



The ITIL® Foundation Examination

Sample Paper A, version 5.1

Multiple Choice

ANSWERS AND RATIONALE

© AXELOS Limited 2012

All rights reserved.

Reproduction of this material requires the permission of AXELOS Limited.

The swirl logo™ is a trade mark of AXELOS Limited

ITIL® is a registered trade mark of AXELOS Limited

Answer Key and Rationale:

Q	A	Syllabus Ref	Book Ref	Rationale
1	B	05-51	ST 4.2.4.3	A change request is a formal communication seeking an alteration to one or more configuration items (CIs). Services, SLAs and computers are examples of CIs. A business strategy is not normally a CI and would be out of scope for change management.
2	A	02-09	SO 1.1.1	Each of these are a purpose of service operation except for option A, undertaking testing to ensure services are designed to meet business needs. Option A is part of service transition.
3	B	06-02	SO 6.5.1.1	IT operations control oversees the execution and monitoring of the operational activities and events in the IT infrastructure.
4	C	05-63	ST 4.3.1	Part of SACM's purpose is to maintain accurate information about assets, including the relationship between assets.
5	A	07-02	SD 3.7.4.1	RACI is a responsibility model used by ITIL to help define roles and responsibilities.
6	A	03-12	SD 4.3.4	A is the OLA, B is the definition of an SLA, C doesn't correspond to an ITIL definition, D involves a third party and is a contract.
7	D	05-42	SD 4.4.1	A is a supporting element of availability management, not a main purpose. B relates to service level management. Availability management does not offer guarantees as identified in C. D is the main purpose of availability management: - "to ensure that the level of availability delivered in all IT services meets the agreed availability needs... of the business."
8	C	02-07	ST 1.1.1	All three are in scope for service transition as all three involve major change.
9	A	02-02	SS 1.2	Service optimization is the correct answer
10	D	03-18	ST 4.3.4.3	A: a CMS can contain corporate data about users / customers such as location or department. B and C: there may be more than one CMDB but they will be part of a single CMS. D is correct as a CMS still helps to control and report on the infrastructure when IT services are outsourced.
11	A	05-45	SD 4.5.4.3	Book answer...business, service and component capacity management are the three sub-processes
12	D	03-19	ST 4.3.4.4	The DML contains master copies of all controlled software in an organization ... "along with licence documents or information". The change schedule would not be included.
13	B	05-31	SD 4.3.1	Service level management has responsibility for negotiating and agreeing OLAs.
14	D	07-01	SD 6.3.2	Book answer. A process owner should ensure process documentation is current and available.
15	A	05-61	ST 4.4.1	The two correct answers (1 and 2) are included in release and deployment objectives. Option 3 is addressed by change management.
16	D	01-10	SS 2.2.2	Measurability, delivery of specific results, and delivery of results to a customer or stakeholder are all characteristics of a process.
17	B	01-02	SD 1.4	Option 4 is incorrect, ITIL is not a standard: ISO/IEC 20000 would be an example of a standard. ITIL is vendor-neutral, non-prescriptive, and provides a best practice framework.
18	C	05-43	SD 4.7.4.1	In most cases the policies should be widely available to all customers and users and referenced in SLAs, OLAs and UCs.
19	D	03-14	SD App A	All of the elements identified are included in the service design package passed to service transition.
20	C	08-02	SS 7.1	1 would be used to support a DML. 2 helps change management. 3 is a release and deployment tool. 4 can help with testing and validation. They all support service transition.

Q	A	Syllabus Ref	Book Ref	Rationale
21	C	05-72	SO 4.4.2 and 4.4.6.4	Book answer. They are both valid roles for problem management.
22	A	05-82	SO 4.3.1	Request fulfilment is the process responsible for dealing with service requests from the users. ‘All requests’ (B) is too wide a scope for the process. Change management looks after change requests (C). Service level management is responsible for D.
23	A	04-02	SS 3.2.3	D is incorrect; customer preferences drive value perception. C is incorrect; delivering on customer outcomes is vital. B is incorrect; the value of a service can be financial but other factors are also relevant. A is correct; customer perception is a vital element in defining how much a customer values a service.
24	D	01-04	SS 3.2.1.2	D is the correct response. Both internal and external customers should be provided with the agreed level of service, and with the same level of customer service.
25	D	01-03	SS 2.1.1	A service is a means of delivering value to customers. IT needs capabilities to deliver services. Cost and risk are what IT helps to manage.
26	C	05-31	SD 4.3.5.6	C is correct: monitoring the SLAs and performance against them is a vital part of the service level management process. A - designing the CMS is a service asset and configuration management activity. B – technology metrics are likely to be created within capacity management or other design processes. D – training the service desk is a service desk role.
27	A	05-81	SO 4.1.1	A - the ability to detect events, make sense of them and determine the appropriate control action is provided by event management. B includes some incident management responsibilities. C is a technical management task. D is likely to be shared between availability management and service level management.
28	D	05-41	SD 4.2.1	The service catalogue should contain details of all operational services.
29	D	03-01	SS 2.1.6	A is part of the definition of utility. B is unrealistic. C could be feasible as a warranty statement from another industry but is not the definition of warranty as used by ITIL. D is a good summary of warranty as defined by ITIL.
30	A	04-09	CSI 3.1	The improvement approach begins with embracing the vision by understanding the high-level business objectives.
31	C	05-71	SO 4.2.4.2	Incident models are designed to provide reusable steps that can be used to restore service after known incident types.
32	A	05-71	SO 4.2.5	The correct order is given in the diagram in the incident management process, and in the subsections of 4.2.5.
33	A	04-04	SD 3.1.1	Measurements and metrics should be included in the design for a new or changed service.
34	D	05-43 05-46	SD 4.7.2 SD 4.6.5.2	IT service continuity management carries out risk assessment as part of defining the requirements and strategy. Information security also needs to analyse security risks before taking action to mitigate them. Service catalogue management does not carry out these assessments.
35	C	04-10	CSI 5.5	Personnel metrics are not one of the three types of metrics described in CSI
36	B	03-16	ST 4.7.4.3	A is the wrong way round. C is incorrect as the SKMS contains more information than the CMS. D is incorrect as the CMS is part of the SKMS.
37	C	05-51	ST 4.2.5.11	The emergency change advisory board (ECAB) provides assistance in the authorization of emergency changes.

Q	A	Syllabus Ref	Book Ref	Rationale
38	B	06-01	SO 6.3	The service desk should be the single point of contact for IT users on a day-by-day basis. The service desk manager may also be the incident management process owner but would not normally be the owner of problem management.
39	D	04-03	SD 3.1.5	Book answer: people, processes, products (services, technology and tools) and partners (suppliers, manufacturers and vendors).
40	B	05-72	SO 4.4.5.6	A is incorrect; the problem record must remain open as it hasn't yet been resolved. B is correct to document the workaround on the problem record, not on each Incident record [C], nor on an RFC [D].

© AXELOS Limited 2012
All rights reserved.

Reproduction of this material requires the permission of AXELOS Limited.
The swirl logo™ is a trade mark of AXELOS Limited
ITIL® is a registered trade mark of AXELOS Limited

Arquitetura Neoway

Da captura à disponibilização dos dados





Kamila Hinckel

Developer at Neoway
Data Platform Team



Ricardo Longa

Developer at Neoway
Core Team

Neoway



Big Data



**Sede em
Floripa**

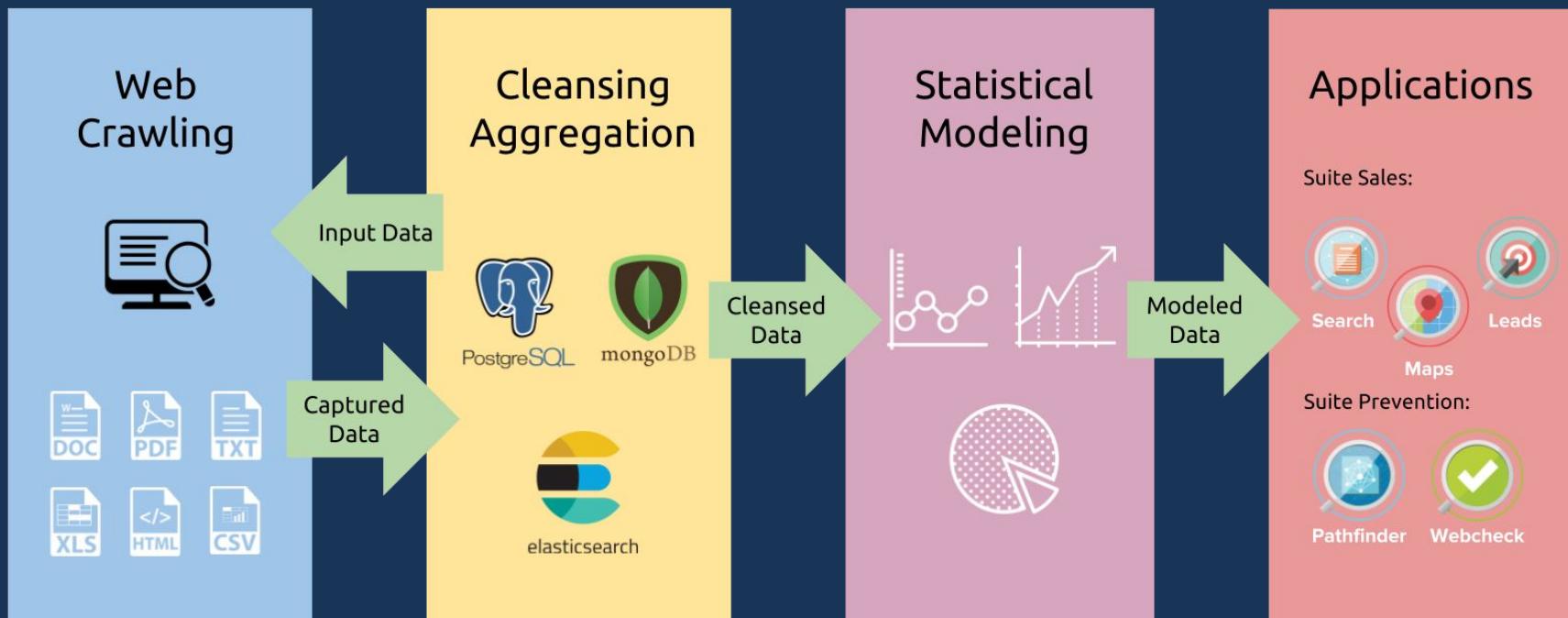


**GPTW:
6^a melhor empresa
para se trabalhar**



**Times
independentes**

O que a Neoway faz?



+400
Clientes

+3000
Bancos de dados

PaaS

+30
Parceiros de negócios

+250
Colaboradores

DADOS

- +35 milhões de **empresas**
- +194 milhões de **pessoas** no Brasil
- +180 milhões de **processos judiciais**
- +45 milhões de **companies USA**

*Em 2018, mais da metade das **grandes organizações mundiais** concorrerá utilizando análises avançadas e algoritmos proprietários, provocando a **ruptura de indústrias inteiras**.**

Como processamos
mais de **6 milhões** de dados
por dia?

Arquitetura



Cliente

Receita Federal do Brasil

www.receita.fazenda.gov.br/pessoajuridica/cnp

Fazenda
Ministério da Fazenda

Emissão de Comprovante de Inscrição e de Situação Cadastral

Contribuinte,

Esta página tem como objetivo permitir a emissão do Comprovante de Inscrição e de Situação Cadastral de Pessoa Jurídica pela Internet em consonância com a [Instrução Normativa RFB nº 1.470, de 30 de maio de 2014.](#)

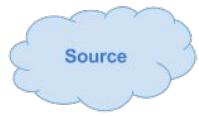
Digite o número de CNPJ da empresa e clique em "Consultar".

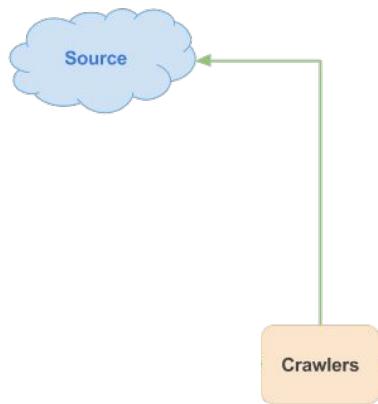
CNPJ :

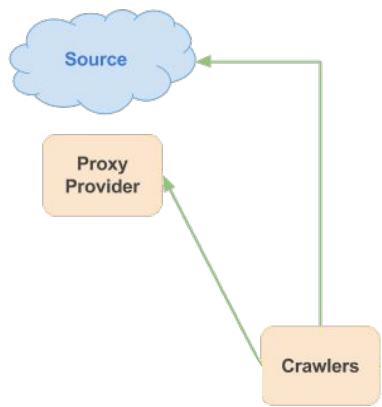
Digite os caracteres ao lado: 



Se os caracteres da imagem estiverem ilegíveis, [gerar outra imagem](#)







Cliente

Receita Federal do Brasil

www.receita.fazenda.gov.br/pessoajuridica/cnp

Fazenda
Ministério da Fazenda

Emissão de Comprovante de Inscrição e de Situação Cadastral

Contribuinte,

Esta página tem como objetivo permitir a emissão do Comprovante de Inscrição e de Situação Cadastral de Pessoa Jurídica pela Internet em consonância com a [Instrução Normativa RFB nº 1.470, de 30 de maio de 2014.](#)

Digite o número de CNPJ da empresa e clique em "Consultar".

CNPJ :

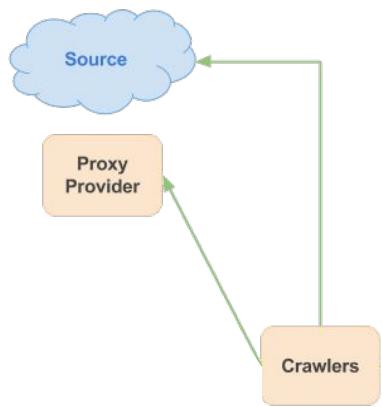
Digite os caracteres ao lado: 

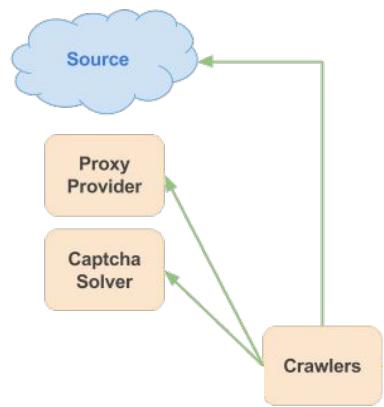


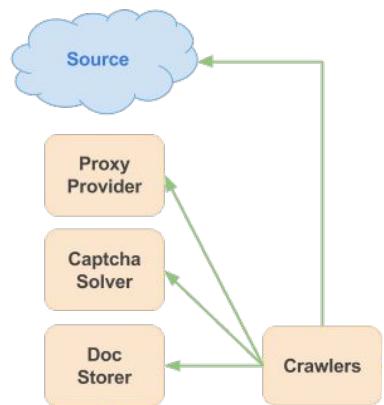


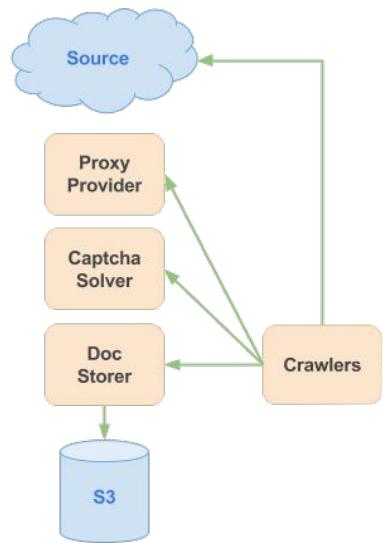
BRACON

Se os caracteres da imagem estiverem ilegíveis, [gerar outra imagem](#)



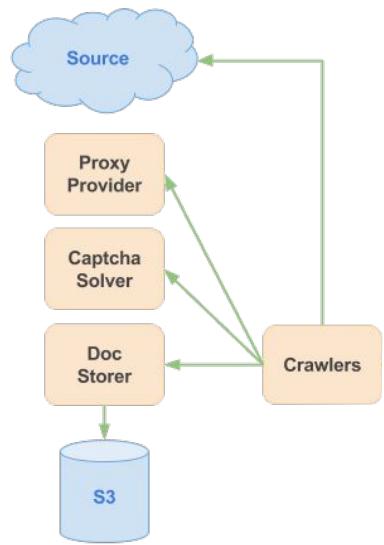


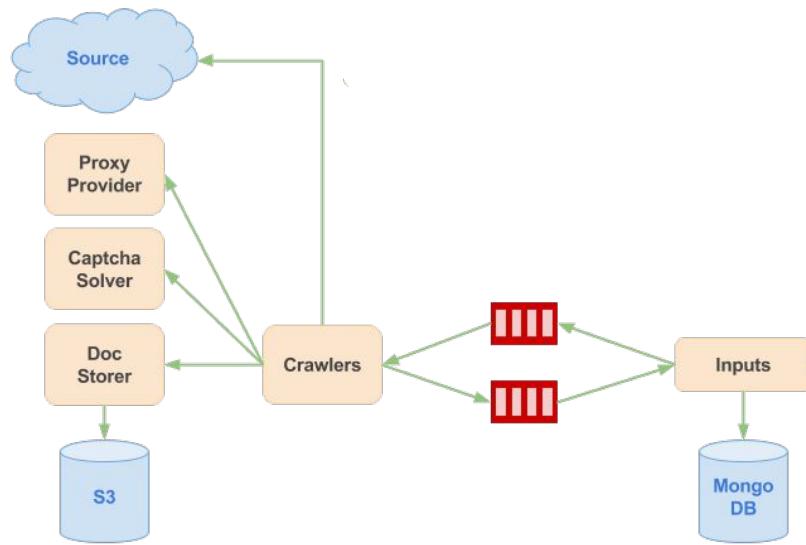






200+ Crawlers!

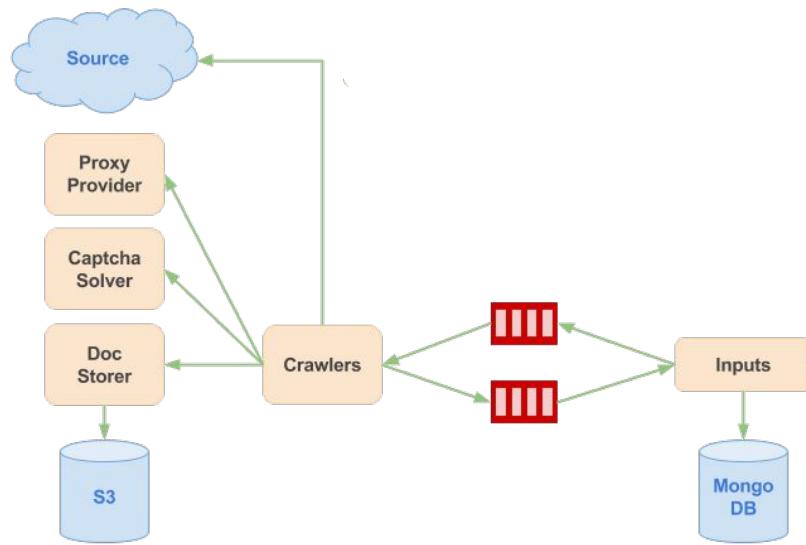


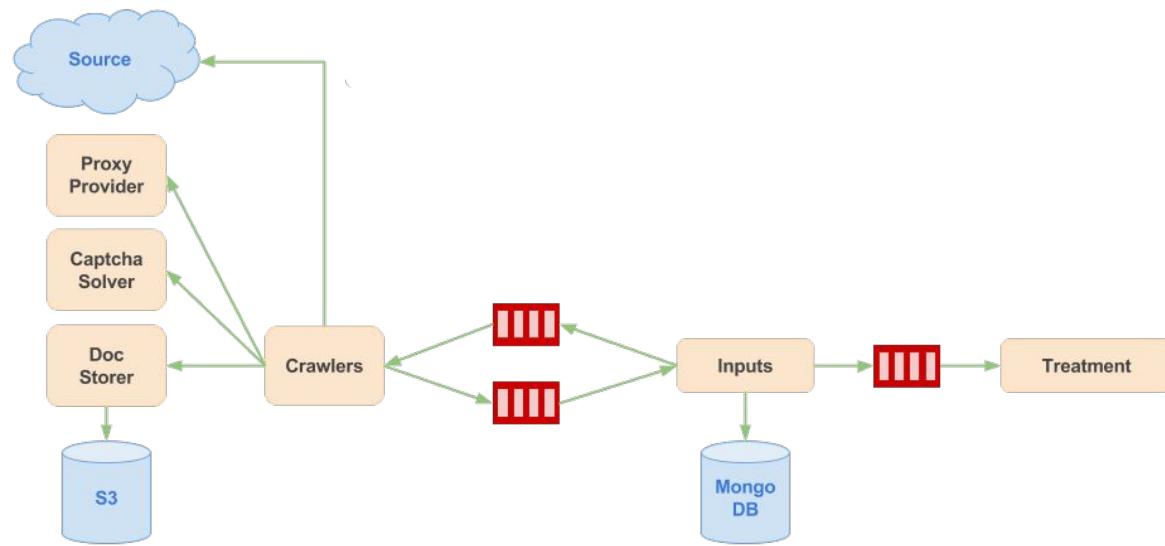


Inputs

Exemplos:

- CNPJ -> 05.337.875/0001-05
{ "cnpj": "05337875000105" }
- Sequencial -> 061256-01
{ "protocolo": "06125601" }
- CNPJ + Razão Social:
{ "cnpj": "05337875000105", "razaoSocial": "NEOWAY" }

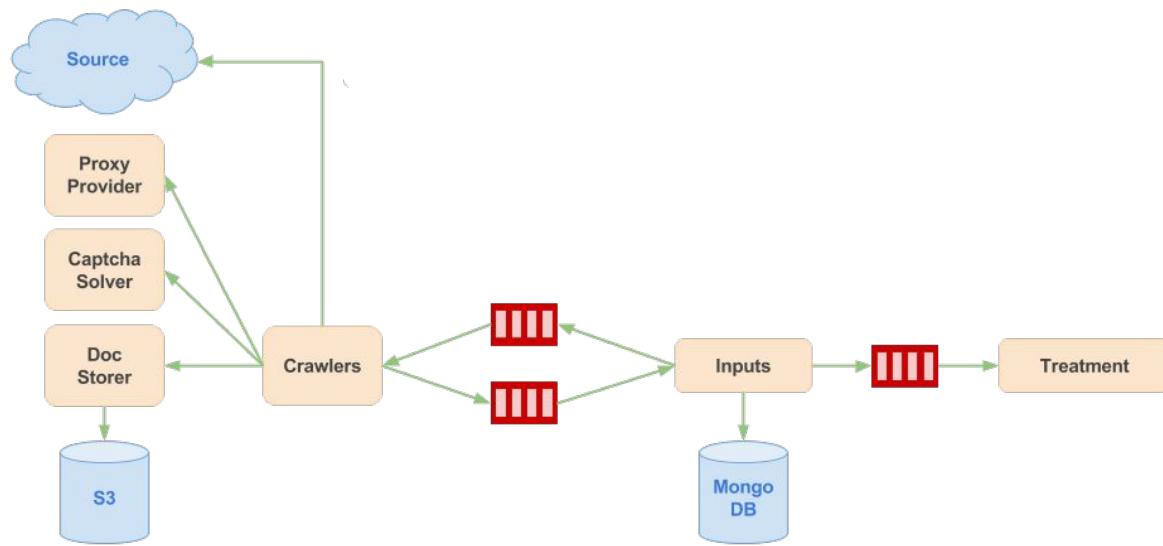


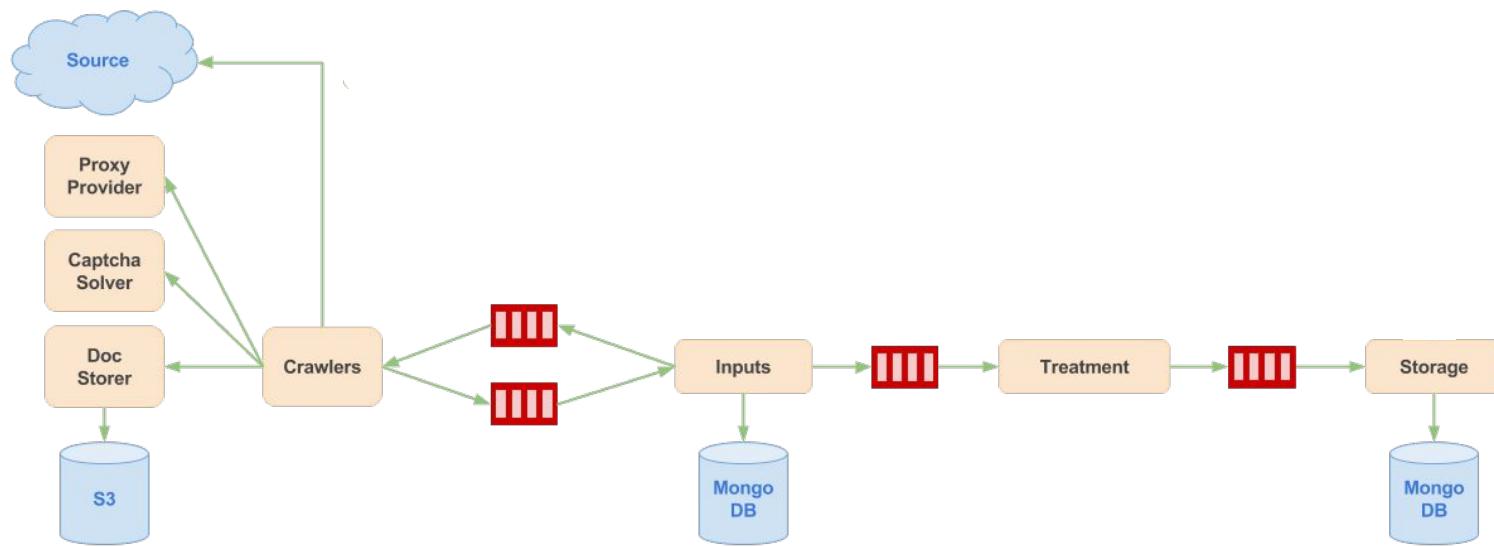


Tratamento e Enriquecimento

Exemplos:

- QCon São Paulo -> QCON SAO PAULO
- NEOWAY BUSINESS SOLUTIONS -> 05.337.875/0001-05







Integração

Exemplo:

Informações das empresas da Receita Federal

+

Informações dos sócios das Juntas Comerciais

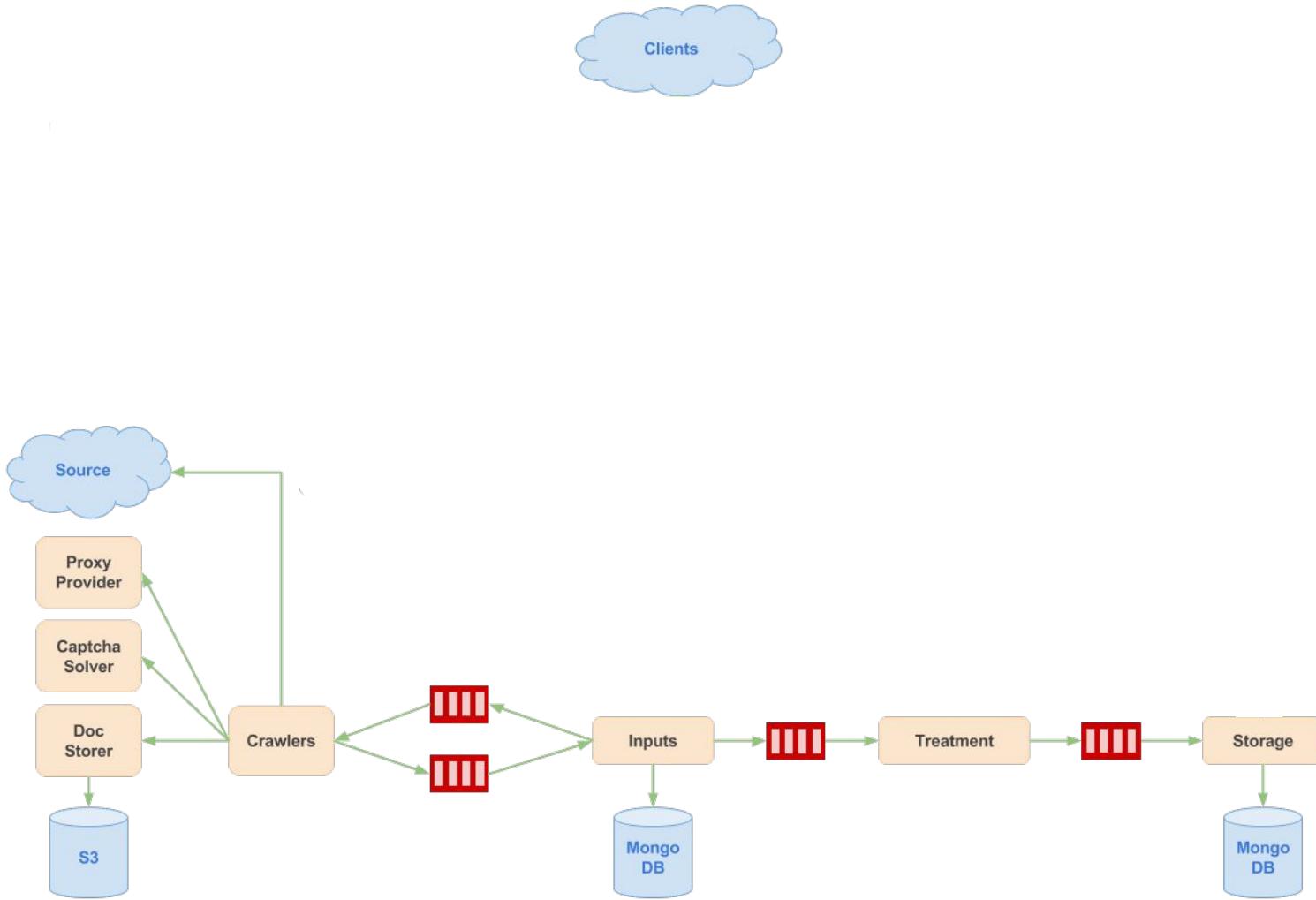
=

Dados completos de empresas do Brasil

Histórico

Exemplo:

- Dia 1: Nome da empresa -> Neoway LTDA
- Dia 2: Nome da empresa -> Neoway SA



Golang



- Compilada, fortemente tipada
- Standard library completa
- Ferramentas Built-in (go test, go bench)
- Pausas do GC na escala de nanosegundos

Golang



- Produtividade
 - Simplista (less is more)
 - Fortemente tipada
- Performance
 - Goroutines (thread 1mb / goroutine 2kb)
- Economia de recursos
 - Baixo consumo de memória

Golang



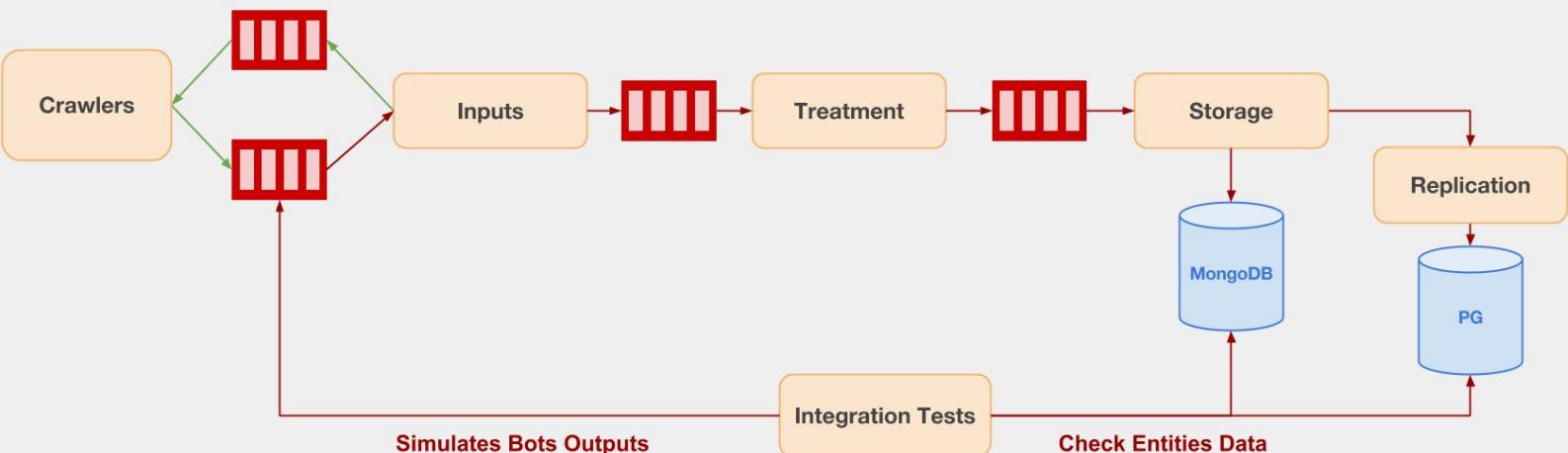
```
package main

import (
    "fmt"
    "net/http"
)

func handler(w http.ResponseWriter, r *http.Request) {
    fmt.Fprint(w, "Hi there, I love this conference!")
}

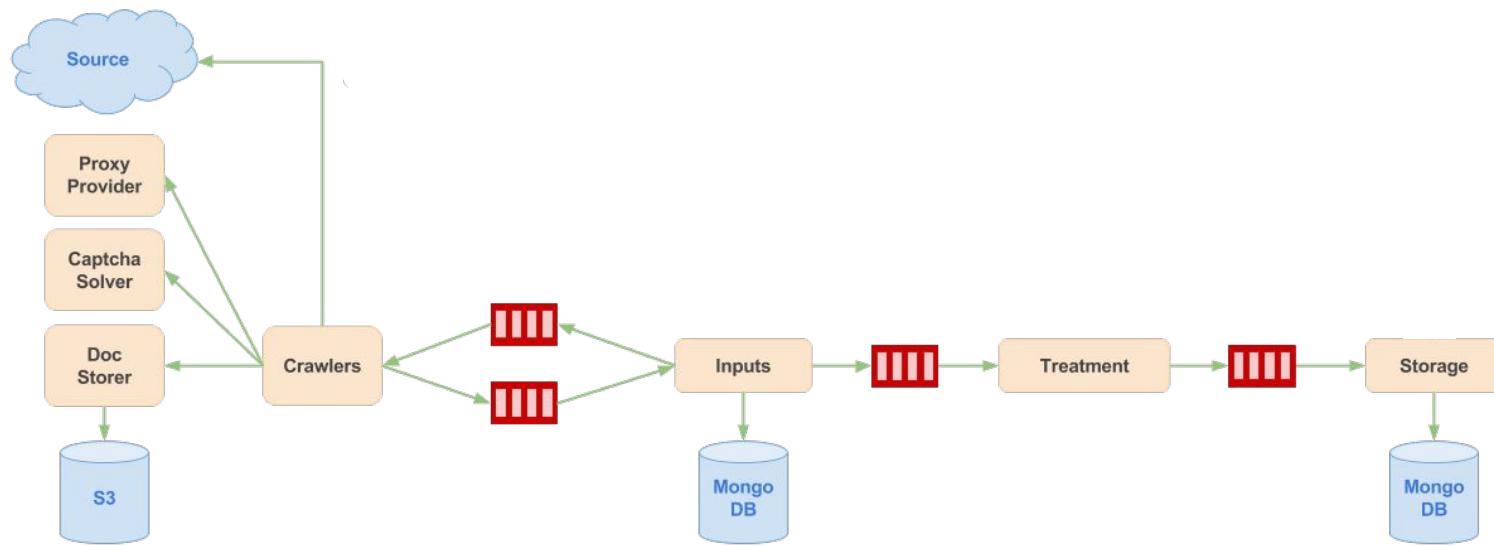
func main() {
    http.HandleFunc("/", handler)
    http.ListenAndServe(":8080", nil)
}
```

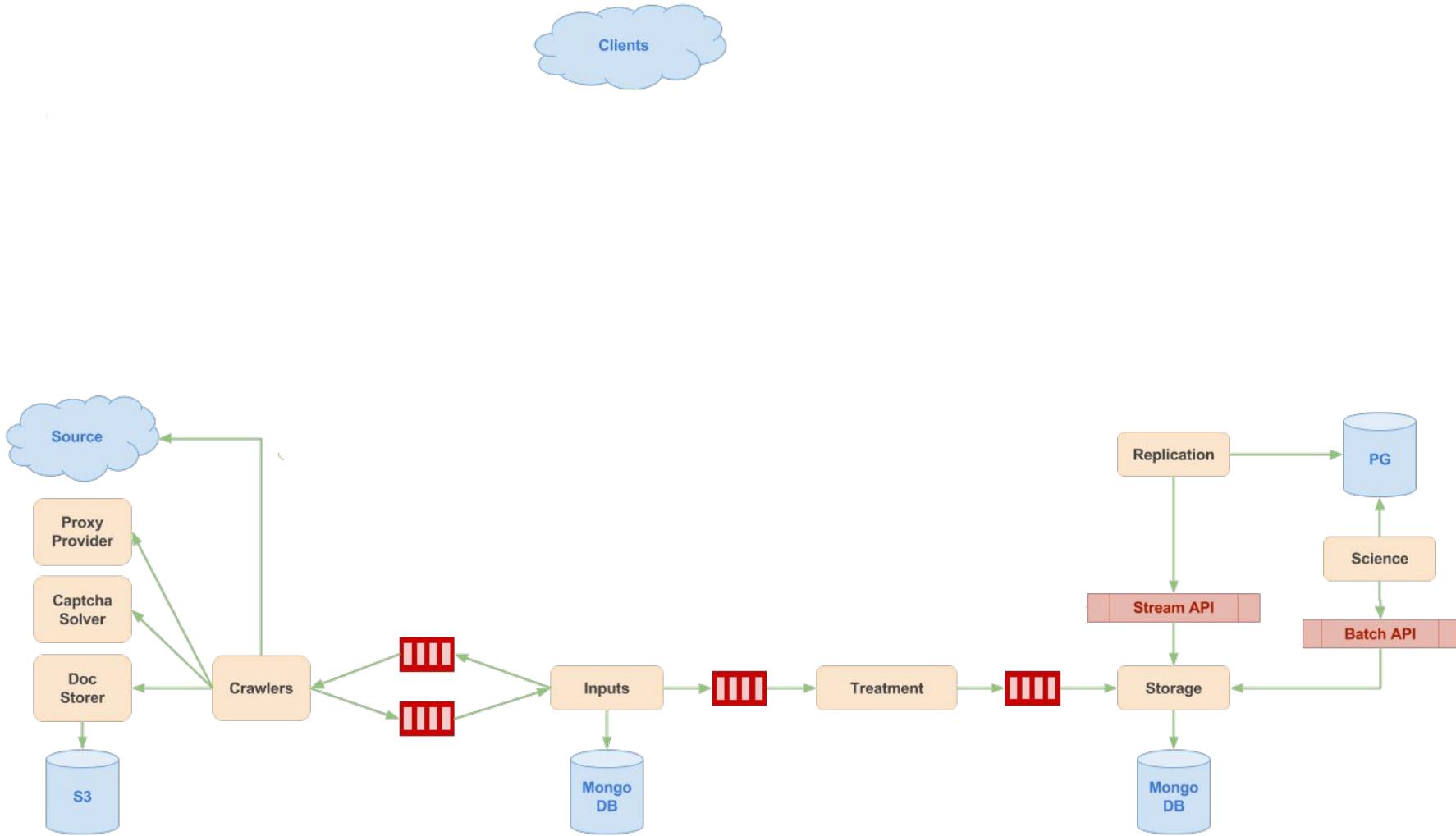
Testes de integração



Testes de integração

- Garante que os serviços funcionem de forma integrada
- Testes integrados com o banco e serviço de mensageria
- Desenvolvedor consegue executá-los em sua máquina

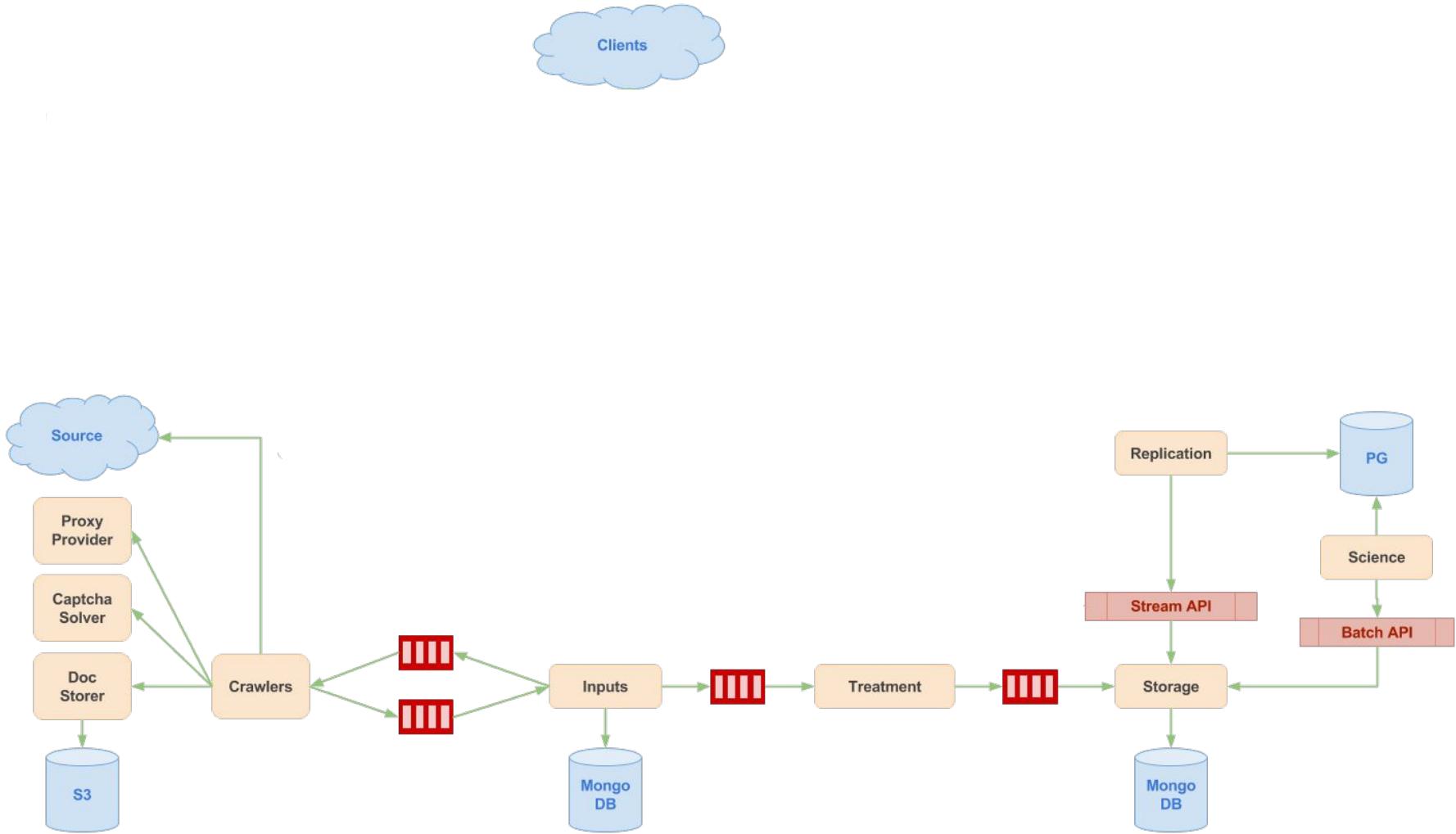


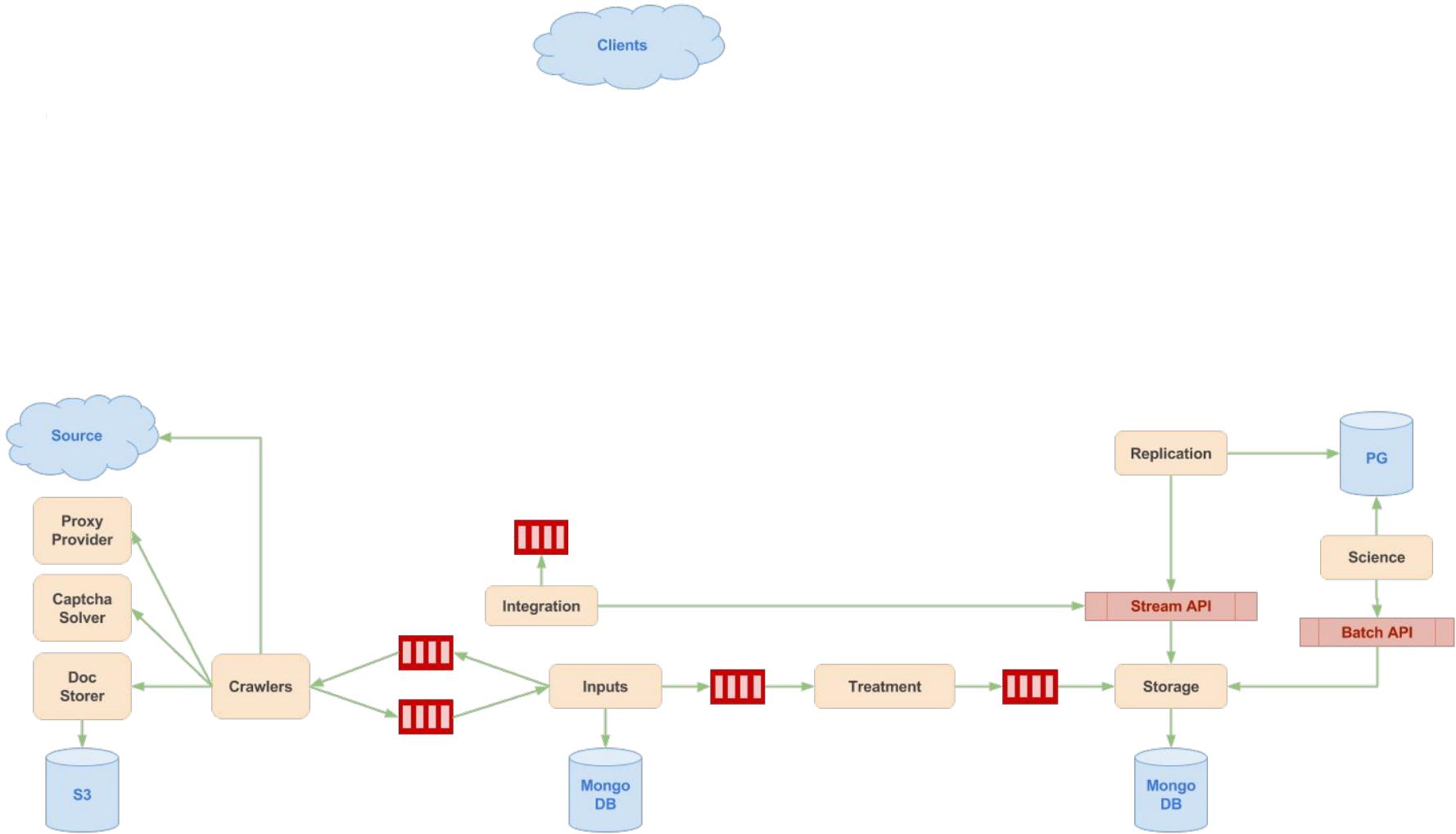


Modelos Estatísticos

Exemplos:

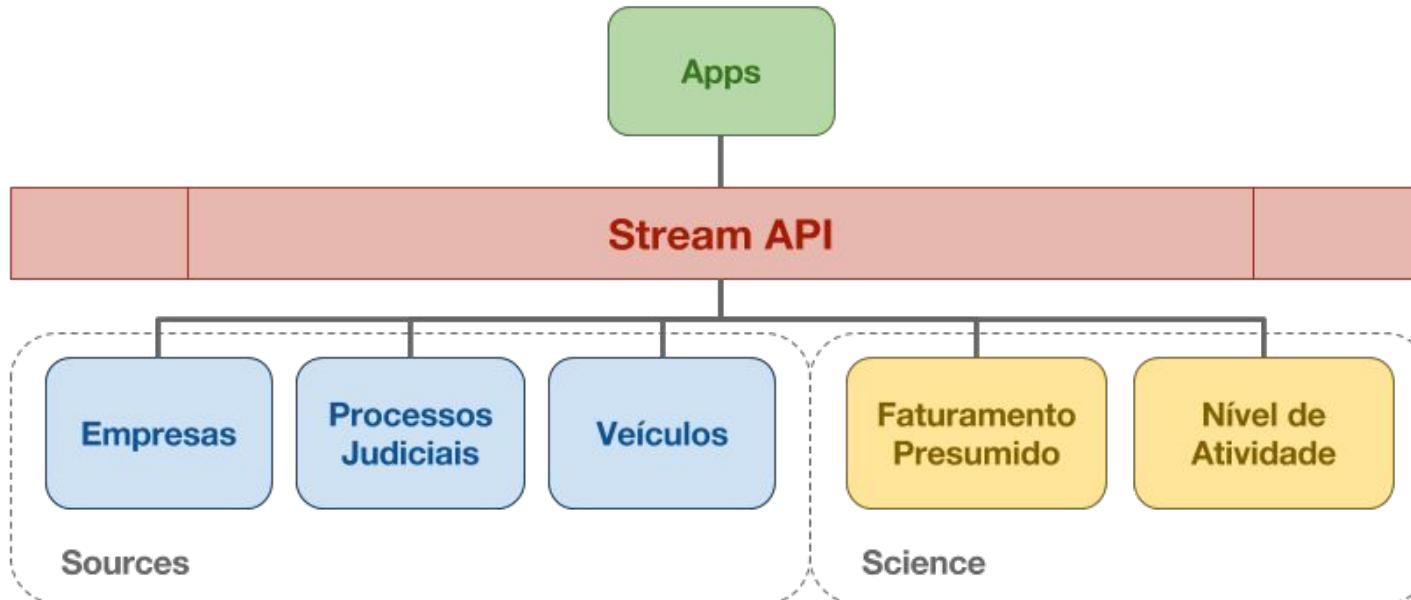
- Nível de atividade de uma empresa
- Faturamento Presumido





Stream API

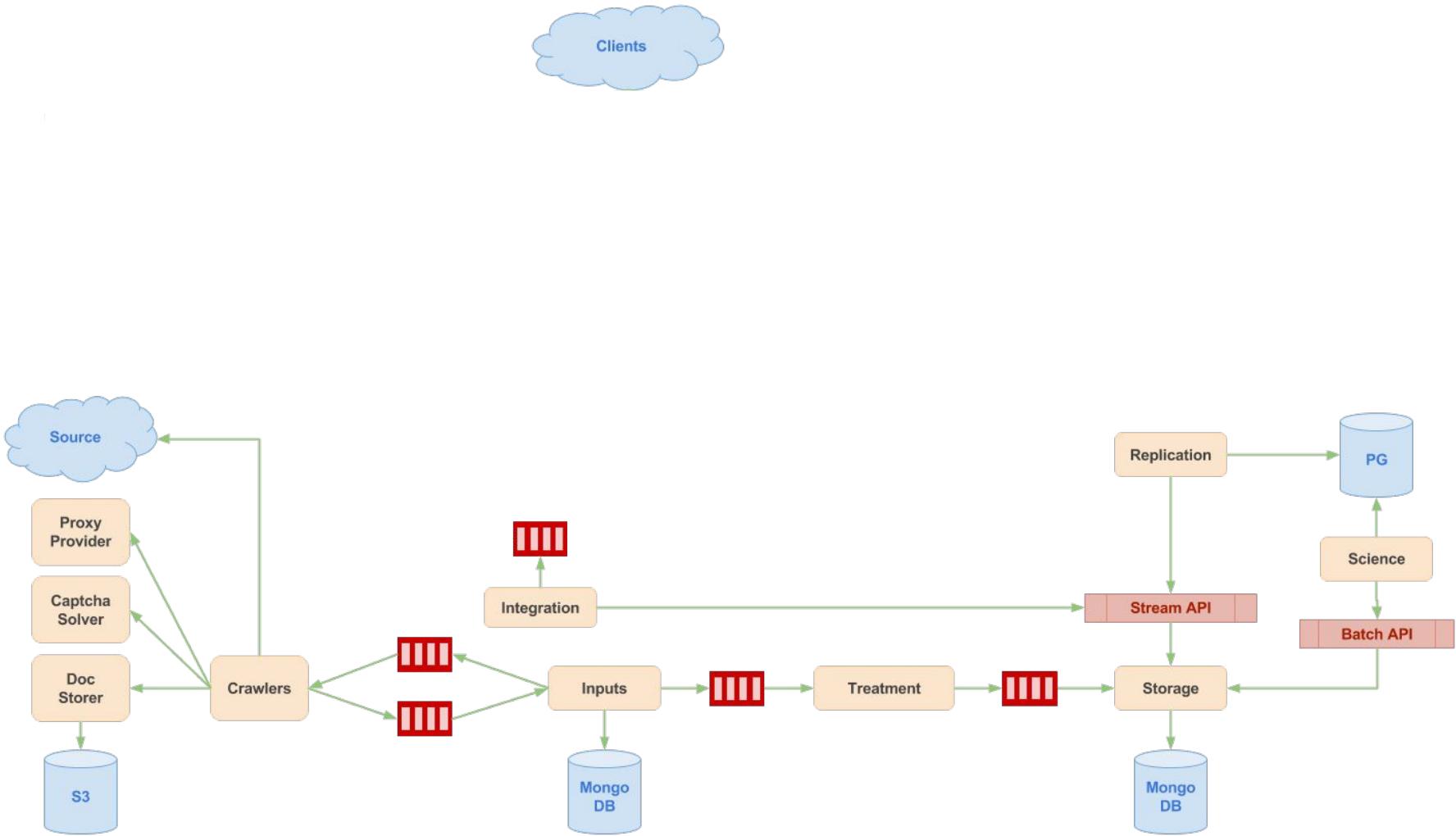
- Conexão persistente
- Novos dados são empurrados pelo servidor
- Sem necessidade de polling
- Controle de processamento via timestamp

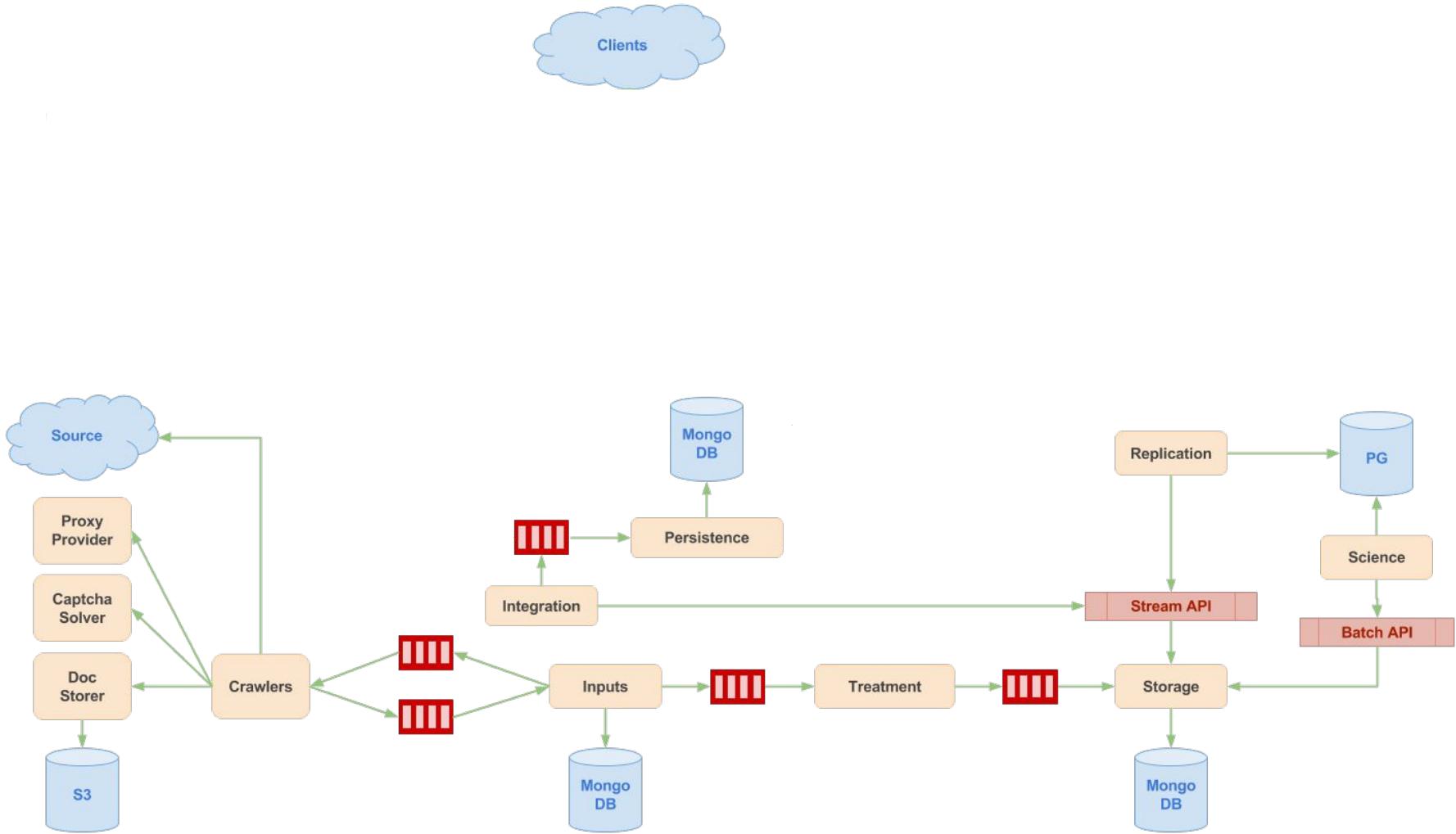


S3

DB

DB





Persistência

- Interessado em quaisquer tópicos com **data-domains.***
- Na sequência, publica a mesma mensagem em **domains.***



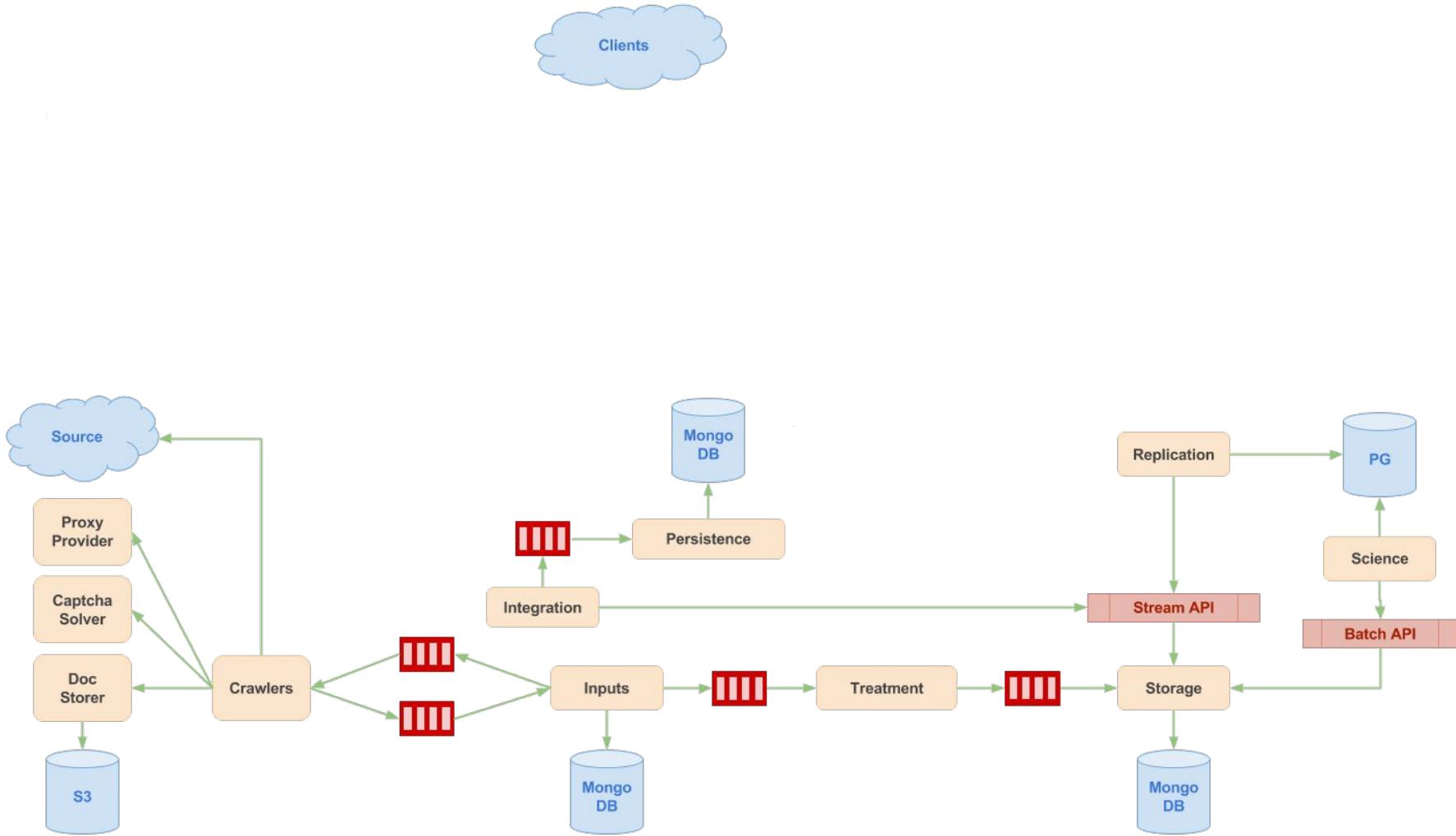
MongoDB

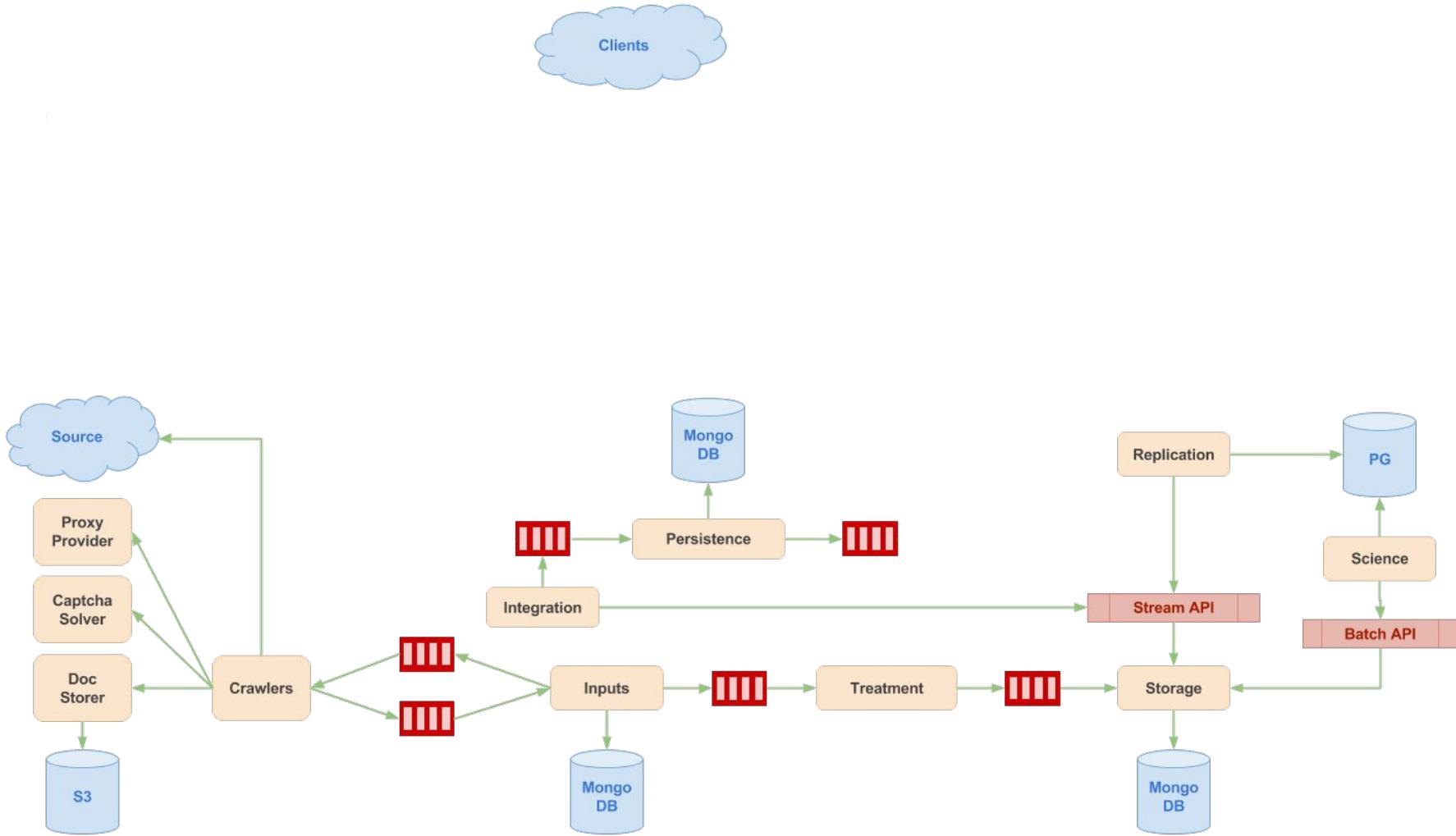
- Utilizado p/ pesquisas simples (sempre pelo _id)
- Réplica com três nós
- Sub-arrays separados em collections



MongoDB

- **empresas** (database)
 - **empresas** (collection - parent)
 - **funcionarios** (collection - child)
 - **socios** (collection - child)
 - **veiculos** (collection - child)
 - **arts** (collection - child)
 - **clientdata_X** (collection - child / por cliente)
 - **tags_X** (collection - child / por cliente)







Mensageria

Como usamos:

- Convenção de nomes de tópicos
 - `data-domains.empresas.funcionarios` (-> Mongo)
 - `domains.empresas.funcionarios` (-> ES)



Mensageria

- NSQ.io - Plataforma de mensageria
- Distribuída (without any centralized brokers)
- Admin UI
- Official Docker image
- Feito em Go



Streams / data-domains.empresas.main

Topic: **data-domains.empresas.main**[Empty Queue](#)[Delete Topic](#)[Pause Topic](#)

Topic Message Queue

NSQd Host	Depth	Memory + Disk	Messages	Channels
✗ rkt-0ff651c4-3be6-4964-8d27-ed18f6b22239:4151 (10.11.11.133:4151)	0	0 + 0	58,759,785	1
✗ rkt-5ffa1fbb-ed98-4c48-ac9d-5d722b5ac12e:4151 (10.11.12.41:4151)	0	0 + 0	65,637,874	1
✗ rkt-84d90d96-9636-4155-a913-4fc255d6ed1f:4151 (10.11.12.115:4151)	0	0 + 0	62,678,894	1
✗ rkt-c069ae97-24df-4867-83ba-3f254b923ab4:4151 (10.11.11.169:4151)	0	0 + 0	53,215,886	1
Total:	0	0 + 0	240,292,439	1

