GREYSKULL ANALYTICS

A GREYSKULL ANALYTICS FIELD GUIDE

BARE BONES DATA STRATEGY

The questions worth answering before you buy another platform.

BEFORE YOU START

WRONG ROOM, WRONG ORDER

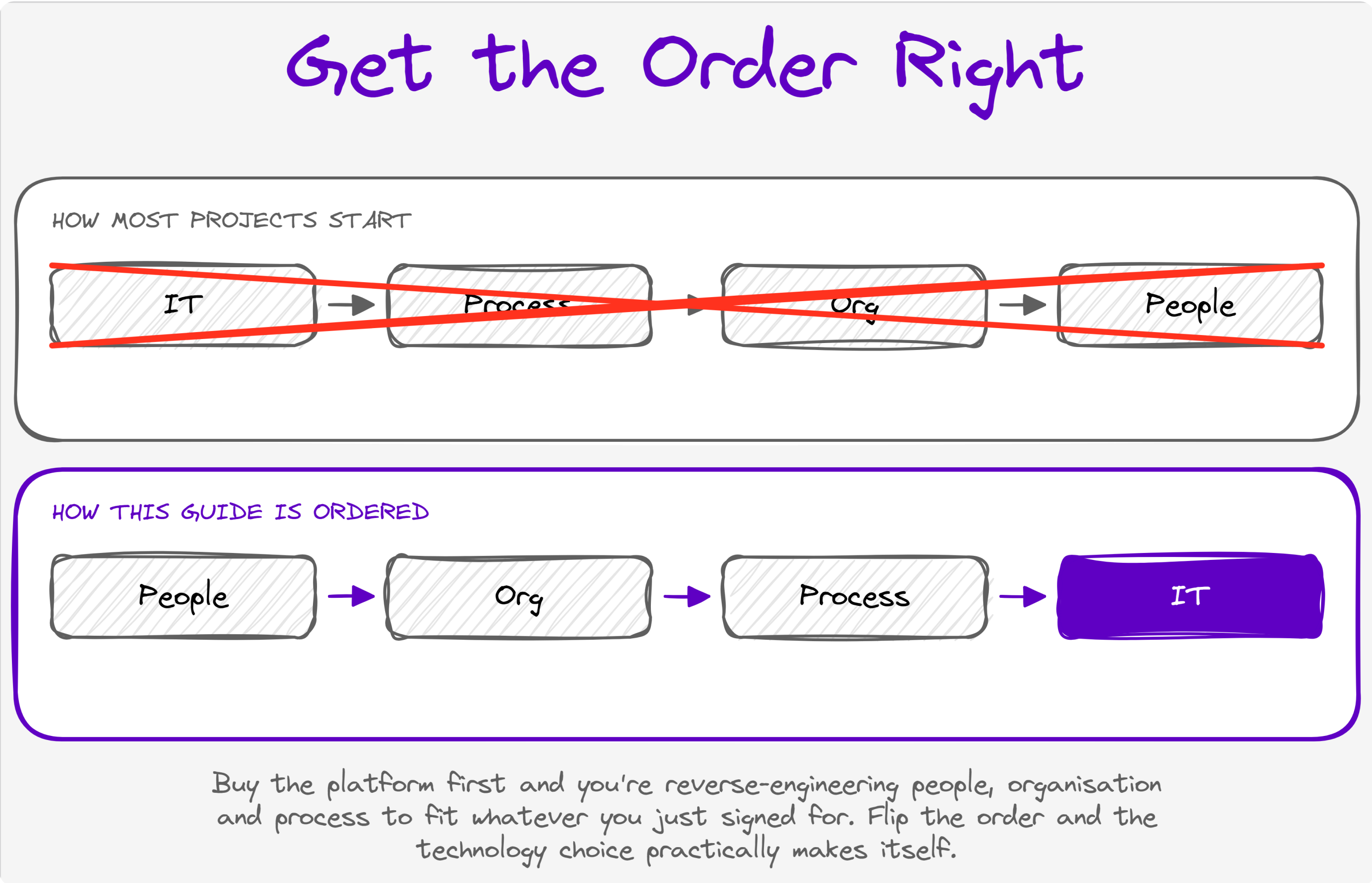
Most data strategy conversations start in the wrong place, and it costs people real money.

