



Leve o seu SCM à outro nível com FeatureToggle (aka FeatureFlag)

CIST



André Paulovich

Systems Architect & Community Manager - CI&T

andrep@ciandt.com

MCP | MCTS | MCT | MCAD | MCSD.Net | MVP Asp.Net 2011 à 2014



@andrepaulovich

me add aí



Use a hashtag **#TDC** **#dotnet** **#toggle**



Tirem muitas fotos!



Qual é o problema que
estamos tentando
resolver?



CI

CD



CI

CD

VIDA DE PROGRAMADOR COM.BR

/* Definição de Pronto */

```
String[] senders = {  
    "Edson Yanaga",  
    "Bruno Souza"  
};
```



#1438







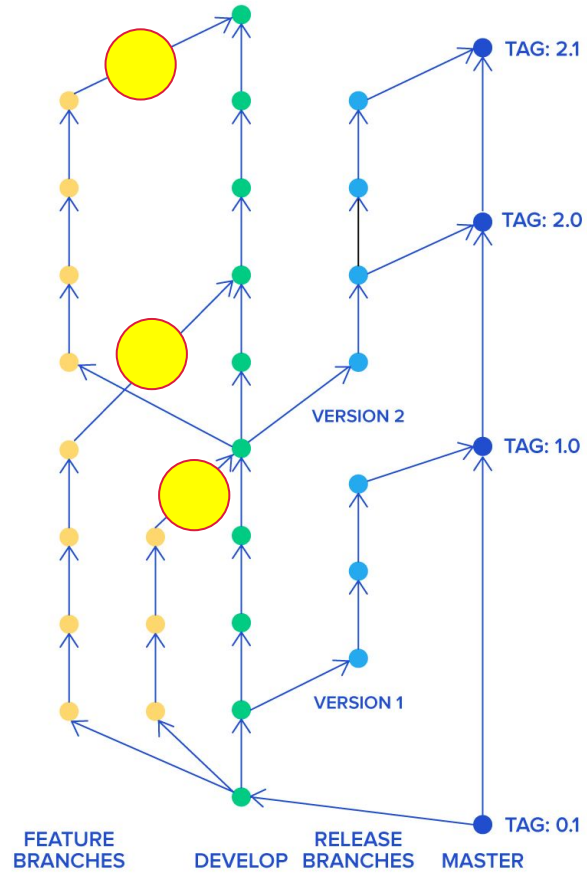




São tantos possíveis
problemas envolvidos
na **“implantação”** de
uma aplicação...



 “ponto de falha”?



GIT MERGE

**Isso acontece porque o
nosso SCM é físico**





Normalmente estes
problemas são ainda
piores em cenários em
que a implantação é
um **“big bang”**



Nova Feature



Nova Feature

BUUMMM!



**Existe “vida” além do
GitFlow... a alternativa
são os toggles!**



O que são FeatureToggles? FeatureFlags?

Que negócio é esse que todos andam falando?




