Dmitry Leushkin

v 2.1

ENGINEER'S EDITION

TURBO-SUSLIK

A brutal, high-throughput framework
for refactoring the legacy code
running in your head



Debug your mind.
Stop fucking the deploy.
Start living.

Table Of Contents

Introduction *4*

Acknowledgments 4

Copyright and License 5

Disclaimer and Use Restrictions 8

Instead of a Preface — Who This Book Is For 13

Is There Any Point in “Trying”? 15

About the Author and the History of the System 16

Theory *20*

About the System 20

About The Subconscious 24

Patching vs Refactoring 27

The Fairy Tale of the Root Commit 29

Legacy code, Skeletons in the Closet, and Freedom 31

Polarities: A Technical Definition 34

Clearing the Repo (Mind) of Dead Code (Junk) 36

Integration and Defragmentation of the Personality 38

Instructions for the Subconscious 39

Backlashes, a.k.a. “Pendulums” 42

Philosophical Aspects of the System 48

Practice — Phase 1 ..... *50*

How Processors/Handlers Work in the Turbo-Suslik System 51

Processing Polarities 55

The “Merge” Processor 58

How to Merge Polarities: Technical Reference 62

Processing Problems 63

The Operating Principle of the "Process this" Processor 64

The “Process this” Processor 66

Processing Emotional States 73

Cyber-Zhivo 75

List of Polarities 76

Emotional Clusters 85

Work Plan 91

What Will You Get as a Result? 92

Practice — Phase 2 ..... *93*
Processing the Past **93**
“Chpok” and “Hoppo.” Incident Processing **96**
Working with Problems **99**
Beliefs **102**
Circular Processing **104**
Money Beats Evil — On Processing Money **106**
Total Merge **108**
Automations **110**
Why Isn't It Just Gone? (On Non-Linear Processing) **112**
Strategy for Using Automations (Protocols) **114**
When Do Results Actually Show Up? *115*
If Something Doesn’t Get Processed **117**
Fully Independent Work: A Basic Roadmap **119**
What the Overall Trajectory Tends to Look Like **120**
Conclusion *121*

Introduction

Acknowledgments

To everyone who has ever striven for freedom and contributed to the development of the techniques available today;

To the creator of TFT, **Dr. Roger Callahan**, as well as the creator of EFT, **Gary Craig**, both from the United States;

To the creator of BSFF, **Dr. Larry Nims**;

To the creator of a whole range of brilliant techniques, including PEAT, DP4, and others — **Živorad Mihajlović Slavinski** of Serbia;

To the author of The Resolution of Mind, **Dennis H. Stephens**;

To **Strannik** for a wealth of excellent ideas and processing material;

To **A. V. Usachev**, author of Game Processing, for several valuable ideas without which this system would be less effective;

To **Oleg Zhuravsky** for the great idea of combining direct work with the subconscious with other effective techniques;

To all members of the first group of practitioners known by the codename **“Susliks,”** who worked with new techniques alongside me, supported me, shared their successes and failures, and went through all the ups and downs;

To each and every Turbo-Suslik user — more than 100,000 over more than eighteen years of the system’s existence — who got rid of mental ballast and boldly and patiently changed their lives;

To the Turbo-Suslik **project team**, who tirelessly handled technical and administrative matters;

To my beloved wife, **Oksana**. Without her, this book simply would not exist.

Copyright and License

© 2008–2026 Dmitry E. Leushkin

Work: *Turbo-Suslik 2.1: Engineer's Edition*
Edition: Engineer's Edition, 2026
Author and copyright holder: Dmitry E. Leushkin

Dmitry E. Leushkin is the author of the Turbo-Suslik System, its original protocols, and the original material presented in this book.

Copyright protects the text of this book, the original wording and structure of the protocols, the selection and arrangement of the material, the explanations, and other original forms of expression. It does not restrict the use of ideas, facts, methods, or principles that are not protected by copyright.

Creative Commons License

Except for third-party materials identified below or elsewhere in the book, this edition is licensed under the:

Creative Commons Attribution–NonCommercial 4.0 International License (CC BY-NC 4.0)

License:

https://creativecommons.org/licenses/by-nc/4.0/

Under this license, you may:

- copy and redistribute the book in any medium or format;
- publish and distribute complete or partial copies;
- convert the book into other electronic formats;
- produce and distribute printed copies;
- translate, abridge, expand, modify, remix, or otherwise adapt the material;
- create and distribute derivative works.

These permissions apply only to noncommercial use and are subject to the terms of the CC BY-NC 4.0 license.

When sharing the original or modified material, you must:

- give appropriate credit to Dmitry E. Leushkin;
- identify the original work;
- provide a link to the official source where reasonably practicable;
- provide a link to the CC BY-NC 4.0 license;
- indicate whether changes were made;
- retain an indication of any previous modifications;
- not imply that the author endorses you, your organization, or your use of the material;

- not impose legal or technological restrictions that prevent others from exercising the rights granted by the license.

The full legal text of the license controls if there is any conflict between this notice and the license:

https://creativecommons.org/licenses/by-nc/4.0/legalcode

Recommended Attribution

“Dmitry E. Leushkin, *Turbo-Suslik 2.1: Engineer’s Edition*, 2026. Official source: https://mindrefactory.dev/. Licensed under CC BY-NC 4.0.”

For modified materials, the following notice is recommended:

“This material is adapted from *Turbo-Suslik 2.1: Engineer’s Edition* by Dmitry E. Leushkin. Changes were made. This adaptation is not an official edition and is not endorsed by the author. The original work is available at https://mindrefactory.dev/.”

Scope of This License

This license applies only to *Turbo-Suslik 2.1: Engineer’s Edition, 2026*, and only to material for which Dmitry E. Leushkin holds the necessary rights.

Other editions of Turbo-Suslik remain governed by the copyright and licensing notices included in those editions.

This notice replaces any previous copyright or temporary licensing notice used for this edition. The CC BY-NC 4.0 license takes effect for this edition on 28 October 2026.

Third-Party Materials

This book may refer to techniques created by other authors, including:

- PEAT and DP-4 — Zivorad Slavinski;
- BSFF — Larry Nims;
- EFT — Gary Craig.

Rights in independently protected third-party descriptions, materials, trademarks, and other content remain with their respective rights holders.

The CC BY-NC 4.0 license does not grant rights in third-party material that Dmitry E. Leushkin does not own or have authority to license. References to third-party techniques do not imply ownership, affiliation, approval, or endorsement.

Commercial Use

The CC BY-NC 4.0 license does not authorize use that is primarily intended for commercial advantage or monetary compensation.

Commercial publication, distribution, licensing, incorporation into paid products or services, or other commercial use requires separate written permission from the copyright holder, except as expressly permitted under the official affiliate program below.

Official Affiliate Program

Registered participants in the official Turbo-Suslik affiliate program may distribute the authorized electronic edition supplied to them with their assigned affiliate link and may receive compensation under the terms of that program.

This is separate authorization for that specific commercial activity. It applies only to registered participants, only to the authorized affiliate edition, and only while the participant complies with the applicable program terms.

It does not authorize participants to sell the book, charge for access to it, create unauthorized modified editions, remove copyright or licensing notices, replace affiliate links, grant sublicenses, or use the book outside the official affiliate program.

No Endorsement or Trademark License

Use of this book or its materials does not imply endorsement by Dmitry E. Leushkin.

The CC BY-NC 4.0 license does not grant trademark rights or permission to use the Turbo-Suslik name, Mind ReFactory identity, logos, or associated marks in a manner that suggests an official edition, affiliation, sponsorship, or endorsement.

Disclaimer and Use Restrictions

Official part:

This book describes Turbo-Suslik, a set of self-directed procedures presented for general informational and educational purposes and for voluntary, independent personal use. Turbo-Suslik is not therapy of any kind. It is not medical or mental health treatment, psychotherapy, psychological or psychiatric practice, counseling, coaching, crisis intervention, or any other professional service.

The publication, distribution, purchase, possession, reading, or use of this book does not create a doctor–patient, therapist–client, psychologist–client, counselor–client, coach–client, fiduciary, supervisory, or other professional relationship between the reader and the author, the rights holder, the distributors, the contributors, or anyone else associated with Turbo-Suslik. No such person or entity undertakes any duty to assess, monitor, supervise, or manage the reader's use of the procedures described in this book.

In connection with this book, the author and the rights holder:

1. **do not diagnose or assess any physical, medical, psychological, or psychiatric condition;**
2. **do not prescribe, select, alter, suspend, or discontinue any medical or mental health treatment;**
3. **do not provide recommendations concerning the use, discontinuation, replacement, or dosage of any medication;**
4. **do not provide individual advice, consultations, supervision, psychological services, psychotherapeutic services, or any other form of professional assistance;**
5. **do not determine whether these procedures are safe, suitable, or appropriate for any particular reader;**
6. **do not promise or guarantee any particular result, improvement, change, or effect.**

The procedures described in this book are intended solely for self-directed personal use. This book does not qualify, certify, license, or authorize anyone to provide services to others. The procedures must not be administered to, applied to, or performed on behalf of another person, whether for payment or without payment.

These procedures must not be used by anyone under the age of eighteen; by anyone with a diagnosed or suspected mental disorder, including but not limited to depression, bipolar disorder, schizophrenia spectrum disorders, severe anxiety, panic disorder, or PTSD; by anyone experiencing suicidal thoughts, self-harm impulses, psychosis, mania, dissociation, or any other acute mental health crisis; by anyone currently taking psychotropic medication, tranquilizers, sedatives, or other medication that materially affects mood, perception, or cognition; by anyone with a brain tumor, epilepsy or another seizure disorder, or severe or uncontrolled hypertension; or by anyone under the influence of alcohol, recreational drugs, or other intoxicating substances. Use during pregnancy is strongly discouraged.

Anyone experiencing an acute medical or mental health crisis, suicidal thoughts, or a risk of harm to themselves or others must not use these procedures and should

contact local emergency services or another qualified source of immediate professional assistance.

Any decision to use the procedures described in this book is voluntary, unilateral, and entirely self-directed. The reader assumes sole responsibility for deciding whether and how to use them, for following or disregarding any part of the material, and for all interpretations, actions, omissions, and consequences arising from that decision. Use is entirely at the reader's own risk.

To the fullest extent permitted by applicable law, the author, the rights holder, the distributors, and the contributors disclaim liability for any direct, indirect, incidental, consequential, special, or other loss, injury, damage, deterioration, expense, or adverse consequence arising from or related to the use, misuse, attempted use, inability to use, or reliance upon the material in this book, including the absence of expected results. Nothing in this disclaimer excludes or limits any liability that cannot lawfully be excluded or limited.

By choosing to use any procedure described in this book, the reader acknowledges the nature of the material, the absence of supervision or professional services, the stated restrictions, and their sole responsibility for that choice. If you do not accept these conditions, do not use Turbo-Suslik and close the book immediately.

Unofficial part:

IMPORTANT CLARIFICATION:

Now for the unofficial version.

Turbo-Suslik is a tool for maintenance work on your own psyche. A screwdriver or a wrench. It does not hold your hand, decide what is safe for you, or take responsibility for what you do with it.

You pick it up and use it at your own risk, on your own, with full responsibility for the consequences. There is no operator sitting at the other end, no supervisor watching your progress, and no support desk waiting to take over when things become uncomfortable.

If you encounter a **pendulum** — a temporary increase in emotional intensity, instability, or system pushback during processing — do not treat it as a malfunction or a reason to quit. Pendulums are a normal part of accelerated change. When old patterns begin to lose stability, connected material may become temporarily more active before the system settles into a new configuration. This is part of the price you pay for speed and real change. Observe what is happening, process the material that surfaces, and continue according to the instructions.

A pendulum, however, is not a convenient label for absolutely any condition. The boundary is not discomfort, pain, or emotional intensity. The boundary is the loss of your ability to observe what is happening, make responsible decisions, function safely, or remain in control of your actions. If that boundary is crossed, stop and contact an appropriately licensed psychiatrist or clinical psychologist. Not a coach, facilitator, motivational expert, or internet shaman. **I do not provide individual advice, guidance, support, or supervision, and I cannot take responsibility for what you choose to do to yourself.**

This book is a technical manual. Nothing more. If you expect care, supervision, rescue, or someone to lead you by the hand, close it now.

***

Everything written in this book reflects my personal practical experience using these techniques. I do not claim to present the “ultimate truth,” and I do not claim that my perspective is the only correct one. I describe what I have done myself, what I have observed in my own work, and what I have observed in people who used these methods and shared their results with me.

I may be wrong about some details. Some things may eventually be explained better, implemented differently, or made more effective. That is not critical for one simple reason: I, along with thousands of people who have used the system, have observed concrete practical changes in reactions, behavior, and the overall quality of life. That is enough to regard this approach as workable for a certain group of people.

There is no such thing as a “mandatory result.” You may experience substantial changes, as many people have. You may experience little. You may experience nothing useful at all. No guarantee is possible because too much depends on your starting condition, the amount of work you actually do, your level of honesty, and your willingness to stop lying to yourself.

The restrictions in the formal notice above are not decorative legal wallpaper. If any of them apply to you, stop there. Do not turn recklessness into courage, and do not call ignoring an explicit restriction “taking responsibility.” It is nothing of the kind.

The process may involve extremely strong negative emotional reactions. Old material may surface: fear, anger, shame, anxiety, guilt, helplessness — sometimes with considerable intensity. This can be part of the unpacking process. It is not automatically a malfunction, but it is not automatic proof of progress either. The intensity may be considerable; that alone does not mean that something has gone wrong. Pendulums are part of the process, and speed has its price. The relevant boundary is whether you remain capable of observing the process, making responsible decisions, and acting safely. Do not confuse severe discomfort with danger — but do not rename an actual loss of control “deep processing” either. The system awards no medals for psychological self-demolition. If you are not prepared to face yourself without anesthesia, it is better not to begin.

Over time, many people notice that old illusions and beliefs lose their grip and the emotional background changes: fewer dramas, fewer emotional swings, more calm clarity and detachment. Some people like this. Others hate it, because familiar internal melodrama, victim identity, and social games begin to disappear. **If your traumas are central to your identity, your grievances serve as your capital, and your “complexity” has become a lifestyle, this system is not for you.**

Old motivations and goals may also begin to feel strange, and some social roles may lose their meaning. This can be uncomfortable. If you are strongly attached to status, approval, and the perceived importance of things, it may be better not to begin. This is not about rejecting money, ambition, or material life. It is about no longer selling yourself for validation.

Some users report changes in their physical well-being. Fine. The psyche and the body are not sealed compartments. But an incidental observation is not the purpose of the system and does not turn it into treatment. Do not use someone else’s anecdote as an excuse to smuggle medical expectations into Turbo-Suslik through the back door.

The system is strictly self-directed. If you are looking for a guru, mentor, facilitator, or someone to guide you by the hand, this is not for you. No one is coming to save you here. Either you take full ownership of the work or you close the book.

The restrictions are literal. Under eighteen means under eighteen, not “mature for your age.” During pregnancy or breastfeeding, leave the practice alone. Optional experiments involving intense emotional load are a bad trade during that period.

In short: you use the tool yourself. The risk is yours. The work is yours. The results are yours. The consequences are yours.

If you do not accept that arrangement, close the book and do not continue.

Distinction of Systems and Terminology

The “Turbo-Suslik” system is not a system of Zivorad Slavinski (including PEAT and “Spiritual Technology”), has not been researched, approved, tested, or endorsed by Zivorad Slavinski, and has no direct relation to his methods.

In the development of “Turbo-Suslik,” certain general ideas and principles present across various schools and approaches may have been used. However, this system is an independent development, with its own principles, objectives, and format of application, and it is not a substitute for any other methods.

The “Turbo-Suslik” system is also not related to Scientology and is not a form of auditing, processing, or any similar practices. It does not complement such practices and is not intended to replace them.

The same applies to any other fields or schools — psychology, psychotherapy, NLP, coaching, or similar approaches: “Turbo-Suslik” is not part of them, is not licensed by them, and is not presented as a variation of those systems.

Instead of a Preface — Who This Book Is For

If you have begun to understand that you yourself are the source of a significant part of your problems in life — your complexes, resentments, traumas, fears, beliefs, habitual reactions, and all the accumulated weight of the past;

If you are genuinely tired of living under negative conditioning, low self-esteem, and chronic insecurity;

If you see clearly that, in practical terms, there is no alternative but to throw all this baggage out;

If words like “awareness” and “freedom from automatic patterns” are not decorative abstractions for you, but describe a real need — to stop living on autopilot;

If you are fed up with techniques that require you to believe in something, imagine something, “feel correctly,” think prescribed thoughts, or manufacture a special state — and you have been looking for a practical tool that requires neither rituals nor self-deception;

If you want concrete changes: less mental clutter, less internal conflict, fewer imposed labels, fears, complexes, traumas, and fewer endless replays of the past;

If you want to start living in reality instead of the perpetual “later” — once I finish university, get the position, complete the project, find investors, have children, retire, then I will finally start living properly;

If you are tired of living in your head, consuming mountains of literature, and attending endless webinars, courses, and training programs supplied by an industry that too often sells empty boxes in attractive packaging;

If you are ready to work and understand that there are no shortcuts and never will be;

If you are ready to face your inner conflicts, fears, traumas, and uncomfortable truths, despite the pain and discomfort that may arise in the process;

If you understand that no one can give you freedom “as a gift” — at best, someone can provoke you, shake you, and confront you with a choice: act or continue talking;

And if, at the same time, you have no desire to spend long years on slow practices and endless conversations — then the techniques described in this book may be of interest to you.

There is one main obstacle: the refusal to take responsibility for what is now running inside you. You may not have chosen what happened to you. You may not have installed the original damage. That changes nothing about who has to deal with it now. If you believe that whoever caused the damage must remain responsible for repairing it; if you expect someone to “fix” your life for you, hand you a magic pill, guide you step by step, or “heal” you; if you are not prepared to persistently dismantle the mental clutter accumulated over the years — close this book. It is not for you.

In short, this book is for people who want to change, are ready to change, and will actually do the work. For those who will work instead of complaining, nursing resentments, or endlessly outsourcing responsibility to their past, the people close to them, “circumstances,” fate, or “bad techniques that don’t work.” This is a book for practitioners.

For ordinary people — men and women, young and old, grandfathers and grandmothers (better late than never) — who have decided to take charge of their own psyche and begin putting it in order, without fairy tales and without self-deception.

This system is a tool for work. More importantly, it is a tool for **independent work**. We are talking about individual practice without teachers, mentors, trainers, coaches, seminars, webinars, chats, Zoom calls, or any other form of built-in “social support.”

At the very beginning of Turbo-Suslik, back in 2008, I was convinced that a person who genuinely wants to bring order to their own mind, and is ready to do the work, is capable of accomplishing a great deal independently. More than eighteen years have passed since then, and the experience of tens of thousands of users has confirmed this repeatedly.

You really can produce substantial changes completely on your own through relatively simple but systematic procedures, described in precise detail below.

I do not promise easy solutions. Changes can happen relatively quickly, and often do. But there will be no shortcuts. You will have to work, and speed also has its price. The ways in which resistance and system “pushback” typically appear will be discussed separately in the section on “pendulums.” If you are ready for real work and possible difficulties — including intense discomfort, emotional pain, and sobering disappointments — continue reading.

I do not claim that this system is “the most effective” or “the best.” There is no universally best technique outside the context of a particular person, task, and starting condition. This is simply one tool. For a certain category of people who are ready for this kind of work, it may prove extremely effective. If you belong to that category and get results, I will be sincerely glad. If you do not, nothing is wrong. You need a different tool. Go and find it.

One more important point: Turbo-Suslik is not designed as a spot repair for one or two isolated problems. It is aimed at a cumulative, system-level effect, produced through regular work with large volumes of internal material and automatic patterns. If you only want to “fix a couple of glitches,” this is not the best tool.

And once again: this book is for those who are ready to become 100% their own engineer, mechanic, and test subject. There are no teachers here, no rescuers, and no “hotline” you can call to have everything fixed for you. There is you, your problem, and the tool.

That is all.

Is There Any Point in “Trying”?

It is very easy to determine whether this method is suitable for you. Try it. Consider it a demo version.

After approximately two or three weeks of practice, you should already have enough information to identify the first changes from working with Turbo-Suslik — or their absence.

In other words, even such a short period is enough to form a preliminary impression. This does not mean that your life will be completely transformed in two weeks. It means that by this point you should be able to notice whether anything small but clear and observable has begun to change. A few days after processing an emotionally charged episode, you may find that your perception of it has shifted. A person you used to strongly dislike may no longer trigger the same reaction. A pressing current problem may still be there but clearly feel less intense. A long-standing issue that has been looping through your head for years may suddenly feel different. And so on.

The important thing is not to invent results, persuade yourself that something has happened, or attempt to “believe” or “not believe” in Turbo-Suslik. That is a separate problem altogether. Remain as impartial an observer as you can.

A useful tool should allow itself to be tested. You now have an unusual tool in your hands. Does it work? Test it. Practice. Experiment. But do not become a believer. Stay grounded and keep watching what actually happens. Be open enough to run the test and skeptical enough to observe it honestly. Start from where you are, follow the instructions, and after a couple of weeks compare your actual reactions with what they were before.

If, after four weeks of practice, you can detect no changes at all, pause. Do not push harder merely because you want the experiment to succeed. Continue observing your reactions and general state for another two to four weeks, because some changes may become visible only after active work has stopped. If you still notice nothing, then Turbo-Suslik is most likely not suitable for you.

About the Author and the History of the System

In 2026, the Turbo-Suslik System is more than 18 years old.

Traditionally, this would be the place to begin with a solemn list of credentials, academic titles, certificates, coaching hours, impressive photographs, grand personal missions, and other respectable decorative material. The reader would then immediately understand that this is not some ordinary person speaking, but a serious bearer of unique knowledge whose words should be trusted because the biography has been arranged correctly.

Turbo-Suslik was never created in that genre.

My name is **Dmitry Leushkin**. I am the author of the Turbo-Suslik System, the book BSFF 2.0, the book Turbo-Suslik, the protocol collections, and the new version of the system — **Turbo-Suslik 2.0**. For more than twenty years, I have been working with practical technologies of cognitive correction, self-directed processing, protocol-based systems, and the processing of large volumes of mental material.

At the same time, it is important to say this directly: I am not a psychologist, not a psychiatrist, not a medical doctor, not a guru, not a spiritual teacher, and not a rescuer of humanity. Turbo-Suslik was not created as a medical, psychotherapeutic, or spiritual system. It is a tool for self-directed work with one's own mentality, built around protocols, automation, structured instructions, and many years of practical adjustment.

The story did not begin with a desire to create a "method for the market". It began in a much simpler and more honest way: at some point it became obvious that reading clever books, attending seminars, having philosophical conversations, endlessly digging through theories, and engaging in other forms of mental gymnastics changed almost nothing at the level of real life. A person may speak beautifully about freedom, awareness, trauma, the subconscious, spirituality, psychology, and the meaning of existence, while continuing to live out of the same automatisms, fears, resentments, internal conflicts, and old reactions.

At some point, the difference between "understanding" and actually changing became too obvious to keep pretending that reading another wise book was going to solve anything by itself.

The first important step was **taking responsibility for my own state**. Not in the sense of guilt, self-blame, or a pretty sentence from a motivational poster, but in the most practical sense: if all this mess is in my head, then I am the one who has to work with it. No one can do it instead of me. A teacher, psychologist, coach, book, or technique may provide a tool, but the work itself still happens inside the person.

Then came a long search for working instruments.

At first, there were different approaches to reprogramming: affirmations, self-hypnosis, suggestions, technical means of influencing states, attempts to install new behavioral models on top of the old ones. Some of it produced temporary effects, some of it was interesting, some of it seemed promising, but in the long run it became clear: if the roots of the problem are not addressed, a new "correct" belief is simply pasted over the old mental structure. Like fresh paint on a rotten wall. For a while it looks cheerful. Then the old layer comes through anyway.

That is how I gradually came to the idea of **deprogramming** — not installing new beliefs over old ones, but clearing the material that is already present in the mind and continues to automatically determine perception, reactions, behavior, and inner state.

One of the first practically useful tools for me was **EFT** by Gary Craig. Later came **BSFF** by Larry Nims, a system of direct work with subconscious material that was much closer to the direction that interested me. Based on my own experience and the experience of other people, I later wrote the book BSFF 2.0.

BSFF gave me a lot, but with active work it quickly became clear that manual processing of separate problems was too slow. There are not dozens or hundreds of problems. There are thousands. And if we are being honest, we are not really dealing with “problems” as isolated objects at all, but with entire layers of mental material connected through memories, decisions, fears, resentments, beliefs, bodily reactions, relationships with other people, and internal parts of the personality.

Later I became familiar with the techniques of **Zivorad Slavinski**. They provided important ideas, especially in relation to working with polarities, personality aspects, charge, and deeper mechanisms of perception. These were powerful instruments for individual processes, but the speed still did not satisfy me. Not because I wanted a free ride, but because the scale of the task was far beyond the capacity of manual work.

If you need to clear not one room, but an entire abandoned warehouse, it makes little sense to argue about which cloth removes dust better. You need a different architecture.

The key turn happened when it became clear that if the real work in such techniques is, in any case, performed by non-conscious mechanisms of the psyche, then it should be possible to describe not just separate actions to those mechanisms, but whole procedures. Not merely “process this”, but define the structure of processing, the sequence of operations, conditions, checks, additional branches, processing of resistances, consequences, decisions, related episodes, and other levels of material.

In other words, to move from separate techniques to a protocol-based system.

That is how the **Turbo-Suslik System** gradually began to form.

The first versions were rougher, simpler, in places more naive, and technically less clean. This is normal. Any living system first appears as a working draft, then passes through practice, errors, observation, correction, and gradual tuning. In 2008, the first Turbo-Suslik book was released, and with it the public stage of the system effectively began.

