A photograph of three professionals in a construction or engineering setting. In the foreground, a man wearing a yellow hard hat and a white shirt is looking down at a large sheet of architectural blueprints spread out on a table. Behind him, another man in a yellow hard hat and a woman in a red hard hat are also examining the plans. They are all dressed in professional attire like shirts and ties. The background is slightly blurred, showing more of the office environment.

THE DOMAIN-DRIVEN CODING ARCHITECT

NICK TUNE
@NICKCODING

DEVELOPERS ARE EVOLVING!



“

Consistently the best source of new ideas are the developers!...

Good teams ensure their engineers contribute to make the product better

– Marty Cagan (@cagan)

”



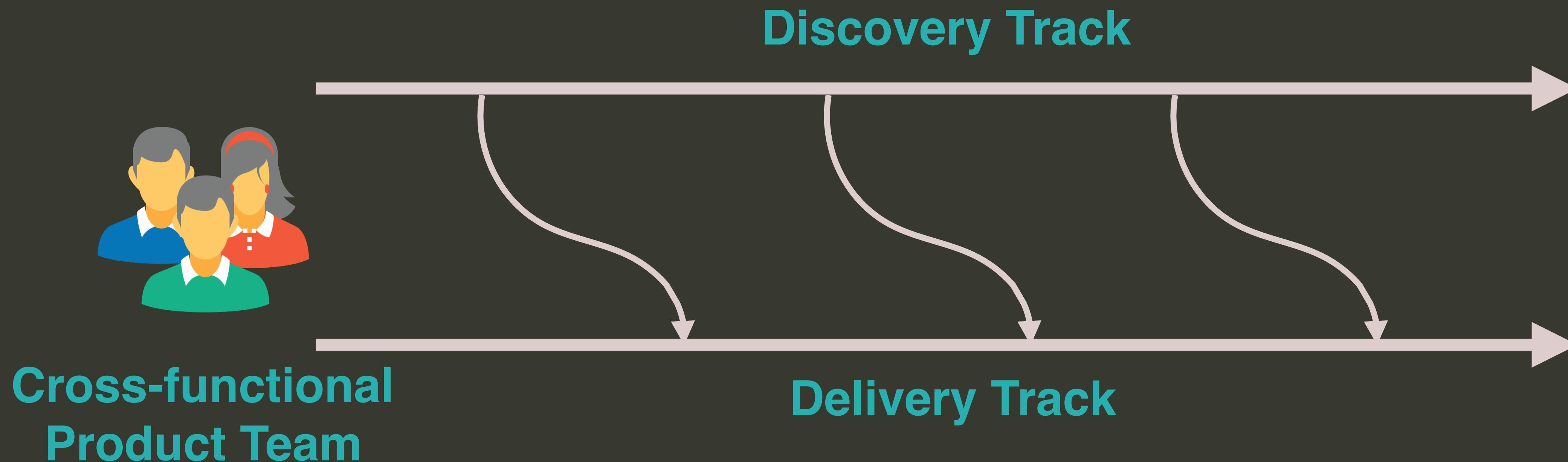
Nick
@ntcoding

Flying cars business model 😂😂😂

#CraftConf



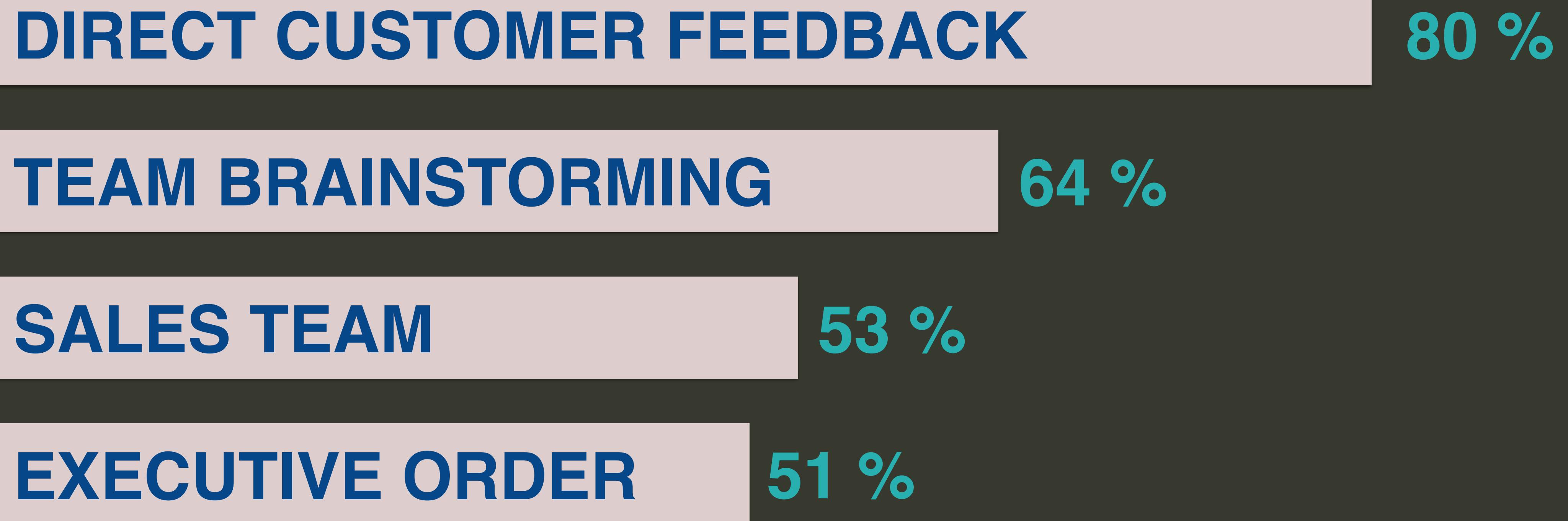
CONTINUOUS DISCOVERY & DELIVERY



The whole team get involved in research sessions and meeting users so they can empathise with people who use the services we're building

– Simon Hurst
User Researcher at Coop

TOP SOURCES OF PRODUCT IDEAS



* % respondents who agreed.

Source: Alpha UX Product Management Insights 2017

“
[company] regularly
send their developers to
work on farms with
their customers
[farmers]”

