WHY YOUR ARCHITECTURE PROGRAM WILL FAIL

– and what to do about it



Roland Woldt | BOC Group Webinar November 2026

TWO TYPES OF ORGANIZATIONS

"Give me the licenses and go away. We'll figure out the rest."

- Tool treated as drawing software
- Three people use it. Nobody else.
- Architecture Review Board = "committee of no"
- 18 months later: shelf ware
- Budget pulled. Program cancelled.

VS.

"This is a capability investment. We own it. We drive it."

- Business-led, not IT-led
- Clear strategy before touching the tool
- Governance defined upfront
- Adoption planned from day one
- Measurable outcomes, not just deliverables

THE FAILURE PATTERN

Three warning signs you've seen before

1

Nobody can answer the basic question

"What is our architecture program actually for?" Ask ten people. Get ten different answers — or silence. When the program has no clear purpose, every stakeholder has a different expectation. And nobody can be held accountable for success.

2

The tool is used by three people

Typically in IT. Occasionally in one business unit. Everyone else either doesn't know it exists or has decided it's not relevant to their work. The tool becomes a personal productivity tool for analysts instead of organizational infrastructure.

3

The Architecture Review Board is called “The Committee of No"

This one is almost universal. When architects are seen as gatekeepers rather than enablers, they get excluded from decisions. By the time they're consulted, choices are already made and the "review" is a formality.

WHAT A CAPABILITY LOOKS LIKE



Foundation

Getting started and run an Enterprise Architecture Program.

Examples of sub-capabilities are:

- Platform setup
- Governance design
- Method config
- Enablement, etc.

How do I get started / set up my EA?



Developm. & Analysis

This capability includes the creation and analysis of architecture views, such as

- Business Architect.
- App Architecture
- Data Architecture
- Service Architecture
- Infrastructure Arch.
- Risk & Controls Architecture

What is my EA?



Architecture Usage

After definition of the arch. views, it is time to use it and implement projects based on them.

This category includes capabilities, such as Bus./IT alignment, solution portfolio management, and solution architecture.

What do I do with my EA?



Visibility

Once you start building your architecture and implementing programs and projects, the question “where are we?” comes up.

The capabilities in this category will answer this question, for example, with real-time dashboards, reports, or EA health checks.

How do I know what is going on in my EA?



