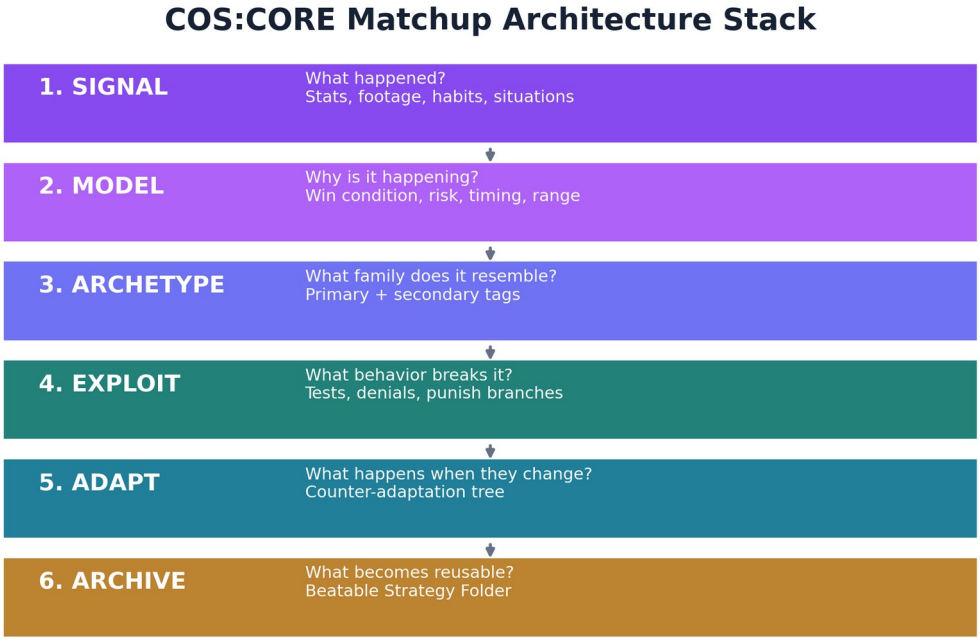
COS:CORE

MATCHUP ARCHITECTURE

HANDBOOK

Archetype Deconstruction • Beatable Strategy Folders • Live Set Adaptation



BOOK ONE — THE MATCHUP SKELETON

A visual coaching system for turning player behavior into reusable, testable counter-strategies. Built around COS:CORE Cognition, Originality, Response and the Tell–Show–Do teaching architecture.

Coach Cosmic Framework • v1.0 • 2026

FOUNDATION

1. The Core Thesis

Do not memorize opponents. Model the architecture that produces their decisions.

A matchup becomes easier to coach when the opponent stops being “a difficult player” and becomes a system with inputs, habits, dependencies and failure states. The goal is not to label someone once. The goal is to build a folder that tells you what to test, what to deny, what to punish and what to watch when they adapt.

> **THE CENTRAL RULE**
>
> An archetype is useful only if it predicts behavior. A counter-folder is useful only if it creates a repeatable advantage.

COS:CORE translates the problem through three lenses:

