[DRAFT] Human Cooperation System (HCS)

A diagnostic lens for understanding how human cooperation enables effective work.

Viktor Jevdokimov, Vilnius, Lithuania

Table of contents

1. INTRODUCTION & QUICK START

1.1 Human Cooperation System (HCS) – From Context to Practice

> A diagnostic lens for understanding how human cooperation enables effective work.

⚠ **DRAFT VERSION**

This document is a **working draft** of the *Human Cooperation System (HCS)*.
Content, terminology, and structure are **subject to revision**.

1.1.1 Introduction

The **Human Cooperation System (HCS)** explores how people manage to work together — not as isolated individuals, but as interdependent actors connected by shared goals, constraints, and meaning.
It focuses on the *invisible architecture* that makes cooperation possible: the conditions, needs, and functions that hold any collaborative system together.

When collaboration fails, it is rarely due to lack of skill or effort — it usually breaks at the **system level**: misaligned expectations, unmet human needs, or missing functions that maintain trust and coherence.
HCS helps reveal these root causes before they manifest as friction, disengagement, or project failure.

1.1.2 Who It's For

HCS is designed for people who lead, enable, or study cooperation:

- **Project and Product Leads** seeking to understand why delivery struggles despite competent teams.
- **Engineering and Design Managers** working to improve cross-functional collaboration and autonomy.
- **Consultants and Coaches** diagnosing organizational misalignment or low trust between groups.
- **Researchers and Framework Builders** exploring universal patterns behind effective human cooperation.

If you've arrived here from **3SF (3-in-3 SDLC Framework)** or similar models, HCS represents the *layer beneath practice* — the foundational system that governs why engagement, delivery, and value either flow or fail.

1.1.3 Purpose

HCS serves as a **diagnostic system** for understanding and improving cooperation across all forms of human work — from teams and organizations to partnerships and ecosystems. It does not prescribe methods; instead, it offers a *lens* to evaluate which functions must be present and coherent for cooperation to succeed.

- It describes the **preconditions** that enable cooperation.
- It identifies **human needs** that sustain motivation and trust.
- It defines **stable functions** that turn shared intent into collective outcomes.

Together, these dimensions form the **Systemic Basis of Work** — what must exist before practices, tools, or frameworks can be effective.

1.1.4 Read This First

1. Start with the **Matrix** to see the five-by-five landscape of *conditions × needs*.
2. Use the **Pyramid** to understand levels from *preconditions → practices*.
3. Apply **The Level Rule** before prescribing practices.
4. Run the **Diagnostic Workflow** on real observations.

1.1.5 Positioning

Think of **HCS** as the *operating system* beneath all frameworks and practices.
Tools like Agile, Lean, or management methodologies can only succeed when the system-level conditions of cooperation are stable.
By understanding and diagnosing those foundations, leaders and teams can work with greater clarity, trust, and purpose.

2. THEORY

2.1 Vision, Principles, and Beliefs

The **Human Cooperation System (HCS)** defines a universal model for understanding how people and organizations **sustain cooperation** under changing conditions.
It establishes the underlying *physics* of collaboration — the invisible structures that determine whether human systems remain stable, adaptive, and meaningful over time.

2.1.1 Vision

To enable individuals, teams, and organizations to **work together intentionally** — with clarity, trust, and shared purpose — regardless of role, method, or domain.
HCS envisions a world where cooperation is treated as a **systemic discipline**, not a matter of personal style or organizational culture.

> **Vision Statement:**
> *A cooperative world where systems of work evolve with people — not against them.*

2.1.2 Principles

HCS is governed by a small set of **systemic principles** that translate into all later models and diagnostics.
They describe *how cooperation remains stable* and *why interventions must follow structure, not trend.*

Principle	Description
1. Function-First	Every method or practice must serve a clear cooperative *function*. Tools and rituals are effective only when matched to their purpose.
2. The Level Rule	A dysfunction cannot be corrected at a higher level than where it originates. Stability builds from the bottom up — from human conditions to frameworks.
3. Evidence Over Opinion	Cooperation is observable. Use behavior and outcomes as evidence before assuming intent or motivation.
4. Feedback Closes the Loop	Stability depends on timely, accurate feedback — at personal, team, and systemic levels. Unclosed loops lead to drift and mistrust.
5. Autonomy with Accountability	Freedom without shared responsibility fragments; control without autonomy suffocates. Balance creates flow.
6. Clarity Before Speed	Communication fidelity and shared understanding are prerequisites for efficiency. Misalignment is the costliest waste.
7. Trust Before Control	Trust enables coordination with minimal overhead. Control mechanisms are substitutes for trust, not its replacement.
8. Reflection Enables Evolution	Continuous introspection — individually and systemically — transforms experience into adaptation.

2.1.3 Beliefs

HCS is built on several **core beliefs** about human work systems:

1. **Cooperation precedes performance.** Teams fail for relational reasons long before technical ones.
2. **All systems drift without feedback.** Stability is not static — it's actively maintained through learning.
3. **Trust is measurable through behavior.** Reliable follow-through and transparency are its most objective indicators.
4. **Shared purpose is a living construct.** It evolves as understanding deepens and context changes.
5. **Frameworks succeed only atop stable cooperative functions.** No process can compensate for missing trust, clarity, or shared intent.
6. **Every organization is a learning organism.** The health of its feedback loops determines its adaptability.

2.1.4 Scope and Non-Goals

HCS does not prescribe **how** to manage projects, design organizations, or measure performance.
Instead, it defines **what makes cooperation functionally possible** — a diagnostic foundation for any framework, method, or governance model.

HCS is **not**:

- A replacement for Agile, Lean, or management systems.
- A behavioral theory of personality or motivation.
- A fixed sequence of practices or rituals.

