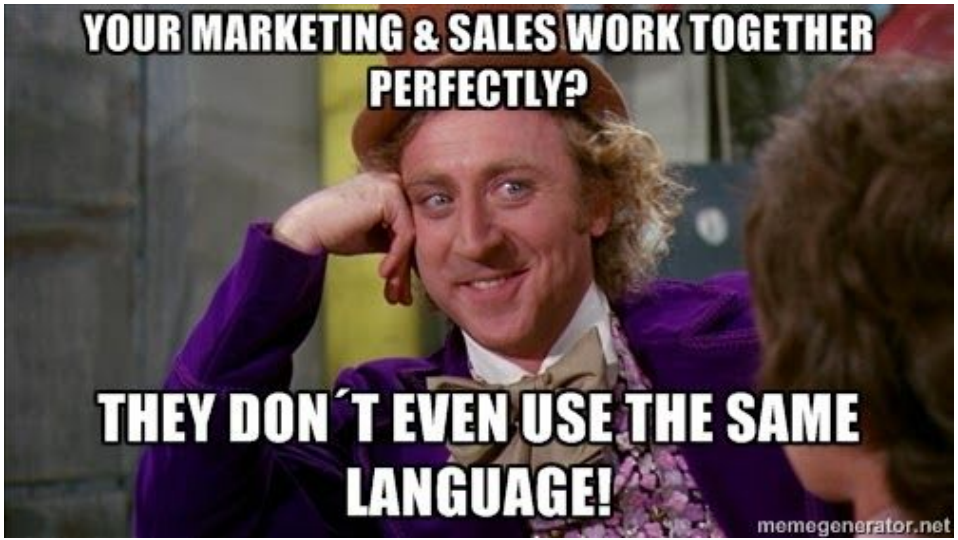
Ouro





- **A restrição(objetiva) é o que permite a longevidade da arquitetura.**
- **A restrição(objetiva) sustenta a flexibilidade.**
- **é sempre contextual.**
- **A arquitetura “correta” é a mais “simples” de mudar (Refatorar).**
- **complexidade não deixa de existir, apenas muda de lugar.**

Falácia: A lógica de negócios pode e deve ser centralizada.