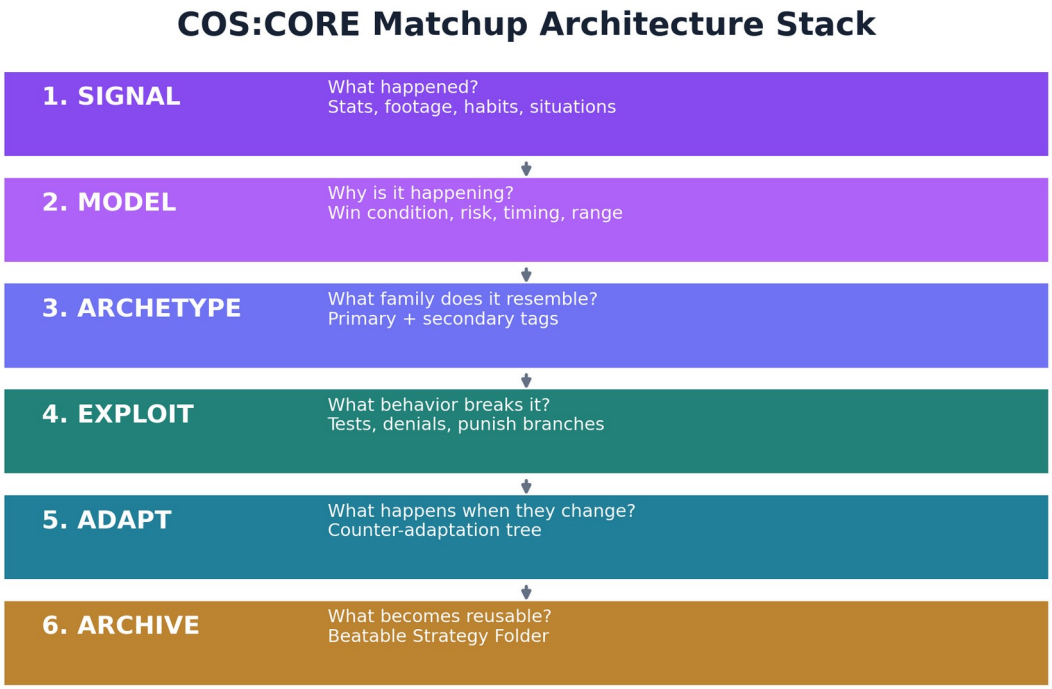
COGNITION READ Recognize the situation before reacting to the move. **WIN CONDITION:** Identify the opponent’s preferred state. **BEAT IT:** Name the dependency that makes that state possible. **WATCH FOR:** False positives from one-off actions.	**ORIGINALITY** CREATE Generate more than one valid answer to the same problem. **WIN CONDITION:** Build branches instead of a single “counter.” **BEAT IT:** Change timing, range or reward while preserving the objective. **WATCH FOR:** Overcomplicating a simple solution.
RESPONSE EXECUTE Deliver the chosen answer consistently under pressure. **WIN CONDITION:** Make the counter automatic enough to survive a real set. **BEAT IT:** Drill the trigger, not just the move. **WATCH FOR:** Knowing the answer but missing the moment.	**CORE** INTEGRATE Connect reading, execution, review and adaptation into one system. **WIN CONDITION:** Archive the lesson so it can transfer to new opponents. **BEAT IT:** Update folders after every meaningful adaptation. **WATCH FOR:** Treating notes as static truth.

The handbook therefore treats every matchup as a loop: observe → classify → test → break → watch → update. The folder is the memory. The live set is the laboratory.

SYSTEM MAP

2. The Six-Layer Matchup Architecture

Every good read should travel through the same architecture before it becomes advice.



Layer	Question	Output
Signal	What repeatedly happened?	Observable behavior; no interpretation yet.
Model	What condition seems to cause it?	Range, turn, resource, risk or timing hypothesis.
Archetype	What family best predicts the pattern?	Primary tag + one secondary tag.
Exploit	What low-cost behavior stresses the dependency?	Proof test and denial plan.
Adapt	What changes after the first answer is shown?	Counter-adaptation branch.
Archive	What should be reusable next time?	Versioned Beatable Strategy Folder.

ARCHITECTURAL DISCIPLINE

Never jump from Signal directly to Exploit. “He used a low twice, so low-parry him” is a guess. “He uses the low after blocked jab pressure to re-open offense; interrupting the reset removes the loop” is a model.

SCAN

3. Observation: Build the Fingerprint Before the Folder

Separate evidence from interpretation so the coaching system does not hallucinate patterns.

Dimension	What to log	Typical question
Pace	How often they initiate or reset offense	Do they create the first interaction or wait?
Range	Where they prefer to operate	What distance makes their plan easiest?
Turn behavior	Challenge, block, evade, retaliate	What do they do immediately after losing initiative?
Risk	Safe pokes vs hard commitments	How much variance are they willing to create?
Defense	Block, movement, throw defense, panic options	Which defensive layer is carrying them?
Conversion	Damage, wall, oki, resource spend	What happens after they win one interaction?
Adaptation	How quickly options change after failure	Do they branch or repeat?
Clutch	Behavior while behind or near KO	Do they stabilize, gamble or accelerate?

