

PINGUINISTA / 2026

The Augmented Mastery Manifesto

The person who mastered a craft can become a stranger to it.

LEONEL SALVO

Revised edition · 14 September 2026

penguinista.com/involut

The diagnosis

Generative productivity does not guarantee human growth. Delegation can remove the practice that built and sustained our judgment.

This can happen in programming, writing, design, analysis and other crafts: a machine delivers the finished result and may also remove opportunities to learn how to produce or judge it. The risk depends on what is delegated, how assistance is used and what practice remains. Code is our first application; the ambition is broader and must be tested in each domain.

The exchange can go unnoticed. Speed appears in a delivery, a deadline met, an invoice. Understanding needs its own assessment: producing more does not demonstrate understanding more.

Against that, authorship. Someone responsible for a work needs to understand the scope of their decisions without executing every step. We borrow two Latin words as working metaphors: *potestas*, the capacity to execute; *auctoritas*, authority over purposes and acceptance. These are neither technical categories nor a complete account of Roman law. Here, an *auctor* can explain, question and answer for a defined scope of a work.

The line we propose is this. The machine has latitude over means because that freedom is useful. People retain authority over purposes and acceptance within their responsibilities. Intervention requires understanding, access and the ability to act; a signature is not enough. Healthy delegation preserves the ability to check, stop, correct or seek help with what was delegated.

The danger has a name: *potestas sine auctoritate*. Execution without anyone willing and able to answer for it. Not a machine rebelling, but a capacity operating after the person responsible has disengaged.

What follows concerns how to sustain that line when machines produce faster than a person can understand.

Children of the magenta line

Warren Vanderburgh's *Children of the Magenta Line* (1997) offers a useful image: learn to manage different levels of automation and retain the ability to intervene when the selected level no longer serves the situation.

The response is not simply less automation. It is knowing when to move down a level and practising that level.

Fourteen years earlier, Lisanne Bainbridge described an irony of automation: the supervisor may need skills that automated work no longer allows them to practise. The analogy guides the question; it does not prove that every craft suffers the same effect.

Two different harms

The learner can deliver without acquiring the skills they need. This is learning debt: an acceptable result can hide an inability to explain, evaluate or change it. Rebuilding everything by hand is not required; practising the capabilities that will later be needed is.

The experienced practitioner can lose skills they once had. We call this risk from insufficient relevant practice cognitive atrophy. It differs from never learning, and a lower performance result alone cannot diagnose it. The organisation of work helps decide which capabilities are exercised and which are abandoned.

These are different paths: failing to build mastery, and becoming a stranger to something once understood. The method must accommodate both without confusing them or judging the person's worth.

Human growth does not require machines to stagnate

Studies of model collapse examine training conditions, not the fate of creativity. Gerstgrasser and colleagues found that retaining real data alongside synthetic generations avoided collapse in the settings they studied. That establishes neither a general ceiling on capability nor a guarantee of indefinite improvement.

Search with experiments and verifiers can also produce new results. AlphaEvolve is a bounded example in algorithms. It does not demonstrate universal autonomy; it does rule out claiming that every new result must arrive as an already written human answer.

Our commitment does not require the machine to freeze. Understanding, developing judgment and intervening remain human purposes even when machines execute many tasks better.

Growth in a machine's capability does not certify growth in its user. They are different outcomes and need different assessments.

This manifesto does not depend on an unemployment forecast or promise salary protection. It advocates a practice of human development; its learning effects need evidence and its economic effects are not guaranteed.

To those who say machines will surpass us anyway

We do not know the future. A machine doing a task better does not settle what we want to learn, what we need to verify or which outcomes we must answer for. Mastery is not only a competition to execute faster.

Choosing to keep growing is the commitment of this document.

Recovery takes time

Recovering skills is possible work, but its extent and cost vary. Renewed practice, feedback and assessment offer a path; we promise neither complete recovery nor a universal timetable.

For a defined work and scope, we propose three working states. **Ajeno**: needs orientation to build an explanation. **Partícipe**: can explain and question some decisions. **Situated mastery**: can justify relevant decisions, intervene and recognise limits. The same person can occupy different states in different parts of a work. These are provisional descriptions requiring evidence, not personal ranks or certifications.

A newcomer can ask questions and identify problems without a complete alternative. Orientation, explaining something back and prediction serve different purposes depending on the case.

Time spent producing without learning cannot be spent again. A work can yield both a result and growth in its participants. We want to organise work so that both happen.

Values

When these conflict, we favour the left.

