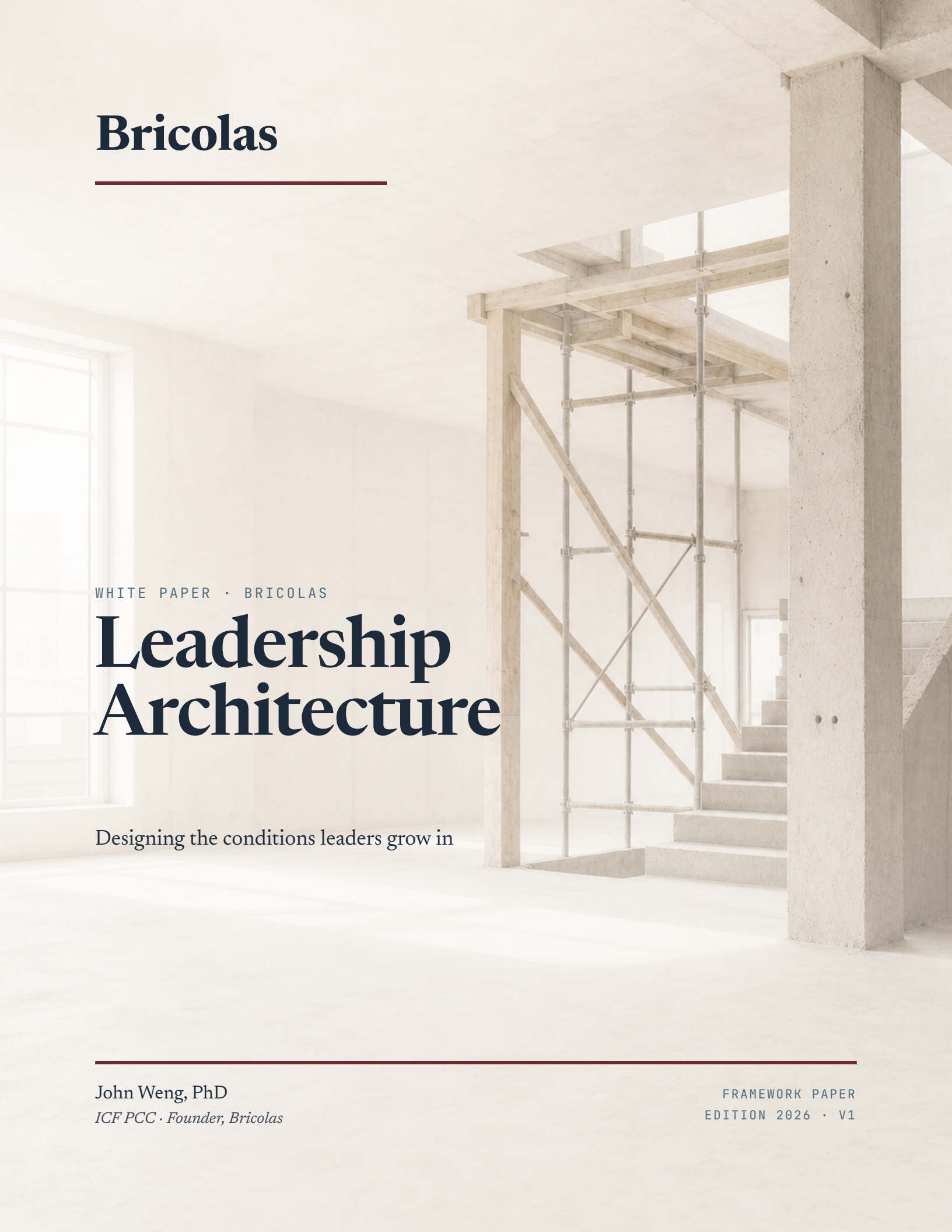




Bricolas

WHITE PAPER • BRICOLAS

Leadership Architecture

Designing the conditions leaders grow in

John Weng, PhD
ICF PCC • Founder, Bricolas

FRAMEWORK PAPER
EDITION 2026 • V1

Contents

Executive summary	3
1 · The leader is not the unit	4
2 · Leadership architecture, defined	5
3 · Why it is necessary, not optional	6
4 · The diagnostic: authorization, then four pulls	7
5 · How the work happens	8
6 · What this paper does not claim	10
7 · What this means for practice	11
References	12

About this paper

This is a framework paper. It synthesizes the leadership-development and behavioral-science literature and introduces a model, rather than reporting an empirical study. It is written for three readers: practitioners who build or buy leadership development, leaders who have been through it and noticed it didn't quite take, and the buyers who fund it.