Use three evidence buckets:

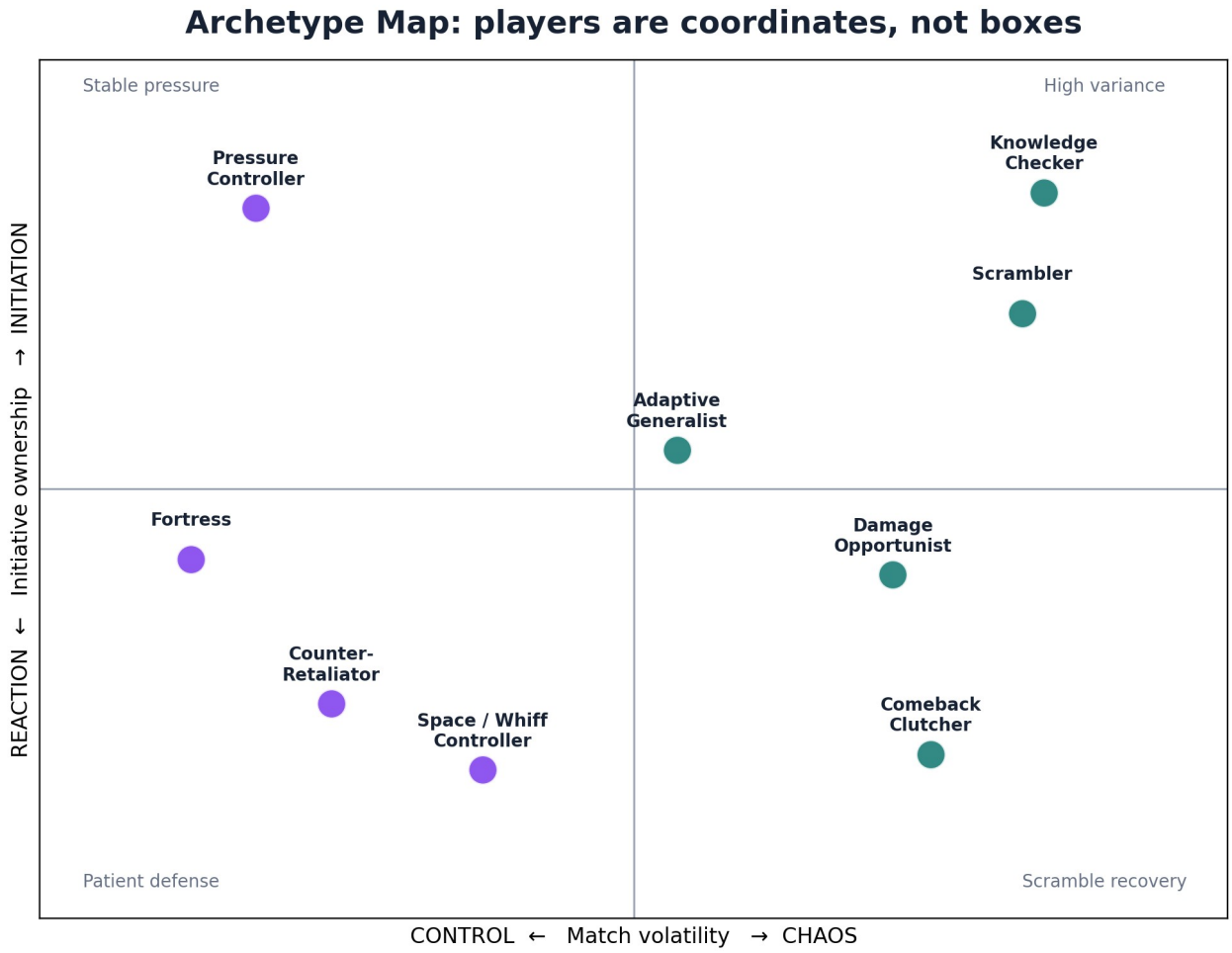
- FACT — directly observed: “After blocked jab, he sidestepped right 4 of 6 times.”
- INFERENCE — likely mechanism: “He is using movement as his preferred turn-escape.”
- TEST — cheapest way to prove it: “Delay a tracking mid after the jab and see whether the escape repeats.”

> **THE 3-CONFIRM RULE**
>
> Do not promote a habit into a folder dependency until it appears in at least three comparable situations, or one high-confidence sequence is independently confirmed by stats/footage.

CLASSIFY

4. Archetypes Are Coordinates, Not Boxes

A player can be 60% Pressure Controller, 25% Counter-Retaliator and 15% Scrambler. Hybrid tags are expected.



