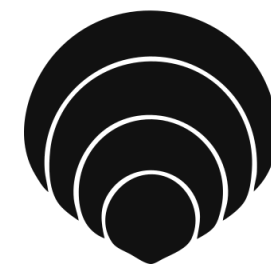


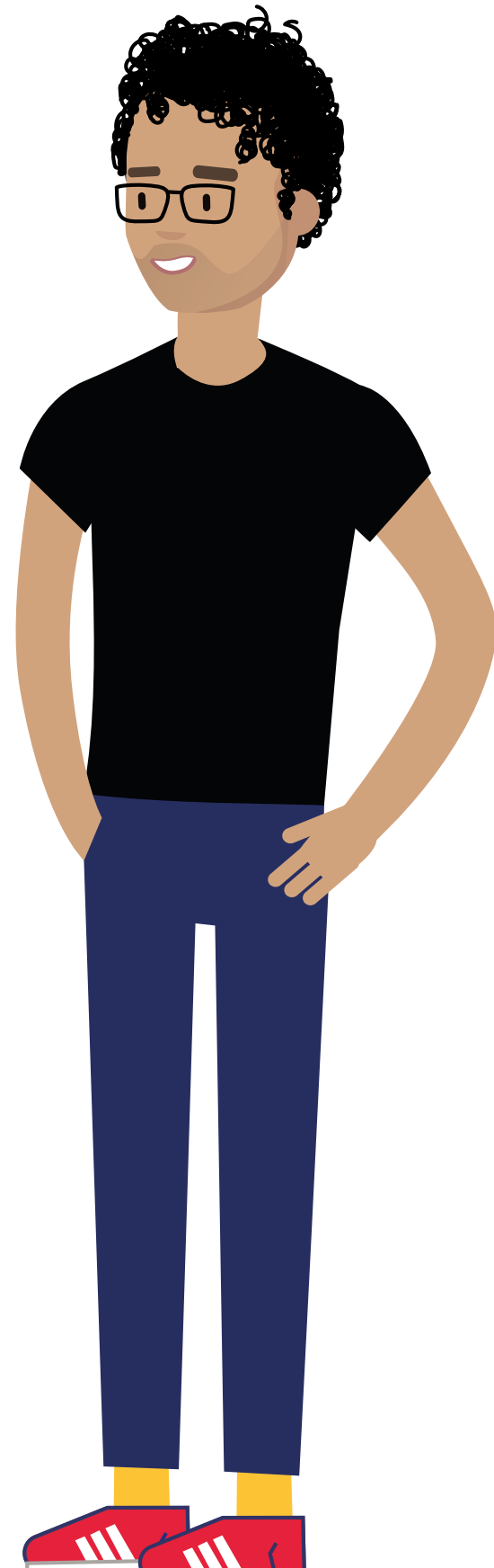
Decoding Paradoxes

Why are many good ideas in
Software Delivery counterintuitive?

Hibri Marzook (he/him)



AGILE ON
THE BEACH



Hibri Marzook

Likes to help software teams deliver fast and enjoy doing it

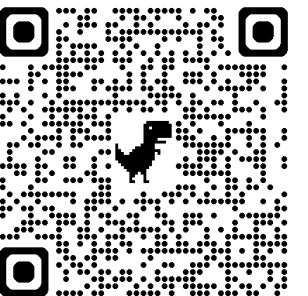
Principal Engineer

Clear.Bank

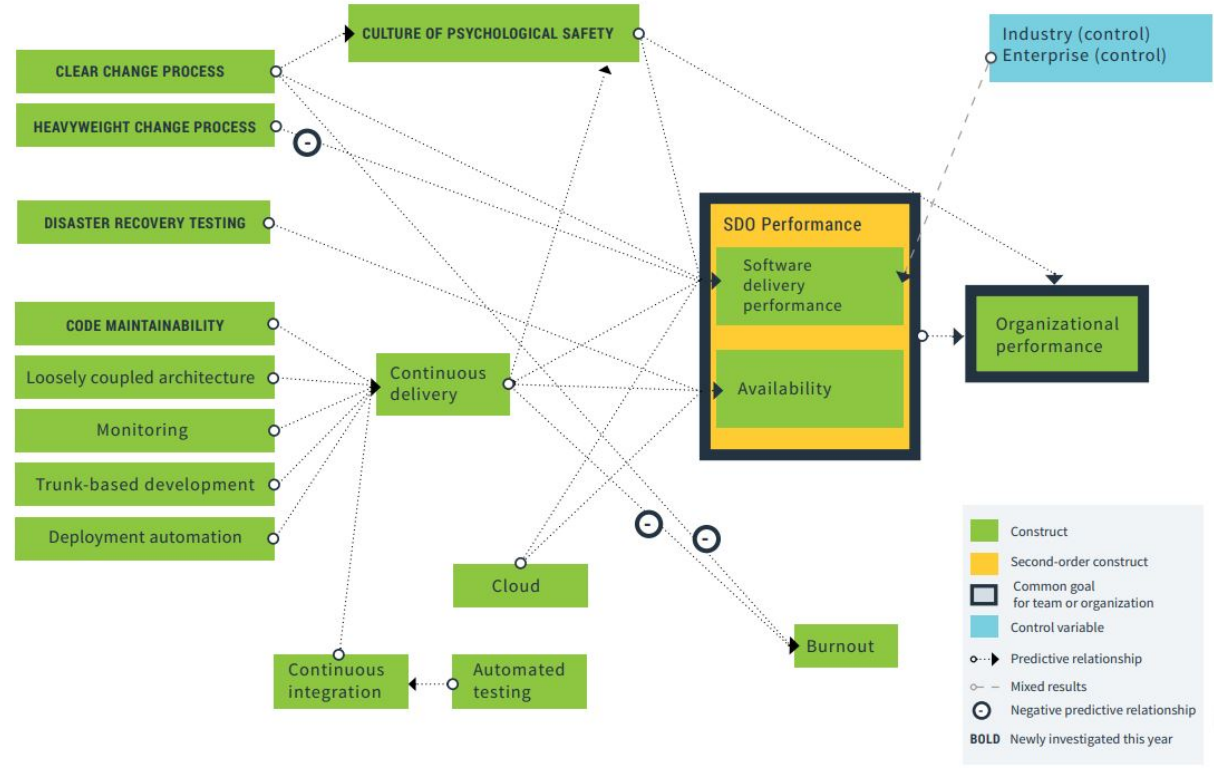
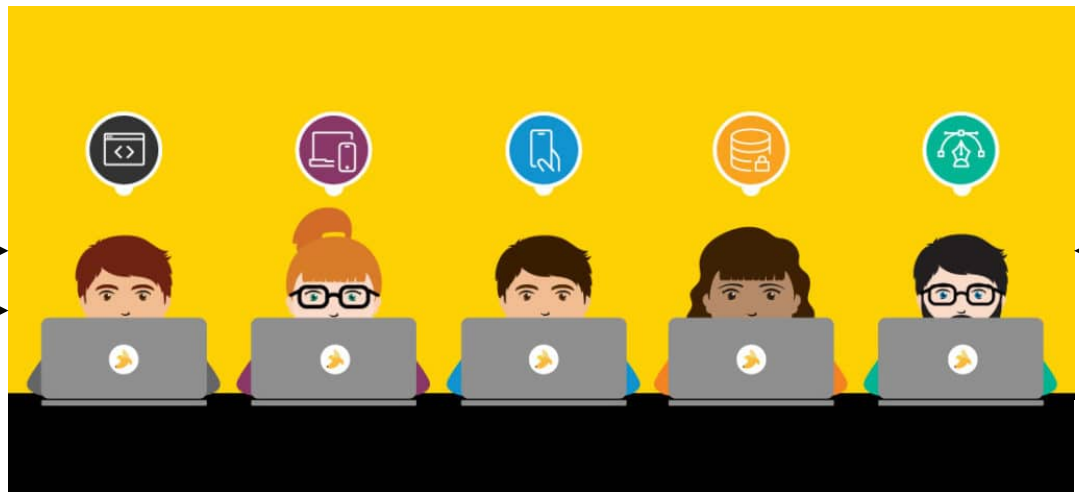
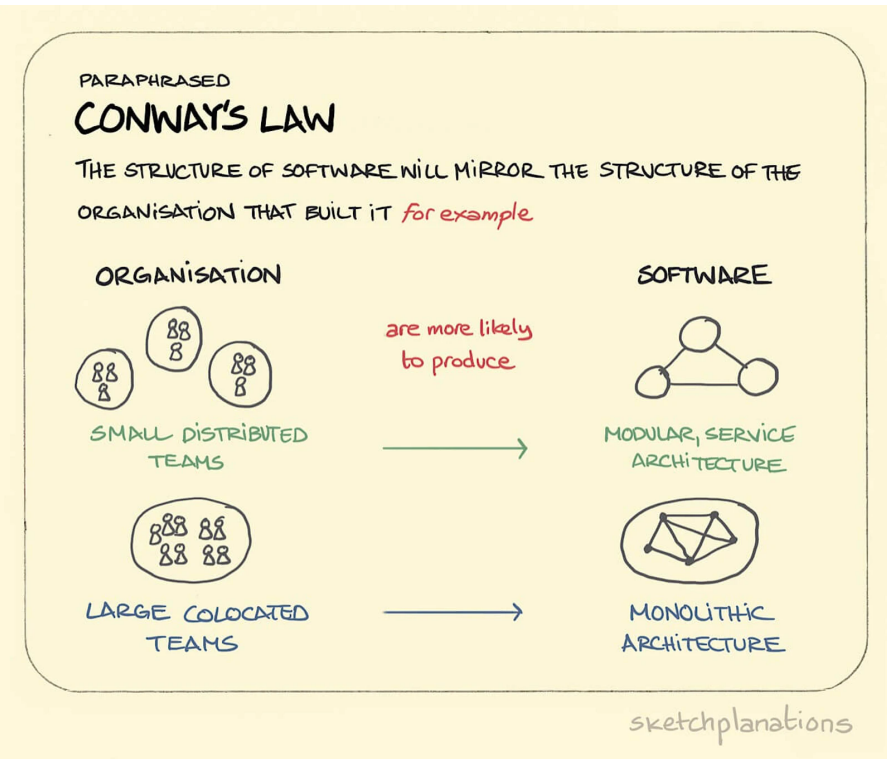
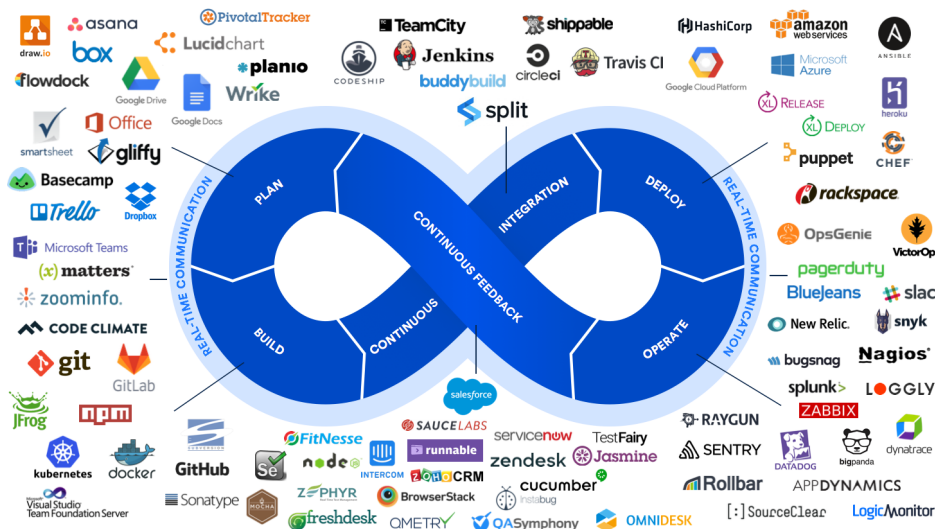
@hibri

www.hibri.net

<https://techhub.social/@hibri>



Software Delivery is complex



**I just want to build
cool stuff**



Is it a Wicked Problem?