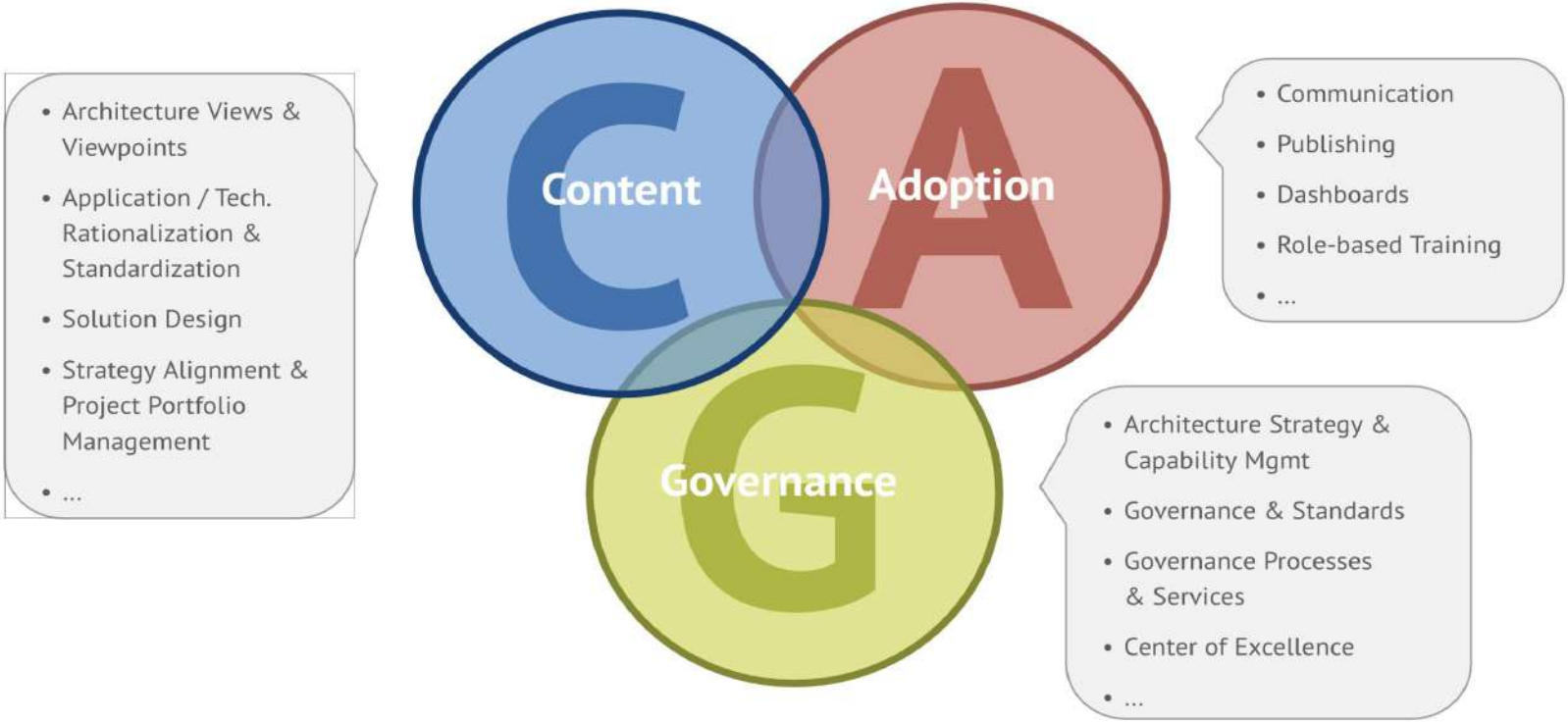
EA Methodology

One problem with EA is that plenty of different frameworks and interpre-tations have been established, which are not necessarily in sync with each other.

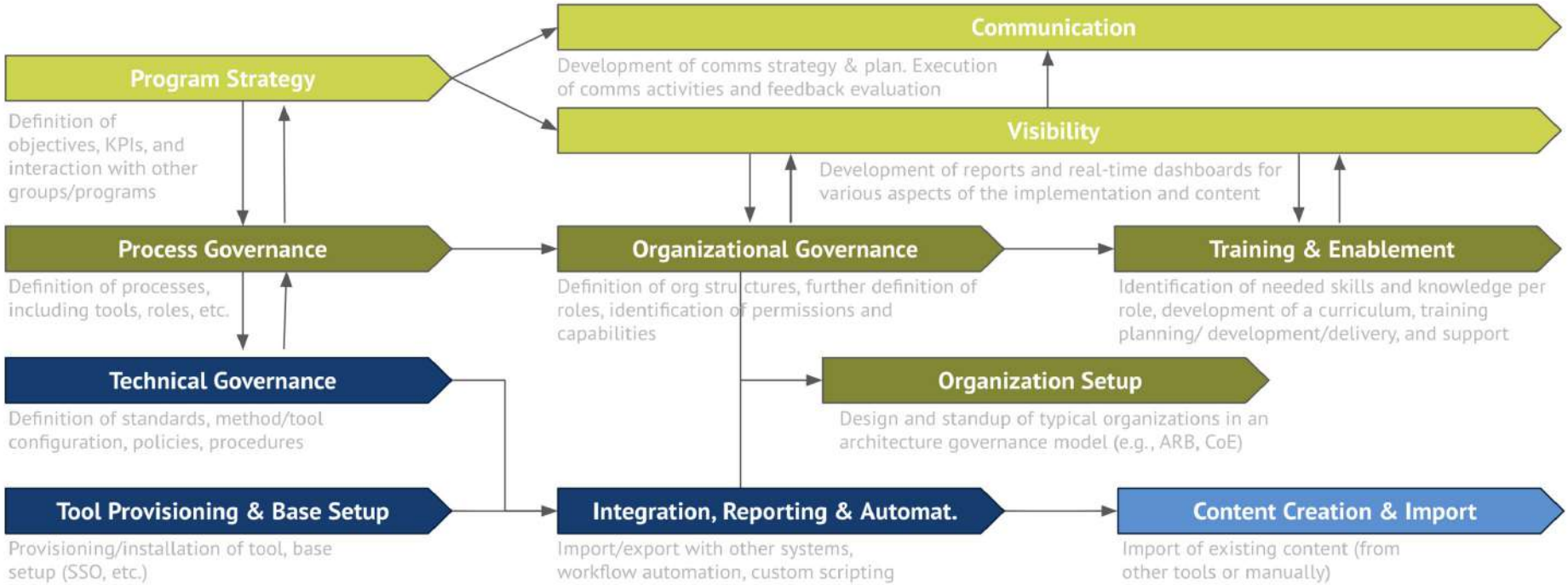
This capability set will create clarity through method analysis, implementation, and mentoring.

