

Technology Day 2025

Komplexität und Kompliziertheit in Software

INNOQ



FABIAN WALTHER
INNOQ.SOCIAL/@FABIAN

Disclaimer

Kontext

- Fokus auf Geschäftsprozessanwendungen in Java
- Keine Gameengines, Kryptographie-Bibliotheken, Embedded-Systeme
- Unterschiedliche Settings von 1 bis N Entwicklungsteam(s)
- Java / JVM -> Objektorientierung, Architektur Clean-, Onion- und Konsorten, DDD taktisch und strategisch
- Von der Arbeit mit Code zur Arbeit mit Menschen

Motivation

Domain-Driven Design und die vielen Missverständnisse

„Der Code ist zu kompliziert! Das geht doch einfacher!“

„Das System ist viel zu komplex!“

Was denn nun???

Steile These:

**Unsere Software muss kompliziert(er)
werden, um Komplexität zu bändigen**

Was is Komplexität?

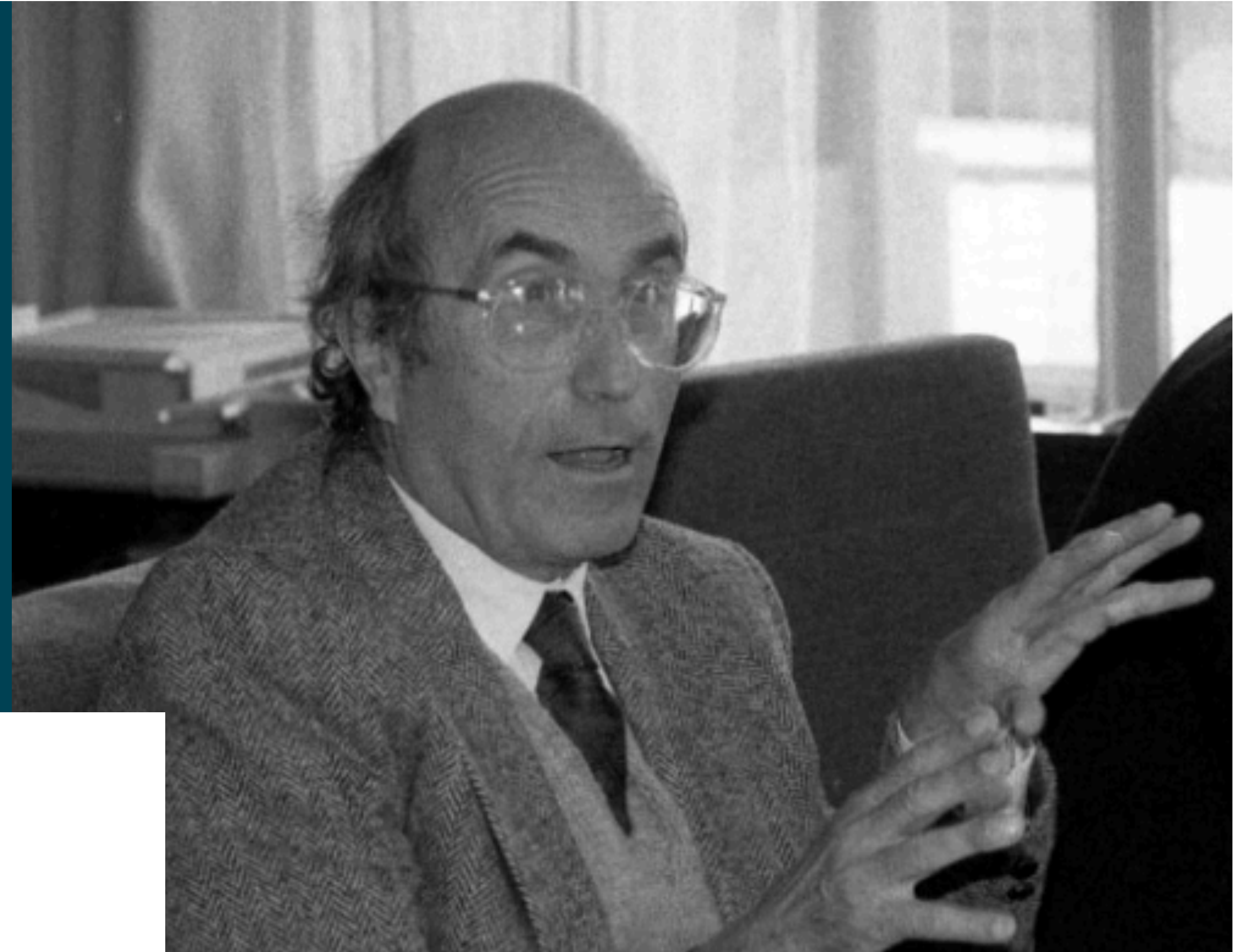
Ein paar Schritte zurück ...

Was ist ein System?

**Wir bewegen uns in der Sphäre
Soziotechnischer Systeme**

**Die Welt ist
unendlich komplex...**

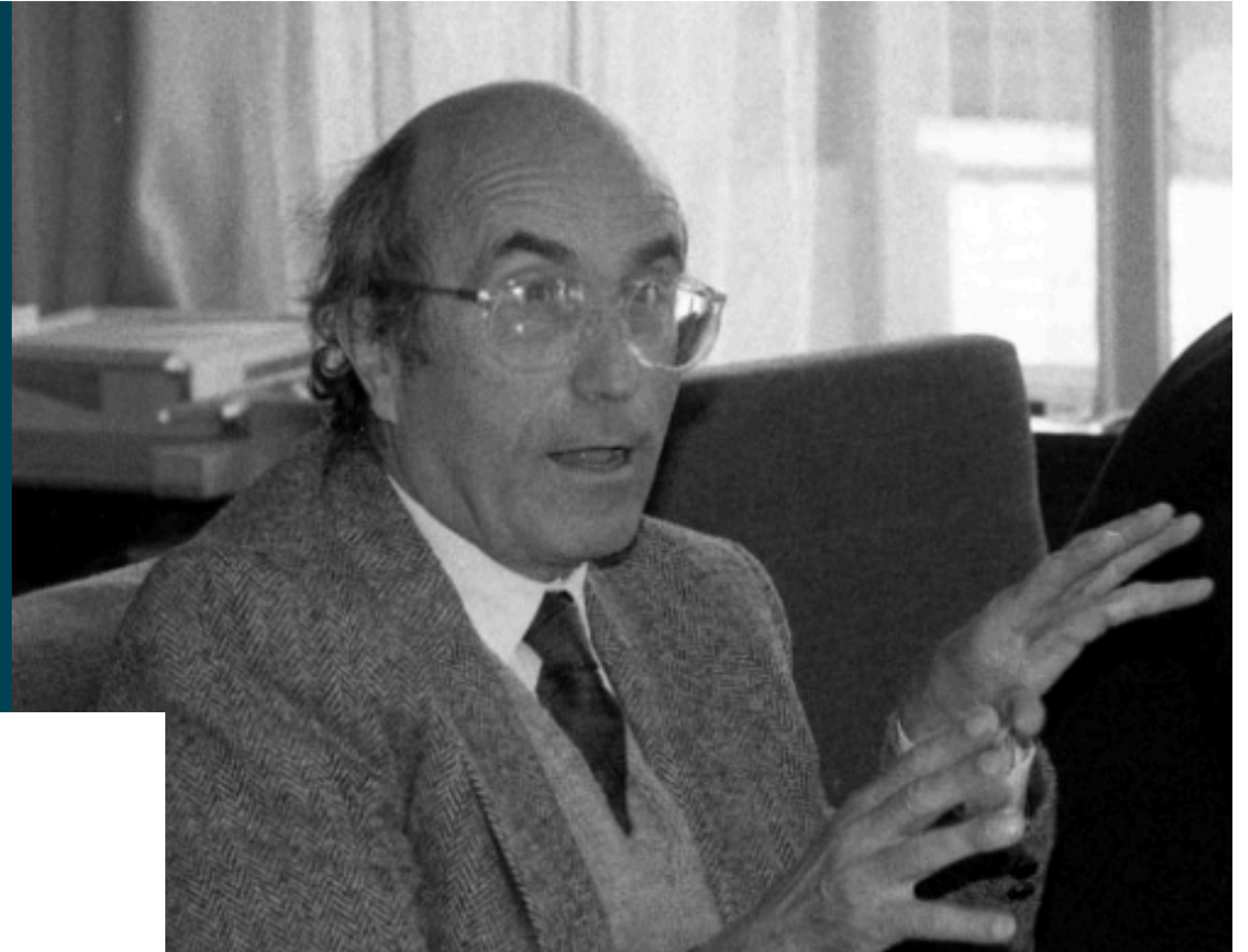
Niklas Luhmann



[Quelle: Universitätsarchiv St.Gallen | HSGH 022/000941 | CC-BY-SA 4.0](#)

**„System“ ist
Abgrenzung von der
Außenwelt**

Niklas Luhmann



Ein System (nach Luhmann) steuert bewusst die Kontaktpunkte zur Außenwelt, um in einer unendlich komplexen Welt handlungsfähig zu bleiben.

**Soziale Systeme stabilisieren sich selbst, durch
Künstliche Vereinfachung ihrer Umwelt, um
Handlungsfähigkeit & Funktionsfähigkeit zu erhalten**

Wer (mehrere) Kinder hat, wird das kennen ...

Regel im Familiensystem: „Nur einer redet gleichzeitig“

**Bewusste Reduktion der Kontaktpunkte des System
„Eltern“ mit dem System „Familie“**

Zur Selbsterhaltung ...

Regel im System Eltern: „Entscheidungen über Bildschirmzeit werden abgestimmt und einheitlich nach außen vertreten.“

-> System „Eltern“ bleibt handlungsfähig

Vereinfachte Umwelt
Weniger Rückkopplung
Weniger Komplexität

Komplex vs. Kompliziert

Ist das Schienennetz komplex?



Von Hbf878, OpenStreetMap contributors - Eigenes Werk, using OpenStreetMap data, CC BY-SA 2.0, <https://commons.wikimedia.org/w/index.php?curid=89862775>

Ist das Schienennetz komplex?



Von Hbf878, OpenStreetMap contributors - Eigenes Werk, using OpenStreetMap data, CC BY-SA 2.0, <https://commons.wikimedia.org/w/index.php?curid=89862775>

Ist das Schienennetz komplex?



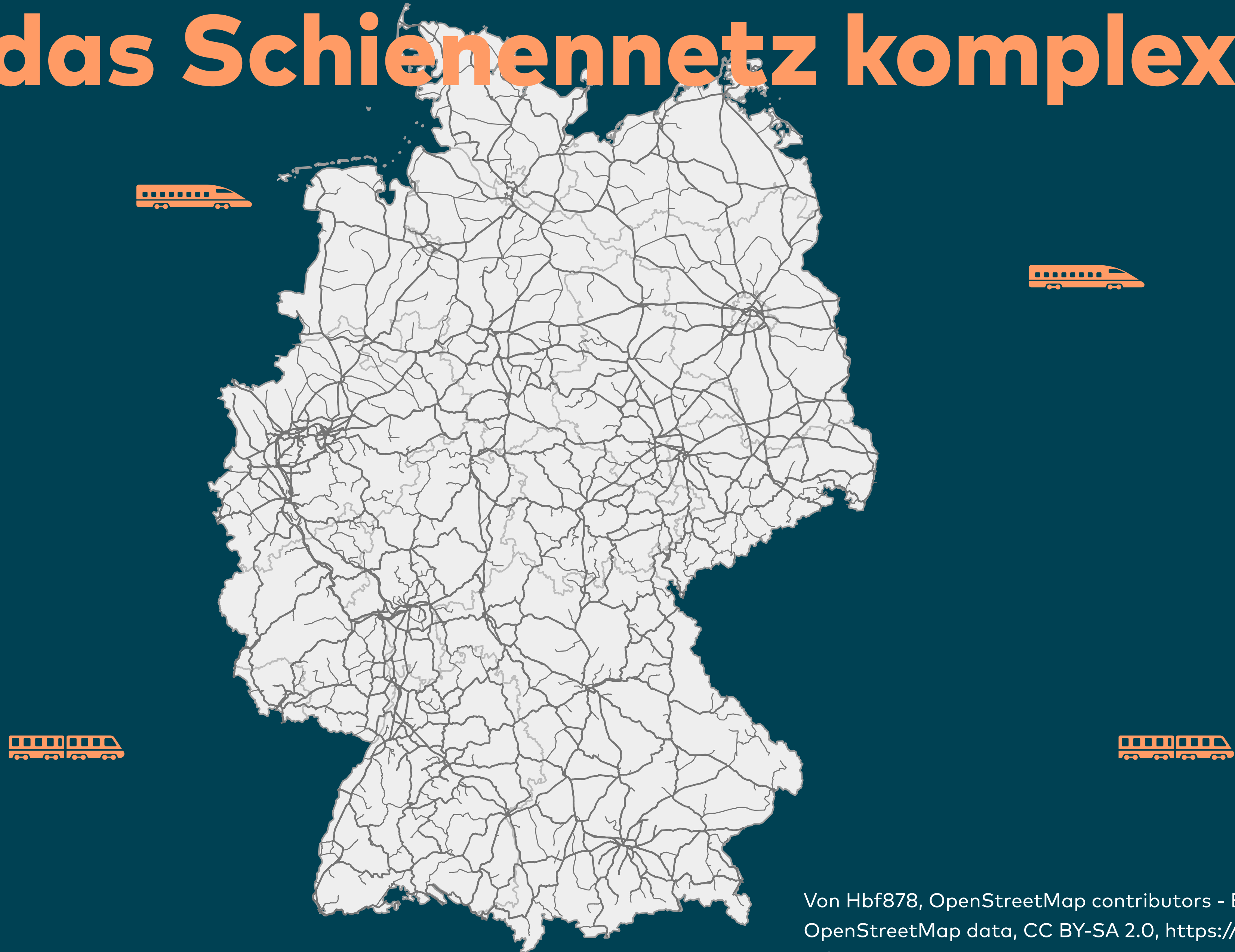
Von Hbf878, OpenStreetMap contributors - Eigenes Werk, using OpenStreetMap data, CC BY-SA 2.0, <https://commons.wikimedia.org/w/index.php?curid=89862775>

Ist das Schienennetz komplex?



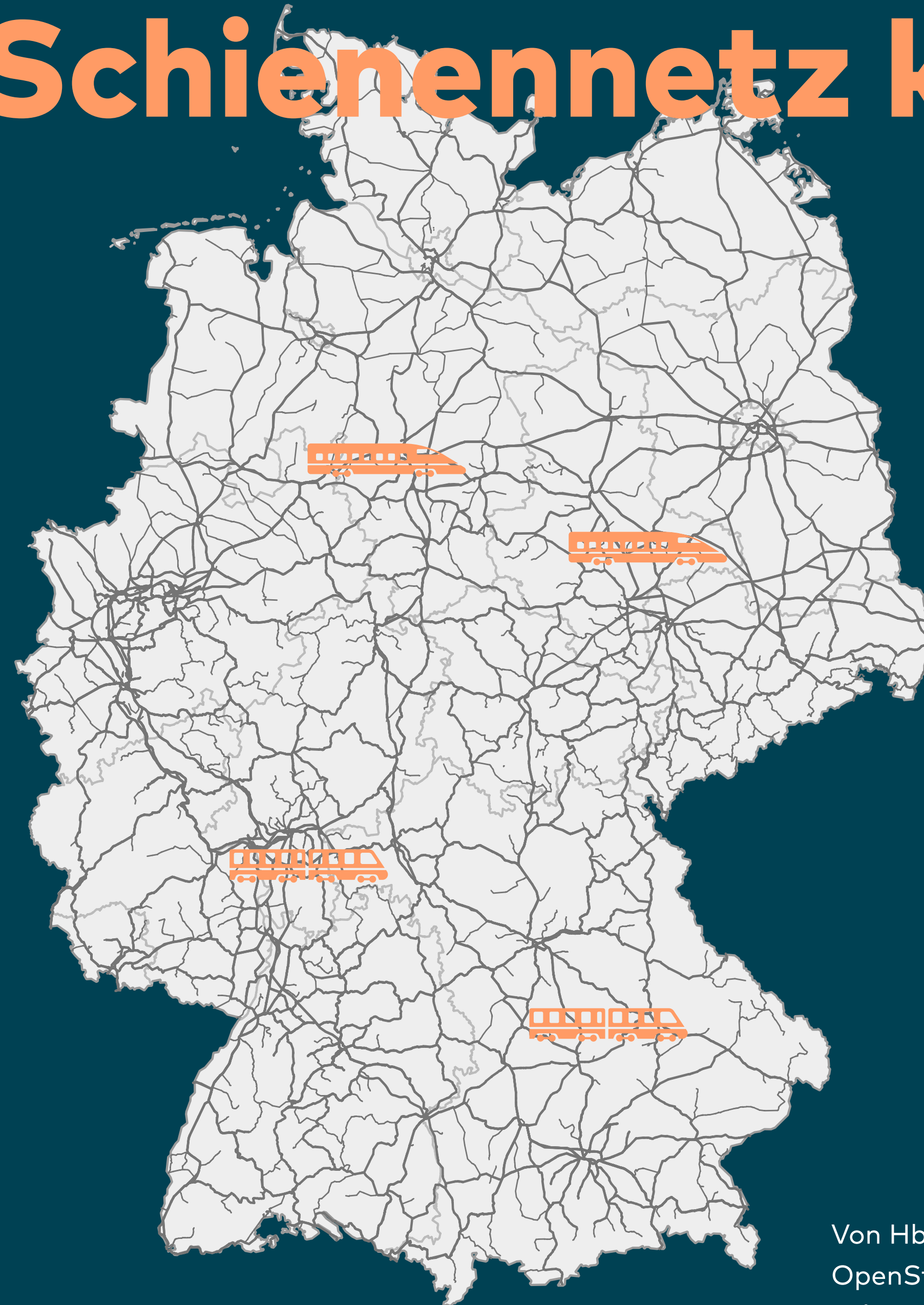
Von Hbf878, OpenStreetMap contributors - Eigenes Werk, using
OpenStreetMap data, CC BY-SA 2.0, [https://commons.wikimedia.org/
w/index.php?curid=89862775](https://commons.wikimedia.org/w/index.php?curid=89862775)

Ist das Schienennetz komplex?



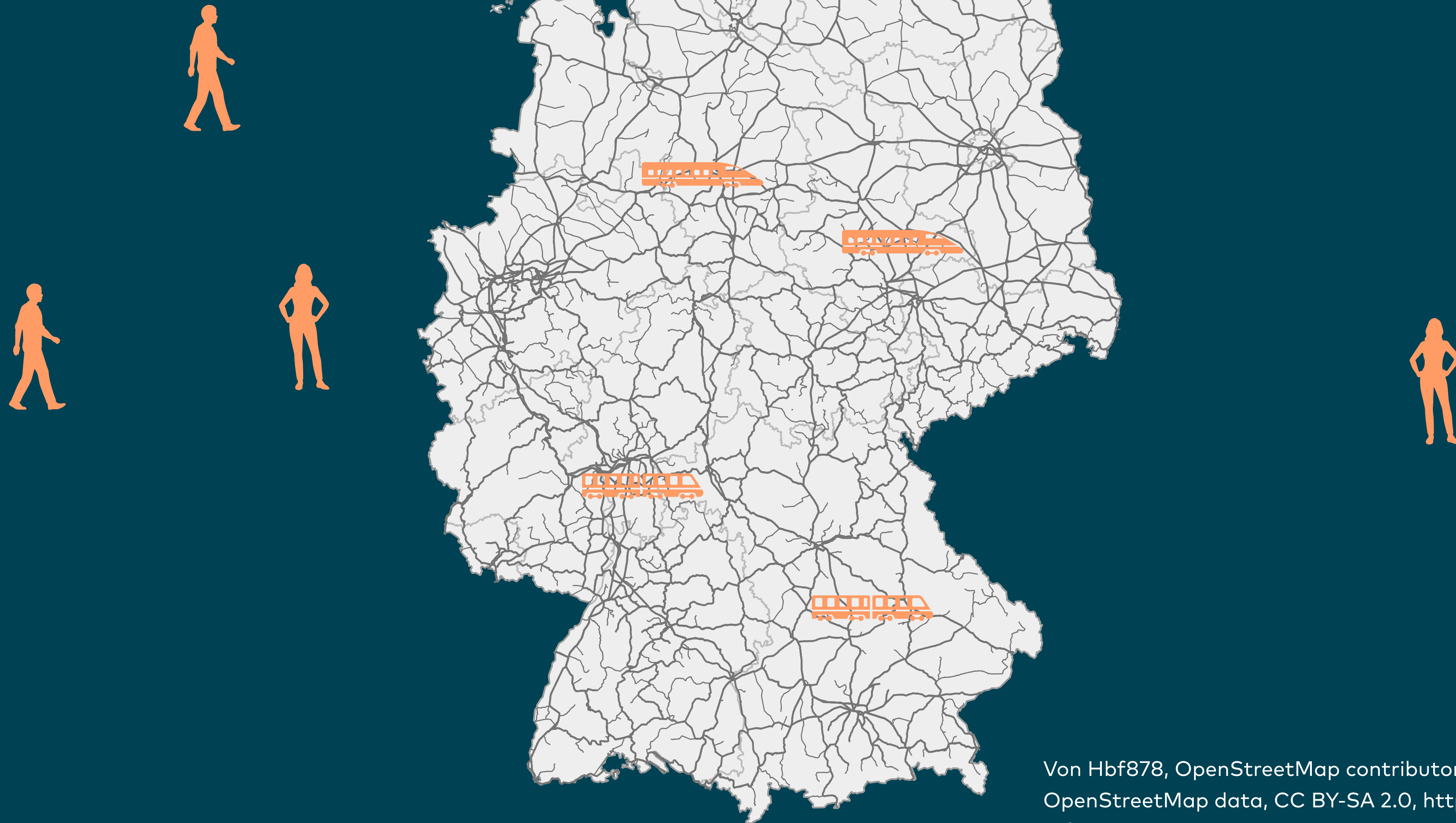
Von Hbf878, OpenStreetMap contributors - Eigenes Werk, using OpenStreetMap data, CC BY-SA 2.0, <https://commons.wikimedia.org/w/index.php?curid=89862775>

Ist das Schienennetz komplex?