*'ill-defined, ambitious and associated with **strong moral, political and professional issues**. Since they are strongly stakeholder dependent, **there is often little consensus about what the problem is**, let alone how to resolve it. Furthermore, wicked problems won't keep still: they are sets of **complex, interacting issues evolving in a dynamic social context**. Often, new forms of wicked problems emerge as a result of trying to understand and solve one of them.'*

'Dilemmas in a General Theory of Planning' (Rittel and Webber, 1973)

Involves changing people's behaviour

Involves changing the relationships between individuals and teams

Involves changing the nature of relationships between individuals and teams

Involves seeing other perspectives

Involves seeing the whole

Software delivery is full of counterintuitive ideas

if it hurts,
do it often

slack time
improves flow

releasing more
often improves
quality

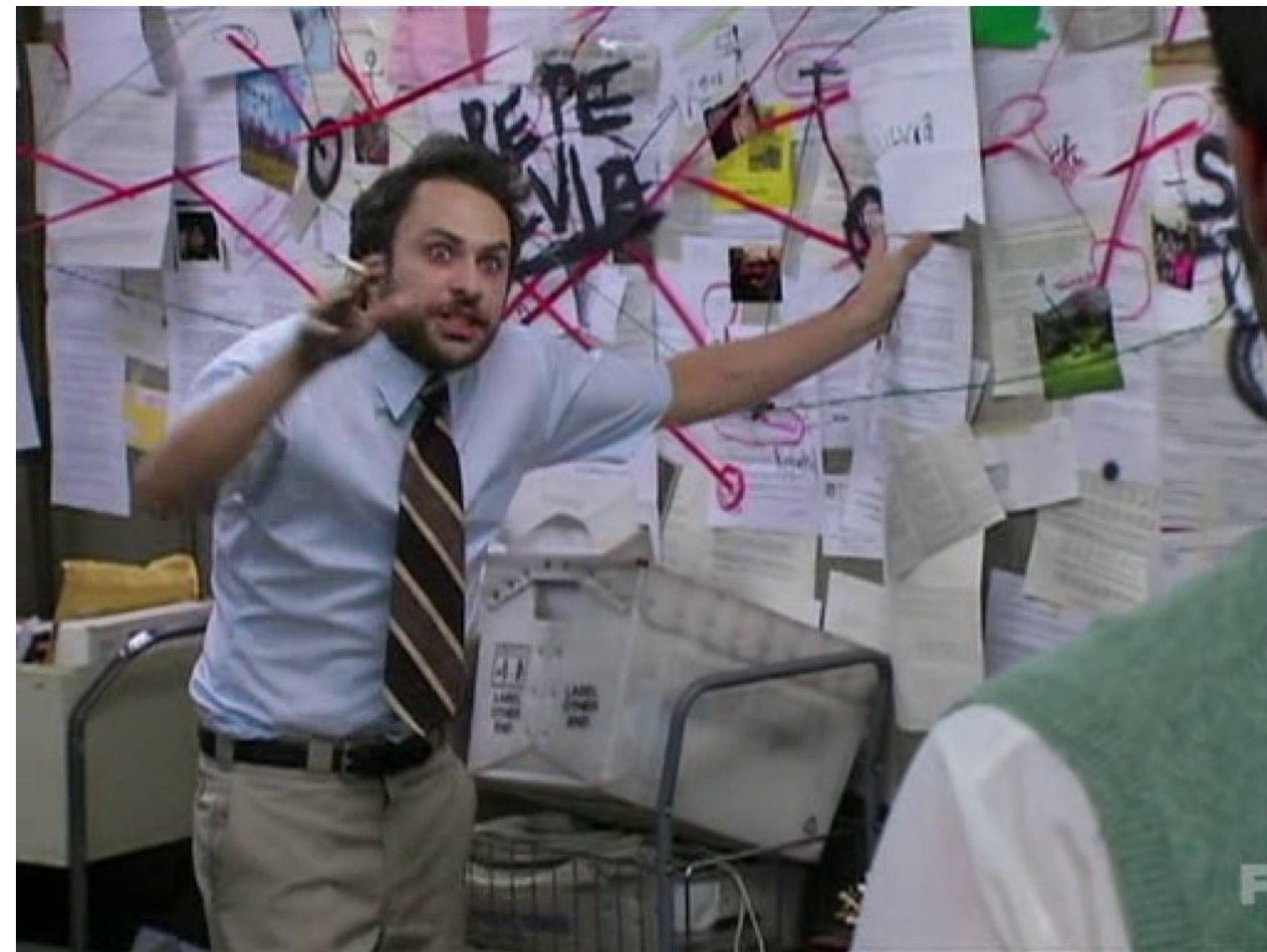
'The simplest thing you can do to improve quality is to deploy more frequently'

Nicole Forsgren

But how does that work?

We need better models to
understand and explain
counterintuitive ideas

When I try to explain how to improve software delivery



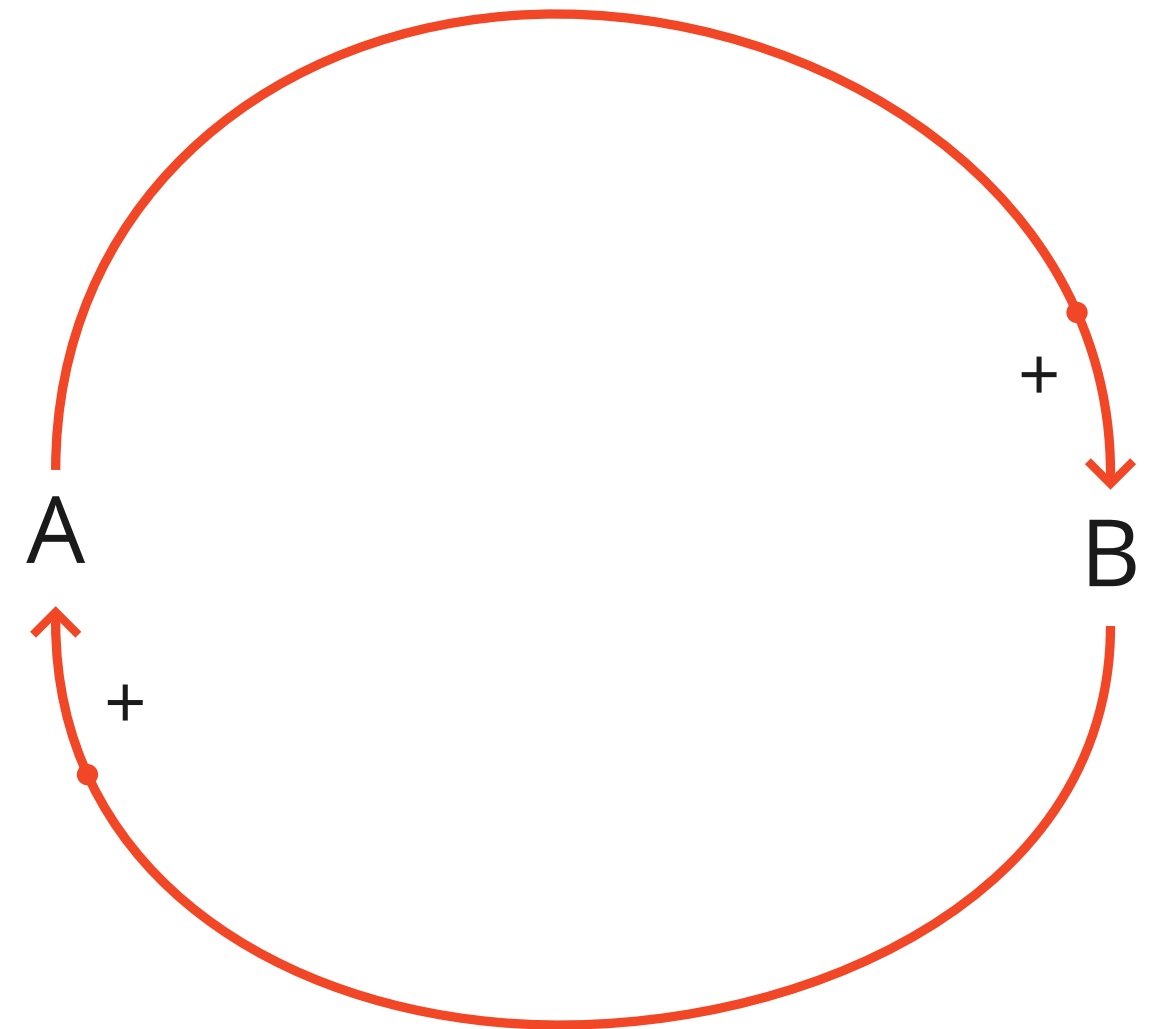
Lets tell the story with causal
loop diagrams

'All models are wrong some models are useful'

George Box

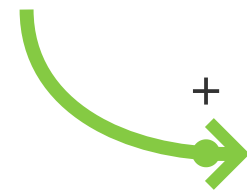
Why use causal loop diagrams?

A influences/increases/affects B



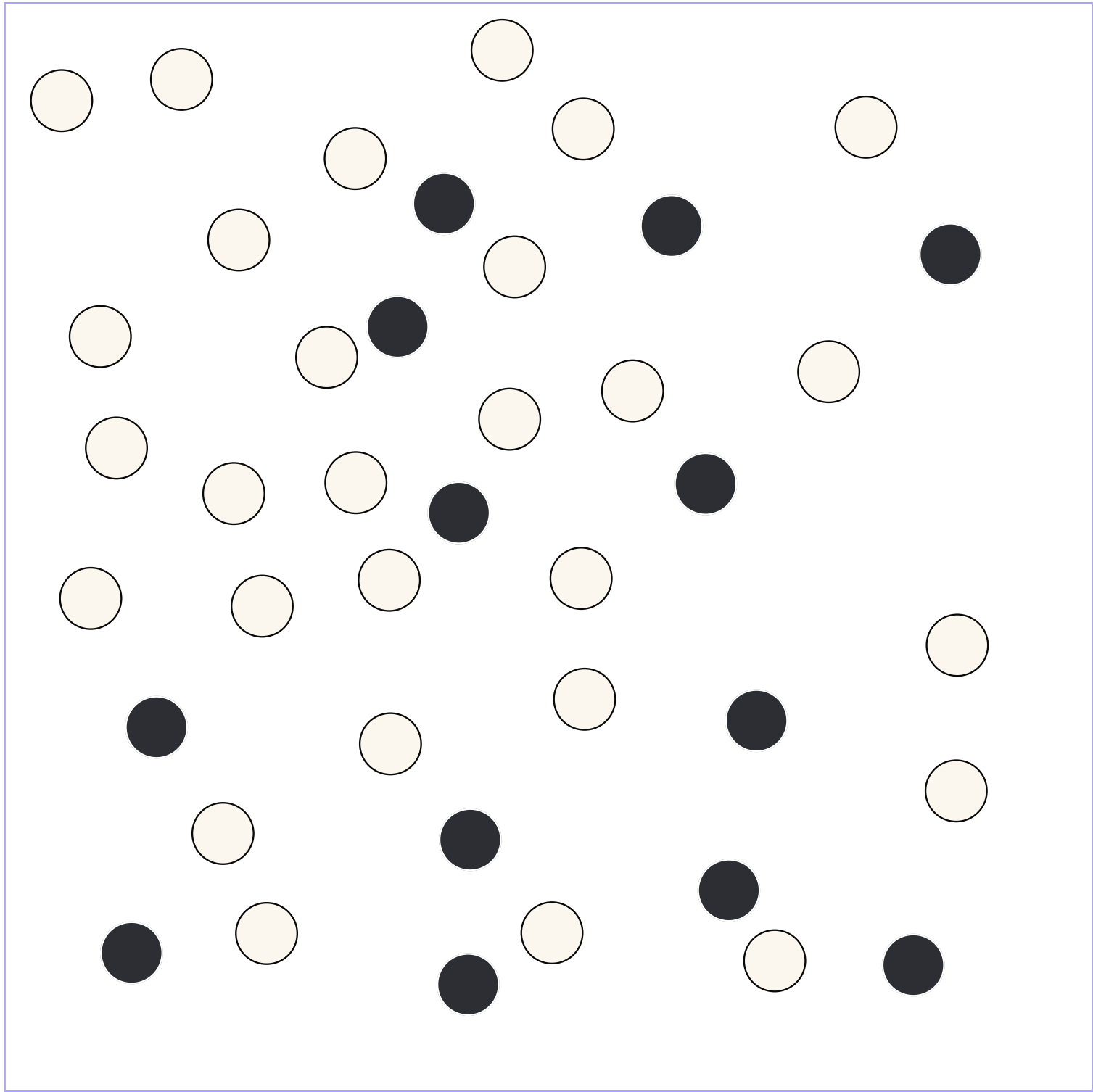
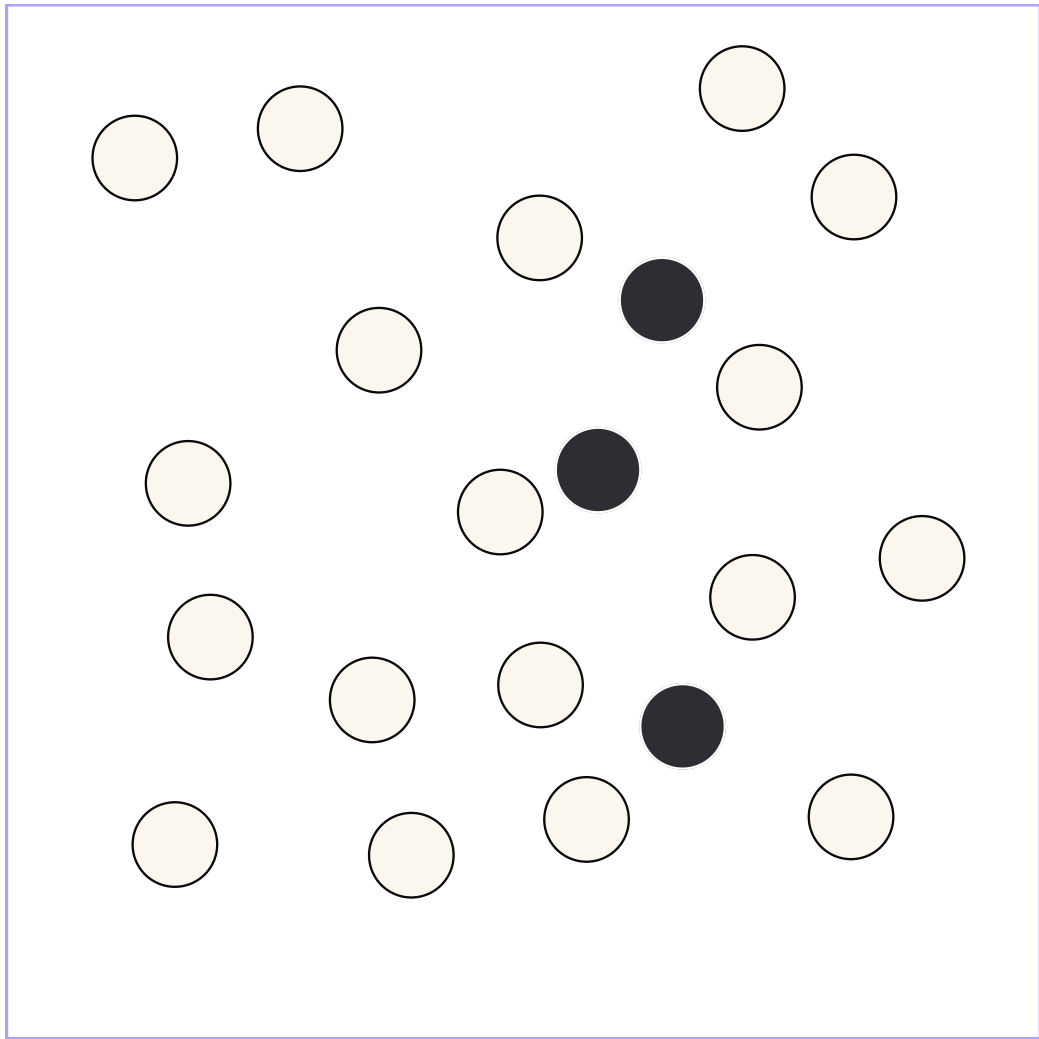
size of a release