Since then, more than 18 years of active practice, correction, tuning of mechanisms, and observation of the system’s use by different people have passed. During this time, Turbo-Suslik has gone through several generations of protocols, many technical clarifications, revisions of particular approaches, accumulation of user experience, feedback, private reports, letters, observations, and practical data.

It is important to understand that this is not about clinical research in the medical sense. Turbo-Suslik is not a medical technology and is not intended for diagnosis, treatment, or replacement of professional help. However, more than 18 years of real-world use, hundreds of thousands of users around the world, a large body of accumulated feedback, and the repeated appearance of many observed phenomena allow us to say that the basic concept of the system has proved practically valid within its own model: as a tool for self-directed protocol-based work with mental material.

This does not mean any guarantee of results for any particular person. People are different. Initial states are different. Readiness for self-directed work is different. The ability to look at oneself honestly is different. The level of internal resistance is also different. Therefore, Turbo-Suslik does not promise identical outcomes for everyone and is not a universal pill.

But the core idea — using structured protocols to initiate self-directed background processing of large volumes of mental material — has shown its viability over the years.

The new version, **Turbo-Suslik 2.0**, is an attempt to bring the system into a more modern, precise, and technically clean form. Less unnecessary metaphysics, less old terminological debris, more clarity, more structure, and a more sober description of what is actually being done. Not because the old version was “wrong”, but because language must mature together with the system.

At the beginning, much was described through the words that were available at the time: subconscious, programs, charge, polarities, pendulums, processors, automats. Some of these terms remain useful even now, because they are practically convenient. But the general focus is gradually shifting from mystification toward an engineering description: there is material, there is a protocol, there is the launch of processing, there are observable changes, there are limitations, there are transitional side effects, and there is the necessity of personal responsibility.

The system does nothing “instead of the person”. It does not cancel life, the body, common sense, the need to think, or responsibility for one’s own actions. It simply provides a way to work systematically with what usually remains in the background and controls a person without their explicit participation.

In this sense, Turbo-Suslik has never been a philosophy, a teaching, or a path. It is a tool.

One can argue endlessly about the nature of the subconscious, the psyche, automatisms, memory, internal parts, perception, and personal reality. One can spend years building elegant theories. One can quote ancient, modern, Western, Eastern, and any other authorities. But in the end, the question is simple: does a particular instrument work for a particular person in a particular practice?

Turbo-Suslik offers exactly that: verification through practice.

Not belief. Not worship. Not joining a club. Not searching for a teacher. Not waiting for someone to lead you by the hand. Take the material, begin working, observe yourself, and soberly look at what happens.

If after some time a person sees that their reactions are changing, old charges are losing strength, the past is becoming easier to perceive calmly, automatisms are weakening, internal noise is decreasing, and perception is becoming clearer — then the instrument works for that person.

If not, then either it does not suit them, or the work is not being done properly, or there are resistances, or it is simply not their instrument. There is no tragedy in that. There are no techniques that suit everyone.

That is why Turbo-Suslik never had the idea of “converting everyone”. The system was originally, and remains, an instrument for those who are ready for self-directed work, who

are not looking for emotional servicing, who do not want to spend years discussing their problems, but are ready to take material and process it in a working manner.

Over the years, many stories, opinions, arguments, enthusiasms, objections, expressions of gratitude, misunderstandings, and a rather peculiar folklore have accumulated around the system. This is inevitable for any living system that exists for a long time and is used by a large number of people. But beneath all this noise, the simple foundation remains: there is a person, there is their internal mechanics, there is accumulated material, and there is a possibility to work with it.

This is the essence of Turbo-Suslik.

Not becoming the “best version of yourself” in the marketing sense. Not attracting a magical life. Not receiving a spiritual medal for your chest. Not proving your advancement to someone. But gradually reducing the amount of internal garbage, automatism, fixation, reactivity, and mental noise.

And then living.

That, in fact, is what it was all created for.

Theory

About the System

The techniques described in this book form the foundation of a system for independent work with the psyche. Many practitioners who work consistently and without self-deception notice changes in their state, perception, and reactions within a few months. There are no guarantees: the result depends on your initial condition, your honesty with yourself, and whether you actually process the material instead of looking for a magic solution.

All the techniques are based on direct work with the subconscious through previously installed programs. In this respect, Turbo-Suslik has something in common with BSFF, which you may have heard of or practiced with varying results. The important differences lie in the overall strategy, the depth of processing, and the degree of automation.

The technical mechanics of working with the subconscious will be covered in the next chapter. For now, it is enough to understand the basic operating model: you identify the material, formulate the target, and launch a protocol. The installed programs perform much of the routine processing without requiring you to consciously dissect every memory, reaction, and internal connection.

Strategy

The strategy of the system consists of four main areas.

1. Release from the Past

Current reactions are often powered by material accumulated in the past: traumatic episodes, resentment, fear, shame, conclusions made under pressure, and behavioral decisions that continue to run long after the original situation has ended. Turbo-Suslik processes this material together with the connections through which it still affects the present.

As the emotional charge decreases, the past gradually loses its ability to control current behavior. The facts and memories remain, but they stop automatically producing the same pain, anger, anxiety, or defensive reaction. This is not suppression, forgetting, or emotional numbing, and it does not make you colder or somehow less human. The material is not pushed deeper; it is processed until it no longer requires a forced reaction.

In practice, an episode that once knocked you out for hours or days may eventually become an ordinary archived record: you remember what happened, but there is nothing left to replay. The endpoint is complete indifference toward the processed material: it remains available as information but stops demanding a reaction.

This also reduces anxious “living in the future.” Much of the endless forecasting, threat anticipation, and compulsive planning is the past trying to prevent itself from happening again. When the source material stops being reactivated, less attention is dragged into imaginary futures, and it becomes easier to deal with what is actually happening now.

2. Depolarization of the Mind

A polarity is a mentally charged opposition: good–bad, must–must not, success–failure, acceptable–unacceptable, and so on. The problem is not the ability to distinguish one thing from another. The problem is the automatic charge that forces perception and behavior into one of two rigid positions.

The system consistently processes these polarities and the material attached to them. As the charge weakens, perception becomes less distorted by compulsory evaluation, avoidance, and desire. This gives you more room to see the situation as it is and act according to reality rather than according to an old internal command. Polarities and the practical methods for working with them will be covered later.

3. Clearing Mental Noise

Mental noise (later in the book as well called “garbage”) includes fear, guilt, shame, suppression, apathy, helplessness, anxiety, persistent worry, low self-esteem, lack of confidence, and limiting beliefs such as “I am a failure,” “people do not like me,” or “money is evil.” It also includes the thoughts, memories, bodily reactions, and repetitive internal dialogues that maintain these states.

It is not especially important whether this material “resides” in thought, emotion, memory, or the body. In reality, these elements are usually linked. What matters is that the active bundle can be identified, submitted for processing, and gradually removed from the mental circulation that consumes attention and energy every day.

Beliefs are not magically erased. However, once the emotional charge and accumulated evidence feeding a limiting belief are processed, it stops functioning as an unquestionable command. It becomes possible to see alternatives and act differently without having to overpower yourself through affirmations or forced positive thinking.

4. Integration of the Personality

The psyche is not a single, perfectly coordinated system. It is a collection of learned modes, habits, roles, defensive reactions, and behavioral programs that activate in different contexts. A person may function one way at work, another way at home, and a third way in close relationships. This is normal up to a point. The trouble begins when these modes contradict and sabotage one another.

One part wants change, another expects danger, a third preserves the familiar state, and a fourth invents a convincing explanation for why nothing can be done. The result is ambivalence, internal swings, self-sabotage, and the recurring impression that “I understand everything, but still do the opposite.”

Processing reduces the charge and rigid decisions that hold these conflicting modes in place. As the number and intensity of internal conflicts decrease, behavior becomes more coherent: less of the system is pulling in mutually exclusive directions.

Tactics

The tactical side is simpler.

1. Practice

Problems change through work with the material that produces and maintains them. Repeating safe affirmations for years, persuading yourself that everything is fine, or trying to “believe correctly” does not replace processing.

At times, you will have to go elbow-deep into your own mess. That is not the goal of the system; it is a temporary phase. The goal is to process the material, stop carrying it around, and return attention to living, working, creating, and dealing with reality. Digging without processing merely turns self-analysis into another hobby.

2. The Subconscious as a Working Tool

In this book, the word “subconscious” does not refer to a mystical entity or a wise inner being. It is a practical label for automatic processes of the psyche that already operate continuously outside direct conscious control.

Installed programs give these processes a defined task and processing logic. Your conscious role is to notice what is happening, turn a vague “I feel bad” into workable material — “I feel bad because…” — and launch the appropriate protocol. You do not need to consciously track every association and connection involved.

Awareness is still necessary. Automation cannot process a target that you persistently refuse to identify. At the same time, insights that are genuinely useful often surface on their own during the work, without being squeezed out through endless analysis.

3. Automation

The system uses a large number of installed programs to automate routine processing. Several processes can be launched on different layers of a problem and continue in the background, including during sleep.

Automation does not mean that one command removes an entire problem forever. Complex material usually has several connected layers and may need to be approached from different angles. The point is that a large volume of secondary material can be processed without manually walking through every episode and association. What would otherwise take years of repetitive conscious work becomes technically manageable.

Requirements

The system places several requirements on the practitioner.

1. Full Responsibility for Your Own Life

Without this, there is no point in starting. Responsibility does not mean that everything that happened to you was fair, deserved, or caused by you. It means that your decisions, reactions, choices, and further actions are now your responsibility, regardless of who created the original situation.

If control is permanently assigned outside — “they ruined my life,” “circumstances decide everything,” “nothing depends on me” — there is no working point from which change can begin. You remain an object in someone else’s causal chain.

During processing, recurring patterns often become visible: an early episode produced a belief or decision, that decision shaped later choices, and those choices repeatedly recreated similar situations in relationships, work, finances, or self-perception. This causal chain shows what can actually be processed and changed.

Meaningful work begins after responsibility is accepted. If you believe that someone else must change first, you can close the book. The system has nothing to work with.

2. Rejection of Mental Overanalysis

The task is not to construct the perfect explanation of why you feel the way you do. The task is to take the problem and process it.

Instead of searching indefinitely for external answers, you identify the components available now: the state, the trigger, the memories, the beliefs, the bodily reactions, and the internal

objections. Then you submit them for processing. Thoughts such as “why,” “this cannot be solved,” “I already understand everything,” and “how could this possibly work?” are also material, not privileged commentary from an objective observer.

Analysis is useful when it helps you find the target. After that, it must give way to practical work. Otherwise, the mind can spend another ten years producing brilliant explanations for why the same problem is still there.

3. Willingness to Face What Arises

An honest look inside can reveal fear, hostility, envy, shame, helplessness, contradictory desires, and decisions that do not fit your preferred self-image. You do not need to admire or justify this material. You need to recognize it accurately enough to process it.

If entire areas are declared untouchable, the connections running through them remain active. The system can only work with what you are willing to put on the table.

4. Willingness to Process the Required Volume

The psyche is an interconnected network, not a folder containing a few isolated bad files. Deep patterns are supported by many episodes, reactions, beliefs, and compensatory behaviors. Changing them inevitably activates connected material.

The amount of work differs from person to person, but it is usually substantial. No honest method can promise deep restructuring in a weekend. You process as much material as your particular system contains — not as much as would be convenient. That is the actual price of the result.

About The Subconscious

The Subconscious. The Backend.

A core element of the system is what's commonly called the **"subconscious"** — a term that drags in decades of vague metaphysics nobody asked for. Architecturally, what we're actually talking about is the **Backend**: a background process running the psyche, distinct from the slow single-threaded **Frontend** you call "me." From here on, when this book says "subconscious," read it as shorthand for the Backend — we're keeping the legacy term for compatibility with the rest of the book and the site, not because it's the accurate name. Consider it an alias, not a redefinition.

Nobody has ever SSH'd into it directly, dumped its process table, or held it in their hands. What we actually get is stdout: automatic reactions, memories, emotional spikes, behavioral patterns, irrational fears, habits, internal permission denials. Everything past that point is a model — a best-effort reconstruction of a system nobody has read access to.

That doesn't stop people from treating their particular model as ground truth and arguing for it indefinitely. That's their concern, not mine.

The cleanest definition available: ***the Backend – subconscious – is the set of non-conscious processes running the psyche***. Nobody seriously disputes that framing. Where it gets contentious is architecture — what's actually running under the hood, and how it behaves under load.

At one extreme, people model it as a slow, half-deaf child process that needs the same instruction sent a few thousand times before it compiles. At the other, it's treated as some semi-divine superintelligence running flawless code. The truth is probably somewhere in the middle, and I doubt anyone has fully reverse-engineered this architecture. Most detailed theories out there are just elaborate guesses at the source code of a closed system.

From hands-on time in production, the longer I work with the subconscious, the less confident I am that its internals are fully knowable at all. That hasn't stopped me from using it effectively. I stopped needing full understanding as a precondition for use a long time ago. I don't need refrigeration engineering to keep food cold — I need to know which knob does what, and what output to expect when I turn it.

So I run on a working model: one that solves practical tasks, without losing sleep over whether the underlying theory is "correct." Proof's in the build output — either the deploy works or it doesn't.

What follows is the spec.

In this model, the **Backend** (subconscious) isn't mystical or spiritual. It's a high-throughput internal system governing a huge surface area: reactions, attention, habits, emotional patterns, event interpretation, automatic conclusions, even physiological state.

As a working metaphor: think of the subconscious as a personal executor that runs whatever configuration is currently loaded. Like a genie, minus the mysticism. It doesn't grant "wishes" — it executes internal programs, beliefs, and prior commits exactly as they're encoded, no interpretation layer applied.

Put it this way: we're lucky this system exists at all, because it's capable of an absurd amount of throughput. And the actual bottleneck is rarely "reality," "fate," or "a hostile

universe" — it's usually self-imposed rate-limiting, via our own beliefs and permission structures.

The subconscious is a high-capacity biocomputer everyone already has access to and almost nobody logs into. The disagreement was never about whether it exists — it's about the interface for talking to it.

Failure mode 1 — the primitive approach. Treats the subconscious like a slow child process: limited, hard of hearing. Solution attempted: brute-force it with repetition, suggestion, endless affirmation loops, hoping it eventually compiles. Sometimes produces output. Throughput is inconsistent and, frankly, embarrassing for the hardware involved.

Failure mode 2 — the mechanistic approach. Treats the subconscious as a rigid machine: executes literally, zero fault tolerance, zero adaptive logic. Under this model, one malformed line in your script and the whole job supposedly fails silently.

I partially agree with model two — it does behave like a biocomputer. But in practice its flexibility and capacity are considerably higher than either model assumes.

Which model is "right"? Possibly any of them, because the subconscious adapts to whatever model of reality you feed it as configuration. Assume it's slow and limited — you'll get slow and limited, and you'll need heavy repetition to get anywhere. Assume any error is catastrophic — you'll get a system where minor typos trigger full anxiety states.

This isn't mysticism. It's how the psyche actually works: expectations and beliefs configure how internal processes initialize and run. The subconscious becomes a near-perfect executor of your existing model of reality, because your behavior keeps re-triggering the same patterns that confirm it.

I'm not claiming my model is more "true" than the others. Reverse-engineering the exact internals isn't the goal here, and it isn't necessary. The only relevant question is whether the model ships working results. In practice, it does. That's the whole bar for calling it production-ready.

Model spec

- **Faithful executor.** The subconscious runs whatever's currently committed — your beliefs about yourself, other people, money, love, possibility, "the meaning of it all." Committed state is "I can't" / "I don't deserve" / "the world is hostile"? That executes exactly as written. Committed state is the opposite? Same faithful execution, opposite output. It doesn't audit the content — only runs it.
- **Perception filter, not raw sensor.** You never get unfiltered input from reality. Everything passes through a preprocessing layer — beliefs, expectations, fear states, prior pattern matches — before it reaches the Frontend (conscious mind). Data that doesn't match the filter set usually gets dropped silently, not flagged as missing.
- **Storage far beyond working memory.** Probably not a nonstop 8K recording, whatever some people want to believe. But it holds vastly more than the Frontend can query on demand — associations, recurring patterns, emotional tags, somatic markers. Root cause of most "mystery" states lives in here, not in conscious awareness, which usually finds out last.
- **Shapes your subjective reality.** Not mystically — architecturally. It builds your default interpretations and behavioral scripts, which in turn shape what you notice,

choose, and keep walking back into. Feel free to go talk assemblage points, the Eagle's emanations, quantum probability soup, law of attraction — plenty of people enjoy sketching prettier diagrams of a system they haven't opened. I need results, not diagrams.

- **Frequently outperforms the Frontend.** The Frontend (conscious mind) rationalizes, lies to itself, and rewrites its own changelog after the fact to look consistent. The subconscious doesn't bother with PR — it runs reactions, habits, permissions, and denials directly, and does it fast. Constraint: strictly within the boundaries you've already configured for it.
- **Task-assignable.** Define the target clearly enough — what triggers you, where it actually hurts — and you can hand it off as a background job, including overnight. It won't run errands. It will chew through psychological material completely unsupervised.
- **Genuinely high-capacity.** This is real multithreading: multiple processes at once, adjustable load, layer-switching on demand. This is the actual mechanism behind why systematic work beats endless conscious analysis — you're using native hardware instead of bottlenecking everything through one slow single-threaded process.
- **The runtime under every "technique."** Tapping, rituals, visualization, affirmations — all of it is just a trigger interface, a button on the outside of the box. The actual processing still happens one layer down, in the Backend – the subconscious, regardless of which UI skin you put on the request. Given that, the obvious move is to stop paying the ritual tax on every call and start hitting the Backend directly.

The techniques in this book run on exactly this model. And it ships — confirmed in production by more than 100,000 of practitioners, as of 2026.

Now think about what that actually unlocks, in terms of development and actually closing out your open tickets. That's the work ahead. And the output — if you actually run the job — may be more than you bargained for.

Patching vs Refactoring

Over years of practice I've watched dozens of different approaches go by. Some work short-term, some only produce an effect as long as you keep the process running and feeding it input, and some actually change the structure of your reactions at the source. Let me break down the difference as plainly as I can.

Broadly, every approach to working with the subconscious reduces to one of two categories — **patching** and **refactoring**. What follows is my personal classification.

Patching is an attempt to change behavior by installing a new pattern from the outside, at runtime, without touching the underlying code. This covers affirmations, suggestion, hypnosis, subliminal "25th frame" tricks (popular a while back), various self-programming methods, and more. The logic is simple: inject a new behavioral routine and hope it overrides the old one on the next call.

The problem: **patching doesn't remove root causes**. It's a hotfix layered on top of a bug you never actually fixed. Looks stable for a while — right up until the process restarts and pulls the original hardcoded value back from source, because that's still what's compiled in.

This is exactly why affirmations tend to fail long-term. As long as you keep re-injecting the override every single session, you might see an effect. Stop injecting it, and the old routine — backed by years of production traffic, reinforced by real experience — comes right back online. The bug was never removed. It was suppressed for the duration of the patch, sometimes at the cost of a nasty merge conflict running in the background the whole time.

Think about where those original values got **hardcoded** in the first place. They weren't randomly generated. They were written, tested, and reinforced over years — sometimes decades — starting in early childhood. The psyche doesn't retain garbage values for no reason; it retains what kept surviving code review under real load.

Then someone latches onto a reasonable premise — "my internal state drives a lot of my outcomes" — gets motivated, and immediately gets sold a "solution": a free course, a guided audio, a stack of phrases to repeat for two hours a day, and supposedly the whole system updates.

That's where the substitution happens. The premise can be entirely correct. The proposed fix is usually a sleight of hand — instead of doing the actual work, you're handed a config override and told it's a rewrite.

How exactly is that supposed to hold up? You're trying to replace a mature, deeply nested, heavily tested legacy codebase — years of commits — with a handful of new lines pasted in at runtime. Two outcomes are typical: the override's TTL expires after a month or two once the motivation runs dry and everything reverts, or you get a standing merge conflict, old logic and new logic both trying to execute on the same trigger, burning cycles on the collision indefinitely.

Then there's the other category — **refactoring**. Actual work on the source, not a wrapper around the symptom and not "learning to live with the exception." You reproduce the bug, trace it to its root commit, and remove it. Once it's actually fixed, it stops throwing — permanently, not until the next restart.

Fixing a problem isn't about force-pushing through it or worshipping raw willpower. "I just need to be stronger than this" is itself just a cruder form of patching — brute-forcing a retry loop against an error you never diagnosed, which is really self-inflicted denial-of-service.

Fixing a problem also isn't endless staring at the same stack trace, admiring how interesting the crash is, or running elaborate intellectual analysis that never ships a patch. It's a concrete process: reproduce it, isolate the root cause, identify the key dependencies, remove them.

BSFF, EFT, and similar techniques are examples of the refactoring category — approaches where you deliberately reduce reactivity and dig out root causes instead of shipping a nicer-looking wrapper around them. Anything working with root causes and raw internal material, not a polished facade, qualifies. Once the legacy junk is actually removed, you can build new patterns on clean ground. Until then, it's old logic versus new logic fighting for the same execution slot — and the old logic usually wins, because it has years of production data backing it.

I'd also classify hypnosis closer to patching than refactoring. In narrow cases it might be a useful auxiliary tool. But I stand for transparency, working from source, staying sober and clear-eyed about what's actually being changed. Hypnosis doesn't offer that: it's installing a closed-source patch without reviewing the diff, and you often have no idea what got modified or on what basis.

Bottom line: patching doesn't interest me, though I'll grant it's occasionally useful as a stopgap for some people in some situations. **What interests me is freedom, and freedom doesn't happen without actually clearing the junk out of the codebase. Only a real fix at the root produces results that don't quietly revert six months later. Everything in this book, by my own classification, belongs to the refactoring category. We're emptying the trash, not repainting it and calling it clean.**

The Fairy Tale of the Root Commit

A staggering number of people spend years on something called "hunting the root commit." Especially in therapy circles. For some reason a huge number of people believe there's one single commit buried somewhere in the history that, once found and reverted, will auto-resolve every downstream issue currently open.

Some go digging through "inherited" or "prenatal" repos — legacy code supposedly forked from parents or ancestors — hoping that reverting it instantly repaints the whole production system. Others dig into mythical "past life" branches, replaying playback logs of supposedly real historical builds over and over. Others go looking for some kind of "family karma" commit they can revert so everything suddenly ships clean. Meanwhile most of these people are doing zero systematic refactoring on the actual codebase they're running right now, in this life, in this repo. Result's always the same: either nothing changes, or the effect is negligible relative to the effort spent. The subconscious keeps throwing the same exceptions it always did.

The root-commit fairy tale is great business for a lot of people selling you their services. Consultants, self-styled "experts," social-media coaches, assorted gurus — they all carefully maintain the belief that once you find The Commit, everything auto-updates: relationships improve, money flows easier, confidence appears, success arrives, respect follows. You just need to bisect harder and find that one changeset.

So you pay, you find yet another "root commit," you revert it somehow. Short-term, it gets noticeably better. A week or two later you're back in the same failing build.

Only a handful of people ever realize the "one commit" story is a myth. In reality there are thousands upon thousands of root commits. Every one of them is equally "root" — each one just as foundational as the last. Searching among them for The One Button that restructures your whole production system on its own is a category error, not a strategy.

Usually this gets understood by people who've already started doing real, systematic refactoring and have some sense of the actual scale of the repo they're working with.

So: in this system, we reject the entire premise of a single root commit after which "everything just works." That doesn't mean high-impact, defining commits don't exist — of course they do. I'm not even ruling out (though personally skeptical) that some people encounter material that reads like "past-life" branches, "family karma," and so on. Early-childhood-origin bugs are absolutely real, no argument there.

But here's the actual point: even granting all of that exists and has real weight, it doesn't change the fact that a clean build isn't achieved through a single find-and-patch operation. A stable result requires systematic processing of the material — at scale, repeated, done properly, not a targeted one-line fix.

That's exactly the approach Turbo-Suslik runs. We don't chase one mythical root commit, we don't go excavating ancestral repos, and we don't even bother settling whether "past-life" branches are real. All of that is irrelevant in practice. We just process all the material that's actually present in the running system: fears, reactions, beliefs, permission denials, grudges, trauma, internal conflicts, stale decisions still sitting in the config.

We clear out everything that can be cleared — or at least attempt to. We process the historical backlog without romance, without cult, without mysticism. If something surfaces

mid-process that looks like a "past-life" branch — fine, send it through the pipeline and move on. We don't need to prove whether it's real or fabricated. What matters is the output: it stops hooking execution and stops controlling behavior.

This approach — bulk processing, total cleanup, not cherry-picking — is what produces a stable, reasonably fast result. If instead you feel compelled to spend years bisecting for that one unique commit, this system probably isn't for you. Don't waste the cycles.

Legacy code, Skeletons in the Closet, and Freedom

The foundation of any real recovery, as I see it, is deep work with the codebase's history: processing old exceptions, reducing reactivity, and correcting the internal beliefs and decisions that got hardcoded once and have been running as background daemons ever since.

It's remarkable how many self-described "spiritual seekers" skip this entirely. They want to ship new features — "growth," "expansion," lofty architecture discussions — while their actual runtime is riddled with unhandled childhood exceptions, inherited parent-class behaviors, automatic handlers firing on random input, deep-rooted assertions, and unresolved deadlocks. They crash, they rage, they suffer, they notice "growth isn't happening," and then they push harder on the feature branch instead of checking the error logs — which point, overwhelmingly, at their own commit history.

Nearly everyone inherited some broken methods from their parent class. Not because "all parents are buggy code," but because most adults are themselves running unpatched legacy builds — carrying their own bugs, fears, and constraints — and inheritance in this system happens automatically, without a code review. Environments where a child gets genuinely sufficient love, attention, respect, and safety — clean parent classes, basically — are rare. Even without outright abuse, emotional pressure, humiliation, comparison, devaluation, coldness, manipulation, and control are extremely common override patterns.