Someone in a leadership meeting says the word "AI." Three weeks later there's a six-figure platform contract signed, and nobody's agreed what problem it actually solves.

The order matters. Technology is the easy part. It's also the part everyone wants to discuss first, because buying software feels like progress, and asking "who's actually going to use this" doesn't.

This guide walks through the questions I take clients through before we talk about tools. It's built around a framework called **POPIT**: People, Organisation, Process, Information Technology. Notice where technology sits. Last. On purpose.

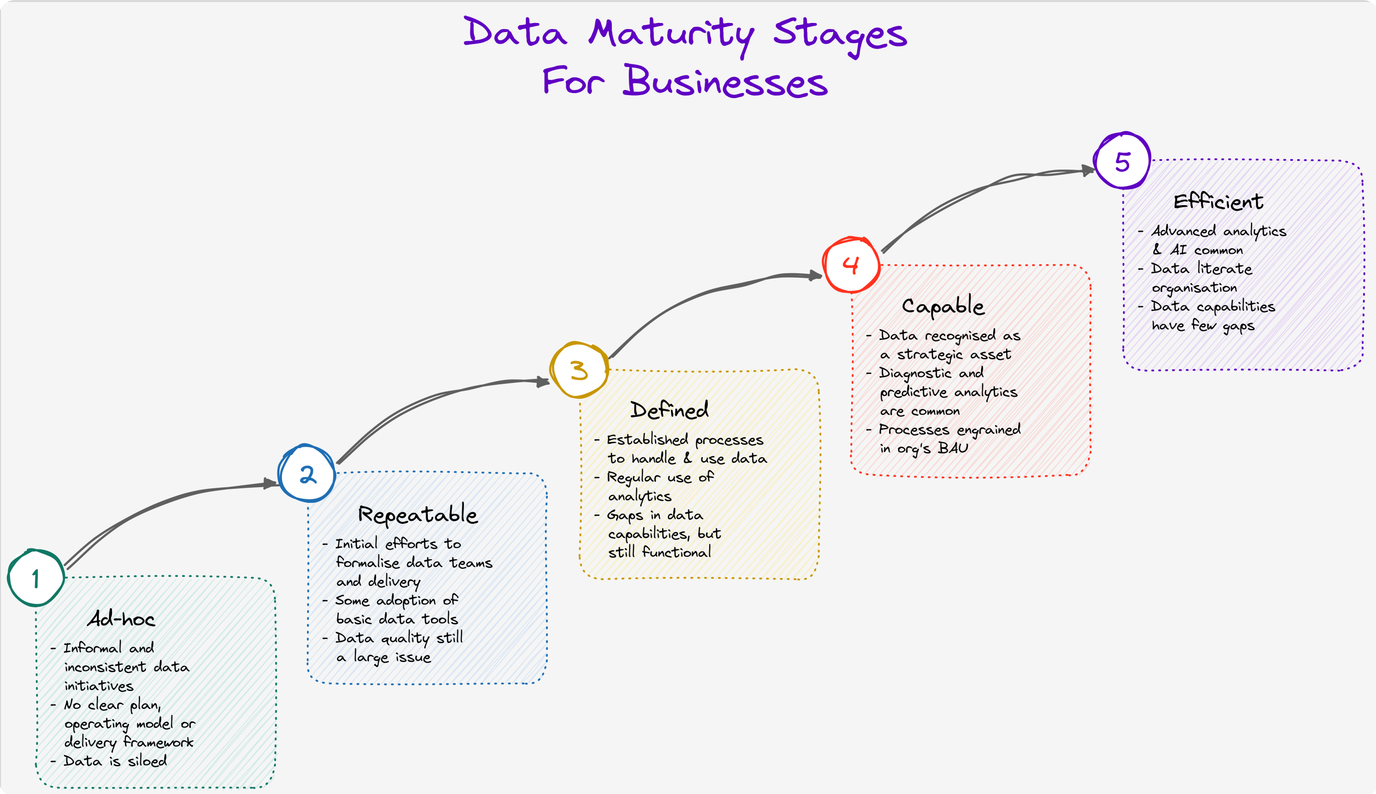
> Work through this roughly in order. Skip what doesn't apply to you. There's no prize for finishing every page, just fewer expensive mistakes at the other end.