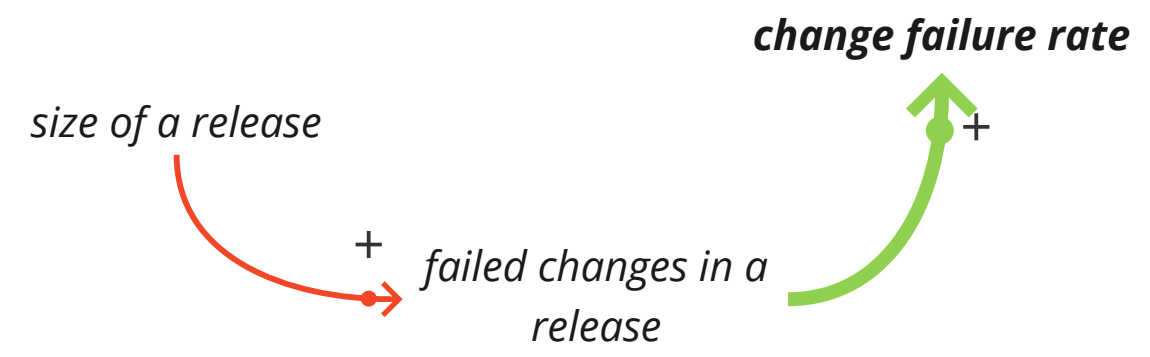
size of a release

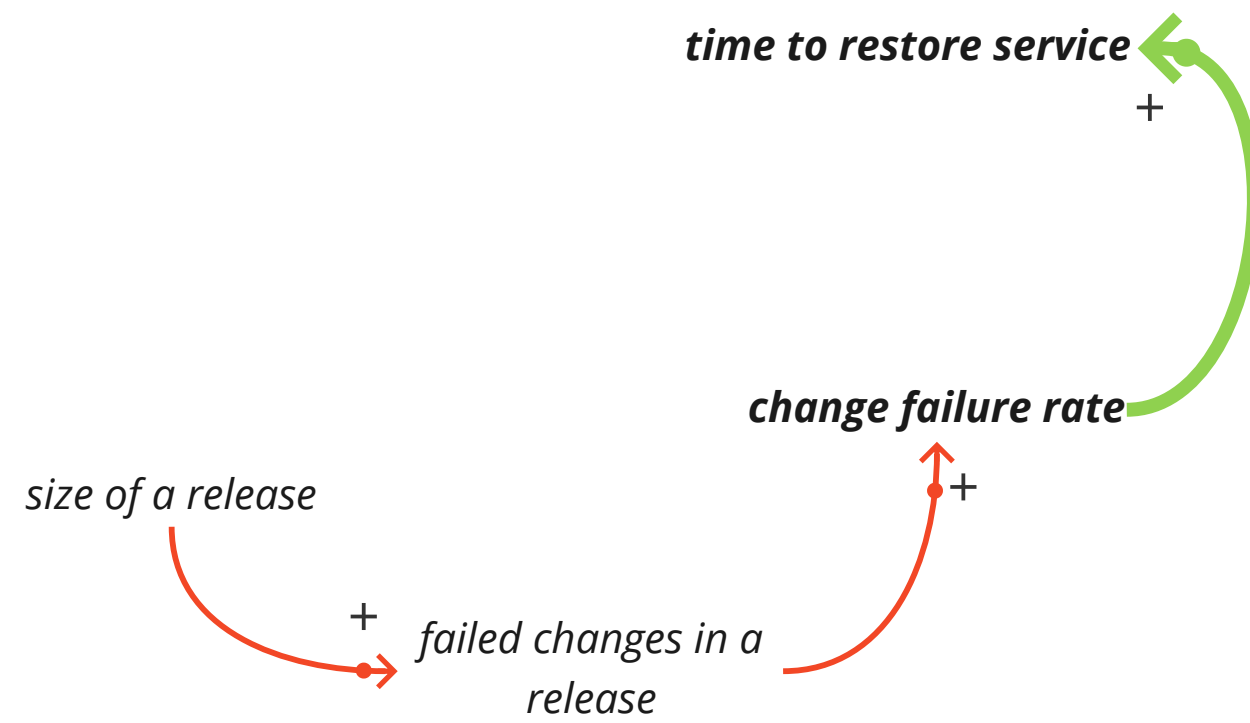


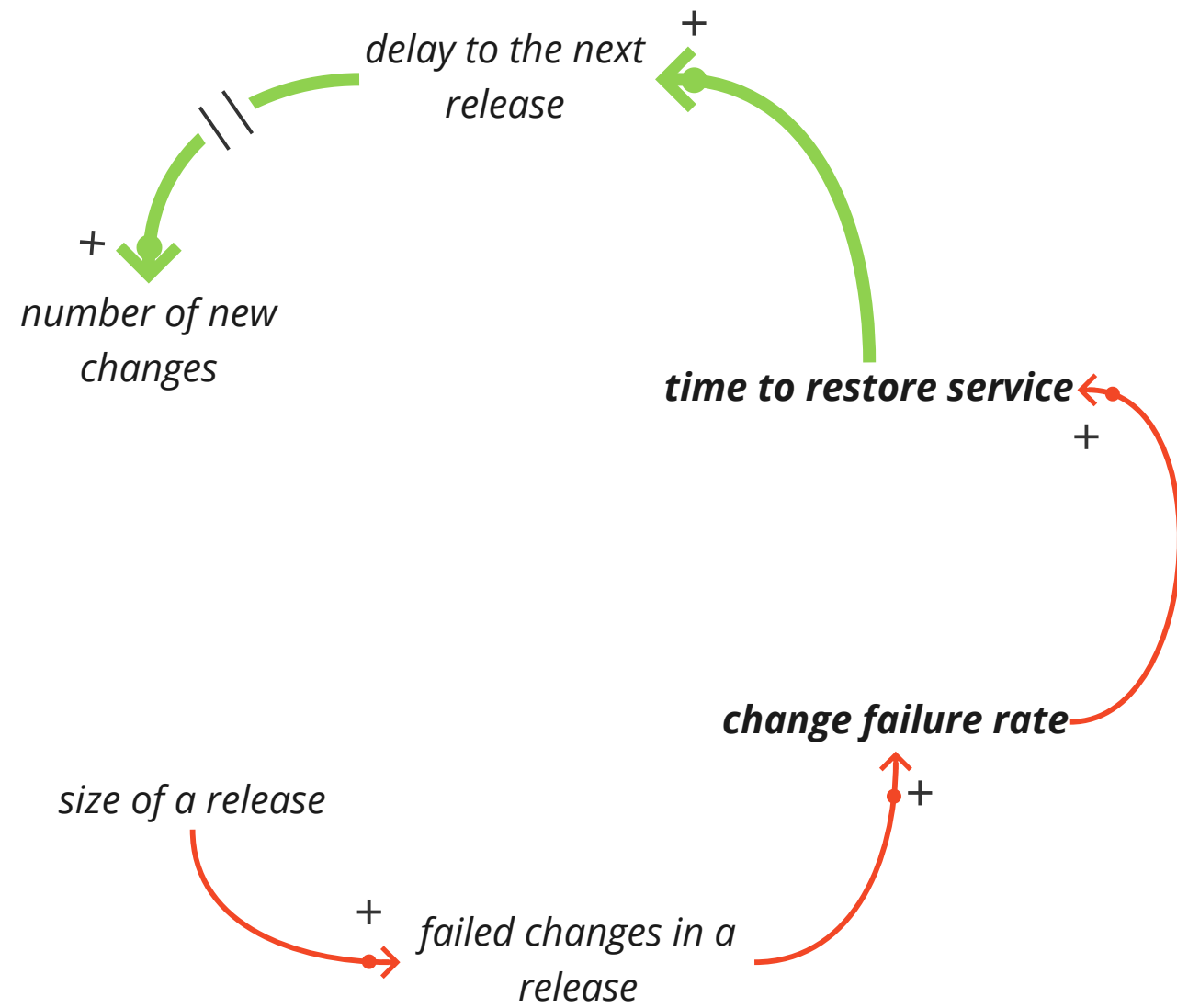
+