What do I have to know about EA?

3 AREAS OF SUCCESSFUL IMPLEMENTATIONS



THE IMPLEMENTATION APPROACH — THE MAP



WORK AREA 1: STRATEGY



What failing programs skip

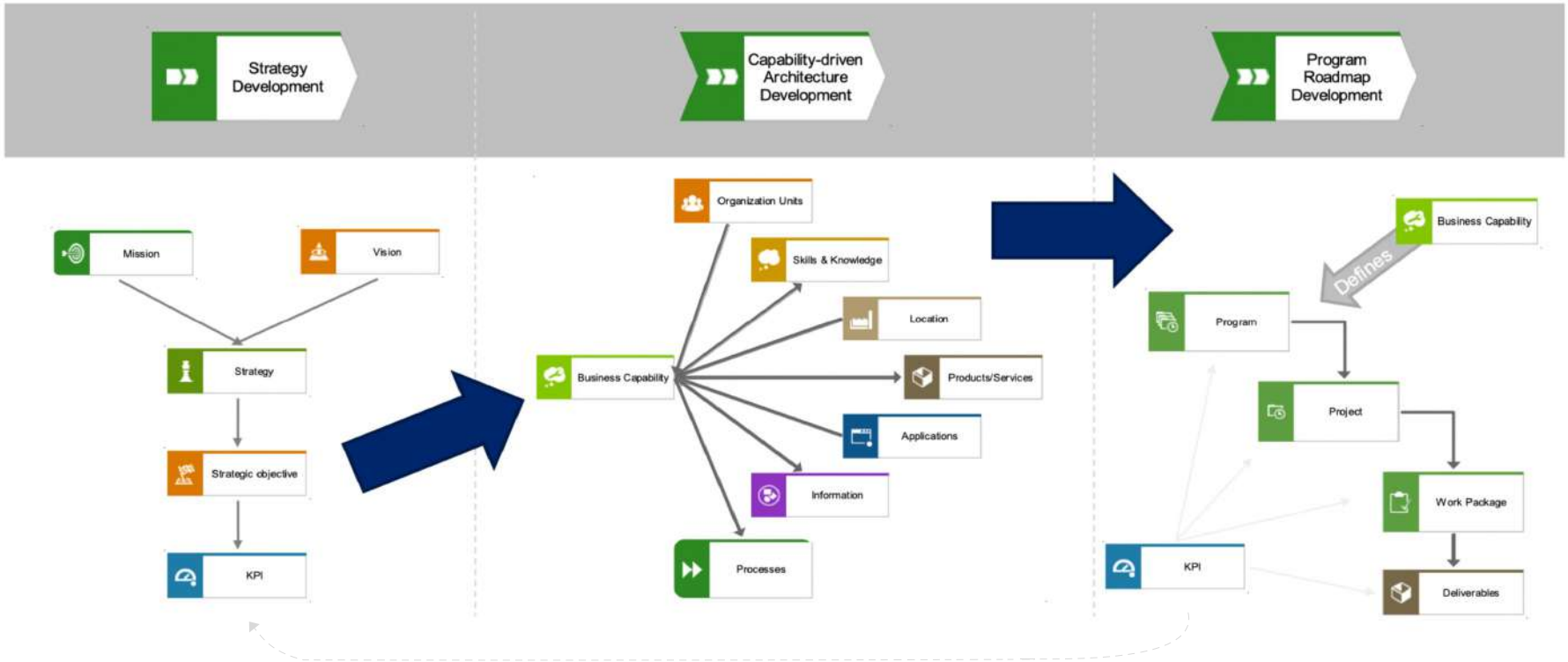
Jumping straight to tool configuration. First conversation: "How many licenses?"

What successful programs do

Answer three questions before touching the tool:

1. What are we for?
2. Who are we serving?
3. How will we measure success?

STRATEGY AS FIRST STEP OF CHANGE



WORK AREA 2: GOVERNANCE



Three things you must (at least) define before go-live

Who can create content?

Who has modeling rights? Under what conditions? What approval is needed before something is published?

Who approves it?

What is the review process? Who signs off? What happens to content that doesn't meet the standard?

What happens when rules are broken?