STEP ONE

WHERE ARE YOU ACTUALLY AT?

Not where the board deck says you are. Where you actually are.



Source: Dylan Anderson, thedataecosystem.substack.com, https://thedataecosystem.substack.com/p/issue-34-top-ten-data-ecosystem-predictions

Before you plan where to go, get honest about where you're starting from. Three questions, worked through in this order:

1. Where are you now on the maturity curve? Ad hoc reporting in spreadsheets is a stage, not a crime. Own it.
2. Where do you actually need to get to? Not "AI-ready," whatever that means this week. What decision are you struggling to make today because the data isn't there?
3. What's the next step, not the last one? You don't jump from spreadsheets to a real-time ML platform in a single project. Find the step directly in front of you and take it.

> Maturity isn't a badge you earn once. Most businesses are further along in some areas than others, finance reporting might be genuinely mature while product analytics is still held together with exported CSVs. That's normal. Map it honestly, stage by stage.

1. Ad-hoc. Informal, inconsistent data initiatives. No clear plan, operating model or delivery framework. Data is siloed.

2. Repeatable. Initial efforts to formalise data teams and delivery. Some adoption of basic data tools. Data quality still a large issue.

3. Defined. Established processes to handle and use data. Regular use of analytics. Gaps in data capabilities, but still functional.

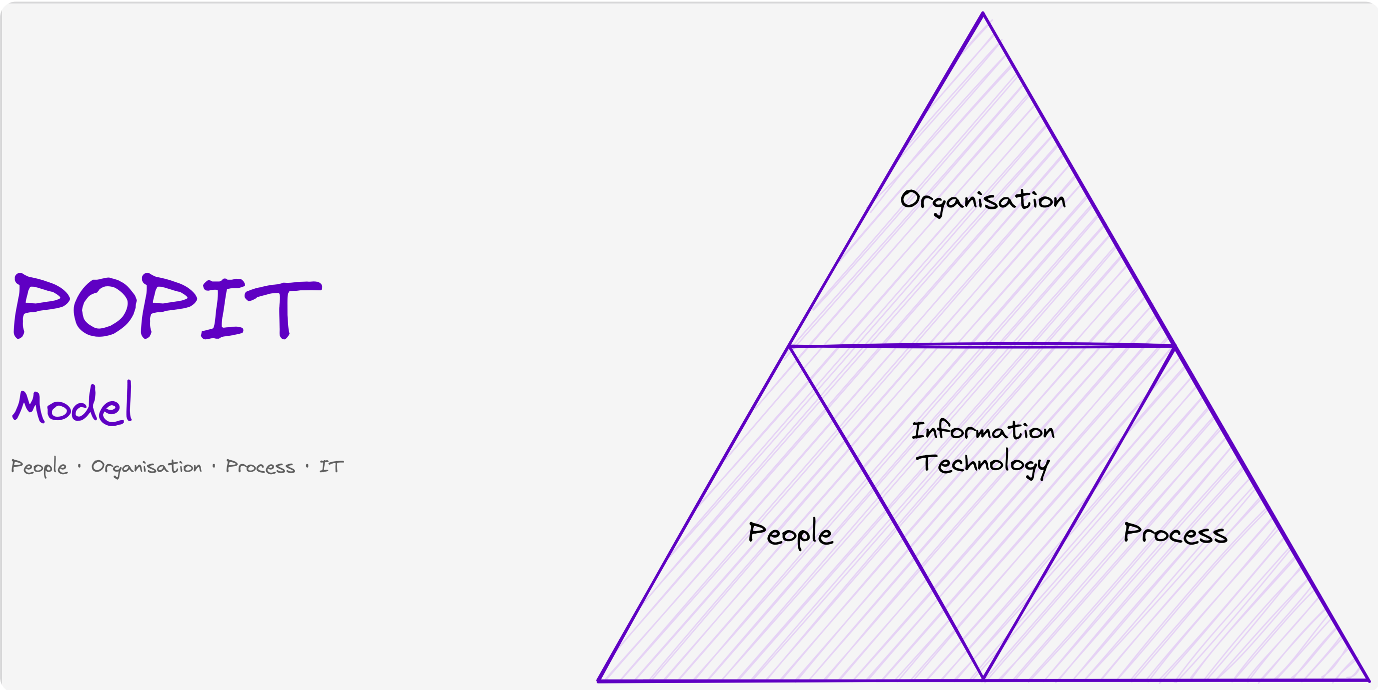
4. Capable. Data recognised as a strategic asset. Diagnostic and predictive analytics are common. Processes engrained in the org's BAU.