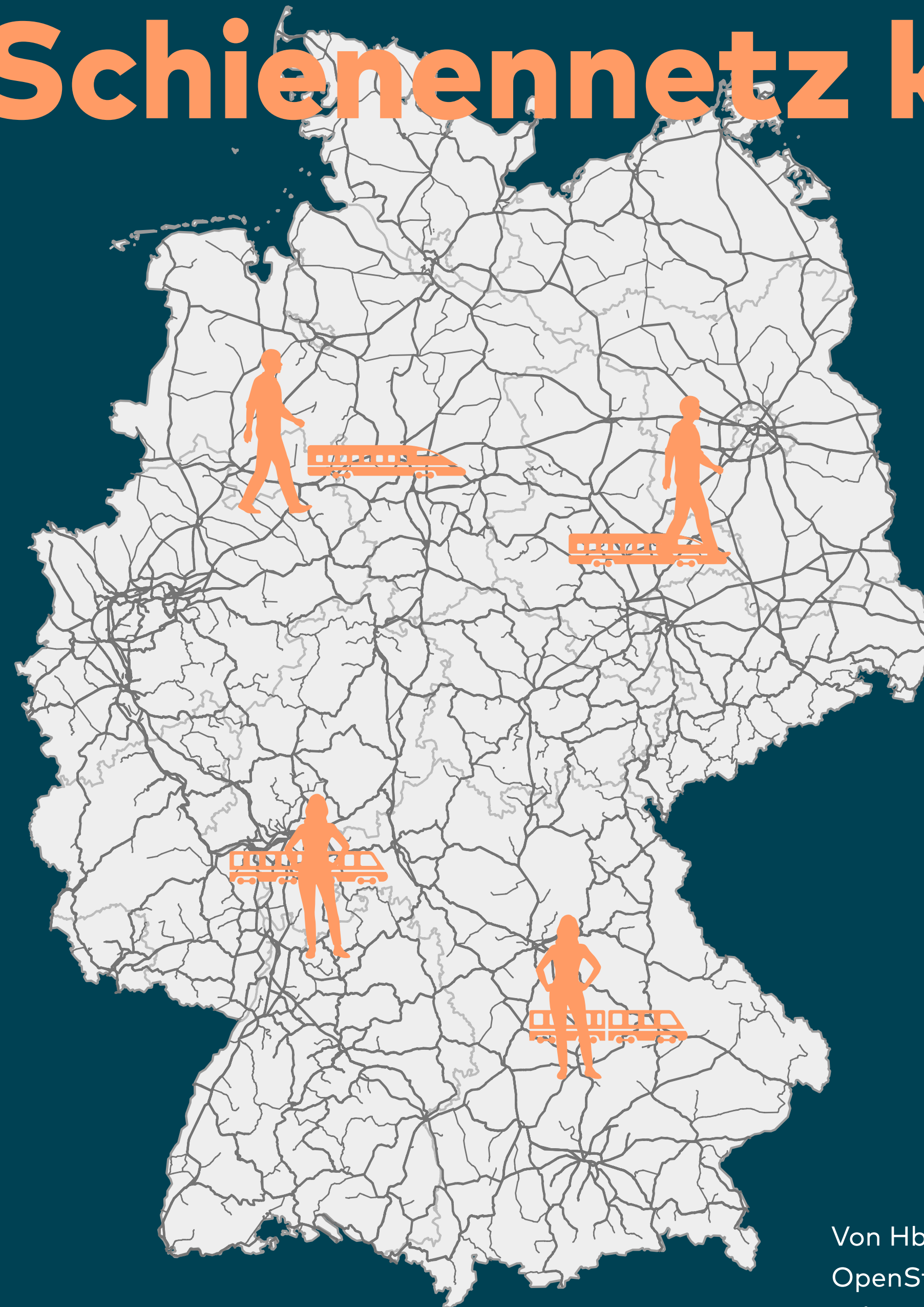
Von Hbf878, OpenStreetMap contributors - Eigenes Werk, using OpenStreetMap data, CC BY-SA 2.0, <https://commons.wikimedia.org/w/index.php?curid=89862775>

Ist das Schienennetz komplex?



Von Hbf878, OpenStreetMap contributors - Eigenes Werk, using OpenStreetMap data, CC BY-SA 2.0, <https://commons.wikimedia.org/w/index.php?curid=89862775>

Ist das Schienennetz komplex?



Von Hbf878, OpenStreetMap contributors - Eigenes Werk, using OpenStreetMap data, CC BY-SA 2.0, <https://commons.wikimedia.org/w/index.php?curid=89862775>

System zeigt ein emergentes, nicht vorhersehbares Verhalten durch Rückkopplungseffekte.^{^1}

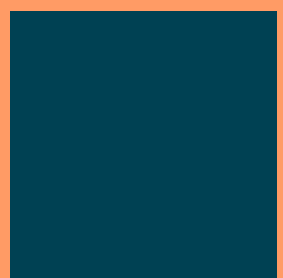
^{^1} <https://www.zdf.de/wissen/frag-den-lesch/komplex-oder-kompliziert---was-macht-den-unterschied-100.html>

**Ein kompliziertes System kann ich
simulieren**

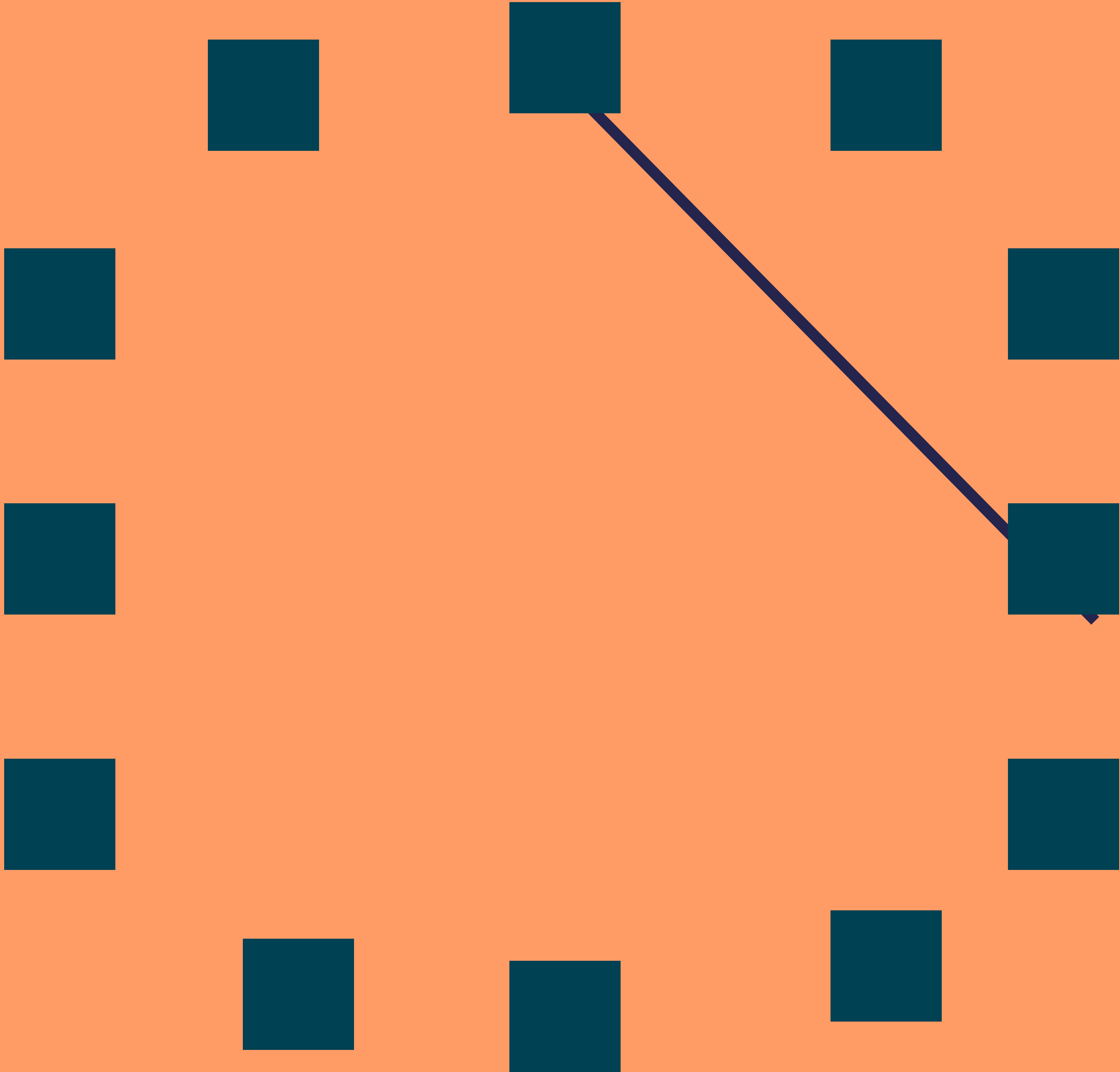
Ursache -> Wirkung ist vorhersehbar

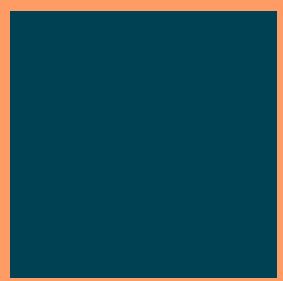
^{^1} <https://www.zdf.de/wissen/frag-den-lesch/komplex-oder-kompliziert---was-macht-den-unterschied-100.html>