It is a **system lens** — a way to see cooperation as an interconnected structure that either supports or constrains all other methods.

2.1.5 Bridge to Practice

The **3-in-3 SDLC Framework (3SF)** applies HCS principles to the domain of software delivery and client-vendor collaboration.
Where HCS defines *how cooperation works in theory*, 3SF shows *how to build it into contracts, roles, and delivery systems*.

Together, they form a continuum:

> **HCS** — the theoretical foundation of cooperative stability. **3SF** — the applied framework that operationalizes it.

This page serves as the philosophical foundation for all other HCS sections — the Pyramid, Matrix, and Diagnostic Workflow each elaborate these principles in context.

2.2 The Human Cooperation System Matrix

⚠ **DRAFT VERSION**

This document is a **working draft** of the *Human Cooperation System (HCS)*.
Content, terminology, and structure are **subject to revision**.

Cooperation exists because of interdependence — when one person alone can't or doesn't want to do everything, *work becomes a shared system of value exchange*.

If one person could do everything alone, work would be **pure autonomy**.

But the moment **two or more people are involved**, *working together becomes a system* — and coordination, communication, trust, and shared purpose are no longer optional; they are essential.

2.2.1 Why the Matrix Exists

The **HCS Matrix** captures the *minimum viable structure of cooperation*.
It defines the smallest set of interacting forces required for human collaboration to emerge and remain sustainable.

It is not a framework or process map — it's a **diagnostic lens**.
It helps you identify *which essential conditions and needs are present, missing, or misaligned* in any collaborative system — from teams and partnerships to entire organizations.

When mapped across real situations, the Matrix shows *why cooperation either flourishes or fractures*, long before outcomes or metrics reveal the symptoms.

2.2.2 Dimensions of the Matrix

> **Naming note:** The term **Change/Uncertainty tolerance** is used consistently across HCS to emphasize both the presence of change and the system's capacity to act under it.

- **Horizontal axis — Core Human Needs** for Cooperative Work
  Represents what individuals and groups require to function together effectively.
- **Vertical axis — Core Work Conditions**
  Represents the external and structural contexts that must exist for cooperation to take place.

Each **cell** in the matrix expresses a **minimal system function** that must be present for collaboration to emerge.
Examples illustrate how these dimensions intersect:

- *Common Purpose × Shared Understanding → Alignment on why* — shared intent behind joint work.
- *Interdependence × Feedback Loops → Outcome reflection* — awareness of mutual impact and adaptation.
- *Change/Uncertainty × Autonomy & Agency → Adaptability* — the capacity to act locally under changing conditions.

The Matrix defines the **existential layer** of the Human Cooperation System — the foundation that makes collaboration possible, preceding any specific framework, tool, or methodology.
Above this layer, operational frameworks such as Agile, DevOps, or OKRs can be seen as implementations of the underlying functions identified here.
The **Principle–Practice Matrix** later extends this logic to the operational layer, describing how cooperation is maintained and optimized once these foundational functions are active.

2.2.3 The Human Cooperation System Matrix – Descriptions

This section defines and explains each element of the Matrix that outlines the foundational **conditions** and **needs** of cooperative work.

Core Work Conditions (Vertical Axis)

1. **Common Purpose** – Shared meaning or goal that unites people working together. It gives direction and motivation, enabling alignment of actions across individuals or groups.
2. **Interdependence** – A condition where the outcome of one's work depends on others and vice versa. It requires awareness of mutual impact and encourages collaboration.
3. **Communication** – The exchange of information, meaning, and intent between individuals or groups. It enables coordination, understanding, and feedback.
4. **Trust** – A belief in the reliability, competence, and good intent of others. Trust reduces the cost of coordination and enables delegation, openness, and cooperation.
5. **Change/Uncertainty** – The inevitable presence of external or internal shifts that affect work. Handling change requires resilience, learning, and flexibility in both individuals and systems.

Core Human Needs for Cooperative Work (Horizontal Axis)

1. **Shared Understanding** – A mutual grasp of language, goals, constraints, and contexts among collaborators. It ensures that all parties interpret information similarly.
2. **Mutual Commitment** – A shared willingness to contribute to common goals. It implies dedication, accountability, and follow-through from all parties involved.
3. **Feedback Loops** – Mechanisms to observe results, evaluate progress, and make corrections. Enables learning and continuous improvement in a shared system.
4. **Distribution of Roles** – Clear delineation of who does what, ensuring responsibilities are known and efforts are coordinated.
5. **Autonomy & Agency** – The ability and permission to act with intention. Agency empowers individuals to make decisions, contribute meaningfully, and take ownership.

2.2.4 Matrix Cell Descriptions (5x5 = 25)

1. **Common Purpose × Shared Understanding = Alignment on why** – People must align on why they're working together and what success looks like. Without this, efforts may diverge.
2. **Common Purpose × Mutual Commitment = Willingness to act** – A goal alone is not enough; people must genuinely commit to working toward it together.
3. **Common Purpose × Feedback Loops = Learning intent** – Continuous evaluation of whether actions still serve the agreed purpose. Prevents mission drift.
4. **Common Purpose × Distribution of Roles = Contribution clarity** – Each role must support the purpose, ensuring no effort is wasted or misaligned.
5. **Common Purpose × Autonomy & Agency = Room for initiative** – Individuals must be able to pursue the shared purpose with self-direction and initiative.
6. **Interdependence × Shared Understanding = Task relationships** – Team members need clarity about how their tasks relate and depend on one another.
7. **Interdependence × Mutual Commitment = Responsibility** – Each person must commit to their role knowing others rely on them.
8. **Interdependence × Feedback Loops = Outcome reflection** – Real-time signals help adjust coordination and prevent cascading failures.
9. **Interdependence × Distribution of Roles = Coordination** – Role clarity enables smooth handoffs and cooperation.
10. **Interdependence × Autonomy & Agency = Local decision-making** – Teams must balance dependencies with local autonomy to avoid bottlenecks.
11. **Communication × Shared Understanding = Language/Terms** – Shared vocabulary and mental models are essential for messages to land as intended.
12. **Communication × Mutual Commitment = Social contract** – Open, honest communication supports commitment and builds accountability.
13. **Communication × Feedback Loops = Signal/response** – Effective communication delivers feedback in a usable form and closes the loop.
14. **Communication × Distribution of Roles = Clarity in interaction** – Communication supports clarity on who does what and when.
15. **Communication × Autonomy & Agency = Permission to act** – People need communication channels to raise concerns, ask for support, or declare decisions.

