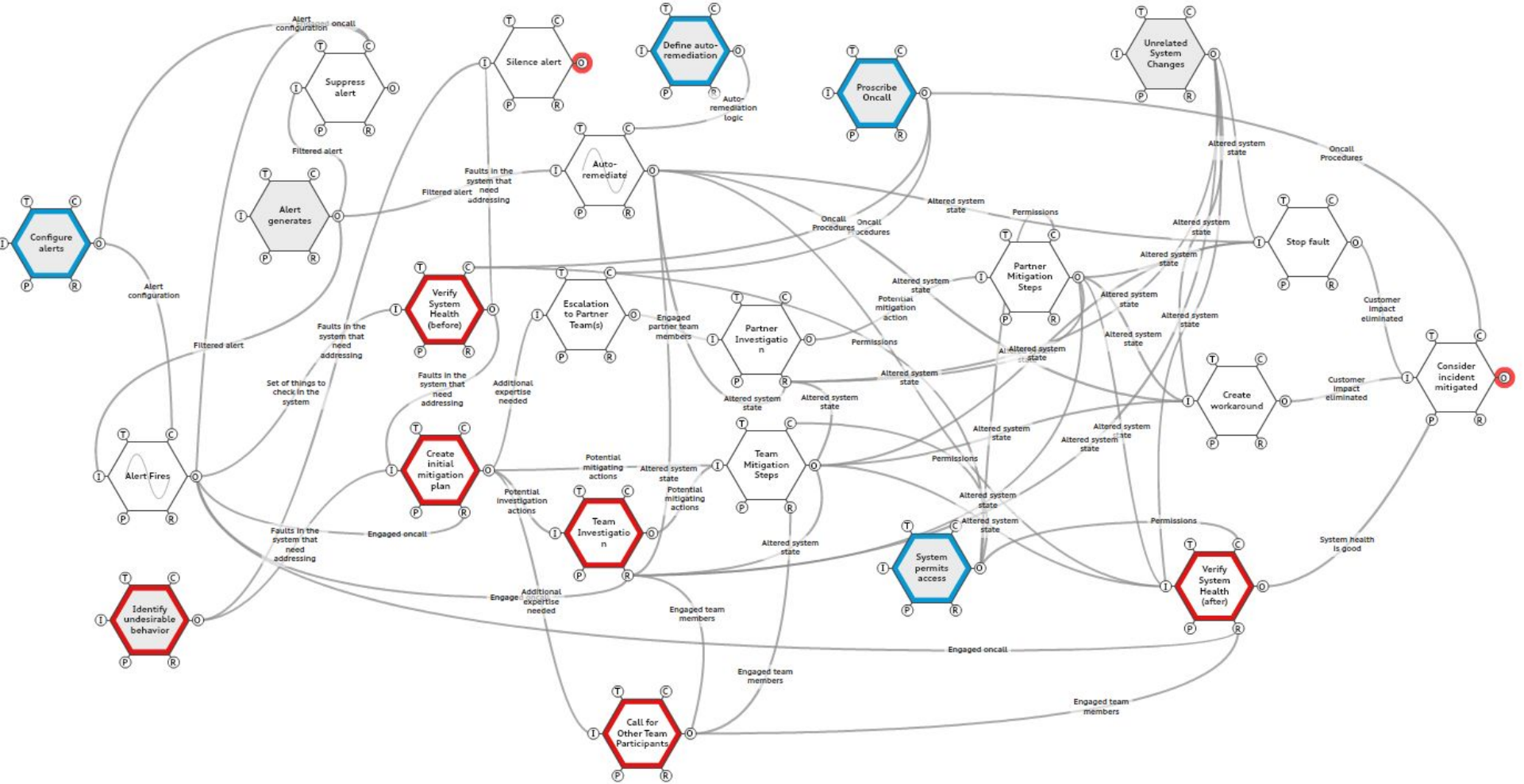
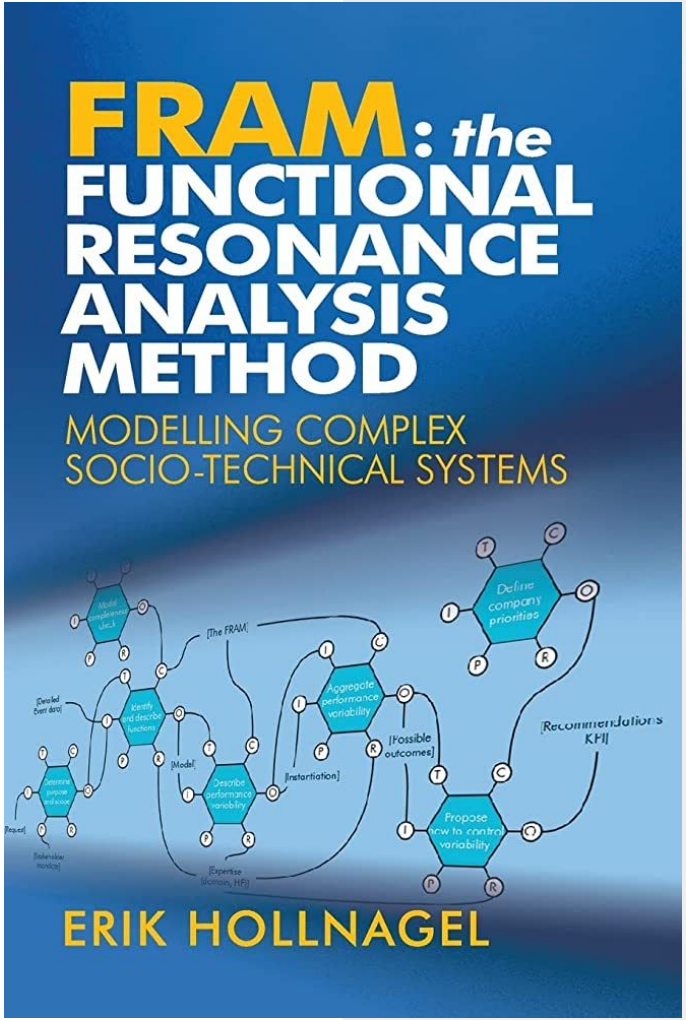
indeed

Functional Resonance Analysis: Diagramming Your System

Tanner Lund

2023


Configure alerts
Alert configuration
Alert generates
Filtered alert
Suppress alert
Alert configuration
Oncall Procedures
Silence alert
Define auto-remediation
Auto-remediation logic
Auto-remediate
Faults in the system that need addressing
Filtered alert
Proscribe Oncall
Unrelated System Changes
Altered system state
Oncall Procedures
Permissions
Stop fault
Customer impact eliminated
Partner Mitigation Steps
Potential mitigation action
Verify System Health (before)
Escalation to Partner Team(s)
Engaged partner team members
Partner Investigation
Permissions
Set of things to check in the system
Additional expertise needed
Create workaround
Customer impact eliminated
Consider incident mitigated
Alert Fires
Create initial mitigation plan
Potential mitigating actions
Potential investigation actions
Team Mitigation Steps
Team Investigation
Engaged oncall
Engage
Additional expertise needed
Engaged team members
System permits access
Verify System Health (after)
System health is good
Identify undesirable behavior
Engaged oncall
Call for Other Team Participants
Engaged team members


FRAM: the FUNCTIONAL RESONANCE ANALYSIS METHOD
MODELLING COMPLEX SOCIO-TECHNICAL SYSTEMS
ERIK HOLLNAGEL

indeed

Why FRAM?