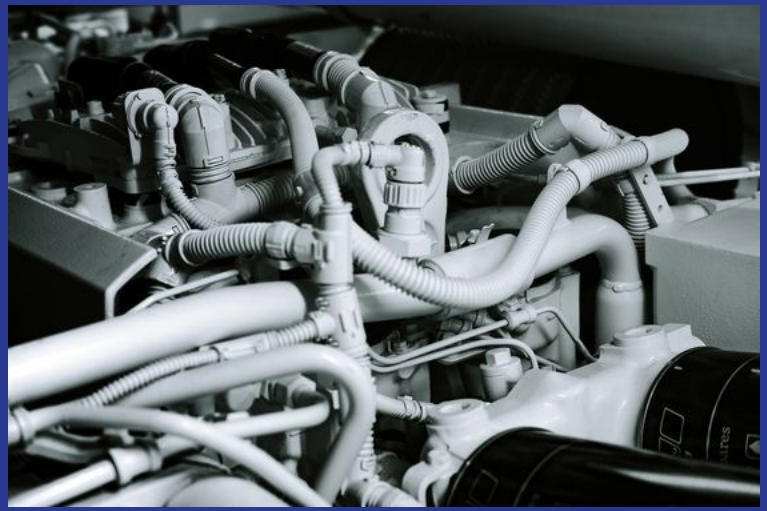

MARKETING
CUSTOMER SUCCESS
SALES
Sales Playbook
LUNCH TIME IN THE OFFICE

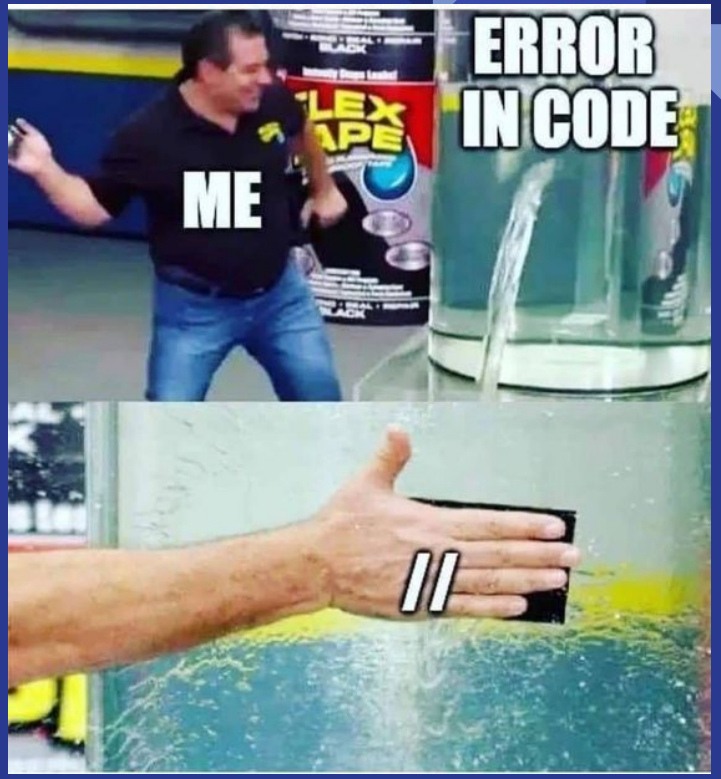

SALES & MARKETING
DEVELOPERS

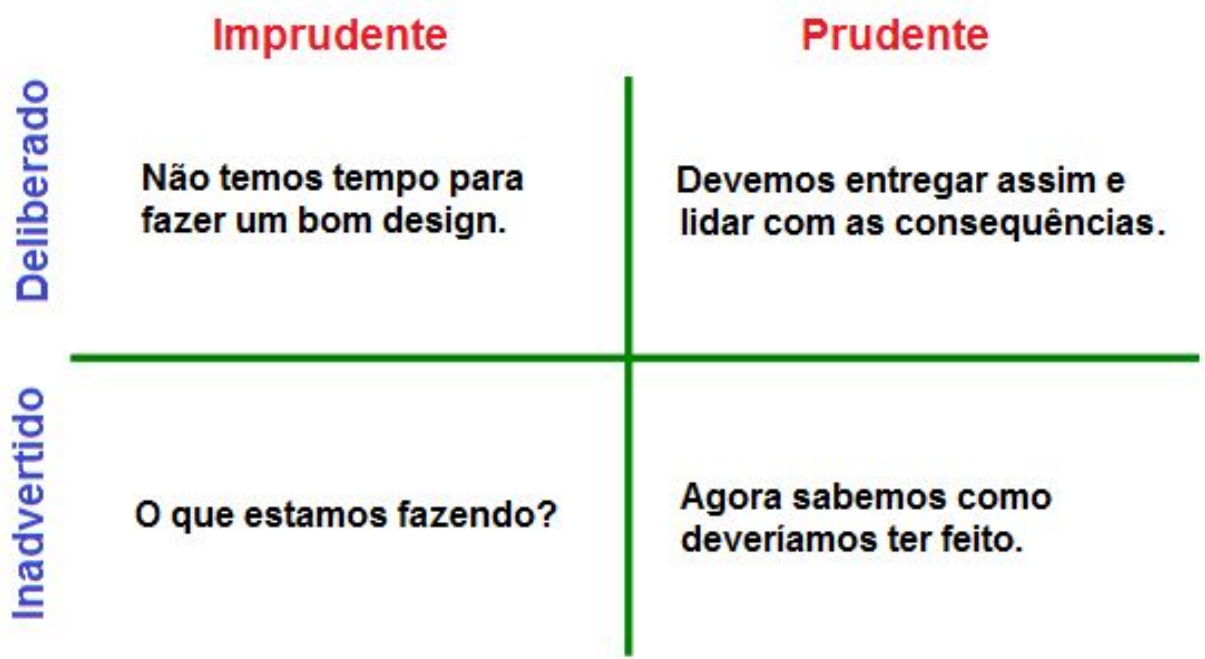

My job here is done
Hey, where is the documentation?

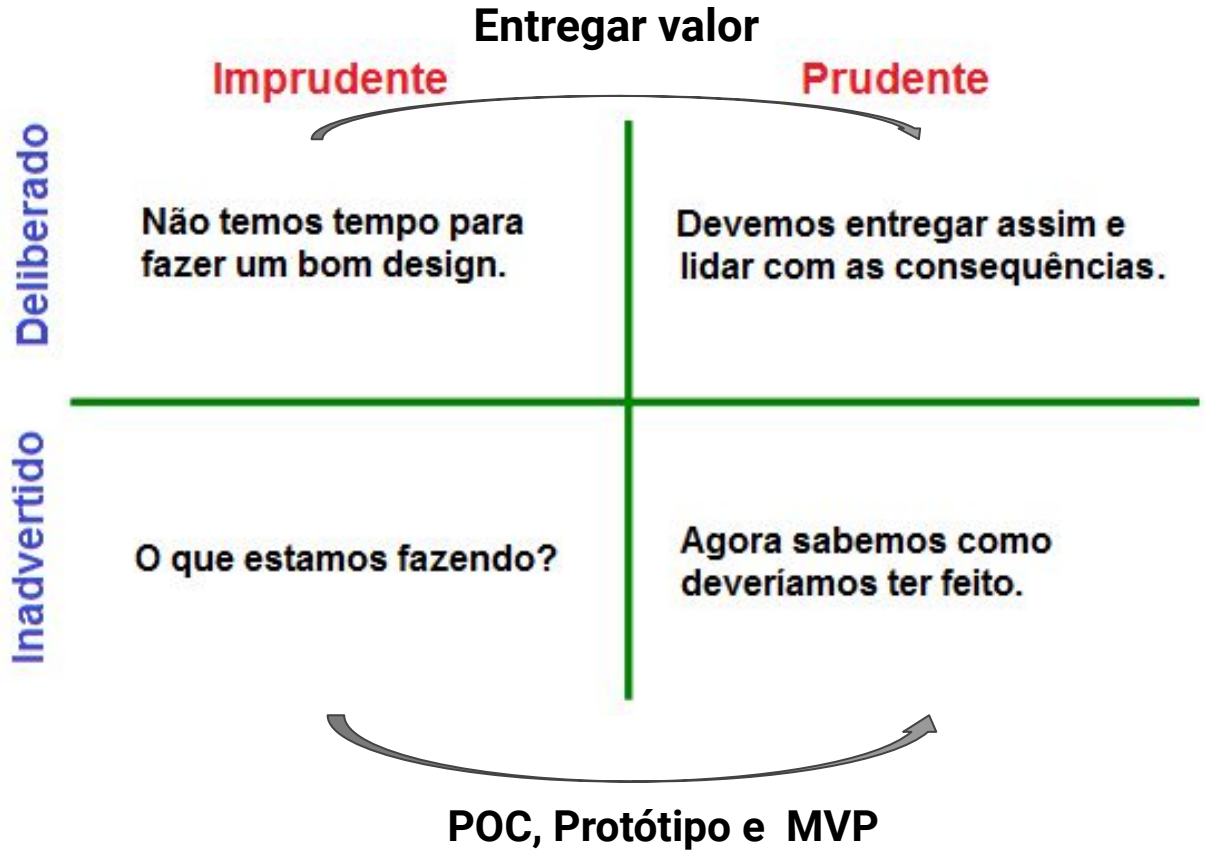
- **Isolar o contexto (DDD - Domain driven design, Data-mesh).**
- **Código limpo ajuda a distribuir o conhecimento no time.**
- **Documentação.**
- **Relacionamento ruim entre áreas.**