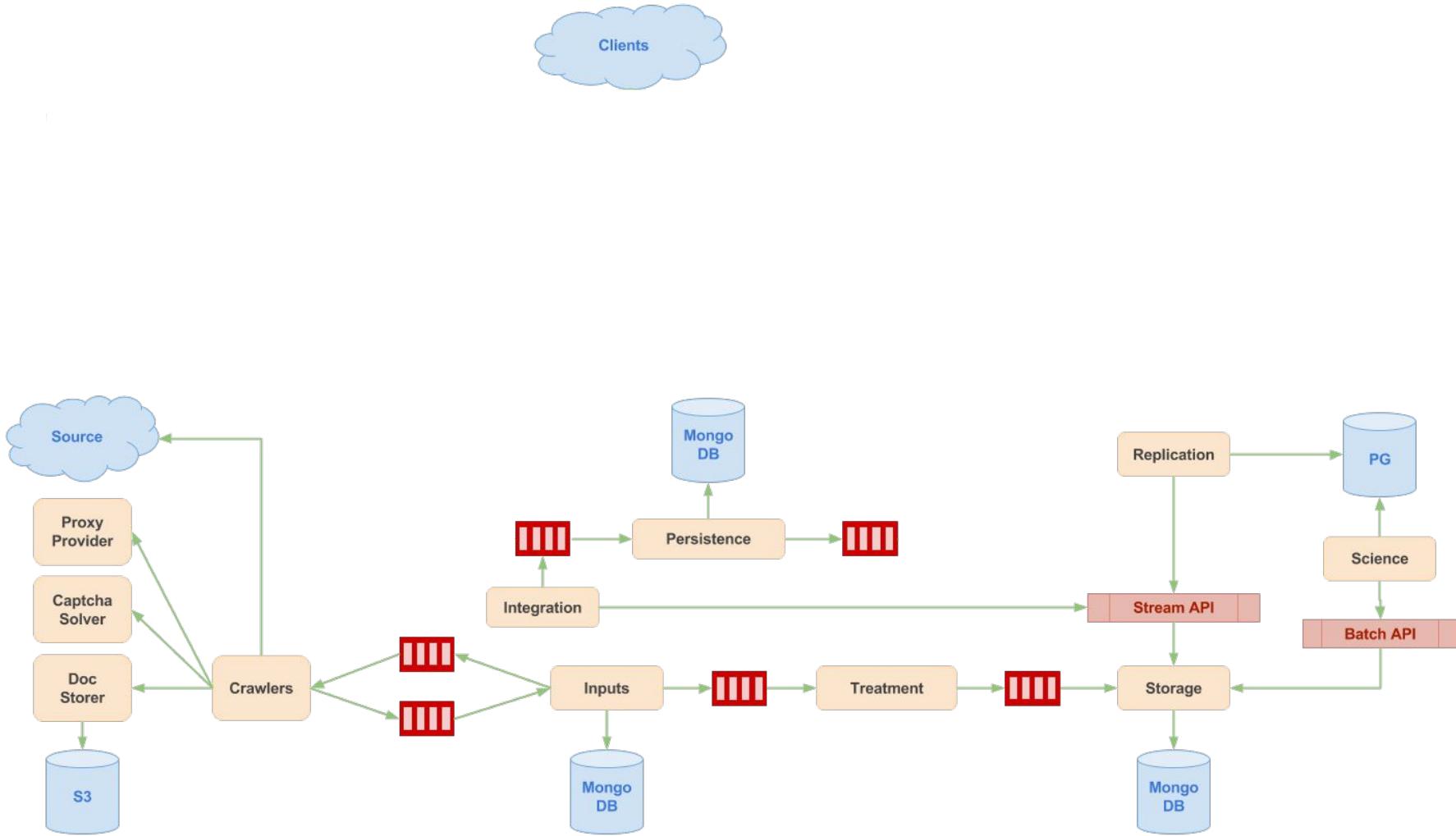
Channel Message Queues

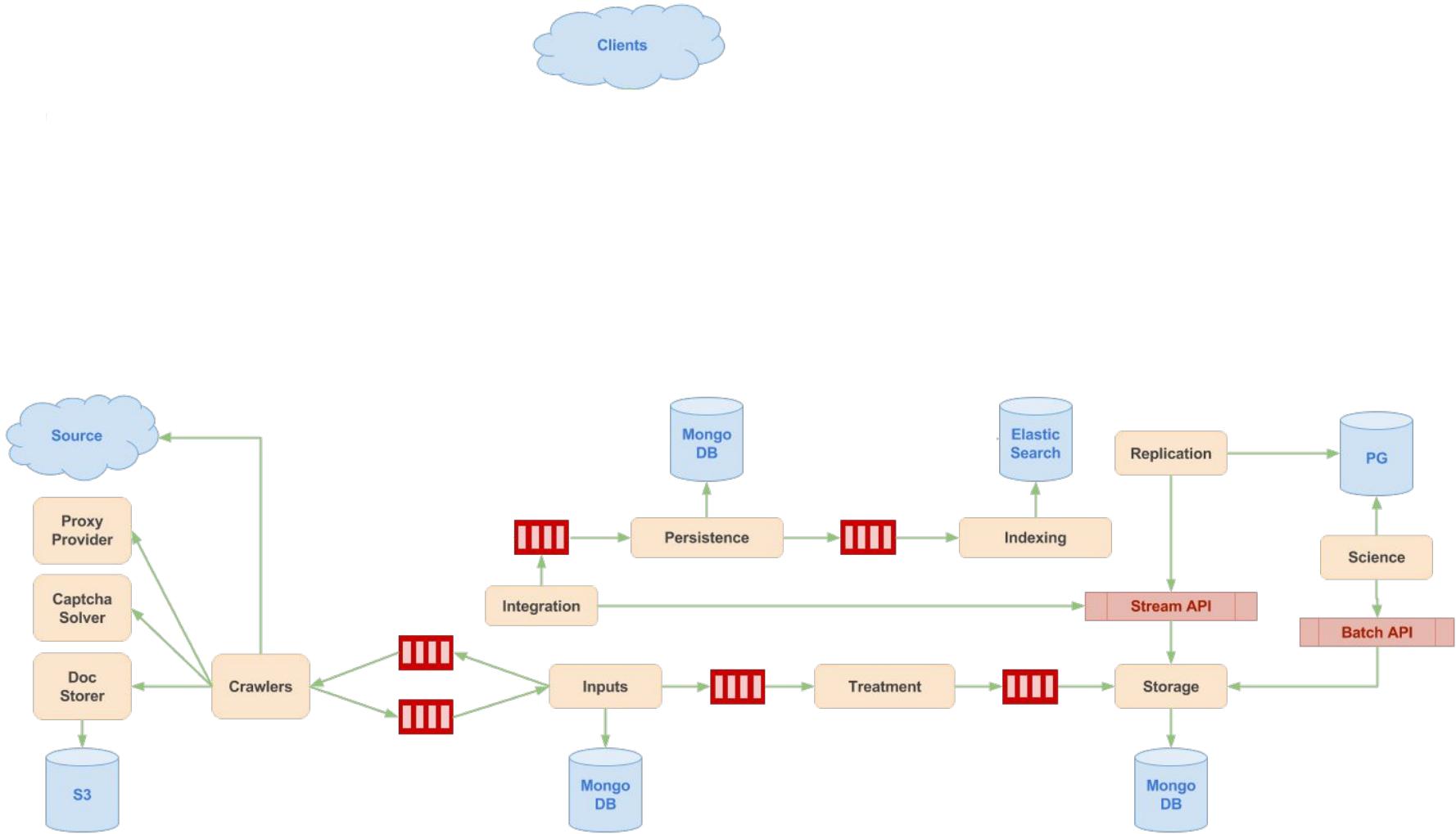
Channel	Depth	Memory + Disk	In-Flight	Deferred	Requeued	Timed Out	Messages	Connections
store	0	0 + 0	0	0	3	0	240,295,088	16

S3

DB

DB





Indexação

- Interesse em quaisquer tópicos com domains.*



Elastic Search

Mapping:

- Estrutura similar ao Mongo (parent / child)

Elastic Search

Dados indexados:

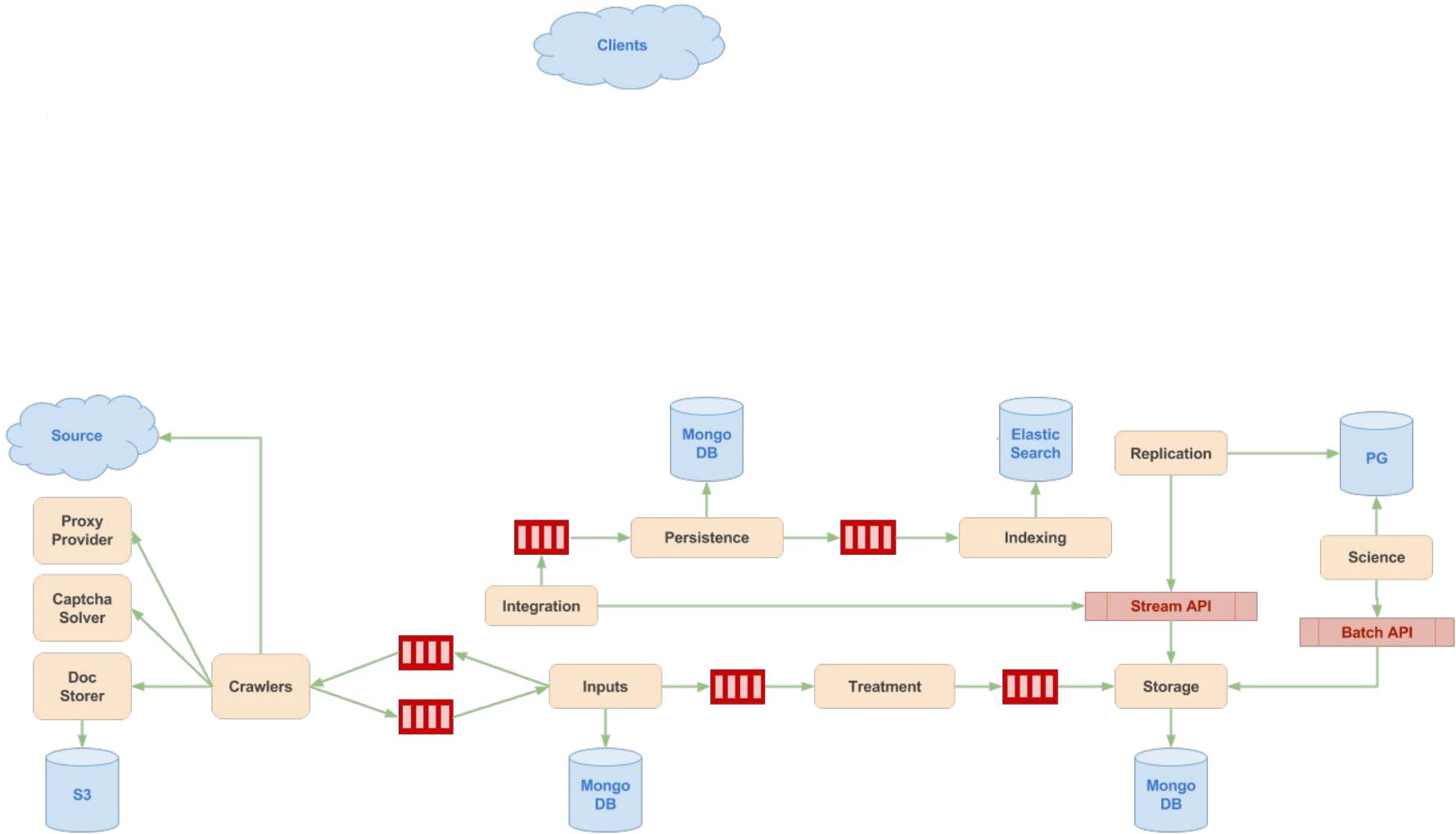
- Empresas: **6,3 TB**
- Pessoas: **3 TB**
- Processos Judiciais: **2,2 TB**
- Companies USA: **178 GB**

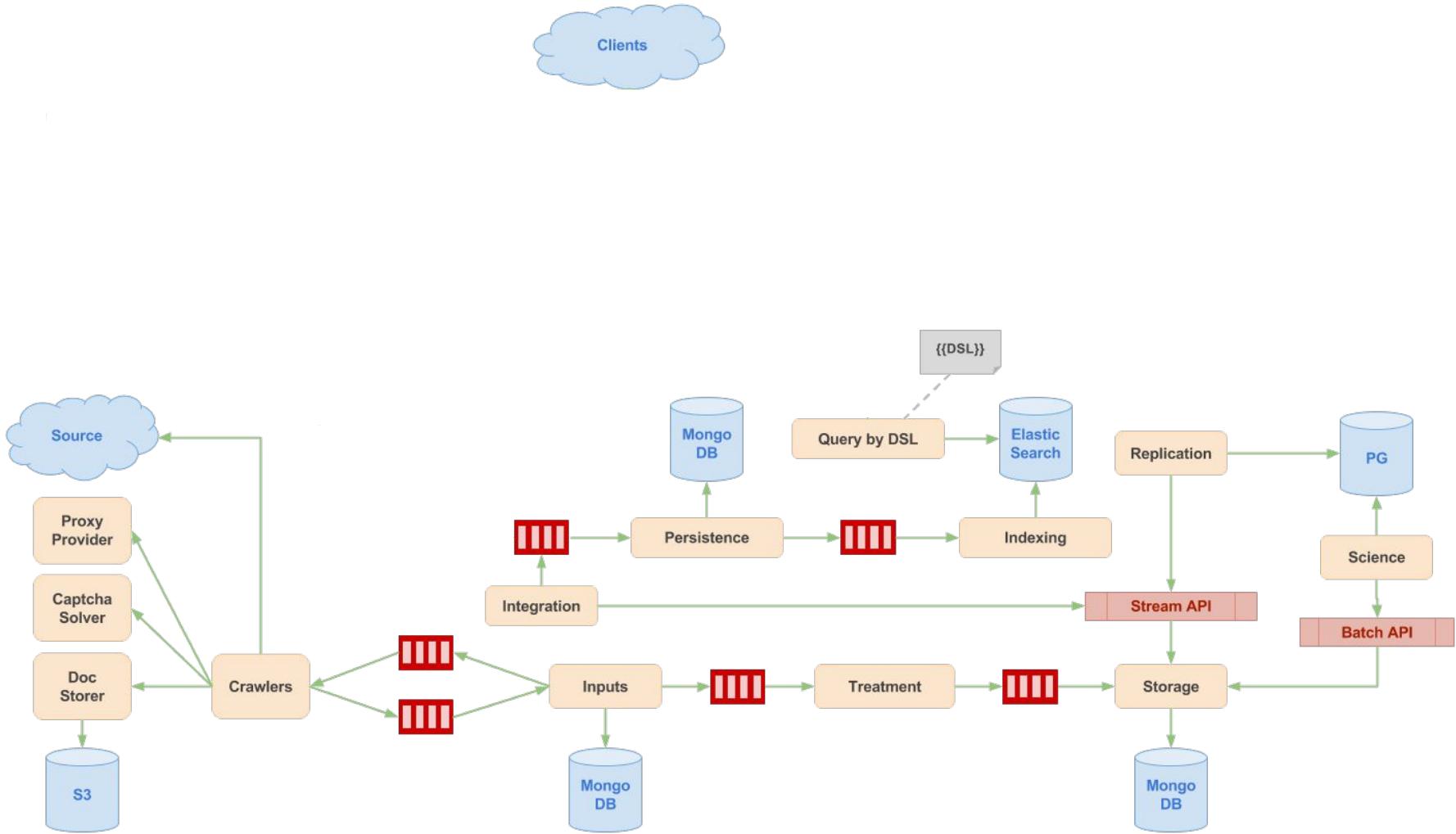


Elastic Search

Um cluster com:

- 13 data nodes
- 3 master nodes





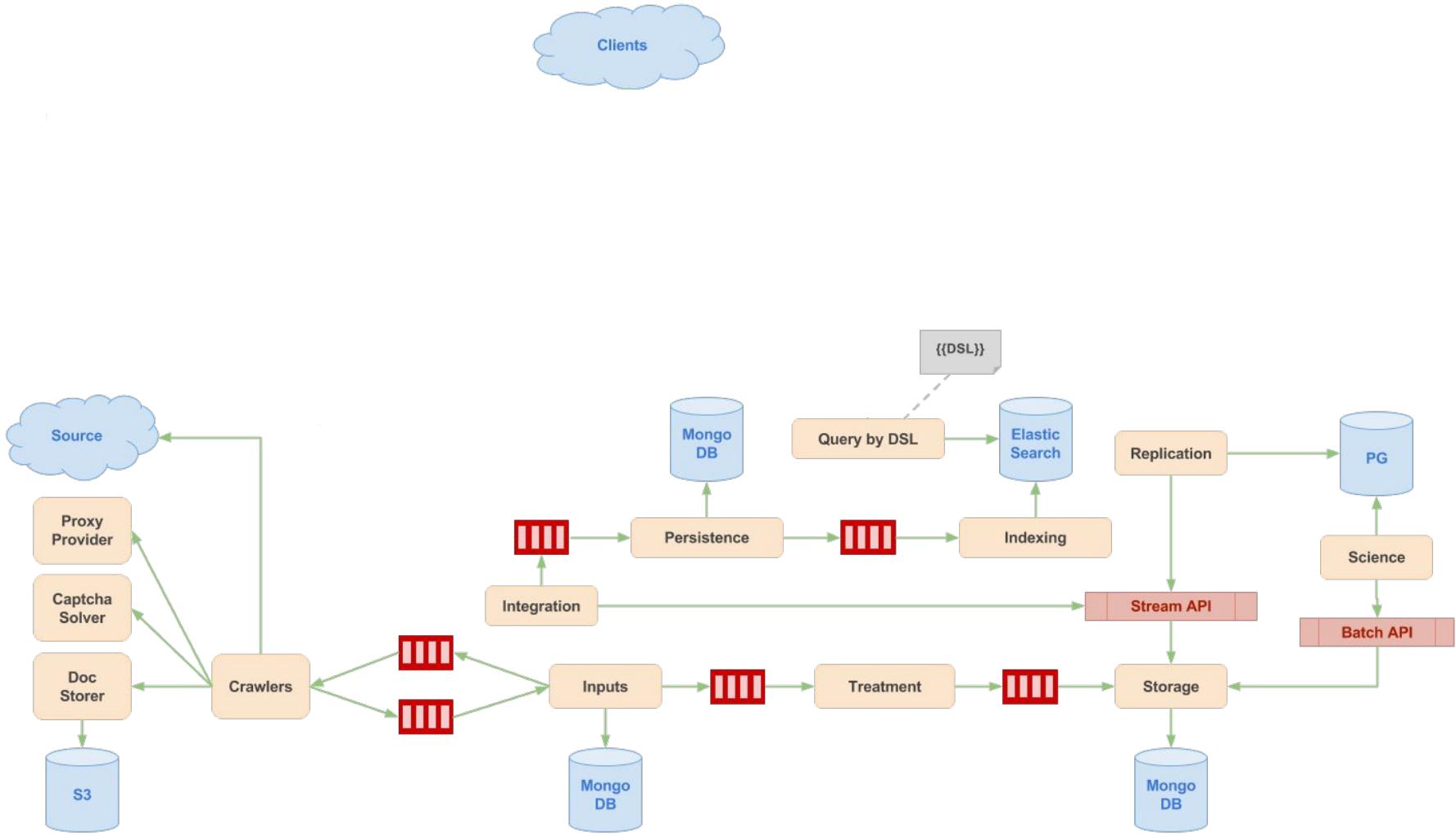
Neoway Search DSL

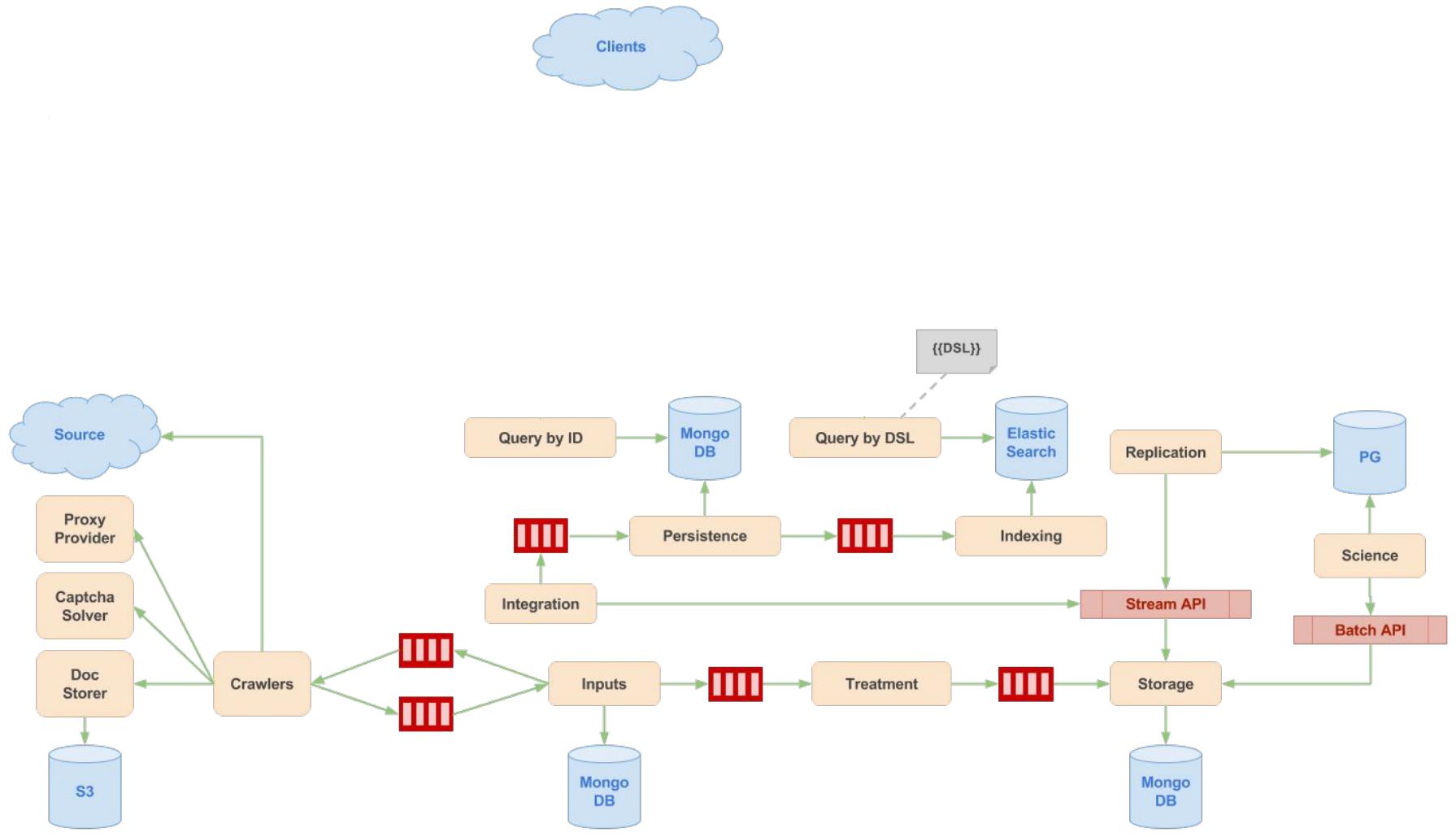
Motivo:

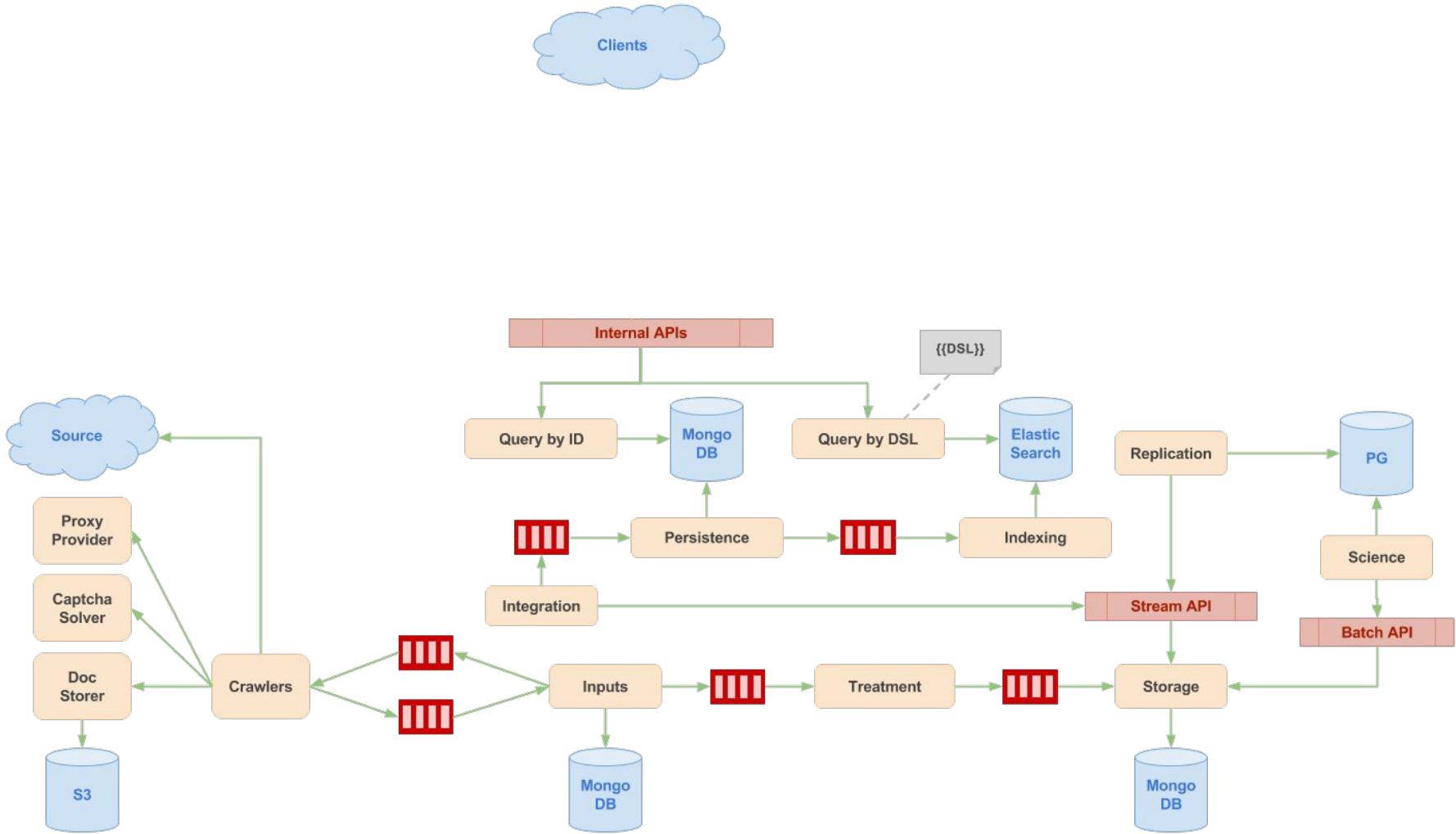
- Abstração da sintaxe do Elastic Search

Neoway Search DSL

```
{  
    "from": 0, "size": 10, "fields": ["cnpj", "razaoSocial"],  
    "query": {  
        "$or": [  
            {  
                "$and": [  
                    {"endereco.uf": "SP"},  
                    {"socios.cpf": "12611122211"}  
                ]  
            },  
            {"totalFuncionarios": {"$lt": 100}}  
        ]  
    }  
}
```





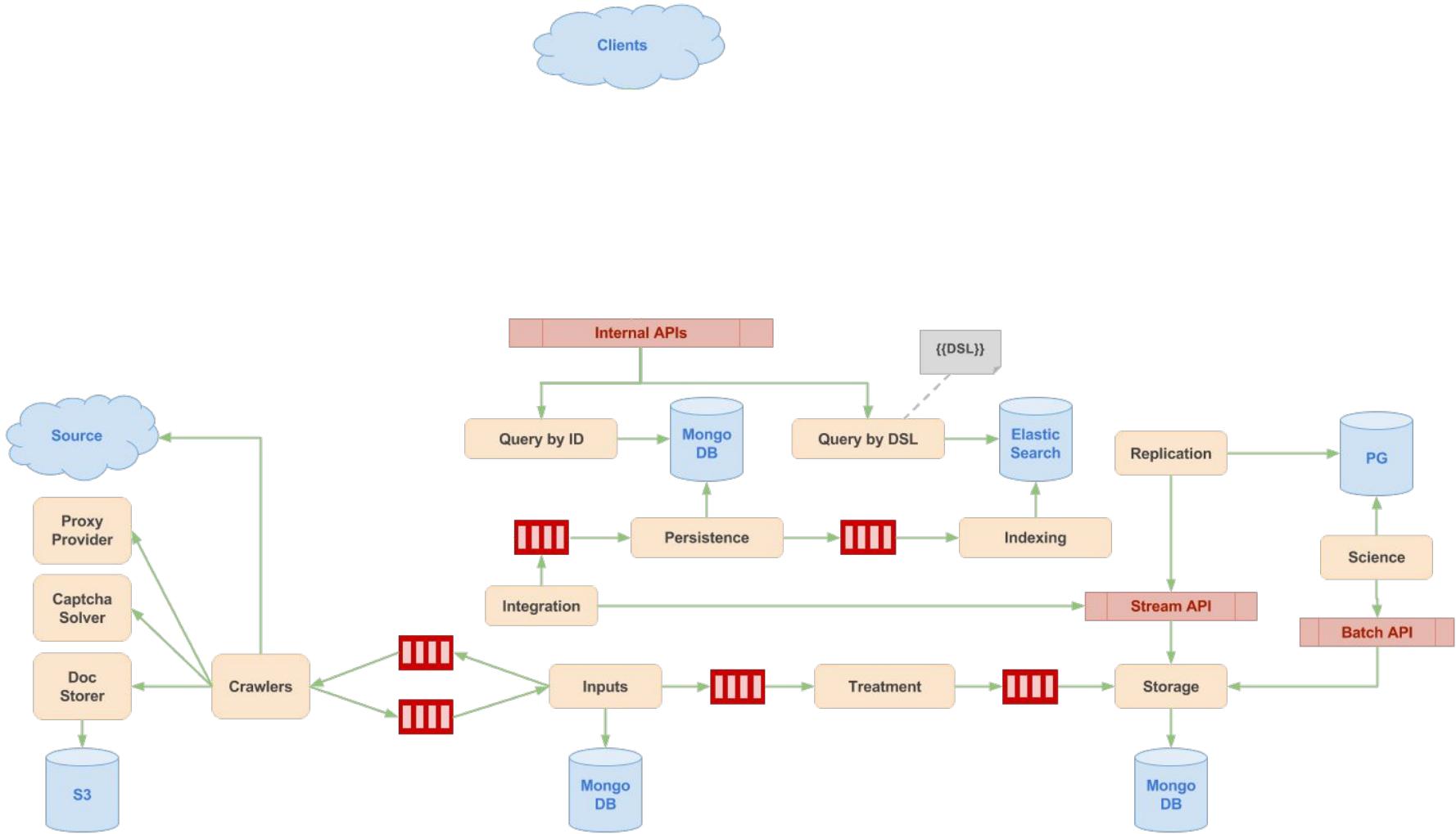


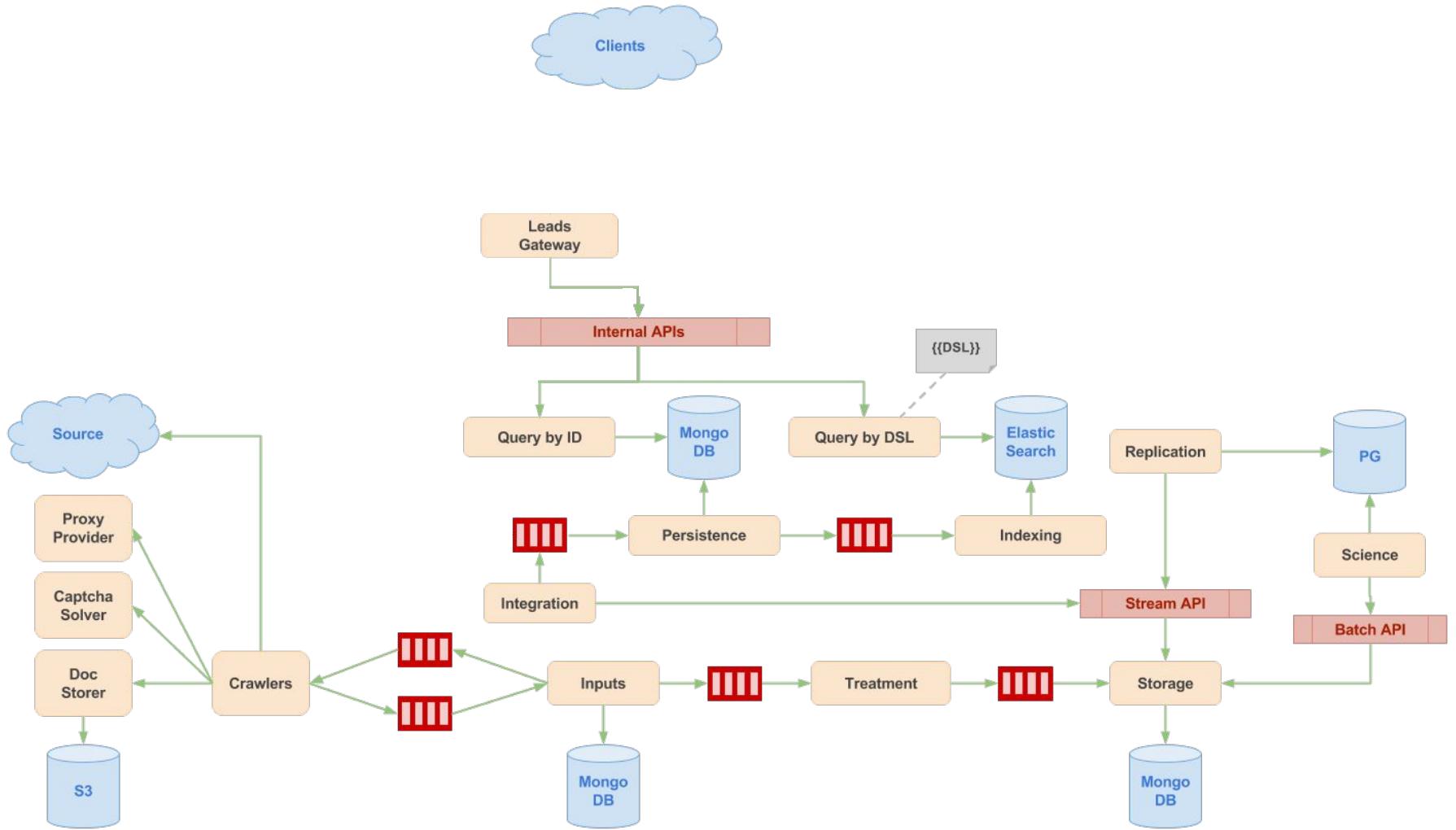
Internal Router

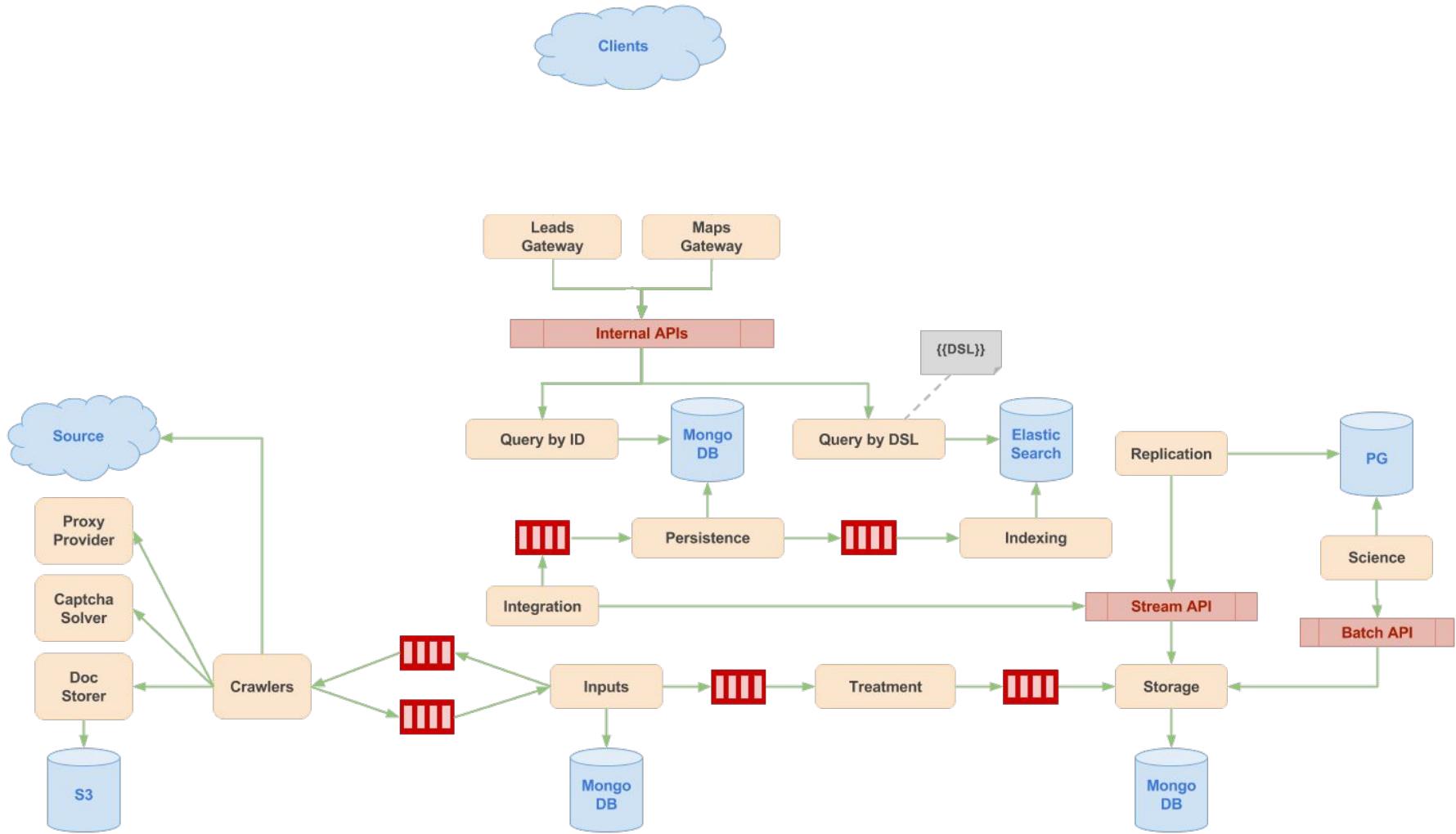
Como usamos:

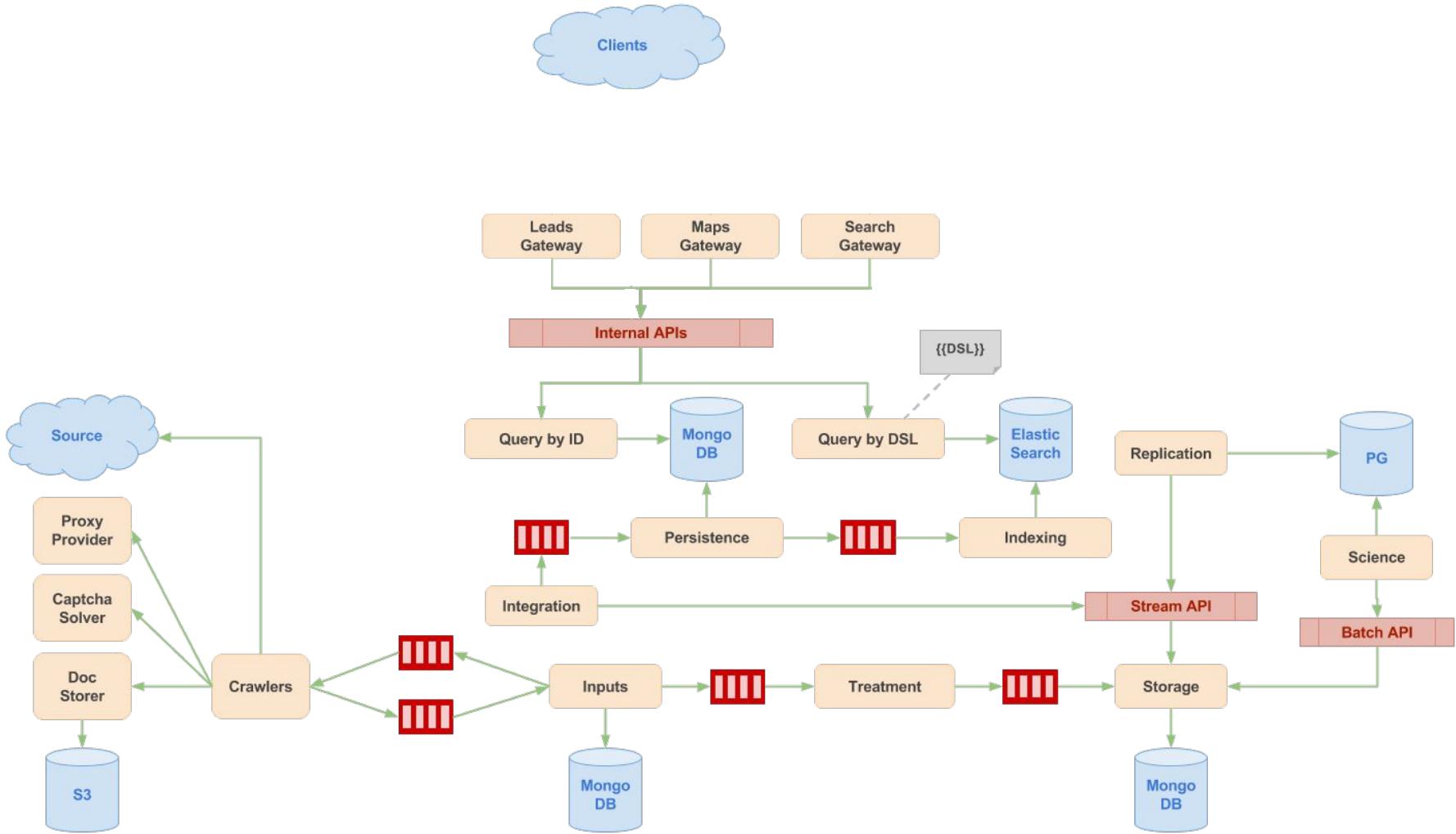
- <http://vulcand.github.io>
- Rotas configuradas via ETCD

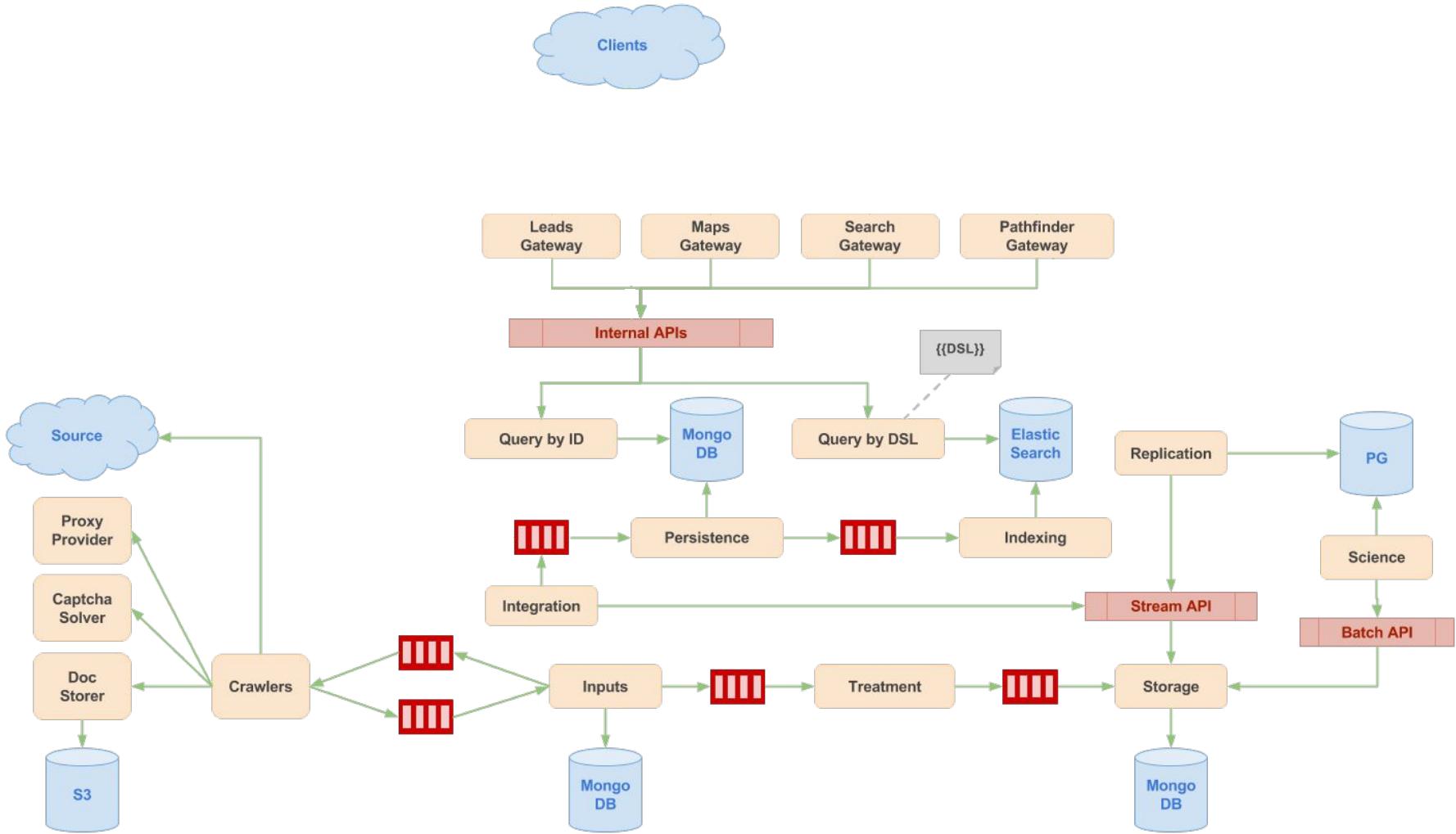
"It uses Etcd as a configuration backend, so changes to configuration take effect immediately without restarting the service."

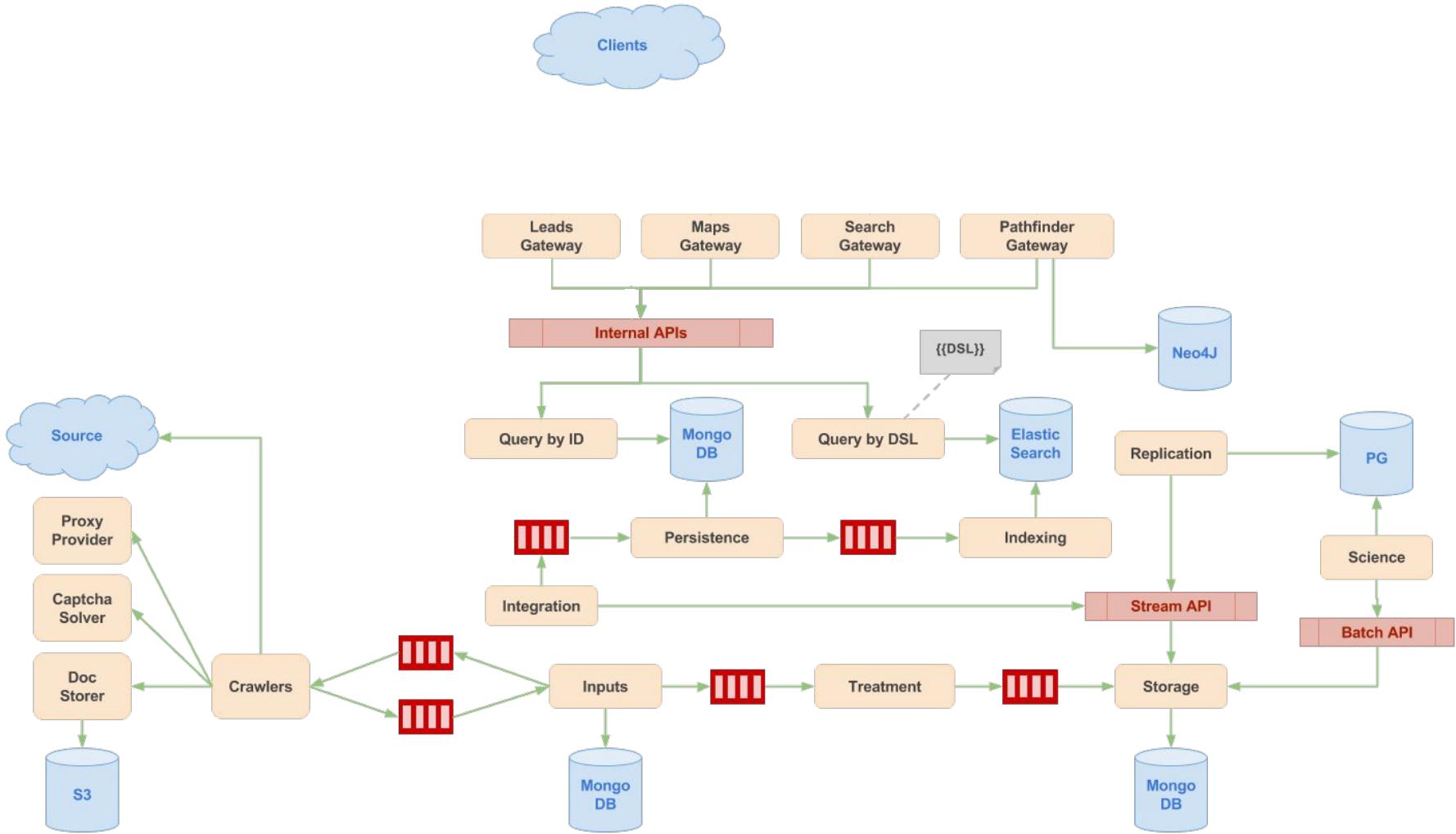








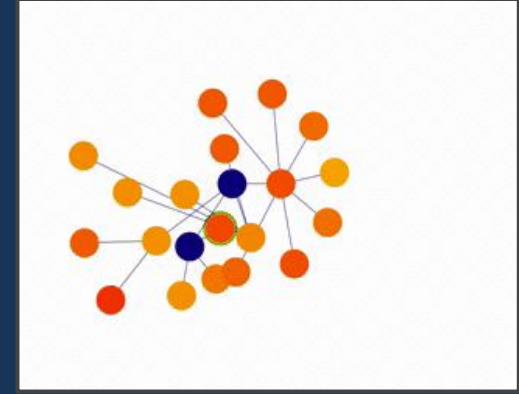
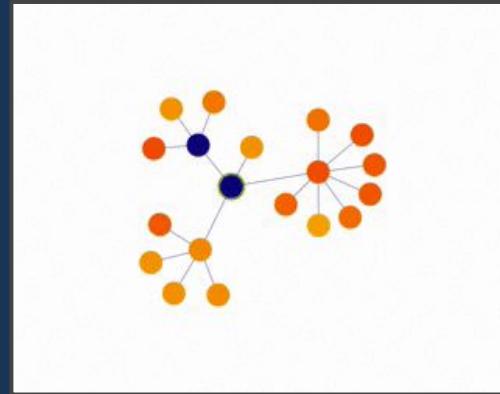


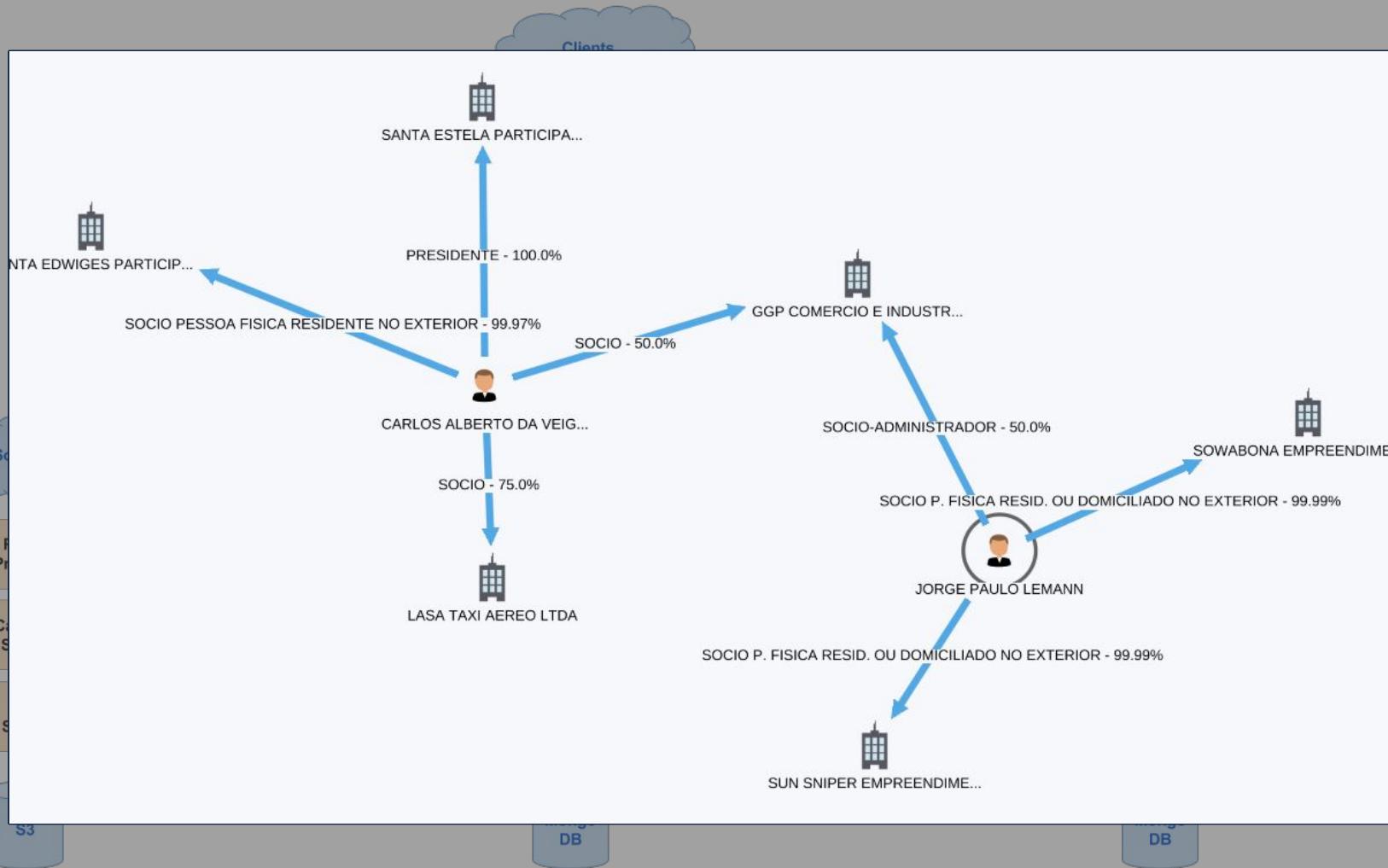


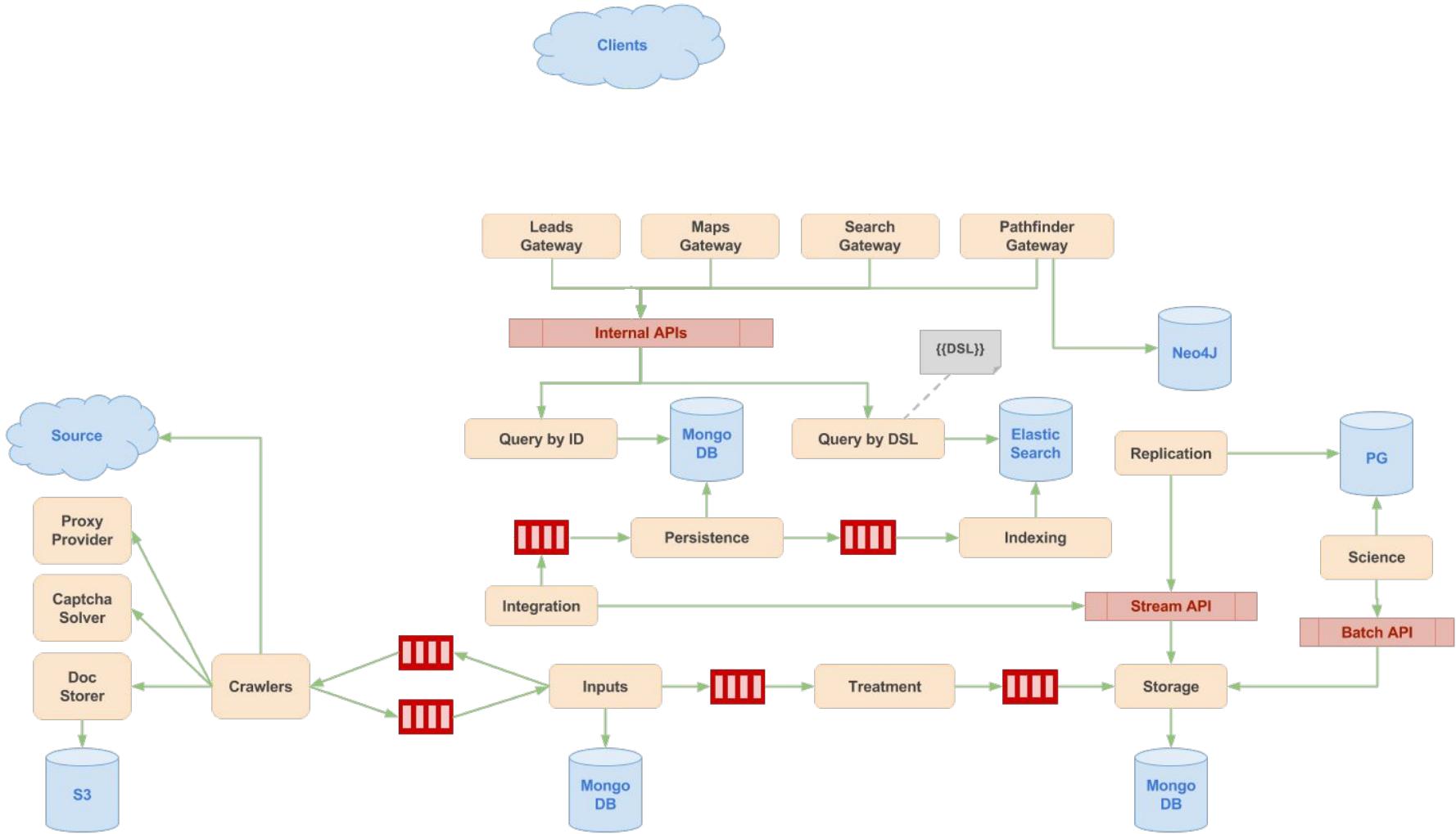
Neo4J + Keylines

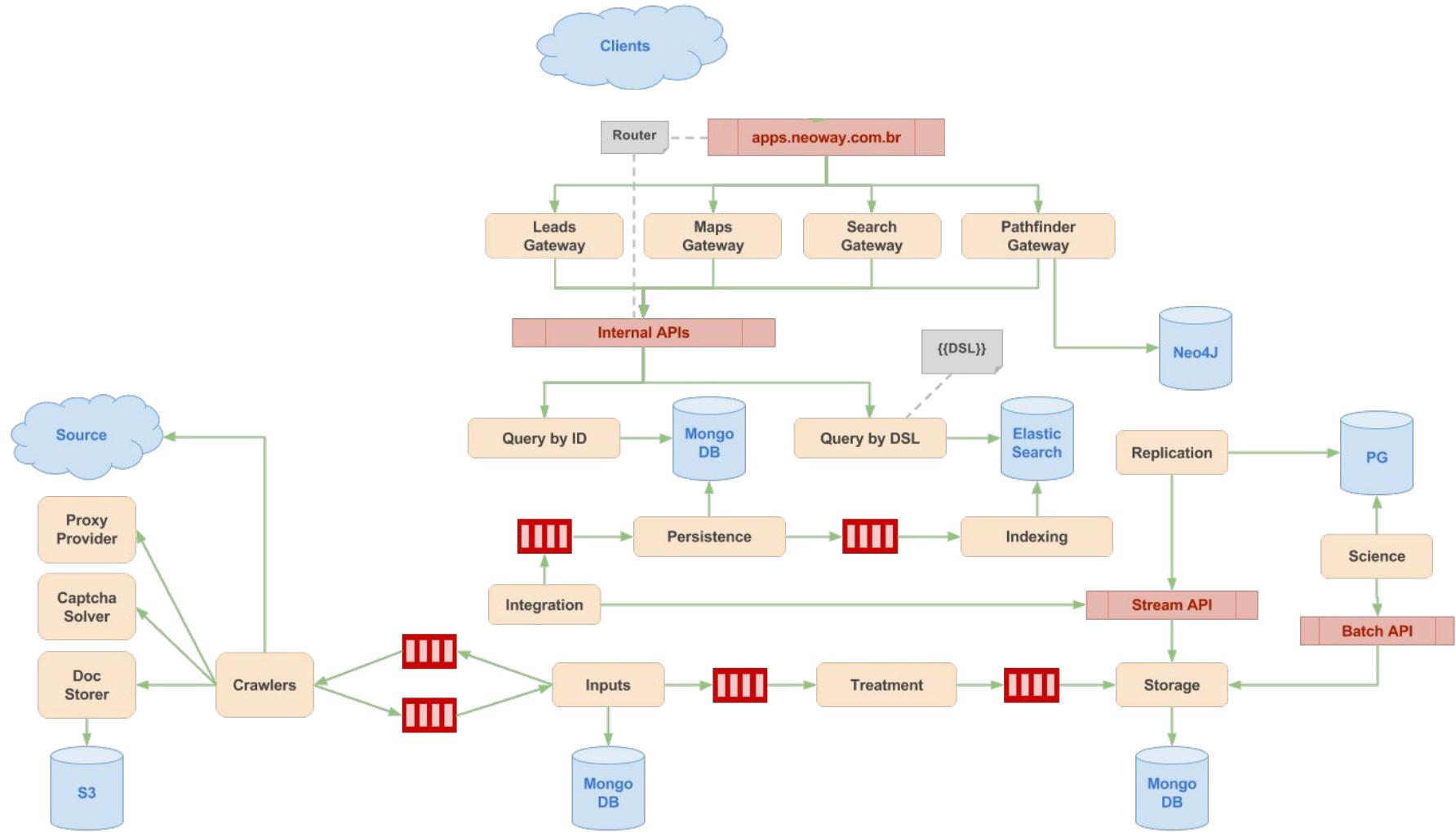
Necessidade:

- Analisar a ligação entre registros de forma visual



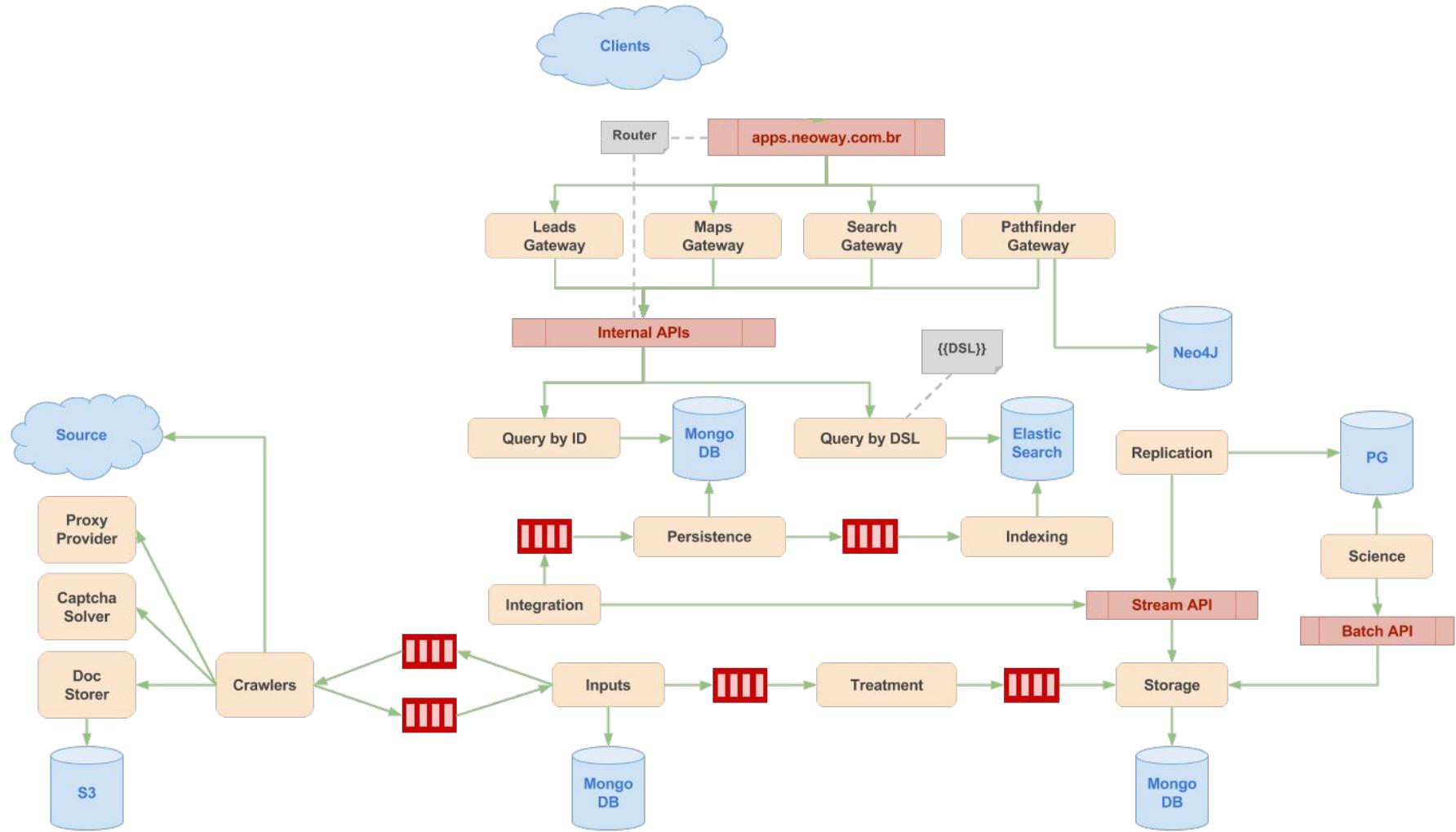


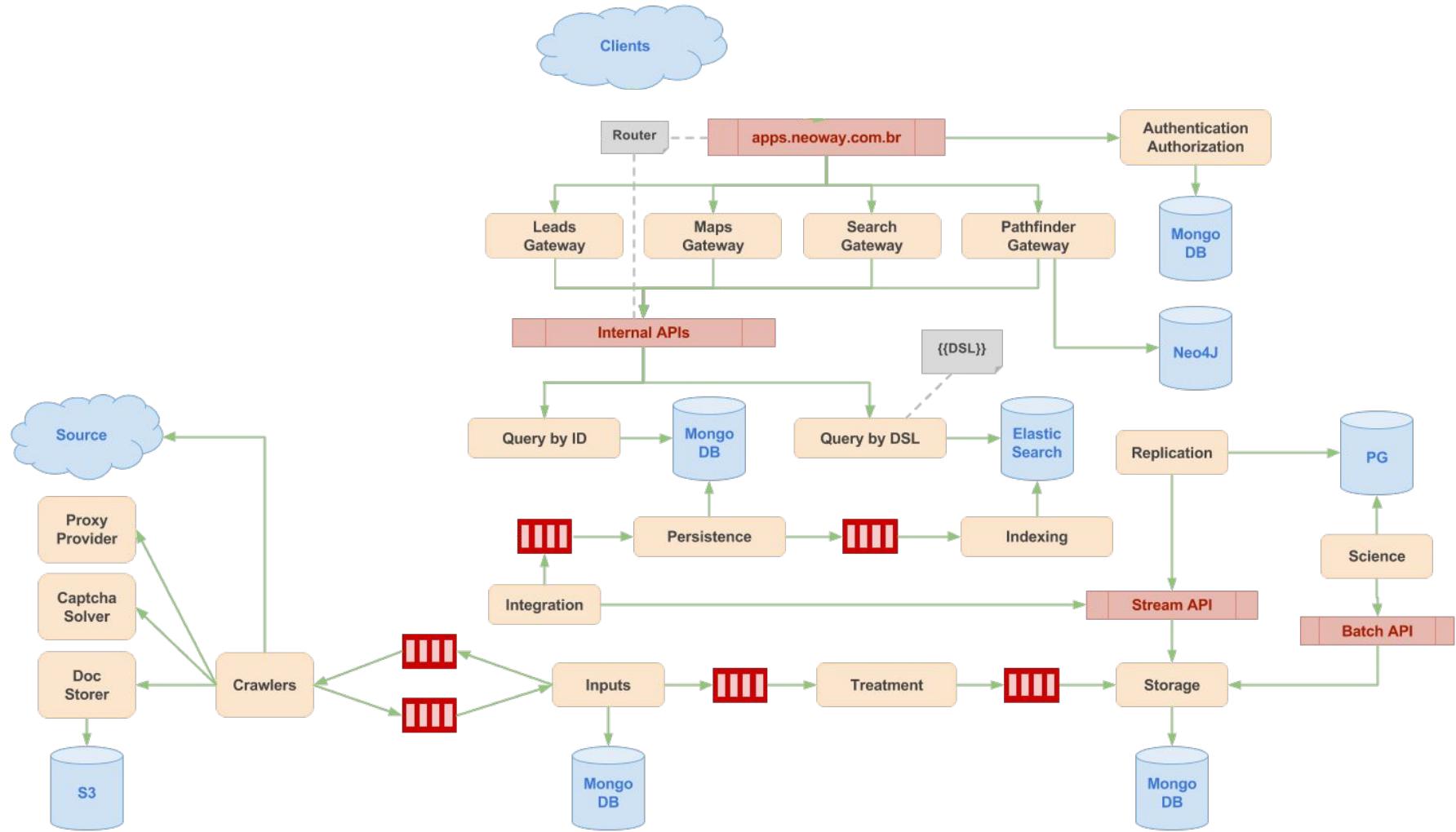


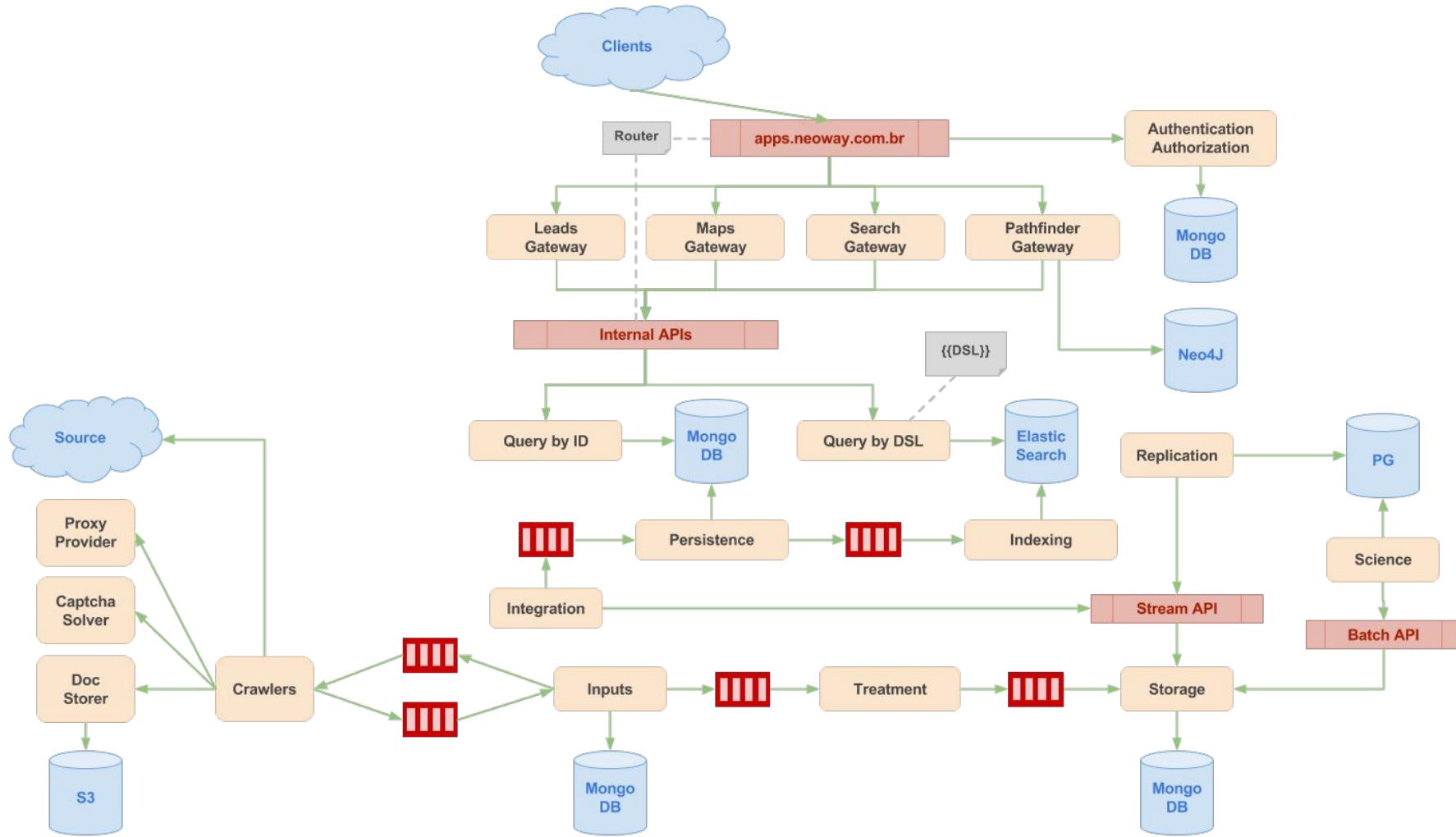


External Auth./Auth./Router

- Outro serviço com VulcanD
- Rotas também configuradas via ETCD
- Custom middleware (redireciona p/ serviço de autenticação/autorização)
 - Autenticação
 - Autorização









Marcadores

Requisitos:

- Cliente marca empresas em lote
- Cliente busca as empresas com suas marcações

Clients

Neoway Selecte o campo... Search icon MODO DE PESQUISA RÁPIDA ricardo.longa@neoway.com.br EMPRESAS DO BRASIL Notifications: 2 Settings Help

RAZÃO SOCIAL RICARDO LONGA Close icon Enviar para

ORDENAÇÃO 10 Filter icon

RICARDO DA SILVA LONGA MEI

RESUMO DA PESQUISA NEOWAY TECNOLOGIA INTEGR... X RICARDO DA SILVA LONGA MEI X

RICARDO DA SILVA LONGA MEI

NÍVEL DE ATIVIDADE INATIVA

CERTIDÃO NEGATIVA ALGUMAS CND IDENTIFICADAS

adicionar...

