

# Topologia de times

Construindo times ~~de alta performance~~ confiantes



# Fernando Cezar



Quase 20 anos de carreira

Dev, Product Owner, Product Manager, Scrum master, DevOps, Diretor de TI e mais algumas coisas que esqueci.

Atualmente:

- Instrutor Freelancer - Alura para Empresas
- #OpenToWork

Certified Scrum Master (CSM®) pela Scrum Alliance

Mentor para evolução ou transição de carreira e agilidade

Opinador profissional do trabalho dos outros

Bebedor semi-profissional de cerveja

Produtor de conteúdo:

- Host - [devscansandos.com.br](https://devscansandos.com.br)
- Host - [Codecon.dev](https://codecon.dev)



@focezar em todas  
as redes sociais

# A teoria da topologia de times



# De onde vem a teoria da topologia de times?

- Construída baseada em problemas de times de DevOps e tem como base a Teoria da Cibernética, Framework Cynefin e Pensamento Sistêmico
- Não aborda exatamente a produtividade do times, mas sim a confiança entre os times.
- Tem foco em TI, mas pode ser aplicado a qualquer área que tenha times que têm dependência para realizar seus trabalhos.

# O livro

- Escrito por Matthew Skelton e Manuel Pais
- Lançado em Setembro de 2019
- O livro é dividido em 4 grandes temas:
  - Carga Cognitiva
  - Lei de Conway
  - Topologia dos times
  - Modos de comunicação

# Carga Cognitiva



# O que é Carga Cognitiva?

- É a quantidade total de esforço mental sendo usada na memória de trabalho.
- Carga cognitiva está relacionada com a quantidade de informação que a memória de trabalho pode armazenar ao mesmo tempo.
- Um time nunca deveria lidar com uma carga cognitiva que vai além do seu limite.

**A ideia de entender a carga cognitiva e a relação dela com a construção de times é importante, principalmente, devido às perdas de não considerá-la nesse processo. Quando a carga cognitiva não é considerada, as equipes se espalham, tentando cobrir uma quantidade excessiva de responsabilidades e domínios.**

# A carga cognitiva pode ser classificada de 3 maneiras

- Intrínseca – lida com a complexidade de uma nova informação;
- Estranha (Irrelevante) – informação distrativa ou desnecessária;
- Germane (Relevante) – processa e integra as informações existentes e as novas.

**O ideal é que os times tenham tempo para focar na carga Relevante (Germane). Isso porque, dessa forma, eles poderão pensar o seu domínio de maneira mais complexa.**

# Lei de Conway



# O que é a Lei de Conway?

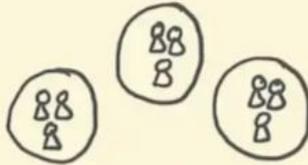
- A Lei defende que, ao desenhar seus sistemas, as companhias estão fadadas a produzir cópias de suas estruturas de comunicação;
- Portanto, se sua comunicação com a área de negócio não é clara, não adiantará trabalhar com as melhores tecnologias e engenheiros, a chance de terem um produto de sucesso será pequena;
- Por outro lado, se uma organização é arranjada em silos funcionais como QA, Data, Frontend, Backend, etc, é improvável que se produza um produto bem arquitetado de ponta a ponta.