Mitigando a Dívida Técnica

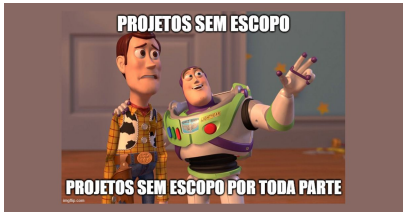


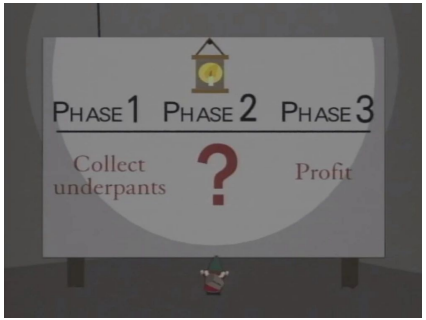



Imprudente
Prudente
Deliberado
Inadvertido
Não temos tempo para fazer um bom design.
Devemos entregar assim e lidar com as consequências.
O que estamos fazendo?
Agora sabemos como deveríamos ter feito.


Entregar valor
Imprudente
Prudente
Deliberado
Inadvertido
Não temos tempo para fazer um bom design.
Devemos entregar assim e lidar com as consequências.
O que estamos fazendo?
Agora sabemos como deveríamos ter feito.
POC, Protótipo e MVP

Inicie “corretamente”





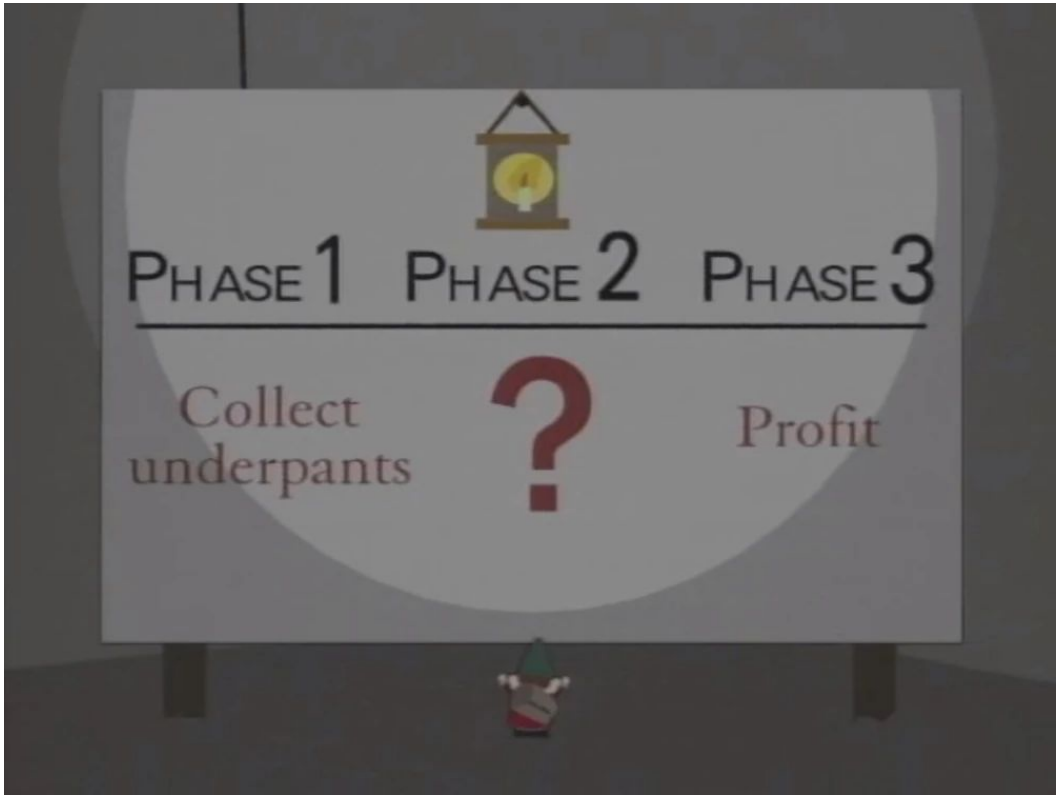
Desenvolvedor

QUALIDADE	RECURSOS HUMANOS	ESCOPO
AQUISIÇÕES		COMUNICAÇÕES
CUSTO	RISCOS	TEMPO

Definido pelo cliente/negócio/analista (mas não sabe como, quer apenas o resultado)

