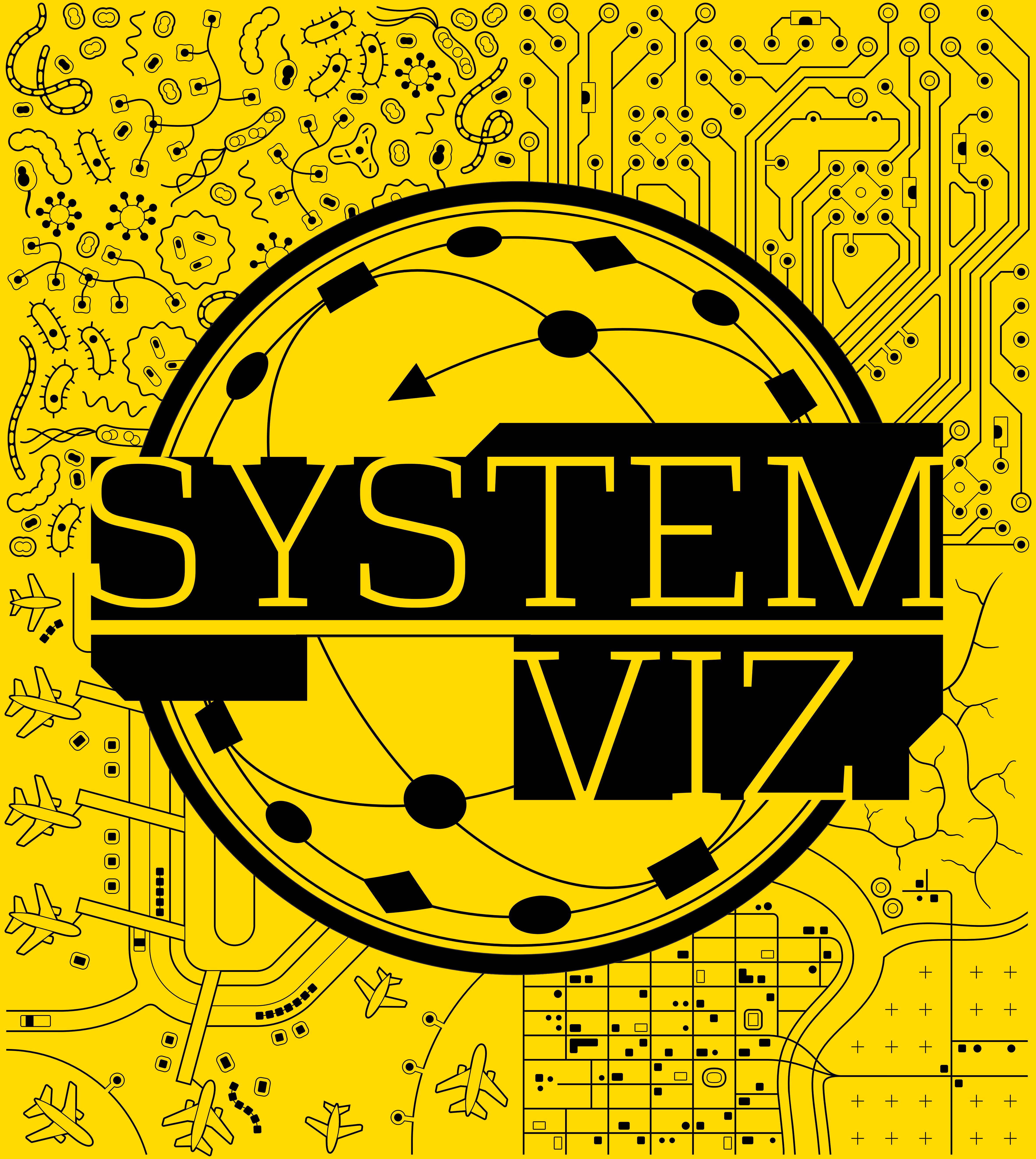
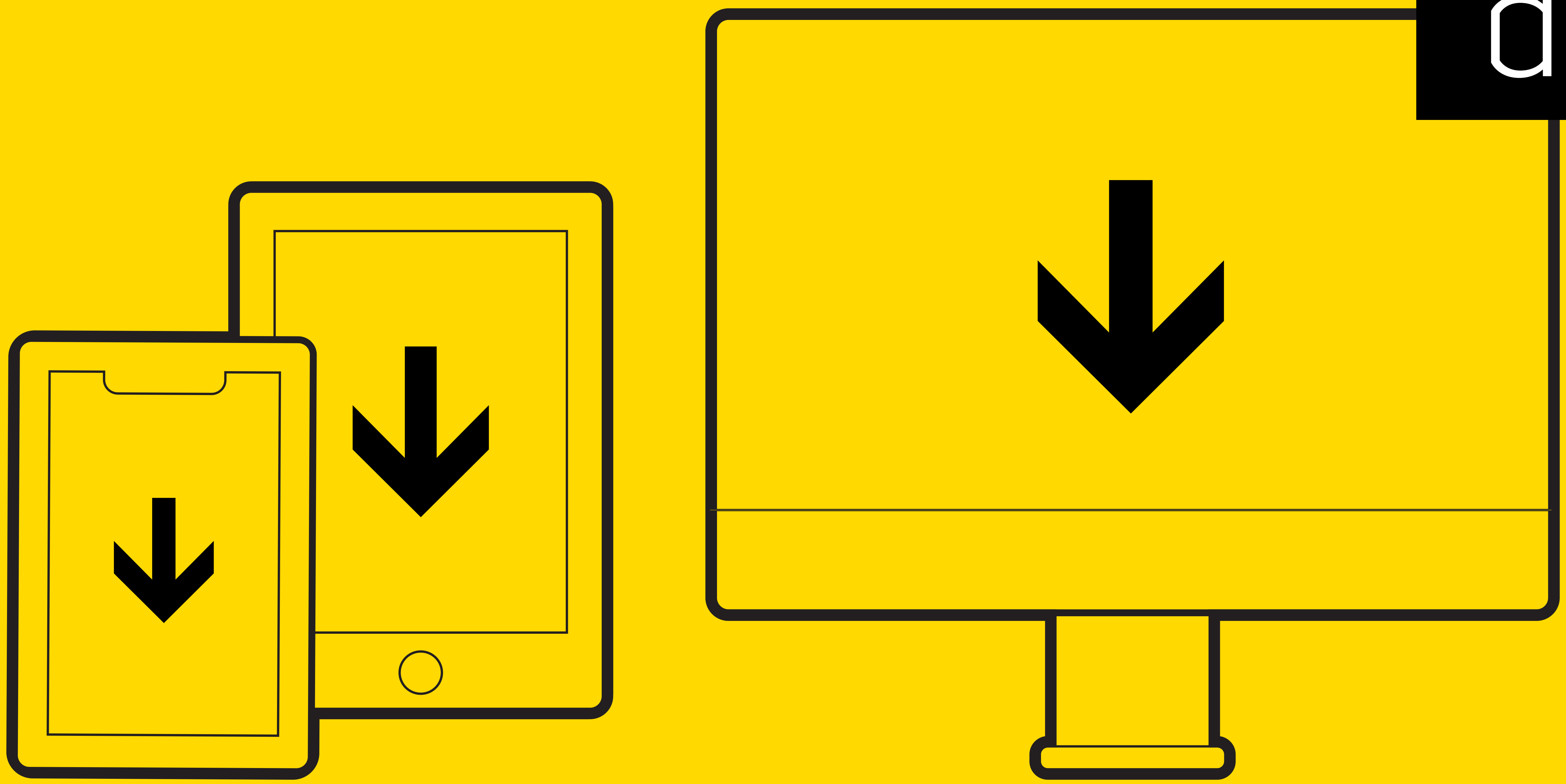