Feature Flags

separam **deploys de**
código do lançamento
de **features.**



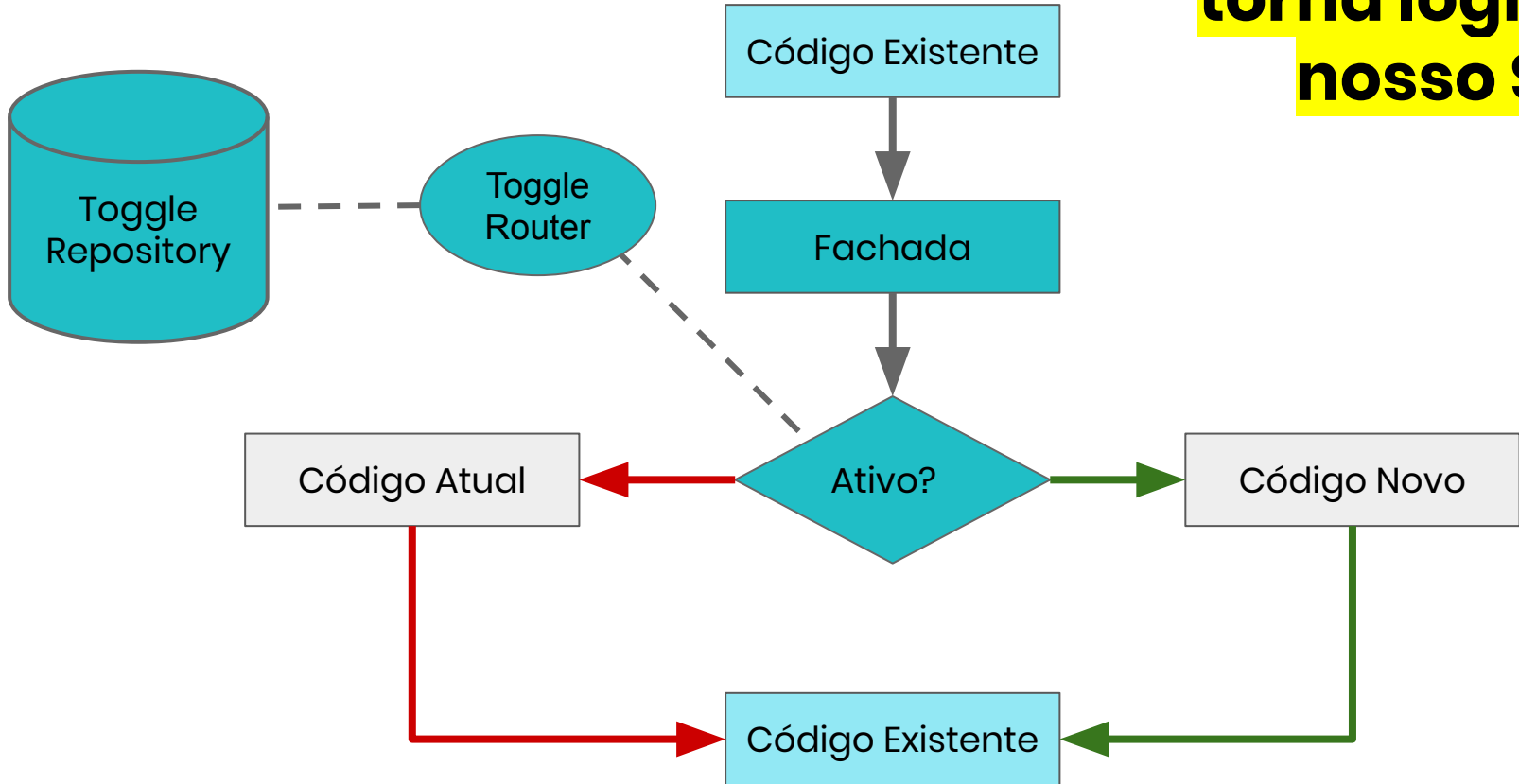
Como funciona?

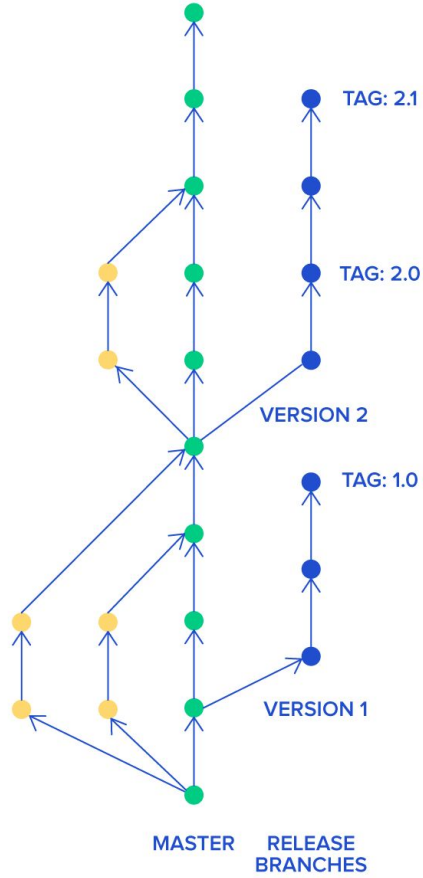
Explica mais aí...



```
if ( Flag.AlgumMecanismo.ativo )
{
    return funcionamentoNovo();
}
else
{
    return funcionamentoAntigo();
}
```