What it claims: that leadership development, done honestly, is the design of the system around the leader (not the repair of the leader alone), and that there is a practice for doing this. **What it does not claim:** proof in the strict scientific sense, or that every engagement can or should become system redesign. The boundaries are named in *What this paper does not claim*.

How to read this

The argument runs across the first four sections: the leader is not the unit, what leadership architecture is, why it is necessary, and the diagnostic that opens the work. Section 5 is the practice: how the work actually happens, with a real engagement that shows the cost of skipping the read. Sections 6 and 7 name the boundaries and what it means for practice. And if you only have the executive summary, the claim is there in full.

The flower and the soil

When we want a flower to bloom, we tend the soil, the water, the light – we rarely fault the flower itself. We grow leaders the other way around: we pour development into our highest-potential people, then send them back to a system no one thought to tend.

Leaders come back from the best programs ready to think differently, listen more deeply, make the hard calls. Within weeks, the world they returned to has pulled them back, and the conclusion is almost never "the conditions pulled them back." It is "the leader needs more development." So we send them again.

This isn't only a metaphor. Forty years of research on whether training transfers to the job (Baldwin & Ford, 1988) points to the same place: the conditions a leader returns to decide whether development holds, more than the quality of the program ever does. **Leadership architecture** is the deliberate design of those conditions: who holds which decisions, the roles a team pulls its people into,

the everyday processes, and what actually gets rewarded. It is less a new idea than a name for the half of the work the field has always known mattered and rarely built for.

The claim of this paper is that architecture is not an optional upgrade to leadership development. For the kind of leadership worth developing, it is necessary. And it is a practice, not a slogan: a diagnostic (authorization, then four structural pulls), and a way of working that runs in two modes: a lens woven into every engagement, and, when the work warrants it and the room authorizes it, a formal expansion of what the engagement is for.

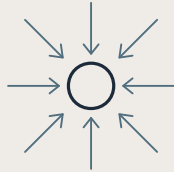
The leader is not the unit

WHERE DEVELOPMENT AIMS

THE USUAL FRAME

Develop the leader

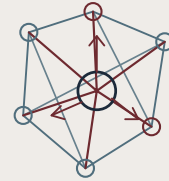
Send the person to a workshop, give them a coach, run a 360. The leader is the thing you try to change.



THE ARCHITECTURE FRAME

Develop the conditions

Change the system around them: whether there's room to try new behavior, what the team expects of the leader, and what behavior actually gets rewarded.



Most leadership development treats the leader as the unit of intervention: the one thing you change to get a different result. Send them to a workshop, give them a coach, run a 360, send them back. It's the package the whole industry sells. The premise is that if the leader learns the right thing, the leadership will follow.

The premise is wrong, and the gap between what gets learned and what gets done is the evidence. The learning is usually real. The doing doesn't land, and the reason is rarely the leader.

The transfer-of-learning literature has been making this point quietly for decades. Burke and Hutchins (2007) and Blume, Ford, Baldwin and Huang (2010) converge on a single finding: whether training sticks comes down mostly to the workplace a leader returns to, the *transfer climate*, not to the quality of the training itself. More recent work pins it down: Hughes, Zajac, Woods

and Salas (2020) find that work-environment support explains about a third of whether training transfers to the job, and of the three kinds they test (peer, supervisor, and organizational), *peer support* carries the most weight. Pfeffer and Sutton (2000) named the same problem from the management side in *The Knowing-Doing Gap*: organizations know what to do and fail at doing it, and the failure is structural, not informational.

Treating the leader without diagnosing the system is malpractice.

A leader slipping back to old habits, then, is not a personal failure. It is a condition of the environment: a rational response to a system that still rewards the old move and still pulls the old role. Which means the unit of intervention was wrong from the start.

Leadership architecture, defined

Leadership architecture is the deliberate design of the system around the leader: the decision rights, the role pulls, the processes, and the reward signals that determine whether what gets developed actually gets done.

"Leadership development" has collapsed into *leader* development, the individual as the project (Day, 2000; Pontes & Weng, 2024). The field's most established voices have begun pushing in

this direction publicly, arguing for leader-development systems rather than one-off programs (Allen & Day, 2026). But the system they describe is still the development system: assessment, pathways, training, coaching, aligned. Architecture restores the other half: the system the leader operates inside, doing more of the work, for good or ill, than the individual's competence ever could.