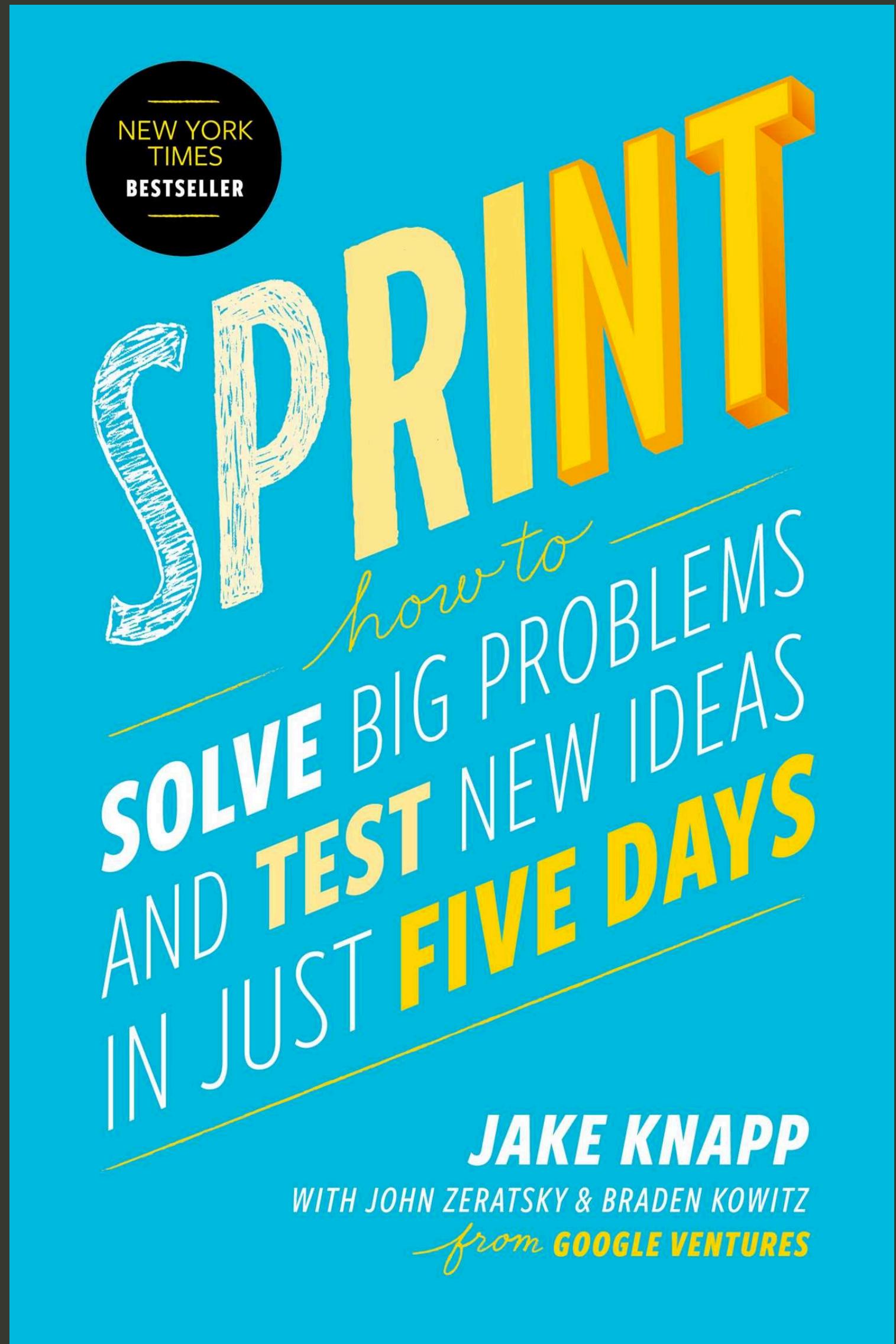


Melissa Perri
CEO Produx Labs
@lissijean





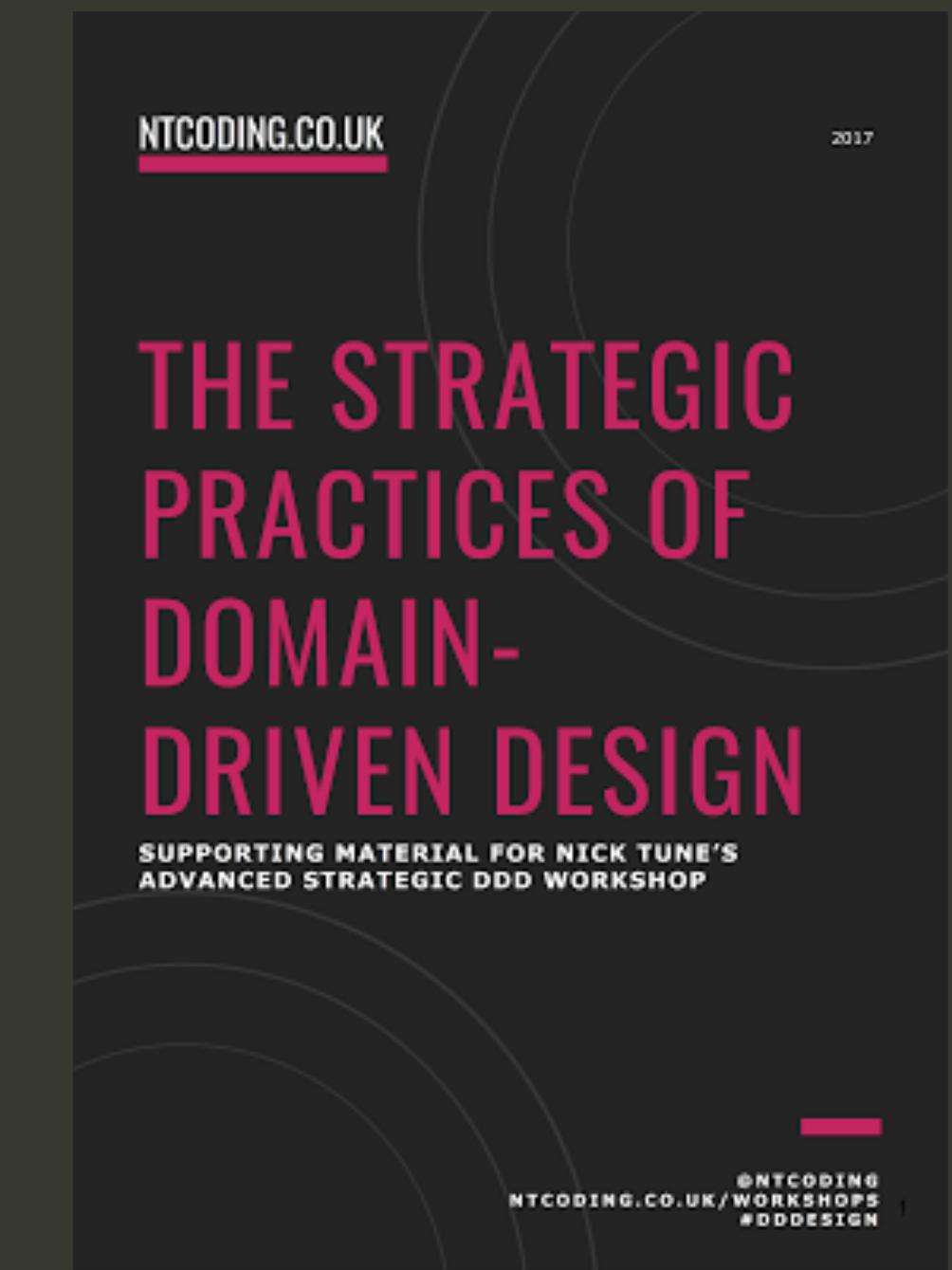
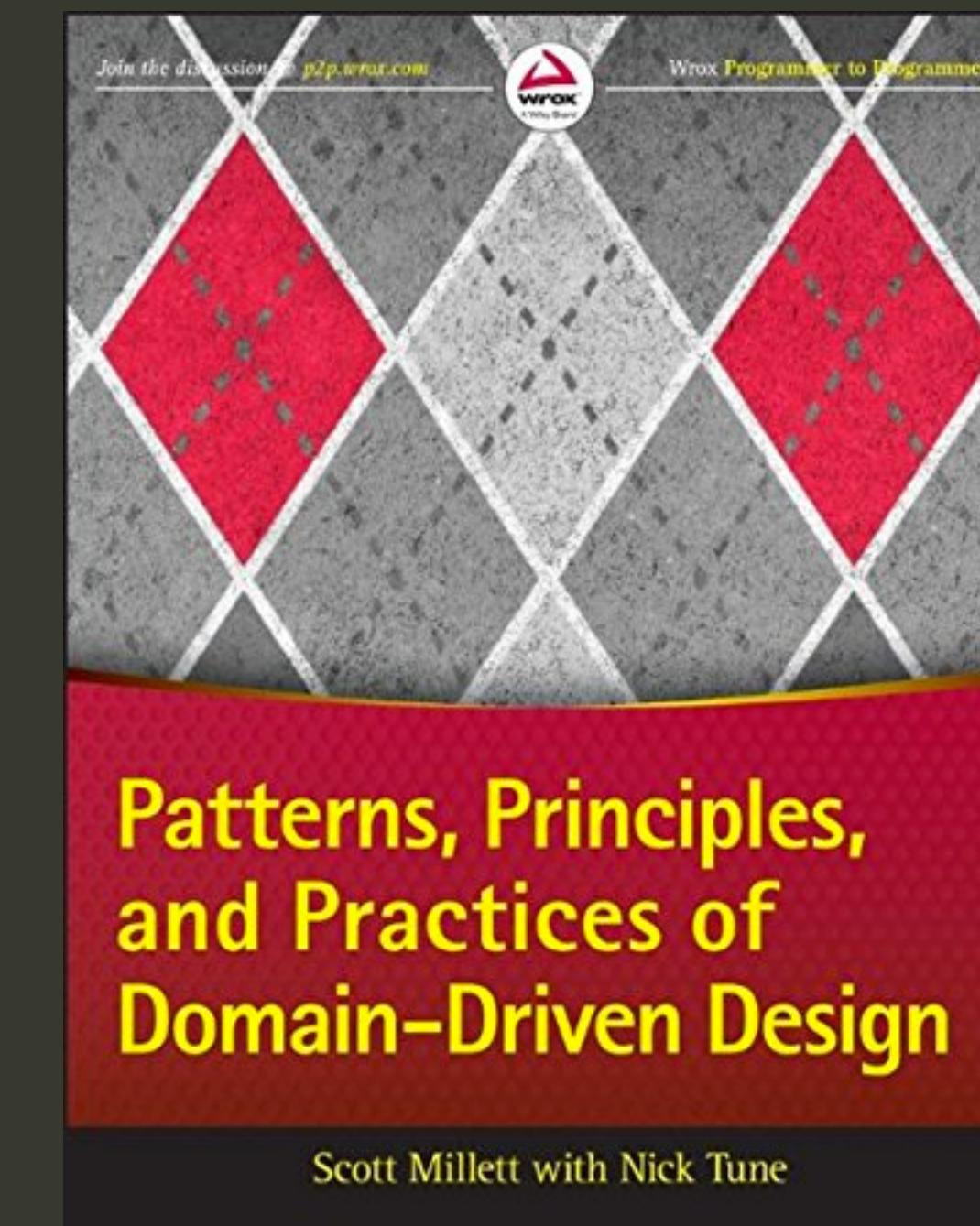
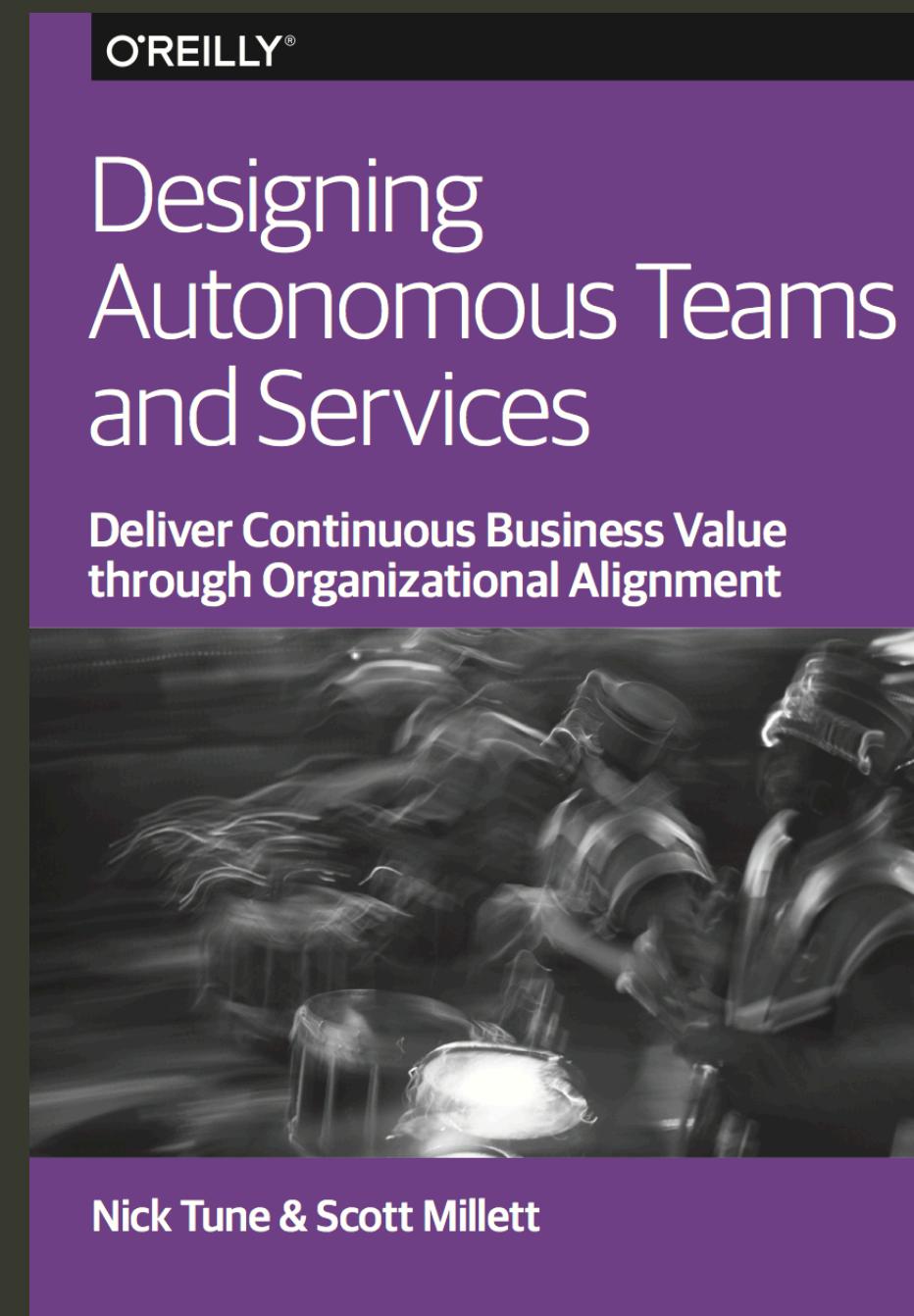




EGO SLIDE



Nick Tune
Principal Engineer,
Salesforce





How can a new perspective of DDD help us **architect** for continuous discovery and delivery?