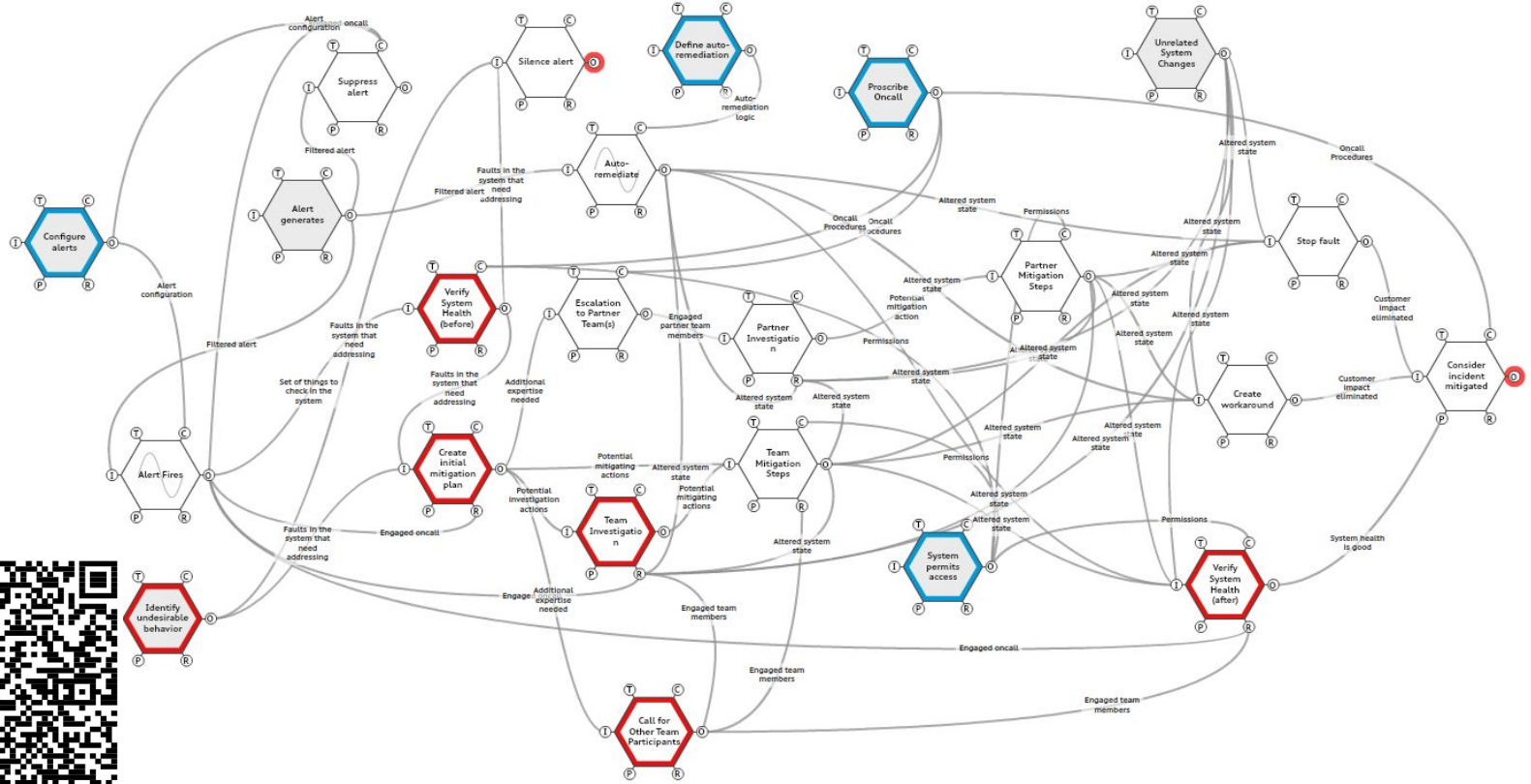
“Who’s Fram?”

“I dunno, I joined late…”

“Is it good to think with?”

- Claude Lévi-Strauss

Relative Ignorance, Complexity, and Intractability



indeed

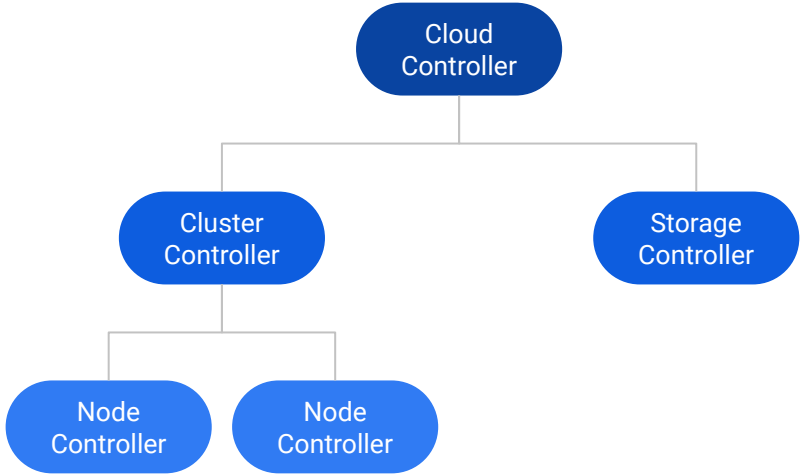
FRAM Principles

“Is FRAM not a model-cum-method then?”

“Apparently it’s more of a method-sine-model”