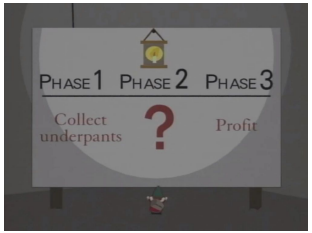
Já está definido.

Já está definido.


PHASE 1 PHASE 2 PHASE 3
Collect underpants
?
Profit

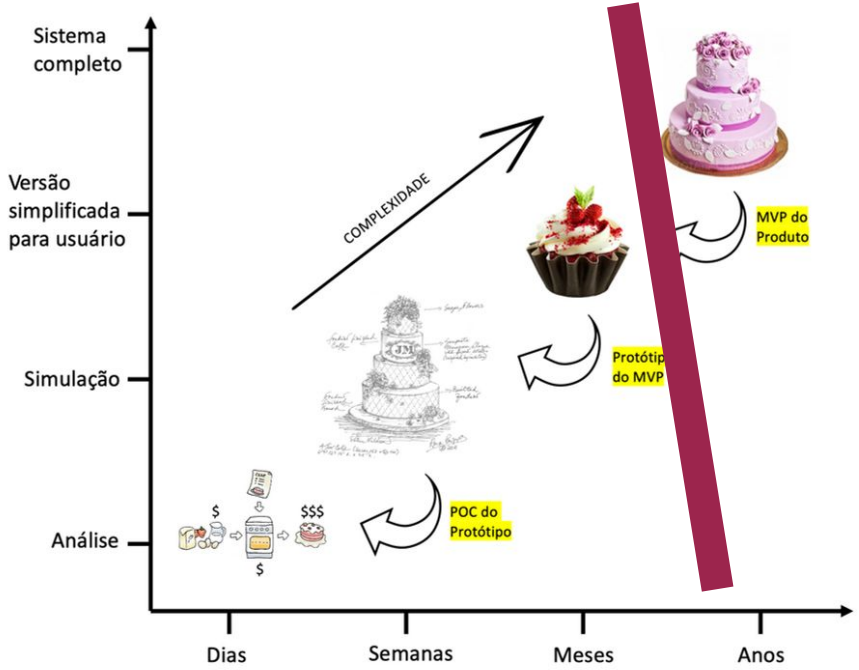

AGILE - SCRUM
I CAN'T GIVE YOU ALL OF THESE FEATURES IN THE FIRST VERSION.
AND EACH FEATURE NEEDS TO HAVE WHAT WE CALL A "USER STORY."
OKAY, HERE'S A STORY: YOU GIVE ME ALL OF MY FEATURES OR I'LL RUIN YOUR LIFE.
www.dilbert.com scottadams@aol.com
© 2002 United Feature Syndicate, Inc.