**Essa abordagem
torna lógico o
nosso SCM**

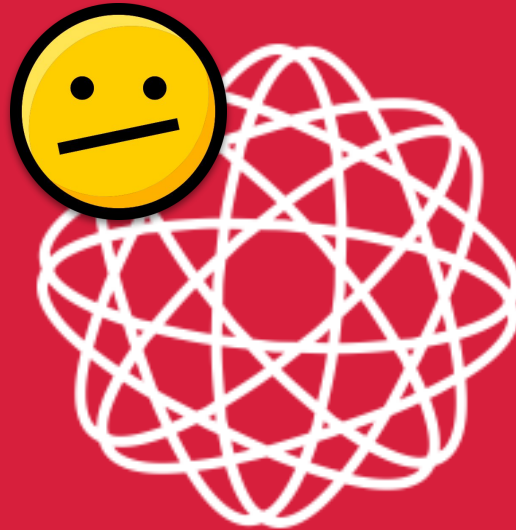




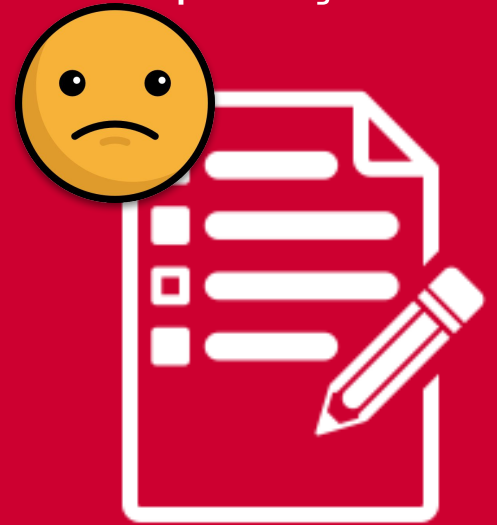
Aumenta a
velocidade de
entrega de valor.



Maior complexidade
de código, porém
com menores riscos
na gestão de
produtos.



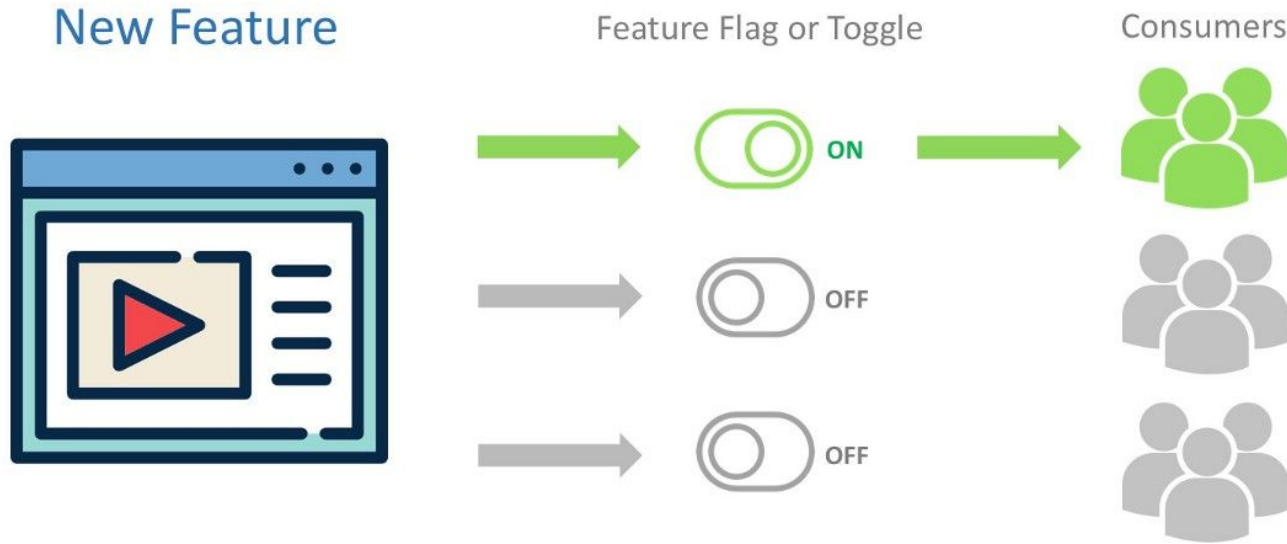
Quebra de
paradigma
convencional de
construção de
aplicações.