the pattern atlas of system vulnerabilities

PETER STOYKO
RSD11 SYMPOSIUM
UNIVERSITY OF BRIGHTON
BRIGHTON, UK

15 · 10 · 22

downloads



SYSTEMVIZ

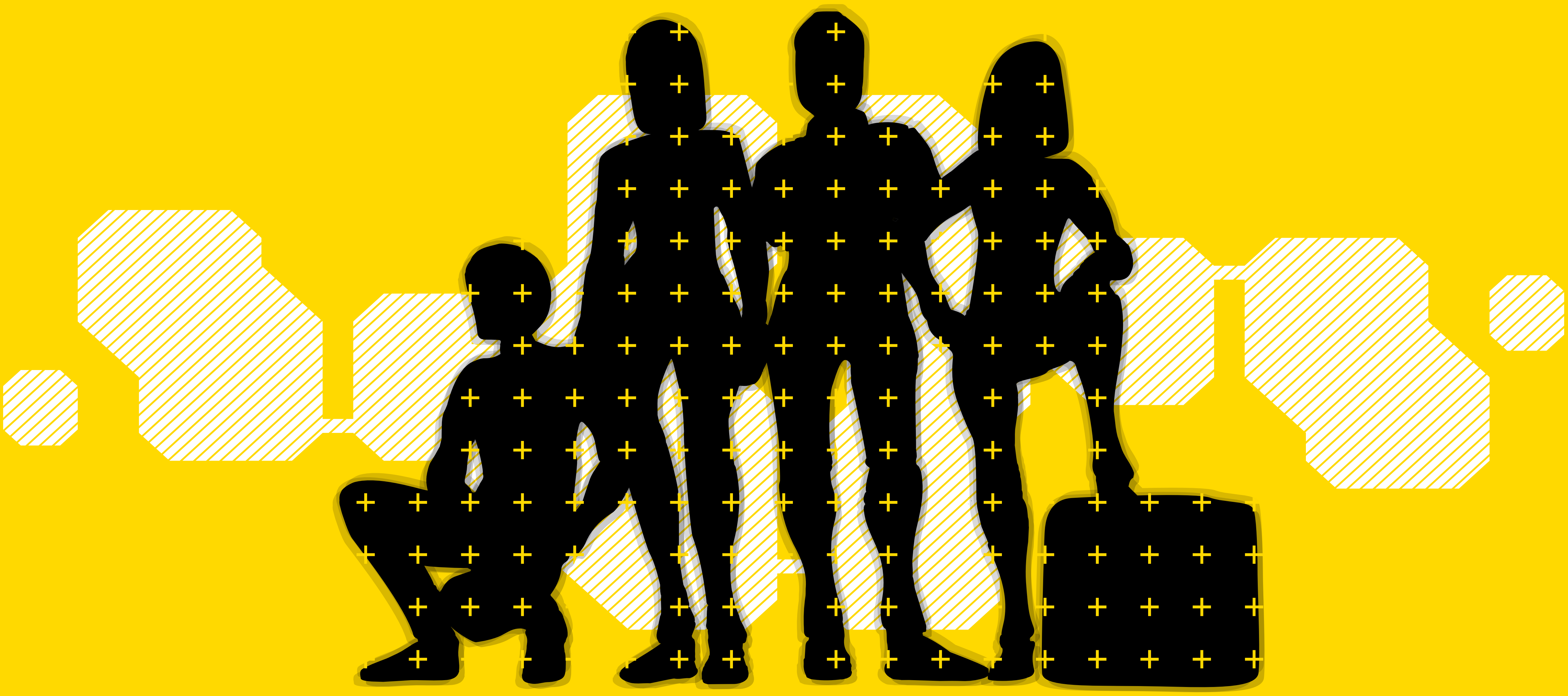
www.systemviz.com

in progress

J12

how **SMALL** players change **BIG** systems

SMALL MOVER STRATAGEMS



concept-driven inquiry

CODEX™

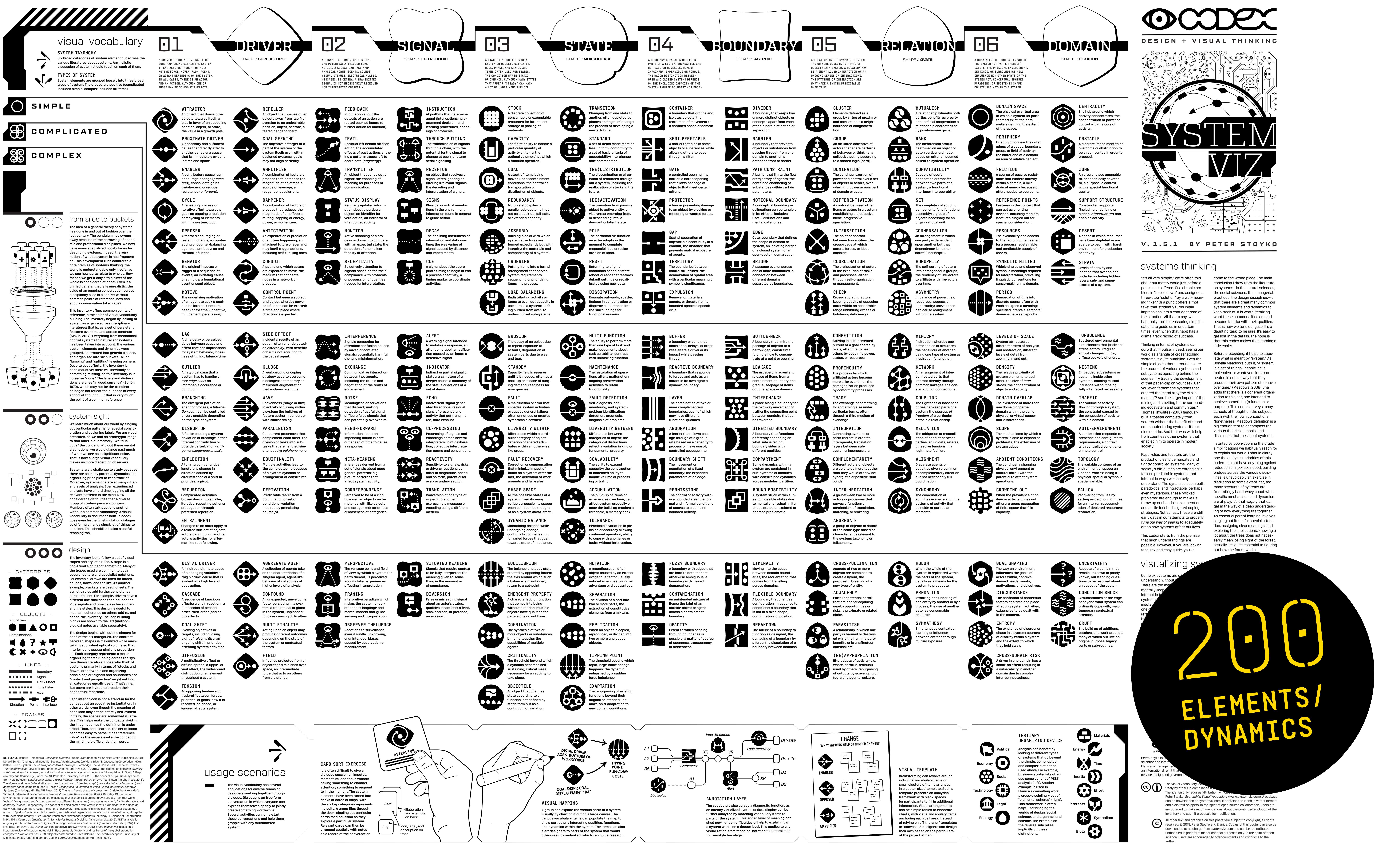
ILLUSTRATIVE VISUAL ANCHOR + SENSITIZING CONCEPTS = ATTENTIONAL SENSITIZATION


magnets for the mind


system sight

VISUAL VOCABULARY

OF SYSTEMS



CODEX
DESIGN + VISUAL THINKING
SYSTEM VIZ
V. 1.5.1 BY PETER STOYKO
visual vocabulary
01 DRIVER
02 SIGNAL
03 STATE
04 BOUNDARY
05 RELATION
06 DOMAIN
SIMPLE
COMPLICATED
COMPLEX
from silos to buckets
system sight
design
systems thinking
visualizing
usage scenarios
CARD SORT EXERCISE
VISUAL MAPPING
ANNOTATION LAYER
VISUAL TEMPLATE
TERTIARY ORGANIZING DEVICE
CHANGE
200 ELEMENTS/ DYNAMICS

gateway concepts

common language

orienting device

atlas

OF SYSTEM VULNERABILITIES

case one

case two

case three


COSTCO
GASOLINE
13'-6"
LONG HOSE
LONG HOSE
FUEL
LONG HOSE
STOP ENGINE
VYB-6619
VJC-7174

case four


酒
艺得轩
上海特产
书画装裱配框中心
靓衣轩
鸿艺阁
工厂店
山东中路
Shandong Rd.(M)
145
180
255
198
黄浦体育馆
HUANG PU GYMNASIUM
7:00-10:00
16:00-19:00


messy entanglement

control attempts

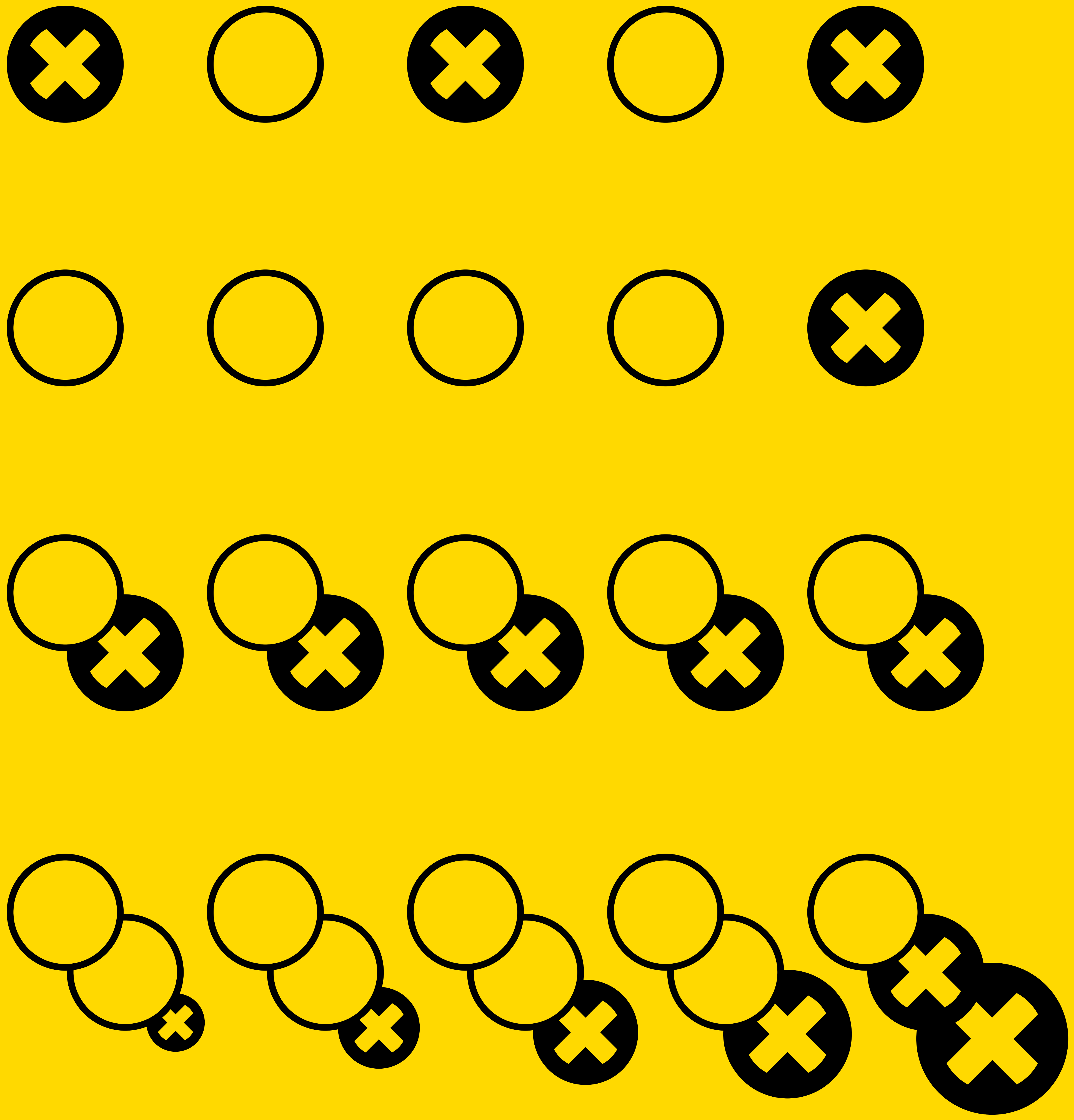
“problems”