5. Efficient. Advanced analytics and AI common. Data literate organisation. Data capabilities have few gaps.

PEOPLE, ORGANISATION, PROCESS, THEN IT

POPIT gives you four lenses to look at a data strategy through, in a deliberate order:

- **People:** who's involved, what skills do they have and what do they actually need?
- **Organisation:** how's the team structured, and who owns what?
- **Process:** what governance keeps the wheels on?
- **Information Technology:** what you actually build and buy.



I put people first because software doesn't fix a broken relationship between a data team and the business. It never has, and it never will.

Get the first three right and the technology choice gets a lot easier. Get them wrong and the fanciest platform on the market won't save you from yourselves.

> This isn't a rigid checklist. It's a lens. Some projects only need you to think hard about one or two of these quadrants. The point is to look at all four deliberately, in this order, instead of skipping straight to the one everyone finds most exciting.

WHO NEEDS TO BE IN THE ROOM

Two groups. Two very different sets of questions.

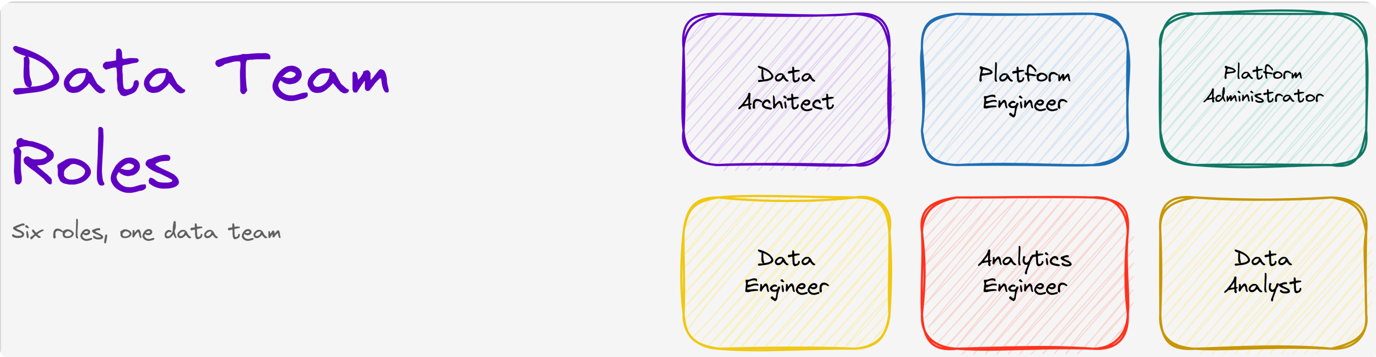
BUSINESS STAKEHOLDERS

Who's the audience for the data you're producing, and does what you're building actually trace back to a business objective they're accountable for? Get that link wrong and no delivery method saves you, you've just built the wrong thing beautifully. Once the content's right, how you actually serve them is the next real decision. Self-service or silver-service is a real decision, not a buzzword, pick it deliberately. And be honest about whether your end users have the data literacy to use what you're about to give them. A brilliant semantic model is wasted on an audience that can't open it, so make sure you're geared up to deliver your data outputs in a variety of guises to give your business stakeholders the ability to choose the method that best serves them.