Functions, not Components

>

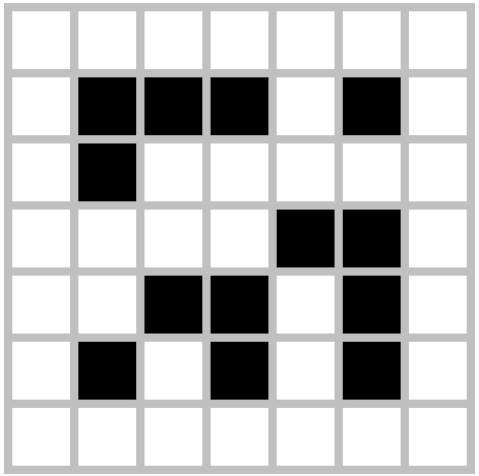


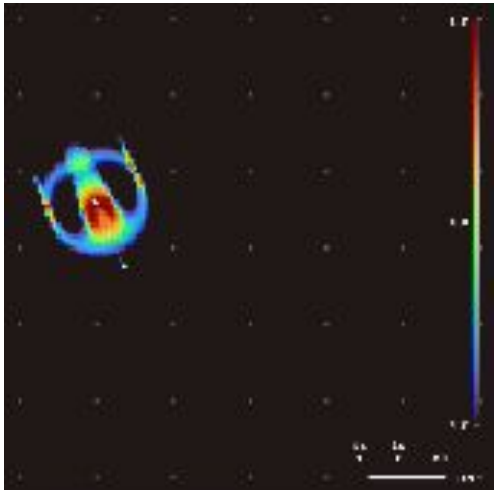
Failure

==

Success

Emergence

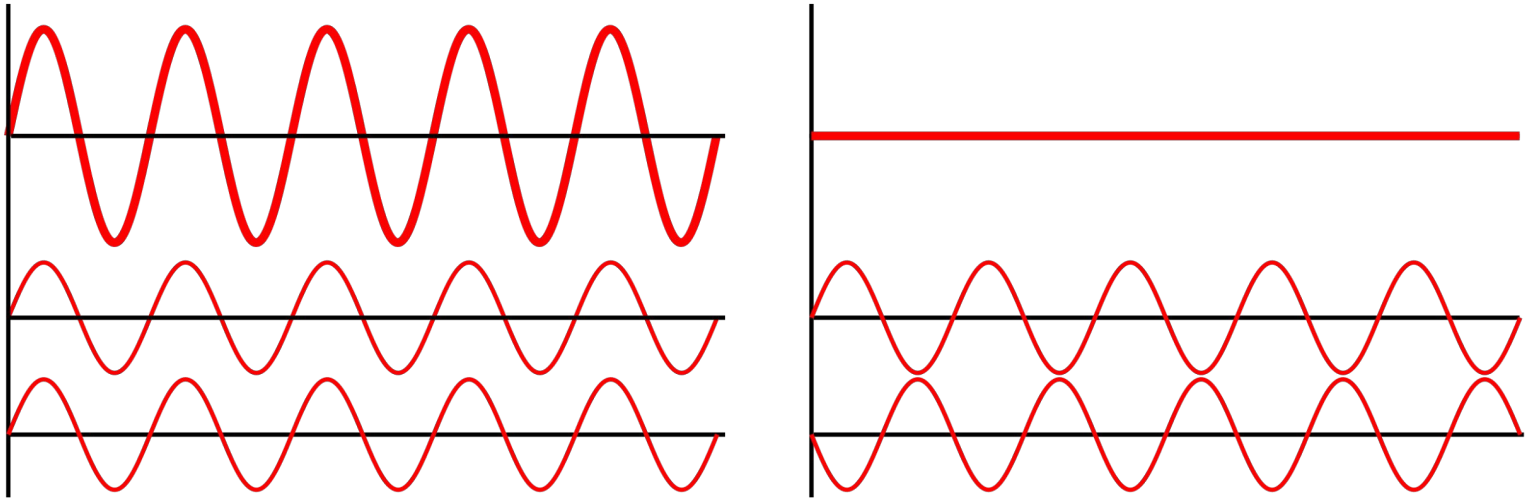




https://chakazul.github.io/lenia.html

Resonance

Stochastic Resonance, Functional Resonance

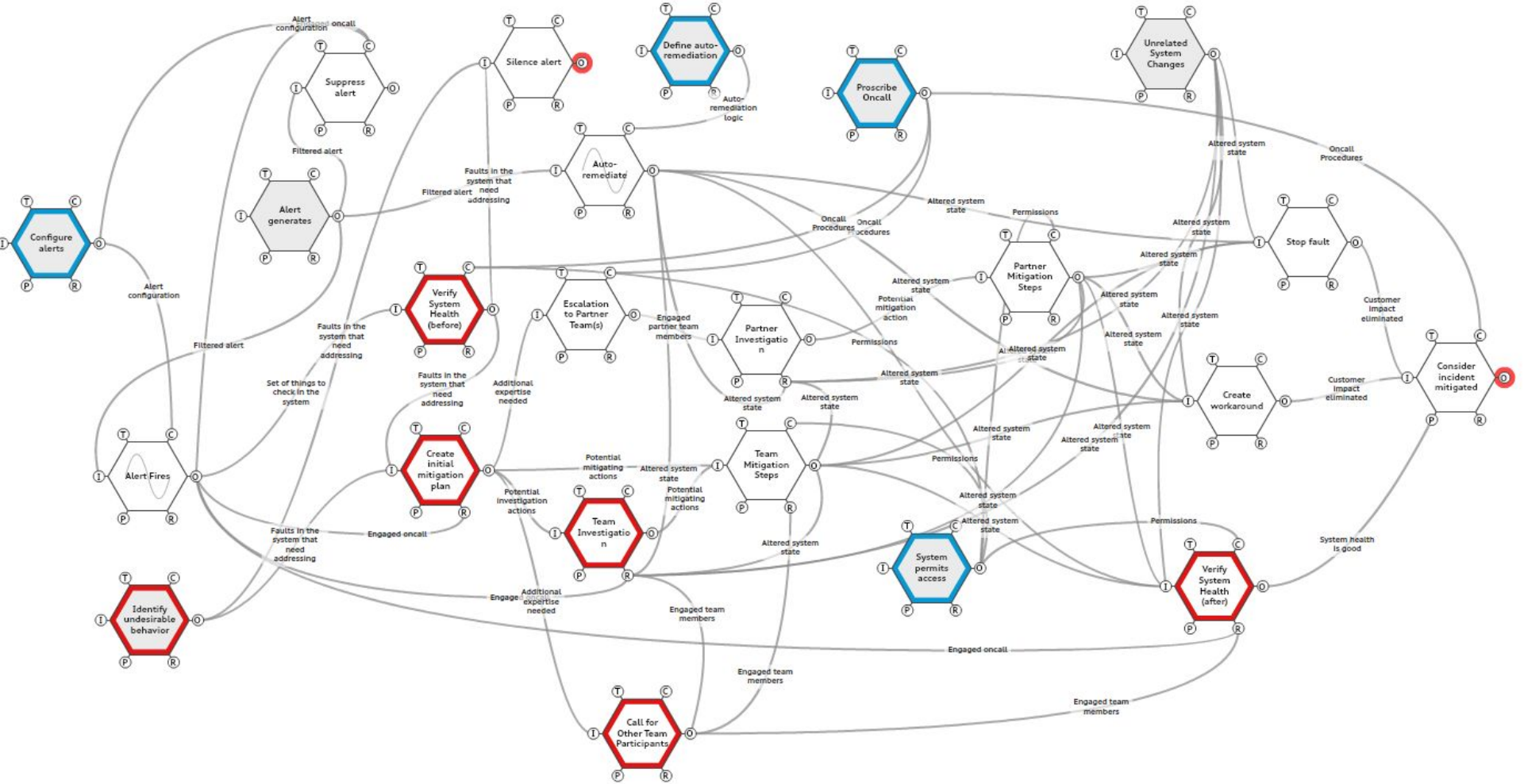


indeed

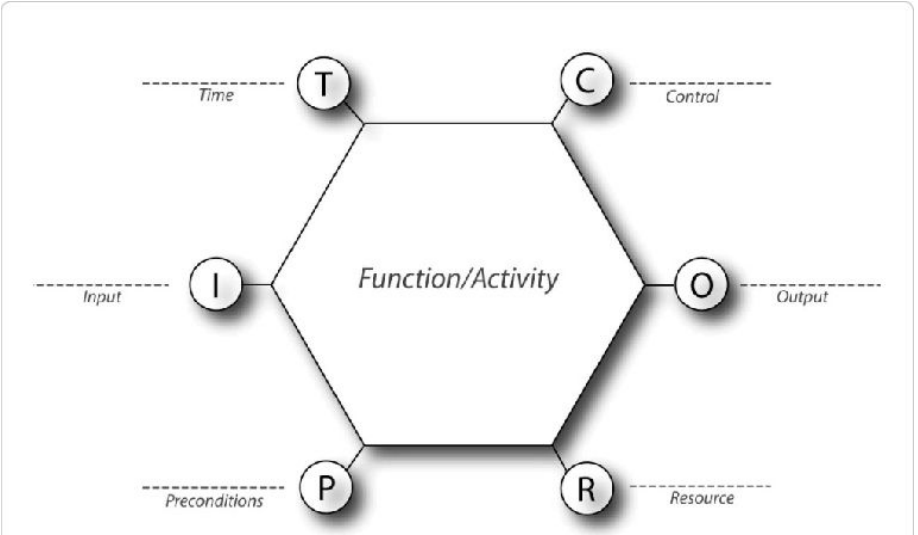
FRAM Models

“So there ARE models then!”