Not if — when. How does the program respond? How are exceptions handled?

WORK AREA 3: TECHNOLOGY



The only area many programs focus on.

Technology is one of four work areas.

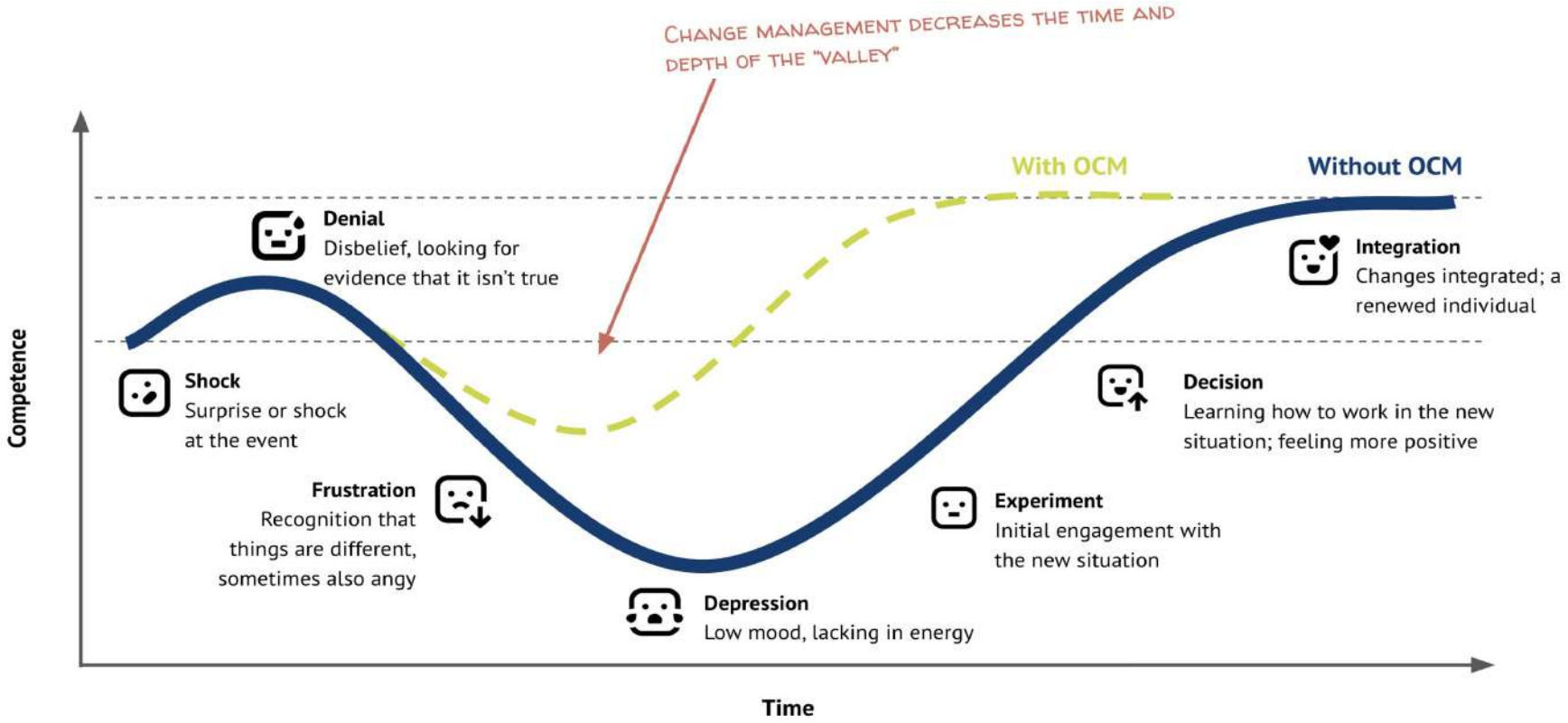
Not the only one.