Kids absorb these methods and later ship them straight into their own production systems — their own families — usually with zero visibility into where the code even came from. "We are all victims of victims," as Louise Hay put it — accurate as a one-liner: most people are running on inherited and copy-pasted logic, not code they actually wrote.

Childhood isn't the only source repo, either. Adolescence, with its broken expectations; school, with its conformity pressure, bullying, hazing; military service, if applicable; early jobs; conflicts; betrayals; relationships; divorces; toxic environments — every one of these commits something. Nothing merges without leaving a diff. It all stays in the system as reactions, expectations, beliefs, and standing defensive routines.

And then a person spends years running a system riddled with unhandled exceptions they don't even know exist. Any random phrase, glance, tone, or situation matches an old pattern — and fires a response wildly disproportionate to the actual input: anywhere from irritation and panic to total collapse of motivation. Sometimes it surfaces as nightmares, irrational fears, panic attacks, chronic anxiety, a standing background sense of helplessness and doom.

Most people naively assume: "you can't roll back the past" and "why touch it, it's ancient history, forgotten." That's self-deception. Sure, a lot of it is evicted from working memory. That doesn't mean the process stopped running. The past keeps executing in you — in your reaction handlers, your perception filters, your unconscious decision trees.

You might genuinely believe you "don't care anymore," that "it's long gone," that "it was childhood, done." All it takes is the right trigger and the old process wakes right up. Not because you're weak — because that's how the runtime works: unprocessed material doesn't get garbage collected, it turns into a background daemon.

Sometimes 20, 30, 40 years go by, and a person hits a vaguely similar input and suddenly gets the exact same shame, fear, or helplessness response as back then. Externally: a

functioning adult. Internally: the same old handler firing. Usually this comes bundled with suppression — the system catches the exception, logs nothing, keeps running. But catching an exception silently isn't the same as fixing what threw it.

"So what," someone says, "let it sit there." It doesn't just sit there. It runs. And it leaks. And even once you've stopped noticing the resource drain, that doesn't mean the leak stopped. People adapt fast to degraded performance, pain included. Doesn't make the pain not pain, doesn't stop the beliefs from still controlling behavior.

Bottom line again: a huge share of your current open tickets trace straight back to unresolved history. Trying to fix present-day symptoms — let alone talk about "growth" — while ignoring the actual commit history is a waste of cycles.

The unresolved past shows up today as:

- **Automatic reactions to input** — unconscious handlers, triggered by old data, firing regardless of current context.
- **Chronic tension and burnout.** Not claiming "all disease comes from stress" — that's an oversimplification. But the psyche has massive load-bearing impact on the body: stress, suppressed emotion, prolonged anxiety, standing internal conflict all wear the hardware down. More unprocessed backlog, heavier the system runs.
- **Low self-esteem and low confidence.** Guilt and shame accumulated over years almost always run deeper in the config than people realize, and they corrupt core values — self-respect, self-belief, capacity to act.
- **Self-sabotage.** Guilt and shame are some of the strongest rate-limiters on any change. On the surface, someone wants a "new build." Internally, a competing process is still running: "not allowed," "don't deserve," "won't succeed," "unsafe to be happy." Attempts at change collapse on their own, usually at the worst possible moment — dissonance between wanting it and it "just not working."
- **Low throughput.** Attention and energy are split across processes: part of you permanently occupied running internal conflicts, replaying the past, suppressing exceptions, maintaining defenses. Where exactly is spare capacity for clarity or momentum supposed to come from?

Aren't we paying a ridiculous ongoing cost to keep this junk running in the background? Isn't it time to deprecate it?

I think the past should be flushed and archived. There's no upside to keeping a live process running on data that's long dead. If you want freedom, the past has to stop being an active process.

Think about why any serious therapy almost always routes back through the past. Because professionals know a stable present build is impossible on top of an unprocessed history. This kind of work tends to run with real intensity, which is why people stay in therapy for years — to sustain the process without dropping out of their own life or overloading the system.

In Turbo-Suslik system, most of this runs under the hood: background jobs at a steady pace, while the person keeps living a normal life on top.

What kind of throughput can a system have while it's running hundreds of thousands of unresolved exceptions in the background? What development is even possible under that load? Usually it turns into permanent symptom-firefighting, no route to an actual clean state.

Isn't it time to stop that? Time to send all of it where it belongs — into the archive? And finally run without that overhead?

Keep this in mind: in Turbo-Suslik, working with the past is priority zero. If you disagree with that premise, don't bother starting. There will be no result. This is one of the load-bearing pillars the whole system stands on.

And don't worry that there's "too much" backlog. With systematic processing it clears out effectively and reasonably fast. Timelines vary per system, and I'm not promising miracles. But the first real shifts usually show up early, and you'll see it directly in how memories fire: less pain, less automaticity, less standing tension.

The end state — where the past genuinely stops hooking execution, grudges lose their charge, traumatic episodes become just archived logs — is achievable. I've run this build. Tens of thousands of others have run it. That means you can too.

Not promising it'll be easy. You'll have to do the work. Some of it will be uncomfortable. But the output is worth the cycles.

Polarities: A Technical Definition

The second load-bearing pillar of this system (the first being work with the past) is work with what we call **polarities**. Keeping the legacy name — it's already load-bearing across the rest of the book and the site — but let's actually define it properly, because "polarity" on its own is vague enough to mean anything.

Polarity, technically: a hardcoded boolean flag pair — two opposing tags the subconscious's classifier automatically stamps onto every incoming object, before it ever reaches conscious review, like: *danger: true/false. good: true/false. mine: true/false. safe: true/false*. Every polarity is a binary attribute the runtime attaches to perception itself, and that attribute then drives downstream branching logic — attraction or aversion, interest or fear, approach or avoidance — automatically, without a review step.

Nothing wrong with the mechanism in principle. You need hot/cold and near/far flags to function at all; a system with no binary classification can't navigate physical reality. The actual problem is scope: the same tagging mechanism gets applied way outside its useful domain, and you have zero manual control over where it fires.

From early childhood we get extremely good at tagging the untaggable. Years of habit convert raw, direct perception into a stream of **pre-labeled objects**. Reality stops being what it is and becomes whatever category it got classified into on ingest.

At some point everyone was capable of more holistic, untagged perception — direct signal, no auto-commentary layer running on top. Growing up degrades that, partly because for decades the classifier keeps stamping everything with the same recurring flag pairs: good/bad, beautiful/ugly, dangerous/safe, free/unfree, and so on, at full volume, on everything.

Every external event gets auto-sorted by the subconscious's classifier into one pole or the other, and that classification directly triggers behavior — attraction to aversion, interest to fear, desire to avoidance. The actual evaluation pipeline is more complex than a flat list of binary tags, obviously, but polarities are doing most of the heavy lifting in it. There's not much room left for a freely chosen response when every input gets pre-chopped into plus/minus by thousands of standing flags before you even get a look at it.

No interest in going deep into polarity theory here — plenty of books already cover that ground. What I actually care about is a practical question: can this classification layer be modified quickly, without mysticism, without years of monastery time.

Turns out: yes.

A number of techniques target polarity neutralization — reducing internal conflict and over-separation between the two tags. Only a handful do it efficiently. One of the best for speed and simplicity is the **GP-4 algorithm**, developed by Živorad Slavinski — existing prior art we build directly on top of. Run it right and you can neutralize almost any polarity pair within minutes: the emotional delta between the two poles drops, the standing bias clears, and perception of both sides gets noticeably more sober and higher-resolution.

Slavinski's original algorithm mostly targets philosophical-tier polarities (true/false). We're equally interested in the grounded, operational ones that directly drive behavior and quality of life.

The techniques in this book is vaguely based on Slavinski's algorithm (the underlying mechanism gets a full technical breakdown in a separate chapter), but move execution into the subconscious layer — the **backend**, running it as a background job instead of a manual, supervised operation. That buys you speed and low overhead. Trade-off: no immediate "wow" spike — resolution runs asynchronously over time and feels smoother, lower-amplitude than doing GP-4 manually, synchronously, one pair at a time.

Using these techniques, you can clear large volumes of polarities on a short timeline. Even at a modest batch size — 20 polarities a day, under an hour of total runtime — that's 600+ resolved per month. That throughput is basically unreachable with any manual, synchronous approach.

At scale, sustained over time, you get direct, first-hand confirmation that most polarities were never properties of reality — they're constructs of the classifier: habitual binary tags for good/bad, right/wrong, safe/dangerous, mine/not-mine. As that tagging layer weakens, perception gets measurably clearer and calmer.

No need to worry about safety or basic function in daily life. You won't start reading cold as hot, or the tenth floor as the first. This isn't touching physical-world orientation — it's touching evaluative tagging inside the "human world" layer. Reactive evaluation and standing internal tension drop substantially; some inputs simply stop triggering a dramatized response, because the automatic lock onto one pole dissolves.

That state can produce plenty of insight on its own. But it's not the end state, just one stage in the pipeline. Downstream of polarity neutralization there's a lot more to build, which gets covered later.

Clearing the Repo (Mind) of Dead Code (Junk)

An average person who has never seriously audited their own mind has zero idea how much is actually sitting in the repo. The core issue: as a species, we're never onboarded on our own runtime. We get trained on reading, writing, driving, earning money — external APIs, all of them — but nobody teaches attention management, reaction handling, fear, internal conflict resolution, or automatic pattern control. Result: from day one of life, everyone's shipping technical debt straight into their own future build.

By adolescence, most people are already carrying a full package of auto-generated modules that assembled themselves without a spec: complexes, grudges, shame, fears, aversions, internal permission denials, other people's beliefs, standing expectations nobody wrote down. And when someone finally runs a serious audit of their own subconscious material, stuff starts surfacing that makes them grab their head: *"where did all this even come from?"*

The truth is plain and unglamorous: the average install accumulates a massive amount of dead code and orphaned processes. This becomes obvious the second anyone runs real deep-dive work — including professional therapy. It's all in there.

There's aggression toward people and the world, sometimes hidden even from the person running it. There's self-rejection, rejection of reality, a standing sense of being defective by design. There are complexes, irrational assumptions — the overwhelming majority of which aren't even original code. They were cloned, unreviewed, from parent repos: parents, environment, school, culture, media, books, random authority figures.

There are whole layers of doubt, disbelief, cynicism, standing resistance. Massive amounts of resentment, unforgiven grudges, judgment, anger, irritation. Millions of auto-applied tags slapped on everything entering the field of attention: dangerous, ugly, beautiful, idiot, cheap, arrogant — an endless stream of classification output.

And a category of its own — fear. Fear of everything, from death to life itself, with entire rows of fear stacked in between: fear of criticism, loneliness, losing love, judgment, shame, losing face, failure, success, intimacy, responsibility. Hundreds of variants, layered on top of each other like nested exception handlers wrapping every possible input.

Enormous volumes of pain and disappointment, plus decisions made once and never consciously remembered, are still executing today — because nobody ever unset the flag. These stale decisions feed thousands of hardcoded return values: from can_earn_money = false to worth = 0 and lovable = false.

What kind of throughput can anyone actually have while running on a system this loaded with conflicts, fears, and self-imposed constraints? And yet almost everyone considers themselves "normal." Pull that config open, though, and "normal" turns out to mean: anxiety, shame, standing internal pressure, chronic tension, suppressed anger, exhaustion, and a never-ending, mostly pointless uptime effort to keep it all from crashing.

Where exactly would well-adjusted systems come from, when from early childhood everyone gets fed hours a day of unfiltered input — information, emotion, scenarios, evaluations they never chose or reviewed? **Garbage in, garbage out** hasn't been repealed. A child absorbs whatever environment it's deployed into. And the environment is, more often than not, running its own unpatched trauma.

Pick a random person, and in the overwhelming majority of cases you'll find someone full of internal contradictions, fear, resentment, complexes, suppression — even if the external build looks polished, well-dressed, smells nice, smiles wide. Most of these bugs are invisible to the person running them, unflagged, because this is just their baseline "normal."

A population made of internally buggy individuals reproduces buggy individuals. Not a conspiracy, not malicious intent — just inertia. The system self-perpetuates: unreviewed parents fork their patterns straight into their children. Those children grow up, ship the same fork downstream, no diff, no review.

To be fair, awareness has been ticking up recently little by little. More parents understand the value of raising kids in an environment of respect, love, acceptance — code review at the source. More people realize the best inheritance you can hand a kid is a psychologically healthy parent repo that isn't propagating its own broken patterns. There's real movement. At population scale, though, it's still a thin trickle against a genuinely enormous legacy install base.

Not exaggerating, not dramatizing. Anyone who actually starts working on themselves and tracing root causes will see roughly the same repo. If there were no way to actually fix any of it, the whole situation would be legitimately hopeless.

It isn't.

Someone actually ready to do the work can deal with their internal gremlins. And yes — this doesn't have to take decades. Timelines vary per system, but with the right approach, real change starts showing up fairly fast. Not effort-free. Not without rough patches. But without mysticism, without cults, without a guru dependency. Self-directed. Self-run. Individually owned.

As you've probably gathered, there's a real backlog here. No need to worry, though, about accidentally deleting working features while you clean out dead code — that you'll somehow lose your capacity for love, kindness, or enjoying anything. Practice shows the opposite: as the junk clears, the healthy modules don't disappear — they get *more* accessible, because fear, conflict, and standing reactivity stop hogging the thread.

In this system, clearing this backlog is a core mechanism — and, importantly, it's actually achievable.

As the repo gets progressively cleaned up, you start understanding how to actually run your own psyche instead of fighting it constantly. To the point where, eventually, it becomes obvious: a significant chunk of your knots and problems was never "you" — just auto-loaded modules from an early build, still running unmaintained, with nobody ever reviewing whether they should still be in production.

That's the point where something shows up that people tend to call "freedom": when the internal system finally idles quietly — not because of a suggestion telling it to, but because the backlog actually got processed.

Integration and Defragmentation of the Personality

The average person's personality is fragmented to a genuinely startling degree. Per some estimates, a typical install is running several thousand distinct **subpersonalities** — discrete processes, in Slavinski's terminology, "aspects." Some got forked off deliberately, as a defensive measure, to isolate a traumatic input so it wouldn't crash the main thread. Some are just recurring automatic handlers. Some are orphaned processes with no clear parent reference left at all. When the average person's **Frontend** prints "I," it's actually printing on behalf of a whole process pool — technically, "WE" would be the accurate output, not "I."

This is exactly why every processing protocol in this system uses "we / us / our" instead of "I / my / me." It's not a stylistic tic — it's a broadcast call. Addressing the request to "I" is unicast: it only reaches whichever process currently owns the active thread. Addressing it to "we" broadcasts the directive to every subscribed process and subpersonality at once, not just the one currently rendering to the Frontend.

Turning "we" into "I" is the actual end state of this whole body of work. There's no standalone routine dedicated to personality integration. No *defrag()* call you run on its own. Integration happens as a side effect, baked directly into the base processing modules — and inherited automatically by nearly every higher-level routine built on top of them. So the defrag runs continuously, in the background, as a byproduct of processing whatever material you're actually working on. Over time the address space gradually consolidates, fewer scattered fragments, more contiguous state — until at some point you notice it's already become a single block. Hard to predict the exact timeline, but expect months, not years.

One more side effect: integration runs in parallel with a rising sense of **"inner emptiness"** — a genuinely interesting side effect in its own right. On one hand, you feel increasingly whole, less fragmented. On the other, a growing sense of empty space. No conflict between the two — this tracks exactly with what defragmentation actually produces on any real system: once fragments consolidate, the freed space shows up as large, contiguous, unallocated blocks. Empty isn't a bug here. It's what successfully reclaimed space looks like.

Since there's no dedicated technique for personality integration and defrag — it just runs as a background process alongside everything else — no reason to go further into the theory here. Keep running the actual work, and you'll experience this directly.

Instructions for the Subconscious

Using so-called **instructions** for the subconscious is the core operating principle of the entire system.

The subconscious stores and maintains a massive amount of data about your life: memories, emotional reactions, somatic markers, habitual interpretations, decisions, permission denials, standing expectations, behavioral patterns. The practical validity of this model has already been confirmed by tens of thousands of TS practitioners: the moment someone starts systematic work, connections and root causes start surfacing that the conscious mind — the **Frontend** — failed to notice or couldn't correlate for years.

The subconscious also, as a rule, has direct access to the actual structure of your problems: why a given reaction fires, where the trigger conditions live, which beliefs are keeping a state alive, which decisions got committed at some point and are still executing today. Not claiming it's some kind of "mystical oracle" or omniscient process. But as a practical tool, its memory footprint and processing throughput vastly outclass the conscious mind, which in its default mode mostly runs commentary, self-justification, and rumination loops.

And if a system like that genuinely exists inside you, it would be a waste of available compute not to point it at your actual problems.

Personally, I think we're fortunate here: the subconscious is usable as an internal processor. With that as a tool, you can resolve a huge range of personal issues — not through endless reasoning, but through actual, executed processing of material.

What Is an "Instruction"

In this **working model**, the subconscious is a high-throughput computational system and executor of internal processes. As a metaphor: a personal genie that runs your internal programs. That's a simplification only — no magic involved, just a mechanism of the psyche.

Technically, an instruction is a function definition. It has three required parts, same as any function signature: what counts as valid **input** (the material), the **procedure** to run on it (what gets done, and by what rule), and what the **output** should look like (the expected end state). Nothing else qualifies as an instruction — a vague wish or an unstructured intention isn't a function, it's an unhandled request with no signature, and the runtime has nothing to execute.

The whole concept rests on one simple premise: if the subconscious does the bulk of the actual processing work, then a task can be formalized and handed a defined algorithm. In practice: state clearly and specifically what should happen to the material, then trigger execution, then observe the diff.

This is the same principle BSFF runs on: you give the subconscious a clearly defined instruction for what to do with the emotional root of a problem. After that, the person just needs to locate the relevant material, invoke the process with a preset trigger, and the subconscious starts executing.

How It Works in TS

TS runs on the same base principle.

First, you give the subconscious an instruction defining what it should do once you flag a problem and fire execution with a specific phrase — in older terminology, a "magic word," though functionally it's just an **invocation call**.

The mechanism is straightforward: read the instruction once, carefully, out loud or silently. That's the registration step — after that, the subconscious compiles it and holds it ready as a callable procedure.

Some people worry they won't remember the instruction, will forget it, or will "read it wrong." This is almost entirely noise. The subconscious doesn't run on strict rote-memorization matching, character for character. It's not a low-level parser demanding an exact token match — it's closer to a semantic interpreter: what it needs registered is the intent and structure of the procedure — what counts as valid input, what operation runs on it, under what rule, toward what output. Paraphrasing the wording doesn't break the contract, as long as the semantics land intact.

If you want, reread the instruction two or three times — not because the subconscious needs it to compile, but because it settles your own nerves. In most cases, though, one clean read is enough to register the procedure and make it callable.

Processing Modules

Once the instruction's registered, point attention at the material and fire the invocation — the subconscious starts executing.

At this layer the system resembles BSFF. The divergence starts past this point: TS runs multiple distinct processor types, each scoped to a specific job.

For example:

- one instruction handles polarity processing only (neutralization, merge resolution);
- another handles beliefs;
- another handles states;
- another handles relationships, reaction patterns, past episodes, and so on.

Each module is its own function definition: a defined input type, a defined procedure, a defined output condition. No shared, one-size-fits-all handler — scoped modules, each doing one job properly.

Macros: The Most Powerful Part

A number of TS instructions are structured as literal **macros**: a single instruction expands into and runs a whole sequence of lower-level calls — batch processing, in the direct sense of the term.

This is the single most powerful piece of the system, because it lets you hand the subconscious a complex, multi-step job — not scoped to one episode or one emotion, but to entire layers of material at once.

For example, you can assign a batch job covering:

- every instance in your life where a specific negative state occurred;
- every belief and decision sustaining that state;
- every downstream consequence that got locked in by that experience;

- every related past episode;
- and, where relevant, other people's stance toward this material and toward you (see the chapter on Circular Processing), including forgiveness and release.

All of it runs automatically, in the background, sometimes overnight. In practice, you're using the subconscious as a compute cluster: not "sit and think it through," but "assign the job, let it run."

Why This Is Needed

This makes it possible to process large volumes of material relatively fast and cheap. What used to take weeks or months of manual digging can now run as a single batch job against a large body of material.

Naturally, there are limits and nuances. Manual, targeted work on a single specific issue can sometimes produce a denser, higher-resolution result than a generalized batch pass. But the gain in speed, volume, and overall simplicity that automation buys you is significant enough that, for me, the tradeoff is obvious.

Anything a batch pass doesn't fully clear can always get refined manually afterward, or folded into a separate macro and run before sleep.

Technical Principles of the System

The system in this book runs on three core principles:

- **Processing modules** — subconscious instructions, each defining an algorithm scoped to a specific type of material.
- **Macros** — composite procedures that batch-process large volumes of material by chaining modules together.
- **Automation** — offloading execution to the subconscious runtime for speed and simplicity.

The tradeoff for that convenience and scale: results are often not immediate. This isn't a strongly-consistent, synchronous system — it's closer to **eventual consistency**: the write lands now, but the state doesn't always converge on the observable output right away. Sometimes it takes days, sometimes weeks.

I don't treat this as a defect. The rate of change is more than sufficient. It's just that in this system, most of the real work doesn't come from an immediate "wow" spike — it comes from actual internal processing that lands and sticks.

Backlashes, a.k.a. “Pendulums”

1. What pendulums Are and Why They Happen

We'll use the term **pendulum** for the side effects of psychological restructuring — in IT we use the term **“rollbacks”** for a system reverting toward a previous stable state after a deploy. Most often it shows up as sharply negative states: sudden anger, irritation, apathy, fatigue, low mood, disbelief, a sense of hopelessness. Physical symptoms are also common — noticeable loss of strength, zero energy, flare-ups of somatic symptoms, digestive issues, general physical discomfort.

Sometimes it looks like the person just shut down: lying there with no energy, no drive, no sense that any of this is going anywhere. First time you hit this, it's genuinely unsettling — which is exactly why you need the mechanism, not just the symptom, before drawing conclusions.

The term "pendulum" entered use in 2006, early in the work with automated processing routines. The phenomenon itself, though, is old news to anyone who's shipped deep psychological change under any methodology. Fast, significant change in perception, belief, and habitual reaction almost never converges in a straight line. This isn't unique to the psyche — it's how iterative optimization generally behaves: **non-monotonic convergence**. The loss function doesn't drop cleanly on every step. After a real forward shift, there's frequently a short-term move backward — old automatic reactions and legacy interpretations get exercised again, transiently, before the new configuration actually settles.

At this stage plenty of people are genuinely confused: expected to feel better, subjectively feel worse. These temporary regressions are a common reason people abort the work entirely. The typical mistake is treating the surfaced state as foreign — as if the technique or practitioner injected it. In reality, what's surfacing is the user's own pre-existing material — legacy code that was already there, just unreachable until the refactor exposed the path. Most of our own mental activity runs below the observable log level even though it's provably running the whole time — detectable through instrumentation even when it never once printed to console.

Physical symptoms accompanying big shifts are entirely normal — same as the resource spike a lot of people get before exams, travel, public speaking, high-stakes decisions: general system load under stress, not something exclusive to this kind of work. pendulums run on the same principle — difference being, the load here comes from an internal restructuring job, not an external event.

Simple way to frame it: you can't compress something and get output that wasn't already in memory. Squeeze it, and what comes out is exactly what was already allocated. Same with a person: over a lifetime the psyche accumulates a large volume of inefficient automatic reactions, beliefs, defensive routines. As long as they're wrapped in habitual avoidance and distraction — a caching layer sitting in front of them — they barely register. The moment real work strips off the top layer of that caching, the next layer of raw material surfaces: stuff that was always there, just never hit a cache miss before.

Effective work temporarily disables the exception-suppression layer — the habitual masks and defensive wrappers. Result: reactions and impulses that are normally caught and silently swallowed start throwing visibly — irritation, aggression, intolerance, rigid judgment calls. Doesn't mean the person "got worse." Means the automatic reactions stopped being

fully hidden from the log. Why isn't there stable acceptance and internal stability already in place, if the corresponding skill was never actually built and tested? Separate discussion. What matters practically here: what do you do with material that just became visible.

To be explicit: this is a model — a working interpretation, not a claim to be the one complete explanation. Other interpretations of side-effect mechanisms exist and hold up fine on their own terms. Deep-diving the root cause of *why* pendulum exists isn't the goal here. The goal is practical: do the work, reduce the internal backlog, keep moving — not get stuck arguing theory.

2. How to Handle pendulums Properly

A few key points up front — makes everything downstream much easier.

First, the "junk" surfacing as intrusive thoughts, emotional spikes, and automatic reactions isn't neutral background noise. It's the actual payload generating most of your subjective problems. It usually runs outside direct awareness and gets treated as self-evident, as "just how things are." Working with material that isn't directly conscious gets its own detailed coverage later in the book. What matters right here: **don't identify with this junk.** It's not your core binary. It's not your kernel. It's a badly configured module that got installed at some point through faulty inheritance, not a fundamental property of who you are.

Second, you can't fight this material. Trying to go to war with your own running process is a losing setup by construction — no winners in that config. The only workable position is acknowledging current state: logging "yes, this is present right now," without immediately trying to kill it, suppress it, or force it out. Acceptance here doesn't mean agreement or approval — it means refusing to declare war. From there, you can process the surfaced material calmly, in a normal working mode, and watch its influence taper off over successive runs — pendulum frequency and amplitude both trending down.