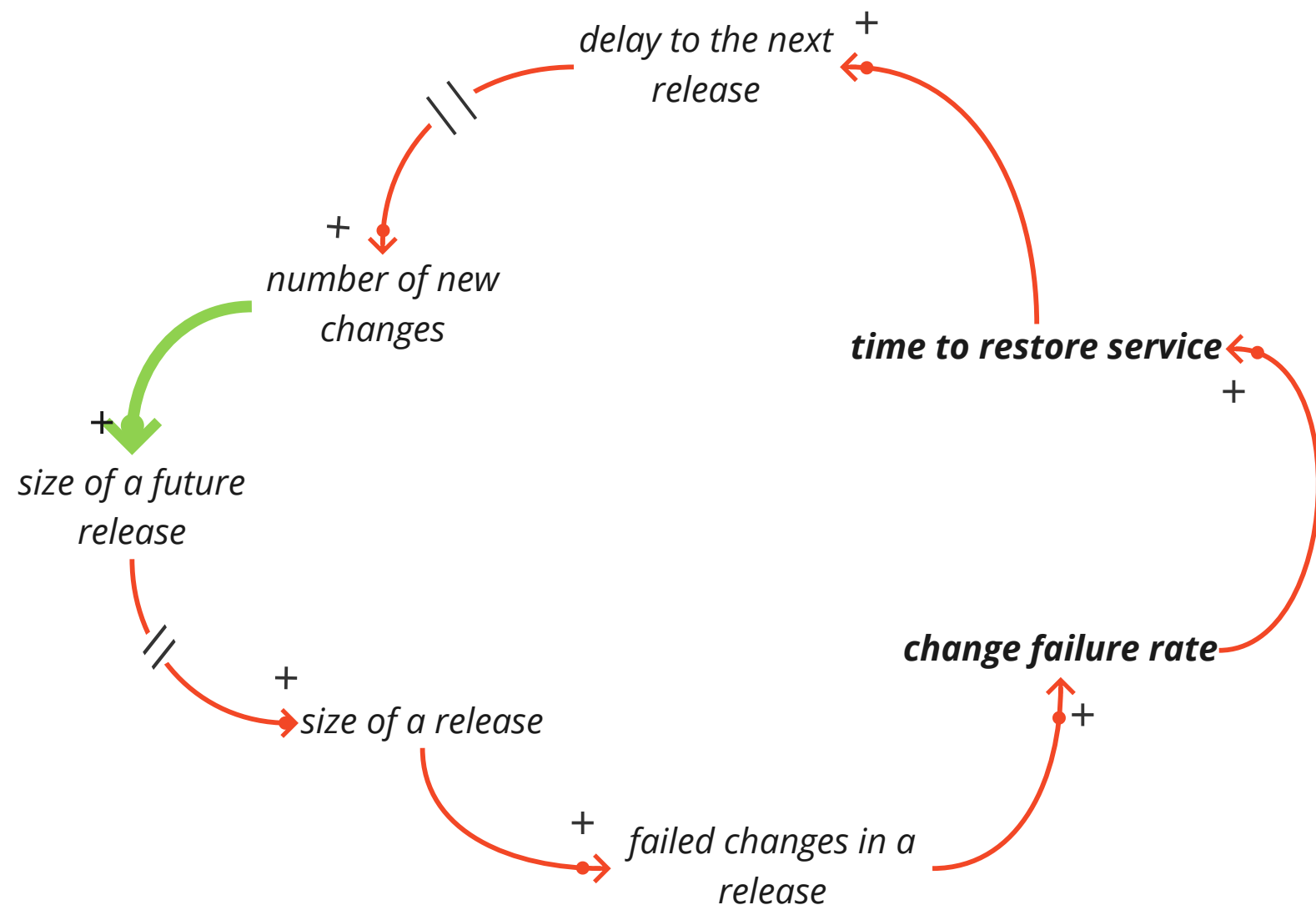
*failed changes in a
release*

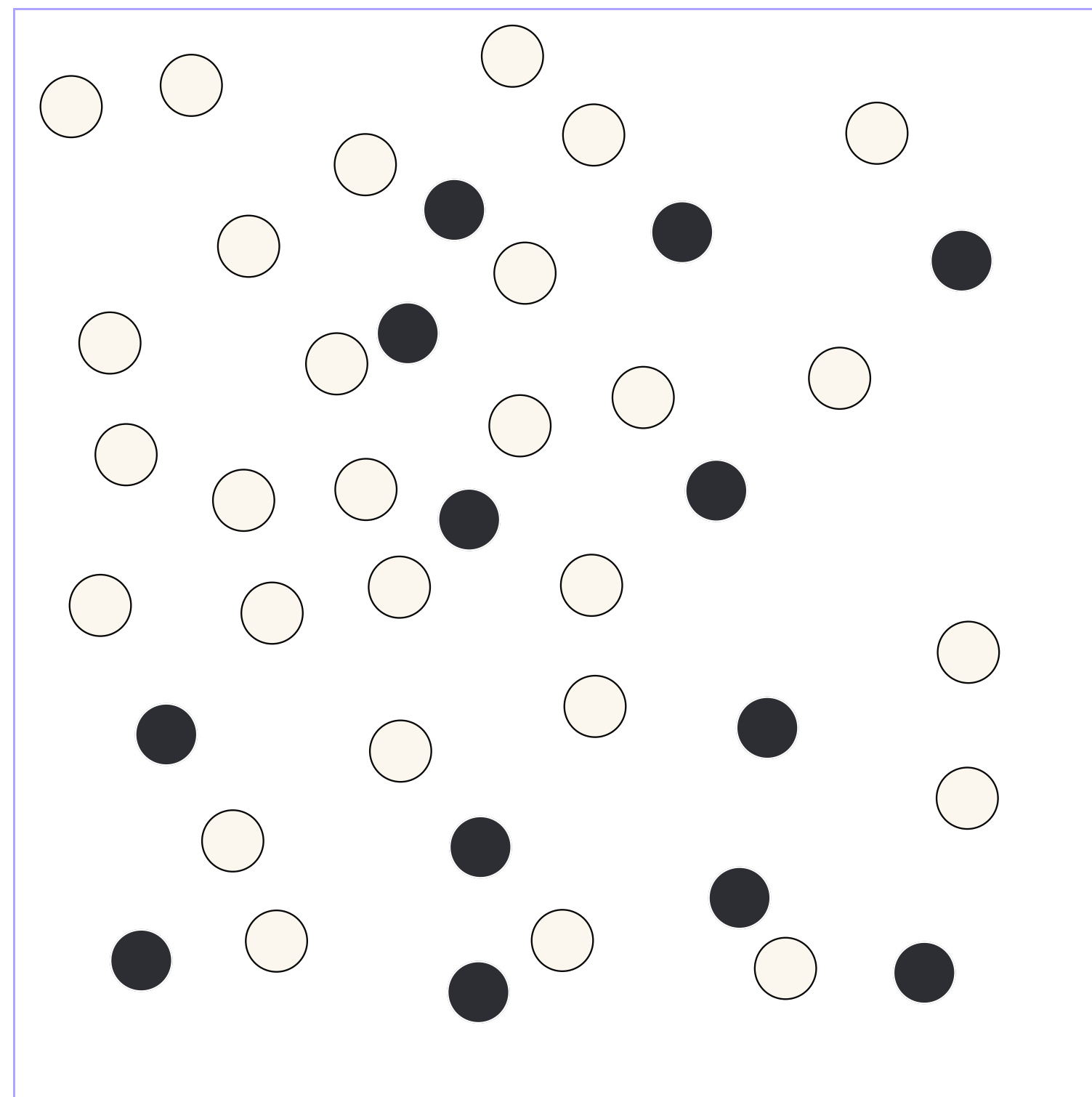
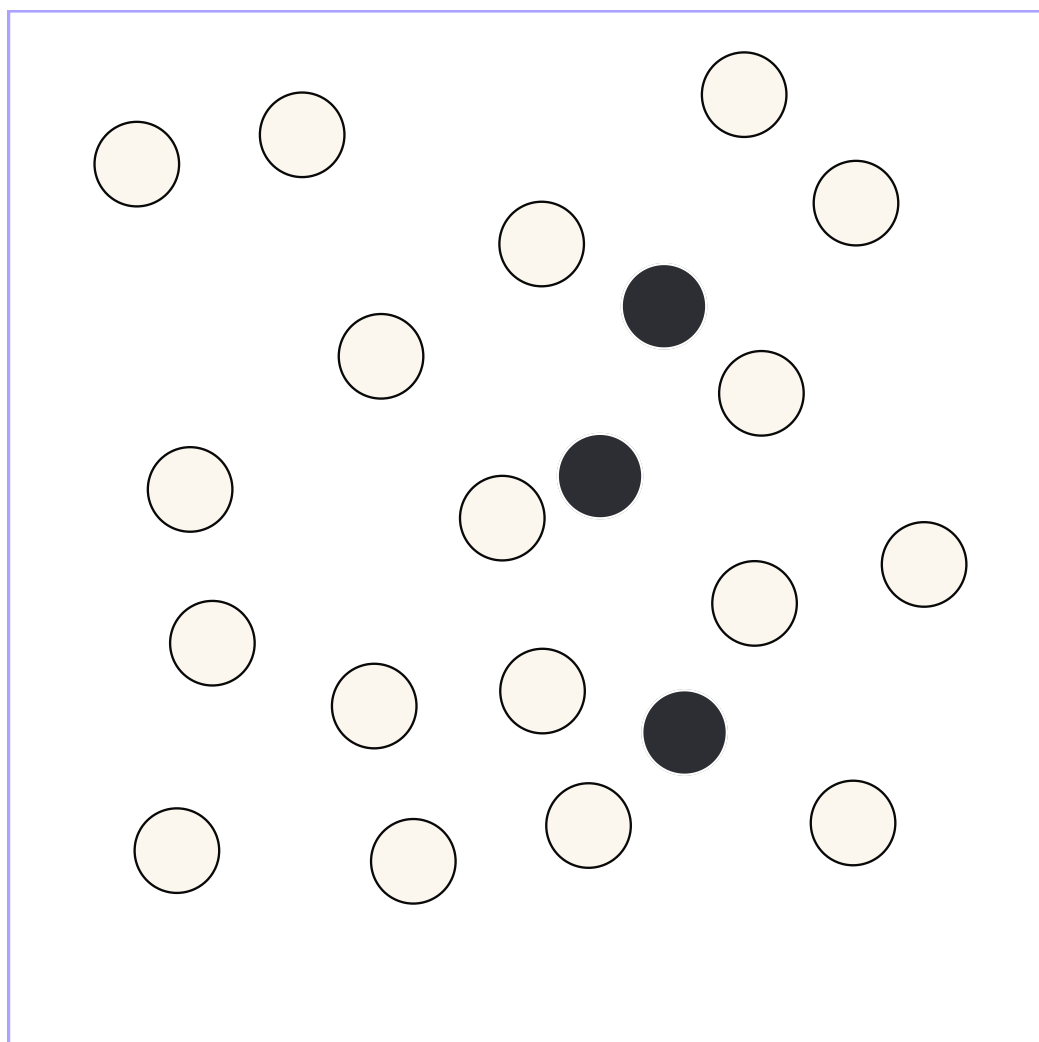