#1

LEARNING TO FOCUS ON THE CORE DOMAIN

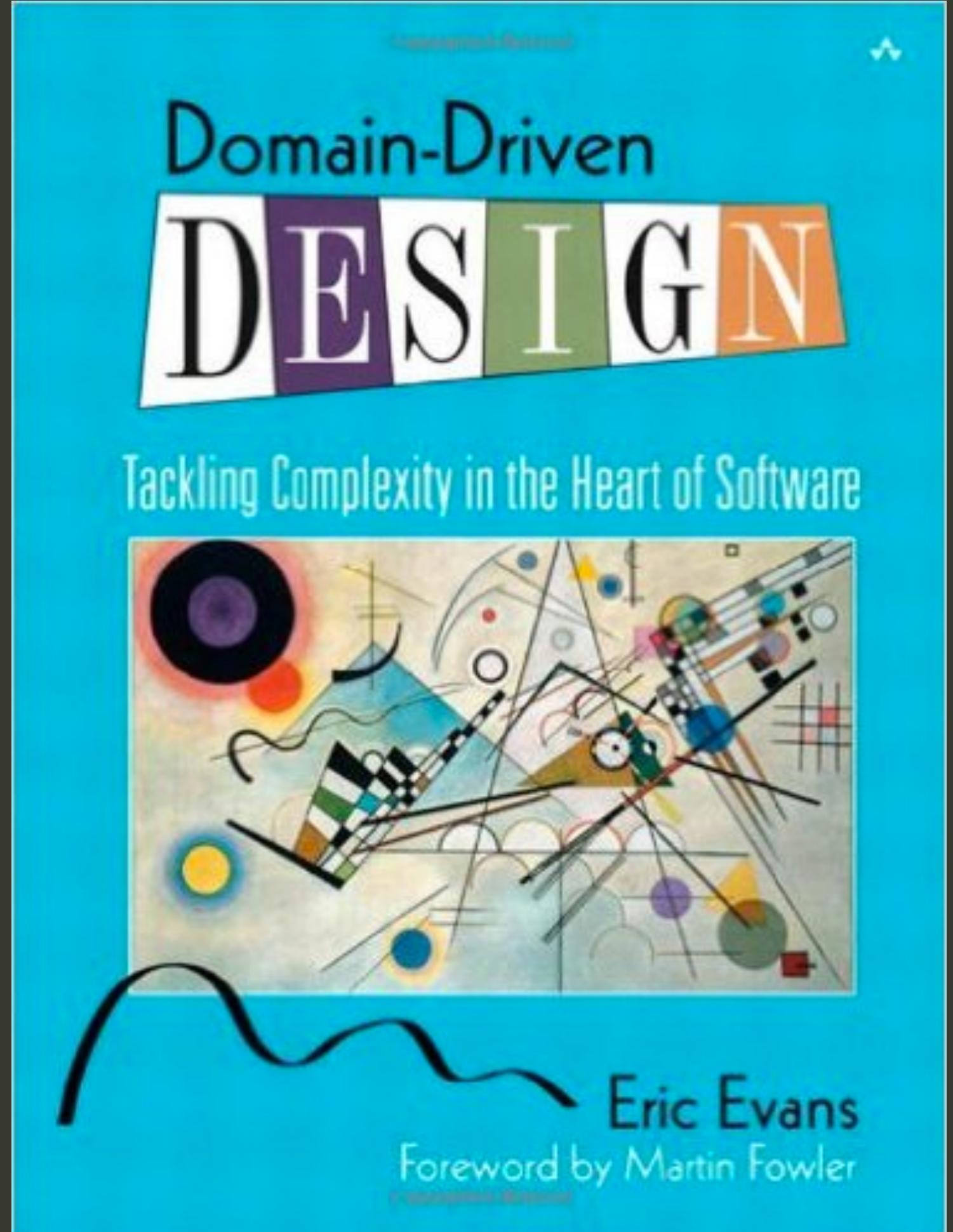


**I've been on a
DDD course!
So I am right and they are
wrong!**

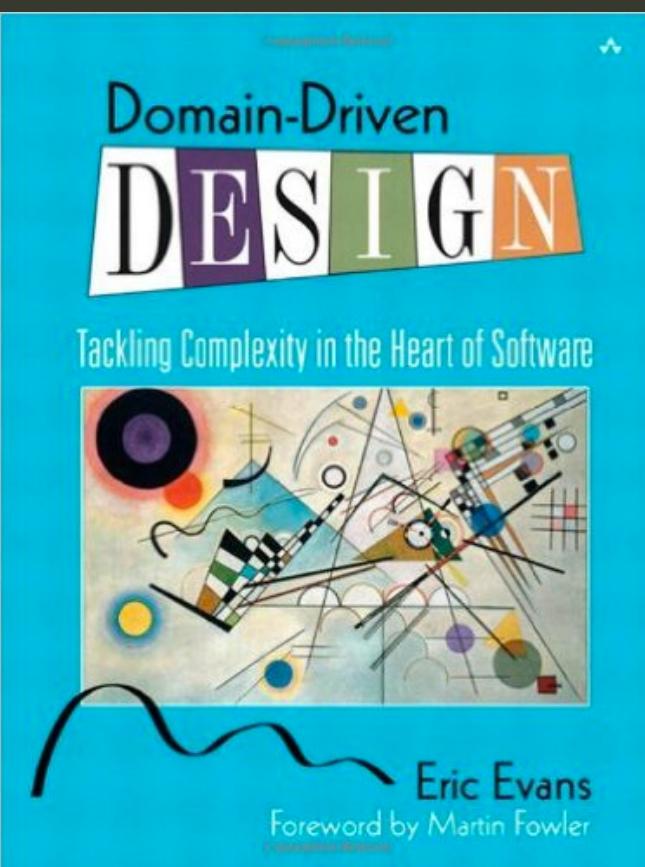
“

*The CORE DOMAIN is
where the most value
should be added in your
system.*

”



“
[non-core] parts of the system seem interesting to computer scientists, and provide better resume material. The core ends up being put together by less skilled developers



– Eric Evans (@ericevans0)

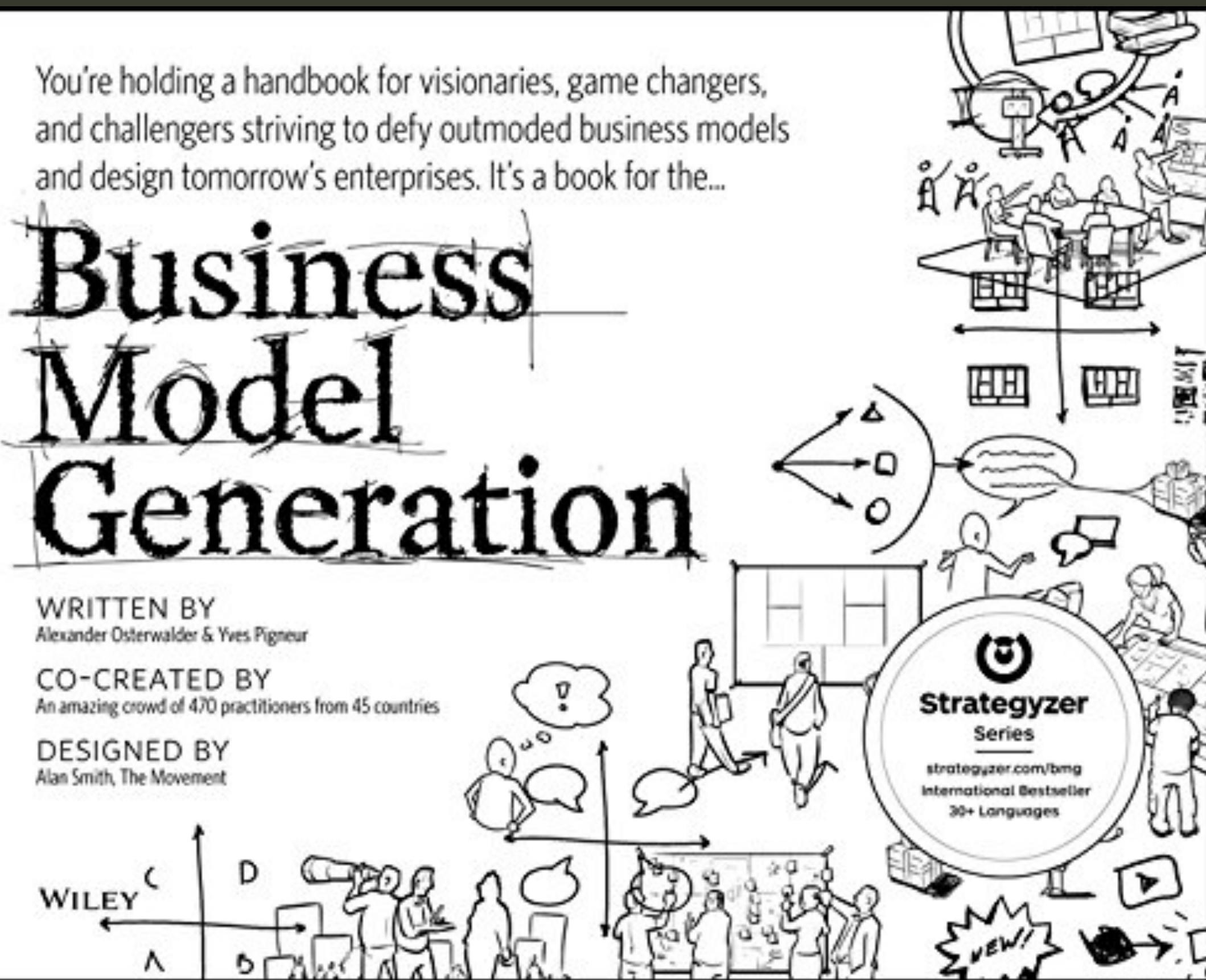
You're holding a handbook for visionaries, game changers, and challengers striving to defy outmoded business models and design tomorrow's enterprises. It's a book for the...