Classify across six axes before assigning the archetype label:

Axis	Left pole	Right pole
Initiative	Reactive	Initiating
Volatility	Control / low variance	Chaos / high variance
Range	Close-space dependent	Long-space dependent
Defense	Guard / absorb	Movement / evade
Toolset	Specialized / narrow	Broad / varied
Recovery	Stabilize while behind	Explode while behind

FOLDER NAMING SYNTAX

[Primary Archetype] + [Secondary Trait] + [Critical Dependency]. Example: “Pressure Controller / Retaliator — requires first-turn ownership.”

FOLDER LIBRARY

5. Archetype Library — Control Families

These archetypes want the match to become predictable on their terms.

PRESSURE CONTROLLER CONTROL + INITIATE

High activity and sustained advantage. Wants the opponent defending repeatedly.

WIN CONDITION: Own the first stable turn, then loop pressure/oki.
BEAT IT: Deny the entry or escape before the second layer becomes stable.
WATCH FOR: Mistaking high button frequency for real control.

COUNTER-RETALIATOR CONTROL + REACT

Blocks or waits, then immediately reclaims initiative.

WIN CONDITION: Make the opponent overextend into a predictable turn ending.
BEAT IT: Use safer endings, delayed timing and baited retaliation.
WATCH FOR: Giving them obvious “your turn is over” cues.

SPACE / WHIFF CONTROLLER RANGE + REACT

Wins through distance, movement and whiff punishment.

WIN CONDITION: Make you swing at bad range, then convert the miss.
BEAT IT: Walk them backward, feint commitment and claim space without swinging.
WATCH FOR: Chasing the player instead of the position.

FORTRESS DEFENSE + CONTROL

Guard-heavy, patient and difficult to open with ordinary offense.

WIN CONDITION: Let you exhaust options, then punish mistakes.
BEAT IT: Use throws, chip/resource pressure, plus-state layering and safe lows.
WATCH FOR: Assuming “patient” means passive.

CONTROL FAMILY RULE

Do not fight control with more predictable control. Either deny the setup before it stabilizes or introduce a variable the archetype does not cover well.

FOLDER LIBRARY

6. Archetype Library — Volatility Families

These archetypes become more dangerous when the match becomes messy, unfamiliar or high-variance.

SCRAMBLER CHAOS + INITIATE

Frequent challenges, unusual timings and rapid state changes.

WIN CONDITION: Prevent the opponent from settling into rehearsed structure.
BEAT IT: Slow the match, protect the life lead, force clean repeated defense.
WATCH FOR: Trying to "out-crazy" the scrambler.

DAMAGE OPPORTUNIST CHAOS + REACT

Accepts neutral losses because one read can reverse the round.

WIN CONDITION: Create one high-reward interaction.
BEAT IT: Reduce launchable risks and make them win many small decisions.
WATCH FOR: Greedy closers when ahead.

KNOWLEDGE CHECKER CHAOS + INITIATE

Tests unfamiliar strings, setups, stance routes or uncommon branches.

WIN CONDITION: Force incorrect answers before the opponent identifies the rule.
BEAT IT: Pause the autopilot; identify the trigger and reproduce it in lab.
WATCH FOR: Calling every unfamiliar option a gimmick.

COMEBACK CLUTCHER CHAOS + RECOVERY

Performance improves when behind; thrives on desperate resource states.

WIN CONDITION: Use opponent urgency against them.
BEAT IT: Keep the closeout boring: no extra risks, no ego challenges.
WATCH FOR: Mistaking "almost dead" for "already beaten."

ADAPTIVE GENERALIST BROAD + FLEX

No single branch stays dominant for long; changes pace, tools and defense.

WIN CONDITION: Force the opponent to solve multiple valid answers.
BEAT IT: Attack the invariant — resource, range, execution burden or risk appetite.
WATCH FOR: Chasing individual moves instead of the common dependency.

SPECIALIST / FLOW SYSTEM DEPTH + REPEAT

A narrow toolset is deeply optimized and repeatedly recreated.

WIN CONDITION: Get back to the same high-confidence situations.
BEAT IT: Break the re-entry route; force unfamiliar branches.
WATCH FOR: Underestimating a "simple" system that is highly polished.

ARCHIVE

7. The Beatable Strategy Folder

A folder is not a list of counters. It is a compact operational model of how to beat one architecture.

Every folder should turn observation into a repeatable game plan

COUNTER FOLDER

01 FINGERPRINT	What repeats? Pace, range, defense, risk, resource use.
02 WIN CONDITION	What must they establish before their strongest loop starts?
03 PROOF TESTS	Low-cost tests that confirm the archetype instead of guessing.
04 DENIAL PLAN	What do we remove first: turn, range, wall, throw, timing, resource?
05 PUNISH TREE	If they answer A → response B. If they adapt B → response C.
06 DRILLS + NOTES	Lab reps, set notes, evidence threshold, version history.