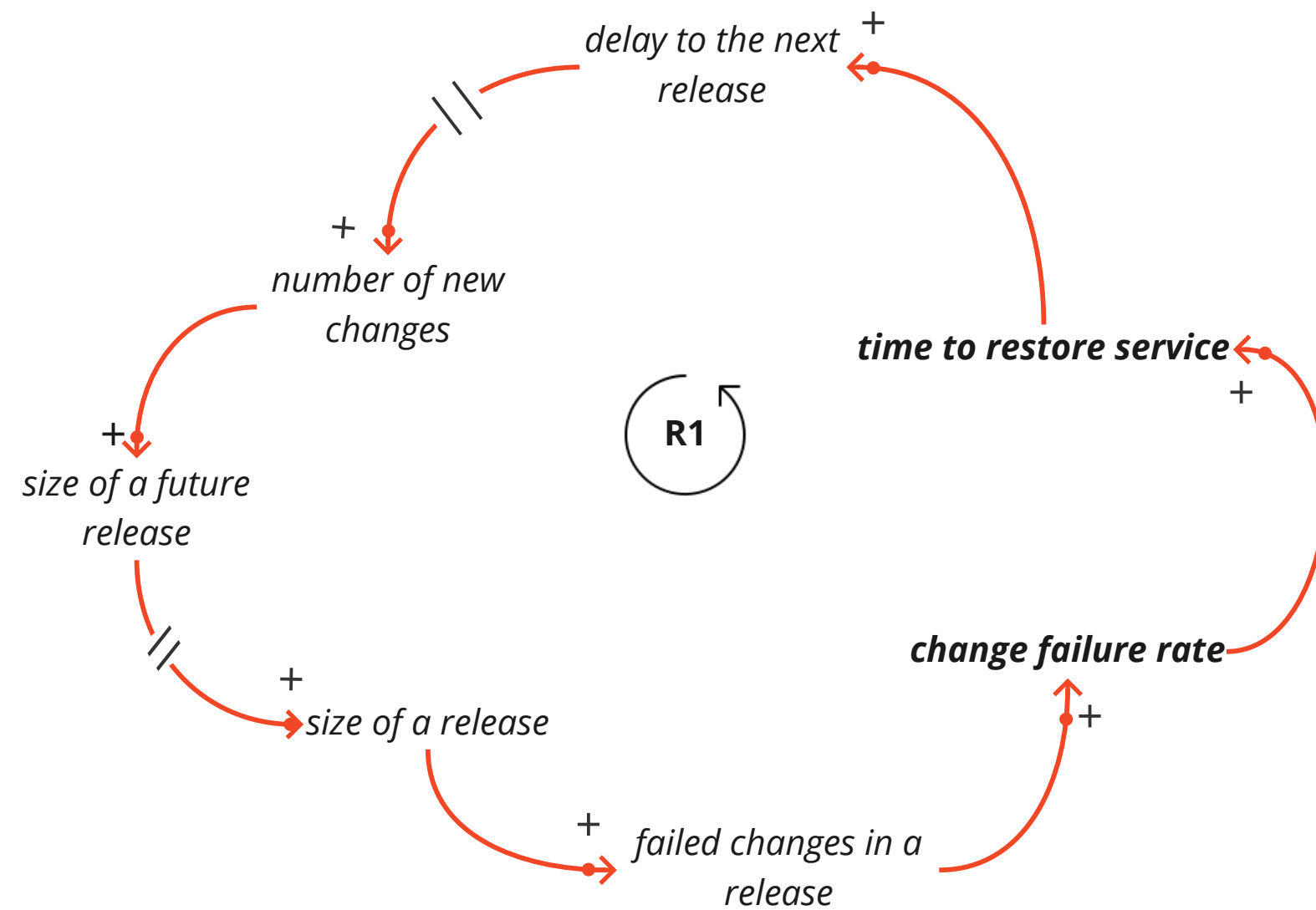




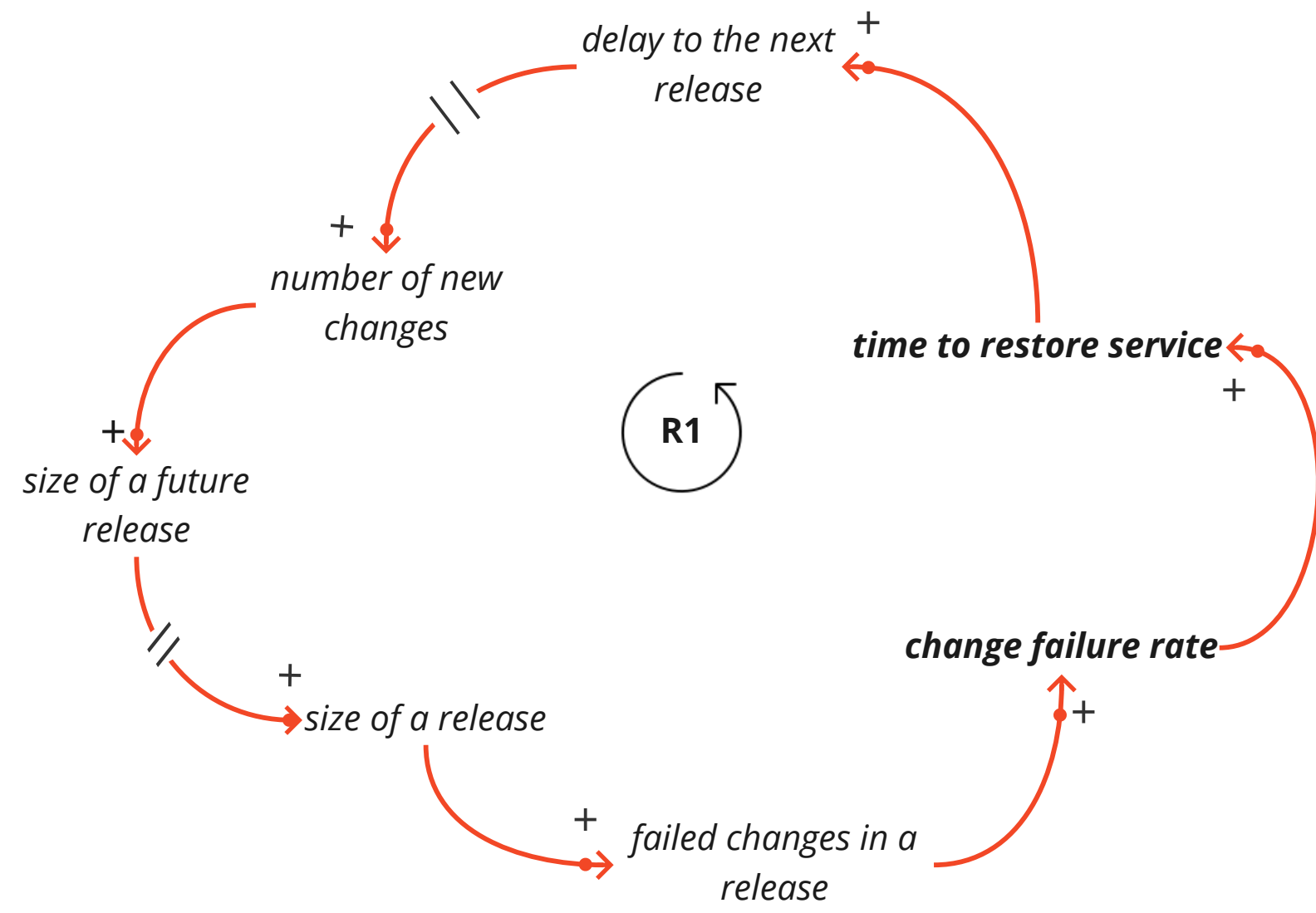


Seeing the *system*

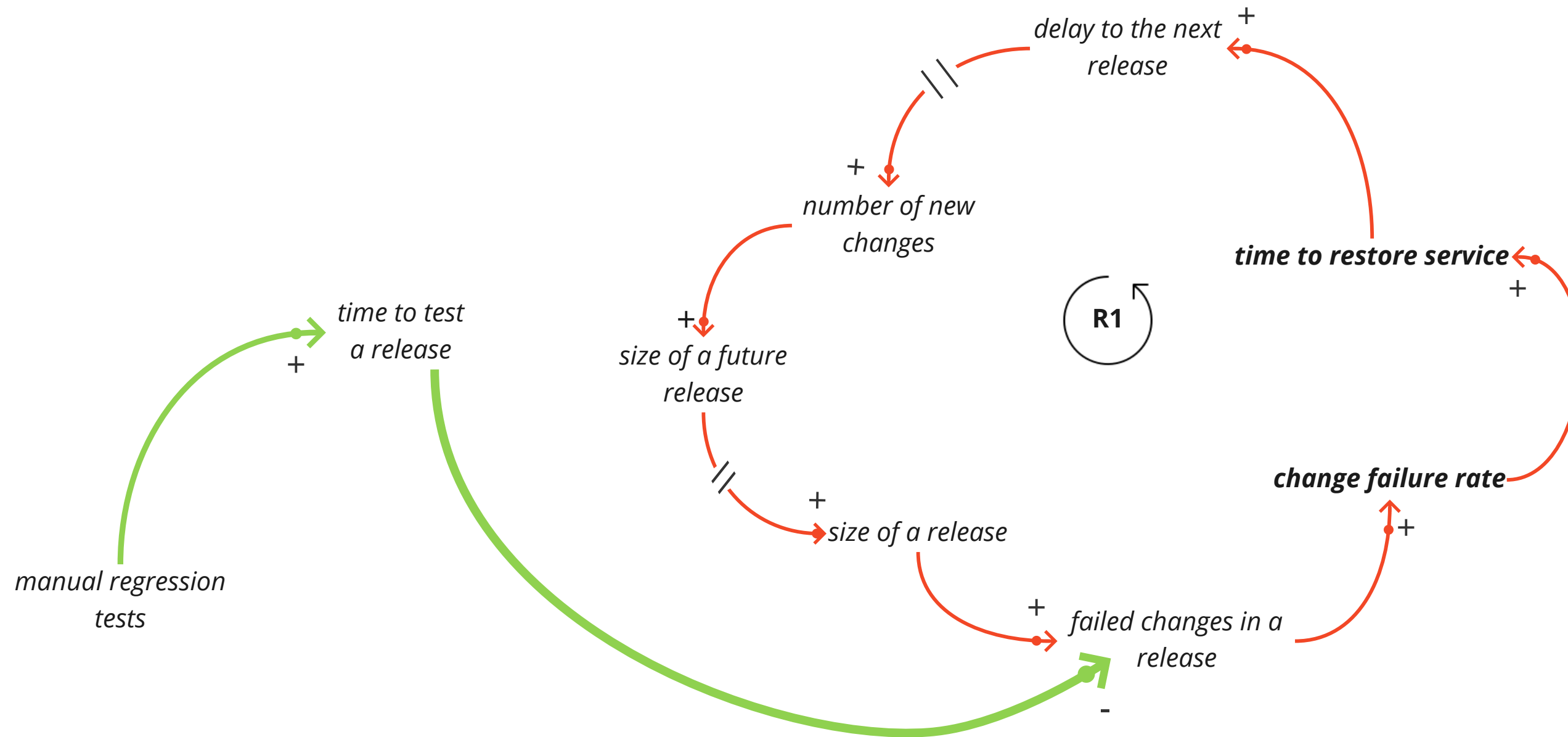
The system engine



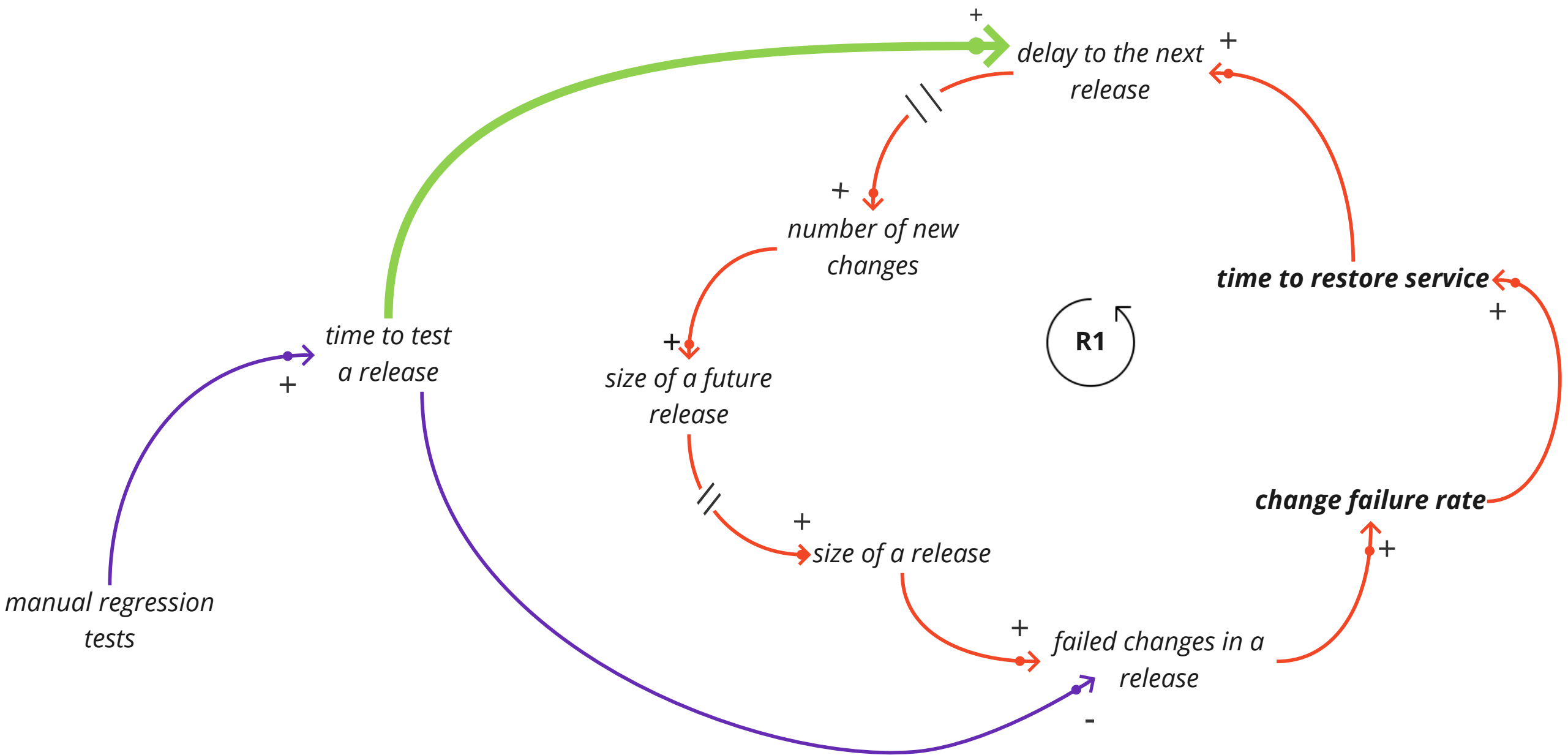
Let's change things