PARAPHRASED

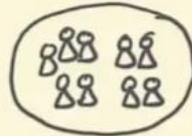
## CONWAY'S LAW

THE STRUCTURE OF SOFTWARE WILL MIRROR THE STRUCTURE OF THE ORGANISATION THAT BUILT IT *for example*

ORGANISATION



SMALL DISTRIBUTED  
TEAMS

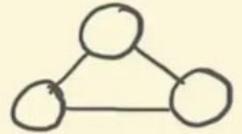


LARGE COLOCATED  
TEAMS

are more likely  
to produce



SOFTWARE



MODULAR, SERVICE  
ARCHITECTURE



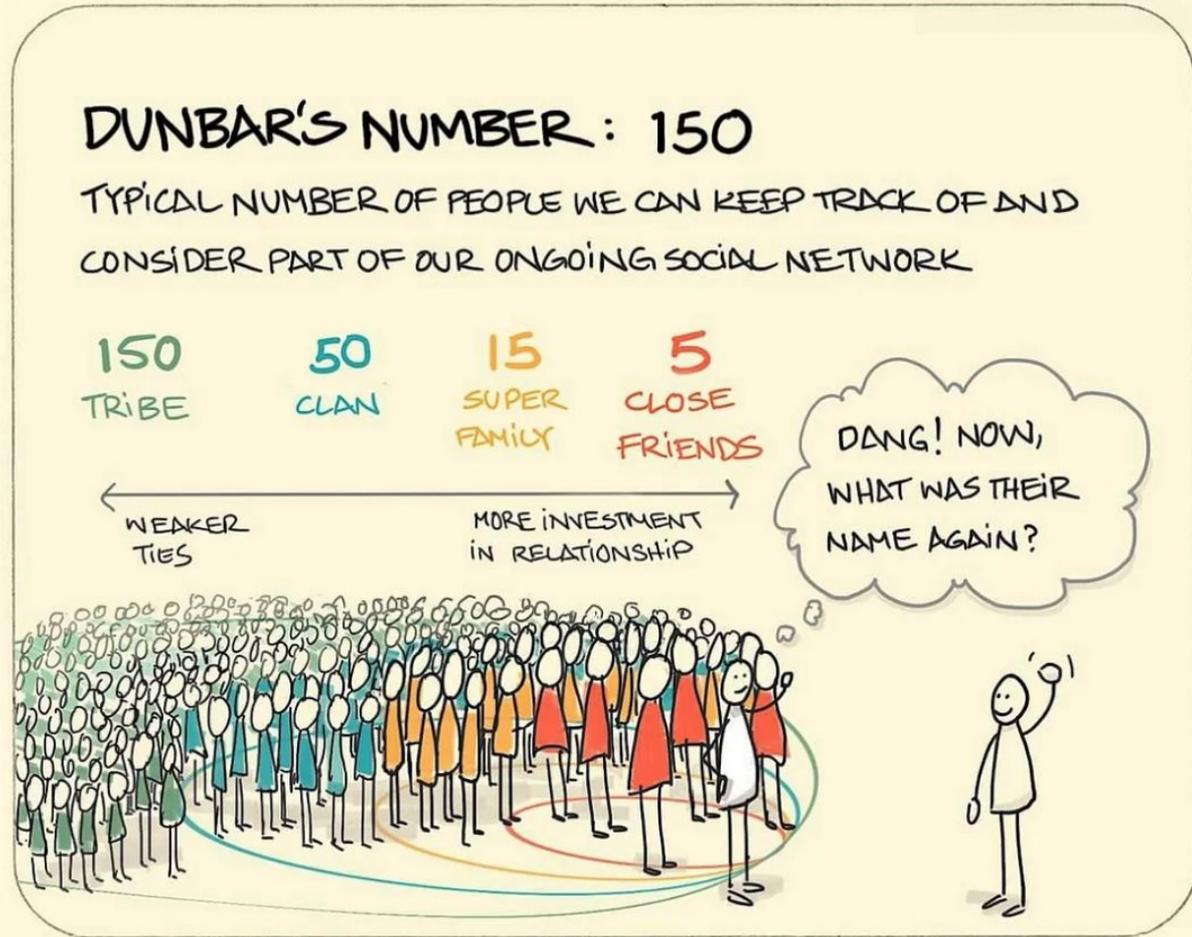
MONOLITHIC  
ARCHITECTURE

# Topologia dos times

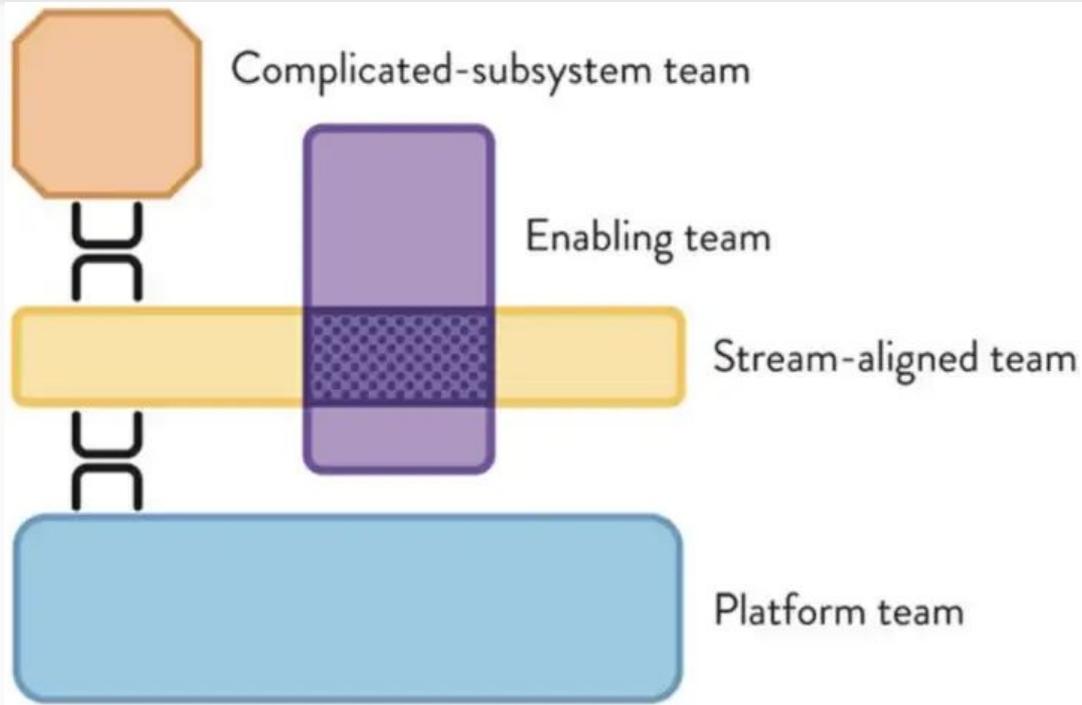


# Tamanho do time

- É padrão da cultura ágil defender times pequenos
- O número de Dunbar representa bem o que estamos buscando, quando falamos de times confiantes.



# Tipos de times





# Stream Aligned Team

(Equipe alinhada ao fluxo)

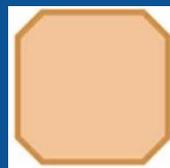
- Se concentram em um fluxo de trabalho único e impactante;
- Tem o poder de criar e entregar valor ao cliente ou ao usuário da forma mais rápida, segura e independente possível;
- As equipes alinhadas ao fluxo trabalham em todo o espectro de entrega, elas estão mais próximas do cliente e, em geral, já são ágeis;
- É o tipo de equipe mais comum nas empresas, a função de outras equipes é definida em relação a ela;
- As equipes alinhadas ao fluxo devem entrar em contato a todo o tempo com as outras equipes para melhorar a velocidade de entrega e a qualidade de seus produtos e serviços sempre.