Business Model Generation

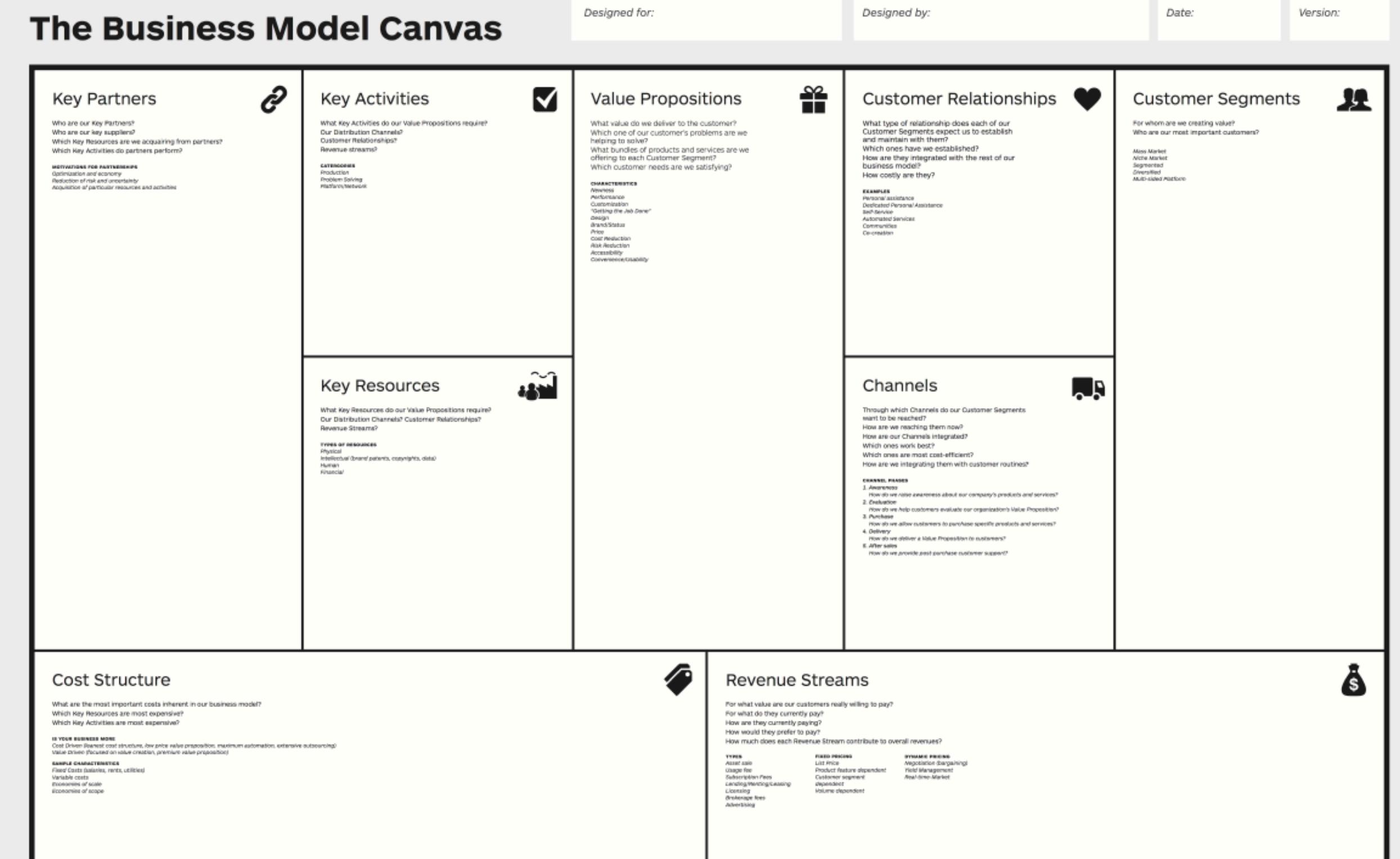
WRITTEN BY
Alexander Osterwalder & Yves Pigneur

CO-CREATED BY
An amazing crowd of 470 practitioners from 45 countries

DESIGNED BY
Alan Smith, The Movement



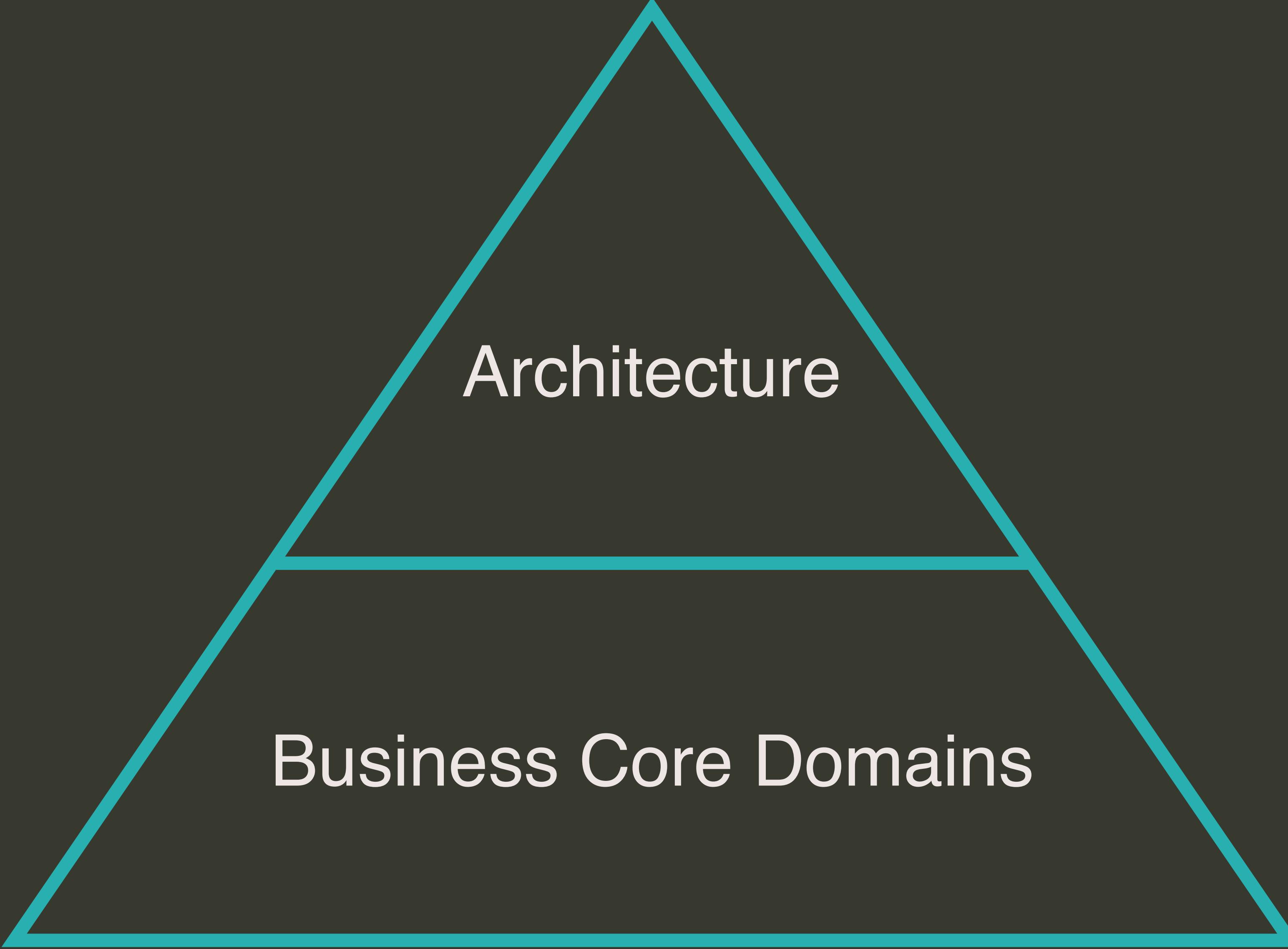
The Business Model Canvas



strategyzer
strategyzer.com

LESSONS LEARNED

- DDD: focus architecture on business core, not just technical patterns
- Need to understand business to find core
- Learn concepts like Business Model Canvas to understand business



Architecture



Business Core Domains

WHAT IS CORE CAN CHANGE...



Slack started life as an internal chat system.
It is now the core domain worth \$4bn.

#2

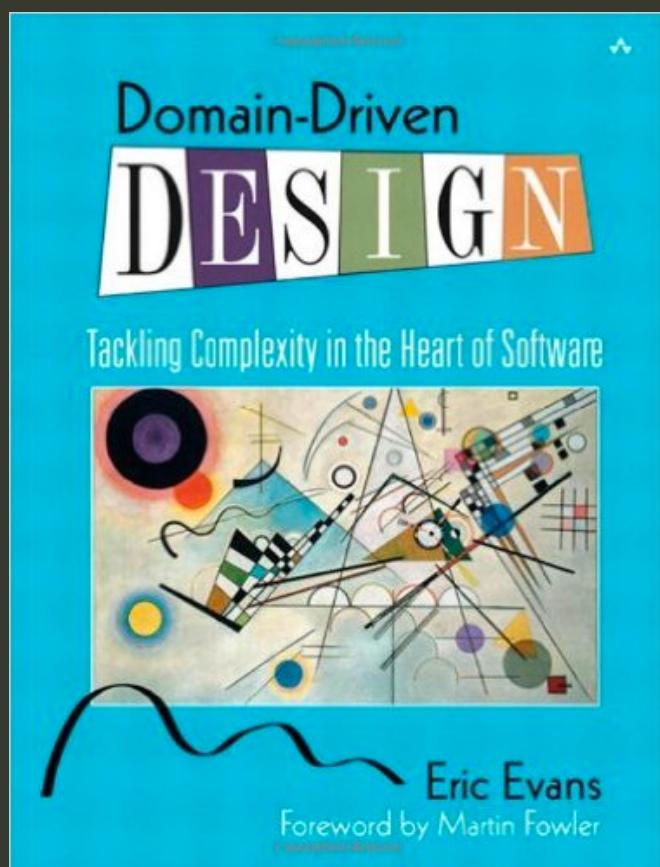
DISCOVERING THE DOMAIN VISION



“

...teams write “vision statements” for management. These documents lay out the specific value the application will bring to the organization.