HAPTIC FEEDBACK
HAPTIC FEEDBACK EVERYWHERE

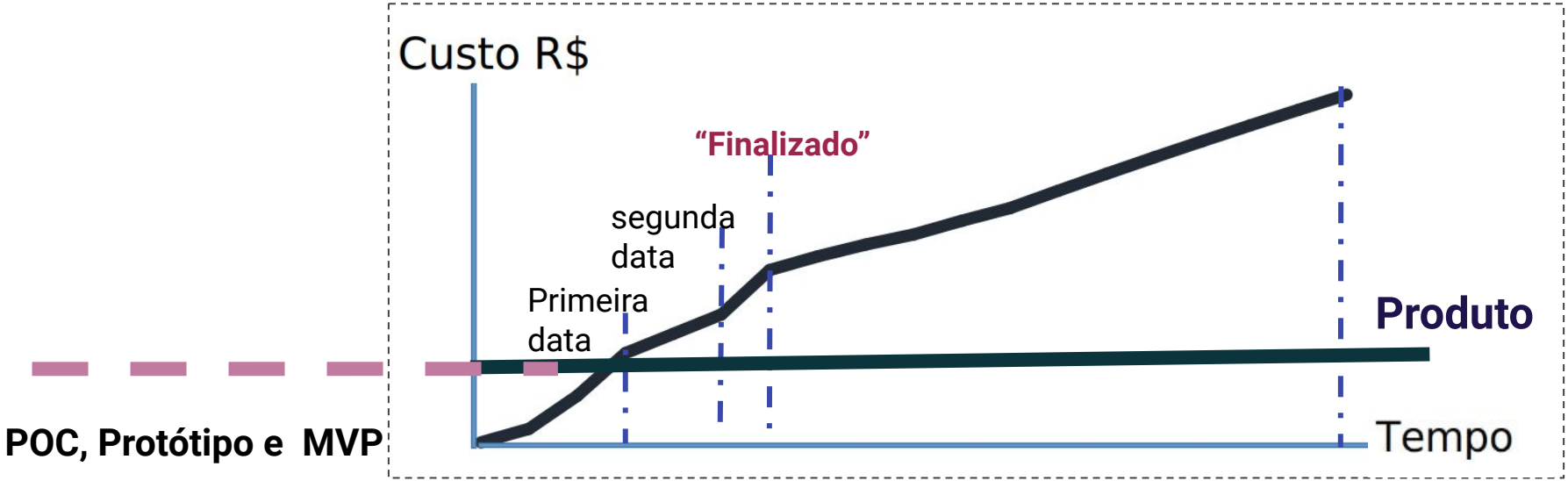


elicitação

Ação ou efeito de elicitar, de fazer sair, de expulsar; eliciação.
Obtenção de informações detalhadas sobre o que se pretende fazer.



https://www.bartosiaki.com.br/post/qual-a-diferen%C3%A7a-entre-poc-prot%C3%B3tipo-e-mvp

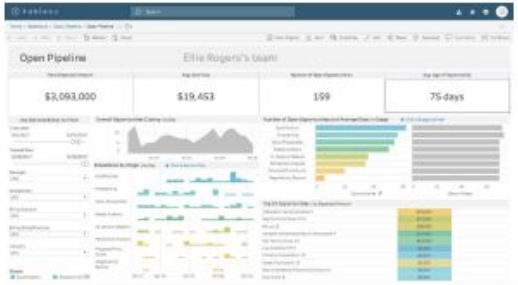

Custo R$
"Finalizado"
segunda data
Primeira data
Produto
Tempo
POC, Protótipo e MVP

Dividir o escopo e eliminar o que não é necessário para entregar valor.

bronze
Table1
Table2
Table3
Table4
Table5
Table6
Table7
Table8
Table9
Table10

silver
Table1
Table2
Table3
Table4
Table5
Table6
Table7
Table8
Table9
Table10

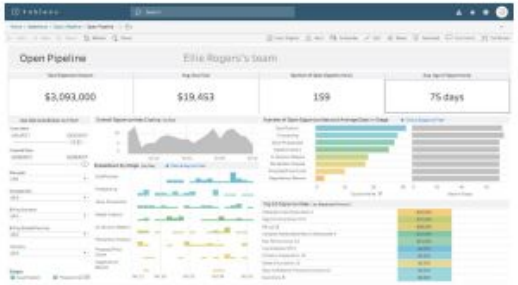
gold
TableX
TableX
TableX
TableX
TableX
TableX
TableX
TableX
TableX
TableX