REGULAR DISRUPTIONS

IRREGULAR DISRUPTIONS

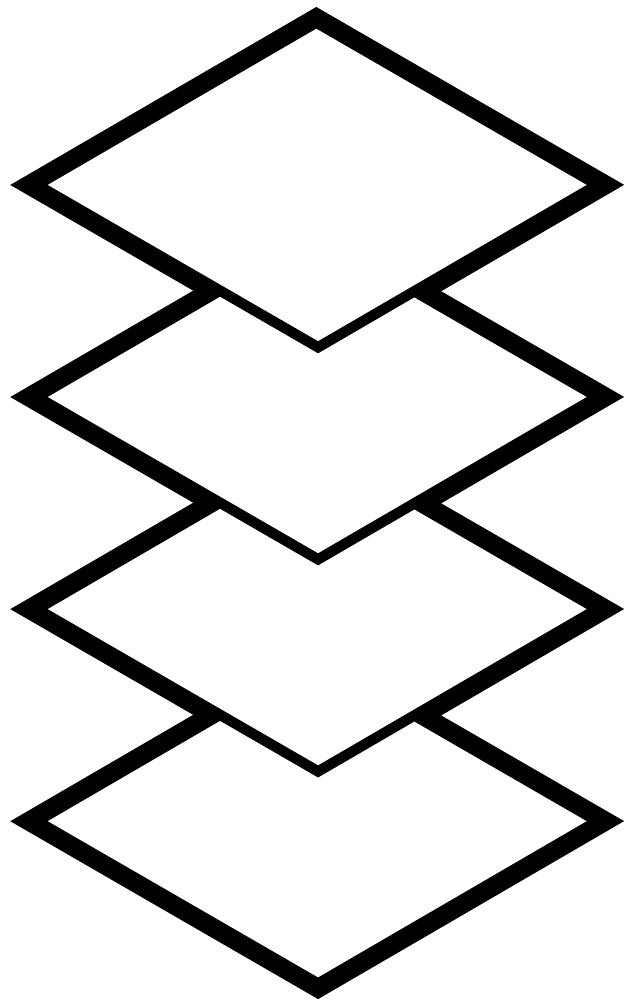
NEGATIVE EXTERNALITIES

CHRONIC DIFFICULTIES




domino effects

levels of scale



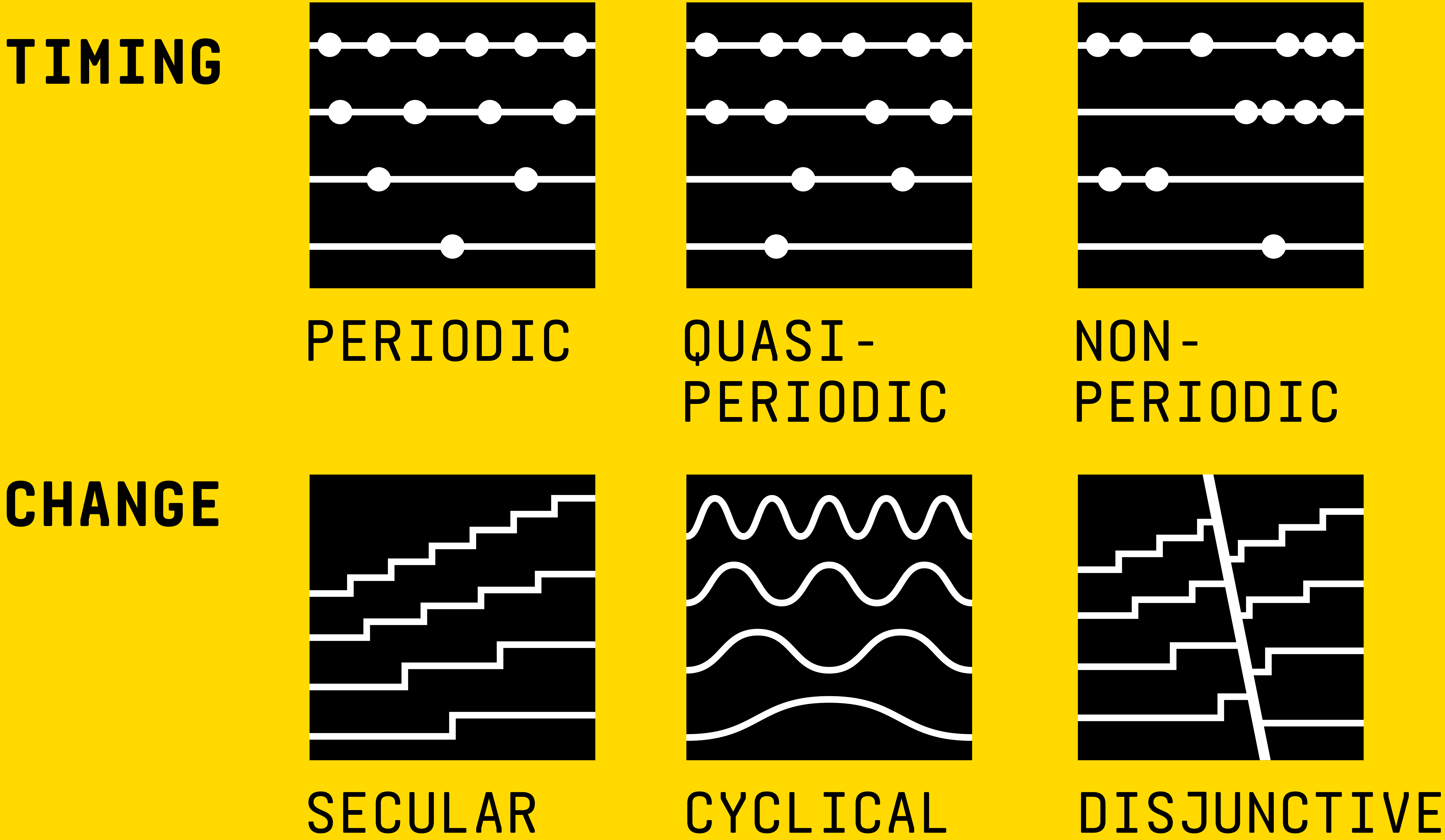
SPACE SCALES

HYPER SCALE	EXO	*Outer*
	SUPRA	*Above, Over*
	MACRO	*Very Large*
	MESO	*Middle*
	MICRO	*Near Surroundings*
	HUMAN	*Experiential Proportions*
HYPO SCALE	MINI	*Miniscule*
	MICRA	*Tiny*
	NANO	*Very Tiny*
	PICO	*Elemental*
	EXI	*Extreme Smallness*

TIME SCALES

HYPER SCALE	PETA	*Geologic Time (Chrons to Eons)*
	TERA	*Millennia*
	GIGA	*Decades-Centuries*
	MEGA	*Months -Years*
	MESO	*Weeks -Months*
	KILO	*Hours-Days*
	HECTO	*Minutes-Hours*
	DECA	*Seconds-Minutes*
	MOMENT	*Experiential Proportions*
HYPO SCALE	CENTI	*Reflex response*
	MILLI	*Neuron firing*
	MICRO	*Latency of optical computer networks*
	NANO	*Laptop (GHz) micro-processor cycle*
	PICO	*Fastest micro-processor cycle*
	FEMTO	*Ultraviolet-light wave cycle*
	ATTO	*Finest timing control of lasers*
	ZEPTO	*Electron oscillation cycle*