**Kompliziert klingt für mich plötzlich
erstrebenswert (im Softwarekontext)**

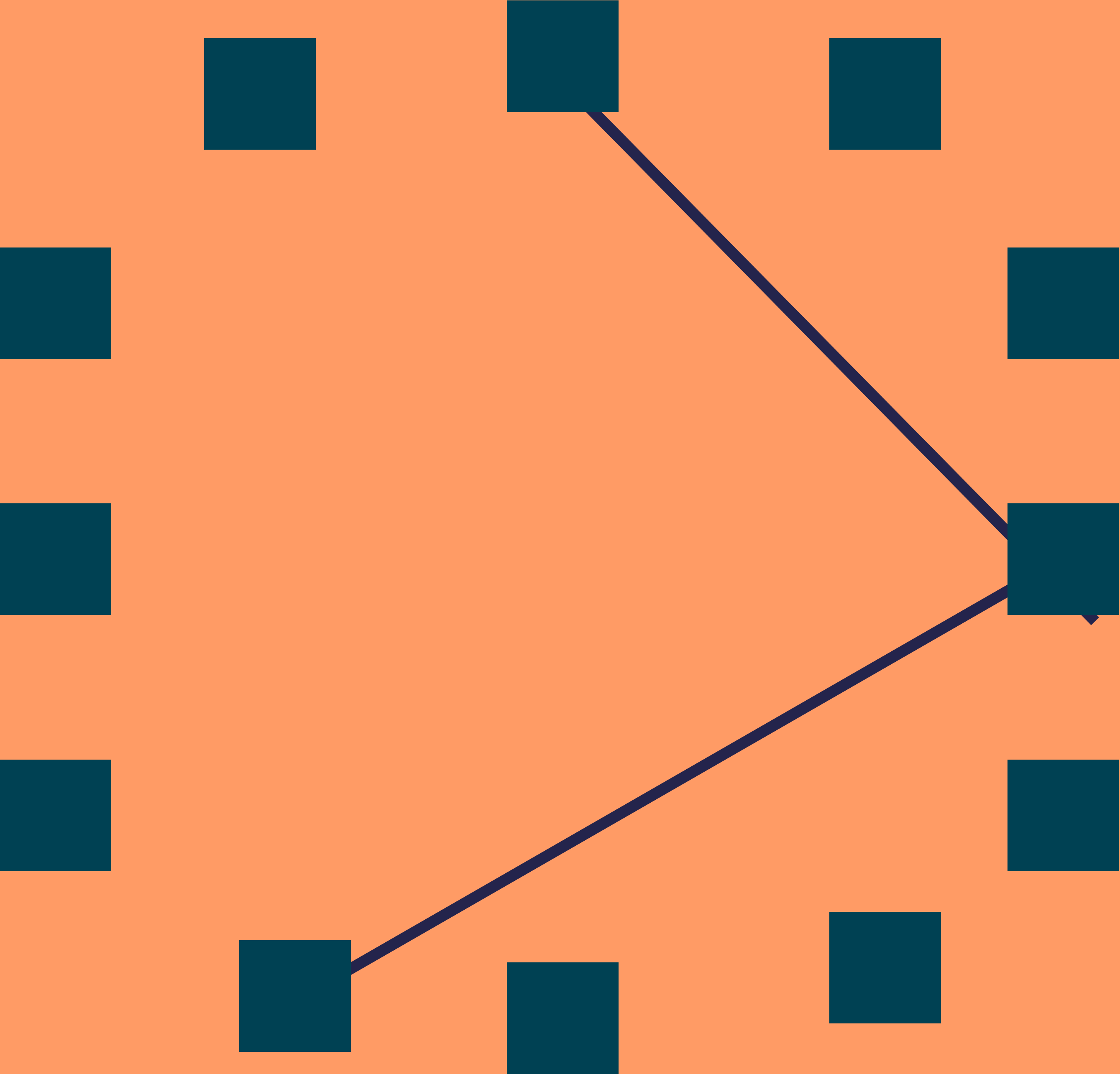


System



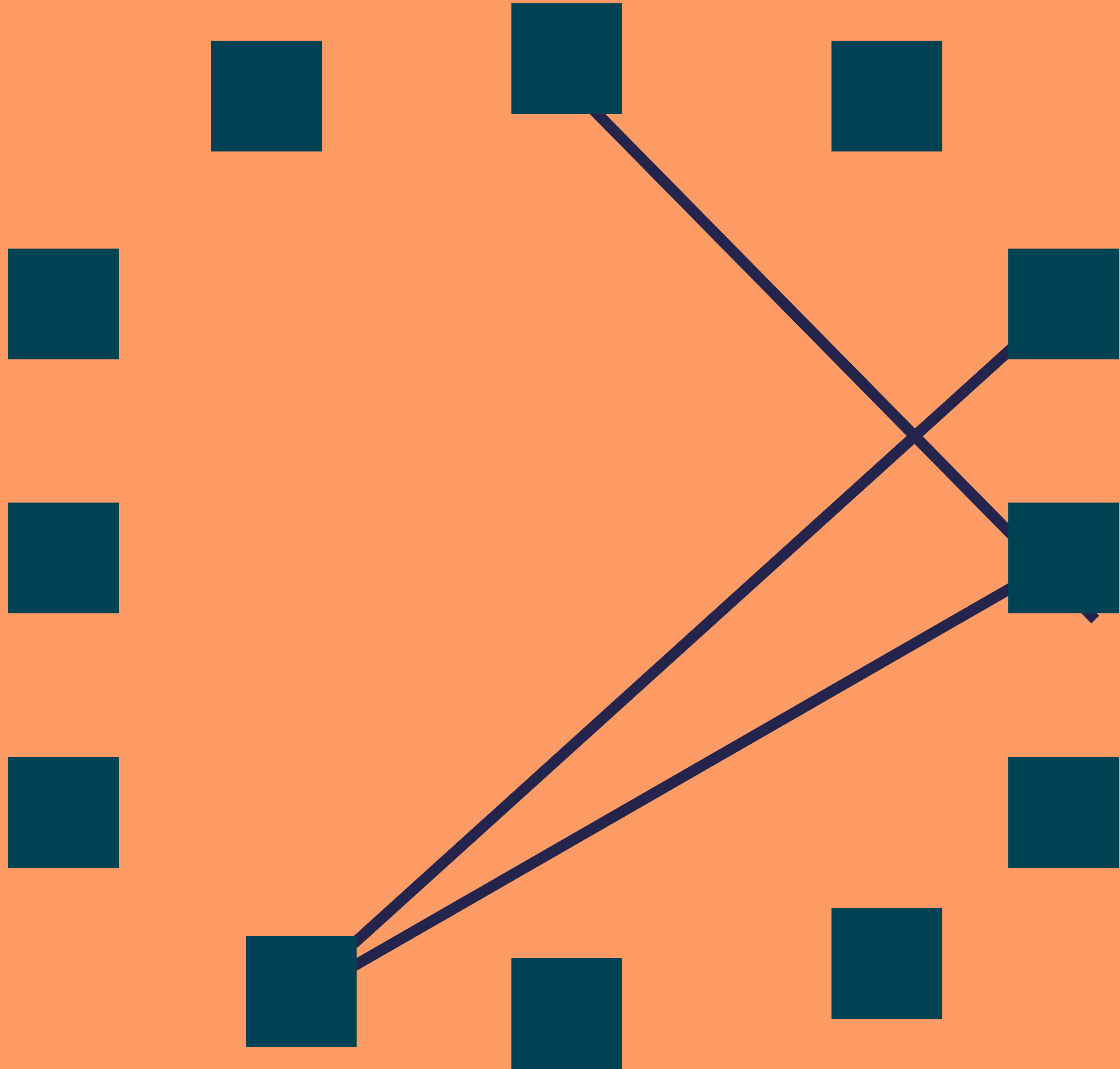


System



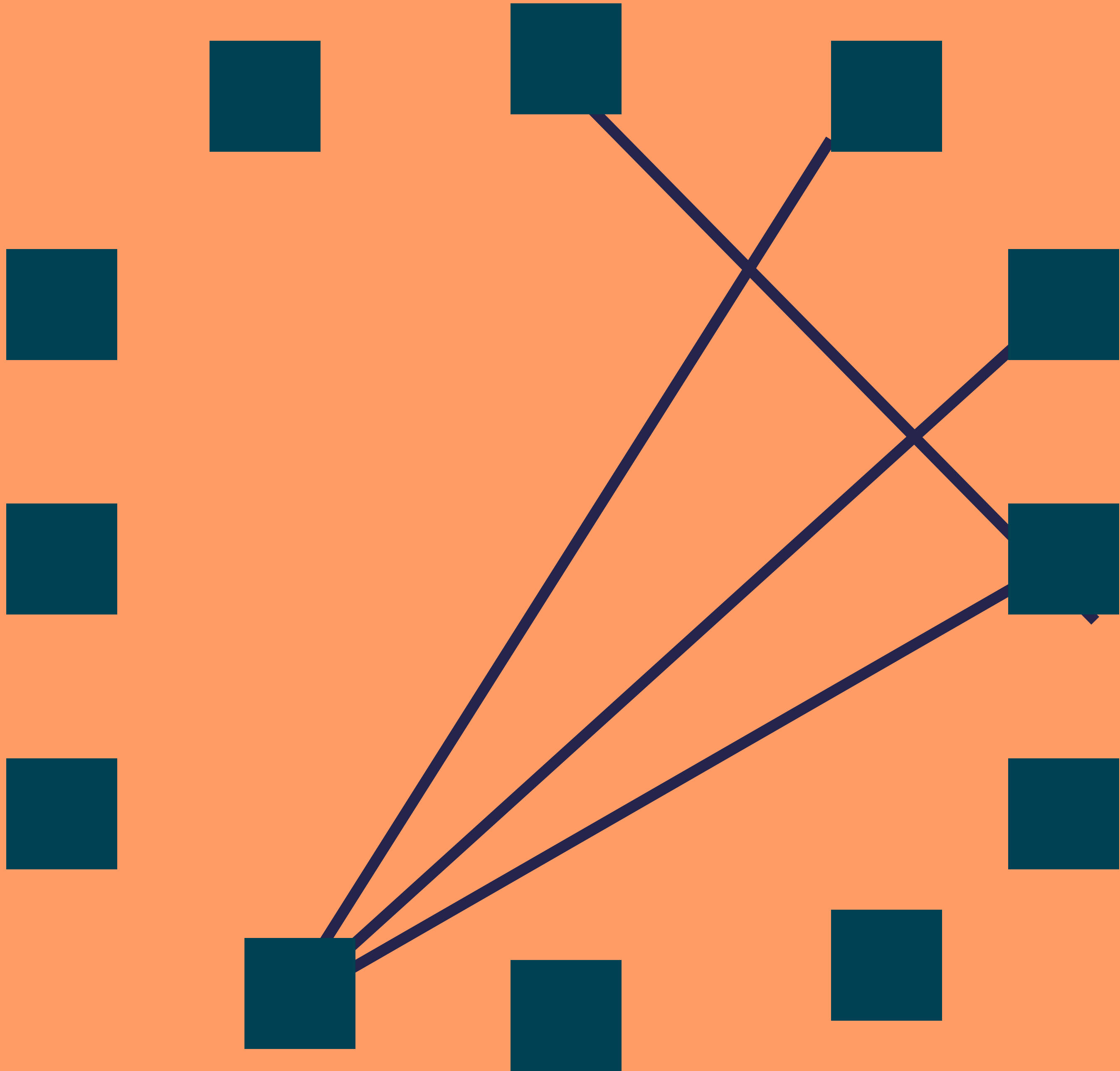


System



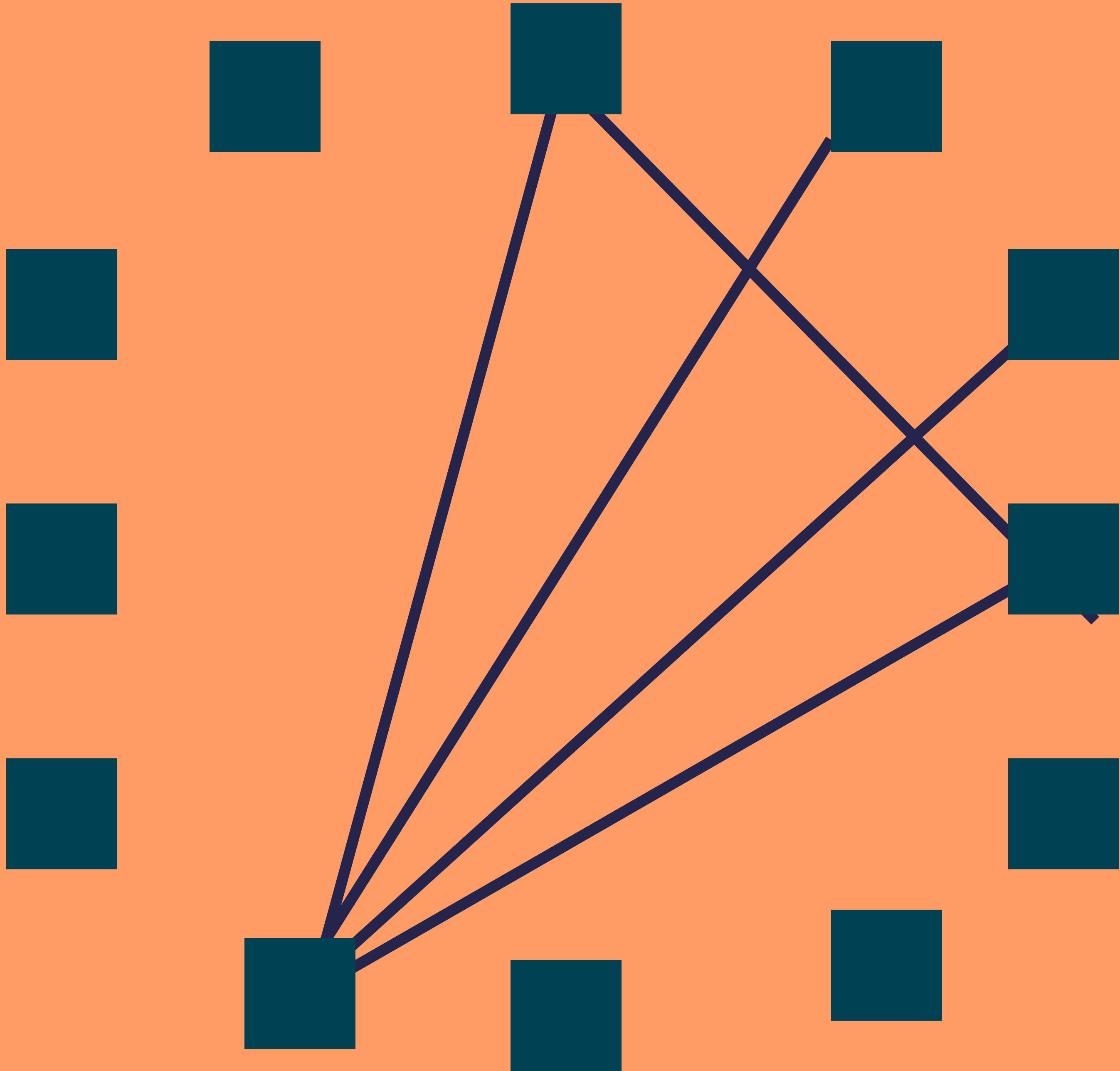


System



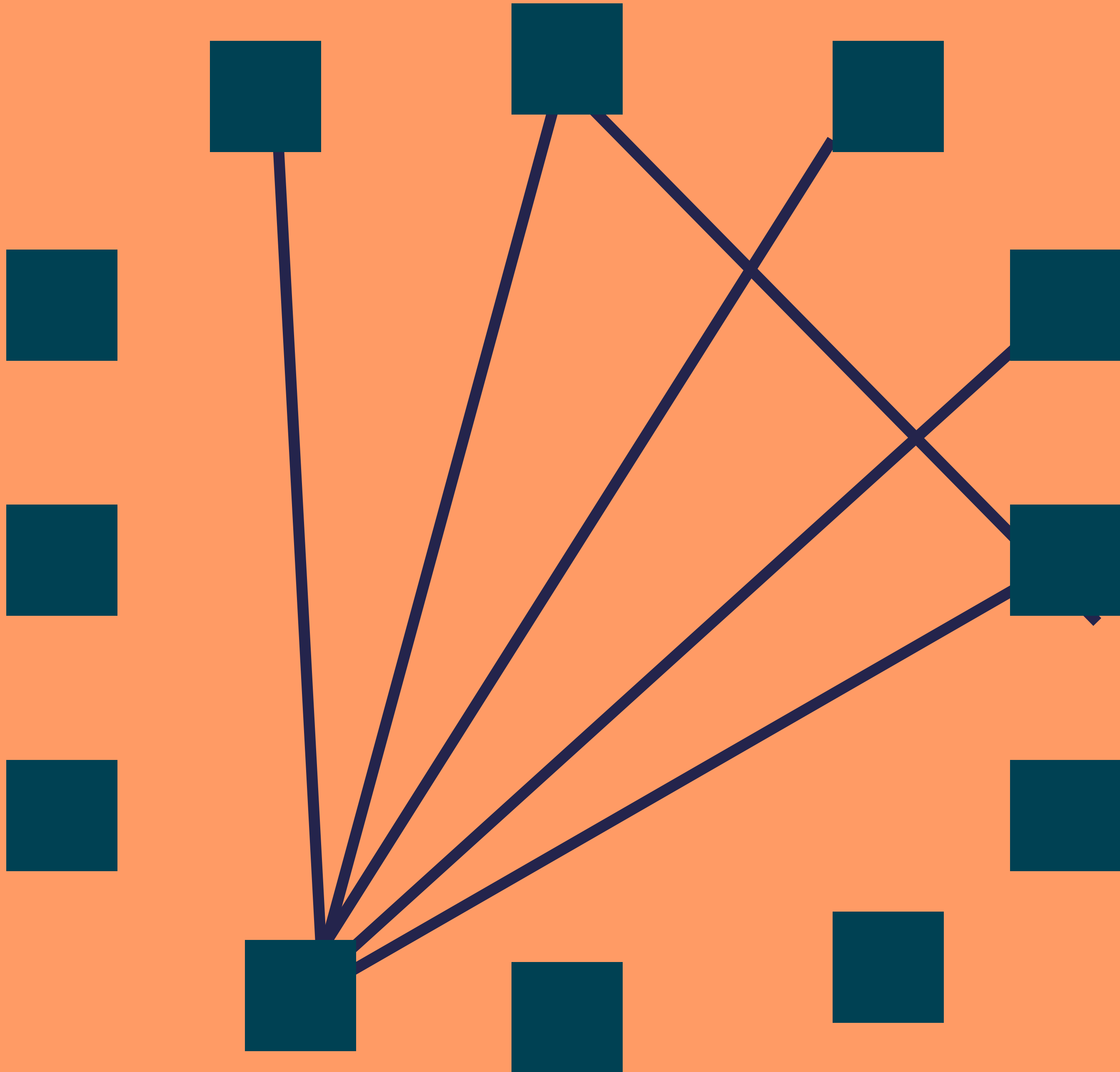


System



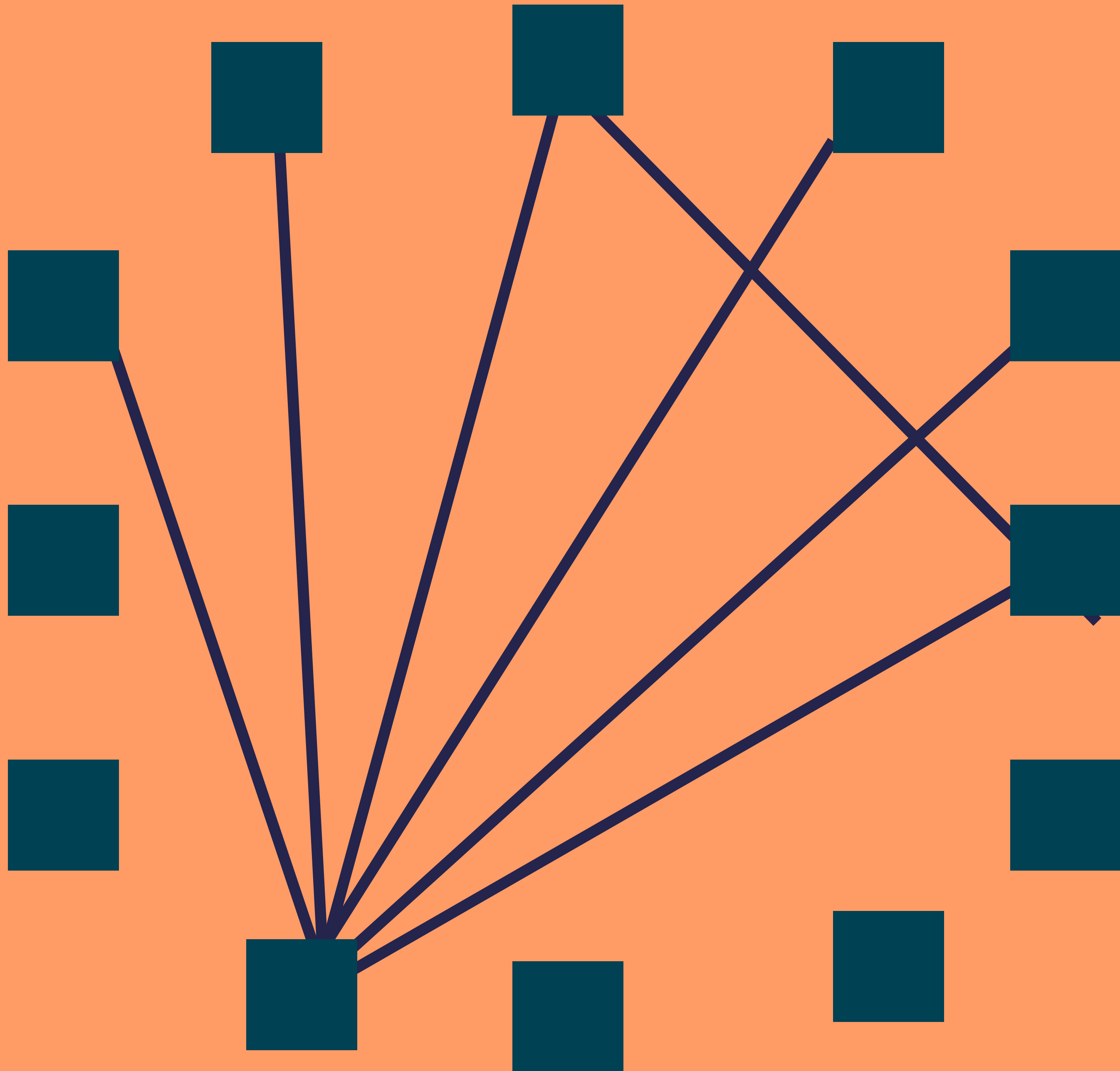


System



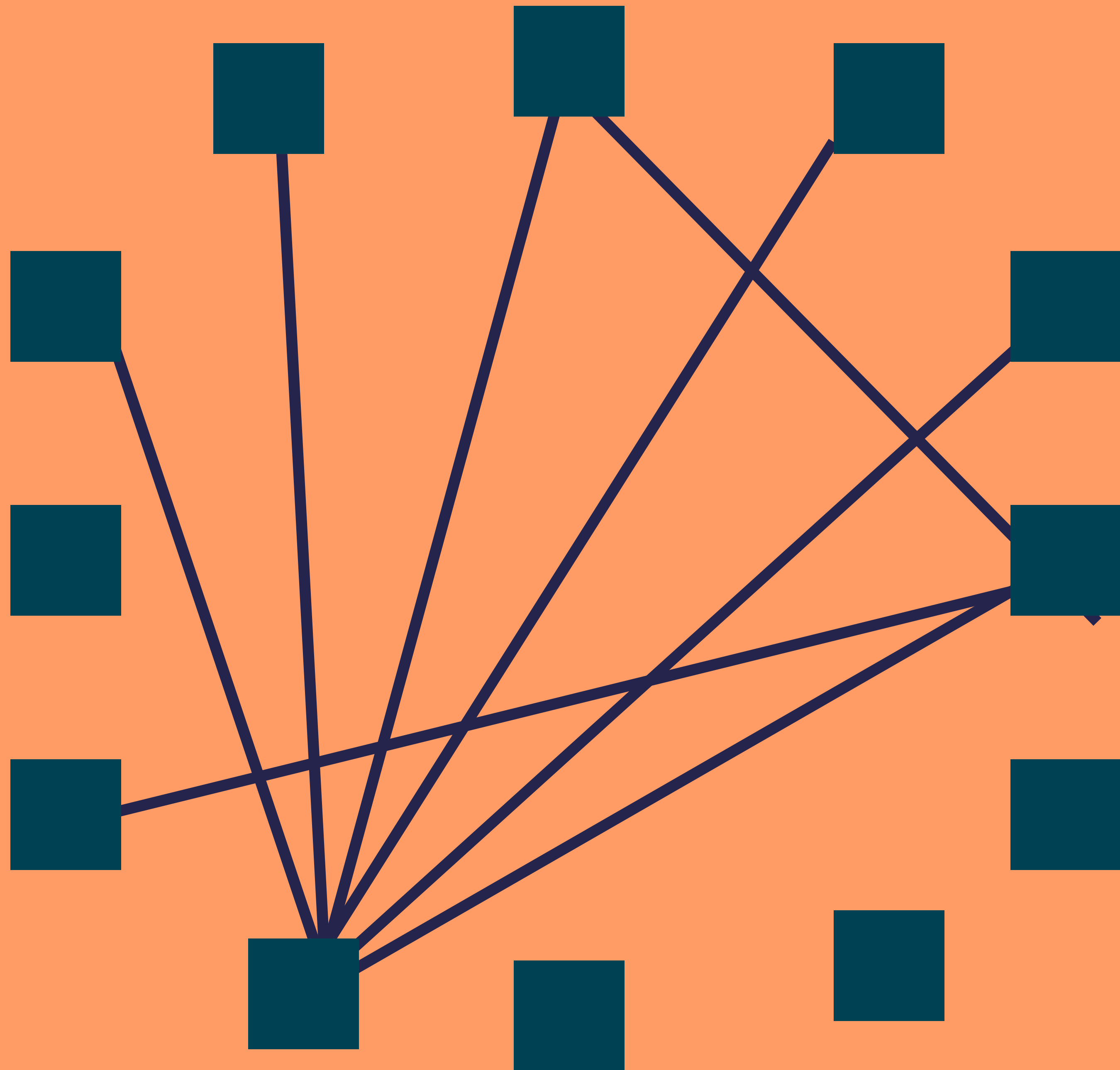


System



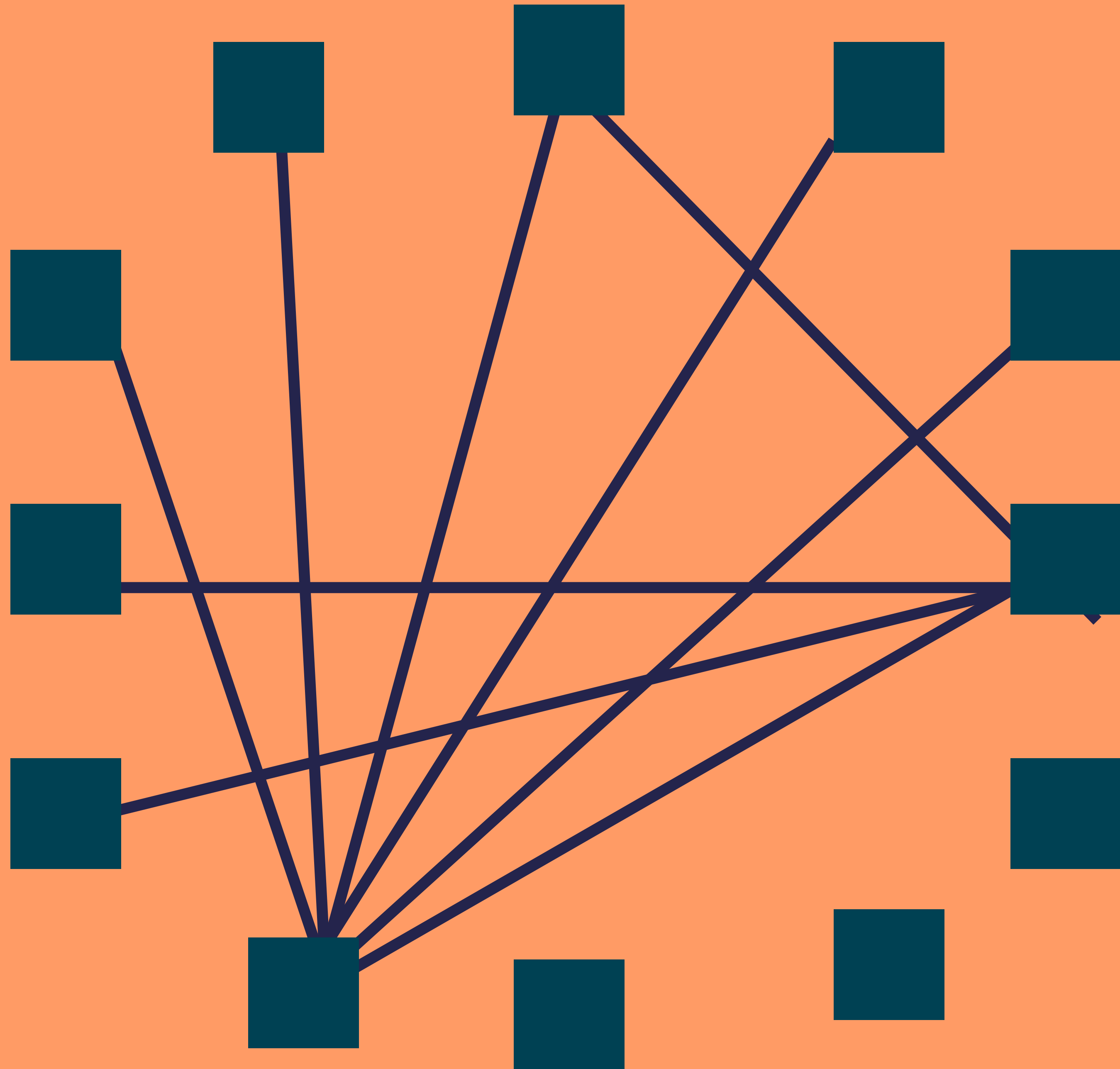


System



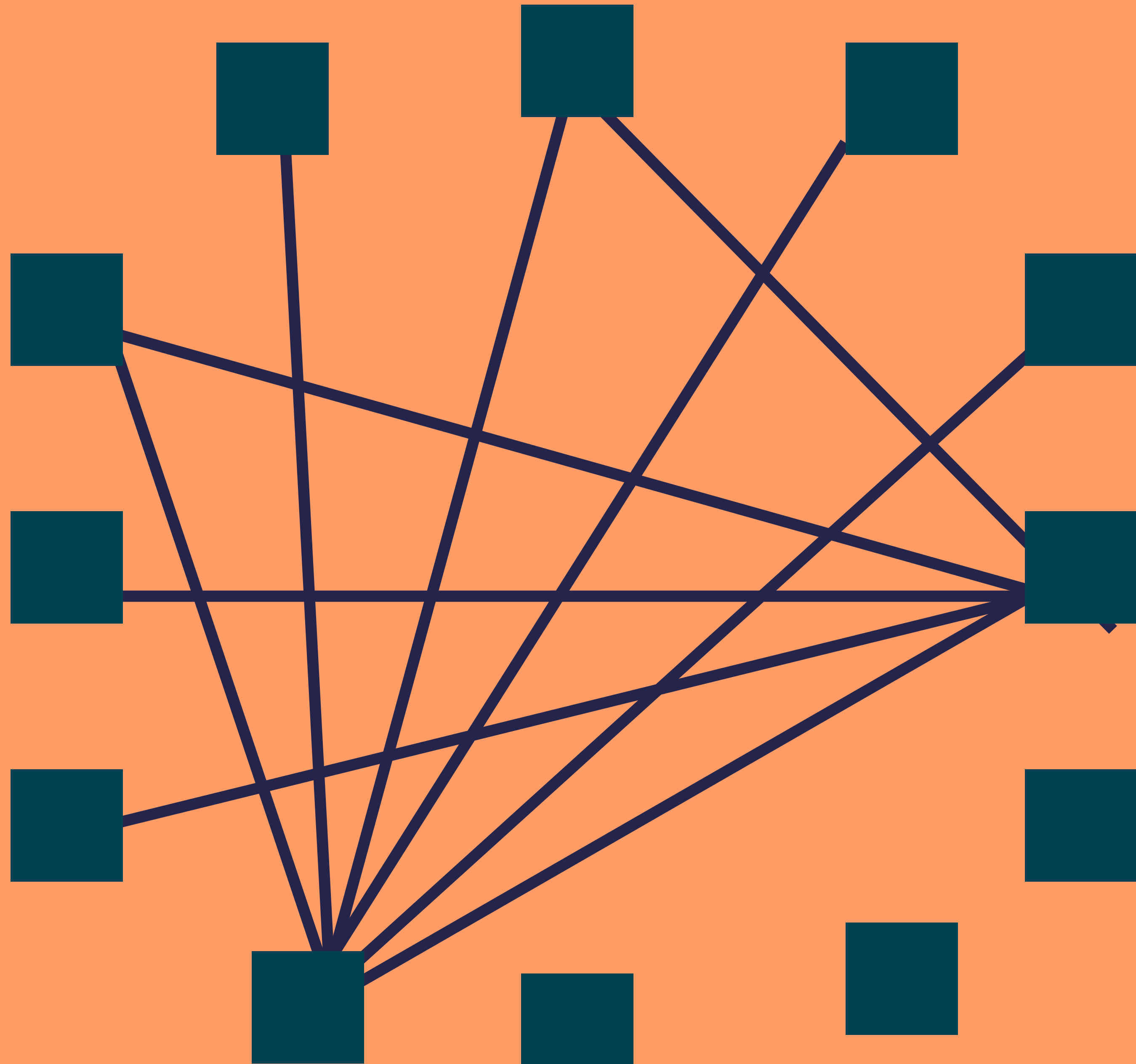


System



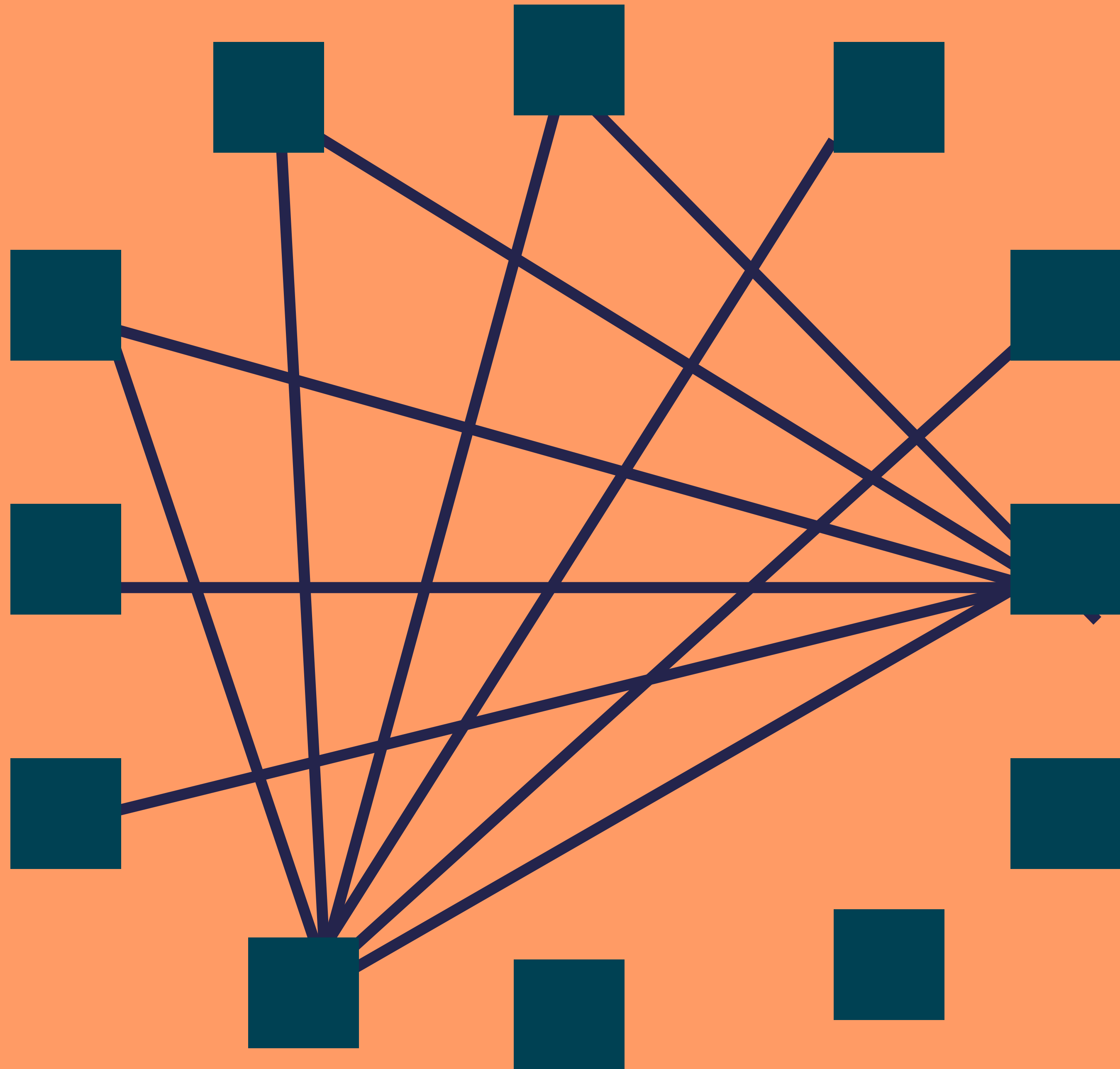