More time to test



More time to test

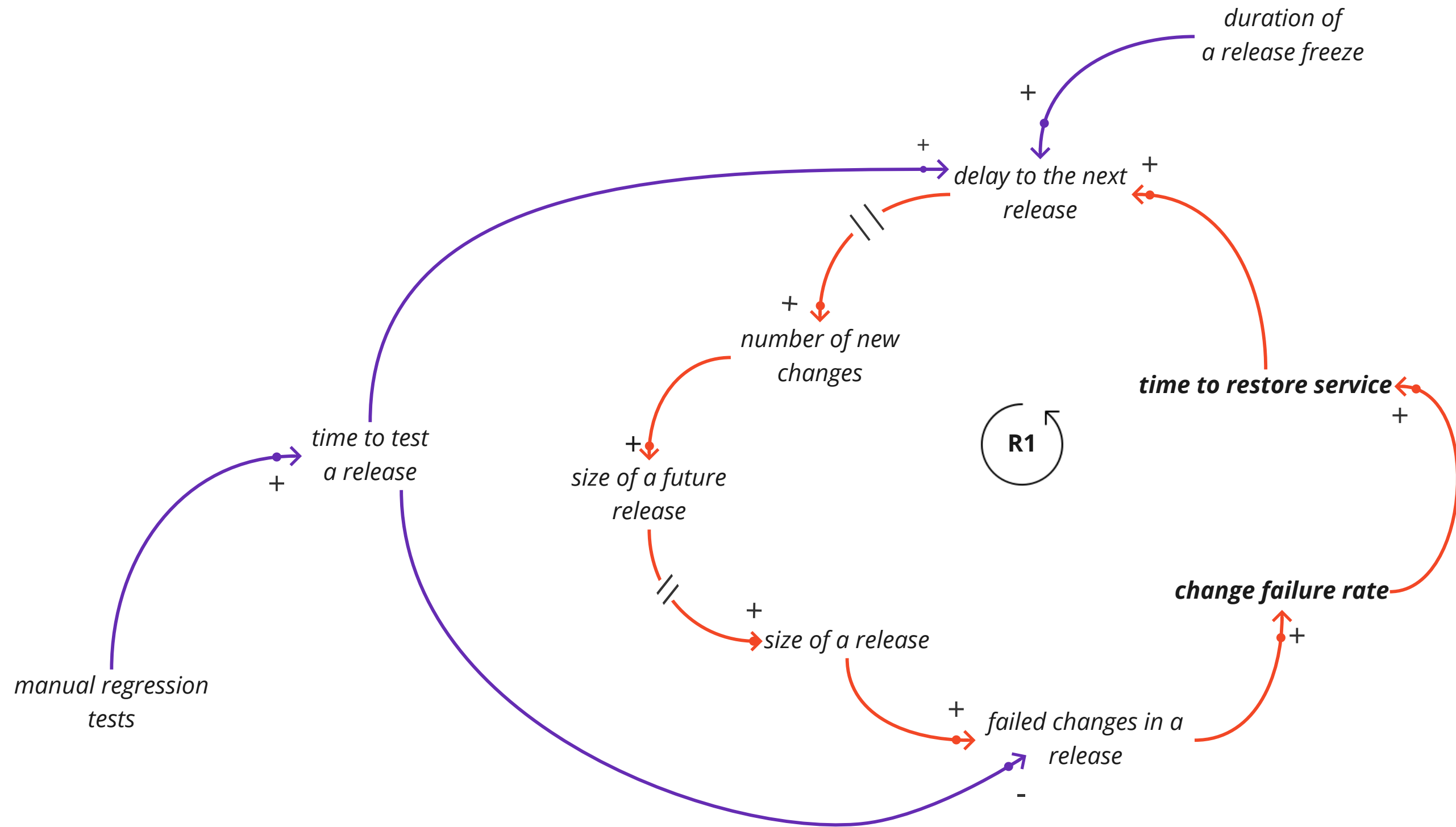


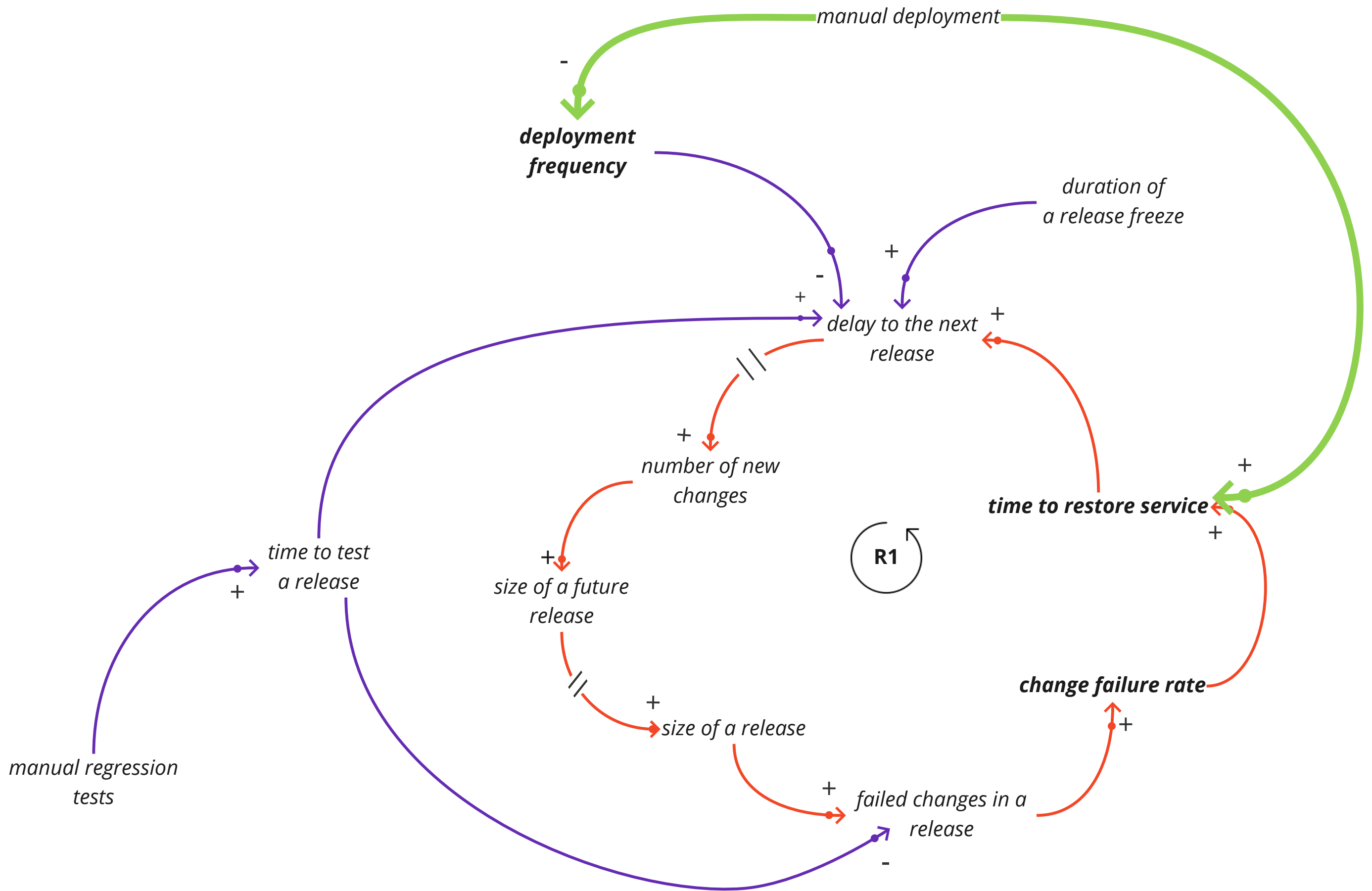
BRACE YOURSELVES



RELEASE FREEZES ARE COMING

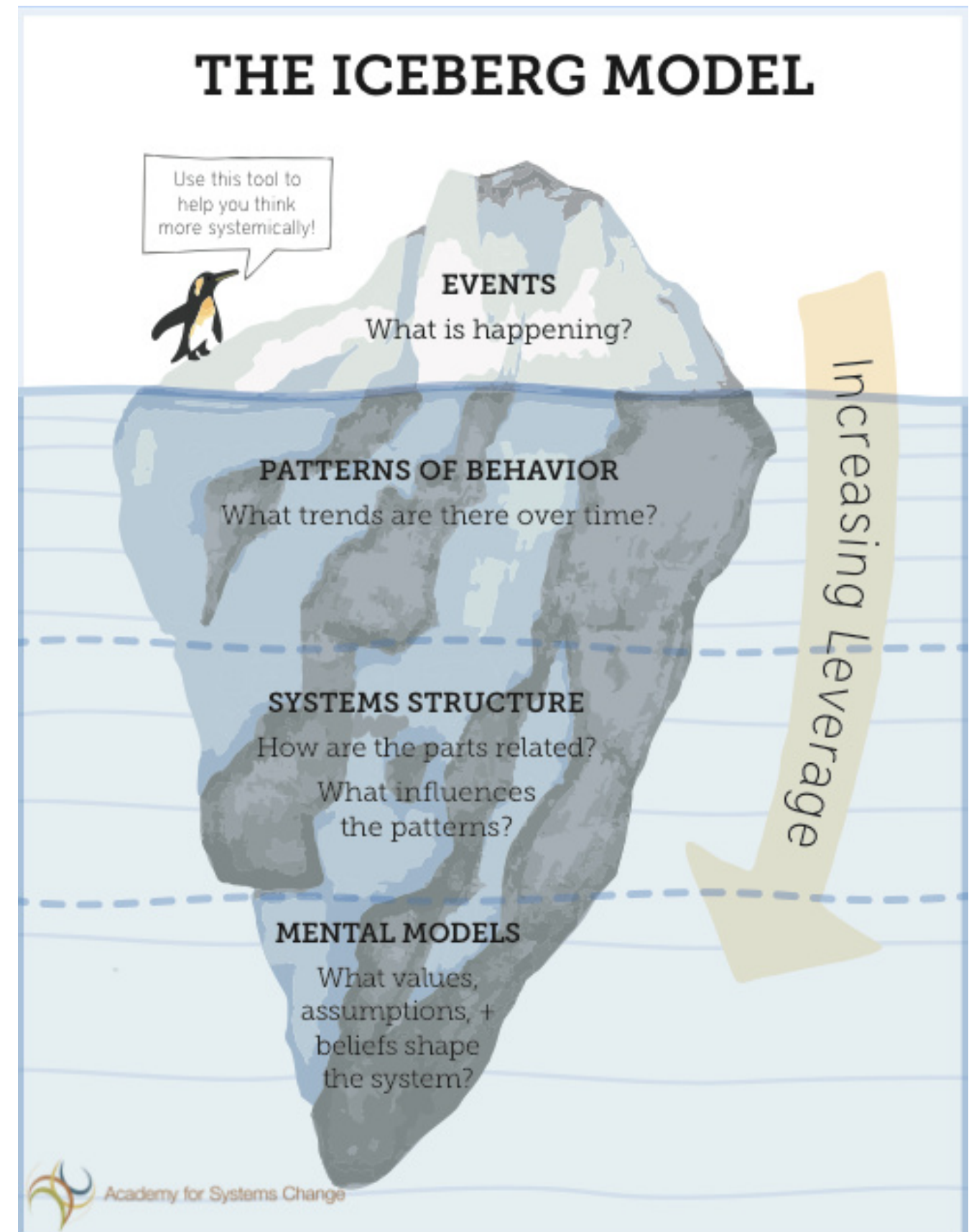
Impact of a release freeze