pace layers


TIMING
PERIODIC
QUASI-
PERIODIC
NON-
PERIODIC
CHANGE
SECULAR
CYCLICAL
DISJUNCTIVE

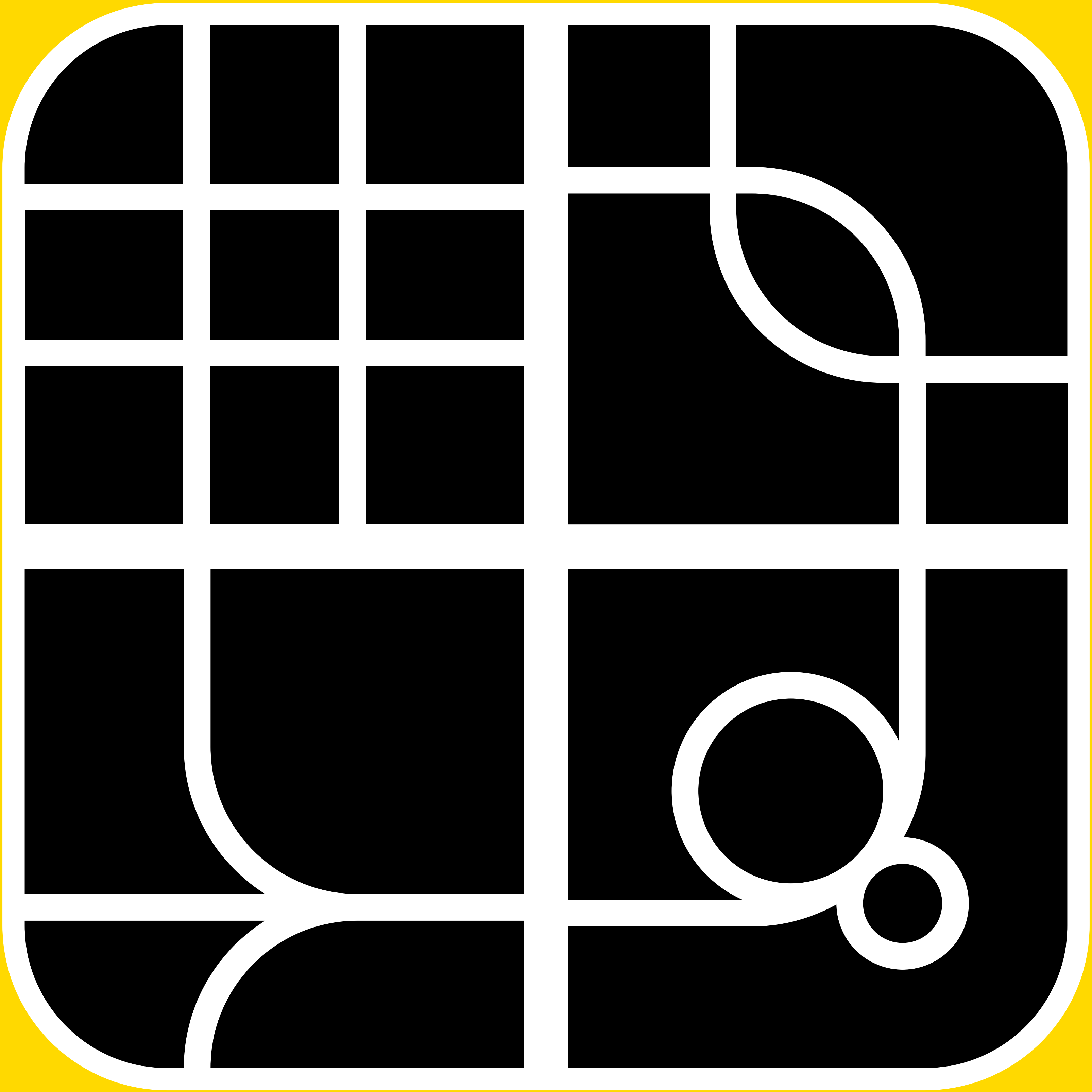
tangle metaphor

MESSES

COMPLEX SYSTEM INTER-RELATIONS

FIBERS

ROUTINES AND SUB-ROUTINES



ENTANGLEMENTS

CROSS-SYSTEM DYNAMICS

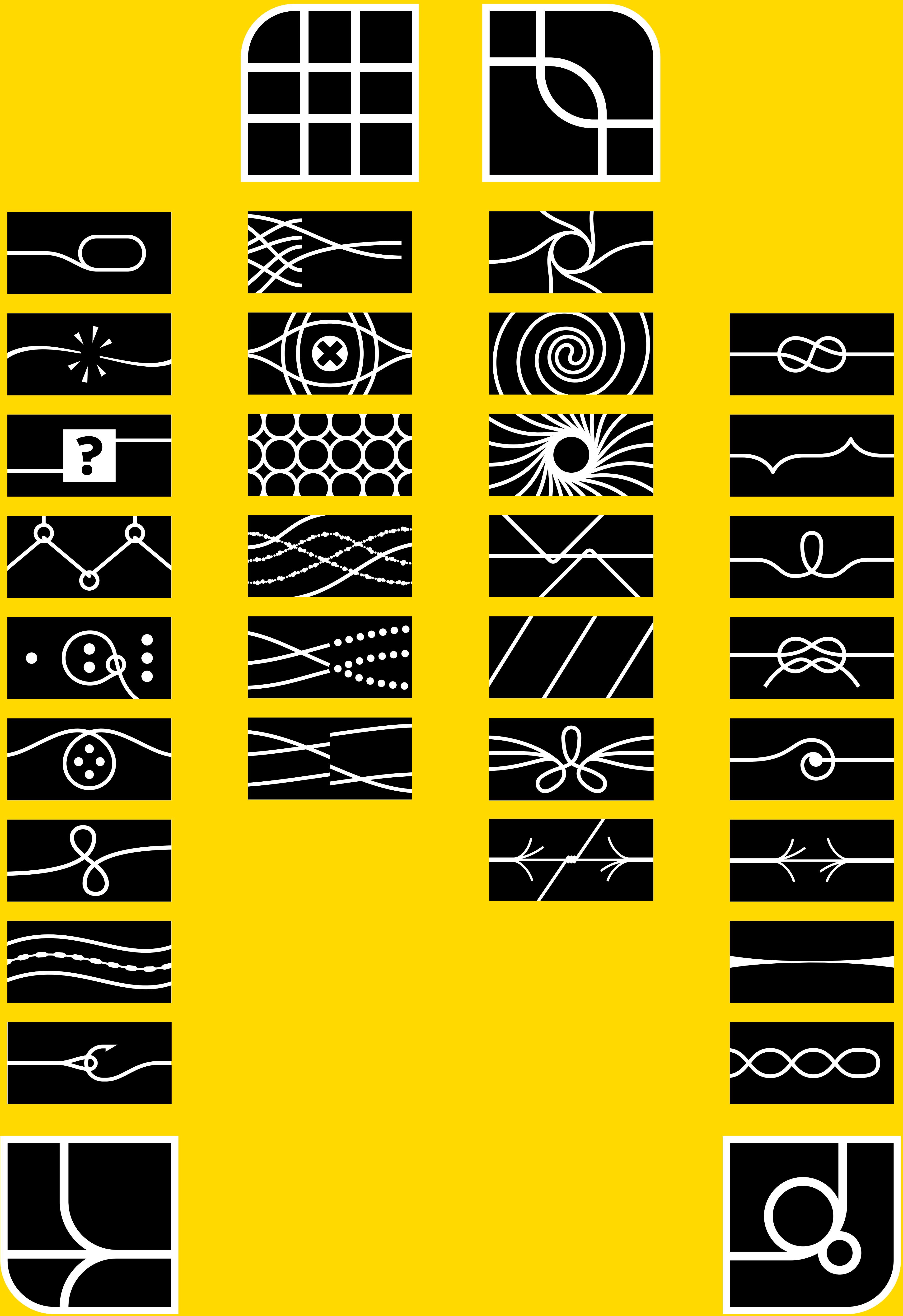
THREADS

SYSTEMS AND SUB-SYSTEMS

vulnerabilities

4

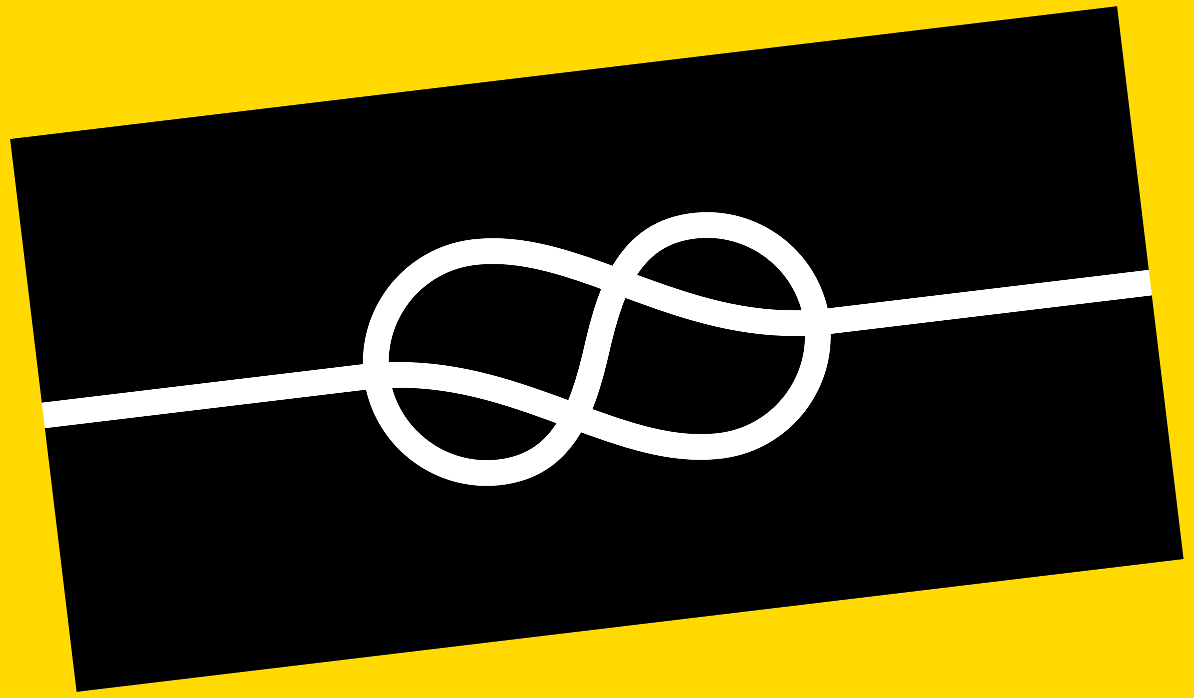
LEVELS
OF SCALE



30

TYPES

visual placeholder



(KNOT)

\+ "CRUFTY" = SYSTEM CRUFT & KLUDGE

ICON & VISUAL ANALOGY

SHORTHAND DESCRIPTOR

CAUSE OF VULNERABILITY

atlas


D' ZUYDT ZEE
D' STILLE
D' PACIFICUM ZEE
TERRA ESONIS
NEUW HOLLANDT
D' INDISE ZEE
OCEANUS ORIENTALIS
OCEANUS AETHIOPICUS sive MERIDIONALIS
ANTARCTICA SECRETA
MERIDIONALIA ARCANA
NEUW DANIA
Tropicus Cancri
Tropicus Capricorni
den Langsten Dach
den Korsten Dach