”



– Eric Evans (@ericevans0)

VISION STATEMENT

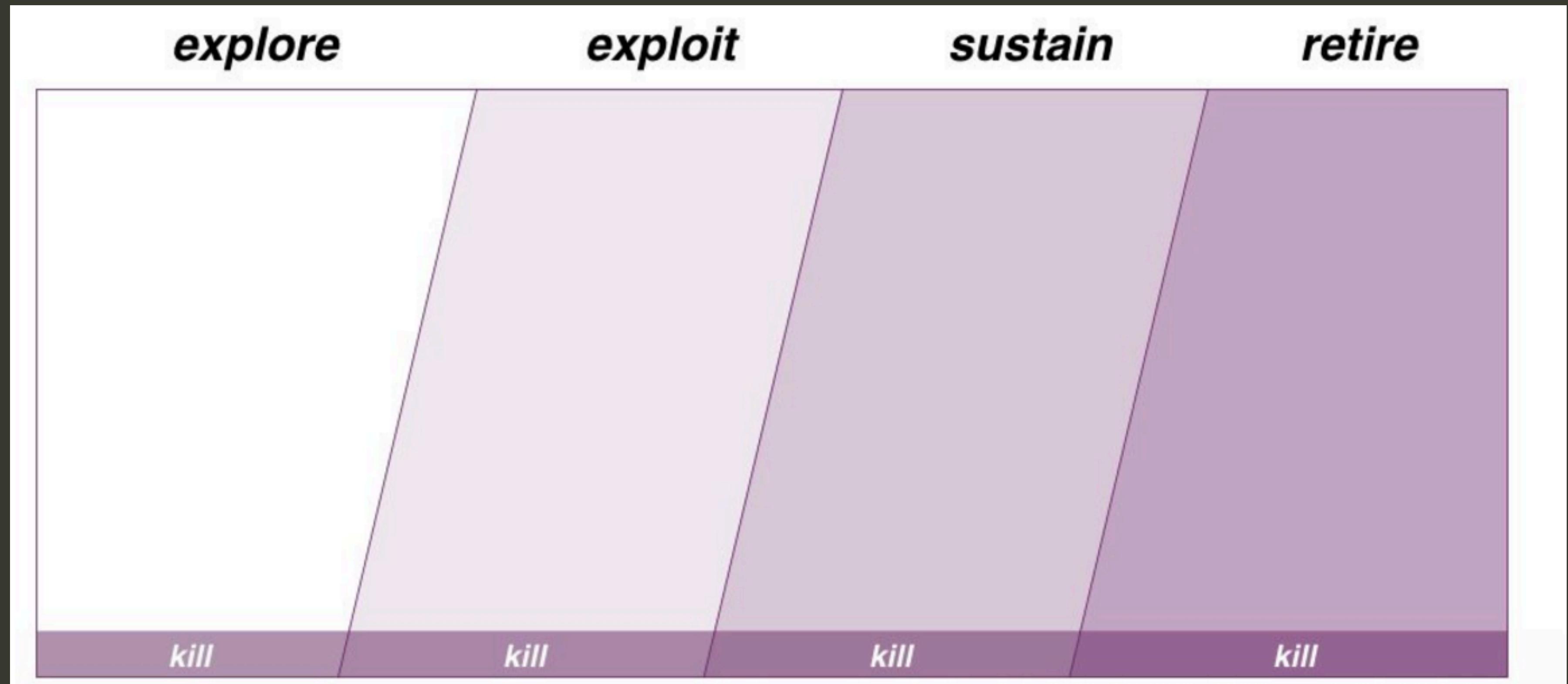
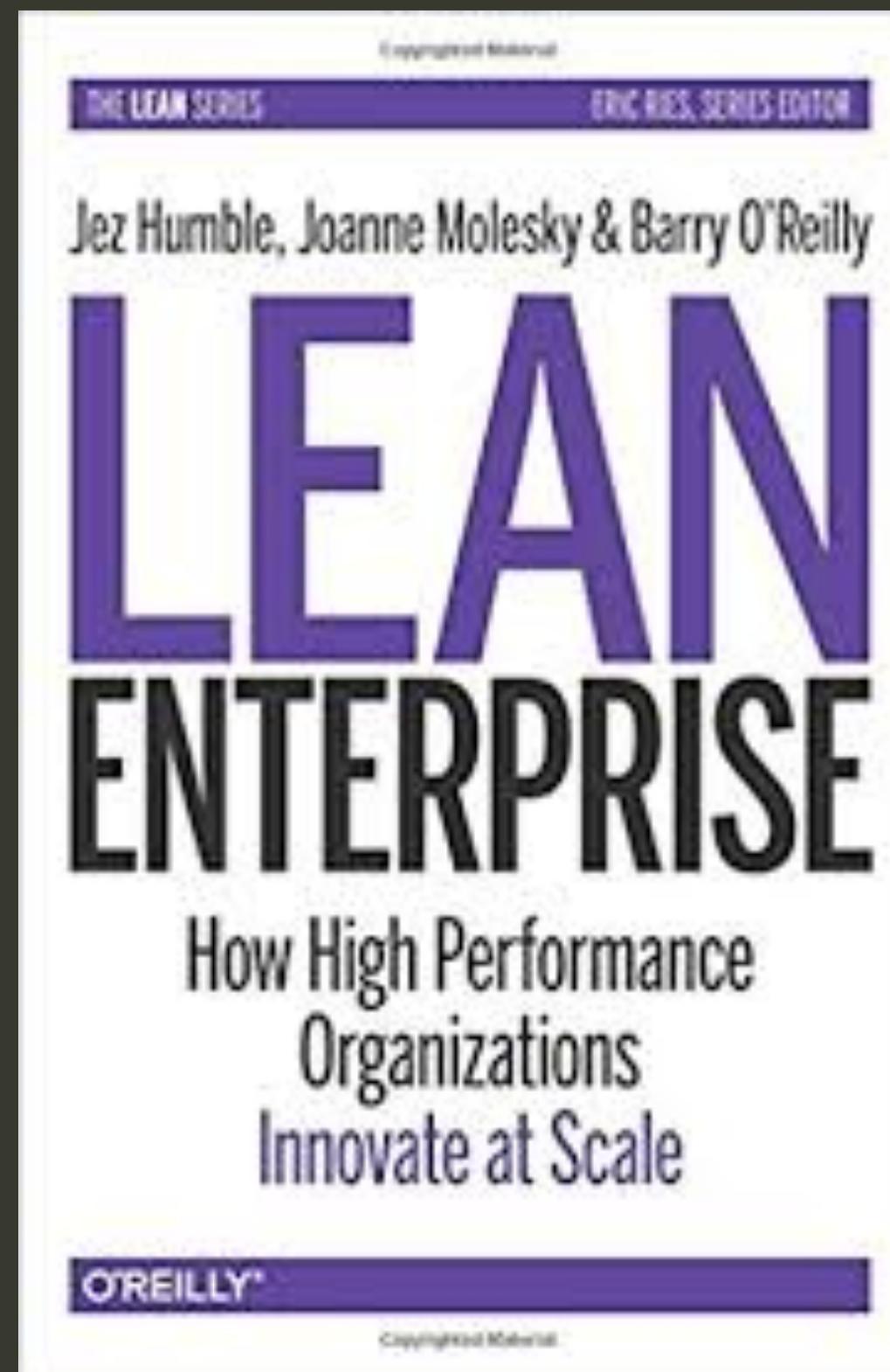
We are a music company.

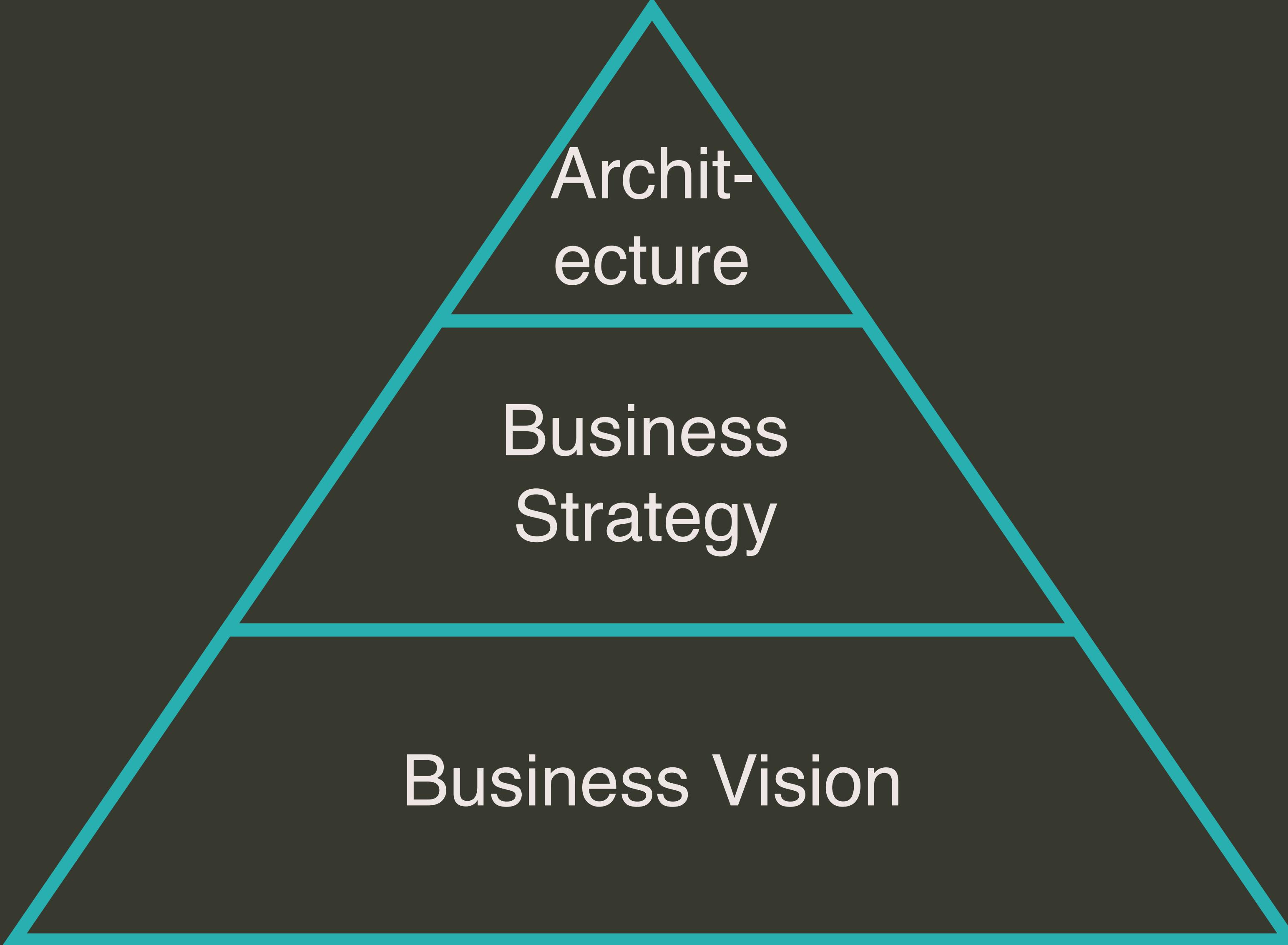
*We help the planet's greatest artists
and biggest hidden talents share their
amazing music with the entire world.*