DADOS CADASTRAIS

CNPJ	21.590.481/0001-02	Data Abertura	19/12/2014
Razão Social	RICARDO DA SILVA LONGA MEI	Idade Da Empresa	1
Nome Fantasia	LONGA CONSULTORIA E TREINAMENTOS	Identificador Do CNAE Principal	8599-6/03
Código Natureza Jurídica	2135	Descrição Do CNAE Principal	TREINAMENTO EM INFORMATICA
Descrição Natureza Jurídica	EMPRESARIO INDIVIDUAL	Setor	SERVICOS
Classificação Natureza Jurídica	PRIVADO	Ramo De Atividade	SERVICOS DE EDUCACAO
Situação Cadastral	BAIXADA	Nível De Atividade	INATIVA
Data Situação Cadastral	17/10/2016	Possui QSA Divergente	NÃO
É Matriz	SIM		

Histórico

Clients

Neoway Selecte o campo... Search icon MODO DE PESQUISA RÁPIDA ricardo.longa@neoway.com.br EMPRESAS DO BRASIL Notifications: 2 Settings Help

RAZÃO SOCIAL RICARDO LONGA Close icon Enviar para

ORDENAÇÃO 10 Filter icon

RICARDO DA SILVA LONGA MEI

NÍVEL DE ATIVIDADE INATIVA

CERTIDÃO NEGATIVA ALGUMAS CNDS IDENTIFICADAS

adicionar...

DADOS CADASTRAIS

CNPJ	21.590.481/0001-02	Data Abertura	19/12/2014
Razão Social	RICARDO DA SILVA LONGA MEI	Idade Da Empresa	1
Nome Fantasia	LONGA CONSULTORIA E TREINAMENTOS	Identificador Do CNAE Principal	8599-6/03
Código Natureza Jurídica	2135	Descrição Do CNAE Principal	TREINAMENTO EM INFORMATICA
Descrição Natureza Jurídica	EMPRESARIO INDIVIDUAL	Setor	SERVICOS
Classificação Natureza Jurídica	PRIVADO	Ramo De Atividade	SERVICOS DE EDUCACAO
Situação Cadastral	BAIXADA	Nível De Atividade	INATIVA
Data Situação Cadastral	17/10/2016	Possui QSA Divergente	NÃO
É Matriz	SIM		

Histórico

S3 DB DB

RICARDO DA SILVA LONGA MEI

CERTIDÃO NEGATIVA ALGUMAS CNDS IDENTIFICADAS

adicionar...

DADOS CADASTRAIS

CNPJ	25.54...	Data Abertura	19/12/2014
Razão Social	ELECAO 2012 RICARDO AZEVEDO LU VEREADOR	Idade Da Empresa	1
Nome Fantasia	LONGA CONSULTORIA E TREINAMENTOS	Identificador Do CNAE Principal	8599-6/03
Código Natureza Jurídica	2135	Descrição Do CNAE Principal	TREINAMENTO EM INFORMATICA
Descrição Natureza Jurídica	EMPRESARIO INDIVIDUAL	Setor	SERVICOS
Classificação Natureza Jurídica	PRIVADO	Ramo De Atividade	SERVICOS DE EDUCACAO
Situação Cadastral	BAIXADA	Nível De Atividade	INATIVA
Data Situação Cadastral	17/10/2016	Possui QSA Divergente	NÃO
É Matriz	SIM		

RICARDO DA SILVA LONGA MEI

CERTIDÃO NEGATIVA NENHUMA CND IDENTIFICADA

adicionar...

DADOS CADASTRAIS

CNPJ	16.404.495/0001-54	Data Abertura	19/12/2014
Razão Social	ELECAO 2012 RICARDO AZEVEDO LONGA VEREADOR	Idade Da Empresa	1
Nome Fantasia	LONGA CONSULTORIA E TREINAMENTOS	Identificador Do CNAE Principal	8599-6/03
Código Natureza Jurídica	2135	Descrição Do CNAE Principal	TREINAMENTO EM INFORMATICA
Descrição Natureza Jurídica	EMPRESARIO INDIVIDUAL	Setor	SERVICOS
Classificação Natureza Jurídica	PRIVADO	Ramo De Atividade	SERVICOS DE EDUCACAO
Situação Cadastral	BAIXADA	Nível De Atividade	INATIVA
Data Situação Cadastral	17/10/2016	Possui QSA Divergente	NÃO
É Matriz	SIM		

Clients

Neoway Selecte o campo... Search icon MODO DE PESQUISA RÁPIDA EMPRESAS DO BRASIL User icon Gear icon Help icon ricardo.longa@neoway.com.br

RAZÃO SOCIAL
RICARDO LONGA Close icon

ORDENAÇÃO Search icon 10 Filter icon

RESUMO DA PESQUISA NEOWAY TECNOLOGIA INTEGR... X RICARDO DA SILVA LONGA MEI X

Enviar para

RICARDO DA SILVA LONGA MEI

Search icon Payment of TAXES icon

NÍVEL DE ATIVIDADE
INATIVA

CERTIDÃO NEGATIVA
ALGUMAS CND IDENTIFICADAS

Visited icon Add icon

DADOS CADASTRAIS

CNPJ	21.590.481/0001-02	Data Abertura	19/12/2014
Razão Social	RICARDO DA SILVA LONGA MEI	Idade Da Empresa	1
Nome Fantasia	LONGA CONSULTORIA E TREINAMENTOS	Identificador Do CNAE Principal	8599-6/03
Código Natureza Jurídica	2135	Descrição Do CNAE Principal	TREINAMENTO EM INFORMATICA
Descrição Natureza Jurídica	EMPRESARIO INDIVIDUAL	Setor	SERVICOS
Classificação Natureza Jurídica	PRIVADO	Ramo De Atividade	SERVICOS DE EDUCACAO
Situação Cadastral	BAIXADA	Nível De Atividade	INATIVA
Data Situação Cadastral	17/10/2016	Possui QSA Divergente	NÃO
É Matriz	SIM		

Histórico

S3 DB DB

A large orange arrow points from the first company entry in the list on the left towards the second company entry, highlighting the transition between entries.

Clients

Neoway  Seleione o campo... 

MODO DE PESQUISA RÁPIDA  ricardo.longa@neoway.com.br
EMPRESAS DO BRASIL    

RAZÃO SOCIAL **RICARDO LONGA**  CRIADOS POR MIM 
VISITADO 

Ordenação  10  

Limpar Pesquisa      Enviar para

ORDENAÇÃO  10  

CNPJ 21.590.481/0001-02
Razão Social RICARDO DA SILVA LONGA MEI
Data Abertura 19/12/2014
Município SAO JOSE

CERTIDÃO NEGATIVA
ALGUMAS CNDS IDENTIFICA... 

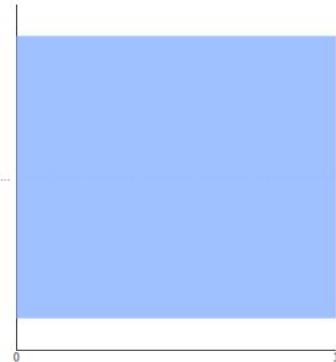
RESUMO DA PESQUISA

1
TOTAL

1
EMPRESAS BAIXADAS

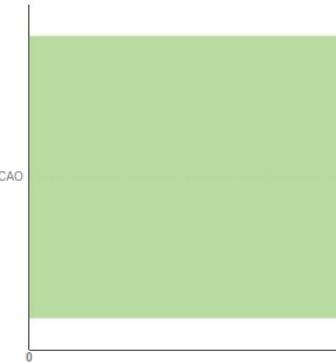
CNAE PRINCIPAL

TREINAMENTO EM INFOR...



RAMO DE ATIVIDADE

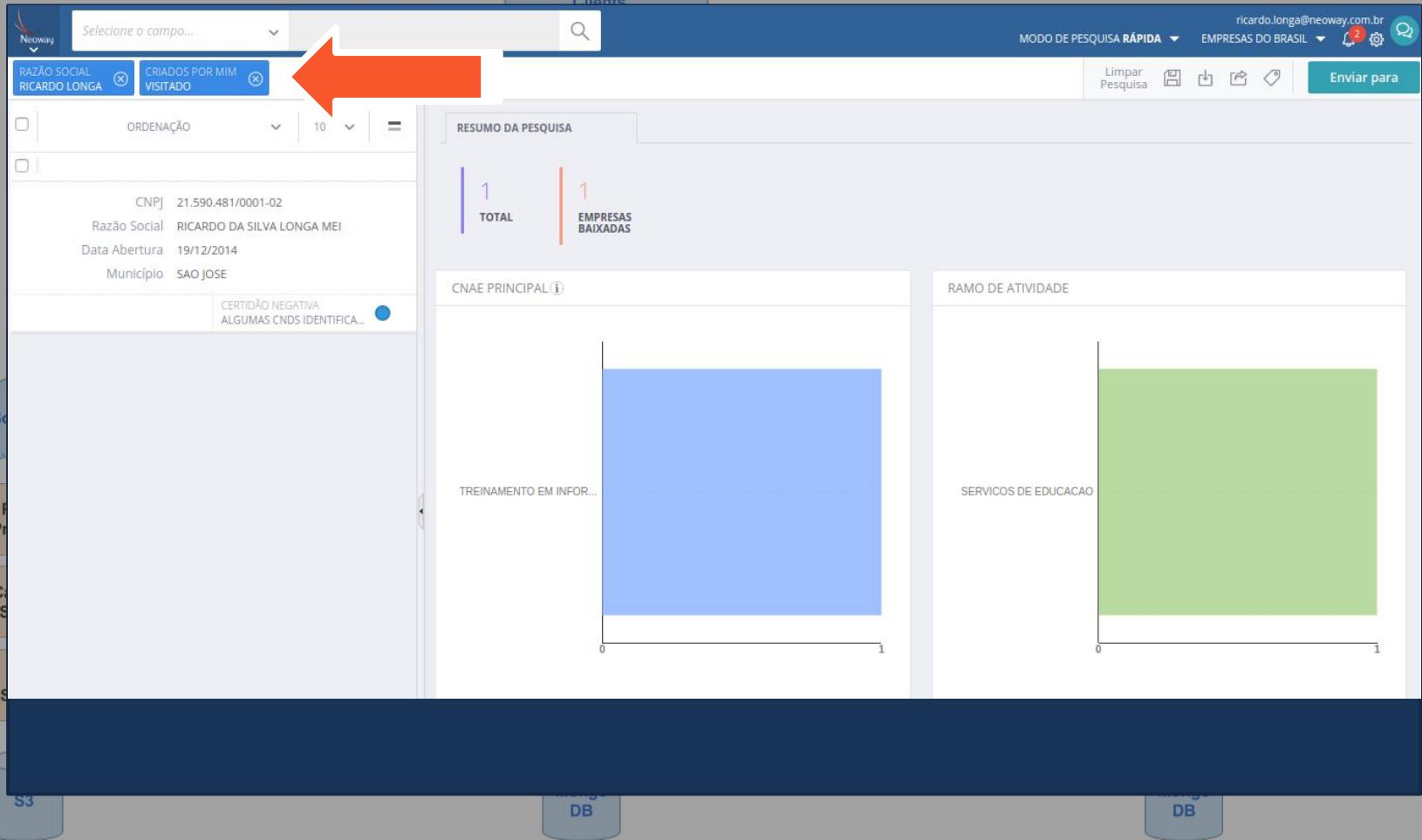
SERVICOS DE EDUCACAO



S3

DB

DB



Clients

Neoway Seleção o campo... MODO DE PESQUISA RÁPIDA ricardo.longa@newway.com.br EMPRESAS DO BRASIL Enviar para

RAZÃO SOCIAL RICARDO LONGA CRIADOS POR MIM VISITADO ORDENAÇÃO 10 =

CNPJ 21.590.481/0001-02
Razão Social RICARDO DA SILVA LONGA MEI
Data Abertura 19/12/2014
Município SAO JOSE

CERTIDÃO N ALGUMAS C

RESUMO DA PESQUISA

1 TOTAL | 1 EMPRESAS BAIXADAS

CNAE PRINCIPAL ⓘ

TREINAMENTO EM INFOR... SERVICOS DE EDUCACAO

RAMO DE ATIVIDADE

0 1 0 1

S3 DB DB

The screenshot shows a business search interface. At the top, there's a header with the Neoway logo, a search bar, and user information like 'ricardo.longa@newway.com.br'. Below the header, a summary box indicates 1 result found. The main content area displays a company profile with details such as CNPJ, Razão Social, Data Abertura, and Município. Two large orange arrows are overlaid on the screen: one pointing from the top-left towards the search bar, and another pointing from the bottom-left towards the company details section.

Clients

Neoway Selecone o campo... MODO DE PESQUISA RÁPIDA ricardo.longa@neoway.com.br EMPRESAS DO BRASIL 2 3

RAZÃO SOCIAL RICARDO LONGA X CRIADOS POR MIM VISITADO X

ORDENAÇÃO 10

RESUMO DA PESQUISA RICARDO DA SILVA LONGA MEI X

RICARDO DA SILVA LONGA MEI

CNPJ 21.590.481/0001-02
Razão Social RICARDO DA SILVA LONGA MEI
Data Abertura 19/12/2014
Município SAO JOSE

CERTIDÃO NEGATIVA ALGUMAS CNDS IDENTIFICADAS

NÍVEL DE ATIVIDADE INATIVA CERTIDÃO NEGATIVA ALGUMAS CNDS IDENTIFICADAS

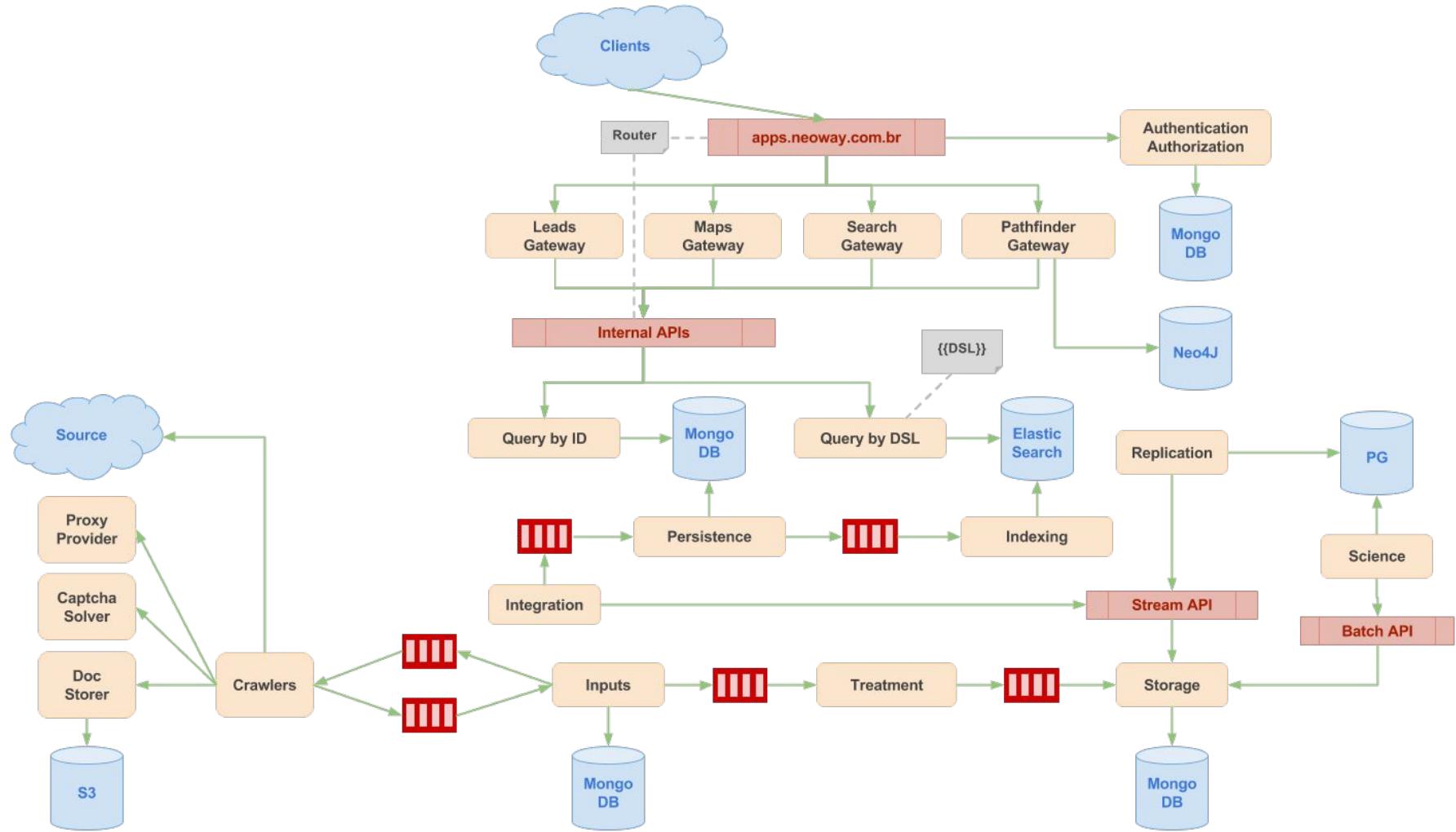
VISITADO X adicionar...

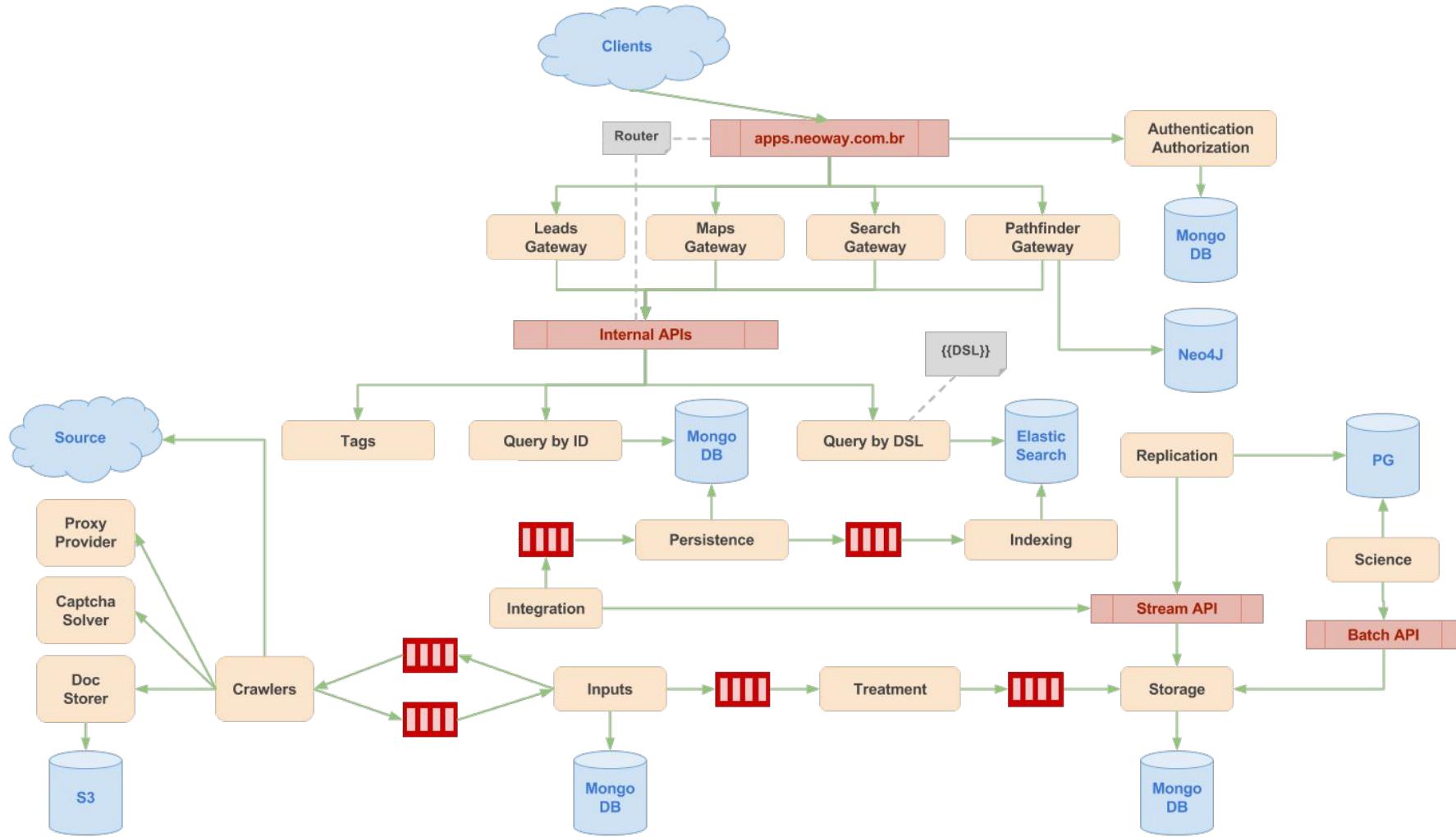
DADOS CADASTRAIS

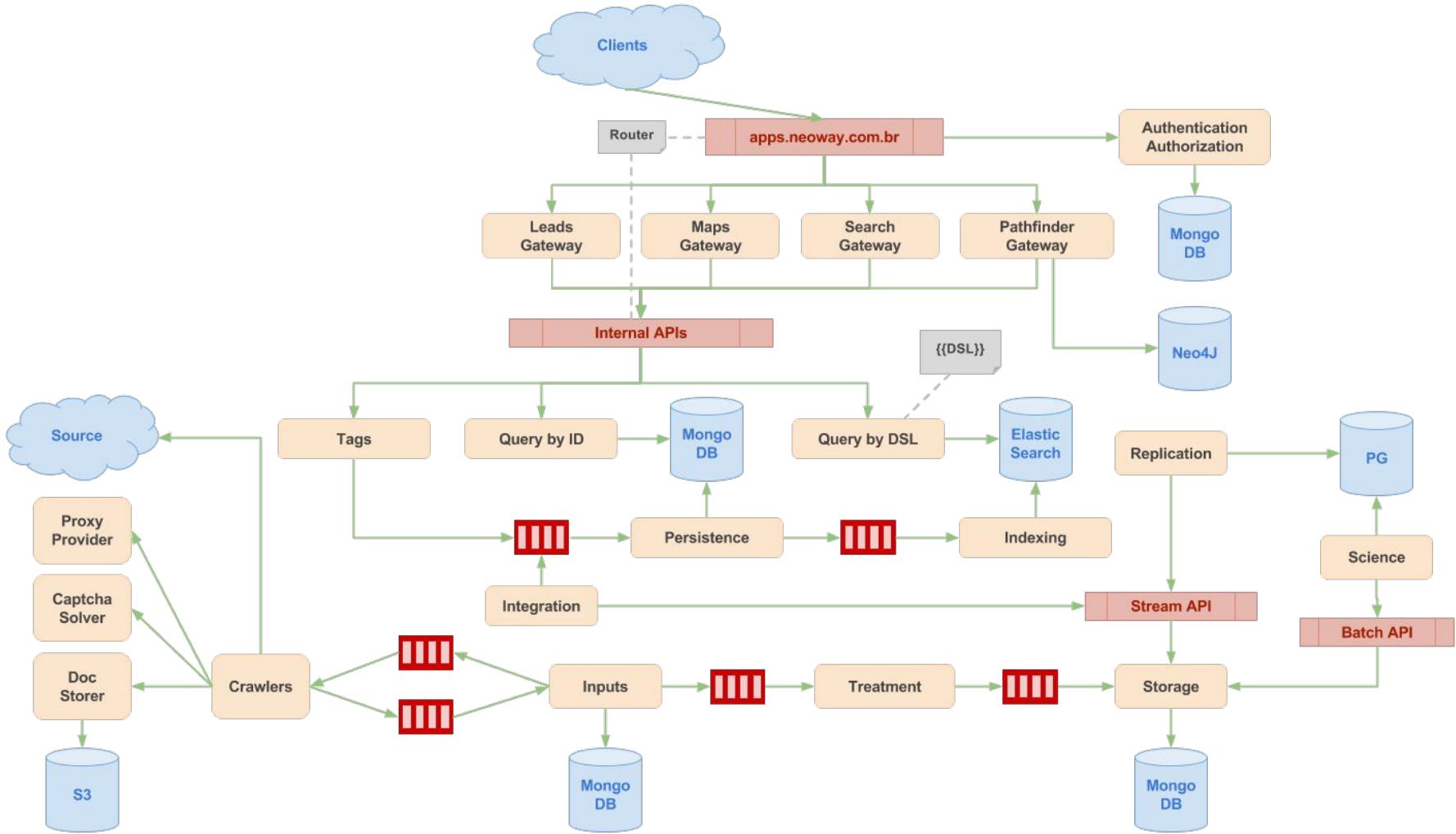
CNPJ	21.590.481/0001-02	Data Abertura	19/12/2014
Razão Social	RICARDO DA SILVA LONGA MEI	Idade Da Empresa	1
Nome Fantasia	LONGA CONSULTORIA E TREINAMENTOS	Identificador Do CNAE Principal	8599-6/03
Código Natureza Jurídica	2135	Descrição Do CNAE Principal	TREINAMENTO EM INFORMATICA
Descrição Natureza Jurídica	EMPRESARIO INDIVIDUAL	Setor	SERVICOS
Classificação Natureza Jurídica	PRIVADO	Ramo De Atividade	SERVICOS DE EDUCACAO
Situação Cadastral	BAIXADA	Nível De Atividade	INATIVA
Data Situação Cadastral	17/10/2016	Possui QSA Divergente	NÃO
É Matriz	SIM		

Histórico

The screenshot shows a business information search interface. At the top, there's a header with the Neoway logo, a search bar, and navigation links for 'MODO DE PESQUISA RÁPIDA' and 'EMPRESAS DO BRASIL'. Below the header, a search result for 'RICARDO DA SILVA LONGA MEI' is displayed. On the left, a sidebar shows basic company details: CNPJ 21.590.481/0001-02, Razão Social RICARDO DA SILVA LONGA MEI, Data Abertura 19/12/2014, and Município SAO JOSE. It also shows a 'CERTIDÃO NEGATIVA' status with the note 'ALGUMAS CNDS IDENTIFICADAS'. In the center, there are two circular icons: one grey with a magnifying glass labeled 'NÍVEL DE ATIVIDADE INATIVA' and one blue with a gear labeled 'CARTA DE IMPOSTOS'. Below these are sections for 'DADOS CADASTRAIS' containing detailed company information like name, address, and industry classification. At the bottom right, there's a 'Histórico' button. A prominent red arrow points from the bottom right towards the 'adicionar...' button next to the 'VISITADO' status indicator.





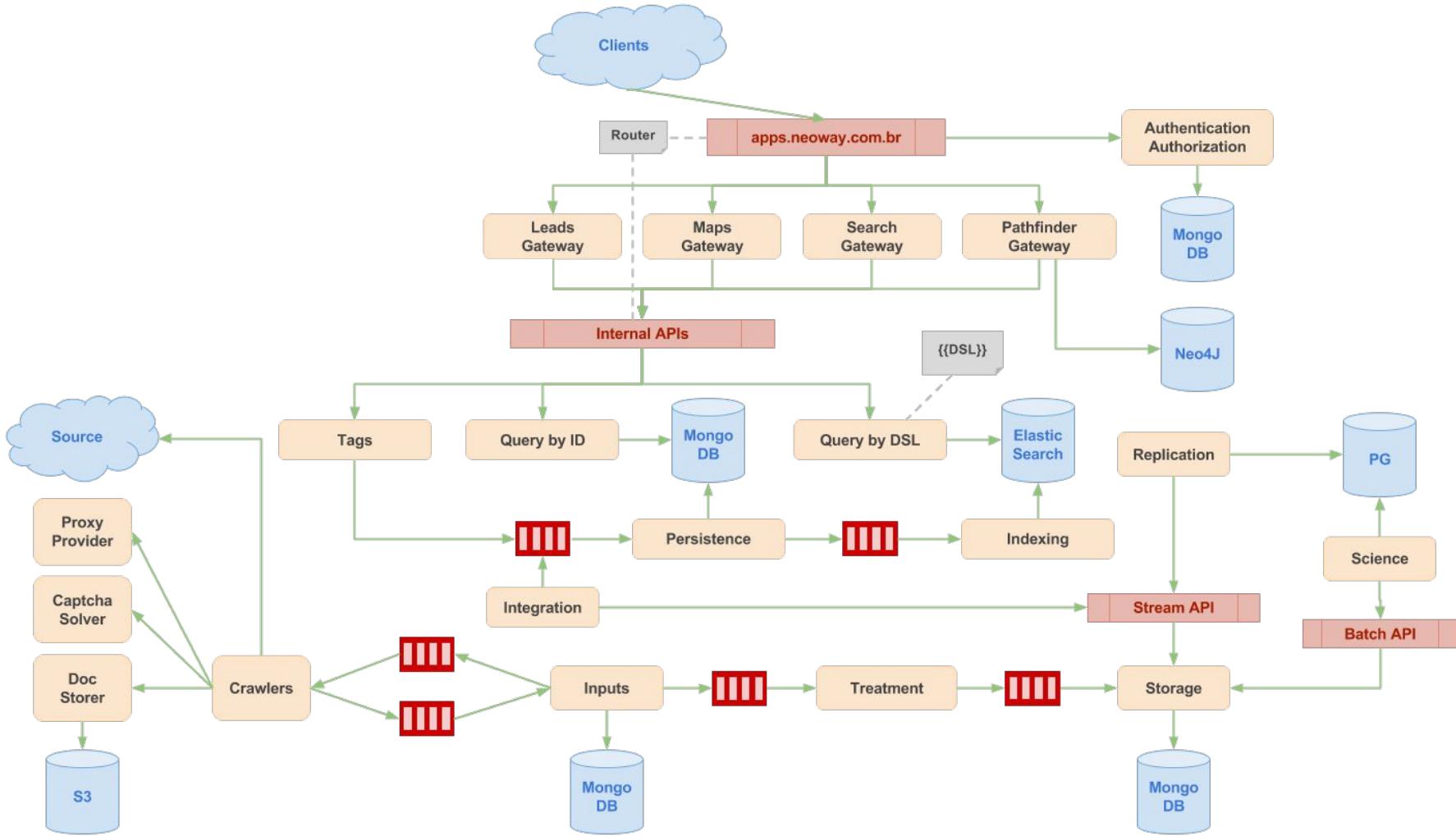


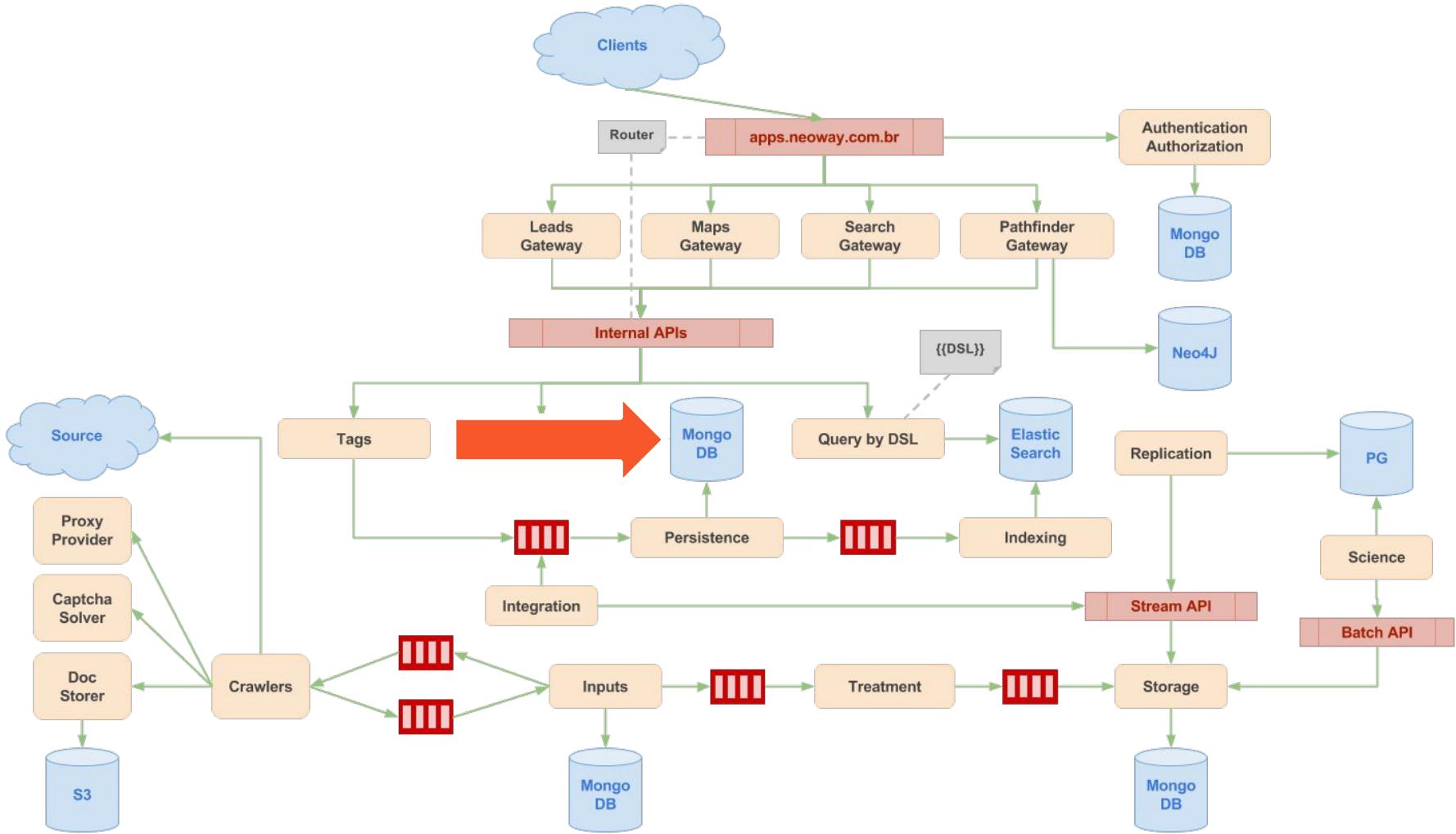
E se as filas
estiverem
congestionadas?

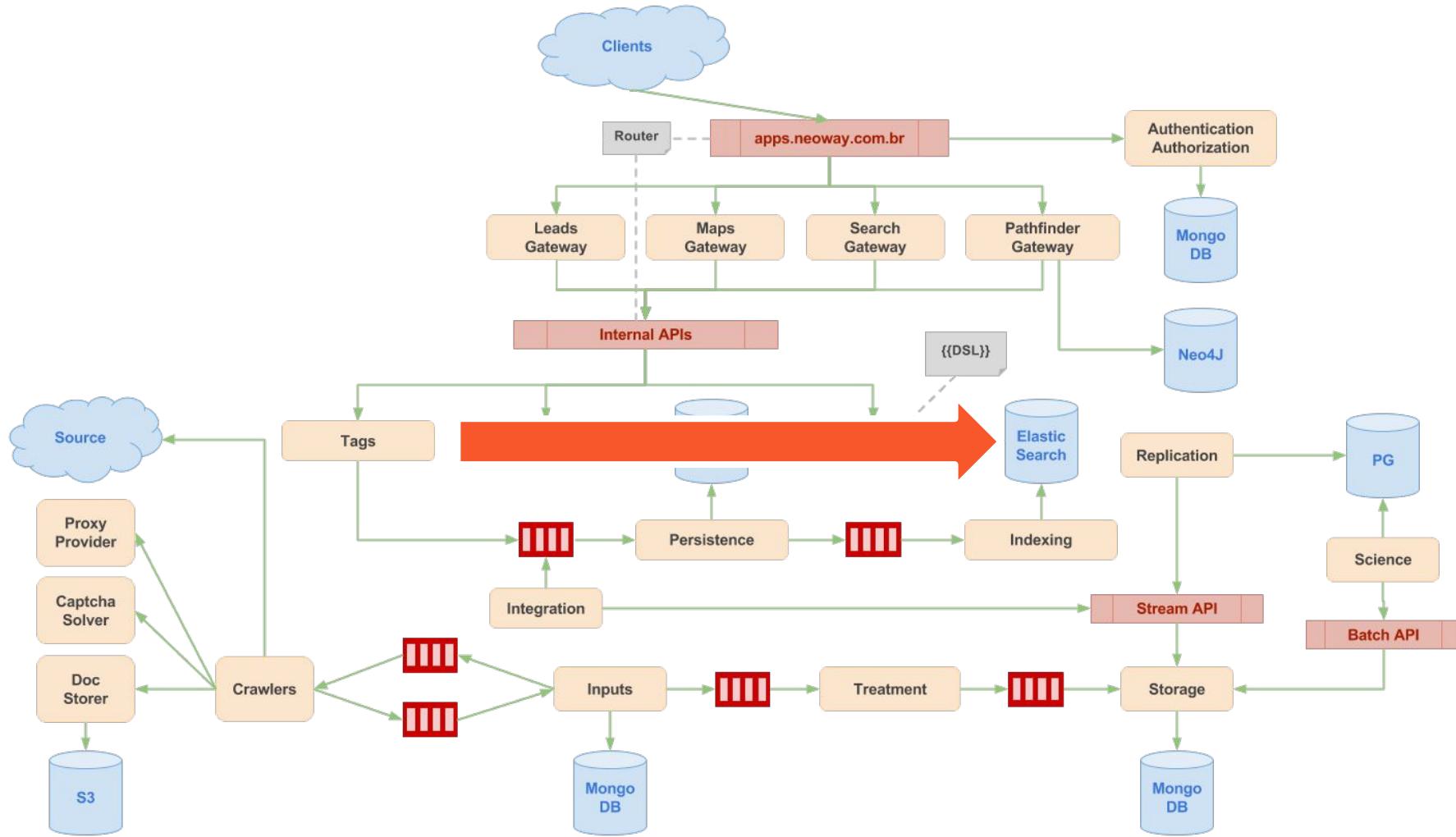
Processamento síncrono

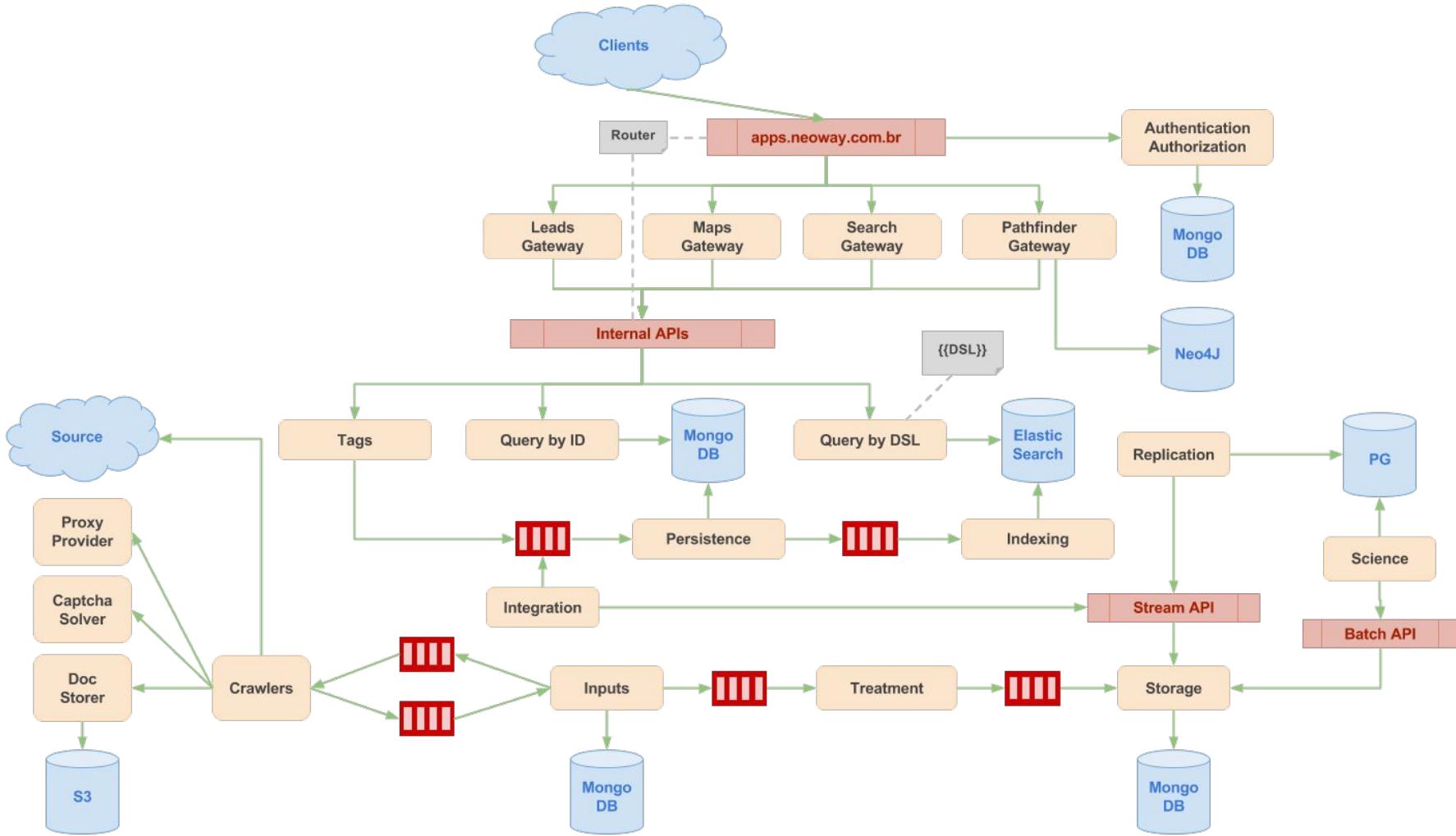
Requisito:

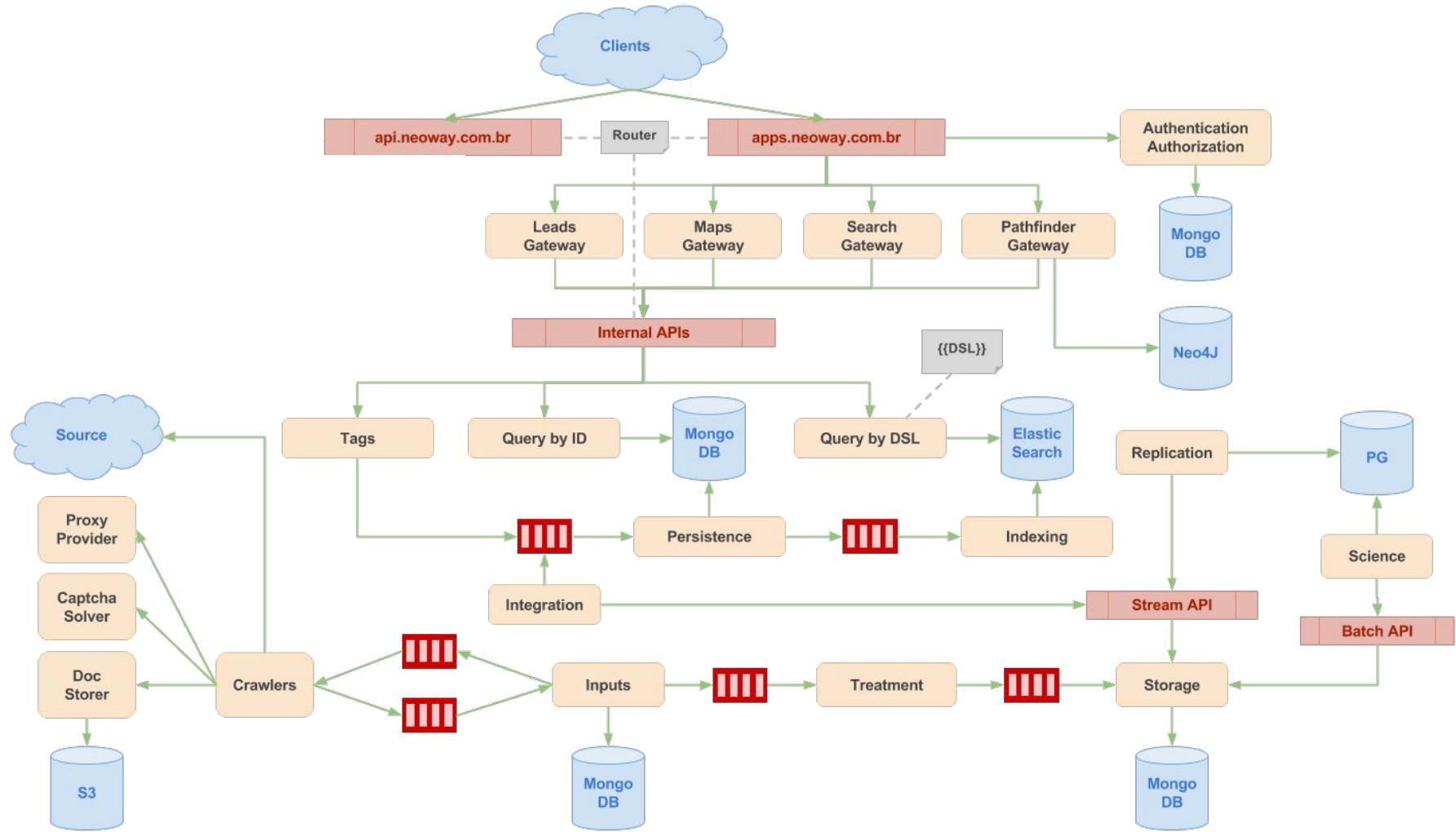
- Ao marcar uma empresa, a busca deve retorná-la imediatamente

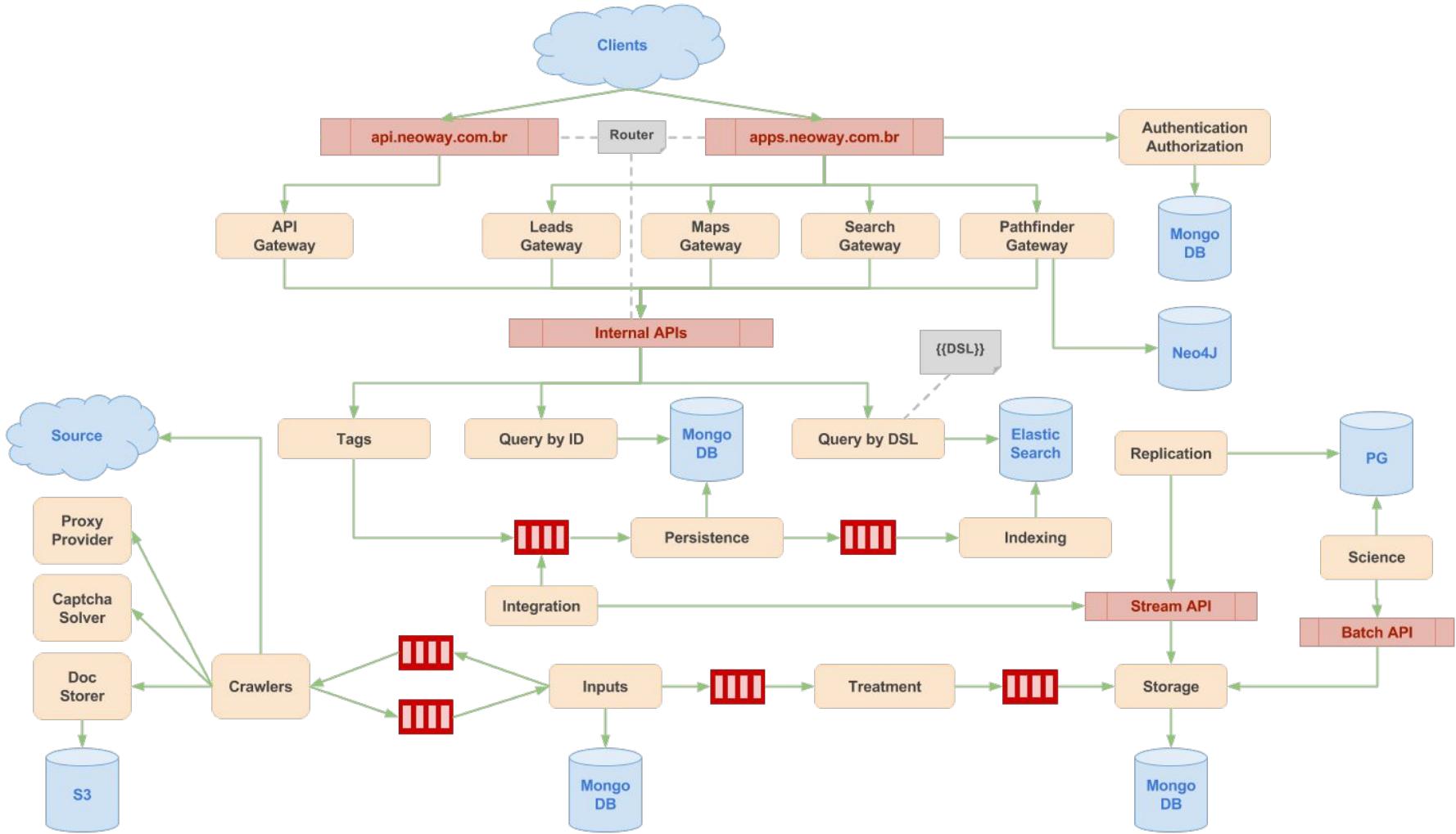


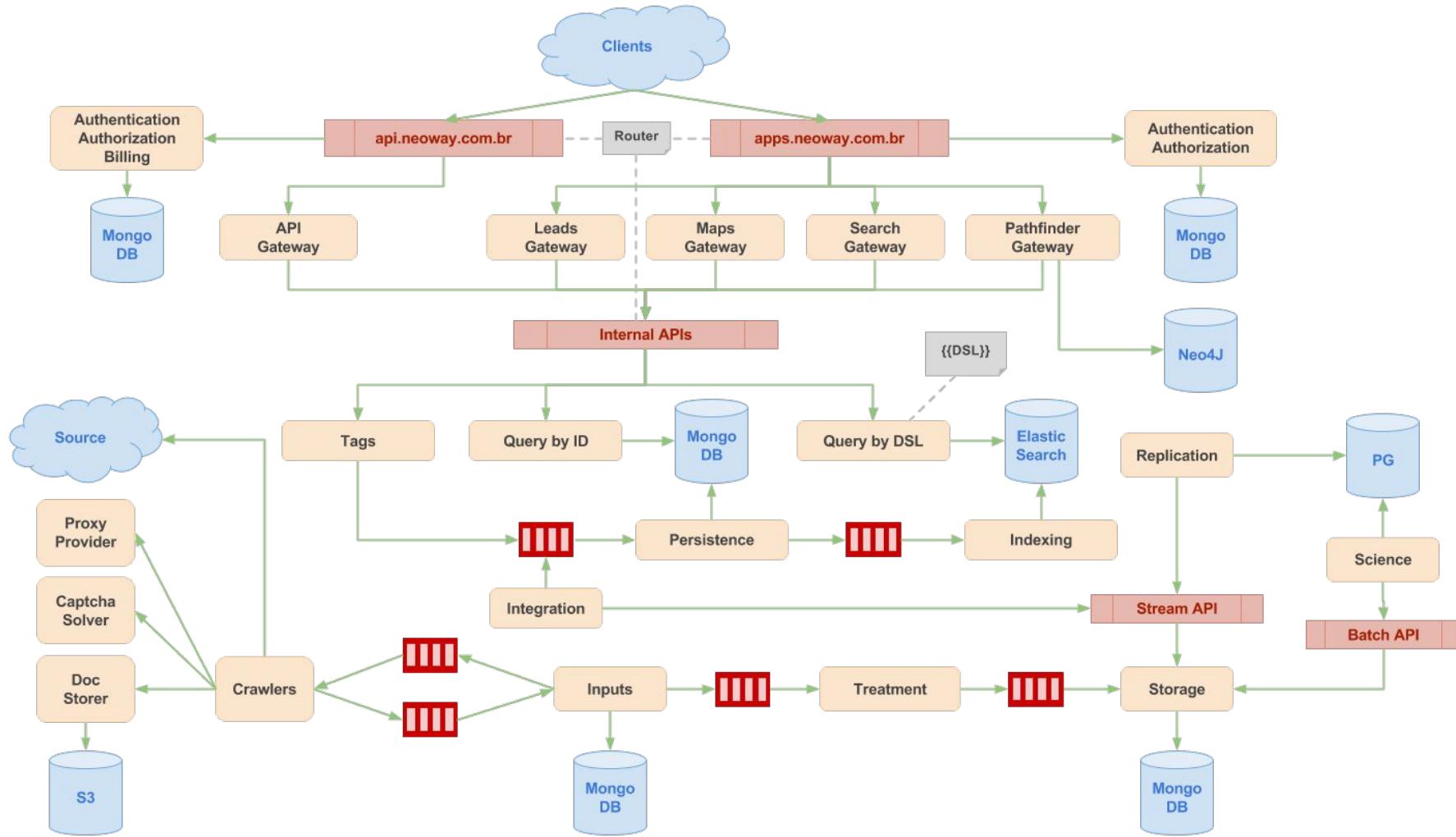












Json Web Tokens

Motivos:

- Sem sessões no servidor
- Informações necessárias trafegam dentro do token
- <https://jwt.io/>

The first step is to pick a signing method. For demonstration purposes we will choose HSMAC256.

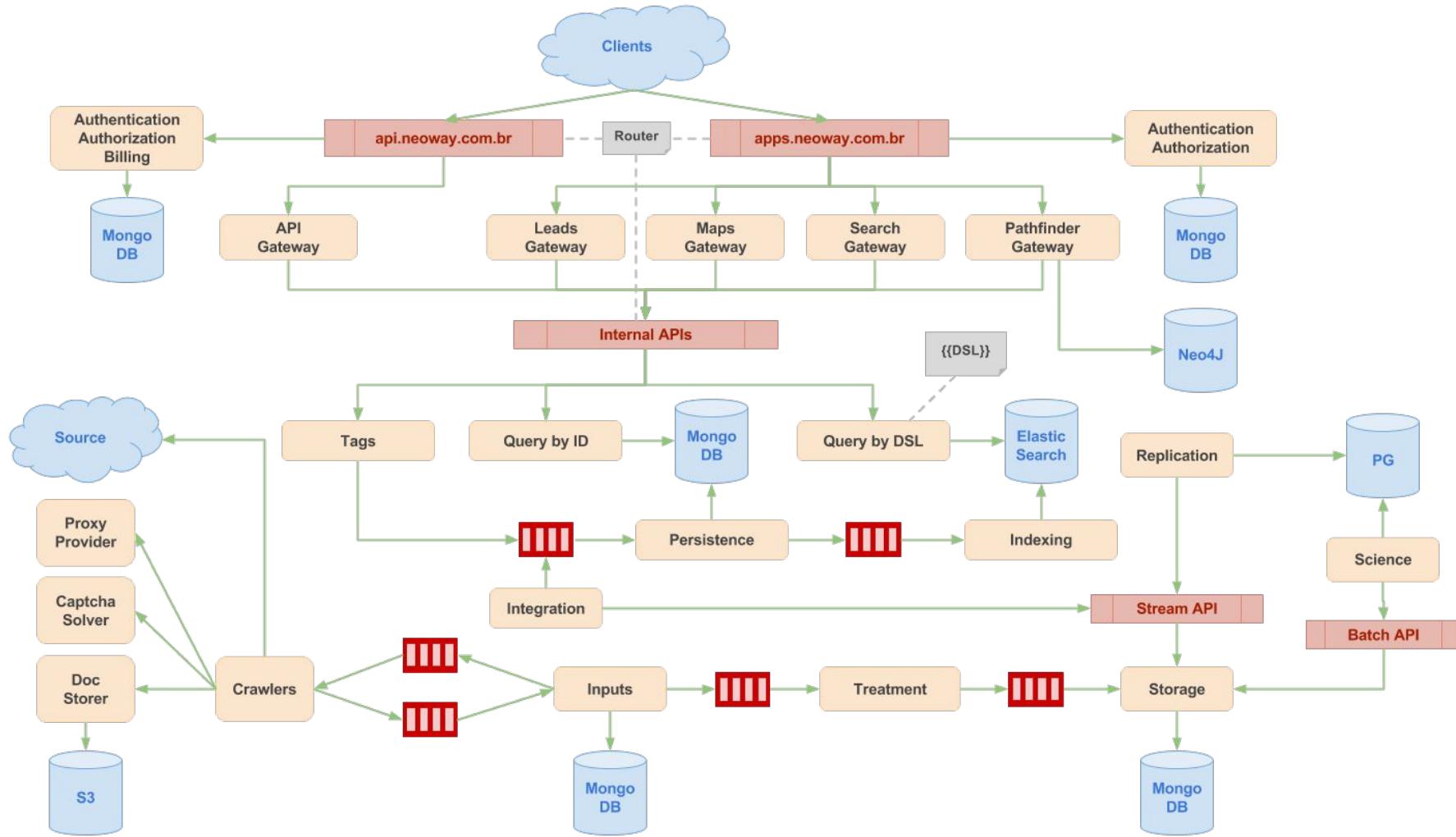
```
algorithm := jwt.HmacSha256("ThisIsTheSecret")
```

Now we need to the claims, and edit some values

```
claims := jwt.NewClaim()  
claims.Set("Role", "Admin")
```

Then we will need to sign it!

```
token, err := algorithm.Encode(claims)  
if err != nil {  
    panic(err)  
}
```



Clients

Cliente feliz!



S3

DB

DB

Scalability

Performance

Cost

Security

API

Aplicações

RAZÃO SOCIAL
NEOWAY XLimpar Pesquisa X X X X X

Enviar para

ORDENAÇÃO		10	=
Data Abertura	04/04/2016		
Município	PALHOCA		
CERTIDÃO NEGATIVA TODAS AS CNDS IDENTIFICA...			
5 USUÁRIOS			
CNPJ	05.337.875/0002-88		
Razão Social	NEOWAY TECNOLOGIA INTEGRADA ASSESSORIA E NEGOCIOS SA		
Data Abertura	02/02/2011		
Município	PALHOCA		
CERTIDÃO NEGATIVA TODAS AS CNDS IDENTIFICA...			
5 USUÁRIOS			
CNPJ	05.337.875/0003-69		
Razão Social	NEOWAY TECNOLOGIA INTEGRADA ASSESSORIA E NEGOCIOS SA		
Data Abertura	05/08/2014		
Município	SAO PAULO		
CERTIDÃO NEGATIVA TODAS AS CNDS IDENTIFICA...			
5 USUÁRIOS			
CNPJ	05.337.875/0001-05		
Razão Social	NEOWAY TECNOLOGIA INTEGRADA ASSESSORIA E NEGOCIOS SA		
Data Abertura	15/10/2002		

RESUMO DA PESQUISA NEOWAY TECNOLOGIA ... X

NEOWAY TECNOLOGIA INTEGRADA ASSESSORIA E NEGOCIOS SA

FATURAMENTO ESTIMADO
DE R\$ 30.000.000,01 A
R\$ 100.000.000,00FUNCIONÁRIOS
DE 101 A 500NIVEL DE ATIVIDADE i
ALTACERTIDÃO NEGATIVA
TODAS AS CNDS
IDENTIFICADASEXPORTADO NEOWAY SUPORTE NW 01 adicionar...

DADOS CADASTRAIS

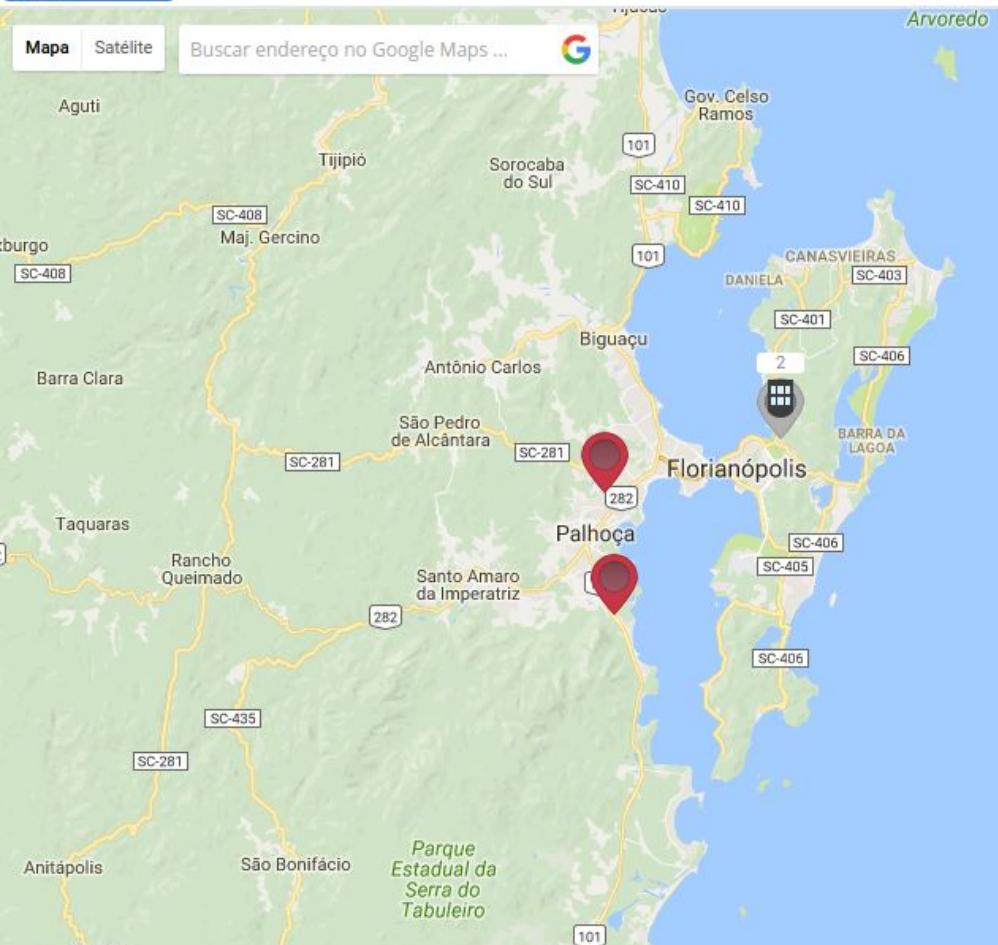
CNPJ	05.337.875/0001-05	Data Abertura	15/10/2002
Razão Social	NEOWAY TECNOLOGIA INTEGRADA ASSESSORIA E NEGOCIOS SA	Idade Da Empresa	14
Nome Fantasia	NEOWAY BUSINESS SOLUTIONS	Identificador Do CNAE Principal	6204-0/00
Código Natureza Jurídica	2054	Descrição Do CNAE Principal	CONSULTORIA EM TECNOLOGIA DA INFORMACAO
Descrição Natureza Jurídica	SOCIEDADE ANONIMA FECHADA	Setor	SERVICOS
Classificação Natureza Jurídica	PRIVADO	Ramo De Atividade	INDUSTRIA DIGITAL
Situação Cadastral	ATIVA	Número De Filiais Ativas	3
Data Situação	15/10/2002	Nível De Atividade	ALTA
Possui OSA Divergente	NÃO		

RAZÃO SOCIAL
NEOWAY XLimpar
Pesquisa

Enviar para

Mapa Satélite

Buscar endereço no Google Maps ...