# Platform Teams

(Equipe de plataforma)



- Permitem que equipes alinhadas ao fluxo entreguem trabalho com autonomia substancial, a equipe de plataforma oferece serviços internos que podem ser usados pelas outras equipes;
- Criam recursos que podem ser usados por várias equipes alinhadas ao fluxo, com pouca sobrecarga;
- É esperado que forneça documentação, suporte e consultoria para equipes alinhadas ao fluxo.



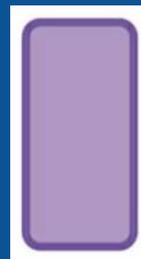
# Complicated subsystem team

(Equipe de Subsistema Complicado)

- Responsável por construir e manter uma parte do sistema que depende de habilidades e conhecimentos específicos
- Os membros da equipe devem ser especialistas em uma área específica do conhecimento para entender e fazer alterações no subsistema
- O objetivo dessa equipe é reduzir a carga de equipes alinhadas ao fluxo que trabalham em sistemas que incluem ou usam o subsistema.
- Com a experiência e os recursos da equipe de subsistemas complicados, as equipes alinhadas ao fluxo não precisam criar recursos em áreas muito complicadas para o trabalho diário.
- Os membros dessa equipe podem ter conhecimento especializado em certos microsserviços (por exemplo, um serviço de faturamento), algoritmos ou até mesmo inteligência artificial.