The one thing to get right:

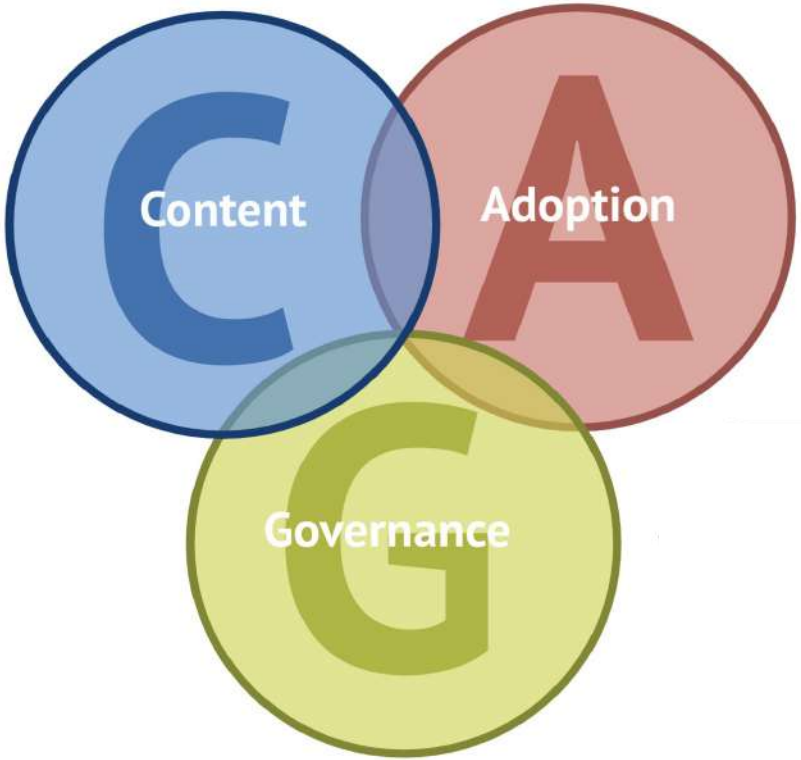
The meta model. It determines the available model types, object types, relationships, and attributes. If you cannot extend it when your needs evolve, you are locked into the vendor's idea of how your organization works.

Evaluate the meta model before you sign the contract. Not the interface. Not the integrations. **The meta model.**

BEFORE WE TALK ABOUT ADOPTION: UNDERSTAND THIS



ADOPTION IS WHERE PROGRAMS LIVE OR DIE



Let's be honest:

- **Content** — most programs have this
- **Governance** — some programs have this
- **Adoption** — almost no programs plan this adequately

Name one person whose job it is to drive adoption of your architecture program.

WORK AREA 4: CONTENT

... and communication & adoption

What this actually means in practice:



Communication starts at step 2, not step 5.

Don't wait until everything is ready. Start communicating when the program is approved. Rumors fill the vacuum faster than you think.

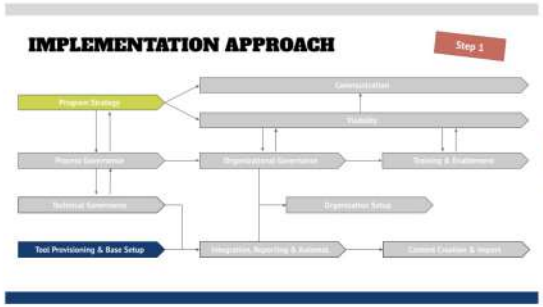
Training is designed from roles, not from features.

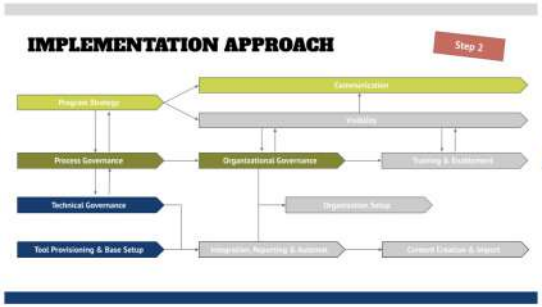
What does a business analyst need to know? What does a reviewer need? Design curriculum from the user's job, not from the tool's menu.

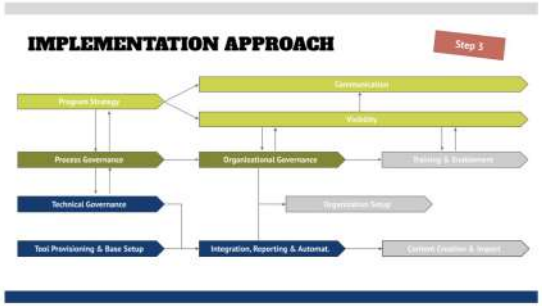
Feedback loops must be closed.