16. **Trust × Shared Understanding = Meaning consistency** – Trust grows when people interpret intentions and actions reliably.
17. **Trust × Mutual Commitment = Reliability** – Trust enables people to believe that others will deliver on their promises.
18. **Trust × Feedback Loops = Safety in feedback** – Safe environments allow for honest feedback without fear of blame.
19. **Trust × Distribution of Roles = Delegation** – Clear roles reduce friction and show respect for expertise, reinforcing trust.
20. **Trust × Autonomy & Agency = Empowerment** – Trust enables freedom to act without micromanagement.
21. **Change × Shared Understanding = Scenario awareness** – Adapting to change requires everyone to see and interpret the new reality similarly.
22. **Change × Mutual Commitment = Resilience** – Teams that weather change together must renew their commitment in the face of shifting conditions.
23. **Change × Feedback Loops = Learning from change** – Change demands fast, reliable feedback to inform next actions.
24. **Change × Distribution of Roles = Flexibility** – Roles may need to shift dynamically; clarity and adaptability are key.
25. **Change × Autonomy & Agency = Adaptability** – People must be empowered to respond quickly and appropriately to emerging challenges.

2.2.5 The Human Cooperation System Matrix – Table View

Work Needs / Work Conditions	Shared Understanding	Mutual Commitment	Feedback Loops	Distribution of Roles	Autonomy & Agency
Common Purpose	Alignment on why	Willingness to act	Learning intent	Contribution clarity	Room for initiative
Interdependence	Task relationships	Responsibility	Outcome reflection	Coordination	Local decision-making
Communication	Language / Terms	Social contract	Signal/response	Clarity in interaction	Permission to act
Trust	Meaning consistency	Reliability	Safety in feedback	Delegation	Empowerment
Change/ Uncertainty tolerance	Scenario awareness	Resilience	Learning from change	Flexibility	Adaptability

2.3 The Human Cooperation System Pyramid

⚠ **DRAFT VERSION**

This document is a **working draft** of the *Human Cooperation System (HCS)*.
Content, terminology, and structure are **subject to revision**.

2.3.1 Why the Pyramid Exists

The **HCS Pyramid** visualizes how cooperation evolves through **five systemic levels** — from the most fundamental conditions of existence to reflective innovation at the top.
It helps identify *where cooperation breaks down* and *which level to address first* when diagnosing teams, organizations, or partnerships.

The Pyramid complements the HCS Matrix:

- The **Matrix** defines *what must coexist* (conditions × needs).
- The **Pyramid** shows *how these elements build on one another* — a hierarchy of dependency where higher levels depend on the stability of lower ones.

In short:

> The Matrix describes **the anatomy of cooperation**,
> while the Pyramid shows **its order of development**.

2.3.2 How to Use the Pyramid

Use the Pyramid as a **diagnostic guide**:

1. Identify which level the issue originates from — conditions, needs, functions, practices, or reflection.
2. Avoid “patching” higher levels (e.g., practices or frameworks) when foundational levels are weak.
3. Move upward only when stability and coherence are achieved below.

Each level in the Pyramid corresponds to a layer of the Human Cooperation System — from the first moment people start working together to when they consciously evolve how they work.

2.3.3 Level 1 – Preconditions for Cooperation (Foundation)

Without these, there’s no “working together” at all.
They’re the *existential conditions* of cooperation — the soil from which all other levels grow.

Examples:

- **Existence of a Common Purpose** – shared “why” to engage at all.
- **Interdependence** – mutual reliance or need for others’ contributions.
- **Basic Communication ability** – shared medium: language, symbols, signals.
- **Basic Trust** – belief that others will not harm and will reciprocate.
- **Capacity to Act under Change** – some tolerance for uncertainty.

Maslow analogy: Physiological & safety needs — the “oxygen” of cooperation.

2.3.4 Level 2 – Core Human Needs for Cooperative Work

These are the *human-level enablers* that make foundational conditions workable day-to-day.
This corresponds to the **horizontal axis** in the Matrix.

Examples:

- **Shared Understanding** — common mental model of goals, constraints, and context.
- **Mutual Commitment** — agreement to invest effort in the shared purpose.
- **Feedback Loops** — ability to observe, learn, and adjust.
- **Distribution of Roles** — clarity on who does what.
- **Autonomy & Agency** — freedom and authority to act.

Maslow analogy: Belonging and esteem needs — creating security in relationships and a sense of contribution.

2.3.5 Level 3 – Cooperative System Functions

Stable **functions** that translate human needs into **coordinated, repeatable work**.
These live in the 25 Matrix cells and define what must happen for collaboration to succeed.

Examples:

- **Problem Discovery** (Clarity of Problem Space × Shared Understanding)
- **Planning & Prioritization** (Interdependence × Distribution of Roles)
- **Monitoring & Feedback** (Communication × Feedback Loops)
- **Enablement & Empowerment** (Trust × Autonomy & Agency)
- **Adaptation & Learning** (Change × Feedback Loops)

These are the “work muscles” — when one is weak, the whole system compensates or breaks. *Maslow analogy*: Cognitive needs — organizing and directing effort.