Pré-requisitos para adotar esta estratégia

Gestão de toggles



**Precisa ser algo simples e intuitivo,
pois vamos compartilhar esta
gestão com os PO's**



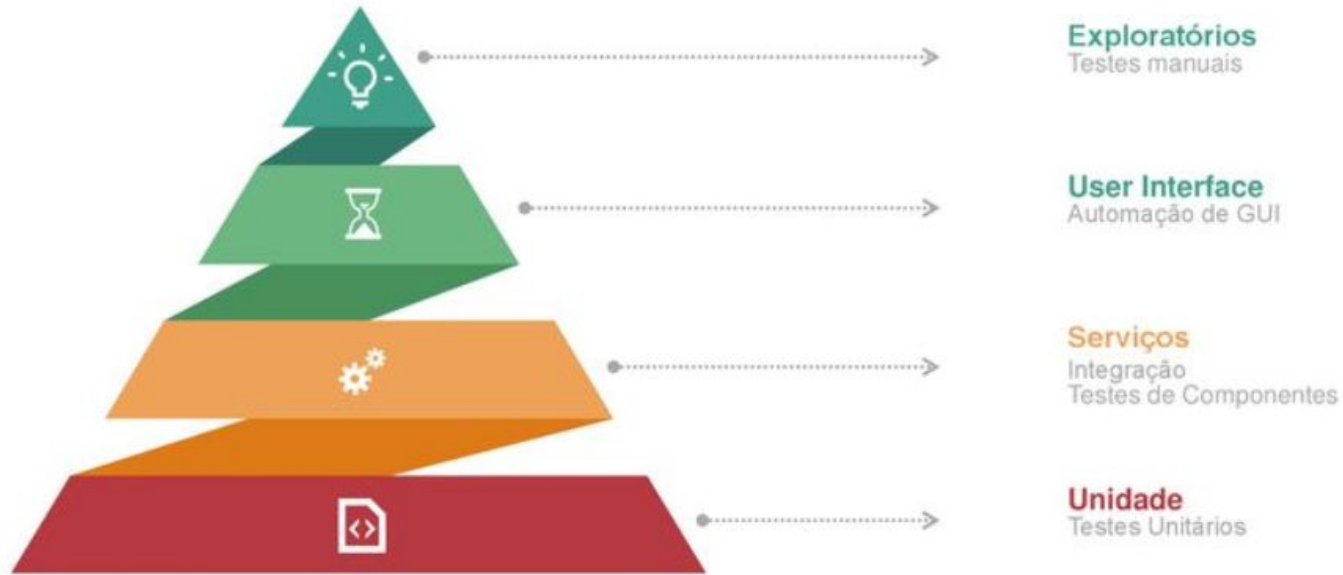
**Todo mundo precisa
aprender a pensar diferente**

Entender que tudo está interligado...