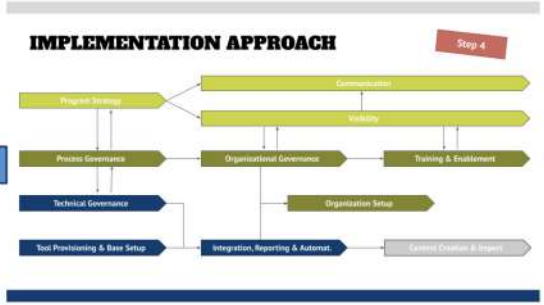
If users raise concerns and nothing changes, they stop raising concerns. And they stop using the tool. Collect feedback. Respond visibly.

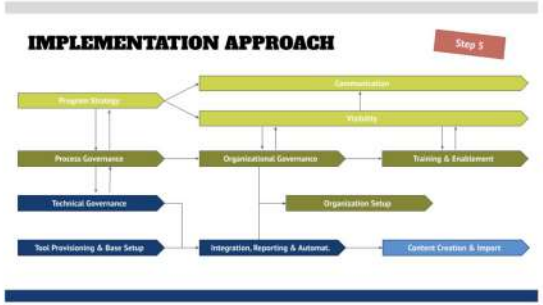
THE SEQUENCE THAT WORKS











TWO MISTAKES IN EVERY FAILING SEQUENCE

MISTAKE 1

Waiting until everything is ready to communicate

By the time you're ready to launch, the organization has already formed opinions — usually based on their last experience with an architecture program. Those opinions are rarely positive. You are fighting the narrative instead of writing it.

Communicate when the program is approved. Not when it's ready.

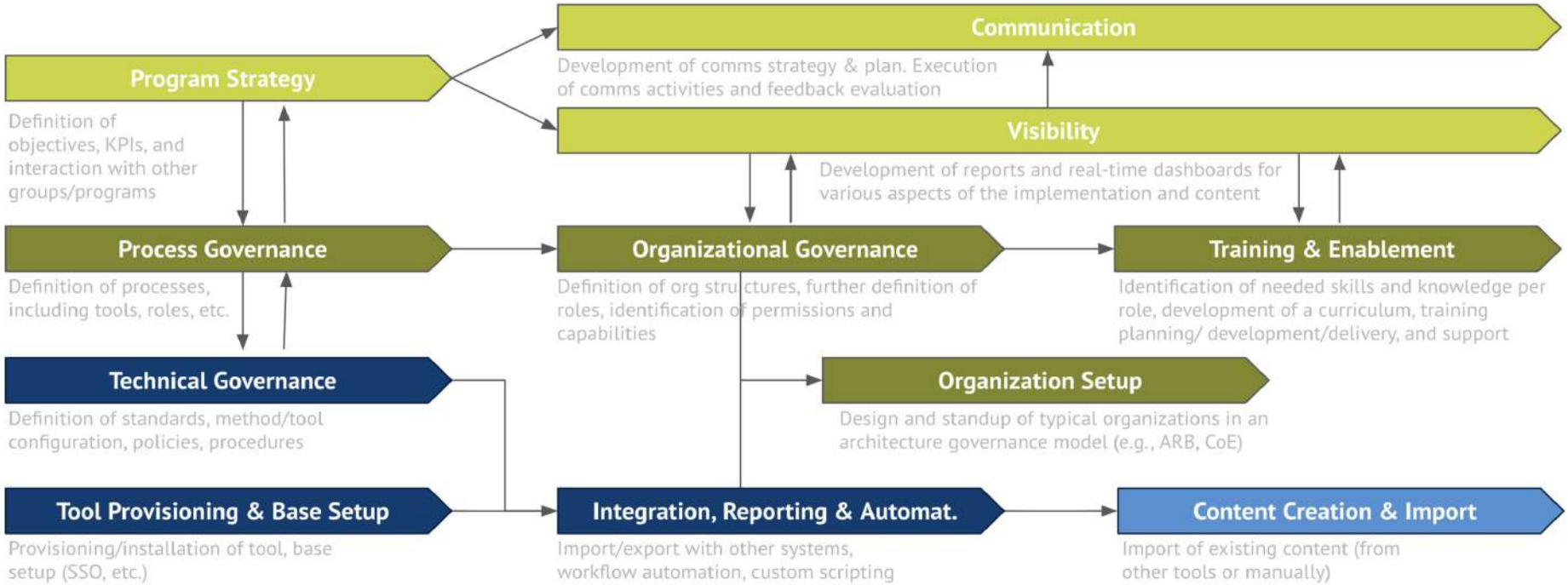
MISTAKE 2

Building technology, then enabling people

Sequential implementation — build everything, then train everyone — means your training is disconnected from the design. By the time people are trained, the tool has been configured based on assumptions about how they work. Some of those assumptions are wrong.