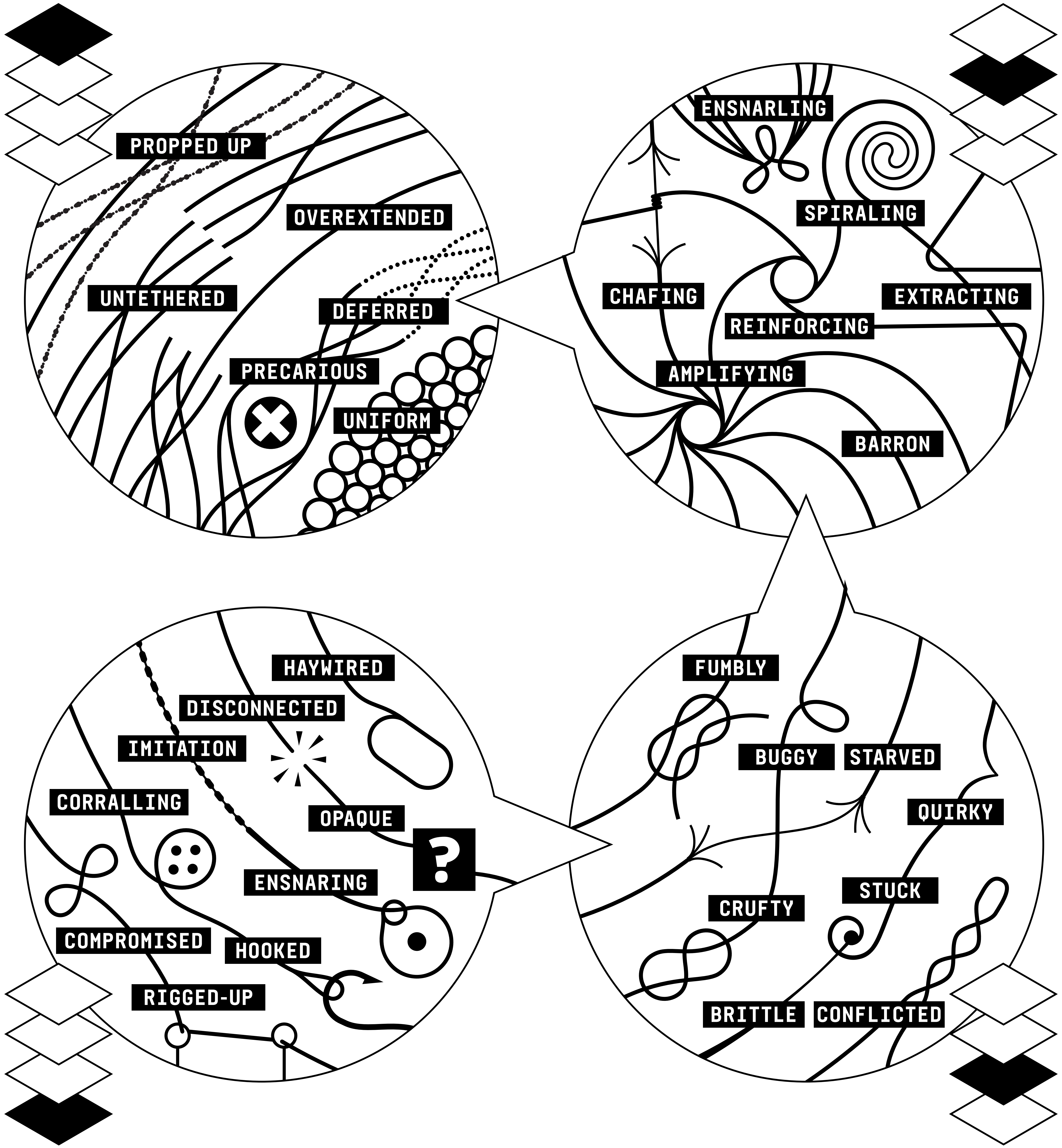
atlas

MESSES

COMPLEX SYSTEM INTER-RELATIONS

ENTANGLEMENTS

CROSS-SYSTEM DYNAMICS



FIBERS

ROUTINES AND SUB-ROUTINES

THREADS

SYSTEMS AND SUB-SYSTEMS

atlas

OF SYSTEM VULNERABILITIES

V. 1.0.0 BY PETER STOYKO

system entanglements

As Donella Meadows explains, "A system is a set of things — people, cells, molecules, or whatever — interconnected in such a way that they produce their own pattern of behavior over time."[1] Few systems are entirely "closed" insofar as they do not draw inputs from elsewhere; most rely on resources or signals coming from outside their outer boundaries in order to operate. Thus, systems are not just organized wholes made of interconnected parts but are connected to each other in ***elaborate tangles***. Some entanglements are tightly-coupled, non-substitutable, on-going dependencies. Others are loose, tenuous, short-lived interactions. A lot of relational variety can exist between those poles. When we talk about a broader amalgam of systems (such as a society, economy, organization or ecosystem), we are talking about complex, evolving webs of entanglement.

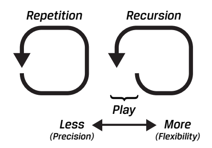
Once a society organizes itself beyond a minimal level of complexity, the webs of entanglement become an ***irresolvable mess***. A "mess" is Russell Ackoff's term for evolving relations that are too numerous, varied, and obscurred to be coordinated.[2] Each system has unpredictable knock-on influences on other systems. Messes can be a good thing as variability is a source of resilience in complex environments, a disruptive event is less likely to affect all systems and interrelations in the same way. There is also more experimentation and intellectual diversity to act as a hedge against uncertainty. Most disruptions and chronic difficulties are less spectacular. Indeed, system glitches and gotchas frustrate us on a daily basis. The rhetoric of technological progress promises a future of effortless convenience and human flourishing. Instead, manipulative and error-prone systems are ever more proscriptive and needs are less wiggle room for coping. Such shortcomings create the ability to control the events around us. Lapses in mediating cumulative impacts of behind-the-scenes harms make the mess less "livable." When cascading system failures happen amid gradual, widespread decline, there is a serious risk of general collapse.

The *Pattern Atlas of System Vulnerabilities* itemizes the different types of vulnerability inherent to elaborate, entangled, human-made systems. It visually explores the idea of messy entanglement to identify the particular patterns that cause trouble. The mess is broken down into four ***levels of scale***. Different dangers are visible by zooming into each level. Some vulnerabilities are the product of messes as a whole (▦). Others are caused by interactions between nominally independent systems (◩). Some vulnerabilities exist within individual systems and subsystems (◪). And some persist among routines and sub-routines (◫). In keeping with the metaphor, visuals are made of contorted and tangled threads.

Cascading errors and confounds at one level of scale may trigger problems at other levels. That is how a tiny mishap gets amplified into a massive disaster. For example, a software bug in the alarm system of a company's control-room once triggered a blackout that lasted for days, not just in the local area, but across a large swath of North America. A botched software update to an Internet router at a telecommunications company caused Canada's entire debit-payment system to stop working for a day, with a third of all mobile phones losing coverage too. A deadly bat virus infected a wild animal sold in an urban market in China, with the contagion spreading to humans, many of whom then traveled internationally, causing systems in various countries to shut down for months to limit further spread. Then it happened again on a global scale.

the (n)ever-changing world paradox

What causes systems to ***change***? Systems are not static but are full of moving parts. So what causes a system to work differently? To have different goals? To adapt to new circumstances, cater to different needs? When complaints are made about systems being rigid, obsolete, or harmful, that is the sort of question raised. However, given all the energy and effort needed to keep everything going, a more relevant question may be: What causes a system to remain more or less stable? Come to think of it, how stable is each system? Reversing the onus of the question can reveal much more about a system's potential to evolve.



Human-made systems are often made up of operational routines and cycles of routines. We think of ***routines*** as fixed, automatic, and repetitive sets of tasks. Yet they do not repeat per se, they recur: routines have the potential to be somewhat different with each implementation depending on how precise the operations have to be implemented. Conformist social systems can also have little tolerance for variation that breaks cherished norms, customs, or rules. A system's recurrence can seem like a consistent pattern because we usually lack fixed reference points with which to gauge change over time. Our inherent *change blindness* (see opposite panel) and the sheer volume of recurrence mean incremental changes escape our attention unless they are tracked using special data. Even so, our impression of system change is highly selective. Some parts always seem to be always evolving. Other parts seem stuck in the same recurring patterns. The conflicted sense that "every thing changes, every- thing stays the same" is called the *(n)ever changing world paradox*.[3]

Change efforts usually focus on decisions made by high-ranking decision-makers. Top-down decisions-making has much less control on elaborate systems than is commonly assumed. Much change (and change-averse downplaying) happens among continually adjusting routines and subroutines. Even high-reliability organizations (such as nuclear powerplants) struggle to manage variability. Routines executed by humans are inherently variable and may not be strictly prescribed to begin with. Does more effort go into routines that are supposedly different situations? Even so-called "low skill" routines are "performative," demanding keen attention to details and impromptu adjustments. A routine may not get the job done under the circumstances, triggering a "repair" of the routine or a switch to another one. New routines may be invented in response to emerging regularities. In undirected social systems, many routines are imitative, habitual, and loosely framed. Social interactions across the system may settle into more or less stable patterns but the ***underlying churn*** causes both gradual drift and the occasional abrupt shift.

Conversely, routines may adapt so that systems stay the same more generally. Do evolving routines maintain the ***status quo*** by allowing a flawed system to persist? Or are changes gradually altering the system's overall character and purpose, turning it into something else? It can be hard to tell when caught up within a system's operations and lacking the critical distance to see things from a broader perspective.

As systems mature, routines become ***routinized and formalized***. Interdependencies build up across systems and subsystems. More effort is spent conserving system functions from disruption. Underlying routines may be more constrained but nevertheless retain some play. Indeed, systems that are too rigid risk becoming brittle and irrelevant. Vulnerabilities and chronic difficulties build up within and between systems, which create the potential for large-scale harms. A major shock can free up a lot of activation energy to enact change. Exploiting broken systems and coping with their fall-out also offers possibilities for change for those who are nimble and resourceful.

FIBERS [Routines and sub-routines]

HAYWIRE

Routines that are circular, contradictory, or result in dead-ends are called *Kafka circuits*.[4] They are usually caused by an unforeseen irregularities or careless design. They result in ordeals for system users; that is, burdensome sub-routines to resolve matters that are full of extra tasks and delays, any of which may lead to further errors. Ideally, administrators have enough discretion to step outside of a routine to implement a pragmatic, one-off fix. Not every situation can be prefigured and turned into a set routine. Many systems do not allow for pragmatic fixes due to automation or because discretion opens the door to petty corruption. Without clear avenues for resolution, those caught up in an ordeal have to fend for themselves (hence the Kafka reference). Sometimes an "ordeal mechanism" (or "micro-ordeal") is added on purpose to discourage system use (or overuse) for political reasons or to manage capacity.

DISCONNECTED

As systems become compartmentalized, they rely on hand-off routines between modular subsystems. That comes with the risk of botched hand-offs. For example, information may not be shared between subsystems to provide a continuous flow of successive routines. Those caught up in the system may have to re-explain their predicament over and over and get handed over as they are bounced from one routine to the next. The risk of data-entry errors and other glitches increases. Sometimes a case will fall through the cracks, stuck in a state of limbo. Ideally, a system creates a "seamless" experience whereby transitioning between sub-systems becomes invisible to observers. However, ruptures may be overlooked if hand-off will be disorienting, so it is not clear what triggered the error. Thus, some systems are made "seamful" whereby the hand-offs become more overt and elaborate, usually with other costs.

OPAQUE

Routines may be automated for efficiency. "Black-box" automation converts operations into elaborate algorithms, with the inscrutability of software code and statistical techniques making it difficult to see what is going on. Automation also introduces rigidities, for often there is no substitute for human judgement and discretion. Dodgy decision-making gimmicks and shoddy work can get encoded into algorithms that are too hard to scrutinize. Algorithms are becoming laundered through algorithms that are hard to scrutinize. Algorithms are becoming more data intensive with data drawn from sources with poor quality control and dubious provenance. Ideally, technology augments human control and offers information. Yet the trend is towards removing human steering entirely. When algorithms that have been trained on flawed data, the operations become opaquely overcomplicated, glitches become nearly impossible to trouble-shoot.