System



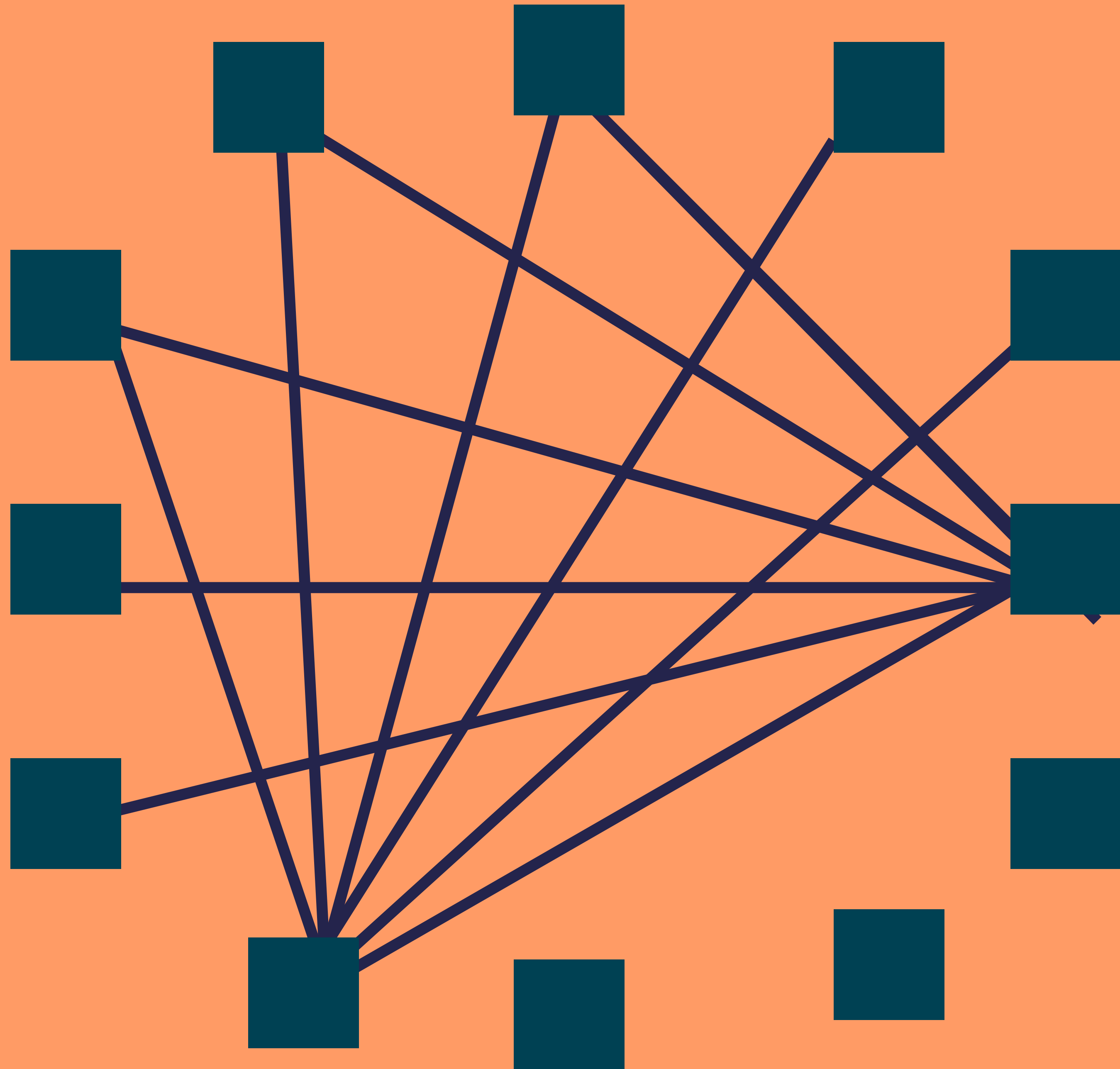


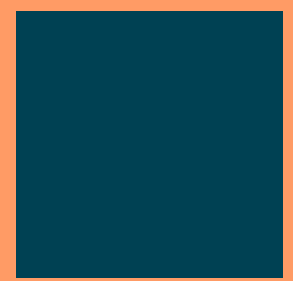
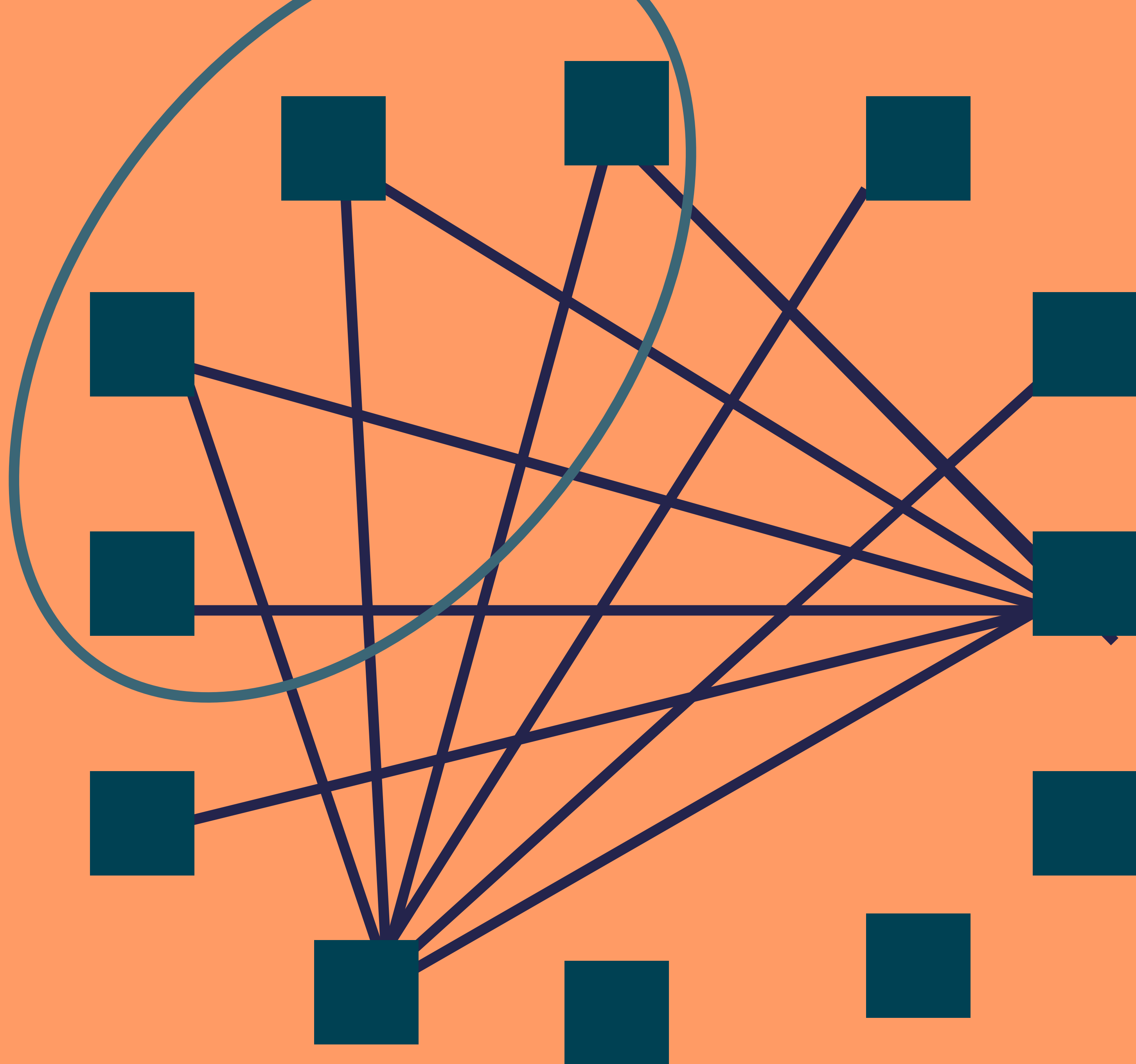
System



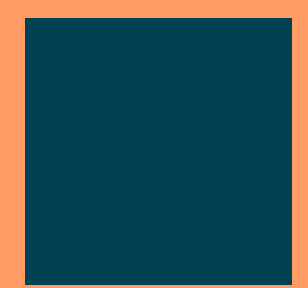
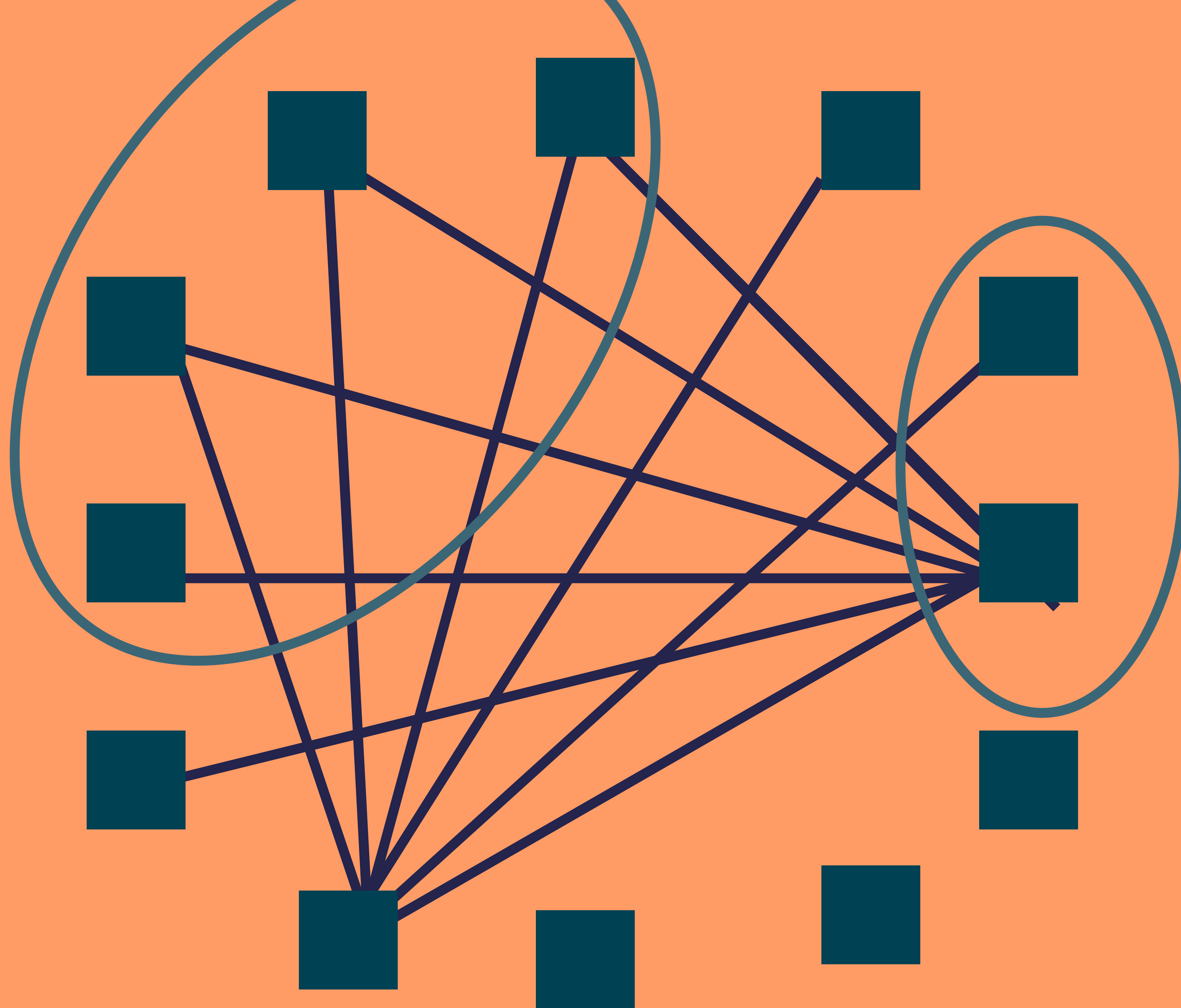


System

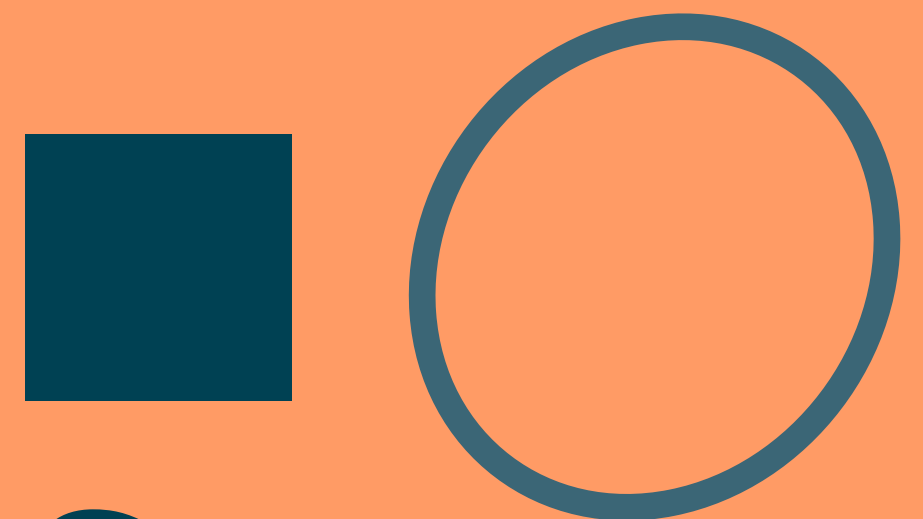
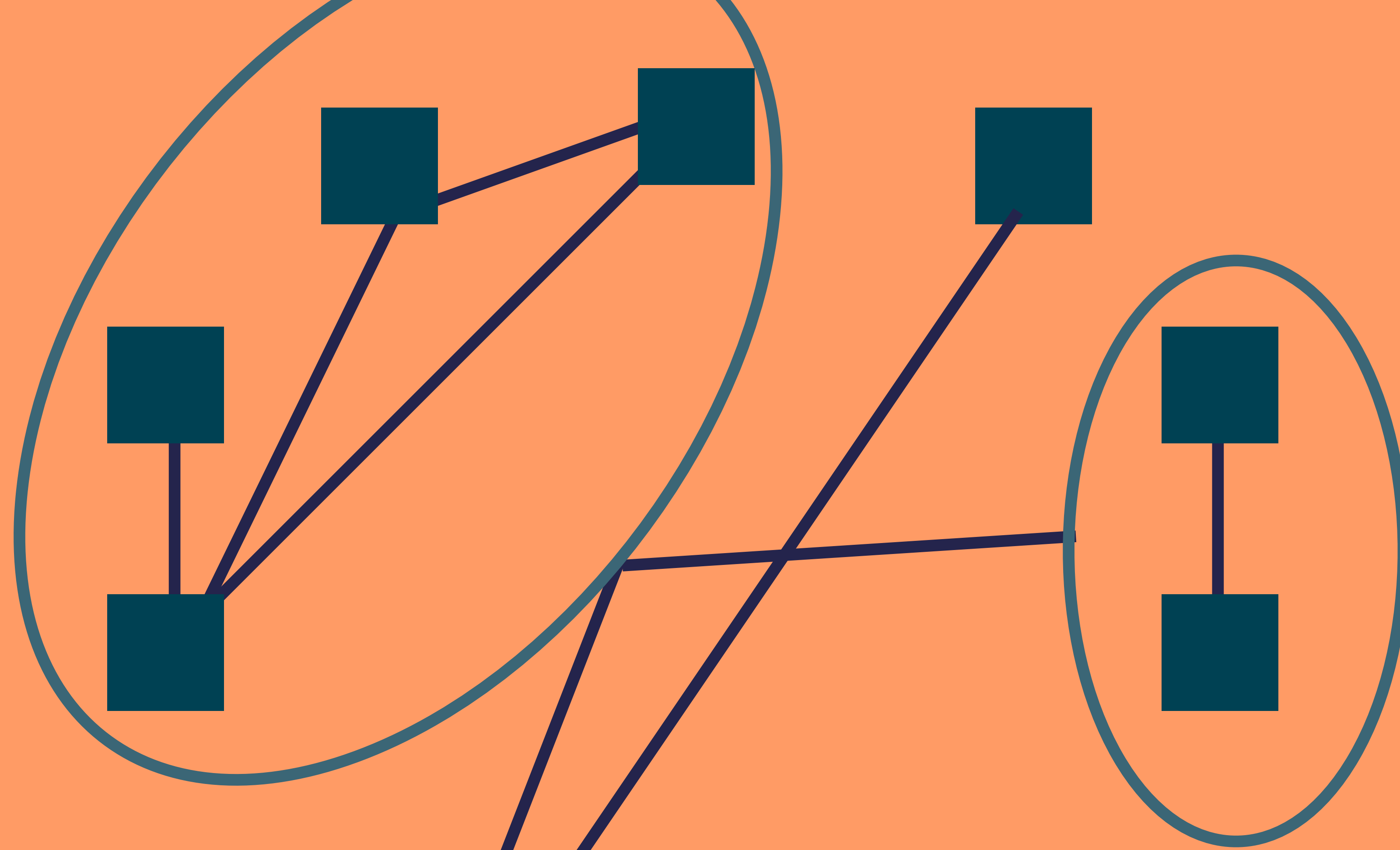




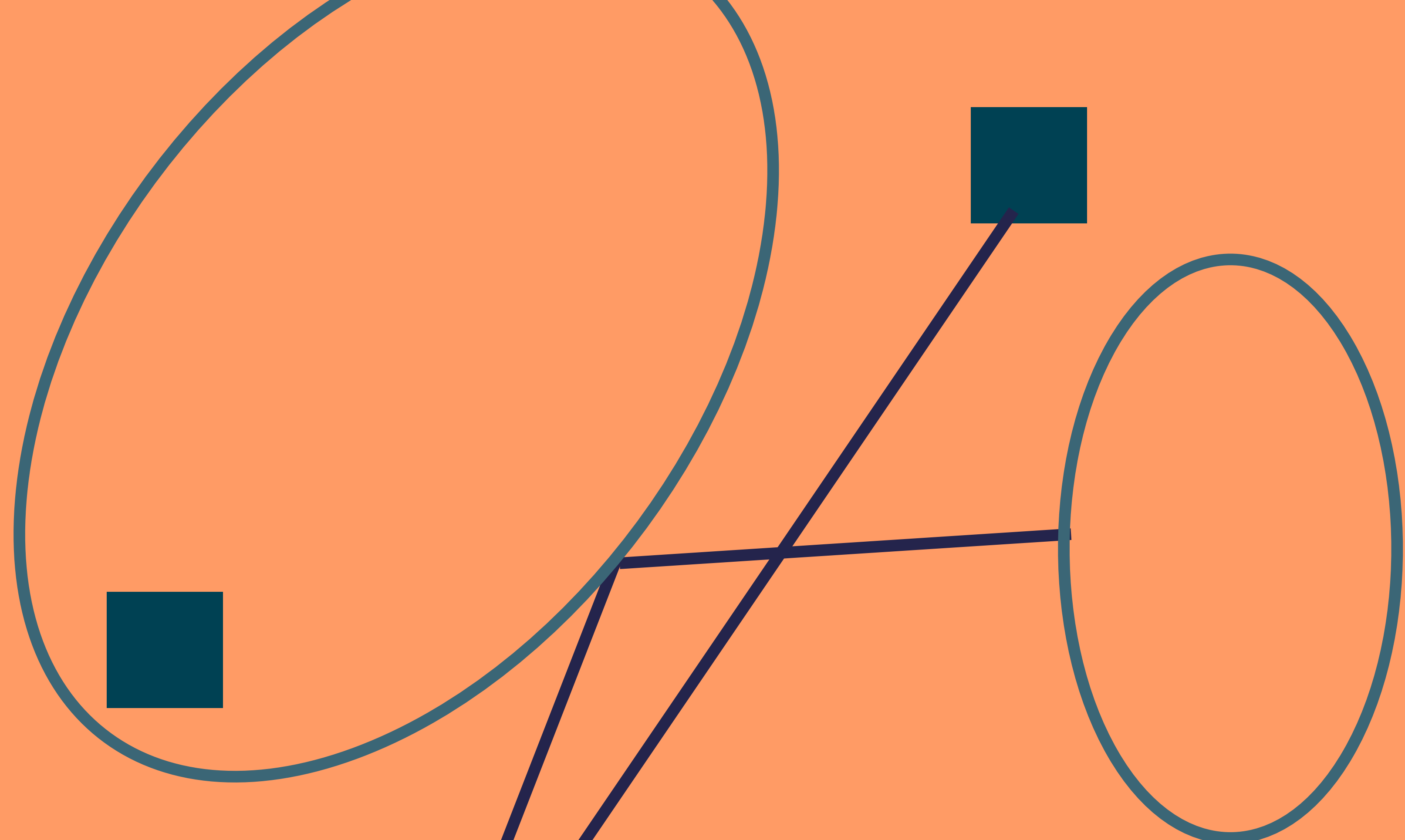
System



System



System



System

**Die Römer hätten gesagt:
„divide et impera“**

Komplexität \neq Komplexität

Essenz und Akzidenz

No Silver Bullet: Essence and Accidents of Software Engineering

by Frederick P. Brooks, Jr.