2.3.6 Level 4 – Practices & Frameworks

Concrete, evolving ways of performing the functions.
Different implementations can fulfill the same function, depending on the organization’s context.

Examples:

- **Retrospectives** fulfill *Monitoring & Feedback*.
- **RACI matrix** fulfills *Distribution of Roles*.
- **Scrum sprint planning** fulfills *Planning & Prioritization*.
- **JTBD interviews** fulfill *Problem Discovery*.
- **Kanban** fulfills *Flow & Focus + Monitoring & Feedback*.

This is where change happens fastest — tools and trends evolve, but underlying functions stay stable.
Maslow analogy: Self-actualization tools — many possible routes to fulfill the same higher need.

2.3.7 Level 5 – Meta-Practices & Innovation (Apex)

The ability to **reflect on and redesign the system itself**.
Here, teams stop merely applying practices and begin **adapting or inventing** them to fit their unique context.

Includes:

- Combining practices into custom playbooks.
- Inventing new practices when existing ones don’t fit.
- Matching practices consciously to functions (avoiding “cargo cult” behavior).
- Teaching and sharing system-level understanding.

Maslow analogy: Self-transcendence — going beyond individual optimization to improve the system for others.

2.3.8 Pyramid View – Start & Top

Always climb levels in order

Don’t patch Level 4 **practices** when Level 1–2 **preconditions or human needs** are weak.
Fix lower levels first — then revisit practices.

- **It starts** at Level 1 the moment two or more people have a reason to interact for mutual value.
- **It tops** at Level 5 when teams become self-evolving systems — capable of diagnosing, designing, and improving their own way of working without external imposition.

Level	Name	Examples	Description
5 (Apex)	Meta-Practices & Innovation	• Designing custom playbooks • Inventing new practices • Matching practices to functions deliberately • Coaching others in system thinking	Teams consciously reflect on, adapt, and redesign their way of working. Practices are tailored, combined, or invented to better serve needs. Knowledge is shared to elevate the whole system.
4	Practices & Frameworks	• Retrospectives (*Monitoring & Feedback*) • RACI matrix (*Distribution of Roles*) • Scrum sprint planning (*Planning & Prioritization*) • JTBD interviews (*Problem Discovery*) • Kanban (*Flow & Focus + Monitoring & Feedback*)	Specific, evolving methods that fulfill stable functions. Practices change over time, but the function they serve remains stable.
3	Cooperative System Functions	• Problem Discovery (*Clarity of Problem Space × Shared Understanding*) • Planning & Prioritization (*Interdependence × Distribution of Roles*) • Monitoring & Feedback (*Communication × Feedback Loops*) • Enablement & Empowerment (*Trust × Autonomy & Agency*) • Adaptation & Learning (*Change × Feedback Loops*)	The stable "muscles" of cooperation — what must happen for work to succeed. Represented by the 25 cells in the Human Cooperation System Matrix.
2	Core Human Needs for Cooperative Work	• Shared Understanding • Mutual Commitment • Feedback Loops • Distribution of Roles • Autonomy & Agency	The human-level enablers that make foundational conditions operational. Corresponds to the Matrix's **horizontal axis**.
1 (Foundation)	Preconditions for Cooperation	• Common Purpose • Interdependence • Communication • Trust • Change/Uncertainty tolerance	The existential conditions for cooperation to exist at all. Corresponds to the Matrix's **vertical axis**. Without these, there is no "working together."

2.3.9 Diagnostic Use

When diagnosing cooperation:

- Weakness in **Level 1 or 2** indicates instability in the foundations — people lack shared context or trust.
- Weakness in **Level 3** suggests missing or broken system functions.
- Weakness in **Level 4** means inappropriate or outdated practices are being applied.
- Weakness in **Level 5** shows that the system can’t yet learn or evolve on its own.

The Pyramid helps shift conversations from *“What framework should we use?”* to *“Which layer of cooperation needs attention first?”*

2.4 The Level Rule

⚠ **DRAFT VERSION**

This document is a **working draft** of the *Human Cooperation System (HCS)*.
Content, terminology, and structure are **subject to revision**.

2.4.1 Why the Level Rule Exists

The **Level Rule** is the single most practical principle of the Human Cooperation System.
It prevents teams and leaders from treating symptoms at the *wrong level* — such as changing practices when the underlying conditions or needs are unstable.

In cooperation systems, **every level depends on the stability of the one below it**.
Just as a house can't rest on a weak foundation, teams can't sustain new practices if trust, clarity, or purpose are missing.
The Level Rule helps you locate *where to intervene first* when diagnosing or improving collaboration.

> The deeper the dysfunction, the lower the level you must repair.

2.4.2 How to Apply the Level Rule

When you observe friction, confusion, or performance drops, follow this diagnostic sequence before deciding what to change:

1. **Start from the bottom.**
   Check that foundational *conditions for cooperation* (Level 1) are present — shared purpose, trust, interdependence, and communication.
2. **Assess human needs.**
   Verify that *day-to-day enablers* (Level 2) are satisfied — shared understanding, mutual commitment, feedback loops, role clarity, and agency.
3. **Evaluate system functions.**
   Review whether *core cooperation functions* (Level 3) are active — such as planning, feedback, enablement, and adaptation.
4. **Then adjust practices.**
   Only once the system is stable below Level 3 should you change *practices or frameworks* (Level 4).
5. **Reflect and evolve.**
   At maturity (Level 5), teams begin to *self-diagnose and adapt* — applying the Level Rule themselves.