Amadurecer todo ciclo de vida de gestão de produto

Testes são código de “produção”



Todo mundo precisa aprender a pensar diferente

with Agile: Software Development Using Scrum – Mike Cohn



**Assim impedimos que
versões “perigosas”
evolucam no pipeline**



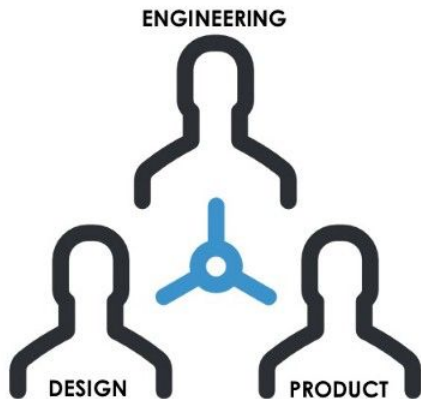
**Vantagens de utilizar
esta abordagem**

**Em caso de
problema, clique
aqui!**

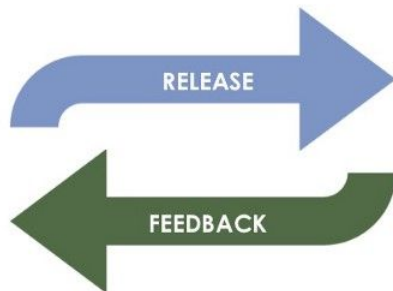


Feedback Loop

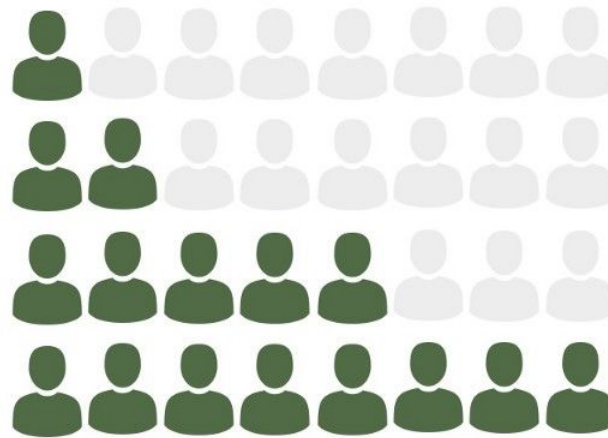
RELEASE STRATEGY



FEEDBACK LOOP



ROLLOUT



**Lançamento incremental
(aka softlaunch)**



Existing page



Experimental page



Testes A/B

**Publicação em
horários específicos**

22
SCHEDULE



Quem utiliza?