Folder field	Minimum standard
Fingerprint	3–6 repeatable signals; separate facts from inferences.
Primary win condition	One sentence: what state must they establish?
Dependency	What must remain true for that state to keep working?
Proof tests	2–3 low-risk tests that can confirm the dependency live.
Denial plan	The first thing to remove before hunting damage.
Punish tree	At least one branch for their most likely adaptation.
Fail-safe	What to do when uncertain: safest reset state.
Drills	Trigger → response reps, not disconnected move practice.
Version	Date, opponent/context, evidence strength, last revision.

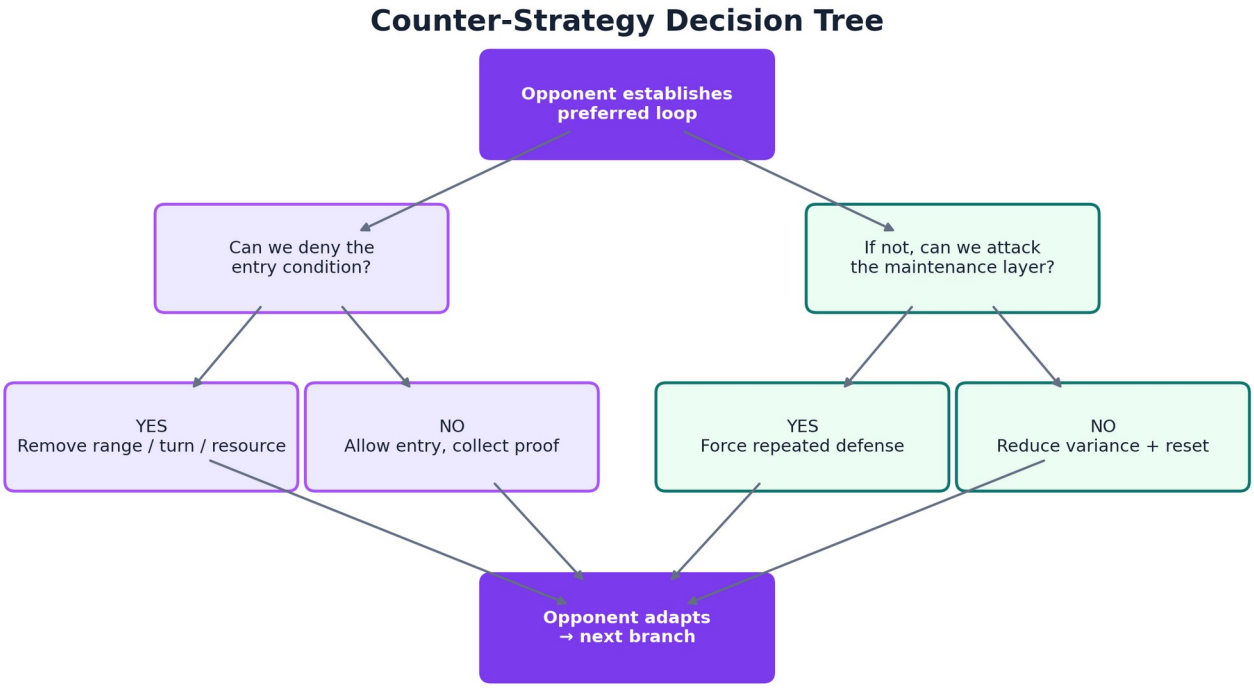
BEATABLE ≠ SOLVED

A strategy is “beatable” when its dependency can be reliably stressed. A player is never permanently solved because humans adapt. Version the folder, not the person.

BREAK

8. Build the Counter Tree

The best counter-strategy has a branch before the opponent shows the adaptation.



Counter priority should usually follow this order:

Priority	Attack this first	Why
1	Entry condition	If the loop never starts, later layers do not matter.
2	Maintenance condition	If entry is inevitable, make the loop expensive to sustain.
3	Conversion	Reduce the reward they receive from successful interaction.
4	Recovery	Prevent the archetype from resetting after failure.
5	Adaptation burden	Force them into branches they have shown least often.