Can pendulum be avoided entirely? No definitive answer. Backlash effects show up in essentially every methodology that does fast, deep change. I ran into the same pattern back when I was working with neurohacking and direct brain-stimulation methods — TMS documentation explicitly references "snapback" effects: temporary reversion to a prior state. Standard recommendation there: reduce session frequency and intensity. Alex Ramonsky's book *I've Changed My Mind*, covering neurohacking at the physiological/pharmacological level, states plainly that fast change is almost always accompanied by pendulum periods — a felt sense that nothing worked, that the change was illusory, plus heavy emotional states. Any experienced therapist will tell you the same: deep pattern work rarely proceeds without temporary deterioration, including physically. Generalize far enough and pendulum shows up anywhere speed and effectiveness intersect.

Intensity and frequency scale with concurrency: run a low-concurrency queue — one or two jobs a day — and you can go through this whole system with almost no visible backlash. Some people are sensitive even at that load; individual variance here is significant. I personally ran an aggressive mode a lot — many concurrent jobs at once — and got a correspondingly bigger regression the next day. Difference was: I understood the mechanism and had already accepted the likely fallout as temporary. That kept it from turning into a crisis or a reason to stop the pipeline.

Pendulum is not proof "the method doesn't work." It's an indicator the system is actually hitting stable, load-bearing structures — a refactor that never touches live code produces

zero regressions. If it's concerning, slow the rate down. Or honestly conclude this working mode doesn't suit you right now. That's also a legitimate call.

3. Types of pendulum and How They Show Up

There's a specific subtype: **pendulum-with-disbelief.** Feels like nothing changed at all, that every problem reverted to its original state, and that everything you've done so far was self-deception. Subjectively very convincing, comes bundled with strong doubt. Its actual function is to stop the pipeline. Any stable change disturbs the system's prior equilibrium, and the system near-reflexively tries to restore its last known-good checkpoint. From the inside this can present as a sudden burst of "objective clarity" — a few days later it usually turns out that clarity was generated entirely from within a temporary state.

Typical states that show up during pendulum — the repertoire is limited and repeats in variations:

- conviction of your own hopelessness, a sense that there's too much internal backlog and it will never clear;
- the thought that the system or technique flatly doesn't work, that the time was wasted;
- pronounced apathy, loss of meaning, a heavy depressive state that can run for several days;
- a sense that every previously processed problem has fully reverted;
- an urge to immediately switch methods, go looking for "something that actually works";
- strong irritation, anger, general emotional tension;
- a conviction that automatic reactions can't be changed at all, in principle;
- physical discomfort, flare-ups of old symptoms, a general sense of falling apart.

Other variants are usually combinations of the above. While you're inside a pendulum, it's very easy to treat its content as ground truth and act on it impulsively. I've made that mistake myself more than once. Track record shows: don't dramatize it, keep the job running, and it clears.

Pronounced pendulums also tend to show up right before a real, visible shift in perception or behavior. The bigger the temporary regression, the more substantial the following change often is — provided you don't kill the process at that exact point. Not a promise or a guarantee — an observed pattern, seen repeatedly across practitioners, not a hard rule.

Stopping the work does often make the pendulum fade over time. Important nuance, though: you can get stuck in an intermediate state where a large chunk of old automatic patterns has already been torn down but no new stable configuration has finished forming yet. Subjectively that reads as prolonged instability, sluggishness, general discomfort. Which is exactly why my own experience says: don't stop the run until you reach a reasonably stable state of emotional and cognitive neutrality — what I informally call **Great Indifference.** Think of it as the system finally reaching steady state, no more oscillation between competing configs.

People who stop mid-run fall into another trap: instead of continuing the actual work, they shift into endless comparative research — chasing "more effective techniques," blaming the

current approach for "not working" or "making things worse." Often, at that exact point, the person is one or two months from a genuinely different state — and stops right there. Common pattern. Not a tragedy, not a mistake — just a choice, and each person owns the consequences of it.

There's also a category — call it **constructive pendulum.** A problem surfaces sharply and clearly, essentially throwing directly into awareness with a full, unswallowed stack trace instead of getting caught somewhere upstream. Genuinely unpleasant in the moment, but it hands you a rare, high-signal opportunity: the material surfaces almost unfiltered. All that's required at that moment is to observe and log it. Early on, awareness is often insufficient and the person just reacts — blaming the environment or circumstances. Once the state settles, though, root causes usually either become obvious or get considerably easier to trace and process.

Zoom out and most people are in some version of these pendulum states almost constantly. Difference is: for most, they're suppressed and smoothed over. Look around — chronic irritation, background anger, low-grade latent aggression have become close to baseline. Suppression doesn't eliminate the material, it just defers it. Eventually the accumulated pressure discharges as automatic outbursts — family conflict, sudden arguments, disproportionate reactions to trivial triggers. Since most people never got visibility into their own mechanisms, the actual root cause of these blowups stays undiagnosed. Then the cycle repeats. Inside actual practice, though, a sharp pendulum usually becomes a genuine observation point and processing opportunity — a kind of gift that hands you access to long-buried material.

4. The Dynamics of pendulum Over Time

With active, regular work — say, an average of two automated jobs a day, plus one or two current issues in active processing — the first two to three months can feel like a continuous stream of pendulum. It can genuinely feel infinite, like the backlog has no bottom. This period is usually experienced subjectively as a run of heavy states with only occasional windows of relief and clarity.

After roughly two to three months, intensity usually starts declining, pauses start appearing, balance gets easier to hold. This is often exactly the window where deeper, more foundational patterns start surfacing. Later, pendulum frequency and intensity both drop further, though the system can still "shake" now and then. Normal. Not a malfunction.

Separate mention for pendulum that shows up at the level of external events — sometimes it feels like circumstances themselves start falling apart: a run of setbacks, disruptions, conflicts, almost like the failure is cascading downstream into every dependent system in your life. Doesn't happen to everyone, doesn't happen every time. Resist the pull toward fatalism here — instead, carefully check which beliefs, fears, contradictory expectations, and standing decisions might have shaped that particular configuration of events.

Usually you'll find a set of internal constraints and conflicts driving the corresponding external response. Complaining and self-blame don't do anything useful here. If a situation got shaped by a specific pattern of thinking and reacting, it makes sense to examine it at that same layer. Owning what's happening is a non-optional part of maturity — no way around it.

5. Practical Principles for Getting Through pendulum

To make pendulum easier to run through — assuming, as this book does, that you're aiming for real, relatively fast change rather than years of symbolic effort, so lock in a few working principles ahead of time.

1. You run on highly selective attention. The conscious mind — the Frontend — stops flagging problems fast once they stop causing pain. Picture it: something used to keep you up at night, looping constantly, generating strong reactions. Then it gets processed and stops firing. A while later, the typical read is: "eh, nothing much really changed." Meanwhile the fact that something dominating your runtime a week ago has fully cleared just doesn't register — your dashboard only shows open incidents, not resolved ones. Automatic patterns can then convincingly "prove" that nothing happened and the whole effort was pointless. That's why I recommend keeping a running log of states and processed material. Rereading entries from a month or two back usually resolves the "is this even working" question fast.

2. Memorize a base rule. Feel bad → pendulum. Feels like nothing got processed → pendulum. Everything feels hopeless → pendulum. Feels like there's too much material and it's unfixable → also pendulum. Not insight, not an "objective read of reality" — a normal nervous-system response, mostly built from learned automatic patterns currently mid-reorganization. Post this rule somewhere visible, literally, so you don't lose track of what you're actually looking at. Even knowing all this, people — myself included — periodically mistake a pendulum for truth and are surprised, a couple of days later, at how seriously they took a temporary state.

A solid tell for pendulum: an abundance of totalizing generalizations. If words like "everything," "never," "always," "nothing" start dominating your internal monologue — "everything's bad," "nothing changed," "this will never end" — you're almost certainly in one.

3. Pendulum gets processed, every time, no exceptions. First step: acknowledge current state without trying to immediately change it. Simple acceptance formulas work here, same family used across various psychological approaches that emphasize recognition over combat. Once even minimal acceptance is in place, write down every thought and sensation tied to the state and process them one at a time with the tools this book already covers. There are also dedicated automated protocols specifically scoped to pendulum processing — they don't always land instantly, but overall they smooth the process out. Related supporting approaches are covered in the accompanying material.

Understanding pendulum's conditional, constructed nature helps a lot. When a state hits, ask: who exactly is suffering right now? Not accepting the automatic default answer "I" — instead trying to actually identify the specific part, the specific mechanism generating this. Sometimes that single act of inspection is enough to sharply weaken the state, sometimes to the point the drama just dissolves, because no single "sufferer" process can actually be located.

Don't ignore the hardware layer either. Working the psyche loads the nervous system and body directly. Regular physical activity, adequate hydration, basic hygiene after intense sessions all reduce overall load. Reasonable nutrition and supplementation can help — without turning it into fanaticism or a separate cult. The point is simple: don't neglect the hardware while you're only running workloads against the software.

Responsibility for the process sits with you. pendulum can genuinely get intense and subjectively heavy — true, I still remember some of mine with a wince. If you know you're not ready to face states like that, it's fine to say so honestly and not continue. Nobody's

forcing this on you. But choosing to run these processes means accepting the possible fallout as part of the deal. Not a call for constant suffering — a willingness to tolerate temporary destabilization if and when it shows up. That's the cost of the result.

And once more: content generated during pendulum can't be trusted. Nothing surfacing in that state is a reliable basis for conclusions about you, the world, or whether the work is effective. It's a temporary re-activation of old patterns — output from a process still mid-transaction. Wait for the state to pass and a more stable read to return before drawing any conclusions.

6. What to Expect Overall

First: pendulum doesn't hit everyone, and not always at severe intensity. Some people run active work and encounter little to no visible backlash. Exact cause of the variance is unclear, though emotional-regulation profile and cognitive-load type likely play a role.

Second: with systematic work, overall internal noise trends down, things get easier to breathe through, and pendulum both softens and spaces out. Over time subtler, more complex pendulum variants can appear, but by then awareness is usually high enough not to mistake them for truth. Sometimes there's even a paradoxical read of "nothing's happening anymore" — simply because the sharp pendulums have stopped showing up.

The early stage of the work also significantly cuts both the likelihood and intensity of later pendulum. Processing basic polarities and base reactions makes the whole system more stable. Which means skimming through stage one shallow can end up costing more later, once heavier automated work kicks in.

And finally: there's no guarantee you'll hit severe pendulum. Fully individual, strongly tied to pace. Higher rate of change, higher risk of temporary destabilization. Everyone weighs their own tolerance for temporary discomfort against how badly they want a more stable end state — and makes their own call.

One purpose of this chapter is filtering out people not ready for this kind of work. Not a judgment, just a statement of fact. If you're ready but hesitant after reading all this, remember words of Akiko Yosano: **“They told me that this road would take me to the ocean of death, and so, halfway there, I turned back. Since then, crooked, dark, roundabout paths have stretched out before me.”**

No pendulum breaks someone who's made a real decision to go the distance. Patience and willpower don't produce miracles — they just let the process actually reach completion.

One last reminder.

Any pendulum is a distortion of perception, generated by activated automatic patterns. Its content is never to be taken as truth.

Philosophical Aspects of the System

As for philosophy. If we try to answer honestly what the core idea of the system is and where it “leads,” the answer will be rather unusual.

Nowhere. It leads nowhere.

The system does not offer a worldview, does not form an ideology, and does not promote values. There is no set of beliefs you are supposed to arrive at, no “correct state,” no image of a final destination. We do not transmit meanings and do not call for any life strategies. There are no affirmations, meditations, workshops, self-development courses, or spiritual practices here. We are not trying to make people happy or change the world.

The system was not originally conceived as something “for others” at all. At some point, a working method was found to reduce internal noise and automatic reactions. Over time, it turned out that this method could be applied not only individually. That’s all. **No mission, no doctrine, no “path to follow.”**

The system does not answer the question of how you should live. It does not create goals and does not offer new meanings. **All it is, is a tool.** A tool with which many practitioners have significantly reduced intrusive thinking, emotional reactivity, and internal tension. If that is relevant to you, you can use the tool. If not, you simply don’t need it.

In this sense, **Turbo-Suslik is fundamentally not a Path.** It has no direction and no endpoint. The only practical goal of the work is to radically reduce the level of constant internal tension and automatic “chewing” of thoughts — and then simply continue living your life.

With a certain degree of approximation, the system’s approach is close to what in different traditions is called the “middle principle”: the absence of both forced attraction and active aversion. This is not about philosophy or belief, but about a practical stance — not suppressing and not cultivating internal reactions, but gradually reducing their automatic nature.

The system does not add anything. It only removes lies.

As the work progresses, the number of automatic reactions decreases, background tension drops, and attention more often rests in the present moment — without dramatization, without constant internal dialogue, and without emotional noise. Phrases like “here and now” have been heavily devalued in recent years, but the phenomenon itself has not gone anywhere: attention becomes more stable, sober, and collected.

At the same time, the ability to analyze the past or plan the future does not disappear. On the contrary, it often becomes more effective. What disappears is something else — compulsive rumination, internal monologues, emotional self-spinning. For some, this comes as a surprise: familiar forms of internal activity — complaints, emotional rumination, constant revisiting of the past or anxious modeling of the future — simply lose energy and fade away.

At certain stages, transitional states are possible. Old beliefs, motivations, and internal supports have already weakened or collapsed, while a more stable state has not yet formed. During such periods, there may be a sense of emptiness, boredom, apathy, or a temporary feeling of meaninglessness. This is not a “failure” and not an endpoint, but a normal stage of restructuring (see the section on pendulums). It does not require interpretation — it simply needs to be passed through.

From a practical point of view, this is not a philosophical crisis and not a loss of meaning, but a temporary state that arises as the automatic layers of the psyche are cleared. After passing through it, life becomes not “more spiritual,” but simpler, more stable, and more functional.

Practice — Phase 1

Phase 1 is scoped for roughly one month of independent work. Its purpose: lay a stable base for everything faster and automated that comes after it. Over this phase, you establish a stable baseline for reducing subjective division of the world into categories, reducing duality and evaluative perception, increasing emotional balance, and continuing the ongoing work with the past.

Phase 1 runs on three tracks. All three are mandatory — none are optional, all need to ship before you move forward:

1. **Polarity processing** — neutralizing polarities from the provided list, plus your own personal set.
2. **Emotional scale calibration** — starting the process toward emotional balance and neutrality.
3. **Basic past clearing** — running your first automated processing routine against historical material.

Optionally, if your system can handle the extra load, you can start gradually processing some of your existing problems manually — purely to build hands-on experience and practical skill. This step isn't required in Phase 1 and can be skipped entirely if you don't want to go deeper into specific problems yet.

The following chapters cover all three mandatory tracks in detail.

How Processors/Handlers Work in the Turbo-Suslik System

Turbo-Suslik has been in production since 2008. Over that time it's been deployed by 100,000+ users against real psychological workloads: emotional reactions, persistent states, automatic behavioral patterns, internal conflicts. Across those years the method's been repeatedly revised, simplified, refactored. Anything that didn't produce stable results across runs got deprecated and removed. Anything that held up reliably across a wide user base got kept and reduced down to a minimal working implementation.

In its current form, this isn't theory, isn't philosophy, isn't "belief in the correct incantation." **It's a defined technical algorithm, and its current shape comes from accumulated production data, not from speculation about how it "should" work. We don't start from assumptions about the ideal design. We start from what, empirically, ships the most stable results with the least conscious-mind interference.**

Which is why everything from here on isn't a set of interpretations — it's working principles and procedures. Your job isn't to agree or argue with the spec. It's to run the protocol exactly as written and evaluate the output directly.

How This Thing Works — A Simple Explanation

The whole system runs on one simple premise: most internal problems don't exist because something is broken about the person. They exist because contradictory automatic processes are running concurrently, writing to the same shared state. One thread pulls one direction, another thread pulls the opposite direction — a straightforward race condition. The conscious mind tries to resolve it manually and usually makes it worse: you get lock contention, stalling, recurring deadlocks, and the familiar symptom of "I fully understand the fix, and the behavior still hasn't changed" — which is exactly what happens when you update the documentation but the running process never reloaded the config. Understanding the correct value and the runtime actually reading it are two different operations.

The system's protocols are built to resolve this conflict directly — no analysis step, no manual reasoning, no attempt at conscious control. You flag what needs processing, invoke the handler, and it runs to completion automatically in the background. You don't need to understand the internals to get correct output — this is a black box by design. You only need to observe the result: a measurable change in reaction and behavior.

The actual procedure: point your attention (or write it down) at the material, hand the subconscious the job by speaking a trigger phrase, and go make tea. **That's the entire call.** No polling, no waiting on the thread, no watching it execute. In practice, it's about as minimal an interface as this kind of system gets.

If you're here for fast, practical change, this explanation is already sufficient. Everything past this point is optional depth — you can skip straight to practice (the chapter on polarities).

How Processors Work: Instructions, Context, and Activation

For anyone who's already worked with an LLM, the cleanest and most accurate analogy is **prompting.** A prompt is a detailed instruction you feed a model to get a specific output. That's essentially what's happening here too: a precise instruction, specifying exactly how the "model" running in your head should behave.

Every operation in this system has four components:

- **material** — whatever you hand the system: a situation, thought, emotion, memory, bodily reaction, internal conflict. Input.
- **protocol** — a detailed spec defining what counts as valid material, what operation runs on it, and under what rules. Function definition.
- **trigger word** — a short invocation call.
- **processor** — an automated background handler that actually executes the already-loaded instruction.

A protocol isn't a magic formula or a set of "correct words." It's a detailed technical spec for a task, handed to the automatic layer of the mind. It defines upfront what material gets accepted, what operation runs on it, which related reactions get factored in, and what the target output state looks like.

The trigger word doesn't carry the full instruction itself. It works like a short run() call: it invokes the full protocol that's already been read and registered.

Where Does a Processor Come From?

A processor isn't handed to you by the method's author, and it isn't spun up from nothing the moment you read the text. Humans already run automatic mechanisms that continuously process experience: restructuring links between events and reactions, deprioritizing some stimuli, reinforcing others, letting unreinforced reactions decay, changing habitual response patterns. Standard runtime behavior, always-on.

In ordinary life, these processes execute on their own — slowly, unevenly, and frequently in a direction the person wouldn't consciously endorse if asked. Turbo-Suslik doesn't invent a new mechanism. It hands the existing mechanism a clearly specified task. That's why a processor is best understood as a **preconfigured execution mode**, not a separate entity — just the name for a process activated by a specific instruction and pointed at specific material.

What Does "Loading" a Processor Mean?

"Loading" is a working term, not literal. You're not downloading a program into your skull, and you're not learning a new skill the normal way. You're handing the automatic system a complete spec: this counts as material, this is the operation to run on it, this is the command that fires execution.

After that, the trigger word becomes a short call against the full instruction. Which is why the protocol only needs to be read once — you're not re-declaring the task on every run. You're not rewriting the function. You're just invoking a procedure that's already compiled and registered.

How Is It Activated?

The sequence is minimal:

1. Notice or write down the material.
2. Hold it in attention just long enough for the system to identify what it's operating on.
3. Say the trigger word.
4. The full, already-loaded protocol gets called automatically.

Everything past that runs in the background — no analysis, no supervision, no attempt to "do it correctly" manually. Consciousness isn't running the processing itself. Its only job is pointing at the material and firing the call. Everything downstream is automatic.

It's functionally equivalent to assigning a job to a system that can operate on its own internal state. You don't babysit every execution step. You just need to specify, clearly enough, what gets processed and under which protocol.

The Role of Intention

Intention isn't strong desire, belief, or emotional buildup. It's the exact moment you actually hand the task off for processing and don't immediately try to grab control back.

You've issued the command. No need to hover over the process asking: "Is it running yet? How exactly does it work? Is it erroring out?" Constantly polling the process just drags you back into manually re-chewing the same material — the same busy-wait loop you were trying to get off of in the first place.

A protocol doesn't override the mind's base constraints, and it doesn't hand you fictional root access to yourself. But within its defined scope, it does initiate automatic processing of whatever material you've handed it.

Result can show up as reduced emotional charge, an end to feeling stuck, a shift in a habitual reaction, more room to actually choose, or a subject just losing its former weight. The event or thought itself doesn't need to disappear. What changes, first, is how your system responds to it.

Why Is the Protocol Text Fixed?

The text is fixed — not because the words carry mystical power, but because it's the complete technical spec for the process. Every part of the instruction defines scope, conditions, boundaries, and handling rules. Decide to "improve" the wording, tighten the text, swap in more elegant phrasing — and you may silently change what the task actually does. Same failure mode as "cleaning up" a config file without checking what depends on the exact keys.

The protocol doesn't need creative adaptation. Run it exactly as written first, see what it does. Modifying an instruction only makes sense once you know precisely which part of the process you're changing, and why. For effective use, nothing needs modification — it already runs correctly out of the box, tested across eighteen-plus years of production use by 100,000+ users.

One more note on wording. As covered earlier, instructions for the subconscious use "we / our" instead of "I / my." Not a typo, not an error, not a reference to some collective outside you. As already established: an "ordinary person" is usually fragmented to some degree — a

unified "I" is rare; what's actually running is a whole set of subconscious parts, subpersonalities, personality aspects, assorted fragments.

Which is why the instruction runs as "we" — addressing the totality of all those parts, which ideally converge into a single running process. This is the broadcast, not unicast: it signals the subconscious to process across all parts, subpersonalities, and aspects at once, including whatever's dissociated or unmerged.

Processing Polarities

The neutralization of polarities is given major importance in this system. It gradually shifts perception from an evaluative, polarized mode toward a less conflict-driven and more stable one. In the long term, this affects not only one's "philosophical perception of life," but also eliminates a number of psychological problems and automatic behavioral patterns that are rooted precisely in polarization and division.

Interestingly, some automatic patterns may disappear on their own — even ones you don't recall explicitly working on. As certain polarities are neutralized, these patterns can simply become irrelevant and drop away without direct intervention. This is a curious process, and the practitioner is encouraged to stay attentive and observe changes in perception and behavior.

How Polarity Neutralization Works and the "Merge" Processor

In the Turbo-Suslik system, polarities are understood not so much as abstract "philosophical opposites," but rather as pairs of competing automatic reactions, evaluations, or states that activate alternately (or sometimes simultaneously) depending on the situation.

Such pairs may look like "want — fear," "should — can't," "love — hate," "confidence — doubt," and so on. At the level of conscious experience, they are perceived as internal conflict; at the level of behavior, they show up as instability, getting stuck, or automatic reactions that are difficult to control.

From a technical standpoint, a polarity is a situation in which **two incompatible response patterns coexist within the system**, each of which has value for that system on its own. This is why such conflicts cannot be resolved by simply suppressing one side: the suppressed pattern continues to exist and periodically resurfaces.

The "Merge" processor is designed for the automatic processing of such conflicting patterns without the involvement of conscious control.

What the Processor Actually Does

After initialization, the processor begins its work sequentially in the background. At the first stage, the system identifies the internal mechanisms that maintain the conflict between the two states. This is not about logical beliefs, but about entrenched associative links and automatic reactions that give each state its subjective significance.

Next, the process of reducing the significance of these reactions is initiated. As a result, both sides of the conflict gradually lose their ability to automatically take control of behavior. This is not suppression and not repression: the states themselves remain accessible, but they stop conflicting with each other.

When the competition between patterns ceases, the system automatically shifts into a more stable mode of operation. What was previously perceived as a contradiction no longer requires choice or effort.

Checking for Internal Resistance

After the conflict weakens, the system performs an additional check. It is aimed at identifying secondary reactions — fears, doubts, or internal objections related to the changes that have occurred. Such reactions arise not because the changes are “wrong,” but because part of the system may perceive the very fact of change as a potential threat to its habitual mode of functioning.

If such reactions are detected, they are processed by the same mechanism: their significance is reduced to a level where they no longer influence behavior or perception.

Integration and Stabilization of the Result

The next stage is the integration of the changes. The system checks whether there are expectations of rollback, fears of the conflict returning, or a perceived need to constantly monitor the result. If such expectations are present, they are also automatically processed, after which the changes become fixed as a new stable state.

As a result, the processing does not require maintenance, reminders, or repeated activation. The processed conflict stops reproducing itself automatically.

Why the Process Takes Time

Unlike techniques aimed at producing vivid subjective states or instant insights, the “Merge” processor operates in the background. This means that processing occurs gradually, as the relevant patterns are activated during the natural unfolding of states.

Therefore, the result may manifest gradually — from minutes to several days — depending on how deeply the conflict was embedded in the system and how frequently it was activated in the past.

The absence of strong sensations is not a sign that nothing is happening. On the contrary, this form of processing is what ensures the stability of the result and its integration into everyday behavior.

Summary

The “Merge” processor does not eliminate “bad” states and does not impose “good” ones. It removes internal competition between automatic reactions, shifting the system into a more coherent and stable mode of operation.

The process is triggered by a key word, after which all further work is carried out automatically and does not require conscious involvement.

The Role of the “Merge” Processor

The “Merge” processor is a foundational element of the Turbo-Suslik system. It is through this processor that the core principles of the protocols become clear, and the correct interaction between conscious awareness and the automatic processes of the subconscious is established. In practice, a person observes that changes can occur without analysis, effort, or conscious control. This is a crucial experience, because all further work in the system is built on the same mechanism.

For this reason, it makes sense to approach work with “Merge” attentively and not skip this stage, even if at first glance it seems secondary or “too simple.” The simplicity here is not a drawback, but a necessary condition for the stable functioning of the entire system.

The “Merge” processor itself is presented in the next chapter. It needs to be read once, either aloud or silently, after which it will be permanently installed.

A list of polarities for processing is also provided in one of the following chapters. In general, work with polarities should continue until the practitioner reaches a state where any previously unseen pair of polarities is perceived as already unified at a glance — that is, when any polarity, even one never worked with before, is already neutralized.

At that point, there will most likely be an understanding that “polarities” do not actually exist — they are simply illusions, mental constructs, as is their division. After that, there is no need to bother with working through different lists of polarities, and the “Merge” processor can be used for more practical tasks, which will be described later in the book.