> Self-service comes in many shapes and forms. It could be giving direct access to a database for skilled and data-savvy business stakeholders, or it might be some sort of serving semantic layer that lets them construct their own content easily in more user friendly environments. In this day and age, it might also be a chatbot type experience, allowing self-service via asking questions to an AI model.
> Silver-service means every analysis has to be done on the stakeholders behalf by a data specialist. They describe what they need and a person goes away and provides it, then hands it over to them on a platter.

DATA TEAM

Do you have the right roles covered? Do the people in them have the right skills, and if not, do you hire or do you train? Here's the shape of a typical team:



Data Architect: designs the technical foundation. Owns the big calls on security, governance and tooling.

Platform Engineer: builds and secures the infrastructure everything else runs on. AKA Infrastructure/DevOps Engineer.

Platform Administrator: keeps the lights on. Manages access, monitors usage.

Data Engineer: gets data in, cleans it up, builds and runs the pipelines and orchestration.

Analytics Engineer: turns raw data into curated datasets. The bridge between engineering and the business.

Data Analyst: turns curated data into answers: dashboards, reports, the analysis someone actually asked for.

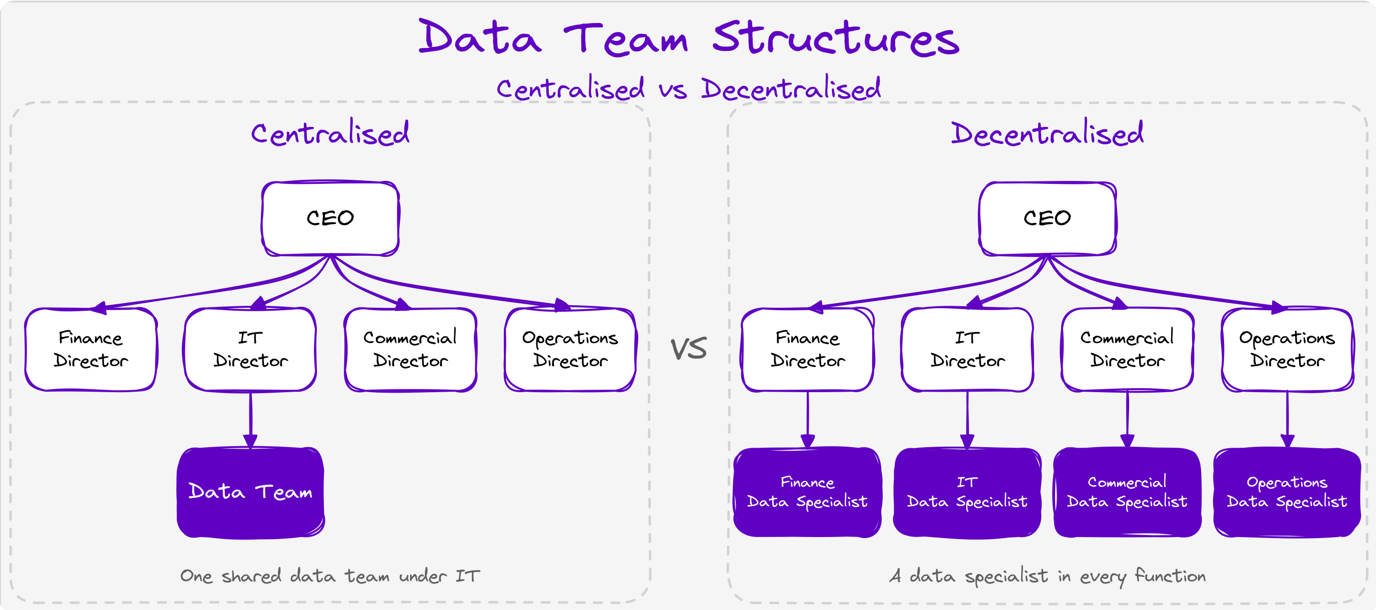
Point of note... this is a number of roles NOT a number of heads. It's possible to be a team of one that carries out all six roles, and for teams of ANY size any given person might wear multiple of these hats. It's also possible that some roles might be outsourced outside of the data team - in some businesses Platform Engineering might be handled by a separate Infrastructure team, not by the data team directly.