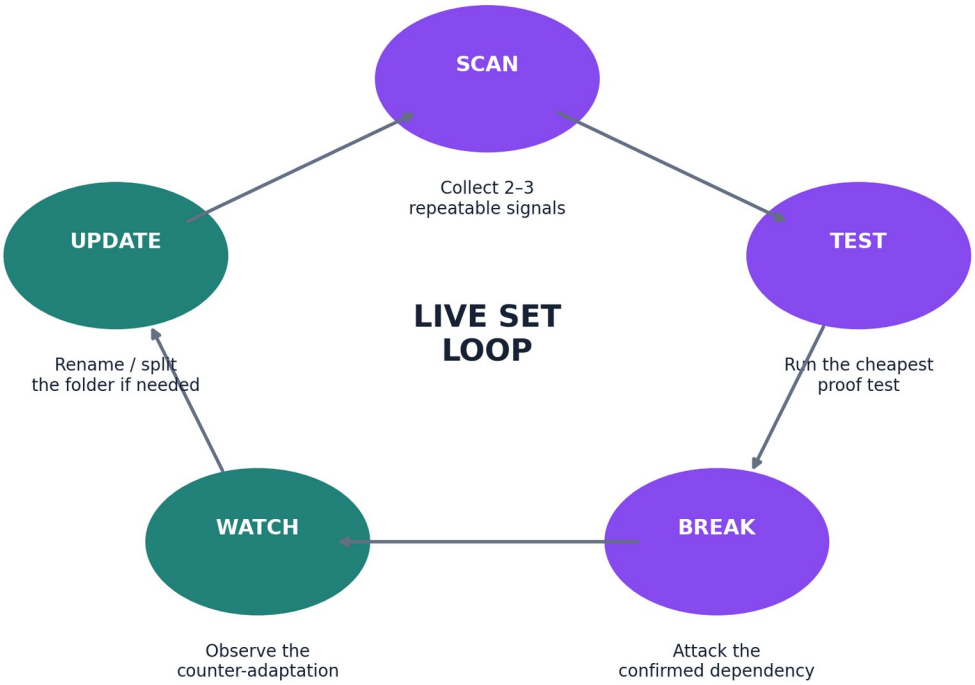
LOW-COST TESTING

A proof test should reveal information even when it loses. The ideal test risks a small amount of health to discover a habit worth an entire set.

EXECUTE

9. Live Set Protocol

The folder must survive real-time pressure. This loop keeps the coach/player from drowning in information.



Set phase	Primary objective	Coach question
Opening game / round	Collect signals	What repeats when nothing has been conditioned yet?
First test	Confirm one dependency	What is the cheapest safe way to make the habit reveal itself?
Exploit window	Repeat the answer	Will they keep paying the same tax?
Adaptation watch	Stop autopilot	What changed after we showed the counter?
Reset	Return to stable state	If we are unsure, what position gives us time to re-read?
Archive	Name the branch	What belongs in the folder after the set?

ONE CHANGE AT A TIME

When a set goes wrong, change one system at a time. If you simultaneously alter movement, timing, offense and defense, you will not know which correction actually worked.

COACHING

10. Teach the Matchup with Tell–Show–Do

A folder becomes curriculum only when the player can recognize and execute it independently.

TELL EXPLAIN Name the architecture, dependency and trigger in plain language. **WIN CONDITION:** Player can predict what the opponent wants before seeing the move. **BEAT IT:** Use one sentence and one decision rule. **WATCH FOR:** Frame-data dumping without a decision model.	**SHOW** DEMONSTRATE Use footage, diagrams or lab examples to reveal the trigger. **WIN CONDITION:** Player can visually distinguish the target situation from similar ones. **BEAT IT:** Show success, failure and the opponent's adaptation. **WATCH FOR:** Only showing perfect examples.
DO PERFORM Recreate the trigger under increasing uncertainty. **WIN CONDITION:** Correct response occurs without coach prompting. **BEAT IT:** Start scripted, then randomize timing/branches. **WATCH FOR:** Practicing the move without practicing recognition.	**TRANSFER** OWN Player names the folder and applies it to a new opponent. **WIN CONDITION:** The lesson becomes a reusable mental model. **BEAT IT:** Ask the player to teach the read back. **WATCH FOR:** Dependency on the coach's live callouts.