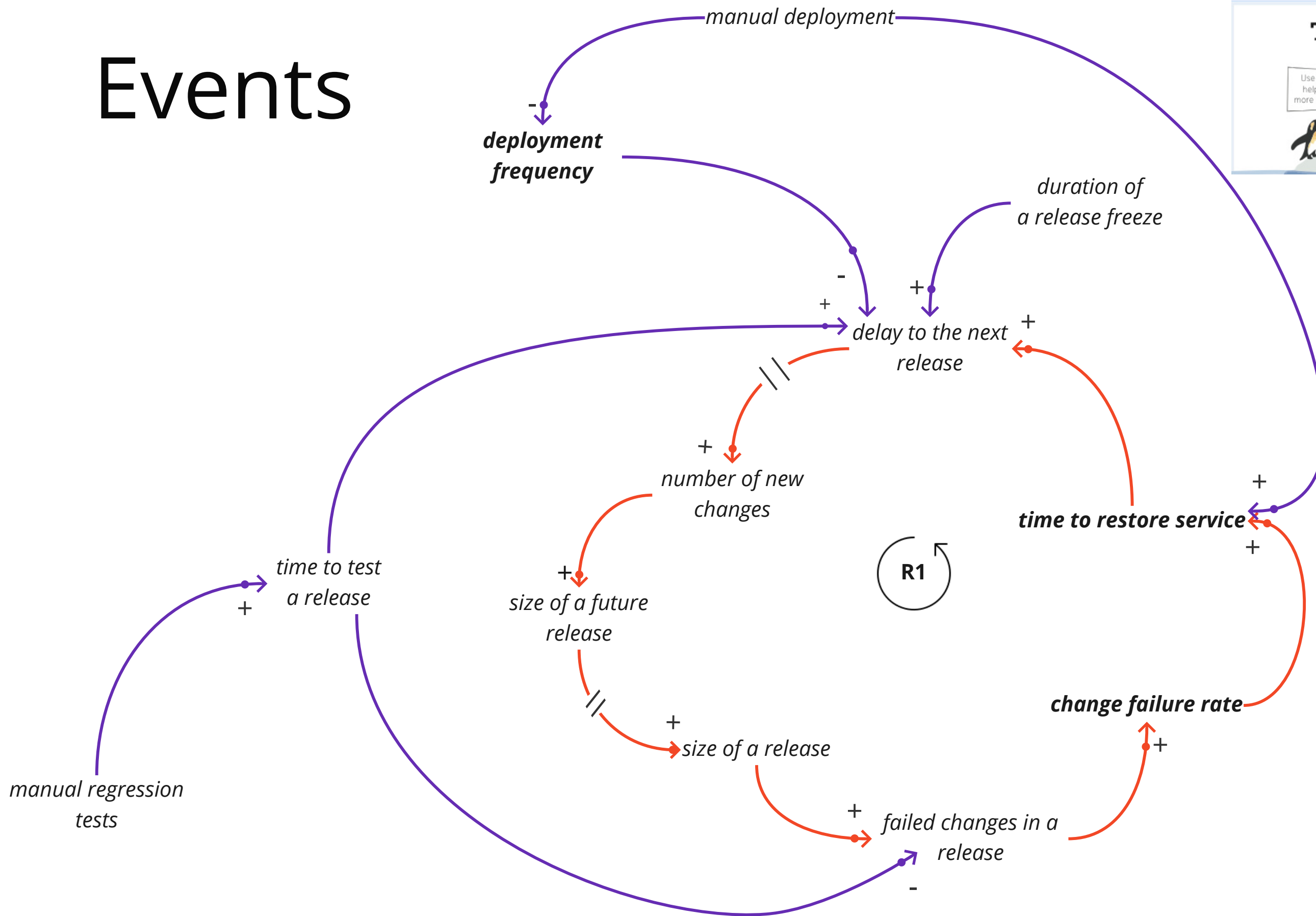


Making lasting change

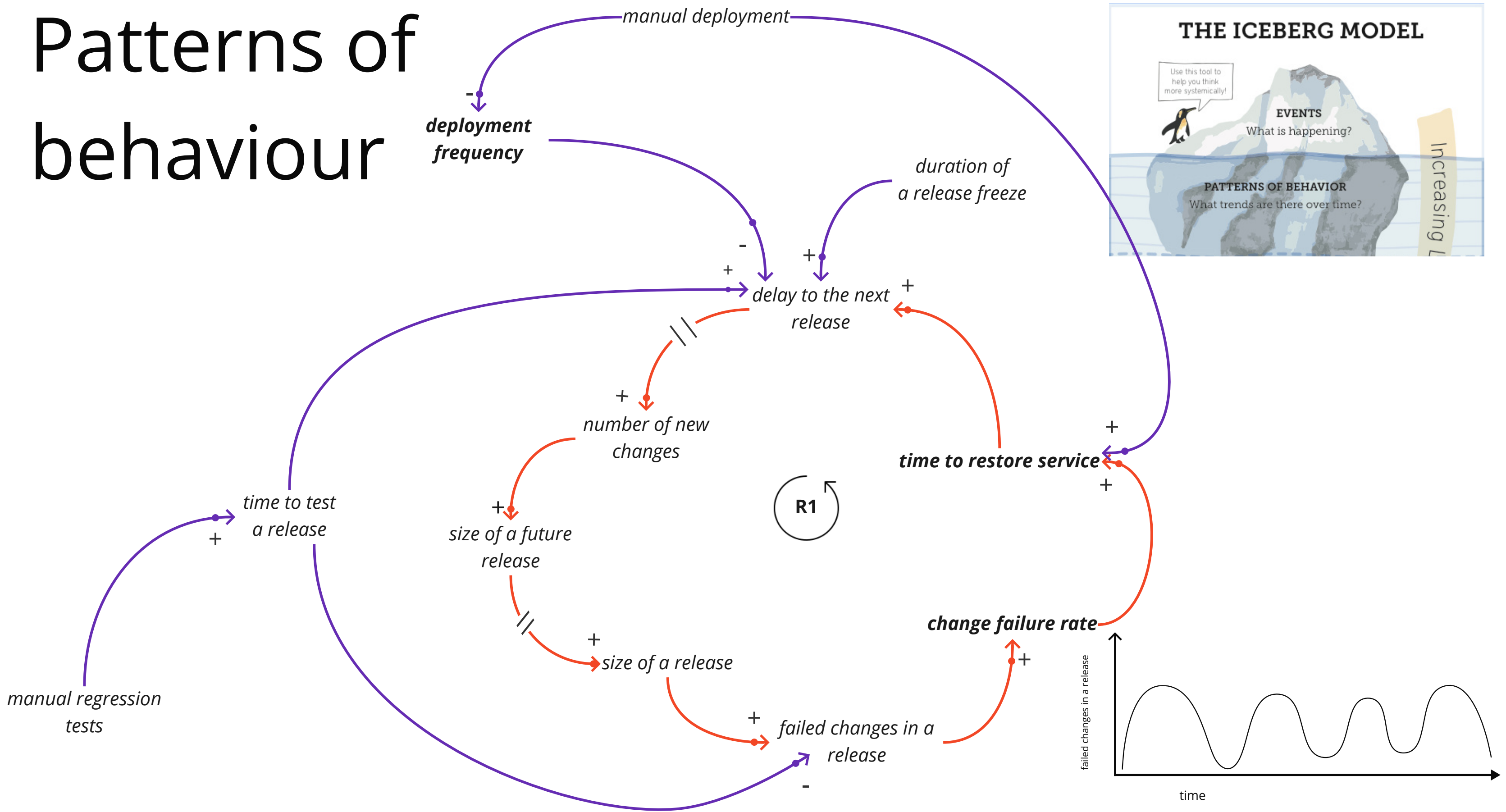
Seeing the iceberg



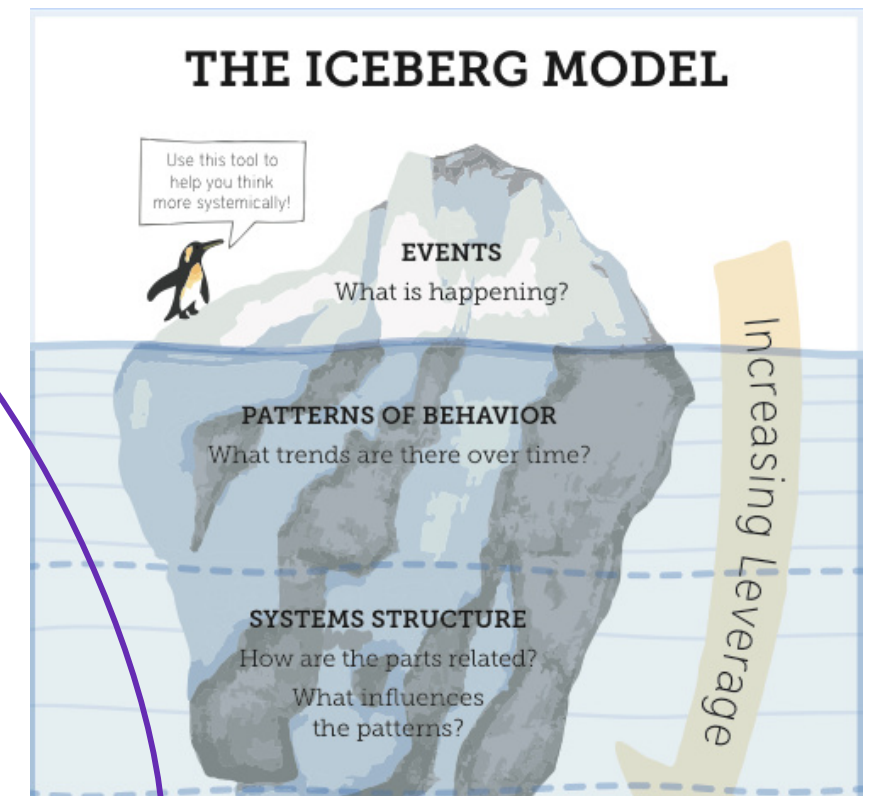
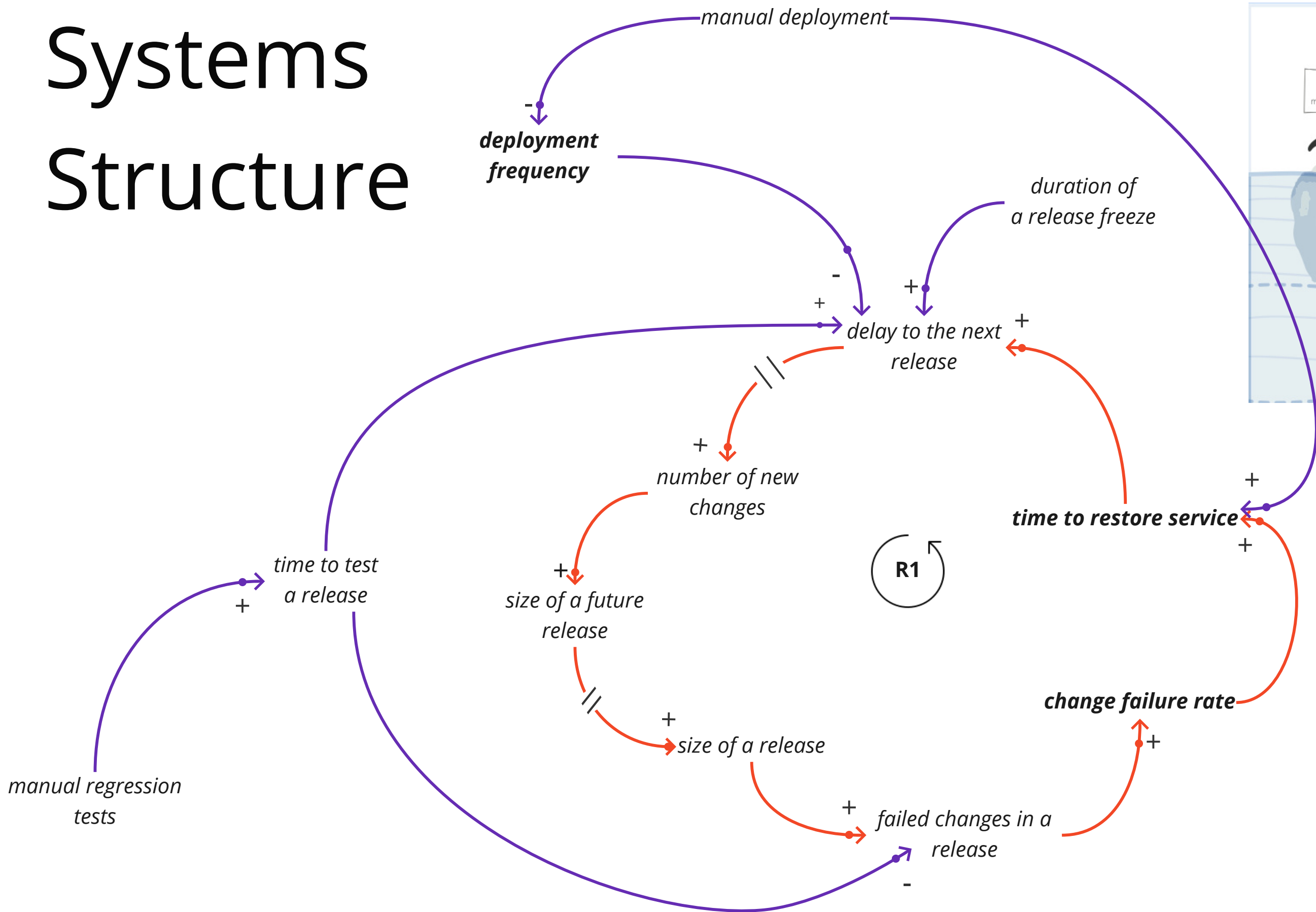
Events