We value	Over
Understanding	volume
Active understanding	passive reading
Acknowledging uncertainty	projecting confidence
Talking back	approving
Growth	maintenance
Real work	exercises

There is value on the right. We value the left more. The form is borrowed, with gratitude, from the Agile Manifesto.

Principles

01 - Productivity does not certify learning. Building and sustaining judgment is an explicit goal.

02 - Form determines whether the flood teaches or buries. A wall of undifferentiated changes makes understanding harder. Representation is part of the method.

03 - Understanding is demonstrated. Reading belongs in the process; explaining, checking and intervening help assess its result.

04 - Alternatives and limits are material too. An agent's explanation is a claim to check, not guaranteed access to its internal reasoning.

05 - Start from work that matters. Use exercises or variations when they isolate a difficulty or help test transfer.

06 - Effort must serve learning. Preserve relevant difficulty and remove what merely gets in the way.

07 - Augmentation may only add, never hide. No layer may be the only route to the original material.

08 - Understanding cannot be delegated. Execution can be delegated; responsibility for judgment cannot be discharged by accepting an answer.

One gesture

Before looking at what the machine produced, take a position.

Before opening a change, draft, image or analysis, formulate an expectation about a decision and its consequences. Then check it against the work and evidence from the domain. Agreement with the machine is not understanding: one can correctly anticipate its mistake.

"I have no position" is a valid answer and an opening for orientation. No answer can also mean insufficient time or an ambiguous question. Participation provides a signal; evaluating understanding requires explanation, evidence and intervention. We do not assign a state from silence or agreement with an agent.

Taking a position before seeing an answer makes expectations and outcomes comparable. It is a candidate practice, not a guarantee of learning. Self-assessment can guide work but needs comparison with observable performance.

And periodically, an audit. Return to a work after an interval, explain it without the agent's answer, solve a variation and intervene on a problem. Distinguish retention, transfer and autonomy; acknowledge assistance and limits. Define the protocol and thresholds before measuring.

This can be done by hand, without installing anything. How to sustain it over time belongs to the methodology.

What this is not

A rejection of machines. Delegation is useful. The aim is to make it compatible with human development and judgment.

Nostalgia for manual work. Some routines build useful patterns; others only consume time. Recover practice for what it teaches, not how much effort it demands.

A course or certificate. The starting material is the person's own work. The states describe a relationship to that work, not a credential.

A replacement for quality control. Finding defects remains necessary. Here the additional question is whether the person understands the decisions and their consequences.

A productivity promise. Time devoted to understanding has a cost. The method must show what learning it produces and whether that cost is sustainable.

In one line

The machine makes the work. This is why we must remain, and become more capable as, the people who answer for it.

From manifesto to method

Auctor is the method under development that turns this philosophy into practice. Its name draws on authorship; it is not a certification. **Epistemic Overdrive** names the selection of relevant cases from a larger volume; it does not promise years of experience in weeks. **The Metacognitive Loop** names the cycle of producing, taking a position, checking and revising one's understanding. These name a property and a way of working, not evidence of effectiveness.

Involut is the product applying Auctor, starting with code. The philosophy and method live in a separate repository from the software. Parts of the methodology remain unfinished. This manifesto does not present those future capabilities as implemented.

Sources and scope

- Bainbridge, L. (1983). *Ironies of Automation*. An antecedent for the supervision/practice problem, not a universal finding about generative AI.
- Naur, P. (1985). *Programming as Theory Building*; Polanyi, M. (1958). *Personal Knowledge*.
- Vanderburgh, W. (1997). *Children of the Magenta Line*. A practical analogy about managing automation.
- Engelbart, D. (1962). *Augmenting Human Intellect*; Carr, N. (2014). *The Glass Cage*.
- Shen, J. H. and Tamkin, A. (2026). [How AI Impacts Skill Formation](#). An experiment on learning a programming library, not proof of inevitable decline across crafts.
- Gerstgrasser, M. et al. (2024). [Is Model Collapse Inevitable?](#). Conditions for avoiding recursive collapse, not a universal ceiling on novelty.
- Novikov, A. et al. (2025). [AlphaEvolve: A coding agent for scientific and algorithmic discovery](#). Search with evaluation in bounded domains.
- *Manifesto for Agile Software Development* (2001) and *Manifesto for Software Craftsmanship* (2009). Antecedents for the values format.

License

Copyright © 2026 Leonel Salvo · Pinguinista. This edition is licensed under [CC BY-SA 4.0](#). Attribute the author, identify changes and share adaptations under the same or a compatible license. Third-party quotations and references retain their own rights. Earlier permissions remain valid; see [the transition record](#). Names and logos are not licensed as trademarks.