A practical drill ladder:

Level	Drill
1 — Static	Opponent reproduces the exact trigger every repetition.
2 — Binary	Opponent alternates target trigger with one decoy.
3 — Variable	Timing, range or branch changes while the dependency remains.
4 — Set simulation	Trigger appears naturally inside a first-to-X set.
5 — Transfer	New character/player expresses the same archetype through different moves.

CASE STUDY

11. Case Study — Coach Cosmic

Veteran Tekken architecture: Pressure Controller / Counter-Retaliator with a specialized, stable loop.

Signal	Value	Interpretation for folder-building
Attack Frequency	25/25	Very high initiation/activity.
Dominance	24/25	Offense often becomes sustained control.
Block	25/25	Guard is a major defensive foundation.
Retaliation	25/25	Strong turn-reclamation behavior.
Composure	21/25	Stable defense under pressure.
Throw Escape	2/25	Clear coverage gap worth immediate proof-testing.
Versatility	12/25	More specialized than broad; force unfamiliar branches.

PRIMARY ARCHETYPE

PRESSURE CONTROLLER / RETALIATOR — requires first-turn ownership and benefits from predictable turn endings.

Beatable Strategy Folder — COSMIC CONTROL:

- PROOF TEST 1: Introduce throws after he settles into guard-based defense. Make him prove the 2/25 throw-escape signal is outdated.
- PROOF TEST 2: Vary the end timing of pressure so Retaliation 25 cannot fire on a fixed rhythm.
- DENIAL: Do not donate the first stable pressure loop. Challenge the entry condition with spacing, lateral movement or pre-emptive position control.
- BRANCH: If he broadens options to answer the first denial, stop forcing the original read and attack the new execution/precision burden instead.
- FAIL-SAFE: When uncertain, reset to neutral rather than contesting his preferred close-range rhythm repeatedly.

COACHING TARGET

Do not “beat Law.” Beat the architecture: first-turn ownership → stable pressure → guarded defense → retaliation → dominance.

CASE STUDY

12. Case Study — Omnivera / James Edwards

Elite cross-game learner: Adaptive Scrambler / Comeback Clutcher whose fighting-game intelligence is ahead of his Tekken-specific database.

Signal	Value	Interpretation for folder-building
Attack Frequency	25/25	Actively seeks interaction despite being new to Tekken.
Retaliation	25/25	Transferable turn-reading is already strong.
Comeback	24/25	Very dangerous after falling behind.
Versatility	23/25	Broad exploration and option switching.
Stage Use	22/25	Quickly incorporates environment into conversions.
Dominance	14/25	Harder time keeping control once earned.
Composure	16/25	Extended stable defense is a productive stress test.

PRIMARY ARCHETYPE

ADAPTIVE SCRAMBLER / COMEBACK CLUTCHER — strongest when volatility rises and the opponent gives him multiple adaptation opportunities.

Beatable Strategy Folder — OMNI VOLATILITY:

- PROOF TEST 1: Take a small life lead and deliberately reduce volatility. See whether he begins manufacturing risk to reopen the round.
- PROOF TEST 2: Force several consecutive clean defensive decisions instead of giving one big read opportunity.
- DENIAL: Do not chase the KO when he is behind. Remove the comeback environment by keeping the closeout boring.
- BRANCH: His Versatility 23 means the first counter may work only briefly. Track what stays invariant when the move selection changes.
- FAIL-SAFE: When the match becomes chaotic, reset spacing and tempo instead of matching his pace.