2.4.3 The Five Levels at a Glance

1. **Level 1 – Preconditions for Cooperation**
   Common Purpose, Interdependence, Communication, Trust, Change/Uncertainty tolerance Must exist for cooperation to even begin. Without these, no system of collaboration can sustain.
2. **Level 2 – Core Human Needs**
   Shared Understanding, Mutual Commitment, Feedback Loops, Distribution of Roles, Autonomy & Agency
   These needs must be met for day-to-day work to flow without friction.
3. **Level 3 – Cooperative Functions**
   The *25 functions from the Matrix* must be minimally expressed so the system can coordinate, learn, and adapt.
4. **Level 4 – Practices & Frameworks**
   Only after stabilizing Levels 1–3 can practices be tuned, replaced, or scaled without "fighting gravity."
5. **Level 5 – Meta-Practices & Innovation**
   Once the system is stable and self-aware, it can safely evolve its own way of working.

2.4.4 Diagnostic Heuristics

Use these quick rules when assessing where to act:

- If a problem **repeats across multiple practices**, it’s likely **not** a Level 4 issue — check Levels 1–3.
- If alignment **collapses under stress**, the root cause is almost always at **Level 1 or 2**.
- If coordination **stalls despite good relationships**, inspect **Level 3** — one or more core functions are missing or unclear.
- If **people lose trust in frameworks**, it may signal a mismatch between the *function needed* and the *practice chosen* (Level 4 problem built on weak lower layers).

2.4.5 Common Anti-Patterns

- **Cargo culting** – copying visible practices (Level 4) to compensate for missing foundations (Level 1–2).
- **Premature scaling** – optimizing or standardizing before core functions are stable.
- **Overdiagnosis** – jumping to structural fixes instead of addressing unmet human needs.
- **Tool worship** – assuming the adoption of a tool solves cooperation gaps.

2.4.6 Practical Example

When a team repeatedly fails to deliver despite switching from Scrum to Kanban:

- Level 4 (Practices) changes are visible but ineffective.
- Diagnosis shows missing **Feedback Loops** and unclear **Distribution of Roles** → both are *Level 2 needs*.
- Root cause lies lower — no amount of framework tuning will fix it until **shared understanding and feedback culture** are restored.

> The Level Rule turns “what framework should we use?” into “what layer of cooperation is failing right now?”

2.4.7 Summary

The **Level Rule** ensures interventions follow the natural order of cooperation:

If you want to fix...	First verify...	Otherwise you risk...
Practices or tools	Preconditions and needs	Surface fixes without stability
Collaboration quality	Trust, commitment, and understanding	Repeating breakdowns
Delivery flow	Functional coordination	Hidden friction and rework
Team autonomy	Clear roles and feedback	Chaos disguised as freedom

Essence of the Rule

> Never try to optimize the system at a higher level than it can currently sustain.

Mastering this principle makes every improvement effort more efficient, humane, and lasting.

2.5 Diagnostic Workflow (Observation → Matrix → Level → Function → Practice)

⚠ **DRAFT VERSION**

This document is a **working draft** of the *Human Cooperation System (HCS)*.
Content, terminology, and structure are **subject to revision**.

2.5.1 Why the Diagnostic Workflow Exists

The **HCS Diagnostic Workflow** turns theory into action.
It's a repeatable process for tracing **observable issues** in cooperation back to their **systemic root causes** — before deciding how to intervene.

Most improvement efforts fail because they start with *practice changes* (Level 4) rather than diagnosing *which level of cooperation is weak*.
This workflow keeps analysis grounded and ensures each corrective action strengthens the right layer of the system.

> The goal is not to fix symptoms — it's to restore coherence between **conditions, needs, and functions**.

2.5.2 How the Workflow Works

Use this short, structured loop on any issue or pattern you observe.

1. Observation — Capture evidence

Write down what you *see, hear, or measure* — without interpretation or blame.
Focus on **observable behavior**, not assumptions.

Example:

> "Critical dependencies were discovered late; downstream team blocked twice this sprint."

This ensures that diagnosis starts from data, not opinion.

2. Matrix Mapping — Locate where it lives

Map the observation to the **HCS Matrix**:
Which **Condition** (vertical axis) and which **Human Need** (horizontal axis) are involved?

Example:

> **Interdependence × Feedback Loops → Outcome Reflection**
> Late discovery means coordination and feedback signals are weak.

This step anchors the issue to a specific **function** — preventing vague problem statements like "communication issue" or "poor planning."

3. Level Check — Find how deep the root is

Determine the **lowest affected level** in the HCS Pyramid:

If the problem is about…	It likely belongs to…
Missing purpose, trust, or communication basics	**Level 1 – Preconditions**
Misalignment, unclear roles, weak feedback, low agency	**Level 2 – Human Needs**
Broken coordination or learning cycles	**Level 3 – Functions**
Ineffective or misapplied methods	**Level 4 – Practices**
Lack of reflection or self-improvement	**Level 5 – Meta-Practices**

Example:

> Repeated late dependency discovery → weak **Level 3 Monitoring & Feedback**
> Possibly thin **Level 2 Feedback Loops** underneath.

This step is where the **Level Rule** applies — fix lower levels first.

4. Function to Practice — Select what to strengthen

Once you've located the function, choose or design a **practice** that fulfills it.

Use the 25 Matrix functions as your reference for what "healthy cooperation" looks like.
The practice can be a known method or a custom one suited to your context.

Example:

> Function: *Communication × Feedback Loops → Signal & Response*
> Practice ideas:
> - Add an explicit dependency scan before each sprint. - Introduce a daily "risk surfacing" moment. - Create a visual "Andon" signal for cross-team blockers.