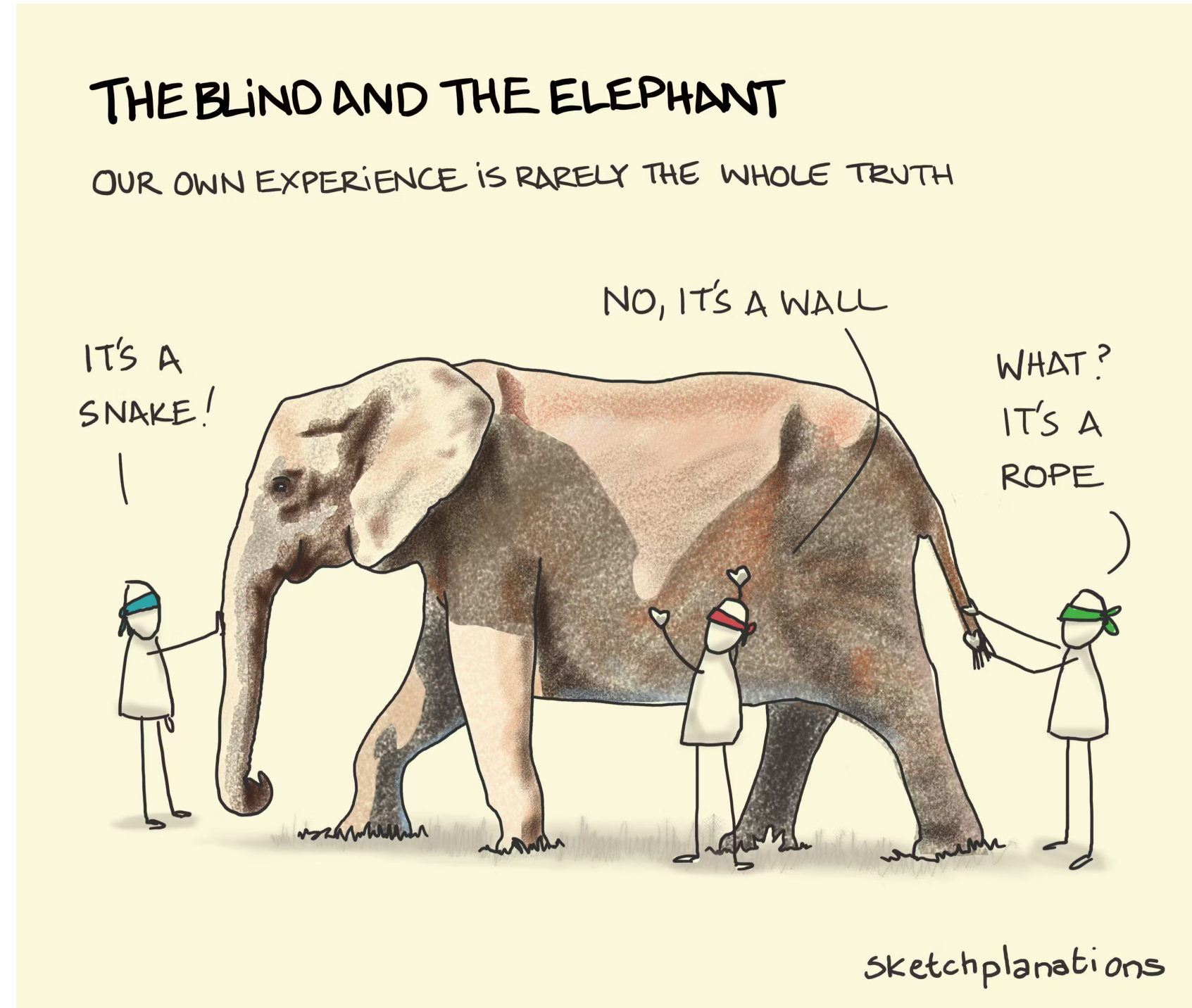
Patterns of behaviour



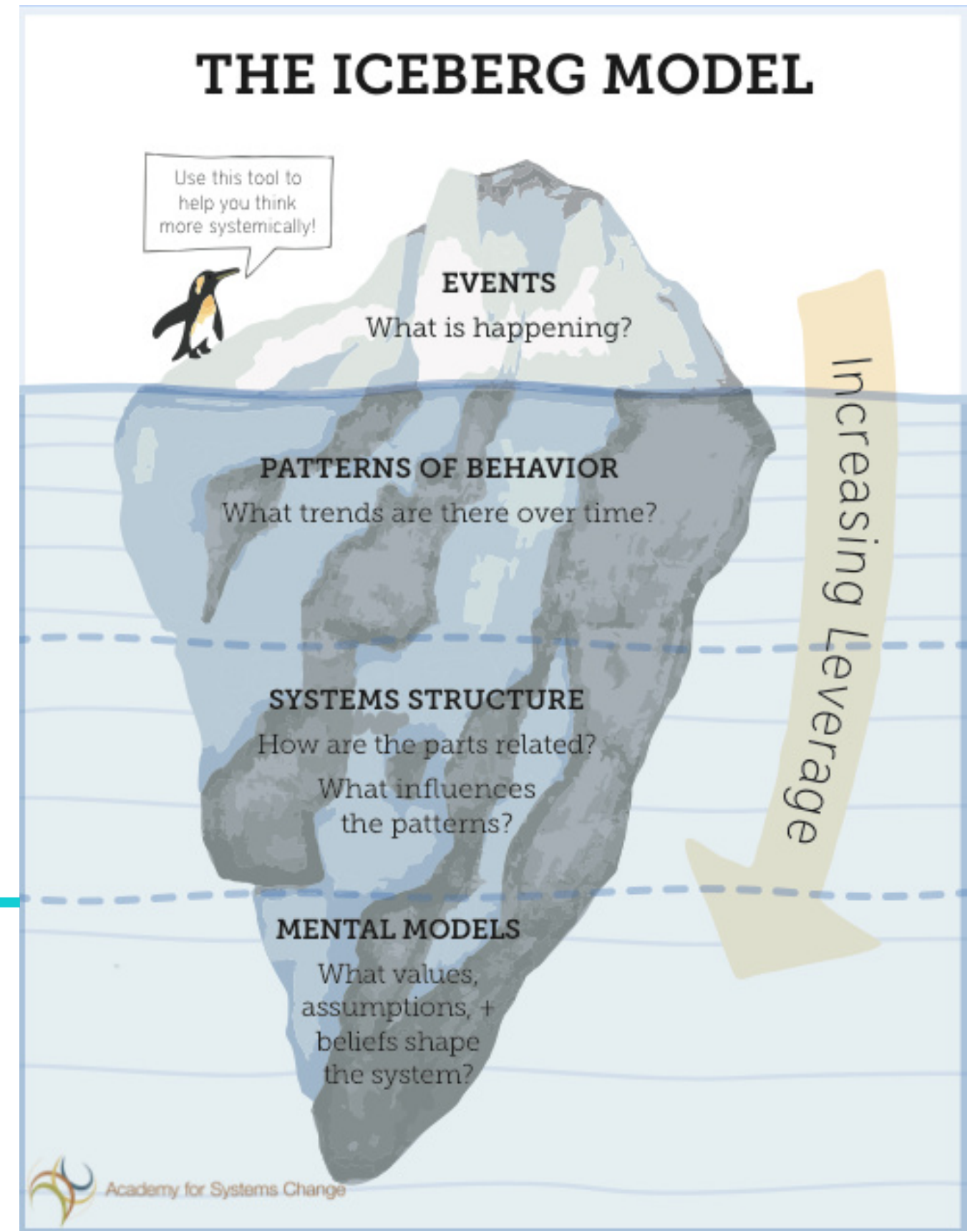
Systems Structure



Collaborative Modelling



Counterintuitive ideas shape our mental models



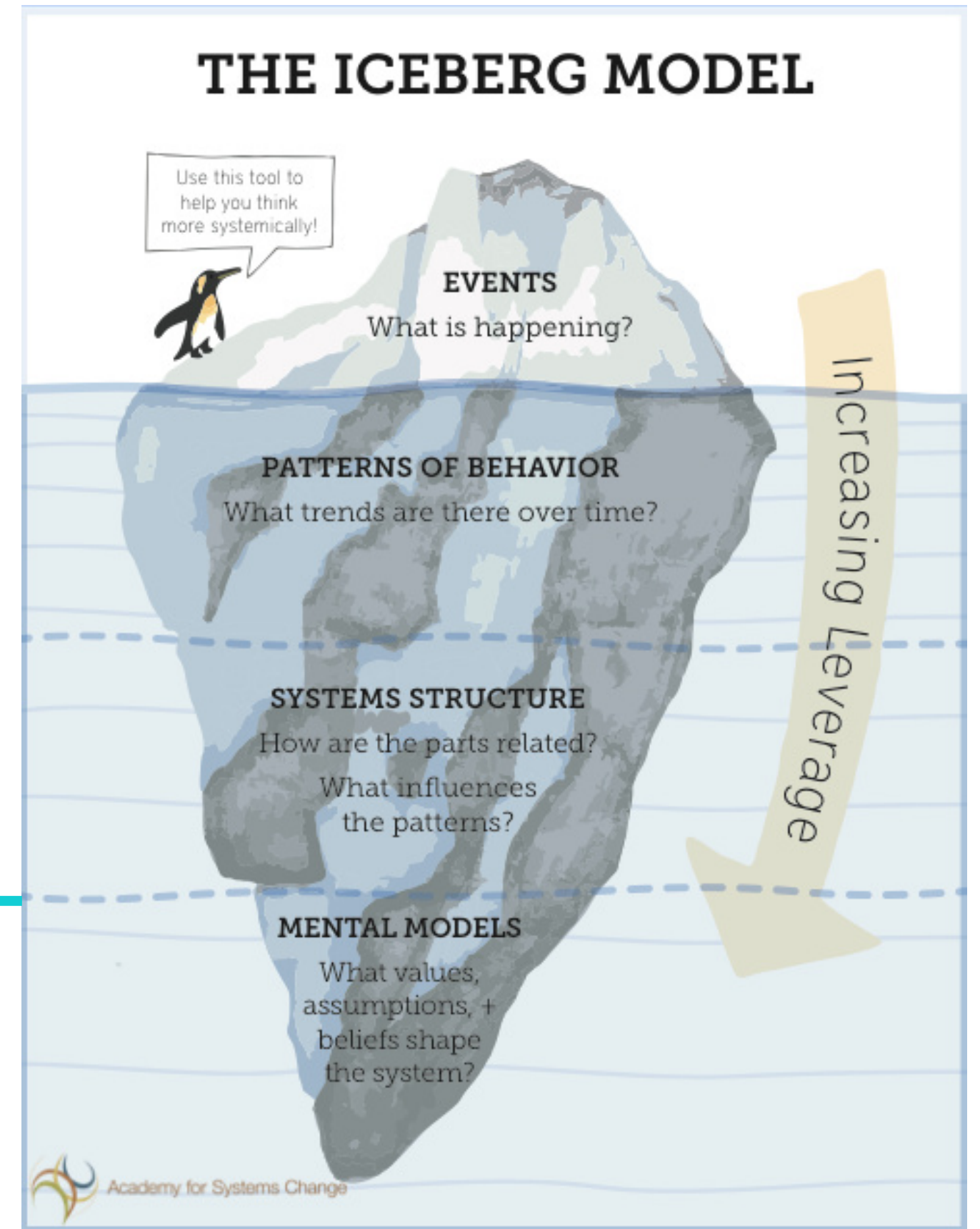
releasing more often improves quality

if it hurts, do it often

stop starting, start finishing

slack time improves flow

What mental models shape our current system?



releasing once a month keeps us safe

hero culture

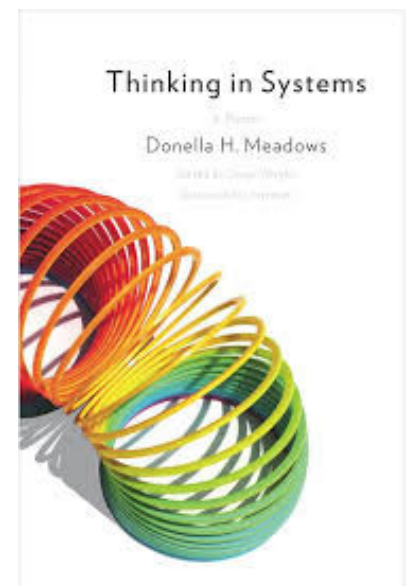
devs and testers in different teams

devs need to be busy all the time

Non-linear strategy

Listen to the wisdom of the system

Donella Meadows

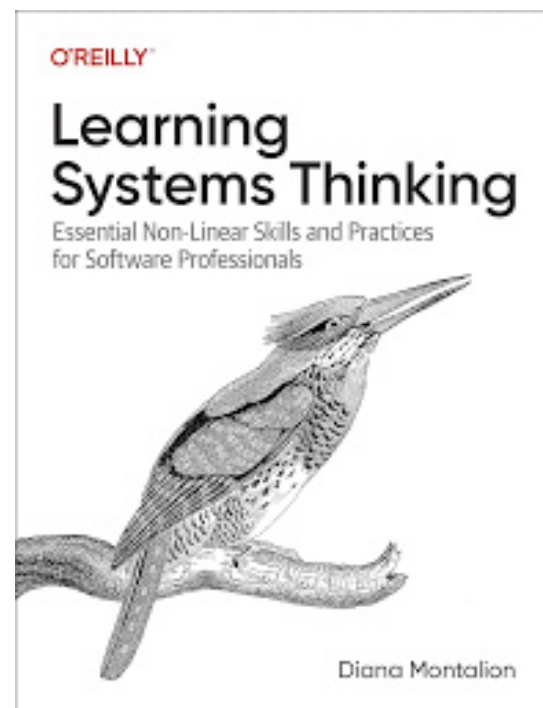


The *system* exists because it
works

Understanding the dynamics
of your system leads to a
better strategy

Continuously improving our
mental models, collaboratively of
the system leads to a better
strategy

'Models are conversations. There is no right way to converse. Modelling is framing questions and exploring answers. Establishing a vibrant, healthy and impactful modelling process is far more important than which tools you use'



Diana Montalio

Dance with the system

Donella Meadows

*"Believe in the fundamental
interconnectedness of all things"*

*Dirk Gently
Holistic Detective*

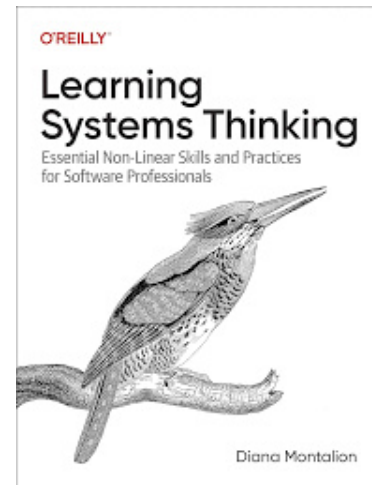
Thank you

The Grammar of Systems
From Order to Chaos & Back



33 Systems Laws and Principles
How to Think like a Systems Thinker

Patrick Hoverstadt 

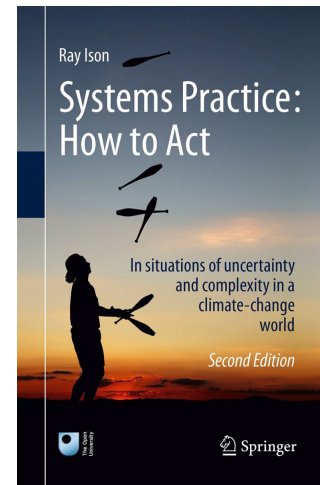


O'REILLY

Learning Systems Thinking

Essential Non-Linear Skills and Practices for Software Professionals

Diana Montalón



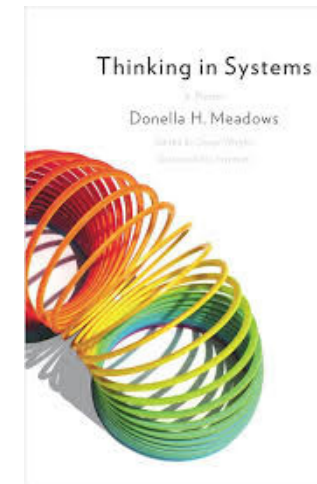
Ray Ison

Systems Practice: How to Act

In situations of uncertainty and complexity in a climate-change world

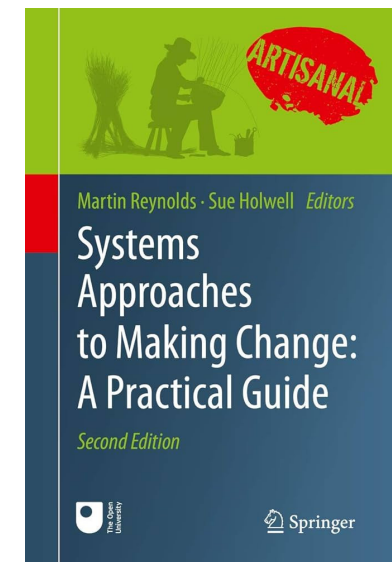
Second Edition

 Springer



Thinking in Systems

by Donella H. Meadows



Martin Reynolds · Sue Holwell *Editors*

Systems Approaches to Making Change: A Practical Guide

Second Edition

 Springer

 Springer

<https://www.open.edu/openlearn/science-maths-technology/engineering-technology/systems-thinking-free-courses>

Questions?

@hibri

www.hibri.net

<https://techhub.social/@hibri>