Run people enablement in parallel with technical build. The feedback you get during training will improve the configuration.

NOW YOU SEE THE WHOLE PICTURE



THE DIAGNOSTIC: THREE QUESTIONS

1

"What is our architecture program for, who does it serve, and how do we measure success?"

If you cannot answer this in one sentence — or if ten people give ten different answers — you have a strategy gap.

Strategy gap

2

"Who approves content, and what happens when someone breaks the rules?"

If you cannot name the approver and describe the process — or if the answer is "it depends" — you have a governance gap.

Governance gap

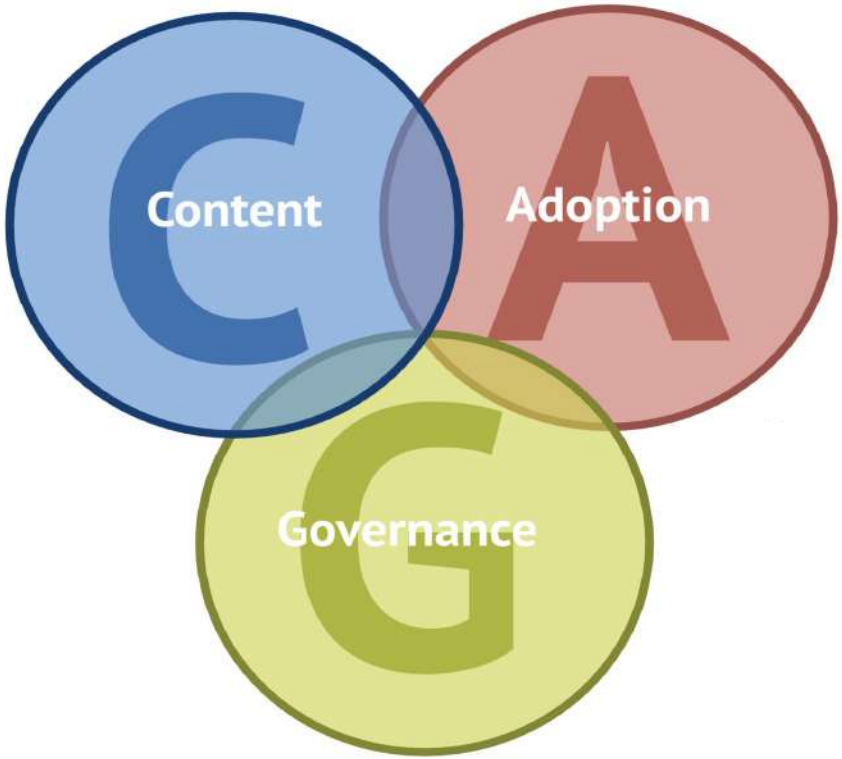
3

"Who is responsible for adoption, and when did your communication plan start?"

If you cannot name one person and the answer to "when" is "after go-live" — you have an adoption gap.

Adoption gap

WHICH CIRCLE IS YOUR PROGRAM WEAKEST IN?



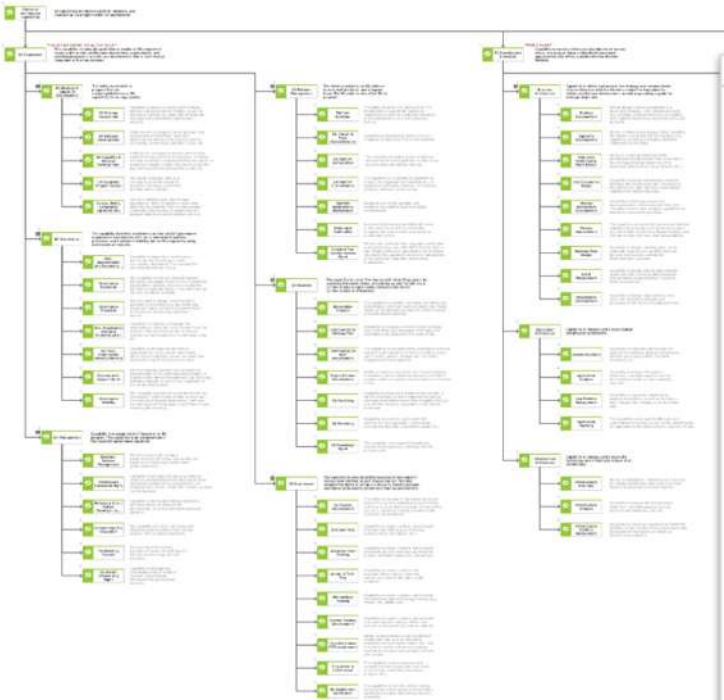
You now have a diagnostic.