LEADER-DEVELOPMENT FRAME		LEADERSHIP-ARCHITECTURE FRAME
UNIT OF WORK	The person: workshop, coach, 360, return.	The system: decision rights, role pulls, processes, reward signals.
ASSUMPTION	If the leader learns the right thing, the leadership follows.	The conditions decide: what the system rewards and permits shapes whether learning holds.
COMMON FAILURE	The learning is real; the doing does not land.	Designs the conditions so the new behavior can hold once the program ends.

This is not a new service bolted onto coaching and workshops. It is a different account of what an engagement *is*: when a practice is hired to develop a team's leaders, it is being hired, whether

the contract says so or not, to change something about the system those leaders work in.

Why it is necessary, not optional

FIVE LITERATURES • ONE FINDING

01 Knowing-doing gap Knowing what to do was never the only gap. The setup is.	02 Self-efficacy Trusting you can do it is half the battle, and that trust is built by the environment.	03 Ecological systems People develop through the environments around them, not on their own.	04 Development in context What a program achieves lives in the surrounding context, not the training event.	05 Adaptive challenge Hard problems shift only when people change how they work, not when an expert fixes them from outside.
PFEFFER & SUTTON (2000)	BANDURA (1977)	BRONFENBRENNER (1979)	GEERTS (2024)	HEIFETZ (1994; 2009)

the conditions decide

ARCHITECTURE IS NECESSARY, NOT OPTIONAL

The case that architecture is necessary, rather than a nice-to-have, is that the empirical record already converges on it.

Pfeffer and Sutton's five barriers to turning knowledge into action are each an organizational *design* choice, not an individual failing: talk substituting for action, memory for thinking, fear, measurement that obstructs judgment, internal competition. Bandura's (1977) four sources of self-efficacy are mostly produced by the system, not summoned by will: the chances to succeed, the models to watch, and the encouragement a leader gets are all afforded or withheld by the conditions around them. Bronfenbrenner's (1979) ecological-systems theory established that human development is shaped by the nested environments a person lives in, not by the individual alone. Geerts (2024) brings that ecological logic to leadership development, synthesizing the field's reviews with the transfer research into a 65-strategy framework that puts impact in the ecosystem and the context, not the training event.

Read together, these are not separate findings. They are one finding from five directions: the conditions decide.

The record is not unanimous about proportion. Lacerenza and colleagues (2017), across 335 studies, found that well-designed programs still produce sizeable gains on their own. The claim here is not that the event is nothing; it is that the conditions decide whether its gains survive.

This is what Heifetz (1994; 2009) named an adaptive challenge: a problem the people in a system have to resolve by changing how they work, not one an expert can fix from outside. Developing the leader treats an adaptive challenge as a technical one. And the leadership worth developing is not a trait one person carries. It is something a system authorizes and people exercise across roles and tasks. Developing that kind of leadership *requires* working on the conditions. Architecture is not optional, because the alternative is rehearsal.

The diagnostic: authorization, then four pulls

Architecture begins with a read, not a redesign.

The read happens in a fixed order: authorization first, then four structural pulls.

Authorization is the gate. Before any condition can be worked, the read starts with a question of authority: is the work authorized to reach the conditions at all, and is whoever holds that authority willing to name the *conditions*, not the

leader, as the real problem? Without that, redesign is just theater, and the honest move is to work at the individual level and say so plainly.

Once authorization is clear, the read lands on four structural pulls. Each one is a property of the system, not a flaw in the person.

NO.	PULL	EVERYDAY FACE	WHAT IT REVEALS
01	Role suction	The team pulls a leader into the role the group needs filled, regardless of what that leader actually brings.	Reveals what the group needs someone to be, whatever this leader actually brings.
02	Decision rights	Who is authorized to make which calls, on what timeline, with what room to reverse, and what even counts as a decision here.	Expanded range dies against rights already settled by accumulation.
03	Bureaucracy	Approval chains and compliance make leading well look like making everyone's job harder; and there is no shared language for what's happening.	Without shared language, no honest structural conversation is possible.
04	Rewards	What the system actually rewards: promotion criteria, recognition, informal signals.	Reveals what it is actually committed to, whatever the values posters say.

These are ways of reading what's stuck, not a checklist to score. To read them we use BART (boundary, authority, role, and task), a disciplined lens for seeing where each one binds.

How the work happens

The work runs in two modes: a lens that is always on, and a formal step the room has to authorize.

The first is always on. In every engagement, including one framed plainly as a leadership-development program, the systems lens is in use and the separate pieces of work are connected on purpose. This is architecture done inside the existing scope, with no new contract. It is most of what the practice is.