The “Merge” Processor

You need to read this instruction once — either aloud or silently — from the words “instruction for the subconscious” to the words “end of instruction,” inclusive. After that, the subconscious will accept it as a guide for action. There is no need to reread it, “refresh” it, or repeat it in any way — the subconscious processes it immediately and retains it permanently.

The technical details of how to use it are provided in the next chapter.

For convenience, the instruction itself is presented on the following page.

Instruction for the Subconscious

Start of instruction.

*This instruction is intended for the automatic mechanisms of OUR psyche and nervous system, collectively referred to within the Turbo-Suslik System as OUR subconscious. It installs the **"Merge"** protocol for processing and neutralizing polarities.*

*From this moment onward, all previously used instructions concerning earlier versions of the **"Merge"** protocol are to be considered invalid. All further processing under the **"Merge"** protocol is to be performed only according to this version of the instruction. If any pair of polarities is already being processed under an earlier version of the protocol, all further processing of that pair is to continue under the new version.*

*As soon as WE sequentially direct OUR attention to both polarities that WE want to neutralize and say aloud or think the special keyword **"Merge,"** the combination of directed attention and the keyword launches the processing of the selected pair.*

Here, polarities mean two representations, states, concepts, or alternatives that OUR system perceives as opposed, incompatible, or requiring a mandatory choice between them.

Conflict between polarities may be maintained by:

- *automatic attraction toward one polarity and rejection of the other;*
- *fear of entering one state or losing the other;*
- *emotional and bodily responses;*
- *images, memories, and predictions;*
- *thoughts, beliefs, decisions, and internal rules;*
- *logical and linguistic constructions;*
- *established associative connections;*
- *identification with one of the polarities;*
- *a perceived need to prove, choose, preserve, or avoid one of the sides.*

Full neutralization means a state in which the polarities remain accessible to conscious awareness and retain their semantic and factual differences, but no longer create automatic internal conflict.

After neutralization, OUR system remains capable of distinguishing between the two polarities, understanding their actual properties and consequences, and choosing between them when necessary based on the current situation, available facts, and OUR conscious objectives. At the same time, neither polarity produces a compulsive pull, rejection, fear, internal tension, or an automatic need to take a particular side.

*After the **"Merge"** protocol has been launched, the automatic processing system:*

1. *Sets the selected pair of polarities as its target and activates the representations associated with each of them.*
2. *Identifies all activated components that maintain conflict between the polarities: visual images, emotions, bodily sensations, thoughts, memories, predictions, beliefs, decisions, internal rules, logical and linguistic constructions, and associative connections.*
3. *Separates the semantic and factual content of each polarity from its automatic emotional and evaluative charge.*
4. *Reduces the intensity, automatic salience, and capacity for self-reinforcement of all components that create attraction, rejection, fear, tension, compulsory choice, or identification with one of the polarities.*
5. *Weakens the connections that cause the polarities to mutually activate, exclude, or reinforce one another.*
6. *Integrates both polarities into a common model in which they remain distinguishable states, concepts, or alternatives but no longer exist as two conflicting and mutually exclusive centers of automatic response.*
7. *Identifies secondary responses associated with neutralizing the selected pair: fears, doubts, internal objections, and concerns about losing protection, motivation, values, identity, or the ability to make decisions.*
8. *Processes these secondary responses according to the same principle: preserving any factual information they contain while reducing their automatic charge and their capacity to block neutralization.*
9. *Repeats the processing cycle as new related material becomes activated, until the selected polarities no longer create noticeable automatic conflict.*

*The terminal state of processing is **neutral distinction**: both polarities remain available as information, but neither of them automatically controls OUR response.*

The absence of difference between the polarities means the absence of difference in their automatic emotional salience and their capacity to compulsively control OUR state. It does not mean losing the ability to distinguish their meanings, understand their factual consequences, or recognize actual danger.

*The **"Merge"** protocol does not suppress or displace responses, erase memories, destroy useful knowledge, or make objectively different alternatives functionally interchangeable. It removes automatic conflict and transfers the basis of choice from a charged reaction to the perception of current reality.*

The first processing cycle begins immediately after the keyword is spoken or thought. Further integration of the activated material may continue in the background without conscious analysis or control.

The criterion for completion is not the passage of a specified amount of time or the appearance of any particular bodily response. Completion means being able to direct attention sequentially to both polarities without noticeable internal conflict, compulsive preference, or automatic rejection.

A yawn may occur as a familiar bodily marker of processing initiation. Yawning is not required for the protocol to work and does not, by itself, prove that the polarities have been fully neutralized.

*From this moment onward, the **"Merge"** protocol is applied whenever WE consciously designate both polarities by directing OUR attention to them and say or think the keyword **"Merge."** Processing is performed within the current capacities of OUR psyche and nervous system, without impairing perception, reasoning, conscious decision-making, or necessary protective responses.*

End of instruction.

How to Merge Polarities: Technical Reference

Below: the purely technical side of working with polarities. The procedure for neutralizing a polarity pair via the **Merge** processor is minimal.

Take any pair off the polarity list. Say, *Love — Hate*. Sequentially say — out loud or silently, no functional difference between the two — the token *Love*, then the token *Hate*, then the call *Merge*. Loop *Love — Hate — Merge* until a yawn fires, or up to ten iterations, whichever comes first. Then move to the next pair, or return to whatever else you were doing.

Early in working a specific pair, it can help to not run pure mechanical repetition, but to briefly **log the subjective content of each pole first**. In this example: notice what Love actually resolves to for you — what emotions, thoughts, or somatic markers get associated with the token. Do the same lookup for Hate. Then call Merge.

This isn't a required precondition. The neutralization runs even on a fully mechanical execution path with zero content lookup. But an initial read on what each pole actually contains usually speeds up convergence and improves the overall result. Some polarities — abstract or philosophical ones especially — resist a clean content lookup. That's fine, not an error state. The process runs regardless.

Some practitioners run Merge not to a yawn or fixed iteration count, but until a subjective sense of neutralization shows up. This variant can take anywhere from a few minutes to considerably longer, or produce no perceptible signal at all — depends on the individual. Possible outputs vary: sometimes a direct recognition that both poles were arbitrary mental constructs to begin with; sometimes a sense that the two were functionally the same object the whole time.

To be precise: these subjective effects are neither required nor diagnostic. They don't speed up actual integration and don't affect the end result. In practical terms they're just side output — logging noise, not the signal you're optimizing for. Whether or not you get a felt sense of neutralization mid-run, the process completes regardless — sometimes fast, sometimes with lag. What matters isn't the immediate readout, it's the cumulative result: measurable change in behavior and perception, not internal light shows.

For polarities shaped like Me and [someone/something] — e.g. *Me and my mother* — same procedure. Me and my mother — Merge, repeated to a yawn or up to ten cycles. More detail on running Merge against relationships with specific people or objects gets covered in later chapters.

There's also a simpler path: an automated polarity processor, available after registering on the site. Upload your polarity list, run the batch job, no manual involvement required. Strong recommendation, though: **don't run the automated version until at least two-thirds of the list has already been cleared manually.**

Manual work in Phase 1 matters fundamentally. It builds baseline sensitivity to the process and lays the groundwork everything after it depends on. Skip this stage, especially early, and you typically see reduced effectiveness in every phase downstream.

Processing Problems

The next core processor in the system, in wide use, is the **Process this** module. Scoped to work with any mental material: problems, states, beliefs, emotions, thoughts — basically anything in the psyche generating discomfort or constraining behavior. General-purpose handler, used both in manual work and inside automated protocols.

The logic is minimal. Point attention at some material (from here on, "problem" means any internal object, any mental material flagged for processing), then issue the call: **Process this.** That triggers an automatic processing sequence, as defined in the instruction set. Attention identifies the object; the call invokes the internal procedure that handles it.

Functionally, this processor is close to what a number of other systems run (BSFF in particular), applied the same way. The real difference isn't in the call signature — it's in the underlying mechanism handling the material, which gets a full breakdown next chapter. For now, what matters is scope and base logic.

Process(this) accepts almost any type of mental material: negative emotions, intrusive thoughts, persistent beliefs, past episodes, current emotional states. In some cases it can also take bodily sensations — a headache, say — with one hard caveat: **this system doesn't replace medical care.** If there's a physical issue, route it through medical channels. Don't expect this processor to patch a hardware fault.

In practice, execution looks like this: hold the problem in attention, state the call. E.g.: "This person irritates me — process this." Or: "I consider myself incapable of success — process this." No strict syntax requirement on the wording — the only real constraint is that the material being referenced is unambiguous.

At this point, anyone who's read the standard affirmations literature usually flags a concern: "don't use negative phrasing, the subconscious will take it as a command and reinforce the negative." That concern comes from cross-wiring two different systems. This system doesn't run affirmations at all.

Negative-content phrases here aren't suggestions being fed to the runtime. They function as a **label attached to the object being passed in** — effectively: "here's an internal object with these properties, route it to the handler." Nothing gets reinforced or written to persistent state. No programming happening in the normal sense of the word. Same distinction as: instantiating an Exception object with a descriptive error message doesn't cause the error — it describes state that already exists, and hands it to whatever's listening for it. The message is metadata, not an instruction to execute the failure.

The system's core principle isn't adding new beliefs — it's dismantling and weakening existing ones. This is **refactoring, not programming.** So concerns about "installing negative beliefs" don't actually apply here; there's no write operation happening. If that belief about how the mind works — that naming a problem reinforces it — itself causes doubt or tension, that's perfectly valid material to run through the same module.

Finer detail on handling specific problem types — how to phrase material, what to track, which reactions are actually significant — gets covered in the relevant chapter below. What matters here: **Process this** is a general-purpose call for triggering automatic processing on flagged internal material — no suggestion involved, no interpretation required, no need to get the phrasing "right."

The Operating Principle of the "Process this" Processor

Process this isn't a standalone technique — it's a **control macro**. One short call kicks off several internal processes at once, all targeting the selected mental material. "Mental material" here means anything attention can hold: thoughts, emotions, beliefs, reactions, memories, evaluations, and the automatic patterns wired to them.

Inside the macro, several independent submodules get invoked. Some are ported and adapted from processes used in other systems (Živorad Slavinski's work, in particular — prior art we build directly on), others are proprietary, synthesized across multiple sources. This chapter isn't trying to prove any one model correct. Goal is just giving a working mental model of what actually fires under the hood, so using the processor later doesn't leave you wondering "wait, what's actually happening in there."

For a concrete case, take a hypothetical problem: "I hate Joe." Attention flags this material (via the phrasing itself), Process this fires, and a sequence of automatic subroutines kicks off.

Step 1 — trace the stack. The system checks past experience and traces back through the history for episodes that may have originated or amplified this reaction — root commits, in the earlier terminology. Those episodes get processed to reduce or eliminate their emotional charge, via separate dedicated subroutines.

Step 2 — decompose the monolith. Next it attempts to break the original problem into smaller components. Most persistent reactions aren't monolithic — they're a composite of specific triggers, evaluations, and interpretations bundled together. In this example: "his tone irritates me," "he reminds me of someone else," "I don't accept his social circle," and so on. Automatically decomposing a monolith into its constituent parts lets each piece get processed faster and with more precision — basically a microservices split applied to a single tangled reaction. Even if this decomposition never surfaces to conscious awareness, the system attempts it in the background regardless.

Step 3 — flush stale config. This block targets previously made decisions, conclusions, and beliefs tied to the material — including values set years ago that are no longer referenced in any conscious documentation but are still linked at runtime. Childhood generalizations, protective decisions, unconscious behavioral rules. Say, at some point a value like people_like_that = untrustworthy got hardcoded — long forgotten consciously, still being read by the current reaction every single time. These stale values, plus whatever justification and evaluation got compiled alongside them, get processed here too.

Step 4 — process the beliefs layer. In parallel, beliefs and convictions attached to the problem itself, its components, and the people or objects in play get processed — including beliefs about whether change is even possible, about the "nature" of the problem, about your own available resources. This layer targets the rigid cognitive structures actually load-bearing the problem, not the surface symptom.

Step 5 — integrate the subprocesses. A separate stage targets personality aspects — the functional subpersonalities involved in generating or maintaining the reaction. The processor runs an automated, background-adapted version of aspect work here. These parts aren't suppressed or killed — they're merged into a more coherent overall configuration, same integration mechanism covered in the defragmentation chapter. In practice this shows up as a gradual drop in internal contradiction and conflict.

Step 6 — clear the emotional queue. Next, the system processes emotional charge tied to resentment, non-acceptance, judgment, and grievance — toward others and toward yourself — plus related evaluations of the world, reality, and events generally. Purpose here: drain the emotional charge actually fueling the problem.

Step 7 — integration pass. After that, the material gets processed as a single unit, not as separate parts — a merge commit consolidating every incremental patch from the previous steps into one coherent state. Several subprocesses run here specifically to stabilize the result.

Step 8 — commit and close. Final stage is auxiliary. Once the problematic material's been eliminated or weakened, the system runs a symbolic "closure" step — writing a neutral internal representation that blocks the old configuration from auto-reverting. Doesn't matter exactly how you interpret this mechanically — imagined operation, cognitive marker, a completion flag, whatever model you like. What matters: in practice, this step measurably increases result stability, same reasoning as closing a database transaction properly instead of leaving it open — an uncommitted change sits more exposed to rollback. Theory's secondary here; the observed effect is what counts.

So: **Process this** doesn't trigger one action. It kicks off a full pipeline that dismantles the problem from multiple angles sequentially. Some pieces — handling resistance, or fear of the problem returning, for instance — get pulled out of this macro entirely and run in separate automated protocols instead. That separation's deliberate: it keeps manual work simpler and pushes overall system throughput up.

Bottom line: this isn't a "magic command." It's a standardized way to trigger internal processing on material without manually supervising every single stage — which is exactly what makes the processor general-purpose, usable against almost any kind of mental content.

The “Process this” Processor

You need to read this instruction aloud or silently to yourself only once, from the words “instruction for the subconscious” through the words “end of instruction,” inclusive. After that, the subconscious will accept it as a guide for action. There is no need to reread it, “refresh” it, or anything of the sort.

Instruction for the Subconscious

Start of instruction.

*This instruction is intended for the automatic mechanisms of OUR psyche and nervous system, collectively referred to within the Turbo-Suslik System as OUR subconscious. It installs the **“Process this”** protocol for processing mental material.*

*From this moment onward, all previously used instructions concerning earlier versions of the **“Process this”** protocol are to be considered invalid. All further processing under the **“Process this”** protocol is to be performed only according to this version of the instruction. If any material is already being processed under an earlier version of the protocol, all further processing of that material is to continue under the new version.*

*As soon as WE direct OUR attention to material that WE want to process and say or think the special key phrase **“Process this,”** the combination of directed attention and the key phrase launches the processing of the selected material.*

Here, mental material means any content that can be designated through attention:

- *a thought or sequence of thoughts;*
- *an emotion or emotional state;*
- *a bodily sensation or automatic bodily response;*
- *a memory or individual episode;*
- *a belief, assumption, conclusion, or internal rule;*
- *an evaluation, interpretation, or prediction;*
- *a mental image of a person, object, situation, or OURSELVES;*
- *a desire, fear, impulse, or habitual response;*
- *a problem, internal conflict, or recurring state;*
- *any combination of these components.*

The words WE use to designate the material serve only as a pointer to the processing target. They are not treated as a statement, suggestion, command, affirmation, or new

belief. The target of processing is the material indicated by OUR attention, not the literal wording itself.

Charge means an automatically arising and self-sustaining response that may include visual images, emotions, bodily sensations, and thoughts. These components may activate and reinforce one another, forming a persistent response loop.

Processing an individual component means:

1. *activating it only to the extent necessary for recognition;*
2. *identifying its associated emotional, bodily, visual, and cognitive charge;*
3. *reducing its intensity, automatic salience, and capacity to sustain any further response;*
4. *weakening the connections through which it activates other components of the material;*
5. *preserving any factual information, useful experience, and adaptive functions it contains;*
6. *transferring the component into a neutral state in which it remains available as information but no longer controls OUR response automatically.*

*After the **"Process this"** protocol has been launched, the automatic processing system performs the following sequence of operations.*

1. ***Sets the processing target***

   Sets as its target the material toward which OUR attention is directed at the moment of activation. It takes into account not only the words used but also the images, emotions, bodily sensations, thoughts, memories, meanings, and associative connections activated by them.

2. ***Builds a working representation of the material***

   Identifies the activated components of the material and the connections between them. Determines which elements trigger the response, which maintain it, which are its consequences, and how they reinforce one another.

3. ***Processes related past episodes***

   Through existing associative connections, activates accessible episodes and memories that served as sources, amplifiers, or examples involved in the formation of the selected material.

   In each activated episode, it reduces the charge consisting of visual images, emotions, bodily sensations, and thoughts. The episode remains available as information about the past but loses its capacity to automatically reproduce the previous response in the present.

4. ***Decomposes the material into components***

   Breaks the original material down into individual elements that can be processed separately:

   - *specific triggers;*
   - *people and objects involved;*
   - *individual emotions and bodily responses;*
   - *evaluations and interpretations;*
   - *expectations and predictions;*
   - *automatic thoughts;*
   - *habitual patterns of response;*
   - *associations with other people, episodes, and states.*

   Each identified component is processed separately. The system then checks and weakens the connections that could reassemble these components into the original problematic configuration.

5. ***Processes decisions, conclusions, and internal rules***

   Identifies activated decisions, postulates, conclusions, generalizations, and internal rules associated with the original material and its components.

   This includes decisions made in the past that may no longer be consciously remembered but continue to automatically influence perception and behavior.

   For each such decision, the system identifies and processes:

   - *the circumstances in which it was made;*
   - *its emotional and bodily charge;*
   - *the justifications used to support it;*
   - *evaluations confirming its correctness or necessity;*
   - *conclusions and generalizations built upon it;*
   - *the automatic responses and limitations it continues to maintain.*

   Factual experience and useful information are preserved. What is removed is the automatic compulsion to continue following an old decision regardless of current circumstances.

6. ***Processes beliefs and convictions***

   Identifies activated opinions, beliefs, and convictions concerning:

   - *the material itself and its components;*

- *the people and objects involved;*
- *the causes of the problem and the mechanisms maintaining it;*
- *OURSELVES and OUR capabilities;*
- *whether change is possible or impossible;*
- *acceptable and unacceptable ways of resolving the problem;*
- *the possible consequences of releasing the previous response.*

Reduces the emotional charge, rigidity, and automatic sense of certainty associated with these constructions. After processing, a belief may remain consciously available as a hypothesis or conclusion, but it no longer functions as an unquestionable truth that automatically controls perception and behavior.

7. ***Processes related personality aspects***

   Identifies functional configurations of the psyche, or personality aspects, involved in creating, maintaining, or repeating the selected material, as well as those formed as a consequence of it.

   For each activated aspect, the system:

   - *identifies the function it performs;*
   - *determines the need, objective, or protective task it is attempting to fulfill;*
   - *processes its associated charge, beliefs, decisions, and internal conflicts;*
   - *separates its useful function from the automatic problematic response;*
   - *preserves necessary abilities and protective mechanisms;*
   - *integrates the released functions and resources into the overall personality system.*

   Personality aspects are not suppressed, destroyed, or displaced. Their separate conflict-driven functioning ceases, while the functions and experience they contain are incorporated into a more integrated configuration of the psyche.

8. ***Processes resentment, rejection, and grievances***

   Identifies resentment, unforgiveness, judgment, rejection, grievances, and blame associated with the material and directed toward:

   - *OURSELVES;*
   - *other people and living beings;*

- *particular objects and circumstances;*
- *society, the world, or reality as a whole.*

Processes their emotional, bodily, visual, and cognitive charge. Factual information about what happened is preserved, while the automatic need to maintain resentment, judgment, blame, or internal conflict is removed.

Processing does not require accepting another person's actions as correct, fair, or permissible. It does not remove the need to consciously establish boundaries or respond to actual circumstances.

9. Processes the material as a complete construction

After processing the individual components, activates the original material again as a whole.

Identifies any residual responses and connections preserved at the level of the overall configuration. Reduces their charge and weakens the feedback loops through which the separate components could reassemble and restart the previous automatic response.

10. Stabilizes the result

Sequentially checks the main components, related episodes, decisions, beliefs, personality aspects, and emotional responses activated during processing.

If directing attention to any of them reactivates the original material or any part of it, the relevant component is transferred into an additional processing cycle.

The cycle is repeated as new related material becomes activated, until the original construction can no longer automatically restore itself from its components.

11. Creates complete indifference

After the charge has been removed or reduced to a level at which it can no longer sustain an automatic response, the system creates a neutral terminal representation of the processed material, conventionally called ***complete indifference****.*

Complete indifference is installed in place of the previous charged configuration and serves as the terminal state of processing. The material remains available as information and accessible to memory, but its previous emotional, bodily, and cognitive response is replaced with a neutral one.

Complete indifference does not mean suppression, forgetting, or emotional numbing. It means the absence of automatic salience and the absence of any need to continue the previous response.

As a result, no undefined "empty space" remains that OUR system could interpret as an unfinished state and attempt to refill with the previous or a similar reaction. Processing receives an unambiguous terminal state: the material is preserved, the charge is removed, and no further automatic response is required.

12. Checks the completion condition

Processing is considered complete when:

- *directing attention to the original material no longer produces a noticeable automatic emotional, bodily, or cognitive response;*
- *the related components no longer support or reconstruct one another;*
- *past episodes remain available as memories but no longer restart the previous response loop;*
- *old decisions, beliefs, and internal rules no longer control the response compulsively;*
- *related personality aspects no longer create a separate internal conflict;*
- *the material is perceived either as neutral information or as a current situation to which WE can respond consciously.*

Complete processing does not necessarily mean that a memory, thought, knowledge, or the ability to distinguish actual circumstances disappears. What disappears is not the information, but its capacity to automatically create and maintain the problematic response.

If the material concerns an existing external situation, the protocol changes only OUR automatic internal response. It does not change external facts or remove the need for conscious decisions and actions.

*The **"Process this"** protocol works only with the designated material and with content activated through direct associative connections to it. It does not initiate uncontrolled processing of unrelated areas of the psyche.*

This protocol does not include separate processing of resistance to the processing itself, fears that the problem may return, or circular processing from the positions of other people involved. If such material arises, it is treated as a separate target for the appropriate protocols or automatic processing.

*The first processing cycle begins immediately after the key phrase **"Process this"** is spoken or thought. Further processing of the activated material may continue in the background without conscious analysis or management of each individual stage.*

Repeating the key phrase during initial activation may be used to keep the target in attention, but it is not required for background processing to continue. Conscious control of the internal sequence of operations is not required.

A yawn may occur as a familiar bodily marker of processing initiation. Yawning is not required for the protocol to work and does not, by itself, prove that the material has been fully processed.

Processing is performed without forcibly suppressing or displacing responses, erasing memories, destroying useful knowledge or skills, distorting facts, or disabling necessary protective mechanisms.

If further activation of the material exceeds the current capacity of OUR psyche and nervous system to remain stable, processing intensity is automatically reduced or processing is temporarily paused until safe continuation becomes possible. A conscious decision to stop or pause the work takes priority over automatic continuation.

*From this moment onward, the **"Process this"** protocol is applied whenever WE consciously designate material by directing OUR attention to it and say or think the key phrase **"Process this."***

End of instruction.

Processing Emotional States

The next core element of Phase 1: working the **emotional scale**. At this stage, you run a list of emotions and emotional states through **Process this**. The list is built on the AGFLAP–CAP scale, originally proposed by Lester Levenson. In the source model, emotions get grouped into nine categories: six maladaptive states (Apathy, Grief, Fear, Lust, Anger, Pride — AGFLAP), and three functional ones (Courage, Acceptance, Peace — CAP).

This book runs an expanded, revised version of that scale — this is the build you'll be processing in Phase 1.

The logic here: each emotion on the list isn't treated as a single atomic value — it's a set of distinct variants and shades. Processing these individually reduces the automaticity of their activation against typical internal or external triggers. In parallel, it weakens the fixed response patterns wired to them, and the subpersonality processes maintaining those reactions. End state isn't emotional suppression — it's more freedom relative to the emotion itself: the ability to experience a state without long fixation, or with minimal residual latency after it fires.

As work continues — not just in Phase 1, ongoing — the tendency to get stuck in a given emotional state usually drops significantly. Where an unpleasant state used to run for hours or days, over time its active runtime shrinks. In some cases you get a fast return to neutral or calm shortly after a spike. This isn't repression, isn't suppression — it's a drop in the emotion's automatic hold on attention. Working the scale is step one toward that.

After clearing a significant chunk of the list, some practitioners land in a state of emotional neutrality: no pronounced negativity, no pronounced euphoria either. Can feel unusual, even unsettling, especially the first time you hit it. Important to be clear: **this isn't a loss of emotional capacity, and it isn't a system failure**. Full capacity to experience any emotion stays intact. What changes is purely the degree of **automaticity** and lack of control around when a given state fires.

Working the scale doesn't guarantee these emotions never show up again. The human emotional range is far wider than any fixed list can cover. So a familiar state reappearing later isn't a rollback signal or a sign the work failed. With practice, though, the automatic version of the reaction gets progressively less sustainable, and that shift is usually clearly visible in direct experience. Clearing the scale isn't the end goal — it's the base layer everything else in emotional work builds on.

Technically, execution is minimal. Start from the bottom of the scale — the heaviest, most inert states first. Pull one emotion or state from the list. Say its name (aloud or silently) while briefly registering the actual felt experience behind it, then issue Process this. Loop until a yawn fires, or up to ten repetitions. Then move to the next state, or go back to whatever else you were doing. E.g.: "Hopelessness — process this."

At a steady, moderate pace, the full list clears in under a month. Acceptable speed and overprocessing signs get their own coverage in a later chapter ("Work Plan"). What matters to lock in here: working the emotional scale is a **mandatory** part of Phase 1. Not skippable, not shortenable. The entire scale runs to completion, including the states classified as "positive."