Resumo Camadas

7 Total de empresas

243 Total funcionários

Salvar pesquisa como camada

FAIXA DE FATURAMENTO

- 1 A 60K
- 60K A 360K
- 360K A 1M
- 1M A 5M
- 5M A 10M
- 10M A 30M
- 30M A 100M
- 100M A 300M
- 300M A 500M
- 500M A 1BI
- APARTIR DE 1BI

0

FAIXA DE FUNCIONÁRIOS

- ATÉ 5

DE 6 A 10

RAZÃO SOCIAL
NEOWAY XLimpar
Pesquisa

Enviar para



VISÃO GERAL

20
CAIXA DE ENTRADA9
LEADS NÃO VISUALIZADO2
LEADS TRANSFERIDOS7
LEADS DESCARTADOS6
OPORTUNIDADES

FASES DA OPORTUNIDADE

CRIAÇÃO DA OPORTUNIDADE

3

REALIZAR CONTATO

2

ENVIO DA PROPOSTA

1

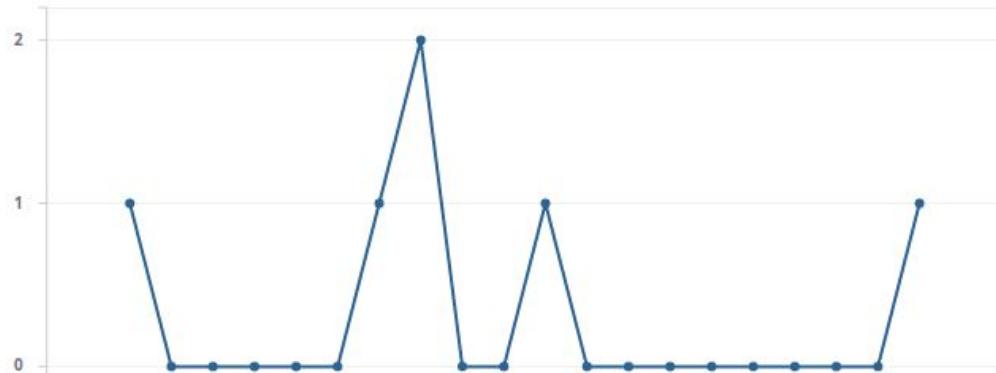
REGISTRO DA VENDA

0

STATUS DAS OPORTUNIDADES



HISTÓRICO DAS OPORTUNIDADES



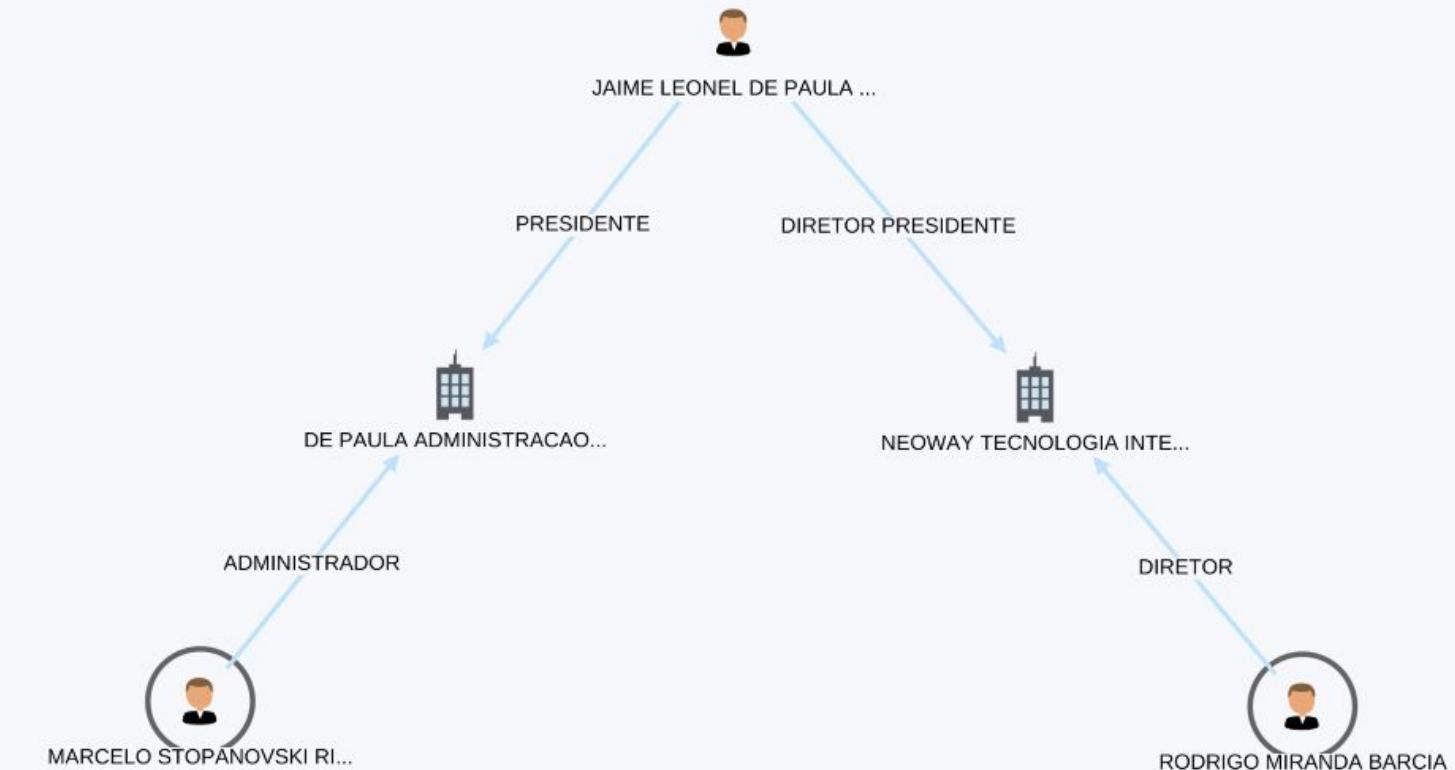


Caminho Mais Curto

RODRIGO MIRANDA BARCIA - 00791777944

MARCELO STOPANOVSKI RIBEIRO - 896526575

ricardo.longa@neoway.com.br



DevOps

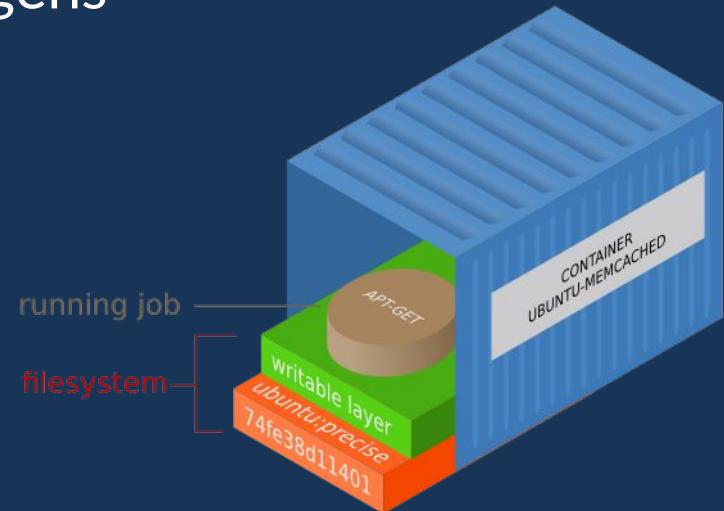
Infraestrutura

- 200+ crawlers capturando dados
- 70+ microservices
- 200+ instâncias no EC2
- 100+ instâncias na Azure



Linux Containers

- Empacotamento de dependências em “containers”
- Permite versionamento das imagens
- Docker compose
- Rkt (uma alternativa minimalista ao Docker)
- Feito em Go

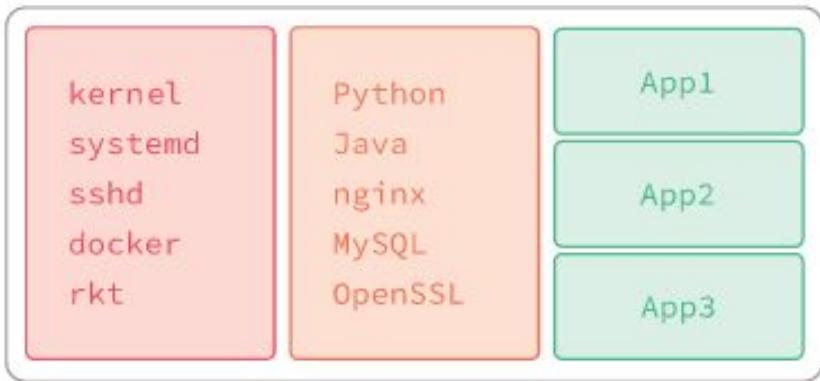


CoreOS



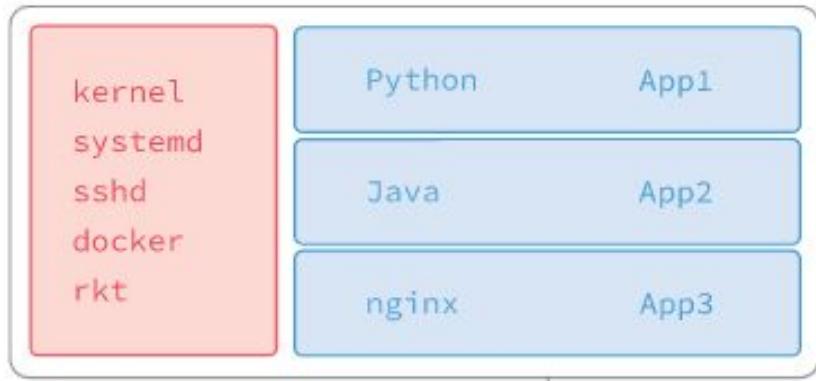
- Distro minimalista
- Focado em containers (Docker/Rkt)
- ETCD - Database (key/value) distribuído
- Rolling updates (facilita atualização de +300 instâncias)
- Cloud Config (provisionamento nativo)

Traditional Distro



CoreOS

Your Containers



required software only

Systemd Units

```
system PRODUCTION # cat tyr.service
[Unit]
After=skydns.service
Before=tyr-announce.service

[Service]
EnvironmentFile=/etc/profile.d/core
ExecStart=/usr/bin/rkt --insecure-options=all run --net=host --no-store=true \
           --inherit-env \
           --volume tmp,kind=host,source=/data/tyr \
           ${TYR_IMAGE} -- \
           -port=${TYR_PORT} \
           -etcdKey=tyr \
           -etcd=http://127.0.0.1:4001
```

Environment Variables

```
system PRODUCTION # cat /etc/profile.d/core
#!/bin/sh

# Globals
GIN_MODE=release
LOGGER_LEVEL=error
LOGGER_OUTPUT=syslog
LOGGER_SYSLOG_ADDRESS=log.neoway.local:1514/udp
LOGGER_SYSLOG_DIR=core
SYSLOG_ADDRESS=log.neoway.local:1514

# Backends
MONGO_URL=mongo1.core.neoway.local:27017,mongo2.core.neoway.local:27017
ELASTICSEARCH_URL="http://x1.elasticsearch.core.neoway.local:9200,http://x2.elasticsearch.core.ne
oway.local:9200,http://x3.elasticsearch.core.neoway.local:9200"
```

Service Discovery

“SkyDNS is a distributed service for announcement and discovery of services built on top of etcd.”

Service Discovery

```
system PRODUCTION # cat tyr-announce.service
[Unit]
Description=Announce tyr
BindsTo=tyr.service

[Service]
EnvironmentFile=/etc/environment
EnvironmentFile=/etc/profile.d/core
ExecStart=/bin/sh -c "while true; do etcdctl set /skydns/local/neoway/core/api/%m '{\"host\": \"'${COREOS_PRIVATE_IPV4}'\"}' --ttl 15 ;sleep 10;done"
ExecStop=/usr/bin/etcdctl rm /skydns/local/neoway/core/api/%m

[Install]
WantedBy=multi-user.target
```

Service Discovery

```
system PRODUCTION # etcdctl ls --recursive /skydns/local/neoway/core/api  
/skydns/local/neoway/core/api/ 4b142d1ced0241a4bad59d4273f65a2e  
/skydns/local/neoway/core/api/ 1104a38250cf48f8bb8daefc25bc40c0
```

```
system PRODUCTION # host api.core.neoway.local  
api.core.neoway.local has address 10.11.12.95  
api.core.neoway.local has address 10.11.11.219
```

Kubernetes



- Gerenciador de containers em cluster
- Criado pelo Google - Escrito em Go
- Fácil de escalar (cluster e serviços)
- Service discovery
- Atualização com zero downtime - Rolling Updates

Admin

Namespaces

Nodes

Persistent Volumes

Namespace

default ▾

Workloads

Deployments

Replica Sets

Replication Controllers

Daemon Sets

Pet Sets

Jobs

Pods

Services and discovery

Services

Ingress

Storage

Persistent Volume Claims

Config

	Name	Labels	Pods	Age	Images	
	✓ celesc	arch: ppd1 name: celesc type: bot	1 / 1	3 months	registry-aws.neoway.com.br/d...	⋮
	✓ cfc-empresas	arch: ppd1 name: cfc-empresas type: bot	1 / 1	3 months	registry-aws.neoway.com.br/d...	⋮
	✓ cfc-profissionais	arch: ppd1 name: cfc-profissionais type: bot	1 / 1	3 months	registry-aws.neoway.com.br/d...	⋮
	✓ cfm	arch: ppd2 name: cfm type: bot	5 / 5	3 months	registry-aws.neoway.com.br/d...	⋮
	✓ cfo	arch: ppd1 name: cfo type: bot	1 / 1	3 months	registry-aws.neoway.com.br/d...	⋮
	✓ cnd-fgts	arch: ppd2 name: cnd-fgts type: bot	10 / 10	2 months	registry-aws.neoway.com.br/d...	⋮
	✓ cnd-fgts-ppd1	arch: ppd1 name: cnd-fgts-ppd1 type: bot	4 / 4	3 months	registry-aws.neoway.com.br/d...	⋮
	✓ cnd-pgfn-cnpj	arch: ppd2 name: cnd-pgfn-cnpj type: bot	10 / 10	2 months	registry-aws.neoway.com.br/d...	⋮
	✓ cnd-pgfn-cnpj-2-via	arch: ppd1 name: cnd-pgfn-cnpj-2-via type: bot	4 / 4	3 months	registry-aws.neoway.com.br/d...	⋮
	✓ cnd-pgfn-cnpj-ppd1	arch: ppd1 name: cnd-pgfn-cnpj-ppd1	1 / 1	2 months	registry-aws.neoway.com.br/d...	⋮

Monitoramento

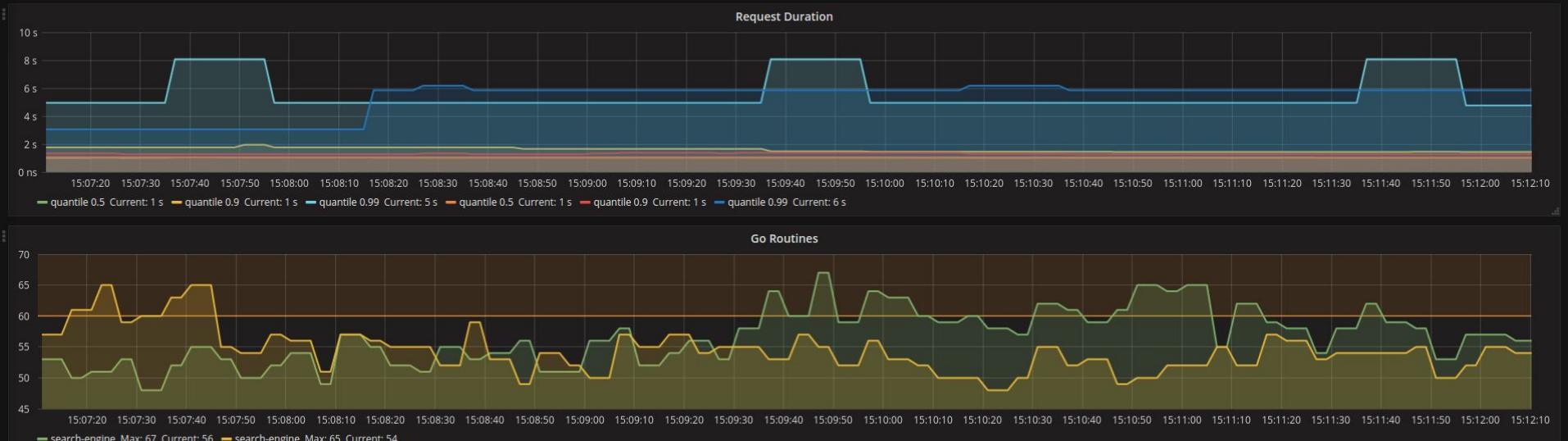


Prometheus + Grafana

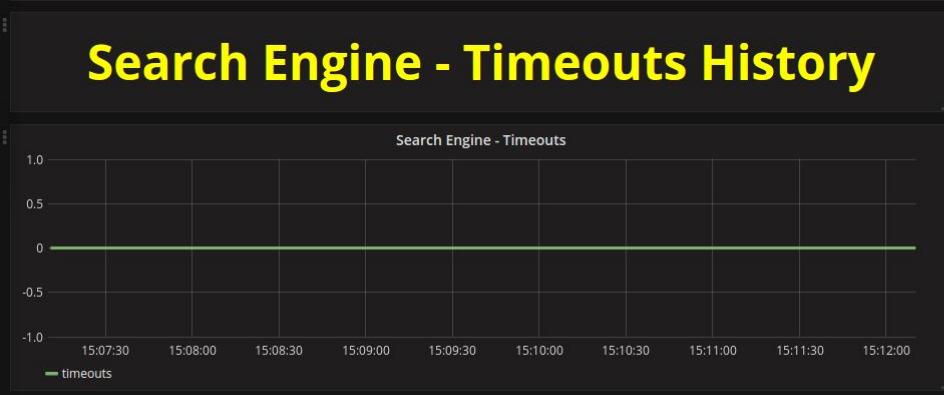


- Prometheus busca as métricas (pull based)
- <https://github.com/google/cadvisor>
- https://github.com/prometheus/node_exporter
- Armazena em um time series database
- Permite configuração de alertas

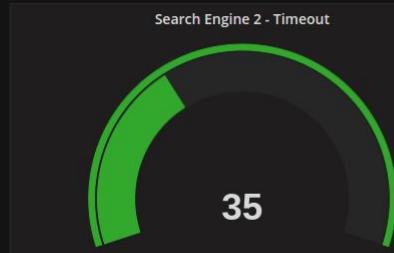
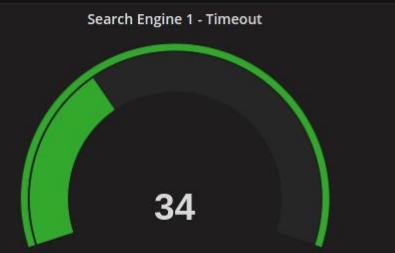
Search Engine - Request Duration and Go Routines



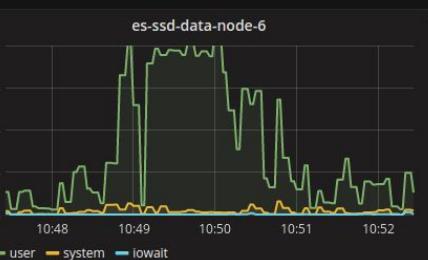
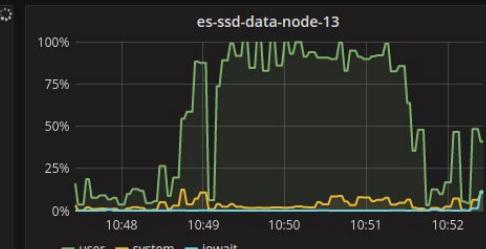
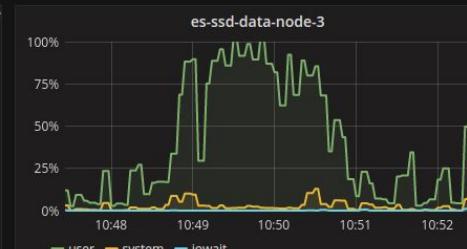
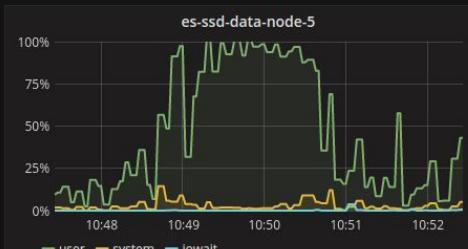
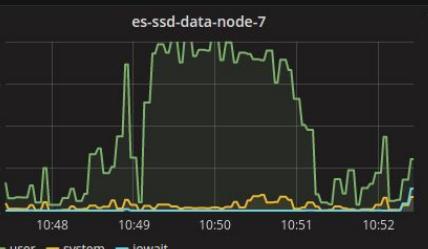
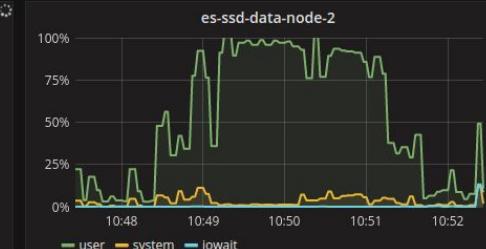
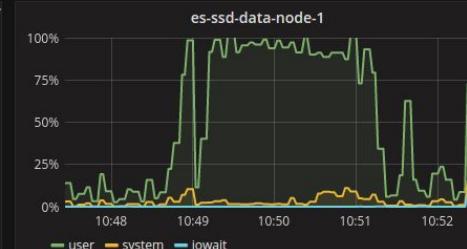
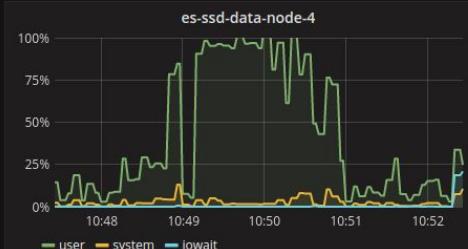
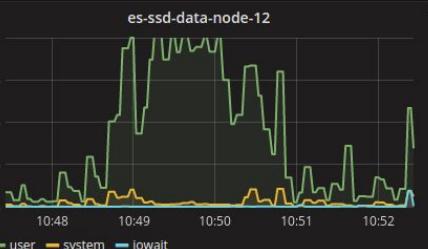
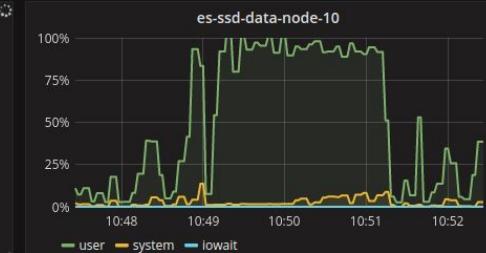
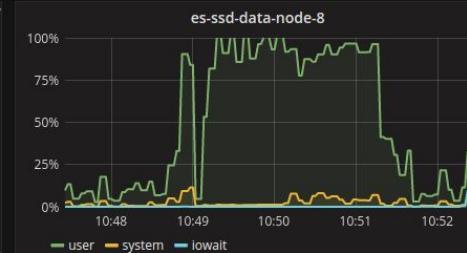
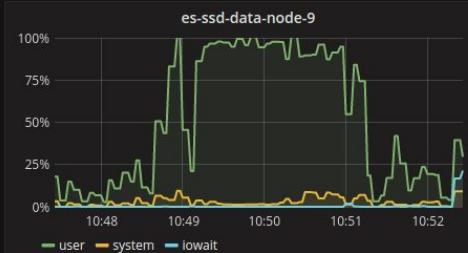
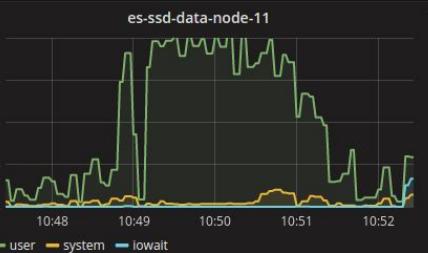
Search Engine - Timeouts History



Search Engine - Timeout Count



+ ADD ROW





core_front_services

All

sales_services

All

pathfinder_services

All

core_services

All

deployment

All

Core Front

heimdall
uploki
upswitch
upbifrost
up

Core API

tyr up	es-ssd-data-node-6 up	eir up	skeeper up	mongo1 up	nsqlookup up	barwoman up	clientdata up	accounts up	leads-indkser up	mongo-backup up	es-ssd-data-node-4 up
es-ssd-data-node-7 up	search-engine up	es-ssd-data-node-10 up	audit up	nsqd up	es-ssd-data-node-12 up	es-ssd-data-node-8 up	es-ssd-data-node-5 up	dory up	mongo2 up	njord up	es-ssd-data-node-2 up
webhook up	domain-metadata up	es-ssd-data-node-13 up	es-ssd-data-node-3 up	config up	bifrost up	thor up	es-ssd-data-node-11 up	export up	es-ssd-data-node-9 up	indkser up	es-ssd-data-node-1 up
sif up											

Sales Suite

mapslayers up	admin-webapp up	carteira-ui up	connector-uploader up	leads-schedule up	dynamics-crm-ws up	search-cluster up	leads-api up	admin-gateway up	dashboard-api up	leads-searchbar-ui up	connector-gateway up
crmin-webapp up	book-webapp up	crmin-worker up	leads-searchbar-api up	leads-schemas up	dashboard-webapp up	crmin-server up	leads-domains up	track up	mapuinotify up	connector-webapp up	leads-webapp up



heimdall
up

loki
up

switch
up

bifrost
up

Core API

Sales Suite

Neoway Open-source Projects

- <https://github.com/NeowayLabs>
 - **nash:** /nash
 - **klb:** /klb



Estamos
contratando!

CARREIRAS

In Technology We Trust.

A **NEOWAY** ESTÁ DEFININDO O FUTURO DOS NEGÓCIOS NO PAÍS.



DADOS

Desenvolvemos um conjunto de aplicações para coleta, visualização e análise estratégica das informações de 29 milhões de empresas.



DESAFIOS

Utilizamos tecnologias inovadoras para resolver nossos desafios. Usabilidade, escalabilidade, data analysis e social network analysis fazem parte do nosso dia a dia.



CLIENTES

Clientes nos segmentos mais importantes da economia, entre eles empresas multinacionais dos mercados Financeiro, Construção Civil, Petróleo e Gás, Comunicações e outros.

Apoio técnico! :)



Obrigado! :)

Kamila Hinckel

kamila.hinckel@neoway.com.br



kamilash

Ricardo Longa

ricardo.longa@neoway.com.br



@ricardolonga



ricardolonga



Projetos reativos com Angular, RxJS e Redux (ngRx)

Loiane Groner



github.com/loiane

loiane.com

loiane.training

QCon
SÃO PAULO

10+ XP TI

**Java, JavaScript, Sencha, Angular,
Phonegap/Ionic**

Blog: <http://loiane.com>

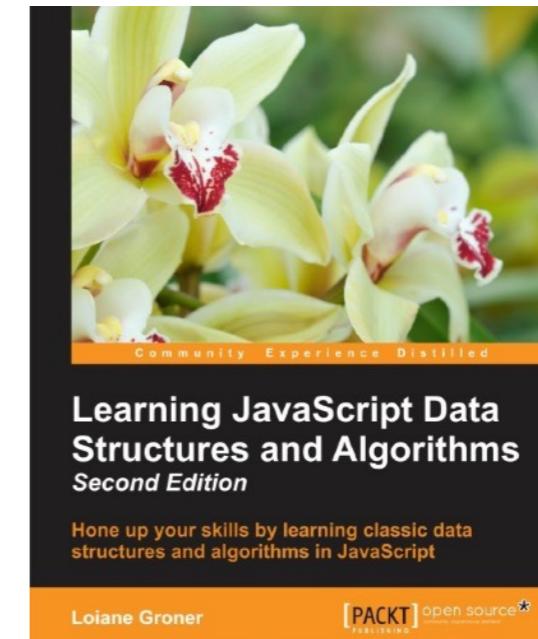
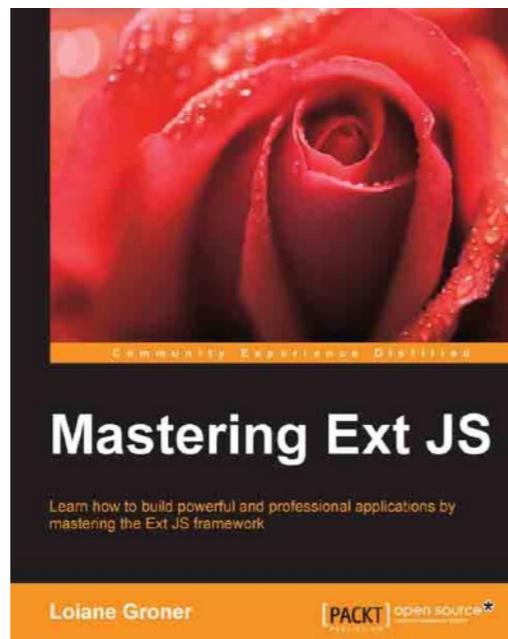
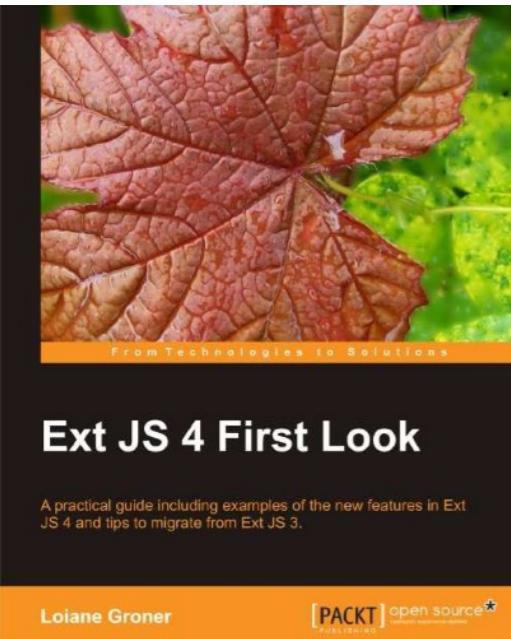
Cursos: <http://loiane.training>

*Andre
noel*



**LOIANE
GRONER**

Meus livros:



Disponível (inglês) na amazon.com.br

Também em português!!!



<https://novatec.com.br/livros/estruturas-de-dados-algoritmos-em-javascript/>



Entender como
desenvolver
funcionalidades
reativas com Angular

#itsJustAngular

- Programação reativa
- Promises x Observables
- Padrão Observer
- Angular reativo I: com forms, http, pipes

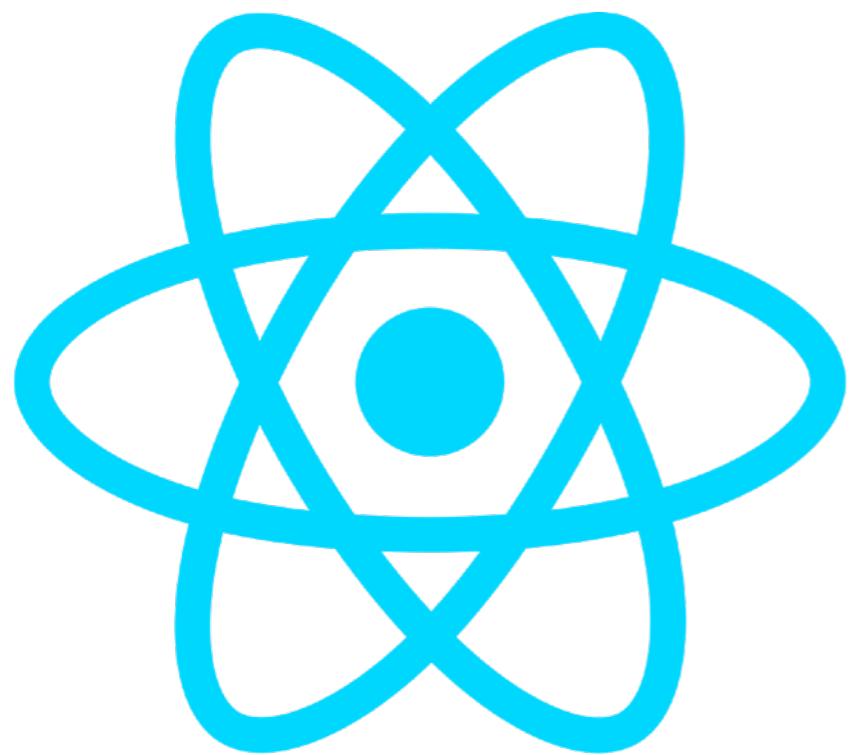
Agenda

- Desafios de projetos grandes e complexos
- Angular reativo II: redux com @ngrx/store
- Operações Imutáveis
- Change detection
- Arquitetura escalável



Programação reativa





React



Manifesto Reativo

<http://www.reactivemanifesto.org/pt-BR>



Reativo

“que reage, responde a estímulos”

Seus componentes são ativos e estão sempre prontos para receber eventos



Reagir a informações



Reagir a usuários



Reagir a erros

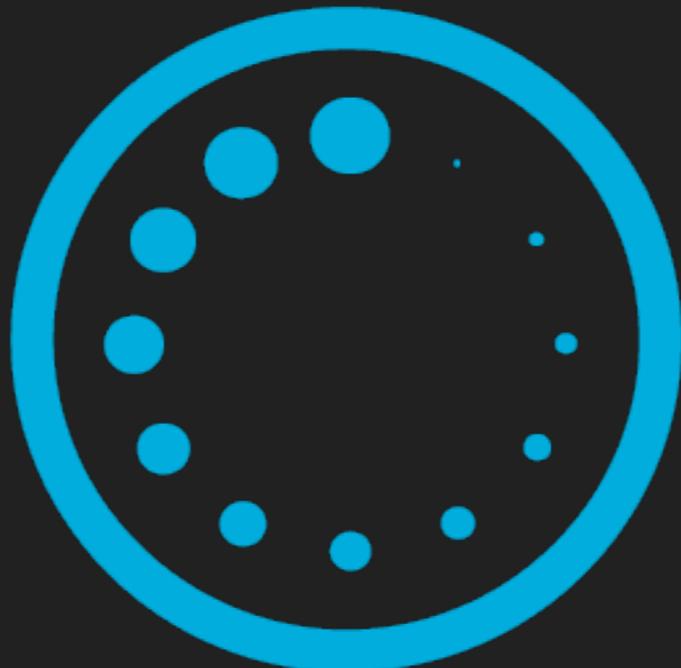


Já fazemos programação reativa

```
$('p').click(function() {  
  $(this).slideUp();  
});
```



Async





Promises

```
img.ready().then(function() {  
  // imagem carregada  
}, function() {  
  // erro  
});
```



Uma promessa pode ser:

- **atendida** - a ação relacionada à promessa teve sucesso
- **rejeitada** - a ação relacionada à promessa falhou
- **pendente** - a ação ainda não foi atendida nem rejeitada
- **definida** - a ação foi atendida ou rejeitada



```
1 function hell(win) {
2   // for listener purpose
3   return function() {
4     loadLink(win, REMOTE_SRC+'/assets/css/style.css', function() {
5       loadLink(win, REMOTE_SRC+'/lib/async.js', function() {
6         loadLink(win, REMOTE_SRC+'/lib/easyXDM.js', function() {
7           loadLink(win, REMOTE_SRC+'/lib/json2.js', function() {
8             loadLink(win, REMOTE_SRC+'/lib/underscore.min.js', function() {
9               loadLink(win, REMOTE_SRC+'/lib/backbone.min.js', function() {
10              loadLink(win, REMOTE_SRC+'/dev/base_dev.js', function() {
11                loadLink(win, REMOTE_SRC+'/assets/js/deps.js', function() {
12                  loadLink(win, REMOTE_SRC+'/src/' + win.loader_path + '/loader.js', function() {
13                    async.eachSeries(SCRIPTS, function(src, callback) {
14                      loadScript(win, BASE_URL+src, callback);
15                    });
16                  });
17                });
18              });
19            });
20          });
21        });
22      });
23    });
24  });
25};
26 }
```



Callback Hell!

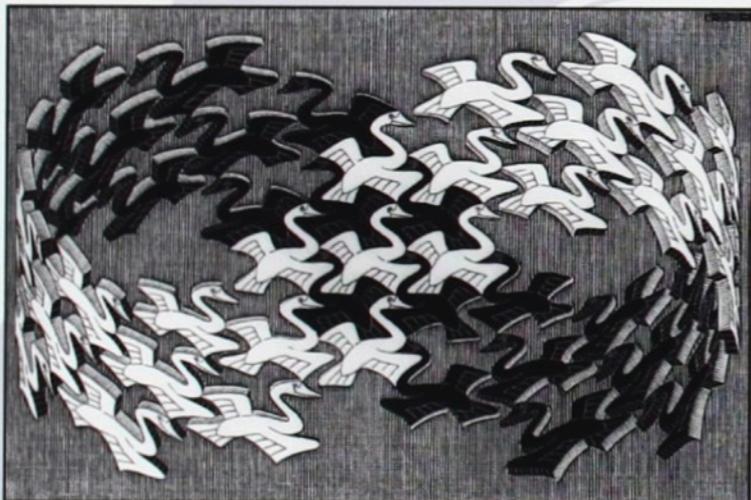
Observables



Design Patterns

Elements of Reusable
Object-Oriented Software

Erich Gamma
Richard Helm
Ralph Johnson
John Vlissides



Cover art © 1994 M.C. Escher / Cordon Art - Baarn - Holland. All rights reserved.

Foreword by Grady Booch

* ADDISON-WESLEY PROFESSIONAL COMPUTING SERIES



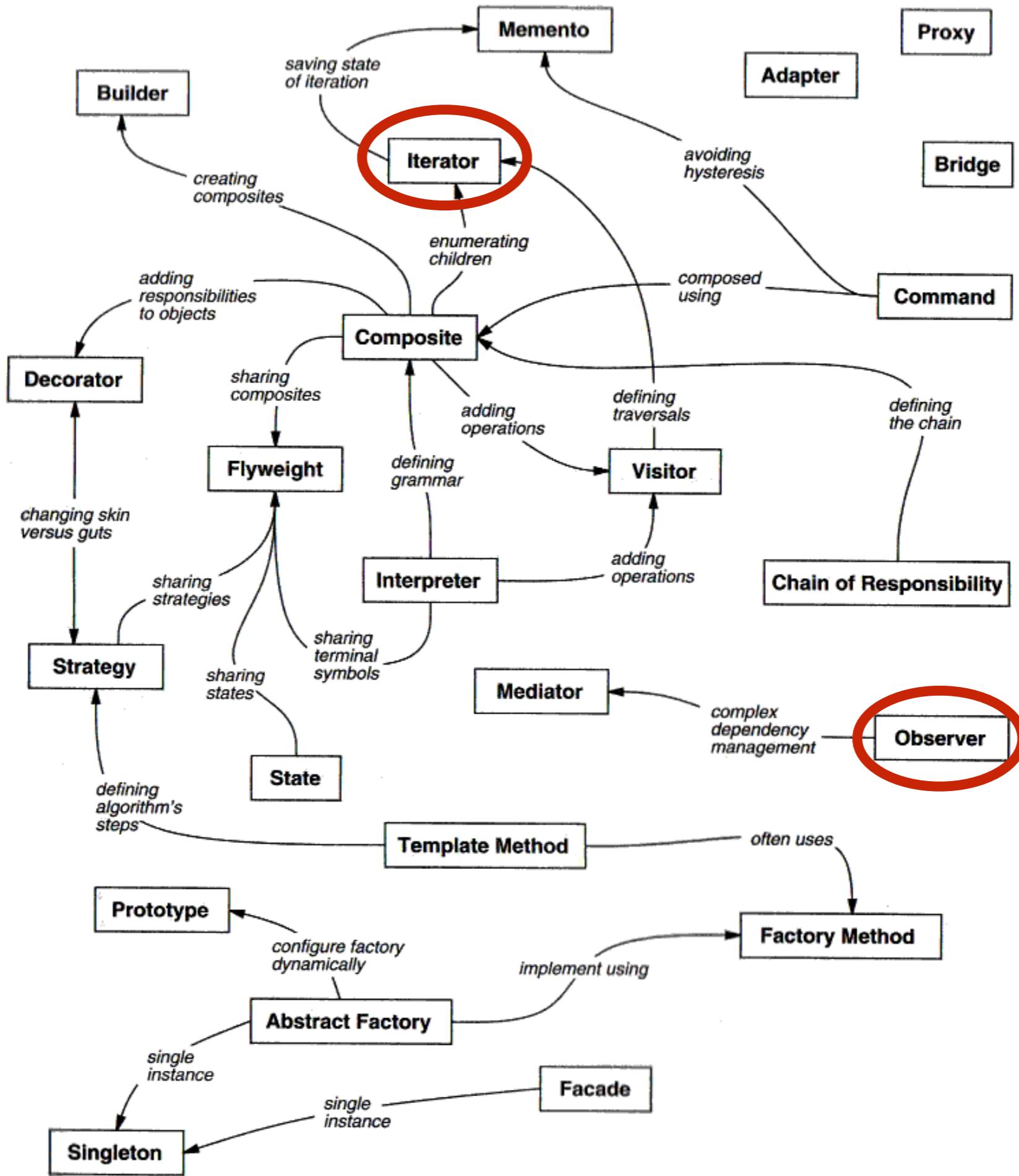
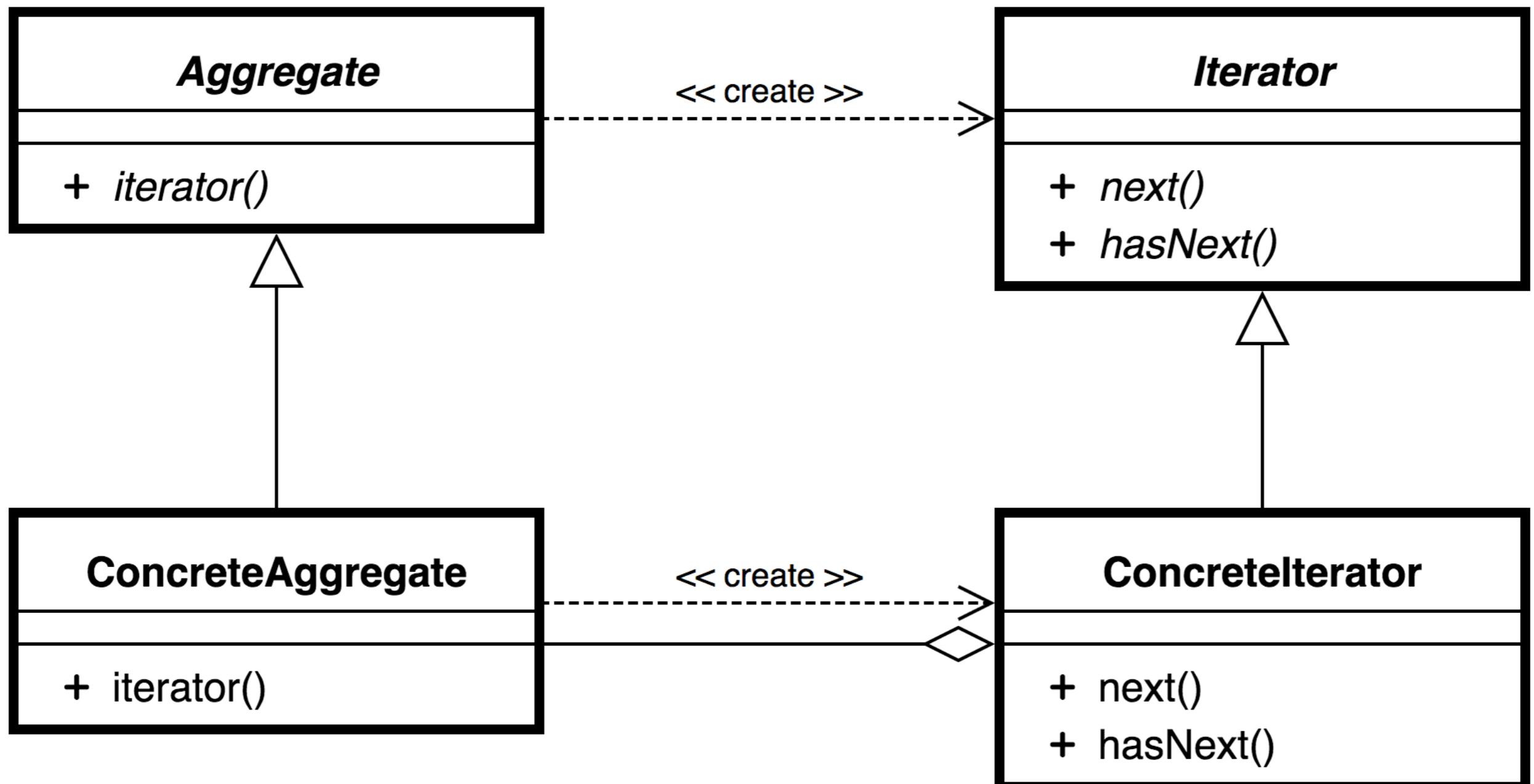
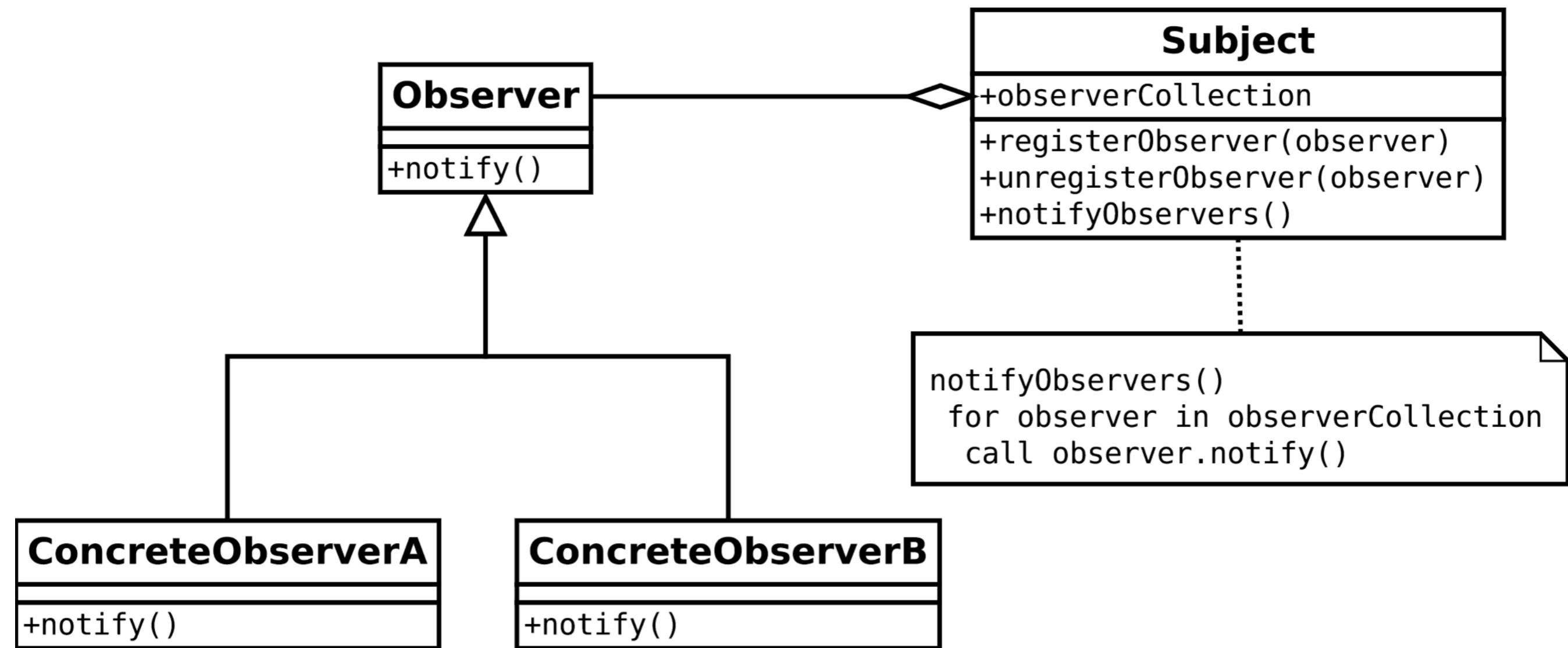


Figure 1.1 Design Patterns Relationships



Iterator ES2015 (ES6)

```
let iterator = getNumeros();
> console.log(iterator.next());
> { value: 1, done: false }
> console.log(iterator.next());
> { value: 2, done: false }
> console.log(iterator.next());
> { value: 3, done: false }
> console.log(iterator.next());
> { done: true }
```



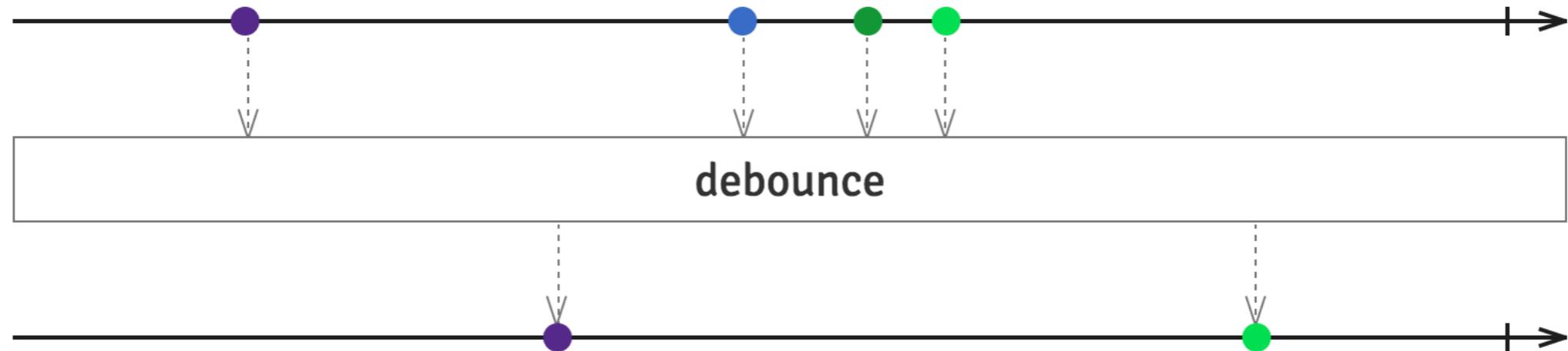


Extensões reativas

<http://reactivex.io/>

The Observer pattern done right

ReactiveX is a combination of the best ideas from
the **Observer** pattern, the **Iterator** pattern, and **functional programming**



CREATE

Easily create event streams or data streams.

COMBINE

Compose and transform streams with query-like operators.

LISTEN

Subscribe to any observable stream to perform side effects.

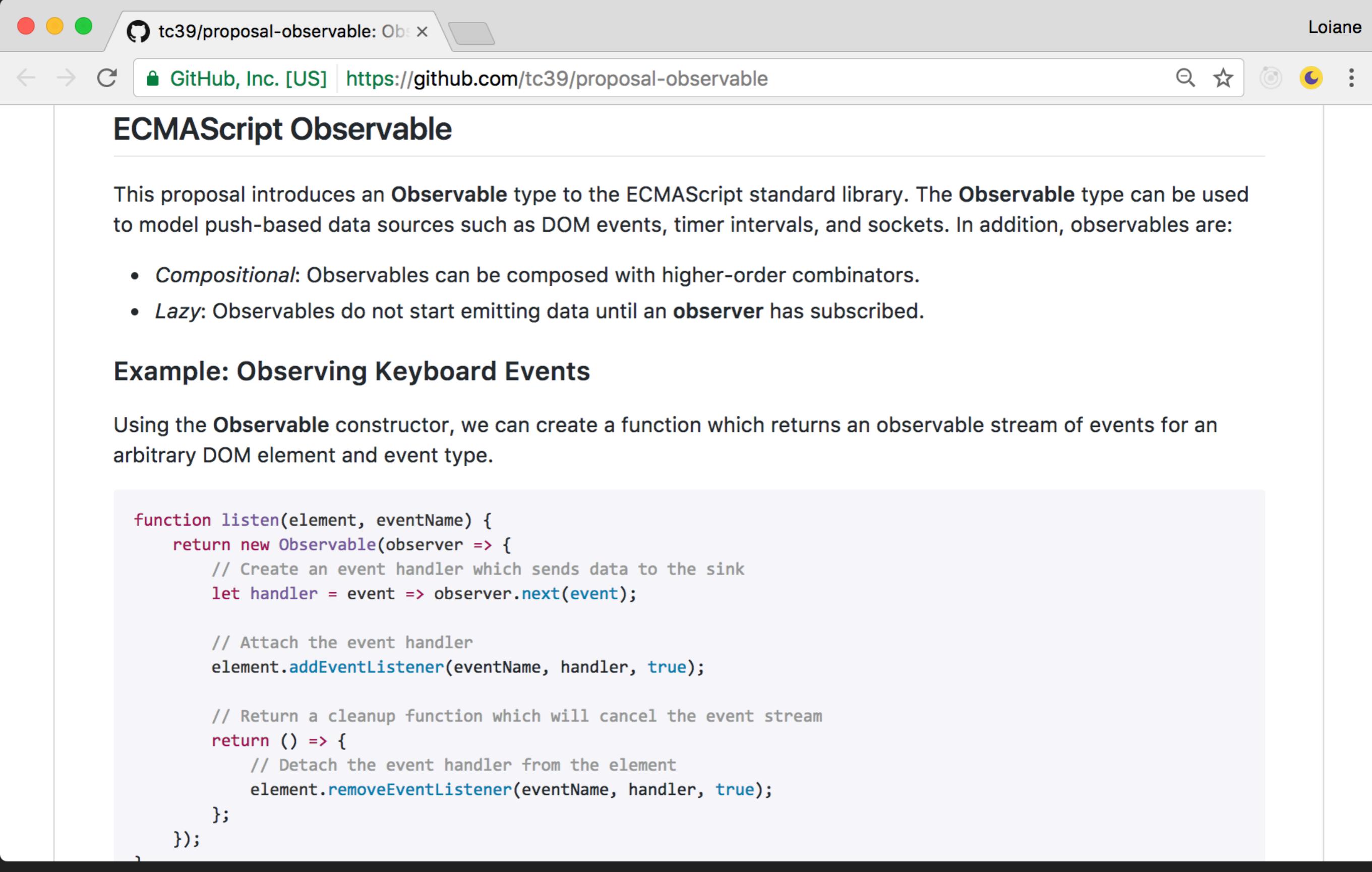
Observable

```
var source = Rx.Observable.return(42);

var observer = Rx.Observer.create(
  function (x) {
    console.log('Next: ' + x);
  },
  function (err) {
    console.log('Error: ' + err);
  },
  function () {
    console.log('Completed');
  }
);

var subscription = source.subscribe(observer);
```

EcmaScript 2018



The screenshot shows a web browser window with the following details:

- Title Bar:** tc39/proposal-observable: Observables
- Address Bar:** GitHub, Inc. [US] | <https://github.com/tc39/proposal-observable>
- User Information:** Loiane

ECMAScript Observable

This proposal introduces an **Observable** type to the ECMAScript standard library. The **Observable** type can be used to model push-based data sources such as DOM events, timer intervals, and sockets. In addition, observables are:

- *Compositional*: Observables can be composed with higher-order combinators.
- *Lazy*: Observables do not start emitting data until an **observer** has subscribed.

Example: Observing Keyboard Events

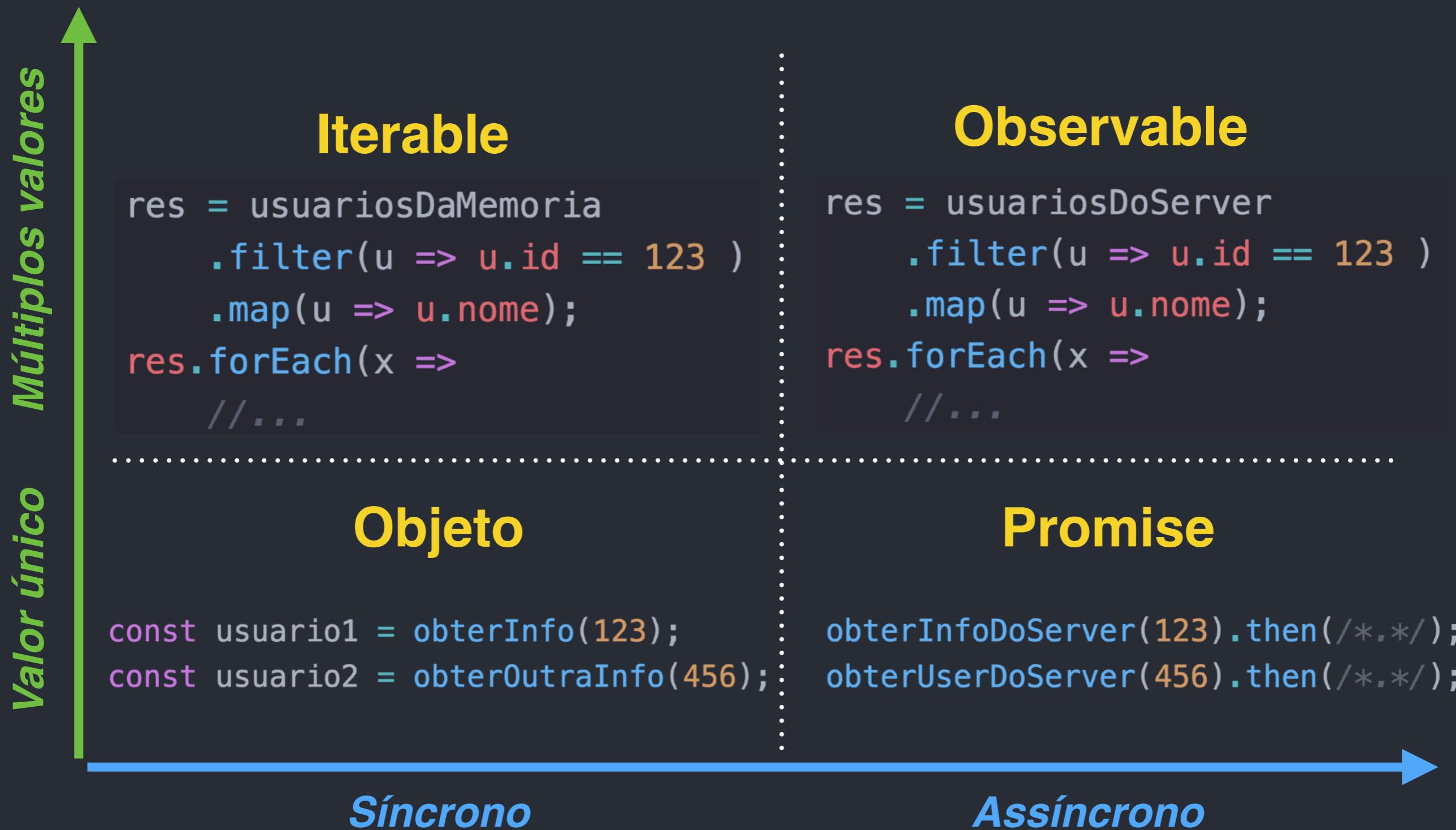
Using the **Observable** constructor, we can create a function which returns an observable stream of events for an arbitrary DOM element and event type.

```
function listen(element, eventName) {
    return new Observable(observer => {
        // Create an event handler which sends data to the sink
        let handler = event => observer.next(event);

        // Attach the event handler
        element.addEventListener(eventName, handler, true);

        // Return a cleanup function which will cancel the event stream
        return () => {
            // Detach the event handler from the element
            element.removeEventListener(eventName, handler, true);
        };
    });
}
```

Panorama da teoria da reatividade





Angular Reativo



Forms

Routing

Http Service

Pipe



Projeto Original: <https://github.com/Reactive-Extensions/RxJS>

Reescrita: <https://github.com/ReactiveX/rxjs>

(melhor performance, modularidade, pilha de debug/depuração => boa compatibilidade com versões anteriores, com poucas breaking changes)



Http Services

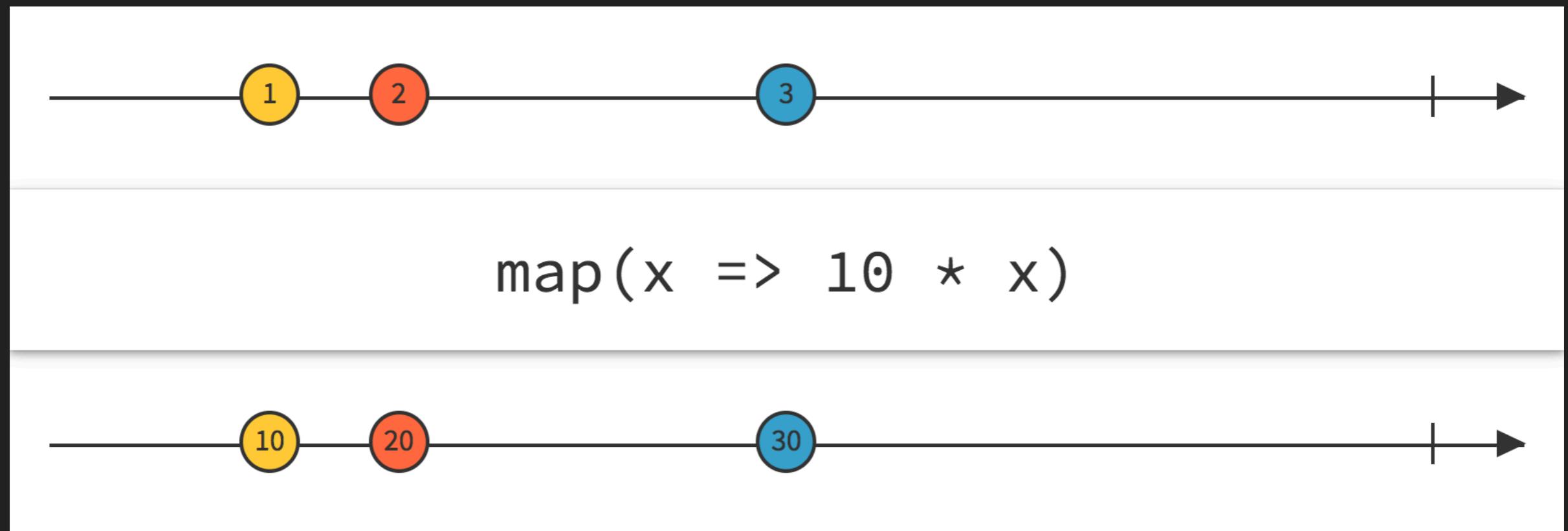
```
getAllPokemon() {  
  if (this.pokemons) {  
    return Observable.of(this.pokemons);  
  } else {  
    return this.http.get(this.baseUrl + 'pokemon.json')  
      .map(res: Response) => res.json().results  
      .do(data) => { this.pokemons = data; })  
      .catch(error: any) =>  
        Observable.throw(error.json().error || 'Server error')  
  );  
}  
}
```



Operadores



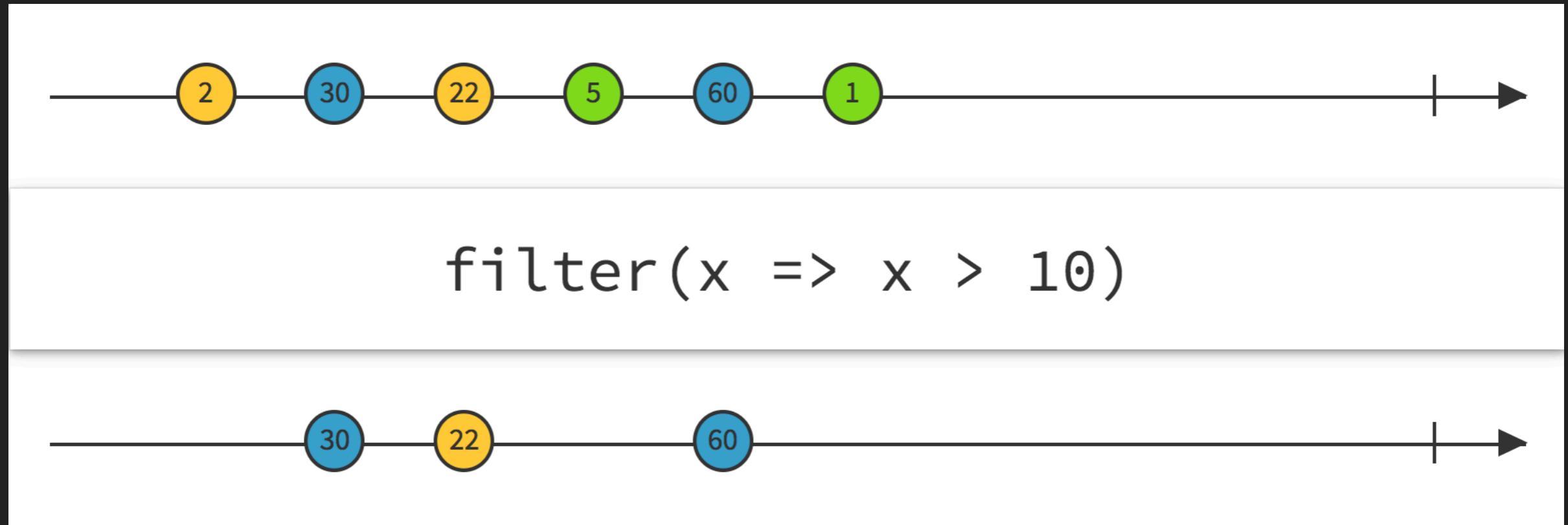
Transformar => map



<http://rxmarbles.com/#map>



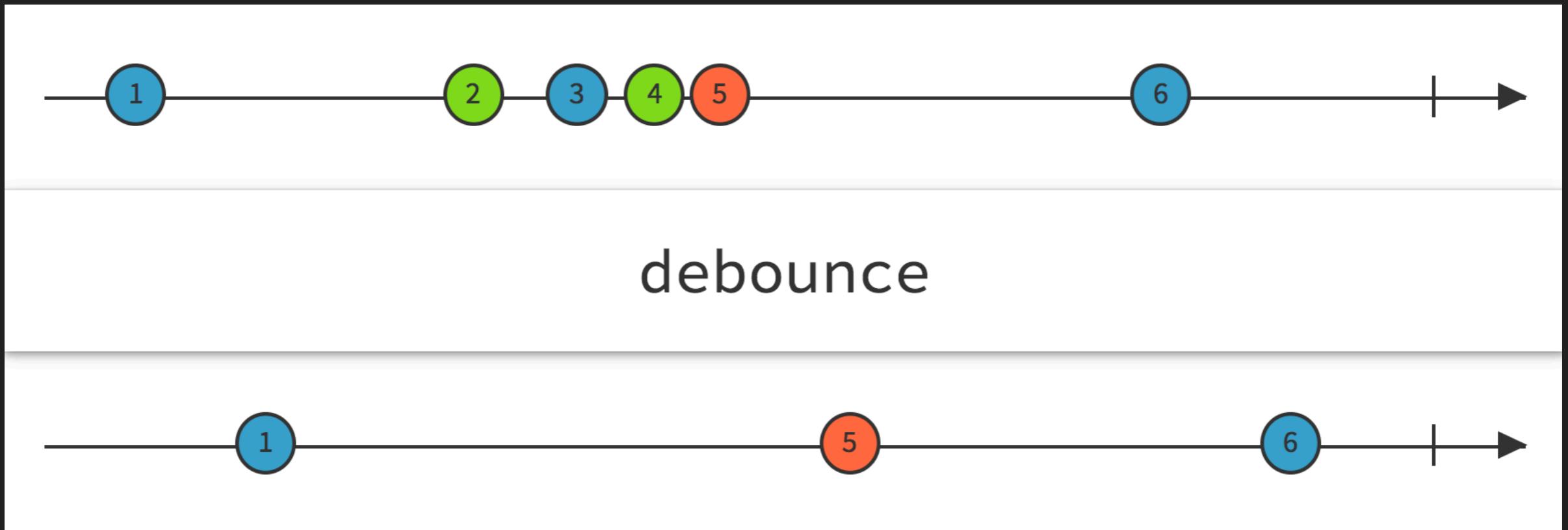
Filtrar informações => filter



<http://rxmarbles.com/#filter>



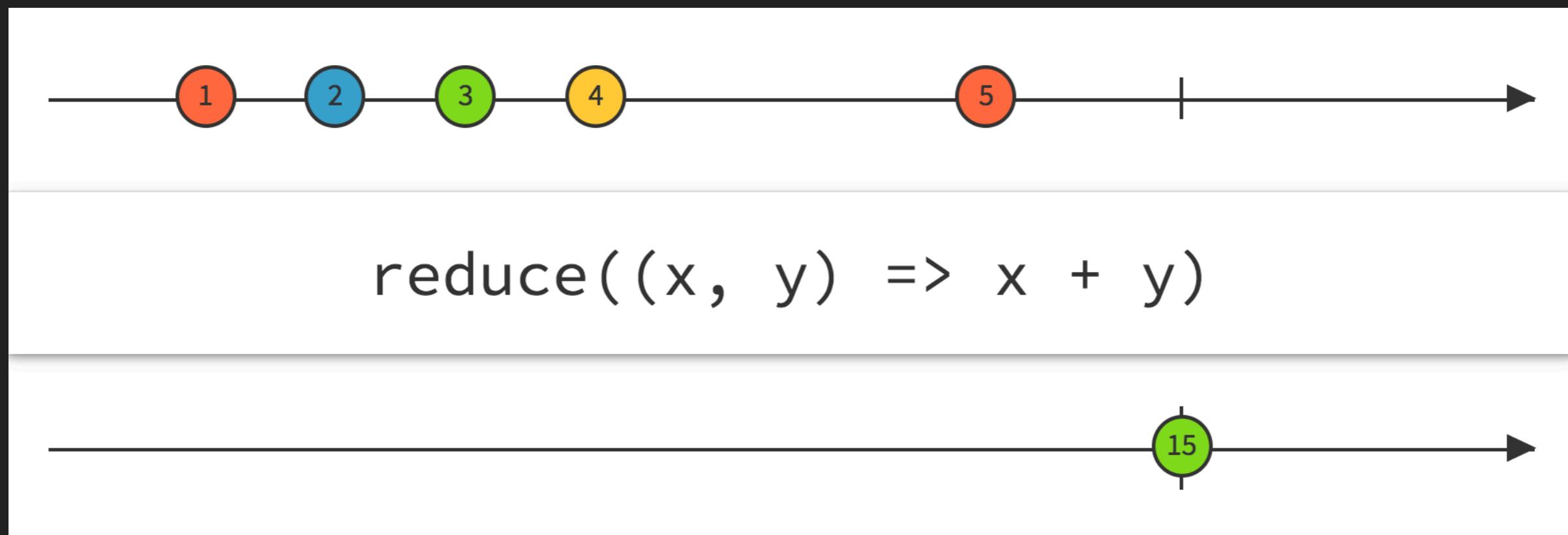
Desativar => debounce



<http://rxmarbles.com/#debounce>



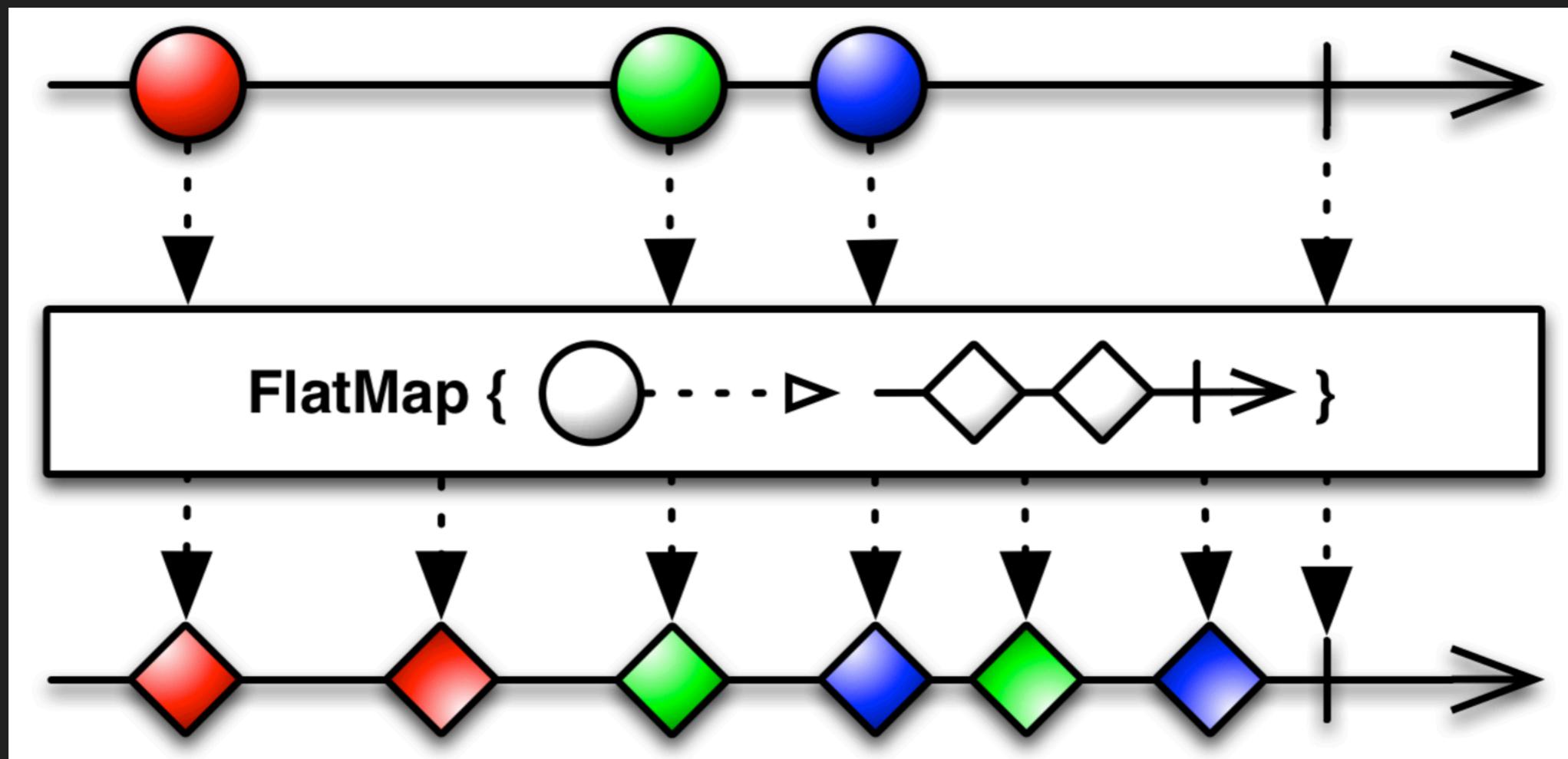
Redução => reduce



<http://rxmarbles.com/#reduce>



Transformar items em Observables => flatmap



<http://reactivex.io/documentation/operators/flatmap.html>



Roteamento

```
ngOnInit() {  
    this.inscricao = this.route.params.subscribe(  
        (params: any) => {  
            const id = params['id'];  
            this.aluno = this.alunosService.getAluno(id);  
        }  
    );  
}
```