# Enabling team

(Equipe capacitadora)



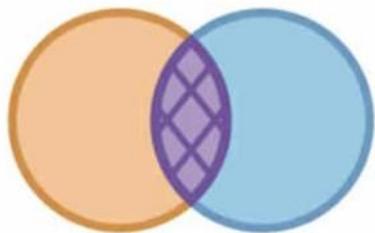
- Composta por especialistas em um determinado domínio técnico (ou produto) que ajuda a preencher essa lacuna de capacidade em equipes alinhadas ao fluxo;
- Se concentram em pesquisa e experimentação para fazer sugestões sobre ferramentas, estruturas e escolhas de ecossistemas que afetam os outros times;
- Busca aumentar a autonomia das equipes alinhadas ao fluxo, aumentando suas capacidades com foco nos problemas, em vez de soluções;
- A equipe capacitadora nunca deve trabalhar em uma dependência permanente.

# Modos de comunicação



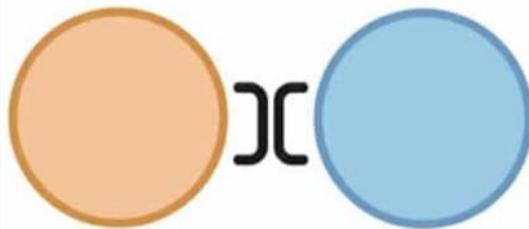
“Nem toda comunicação é positiva, pois aumenta a carga cognitiva dos times. Por isso é necessário mapear e compreender como topologias tão diferentes podem interagir na estrutura.”

# São previstos 3 modos de comunicação



Collaboration

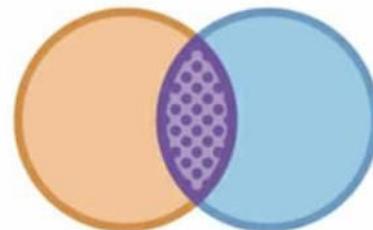
Os times trabalham juntos, exigindo adaptabilidade e alinhamento.



X-as-a-Service

Um time provê alguma coisa para outro time, com o mínimo de colaboração.

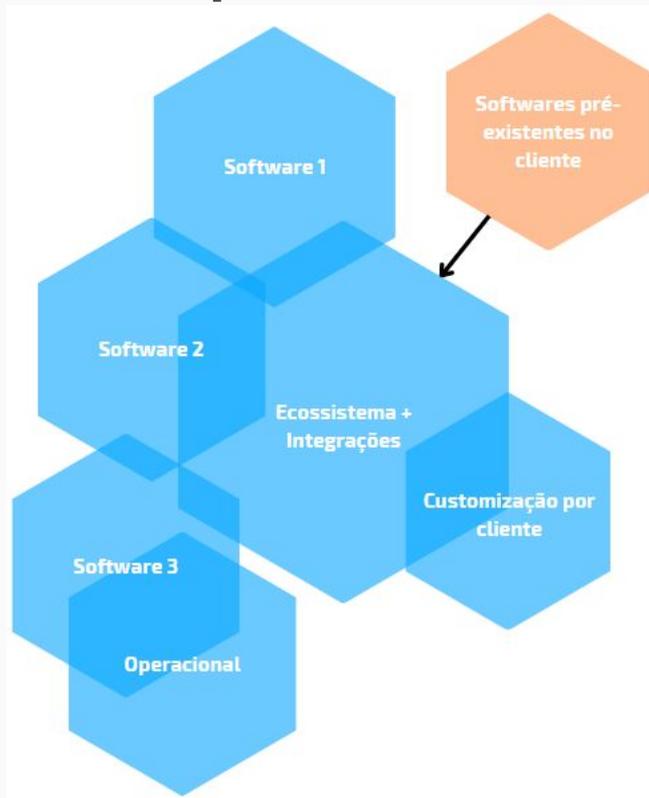
Por exemplo, times que desenvolvem APIs para serem utilizadas por outros times.



Facilitating

Um time ajuda ou pede ajuda para outro time, eliminando impedimentos, principalmente quando é necessário um trabalho ativo para a evolução do trabalho ou para a execução de melhorias.

