

LuAir Governance<sup>•</sup>

# Node Canvas

The blanks that matter. Filled in, this is your node.

Governance, installed.

NODE NAME

---

DATE STARTED

---

LINE OF BUSINESS / SCOPE

---

VERSION

---

ORCHESTRATOR (FIRST NODE? YOUR SPONSOR)

---

# 01

## People & roles

When something goes wrong with your most important data,  
who handles it?



OUR NODE IS (THE BOUNDARY: ONE ROLE, ONE TEAM, A DEPARTMENT)

---

OUR ORCHESTRATOR — WHO THE NODE ESCALATES TO

---

Operator — backup — sponsor: who holds each role is on the staffing record page

OUR NODE AUTHORITY — DECIDED ALONE / REQUIRES THE ORCHESTRATOR'S APPROVAL

---

OPERATOR QUALIFICATIONS (PERSON: SKILLS AND EXPERIENCE — AGENT: MODEL AND EFFORT)

---

FIRST RULES, AND WHO OWNS EACH (RULE NUMBER — RULE — OWNER) · RULE WORDING: SEE THE PLAYBOOK GLOSSARY

---

FOR A DEPARTMENT NODE, THE NODES INSIDE THIS NODE EACH FILL THEIR OWN CANVAS: NODE — OPERATOR — ESCALATES TO (USUALLY: THIS NODE'S OPERATOR)

---

Working when: The node knows who it is: what's inside the circle, and who the orchestrator is / Operator and backup are named on the staffing record — and know it · full list in the playbook

# 02

## Intake

When someone discovers a data problem, where does it go?



OUR INTAKE QUEUE IS (TOOL / PLACE)

---

WATCHED BY

---

ANNOUNCED ON

---

Every item logged with: date — who asked — one line — status

HOW REQUESTS ARRIVING OUTSIDE THE QUEUE GET CAPTURED

---

Working when: Anyone on the team, asked where problems go, gives the same answer / Everything open is on the list — nothing lives only in chat · full list in the playbook

# 03

## Prioritization

When multiple data problems come up at the same time, how do you decide which one to fix first?



CYCLE REVIEW: DAY / FREQUENCY

---

WHO ATTENDS

---

COMMITMENT CEILING PER CYCLE

---

COMMITTED LIST LIVES AT

---

CYCLE REVIEW EXCEPTION: MID-CYCLE ADDITIONS ARE THE SPONSOR'S CALL — SPONSOR INITIALS

---

HOW REQUESTERS HEAR COMMITTED / DEFERRED

---

Working when: The review happens on schedule, even when there's little to review / Each cycle has a dated committed list, and it's short · full list in the playbook

# 04

## Escalation path

When a data issue is too big for whoever found it, what happens next?



OUR ORCHESTRATOR (FROM PEOPLE & ROLES)

---

The three triggers: no rule covers it — beyond the node's authority — aged past its window

OUR AGING WINDOW (ESCALATE AFTER)

---

HOW ESCALATIONS ARE SENT UP (MESSAGE, TAG, CHECK-IN)

---

FIRST REAL ESCALATION — WHAT IT WAS, AND THE DATE

---

ANSWERS THAT BECAME RULES (ANSWER — RULE NUMBER — NEW RULE — OWNER)

---

---

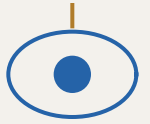
---

Working when: Everyone in the node knows: no rule, no guessing — it goes to the orchestrator / The orchestrator has agreed how escalations arrive · full list in the playbook

# 05

## Awareness

Do you know what your most important data is, what shape it's in, and what rules apply to it?



STATE IS VISIBLE AT

---

STATUSES WE USE

---

CRITICAL DATA: ATTRIBUTE — WHERE IT LIVES — WHOSE RULE GOVERNS IT

---

---

---

---

DATA CONDITION CHECK: RHYTHM / CHECKED AGAINST WHICH RULE / FINDINGS NOTED AT

---

FINDING REACTION: WHAT THE NODE DOES WHEN THE DATA DOES NOT MEET ITS RULE

---

Working when: Anyone can answer 'what's stuck and why' by looking, not by asking around / The critical data list exists, with owners against rules · full list in the playbook

# 06

## Definition of done

How do you know when a data problem is actually fixed?



KINDS OF WORK WE SUPPORT — AND WHAT DONE MEANS FOR EACH (KIND — DEFINITION — OWNER)

---

---

---

---

WORK METHODS: KIND OF WORK — WHERE THE WRITTEN STEPS LIVE — OWNER (AGENT OPERATOR: EACH IS AN AGENT WORK METHOD)

---

---

---

---

Exit condition matched before work starts — confirmed by name at close — no match? it escalates

WHO CONFIRMS DONE

---

REOPENS NOTED WITH REASON

---

Working when: Each kind of work the node supports has a written, owned definition of done / Every open item carries its definition from the moment work starts · full list in the playbook

# 07

## Communication cadence

How does your team find out about data problems?



WHO HEARS FROM THE NODE — HOW OFTEN — THROUGH WHAT

---

---

---

The three-line report: worked on / resolved / stuck (who it waits on)

TEAM CHECK-IN: DAY / CHAIRED BY

---

DECISIONS TRACED AT

---

FIRST THREE REPORT DATES

---

Working when: Three consecutive cycles of the report have gone out on time / The sponsor can say when they'll next hear from the node without asking · full list in the playbook

# The node is live

Seven proofs, witnessed on real work. Initial and date each.

01 A question found its owner without a meeting

INITIALS / DATE

02 Work arrived in one queue, from more than one person

INITIALS / DATE

03 A cycle committed little and finished it

INITIALS / DATE

04 Something escalated and landed with a name

INITIALS / DATE

05 'What's stuck?' was answered by looking

INITIALS / DATE

06 Something closed against its written exit condition

INITIALS / DATE

07 Three reports went out on rhythm, one of them quiet

INITIALS / DATE

SUPERVISED WEEKS (DATES)

REAL CASES HANDLED (MIN. FIVE)

SPONSOR SIGNATURE

COACH CERTIFICATION (OR SPONSOR, SELF-RUN)

# Staffing record

Who holds each role now. Update this page when the person or agent changes — the rest of the canvas stays as written.

| ROLE           | HELD BY | SINCE | QUALIFICATIONS MET |
|----------------|---------|-------|--------------------|
| Operator       |         |       |                    |
| Backup         |         |       |                    |
| Sponsor        |         |       |                    |
| Decision owner |         |       |                    |
| Decision owner |         |       |                    |
| Decision owner |         |       |                    |

# Where this leads

## **Know your shape first**

The readiness assessment — seven questions, a few minutes: [luair.net/assessment.html](https://luair.net/assessment.html)

## **The worked use case**

Node Zero: a complete example of a node staffed by an AI agent — and new use cases and method updates as they're published: [luair.net/node-zero.html](https://luair.net/node-zero.html)

## **Questions, or a coached install**

[contact@luair.net](mailto:contact@luair.net) · [luair.net](https://luair.net)

Governance, installed.