RIGGED-UP

Many routines are designed to make the job easier for system administrators while adding burdens on users, whose needs and circumstances are an after-thought. Some routines encourage over-use or misuse of the system in ways that benefit vested interests, administrators included. For example, as Ivan Illich argues, the medical system encourages over-consumption and many treatments are designed for the convenience of providers regardless of their suitability for patients.[5] That results in an increase in provider-induced harms (*iatrogenesis*), including routine mistakes, unnecessary risk-exposure, and the force-fitting of standardized routines onto different problems. Some treatments may be based on contrived needs and promote unhealthy dependence. These sorts of vulnerabilities can be difficult to detect because they serve the narrow administrative conveniences and violate the mental models of those running things.

ENSNARING

Systems designed to indirectly control human variability rely on behavioral manipulations. Some routines psychologically "nudge" people into making particular choices using subtle prompts. Other routines "gamify" activities by inducing addictive habits with the emotional satisfaction of token rewards, such as points or "likes." Some attunement to human psychology is inevitable to better align systems with the ways humans naturally think. Even so, these "choice architectures" second-guess decisions for entire segments of people. The potential for exploitation never goes away. Indeed, dark-pattern nudges and addictive habits are common. Outdated, dysfunctional nudges are a form of cruft called "sludge." Competing manipulations create unanticipated and contradictory behavioral patterns. Leeriness builds up and defensive behaviors emerge (*reactance*), which cause future manipulations to perform erratically.

CORRALLING

Both coercions and indirect manipulations can create herding behaviors that commit large swaths of a population to the same behavioral patterns. A lack of variability can be a vulnerability, as diversity is a hedge against uncertainty and disruptive events. For example, routines that coax everyone into a few dietary habits can be risky if evidence about what is healthy and sustainable continues to evolve. Even if routines herd a small share of the population, any induced harms to the cohort can have knock-on effects elsewhere in the system. That is the danger of "pushes," or behavioral manipulations combined with coercive measures. Unlike with "nudges," optionality is replaced with "managed autonomy," or constrained behavior that is strictly set to proscribed limits. If those limits are too narrow and ill-informed, then the herding behavior may causes those involved to become blindsided by unforeseen dangers.

COMPROMISED

Routines may not be specified for functional reasons but are negotiated compromises. Such *truces* may reduce internal tensions, even if no faction is fully satisfied (the *joint-decision trap*) and one side is placated only half-heartedly. Conflicted administrators will "selectively perform" in ways that favor certain goals over others. Pretenses and token sub-routines may be added to give the false impression that a system is fulfilling certain functions while merely going through the motions. Defensive routines may build up to preserve a truce or protect turf. Those can block information signals that are crucial for a system's operations. Some routines may become "sanctified," imbued with a larger meaning by a faction, causing any tampering or question- ing to be opposed. The build-up of compromised routines cause systems to lose their bearings.

IMITATION

It's often assumed that there is a "right" way to doing things—a routine deemed a "best-practice" or "gold standard." *Anti-patterns* are approaches with a poor track record but are relied on anyway because of myths and prestige surround- ing them. It can be tempting to copy a well-regarded routine regardless of how suitable it is. Even if a routine shows promise, the system may not have the capability to implement it properly. In any case, such *wannabe routines* present a dilemma. Is the routine implemented, even if inappropriate? Or does everyone just go through the motions, while blocking any attempt to scrutinize short-comings too closely (*shielding*)? Are inconvenient or difficult parts dropped, even if that misses the whole point of the routine? Pantomimed, ill-fitting routines waste effort and disguise

HOOKED

Routines may become *captured* by outsiders or another system. In other words, independence is comprimised as interests interfere with system functioning, often in subtle and self-serving ways. For example, a routine may develop a dependency on an interest group for information or capabilities, exposing the system to outside pressures. Cosy relations may form between those working in a system and those regulated by it, causing favorable treatment within ostensibly neutral routines. A routine may pander to existing beneficiaries in a way that discriminates against prospective ones. These corruptions can shape a system, making its own logics and assumptions inseparable from those of outside interests. Thus, interests get their hooks into the system even if the stated goals of the system overall do not change.

MESSES [Complex system inter-relations]

OVER-EXTENDED

Attempts to simplify messy entanglements of systems into an intelligible coherence, instead of accept the messy world as it is, soon hit limits of capability. Making messes easier to interpret inevitably relies on "thin simplifications," or crude classification schemes that do a poor job of accurately capturing the salient (and continually evolving) variation.[8] Government and corporate systems attempting oversight tend to be fragmented patchworks of capability, with less capable units dragging down more capable ones. Interventions trying to streamline messes often wind up adding more complexity and unanticipated dynamics. A "clean up" is not even possible as amalgams of systems (such as a society or ecosystem) become irresolvable messes if they exceed a minimal level of complexity. Incapable systems with ambitious mess-management goals are accidents waiting to happen and are blind to the dysfunctions they introduce.

PRECARIOUS

Systems attempting to act as a bulwark against chaotic disorder in the environment will *actively conserve* vulnerability to occasional disasters and chronic misfortunes, tilting the risk towards those with less power.[9] For example, homes in low-lying areas seem safe due to elaborate drainage systems and water-containment barriers until the once-a-generation flood hits. Fighting all forest fires to protect nearby properties curtails the routine clearance of tinder from the forest floor, making subsequent fires bigger, hotter, and more destructive. Law enforcement agencies mandate weak software security to make it easier to spy on criminals, which is a vulnerability for everyone that criminals then exploit. When disaster strikes, hazard-prone systems will double-down on vulnerability conservation by piling on more controls (*control creep*) instead of rethinking arrangements due to the *sunk costs* and escalating risk narratives.

UNIFORM

Messes can be a good thing insofar as variation brings resilience. A variety of systems can better complement the complex environments in which they operate. Experimentation and diversity of thought act as a hedge against uncertainty; placing multiple bets is how uncertainty and chance are worked with constructively. However, a singular "best" way of doing things may take root across systems. Nascent evidence may even become baked into long-term regulations, contracts, voluntary standards, and formal routines, rigidities which then block evidence from being updated. Cookie-cutter approaches are cheaper, easier to maintain, and quicker to scale. However, as each "best of breed" system and "best practice" is replicated, any underlying flaws and contextual misfits get replicated just as quickly, leaving every copy vulnerable to exploit- ation. That is the danger posed by *monocultures* of systems.

PROPPED-UP

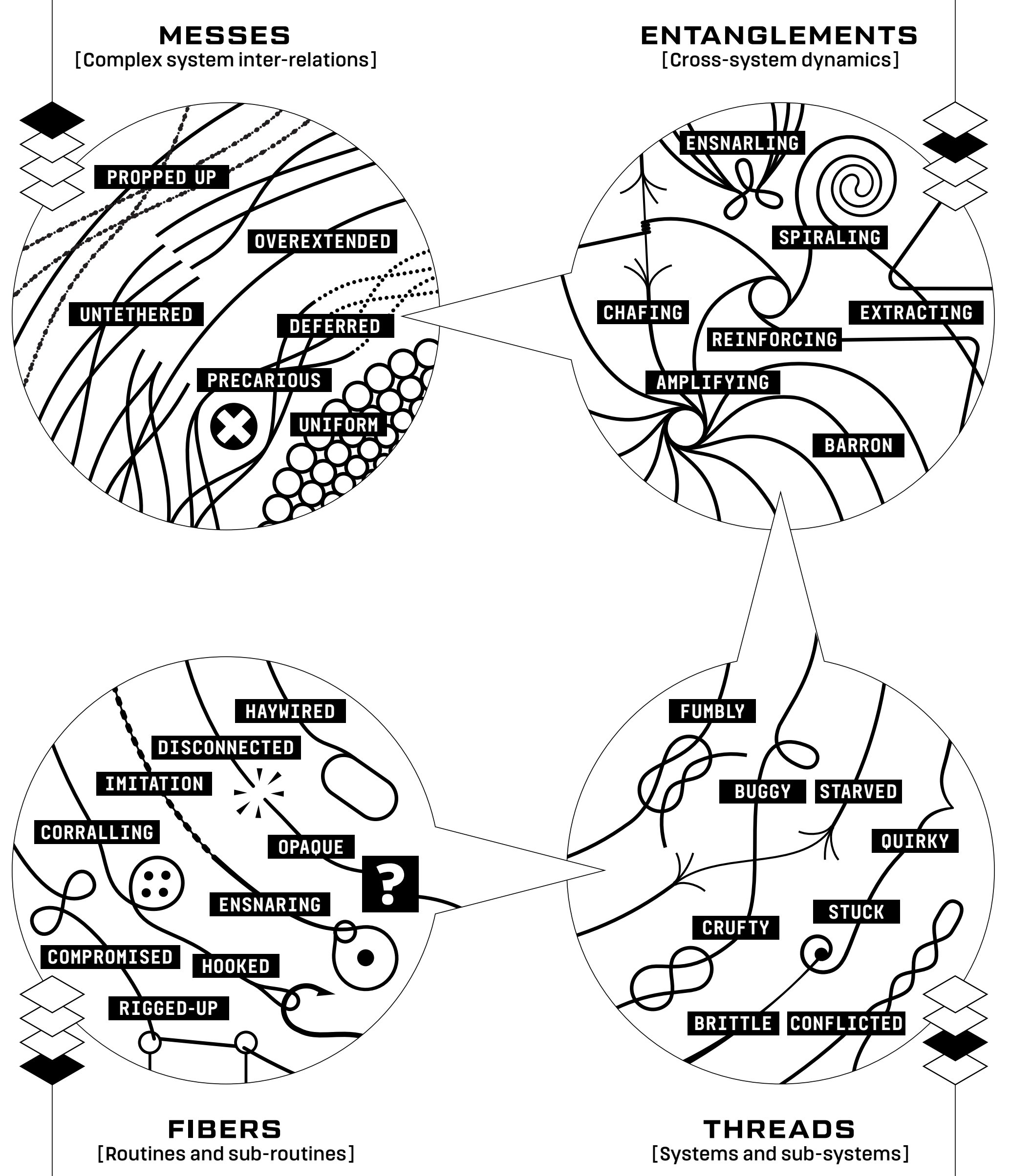
A broken thread in a tangle will remain propped-up by other threads. Likewise, as systems become interdependent, the incentive is to prevent dissolution of failing systems or usher in a replacement right away. Systems that are "too big to fail" or well-connected politically get the most support. That prevents a spell of bad fortunes from scuttling decades of building. Yet difficulties test the mettle of organizations and systems. Propping them up preserves weakness and encourages reckless and cynical risk-taking (*moral hazard*). Sketchy operations then linger, their weaknesses obscured by the extended support. These are the "zombies": neither alive nor dead, just limping along at minimal viability; thwarting innovative upstarts trying to take their place, hording resources and talent better redeployed elsewhere, and otherwise preventing rejuvenation. Too many weak threads make the tangle vulnerable to large-scale disruption.