The second is the formal step: when the structural work is big enough to warrant it, and the

authority is there to take it on, architecture is *offered* as an extension and the engagement is re-contracted. The architecture is the framing and the chaining, not the intervention. A 360 introduced in a workshop, referenced in coaching, revisited in team coaching: that chain is what makes the instrument architectural. The instrument alone is just an instrument.

01 • Read

Explore

PURPOSE

Read the system before touching it.

WHAT HAPPENS

Listening sessions, stakeholder mapping, a read of which pulls are in play, the authorization diagnostic.

BUYER-VISIBLE OUTPUT

A structural diagnosis and an authorization map.

02 • Intervene

Execute

PURPOSE

Change the leader and the conditions at once.

WHAT HAPPENS

Workshops, 1:1 and team coaching, and multi-source feedback, run concurrently with the substantive modules (3D Management, leadership stance) and system-side changes to decision rights, roles, and rewards.

BUYER-VISIBLE OUTPUT

Developed leaders and a system that permits the development.

03 • Transfer

Reinforce

PURPOSE

Make the change hold after we leave.

WHAT HAPPENS

Peer-learning structures, communities of practice, feedback loops, and the celebration and comms that keep the new move rewarded.

BUYER-VISIBLE OUTPUT

Conditions that no longer pull the old behavior back.

The chain across the phases is the architecture. And iteration compounds: each cycle reads the conditions again and tightens the fit.

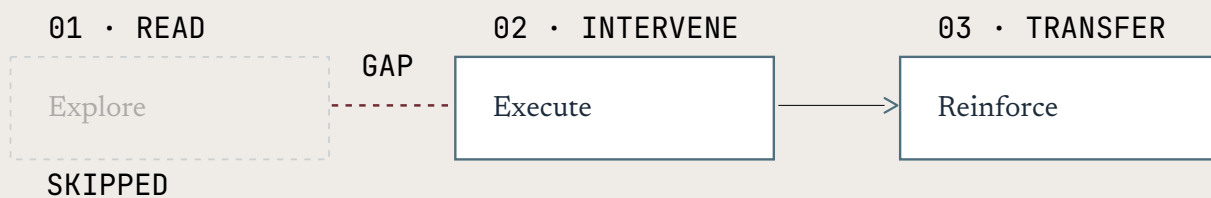
A CASE, ANONYMIZED

With an executive team, a recent leadership-development engagement carried every mark of architecture work: multi-surface delivery, the systems lens throughout, and the structural conditions surfacing as the program's own recurring themes. One kept surfacing: every real decision quietly routed around the new CEO, and no one would say it.

What it lacked was the Explore phase: no upfront listening, no stakeholder diagnosis, no authorization map. We didn't see it at the time. The missing phase became obvious only in what it cost.

The shape was deliverable. What it proved was the cost of the missing phase: everyone could see the decisions routing around the CEO, and no one had been authorized (or felt authorized) to name it. Explore is what makes the unsayable sayable – what turns what a team already sees into something it can act on.

THE ENGAGEMENT'S OPERATING MODEL



For the buyer, the chain is short. ADDIE (the standard analyze–design–develop–implement–evaluate cycle most training is built on) designs the program; architecture designs the system the program lands in, concurrently.

What this paper does not claim

A model has to say what it is not claiming. Three boundaries, named plainly.

NOT A PROOF

It reads several bodies of research together and offers a model, not a controlled study with measured outcomes.

NOT FOR EVERY ENGAGEMENT

The same read that shows when to work on the system also shows when to leave it alone.

NOT WITHOUT AUTHORIZATION

Someone has to be willing to name the conditions, not the person, as the real problem, and to act on it. That can come from bringing in outside help, or from a team deciding to turn the lens on itself. Without it, the work stays at the individual level, and honesty means saying so.

This is a framework paper, and its boundaries matter. I won't pretend this is proof: there are no controlled studies or measured outcomes here. The argument is structural: it reads across several independent literatures and offers a model and a practice, not a tested instrument. And the read could be wrong in a specific way: if transfer turns out to hinge more on the individual's motivation and ability than this paper weighs it, the architecture claim shrinks accordingly.

It does not claim that every engagement can or should become system redesign. Restraint is part

of the model, not a gap in it: the same diagnostic that shows when architecture is warranted also shows when it is not, and forcing a system engagement the conditions won't hold is its own failure mode.