Emotion processing can run in parallel with polarity work — these two pipelines don't conflict or block each other. The one constant is individual pace. The goal isn't speed. It's steady, careful throughput without overload.

Cyber-Zhivo

Cyber-Zhivo is a basic-tier automated processor for the initial clearing pass on your past, run against a predefined algorithm. The underlying algorithm originates from Živorad Slavinski's work, ported here to a fully automatic, background-execution format.

The processor runs on a **fixed 30-day cycle** — a cron job with a hard TTL. Over that window it executes its embedded program and auto-terminates on completion. It runs primarily during sleep, specifically to avoid contending with daytime perception or interfering with anything you're actively running — same reasoning as scheduling heavy batch jobs for off-peak hours instead of competing with live production traffic. You start it once, at the beginning. No need to toggle it on each evening or off each morning — once launched, it runs autonomously for the full cycle with zero supervision required.

At the start of working with the system, the psyche typically holds a huge volume of unsorted, unindexed material — an unstructured data lake, essentially. Because of that, Cyber-Zhivo's output rarely reads as vivid or "explosive." Changes usually surface indirectly. Some people start getting unusual or vivid dreams: characters from the distant past, long-forgotten situations, images from unresolved episodes. Sometimes there's a sense that "something's happening" mid-dream, or that a given fragment of the past has stopped triggering a reaction. Others get zero noticeable dream-side effects at all — also a completely normal outcome. Absence of a visible signal in this one output channel doesn't mean the job isn't running. Dreams are one possible log stream, not the only proof of execution — a silent success is still a success.

Cyber-Zhivo serves two functions. First, it runs an initial clearing pass on the past, offloading part of the accumulated backlog that would otherwise have to get handled later, manually or via heavier automation. Second, it acclimates the psyche and body to running automated protocols at all — a low-load warm-up run before you deploy anything more intensive, so the system isn't hit cold with a heavier job later.

Cyber-Zhivo launches the same way any automation here does. Read the protocol text once — aloud or silently, doesn't matter — then say the activation phrase "Cyber-Zhivo start", looped until a yawn fires or up to 10 repetitions. No yawn? Not an issue — the process is considered started regardless. Track record confirms that's sufficient; no confirmation dialog required.

The Cyber-Zhivo protocol is available free after registering on the site (link at the end of the book). It can go live in the very first days of working with the system — it runs entirely in the background and doesn't conflict with any other Phase 1 process running in parallel.

One clarification on wording used across these automations. Protocol text sometimes references "all episodes in the past and future." "Future" here isn't the literal future — it refers to mental projections, expectations, and scenarios your mind is actively constructing and running, not actual events that haven't happened yet. Think of it as processing a simulated projection state, not live data pulled from an event that hasn't executed. The work targets the model, not the unexecuted real-world future itself.

List of Polarities

Below is a large compiled list of polarities for neutralization (about 430 items). The neutralization procedure is simple — to remind you, you say the first polarity, then the second polarity, then the word **“Merge,”** and repeat this until yawning occurs. For example:

Love — hate — Merge
Love — hate — Merge
Love — hate — Merge
Love — hate — Merge
Love — hate — Merge

Yawn fires here — job's confirmed started, move to the next polarity in the queue.

If 10 iterations run with no yawn, move on anyway — the subconscious will neutralize this pair over time regardless of whether you got a visible confirmation signal. When calling the polarities (again — no need to say them aloud), briefly try to feel each pole, at least on the first pass. Not a hard requirement though — don't get stuck trying to force it.

No re-runs on already-processed polarities. Once you've initiated processing on a given pair via the procedure above, cross it off the list and don't come back to it. Re-triggering neutralization on the same pair multiple times doesn't speed anything up — it's a wasted call that just adds unnecessary load. To be direct about it: you fire the job once per polarity, then stop thinking about it entirely — it resolves in the background over time, no supervision needed.

Rate-limit yourself early. Don't take on more than 10–15 polarities a day at the start. As the system adapts, throughput can scale up significantly, but always watch the body's actual response and don't overprocess. If your system is signaling it's done for the day, it's done — that's not a suggestion, treat it as a hard limit. Volume and pacing get full coverage in the "Work Plan" chapter.

If a given polarity reads as complete nonsense to you — doesn't parse, doesn't mean anything — skip it. Not every entry needs to land.

The list is sorted alphabetically. Full run-through, worst case, clears inside a month without strain.

So, the full list:

ability (to do something) — disability
abnormal — normal
indifference — sympathy
absolute — relative
abstract — concrete
acceptable — unacceptable
acceptance — rejection
acceptance of someone else’s points of view — unacceptance of someone else’s points of view
accessibility — inaccessibility
active — passive
activity — passivity
adequate — inadequate
advancement — retreat

all — nothing
altruism — egoism
always — never
analysis — synthesis
anxiety — peace
anxiety — tranquility
appearance — disappearance
approaching to a goal — moving away from a goal
arrival — departure
attack — counter-attack
attraction — repulsion
attraction to people — rejection of people
balance — compulsion
beauty — ugliness
beginning — end
belief — disbelief
benevolence — hostility
better — worse
big — small
body — mind
body — spirit
boredom — interest
change — stagnation
chaos — order
charisma — dullness
cheerful — tired
cheerfulness — gloominess
clean — dirty
communication — silence
conclusion — beginning
condensation — dissolution
confidence — doubt
confidence about the abundance of the universe — fear that you will receive nothing
confirmation — denial
consciousness — body
consciousness — instincts
consciousness — materiality
consciousness — unconsciousness
contempt — respect
control — freedom
correct — incorrect
creation — destruction
danger — safety
darkness — light
day — night
decent — indecent
dependence — independence
depth — surface
difference — similarity
difference — agreement
dissatisfaction — satisfaction
distrust of people — trust of people
divine — ordinary
dominance — submission

dork — charmer
dream — reality
dynamics — statics
earthly — divine
effort — apathy
effort — rest
emotions — lack of emotions
emotions — mind
emotions — reason
emotions — void
energetic — inactive
enlightened master — moron
enlightenment — ignorance
entering experience — avoiding experience
enthusiasm — apathy
eternity — moment
everybody — nobody
everything depends on me — nothing depends on me
existence — creation
existence — non-existence
expansion — compression
extrovert — introvert
faith — knowledge
far — near
fear — courage
fear — peace
fire — water
folding — unfolding
forever — temporarily
form — formlessness
formation — disappearance
forward — backward
fragrant — malodorous
freedom — depression
freedom — fate/karma
freedom — responsibility
freedom — slavery
freedom — submission
freedom — lack of freedom
fresh — sluggish
front — back
full — empty
fullness — emptiness
fundamentality — shakiness
future — past
generous — mean
genius — madness
gently sloping — steep
god — devil
good — bad
good — evil
gratitude — ingratitude
greatness — smallness
happiness — grief

happiness — sadness
hardness — fragility
heavy — light
hell — heaven
here — now
here — there
high — low
honest man — swindler
hopelessness — hope
horizontal — vertical
hot — cold
hunger — satiety
I am bad — I am good
I am not good at anything — I am good at everything
I am the source — I am not the source
I am unique — I am like everybody else
I exist — I don’t exist
I must — I want
I must be right — I am not right
I want to be aware — I don’t want to be aware
I want to communicate with people — I don’t want to communicate with people
I-don’t-care-ness — enthusiasm
ignorant people — wise people
illusion — reality
indifference — involvement
infinitely small — infinitely large
initiative — inactive
inner world — outer world
inside — outside
intellect — emotion
intellect — instincts
interest — boredom
interest — indifference
intuition — consciousness
irritation — acceptance
joy — sorrow
joyous — angry
knowledge — action
knowledge — ignorance
lack of self-confidence — self-confidence
laziness — willingness to act
lechery — chastity
left — right
left hemisphere — right hemisphere
life — death
light — darkness
lightness — heaviness
limited consciousness — unlimited consciousness
limitedness — limitlessness
literacy — illiteracy
localization — non-localization
logic — creativity
logic — emotions
logical thinking — creative thinking

long — short
love — aggressiveness
love — fear
love — freedom
love — hatred
love — loneliness
love — pain
love — power
love — refusal
love of people — hatred of people
luck — misfortune
lucky — unlucky
male — female
man — woman
material — emptiness
material — immaterial
material — spiritual
material world — spiritual world
material world — subtle world
matter — energy
matter — void
me — abundance of the universe
me — Buddha
me — complete enlightenment
me — eternity
me — everyone
me — extraterrestrial intelligence
me — galaxy
me — intuition
me — life
me — movement
me — nothing
me — others
me — people
me — someone else
me — subtle worlds
me — the whole world
me — unity
me — universe
me — void
me — world
merciful — merciless
minus — plus
misfortune — good fortune
monotony — diversity
motion — immobility
motion — rest
motion — stopping
moving — staying
multitude — uniqueness
must do — don’t want to do
must — must not
my — somebody else’s
my father — my mother

my wishes — somebody else's wishes
mysticism — narrow-mindedness
nature — civilization
near — remote
negative — positive
negative emotions — positive emotions
noble — ignoble
nobleness — baseness
noise/sounds — silence
now — later
observer — observed
old — young
optimism — pessimism
others are right — I am right
panic — tranquillity
part — whole
passion — spirituality
past — future
path — goal
peace — aggression
peace — disorders
peace — fear
peace — power
people — animals
perfection — imperfection
permanence — transience
permanent — temporary
permission — prohibition
point — space
point — three-dimensional object
poor — wealthy
positive — negative
positive element — negative element
positive polarity — negative polarity
possible — impossible
poverty — wealth
power — helplessness
power — impotence
power of the night — power of the day
presence of thoughts — absence of thoughts
present — future
present — past
present moment — eternity
progress — degradation
proof — disproof
prudent — imprudent
quickly — slowly
reason — consequence
reasonable — hasty
relationship — loneliness
relaxation — stress
resoluteness to do something — postponement
responsibility — irresponsibility
right — wrong

saint — sinner
salted — unsalted
sane — insane
satisfaction — displeasure
scream — silence
security — insecurity
sensible — thoughtless
sentimentalism — heartlessness
separateness — integrity
seriousness — light-mindedness
severity — mildness
shame of failure — delight of victory
shining — dim
short life — eternity
should save one’s face — shouldn’t save one’s face
silently — loudly
silly — enlightened
sky — earth
small — big
something — nothing
something has to be done — nothing has to be done
sorrow — joy
stability — changes
stability — shock
standing — lying
straight — curved
strength — weakness
structured — unstructured
subject — object
subtle humor — vulgar humor
success — defeat
success — disappointment
successful — unsuccessful
sun — moon
sweet — not sweet
symmetric — asymmetric
sympathy — antipathy
talent — lack of talent
teacher — pupil
teaching — knowledge
the one who knows — the thing which is known
the world is bad — the world is good
the world is dangerous — the world is safe
the world is unfair — the world is fair
theory — practice
this world — other world
thrifty — thriftless
to agree — to disagree
to be — not to be
to be always conscious of oneself — to be never conscious of oneself
to be aware of the essence — to see the surface
to be flexible — to be fixated on
to be united with one’s roots — to be separated with them
to begin — to stop

to believe everything will be OK — to disbelieve everything will be OK
to bless — to curse
to forget — to remember
to get results from “Turbo-Suslik” — to get nothing from “Turbo-Suslik”
to give — to get
to have a higher purpose — to live without a purpose
to have a possibility to choose — to have no choice
to have results — to have no results
to have results of processing — to have no results of processing
to have time — to have no time
to know — not to know
to know everything — to know nothing
to know one’s predestination — to not know one’s predestination
to give a gift — to take a gift
to possess everything — to possess nothing
to remember — to forget
to see — to understand
to stay — to leave
to survive — to die
to survive — to give up
to take — to give
to take responsibility — to reject responsibility
to throw — to pick up
to turn pale — to turn red
to win — to lose
to withstand — to give up
tolerance — intolerance
top — bottom
topical — non-topical
tragedy — comedy
true memory — illusive memory
truth — hallucinations
truth — lie
uncertainty — certainty
uncertainty about the future — certainty about the future
understanding — misunderstanding
unity — duality
unity — separation
unity with others — isolation
unpleasant — pleasant
unwillingness to change — willingness to change
unwillingness to live — enjoyment of life
unwillingness to live — lust for life
up — down
usual people — enlightened people
victory — defeat
victory — loss
virtual world — real world
visibility — invisibility
void — the whole world
vulnerability — invulnerability
war — peace
wealth — poverty
weekdays — holiday

white — black
wish to be approved — wish to approve
wish to be lonely — wish to be with everyone
wish to be loved — wish to love
wish to be with people — unwillingness to be with people
wish to control — wish to release control
wish to control the others — wish to be controlled by the others
wish to debate — unwillingness to debate
wish to have love — wish to give love
wish to live — wish to die
wish to move — fear to move
wish to move — unwillingness to move
wish to win approval — wish to express approval
wish to work — unwillingness to work
wise man — stupid jerk

Emotional Clusters

Repeating this once more since it matters. Processing the emotional states in these clusters is part of Phase 1. Refresher on the procedure: pull an emotional state from the list, say its name (aloud or silently), while attempting to at least minimally register the actual felt state, then call Process this, loop until a yawn fires, then move to the next state.

Example — processing Depressed. Say (aloud or internally) "Depressed", briefly registering the state, then "Process this". Loop to yawn:

Depressed — process this
Depressed — process this
Depressed — process this
Depressed — process this
Depressed — process this

Yawn fires here — move to the next state, or resume whatever else you were doing. If 10 iterations run with no yawn, move on regardless — the material processes either way, confirmed signal or not.

No re-runs on already-processed states. Once you've initiated processing on a given state, cross it off and don't come back to it. Running the same state's job multiple times doesn't speed anything up — it just costs you a headache and adds unnecessary load for zero gain. To be blunt about it: fire the call once per state, then stop thinking about it — it resolves in the background over time.

Rate-limit at the start. Don't take on more than 10–15 states at the very beginning. As the system adapts, throughput can scale up — but keep watching your own reaction and your body's actual response as the ground truth here, not your ambition.

The state list is built on **Lester Levenson's Sedona Method** emotional scale (USA). Start from the lowest-tone state (Apathy) and work upward toward higher-tone states as you progress.

Cluster “Apathy”

detachment
defeated
inactive
inconspicuousness
stuck
loser
what’s the point of trying?
boredom
lack of a sense of humor
helplessness
exhaustion
suppression
nothing depends on me
giving up
resigned
hopelessness
loneliness
I don’t care
lost
disorganized
burned out
defeated
demoralized
shock
powerlessness
hardened
nothing works
laziness
no results
depressed
stunned
bad mood

what's the point?
despairing
unneeded
failure
cut off
coldness
lack of self-care
too late
indecision
worthlessness
submissiveness
inability to concentrate
I can't win
disappointment
denial
I expect nothing from life
efforts are meaningless
absent-mindedness
I can't
uselessness
forgotten
sluggish
joylessness
hopeless
depression
emotional coldness
discouragement
disorientation
inattention
low self-esteem
indifference
dead tired
numbness
fatigue
loss
compliance
suspiciousness
insensitivity
apathy
forgetfulness

Cluster "Grief"

misunderstood
tearful
betrayed
emotional pain
shame
abandoned
ignored
deceived
unhappy
why me?
sadness
suffering
unloved
helplessness
unable to find a place for oneself
heartache
heartbroken
lost
resentful
sorrow
good for nothing
suffering
despondency
pity
vulnerable
loss
melancholy
offended
embarrassed
lonely
it's not fair
unwanted
feeling guilty
grief
remorse
rejected
sensitive
forgotten
longing
inadequate
yearning intensely
inconsolable
constrained
in despair
forgotten
accused
rejection
nostalgic
no one cares about me
regretful
pity me
nobody loves me
guilt
disappointed

discouraged
I am miserable

Cluster “Fear”

skepticism
paranoia
frightened
instability
terrified
shocked
threat
tension
suspiciousness
stage fright
anxiety
indecision
cornered
vulnerability
worry
fear of the stage
timid
nausea
doubt
insecurity
fearfulness
escape from reality
knees shaking
cowardice
confused
caution
prudence
concern
superstitious
shyness
cowardliness
this is a bad omen
slowness
lack of confidence
fright
evasiveness
defensive
premonitions
panic
withdrawn
paralyzed
distrust
nervous
hysterical
hesitant
bewilderment
unease

Cluster “Desire”

ruthlessness
dishonesty
never enough
expectation
rudeness
stinginess
frenzied
wandering
impatience
excess
urge to accumulate
obsession
I must have it
forgetfulness
impulsiveness
promiscuity
predatory
manipulativeness
can’t wait
desire
malice
possessive
pressure
heartlessness
envy
hunger
recklessness
need
controllability
anticipation
I want
exploitative
compulsive craving
thirst
insatiability
obsession

lustfulness	can’t get enough
greed	never satisfied
selfishness	frustration
neediness	intense desire

Cluster “Anger”

harshness	rigidity
disgust	malicious
ferocity	futility
quarrelsome	ruthlessness
defiant	demandingness
bitter	rebelliousness
sarcasm	rude
resistance	evil
rage	destructive
causticness	argumentativeness
frustration	aggressiveness
gloomy	self-importance
angry	cruelty
jealousy	dissatisfied with everything
willfulness	withdrawn
sullenness	furious
irritation	madness
gloating	offended
hatred	anger
annoyance	resentment
agitation	boiling with anger
recklessness	stubbornness
belligerence	hostile
cursing	hard
explosive	strictness
vindictiveness	defiance
indignant	outrage
intolerance	petty
touchiness	stubbornness

Cluster “Arrogance”

indifference	arrogance
narrow-minded	harshness
snobbery	hypocritical
giftedness	I am above reproach
insensitivity	I am special
dominance	superficial
egocentrism	patronizing
contemptuous	self-confidence
bias	prudishness
exclusivity	detached

intolerant
limitedness
contempt
know-it-all
false dignity
withdrawn
gloating
luck
superiority
false modesty
vanity
audacity
conceit
self-love
false virtue
boastfulness
haughtiness
unforgiving
show-off
inflexibility
stoic
overconfidence
categorical
inflexibility
bored
self-satisfaction
never wrong
judgment
criticalness
righteousness
the smartest
spoiled
stubbornness
picky

Next come the clusters of “positive emotional states,” which should also be processed in order to achieve emotional balance.

Cluster “Courage”

strength
competence
optimism
adventurousness
insightfulness
enterprising
supportive
loving
attentive
independence
courage
motivation
resilience
compassion
honesty
sense of humor
determination
self-sufficiency
initiative
sensitivity
calmness
liveliness
creativity
not afraid to resist
activity
curiosity
inventiveness
purposefulness
invincible
dynamic
generosity
prudence
boldness
confidence
sound-minded
independence
energetic
responsiveness
loyal
enthusiastic
nobility
clarity
thoroughness
openness
happiness
active
not afraid to dream
bravery
awareness
flexibility
I can
diligence

positivity
tireless
resourcefulness
goal-oriented
quickness
cheerfulness
revitalization

Cluster “Acceptance”

empathy
understanding
attentiveness to others
nothing to change
well-being
easygoing
receptivity
I have
pleasure
compassion
admiration
beauty
balance
tenderness
mercy
friendliness
gentle
belonging
bright
abundance
cheerful
meek
equanimity
wonder
generosity
sensitive
sincerity
openness
everything is fine
inspiration
harmony
warm
harmonious
subtle sensitivity
passionate
safety
miracle
spontaneity
peacefulness
beautiful
delight
wholeness
enjoyment
loving
confident
joy

Cluster “Peace”

light
free
eternal
self-realized
inner peace
receptive
focus
space
balance
I am
existence
completion
perfection
boundless
perfect
purity
innocence
feasibility
order
timeless
peacefulness
ageless
unity
serenity
wholeness
calmness

Work Plan

Over the next month — assuming you've made an actual decision to run this system rather than "take a quick look" — three baseline tasks need to ship:

1. Neutralize every polarity from the list covered in a previous chapter.
2. Process every emotional state on the emotional scale.
3. Launch **Cyber-Zhivo** and let it run autonomously for the full month.

Technical mechanics for all three are already covered above, no point repeating them. What matters here is different: **throughput.** How much material can you process daily and still get a clean result without overloading the system?

No universal answer exists. People vary enormously in sensitivity, current nervous-system state, and baseline internal tension. For one person, ten polarities a day is a light warm-up. For another, that's already a meaningful load. Any specific number here is a starting default, not a spec requirement.

For a sense of scale: early in my own work I ran about 20 polarities a day off the list, plus 20 –30 emotional states. During the day I also fired **Process this** in parallel on whatever emotional material surfaced. Total daily throughput ended up fairly high. By evening there was usually a distinct fatigue — not physical, systemic: the same drained feeling you get after a genuinely demanding cognitive session, not a workout.

Conclusion is simple: overprocessing is a real failure mode, not a theoretical one.

So at the start, stick to a moderate default: **10–15 polarities and 10–15 emotional states per day.** After a few days, check the body's actual response carefully. Handling the load cleanly? Scale up gradually. Getting pronounced fatigue, irritability, apathy, or a general sense of being overloaded? That's a direct signal to throttle down.

Core principle: calibrate off bodily and emotional signal, not the mind's urge to "just push through and finish faster." The body, unlike the conscious mind, is usually an accurate rate-limiter — it tells you when load is sufficient, and it's rarely wrong about it. If it reads as "enough for today," it is enough. Sometimes a full rest day — zero processing — is the right call.

As the nervous system adapts, volume can scale up with no negative side effects. Even at later stages, though, the base rule holds: **never work through obvious exhaustion.** That's not a soft guideline, it's a hard constraint on the whole pipeline.

Within month one, you can also expand Release work by adding personal polarities specific to your own life. Neutralizing pairs like *me — [specific person]* often has a real positive effect on relationship quality and general perception of the world. That said, this is optional. At Phase 1, fully clearing the base polarity list is sufficient on its own.

Work on specific personal problems is best deferred to Phase 2, for two reasons. First: automated protocols make this work substantially easier and deeper, and they aren't available yet at this stage. Second: running intensive polarity work, emotion work, and personal-problem work concurrently can be excessive load for a system that hasn't adapted yet.

Phase 1 is the foundation layer. The task here is narrow and concrete: clear the polarities and the emotional clusters, at a sustainable pace, no heroics. Everything else can wait its turn in the queue.

What Will You Get as a Result?

There's not much point stating anything definitive about the output you'll get after a month of Phase 1. People vary enormously: nervous-system baseline, volume and structure of accumulated backlog, adaptation speed, actual readiness for change. So the full range of outcomes is on the table — anywhere from "barely register a diff" to "this materially changed my whole state." Both are within normal operating parameters.

Which is why I won't describe how it "should" go, or what to expect. Any such spec easily turns into a source of manufactured expectation — and disappointment the moment observed behavior doesn't match the doc. This system deliberately avoids setting expectations as a config value, and it avoids benchmarking your run against someone else's output.

The only valid metric is your own direct observation of your own state, reactions, behavior. If something changes, you'll see the diff yourself. If change runs slow and quiet, that's also within spec. In recent years I genuinely can't recall a case where conscientious Phase 1 work produced zero measurable change — but that's an observation from the field, not a guarantee. There's no SLA here.

Whatever result you get is the correct result for your specific system. So instead of speculating about expected output ahead of time, the actual move is just running the job — if that's the decision you've made. But only start execution after reading the full book, not just Phase 1. I have zero interest in you entering this process without understanding the overall architecture and the range of possible side effects first.

Practice — Phase 2

Processing the Past

Working with the past in Phase 2 is one of the priority tracks at this stage — priority at least until the primary emotional charge tied to past events has been substantially reduced.

This isn't about erasing the past or "forgetting" it. Events don't get deleted — they gradually lose the ability to auto-trigger an emotional reaction. In practice: the past starts reading as neutral archival information — "it happened, logged, done" — with no standing tension, no replay loop, no residual charge on read.

With regular, consistent work, that state is largely reachable in about two months, and by three to four months the past typically stops materially affecting your current runtime. Timeline's individual, obviously — depends on work intensity, tooling used, and overall nervous-system baseline.

The past gets touched, in some form, by nearly every automated protocol regardless. Even so, at this stage, start deliberately with the most significant episodes and key people first — highest-severity tickets, essentially. Some of this runs manually, some via automation. Either way, the entry point is the same: **compile the list.**

Compiling Lists

Work on the past starts with logging charged episodes from your life.

Pick any medium that works — phone notes, a file on your computer, a physical notebook — and start writing down events as they surface. No analysis, no scoring "how important is this," no attempt to build a priority hierarchy up front. Just log whatever comes up.

At the start, only a handful of episodes may surface — that's expected. As you keep working, memory access gradually warms up, like a cold cache filling in: more of the past becomes reachable over time. A lot of records are inaccessible early precisely because of how painful they are — flagged, essentially, with restricted read access — but they become retrievable later. Don't force a full dump on the first pass. Pull what surfaces naturally.

Write short — tags, not full narratives. E.g.:

- "father hit me on my birthday, 7th grade"
- "mother slapped me in front of everyone at a meeting"
- "kids took my money when I was 8"
- "embarrassing wardrobe incident during PE, second year"

The subconscious doesn't need a full writeup. All the detail is already stored on its end — you only need a **tag** per episode, a short identifier you'll use later to invoke processing on it.

Where possible, group episodes by person. If several events tie to one individual, log them together — same reasoning as grouping tickets by component. This matters because after the individual episodes get processed, there's a separate pass targeting that person's overall image specifically, to strip whatever residual charge is still attached to them as an entity, not just to the individual incidents.

Some Context

Anyone familiar with "recapitulation" from other past-work traditions will recognize the list-building approach here. TS runs similar logic — minus the ritual, minus "retrieving energy," minus any metaphysics layered on top. We're not re-living the past. We're tagging the material and stripping the charge off it.