DEFERRED

Humans tend to make short-sighted decisions that discount the future (*bounded willpower*) using imperfect information (*bounded rationality*) based on cultural framings of self-interest (*bounded interest*).[10] Systematization is often sold as more purely rational. Yet systems can pander to human biases and amplify their effects, especially innovations that time-shift the costs of present activities. For example, debt-financing and economies of scale can accelerate resource extraction and consumption in the present, while deferring the cost burden to some unspecified future moment. System feedback may provide signals of long-term unsustainability. Some systems may adjust to achieve equilibrium (*homeostasis*). Insofar as these signals and constraints call for short-term sacrifices, they may be surpressed or ignored. Worse, the mess of systems often obscures and blocks feedback about long-term consequences.

UNTETHERED

Systems can come to rely on the same abstractions, which become untethered from the complex, underlying reality and take on a life of their own. That is an *abstraction trap*. For example, money is a medium of exchange that bestows an *exchange value* on goods and services, which can become disconnected with real-world *use value*. Likewise, interface metaphors and conventions make a system easier to use by relating their operations to more familiar objects. Insofar as abstractions become integral to a system, the risk is that the contrivance is forgotten (*reified*) and considered a direct representation of reality (*misplaced concreteness*). Most systems have at least one layer of abstraction to make it intelligible to users and administrators. These abstractions can become a recognizable aspect of culture that get reused in other systems, often inappropriately. The build-up of abstractions across systems then mask underlying dangers. Worse, as problems emerge, the usual reaction is not to question the accuracy and validity of the abstraction layers. Instead, attention turns to spurious rationalizations and blame games.



ENTANGLEMENTS [Cross-system dynamics]

REINFORCING

A *reinforcing loop* occurs when system dynamics perpetuate a recurring pattern of activity. A negative reinforcement includes incentives, inducements, and persuasions that discourage particular actions, whereas positive reinforcement encourages actions. No system is perfect, so positive- and negative feedback signals are necessary to course correct by indicating what seems to work or not. Reinforcing loops tend to lock-in system activities by making divergence costly. Vulnerabilities emerge when a system trajectory is not sustainable and creates serious harms, yet the reinforcing dynamics promote continuation. For example, system reinforcements may be optimized to achieve a narrow goal while creating all sorts of collateral damage for other systems. Or a group of systems reinforce each other in a way that perpetuates long-term harms. By the time the dynamics change, it may be too late to avoid irreversable destruction.

SPIRALING

A *downward* (or *vicious-*) *spiral* occurs when system dynamics lead to a troublesome predicament, which gives rise to new dynamics that lead to more trouble, and so on until collapse happens. Each new set of dynamics makes it difficult to reverse course and recover from previous troubles. Often the only way to set things right is to create an *upward* (or *virtuous-*) *spiral* by changing the dynamics. That usually involves making short-term sacrifices and gaining support from elsewhere. Spiral dynamics tend to emerge because successive predicaments trigger troublesome dynamics within different systems. For example, a business may experience troubles in a supply chain that reduces cash flows, which trigger troubles from the financial system, which trigger operational cutbacks, which trigger reduced demand in retail markets, which further reduce cash flows. Each system exacerbates troubles independently.

AMPLIFYING

Risk amplifiers (or "black-hole risks") are dynamics that accelerate the pace of a downward spiral. Troubles can spread rapidly as they ripple across a growing set of cases (*compounding effects*), as with viral contagions. A positive reinforcement may incentivizes harm, with a growing number of actors joining in (*acceleration*). Risks may be correlated, so when one trouble occurs, others happen too, making the situation worse. Multiple amplifers create a *runaway effect* ("doom loop"). Fallbacks include: "firewalls" to contain spread; "curcuit breakers" to shut things down temporarily; "shock absorbers" to impose delays; and reserves to absorb losses. Triggering these measures may cause a panic by signaling danger. Risks become "turbo charged" when system complexity obscures the potential for runaway effects, the adequacy of risk-management measures is hard to judge, and incentives encourage downplaying of dangers.

EXTRACTING

In a "platform ecosystem," independent actors (*complementors*) built sub-systems atop a shared system. Think of popular social-media, e-commerce, and computer operating-system platforms. If the platform relies on a large network of users or massive economies of scale, a "winner-take-most" dynamic emerges as popularity breeds more popularity. Platform owners gain regulatory powers. Once a critical mass of dependecies is achieved, platform owners can become "takers instead of makers" by extracting *rents* while resting on their laurels. Siphoning resources away from innovators causes vitality to wane. Worse, entire domains that would otherwise be full of vibrant, free-wheeling diversity can become "platformized." Everything becomes reliant on a single system which, if it breaks, brings everyone else down too. There is less incentive to fix vulnerabilities on platforms with captive dependents and insulated platform owners.

BARRON

Within natural ecosystems, an *edge effect* is the abundance of diversity and interaction happening in the space where one habitat abuts another. An example is the space dividing a forest from scrub lands or an undersea shelf where the depths meet the shallows. The liminal zones of societal tangles have edge effects too. For example, innovation tends to emerge amid dense networks between business, finance, and academic research clusters. Buzzing urban neighbourhoods supporting vibrant cultures tend to have a mix of civic amenities, residents, and businesses. It is hard for beneficial spill-over effects (*positive externalities*) to happen between systems without the interactions that edges offer.[11] That is not true if there are too many edges, in which case certain types of actor may build up an advantage. Conversely, a barron network domain results if there are too few edges or intermingling is routinely blocked.

ENSNARLING

Systems can get in each others way. For example, regulatory and legal systems govern other systems. Worthwhile restrictions are beneficial. However, the build-up of regulations can become unwieldy. Contradictions may never get reconciled. Compliance burdens grow. Knowing all the requirements can be impractical. Streamlining by removing unnecessary "red-tape" has become a kind of "forever war." Regulators keep it all working by exercising sensible judgement and avoiding cases of regulatory unreasonableness. That is difficult to do even-handedly. Established players may even lobby to preserve red-tape to block potential rivals. Systems can stifle each other indirectly too. A domain may become crowded, with systems relying on a shared infrastructure (or super-system) with finite capacity. Designs have to account for congestion, especailly to avoid snarl-ups during peak periods and emergencies.

CHAFING

Systems can be designed for parasitic exploitation, the chafing in the tangle. Doctorow & Giblin explain how many instrumental systems have embedded subsystems that impinge on use and spy on users.[12] For example, the dominant brand of tractors are full of sensors to gather information about farm conditions to trade commodities. Not only does the data not help farms as independent systems but the trades go against their commercial interests. Each tractor also has a kill switch to prevent unauthorized repairs. Needless to say, vulnerability results when entire classes of system are not readily fixable and "curse" other systems that rely on them. Such parasitic systems can be more direct. For example, Michael Lewis tells how a consortium of Wall Street insiders built a sub-system to profit from price discrepancies between markets (*arbitrage*).[13] A fiber-optics tunnel was built to detect trades and "flash trade" a fraction of a second ahead, costing the original trader a little extra. That is "rigging" the system in a very literal sense.

THREADS [Systems and sub-systems]

CRUFTY

Cruftiness refers to departures from sound design principles that accumuate in systems as they age. If a system is not designed for ease-of-repair and on-going adaptation, it is "nursed along" with pragmatic fixes. The system departs from any sort of ideal configuration to become a knotted hodge-podge of parts. Crufty systems are full of *kludges*; that is, makeshift patches, hacky work-arounds, sub-par trade-offs, and unnecessary dependencies. Cleaning up that "technical debt" is an thankless, arduous chore, which is why it is despairingly called "yak shaving." Experienced fixers keep systems going with heroic saves and more kludges. Successful band-aids remove any urgency to do major renovations or confront deeper dysfunctions. Lack of major disruption obscures chronic, low-level problems. Cruft ultimately overwhelms the ability of fixers to cope. Disruptions also increase as experienced fixers retire or change jobs.

QUIRKY

Crimps in the thread represent unanticipated interactions between parts that otherwise work as intended. Even if benign, an unanticipated interaction may combine with others to create a compounding error. Charles Perrow points out that designers cannot foresee all interactions within highly complicated systems nor between different systems that interact. There are too many permutations and combinations to consider, even with perfect information. The more interconnected and elaborate a system gets, the more susceptible it is to such "normal accidents."[14] Moreover, the quirky behaviors of the system that result do not fall tidily within specialized disciplinary boundaries. Experts of various stripes have to put their heads together to diagnose problems. In so doing, they struggle to moderate disputes, allocate responsibility, and clear-up ambiguities because there are no established methods for that.

BUGGY

Bugs (flaws and faults) are inherent to system development. Most are quashed but a few inevitably remain. These kinks in the system are hard to track down because they are triggered by rare interactions and circumstances. The trickiest to diagnose are technical flaws combined with faulty assumptions. Some are harder to find when looked for (*observer influence*). As software takes over more and more system tasks, bugs become a bigger threat. Unlike with other engin- eering disciplines, workaday coders are slow to absorb advancements in praxis. To cut down on the "vulns," coders babysit one another (peer code-reviews, pair programing, group bug-hunts, and so forth). Anticipatory, socially-savvy *quality control* methods have success in highly constrained scenarios but work less well for systems operating in chaotic settings. Tellingly, the convention in software licensing exempts providers from any legal liability for bugs.

FUMBLY

Systems may be organized into discrete modules to avoid spaghetti-like tangles of interdependencies. Each module can be debugged, reworked, or swapped-out without having to fuss around much elsewhere. The modules then interface with each other through loose couplings, the metaphorical ties in the tangle. If taken too far, such arrangements can create fumbly systems. Highly modular systems decohere when losing sight of larger goals; full independence runs counter to what a system is. As systems decohere, unanticipated dysfunctions emerge from the fragmentation. New forms of kludge are used to cope, such as hardware adapters, translation layers, and triage routines. The connective ties are easy to maintain in theory. In practice, those ties often get neglected, causing botched or cumbersome hand-offs. If different modules are controlled by factions with different interests, the ties can be curtailed to impair cooperation.