bronze
Table1
Table2
Table3
Table4
Table5
Table6
Table7
Table8
Table9
Table10

silver
Table1
Table2
Table3
Table4
Table5
Table6
Table7
Table8
Table9
Table10

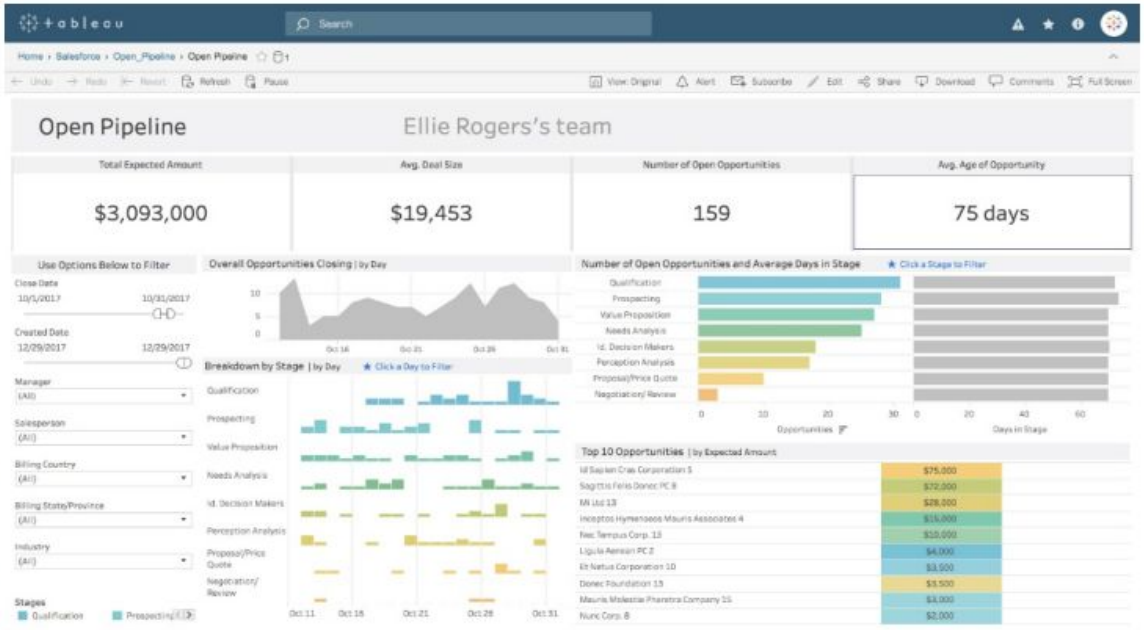
gold
TableX
TableY



gold

TableY
ColY
ColY
ColY
ColY
ColY
ColY
ColY
ColY
ColY
ColY

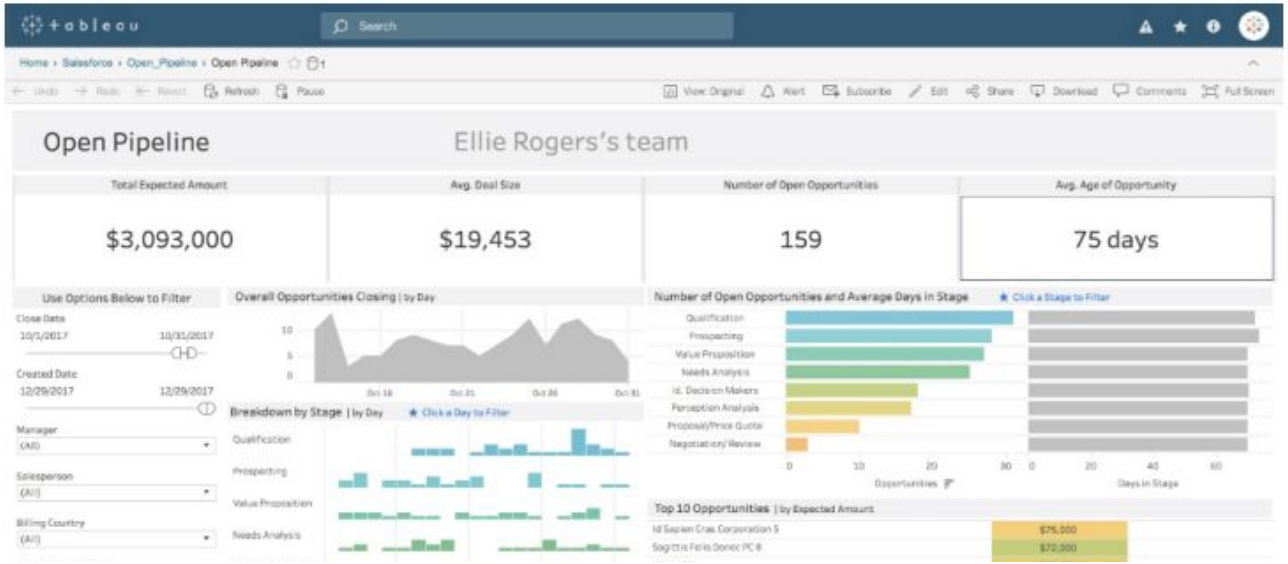
TableX
ColX
ColX
ColX
ColX
ColX
ColX
ColX
ColX
ColX
ColX



gold

TableY
ColY
ColY
ColY

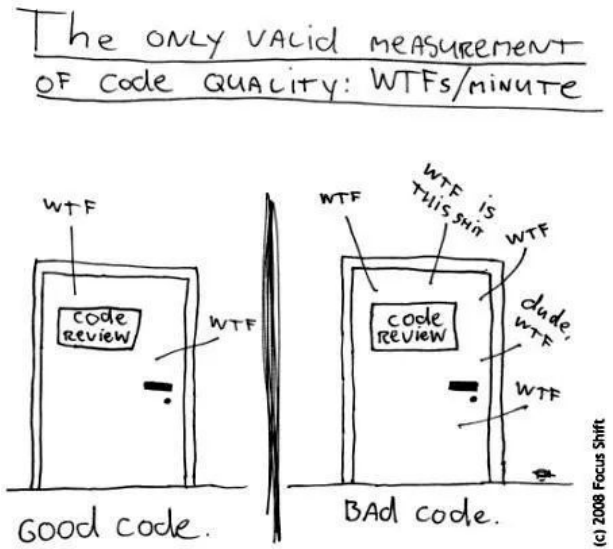
TableX
ColX
ColX



	Tabelas	colunas	colunas (apenas o necessário)
tabelas de sistemas	10	100	
tabelas na gold	2	20	5
	20%	20%	5%
	80% de economia	80% de economia	95% de economia + qualidade

** Profilling (Perfil)

Escreva e refatore



Qualquer alteração ou decisão técnica deve ser aplicada como código.

- Infrastructure as Code
- **Data Quality as code**
- Report as code (mais complicado)
- Pipeline as code

Quando refatorar: Feriu a qualidade do dado

Profilling (Perfil):

- máquina de estado (colunas de “status”)
- ids, joins, merge (null, duplicados)
- limites (venda, compra, desconto..)

Single Column

- Cardinalities
- Patterns & Data types
- Value distributions
- Domain classification

Multi-Column

- Correlations
- Association rules
- Clustering
- Outliers
- Summaries & sketches

Cross-DB dependencies

- Unique column combinations
- Inclusion dependencies
- Functional dependencies

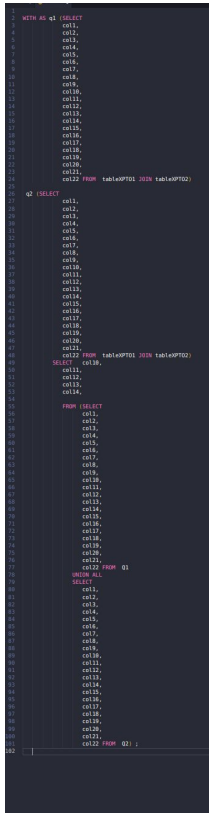
Great Expectations

Reference: Ziawasch Abedjan, Lukasz Golab, and Felix Naumann. 2015. Profiling relational data: a survey. The VLDB Journal 24, 4 (August 2015), 557-581.