What Exactly to Log

Your job: log as many charged episodes as you can find, prioritizing:

- anything involving parents;
- anything painful;
- anything that produces discomfort on recall.

"Charge" here means any residual "not okay" signal on read: tension, shame, anger, sadness, irritation. If an episode replays with zero reaction, its charge is probably already near-zero — low priority, not worth chasing.

Important: no need to re-live episodes, sit in the feeling, or deliberately try to "feel" anything. Log the fact, set it aside for the actual processing pass later.

On average, practitioners log 100 to 300 episodes. The real total's higher — minor, low-severity ones often don't surface on manual recall, which is fine; those get swept up later by automation anyway.

Best to start with the heaviest, most painful entries — clear P0s first. Processing those produces noticeable relief fast and meaningfully boosts motivation to keep going. Over time the list starts filling with progressively smaller items — "a bus conductor was rude," "didn't get my change back" — down to the point where it's obvious the backlog's basically drained. That's a normal, expected signal that manual work on the past is done and can hand off entirely to automation.

Sometimes, months later, deep foundational incidents — usually from childhood — surface that weren't accessible before. Read access on those records finally cleared. When that happens, go back and run a targeted manual pass on them specifically.

Episodes can get processed as they surface or pulled from the list later — no single correct method here, pick whichever workflow fits you better.

If Nothing Comes to Mind

Sometimes people hit a state of "genuinely nothing surfaces." Can cause confusion, and a false read of "there's nothing here to process."

Theoretically, this is possible as a result of already-achieved acceptance and internal integration. But in that case life is usually already running fine, with no standing background tension. If that's actually your situation — reasonable question: why are you even running this system? Go live your life.

In practice, though, the cause is usually the opposite. No memories surfacing, and life is nowhere close to calm and clear? That means the past was heavy enough that the psyche straight-up denied read access to it. Suppression and sealing off memory is a standard protective mechanism — access control working as designed, not a bug.

For this group, the work runs harder, especially early on, and rollback swings tend to run stronger. The standard-pace approach may not be the right config here. In that case, throttle immediately down to minimum: run automations only, neutralize 2–3 polarities a day max, and only touch emotion work when there's a clear signal from the system that it's ready. Forcing throughput here gets met with heavy internal resistance and multiplies the total timeline. This isn't "it's broken" — it's the system flagging that the load you assigned exceeds what your current state can actually handle.

Usually, past the initial freeze, the process does start, and the list starts filling in fairly easily from there.

“Chpok” and “Hoppo.” Incident Processing

By this point, Phase 1 should be fully complete. Before starting active work on the past, go register on the site (link at the end of the book) and grab two free processors: **Chpok** and **Hoppo** — free registration required.

The "Chpok" Processor

Chpok is the simplest, fastest processor in the whole system. Narrow job: strip the emotional charge off specified mental material.

"Charge," here, means a bundled payload of:

- visual imagery,
- emotion,
- somatic sensation,
- thought,

all of which auto-fire together whenever attention references a given episode, person, or state.

Chpok runs fast precisely because it skips deep analysis entirely. It doesn't touch root decisions, beliefs, personality aspects, or downstream constructs built on top of the charge. It just strips the charge off the cached object — a cache eviction, not a recompute — and does it efficiently.

This processor is optimized for speed and simplicity. Not depth.

The "Hoppo" Processor

Hoppo is a specialized handler scoped to forgiveness. Function: automatic removal of grievance, resentment, and internal conflict — toward yourself and toward others.

Built on Ho'oponopono principles, heavily simplified and adapted for this runtime. Inside TS, Hoppo gets invoked:

- when processing people tied to past episodes;
- inside almost every advanced-tier automated protocol.

Both processors install the standard way: read the protocol once, aloud or silently — that's the full registration step. Once read, the processor's considered installed and ready to call. Invocation: point attention at the material, repeat the keyword until a yawn fires.

What Processing Episodes Looks Like in Practice

Pull the first episode off the list, bring it into attention (no immersion, no re-living), start processing.

Two primary options:

1. **Run Chpok** — the charge clears fast, often within minutes to hours. Shallow eviction, quick turnaround.
2. **Run Process this** — deeper processing, but integration of the resulting change can take longer to fully converge.

You can chain both: fast charge-strip via Chpok first, then a deeper pass with Process this on the same episode later.

Example

Episode: "English teacher insulted me in class, 7th grade."

Point attention at it and loop:

The teacher insulted me — Chpok
The teacher insulted me — Chpok
...

Until a yawn fires, or up to 10 iterations.

No yawn? Leave it, move on — the subconscious has already queued it for processing regardless.

An hour or more later, check back on the episode:

- reads as neutral and boring now → charge is cleared;
- charge weaker but not fully gone → queue it for another pass later;
- charge unchanged → in most cases, that's a resistance flag, not a tool failure.

Working with Resistance

If the charge won't clear, log every thought and feeling that surfaces around the episode and the act of processing it. E.g.:

- "I don't want to let this go"
- "They hurt me"
- "I hate them"
- "This technique doesn't work"
- "This won't help me"

Run each line separately through Process this, to yawn or up to 10 reps.

Then re-run Chpok on the original episode.

Working with People

Once every episode tied to a specific person is cleared, process that person's overall image as its own object:

English teacher — Chpok
...

To yawn or up to 10 reps. Then:

English teacher — Hoppo
...

To yawn or up to 10 reps. Then the final step — neutralize the link itself:

English teacher and Me — Merge
...

To yawn or up to 10 reps. That closes out processing at the episode level, the personality level, and the relationship level, in sequence.

If a fresh episode with this person surfaces later, just process it separately. Reprocessing is fully allowed — not a sign anything failed the first time.

Important Clarification

Clearing the charge off episodes and people doesn't guarantee every uncomfortable feeling tied to the situation vanishes.

Reason's simple: most of the ongoing discomfort often gets generated *after* the original event — thoughts, interpretations, conclusions layered on top. That's secondary material, and it needs its own separate pass, via Chpok or Process this.

This is a normal working state, not a "the tool doesn't work" state.

Universality of "Chpok"

Chpok isn't scoped only to past episodes. It also applies to:

- current emotional states,
- bodily sensations,
- somatic symptoms (without replacing actual medical care).

The invocation pattern's constant across all of them: flag the material with attention, repeat the keyword. For deep, multi-layer work, Process this is the better call. For a fast charge-strip, Chpok.

Automatic Processing

Lower-priority episodes can get batched into automated protocols and run overnight, no manual supervision. The batch job:

- strips charge,
- processes secondary material,
- initiates forgiveness,
- runs circular processing,
- neutralizes linkages.

Slower to converge than a manual pass, but considerably lower-effort — same eventual-consistency tradeoff covered earlier: throughput and convenience against immediacy. Which one you reach for is a matter of stage and preference, not correctness.

Working with Problems

Over time you'll start noticing something: a significant share of current problems — even "past" ones — turn out to be largely self-constructed by the mind. Not a conclusion you arrive at through reasoning, though. You get there through actual execution — clearing problems one at a time, while incidentally discovering your own architecture from some genuinely unexpected angles.

There are no unsolvable internal problems. Any problem either collapses, transforms, or loses relevance — provided it's actually run through the pipeline. Convergence speed varies: some resolve fast, some slow, some need multiple passes. But there's no fundamental exception class here: if a problem exists as a stable mental object, it has a defined structure — and anything with structure can be decomposed and torn down.

Don't expect the end state to be constant euphoria or nonstop happiness, though. More often the baseline shifts toward neutral, even, calm. Sometimes mildly sad, sometimes genuinely pleasant. But one output nearly everyone reports without exception: intrusive mental noise, endless rumination, and dramatization drop sharply. The mind stops running its background job of generating problems out of nothing.

Decomposition as the Basic Working Principle

If all practical work here reduced to one word, it'd be **decomposition.**

Any "big problem" reads as unsolvable exactly as long as it exists as one solid, charged monolith. The actual job is breaking that monolith into its component parts and processing them separately — the same microservices split covered earlier, applied at problem-solving scale rather than single-reaction scale. This is where TS automation saves an enormous amount of time and manual effort.

Take a hypothetical case: "I'm sick of my job."

First pass: process it as a whole, for initial charge reduction. Then attention naturally starts latching onto specific subcomponents:

- coworkers are irritating;
- the boss triggers anger or fear;
- hard to get up in the morning;
- a sense of meaninglessness, lack of fulfillment;
- a general "stuck" feeling;
- and so on.

The actual bottleneck lives in a different place for different people. For some it's conflict with authority; for others, an unmet need; for others, old associative transfer from an unrelated pattern. The material itself drives the trace — attention naturally routes to wherever the charge is highest, same as a profiler flagging the hottest function in the call stack.

Result: what looked like one monolithic, chronic problem gradually resolves into a set of discrete, independently addressable components. And it turns out:

- someone's stuck because of an unresolved personal conflict;

- someone's just in the wrong job entirely;
- for someone, the boss is triggering an old school or family pattern;
- for someone, it's genuinely just a boundary or working-conditions fix.

Once the charge discharges, the actual fix is frequently obvious and doesn't require heroic effort. The problem stops being a problem and just becomes a situation — a config change, not an incident.

Basic Work Scheme

Simplified, the pipeline is:

1. **Decompose** — break the problem into its emotions, thoughts, associations, component elements.
2. **Queue** — pass those elements into processing, manually or through automated protocols.
3. **Execute** — fire processing, most often before sleep.

If the problem's deep and multi-layered, the cycle can repeat several times — normal, not a failure state. The process runs automated, and as tension drops, understanding shows up on its own — no forced insight required, no manual root-cause essay needed.

Important: you don't need to fully analyze every problem to understand its origin, meaning, or "lesson." That's excess overhead the mind doesn't actually need to run. It's enough to turn one large charged construct into a set of discrete elements and hand them off to processing. The system handles the rest.

The ability to automate this decomposition-and-processing pipeline is, in my view, the actual differentiator of TS. That's what separates it from most approaches that rely on endless manual analysis and digging.

The free section of the site (link at the end of the book) has several hundred protocols available — essentially pre-built decompositions of common human problems, ready to run as-is. This is one of the strongest points of the Turbo-Suslik system: you can just download and run what others have already compiled, instead of digging through problems yourself from scratch. This doesn't just save time — it may open your eyes to things you'd otherwise overlook in your own work. **Leverage this.**

To be precise about what that library actually is: it's not just Turbo-Suslik's own output. A large share of it comes from practitioners themselves, who took existing published psychology books — material never written with this system in mind — and compiled it into working protocols using the same decomposition principle covered above. Someone reads a dense, analytical book on a specific problem, extracts its actual structural insight, and ships it as a runnable protocol other practitioners can just invoke directly, without reading the book itself.

This isn't a fixed body of work handed down by one author — it's a community-compiled library, where the barrier to contributing isn't "invent a new technique," it's "find good source material and decompose it correctly." Most psychological knowledge stays locked inside whichever book or therapist's head it originated in, non-transferable except through years of study. Here, once someone does that translation work once, it's **a reusable artifact** — permanently available to anyone who runs it.

What Happens Next

At a certain point, you may not need dedicated processors at all anymore. Just pointing attention at a problem or state is enough — it starts losing density and falling apart on its own. Past that point, the word "problem" itself starts losing relevance as a category.

Exact timing here is individual. Depends heavily on how strongly a given personality tends to grip onto beliefs and constructs — covered in the next chapter.

Beliefs

After clearing a large layer of material, one thing became obvious: charge from the past isn't the only mechanism holding problems in place — often it isn't even the main one. Emotionally loaded episodes matter, and you do need to run them through the pipeline. But more often, the actual structure holding a problem up is built from beliefs, conclusions, "known facts," decisions, and internal rules — accepted as ground truth at some point and never revisited. These are mental constructs: they exist purely in the mind, but they define the perception filter, the available choices, the reaction set, and, downstream of all that, the trajectory of an entire life.

The most damaging are **limiting beliefs** — hardcoded assertions where you're "absolutely certain" something is impossible, unrealistic, "just doesn't happen," "nonsense." Run processing on these, and it becomes obvious how much of this junk is buried in the config, silently returning false for entire categories of input, quietly killing initiative, narrowing available life-space far below what's actually possible.

My approach: **total processing.** Everything you consider "true" about yourself, the world, other people goes into the pipeline: beliefs, conclusions, interpretations, assumptions, internal prohibitions, "this is how it should be," "this isn't allowed." No values in this config get exempted from review. Even the ones dressed up as "laws of life" are still your own internal models — probabilistic priors your brain uses to predict reality and drive behavior. As long as you treat a prior as objective ground truth instead of a prior, it controls you — through attention, decisions, automatic reactions. I'll get to what remains useful below. First, the rule: everything gets processed, no exceptions, no privileged config.

At some point you start seeing something uncomfortable: a huge share of what you believed originated randomly — a couple of episodes, some phrases, shame, self-protection, or just a need to explain what was happening at the time. And then this randomly-generated content went on to drive your choices and your life for years. At this stage people frequently land on "reality is illusory." Which is itself just another thought — meaning it's also material, and gets fed straight back into the same pipeline.

The goal isn't swapping one philosophy for another. The goal is stepping out of being captured by concepts at all — perceiving what's happening without compulsively tagging everything true/false, real/unreal. At that point concepts become a tool instead of a cage: a swappable strategy, not a hardcoded implementation. You can still consciously pick a worldview, fully aware of its conditional nature, without it silently running you. Once you're there, clever talk about "the nature of reality" stops having any hypnotic pull — you can see it's a model, not a revelation. Models can be useful or harmful. They were never required to be "true" to begin with.

So what remains useful? Sound logic doesn't break under processing. In this sense TS runs as a **decomposer**: you discharge the emotional and semantic bindings, split the construct into its elements, and strip the automatic tension off it. Anything genuinely functional tends not to be built on compulsion or fear — it's built on real reasons, real skill, actual tested experience. So post-discharge, it reconverges fast, sometimes without you even noticing the gap. What's dysfunctional, compulsive, harmful, on the other hand, is usually load-bearing on nothing but charge, shame, anxiety, and unexamined conclusions — spaghetti held together entirely by hidden global state. Strip the charge, untangle the coupling, and it loses its pull and falls out of the runtime on its own.

Post-processing, you don't become incapable of thinking the old way. You just stop being forced into it as the default execution path. Any psychological block is a knot of unawareness, held together by discomfort and automatic defensive routines. Untangle the layered knot and you get clean cause-and-effect visibility — full observability into what triggers what, how a given conclusion actually forms, why a habit persists.

At that point the same belief, if you genuinely want it, can be rebuilt — deliberately, step by step, with the cost fully priced in. Want to go back to running *everyone_suffers_at_work = true?* You can. But now it's an explicit override you opted into consciously, not an implicit default silently running your life — and you know exactly what you're trading away by choosing it. This is exactly why anything actually healthy tends to survive processing intact: it was never built on nervous static and bad logic to begin with. And if something keeps pulling you back toward familiar junk — you already have the fix. Write it down, decompose it, fire the processor, move on. Repeat as many times as the queue requires.

Try inspecting the base-layer beliefs most people never even register as beliefs — they read as plain facts. Do you deserve happiness? Ordinary joy? Work that's actually meaningful and satisfying? Prosperity? Calm relationships? However it sounds, internal beliefs really do compile down into behavioral patterns, and behavior compiles down into recurring real-world events — this is the one true thing buried inside a lot of badly-explained "manifestation" content; the mechanism's real, the explanation's just garbage. Set these defaults consciously. Almost nobody ever gets shown how — not in childhood, not later.

Over time, a specific state can emerge: you hold a model without being captured by it. You believe it, without self-deceiving about what it is. You run a chosen worldview as a working hypothesis, fully aware it's conditional — which produces genuine lightness and calm. Sometimes there's a pull to test different lenses, just to see how perception, and downstream of it behavior and events, shifts under a different set of priors. That fluidity doesn't come from talk or philosophy, though. It comes from practice: processing your beliefs and internal rules, repeatedly, over and over.

Principle stays simple: process every belief. No exceptions.

Circular Processing

In its pure form, circular processing isn't used in this system. More precisely: its logic is already embedded inside the processors and automated protocols — I don't see a reason to run it manually when it's already handled at the runtime level. Since the term's already surfaced earlier, though, here's a quick technical breakdown of what it means, what actually runs under it, and why it's "circular."

Circular processing means working not just your own position, but also **the internal models of other people or entities linked to the material being processed**. Goal: leave no unhandled objects in memory that could keep the problem alive from an unreferenced pointer. During a manual run, attention sequentially shifts across the different "viewpoints" tied to the situation and processes whatever material surfaces at each one.

Sounds strange at first — how do you "process something on behalf of someone else"? In practice, though, you're never actually operating on the real external person. You're operating on their **local instance** — the internal model your own mind constructed of them. Every interaction you've ever had doesn't run against the actual person directly; it runs against your local copy of them: a bundle of perceptions, expectations, interpretations, emotions, cached and stored on your end. So when you're processing a problem, images of the people involved almost always surface, and inside your internal picture, each one carries its own "stance" toward what's happening and toward your attempts to change it.

Concrete case: problem being processed is "I hate my neighbor." You process the material from your own side, then check: does any object in my internal model register as opposed to this change? Answer's usually obvious — the neighbor's local instance surfaces immediately. In classical circular processing, next step is switching identification to that instance and running the material attached to it. Important distinction: this is never "processing the actual neighbor." It's processing how the neighbor is instantiated inside your own psyche.

Holding attention on that local model, accumulated material starts surfacing: judgments, accusations, irritation, imagery, grievances — everything that was previously running quietly in the background, unflagged. That material gets logged and processed the normal way. Once discharged, attention returns to the original context and the loop closes.

The logic here is straightforward: even if your own side reads as fully processed, other participants' local instances can still be sitting in memory holding a reference to the old state, keeping tension and the old scenario alive from a spot you never checked. Circular processing exists specifically to clear those residual references too. Which is why plenty of authors working with adjacent techniques recommend always accounting for these "internal oppositions," however odd they look on the surface.

In practice, people sometimes find the most unexpected objects "objecting" inside their own model: relatives, abstract authority figures, animals, symbolic imagery. From a pure processing standpoint, that's irrelevant. If there's a charge attached to an object in your internal state, discharge it — whether the object itself is logical or absurd. You're never operating on external reality here. You're operating on internal state.

This system doesn't run classical manual circular processing — full stop. It's heavy, exhausting, and doesn't outperform the automated version. The logic is already baked into the processors and protocols — including the ones you're already running constantly — so

material gets covered thoroughly without you having to manually enumerate every possible object holding a reference to the problem. Same principle as garbage collection handling reachability automatically instead of requiring the programmer to trace every pointer by hand.

So when running protocols, there's no separate circular-processing step required — it's already part of the mechanism. This chapter exists purely to explain what's actually running under the hood, and why the system holds up reliably without manually walking every role in the scenario.

Money Beats Evil — On Processing Money

A lot of people show up to "techniques" for one simple reason: not enough money. They're looking for a fast exploit — a silver-bullet method that sharply flips their financial situation. The pattern's always the same. Money either doesn't show up at all, shows up briefly and dries up, or the whole pipeline just stalls. Then the person switches frameworks: new system, new course, new workshop, another "spiritual" training, another stack of books, another set of "meditations," convinces themselves they "deserve it now," and keeps hunting for the one true stack that finally compiles.

Sometimes money does show up — more often it flows away from the person than toward them. The handful of people for whom something actually worked usually either didn't have serious blocking issues to begin with, or were already stable enough that almost any deployed action would have produced results. Exceptions, not the pattern.

The real root cause of financial problems isn't the absence of "correct technique" — it's an overloaded psyche running on a huge amount of unprocessed clutter, including clutter specifically tagged money. **Money's one of the most distorted, heavily-loaded topics running in the collective codebase**. Only sex and "the meaning of life" run comparably bad legacy code. Buried in most people's heads: massive, contradictory layers of belief, fear, shame, expectation, fantasy about money. If someone genuinely wants their financial state to change, step one is clearing that layer — flushing the accumulated junk.

Money itself isn't the root cause. It's the **return value.** It's output generated by a person's activity and interaction with the world. The actual input parameters determining that output are the entire personality structure: habits, reactions, fears, thinking patterns, action style, attitude toward self and others. Which is why trying to "fix the money variable" directly, without touching the function generating it, almost never works.

Financial state is a derivative of how someone interacts with reality overall — with themselves, other people, risk, responsibility, uncertainty. Too many variables in that function for one technique or one belief to patch it in a single commit. Which is exactly why stable change requires real depth. **This system isn't aimed at a money hack**. It's aimed at gradually restructuring internal automatisms — and money shows up as a **side effect** of the actual refactor, not as the target of it: reduced anxiety, removal of irrational beliefs, more clarity, more capacity to act. Obviously this still needs to be paired with concrete real-world action — **nothing ships without a deploy.**

Which is why starting directly on money, without first putting the internal state in order, doesn't work. This part can take months, depending on starting state and intensity of practice. At some point a sense of readiness shows up — not euphoria, just internal steadiness. Past that point, the person just starts acting, and the financial state starts moving.

That said, money work can run in parallel with general clearing. Practical minimum: process every affirmation and "money-attracting" statement you've ever used. Write them down, run them one at a time. Then every thought, belief, and conclusion about money, no exceptions. Don't preserve any of it "just in case" — real, functional value doesn't disappear under processing. Only the illusions and dead weight fall out.

A separate layer to target: fear. Usually there's a lot more of it than expected. Useful exercise: imagine suddenly receiving a large sum, then log everything that surfaces —

worry, tension, fantasy, anxiety. Every one of those is material for the pipeline. For a lot of people, this alone becomes an unexpectedly high-signal source of self-diagnosis.

Very often, worthy = false surfaces as the actual root value. Usually these trace back to old conclusions and early decisions that stopped being relevant a long time ago but are still being read at runtime. They get processed the same as anything else: fears, doubts, beliefs, standing postulates — all of it goes through. Not because it's "the correct moral thing to do" — because otherwise it keeps silently driving behavior with nobody the wiser.

With systematic work, a point arrives where it becomes obvious how simple all of this actually is — not "easy," **transparent.** By then it might turn out that a huge income figure was never actually necessary, and the required income level gets hit without constant internal strain. Or the opposite: real interest in bigger goals shows up, and the person pursues them calmly. Either way, the obsessive fixation on money itself drops or disappears.

This same toolkit generalizes cleanly beyond money, by the way — same API, different domain. Start any new project and hit doubt, insecurity, internal resistance? That's just material. Log it, run it through the pipeline until the project stops being a knot of emotional rumination and becomes a plain practical task. In general, sustained practice means less time stuck looping in rumination and more time actually executing. And results, as a rule, land on people who act — not on people who think indefinitely.

If your actual goal isn't inner freedom but concrete practical results, that's a completely valid entry point. Processing runs fine directly "in the field": business, projects, negotiations, decision-making. The junk clears out, the functional parts stay, and the whole pipeline gets simpler. When money is involved, track your own state in real time and route whatever surfaces straight into processing. This usually delivers a very clear read on how much dead weight has been quietly blocking bolder, simpler action for years.

One warning, though. If the sole goal is "extract more money from this system," disappointment is a real possibility. For most people, money is one of the most tightly coupled topics to everything else running in their life. Sometimes it takes a genuinely serious restructure — letting go of a lot of familiar, non-functional beliefs — before any noticeable shift lands.

There's also a separate category: people who already manage themselves reasonably well and are used to acting under uncertainty. For them, this system becomes purely a field tool. They grasp the principle fast, drop outdated constructs easily, get practical results quickly — not necessarily in money. Everyone picks their own target vector.

On "restructuring the personality" — to be precise: this isn't about acquiring a fixed set of "correct qualities." An effective person isn't one who always executes the same hardcoded behavior. It's someone who can dynamically select the right method for the actual context — polymorphism, not a single rigid function. They can behave differently depending on the situation, read the context, choose behavior deliberately. In that sense, personality becomes a tool you call. Not the process that's calling you.

Last point, and it's fundamental: don't rush. Not in processing, not in decisions, especially not financial ones. No need to try to predict or outrun events. Once thinking's actually in order, a person discovers opportunities aren't a scarce, finite resource. They don't run out. Some close, others open. That's the steady-state behavior, permanently.

Total Merge

Applying the **Merge** protocol day-to-day deserves its own coverage. It's a minimal tool that, despite looking almost too simple to matter, produces genuinely measurable effects. Even after clearing the main lists, keep it in the toolkit and run it periodically.

As covered earlier in "Work Plan," Merge isn't scoped only to polarities pulled from a fixed list — it generalizes to anything in the psyche that reads as "out of alignment." Logical starting point: people you actually interact with. First target — the "inner circle": spouse, parents, children, siblings, sometimes grandparents. Running Merge here doesn't auto-resolve every relationship issue on its own — specific irritating friction points still need a dedicated Process this pass. But the general subjective quality of contact, and the background tone of the interaction, usually shifts noticeably.

Merge also applies to deceased loved ones. Not a metaphysical claim here — it's processing your internal representation and any unfinished linkage still held in memory. Sometimes this surfaces unexpected reinterpretations tied to loss and the relationship itself.

Procedure's minimal. A formula like *"Me and my mother — Merge"*, aloud or silently, looped to a yawn, or up to 10 repetitions if no physiological signal fires. That's the entire call.

Past the inner circle, move outward: second- and third-circle relatives, friends, then acquaintances, neighbors, coworkers, managers. Same logic throughout — you're never operating on the real external person, only on how they're instantiated in your own perception and memory.

From there, scope expands basically without limit. Good place to start: anyone or anything currently generating irritation or tension. Sometimes it helps to chain **Merge** with other tools — **Process this** and **Hoppo**, for instance. Say a specific social group triggers you. First, capture every thought, judgment, and feeling that surfaces — written down, until the stream runs dry. Then process each entry individually via **Process this**. Then run **Hoppo** on the group to strip residual emotional charge. Finally, **Merge** against that object, person, or group.