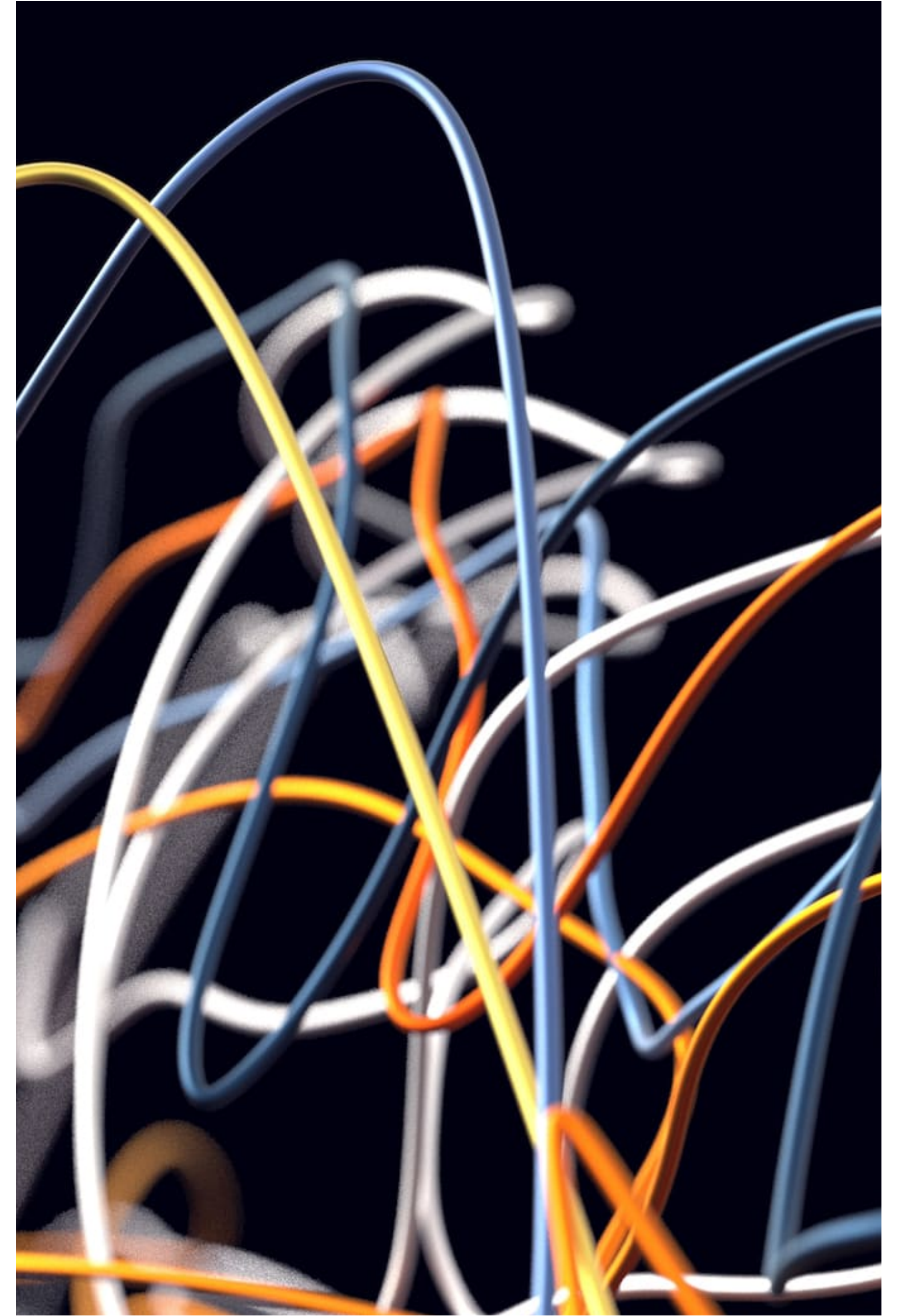
Komplexität in Software

Akzidenz

No Silver Bullet: Essence and Accidents of Software Engineering

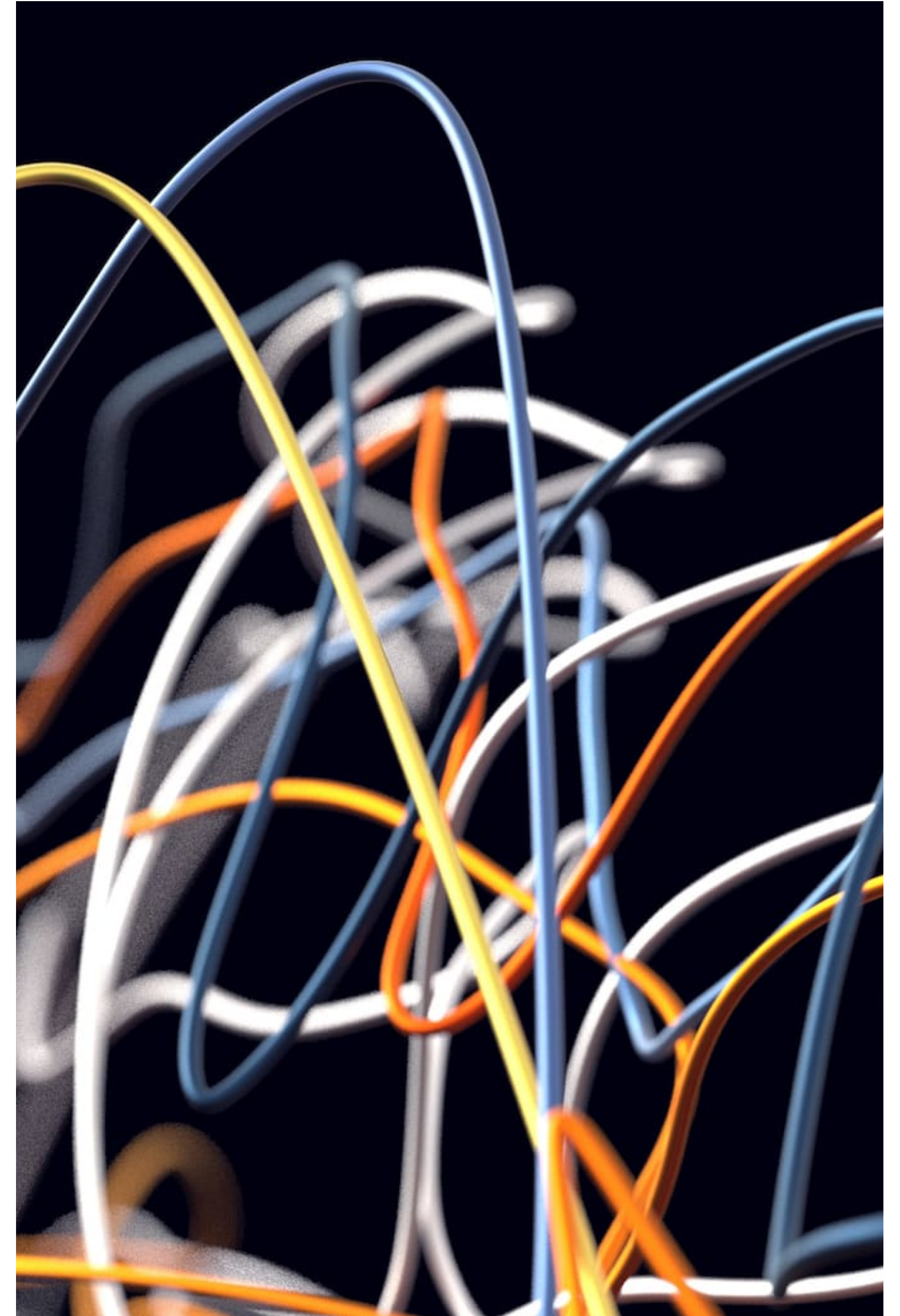
by Frederick P. Brooks, Jr.

Akzidentelle Komplexität



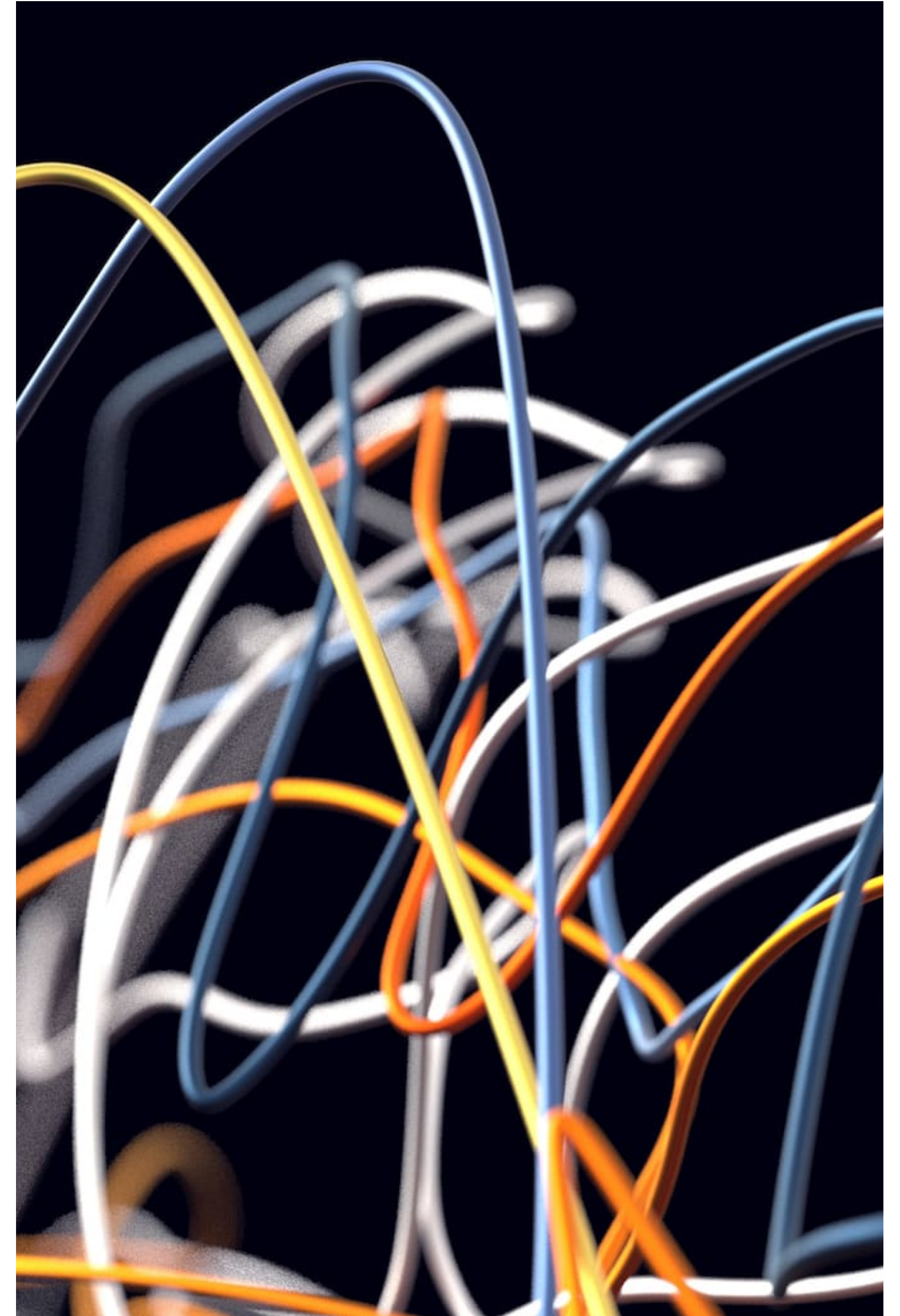
Akzidentelle Komplexität

- Technisch / Organisatorisch selbst erzeugt



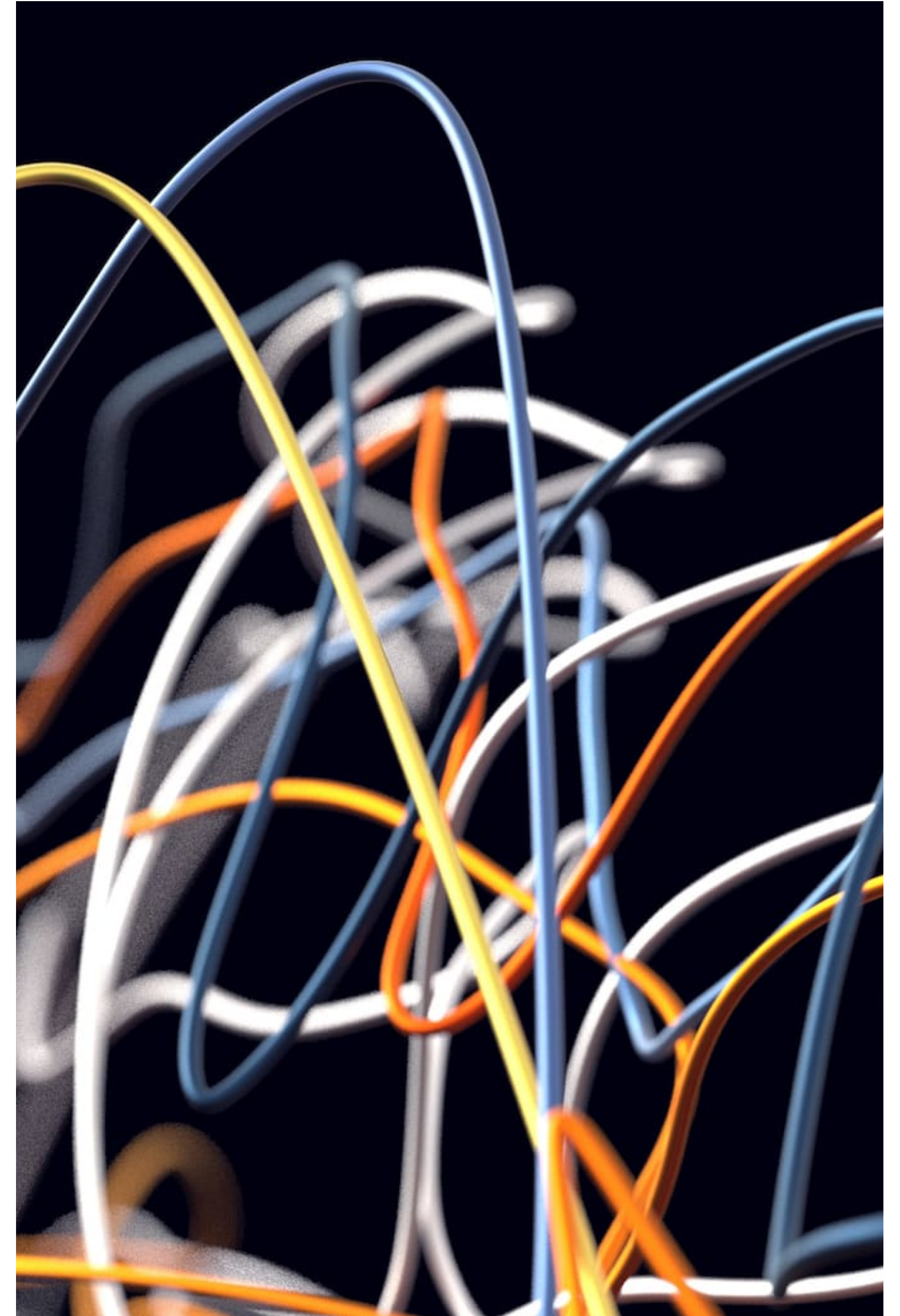
Akzidentelle Komplexität

- Technisch / Organisatorisch selbst erzeugt
- „Architektur Cargo Culting“



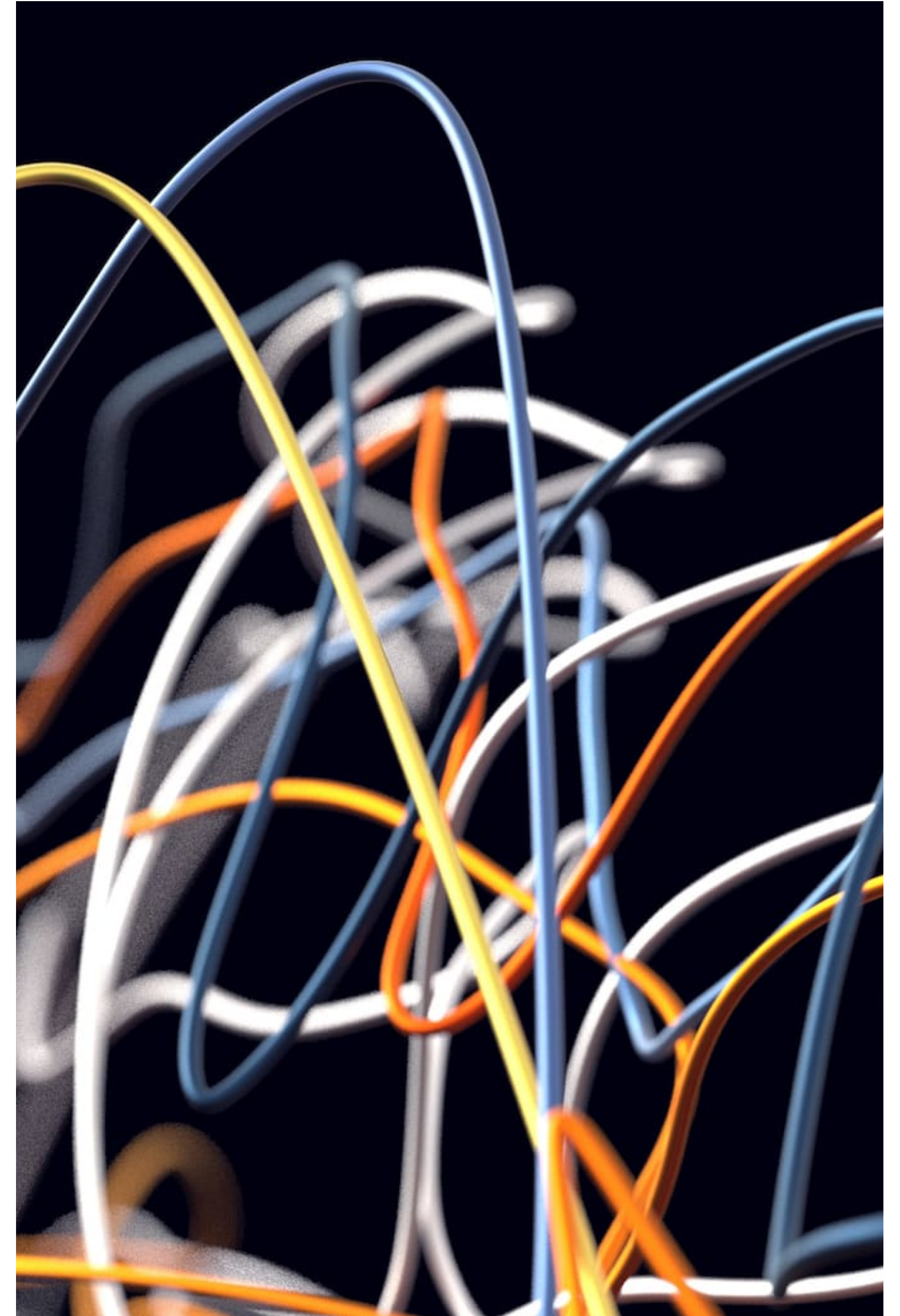
Akzidentelle Komplexität

- Technisch / Organisatorisch selbst erzeugt
- „Architektur Cargo Culting“
- Verteilte Systeme ohne Not