Quando refatorar: código hadouken

```
function register()
{
    if (!empty($_POST)) {
        $msg = '';
        if ($_POST['user_name']) {
            if ($_POST['user_password_new']) {
                if ($_POST['user_password_new'] === $_POST['user_password_repeat']) {
                    if (strlen($_POST['user_password_new']) > 5) {
                        if (strlen($_POST['user_name']) < 65 && strlen($_POST['user_name']) > 1) {
                            if (preg_match('/^[a-z\d]{2,64}$/i', $_POST['user_name'])) {
                                $user = read_user($_POST['user_name']);
                                if (!isset($user['user_name'])) {
                                    if ($_POST['user_email']) {
                                        if (strlen($_POST['user_email']) < 65) {
                                            if (filter_var($_POST['user_email'], FILTER_VALIDATE_EMAIL)) {
                                                create_user();
                                                $_SESSION['msg'] = 'You are now registered so please login';
                                                header('Location: ' . $_SERVER['PHP_SELF']);
                                                exit();
                                            } else $msg = 'You must provide a valid email address';
                                        } else $msg = 'Email must be less than 64 characters';
                                    } else $msg = 'Email cannot be empty';
                                } else $msg = 'Username already exists';
                            } else $msg = 'Username must be only a-z, A-Z, 0-9';
                        } else $msg = 'Username must be between 2 and 64 characters';
                    } else $msg = 'Password must be at least 6 characters';
                } else $msg = 'Passwords do not match';
            } else $msg = 'Empty Password';
        } else $msg = 'Empty Username';
        $_SESSION['msg'] = $msg;
    }
    return register_form();
}
```

Quando refatorar: Código acima de um limite estipulado (50, 100 linhas ?)



Inicie “corretamente”:

Usar poc e/ou protótipos para entender e/ou aprender sobre o contexto, descarta assim que possível - **Feedback, deve entregar valor, elicitação.**

“Fatiou passou”:

Levar apenas o necessário (colunas) para a camada mais exposta ao cliente (relatórios/dashboards) – **refatoração/backfill**

Escreva e refatore:

Todos os objetos da arquitetura devem estar representados em código, código limpo –**refatore..refatore..refatore.**

Dívida deve ser controlada.

- Dívida deve ser adquirida conscientemente.
- Dívida deve ser **paga**.

Falácias:

- O sistema está finalizado.
- A arquitetura é a mesma em todos os lugares.
- A lógica de negócios pode e deve ser centralizada.

Mitigando a Dívida Técnica:

- **Inicie “corretamente”**
- **“Fatiou passou”**
- **Escreva e refatore**

Dívida deve ser controlada.

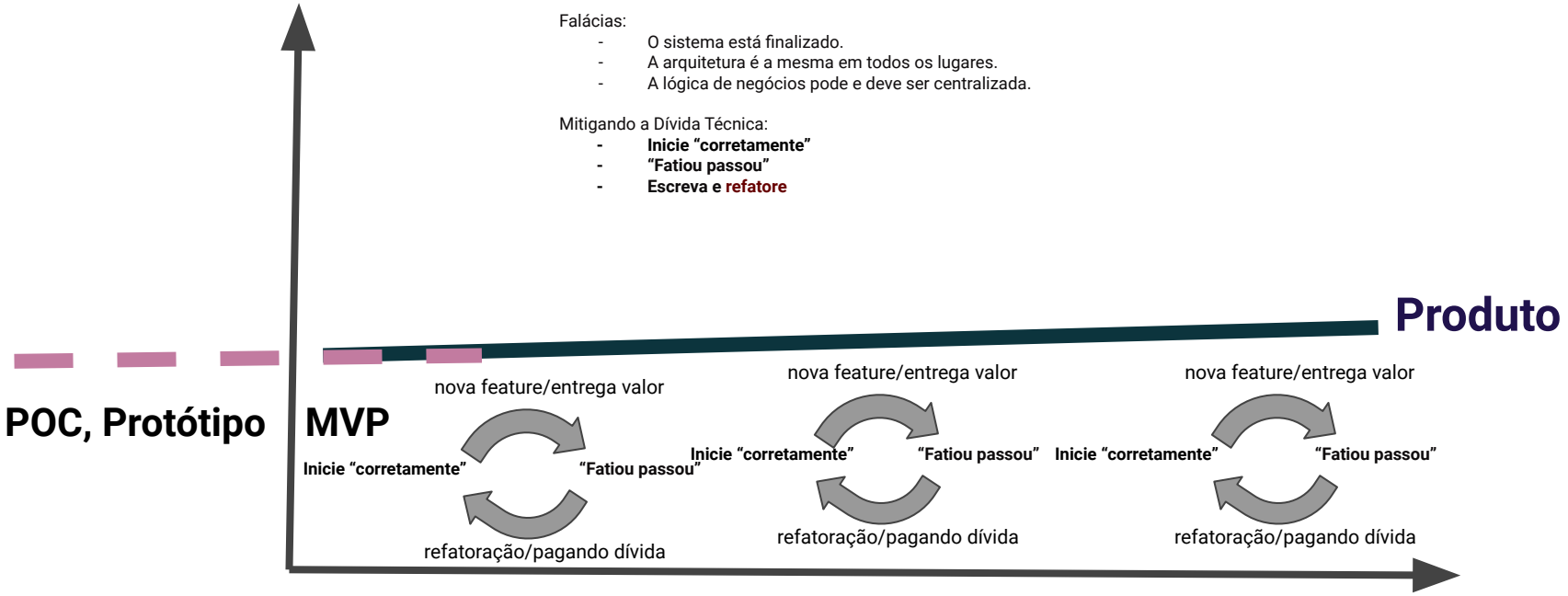
- Dívida deve ser adquirida conscientemente.
- Dívida deve ser **paga**.