NETFLIX

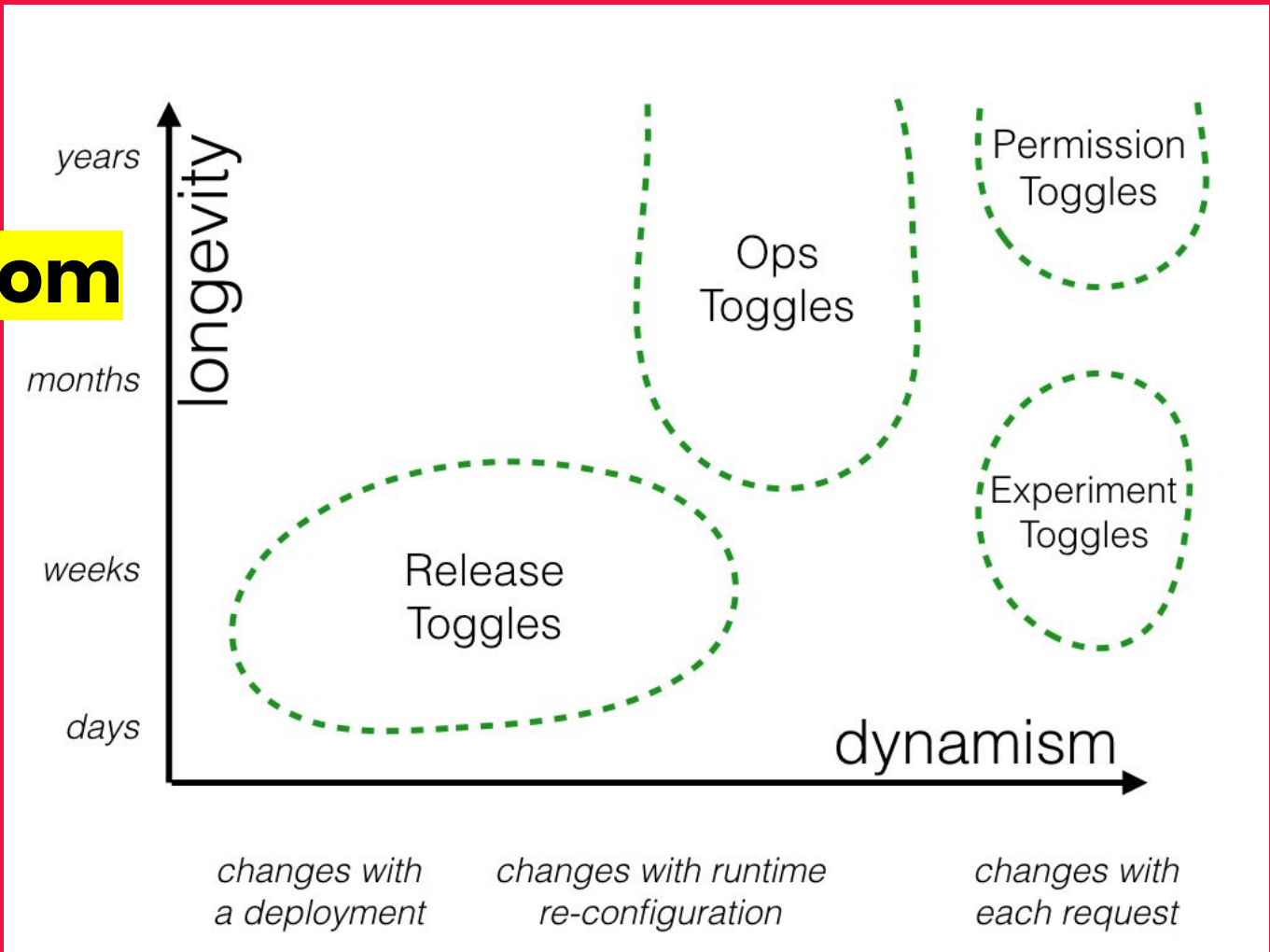




Como começar?




**Começe com
“release
toggles”**





Ferramentas



Launch Darkly
rollout.io
Optimizely
Featureflow
ConfigCat
Unleash
FF4J

**Escolha uma
ferramenta
adequada ao
seu contexto**



Erre cedo...



Fácil demais





Use toggles em todos os ambientes!

Não caia na armadilha de usar git flow nos ambientes de desenvolvimento e testes e toggles como estratégia de ativação apenas.



**A subida é difícil,
mas a vista lá de cima
vale a pena!**



Referências

<https://pt.slideshare.net/geekle/devops-5348895> (Apresentação - Matthew Jones)

<http://bit.ly/2nJyYTC> (Apresentação - João Cunha)

<https://www.toptal.com/software/trunk-based-development-git-flow> (Artigo - Konrad Gadzinowski)

<https://hackernoon.com/feature-flag-driven-releases-7a7a5fee6ba7> (Artigo - Justin Baker)

<https://www.hipsthetic.com/emoji-icon-sets/> (Iconset - Emojis)

O Potencial de Feature Toggles (Apresentação - Carlos Pisani)

<https://martinfowler.com/bliki/TestPyramid.html> (Artigo - Martin Fowler)

<https://www.azuredevopslabs.com/labs/vstsextd/launchdarkly/> (Tutorial - Azure Devops Labs)

CIST



André Paulovich

Systems Architect & Community Manager - CI&T

andrep@ciandt.com

MCP | MCTS | MCT | MCAD | MCS.D.Net | MVP Asp.Net 2011 à 2014



@andrepaulovich

Obrigado!