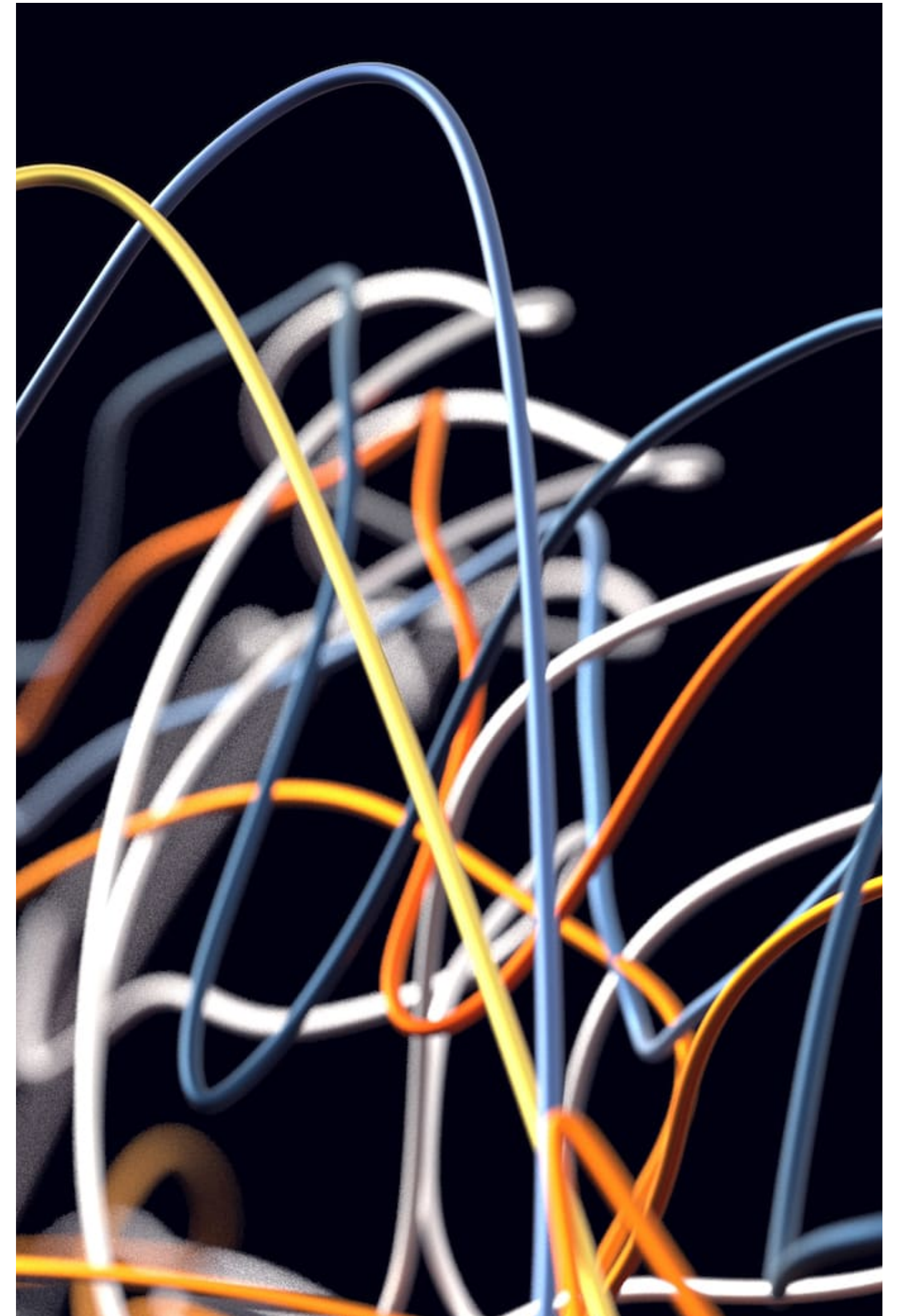
Akzidentelle Komplexität

- Technisch / Organisatorisch selbst erzeugt
- „Architektur Cargo Culting“
- Verteilte Systeme ohne Not
- Premature optimization



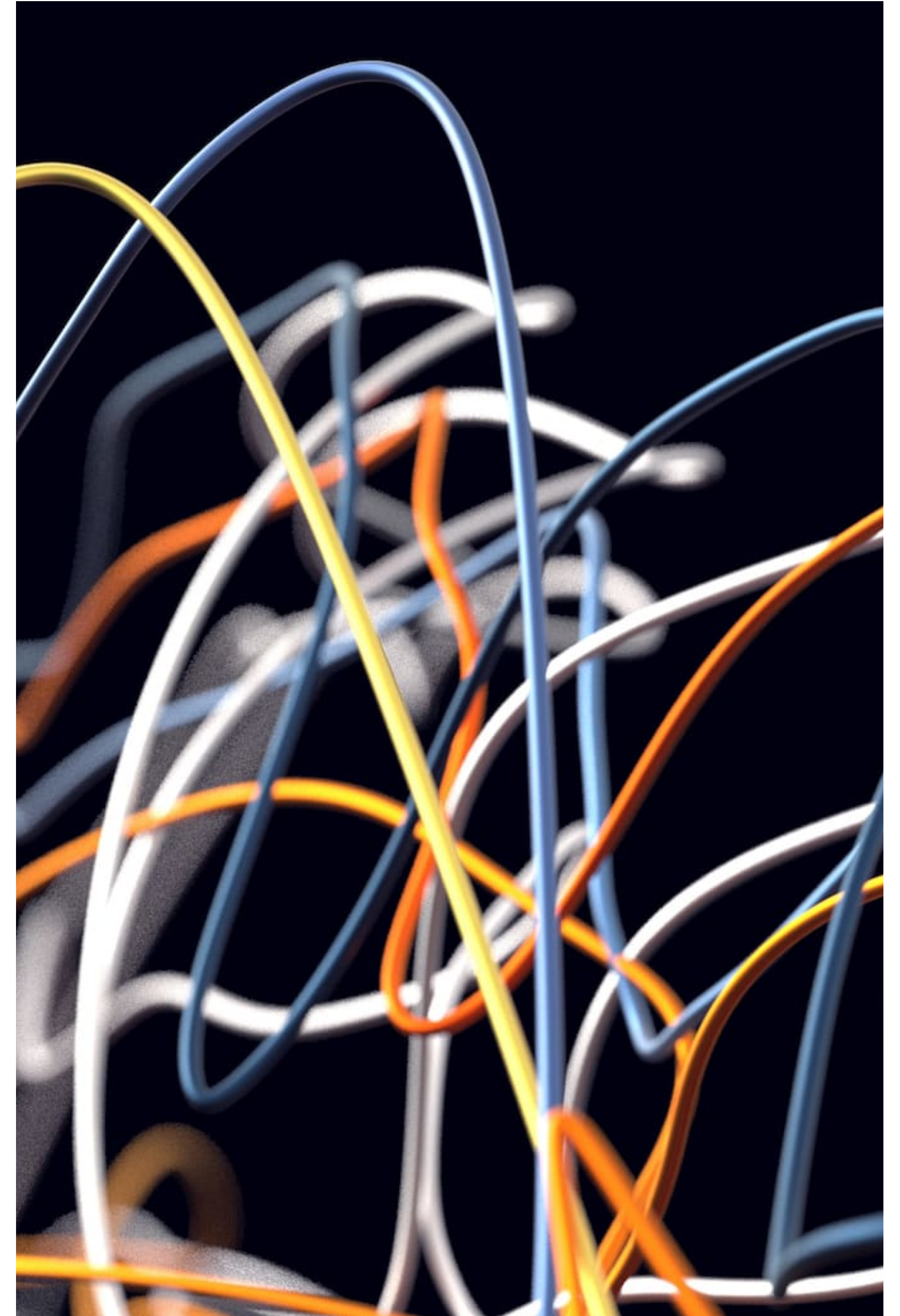
Akzidentelle Komplexität

- Technisch / Organisatorisch selbst erzeugt
- „Architektur Cargo Culting“
- Verteilte Systeme ohne Not
- Premature optimization
- YAGNI



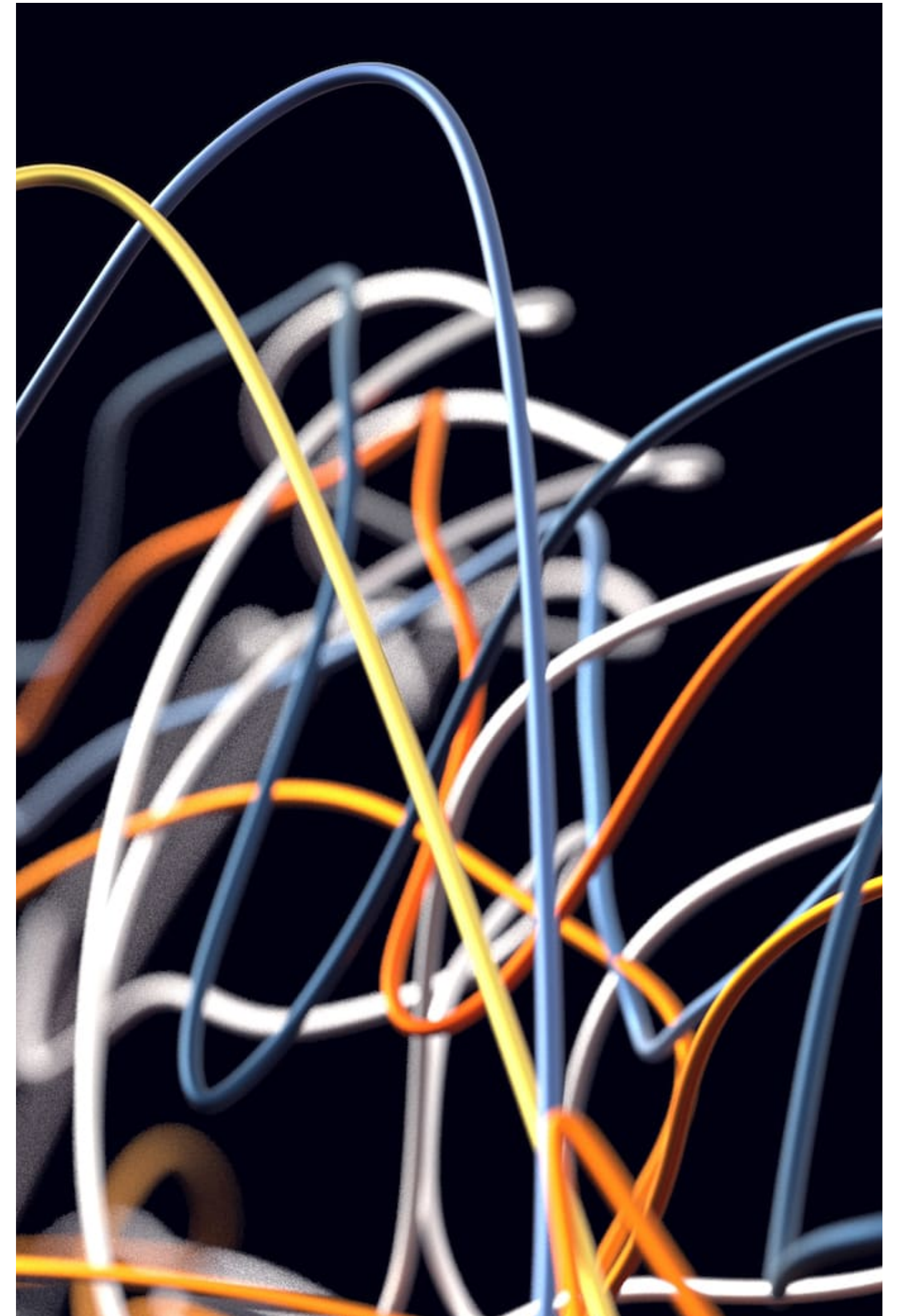
Akzidentelle Komplexität

- Technisch / Organisatorisch selbst erzeugt
- „Architektur Cargo Culting“
- Verteilte Systeme ohne Not
- Premature optimization
- YAGNI
- Conference Driven Development



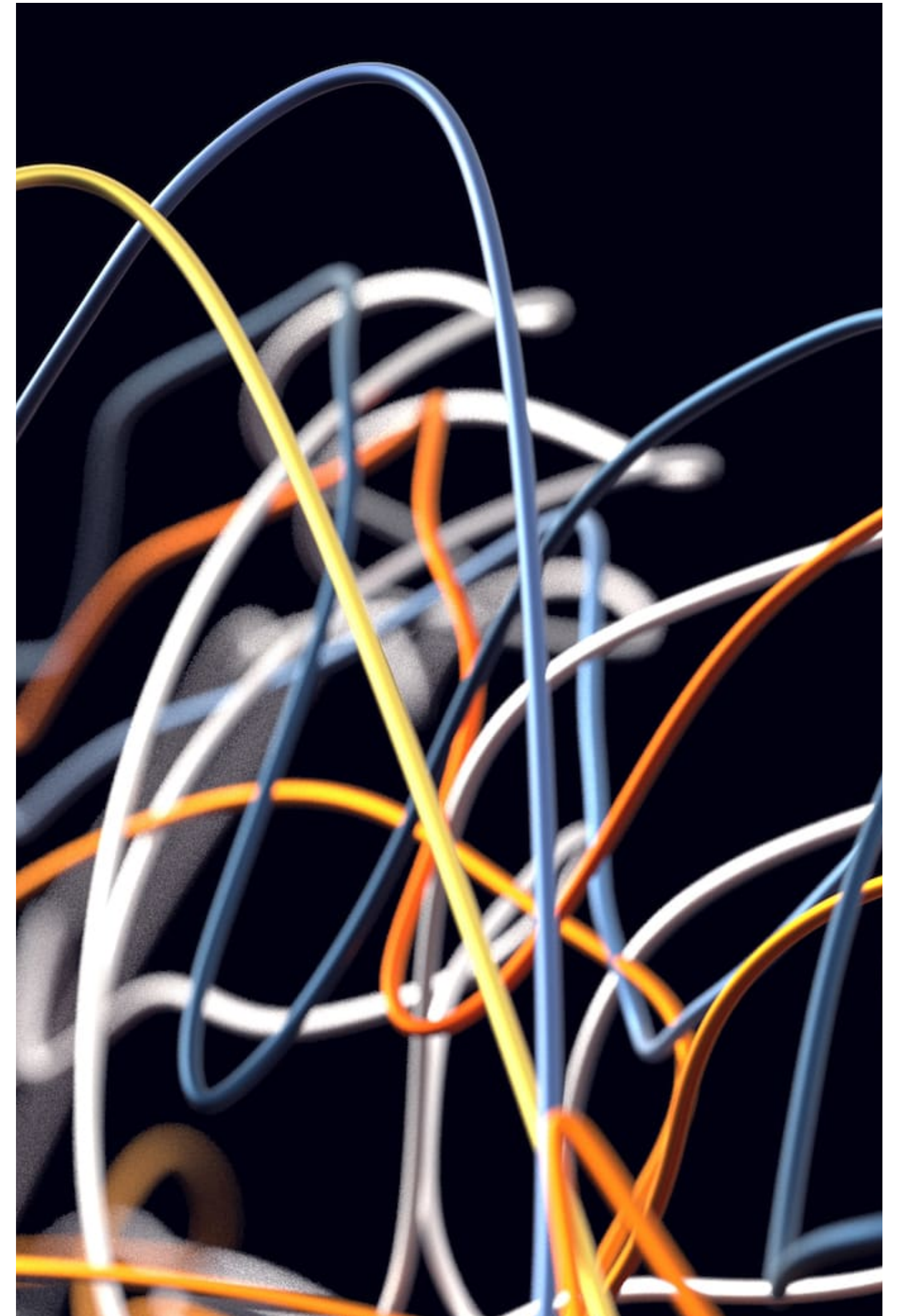
Akzidentelle Komplexität

- Technisch / Organisatorisch selbst erzeugt
- „Architektur Cargo Culting“
- Verteilte Systeme ohne Not
- Premature optimization
- YAGNI
- Conference Driven Development
- Technik > Fachlichkeit



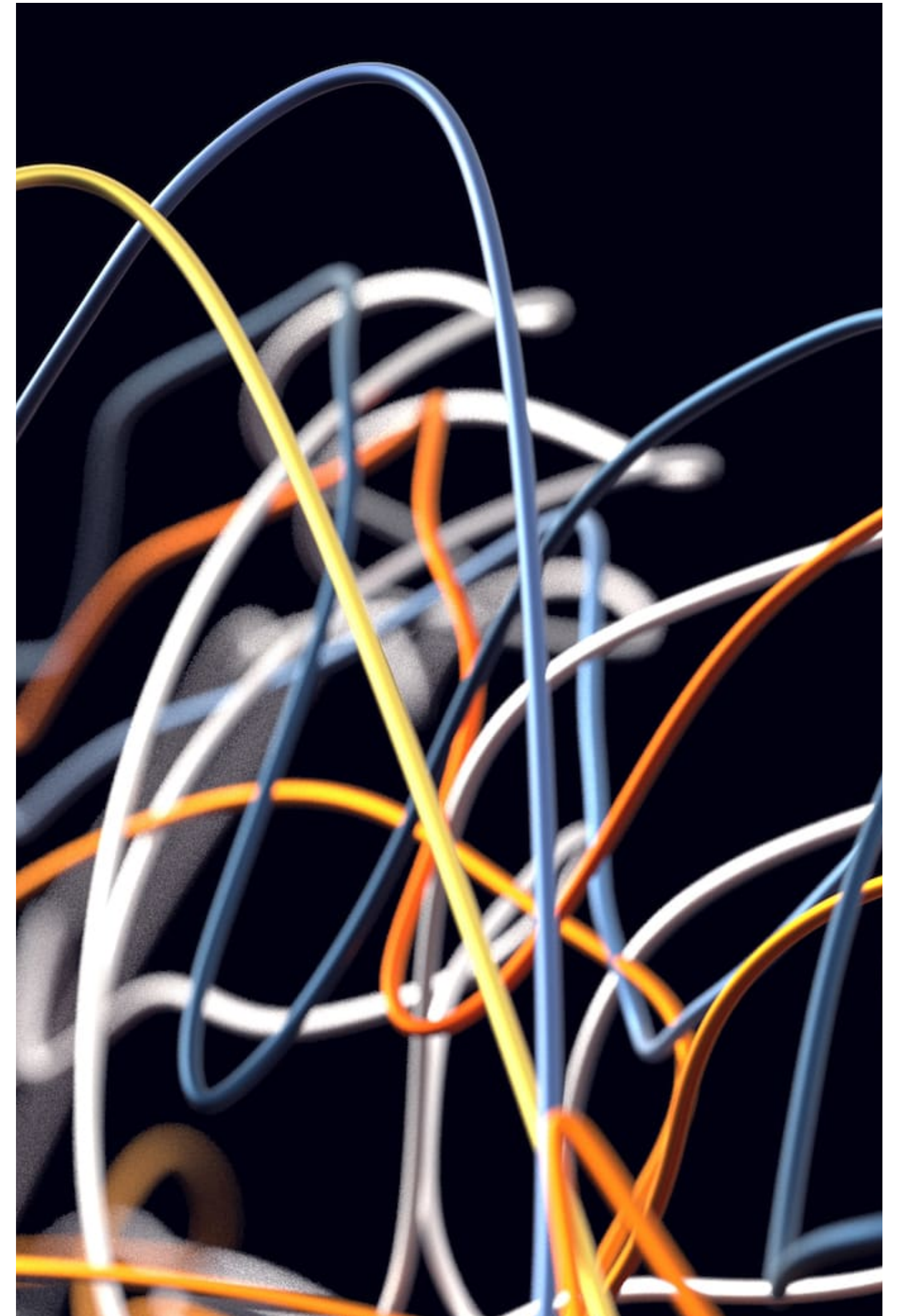
Akzidentelle Komplexität

- Technisch / Organisatorisch selbst erzeugt
- „Architektur Cargo Culting“
- Verteilte Systeme ohne Not
- Premature optimization
- YAGNI
- Conference Driven Development
- Technik > Fachlichkeit
- Obacht: einiges davon könnte auch nur unnötig kompliziert sein, nicht komplex ;-)



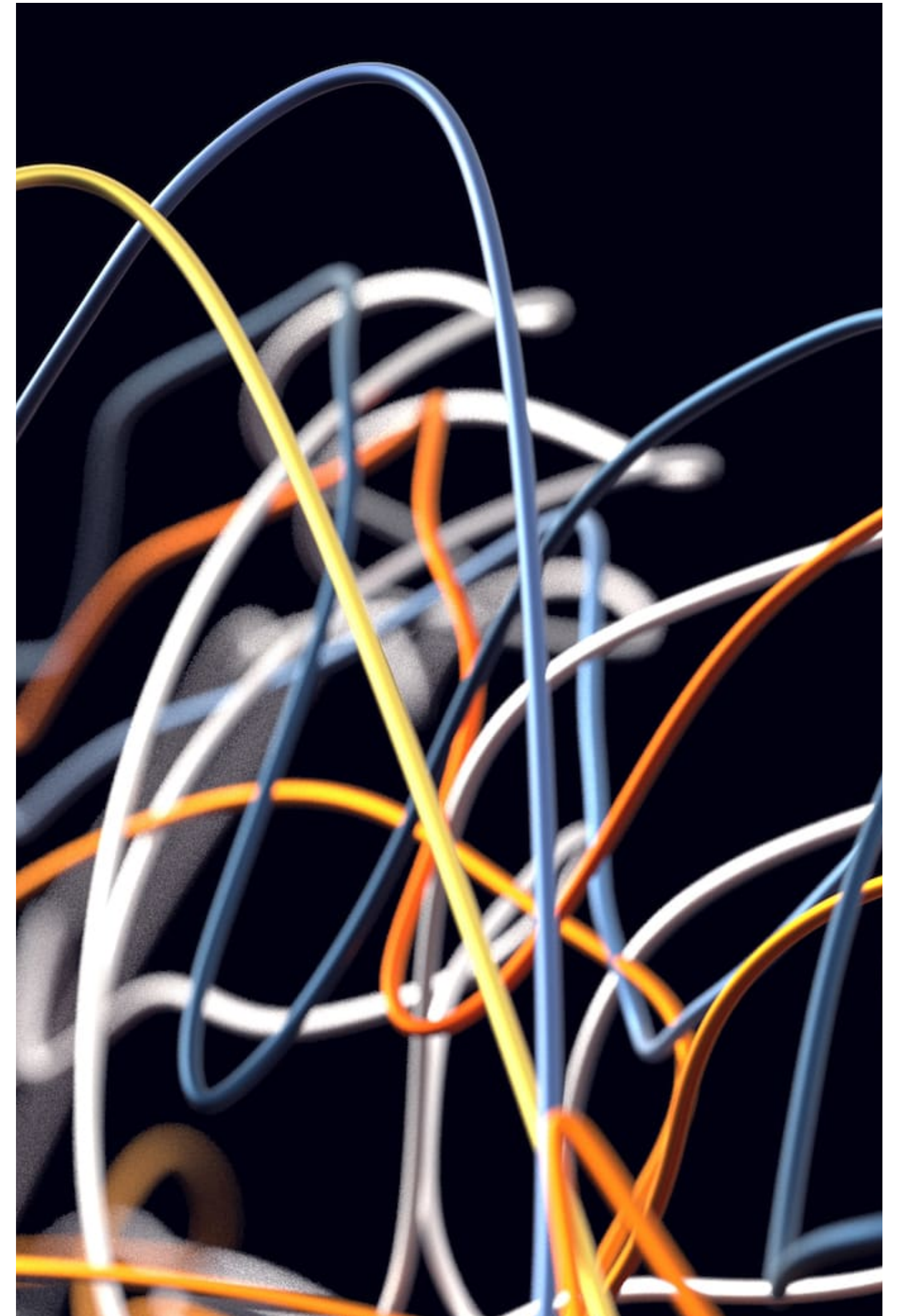
Akzidentelle Komplexität

- Technisch / Organisatorisch selbst erzeugt
- „Architektur Cargo Culting“
- Verteilte Systeme ohne Not
- Premature optimization
- YAGNI
- Conference Driven Development
- Technik > Fachlichkeit
- Obacht: einiges davon könnte auch nur unnötig kompliziert sein, nicht komplex ;-)
- MIMF