LESSONS LEARNED

- Business vision guides the strategy
- Vision empowers teams to keep architecture aligned to business goals
- Vision is communicated as some form of vision statement

MAP YOUR INNOVATION PORTFOLIO





Architecture

Business
Strategy

Business Vision

#3

DOMAIN-DRIVEN ORGANISATION DESIGN

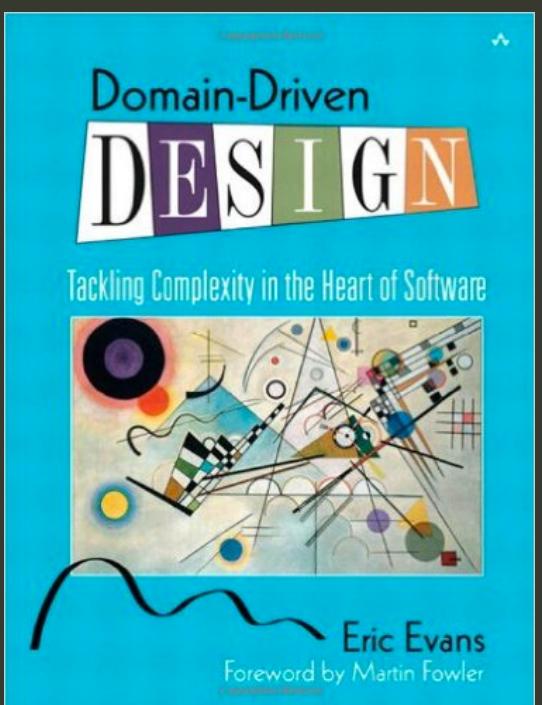


**So many
meetings!**

Waaaaaaa!

“*[bounded contexts] must guide design decisions to reduce the interdependence of parts [code & teams].*

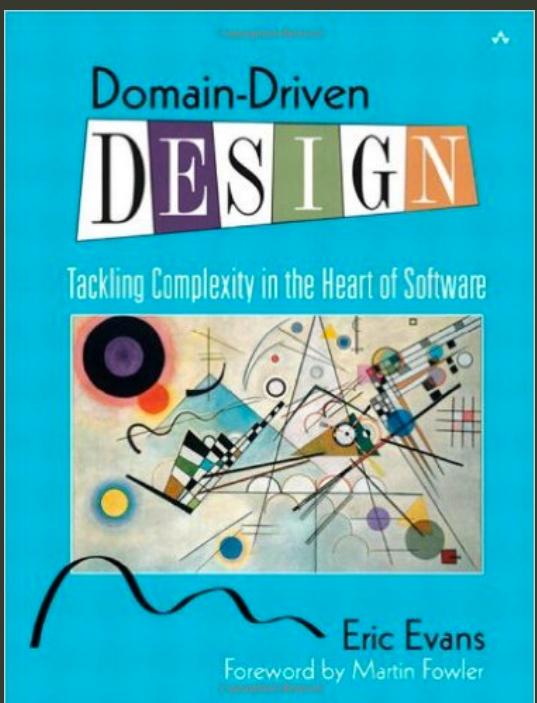
...these two groups had different models, but they did not realize it



– Eric Evans (@ericevans0)

“

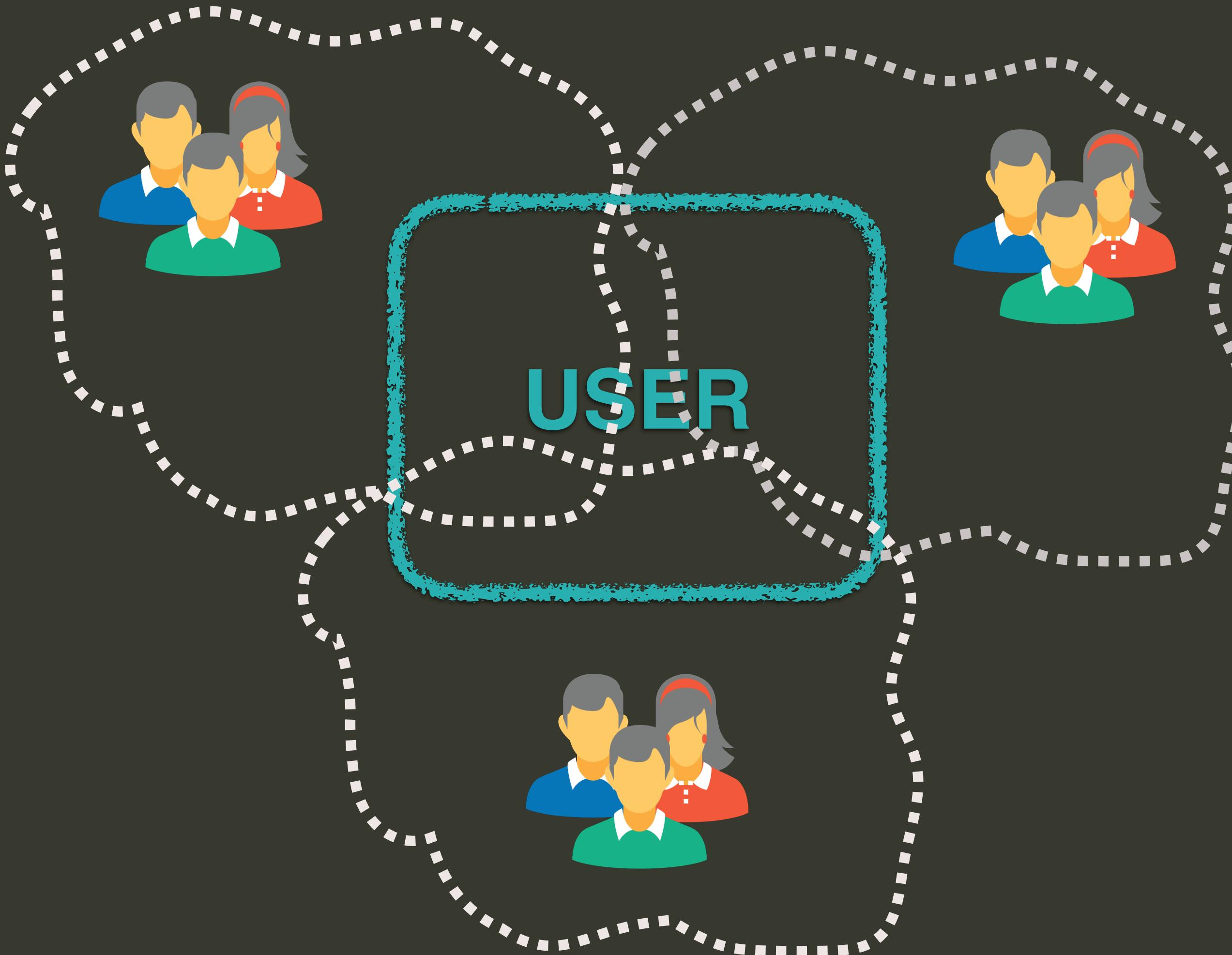
The CONTEXT MAP charts the territory, giving the big picture of the CONTEXTS and their connections...



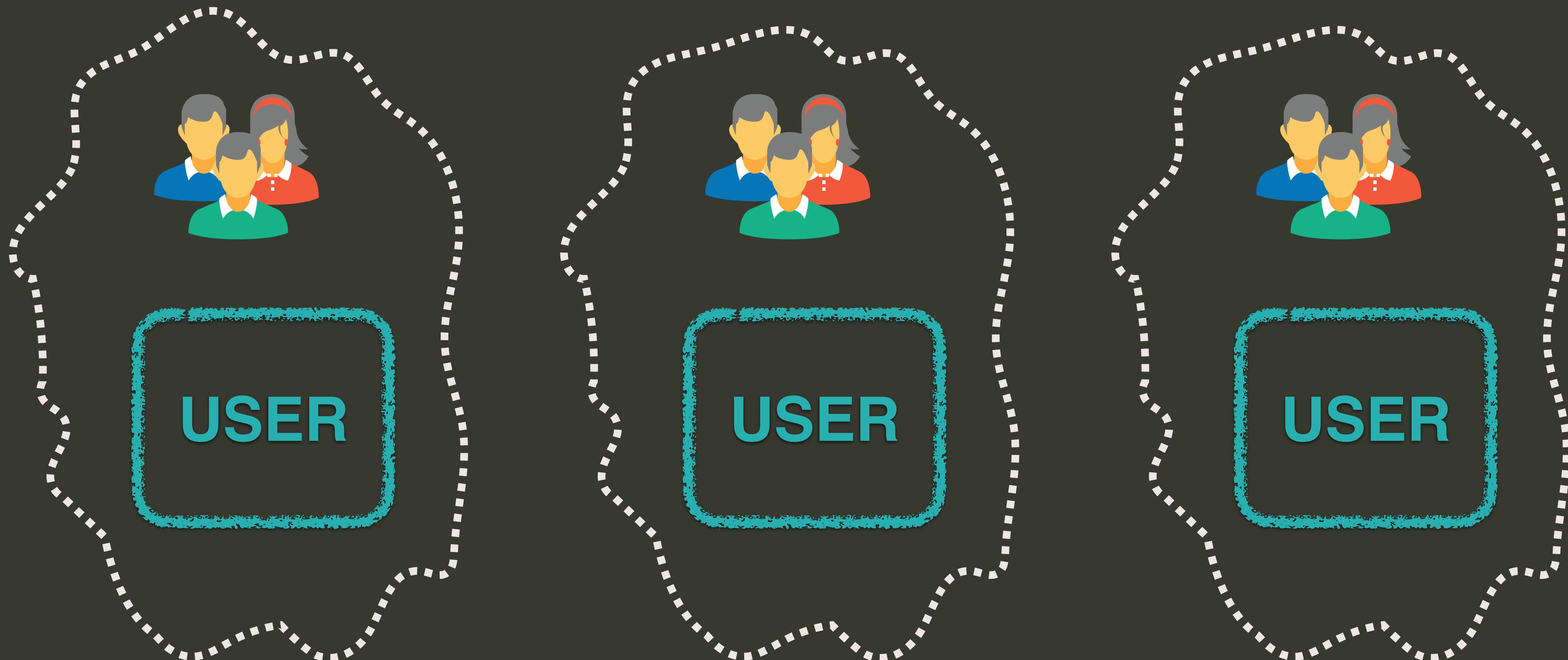
— Eric Evans (@ericevans0)