> The best data teams I've worked with work collaboratively and iteratively with business stakeholders, combining their technical expertise with the domain knowledge of their end users, delivering value early and incrementally.

WHERE IT SITS, AND WHO OWNS IT

Structure follows how decisions actually get made, not how the org chart says they do.

Centralised or decentralised? If central, where does it sit? Inside IT? Finance? Its own function entirely? There's no universally correct answer, there's a correct answer for *your* business, and it depends on how decisions actually get made in your organisation, not how the org chart claims they get made.



Again, the shape of this will likely depend on the data maturity and literacy of your business as well as the size. Smaller organisations may not have enough data work to justify specialists in every part of the business and will get better economies of scale from a central team. For large companies, the domain expertise acquired from having a specialist sit inside their team can be well worth it.

It's also entirely possible to have a hybrid of these approaches, with some responsibilities sitting centrally and others being devolved.

The important part of this is making sure that those roles and responsibilities are defined consciously: deciding your operating model and organisational design should be considered, not left to evolve organically.

> A quick gut check: if someone outside the data team wants a new report or dataset, do they know who to ask? If the honest answer is "it depends who you know," that's a structure problem, not a tooling problem, and no platform migration will fix it.

THE GOVERNANCE THAT KEEPS THE WHEELS ON

Not all of it, all at once. But you need an honest answer for each of these.

Code management. Version control, peer review and a deployment process for the code that runs your pipelines, models and reports. If a change can go straight from a laptop to production with nobody else looking at it, that's not a process, that's a gamble.

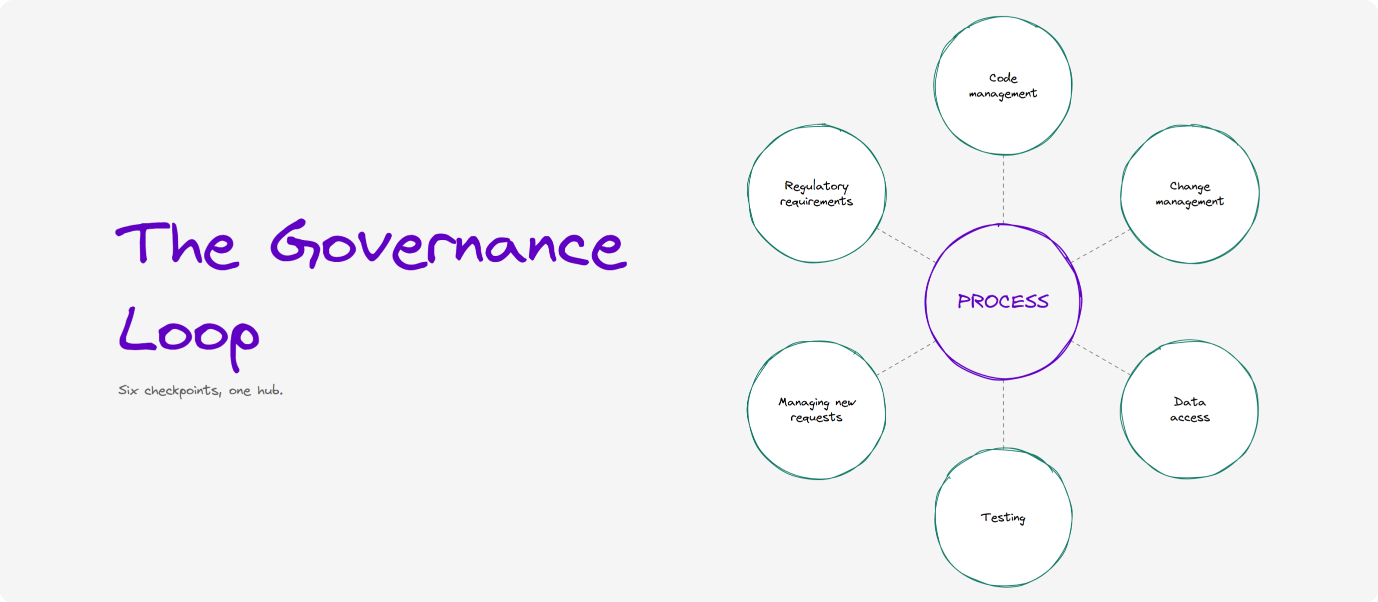
Change management. How do you introduce a change to a table, a metric definition or a pipeline without quietly breaking the dashboard an exec checks every Monday morning? Someone needs to own the answer.