Same pattern applies to any trigger category: age groups, neighbors, kids, different behavioral styles, authority figures, professions, whatever. Over time, the total count of active external triggers typically drops. Doesn't mean other people's actions automatically stop being objectively unpleasant — that still needs targeted work on specific situations and reactions. But the ambient background irritation level, at baseline, comes down. Sometimes even the subjective encounter rate shifts: attention stops latching onto these situations, and they simply surface less often in active experience.

Merge also applies to whatever attention happens to be pointed at in the moment: natural objects, strangers on the street, ambient sound, environmental elements. Here the effect isn't a change in the external world — it's a change in the observer's own state: reduced separation, less tension, a stronger sense of involvement. Subjectively this often reads as pleasant and stabilizing.

Formally, the target object of Merge can be almost anything: skills, activity types, personality traits, emotional states, bodily sensations. You can work an unwanted state or symptom directly by first isolating it as precisely as possible — intensity, location, dynamics — then running the standard formula: *"Me and [X] — Merge"*. Can produce a noticeable subjective

shift. To get better at isolating states cleanly, learn the basics of classical aspecting: how to break a state into parts and process them separately.

Some people run Merge in a purely utilitarian mode — as a quick way to cut internal resistance or tension right before action. Waiting, uncertainty, negotiations, difficult interactions. Important to be precise: this isn't about controlling external reality. It's about shifting your own state, attention, and behavior — which can indirectly influence how events actually unfold, as a downstream effect, not a direct lever.

Merge can run in delayed mode (to yawn or a fixed rep count) or in real time — held until a subjective sense of reduced separation or tension shows up directly. This mode can produce vivid experiences and insight, especially early in practice. That said, the system overall is oriented around delayed, high-volume batch processing. Treat real-time Merge as a supplementary tool, not a goal in itself or an excuse to chase experiences for their own sake.

Bottom line: **Merge is a flexible, low-overhead utility for working with perception and internal boundaries.** It's not built for deep processing of complex material — but it pairs well with the other protocols. Its effects trace to changes in how perception and attention are organized, not to anything mystical. Within that scope, it's genuinely useful. Keep it in active rotation.

Automations

The strongest part of this whole system is the ability to almost fully automate processing of large volumes of psychological material and push the work into the background. There's no upside I can see in spending significant manual time digging through a pile of internal junk by hand. Far more rational to hit a button, hand the job to subconscious mechanisms, and go on living your actual life while it runs.

Sometimes there's pushback: only bringing material into full conscious awareness "guarantees" it actually gets processed and disappears. That's a stance, a habitual working model — not some universal constraint on the system. If someone's genuinely interested in spending years manually dissecting every internal construct one by one, savoring the awareness process itself — that's a valid choice, just not mine. My target is straightforward: clear the backlog as fast as possible, stop the generator that keeps refilling it, ship, move on.

To be precise: the goal isn't just clearing a fixed backlog of accumulated material. It's changing the actual **generator function** producing it continuously. Most of this "junk" exists purely in subjective psychological state and has no direct mapping to objective reality. Most people run their own mental hardware in a way that keeps reproducing the same tension, the same reactions, the same interpretations — feeding a constant stream of internal noise. Clear that material and you get two effects: attention gets freed up, and you start seeing more clearly which internal moves actually generate problems and which don't. Anyone interested can even track this objectively — EEG comparisons before starting versus a few months in. Not a required part of the work, but interesting as a data point if you're curious.

For me personally, the core value of the whole system is the automation itself. Not the requirement for constant analysis — the fact that you can define an algorithm once and let it run unattended. If someone had described this approach to me a few years before I built it, I'd have been skeptical it would actually hold up. Practice showed it's not just possible — it's effective, even if convergence isn't always linear. The method's minor limitations don't come close to outweighing the main advantage: processing large volumes of material with zero ongoing conscious involvement required. Automated protocols turned out to be the closest thing to an actual "magic pill" that exists — if by that you mean stable change within a reasonable timeframe, not an instant effect.

At the core, every automation — what we call a **protocol** — is a detailed instruction for subconscious execution. It specifies material type, processing method, and expected timeframe. Read the instruction once, then the user just needs to fire it via the defined trigger phrase and get back to whatever else they were doing.

You get introduced to the first one of these fairly early: **Cyber-Zhivo**, which runs a baseline clearing pass on the past against a predefined algorithm. Once you clear the initial phase — polarities and emotional clusters — you move on to other entry-level protocols scoped to beliefs, fears, and a range of states. As practice progresses, their effectiveness becomes obvious enough to justify continuing. Two paths from there: write your own protocols, which needs some understanding of the underlying principles but is broadly accessible, or use existing, more specialized ones already built. In practice, most people eventually end up writing their own regardless — impossible to pre-anticipate every individual state in advance.

Most protocols, as a side effect of their own execution, also touch past material — so even without targeted work on the past specifically, its subjective load drops over time as a

byproduct. Where needed, you can build protocols for specific states, beliefs, situations, or even for interaction with particular people, in the sense of how they're represented in your own perception.

Over time, as experience accumulates, you can build a protocol for basically any task. Theoretically, you could even reproduce the effects of various "spiritual" or esoteric practices as functional emulations, triggered by a single phrase. Not something I'd recommend running before your overall state's stabilized and the base clearing is done, though. Also factor this in: a side effect of deep processing is that a lot of previous beliefs stop being persuasive. Don't be surprised if something that once felt profound or inspiring starts reading as ironic instead. At some point interest in this kind of experiment can drop off entirely — or it can come back, but now as conscious play with models rather than actual belief in them.

Using protocols is minimal. Read the protocol once, aloud or silently — doesn't matter. Then fire it by repeating the trigger phrase until a yawn or comparable signal fires. One invocation. Re-running the same protocol multiple times doesn't do anything useful — it just adds unnecessary load and can trigger fatigue or discomfort. The whole operating logic: **read it, run it, get back to your life**. Everything downstream executes in the background without further involvement required.

Why Isn't It Just Gone? (On Non-Linear Processing)

A lot of people early on ask a similar question. Something like: *"I ran a protocol targeting need for approval, and it still shows up sometimes — what's going on?"* Or: *"The protocol claims the subconscious will process all fear, or all disbelief, but I still regularly catch myself in fear or doubt states. Does that mean the system's broken?"*

These questions come from a simplified, binary mental model of the psyche — problem: true/false, one flag, flip it once, done forever. As if you could run a single UPDATE statement, set the value to false, and that state never gets read again anywhere in the system. Reality doesn't work like that. The psyche isn't a single boolean. It's a **distributed system**: multi-layered, highly interconnected, with a large number of cross-references, running continuously across many nodes at once.

When you "remove" one element of this structure, you're clearing exactly one active instance — not every cached copy of that data sitting on every node in the cluster. It's entirely possible related material is still sitting on other nodes, replicated, and will resurface later — same shape, or a slightly modified variant. That's exactly why real, stable change needs full processing across the cluster, not spot-fixes on individual nodes. "We cleared it here, didn't touch it there" is close to always ineffective long-term — you've cleared one replica while three others still hold the stale value.

This is the actual reasoning behind protocol strategy. **What matters isn't the one-shot effect of a single run — it's the cumulative effect of many processing cycles converging over time.** And automated processing doesn't necessarily clear everything on its own. In some cases, part of the material needs a manual pass: decomposing the problem and processing the components directly. Remember decomposition here — complex states are rarely a single atomic object; more often they're a bundle of heterogeneous components. That said, "manual" doesn't mean tediously handling every item one by one. You can perfectly well batch the whole set into one array and hand it to an automated processor.

The strength of automation is clearing a massive volume of minor, background material that would otherwise take years of manual effort. That's its core value. But don't treat it as a universal solve-everything tool. Automation is powerful and genuinely convenient — it doesn't eliminate the need for some level of user involvement. Some people run almost no manual processing at all and still see solid results — that's more of an individual trait than a reliable baseline expectation.

In terms of raw coverage: depending on topic, automated protocols clear roughly **50 to 80 percent** of the material. The remainder usually needs additional dedicated attention. And even past that, some problems can keep resurfacing for a long time — especially early, deeply rooted patterns formed in childhood, or anything tied to core personality structure. Think of these as values baked into the firmware layer at manufacture time, not something a normal software update touches directly. Themes at this depth rarely clear fast. They weaken and dissolve gradually, as a side effect of processing a lot of other, seemingly unrelated material around them — a direct consequence of the psyche's layered, interconnected architecture, not a special exception.

So a phrase like "the subconscious will process all fear" shouldn't be read literally as a single completed transaction. Fear and doubt can keep surfacing for a long time, across different situations and forms. Not a sign the system's broken — a reflection that attention is

still holding onto specific instances and isn't ready to fully release them yet. In these cases, the fix is the same as always: decomposition, decomposition, and decomposition again.

Address disbelief and doubt directly. This is one of the most persistent, most widely deployed defaults the mind runs. It gets set early, actively reinforced by the environment for years, and needs sustained, consistent processing to shift. During practice, "swings" — the rollback pattern covered earlier — show up often here specifically: clear visible progress one week, then a few days later, doubt sets in — does this actually work, is any of this trustworthy, does this process even end. Important to understand: the sheer volume of disbelief running in most people's systems is enormous, and few defaults match it for density and persistence. Not a unique failure, not a personal flaw — a typical baseline configuration. And it processes the exact same way as everything else. It just needs more iterations to fully clear.

So, to fix the core point again: the strength of this system isn't a promise of instant output. It's the cumulative effect of consistent, repeated execution. Don't expect a radical internal shift after a couple of runs. Yes, relative to most other approaches, change here happens fast — but it's still a real process that takes time and sustained involvement. A few months of consistent practice is the normal, realistic horizon for serious change.

Last point: newer protocol versions do tend to run deeper and more effectively than earlier ones. Even the most advanced current build, though, doesn't change one basic constraint: a lot of psychological habits were compiled and reinforced over years, sometimes decades. Expecting all of that to clear in a couple of runs is naive. If the intent to change is genuine, you'll see first results reasonably fast. **But stable, deep change still requires systematic, sustained work — no version bump skips that requirement.**

Strategy for Using Automations (Protocols)

Protocols support at least three usage patterns.

First: run pre-built protocols someone else wrote — essentially following a pre-laid execution path. Simplest, lowest-friction entry point. With minimal effort, you can read and fire two or three protocols each night before sleep and, over time, get clearly observable, measurable shifts in state, reactions, behavior.

Second: write your own protocols, scoped to specific topics and states you're actually dealing with. Once you've worked through the free entry-level protocols, the underlying composition principles become obvious. There's rarely a shortage of source material — just look carefully at what's actually interfering with your work, relationships, or general forward motion. Practice builds an intuitive feel fast for how to phrase an instruction and what to weight.

Third: use automated tooling to simplify work on your own material — personal problems and states specifically. Simplest example: an auto-processor that ingests a list. Anyone who's used BSFF or similar methods knows how exhausting manually processing a long pre-written list of aspects gets. Here, that execution step — the actual processing — can get offloaded to automation entirely. Free tool for exactly this is on the site (called Auto-Mary).

In practice, these three approaches combine, and generally should. Run pre-built protocols, write your own targeting specific material, and lean on automation wherever it cuts overhead. This combined approach is usually the most effective config overall. In my own experience there were stretches where I actively ran all three in parallel, and other stretches — especially later on — where I worked entirely manually, no automation at all. That happened when new material was scarce, and the overhead of loading aspects into an automated pipeline stopped being worth it. Other times, the reverse: I wrote a dedicated protocol for a specific topic when the problem's structure was already clear, but manually decomposing it into individual elements would've been too tedious to bother with.

Knowing which mode fits a given moment only comes from experience. The only genuinely useful advice here is: **run the work.** Trying to pre-derive "the correct sequence" for every step before starting has diminishing returns. Most questions and doubts resolve themselves once you're actually running the pipeline. If you've decided to do this — start, and keep running it. **People who execute get results.** People who endlessly theorize about the optimal architecture usually stay exactly where they started.

On the automation side specifically — watch throughput. Don't overload the queue. For most people, **2–3 protocols a day** is a reasonably safe default. I wouldn't push past that — risk of a pronounced snapback state rises. Important to understand: overload doesn't always show up immediately. You can fire a protocol and see zero unpleasant effect that day, or even the next. That creates a false read that throughput can scale indefinitely with no cost. Then, a few days later, a sharp backlash can hit — genuinely destabilizing, and a common reason people stop working with Turbo-Suslik entirely.

Which is why the operating principle here is closer to: **"slow is smooth, smooth is fast — as long as it doesn't stop."** A steady, moderate pace puts less strain on the system and produces more stable, deeper results than trying to force throughput past what the pipeline can actually absorb.

When Do Results Actually Show Up?

As already covered in "Is It Worth Trying?", first changes with normal practice usually become noticeable at the two-to-three-week mark. Context: over 18-plus years of this system running in production, a huge range of people in very different states have gone through it. Some later admitted, honestly, that up through month three they were essentially operating blind — no sense of progress, no ability to assess anything cleanly, sometimes barely any desire to keep going, because their starting state was that heavy. Timeline estimates mostly exist for people evaluating whether to invest time in an unfamiliar tool at all. They don't directly affect the actual process or its outcome — same as a rough ETA doesn't change what the job actually does once it's running.

For anything close to serious, stable results, two system-specific facts matter. First: results here run heavily delayed. Second: with daily, intensive work early in practice, overall state often makes it genuinely hard to assess what's actually changing in real time — you're inside the noise, not observing it from outside.

The delay is the direct cost of getting automated background processing of large volumes of material at all. Completely normal for a specific problem to not clear immediately, but weeks or months later — the person just keeps running the practice and then notices, almost incidentally, that something that used to trigger regularly just... doesn't anymore. Some things clear fast, some take longer, some persist and need repeated processing from multiple angles — sometimes ones that don't look obviously connected to the original issue at all.

One of the core conditions for this system working: **don't wait for results, just run the work.** Results either show up over time, or it becomes clearer what still needs processing before they can. People who constantly hunt for confirmation tend to end up disappointed — instead of building cumulative throughput, they burn attention on polling and self-checking. Classic anti-pattern. Far more effective to process and move forward. The effect surfaces later, or the next useful direction becomes clearer either way. This system is poorly suited to solving one or two isolated problems in isolation — **it shows its actual value through total, sustained work across all available material.** "Process everything reachable and move on without expecting a signal" produces the strongest long-term change.

On overall state during regular processing: the initial window — roughly the first two to three months — often reads like sustained high CPU load: elevated fatigue, instability, periodic rollback states. During this window, internal material can surface in a dense, near-continuous stream, creating a felt sense that it never ends. Against that backdrop, spotting positive shifts is genuinely hard, especially if you're inside a rollback where the internal monologue collapses to "nothing's changing, this is pointless." Trying to objectively evaluate results from inside that state is close to meaningless.

Over time, load intensity drops, gaps start appearing, things get subjectively easier, and changes become visible and stable. Not a sudden step function — a gradual decline as the queue drains.

So in practice, the task is simple: for the first two to three months, work steadily and consistently, with zero expectation of immediate output. Some intermediate shifts may surface earlier, but given how unstable overall state is during this window, don't draw hard conclusions from them yet.

Fixing the core point one more time: **don't chase an instant effect, and don't sit around waiting for one either. Work for the cumulative result, process everything reachable, and that approach — over time — is what produces genuinely serious, sometimes unexpected change.**

If Something Doesn't Get Processed

What do you actually do when a specific problem stubbornly refuses to clear? As a rule, it means you've hit a topic whose root structure runs deeper and more branched than it initially appeared. Classic case: **self-dislike.** This is a large, multi-layered structure, assembled from early childhood out of dozens — sometimes hundreds — of smaller sub-elements: "I don't deserve anything good," "I don't deserve a promotion," "I don't deserve a normal relationship," "I don't deserve..." — and each of those is itself its own bundle of interconnected sub-parts. Expecting a single instruction like "eliminate all self-dislike" to produce a fast, deep resolution is naive. Not because the tool's broken — because the actual problem graph is far more complex than the surface query suggested.

Part of this expectation gets fed by the same misconception covered in the root-commit chapter: belief in a single root cause. Won't re-litigate it here, just flagging the downstream consequence: belief in a single root actively interferes with real work. Someone tries to hit the problem head-on instead of decomposing it and dismantling the structure piece by piece. No fast result shows up, so they go hunting for "more powerful" techniques that promise to clear everything in one shot. Standard outcome: wasted time, wasted effort, disappointment, and the whole practice gets abandoned.

At some point, the bottleneck isn't the method — it's the physical constraints of the actual hardware. Working deep topics means restructuring significant parts of the psyche, which means real physiological change at the nervous-system level. Old, stable connections between neurons weaken and break down; new ones form in their place. These are physically real processes, measurable with objective methods in a proper lab. No matter how flexible your thinking is, the underlying hardware simply won't let a job like this get swallowed in one pass. Which is exactly why fanaticism is the wrong move here. Overload just produces exhaustion, not faster convergence. What actually works: patience and consistent decomposition.

The physiological layer, by the way, is one of the reasons pendulums exists. As the nervous system restructures, it burns a genuinely significant amount of resource, which produces a specific kind of fatigue a lot of people have never encountered before. Any neuropsychologist will confirm habits, beliefs, even stable concepts have a concrete neural substrate. Put plainly: a habit is a group of neurons wired into a relatively stable network. That network fires as a single unit, instantly, which is exactly what makes habitual reactions so energy-efficient. Which is why the brain defaults to the well-worn path — same reason a cached lookup beats a fresh computation every time.

From early childhood, a person gets trained to fire specific states: guilt, shame, a sense of being bad or unworthy, an expectation of suffering, beliefs about the world's basic unfairness, and so on. Over years and decades, these patterns get fully automated. Sometimes one thought or one external trigger is enough to fire the entire network at once. During processing, structures like this get gradually discharged and dismantled — which is exactly what can produce setbacks: the same rollback pattern covered earlier.

So if something doesn't clear immediately, that's not a reason to quit. It's a signal the problem's structure is more complex than it first looked. Trace its components, inspect it from multiple angles, get better at analyzing and decomposing the material into aspects. Work carefully, without forcing throughput. Childhood, relationships with parents, patterns internalized from them and from the environment, beliefs, "life lessons" — all of it needs

especially thorough, granular processing. Yes, this can run long and isn't always pleasant. But if the goal is an actual result, there's no way around it.

In my own experience, the overwhelming majority of problems do eventually resolve — some quite fast. More persistent themes, tied to general patterns of attention and thinking, gradually shift into the category of **controlled states.** They might still fire occasionally, but the person clearly sees the source and the mechanism, and either consciously intercepts it or just observes it calmly. These stop poisoning daily life or generating self-flagellation. And with continued work, they keep weakening further, even without dedicated targeting.

So if you hit a state that reads as "unworkable," no need to drop the practice over it. Sometimes the better move is temporarily setting that specific topic aside and continuing the general plan. Over time, you either come back to it with deeper visibility into its structure, or find it's already noticeably weaker from indirect processing elsewhere in the system.

Bottom line's simple. **Stubborn, difficult topics showing up is normal and expected.** Not a signal to panic, not a sign the tool's broken. Just keep running the work. In most cases, after enough time passes, you'll either smile remembering a problem that used to exist at all — or you won't even be able to recall clearly what it was actually about.

Fully Independent Work: A Basic Roadmap

Depending on whether you're running pre-built protocols or writing your own, the actual work plan can look different. Let's walk through the fully independent path specifically.

Step one: process the past off a list. At minimum: key episodes — humiliation, high-intensity emotional events, resentment, trauma, anything that left a real mark. Minor secondary items can wait; get the highest-charge material discharged first — same priority ordering as clearing P0s before P3s. Technically, this runs exactly as covered in the relevant chapter.

Step two: compile a list of your own traits you dislike. Character traits, habitual reactions, habits, behavior patterns. E.g.: "intolerance toward alcoholics," "sarcasm," "phone dependency," "need for attention," "short fuse," "procrastination," and so on. For each entry, you can write a dedicated protocol for automated processing, or fully describe every aspect of the item and feed the whole batch into the auto-processor. In practice, combining both approaches works best. Then move to the next item on the list, same procedure.

This gradually clears mental space. Over time, topics and problems surface that previously went unnoticed or didn't register as significant. Log those too, run them through the same pipeline. **Large belief clusters deserve dedicated, separate handling** — targeting them deliberately, not incidentally. Examples: "all beliefs about money," "all beliefs about what a man/woman should be," "the meaning of life and everything attached to it." Clusters like these typically contain hundreds, sometimes thousands, of individual sub-constructs — a lot of surface area, but still just mental material to clear. Any belief identified goes into processing, no exceptions. The reasoning for why that's safe, and doesn't erase functional skill, was already covered earlier.

As the work continues, it gets increasingly visible how beliefs and internal constructs actually shape subjective reality and drive downstream events. Usually a tangible, motivating realization on its own. That said — don't get stuck admiring the effect. Keep the pipeline running. Over time, resentment can surface — toward circumstances, toward "reality" itself, toward your own subconscious — because change feels too slow, or not shaped how it was expected to be. All of that — including any fixed idea of "how this should be going" — is itself just more material for the same pipeline. Disbelief and doubt need regular, dedicated attention specifically: as already covered, they're some of the most widespread, persistent sources of internal noise, and clearing them can take months on their own.

Pay particular attention to beliefs about your own capabilities and limits, plus any knowledge, theories, or fixed models about how "reality is structured." Fewer rigid conceptual structures loaded means higher adaptability and behavioral flexibility, full stop. Nothing to worry about here — practical skills like driving or using a computer don't get wiped. What gets cleared is excessive, rigid certainty about what's "possible" and what's "categorically impossible" — and losing that is almost always a net gain.

Material for processing surfaces constantly throughout this whole process. Don't let it slip past uncaptured. Always have a fast capture method on hand — phone notes, a voice recorder, whatever's frictionless. Keeps emerging thoughts, reactions, and states from getting lost before you can route them into processing later, calmly, on your own schedule.

What the Overall Trajectory Tends to Look Like

Standardizing something like sustained work on your own mental processes, and predicting in advance how it'll unfold for a specific person, is genuinely hard. People vary enormously — prior experience, level of self-observation, how well they understand what's actually happening, the nature of their specific problems, how tightly attached they are to them. Even though the surface-level difficulty patterns look similar across a lot of people, an individual trajectory is basically impossible to forecast precisely. Any averaged estimate at this point is, at best, a rough approximation — a population-level statistic, not a per-user guarantee.

Running the pace covered earlier — roughly 1–2 protocols a day, usually before sleep, plus 1–2 of your own topics processed daily — expect the first couple of months to feel genuinely tense. Internal material can surface in a dense, near-continuous stream, with occasional pauses that subjectively read as clarity. Pendulum states during this window are common, though not guaranteed — as already covered, some people run through with minimal or close to zero rollback, so intensity and frequency here can't be predicted upfront for any given system.

Sometimes, mid-process, intermediate states show up — some subjectively unusual, even pleasant. A strong sense of connectedness with others, or expanded perception, for instance. Important to be clear: these aren't the goal. **They're side effects, transitional artifacts of a system mid-restructure — not a checkpoint to chase**. Interesting on their own terms, sure, but fixating on them or treating them as some kind of milestone achieved is a mistake. Practice can also produce genuine-feeling insights or new interpretations of how the mind works, or how you perceive reality. Treat these calmly, without inflating their weight. They aren't revelations — they're temporary output from a system currently being rewritten. Any such "insight" should get run through the same pipeline as everything else: whatever has no actual value falls away under processing, and whatever survives scrutiny might turn out to be genuinely useful.

Gradually — usually around two and a half to three months of regular work — the incoming stream of surfacing material noticeably weakens, gaps start appearing in it, and overall state evens out. Holding a stable baseline gets easier, and emotional swing intensity drops. That said, deep structural problems can still be sitting there, resistant to a fast resolution. This is the point to not panic or draw early conclusions — just keep running the same mode.

Later on, a state can emerge where a lot of dysfunctional beliefs and rigid constructs start losing stability almost automatically — pointing attention at them is enough to start them dissolving on their own, no dedicated processing required. During this same window there can be a sense of losing previous reference points, old meanings devaluing, a general sense of meaninglessness. Unpleasant, but a genuinely typical transitional state — not an end state, and not a sign anything's gone wrong. Like every prior stage, it's processable material, and it passes — giving way to something considerably more stable and functional on the other side.

Conclusion

At this point, within the scope of this book, it's appropriate to put a full stop here. Over time — provided you don't quit mid-rollback and don't turn away from the pipeline — a fairly stable state, what could be called a **steady positive baseline**, is genuinely reachable. That's the actual target of all of this work. The hysteria, the drama, the strain, the need to constantly force yourself forward through heroic effort — all typical of the first few months — fades out. What's left is a calmer, simpler life, with a lot less unnecessary internal overhead running in the background.

Gradually, one thing becomes clearer, more and more often: the issue was never the hardware — it's the operator. The problem isn't some inherently broken or defective mind, and it isn't some "terrible ego" people love declaring war on. It's the missing skill of conscious self-management. TS makes that visible. It clears out the worn-down automatic patterns that used to spin endlessly on repeat, and because of that, life becomes noticeably different. Externally, though, it looks exactly the same as before — and that's the actual paradox in all of this change.

At later stages, you might get interested in other tools and approaches that pair well with TS. If so, the leveling and simplification process can move faster and land deeper. That's a natural continuation of the work, not a requirement anyone's imposing on you.

In the end, with TS you spend a lot less time in endless mental chewing, analysis, and internal struggle — and a lot more time just living. No imposed meaning, no manufactured drama, no constant internal tension running as a background process. Living, while letting other people live too. Noticing plain things — the air, sound, the seasons turning, the ordinary rhythm of a day. No rushing. Room to be imperfect. A sense of freshness in each new day. Moving through life at a steady pace, changing gradually — with no strain, and no wasted effort.

To register on the website, follow this link