”

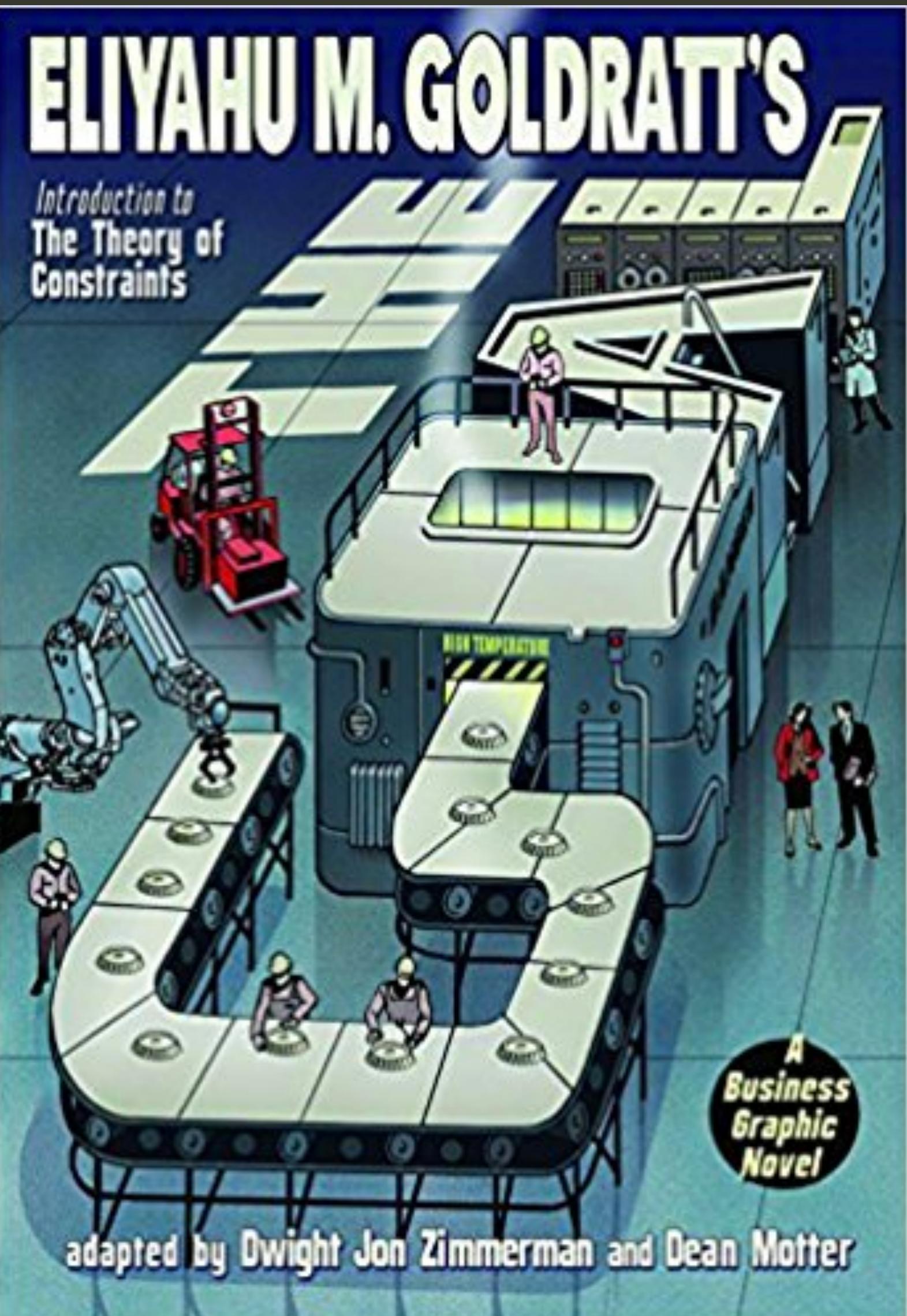
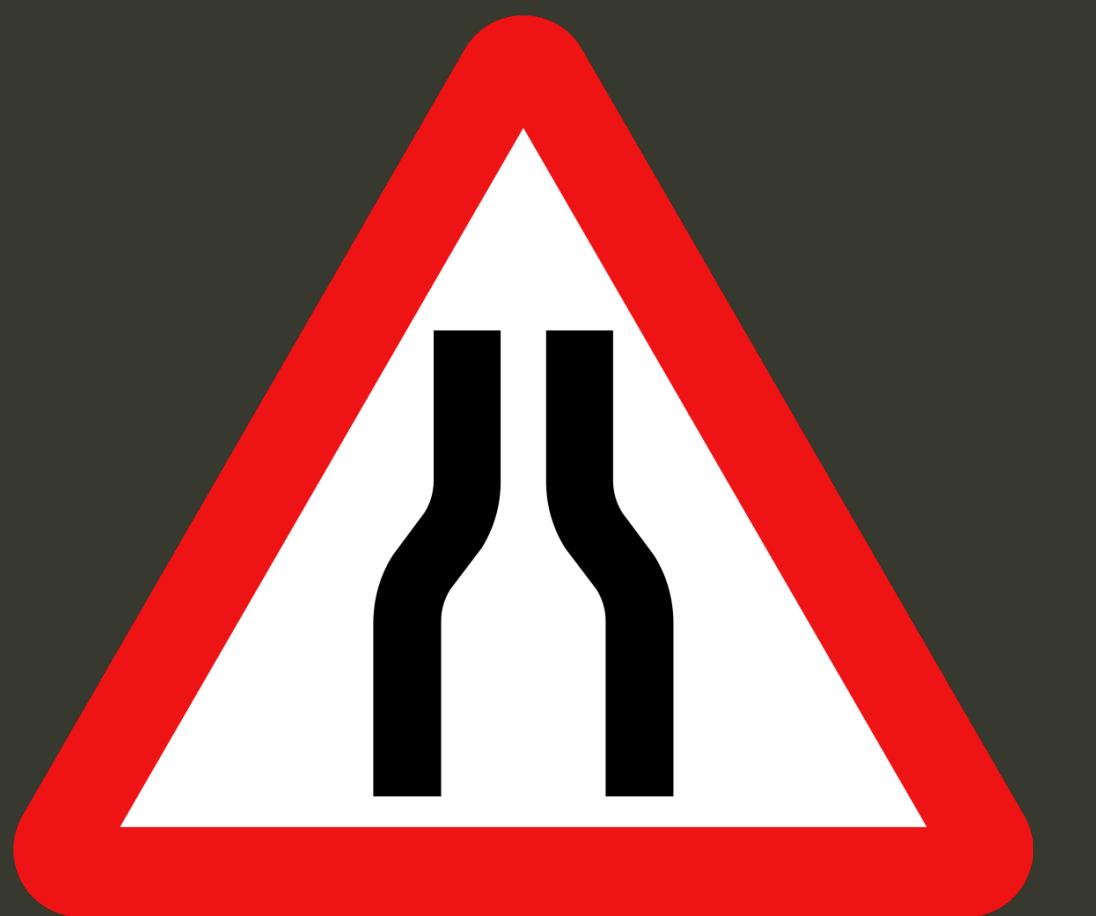
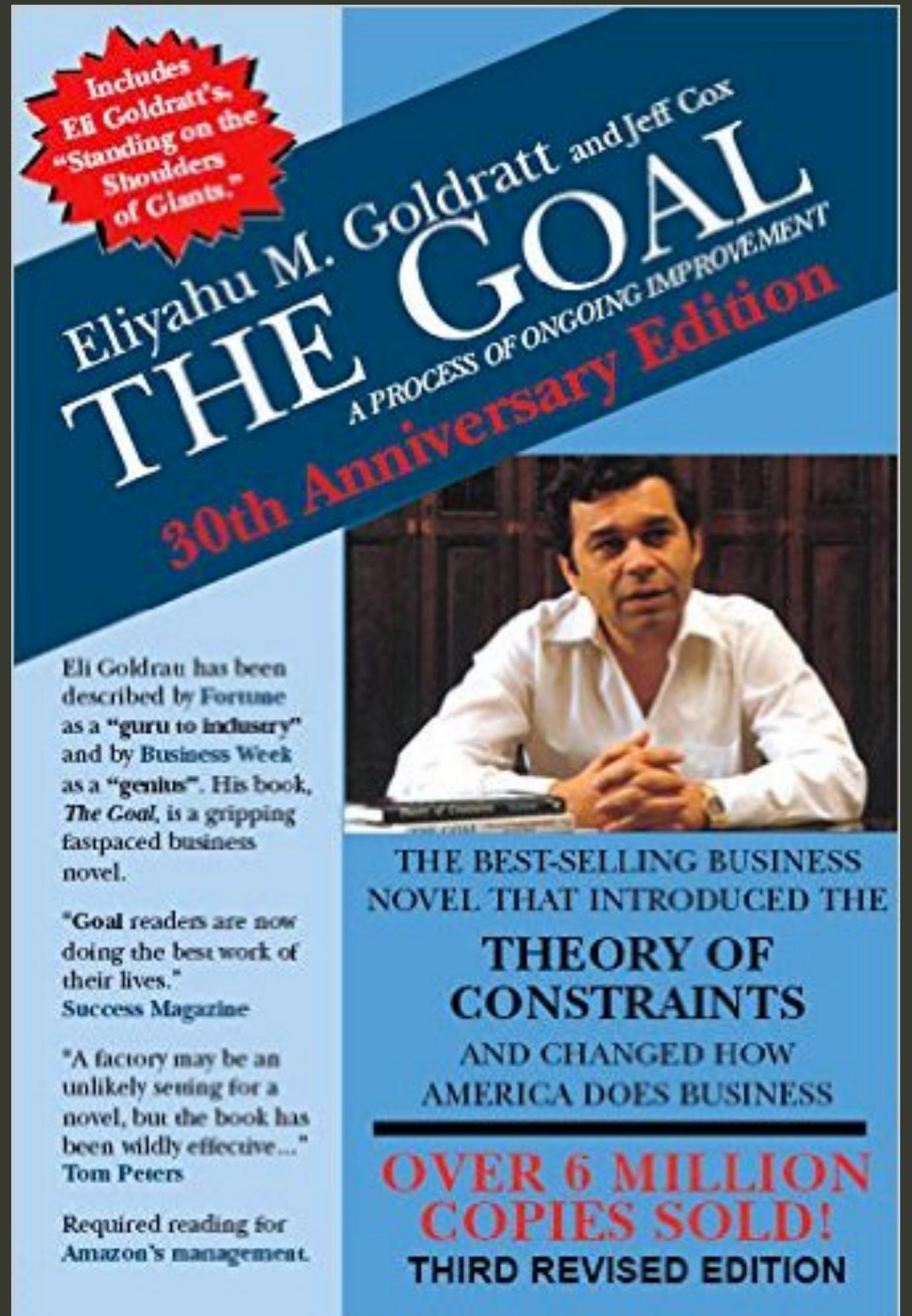
MULTIPLE HIDDEN CONTEXTS