“Yes, but most of the value is in the making of them, apparently”

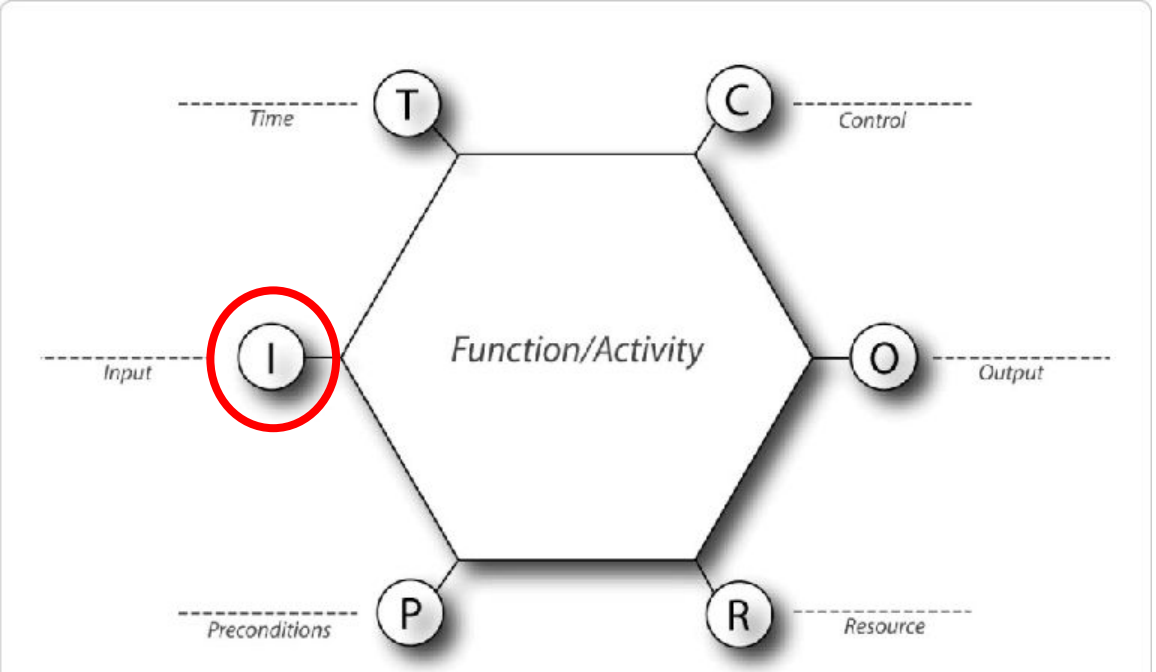

Configure alerts
Alert configuration
Alert generates
Filtered alert
Suppress alert
Alert configuration
Oncall Procedures
Silence alert
Define auto-remediation
Auto-remediation logic
Auto-remediate
Faults in the system that need addressing
Filtered alert
Proscribe Oncall
Unrelated System Changes
Altered system state
Oncall Procedures
Permissions
Stop fault
Customer impact eliminated
Partner Mitigation Steps
Potential mitigation action
Verify System Health (before)
Escalation to Partner Team(s)
Engaged partner team members
Partner Investigation
Permissions
Set of things to check in the system
Additional expertise needed
Create workaround
Customer impact eliminated
Consider incident mitigated
Alert Fires
Create initial mitigation plan
Potential mitigating actions
Potential investigation actions
Team Mitigation Steps
Team Investigation
Engaged oncall
Engage
Additional expertise needed
Engaged team members
System permits access
Verify System Health (after)
System health is good
Identify undesirable behavior
Engaged oncall
Call for Other Team Participants
Engaged team members

H E X A G O N S


Time
T
C
Control
Input
I
Function/Activity
O
Output
Preconditions
P
R
Resource

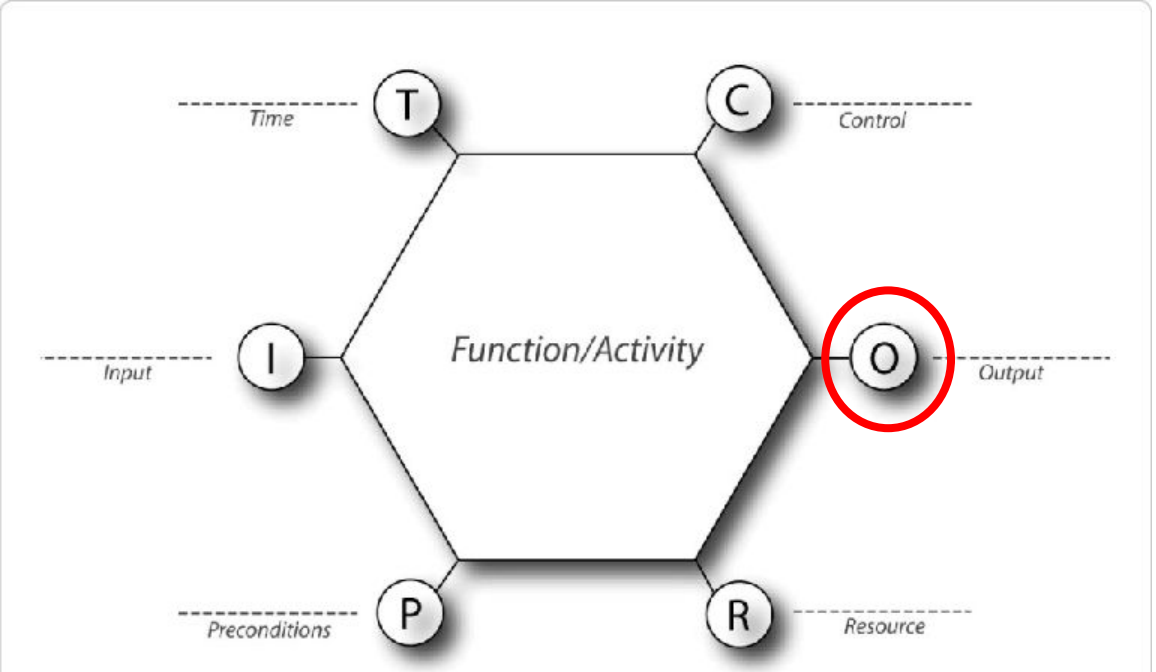
Input

“Open the Door”



Output

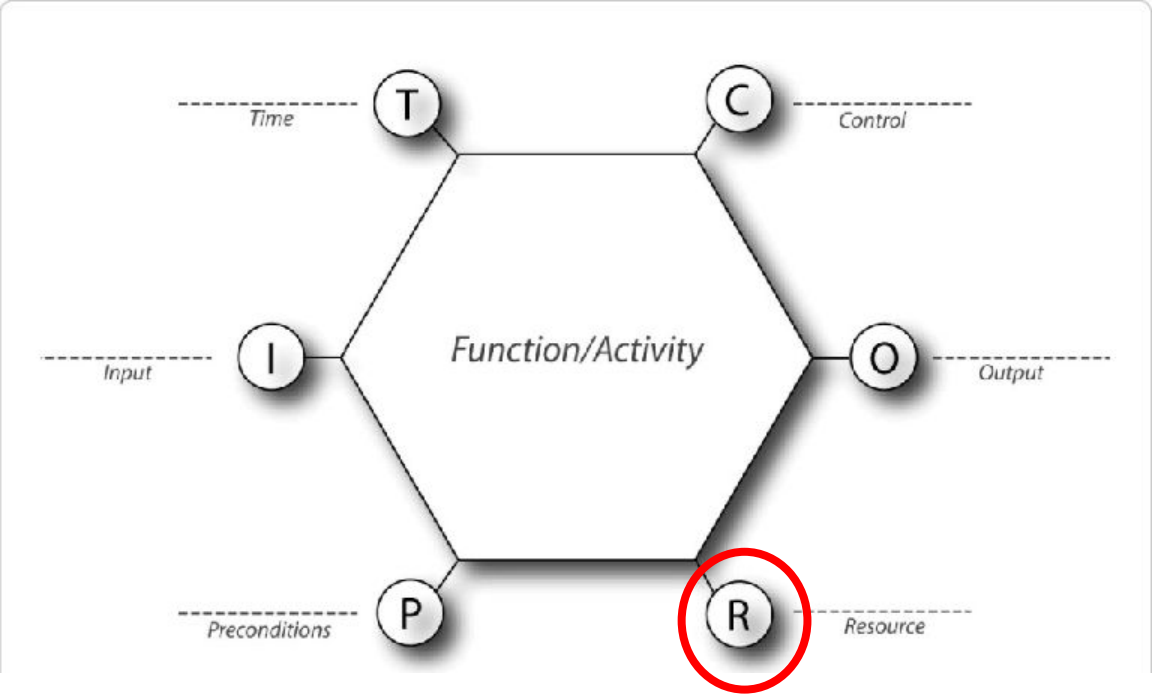
"Lock the Door"