Roteamento

```
inscricao: Subscription;
```

```
ngOnInit() {  
    this.inscricao = this.route.params.subscribe(  
        (params: any) => {  
            const id = params['id'];  
            this.aluno = this.alunosService.getAluno(id);  
        }  
    );  
}
```

```
ngOnDestroy(){  
    this.inscricao.unsubscribe();  
}
```



Pipe Async

```
this.courses$ = this.store.select(states.getCourses);  
  
this.isLoading$ = this.store.select(states.getIsLoadingCourses);
```



Pipe Async

```
this.courses$ = this.store.select(states.getCourses);

this.isLoading$ = this.store.select(states.getIsLoadingCourses);

<div class="preloader-wrapper active"
  *ngIf="isLoading$ | async as isLoading; else listCourses">
  <app-loader></app-loader>
</div>
<ng-template #listCourses>
  <app-courses-list-item
    *ngFor="let course of courses$ | async"
    [course]="course"
    (courseClicked)="goToCourseDetail($event)">
  </app-courses-list-item>
</ng-template>
```



Pipe Async

```
this.courses$ = this.store.select(states.getCourses);
```

```
this.isLoading$ = this.store.select(states.getIsLoadingCourses);
```

Observable<boolean>

```
<div class="preloader-wrapper active">
  *ngIf="isLoading$ | async as isLoading; else listCourses">
    <app-loader></app-loader>
</div>
```

```
<ng-template #listCourses>
```

```
  <app-courses-list-item>
```

```
    *ngFor="let course of courses$ | async"
```

```
      [course]="course"
```

```
      (courseClicked)="goToCourseDetail($event)">
```

```
    </app-courses-list-item>
```

```
</ng-template>
```

Observable<Course[]>



```
| async'
```

async == subscribe + unsubscribe



Pipe Async

Sintaxe melhorada Angular >= 4.x

Observable<Course>

```
<div *ngIf="course$ | async as course">
  <h4 class="header">{{ course?.name }}</h4>
  <div>
    {{ course?.description }}
  </div>
</div>
```



Formulários Reativos I

```
initForm() {  
  this.form = this.fb.group({  
    email: ['', [  
      Validators.required,  
      Validators.email  
    ]],  
    password: ['', [  
      Validators.required,  
      Validators.minLength(6)  
    ]]  
  });  
}
```



Formulários Reativos II

```
this.form.valueChanges.subscribe(data => {  
  console.log('Mudanças no form', data);  
});
```

```
this.form.controls['email'].valueChanges  
  .subscribe(novoValor => console.log(novoValor))  
);
```



Everything is a stream

Blocos Principais

MÓDULOS

COMPONENTES

TEMPLATE

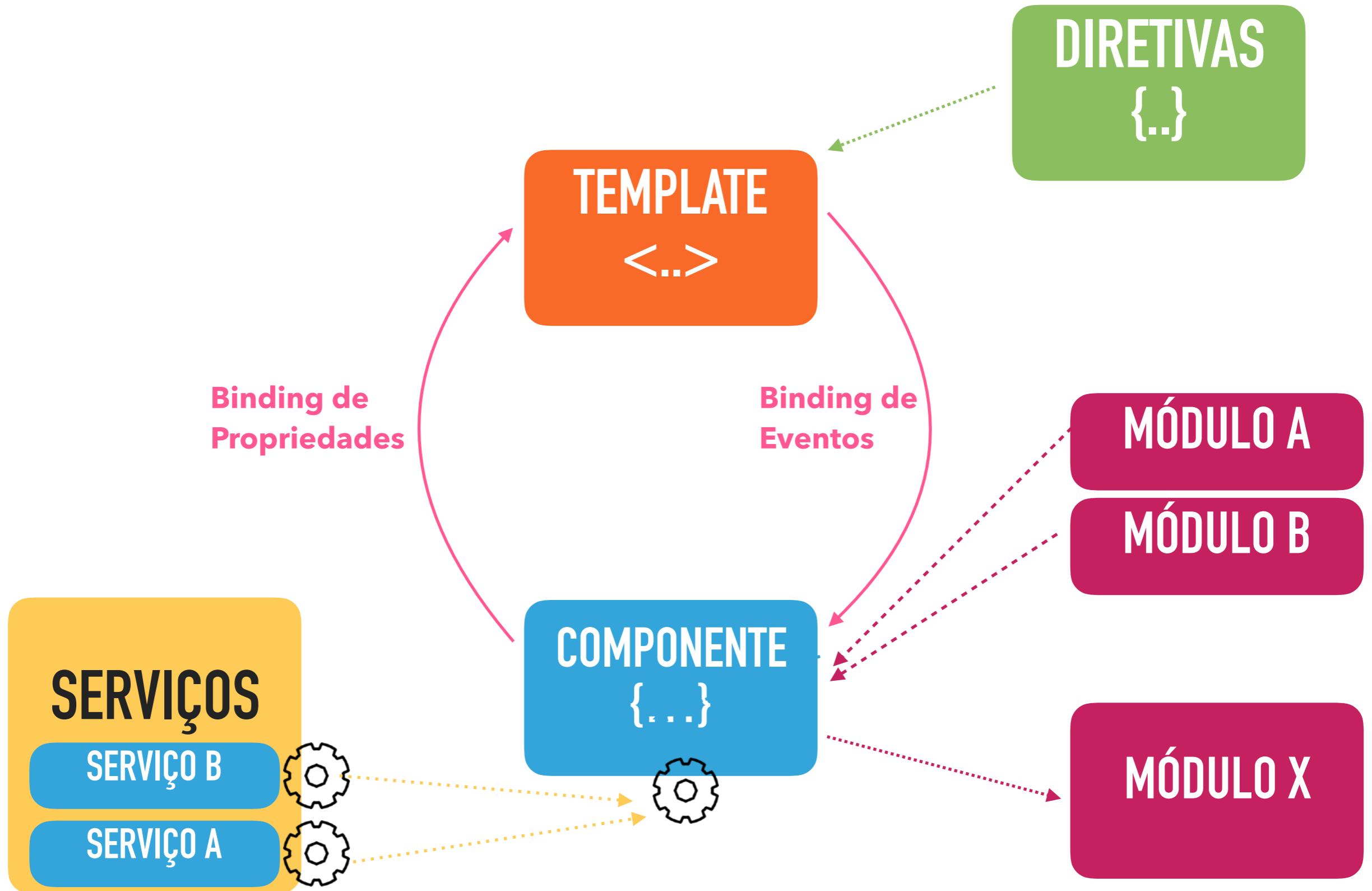
DIRETIVAS

SERVIÇOS

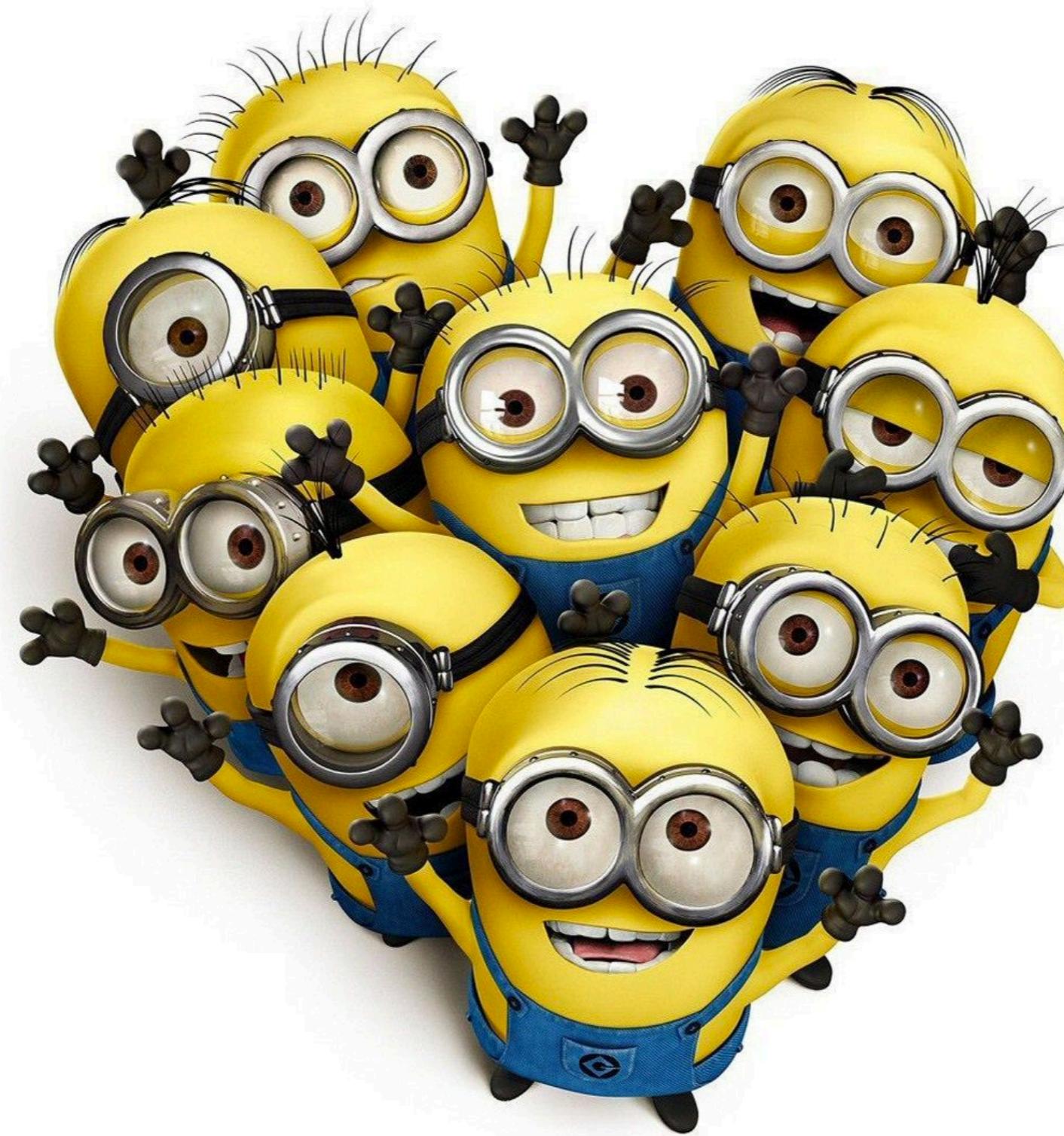
INJEÇÃO
DEPENDÊNCIA

DATA BINDING

ROTEAMENTO



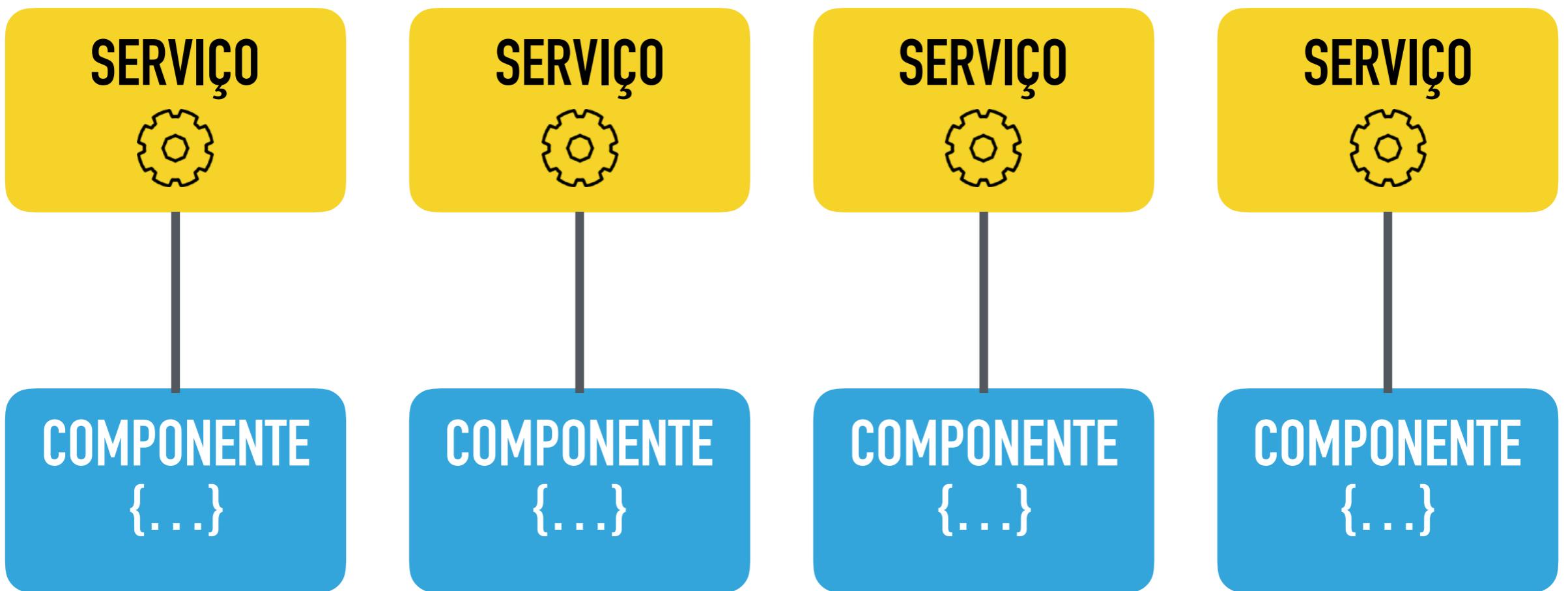
Yay! Projeto novo em Angular!



SERVIÇO

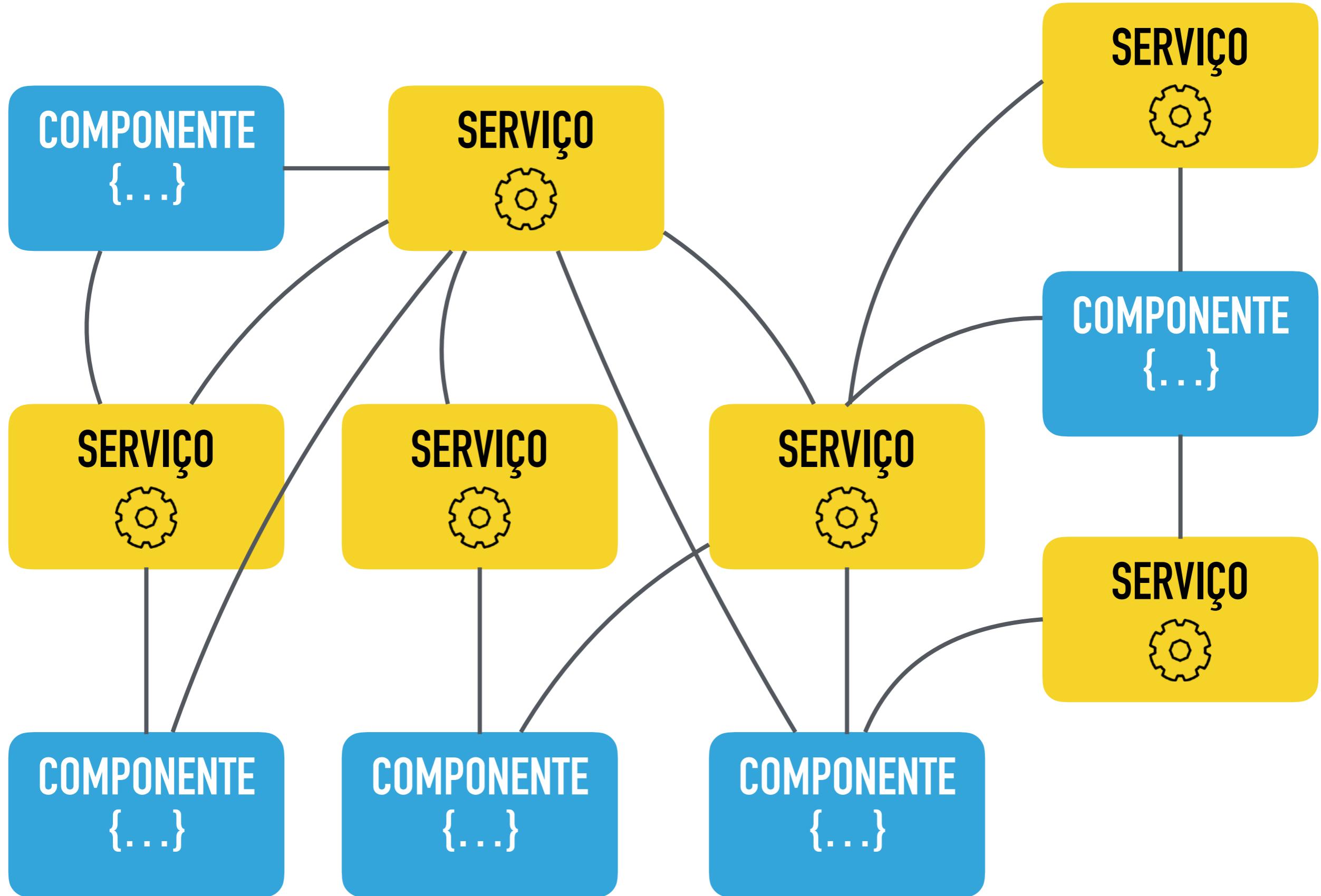


COMPONENTE
{...}



uma semana depois...







OMG WTF



Gerenciamento de estado...



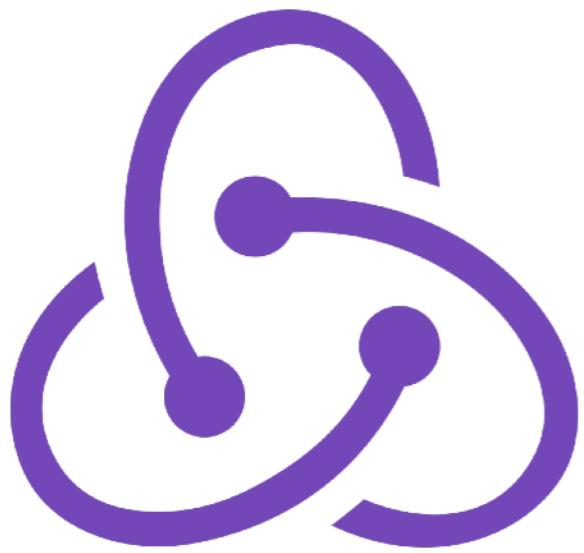
**Dados que vem/enviados do/para
servidor, estão pendentes e resultaram
em erro...**

Estado UI, como toggle, alerts e erros

User input, filtros, buscas...



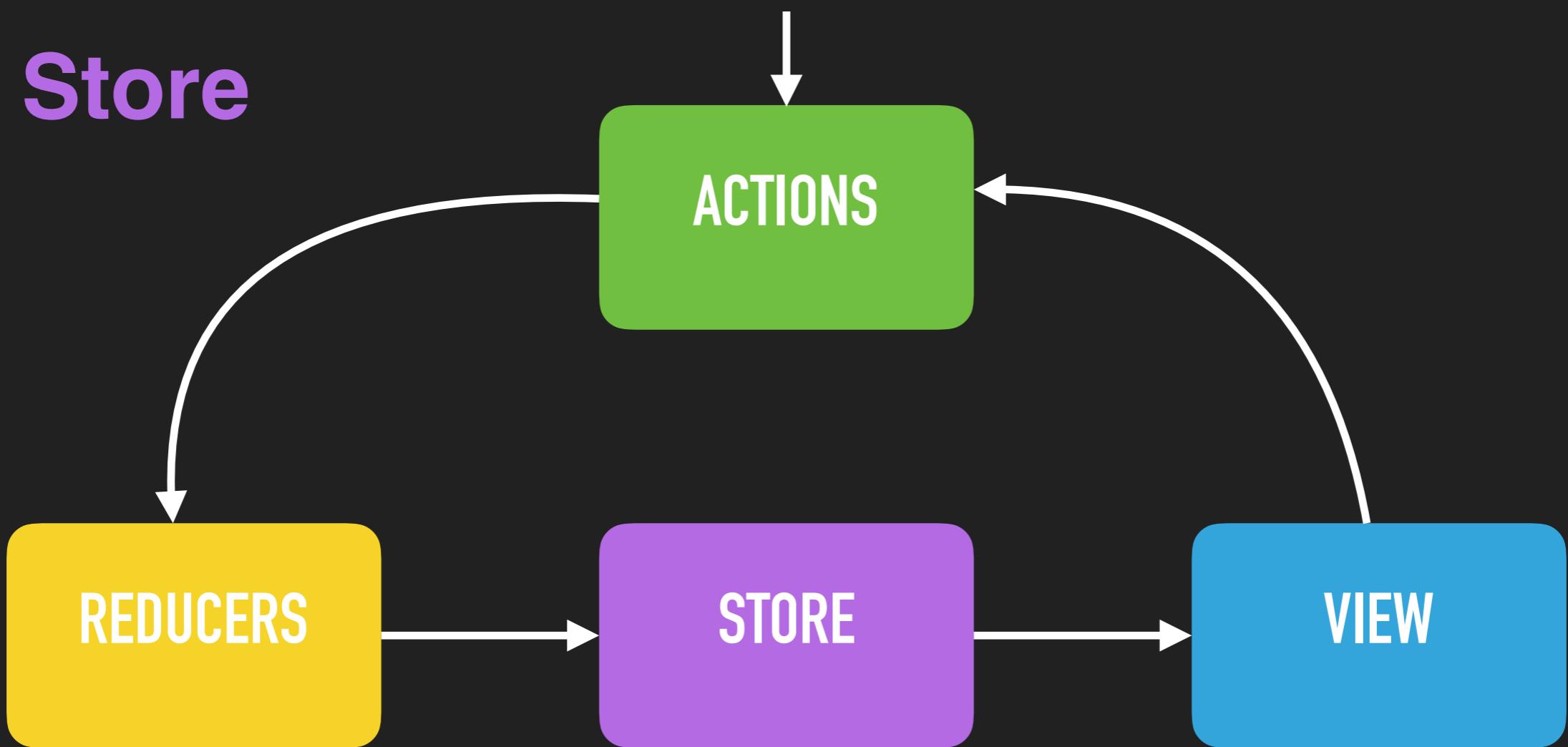
E se pudéssemos gerenciar tudo
num lugar só...



Redux



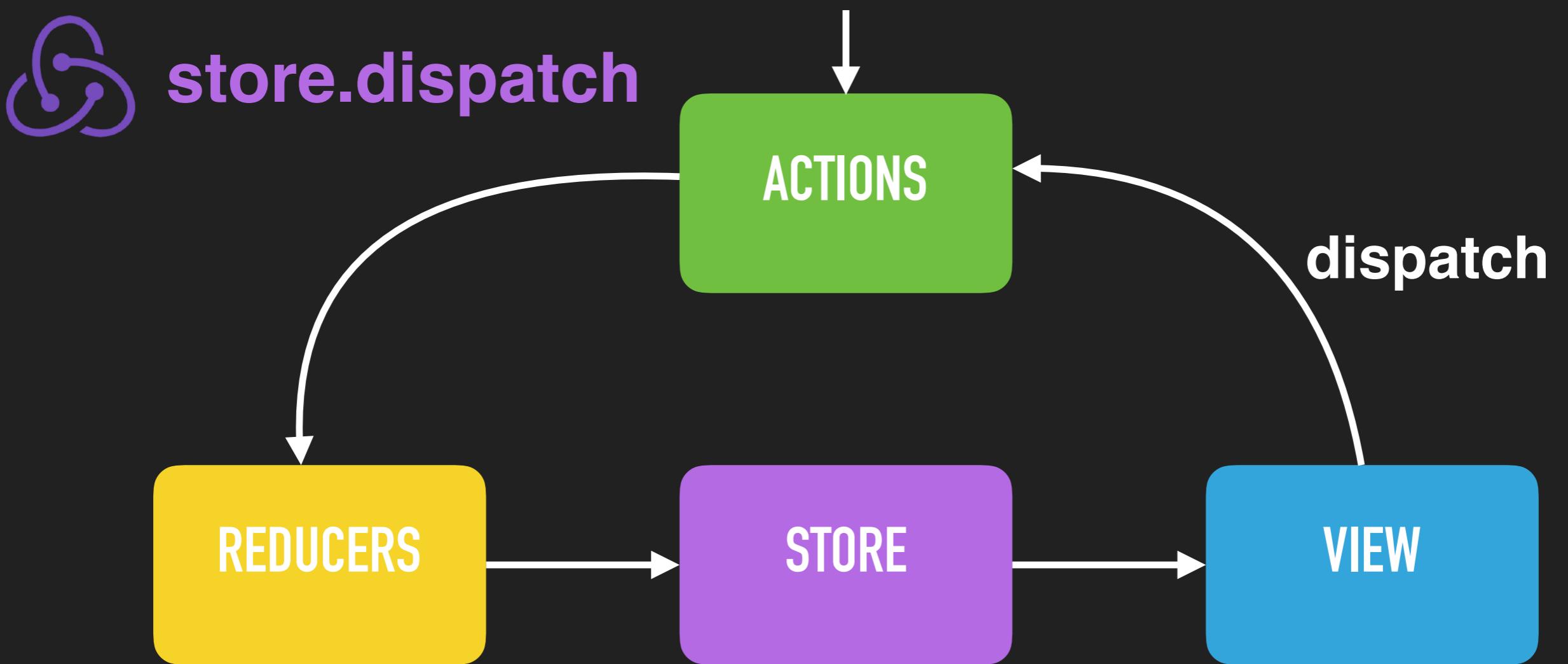
Redux é uma biblioteca, e também
é um **padrão**



“Banco de dados” da aplicação

Manipulação do estado acontece nos reducers que registram o estado na store

Pode executar ações pré ou pós reducer (middleware)

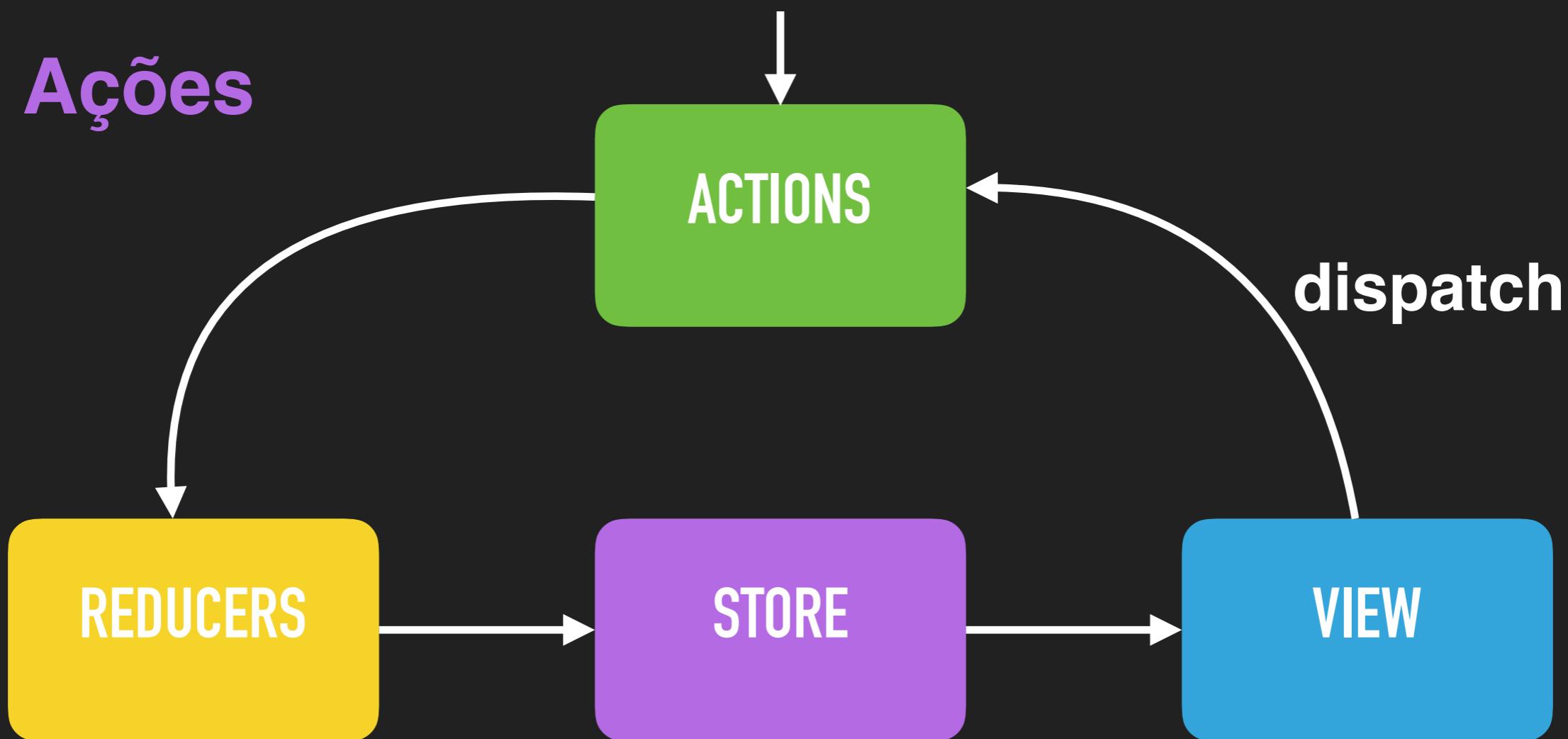


Envia uma ação para a store, que executa o reducer apropriado

É chamado de um componente ou serviço



Ações



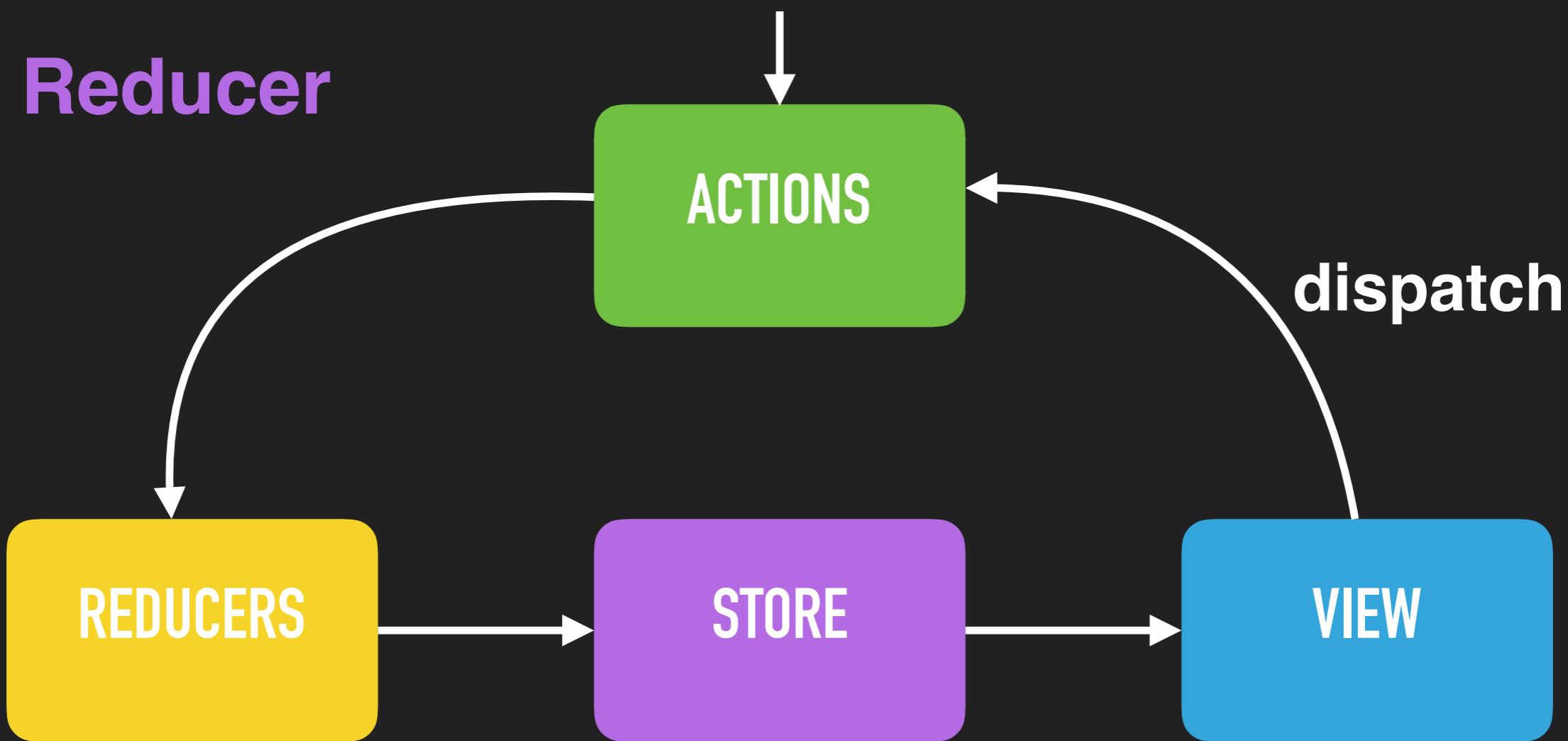
Funções que disparam os eventos para os reducers

Possuem um tipo e um payload (carga)

**Baseado na ação, o reducer obtém o payload
retorna um novo estado**



Reducer



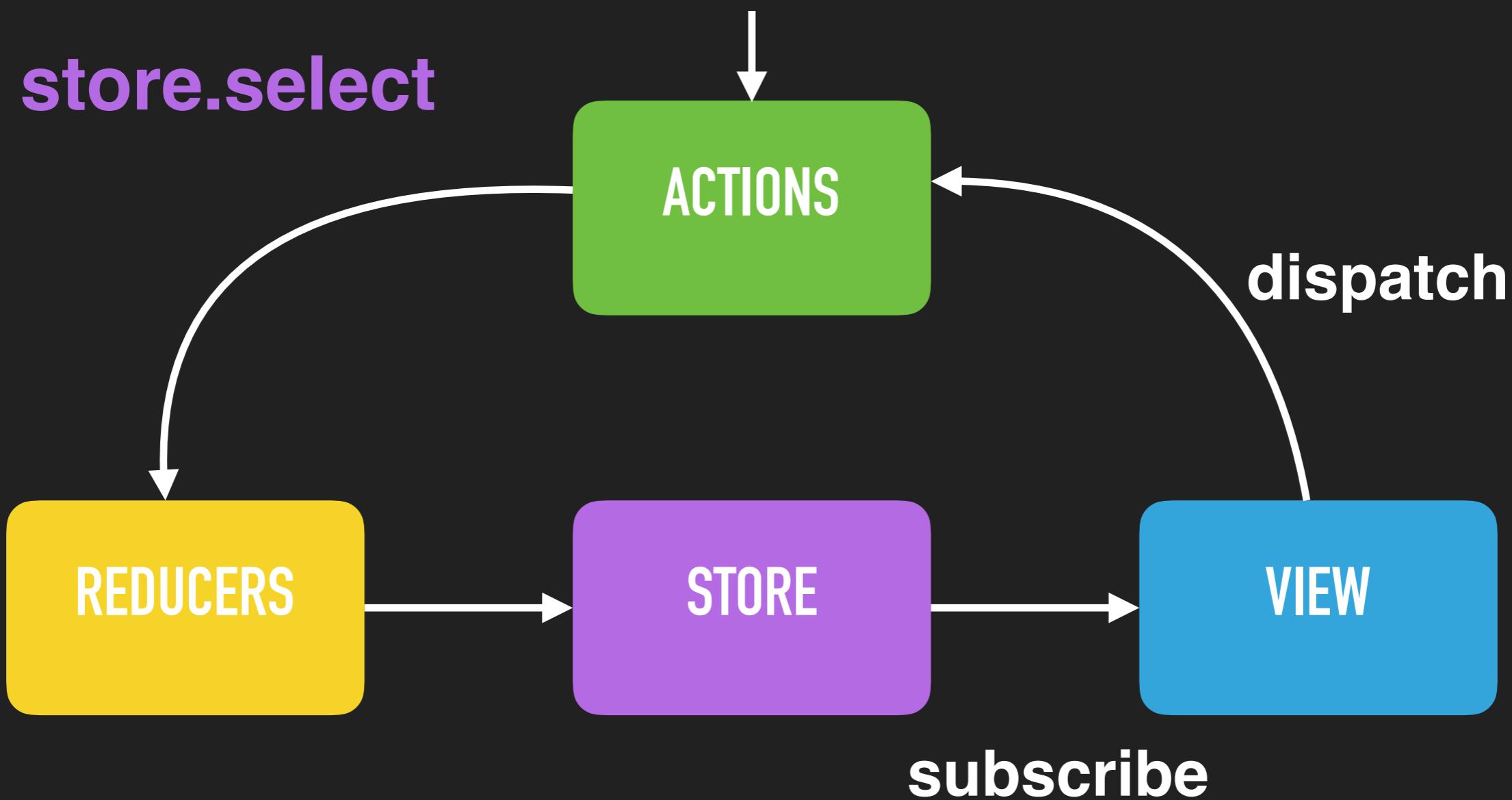
Função que recebe o estado atual e uma ação

Retorna um estado novo baseado na ação

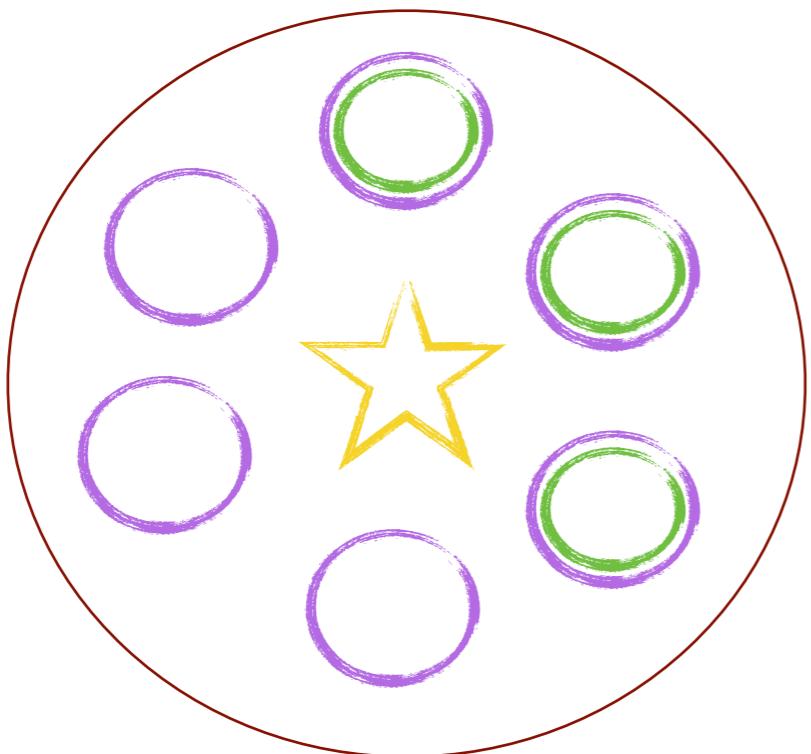
Funções reducers devem ser funções puras



store.select

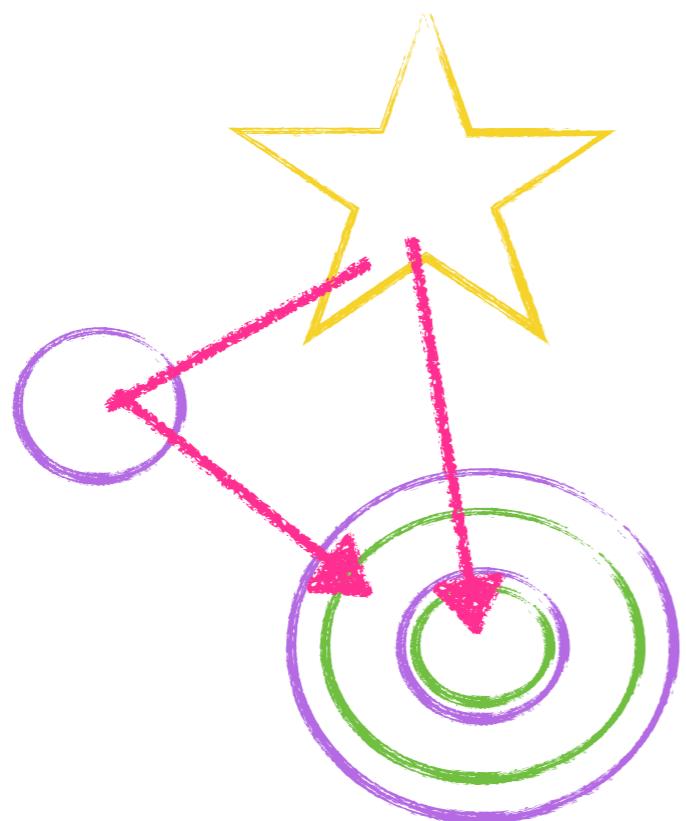


Fornece a informação do estado que queremos selecionar/consultar



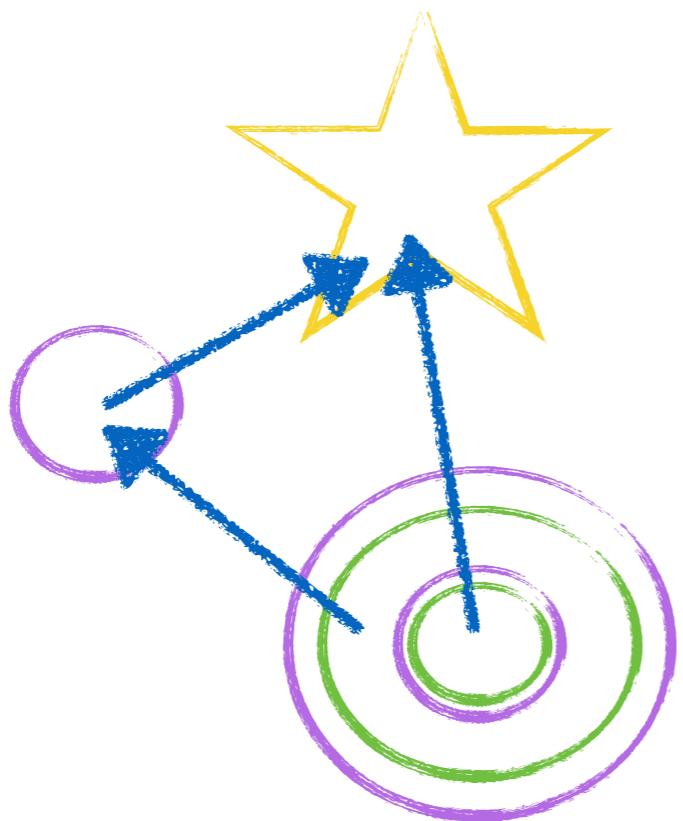
- ★ **store**
- **serviço**
- **componente**

Árvore de estado única



★ **store**
○ **serviço**
○ **componente**

Estado flui de cima pra baixo



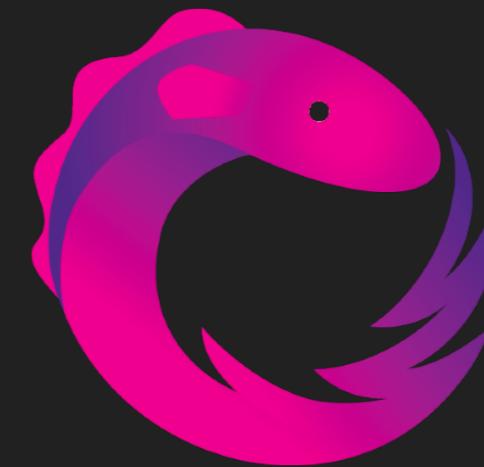
★ **store**
○ **serviço**
○ **componente**

Eventos fluem de baixo pra cima



<https://github.com/InfomediaLtd/angular2-redux>

Implementação bem parecida com o redux original



<https://github.com/ngrx>





Hello World!



Não reativo

```
<div class="content">
  <button (click)="increment()">+</button>
  <button (click)="decrement()">-</button>
  <button (click)="reset()">Reset Counter</button>
  <h3>{{ counter }}</h3>
</div>
```



Variável comum <number>



Não reativo

```
export class CounterComponent {  
  
  counter: number = 0;  
  
  increment() {  
    this.counter++;  
  }  
  
  decrement() {  
    this.counter--;  
  }  
  
  reset() {  
    this.counter = 0;  
  }  
}
```



Reativo

```
<div>
  <button (click)="increment()">+</button>
  <button (click)="decrement()">-</button>
  <button (click)="reset()">Reset Counter</button>
  <h3>{{counter$ | async}}</h3>
</div>
```



Observable<number>



Definir ações

```
export const INCREMENT = 'INCREMENT';
export const DECREMENT = 'DECREMENT';
export const RESET = 'RESET';
```



No componente:

counter\$: Observable<number>; **(2): Declarar variável do template**

```
constructor(private store: Store<number>) { }
```

(1): Injetar Store

```
ngOnInit() {  
    import { Store } from '@ngrx/store';  
    this.counter$ = this.store.select<number>('counter')  
}
```

```
increment() {  
    this.store.dispatch({ type: INCREMENT });  
}
```

```
decrement() {  
    this.store.dispatch({ type: DECREMENT });  
}
```

```
reset(){  
    this.store.dispatch({ type: RESET });  
}
```



No componente:

```
counter$: Observable<number>;  
  
constructor(private store: Store<number>) { }  
  
ngOnInit() {  
  this.counter$ = this.store.select<number>('counter')  
}  
  
increment() {  
  this.store.dispatch({ type: INCREMENT });  
}  
  
decrement() {  
  this.store.dispatch({ type: DECREMENT });  
}  
  
reset(){  
  this.store.dispatch({ type: RESET });  
}
```

**(3): Obter valor iniciar do counter\$
store.select (pedaço do estado)**



No componente:

```
counter$: Observable<number>;  
  
constructor(private store: Store<number>) { }  
  
ngOnInit() {  
  this.counter$ = this.store.select<number>('counter')  
}  
  
increment() {  
  this.store.dispatch({ type: INCREMENT });  
}  
  
decrement() {  
  this.store.dispatch({ type: DECREMENT });  
}  
  
reset(){  
  this.store.dispatch({ type: RESET });  
}
```

(4): No lugar de incrementar, decrementar, disparar eventos store.dispatch



Reducer

Recebe o estado

```
export const counterReducer: ActionReducer<number> = (state: number = 0, action: Action) => {  
  switch (action.type) {  
    case INCREMENT:  
      return state + 1;  
    case DECREMENT:  
      return state - 1;  
    case RESET:  
      return 0;  
    default:  
      return state;  
  }  
};
```

Recebe a ação + payload

Retorna um novo estado



Integra com a aplicação Angular

```
imports: [  
  BrowserModule,  
  StoreModule.provideStore({ counter: counterReducer })
```



**Fornece a store (AppState) para toda a aplicação
 AppModule.ts**



CRUD com Ajax



Definir ações

```
LOAD: type(' [Task] -LOAD Requested-'),  
LOAD_COMPLETED: type(' [Task] -LOAD Completed-'),  
LOAD_ERROR: type(' [Task] -LOAD Error-'),
```

Para cada chamada Ajax: 3 ações:
Pedido (request)
Request Completo (altera o estado)
Erro

```
export class LoadCompletedAction implements Action {  
  type = ActionTypes.LOAD_COMPLETED;  
  
  constructor(public payload: {tasks: Task[]}) { }  
}
```



No componente

```
store.dispatch(new actions.LoadAction());
```

Evento de pedido para carregar a informação do servidor

```
this.tasks$ = store.select('tasks')
```



Reducer

```
export const taskReducer: ActionReducer<Task[]> = (state: Task[] = [], action: TaskAction) => {  
  switch (action.type) {  
    case ActionTypes.LOAD_COMPLETED:  
      return [...state, ...action.payload.tasks];  
  
    case ActionTypes.CREATE_COMPLETED:  
      return [...state, action.payload.task];  
  
    case ActionTypes.REMOVE_COMPLETED:  
      return state.filter((task: Task) => {  
        return task.id !== action.payload.task.id;  
      });  
  
    default:  
      return state;  
  }  
}
```

NÃO mudar o estado diretamente
Estado deve ser imutável



Reducer DEVE ser uma função PURA
Programação Funcional



Dados imutáveis

```
const meuArray = [1, 2, 3, 4];  
meuArray.push(5);
```

```
const meuArrayImutavel = [...meuArray, 6];
```



Dados imutáveis

```
const meuArray = [1, 2, 3, 4];  
meuArray.push(5);
```

```
const meuArrayImutavel = [...meuArray, 6];
```

```
case AuthActionTypes.LOGIN_COMPLETED: {  
    return Object.assign({}, state, {  
        userData: action.payload.user,  
        isLoggedIn: action.payload.user != null,  
        error: null  
    });  
}
```

IMMUTABLE



Função pura x impura

```
function calcularAoQuadrado(x) {  
    return x * x;  
}
```

```
function quadradoArray(items) {  
    for (var i = 0; i < items.length; i++) {  
        items[i] = items[i] * items[i];  
    }  
    return items;  
}
```



Função pura x impura

```
function calcularAoQuadrado(x) {  
    return x * x;  
}
```

```
function quadradoArray(items) {  
    for (var i = 0; i < items.length; i++) {  
        items[i] = items[i] * items[i];  
    }  
    return items;  
}
```

Não modifica o estado e não tem efeitos colaterais



Components: Dumb Components

```
<li class="collection-item">
  <span [class.task-completed]="task.completed">{{ task.title }}</span>
  <a class="secondary-content btn-floating"
    (click)="onRemove()">
    <i class="material-icons circle">delete</i>
  </a>
  <a class="secondary-content btn-floating"
    (click)="onComplete()">
    <i class="material-icons circle">done</i>
  </a>
</li>
```



Components: Dumb Components

```
export class TaskItemComponent {  
  
  @Input() task: Task;  
  @Output() remove: EventEmitter<any> = new EventEmitter(false);  
  @Output() complete: EventEmitter<any> = new EventEmitter(false);  
  
  onRemove(){  
    this.remove.emit();  
  }  
}
```

- Apenas recebem informações via Input properties e disparam eventos com Output properties
- NÃO sabem nem se comunicam com Store ou Estado da aplicação



Containers: Smart Components

```
<div class="row">
  <div class="col s6 offset-s3 input-field">
    <app-task-form (createTask)="onCreateTask($event)"></app-task-form>
    <app-tasks-list
      [tasks]="tasks$"
      (remove)="onRemoveTask($event)"
      (complete)="onUpdateTask($event)">
    </app-tasks-list>
  </div>
</div>
```

**Escutam os eventos dos
Componentes filhos e fazem o
dispatch**

Lidam com o ngRx e a Store

Mas e a comunicação com servidor?



Redux apenas se interessa pelo estado do cliente (frontend)



Middleware



Store side effects

Efeitos Colaterais

`@ngrx/effects`



Effects

```
@Effect()
loadAction$: Observable<Action> = this.actions$  
  .ofType(taskActions.ActionTypes.LOAD)  
  .switchMap(() => this.api.loadTasks()  
    .map(res => new taskActions.LoadCompletedAction({tasks: res}))  
    .catch(() => Observable.of({ type: taskActions.ActionTypes.LOAD_ERROR }))  
);
```

**Escuta a ação de Pedido e faz dispatch da ação de
“Completo” - que muda o estado**



Effects + Service API

```
loadTasks(): Observable<Task[]> {  
  return this.http.get(this.API_TASKS_URL)  
    .map((res: Response) => res.json());  
}
```

```
createTask(body: any): Observable<Task> {  
  return this.http.post(this.API_TASKS_URL, this.getBody(body))  
    .map((res: Response) => res.json());  
}
```

o Serviço da API não sabe do estado nem do redux



Effects

Payload é a resposta do servidor

```
@Effect()
createAction$: Observable<Action> = this.actions$  
  .ofType(taskActions.ActionTypes.CREATE)
  .map(action => action.payload.task)
  .switchMap(payload => this.api.createTask(payload)
    .map(res => new taskActions.CreateCompletedAction({task: res}))
    .catch(() => Observable.of({ type: taskActions.ActionTypes.CREATE_ERROR }))
);
```

RxJS <3

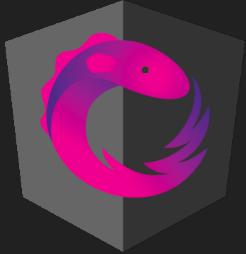


Executando os Effects

```
@NgModule({  
  imports: [  
    CommonModule,  
    FormsModule,  
    TasksRoutingModule,  
    EffectsModule.run(TaskEffects)  
,  
  providers: [  
    TasksService,  
    TaskEffects  
,  
  ]  
})  
export class TasksModule { }
```



Auth + Múltiplos módulos com Firebase

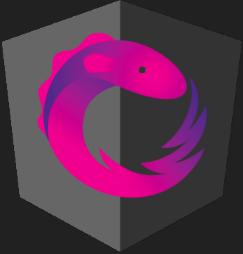


Lidar com roteamento?



Effects

```
@Effect()
loginAction$: Observable<Action> = this.actions$  
  .ofType(auth.AuthActionTypes.LOGIN_REQUESTED)  
  .map(action => action.payload)  
  .switchMap(payload => this.authService.signIn(payload.user)  
    .map(res => (new auth.LoginCompletedAction(new auth.AuthUserPayload(res))))  
    .do(() => this.router.navigate(['/home']))  
    .catch((error) => Observable.of(new auth.AuthErrorAction({error: error})))  
);
```



Projeto com vários módulos?



combine reducers e estados

```
const reducers = {  
  auth: fromAuth.authReducer,  
  course: fromCourse.courseReducer,  
  lesson: fromLesson.lessonReducer  
};
```

`combineReducers(reducers);`

```
export interface AppState {  
  auth: AuthState;  
  course: CourseState;  
  lesson: LessonState;  
};
```



store.select de um pedaço do estado

```
export const getCourseState = (state: AppState) => state.course;

export const getCourses = createSelector(getCourseState, fromCourse.getCourses);
export const getIsLoadingCourses = createSelector(getCourseState, fromCourse.getIsLoading);
export const getSelectedCourseUrl = createSelector(getCourseState, fromCourse.getSelectedCourse);
export const getCourseLessons = createSelector(getCourseState, fromCourse.getCourseLessons);

// LESSONS
export const getLessonState = (state: AppState) => state.lesson;

export const getSelectedLessonUrl = createSelector(getLessonState, fromLesson.getSelectedLessonUrl);
export const getSelectedLesson = createSelector(getCourseLessons, getSelectedLessonUrl,
  (lessons, selectedUrl) => lessons.find(lesson => lesson.url == selectedUrl)
);
```

reselect

```
this.lesson$ = this.store.select(fromRoot.getSelectedLesson);
```



Outras ferramentas

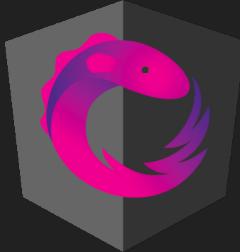


@ngrx/router-store: estado do roteamento na store

@store-devtools: integração redux-devtools

@store-log-monitor: monitoramento da store

ngrx-store-freeze: garante estado imutável no desenvolvimento do projeto



Desafios (ainda não 100% definidos)

Lazy loading dos arquivos ngRx:

<https://github.com/ngrx/store/issues/197>

<https://www.npmjs.com/package/ngrx-domains> (alfa)



Detecção de Mudanças



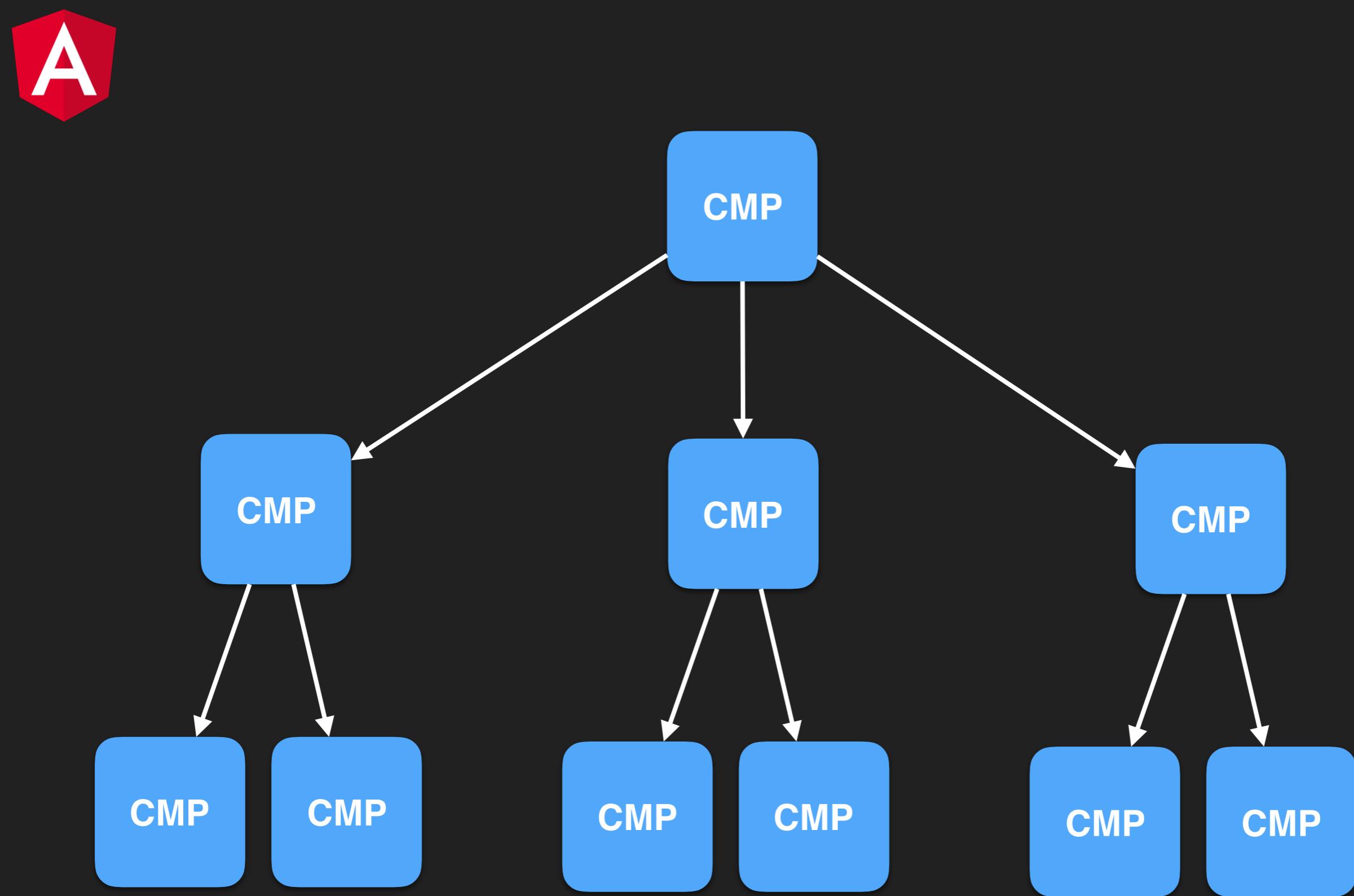
Muda o estado, muda a aplicação

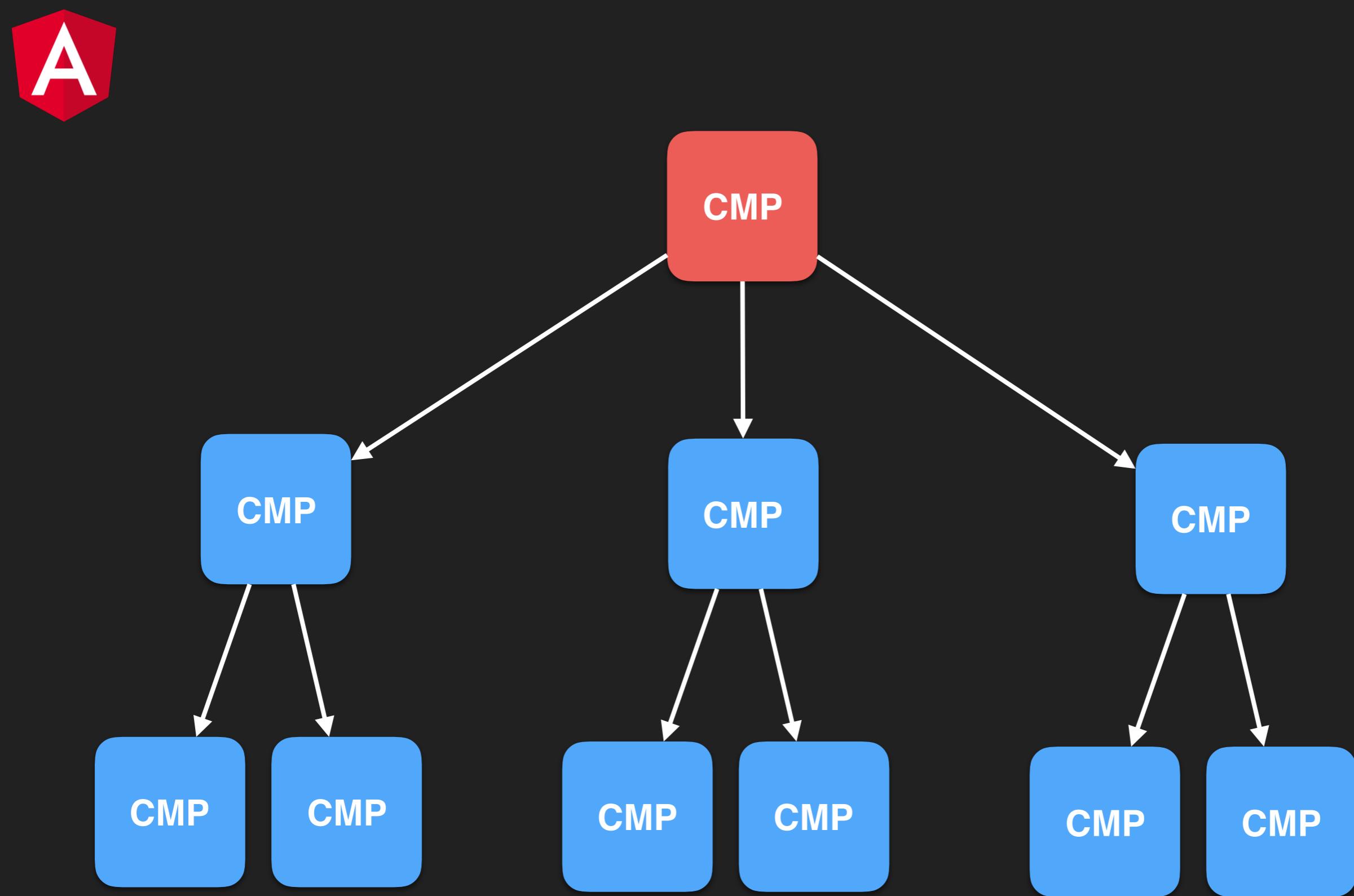


Como funciona?