Akzidentelle Komplexität

- Technisch / Organisatorisch selbst erzeugt
- „Architektur Cargo Culting“
- Verteilte Systeme ohne Not
- Premature optimization
- YAGNI
- Conference Driven Development
- Technik > Fachlichkeit
- Obacht: einiges davon könnte auch nur unnötig kompliziert sein, nicht komplex ;-)
- MIMF



Essenz

Essenzielle Komplexität

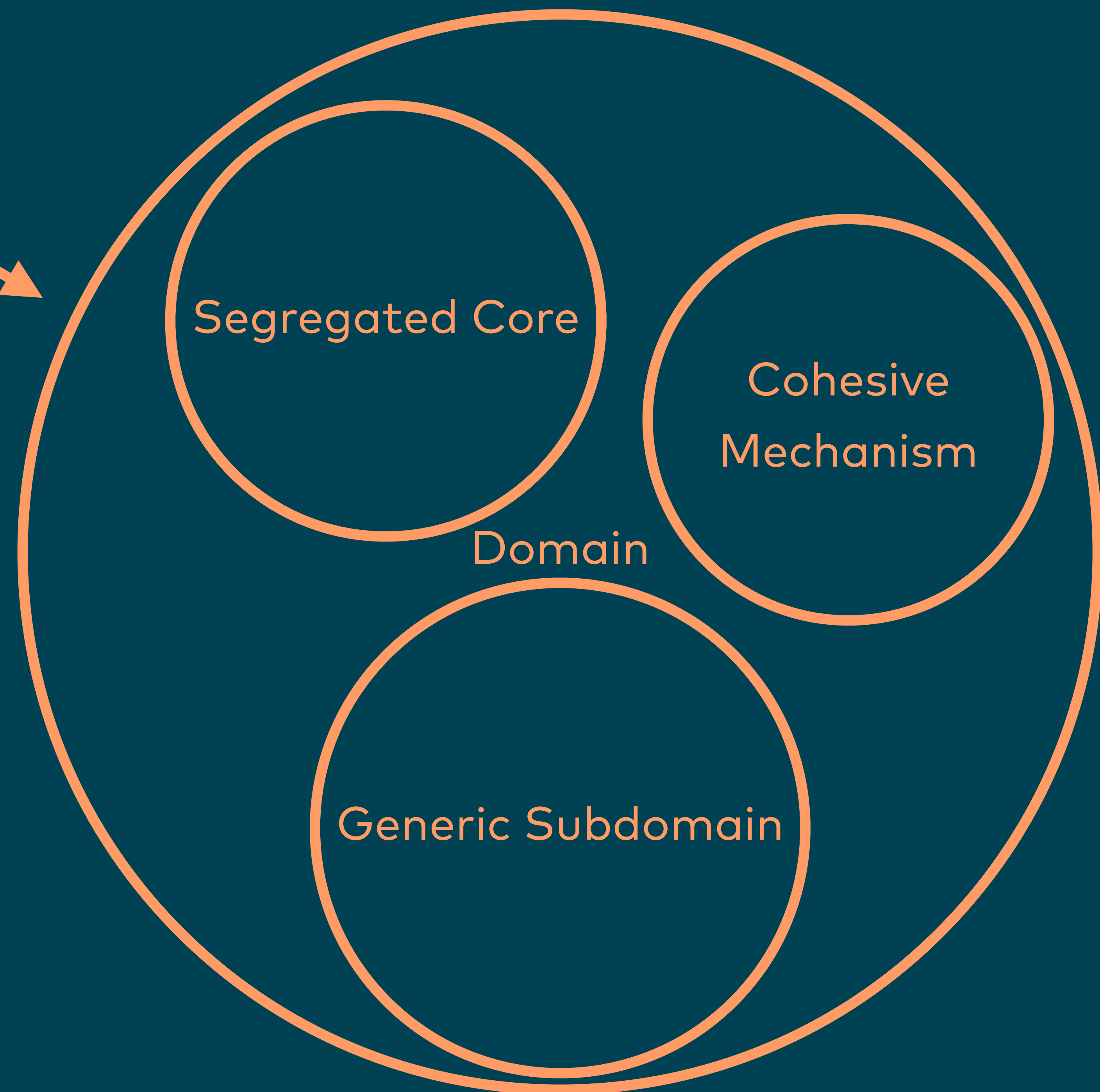
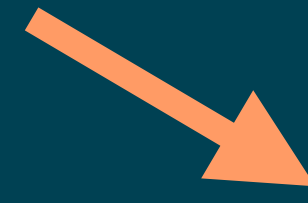
- Lässt sich nicht reduzieren*
- (*Außer durch ein anderes Fachmodell)
- Lässt sich nur beherrschbar(er) machen



Domain Distillation

Die Suche nach dem fachlichen Kern

Core Domain

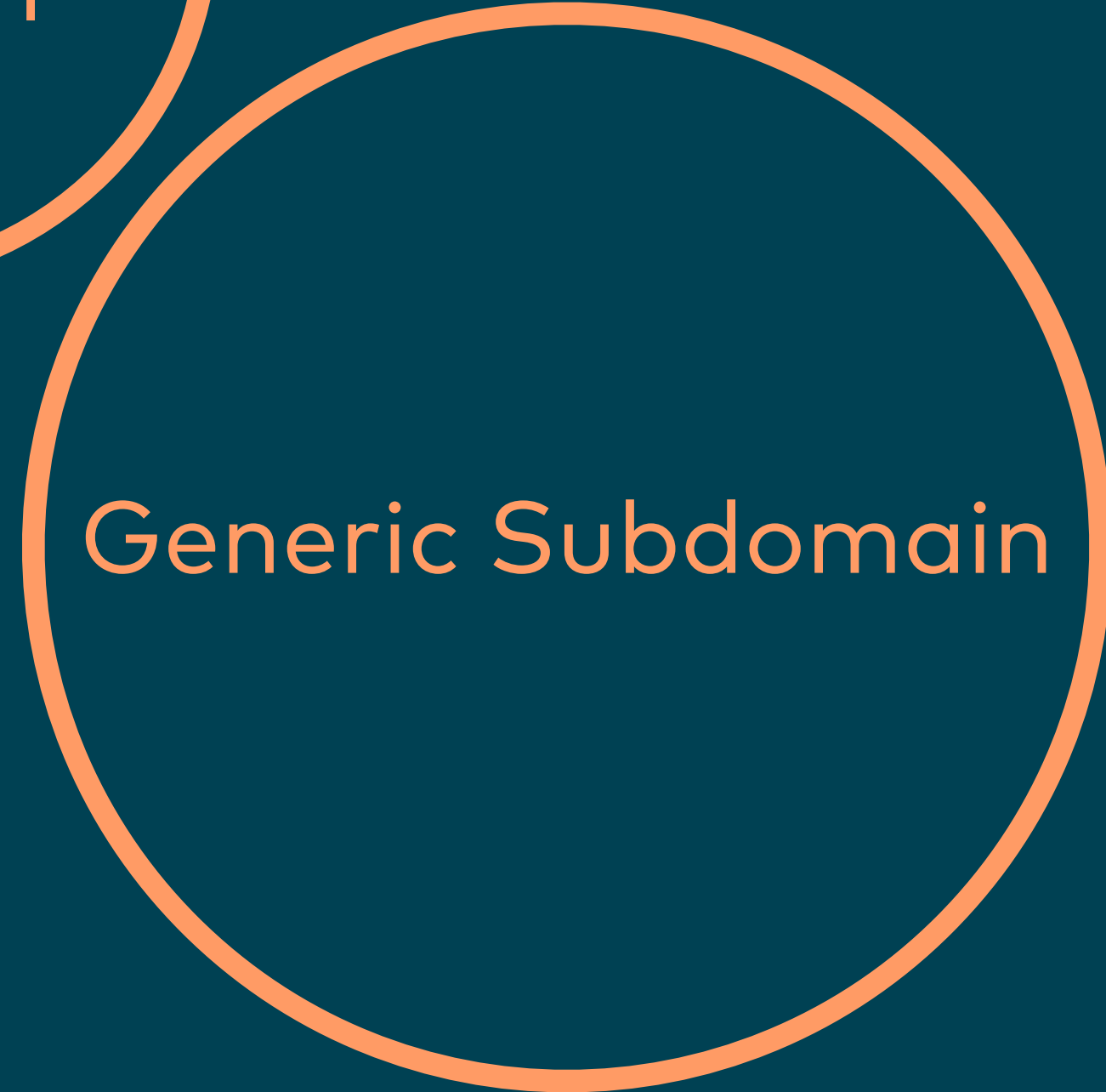
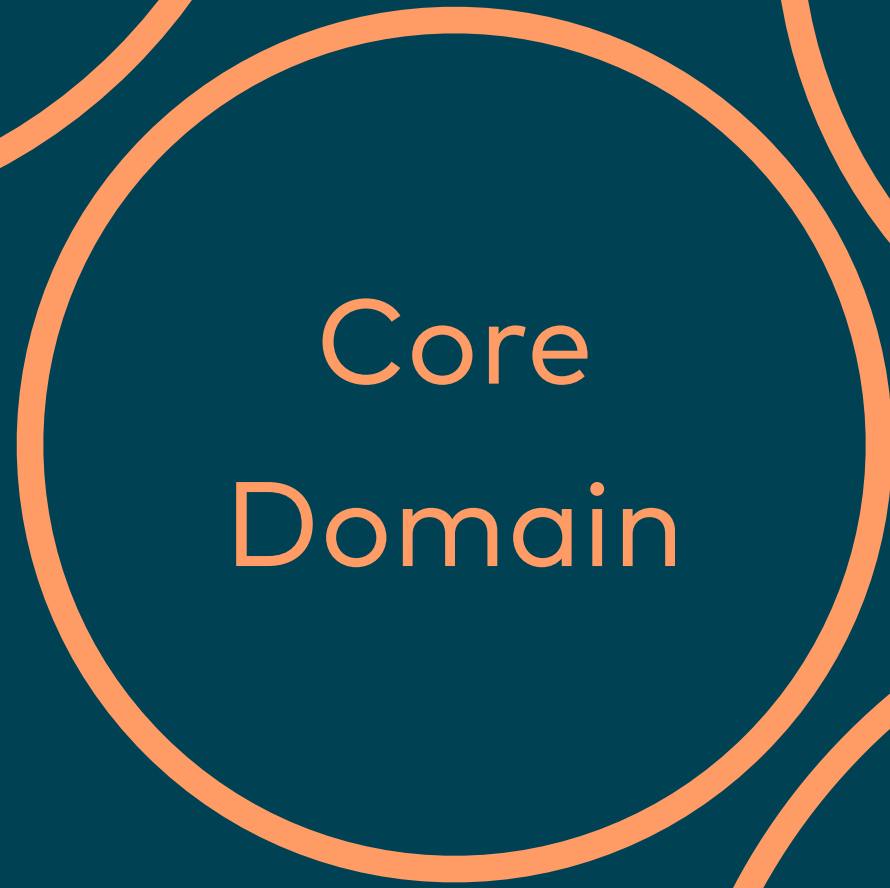


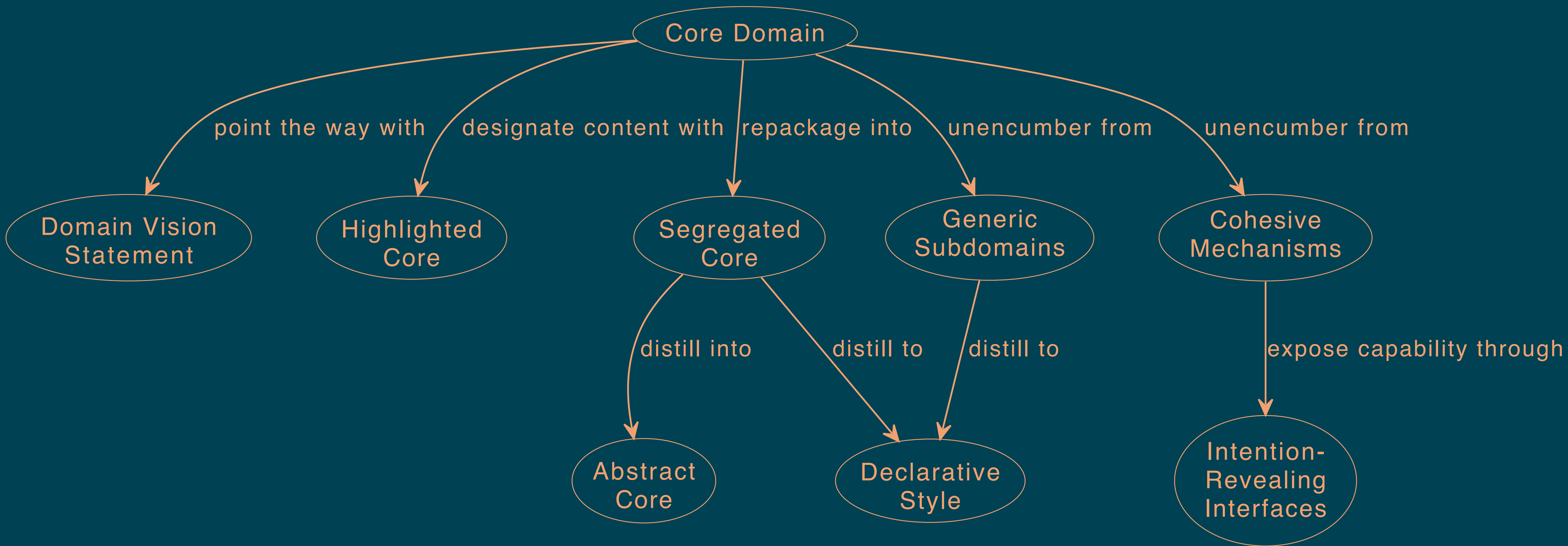
Segregated Core

Cohesive
Mechanism

Domain

Generic Subdomain





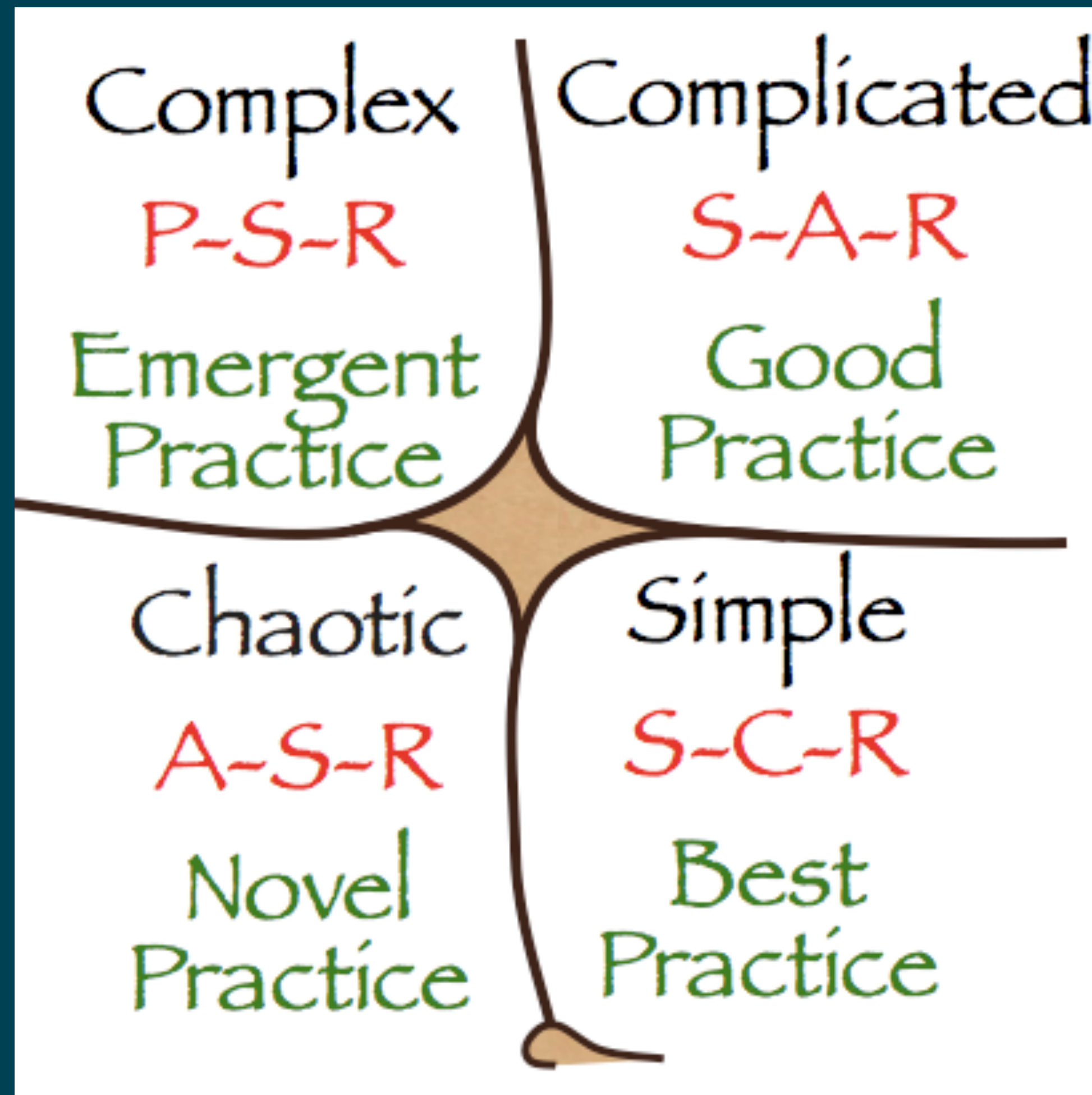
Lösungsansätze

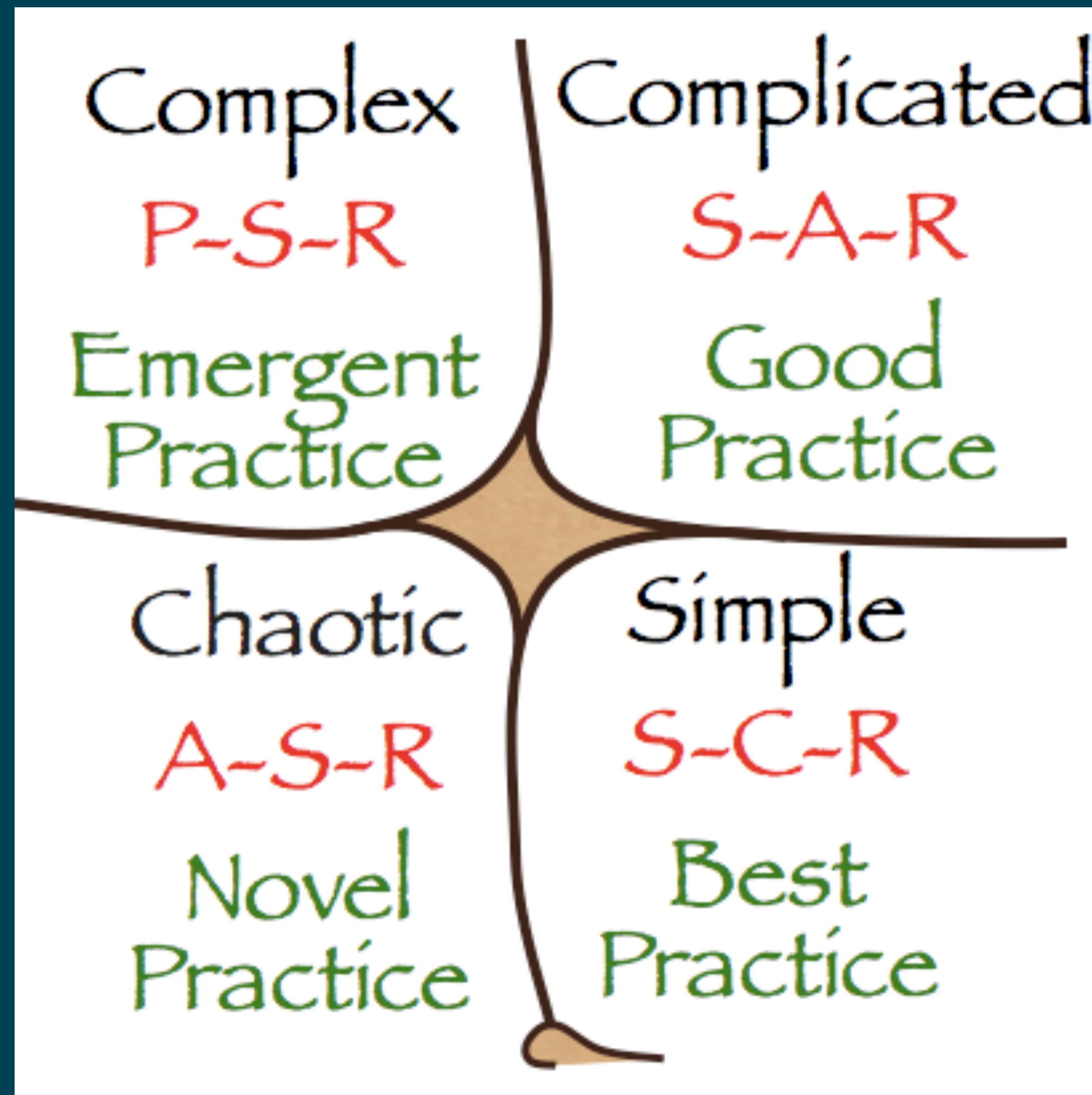
Erst das Organisatorische

Welches Werkzeug muss ich aus dem Kasten ziehen?