Data access. Who can see what, how access is requested and granted, and how often it's reviewed and revoked. Access that's easy to grant and never revisited is a liability, not a convenience.

Regulatory requirements. GDPR, sector-specific compliance, data retention rules. What you're required to do, not just what's good practice, and who's accountable if it's not done.

Managing new requests. An actual intake process, so work gets prioritised on merit rather than on who shouted loudest or sits closest to the data team.

Testing. Validating data quality and pipeline logic before it reaches production, not after someone in a board meeting notices the numbers look wrong.



> Boring? Yes. Skip it and you'll find out exactly why it matters, usually during an audit, or the week your best analytics engineer leaves and nobody can find the documentation.

WHAT YOU'RE ACTUALLY BUYING

Everything before this page should happen first. If you're shortlisting vendors before answering the people, organisation and process questions, you're optimising the wrong variable. The buying consideration categories below borrow directly from Joe Reis and Matt Housley's Fundamentals of Data Engineering, a book worth owning if you're going to be making these calls.

Team size and capabilities. Buy for the team you have, not the team you wish you had.

Speed to market. What does another quarter of waiting actually cost you?

Interoperability. Does it work with what you already run, or does it quietly want to replace everything?

Cost and business value. Total cost of ownership, opportunity cost and whether it actually pays for itself.

Today versus the future. Every technology eventually gets replaced. Build on what's stable and treat the rest as replaceable.

Location. On premises, cloud, hybrid or multicloud, and where the data needs to physically and legally live.

Build versus buy. Building gives you control. Buying gives you speed. Be honest about which one you need right now.