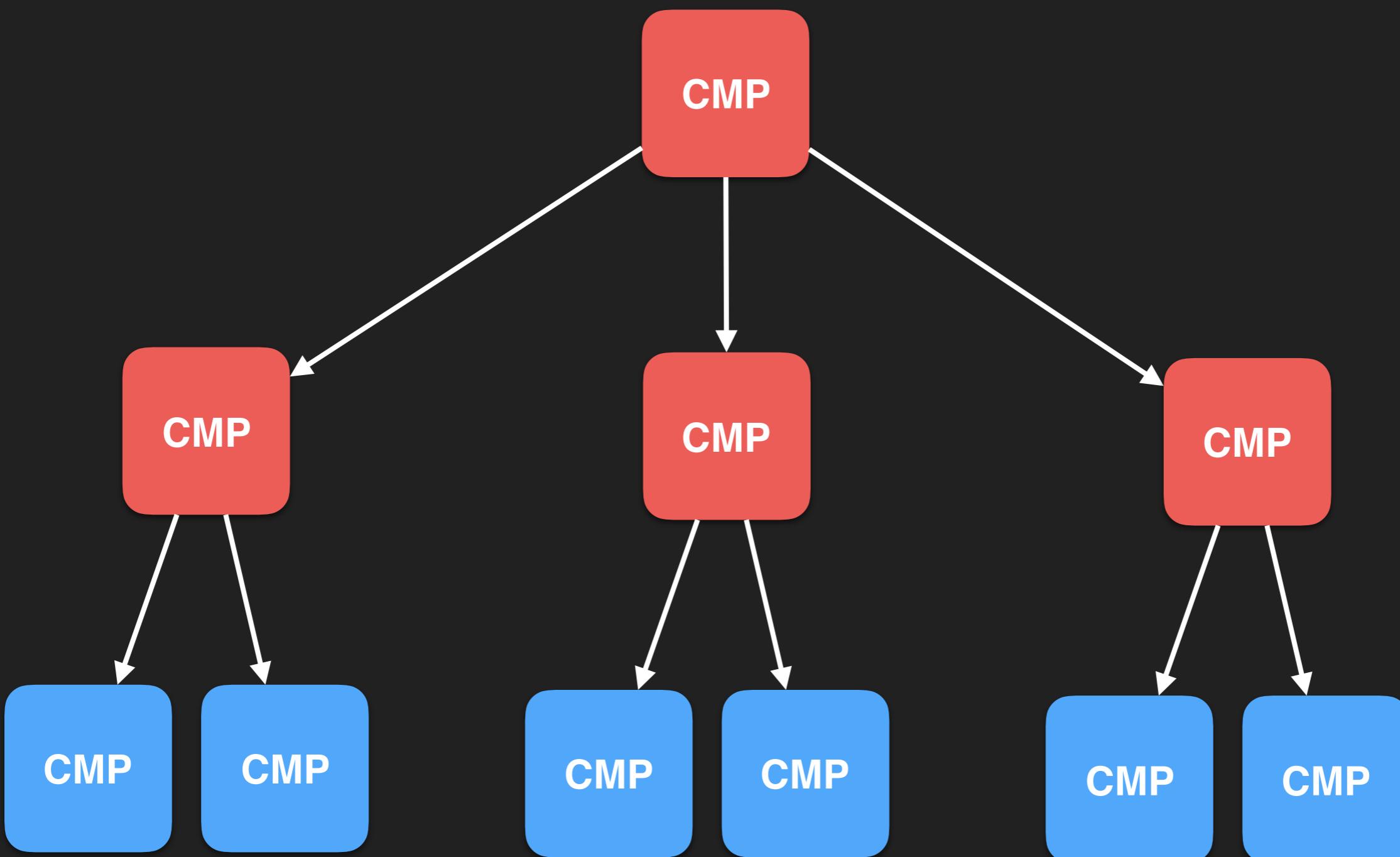
Quando uma mudanças acontece, o Angular compara cada valor da propriedade do Componente com o valor anterior que está sendo usado no Template

Se o valor atual e anterior são diferente, seta o valor do component como isChanged = true;

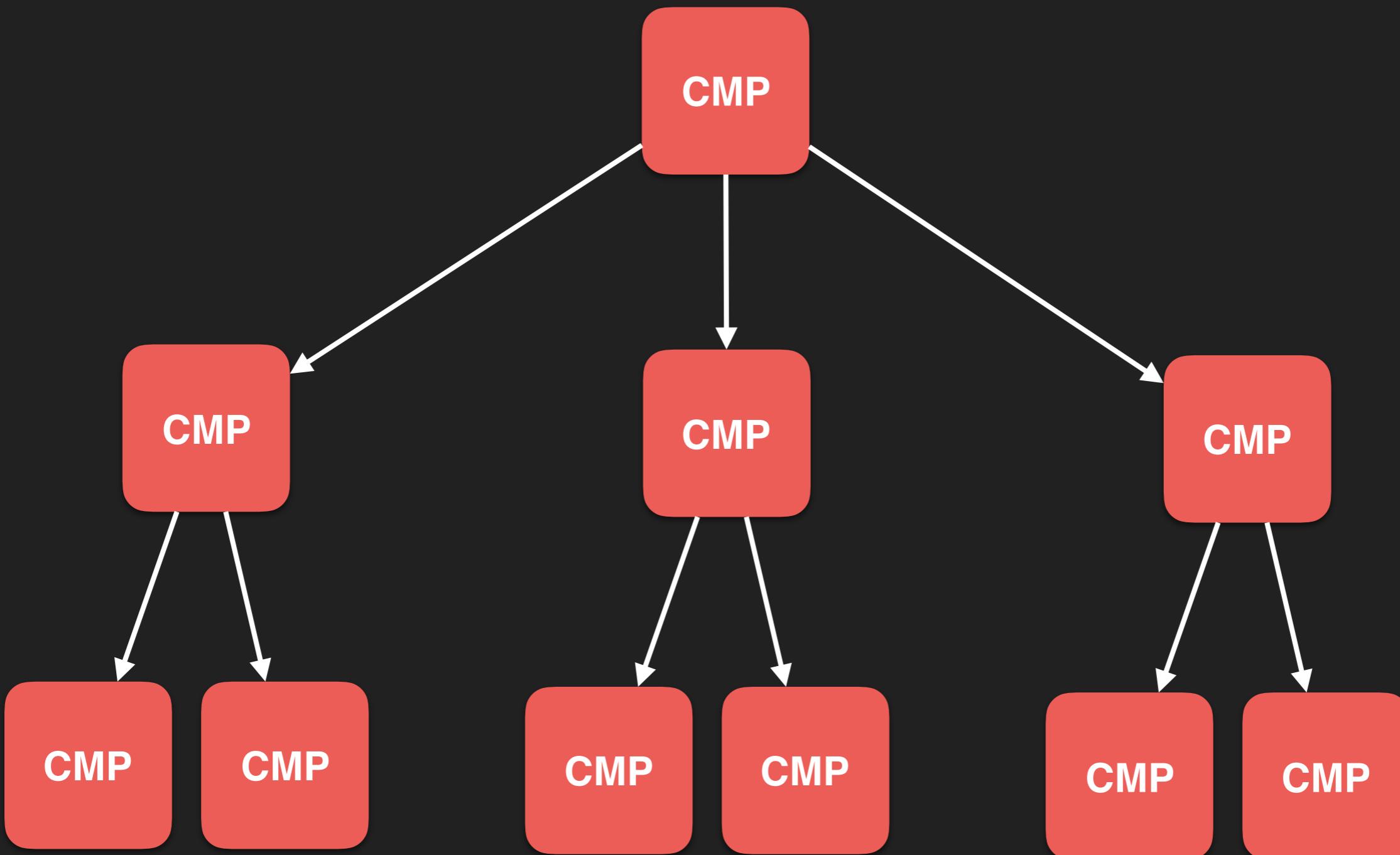




A



A





Como otimizar?



Como funciona?

ChangeDetectionStrategy.OnPush: comparação por referência (bom para imutabilidade)

ChangeDetectionStrategy.Default: padrão, sempre irá fazer uma verificação profunda (por propriedade)



Demo



Prós e Contras:

1. Fluxo unidirecional ✓

2. Debug volta ao tempo (DevTools) ✓

3. Separação do código ✓

4. Fácil debug e bug fix (1, 2, e 3) ✓

5. Mais fácil pra testar devido à funções puras ✓

6. Melhor performance (onPush) ✓

7. Serialização do estado ✓

8. Mais uma camada == mais código -



Arquitetura escalável?





Gerenciamento de estado previsível

Fácil de testar (estado imutável + DI)

Desacoplado de formatos de mensagem e servidor

Serviços não sabem do estado da aplicação

Sistema modular + lazy loading



Pra estudar mais...

- <https://angular.io/docs/ts/latest/guide/reactive-forms.html>
- <https://angular.io/docs/ts/latest/guide/server-communication.html>
- <https://angular.io/docs/ts/latest/guide/router.html>
- <https://github.com/Reactive-Extensions/RxJS/blob/master/doc/gettingstarted/categories.md>
- <https://gist.github.com/btroncone/d6cf141d6f2c00dc6b35>
- <http://rxmarbles.com/>
- <http://reactivex.io/documentation/operators>
- <https://github.com/ngrx>
- <https://github.com/ngrx/example-app>
- <https://auth0.com/blog/understanding-angular-2-change-detection/>
- <http://blog.brecht.io/A-scalable-angular2-architecture/>
- <http://blog.mgechev.com/2016/04/10/scalable-javascript-single-page-app-angular2-application-architecture/>



Código dos exemplos

<https://github.com/loiane/angular-redux-ngrx-examples>

- angular-ngrx-counter
- angular-ngrx-todo (com backend em node.js)
- angular-ngrx-auth-firebase (auth e roteamento)
- angular-ngrx-lms-firebase (auth, roteamento, múltiplos reducers, lazy loading)

<https://github.com/loiane/angular-change-detection-example>

Quer aprender mais sobre Angular (2.x e 4.x)?



Angular 2



Curso de Java Básico Módulo 2:
Gratuito com Certificado e Forum



Curso Estrutura de Dados e
Algoritmos Java



Curso Phonegap / Cordova



ExtJS 4: CRUD MVC - Completo



Curso de Java Básico Gratuito
com Certificado e Forum



Screencasts Extras



Curso Gratuito CSS3 com Sass e
Compass



Curso Gratuito ExtJS 4



/loianegroner

obrigada



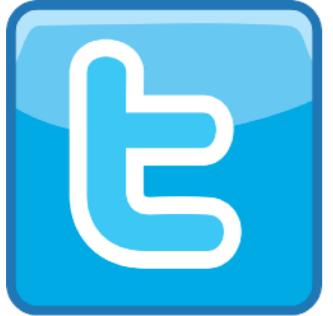
<http://loiane.training>



<http://loiane.com>



<facebook.com/loianegroner>



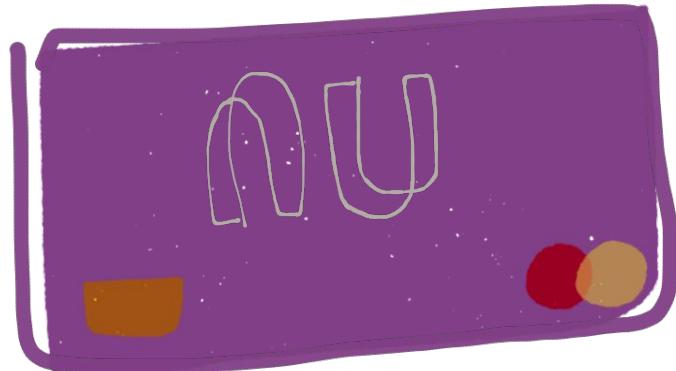
<twitter.com/loiane>



<https://github.com/loiane>



<youtube.com/loianegroner>



Infraestrutura de Dados no nu bank



André
Mideo
@andremideo

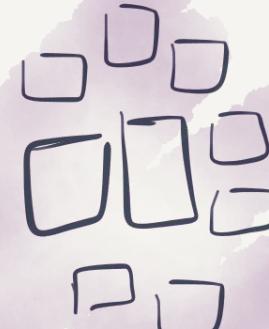
Alessandro

Androni

@metamatema



Data



Infra

B
I
L
L
S

F
I
N
A
N
C
E

A
U
Q
I
S
-
T
I
O
N

DO
DO
DO
DO

PLATFORM 

B
I
L
L
S

F
I
N
A
N
C
E

A
C
Q
S
-
T
I
O
N

Do
Do
Do
Do

PLATFORM 

Data Infra 

Plataforma

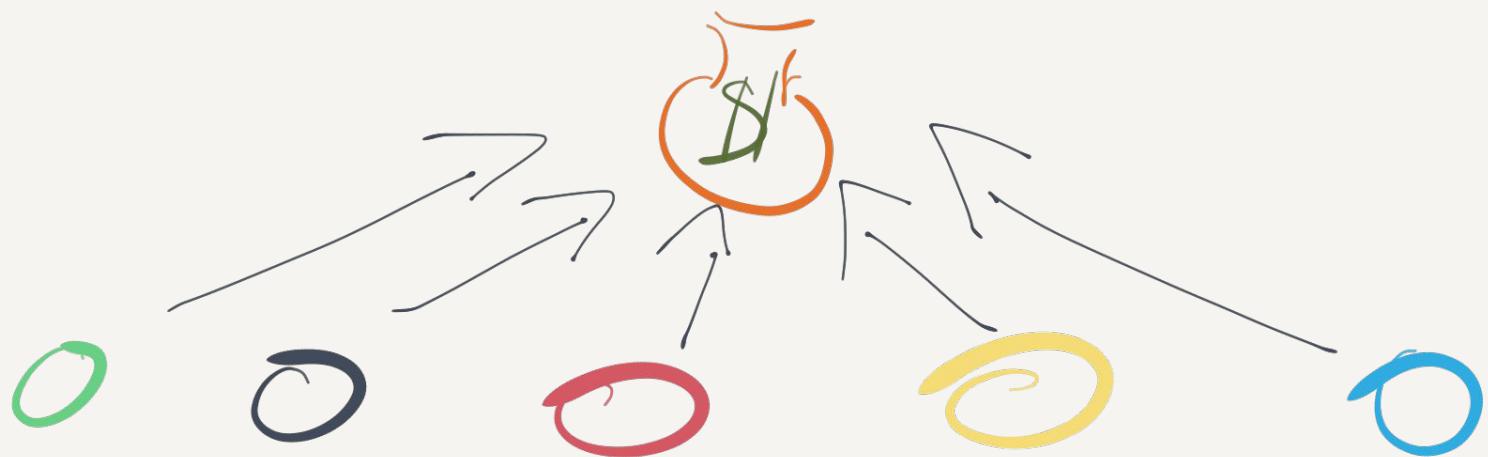
Dados



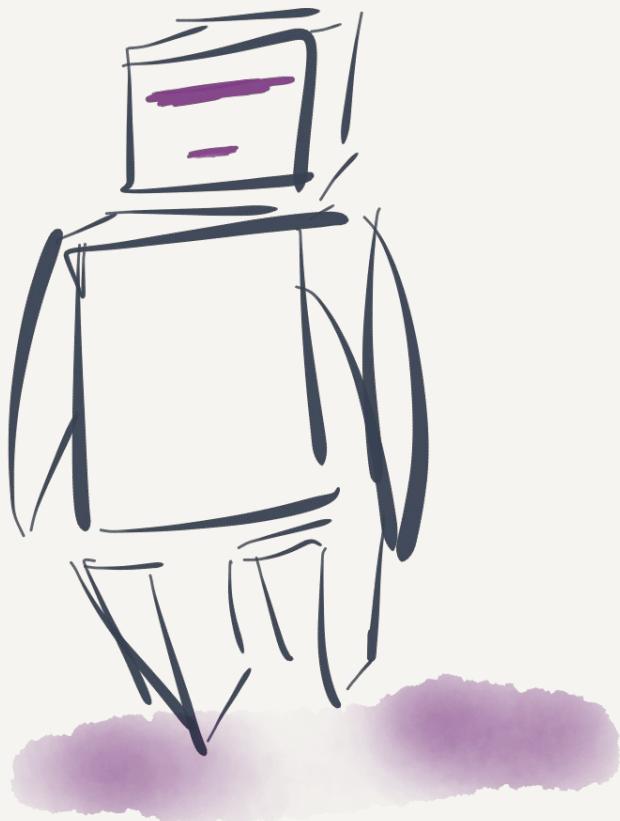
EMPODERAR



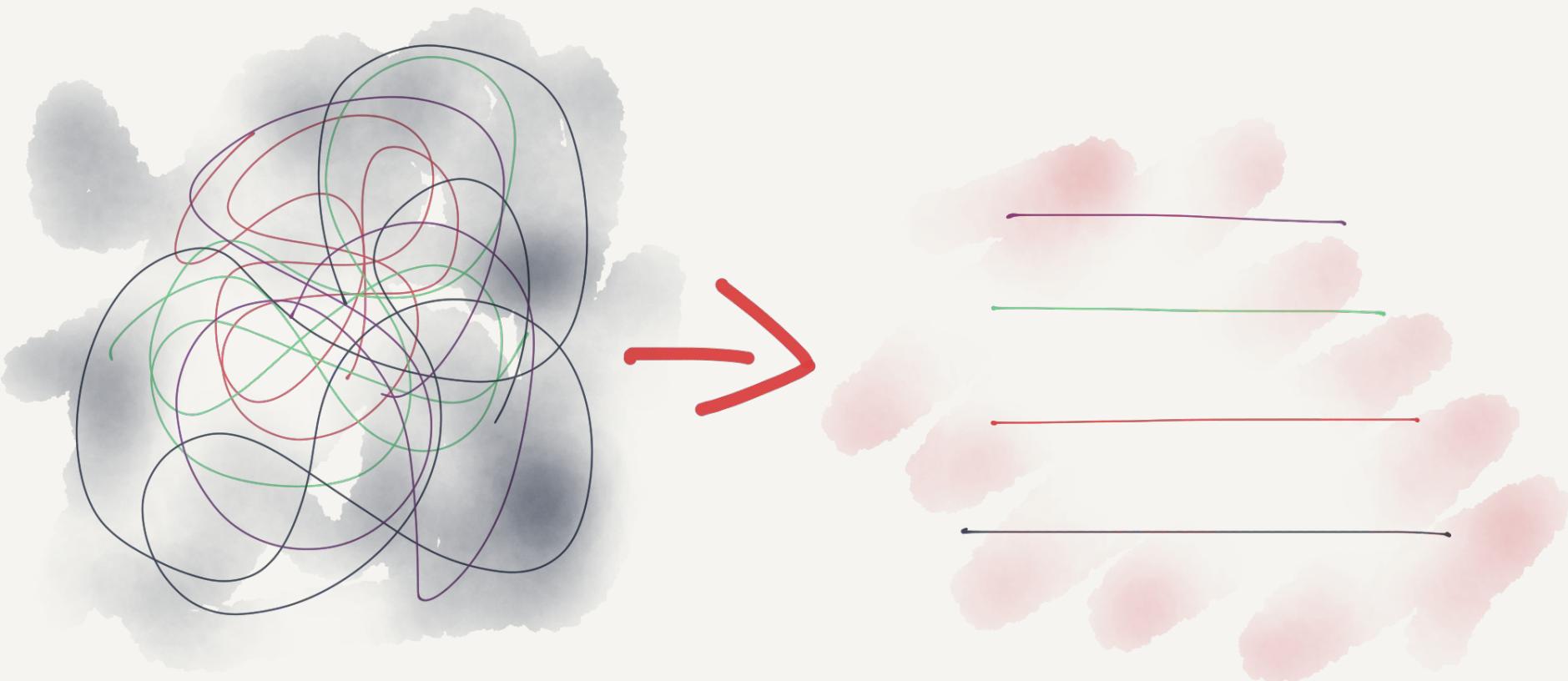
Descentralizar



Autonam + IZIAR



Simplificar



E...

APRENDÉMOS

ALGUMAS

COISAS



WARNING

Tudo isso vai

mudar em 6
meses !!!
600

O?
O.

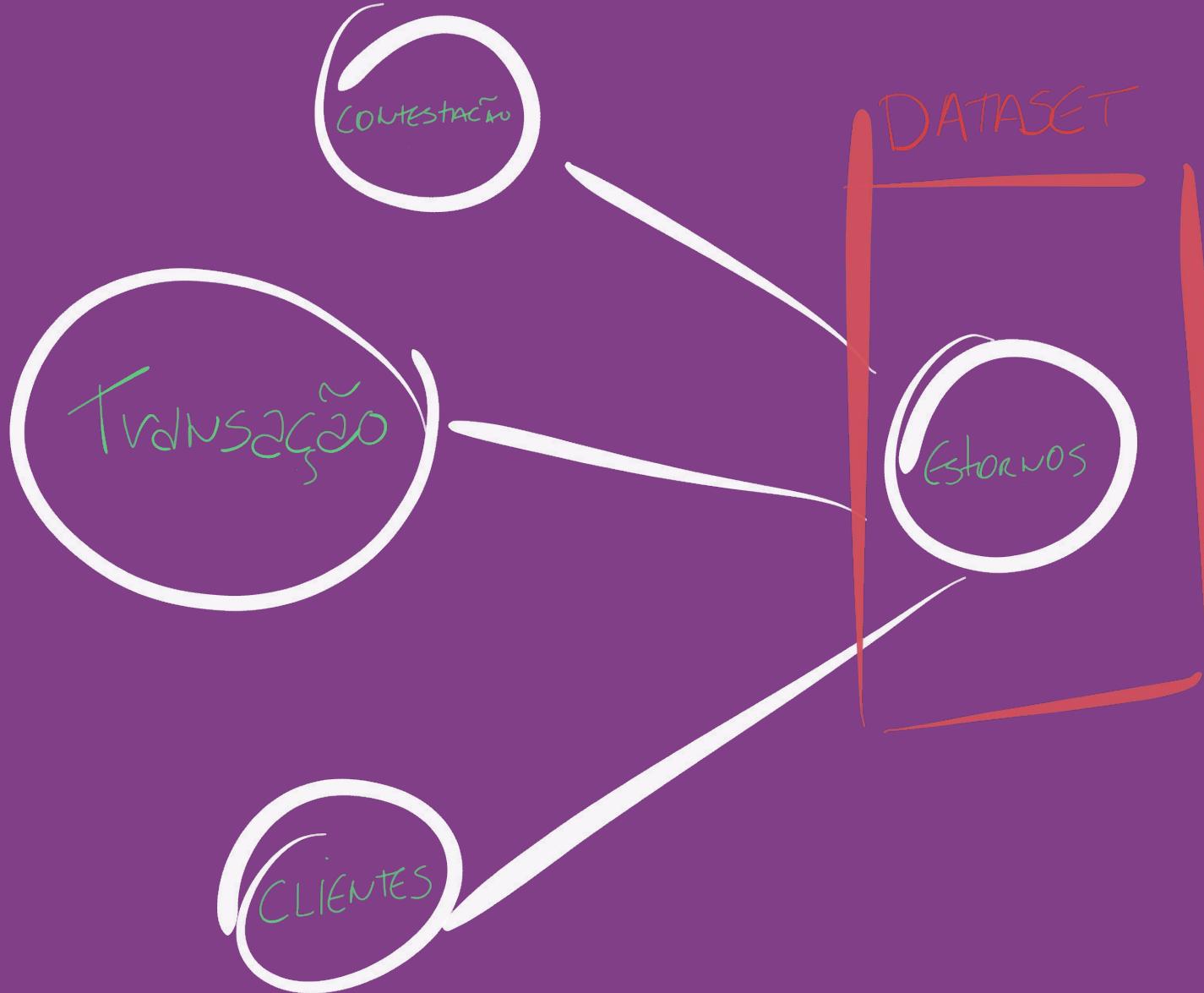
Preciso construir
um DATASET
SOBRE Estor WOS

Como?

f(Todos os Dados)

-stornos

```
SELECT SUM(transaction__amount)
FROM chargebacks
JOIN transactions
JOIN customers
WHERE chargebacks.status = 'Estornado'
```



Contratos

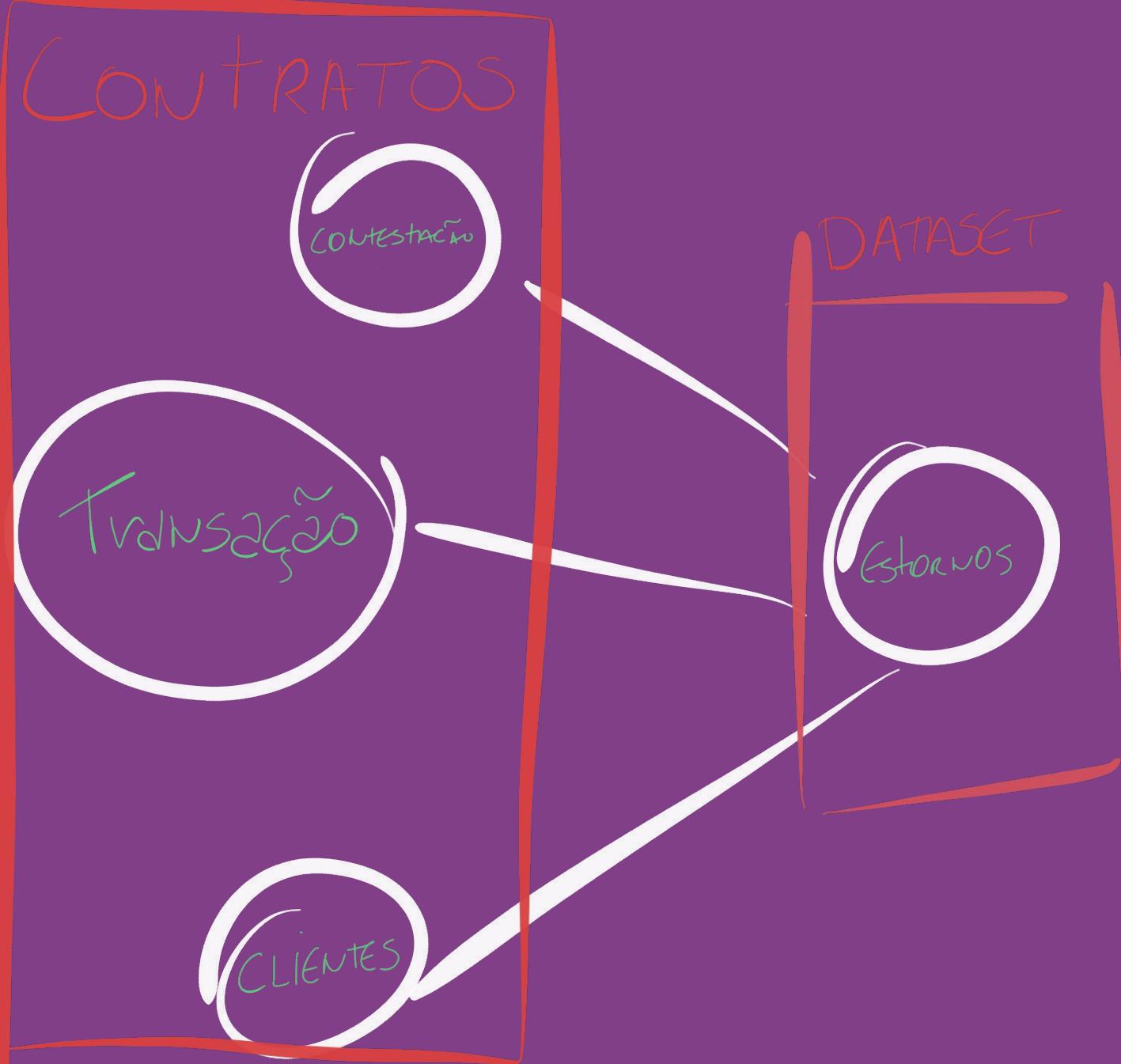
Contestação

Transação

CLIENTES

DATASET

Estornos



Descentralizado

```
(def skeleton
{:enrollment/id
  {:schema Uuid :id true :required true
   :eg #uuid "afb6f3ee-c698-42cf-9276-b5780fc0c7a1"
   :contract/history true}
 :enrollment/customer-id
  {:schema Uuid :index true :required true
   :eg #uuid "b227594c-9ff5-11e4-8c68-7831c1ce75c6"
   :contract/name :customer/id}
 :enrollment/account-id
  {:schema Uuid :index true :required true
   :eg #uuid "884f4585-10a6-4fe1-ae23-c8f04a88253f"
   :contract/name :account/id}
 :enrollment/product
  {:schema Product :required true
   :doc "Product for this enrollment"}
 :enrollment/plan
  {:schema Plan :required true
   :contract/history true
   :doc "Plan associated with this enrollment.
The plan represents all the billing aspects of the enrollment:
amount, frequency etc.
The plan associated with an enrollment may change, creating a new subscription"}
 :enrollment/status
  {:schema Status :required true
   :eg :enrollment.status/active
   :contract/history true
   :doc "Current status of the enrollment
(trial, active, unsubscribed, canceled...)"}))
```

Contrato E modelo do serviço

DADOS
(OPERACIONAIS)



CONTRATO

MAPEAMENTO

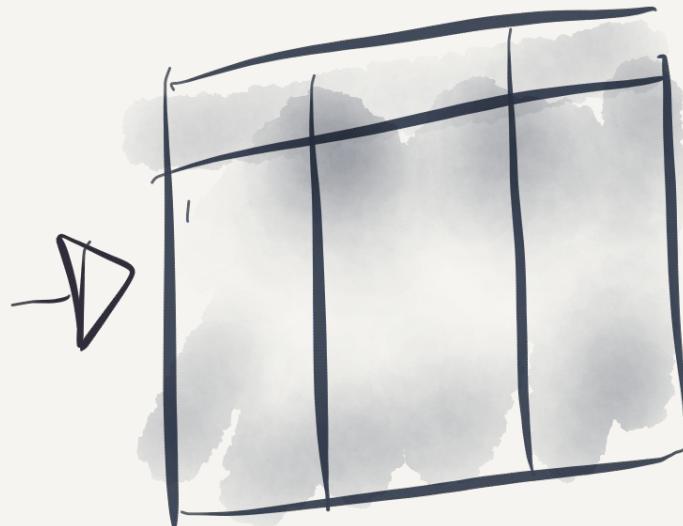
Correções

PII

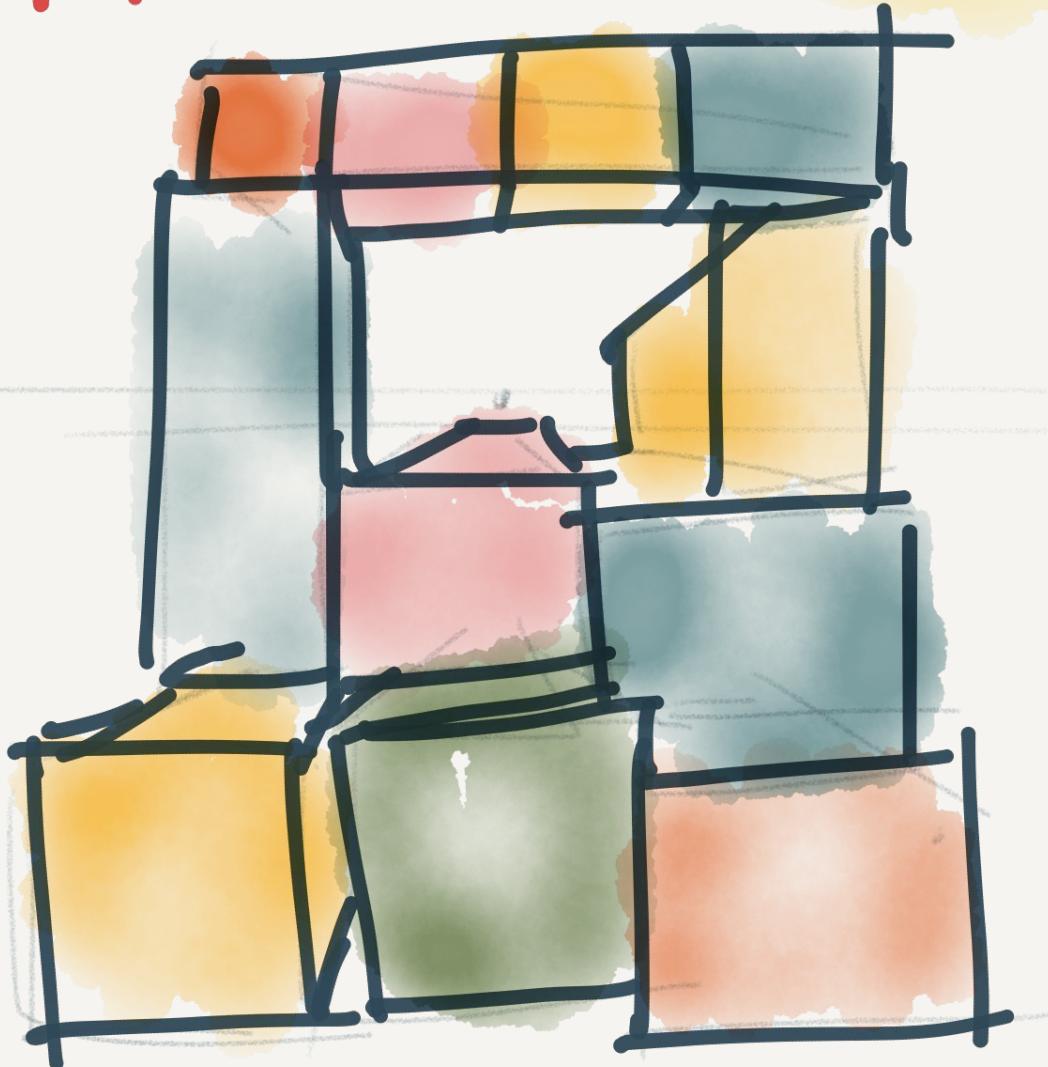
SHARDING

Integridade

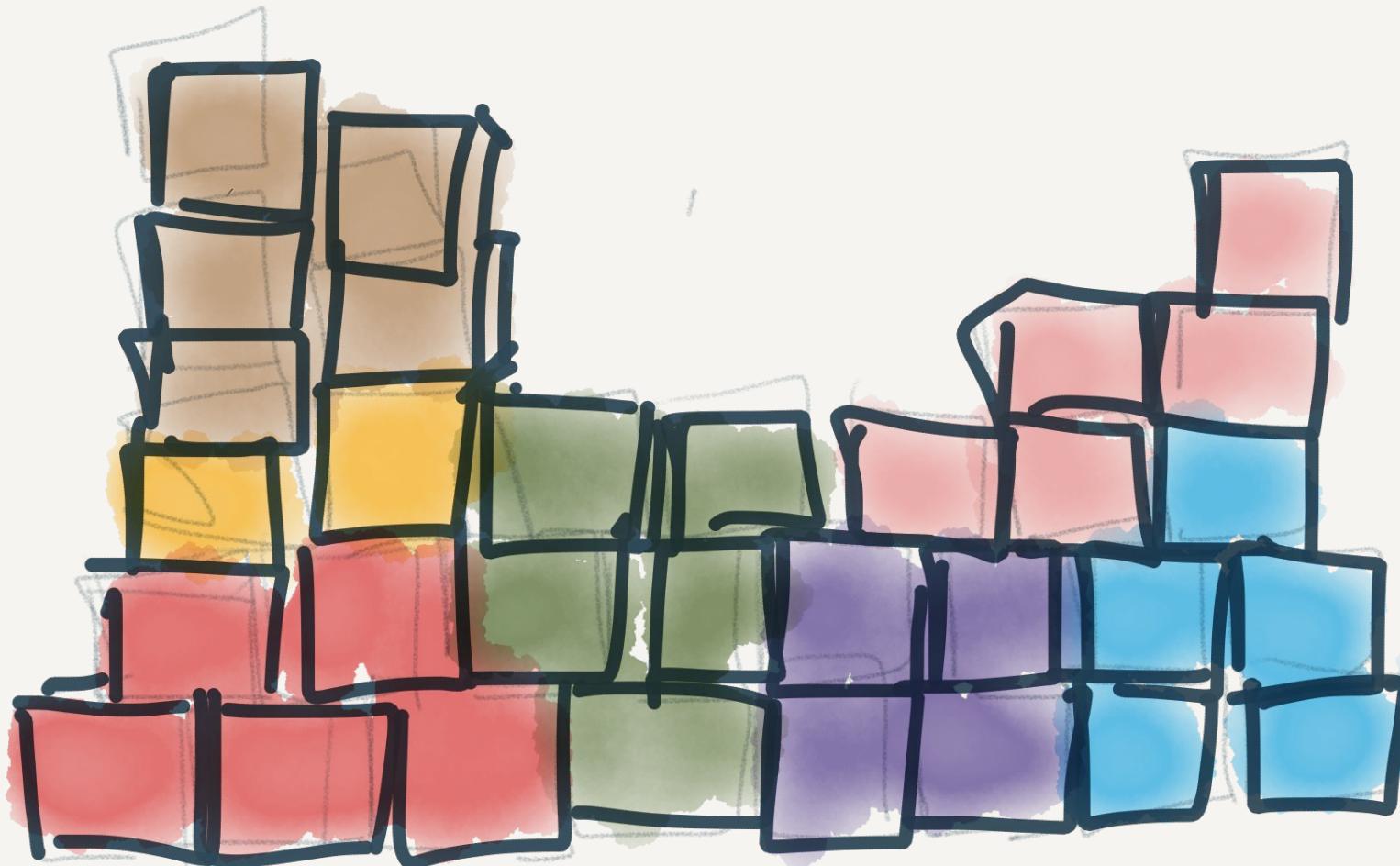
DADO PARA
,
ANÁLISE



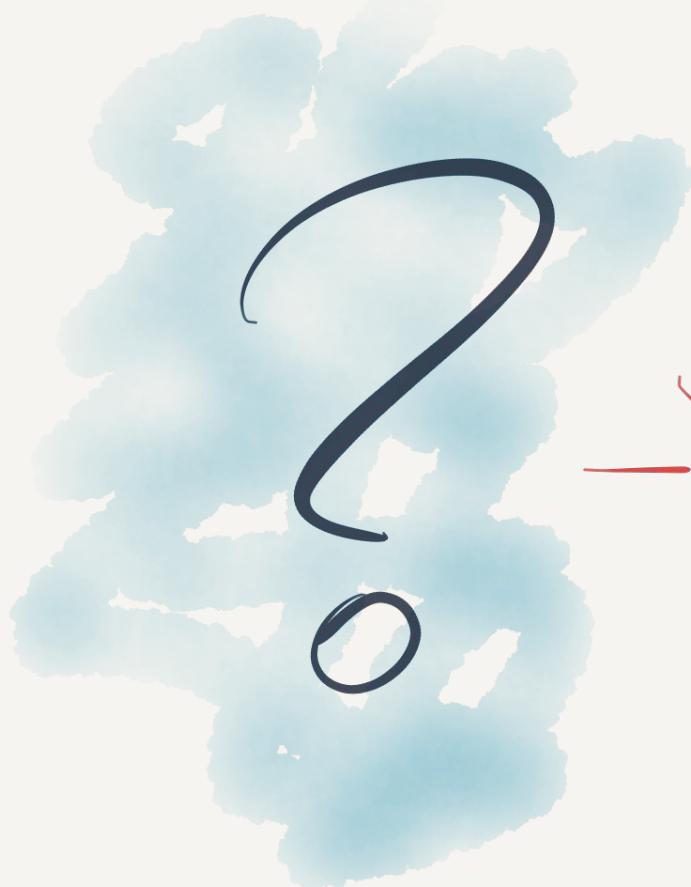
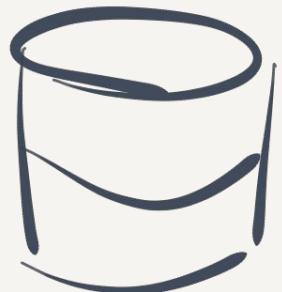
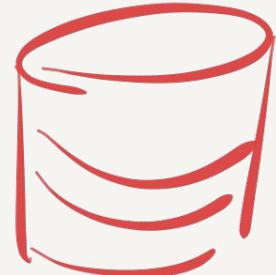
PAIN ON READ

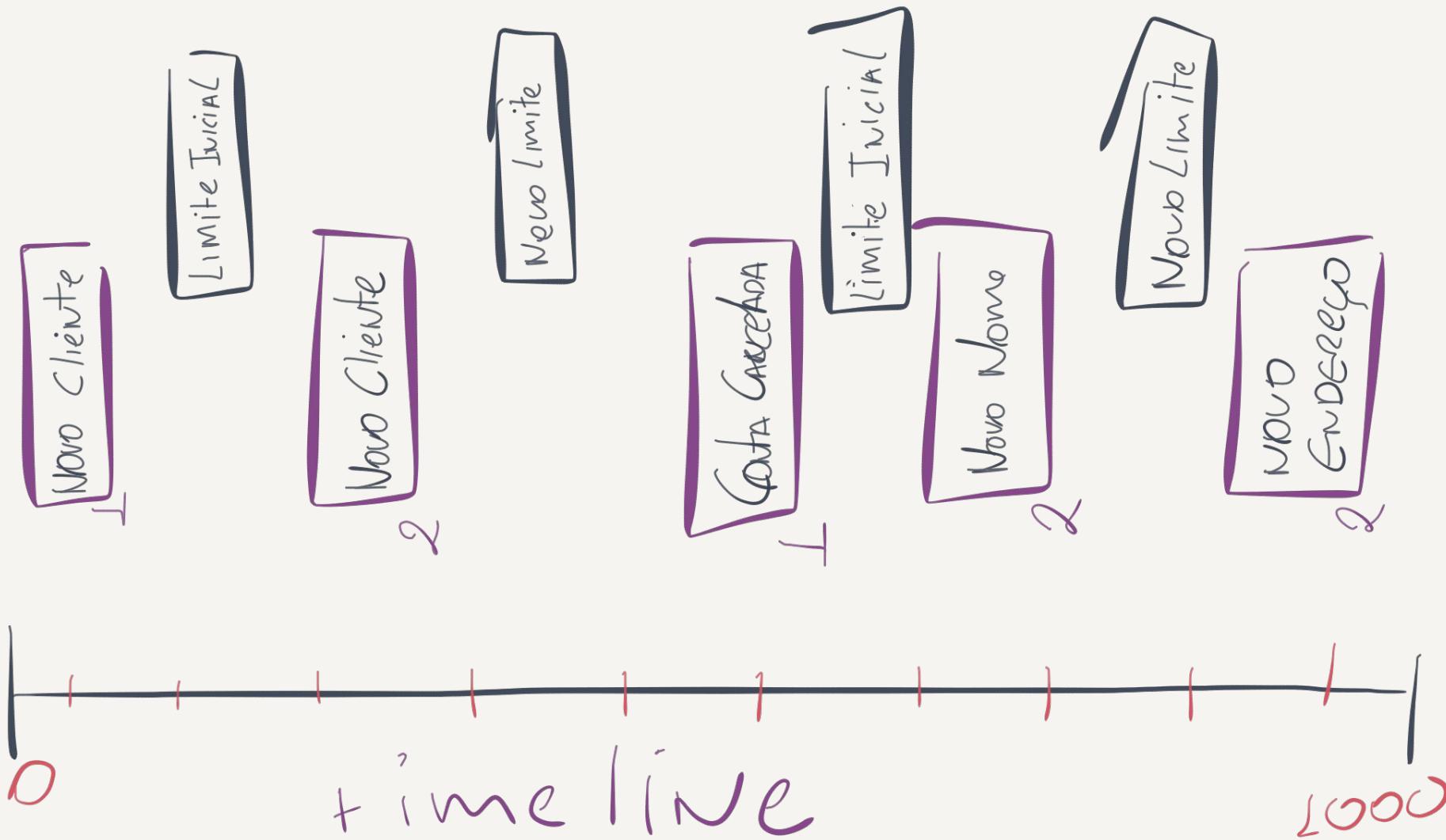


PAIN ON WRITE

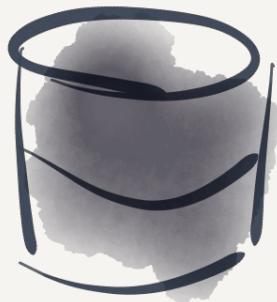


DADOS DE
PRODUÇÃO





DADOS DE
PRODUÇÃO



SERVICIO

ULTIMARIO
A

B

C



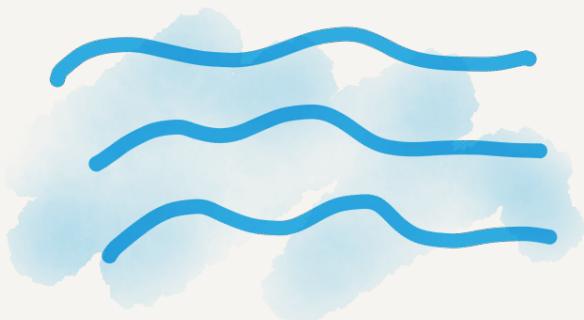
LOG
EXTRAÍDO

FORMATO
ÚNICO



Contrato
A

B



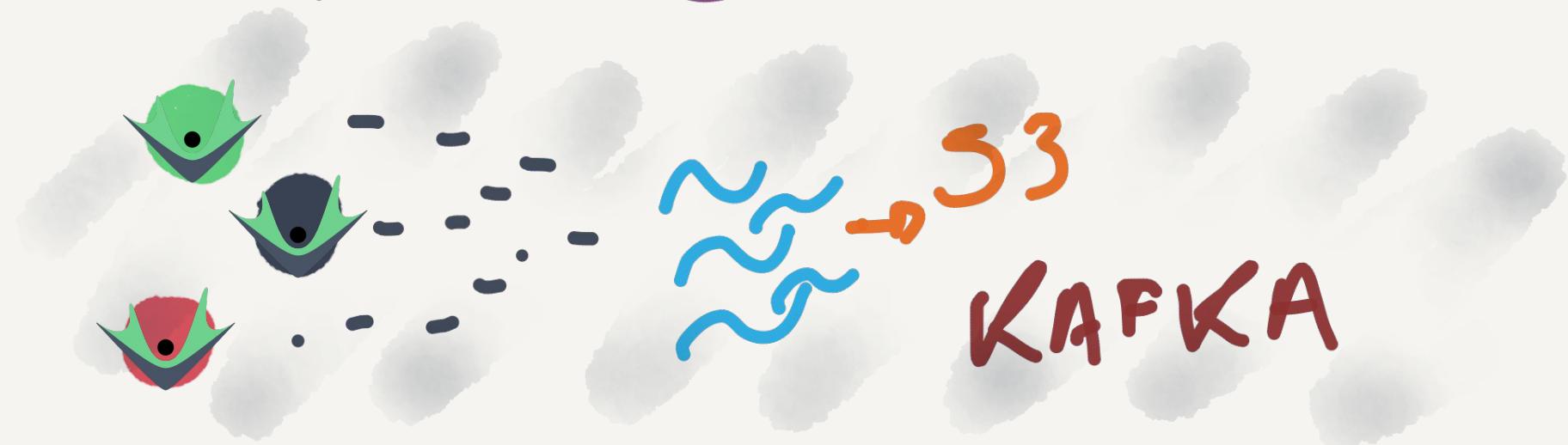
Correnteza

Extractor Streaming

DE NOSSOS BAUCOS

DATOMIC

- Schema PADRÃO
- S3 ou KAFKA
- Auto Corretivo

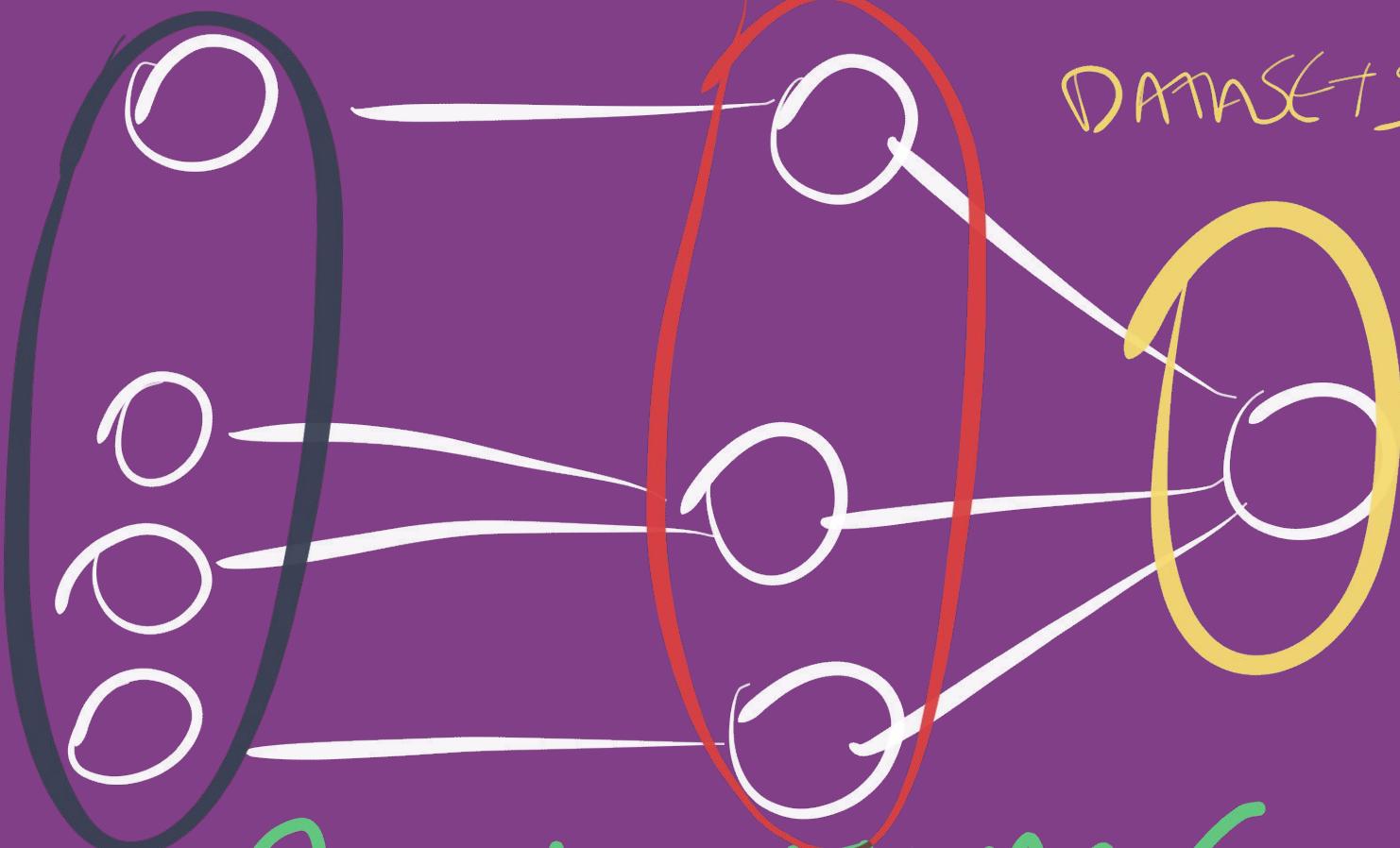


LOGS

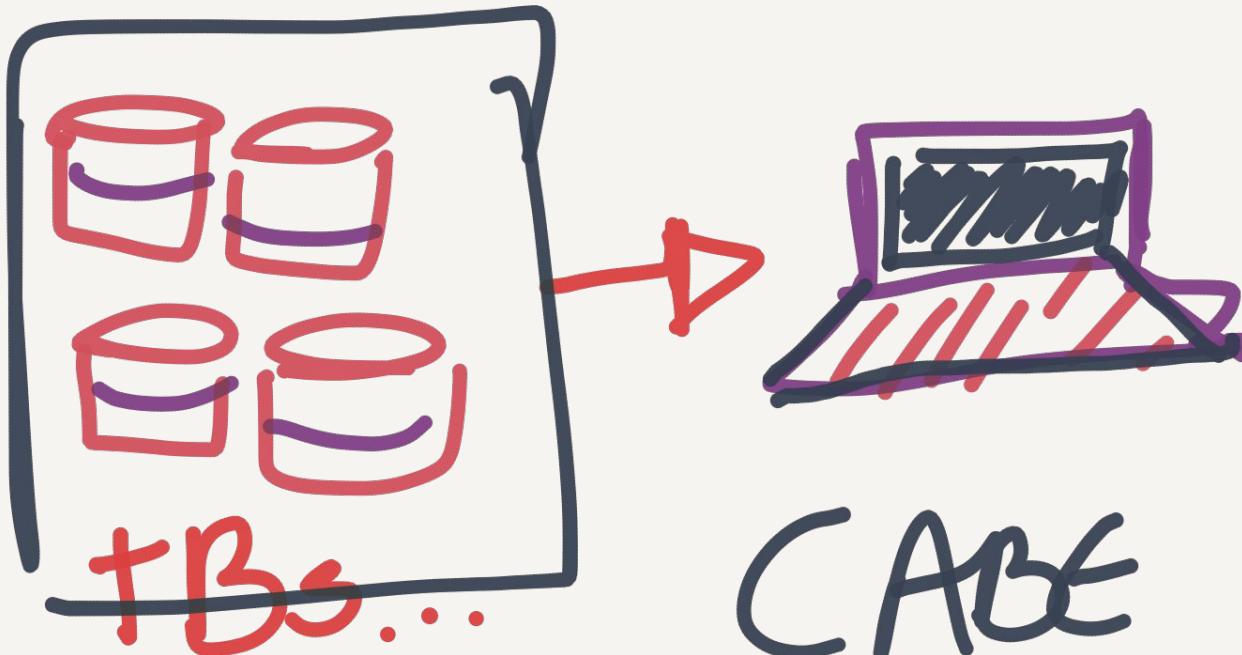
CONTAINERS

DATASETS

RUNTIME



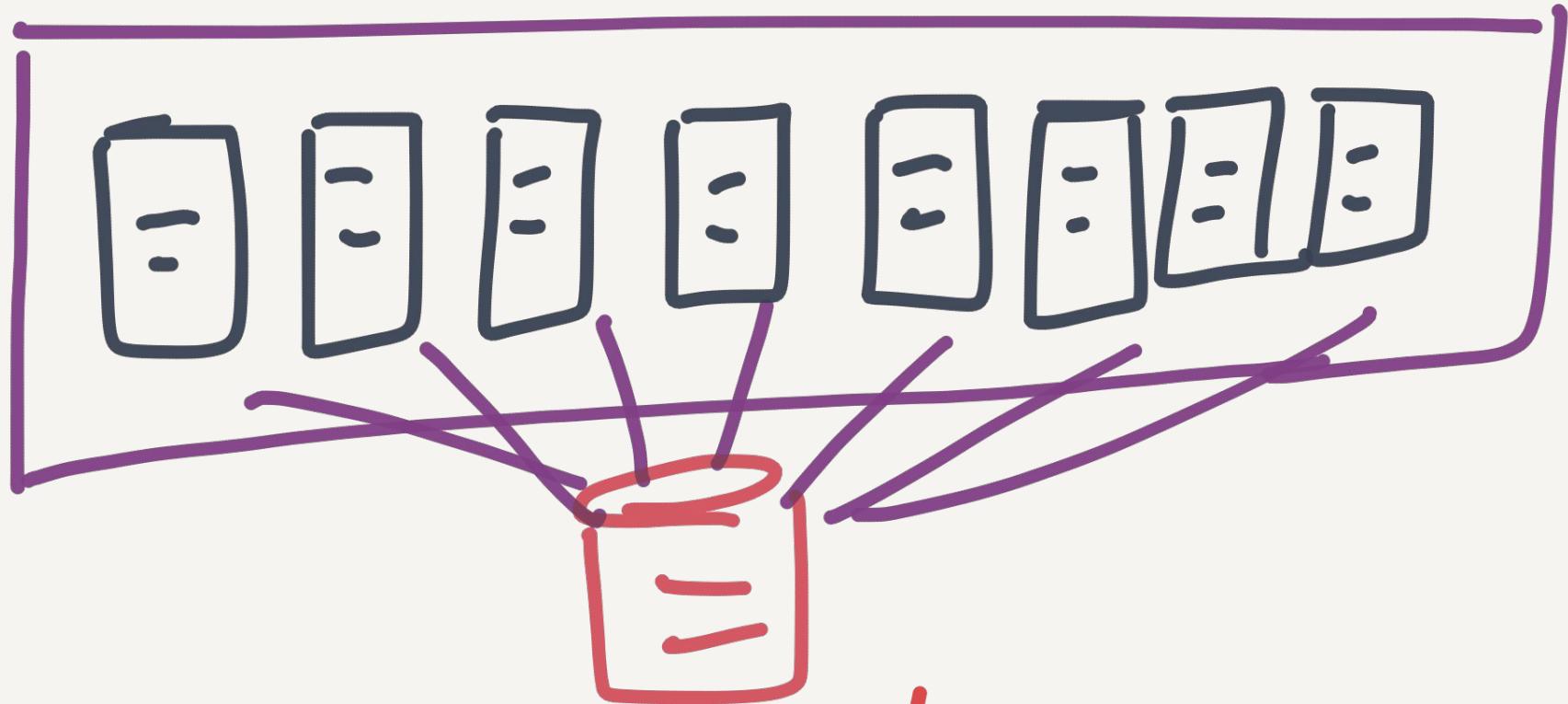
Como
Processamos?



~~TBs...~~

CABE NA
MEMORIA?

:(Nope...



PROCESSAMENTO
DISTRIBUÍDO



Uhhmmmm

e o o

???

Vou tentAR ESSE

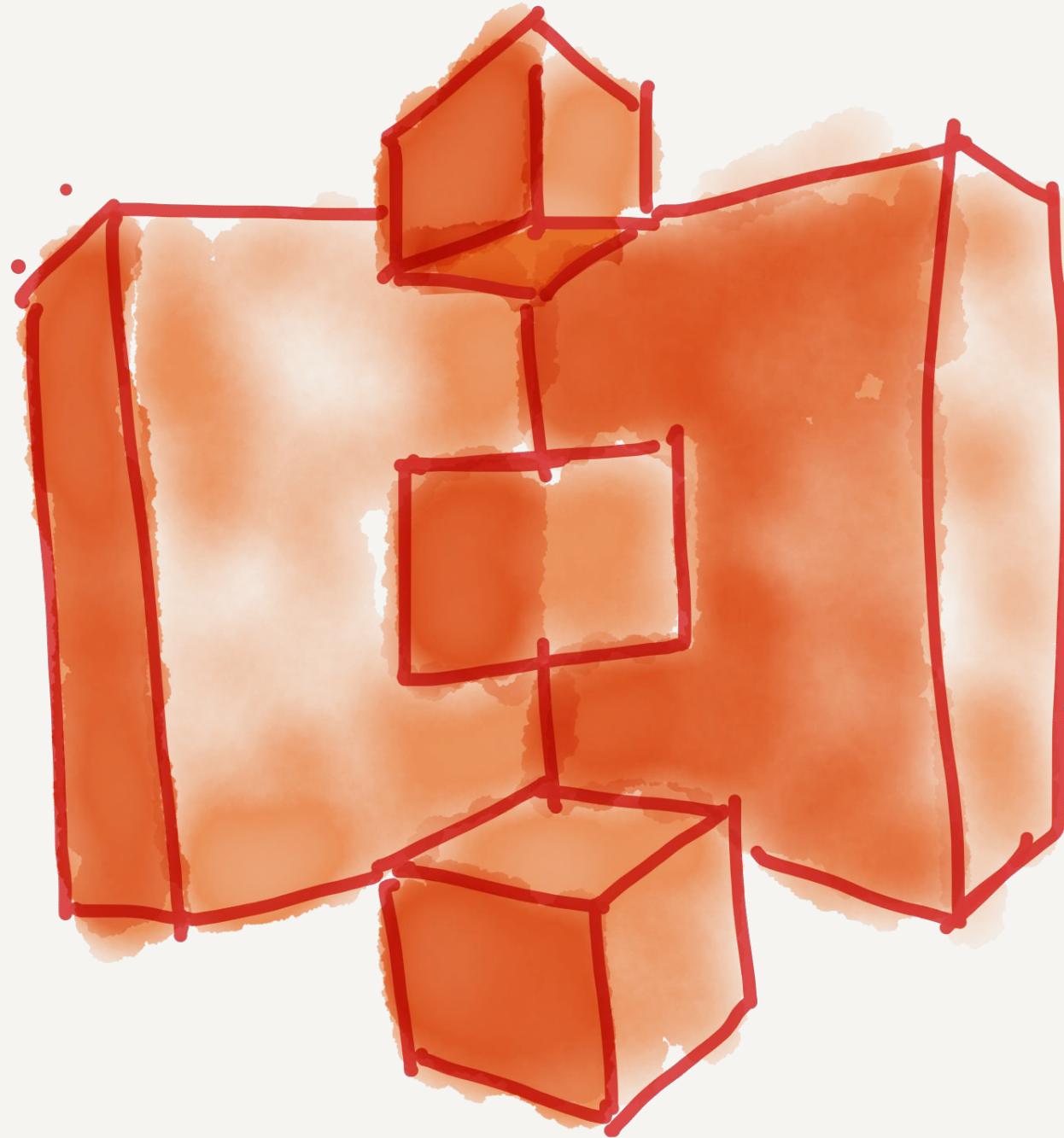
MAPReduce

```
public class WordCount {
    public static class Map extends MapReduceBase implements Mapper<LongWritable, Text, Text, IntWritable> {
        private final static IntWritable one = new IntWritable(1);
        private Text word = new Text();
        public void map(LongWritable key, Text value, OutputCollector<Text, IntWritable> output, Reporter reporter) throws IOException {
            String line = value.toString();
            StringTokenizer tokenizer = new StringTokenizer(line);
            while (tokenizer.hasMoreTokens()) {
                word.set(tokenizer.nextToken());
                output.collect(word, one);
            }
        }
    }
    public static class Reduce extends MapReduceBase implements Reducer<Text, IntWritable, Text, IntWritable> {
        public void reduce(Text key, Iterator values, OutputCollector<Text, IntWritable> output, Reporter reporter) throws IOException {
            int sum = 0;
            while (values.hasNext()) {
                sum += values.next().get();
            }
            output.collect(key, new IntWritable(sum));
        }
    }
    public static void main(String[] args) throws Exception {
        JobConf conf = new JobConf(WordCount.class);
        conf.setJobName("wordcount");
        conf.setOutputKeyClass(Text.class);
        conf.setOutputValueClass(IntWritable.class);
        conf.setMapperClass(Map.class);
        conf.setCombinerClass(Reduce.class);
        conf.setReducerClass(Reduce.class);
        conf.setInputFormat(TextInputFormat.class);
        conf.setOutputFormat(TextOutputFormat.class);
        FileInputFormat.setInputPaths(conf, new Path(args[0]));
        FileOutputFormat.setOutputPath(conf, new Path(args[1]));
        JobClient.runJob(conf);
    }
}
```

APACHE
Spark

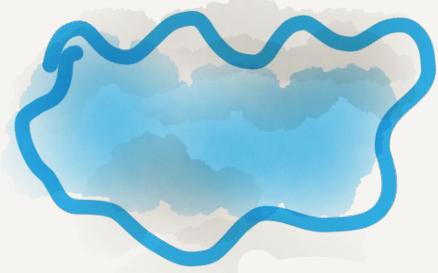


```
val ds = sqlContext.read.text("s3a://bucket/datasets/lines")
val wordCount =
  ds
    .flatMap(_.split(" "))
    .filter(_ != "")
    .groupByKey(_.toLowerCase())
    .count()
```



S3 como A estrutura
de nosso Ecosystem

"INFINITO"



CUIDADO COM SUAS PARTIÇÕES



s3://seu-bucket/key/prefix/data/formato/name



s3://seu-bucket/18196a77-53ca-4047 ...

DATAFRAME

```
val estornos =  
    chargebacks.join(customers, $"customers.id" == $"chargebacks.customer")  
        .join(transactions, $"transactions.id" === $"chargebacks.transaction")  
        .where($"chargebacks.status" === "Estornado")  
        .groupby("customers.id").sum($"transaction_amount")
```

Estornos é apenas uma representação
da computação

Isso possibilita.....

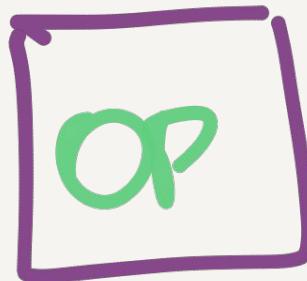
Construindo abstrações simples e reutilizáveis

```
import org.apache.spark.sql.DataFrame

trait SparkOp {
    val name: String
    val inputs: Set[String]
    val definition: Map[String, DataFrame] => DataFrame
}
```

Não precisa se preocupar com formato, partições, tamanho do cluster, caminhos de entrada e saída...