# Case: Reestruturando o passado



- Ecossistema com 3 grandes softwares
- Necessidade de integração entre os 3 softwares
- Software operacional integrado a 1 dos softwares
- Algumas regras eram implementadas a cada novo cliente baseadas em legislação local
- Os clientes já possuíam softwares
- Implantação nos clientes vinham prazos fechados
- Todos os times se envolviam na implantação

Como eu estruturaria essa operação utilizando a Topologia de Times?

# Stream Aligned Teams

- Geralmente, essas equipes são formadas para cada implantação
- Focadas nas entregas de valor de negócio
- Diretamente impactadas pelas decisões dos contratos de vendas

Entregas de valor

Customização

Legislação exclusiva

Operacional

# Platform Teams

- Times o mais consistentes possíveis
- Responsáveis pela estabilidade das plataformas
- Responsáveis pelas decisões de novas features das dos softwares
- Também responsáveis por MVP's e POC's

Software 1

Software 2

Software 3

# Enabling teams

- Times com profissionais versáteis
- Capaz de se adaptar rapidamente a diferente contextos
- Agrega e influencia as atividades dos times com os quais se relacionam

Integrações

Implantação

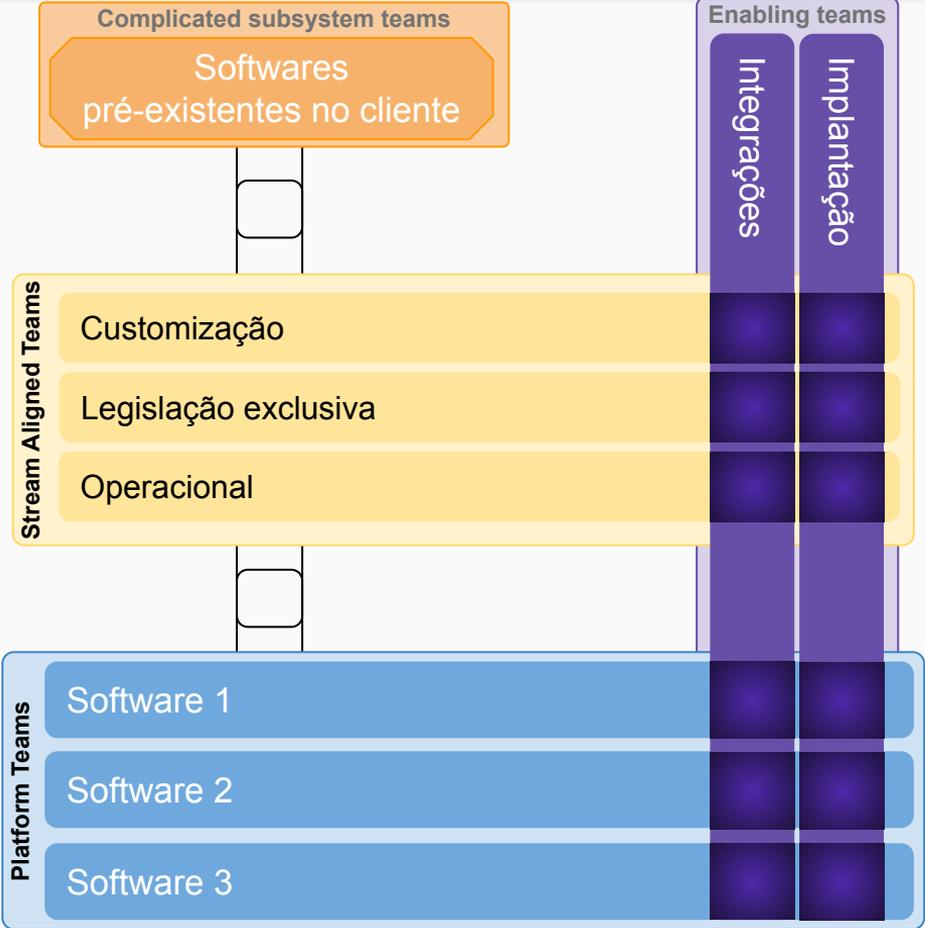
# Complicated subsystem teams

- Equipes existentes na estrutura dos clientes
- Precisam estar motivadas a entregar o que os times alinhados ao fluxo solicitam



Softwares pré-existentes

# Estrutura Proposta



# Obrigado!



[linktr.ee/focezar](https://linktr.ee/focezar)