MULTIPLE BOUNDED CONTEXTS







LESSONS LEARNED

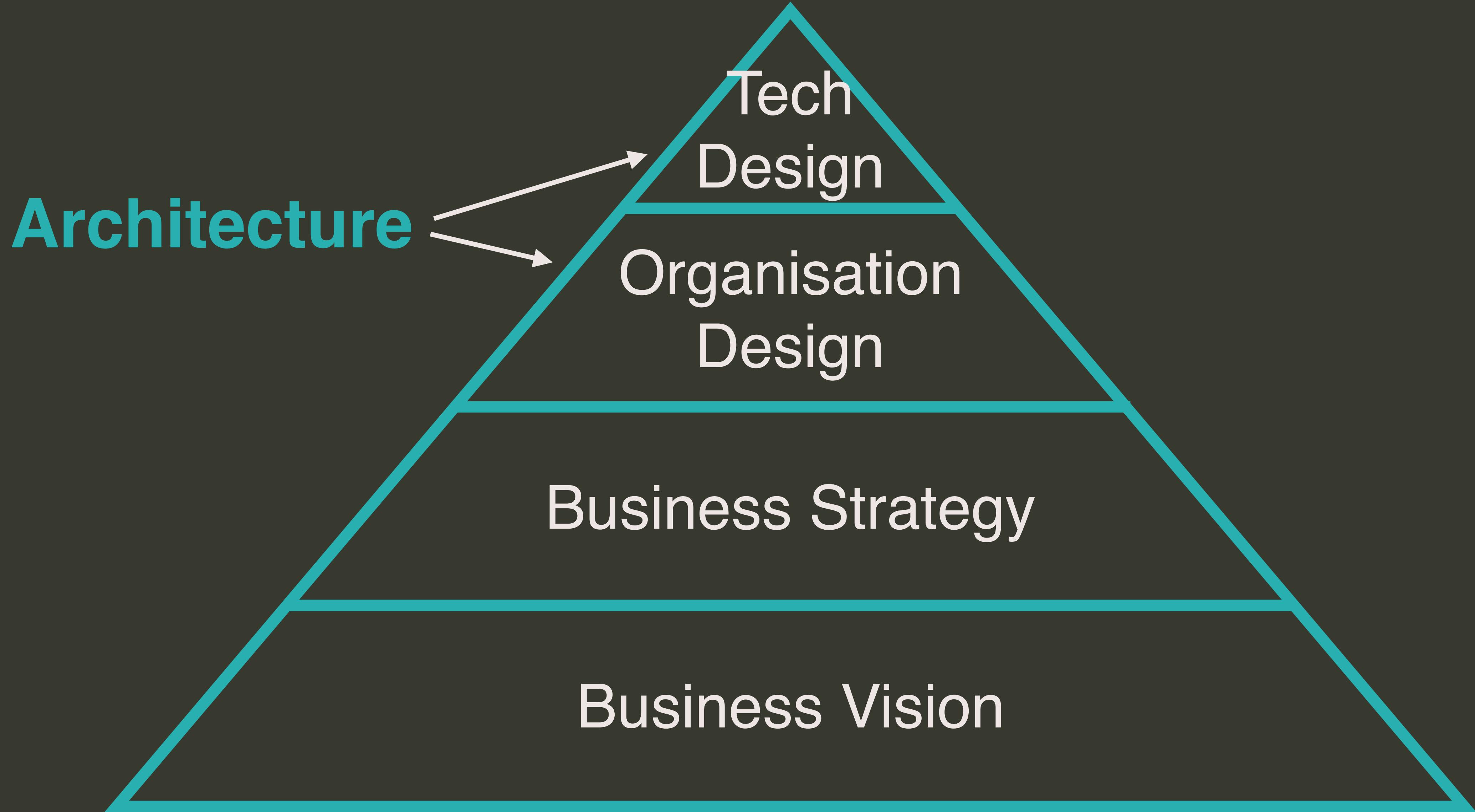
- Bounded contexts decouple **PARTS**
- Parts = teams & code
- Find contexts that minimise bottlenecks to make teams autonomous
- Use context maps to design & communicate boundaries



Software architecture

Sociotechnical Systems

Architecture



#4

ARCHITECTING FOR CONTINUOUS DISCOVERY & DELIVERY

BECOME A STRATEGIC DDD ARCHITECT

- Find and exploit core domains
 - Learn Business Model Canvas
- Identify innovation aligned with vision
 - Find the vision statement
- Architecture = code & organisation design
 - Align teams with *real* bounded contexts

YOU NOW HAVE SUPERPOWERS



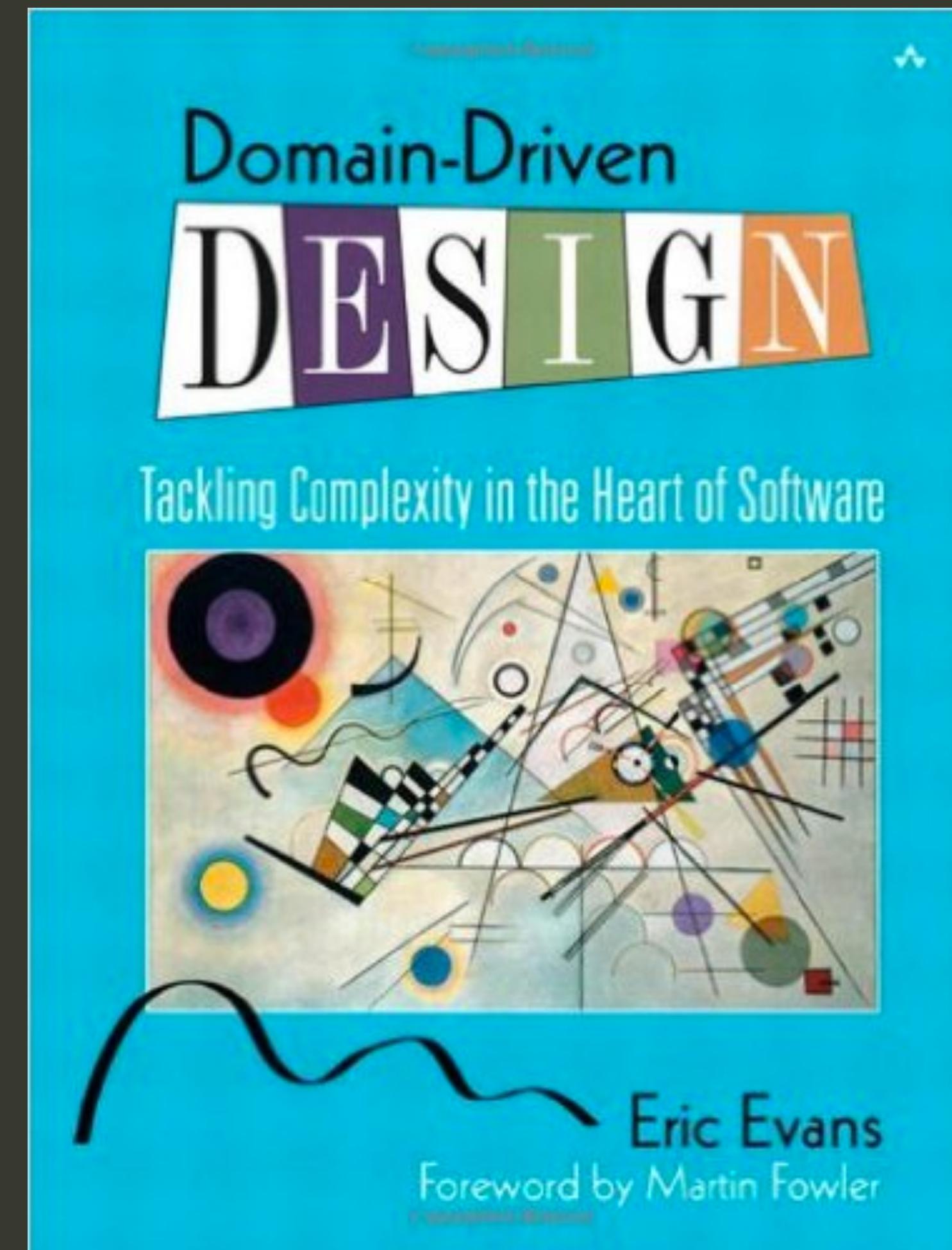
Black Ops DDD with the Business Model Canvas



Javier Fernandez

Watch: bit.ly/alignment-experiment

READ IT PROPERLY THIS TIME!



MORE...

ntcoding.co.uk/workshops

ntcoding.co.uk/blog



@*ntcoding*



/in/*ntcoding*

O'REILLY®

Designing
Autonomous Teams
and Services

Deliver Continuous Business Value
through Organizational Alignment

Nick Tune & Scott Millett