5. Trial & Learn — Validate and iterate

Run the chosen practice for 1–2 cycles and track a **specific observable signal** (not vague satisfaction scores).
If it doesn't produce the expected outcome, recheck **lower levels** before trying another practice.

Example:

> Watch: *Reduction in repeated dependency delays within two sprints.*
> If not improved → revisit Levels 2 and 3 for unaddressed issues in feedback culture.

2.5.3 Output Template

Use this lightweight template for documentation or reflection:

```
Observation:
Matrix cell:
Level(s) involved:
Function to strengthen:
Practice(s) selected:
Signal/metric to watch:
Review date:
```

You can use it in retrospectives, coaching sessions, or system diagnostics — it fits equally well in individual reflection or team learning contexts.

2.5.4 Practical Example (Full Loop)

Observation:
“Design team’s updates rarely align with development progress; misinterpretations surface during QA.”

Matrix Mapping:
Communication × Shared Understanding → Language / Terms

Level Check:
Level 2–3 issue: weak shared understanding and coordination function.

Function to Strengthen:
Clarify shared vocabulary and decision handoffs.

Practice(s):
- Introduce weekly alignment review with both design and dev leads. - Create a shared “definition of ready” document.

Signal/Metric:
Reduced QA rework or clarifications needed after handoff.

2.5.5 Why It Matters

This workflow transforms **improvement discussions** from reactive fixes to **systemic diagnosis**.
It replaces “who made the mistake?” with “which part of the system is under strain?” — building psychological safety while improving results.

> The HCS Diagnostic Workflow is the bridge between **theory and daily practice** — a loop that keeps cooperation systems alive, adaptive, and learnable.

Summary Table

Step	Purpose	Output
1. Observation	Capture neutral evidence	Statement of what happened
2. Matrix Mapping	Locate systemic dimension	Condition × Need = Function
3. Level Check	Identify root depth	Level(s) causing dysfunction
4. Function → Practice	Define safe intervention	Function and matching practice
5. Trial & Learn	Validate and adjust	Measurable feedback loop

Essence

> Every cooperation issue can be traced through the same lens:
> **Observation → Matrix → Level → Function → Practice → Learning.**

This is how theory becomes a living system.

3. REFERENCE

3.1 Glossary

The **Human Cooperation System (HCS)** defines a foundational language for describing the dynamics of **cooperative work systems** — how individuals and groups align, commit, and adapt together across levels of stability and change.
This glossary supports consistency across HCS Theory, Pyramid, Matrix, and Diagnostic Workflow.

3.1.1 Core System Terms

Term	Meaning
Human Cooperation System (HCS)	A diagnostic system that explains how cooperative work remains stable and adaptive across changing conditions. It models the "physics" of collaboration using the Pyramid, Matrix, and Level Rule.
Pyramid Model	Hierarchical structure of stability, from foundational human conditions (Level 1) to reflective meta-practices (Level 5).
Matrix Model (5×5)	Two-dimensional map crossing five cooperative dimensions with five human conditions to identify functional gaps or tensions in a system.
Level Rule	Governing principle stating that dysfunction at a lower level cannot be resolved by interventions at higher levels. Stability must be restored bottom-up.
Diagnostic Workflow	A repeatable method for observing behaviors, identifying affected levels and functions, and selecting interventions that respect the Level Rule.

3.1.2 Foundational Conditions (Pyramid Level 1)

Term	Meaning
Common Purpose	The shared intent that gives meaning and direction to collective effort.
Shared Understanding	The degree to which participants interpret information and goals in the same way.
Mutual Commitment	The willingness of each participant to invest effort and uphold agreements that sustain cooperation.
Trust	Confidence that others will act reliably and with integrity within shared norms.
Interdependence	Recognition that outcomes depend on others, making collaboration both necessary and valuable.

3.1.3 Systemic Enablers (Pyramid Levels 2–3)

Term	Meaning
Autonomy	The ability to make decisions within agreed boundaries; freedom coupled with responsibility.
Agency	The individual or group's capacity to act intentionally to influence outcomes within a system.
Distribution of Roles	How authority, accountability, and contribution are shared to balance autonomy and coherence.
Feedback Loops	Mechanisms through which information about performance or environment is used to adjust behavior and improve outcomes.
Change / Uncertainty	Natural variability in environment, priorities, or conditions that tests the adaptability of the system.

3.1.4 Cooperative Function Groups (Level 3)

Term	Meaning
Strategic Alignment	Ensures clarity of *what* and *why* — aligning purpose and understanding across participants.
Execution & Coordination	Maintains flow of work, role clarity, and cross-functional collaboration.
Systemic Governance	Establishes decision-making, delegation, and accountability mechanisms that balance trust and control.
Learning & Adaptation	Converts experience and feedback into collective improvement and resilience.
Communication Fidelity	Ensures that messages are transmitted, received, and interpreted as intended.

3.1.5 Principles and Rules

Term	Meaning
Function-First Principle	Effective practices must be matched to the cooperative function they serve, not adopted by imitation.
Lowest-Level Fix Rule	Interventions should always address the root level where instability originates.
Whole-System View	Understanding that all cooperative dimensions interact; improvement in one area affects others.
Reflective Practice	Periodic examination of how cooperation itself functions, leading to intentional redesign (Level 5 behavior).

3.1.6 Relationships to Other Frameworks

Term	Meaning
3-in-3 SDLC Framework (3SF)	A meta-framework derived from HCS principles that operationalizes cooperation in software delivery ecosystems.
Agile / Lean / Scrum	Examples of Level 4 frameworks that depend on stable lower-level cooperative functions to succeed.
Systems Thinking	The philosophical foundation of HCS — viewing cooperation as an interconnected adaptive system rather than a set of isolated practices.