And it does not claim the practitioner can do this without authorization. Architecture no one will authorize is not architecture; it is an individual-development engagement wearing borrowed language, and naming that honestly is part of the discipline.

What this means for practice

Four moves carry across any engagement that takes this seriously.

1 Diagnose the system before designing the intervention.

THE STRATEGIC HINGE

Read the decision-rights structure, where role suction lives, and which bureaucratic pains leaders absorb as personal failures, before choosing the intervention. The system's authorization architecture is doing more work than any individual's competence.

2 Engineer for transfer climate, not event quality.

IF THE SYSTEM PULLS BACK, THE WORKSHOP DID NOT MATTER

Build for the conditions that let the new behavior hold, and put peer structures first, given the evidence.

3 Move the unit from individual to team to system.

TEAM COACHING IS A REAL UPGRADE; SYSTEM IS THE NEXT MOVE

Let the work follow the problem's real unit of action, not remain at the individual level by habit.

4 Do not pretend AI changes the diagnosis.

THE STRUCTURAL PROBLEM DOES NOT CHANGE BECAUSE THE DELIVERY CHANNEL DID

AI used for individual coaching mostly scales the same frame; AI used for system diagnosis is the frontier.

References

- Allen, S. J., & Day, D. V. (2026, January 6). The six elements of a successful leadership development system. *Big Think+*. <https://bigthink.com/plus/the-six-elements-of-a-successful-leadership-development-system/>
- Baldwin, T. T., & Ford, J. K. (1988). Transfer of training: A review and directions for future research. *Personnel Psychology*, 41(1), 63–105.
- Bandura, A. (1977). Self-efficacy: Toward a unifying theory of behavioral change. *Psychological Review*, 84(2), 191–215.
- Blume, B. D., Ford, J. K., Baldwin, T. T., & Huang, J. L. (2010). Transfer of training: A meta-analytic review. *Journal of Management*, 36(4), 1065–1105.
- Bronfenbrenner, U. (1979). *The ecology of human development*. Harvard University Press.
- Burke, L. A., & Hutchins, H. M. (2007). Training transfer: An integrative literature review. *Human Resource Development Review*, 6(3), 263–296.
- Day, D. V. (2000). Leadership development: A review in context. *The Leadership Quarterly*, 11(4), 581–613.
- Geerts, J. M. (2024). Maximizing the impact and ROI of leadership development. *Behavioral Sciences*, 14(10), 955.
- Heifetz, R. A. (1994). *Leadership without easy answers*. Belknap Press of Harvard University Press.
- Heifetz, R. A., Grashow, A., & Linsky, M. (2009). *The practice of adaptive leadership*. Harvard Business Press.
- Hughes, A. M., Zajac, S., Woods, A. L., & Salas, E. (2020). The role of work environment in training sustainment. *Human Factors*, 62(1), 166–183.
- Lacerenza, C. N., Reyes, D. L., Marlow, S. L., Joseph, D. L., & Salas, E. (2017). Leadership training design, delivery, and implementation: A meta-analysis. *Journal of Applied Psychology*, 102(12), 1686–1718.
- Pfeffer, J., & Sutton, R. I. (2000). *The knowing-doing gap*. Harvard Business School Press.
- Pontes, T., & Weng, J. (2024). Leadership development research and scholarship. *New Directions for Student Leadership*, 2024(183), 43–49.

About the author

John Weng, PhD is the founder and principal of Bricolas, a leadership-development practice in San Diego focused on executive coaching, team development, and supervision for senior leaders, partners, and founders. He is also a leadership coach and consultant at the University of Cambridge Judge Business School, supporting participants of the General Management and Advanced Leadership Programmes.

PhD in Leadership Studies (University of San Diego, 2024); ICF Professional Certified Coach (PCC) and Advanced Certificate in Team Coaching (ACTC); Certificate of Advanced Study in Coaching Supervision. The practice draws on group-relations methodology, the adaptive-leadership tradition, developmental scholarship, and coaching-supervision practice.

Bricolas

Bricolas is composed practice — moves drawn from a wide repertoire of frameworks, designed for this team, this leader, this room as it actually is.

San Diego, California

hello@bricolas.com · bricolas.com

CITE AS

Weng, J. (2026). *Leadership Architecture: Designing the conditions leaders grow in*. Bricolas. v1.0 · 2026. <https://doi.org/10.5281/zenodo.21923578>

© 2026 Bricolas LLC.

Released under CC BY-NC-SA 4.0. creativecommons.org/licenses/by-nc-sa/4.0/