COACHING TARGET

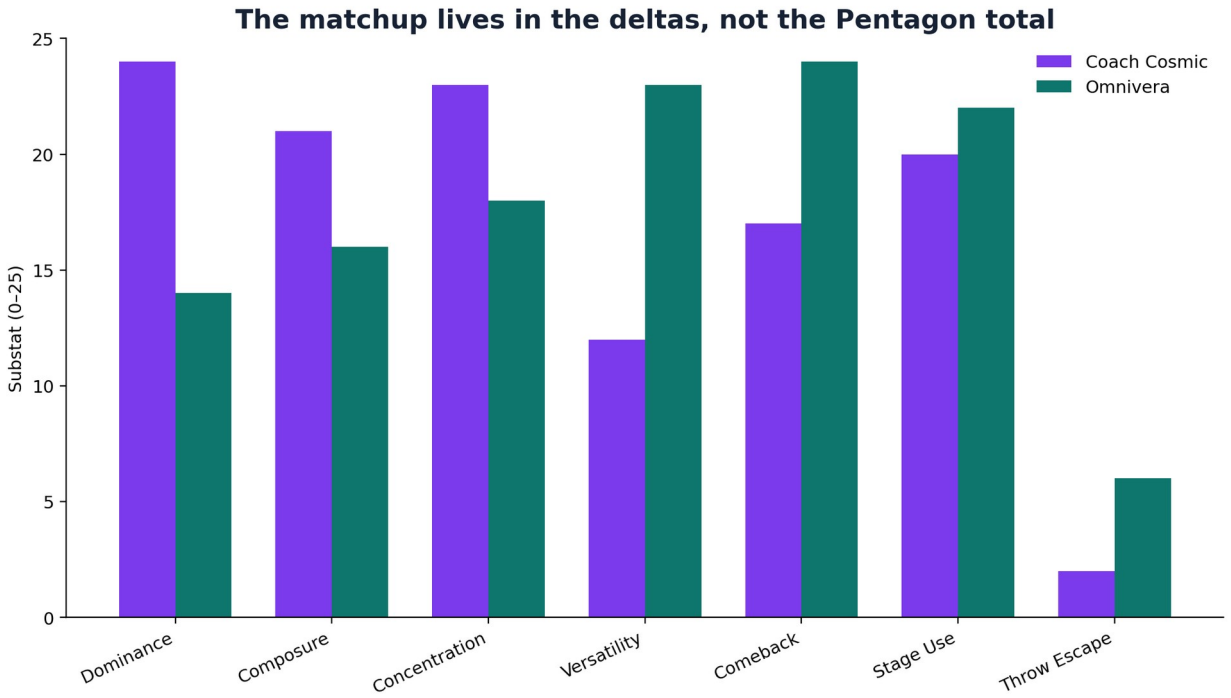
Do not “beat Paul.” Beat the architecture: broad experimentation → reactive adaptation → high-variance reversal → comeback pressure.

MATCHUP MODEL

13. Cosmic vs. Omnivera — Architectural Collision

The Pentagon is most useful when it exposes the difference between two systems, not when it is treated as an overall score.





Collision point	Cosmic objective	Omnivera objective
Tempo	Stabilize the match around a known loop.	Break rhythm before the loop becomes stable.
Defense	Block, remain composed, retaliate.	Create varied threats and force coverage checks.
When ahead	Maintain dominance; avoid unnecessary variance.	Create volatility and engineer a comeback state.
When behind	Use concentration and structured recovery.	Lean into breadth, stage use and comeback strength.
Long set	Expand branches so specialization is not solved.	Convert experimentation into repeatable dominance.

THE MATCHUP SKELETON

Cosmic wins by making the set smaller and more predictable. Omnivera wins by making the set wider, more variable and harder to model.

WORKSHEET

14. Blank Archetype Worksheet

Use this page before creating the actual Counter Folder.

Field	Coach notes
Opponent / Character	
Context / Date / Mode	
Primary archetype	
Secondary archetype	
Preferred range	
Preferred pace	
Preferred defense	
Risk profile	
Primary win condition	
Dependency hypothesis	
Evidence #1	
Evidence #2	
Evidence #3	
Cheapest proof test	
What would falsify the hypothesis?	
Most likely adaptation	

QUALITY CHECK

If you cannot write what would falsify the archetype hypothesis, you are describing a belief—not testing a model.

WORKSHEET

15. Blank Beatable Strategy Folder

Archive only what you can explain, test and train.

Folder field	Write-in
FOLDER NAME	[Primary] / [Secondary] — [Critical dependency]
FINGERPRINT	3–6 signals that identify this architecture
WIN CONDITION	What state must they establish?
ENTRY CONDITION	What allows the state to begin?
MAINTENANCE CONDITION	What keeps it working?
PROOF TEST 1	
PROOF TEST 2	
PROOF TEST 3	
DENIAL PLAN	
PUNISH / RESPONSE TREE	If A → B; if B adapts → C
FAIL-SAFE RESET	What do we do when the read is uncertain?
TELL	One-sentence explanation
SHOW	Footage / lab example
DO	Trigger-response drill
TRANSFER TEST	New opponent / character expressing same archetype
VERSION NOTES	Date • evidence strength • changes

FINAL PRINCIPLE

The goal is not a giant encyclopedia of moves. The goal is a library of opponent architectures that become easier to recognize every time you see them in a new form.