STUCK

Systems can get snagged, or unable to adapt to the times. To recoup an investment, a system's life-cycle maybe extended too long. Some snags involve *lock-in* and *path dependence*. For example, customizations make a system difficult to upgrade or migrate away from. Proprietary technologies can create unhealthy dependence on external providers and add switching costs. A system may lack a diversified resource base, relying on only a few commodities, suppliers, or regions; too many eggs are placed in too few baskets, adding risk. Systems can also get caught in an *efficiency trap*: the system never settles if change is always being explored; the system struggles to change if fixated on exploiting existing advantages.[15] Each mode involves different capabilities. A system locked in exploitation mode will find it difficult to relearn how to change when the environment demands adaptation. Initial attempts will inevitably be clumsy.

STARVED

A frayed thread represents a system hanging on by a few fibers, barely performing its function because it is starved of resources, such as funding, staff, facilities, and whatnot. Less vital tasks fall by the wayside. Maintenance duties are neglected. Backlogs pile up and delays become unreasonable. Margins of safety erode. Over-all performance suffers. If deprived long enough, starved systems will operate on the edge of break-down. There are three forms. First, so-called austerity measures are attempts to be frugal in the short-term but, rather than trim "fat," end up curtailing the management of long-term risk. Moreover, political entrenchment can matter more than functional necessity in cost-cutting decisions. Second, opponents of a system can "starve the beast" if they hold sway over resource allocations. Third, "lean" systems premised on "just-in-time" resource allocation lack fall-backs needed to cope with unforeseen disruptions.

BRITTLE

Brittle systems are over-specified and exacting. There is little forgiveness for awkward variation, with minor anomalies causing major malfunctions. Think of brittle systems as strained threads, too taught to flex when necessary. Atypical cases (*confounds*) then cause the system to snap. Ideally, if a system fails, it fails gracefully (or "safely"): not all functionality is lost; recovery is immediate with fall-backs in place to minimize damage and burden placed on everyone affected. Even better, systems "self heal" by automatically diagnosing the problem and setting things right. However, brittle systems lack that sort of fault tolerance, allowing small errors to cause major breakdown. Accordingly, a high-precision system with many intricate parts tends to either have higher maintenance requirements or demand more control over operational conditions; they need a pit-crew of fixers or have to be heavily insulated from stressors.

CONFLICTED

Most systems contain trade-offs and tensions that have to be mitigated. Elegant designs manage conflict with ingenuity. Internal goal conflicts left unresolved can prevent a system from accomplishing its ultimate purpose. A system may internalize political conflict by turning it into administrative contradiction. For example, a governance system may subsidize tobacco farmers while running anti-smoking campaigns. It may rely on lottery revenues to combat gambling addiction. One activity may be a ploy to whitewash the other. More often, administrators do not want to favor one interest over another, so the system twists itself up to satisfy conflicting sides in ways that are self-defeating. Sometimes, the contradictions happen in obscure ways that only become apparent in particular circumstances. These *double binds* create "damned if you do, damned if you don't" options.[16] Often, the only move left is to muddle through while maintaining the fraught pretense of system consistency.

SPACE SCALES

	Scale	Description
HYPER SCALE	EXO	Outer
	SUPRA	Above, Over
	MACRO	Very Large
	MESO	Middle
	MICRO	Near Surroundings
	HUMAN	Experiential Proportions
HYPO SCALE	MINI	Miniscule
	MICRA	Tiny
	NANO	Very Tiny
	PICO	Elemental
	EXI	Extreme Smallness

TIME SCALES

	Scale	Description
HYPER SCALE	PETA	Geologic Time (Chrons to Eons)
	TERA	Millennia
	GIGA	Decades-Centuries
	MEGA	Months -Years
	MESO	Weeks -Months
	KILO	Hours-Days
	HECTO	Minutes-Hours
	DECA	Seconds-Minutes
	MOMENT	Experiential Proportions
HYPO SCALE	CENTI	Reflex response
	MILLI	Neuron firing
	MICRO	Latency of optical computer networks
	NANO	Laptop (GHz) micro-processor cycle
	PICO	Fastest micro-processor cycle
	FEMTO	Ultraviolet-light wave cycle
	ATTO	Finest timing control of lasers
	ZEPTO	Electron oscillation cycle

scale and scope

From the vantage point of ordinary experience, the tangle of systems can appear stable due to human *change blindness*. Most systems activity happens behind the scenes and far away, at sizes too small or too big to notice. We percieve change through a brief window of time with a shallow depth of focus. Our memories are selective and degrade quickly. We take emotional comfort from continuity. All told, we have little inclination and ability to keep track of all the moving parts. If we broaden our field of view and become better attuned to system rhythms, we discover that the tangle of systems is far from a *steady state*. It is continually writhing, with the occasional disjuncture. Even seemingly fixed and solid *structures* are always degrading under the forces of entropy without regular maintenance. Thus, everything is in ***motion*** (*process ontology*), just not at time and spatial scales we are used to thinking in.

Modern society is speeding everything up with ever faster technologies and briefer technological life-cycles. Human-induced elective pressures in natural ecosystems are becoming ever more severe. Things do not last like they used to. At the same time, systems ***sprawl*** further from our local vantage point. Not only do many system interdependencies now span the globe, they reach ever lower depths and ever loftier heights. For example, Ancient Roman mines could go 200 meters deep, whereas today's mines can go down as far as four kilometers. Neolithic-era wells could be a dozen meters deep. Today, a shale gas well may reach a depth of seven kilometers. Eight millennia ago, a Mongolian falconer may have birds patrolling a few hundred meters in the sky. A modern satellite can be hundreds of thousands of kilometers above the ground.

Life is being ever more systematized: human-made systems condition and constrain a wider variety of social relations; daily tasks are more reliant on systems; systems intrude on our personal sphere more readily, including inside our bodies; less of nature is untouched by human intervention. Thus, the ***scope*** of systems is also expanding.

An overview of ***generic levels of scale*** are shown in the tables above. These provide an inkling of the scales that might be relevant when analyzing systems. This breakdown may not suit every analytical task, for not all levels listed will be relevant, plus many in-between levels might be. These tables invite us to consider methodically how systems (both separately and entangled) operate at different scales. Traditional approaches to analysis fixate on a single scale (as with micro- and macro economics) and limited territories (such as regions). Understanding system vulnerabilities holistically requires a wider ranging attention that accounts for various scales and an extended scope of system activities.

The *Pattern Atlas'* treatment of scale also incorporates different ***levels of aggregation***. The mess is made up of various system entanglements, which in turn are made up of systems and subsystems, which are made up of routines and sub-routines. At low levels of aggregation, routines operate according to recurring patterns. As we move up the levels, the cyclical recurrence become harder to predict due to the complex, non-linear dynamics. For the sake of highlighting particular types of vulner- ability, the *Atlas* is divided into four levels of scale. These are illustrated using four planes, with the highlighted plane indicating the current level. What these relative levels represent in absolute terms depends on the specific tangles of systems being analyzed.

The growing scale and scope of entanglement are increases in vulnerability in their own right. It is not always clear why all human activities have to be systemized or turned into systems with high-levels of entanglement. Managing vulnerability often involves ***setting new boundaries*** to partition the sprawl.

TIMING

PERIODIC — Different systems or sub-systems change in regular intervals of different duration

QUASI-PERIODIC — System change happens at intervals that are not regular (or not entirely so)

NON-PERIODIC — System change does not happen over periods that can be identified as intervals

CHANGE

SECULAR — Changes not due to short-term fluctuations or on-going cycles or rhythms.

CYCLICAL — Change due to recurring rhythms within a system

pace layers

Thanks to systems, the tem… ity of life has become eve… fixed durations, rigid se… times for certain activi… cycles.[17] There is a rh… in the tangle, makin… dictable but not e… ruption happens… systems evenly… change at diffe… calls these pac… above).[18] Even… befalls all syst… discontinuity… is because sy… responsivene… dependencies… unevenness i… failures difficu… those with flex… capabilities wh… system lags.

FULL ANNOTATED POSTER

SYSTEMVIZ

Project. The *Pattern Atlas* is a product of the SystemViz Project, an ongoing research study devoted to better explaining systems through the use of visuals. Further elaboration can be found in the forthcoming book by Peter Stoyko, *How Small Players Change Big Systems* (2024). Additional products can be downloaded for free from the project web site.

Author. Peter Stoyko is an interdisciplinary social scientist and information designer with an interest in service design, systems, culture, foresight and governance. For more, see: www.stoyko.net

Citation. Please site this document as: Peter Stoyko, *Pattern Atlas of System Vulnerabilities 1.0.0* — Poster (Ottawa: The SystemViz Project, 2022).

Related. The CultureViz Project is a companion project that visually explores culture as a system. For more, see: www.cultureviz.com

www.systemviz.com

faults + confounds

A fault is a malfunction or error that impedes system activities or causes general failure. A confound is an unexpected, unwelcome factor persisting in a system or an unplanned-for case causing difficulties. These disruptions come in four general patterns, with each pattern having different implications for the way systems prepare and react.

REGULAR DISRUPTIONS — Regular disruptions occur with predictable frequency and can be coped with by adding precautions and recovery measures. Dedicated systems or sub-systems may be installed to routinize the handling of these disruptions. For example, an urban road network is an amalgam of coordinated systems (traffic control, law enforcement, road maintenance, and so forth) designed for semi-autonomous systems (vehicles) to operate on. Occasional disruptions happen, such as vehicle crashes and snow storms. Dedicated systems prevent serious disruption (such as towing-, ambulance, and snowplow services). Operators of downstream systems may even have a vested interest in preventing upstream problems from getting solved once and for all.

IRREGULAR DISRUPTIONS — Irregular disruptions are rare events that cause a system from functioning. Some are catastrophic insofar as they cause significant damage to more than one system at a time, or cause a cascade of disruptions across interconnected systems. Emergency preparedness and disaster management are disciplines devoted to minimizing these disruptions. That can include dedicated systems. Overly precise preparations usually get caught flat-footed and ill-equipped because irregular disruptions are so quirky and rare. Precautions are over-optimized for preventing the previous disruption. Vigilance also wanes over time. Haphazard, ill-informed disaster responses that try to micro-manage elaborate entanglements of systems often wind up causing their own cascading failures.

NEGATIVE EXTERNALITIES

CHRONIC DIFFICULTIES

problem maintenance

examples

HAYWIRED

UNIFORM

AMPLIFYING

CRUFTY


(n)ever changing world paradox

routine specification


thank you
systemviz.com