Most programs identify at least one gap — often the same one.

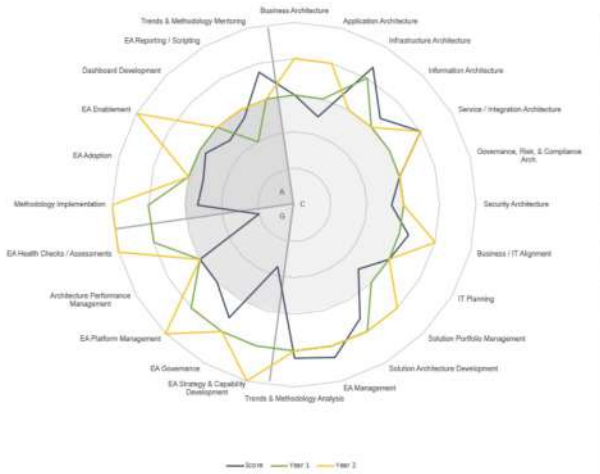
Knowing which circle is weak is the first step to fixing it.

The gap won't fix itself.

WHAT GOOD LOOKS LIKE: CAPABILITY MATURITY



ID	Capability / Evaluation	Final Score	Year 1	Year 2
Content		**3.4**	3	4
2.1	Business Architecture	**3.0**	3	4
2.2	Application Architecture	**2.5**	3	4
2.3	Infrastructure Architecture	**4.3**	4	3
2.4	Information Architecture	**3.3**	3	3
2.5	Service / Integration Architecture	**4.0**	3	4
2.6	Governance, Risk, & Compliance Arch.	**3.0**	3	3
2.7	Security Architecture	**2.7**	3	3
3.1	Business / IT Alignment	**3.3**	3	4
3.2	IT Planning	**3.0**	3	3
3.3	Solution Portfolio Management	**2.5**	3	4
3.4	Solution Architecture Development	**3.6**	4	4
1.3	EA Management	**4.3**	4	4
5.1	Trends & Methodology Analysis	**4.2**	4	4
Governance		**2.5**	4	5
1.1	EA Strategy & Capability Development	**1.8**	4	5
1.2	EA Governance	**3.6**	4	4
1.4	EA Platform Management	**3.0**	4	5
4.1	Architecture Performance Management	**3.0**	3	3
4.2	EA Health Checks / Assessments	**1.0**	4	5
5.2	Methodology Implementation	**2.7**	4	5
Adoption		**2.9**	3	3
1.5	EA Adoption	**2.6**	3	3
1.6	EA Enablement	**2.8**	3	5
4.3	Dashboard Development	**2.5**	3	3
4.4	EA Reporting / Scripting	**2.8**	2	3
5.3	Trends & Methodology Mentoring	**3.8**	3	3



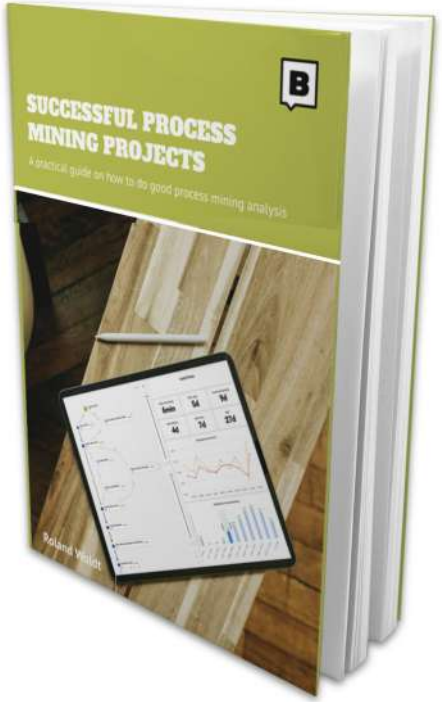
WANT TO LEARN MORE?

Use code "Adonis" for 25% off

whatsyourbaseline.gumroad.com







Q&A

Please reach out!

roland@whatsyourbaseline.com