Resource

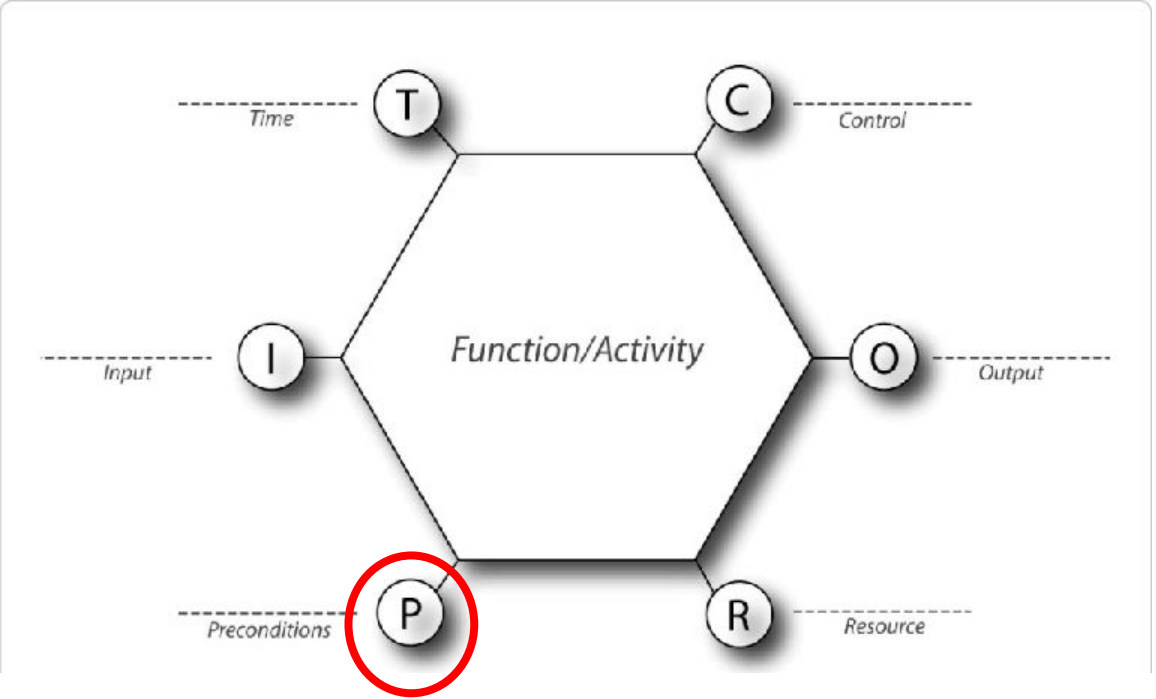
"Drive to the Job Site" ← Gas

Save the File ← Storage Capacity



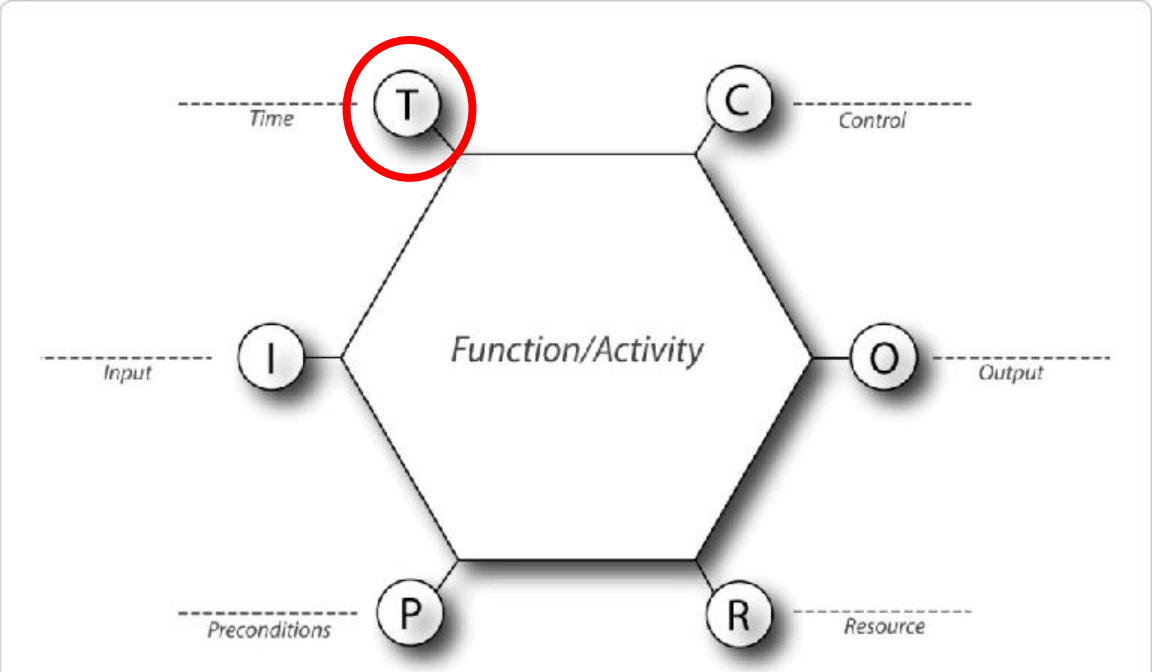
Preconditions

Close Up the Wound ← Check For All Surgical Tools / Apply Analgesic



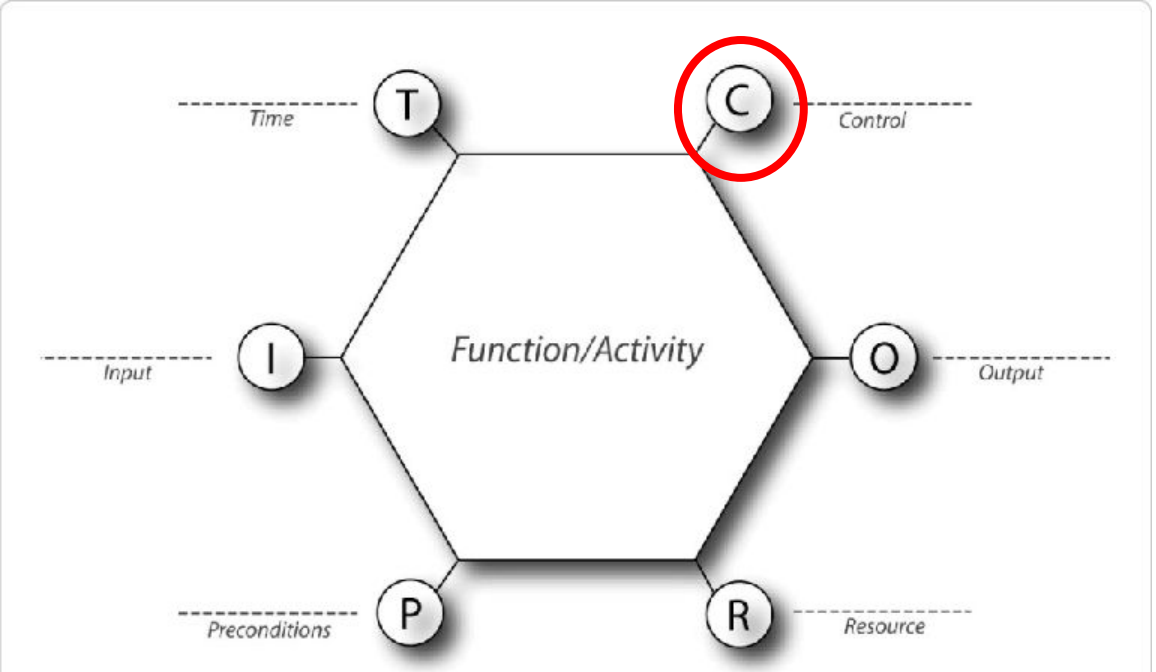
Time

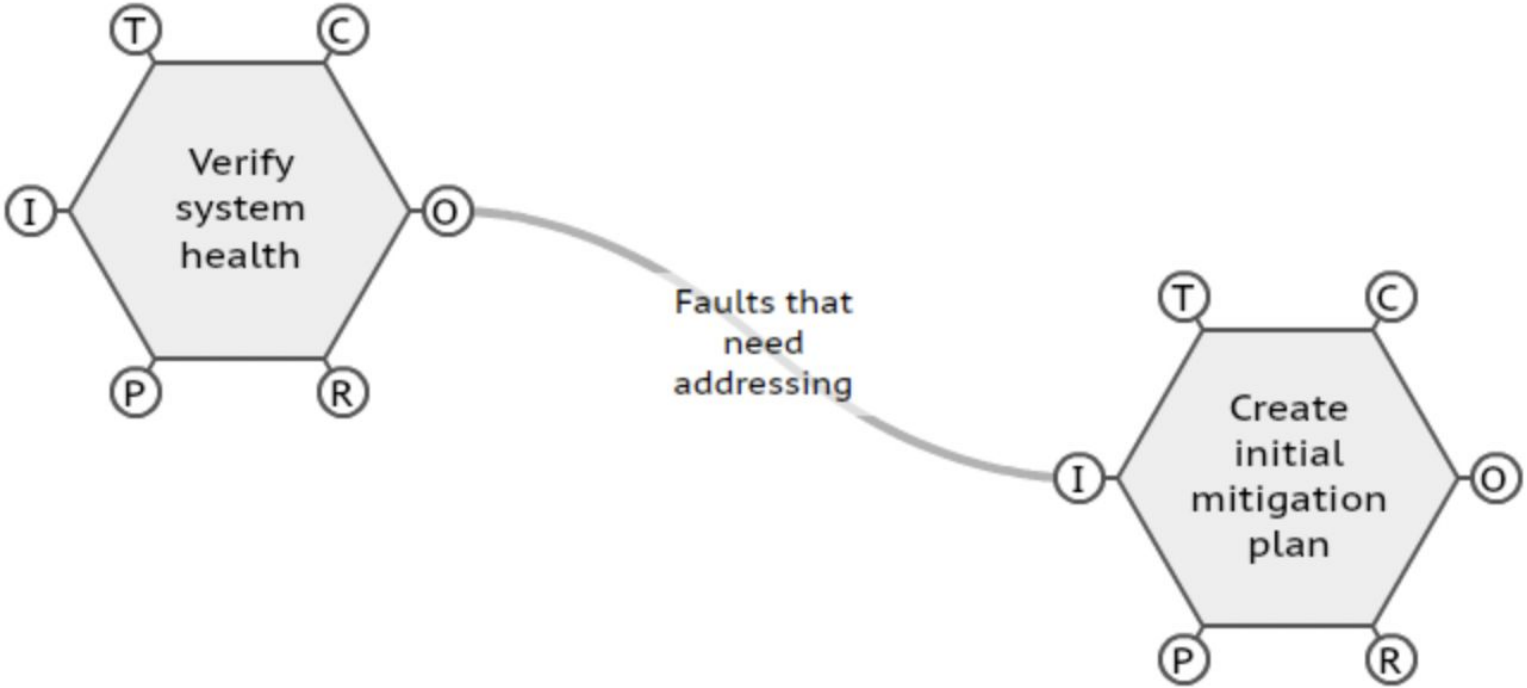
Ship Sailing Schedule

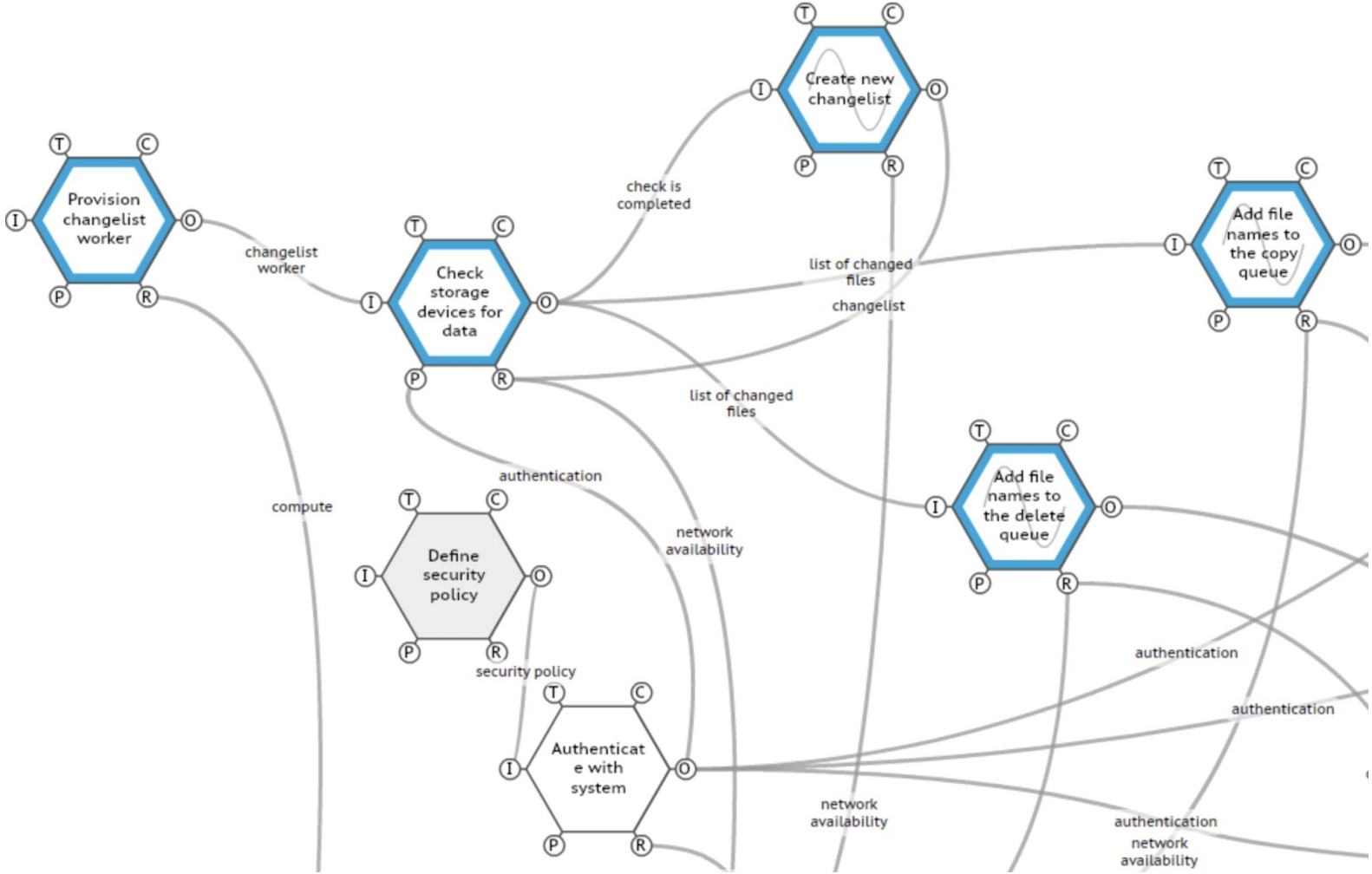


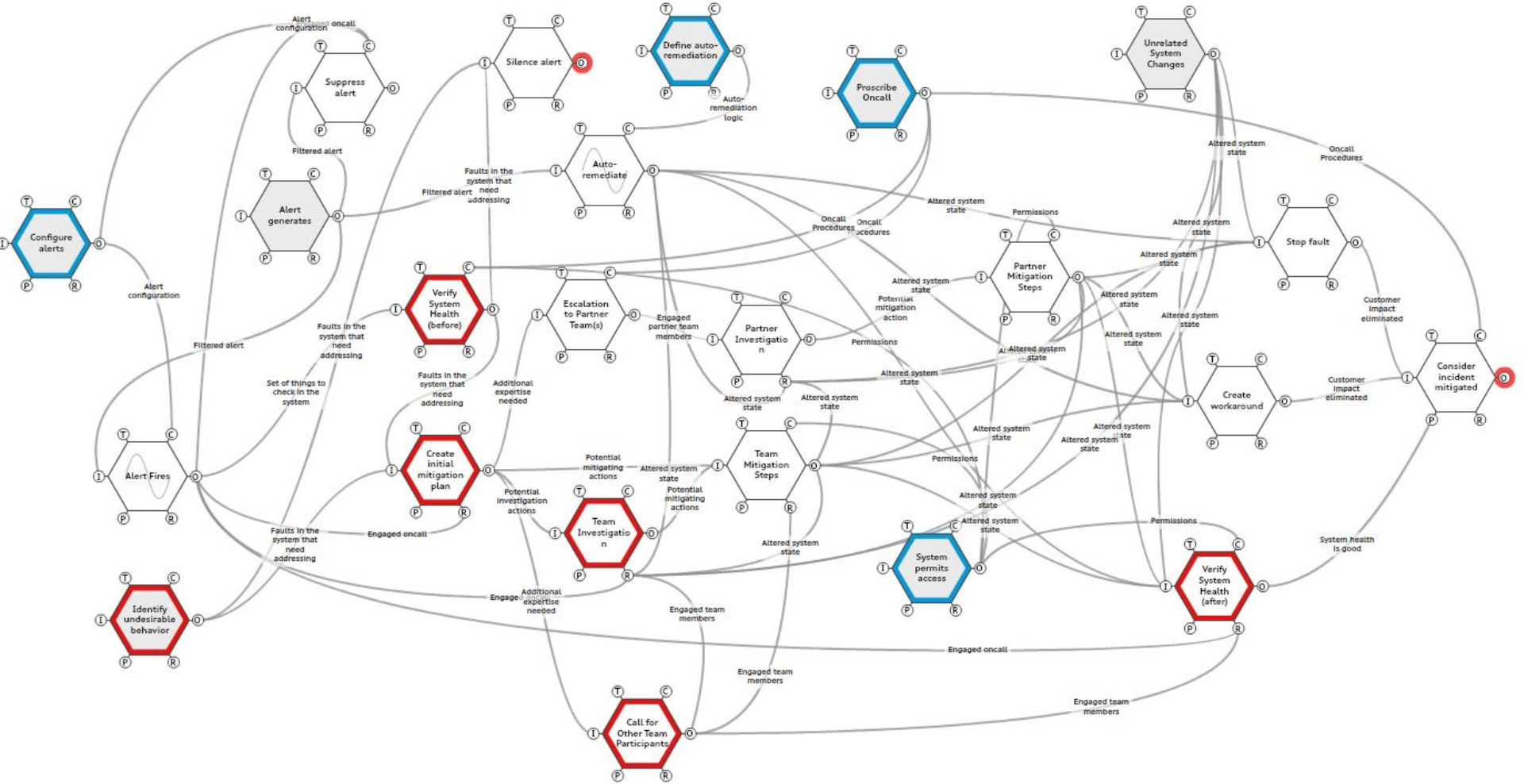
Control

- Runbook
- Ship’s Standing orders
- Code Deployment Configuration




T
C
I
Verify system health
O
P
R
Faults that need addressing
T
C
I
Create initial mitigation plan
O
P
R


Provision changelist worker
changelist worker
Check storage devices for data
check is completed
Create new changelist