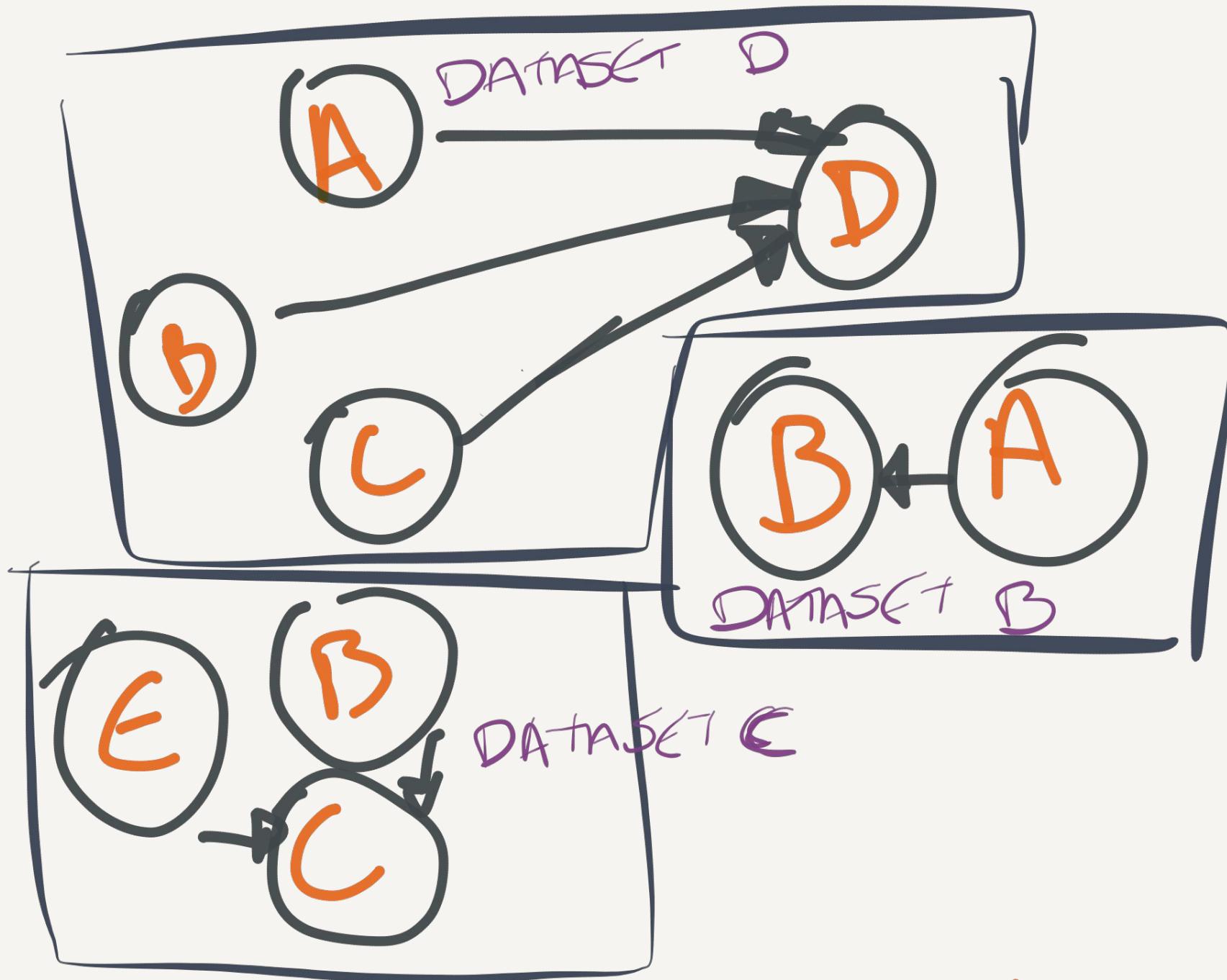
ITAPIU



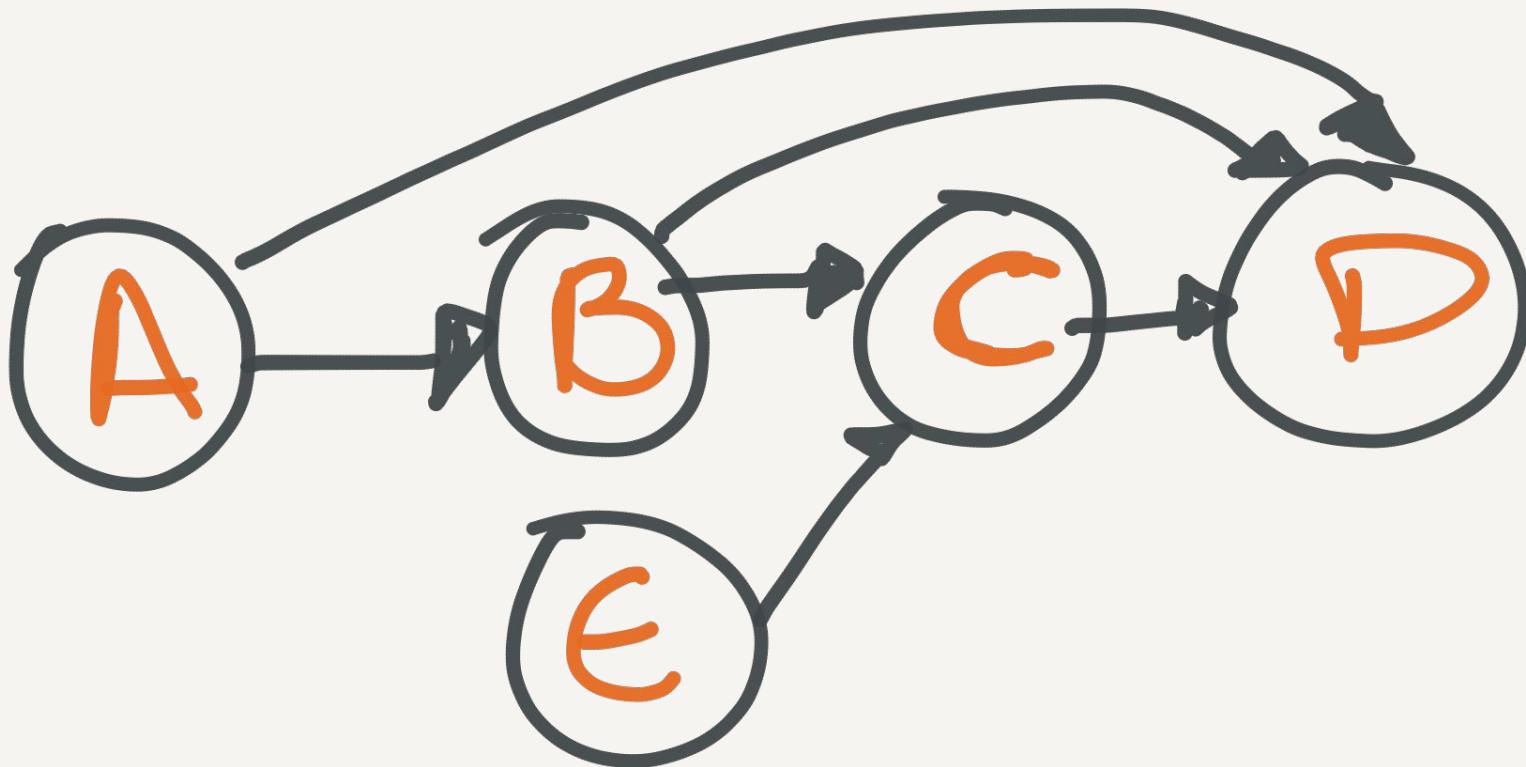
Especializações

MLOP → XgBoost

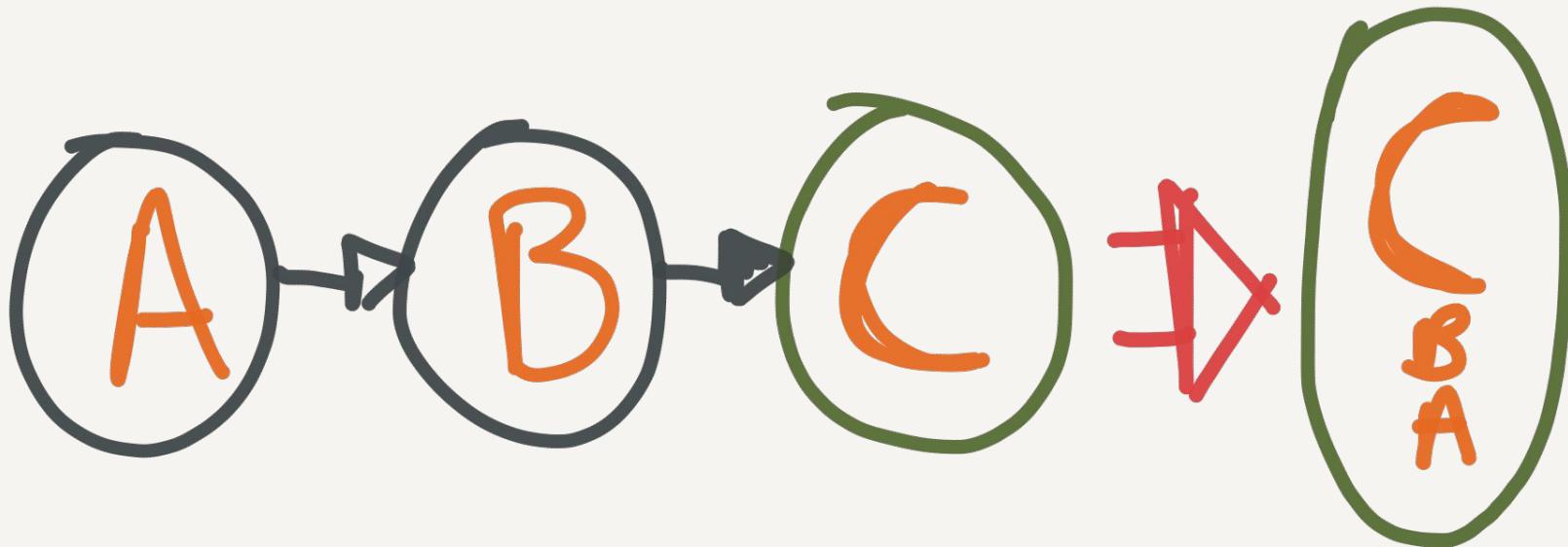
DimOp → Dimensional Modeling



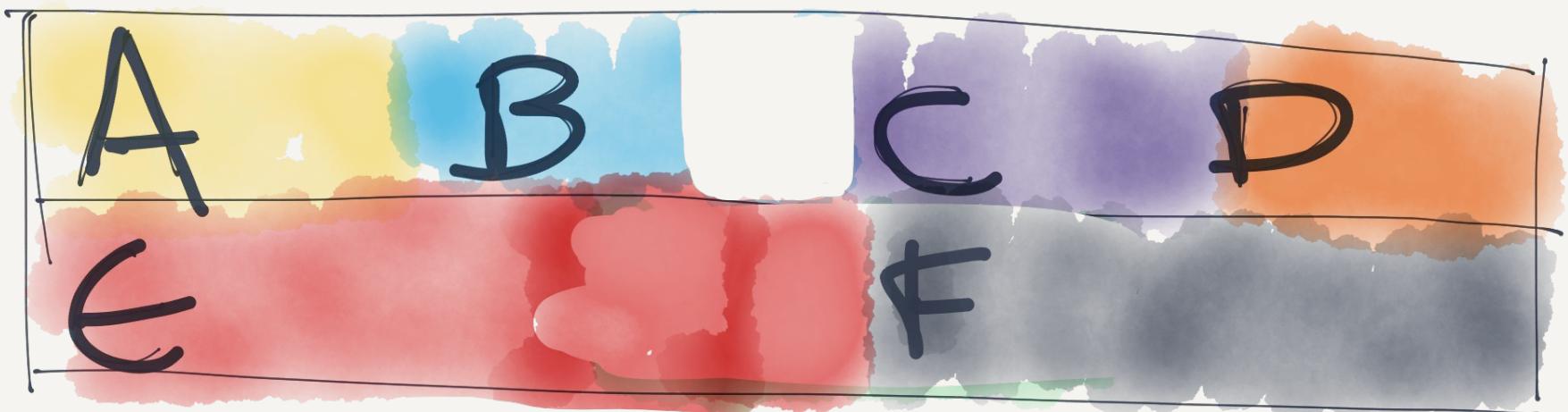
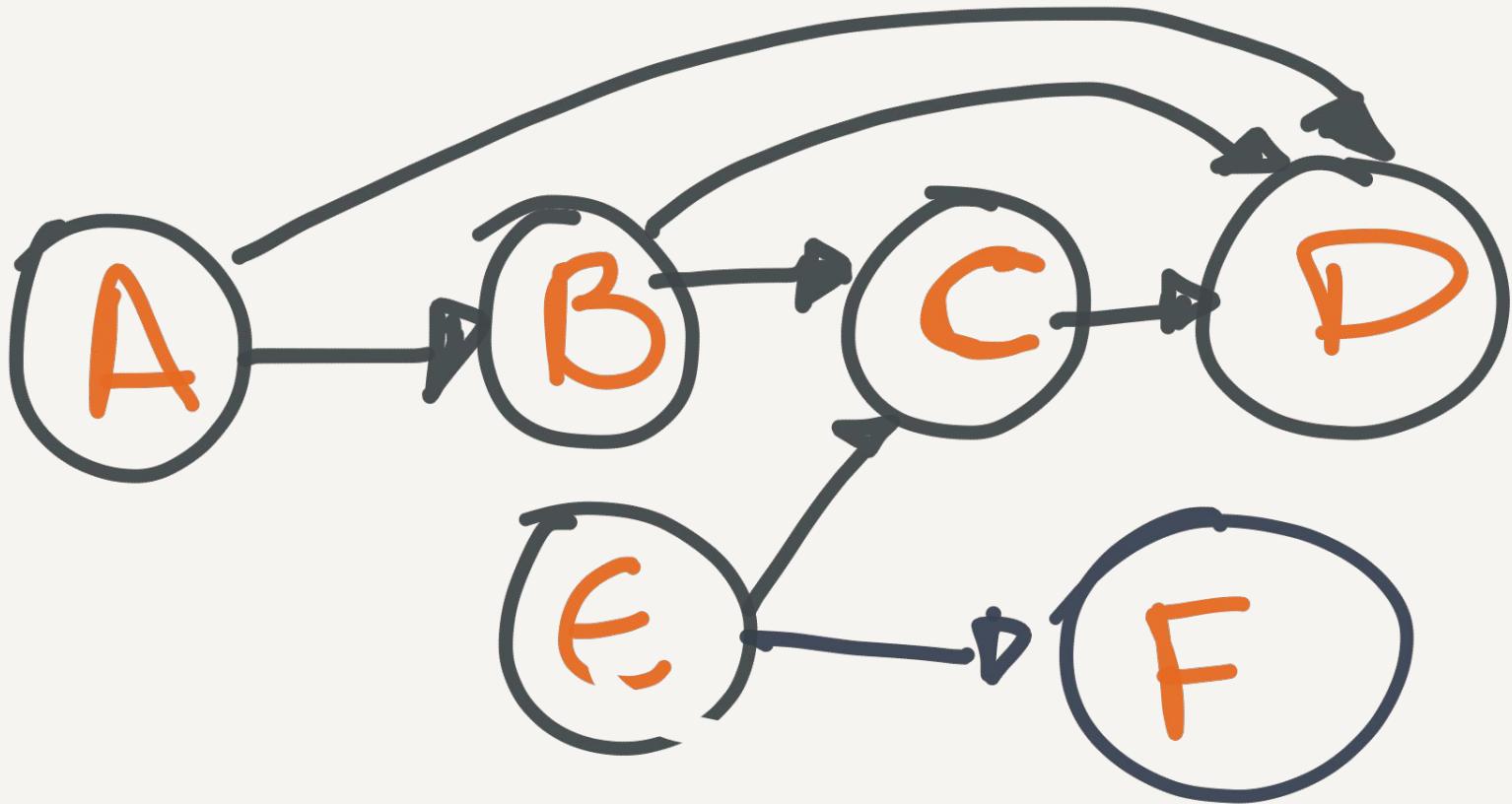
DAG



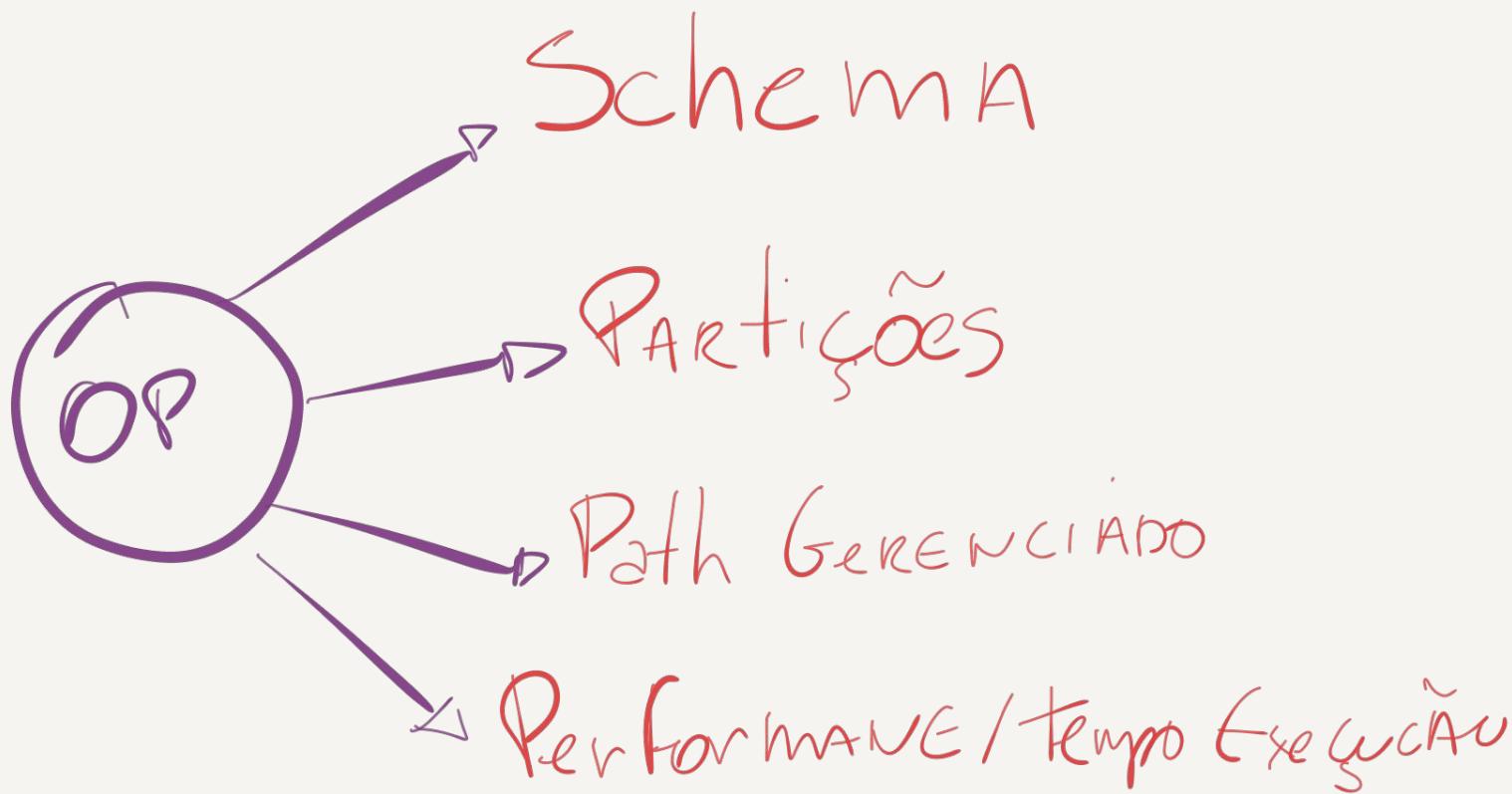
Compilador



InLine



METADADOS



Otimizar Partições (com custo BAIXO)

Ontem

100GB



Coalesce

2000
partições

Outputs

Op → Redshift
with Schema

Logical types



41600+

Spark Ops



Recap

Visite Nossa
STAND

(teremos sorteios)

Open Land!

QCON SP, APRIL 24TH, 2017

322/487

Styling a design system based on Bootstrap 4 and Web Components

@andresgalante



Hola!

@andresgalante





danielpassos 12:03 PM

Palestre em portugues

As pessoas amam quem nao e nativo e fala portugues

Ninguem liga se vc fala alguma coisa errada ou nao

pt.R

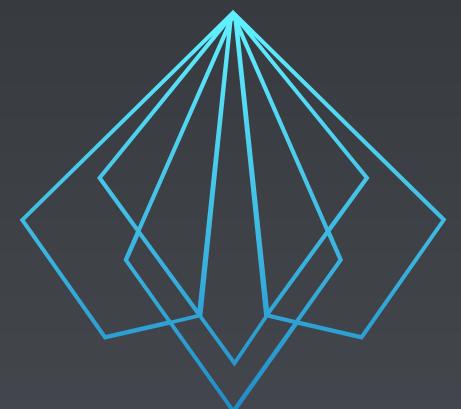
327/487



redhat.[®]

UXD

EXPERIENCE MATTERS



P A T T E R N F L Y

CSS MODULARITY

328/487



Agenda

329/487

Design Systems

CSS Specificity

Shadow DOM

Good CSS Architecture

The PatternFly Use Case

Q&A

330/487



PATTERNFLY

CODE
IS MY
SPIRIT
ANIMAL

EVOXX
STATES



Um argentino incomoda
muita gente...

332/487

A wide-angle photograph of a coastal road at sunset. The sky is filled with dramatic, wispy clouds in shades of orange, yellow, and blue. A winding asphalt road curves along the right side of the frame, bordered by a white guardrail. In the distance, a person is walking on a grassy hillside to the left of the road. The overall atmosphere is peaceful and inspiring.

LET'S START...

333/487

Design systems.
The road to consistency
and modularity.



“A design system is everything
that makes up your product.”

– Mark Otto

<https://speakerdeck.com/mdo/build-your-own-bootstrap>

Consistency





“...when things always behave the same, users don't have to worry about what will happen. Instead, they know what will happen based on earlier experiences”

– Jakob Nielsen

<https://www.nngroup.com/articles/top-10-mistakes-web-design/>

$$5 + 8 = ?$$

$$5 + 8 = 13$$

$$123 + 34 = ?$$

$$123 + 34 = 157$$

$$123 + 34 = ?$$

$$123 + 34 = 157$$

Cache Memory



Average attention span of a human:

2000: **12 seconds**

2015: **8 seconds**



345/487

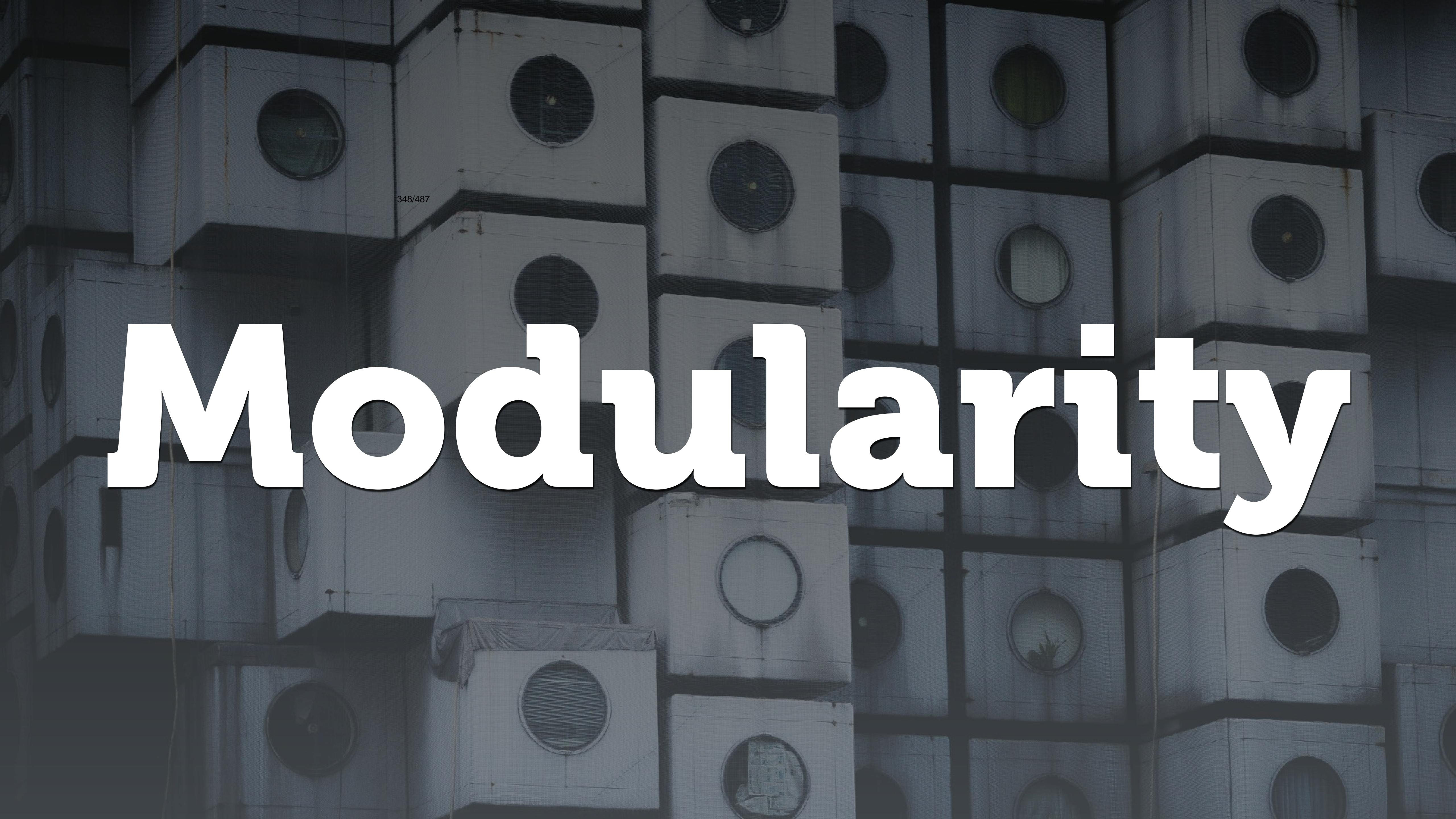
9 seconds

<http://www.statisticbrain.com/attention-span-statistics/>



P A T T E R N F L Y





348/487

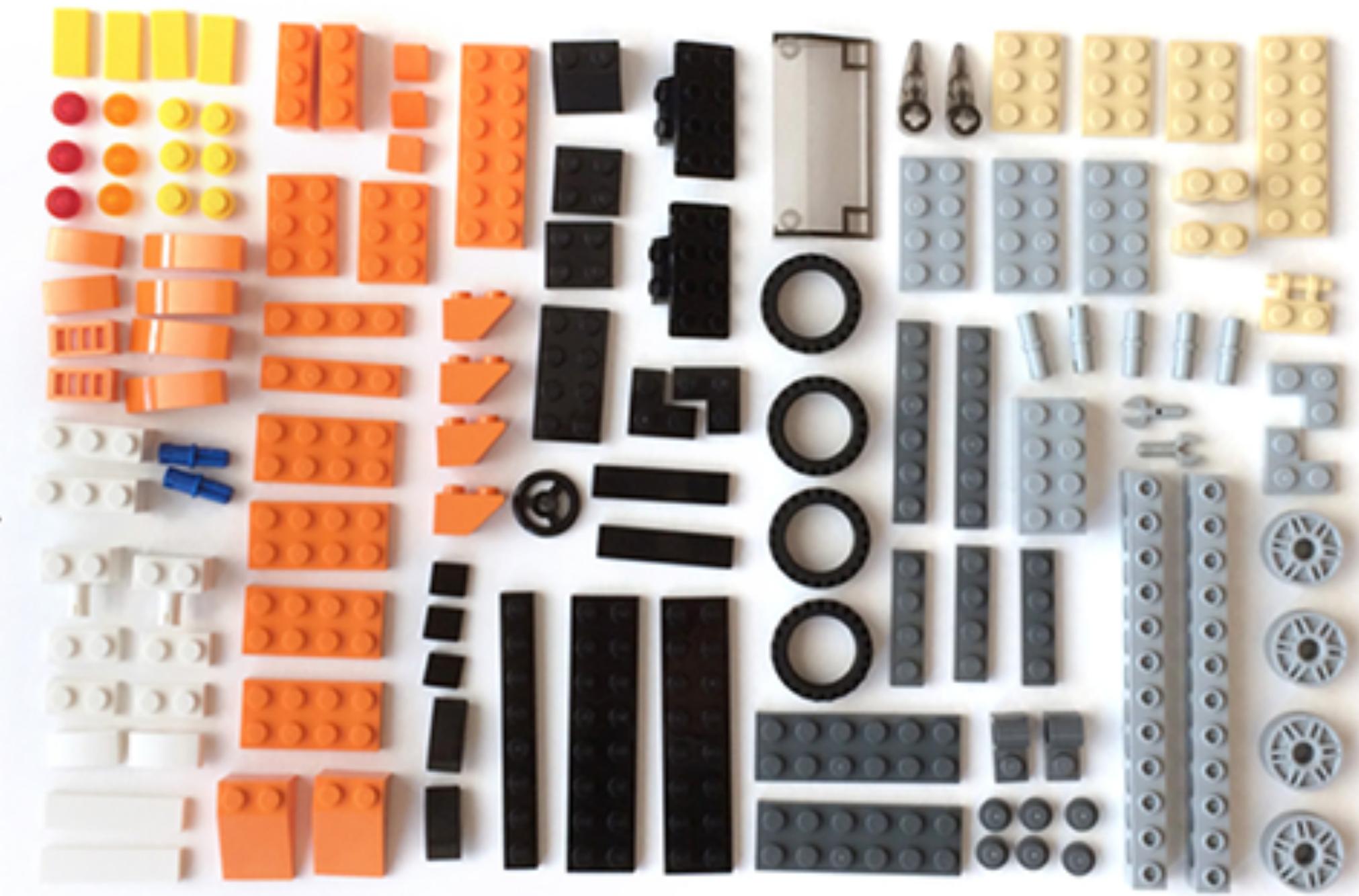
Modularity

349/487



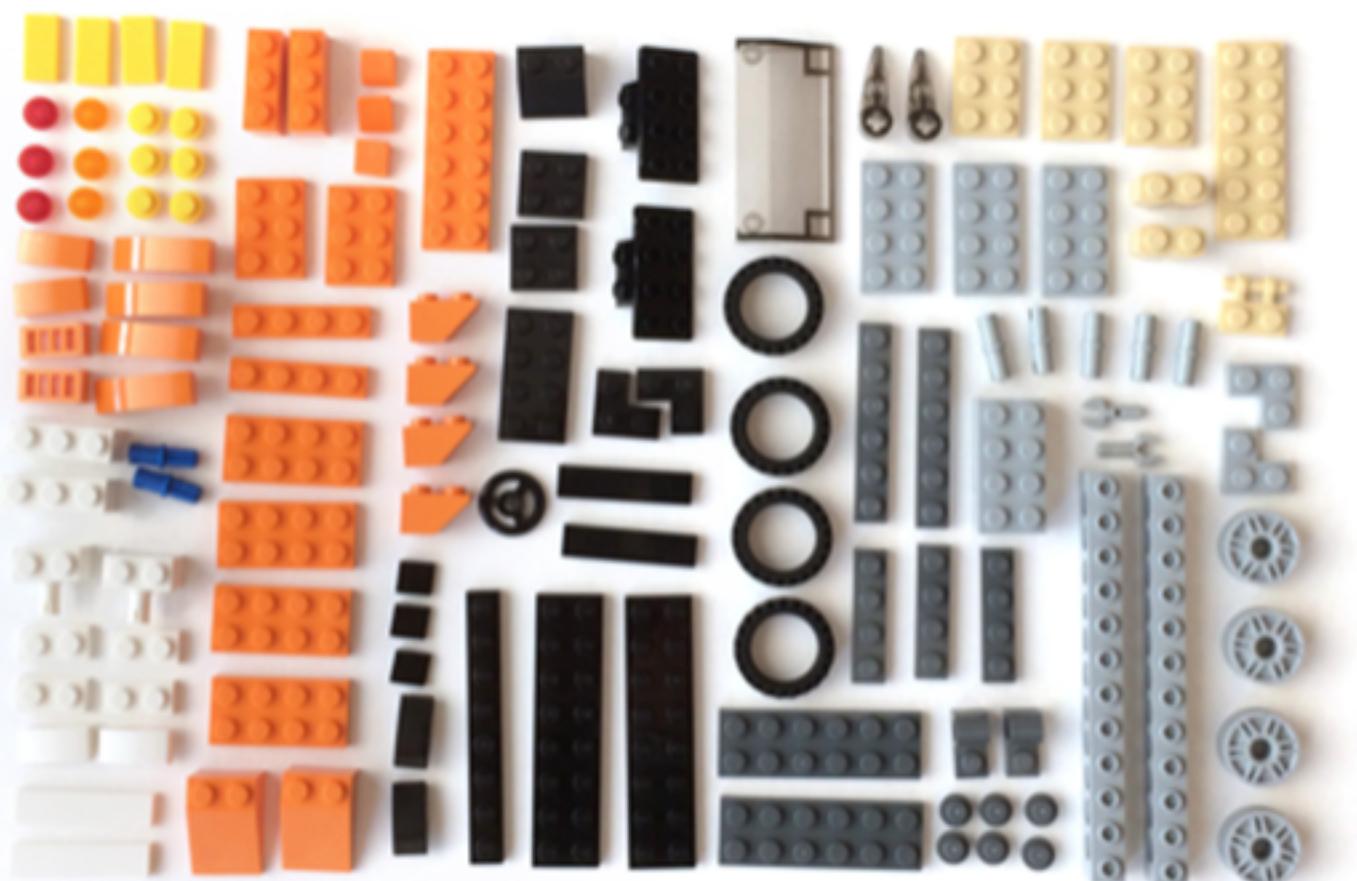
<http://atomicdesign.bradfrost.com/chapter-4/>

350/487



<http://atomicdesign.bradfrost.com/chapter-4/>

351/487



<http://atomicdesign.bradfrost.com/chapter-4/>

CSS Villains

Globally operating

Dependent on location

Very leaky

Based on inheritance

A close-up photograph of a Skeletor action figure. He has a pale, skeletal face with a yellow skull-like mask, wearing a black hooded cloak with purple lining. He is riding a purple, scaly, multi-headed dog-like creature. He holds a purple sword with a blue hilt. The background is plain white.

CSS Specificity

353/487

CSS Specificity

354/487

How the browsers decides which
CSS declaration is the most
relevant to apply to an element.

CSS Specificity

355/487

If one selector is more specific than another, it overrides it.

If two selector share the same weight, the later one is applied.

Selector Weight

356/487

Selector	Weight	Example
Type Selector	1	p { ... }
Class Selectors	10	.success { ... }
ID Selectors	100	#header { ... }
Inline Styles	1000	<p style="color: red">

```
p.success {  
    color: green !important;  
}
```

358/487

CSS

Global
HD

Demo

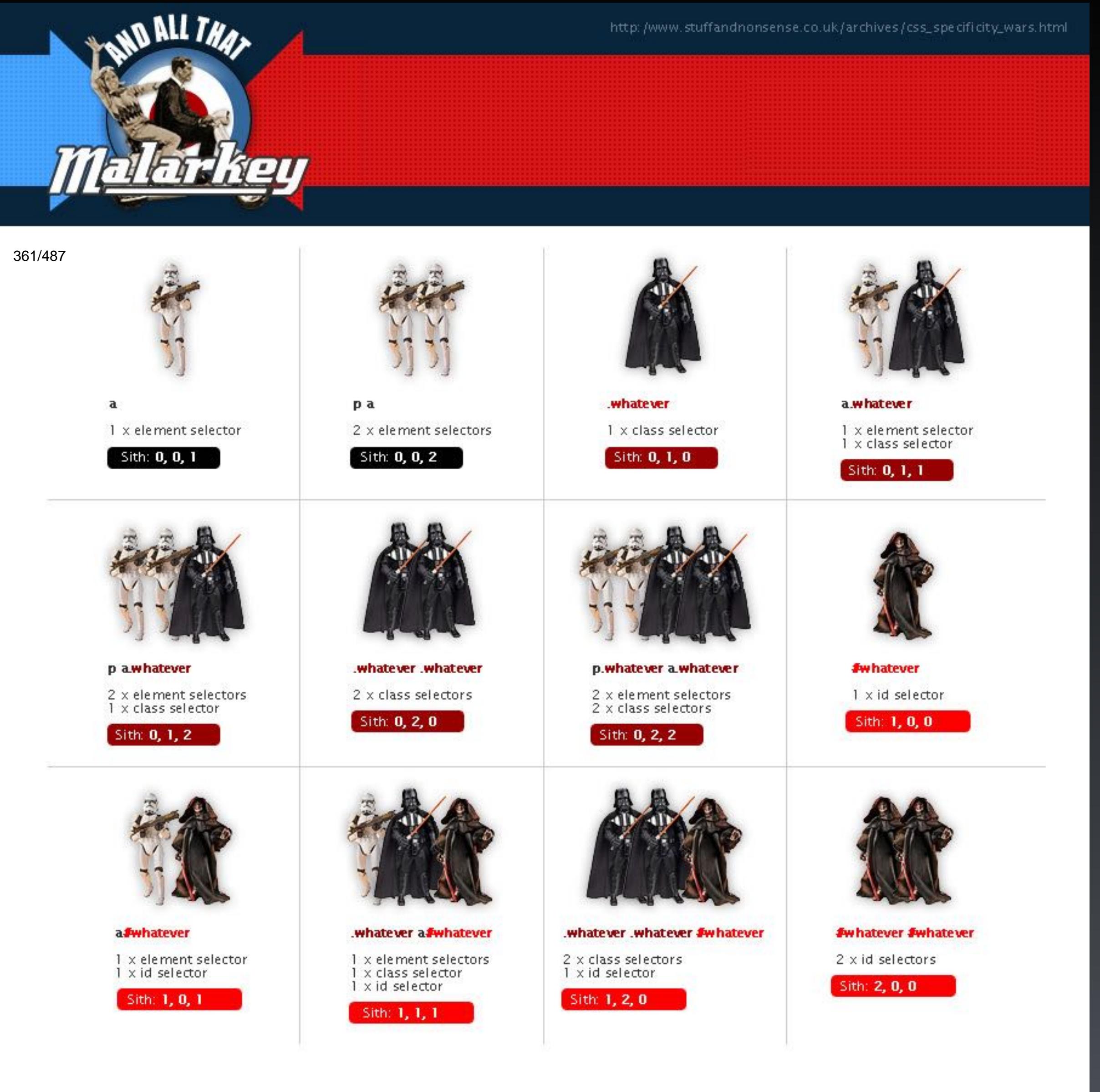
Specificity

Class ✕ Type Selectors

IDs ✕ Class

Inline styles ✕ IDs

!important 💣 all of them.



https://stuffandnonsense.co.uk/archives/css_specificity_wars.html

SHADOWDOM

362/487

GOOD CSS



WebComponents

363/487

Web Components

364/487

ShadowDOM

HTML Imports

Custom Elements

HTML Templates

https://www.w3.org/standards/techs/components#w3c_all

Web Components

365/487

Encapsulate all of their HTML and CSS.

JavaScript behaviours are native on the browser.



366/487

SHADOW DOM

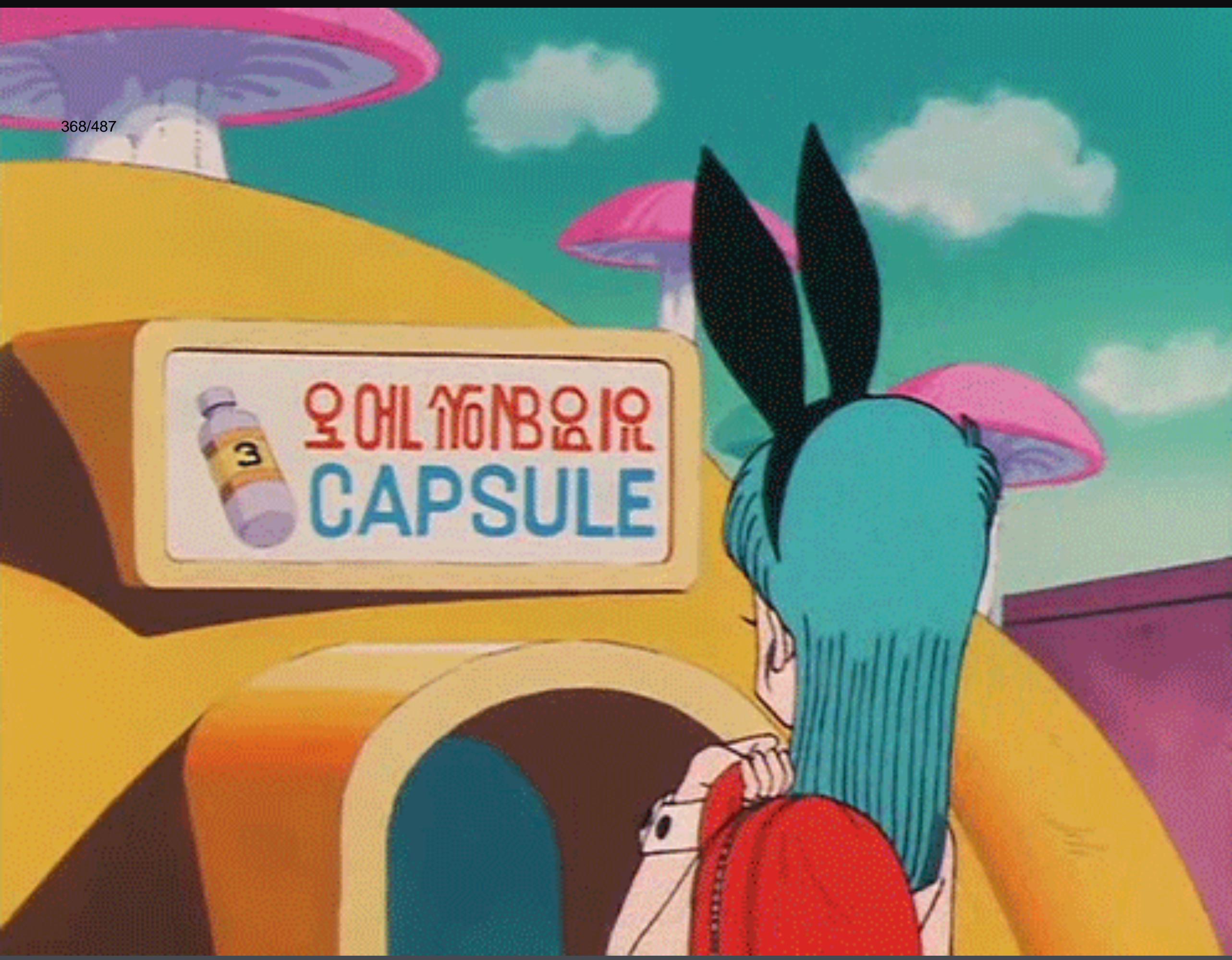
Custom elements

+

Shadow DOM

=

Self contained HTML, CSS and JS



368/487



369/487

CAPSULE

RP.

370/487

css

The ShadowRoot

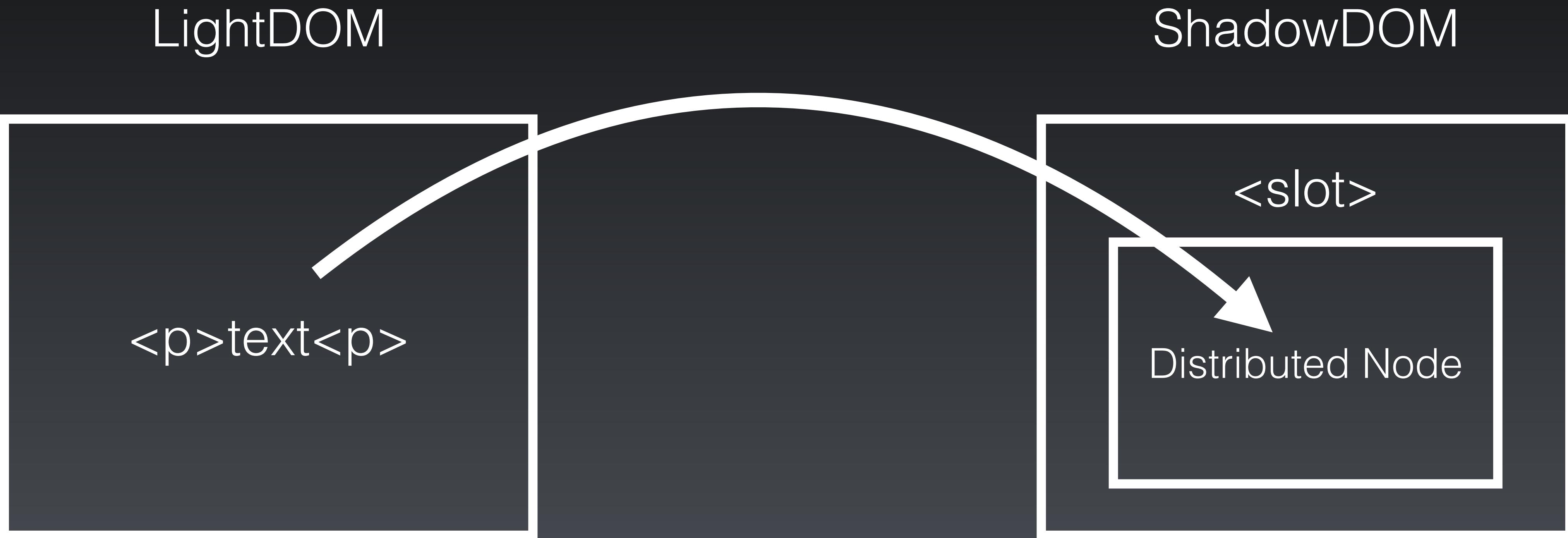
A shadow root is a document fragment that gets attached to a “host” element.

Demo

ShadowRoot

Light DOM + Shadow DOM = Composed DOM

373/487



<slot>

Placeholders inside your component that users can fill with their own markup.

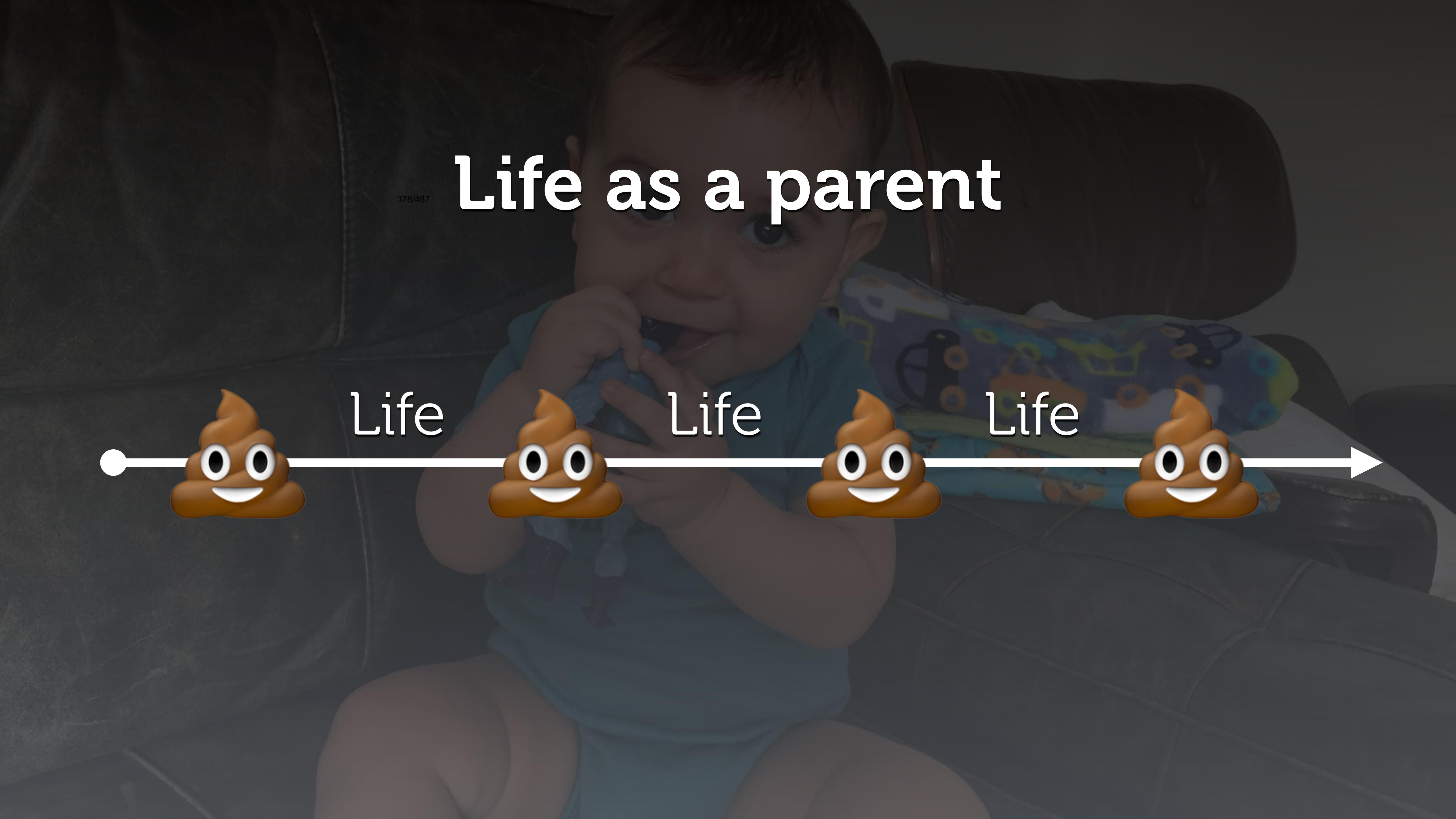
Demo

Slots

Stylin' the Shadow DOM

377/487



A close-up photograph of a baby with light brown hair and blue eyes, wearing a teal onesie. The baby is sitting on a dark-colored couch, looking slightly to the right. A pacifier is in their mouth. In the background, there's a colorful blanket with various patterns.

Life as a parent

378/487



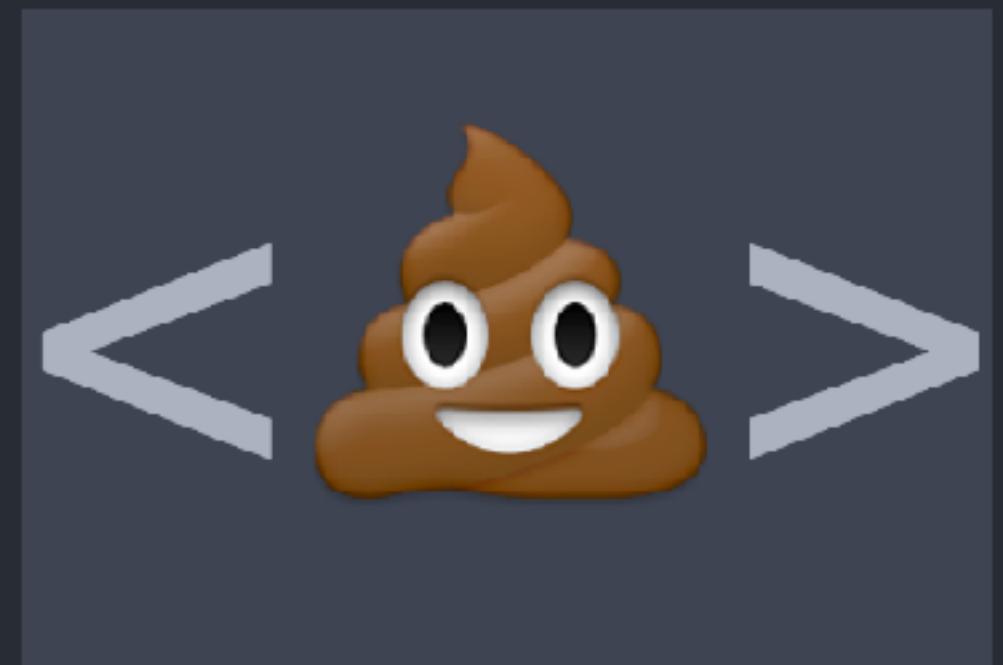
9

..<body>~

379/487

10~

11~



12~

13~

CSS selectors from the outer page
don't apply inside your component.

Styles defined inside don't bleed out.
They're scoped to the host element.

:host(<selector>) selector

Allows you to target the host if it matches a <selector>.
381/487

::host-context(<selector>) selector

Matches the component if it or any of its ancestors
matches <selector>.

:slotted(<compound-selector>) selector

Matches nodes that are distributed into a <slot>

Demo

host and slotted selectors



© El Juguetonista

384/487



<https://www.polymer-project.org/>

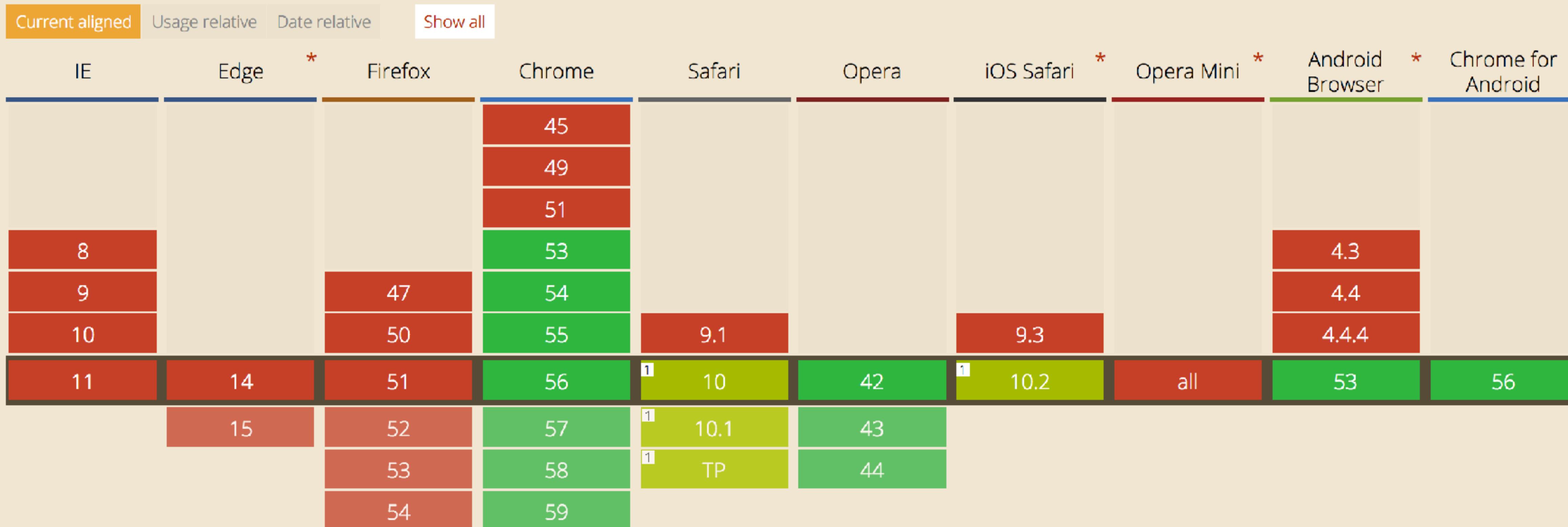


<https://www.webcomponents.org/>

Shadow DOM v1 - WD

Method of establishing and maintaining functional boundaries between DOM trees and how these trees interact with each other within a document, thus enabling better functional encapsulation within the DOM & CSS.

385/487

[Notes](#)[Known issues \(0\)](#)[Resources \(8\)](#)[Feedback](#)
<http://caniuse.com/#feat=shadowdomv1>

Shadow DOM v0 was implemented in Chrome/Opera but other browser vendors are implementing v1.

MS Edge status: Under Consideration



Shadow DOM

UNDER CONSIDERATION

★ 9737 Votes



Enables DOM tree encapsulation. Without it, widgets may inadvertently break pages by using conflicting CSS selectors, class or id names, or JavaScript variables (often used in Web Components).

386/487

Roadmap Priority: High — We intend to begin development soon.

☆ 9737 Votes

[Detailed browser support](#)

[Working draft or equivalent](#)



Under Consideration in Microsoft Edge



Shipped in Chrome



In Development in Firefox



Not Supported in Internet Explorer 11



In Development in Opera



In Development in Safari

<https://developer.microsoft.com/en-us/microsoft-edge/platform/status/shadowdom/>

Shadow DOM LANDED LANDED LANDED CONSIDERING DEVELOPING

Docs Specification caniuse 59% 0/3 Resolved

Provides encapsulation for JavaScript, CSS, and templating in a [Web Component](#).

<https://platform-status.mozilla.org/#shadow-dom>

Browser support



CHROME



OPERA



FIREFOX



SAFARI



EDGE

387/487



TEMPLATES



STABLE



STABLE



STABLE



STABLE



STABLE



IMPORTS



STABLE



STABLE



POLYFILL



ON HOLD



POLYFILL



ON HOLD



POLYFILL



CONSIDERING



CUSTOM ELEMENTS



STABLE



STABLE



POLYFILL



DEVELOPING



POLYFILL



DEVELOPING



POLYFILL



CONSIDERING



SHADOW DOM



STABLE



STABLE



POLYFILL



DEVELOPING



STABLE



POLYFILL



CONSIDERING

Shadow DOM

388/487

- 👍 Excellent name
- 👍 Scoped Styles 🎉
- 👍 Modular components
- 👎 Bad browser support

A wide-angle photograph of a coastal road at sunset. The road curves from the bottom left towards the center of the frame, bordered by a white guardrail on the right. A lone figure in a blue shirt and dark shorts runs away from the camera on the road. To the left is a steep, green-covered hillside. The sky is filled with wispy, orange and yellow clouds against a deep blue background. In the distance, a small town or cluster of houses is visible at the base of the hills.

CSS GUIDELINES

389/487

Let's write really good CSS!

4 Rules to keep your CSS tidy

390/487

- 1 Use a global namespace
- 2 Use BEM
- 3 Keep selectors flat
- 4 Write great CSS documentation



1

Use a namespace
to avoid conflicts.

```
.pf-masthead {  
    color: white;  
}
```

2

BEM syntax

`{{block}}_{{element}}--{{modifier}}`

Block

394/487

.pf-masthead

Element

.pf-masthead_section

Modifier

.pf-masthead--inverse



3

**Use flat Selector
to avoid specificity issues.**

- 👍 Whenever possible don't nest selectors
- 👍 Use classes, never IDs
- 👍 Never use `!important`

A large white circle with a thin black outline, centered on the slide. Inside the circle is the number '4' in a bold, white, sans-serif font.

4

Write good CSS docs
for a correct CSS implementation.

https://github.com/patternfly/patternfly-css/blob/master/.github/DOCUMENTATION_TEMPLATE.md

Class name: .classname

398/487

Applied to: <element>

Outcome: what does it do?

Required: Yes or no

Remarks: General remarks

Accessibility: role, aria and a11y comments

```
<header class="pf-masthead">
```

Class name: .pf-masthead

400/487

Applied to: <header> or <div>

Outcome: Initiates a masthead

Required: Yes

Remarks: Always use it with the grid

Accessibility: role="navigation", skip link, etc

Lint CSS!

[Docs](#)[FAQ](#)[Pricing](#) [@houndci](#) [Sign In with GitHub](#)

402/487

Code review tool for GitHub pull requests

Hound comments on style violations in GitHub pull requests, allowing you and your team to better review and maintain a clean codebase.

[SIGN IN WITH GITHUB](#)<https://houndci.com>

OOCSS & BEM

403/487



Vanilla CSS



Transparent structure



Modular components



It can go wrong really fast



404/487

USE CASE

405/487

PatternFly

Putting everything together



P A T T E R N F L Y

Custom Elements:  yes!

Shadow DOM:  not now.



Home Frameworks ▾

Components Github

408/487



patternfly-webcomponents

Components for the web built with Patternfly ❤️

[View on Github](#)

[View Components](#)

Installation

NPM

Install patternfly-webcomponents via NPM

<https://patternfly-webcomponents.github.io/>

`npm install --save patternfly-webcomponents`

CSS PatternFly Guidelines

<https://github.com/patternfly/patternfly-css/blob/master/CODE-GUIDELINES.md>

B



Bootstrap 4 is awesome



happy me

The screenshot displays the Patternfly dashboard interface, which includes a top navigation bar with sections like Section One through Section Six, and a header bar showing system statistics: 361 Users, 29,210 Hosts, and 465 Errata. The main content area is divided into several panels:

- Datacenter Northeast**: A section showing a line graph for Datacenter Load (Memory Utilization) from January to May, with a notable peak in April. It also displays critical server events (625, up 11%) and a link to View All Events.
- Patternfly Sidebar**: A dark sidebar containing the Patternfly logo and a list of navigation items: Dashboard, Dolor, Ipsum, Adipiscing, and Lorem.
- Dashboard Overview**: A grid of four cards: 0 Ipsum (0 users, 0 hosts, 0 errata), 20 Amet (4 users, 1 host, 4 errata), 9 Adipiscing (9 users, 9 hosts, 9 errata), and 12 Lorem (12 users, 1 host, 1 errata).
- Top Utilized Clusters**: A list of clusters with their usage status: RHOS6-Controller (190.0 of 200.0 GB Used, red bar), CFMEQE-Cluster (100.0 of 200.0 GB Used, green bar), RHOS-Undercloud (140.0 of 200.0 GB Used, orange bar), and RHEL6-Controller (153.0 of 200.0 GB Used, orange bar).
- Quotas**: A section showing resource utilization: CPU (115 of 460 MHz), Memory (8 of 16 GB), Pods (5 of 8 Total), and Services (2 of 2 Total).
- Utilization**: Real-time metrics for CPU (50 Available of 1000 MHz), Memory (256 Available of 432 GB), and Network (200 Available of 1300 Gbps). Each metric is accompanied by a circular gauge chart.
- Recently Events**: A list of three recent events: Event One (The following snippet of text), Event Two (The following snippet of text), and Event Three (The following snippet of text).

Demo

PatternFly masthead

...and they all live happily ever after.

Muito Obrigado :)

415/487



P A T T E R N F L Y

github.com/patternfly

slack.patternfly.org

patternfly.org

@andresgalante



Central de Atendimento
51 3214 1414

Central de Locações
51 3214 1400

Central de Vendas
51 3214 1415

Central de Condomínios
51 3214 1432

www.creditoreal.com.br

CNPJ 92.691.336/0001-66

DOC DE LOCAÇÃO

IMÓVEL : 00006038 - REALIDADE
 ENDEREÇO : R LUIZ MANOEL, 31 - Apartamento 404 - PORTO ALEGRE/RS
 LOCATÁRIO : 0012590 - ELEONOR VINICIUS DUDEL MAYER
 COMPETÊNCIA : 10/2017 (NORMAL)
 PROPRIET. : 0008367 - MILNITSKY NEGOCIOS IMOBILIARIOS LTDA CNPJ: 05.935.669/0001-99

VENCIMENTO

05/11/2017

DESCRÍCÃO DE TAXAS

00001 ALUGUEL	1.100,00
00218 CONDOMINIO REPASSE	151,20
00225 IPTU - REPASSE	76,57
00242 TAXA PORTE	1,80
04022 DMAE-REPASSE	(09/2017 - APARTAMENTO 404) 29,25
00122 TAXA BANCARIA	4,00
-----	-----
TOTAL	1.362,82
VLR.MULTA	135,70
TOTAL C/MULTA APÓS VENCIMENTO	1.498,52

Próx. Reaj.: 18/08/2018

INFORMATIVO

PAGAMENTO SEM TAXA DOC NO BANCO ITAU - RUA SETE DE SETEMBRO Nº 746. O pagamento deste DOC não quita débitos anteriores. Após 5 dias vencido, somente no escritório de advocacia. Informe-se sobre Seguro conteúdo/Assistência 24 h. Ligue 3214.1414 e solicite o setor de condomínios (condôminos) ou administração de bens (locatários). NOVIDADE: Agora ficou mais fácil consultar a 2ª via do seu doc em nosso site. Acesse agora mesmo www.creditoreal.com.br, informe apenas o seu CPF ou CNPJ no campo indicado e você conseguirá emitir a 2ª via do seu boleto até a data do vencimento.

Após vencimento pagável conforme instruções

Autenticação Mecânica

Itaú Banco Itaú S.A.

341-7 | 34191.09008 13448.560899 70866.820007 7 73340000136282

Local de Pagamento REDE BANCARIA					Vencimento 05/11/2017
Beneficiário CREDITO REAL IMOVEIS E CONDOMINIOS S/A CNPJ 92.691.336/0001-66					Agência / Código do Beneficiário 0897/08668-2
Data Documento 18/10/2017	Nº Documento 00006038.01.10/2017.N	Espécie Doc. N	Aceite X	Data de Processamento 18/10/2017	Nossa Número 109/00134485-6
Uso do Banco	Carteira 109	Moeda REAL	Quantidade	Valor X	(=) Valor do Documento 1.362,82
Instruções					(-) Desconto
*PAGAVEL REDE BANC. ATÉ 05 DIAS APÓS O VENCIMENTO, ACRESCIDO DE MULTA. *APÓS 05 DIAS VENCIDO, PAGAMENTO SOMENTE EM ESCRITÓRIO DE ADVOCACIA. *O PAGAMENTO DESTE DOC NAO QUITA DEBITOS ANTERIORES. *E EXPRESSAMENTE PROIBIDA A DEDUCAO DE VALORES. *PAGAMENTO SEM A TAXA DOC, NO BANCO ITAU - DA RUA SETE DE SETEMBRO , Nº 746. **APÓS O VENCTO, MULTA DE R\$135,70. JUROS:1%AM + CORREÇÃO MONET.					(-) Outras Deduções
					(+) Mora / Multa
					(+) Outros Acréscimos
					(=) Valor Cobrado

Pagador ELEONOR VINICIUS DUDEL MAYER

R LUIZ MANOEL, 31 - 404

90040-390 SANTANA - PORTO ALEGRE-RS

Imov.: 6038

Pagador Avalista

Código de Baixa

Autenticação Mecânica Ficha de Compensação



ARPG – THE AGILE RELEASE PLANNING GAME

Facilitator's guide – version 1.1 April 2010

A game where participants run an agile project.

Contents

The game.....	3
The story so far	3
What needs to be done.....	3
Equipment at hand	3
Getting started	4
Break the group into teams.....	4
Explain the story so far.....	4
Set the board up	4
Place the deadline cards on the board.....	4
Explain velocity and story points	4
Prepare an initial release plan.....	4
Running the first iteration.....	5
Confirm the plan for the iteration	5
Determine the actual velocity	5
Deal with existing defects	5
Allocate the remaining velocity.....	5
Identify new defects	5
Make scientific discoveries.....	5
Update the status reporting sheet	5
Running remaining iterations.....	6
Ending the game.....	6
Debrief	6
Variations and optional rules	6
Everyone has the same luck	6
A more focussed outcome	6
Improved status reporting	6
Iteration story wall.....	6
Hang over stories.....	7
Dependencies	7

The game

The game is designed to help participants understand the concepts involved in release and iteration planning.

Participants must plan their work across a series of iterations and then adapt their plans as the project evolves.

In order to be successful, participants must meet critical deadlines, minimise defects, monitor their velocity and replan as they gain further insight into the evolving project.

Variants of the game allow participants to explore further concepts through:

- Managing within iterations by using story walls; and
- Using velocity charts and other tools to track progress.

The story so far

The team are the advanced party for the first mission to Mars.

The team got to Mars without any issues and a robot team landed to establish a base. Unfortunately the team's own landing did not go so well.

For some reason, the automatic pilot fired the retro-rockets far too late. Instead of making a gentle descent, the team plunged into the Mars base, virtually destroying it.

All communication has been lost and the team need to set priorities without any guidance.

Worse still, the team will run out of crucial resources in only a short time.

What needs to be done

Just before shutting down, the mission computer prepared an analysis of the team's chances of survival.

If the team repair the landing module then they can access a vast cache of supplies and even go home. Survival is thus assured.

If not then they will need to be rescued.

Fortunately another exploration vessel is already on route. Unfortunately this will take time, so the team will need to survive for 8 complete iterations before being rescued.

It would also be good if the team could make use of their time on Mars to complete the original objective of their mission – scientific research.

If the team live long enough, they can rebuild the original science labs and fulfil their original mission.

Equipment at hand

The mission computer produced a series of story cards for the team to use. Each card contains the name of a feature the team can build (or repair) and the number of "story points" required to do so.

The amount of story points the team can complete each iteration (or turn) is known as the team's velocity.

The computer estimates that the team's velocity is approximately 15 story points per iteration. This number is expected to change over time as mission evolves.

The computer has also produced some other tools for the team to use:

1. A mission glossary, explaining the features that can be built, as well as the meaning of other terms;
2. A status report for the team to use in tracking their progress; and
3. An iteration planning board for the team to use in planning iterations.

Getting started

There are a number of variations of the game available and these are explained a little later. But most teams start with the basic game.

The game can be introduced by first presenting an explanation of the concepts of release planning or it can be run prior to the explanation of the theory.

Break the group into teams

To introduce the game, break the group into teams. Each team will need:

1. Team members. The best team size is between 2 and 8 people.
2. The game board, defect counters, deadline cards and story cards.
3. A pair of dice.
4. A status report sheet and a pen. You are welcome to photocopy the status report for use in the game.
5. A story glossary describing the stories they can work on.

Explain the story so far

Explain the background story to the group.

Set the board up

Have each team lay the board out in front of them.

Place the deadline cards on the board

The purpose of the mission is to make new scientific discoveries. But to survive, the team will need to either repair the Mars Lander or create the right features to last 8 iterations.

But the base is collapsing around the team's ears. In order to merely survive, the team will need to meet a set of critical deadlines.

Deadlines are marked on the board. They can be met by building certain features in time, or by deferring the deadline through implementing smaller stories.

Place the deadline cards on the board where indicated. This way, if the deadline is met the card can be deferred to a later iteration or removed from the board to show it has been permanently addressed.

Explain velocity and story points

The team will survive by "completing stories" to meet deadlines or build features.

Each story is represented by a story card. Each card has a heading, a story number and a story point rating.

- The story number can be used to find the story in the project glossary.
- The story point rating is the cost to complete the story.
- The velocity is the number of story points the team can expend to complete stories each iteration.

Prepare an initial release plan

The final step in the initiation phase of the project is to prepare a release plan.

Give each team 10 minutes to:

- Review the project glossary to understand the benefit of the different features; and
- Come up with a release plan.

Note the team will decide how many iterations they plan for, but they must at least allocate stories to the first iteration.

Players can allocate more stories than they expect to complete, since they will be able to confirm which order stories are done during each iteration.

Running the first iteration

Confirm the plan for the iteration

The team confirm the stories to be attempted before the start of the iteration.

In future iterations, the team need to decide whether to reduce outstanding defects. Fixing a defect requires 1 story point per defect.

However, in the first iteration there will be no defects to fix. So the team should place the “zero” defect counter on the board.

Determine the actual velocity

We predict the velocity for the first iteration will be 15. This is a good indicator of how much work the team will get done, but it is not a perfect prediction.

So we roll dice to represent the impact of unpredicted occurrences, as well as the ongoing improvements made by the team.

Have each team roll two dice and consult the velocity change table on the game board.

Add (or subtract) the amount shown in the table to calculate the actual velocity for this iteration. This number will also be the expected velocity for the following iteration.

Deal with existing defects

Where defects exist, the team need to make running repairs, use workarounds and perform other work that will distract the team from both creating new features and even repairing the defects themselves. We refer to this as technical debt.

Consult the “Drain due to technical debt” table on the game board. The number shown represents the amount of effort the team must spend coping with existing defects in production.

This number is subtracted from the velocity each iteration before the velocity can be spent fixing defects or building stories.

Allocate the remaining velocity

If the team had committed to fixing defects during the iteration then these would need to be completed before work begins on stories..

The team can then choose which stories to complete with the remaining points.

Incomplete stories must be deferred until a later iteration and any points that cannot be allocated to complete a story will be wasted.

Identify new defects

The team is likely to find defects in the work once it has been completed.

To represent this, the team roll two dice and consult the new defects table on the game board. This represents the number of defects that will carry forward into the next iteration.

Make scientific discoveries

In addition to merely surviving, the team has the opportunity to become one of the greatest scientific expeditions of their age.

If the team has built any science labs, then for each one, the team roll two dice and consult the “New discoveries” table.

Note that the team can add 1 to each roll for each Mars Rover they have built.

In addition, the first time a 10 or more is rolled for the life research project, the team discover life. And they gain 10 kudos points instead of 2.

Update the status reporting sheet

The team update the status report to reflect what has occurred during the iteration.

Running remaining iterations

Assuming the team survive they now move onto the next iteration.

The process is essentially the same as running the first iteration, except that defects may now exist.

Ending the game

For a single team, the game ends when either:

- The team do not survive the iteration;
- The team are rescued at the end of the 8th iteration; or
- The team have completed the repairs to the Mars Lander.

If more than one team is playing then other teams can continue, or the game can end for the whole group if preferred.

Debrief

Once the teams have completed the game, perform a debrief as a group.

One way to do this is to hold a retrospective:

- Have each team answer 3 questions:
 - What worked well?
 - What caused problems?
 - What can we learn from this?
- If there are multiple teams, then have the teams discuss their findings as a group.

Variations and optional rules

Everyone has the same luck

Allowing each team to roll the dice for themselves allows the facilitator to move freely between teams during the game. But it also means that different teams may have different outcomes based on luck rather than planning.

An alternative approach is for the facilitator to roll the dice each iteration and announce the results.

A more focussed outcome

The basic games requires people to set their own priorities.

Instead it is possible to have a more defined outcome for the teams to race towards.

In this case the game ends when one team achieves the objective.

Two typical objectives are:

- **We will not be rescued.** The teams will therefore need to build the Mars Lander. The first team to do so wins.
- **It's all about the science.** This is the same as the basic game, except that the winning team is the one with the highest scientific kudos at the end of 8 iterations.

Improved status reporting

The status sheet allows the teams to keep track of what is going on. But better tools are available to Agile development teams.

The game works well with participants completing the same reporting tools that they would utilise in a real project, for example:

- Velocity chart (or burn up chart);
- Defect tracking chart; or
- Iteration contents chart.

Using these tools will require longer turns and in return it will help to explain and imbed the use of these tools.

Iteration story wall

There is an extension game that tracks the same stories as they are built within the iteration.

This helps to explain the use of the story wall in a project.

However using this extension throughout the whole game will lengthen the playing time.

An alternative is to use the story wall only after iteration 4, or to use it as a standalone tool to explain story walls as a separate exercise.

Hang over stories

If the team is happy to play a more complex version of the game then they can introduce “hang over stories”.

These are stories that are almost completed in an iteration but “hang over” into the next iteration before being completed.

However the team will experience a level of inefficiency when completing stories over multiple iterations. So they must “waste” a story point.

Furthermore, if the story is not completed in the next iteration then the points allocated to the story are lost and the story is no longer a hangover one.

For example:

- The team then has 2 points left to allocate to stories at the end of the iteration and decides to allocate them to a 5 point story that will be hung over to the next iteration.
- The team can now complete the story in the next iteration, but they have wasted one point, so they still need to use a further 4 points to complete the story.

Dependencies

Dependencies add great complexity to planning projects in the real world.

To duplicate this in the game, make the following stories dependent:

Story	Is dependent on
19c Mars Lander fittings	19g Mars Lander hull structure
19d Mars Lander communications	19e Mars Lander CPU
19f Mars Lander controls	19e Mars Lander CPU
19i Mars Lander navigation unit	19e Mars Lander CPU
19k Mars Lander atmosphere unit	19j Mars Lander power plant
20 Lander final tests	All other Lander stories (stories 19a through 19k)
22 Robot development engineer	26 Robot factory
23 Robot test engineer	26 Robot factory
24 Robot TV crew	26 Robot factory
25 Robot chef	26 Robot factory
34 Water recycling	33 Water production plant

Where a story is dependent on another one, it can be completed in the same iteration as the other story, but cannot be completed in a prior one.