3.1.7 Reference Purpose

This glossary ensures terminological clarity across all components of the Human Cooperation System and supports:

- Consistent interpretation of key terms in diagnostics and workshops.
- Alignment between HCS theory and practical applications such as 3SF.
- Educational use in training, coaching, and organizational learning contexts.

3.2 Sources and Frameworks

The **Human Cooperation System (HCS)** builds upon a wide range of theories, models, and disciplines that have shaped how humans understand **cooperation, trust, and adaptive systems**.
This section lists the **academic and conceptual influences** that inform HCS theory and its derivatives (including 3SF).

Each source is categorized by its **theoretical contribution area** — systemic, psychological, communicative, or organizational — highlighting how HCS integrates existing knowledge into a unified diagnostic model of human cooperation.

3.2.1 Systems Thinking and Cybernetics

Source / Framework	Core Idea	Relevance to HCS
General Systems Theory (Bertalanffy)	All systems — biological, social, or technical — share structural and functional patterns.	Provides the foundation for viewing cooperation as a *system of interacting elements* rather than isolated behaviors.
The Fifth Discipline (Peter Senge)	Organizations learn and adapt through feedback and systemic awareness.	Inspires HCS's focus on *feedback loops* and *collective learning* as stabilizing forces.
The Viable System Model (Stafford Beer)	Describes how systems maintain internal stability and external adaptability through recursive control loops.	Informs HCS's concept of *Systemic Governance* and *Autonomy–Control balance*.
Cybernetics (Wiener / Ashby)	Studies self-regulating systems and the role of feedback and control.	Underpins HCS's diagnostic logic and *Level Rule* — stability emerges from functional feedback, not hierarchy.
Complex Adaptive Systems (Holland / Gell-Mann)	Systems evolve through adaptation and local interaction under uncertainty.	Reinforces HCS's treatment of *Change and Uncertainty* as natural, not exceptional, conditions.

3.2.2 Human Motivation and Organizational Psychology

Source / Framework	Core Idea	Relevance to HCS
Hierarchy of Needs (Maslow)	Human motivation progresses from basic to self-actualizing needs.	Provides conceptual ancestry for the *HCS Pyramid* and the *Level Rule* — higher forms of cooperation rely on lower-level stability.
Self-Determination Theory (Deci & Ryan)	Autonomy, competence, and relatedness drive intrinsic motivation.	Directly informs HCS's dimensions of *Autonomy* and *Mutual Commitment*.
Theory X and Theory Y (McGregor)	Management assumptions influence employee motivation and trust.	Supports HCS's framing of *Trust* and *Agency* as emergent, not enforceable, conditions.
Psychological Safety (Edmondson)	Teams learn and innovate when members feel safe to take interpersonal risks.	Embedded in HCS's *Learning and Adaptation* function.
Social Exchange Theory (Blau)	Relationships are sustained through reciprocal benefit and fairness.	Grounds HCS's view of *Mutual Commitment* as both emotional and contractual.

3.2.3 Communication and Shared Meaning

Source / Framework	Core Idea	Relevance to HCS
Shannon–Weaver Model of Communication	Information transmission depends on reducing noise and distortion.	Forms basis for HCS's *Communication Fidelity* function.
Double-Loop Learning (Argyris & Schön)	True learning requires questioning underlying assumptions, not just correcting errors.	Informs *Reflective Practice* and *Level 5 Meta-Systems Thinking*.
Sensemaking (Weick)	People construct meaning retrospectively to understand ambiguous situations.	Explains why *Shared Understanding* is a dynamic, co-created process.
Nonviolent Communication (Rosenberg)	Empathic communication strengthens relationships and trust.	Reinforces the emotional intelligence aspect of *Trust* and *Feedback Loops*.

3.2.4 Organizational Design and Governance

Source / Framework	Core Idea	Relevance to HCS
Sociotechnical Systems Theory (Trist & Emery)	Optimal performance arises when social and technical subsystems are jointly optimized.	Validates HCS's integration of human and procedural stability layers.
Lean Thinking (Womack & Jones)	Continuous removal of waste to improve flow and value.	Reflects in HCS's *Execution & Coordination* function.
Kaizen	Continuous small improvements by all participants.	Embedded in *Learning & Adaptation* and the *Diagnostic Workflow*.
Adaptive Leadership (Heifetz)	Leadership is about enabling systems to adapt, not control.	Mirrors the *Systemic Governance* and *Agency* principles in HCS.
Organizational Learning (Argyris, Senge)	Organizations evolve when individuals learn within systemic feedback structures.	Reinforces *Systemic Learning* and the importance of *feedback closure*.

3.2.5 Philosophical and Ethical Foundations

Source / Framework	Core Idea	Relevance to HCS
Aristotle – Nicomachean Ethics	Virtue arises from practiced balance between extremes.	Influences HCS's pursuit of equilibrium between autonomy and control.
Kantian Ethics	Human cooperation is grounded in respect for rational agency.	Echoes in HCS's concept of *Agency* and moral interdependence.
Ubuntu Philosophy ("I am because we are")	Human identity and well-being are inseparable from community.	Embodies the relational worldview behind *Interdependence* and *Mutual Commitment*.
Phenomenology (Husserl / Merleau-Ponty)	Meaning is constructed through lived experience.	Supports HCS's principle that *Shared Understanding* emerges through interaction, not instruction.

3.2.6 Bridging Toward Practice

Source / Framework	Core Idea	Relevance to HCS
Agile / Lean / Scrum	Frameworks that operationalize adaptability and feedback.	Represent Level 4 practices that rely on stable lower-level HCS conditions.
3-in-3 SDLC Framework (3SF)	Systemic governance model for client–vendor ecosystems.	Serves as a *Level 5 derivative* — an applied embodiment of HCS principles.
Team Topologies (Skelton & Pais)	Structures teams for flow and cognitive load balance.	In practice, can be evaluated through HCS's *Execution & Coordination* function.
VMOSA / OKR / Wardley Maps	Strategic planning and situational awareness frameworks.	Connect to *Strategic Alignment* at Level 3, translating cooperative intent into measurable direction.