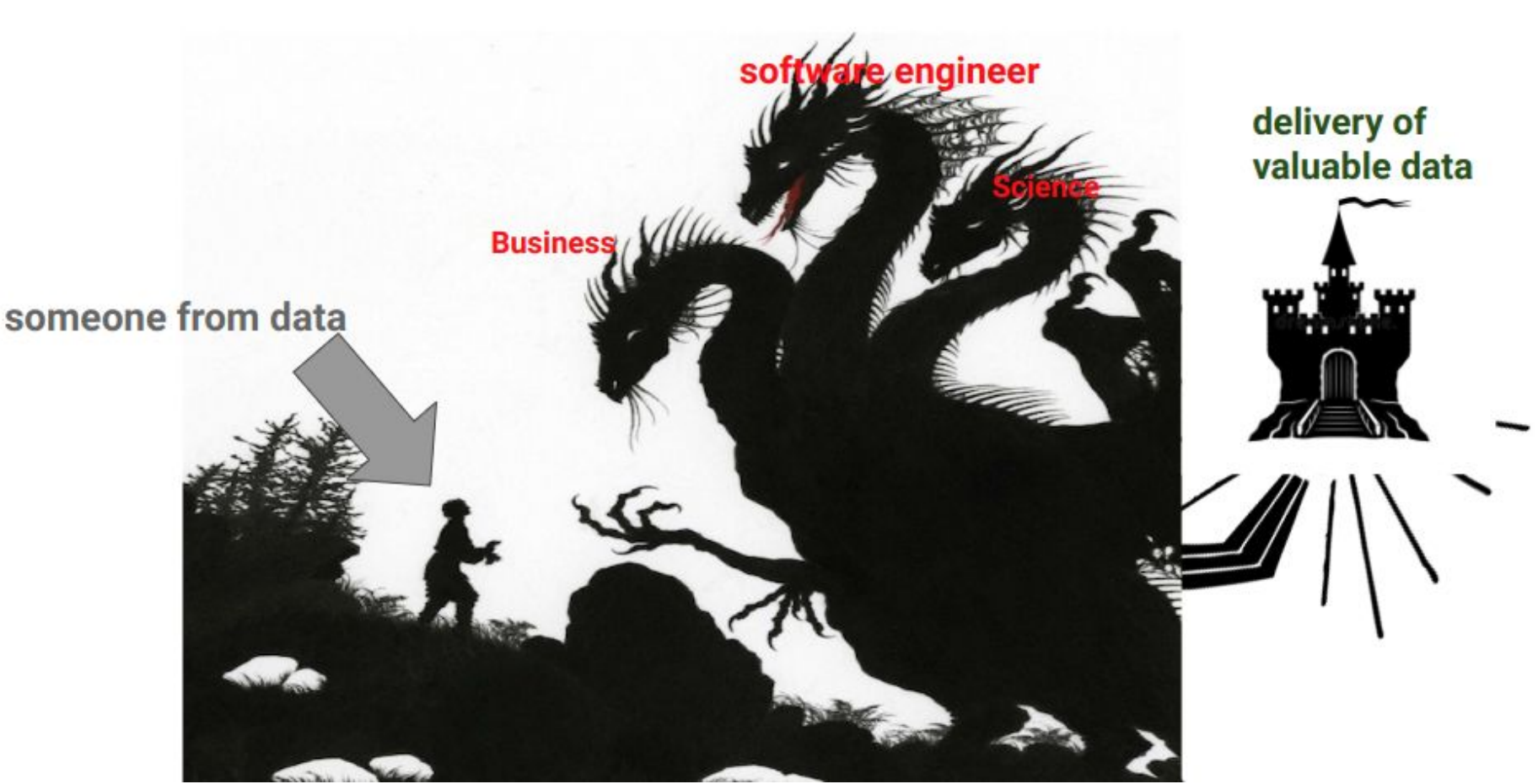
Falácias:

- O sistema está finalizado.
- A arquitetura é a mesma em todos os lugares.
- A lógica de negócios pode e deve ser centralizada.

Mitigando a Dívida Técnica:

- **Inicie “corretamente”**
- **“Fatiou passou”**
- **Escreva e refatore**




software engineer
Science
Business
someone from data
delivery of
valuable data

Dúvidas ?