ARPG – The Agile Release Planning Game.

Copyright © James King 2009. To learn how to plan and run projects in the real world, go to www.softed.com

Iteration 1 Repair construction centre	Iteration 2 Need power source	Iteration 3 Need air	Iteration 4 Need water	Iteration 5	Iteration 6 Need food supply	Iteration 7 Need Laundry	Iteration 8 Need bathing facility Need hospital
Defects to fix	Defects to fix	Defects to fix	Defects to fix	Defects to fix	Defects to fix	Defects to fix	Defects to fix
Story 1	Story 1	Story 1	Story 1	Story 1	Story 1	Story 1	Story 1
Story 2	Story 2	Story 2	Story 2	Story 2	Story 2	Story 2	Story 2
Story 3	Story 3	Story 3	Story 3	Story 3	Story 3	Story 3	Story 3
Story 4	Story 4	Story 4	Story 4	Story 4	Story 4	Story 4	Story 4
Story 5	Story 5	Story 5	Story 5	Story 5	Story 5	Story 5	Story 5
Story 6	Story 6	Story 6	Story 6	Story 6	Story 6	Story 6	Story 6

Dice roll	Velocity change	New defects	New discoveries
2	Subtract 3 velocity points	No new defects	No discovery
3	Subtract 2 velocity points	1 new defect	No discovery
4	Subtract 1 velocity point	2 new defects	No discovery
5-7	No change velocity	3 new defects	No discovery
8-9	Add 2 velocity points	4 new defects	No discovery
10-11	Add 4 velocity points	5 new defects	Small find - Add 1 kudos.
12	Add 6 velocity points	6 new defects	Big find - Add 2 Kudos

Rate	Drain due to technical debt	Impact of scientific kudos
0-1	No impact	No kudos - what a wasted opportunity.
2-3	2 velocity points	You should find employment in universities or on another mission
4-5	3 velocity points	People quote your names in their research. Well done
6-8	5 velocity points	You have offers to join the faculties of the several prestigious universities
9-13	8 velocity points	You can publish a book and live on the lecture circuit
14-21	13 velocity points	The discoveries you made have become new categories of scientific research in their own right
22+	21 velocity points	Schools, universities and cities will be named after you

ARPG – The Agile Release Planning Game.

Copyright © James King 2009. For additional copies of this game please email James King at james@kingsinsight.com

1. Confirm the plan for the iteration

Decide whether to reduce outstanding defects.

Fixing a defect requires 1 story point per defect. To identify the number of defects to be fixed, place a defect counter with the appropriate number in the “defects to fix” box for the iteration.

Next, confirm the stories to be attempted before the start of the iteration. You are not allowed to work on stories that were not placed on the game board within the iteration, but you are allowed to defer stories that were chosen if you do not have the points to complete them.

2. Determine the actual velocity

The expected velocity for the iteration is the same as the velocity actually achieved last time.

To find the actual velocity, roll two dice to and consult the velocity change table on the game board.

Add (or subtract) the amount shown in the table to calculate the actual velocity for this iteration. This number will also be the expected velocity for the next iteration.

3. Deal with existing defects

If defects exist, then you will need to cope with them in production. This involves patching holes, using workarounds, rebooting systems and so forth.

This will distract the team from both creating new features or even repairing the defects themselves. This is referred to as “technical debt” – the complexity caused by the work done in previous iterations.

Determine how many outstanding defects you have from prior iterations and consult the “Drain due to technical debt” table on the

game board. The number shown represents the amount of effort you must spend coping with existing defects in production.

This number is subtracted from your velocity before it can be spent fixing defects or building stories.

4. Allocate the remaining velocity

If you committed to fixing defects during the iteration then these need to be completed before work begins on stories.

You can then choose which stories to complete with the remaining points. Incomplete stories must be deferred until a later iteration and any points that cannot be allocated to complete a story will be wasted.

5. Identify new defects

To determine if any defects slipped through your testing this iteration, roll two dice and consult the new defects table on the game board.

This represents the number of defects that will carry forward into the next iteration, in addition to any outstanding defects that you have not repaired this iteration.

6. Make scientific discoveries

Being on Mars gives you the opportunity to be one of the greatest science expeditions of your age ... or to be a mere footnote in history.

If you have built any science labs, then for each one, roll two dice and consult the “New discoveries” table.

In addition, the first time a 10 or more is rolled for the life research project, the team discover life. And they gain 10 kudos points instead of 2.

7. Update the status reporting sheet

The team update the status report to reflect what has occurred during the iteration.

ARPG –AGILE RELEASE PLANNING GAME

Iteration planning extension – version 1.1 April 2010

A training game where participants experience the planning and running of iterations in an agile project.

The game

This extension is designed to work with the Agile Release Planning Game.

The wider game is designed to help participants understand the end to end process of release planning. This extension focuses on using a story wall to track work within the iteration.

The aim

The aim of this game is for colonists to build new features for use on the base.

To do so involves 3 steps

- Identify, find and assess the materials needed to construct the feature. These materials must be scavenged from the wreckage of the old base.
- Use these materials to build working features; and
- Test and install the features. This involves making sure they work as designed, do not leak and so forth.

The story wall

The game includes a story wall. The story wall is the tracking mechanism that teams in the real world use to track the progress of stories (or tasks or features) during the iteration.

This particular story wall has the following columns:

- **Salvage and analysis.** The stories where the team are salvaging useful materials.
- **Building.** The stories representing features currently under construction.
- **Testing and installation.** Stories representing features being tested and installed in the base.
- **Open for business.** Features that have already been deployed during this iteration.

Allocating velocity

The team's velocity represents the amount of work the team can get through in an iteration. For our purposes we will assume that each point of velocity is equivalent to one "resource" that the team can allocate to work.

The team can allocate their resources as they see fit between:

- Salvage and analysis;
- Building; and
- Testing and installation.

The resources can be allocated to different tasks each iteration. But must remain allocated to the same for the duration of the existing iteration.

So if two points are allocated to testing and there are no stories to test at some point in the iteration, then the team members working on testing sit idle for the day.

Determining the stories and velocity

The stories and the velocity for the iteration can be taken from any iteration in the main game. Alternatively, a base set is provided in the appendix to this extension.

Running each iteration

Set up and plan the iteration

Each iteration consists of 5 days, during which the team must move as many stories and defects as possible from the Salvage column to the Open for Business Column.

Place the stories in the salvage column

To begin, all the stories allocated to the iteration are placed in the salvage column.

Place the defects in Salvage and Build

Some defects require new materials to repair, while others only need "tinkering" to be fixed.

Place half of the defects to be worked on in the Salvage column and half in the Build column. If there is an odd number then place the remaining defect in the Salvage column.

Deploy the resources

The team are free to allocate the available resources to any column they choose.

For example, if the team has a velocity of 15, then they can allocate up to five resources to each of Salvage, Build and Test. There is no need to allocate any resources to “open for business” as there is no work to be done in this column.

Day one the salvage team

On the first day, the salvage team roll a die per point allocated to the salvage task. The number rolled represents the number of story points that the team can move into the Build column.

- Only complete stories (or defects) can be moved to the building column, no partial stories can be migrated.
- If there are no stories in the column then the team sit idle.
- However if there are both points and stories left at the end of the day then the team can carry the remaining points over to the following day.

For example if the team allocate two resources to salvage, they might roll a 4 and a 3 on their dice. So they can move 7 points worth of stories to the Build column. If the only stories they have are 5 point stories then they move one story to the building column and carry two points forward to the next day.

Day 1 – the other teams

The build and test teams repeat the process followed by the salvage team.

Note however that they cannot start work on stories that move into their column during the

day. Instead they can only work on stories that are there at the beginning of the day.

Days two through five

The rest of the iteration continues in the same way as the first day.

Appendix A – Sample iterations

If the team do not have iterations to use from the wider game, then they can choose to run these ones.

Iteration 1

Velocity: 15 points. Defects to fix: 4 points

Stories available: Air generator (5 points, story 1); Basic accommodation unit (2 points, story 2); Mars Rover (3 points, story 18); Mars Lander Communications (3 points, story 19d); Generator (8 points, story 10)

Iteration 2

Velocity: 18 points. Defects to fix: 8 points

Stories available: Any remaining from iteration 1; Hospital (5 points, story 11); Basic Karaoke machine unit (1 points, story 14); Luxury accommodation (13 points, story 16); Mars Lander CPU (5 points, story 19e); Training unit (8 points, story 31)

ARPG Iteration planning extension

Iteration Story wall for iteration _____.

Available velocity: _____

Salvage and analysis	Build	Test and installation	Open for business
Resources allocated:	Resources allocated:	Resources allocated:	Resources allocated: Nil

Defect counters for use on the Iteration story wall. These can be used to track individual defects during the iteration:

Defect 1 SP							
Defect 1 SP							
Defect 1 SP							

Agile Release Planning Game – Mission Glossary

Item	Meaning	Pts
Air for 4 iterations	If there is no atmospheric plant, the team can go outside and create air from the ice. This will generate enough air for another 4 iterations	2
Air generator	Creates air for the Mars base. Will function indefinitely	5
Basic accommodation unit	Required for the colonists to survive. Still requires air, power, water and food	2
Basic bathing facility	Allows basic bathing and other sanitary activities	3
Cinema and movie studio	Shows the latest films from earth and also allows local production	13
Clothing factory	Allows the manufacture of new clothing and space suits as well as repairs to existing clothes	3
Communication unit	Allows communication with earth and any other colonies	3
Construction centre	Allows construction of units beyond the first three iterations	5
Food processing unit	Creates the food needed for the colony	5
Generator (fission)	This involves fixing and installing the fission drives from the ship. Should work fine for 5 iterations and will then fail on a dice roll of 1 every iteration afterwards	3
Generator (solar)	Build a new power generator to use solar power	8
Hospital	Minimises the risk of lost work due to injury or illness	5
Hydroponic plant unit	Allows the production of organic vegetables	5
Iphone network	Allows colonists to use their iphones to communicate	2
Karaoke machine	Allows the team to unwind while singing to each other	1
Laundry unit	Allows cleaning of clothing	3
Luxury accommodation	Allows the team to live a more comfortable existence	13
Luxury baths	As per the basic bath, but allows a luxurious experience	8
Mars rover	Allows wider exploration of the planet. Add 1 to each dice roll when attempting to gain scientific kudos, for each rover in operation.	3
Mars Lander	The mars lander is a rocket capable of reaching the orbiting module. However it is currently shattered following a rough landing on the planet's surface. Multiple stories are needed to repair the lander. Once these are completed then final tests can be run on the lander and then it is ready for takeoff	n/a
Mars Lander hydraulics	Needed for the Mars Lander to be able to operate successfully	3
Mars Lander docking station	Needed for the Mars Lander to be able to operate successfully	1
Mars Lander furnishings	Needed for the Mars Lander to be able to operate successfully	1
Mars Lander communications	Needed for the Mars Lander to be able to operate successfully	3
Mars Lander central computer	Needed for the Mars Lander to be able to operate successfully	5

Agile Release Planning Game – Mission Glossary

Mars Lander controls	Needed for the Mars Lander to be able to operate successfully	2
Mars Lander hull structure	Needed for the Mars Lander to be able to operate successfully	13
Mars Lander engines	Needed for the Mars Lander to be able to operate successfully	8
Mars Lander navigation	Needed for the Mars Lander to be able to operate successfully	2
Mars Lander power plant	Needed for the Mars Lander to be able to operate successfully	5
Mars Lander atmosphere	Needed for the Mars Lander to be able to operate successfully	3
Mars Lander - final tests	Testing needs to be completed after the Mars Lander has been built.	1
Robot build engineer	Adds 1 point to velocity for future iterations	5
Robot chef	Frees colonists up from cooking duties	3
Robot factory	Allows the construction of robots	3
Robot test engineer	Reduces each defect leak by one point	3
Robot TV crew	This robot allows the team to create a reality show based on their adventures. This will provide some handy income for the team if they make it back home to spend it	5
Science unit (general research)	Allows the team to conduct scientific research	2
Science unit (Mars life project)	Allows the team to look for life on Mars and potentially study new life forms	5
Sheltered park	Allows the team to move about and enjoy themselves, going outside without space suits on.	5
Sports centre	Allows the team to exercise and enjoy themselves	5
Training unit	An online training system that allows the team to develop new skills. Add one to the velocity for future iterations	8
Water for 4 iterations	If there is no water plant, the team can go outside to melt and purify ice. This will provide enough water for four more iterations	2
Water production plant	Allows the Mars base to create its own water indefinitely	5
Water recycling plant	Allows more efficient use of water	2

Story cards

433/487

Story 1	5 SP	Story 2	2 SP	Story 3	3 SP	Story 4	13 SP
Air generator		Basic accommodation unit		Basic bathing facilities		Cinema and movie studio	
Story 5	3 SP	Story 6	3 SP	Story 7	5 SP	Story 8	5 SP
Clothing factory		Communication unit		Construction centre		Food processing unit	
Story 9	3 SP	Story 10	8 SP	Story 11	5 SP	Story 12	5 SP
Generator (dodgy)		Generator (permanent)		Hospital		Hydroponic plant farm	
Story 13	2 SP	Story 14	1 SP	Story 15	3 SP	Story 16	13 SP
I-phone and blackberry networks		Karaoke machine		Laundry unit		Luxury accommodation	
Story 17	8 SP	Story 18a	3 SP	Story 18b	3 SP	Story 19a	3 SP
Luxury baths		Mars rover		Mars rover		Mars lander hydraulic unit	
Story 19b	1 SP	Story 19c	1 SP	Story 19d	3 SP	Story 19e	5 SP
Mars Lander docking station		Mars Lander furnishings and fittings		Mars Lander communications		Mars Lander CPU	
Story 19f	2 SP	Story 19g	13 SP	Story 19h	8 SP	Story 19i	2 SP
Mars Lander controls		Mars Lander hull structure		Mars Lander engines		Mars Lander navigation unit	
Story 19j	5 SP	Story 19k	3 SP	Story 20	1 SP	Story 21	2 SP
Mars Lander power plant		Mars Lander atmosphere unit		Mars Lander final tests		Air for 4 iterations	
Story 22a	5 SP	Story 22b	5 SP	Story 22c	5 SP	Story 22d	5 SP
Robot build engineer		Robot build engineer		Robot build engineer		Robot build engineer	
Story 23a	3 SP	Story 23b	3 SP	Story 23c	3 SP	Story 23d	3 SP
Robot test engineer		Robot test engineer		Robot test engineer		Robot test engineer	
Story 24	5 SP	Story 25	2 SP	Story 26	3 SP	Story 27	5 SP
Robot TV crew		Robot chef		Robot factory		Science lab (Mars life project)	
Story 28a	2 SP	Story 28b	2 SP	Story 28c	2 SP	Story 28d	2 SP
Science lab (general research)		Science lab (general research)		Science lab (general research)		Science lab (general research)	
Story 29	5 SP	Story 30	5 SP	Story 31	8 SP	Story 32	2 SP
Sheltered park		Sports centre		Training unit		Water for 4 iterations	
Story 33	5 SP	Story 34	2 SP	Story	SP	Story	SP
Water production plant		Water recycling plant					

Deadlines

434/487

Need power source (Initially iteration 2)	Need air (Initially iteration 3)	Need water (Initially iteration 4)
Need construction centre (Iteration 1)	Need Laundry (Initially iteration 7)	Need food supply (Initially iteration 6)
Need bathing facility (Initially iteration 8)	Need hospital (Initially iteration 8)	

Defects

1 Defects	2 Defects	3 Defects	4 Defects	5 Defects	6 Defects	7 Defects	8 Defects
9 Defects	10 Defects	11 Defects	12 Defects	13 Defects	14 Defects	15 Defects	16 Defects
17 Defects	18 Defects	19 Defects	20 Defects	0 Defects			

ARPG team status report

Team name:

Iteration	1	2	3	4	5	6	7	8
Survival status	Alive							
Expected velocity (velocity last time)	15							
Add (less) velocity change								
Actual velocity								
Less defect drain								
Velocity available for use								
Velocity used to fix defects								
Velocity used to build new features								
Velocity wasted								
Defects at start	0							
Less defects repaired								
Add new defects								
Defects for next iteration								
Life science labs at start	0							
New life science labs								
Life science labs at end								
Other science labs at start	0							
New labs								
Other labs at end								
Mars Rovers at start	0							
New Mars Rovers								
Mars Rovers at end								
Scientific kudos at start	0							
Add new discoveries								
Scientific kudos at end								



Inteligência em Tecnologia da Informação

Gestão Ágil de Projetos com SCRUM

Eleonor Vinicius Dudel Mayer

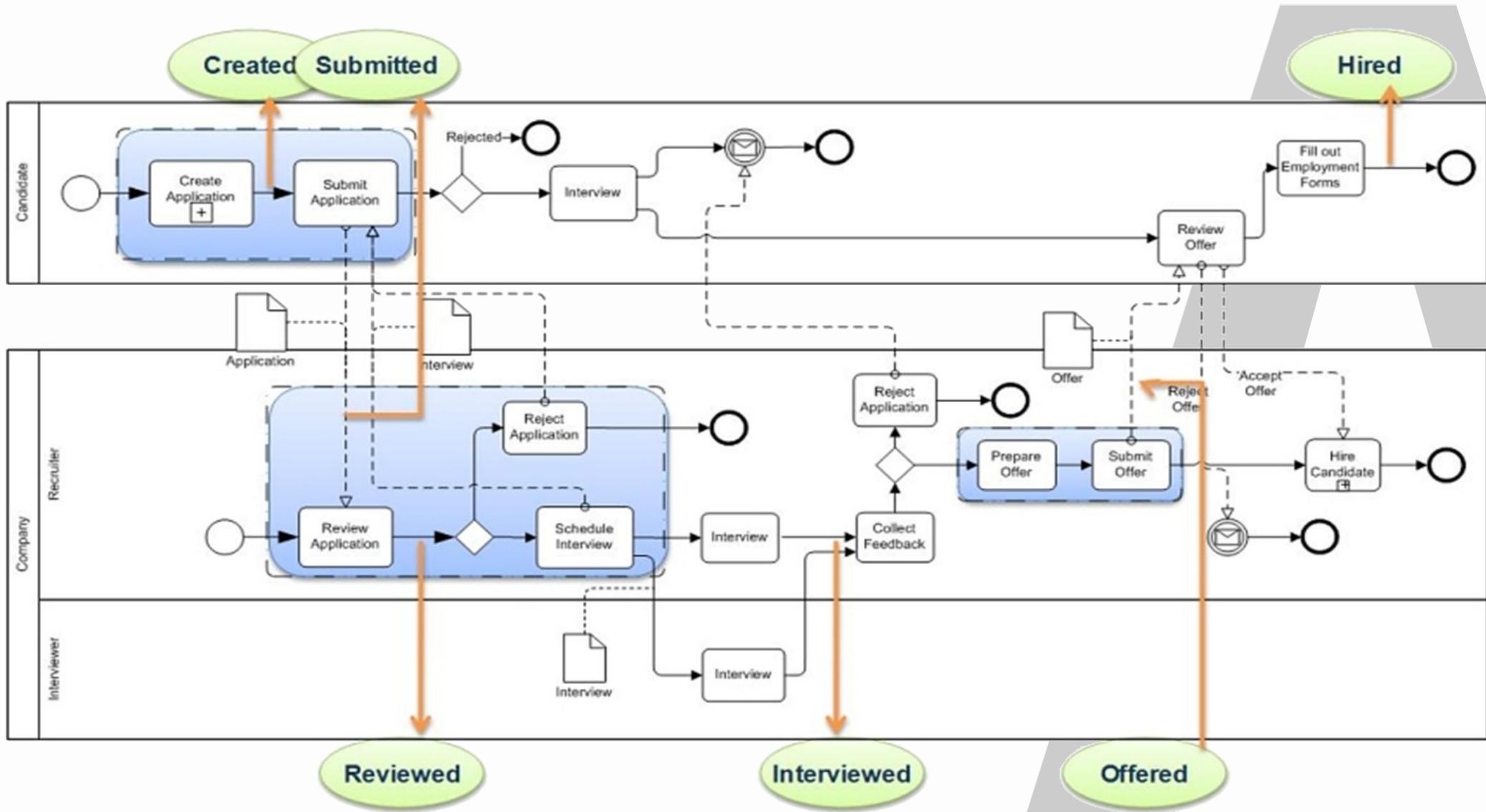
vinicius.mayer@advancedit.com.br

Agenda

- **TIPOS DE PROCESSOS**
- Revolução Científica
- Choque de Paradigmas
- Gestão de
Processos Empíricos
- SCRUM
 - » Artefatos
 - » Papéis
 - » Reuniões
 - » SCRUM Flow



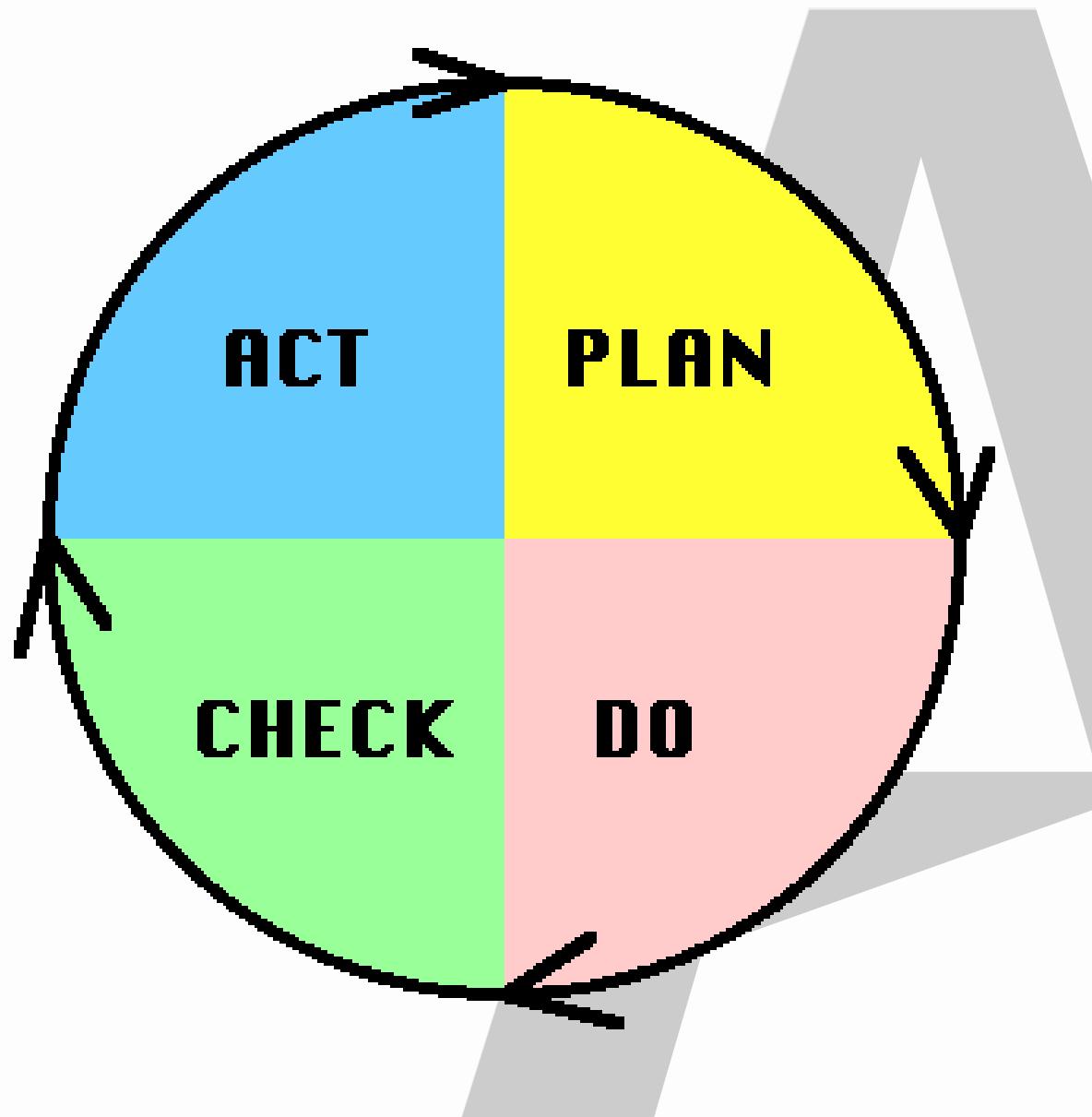
Tipos de processos



Tipos de processos

- Processos definidos

- » Mecanismos subjacentes claramente entendidos
- » Sucesso de atividades claramente entendidas e lineares
- » Capacidade de estimar
- » Tempos de execução



Tipos de processos

- Processo Definido (“Taylourista”)

- » Process-centric (Processos são importantes, Pessoas são recursos)
- » Pré-definição e rigidez top-down (“The One Best Way”)
- » Hierarquia, Comando e Controle
- » Modelo de Qualidade
 - **Escopo fixo entregue dentro do prazo e orçamento**
- » Representantes
 - Waterfall, RUP (antes de 2005), CMMI, ISO, PMI

Tipos de processos



Tipos de processos

- Processos empíricos

- » Complexos, caóticos e com detalhes desconhecidos
- » Atividades podem ser cíclicas e tem durações com muitas variações
- » É difícil de estimar tempos de execução ou mesmo definir atividades a serem realizadas

Paradigmas

● Processo Empírico

- » People-centric
 - Pessoas e Princípios são o foco principal
 - Processos são instrumentos
- » Adaptabilidade e Aperfeiçoamento Contínuo
- » Colaboração, Liderança e Auto-gerenciamento
- » Modelo de qualidade
 - **Satisfação do cliente** por meio da aderência ao seu processo de negócio
- » Representantes
 - SCRUM, XP, FDD, ASD, Lean Development, DSDM, Agile Modeling, MSF Agil, Iconix, OpenUP, RUP (à partir de 2005)

Agenda

- Tipos de Processos
- **REVOLUÇÃO CIENTÍFICA**
- Choque de Paradigmas
- Gestão de
Processos Empíricos
- SCRUM
 - » Artefatos
 - » Papéis
 - » Reuniões
 - » SCRUM Flow



Revolução Científica

- 1905
 - » Albert Einstein e a Teoria da Relatividade
 - » Nova visão de mundo
 - » Um olhar diferente para descrever o mundo → Mudar a visão que cada um tinha do mundo
 - » **Mecânico e Certo (Newton) → Relativo e Incerto (Einstein)**
- A física sofria o que se chama de “**Mudança de Paradigma**”

Revolução Científica

- 1965

- » Thomas Kuhn publica “The Structure of Scientific Revolutions”
- » “**A ciência evolui não pelo acúmulo direto de descobertas, mas pelo surgimento de novas visões de mundo ou 'paradigmas' .**”.

Revolução Científica

- Final do Século XX (para Projetos de Tecnologia da Informação)
 - » Resultados ruins na aplicação das teorias em vigor
 - » Busca por novas formas e teorias para obter melhores resultados
 - » Diversos métodos e técnicas são formados
 - » Tais métodos são convergentes
 - » Novo conjunto de princípios teóricos que contradizem a teoria em vigor

Revolução Científica

- Fevereiro de 2001
 - » Publicação do Manifesto Ágil
 - » Novos princípios formalizados
- Testada e aprovada em projetos do mundo inteiro
- Adeptos do paradigma corrente duvidam e questionam as práticas difundidas por essa nova corrente
- Novo paradigma conquista novos adeptos

Revolução Científica

- Processo de Substituição Conceitual (de Thomas Kuhn)
 - » Anomalias são detectadas nas teorias em vigor
 - » A desconfiança conceitual se instaura
 - » Aplicação e testes de novas idéias são realizados
 - » **Acontece um “Choque de Paradigmas”, que inicia uma batalha conceitual entre as duas correntes de pensamento**
 - » **Ocorre então a expansão da aceitação do novo paradigma**
 - » Substituição do Paradigma (**Revolução Científica**)

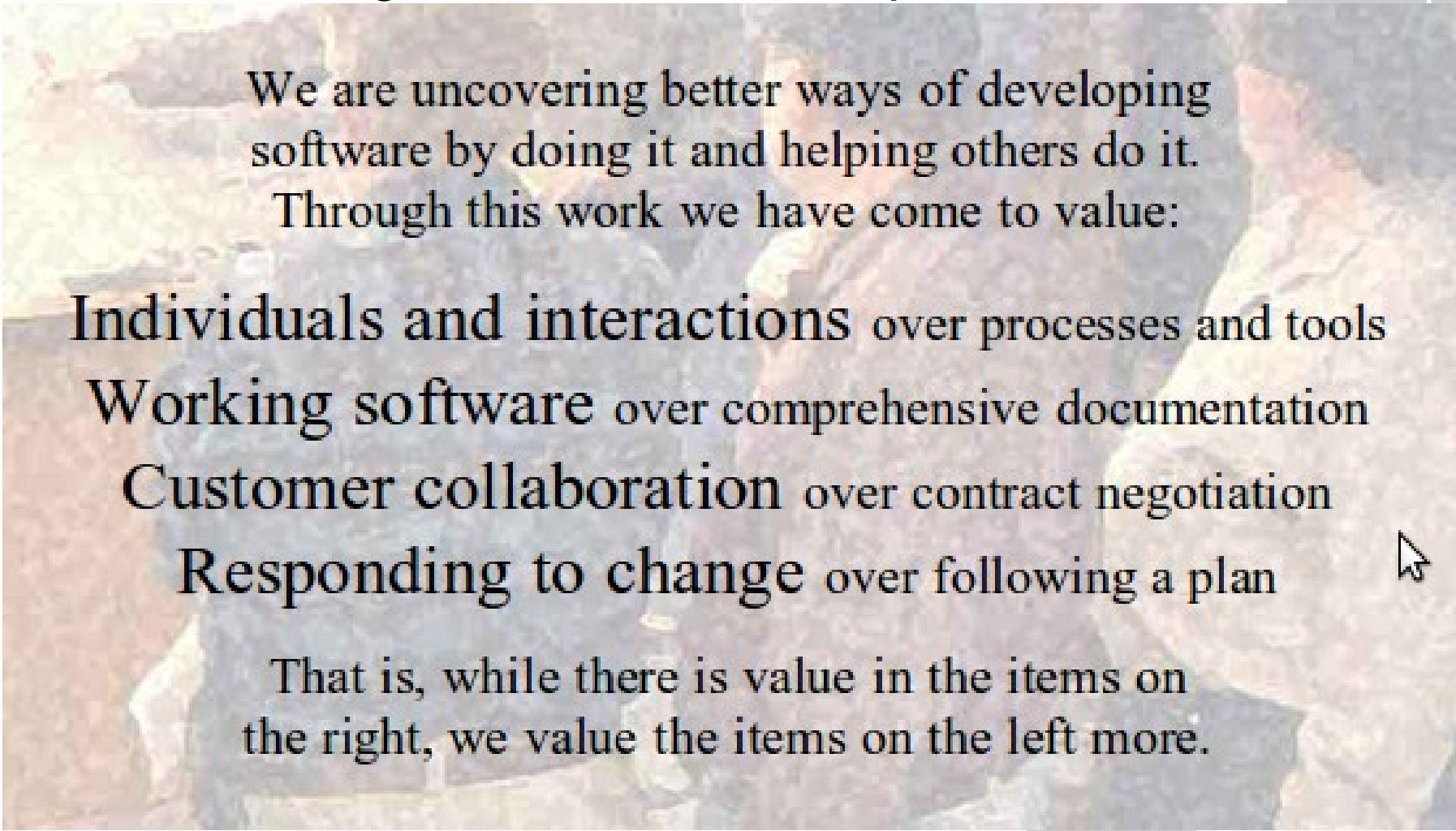
Agenda

- Tipos de Processos
- Revolução Científica
- **CHOQUE DE PARADIGMAS**
- Gestão de
Processos Empíricos
- SCRUM
 - » Artefatos
 - » Papéis
 - » Reuniões
 - » SCRUM Flow



Choque de Paradigmas

- Manifesto for Agile Software Development



We are uncovering better ways of developing software by doing it and helping others do it.

Through this work we have come to value:

Individuals and interactions over processes and tools

Working software over comprehensive documentation

Customer collaboration over contract negotiation

Responding to change over following a plan



That is, while there is value in the items on the right, we value the items on the left more.

Choque de Paradigmas

- Principles behind the Agile Manifesto

- » Our highest priority is to satisfy the customer through early and continuous delivery of valuable software.
- » Welcome changing requirements, even late in development. Agile processes harness change for the customer's competitive advantage.
- » Deliver working software frequently, from a couple of weeks to a couple of months, with a preference to the shorter timescale.
- » Business people and developers must work together daily throughout the project.
- » Build projects around motivated individuals. Give them the environment and support they need, and trust them to get the job done.
- » The most efficient and effective method of conveying information to and within a development team is face-to-face conversation.

Choque de Paradigmas

● Principles behind the Agile Manifesto

- » Working software is the primary measure of progress.
- » Agile processes promote sustainable development. The sponsors, developers, and users should be able to maintain a constant pace indefinitely.
- » Continuous attention to technical excellence and good design enhances agility.
- » Simplicity--the art of maximizing the amount of work not done--is essential.
- » The best architectures, requirements, and designs emerge from self-organizing teams.
- » At regular intervals, the team reflects on how to become more effective, then tunes and adjusts its behavior accordingly.

Choque de Paradigmas

● Planejamento

	Tradicional (Definido)	Ágil (Empírico)
Planejamento de	Atividades	Entregas
Plano de	Projeto	Releases
Milestones	Fases finalizadas	Metas de negócio atendidas
Progresso	Atividades concluídas	Funcionalidades entregues
Quando revisado procura-se por	Atividades esquecidas	Features esquecidas
Plano	Seguido à risca	Definido conforme o rendimento medido
Acompanhado pelo	Gerente de projeto	Por toda a equipe
Atualização	Complexa	Simples (ou automatizada)

Choque de Paradigmas

● Levantamento de Requisitos e Análise

	Tradicional (Definido)	Ágil (Empírico)
No início do projeto os requisitos são descritos	Detalhadamente	Em alto nível
Especificações	Contém informações para orientar o desenvolvedor	Contém informações geradas durante o desenvolvimento
O cliente participa	Apenas descrevendo o problema	Descrevendo o problema e elaborando a solução
O cliente aprova	Especificações	Software funcional
Mudanças nos requisitos são	Evitadas	Incentivadas
Cenários de Negócio são	Descritos em documentos	Automatizados e utilizados para validação do comportamento do software

Choque de Paradigmas

● Projeto (de software) e Codificação

	Tradicional (Definido)	Ágil (Empírico)
Modelos são utilizados para	Documentação	Exploração do requisito
Representado por	Documentação em papel	Código-fonte
	Código é gerado à partir de modelos	Modelos são gerados à partir do código
O projeto (de software) é feito por	Um arquiteto	Pela equipe
Código é escrito por	Um programador	Um desenvolvedor
Alterações no código	Só para novas funcionalidades	Também para melhorar o projeto (de software)
Rastreabilidade	Impositiva para gerenciar alterações	Conquistada por meio da interligação automatizada dos artefatos

Choque de Paradigmas

● Testes e Qualidade

	Tradicional (Definido)	Ágil (Empírico)
Testes são	Manuais	Automatizados
Função da equipe de testes	Encontrar defeitos	Evitar defeitos
Qualidade garantida por	Inspeções ao final do processo	Incorporação de qualidade ao produto e antecipação das inspeções
Qualidade é perseguida com	Aumento do controle e da burocracia	Disciplina no uso de boas práticas, excelência técnica, investimento no conhecimento e na aquisição de novos skills pelas pessoas

Agenda

- Tipos de Processos
- Revolução Científica
- Choque de Paradigmas
- **GESTÃO DE
PROCESSOS EMPÍRICOS**
- SCRUM
 - » Artefatos
 - » Papéis
 - » Reuniões
 - » SCRUM Flow



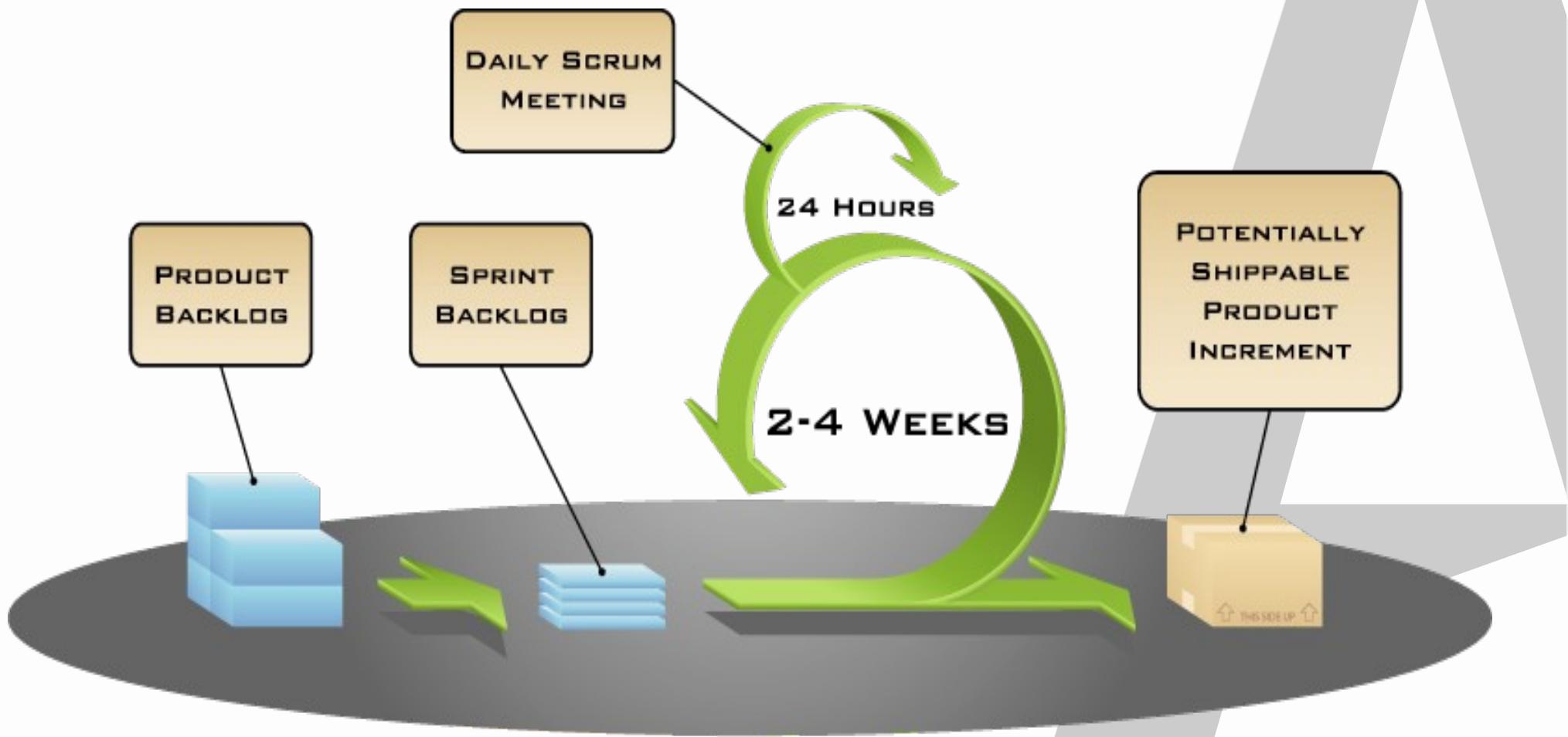
Gestão de Processos Empíricos

- Fixar a maior quantidade de parâmetros possível
 - » Contexto
 - Prazo, esforço, estrutura da equipe, custo
 - » Entradas
 - Product Backlog
 - Prioridades
 - Estimativas
 - » Saídas
 - Objetivo
 - Critérios de avaliação



Gestão de Processos Empíricos

- SCRUM



COPYRIGHT © 2005, MOUNTAIN GOAT SOFTWARE

Agenda

- Tipos de Processos
- Revolução Científica
- Choque de Paradigmas
- Gestão de
Processos Empíricos
- **SCRUM**
 - » ARTEFATOS
 - » Papéis
 - » Reuniões
 - » Artefatos



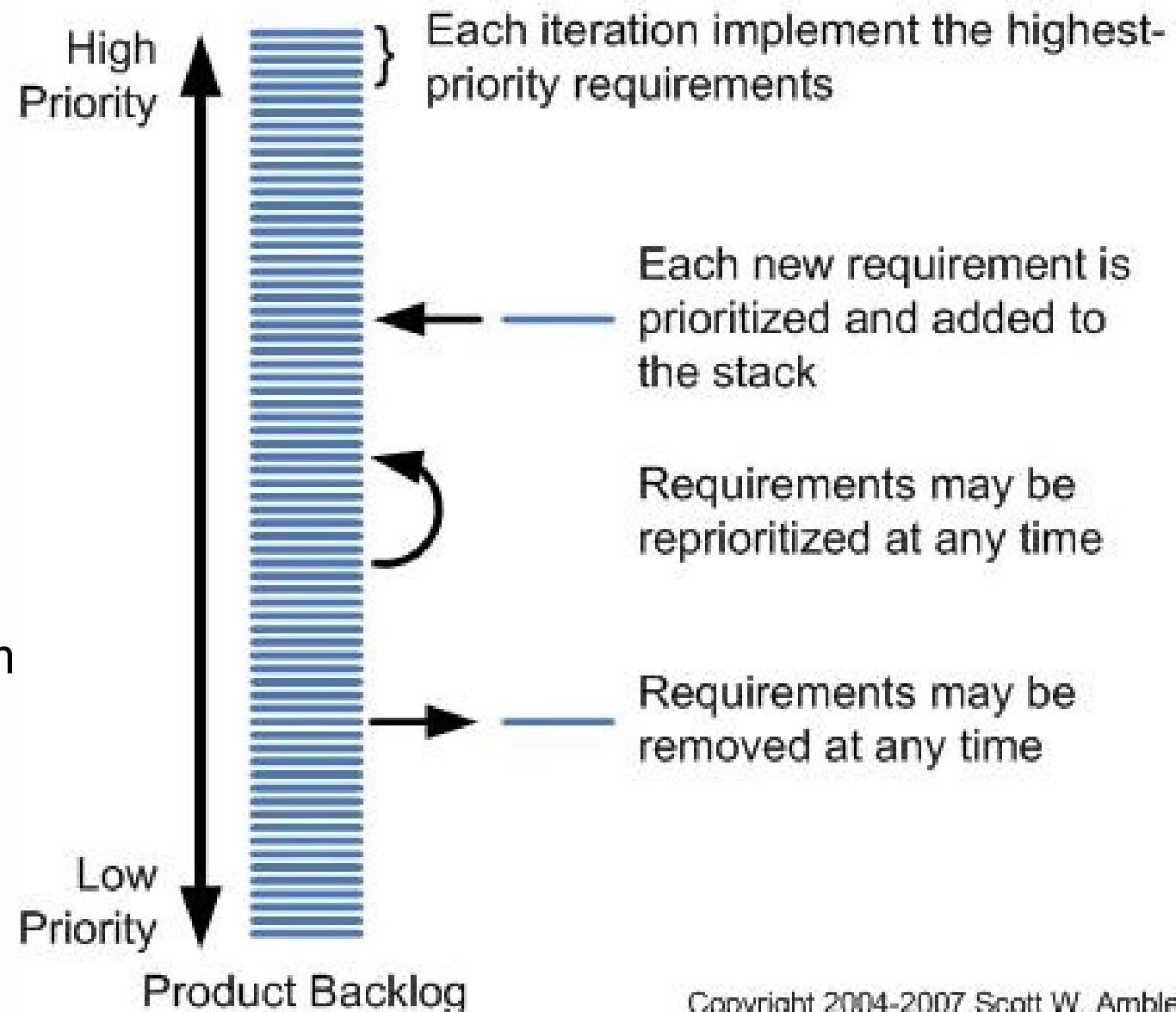
Artefatos

- Product Backlog
 - » Entregáveis
 - » User Stories
 - » Requisitos funcionais
 - » Priorizado e estimado
 - » Itens de maior prioridade com mais detalhes
 - » Mantido e postado de forma visível

Artefatos

● Product Backlog

- » Estregáveis
- » User Stories
- » Requisitos Funcionais
- » Priorizado
- » Estimado
- » Mais detalhado em itens de maior prioridade



Copyright 2004-2007 Scott W. Ambler

Artefatos

● User Stories

- » Como <papel do usuário>, quero <funcionalidade> para <valor de negócio/objetivo>
- » A fim de <valor de negócio/objetivo>, como um <papel de usuário>, quero <funcionalidade>

- » Forma simples e objetiva de “representar” um “requisito funcional”
- » É uma descrição resumida de uma funcionalidade que trará valor para o usuário do software e seu cliente
- » Enfatiza a **comunicação verbal** ao invés da escrita; ou seja, o foco está na representação do requisito e não na documentação detalhada

Artefatos

● User Stories

» INVEST in good stories and SMART tasks

- Independente (S): deve ser possível demonstrar a funcionalidade se dependências
- Negociável (S): deve existir valor agregado a realização da história
- Valorável (R): deve ser possível verificar o valor agregado ao produto
- Estimável (M): deve ser possível definir o tamanho do problema
- Pequena (T): deve ser suficientemente pequena para caber num ciclo de produção (Sprint)
- Testável (A): deve ter critérios objetivos de comprovação da realização da mesma

Artefatos

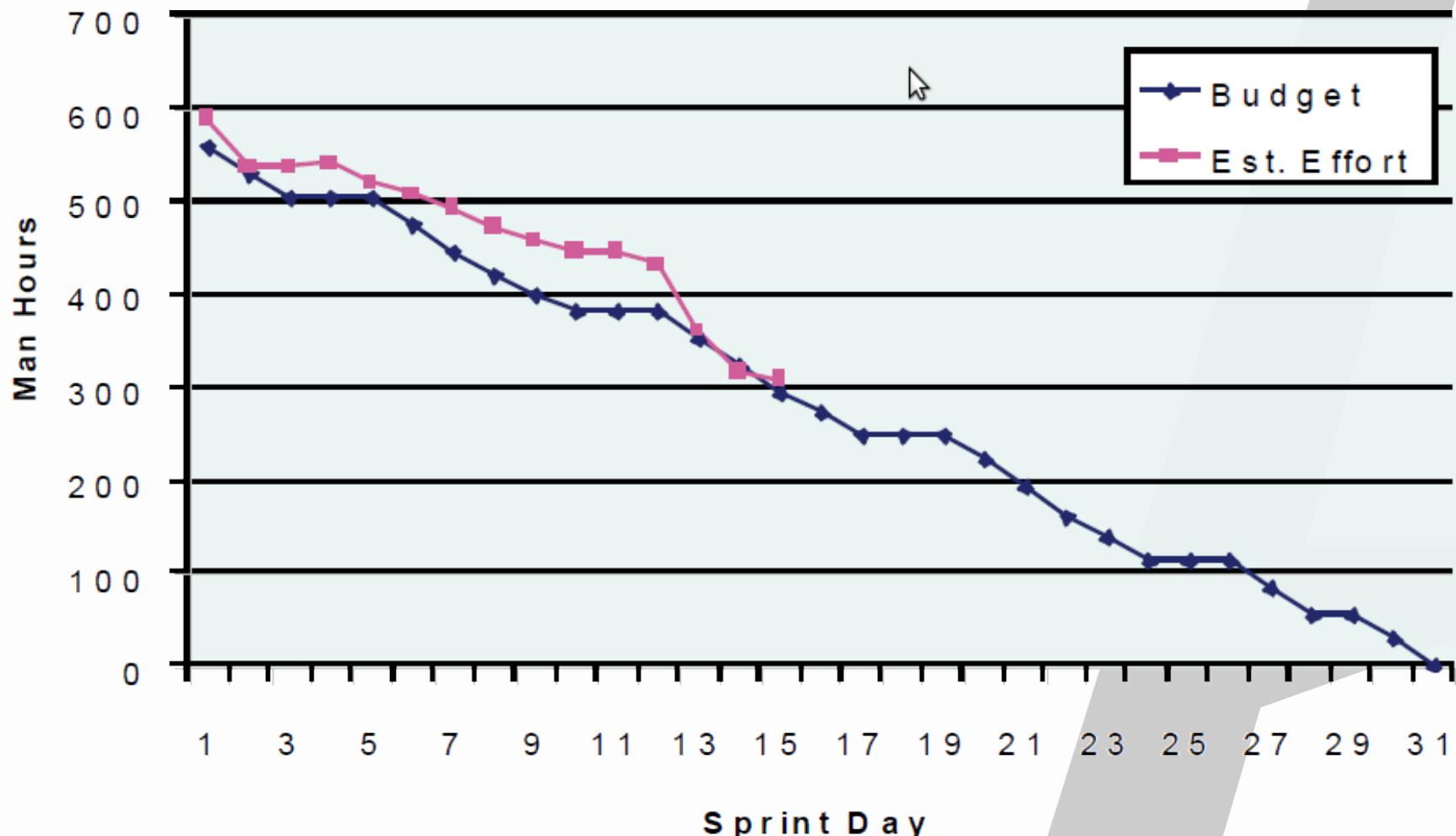
● User Stories

- » Como hóspede, quero fazer uma reserva no hotel
- » Como hóspede, quero cancelar minha reserva
- » Como agente de viagens, quero ver as fotos do hotel

- » Como auxiliar do hotel, posso confirmar o cancelamento das reservas
 - Como hóspede PREMIUM, posso cancelar minha reserva até o último minuto, sem pagar multa
 - Como hóspede comum, posso cancelar minha reserva até 24 horas antes, sem pagar multa
 - Como qualquer hóspede, receberei a confirmação de cancelamento da reserva por e-mail

Artefatos

- Burndown Chart



Agenda

- Tipos de Processos
- Revolução Científica
- Choque de Paradigmas
- Gestão de
Processos Empíricos
- **SCRUM**
 - » SCRUM Flow
 - » **PAPÉIS**
 - » Reuniões
 - » Artefatos



Papéis

- Product Owner
- SCRUM Master
- Team



Papéis

● Product Owner

- » Define as funcionalidades do produto
 - Levantar requisitos
- » Decide datas de lançamento e conteúdo
 - Gerenciar o plano de entregas (Release Plan)
- » Prioriza funcionalidades de acordo com o valor de negócio
- » Ajusta funcionalidades e prioridades
- » Aceita ou rejeita o resultado dos trabalhos
 - Aceitar o software no final de cada iteração
- » Gerenciar e priorizar o Product Backlog
- » **Responsável pela lucratividade do projeto (ROI)**

Papéis

● SCRUM Master

- » Trabalhar com o Product Owner
 - Representa a gerência para o projeto
- » Remover impedimentos
- » Manter o processos funcionando
- » Socializar SCRUM na organização
- » Responsável pela aplicação dos valores e práticas do Scrum
- » Garante a plena funcionalidade e produtividade da equipe
- » Garante a colaboração entre os diversos papéis e funções
- » Escudo para interferências externas

Papéis

● Team

- » Entre 5 e 9 pessoas
- » Multi-funcional (Programadores, testadores, desenvolvedores de interfaces, etc.)
- » Tempo integral - Raras exceções (Ex.: Administrador de Base de Dados)
- » Auto-organizável - Idealmente, sem cargos definidos
- » Trocas só na mudança de Sprints

- » Estimar o tamanho dos itens do Product Backlog
- » Assumir o compromisso com a entrega de um incremento de software funcional
- » E ENTREGÁ-LO!
- » Manter o seu próprio progresso (com o auxílio do SCRUM Master)
- » Auto-organizados, mas responsáveis pela entrega daquilo prometido para o Product Owner

Agenda

- Tipos de Processos
- Revolução Científica
- Choque de Paradigmas
- Gestão de
Processos Empíricos
- SCRUM
 - » Artefatos
 - » Papéis
 - » **REUNIÕES**
 - » SCRUM Flow



Reuniões

- Planning 1 Meeting
 - » Estimation Meeting
- Planning 2 Meeting
- Daily SCRUM Meeting
- Sprint Review Meeting



Reuniões

● Estimation Meeting

- » Preparação do Sprint Planning
- » Estimativas formais
- » Pelo menos duas reuniões por Sprint
- » Estimar unicamente tamanho, nunca tempo
- » Atualizar o Product Backlog com as estimativas
- » É uma das entradas para fazer o Release Plan

Reuniões

● Planning Meeting

» Entradas

- Product Backlog
- Capacidade (“produtividade”) da equipe
- Estabilidade da tecnologia
- Novas funcionalidades entregues

» Saidas

- Objetivo do próximo Sprint
- Selected Product Backlog
- Sprint Backlog

Reuniões

● Daily SCRUM Meeting

- » Reunião diária de 15 minutos
- » Mesmo local e hora todos os dias
- » Sala de reuniões fora do ambiente de trabalho
- » Membros da equipe respondem
 - O que eu fiz ontem?
 - O que vou fazer hoje?
 - Existe algum impedimento?
- » Impedimentos e decisões



Reuniões

● Sprint Review Meeting

» Quando uma equipe diz “Pronto!” o que isso significa?

- Código aderente aos padrões
- Está “limpo”!
- Refactoring
- Testados unitariamente
- Integrado
- Suite de testes automatizados
- Ambiente de desenvolvimento centralizado:
 - Código
 - Padrões de codificação
 - Builds automatizados

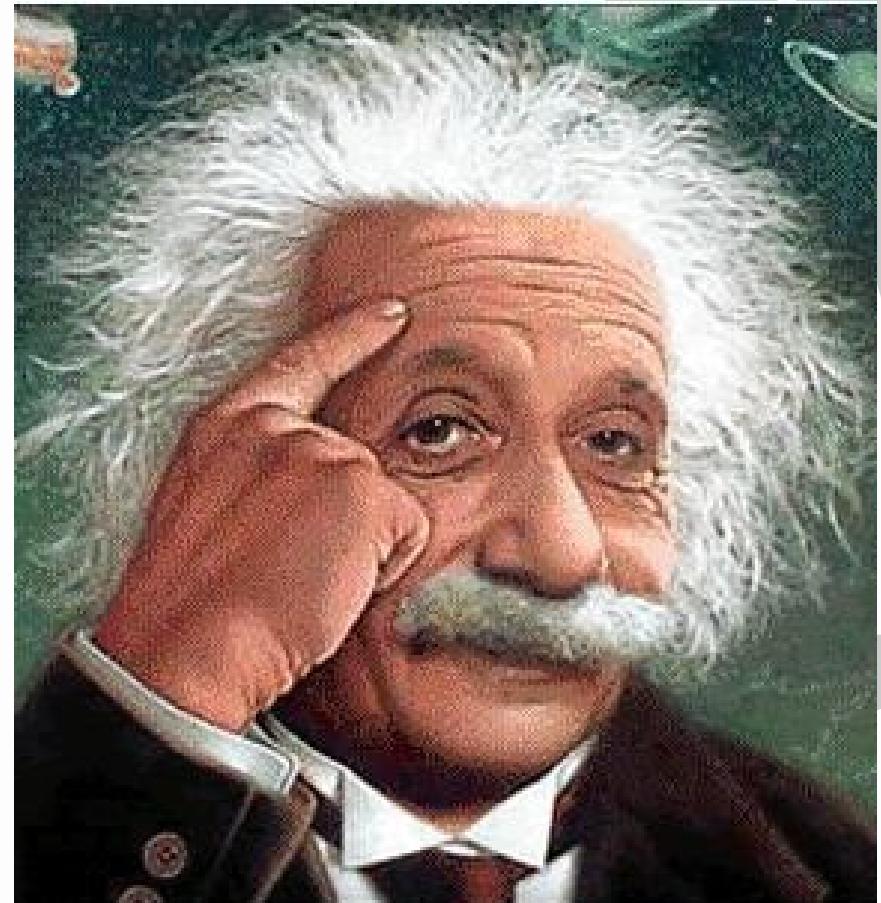
» Apresentação das funcionalidades implementadas

- Avaliação do projeto
- Relacionado aos objetivos do Sprint

Reuniões

● Sprint Retrospective Meeting

- » Lições aprendidas!
- » Processo de retrospectiva
 - O que foi bom?
 - O que foi ruim?
 - Quem tem o controle?
 - Priorizar
- » O Product Owner não participa
- » Revisão e melhoria do processo



Reuniões

● Sprint Retrospective Meeting

- » Devolver funcionalidades finalizada para o Product Backlog e re-priorizar
- » Remover funcionalidade do Product Backlog que a equipe inadvertidamente possa ter implementado
- » Trabalhar com o SCRUM Master para reformular a equipe
- » Re-priorizar o Product Backlog para poder tomar vantagem de oportunidade que as funcionalidades finalizadas apresentam

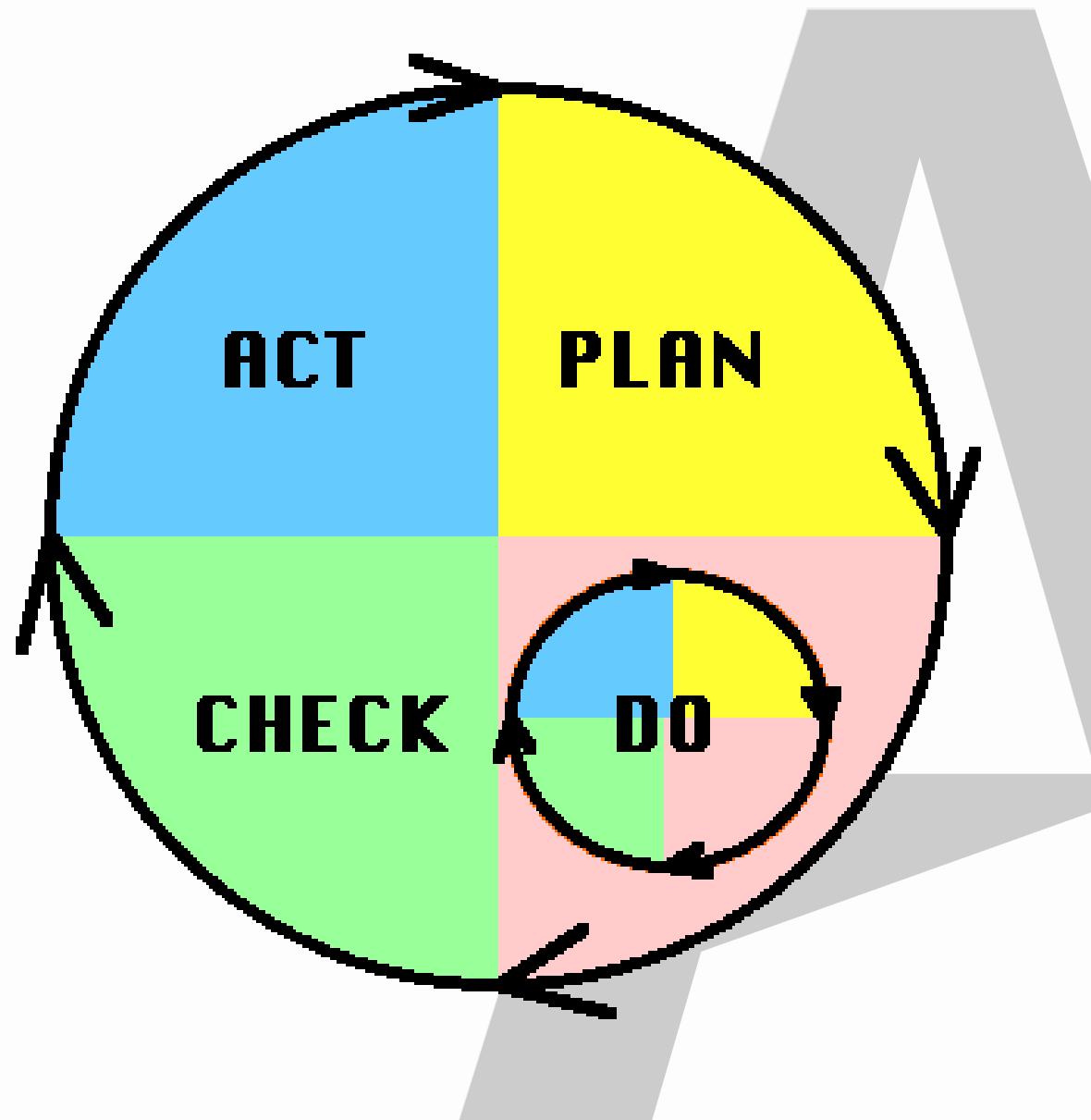
Agenda

- Tipos de Processos
- Revolução Científica
- Choque de Paradigmas
- Gestão de
Processos Empíricos
- SCRUM
 - » Artefatos
 - » Papéis
 - » Reuniões
 - » SCRUM FLOW

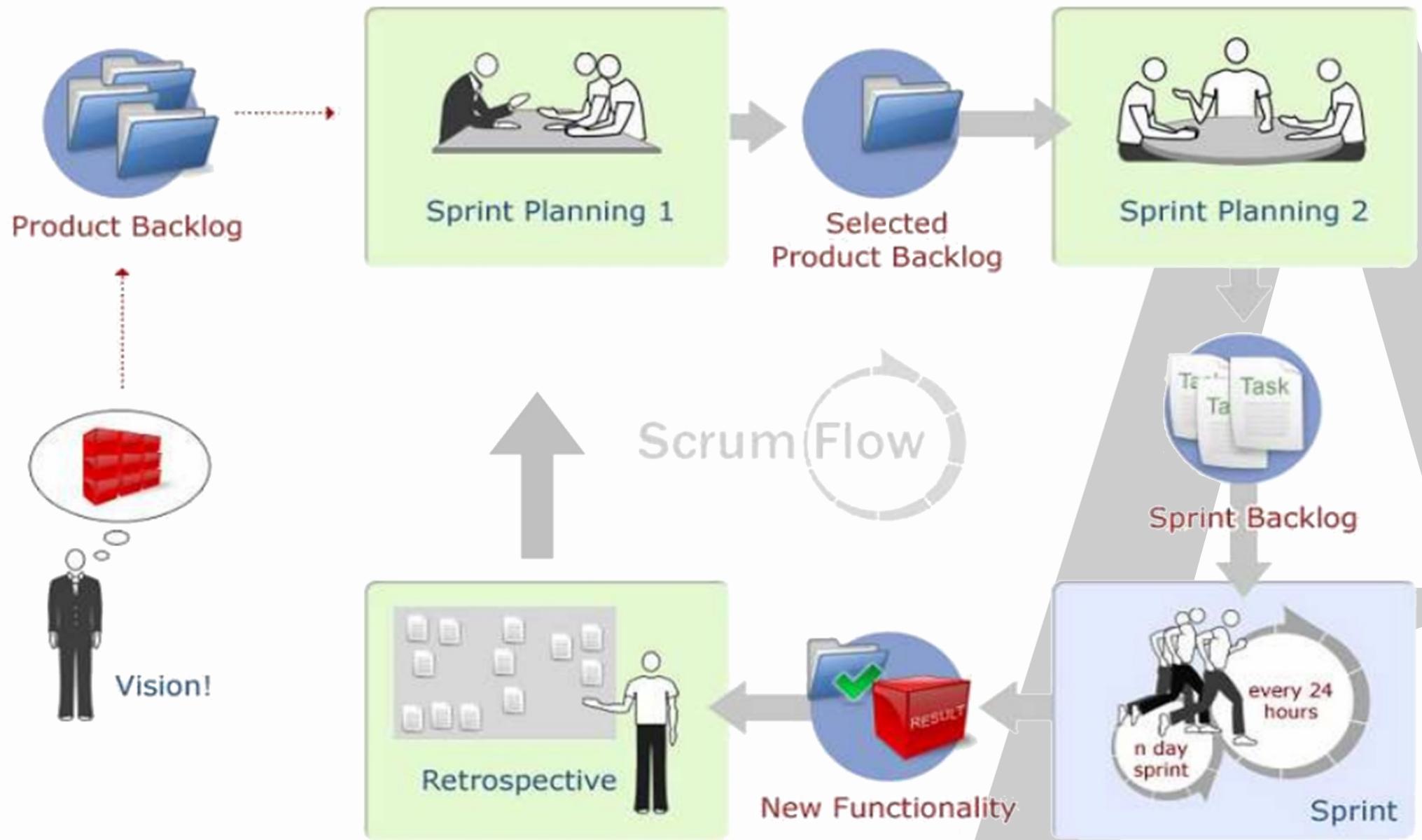


SCRUM Flow

- Sprint Planning
 - » PLAN
- Sprint
 - » DO
- Sprint Review
 - » CHECK
- Sprint Retrospective
 - » ACT



SCRUM Flow

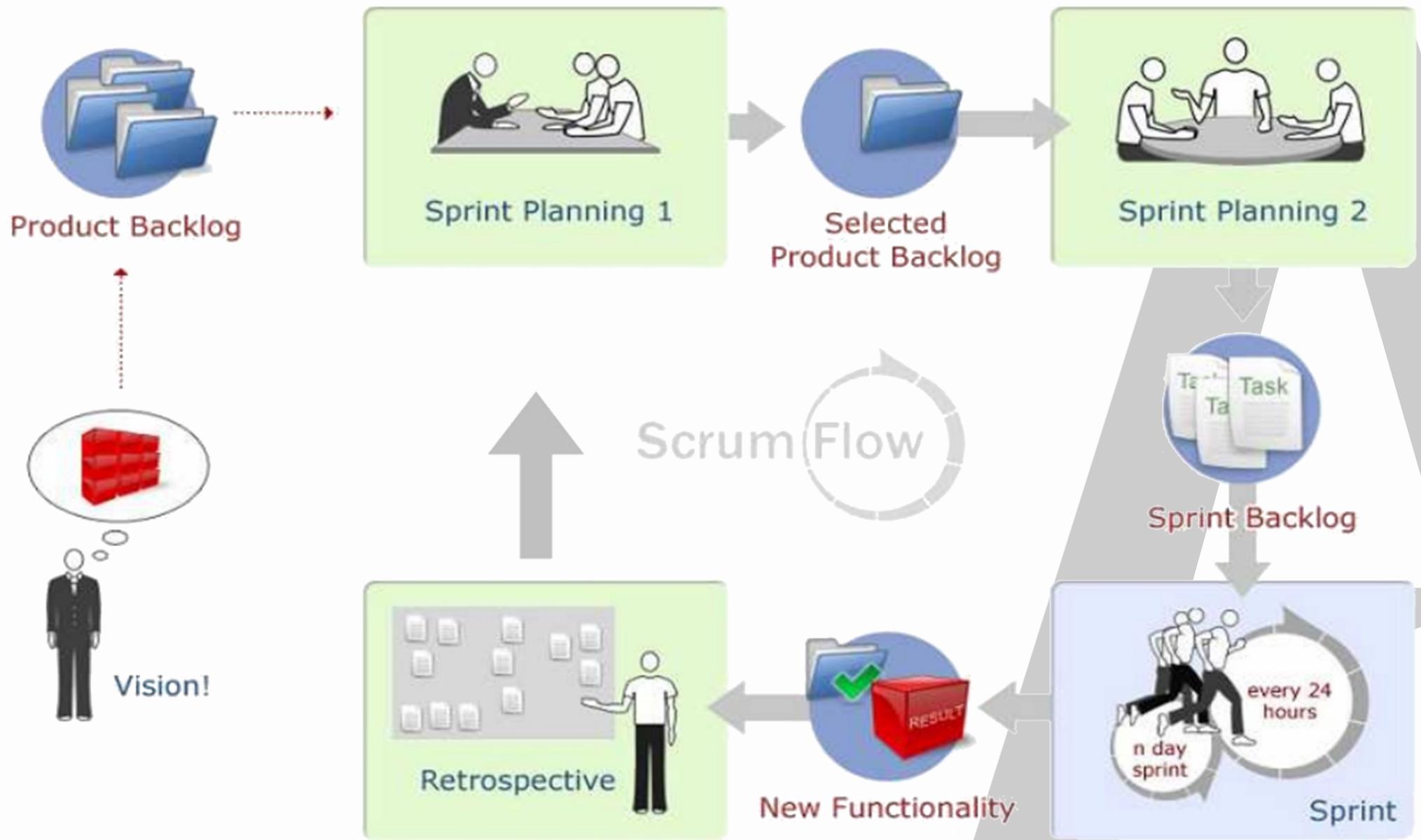


SCRUM Flow

● Sprint

- » Normalmente 30 dias úteis – corridos, ou de 2 a 4 semanas
- » Projetos Scrum progridem em uma série de “sprints”
- » Um período constante leva a um melhor “ritmo”
- » O produto é projetado, codificado e testado durante o Sprint
- » Nenhuma mudança durante o Sprint
 - Planejar a duração dos Sprints de acordo com o máximo tempo com o qual você pode se comprometer a deixar as mudanças fora deles

SCRUM Flow



Duvidas



Obrigado!

- Sem importar o que descubramos, entendemos e realmente acreditamos que todo mundo fez o melhor trabalho que pode, dado o que conheciam na hora, suas habilidades, capacidades, recursos disponíveis e, a própria situação no momento.

Norman Kerth, 2001