3.2.7 Reference Philosophy

HCS does not seek to **supersede** these theories — it **synthesizes** their enduring principles into a unified, practical system for diagnosing cooperation.

> **In essence:**
> **HCS = (Systems Thinking + Organizational Psychology) × (Communication + Adaptive Governance)**
> structured through the **Pyramid** (stability hierarchy) and **Matrix** (functional map),
> providing the theoretical foundation for applied frameworks like **3SF**.

3.3 Version and Licensing

This section documents the current version, license, and attribution principles for the **Human Cooperation System (HCS)**.
It ensures transparency, traceability, and consistency across all derivative works, educational materials, and frameworks built upon it — including the **3-in-3 SDLC Framework (3SF)**.

3.3.1 Version Information

Attribute	Description
System Name	Human Cooperation System (HCS)
Version	**V1.0**
Status	Stable — foundational version established as the theoretical core of cooperative system design.
Release Date	November 2025
Maintained by	3in3.dev
Repository	GitHub – vitar/hcs

Version Summary

Version 1.0 consolidates the **core theoretical architecture** of the Human Cooperation System, including:

- **Pyramid Model** — five hierarchical levels of stability, from human conditions to meta-practices.
- **Matrix Model (5×5)** — mapping of cooperative dimensions across fundamental human needs.
- **Level Rule** — the governing principle of hierarchical interdependence and systemic stability.
- **Diagnostic Workflow** — structured method for identifying dysfunctions and matching them to system functions.
- **Reference Section** — Glossary, Theoretical Grounding, Licensing, and Author information.

This release establishes HCS as the **foundational theory** upon which practical frameworks (such as 3SF) and diagnostic tools can be built.

3.3.2 Licensing

The **Human Cooperation System** and all related documentation are licensed under the:

Creative Commons Attribution 4.0 International (CC BY 4.0)

You are free to:

- **Share** — copy and redistribute the material in any medium or format.
- **Adapt** — remix, transform, and build upon the material for any purpose, including commercial use.

Under the following terms:

- **Attribution** — You must give appropriate credit, provide a link to the license, and indicate if changes were made.
- **No additional restrictions** — You may not apply legal terms or technological measures that legally restrict others from doing anything the license permits.

License Reference

Creative Commons Attribution 4.0 International (CC BY 4.0)

To attribute this work, please use the following reference:

> *Human Cooperation System (HCS) by 3in3.dev — licensed under CC BY 4.0 via GitHub repository vitar/hcs.*

3.3.3 Versioning Policy

- **Major versions (V2, V3, …)** introduce new theoretical constructs or expanded diagnostic models.
- **Minor revisions (e.g., V1.1)** include refinements, clarifications, or terminology alignment with derivative frameworks (e.g., 3SF).
- All published versions will remain **permanently available** for reference and citation.
- Future releases will maintain **backward compatibility** with the foundational definitions, rules, and models of HCS.

3.3.4 Attribution Guidelines

If reusing or adapting HCS content:

1. Include a visible credit line referencing *3in3.dev* and the license type.
2. Retain section numbering and core definitions where possible to preserve structural consistency.
3. When combining HCS content with other frameworks or methods, clearly separate attribution and derived materials.
4. For translations or derivative works, add a note identifying the adaptation (e.g., “Adapted from the original Human Cooperation System V1.0 documentation licensed under CC BY 4.0”).

3.4 About the Author

Viktor Jevdokimov, Vilnius, Lithuania — Creator of 3in3.dev, HCS, and 3SF

Viktor Jevdokimov is a software engineering leader, systems thinker, and framework designer with over 30 years of experience in software product delivery, modernization, and team alignment.

He is the creator of the **Human Cooperation System (HCS)** and the **3-in-3 SDLC Framework (3SF)**, and founder of the **3in3.dev** initiative — an independent platform dedicated to advancing collaboration and alignment between **Client**, **Vendor**, and **Product** ecosystems.

Professional Background

- Began career supporting distributed banking software on DOS and Windows, developing a deep appreciation for troubleshooting and system design.
- Progressed through roles of **developer**, **architect**, **delivery lead**, and **practice lead**, working with international clients on modernization and cloud migration initiatives.
- Specializes in **Client–Vendor relationship design**, **project leadership**, and **delivery system diagnostics**.
- Advocates for *“Context before Method”* and *“Trust before Control”* as guiding principles of effective collaboration.

Creative and Personal Work

Beyond software, Viktor is an **active musician and live sound engineer**, performing and mixing with the *Great Things* cover band.
He approaches both sound and systems with the same mindset: striving for **clarity, balance, and authenticity**.

About 3in3.dev

3in3.dev is an independent research and publishing initiative founded by Viktor Jevdokimov.
It consolidates his experience and experimentation into open frameworks that help organizations improve how they **engage, deliver, and measure value** across collaborative ecosystems.

3in3.dev publishes:

- The **Human Cooperation System (HCS)** — theoretical foundation for cooperative system design.
- The **3-in-3 SDLC Framework (3SF)** — practical application of HCS principles in software delivery.
- Supporting tools, templates, and learning materials under an open license.

> “These systems aren’t about control — they’re about clarity, trust, and the shared intent that makes collaboration work.”
> — Viktor J., Creator of 3in3.dev

Connect and follow on **LinkedIn** for updates and professional discussions.
For contact, collaboration, or speaking requests, visit **https://3in3.dev**.