list of changed files
changelist
Add file names to the copy queue
list of changed files
Add file names to the delete queue
authentication
compute
Define security policy
network availability
security policy
Authenticat e with system
authentication
authentication
network availability
authentication
network availability


Configure alerts
Alert configuration
Alert generates
Filtered alert
Suppress alert
Alert configuration
Oncall Procedures
Silence alert
Define auto-remediation
Auto-remediation logic
Auto-remediate
Faults in the system that need addressing
Filtered alert
Proscribe Oncall
Unrelated System Changes
Altered system state
Oncall Procedures
Permissions
Stop fault
Customer impact eliminated
Partner Mitigation Steps
Potential mitigation action
Verify System Health (before)
Escalation to Partner Team(s)
Engaged partner team members
Partner Investigation
Permissions
Set of things to check in the system
Additional expertise needed
Create workaround
Customer impact eliminated
Consider incident mitigated
Alert Fires
Create initial mitigation plan
Potential mitigating actions
Potential investigation actions
Team Mitigation Steps
Team Investigation
Engaged oncall
Engage
Additional expertise needed
Engaged team members
System permits access
Verify System Health (after)
System health is good
Identify undesirable behavior
Engaged oncall
Call for Other Team Participants
Engaged team members

indeed

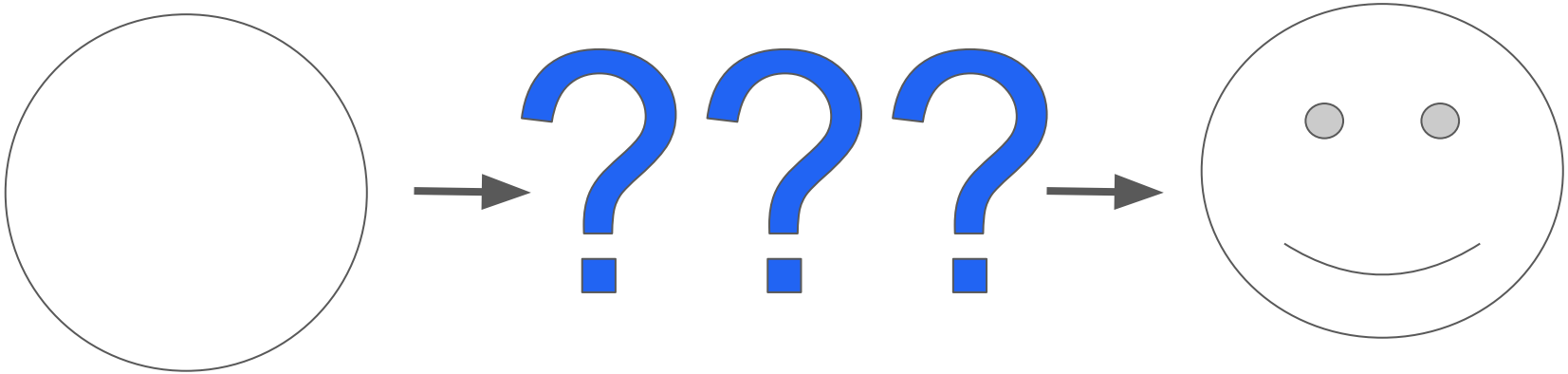
How FRAM

“How FRAM of you, sir!”