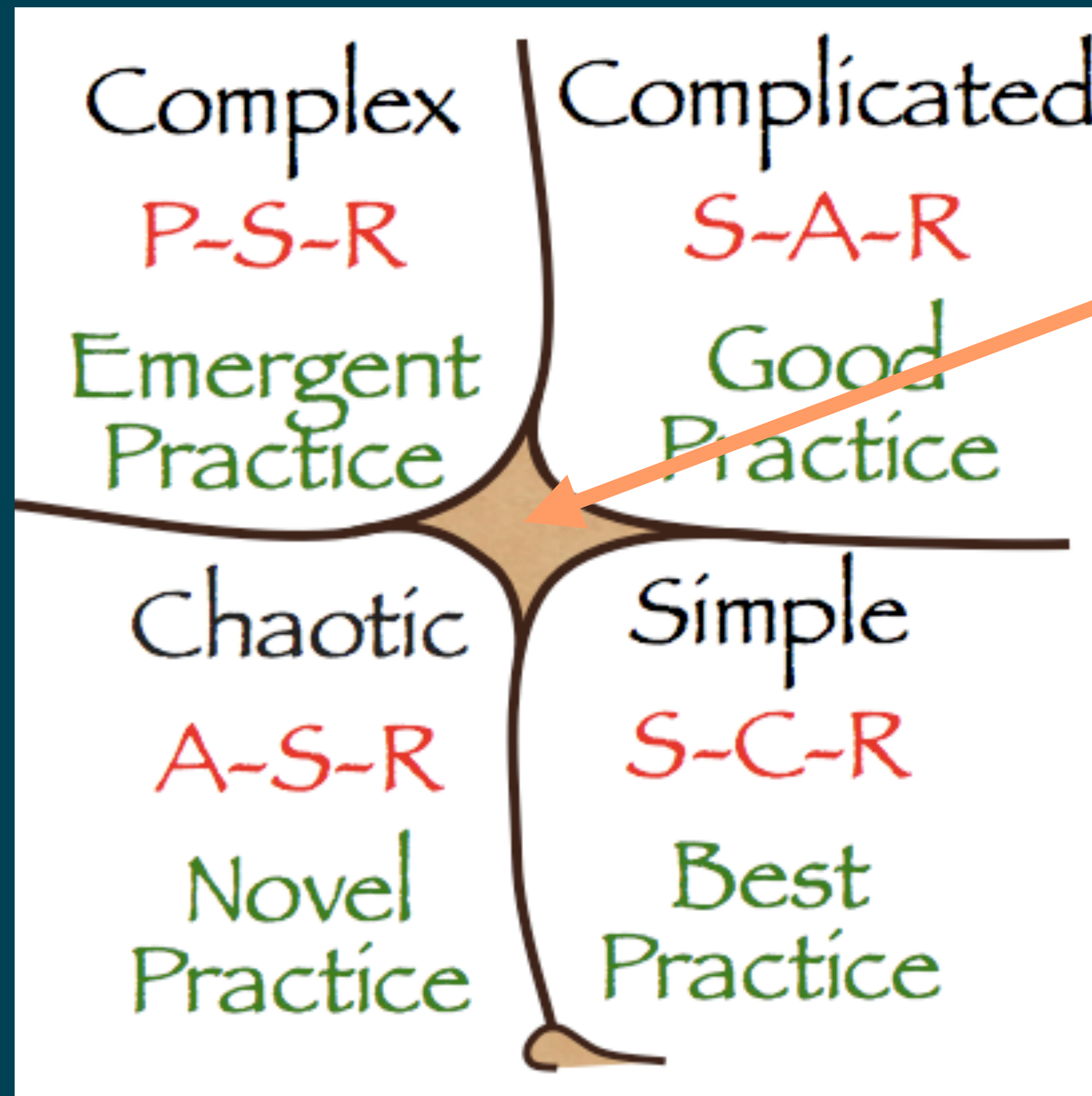
CYNEFIN

Will ich überhaupt ein komplexes Problem lösen?





Disorder



Disorder

Simple

Sense

Categorise

Response

Einfach halten

- KISS
- CRUD is ok
- Aufwendige Aggregate für Invarianten evtl. Over-Engineering
- Standard-Lib(s) ohne aufwendige fachliche Abstraktion verwenden
 - Quartz-Scheduler
 - Resilience4j

Complicated

Sense

Analyse

Response

Ursache und Wirkung...

- ... meinen wir verstanden zu haben
- Es gibt mehre logische und erklärbare Lösungsansätze
- Wir können direkt in den taktischen DDD-Werkzeugkasten greifen
- Komplexe fachliche Regeln existieren, aber können klar mit Fachexperten modelliert werden (Bounded Contexts, Aggregate, Value Objects, Entities und Regeln als Policies, usw.)

Complex Probe Sense Response

Strategic DDD to the rescue

- Direktes Modellieren in Software ist verfrüht, zu vieles ist noch unklar
- Domäne explorieren und iterativ umsetzen
- Eventstorming (Process- / Design-Level)
- Prototyping & Agile Entwicklung
- Event-Sourcing für schrittweise Evolution und Transparenz von Prozessen

Chaotic Act Sense Response

Handeln! Jetzt!

- Bei production incidents mache ich kein Eventstorming ;-)
- Handeln, stabilisieren
- Monitoring, Logging erweitern, Rollback
- Dann Informationen sammeln und Ergebnisse bewerten
- Wenn wieder in kompliziertem oder komplexem Fahrwasser, s. o.

Disorder

Wo stehen wir?

- Unklar in welchem Quadranten wir uns befinden
- Unterschiedliche Ansichten bei den Stakeholdern
- Collaborative Workshops, um die Themen zu sortieren
- Big-Picture-Eventstorming, aber vieles andere ist möglich

MIMF

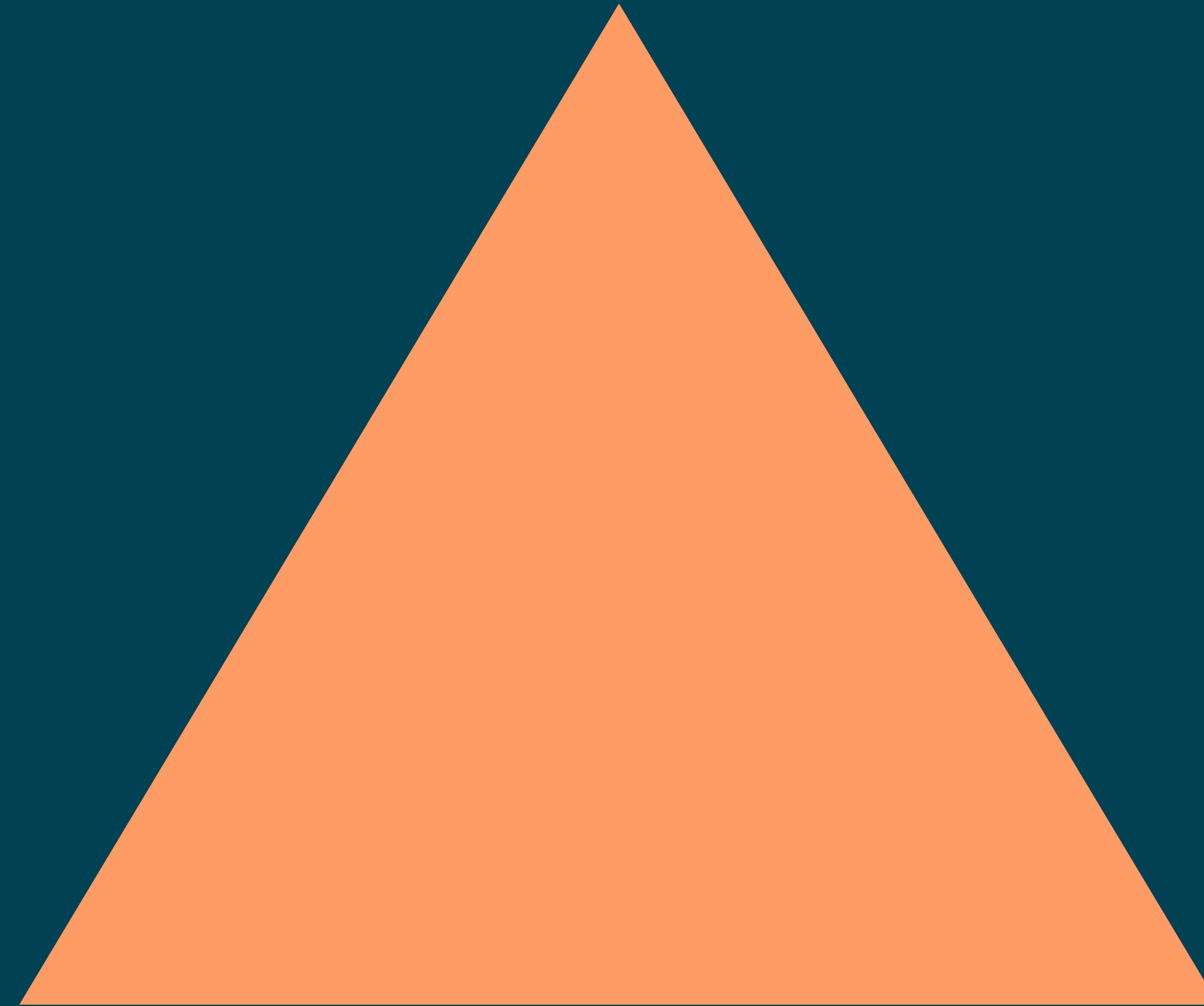
Minimal Implementation Model Fallacy

Warren Thorngates Postulat

Warren Thorngates Postulat

Der angemessenen Komplexität

Allgemein



Einfach

Präzise

Quelle: [Yousef Mehrdad Bibalan](#)

Quelle: [Thorngates Postulat der angemessenen Komplexität](#)

Allgemein



Komplex

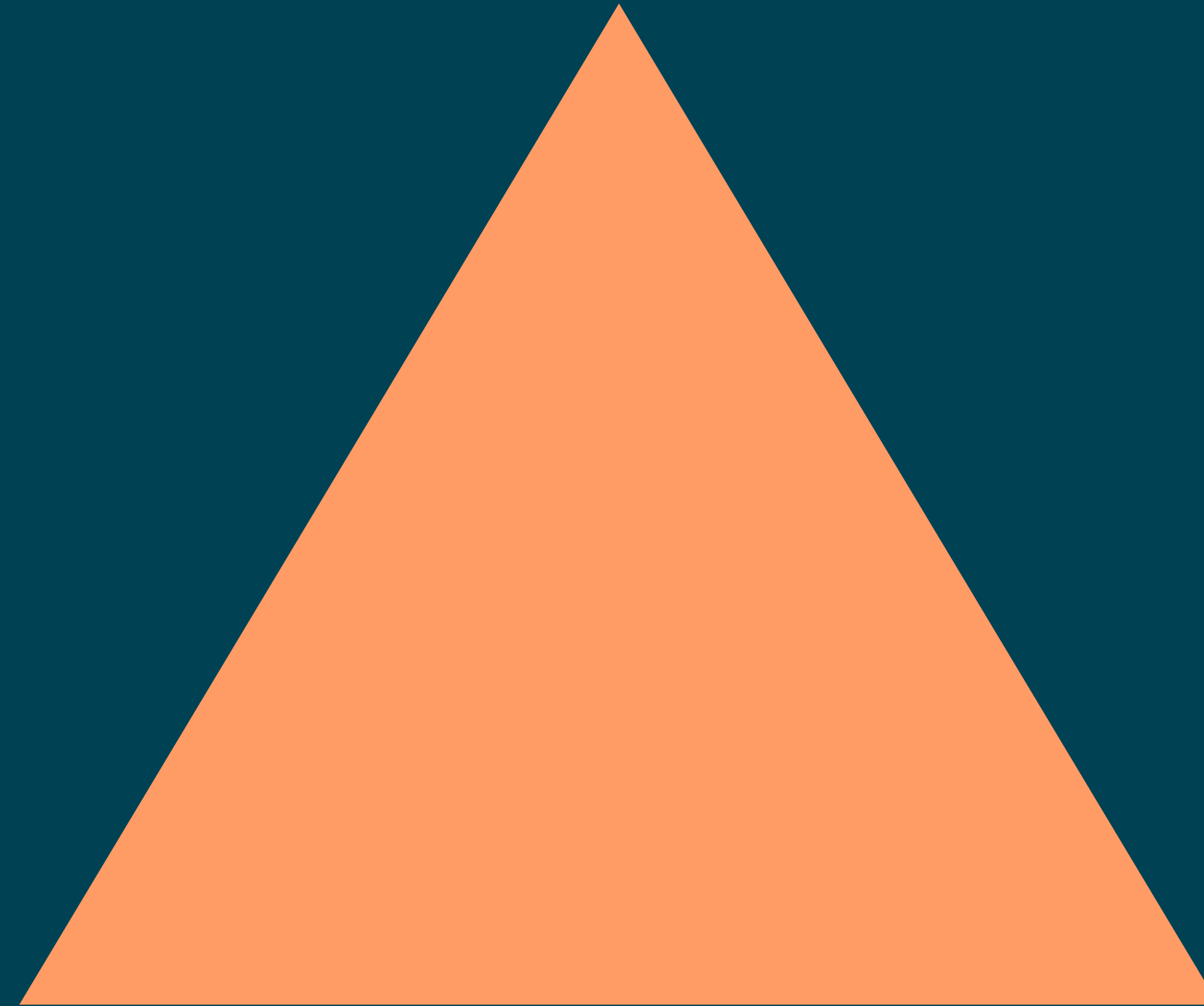
Einfach

Präzise

Quelle: [Yousef Mehrdad Bibalan](#)

Quelle: [Thorngates Postulat der angemessenen Komplexität](#)

Allgemein



Einfach

Präzise

Quelle: [Yousef Mehrdad Bibalan](#)

Quelle: [Thorngates Postulat der angemessenen Komplexität](#)

Allgemein



Core Domain

Einfach

Präzise

Quelle: [Yousef Mehrdad Bibalan](#)

Quelle: [Thorngates Postulat der angemessenen Komplexität](#)

Allgemein



Core Domain

Einfach

Supporting
Subdomain

Präzise

Quelle: [Yousef Mehrdad Bibalan](#)

Quelle: [Thorngates Postulat der angemessenen Komplexität](#)

Allgemein

Generic Subdomain

Core Domain

Supporting
Subdomain

Einfach

Präzise

Quelle: [Yousef Mehrdad Bibalan](#)

Quelle: [Thorngates Postulat der angemessenen Komplexität](#)

Denkt an MIMF

Denkt an MIMF

Wenn „Allgemein“ mit „Präzise“ kombiniert wird, wird's
komplex

Allgemein



Einfach

Präzise

Quelle: [Yousef Mehrdad Bibalan](#)

Quelle: [Thorngates Postulat der angemessenen Komplexität](#)

Allgemein



Einfach

Präzise

Quelle: [Yousef Mehrdad Bibalan](#)

Quelle: [Thorngates Postulat der angemessenen Komplexität](#)

**„A model is a lie that helps you see the
truth“**

Howard Skipper

"All models are wrong but some are useful"

George Box

(Fachliche) Komplexität ist wie Luft im Ballon

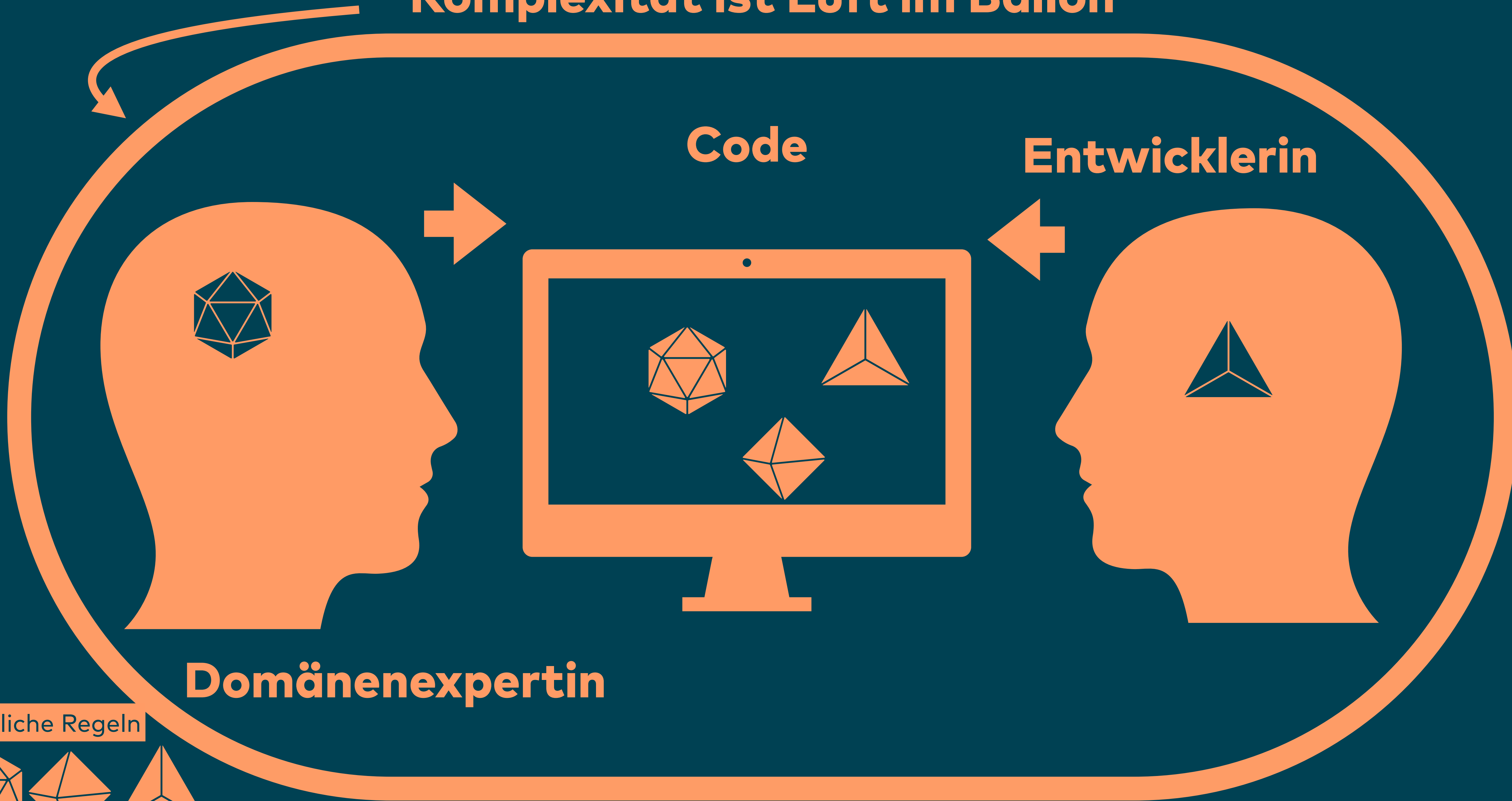
Komplexität ist Luft im Ballon

Code

Entwicklerin

Domänenexpertin

Fachliche Regeln



**MIMF in kurz: Komplizierter Code wird als
komplex missverstanden und komplexer
sieht gern unkompliziert aus**

**Wenn eine fachliche Änderung keine Codeänderung
erfordert, ist der Code nicht kompliziert, aber dafür
eine komplexe Blackbox**

Fazit

**In Software versuchen wir häufig
Systeme „von außen“ zu konstruieren.
Wir erstellen ein Organigramm und
erwarten, dass es Wirklichkeit wird.**

**Welches Organigramm hat je die
informelle Organisationsstruktur
abgebildet?**

Welches Organigramm hat je die informelle Organisationsstruktur abgebildet?

... die dafür sorgt, dass Unternehmen überhaupt funktionieren

Soziale Systeme stabilisieren sich selbst

Soziale Systeme stabilisieren sich selbst

**Software kann das nicht. Wir müssen (kontinuierlich)
ran.**

Viele kleine, gut abgegrenzte Dinge mit klaren Schnittstellen werden zu einem komplizierten Ganzen.

Viele kleine, gut abgegrenzte Dinge mit klaren Schnittstellen werden zu einem komplizierten Ganzen.

Kompliziert heißt aber: nachvollziehbar.

Ein perfektes Modell der Realität wäre die Realität.

Und die Realität ist unendlich komplex.

Ein perfektes Modell der Realität wäre die Realität.

Und die Realität ist unendlich komplex.

Modelle sind toll. Lasst uns viele kleine haben, die wir gut verstehen können (aber nicht immer gleichzeitig auch müssen)

Ein nützliches Fachmodell finden.

Das kleinstmögliche Modell der Domäne heraus destillieren.

Diese Essenz iterativ implementieren.

Akzidentelle Komplexität vermeiden.

**Große Blackboxen
erzeugen Unsicherheiten
bei Menschen**

**Viele kleine Blackboxen
lassen sich beherrschen
und vermitteln Kontrolle**

Aber nur mit klaren Schnittstellen

Weitere Links und Mentions

- Eric Evans: <https://www.amazon.de/Domain-Driven-Design-Tackling-Complexity-Software/dp/0321125215>
- Margot Berghaus - Luhmann leicht gemacht
- Frederick P. Brooks, Jr. - The Mythical Man-Month
- https://de.wikipedia.org/wiki/Thorngates_Postulat_der_angemessenen_Komplexit%C3%A4t

Feedback? Contact!



Fabian Walther
fabian.walther@innoq.com
innoq.social/@fabian

LinkedIn



innoQ Deutschland GmbH

Krischerstr. 100
40789 Monheim
+49 2173 3366-0

Ohlauer Str. 43
10999 Berlin

Ludwigstr. 180E
63067 Offenbach

Kreuzstr. 16
80331 München

Hermannstrasse 13
20095 Hamburg

Erftr. 15-17
50672 Köln

Königstorgraben 11
90402 Nürnberg