“Sorry, I didn’t mean to analyze your system like that…”

Identify & Describe The Functions

How to Draw a Smiley Face:

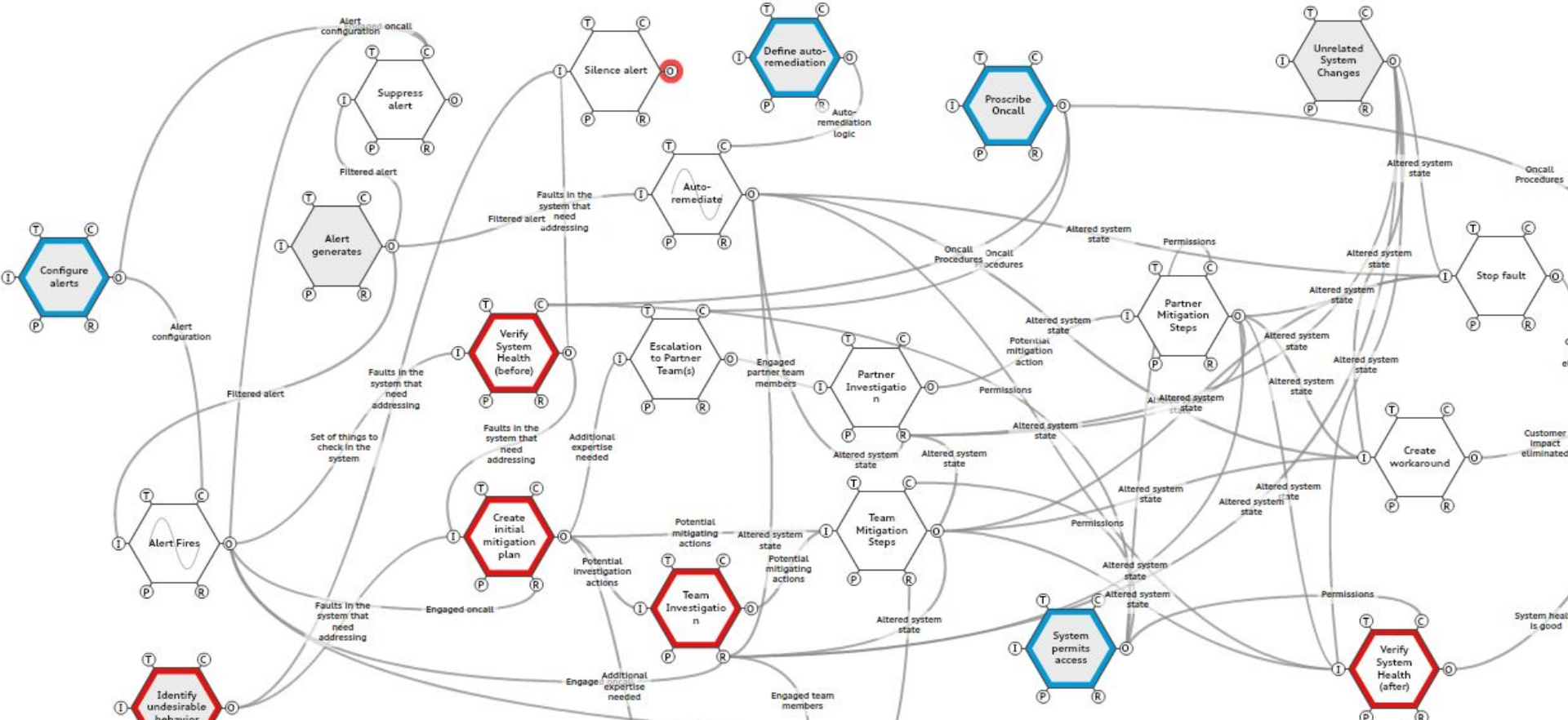


1. Draw a Circle

2. ???

3. Draw the Rest of the Smiley Face

Identify Variability



Aggregate and Categorize **Input** and **Output** Variability

1. Internal
2. External
3. Upstream/Downstream

Timing

- Too early
- Too late
- Omission

Force/Distance/Direction

- Too weak
- Too strong
- Wrong direction
- Wrong type

Object

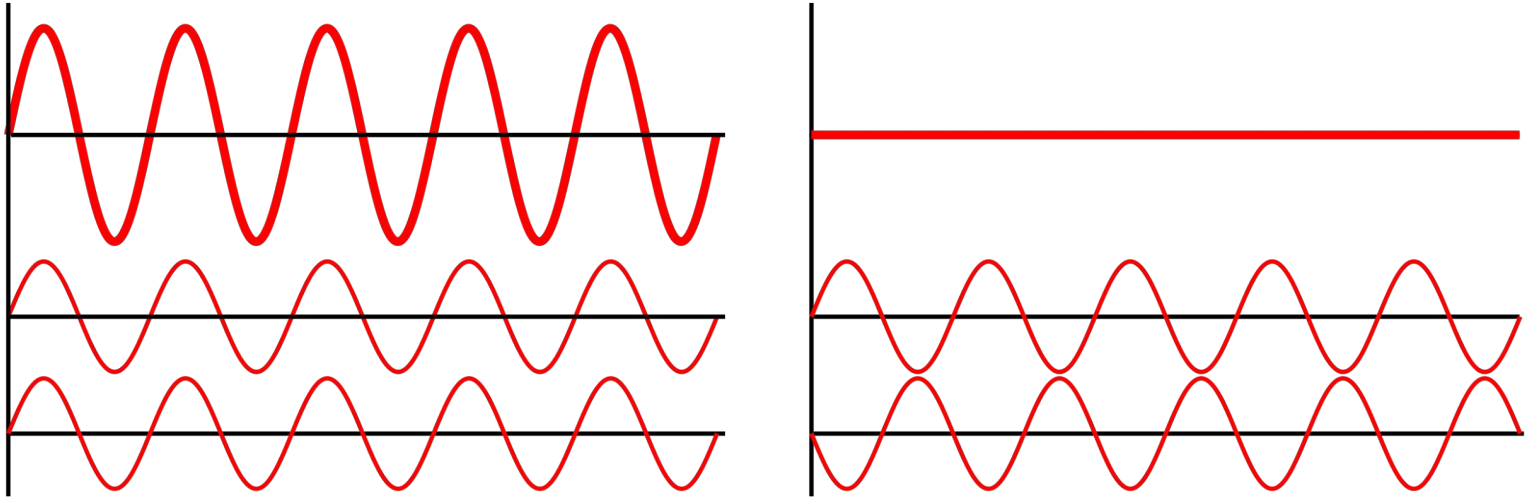
- Wrong object

Sequence

- Omission
- Skipping
- Repetition
- Reversal/Swap
- Extra wrong object

But don’t overthink the categories, eh?

How does the variability of the output affect downstream functions?



Handout!

- FRAM Visualizer Software Download
- An exercise for practicing FRAM
- FRAM website & tutorials
- FRAM book link
- ~~My soundcloud link (again)~~
- Contact information

indeed

When FRAM?

“When? Is FRAM late for something?”

“I dunno, I missed that part…”

indeed

@101010lund

Tanner Lund