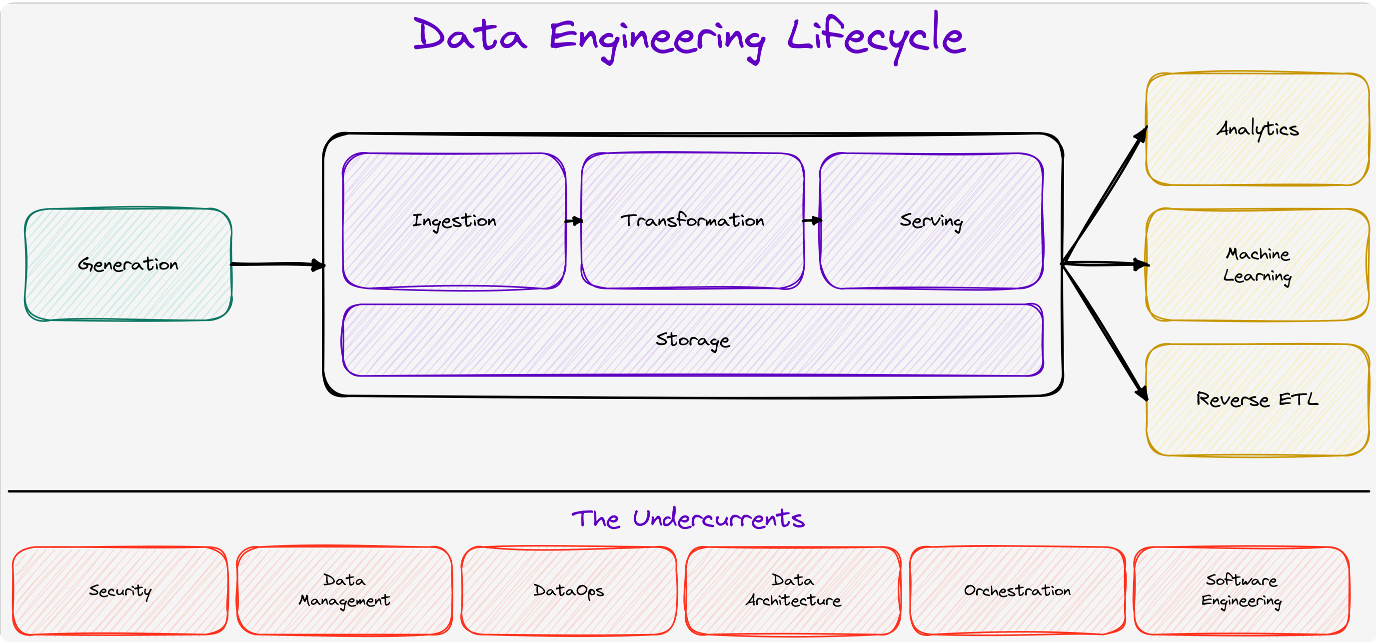
Monolith versus modular. One tightly coupled system that's simple to reason about, or best-of-breed tools stitched together.

Serverless versus servers. Pay only when your code runs, or keep control of the infrastructure underneath it.

> None of these questions have a universally right answer. That's the point. Anyone selling you a platform before asking about your team, your organisation and your processes is selling you their answer to someone else's question.

THE DATA ENGINEERING LIFECYCLE

The same book also sets out the Data Engineering Lifecycle, a useful blueprint for understanding what technology areas you will need to consider. This is NOT an architecture diagram, it's much more process focussed than that, but it maps well to the components a technical architecture needs to consider. The main lifecycle maps well to classic data processing patterns whereas the undercurrents map well to the underlying process and governance needed to support this.



Source: Housley, M. and Reis, J. (2022) *Fundamentals of Data Engineering*. O'Reilly Media, Inc. https://link.amazon/B0hndFpCC

ONE IDEA WORTH KNOWING

THE DATA LAKEHOUSE

DATA LAKEHOUSE

If you've worked through the POPIT framework and it does become clear that investing in a new data platform is the right move for you, you'll want to familiarise yourself with the term Data Lakehouse.

The Data Lakehouse has become the de facto architecture for data platforms in the current era. A Lakehouse is a Data Lake and a Data Warehouse that stopped arguing. Cheap object storage, open table formats, one place for BI and AI/ML instead of two separate platforms.

Worth it if: you're scaling, want to avoid vendor lock-in and you're happy investing engineering time to implement it properly.
Worth pausing on: table formats like Delta Lake, Iceberg and DuckLake are still maturing: "simple" implementations turn code-heavy fast.

Modern data vendors such as Databricks (oft cited as creators of the term Data Lakehouse) and Microsoft Fabric are built on the underlying concepts of a Data Lakehouse, as is Greyskull Analytics' own Oriel platform, a cost-conscious Data Lakehouse aimed at smaller businesses.

> You don't need to master the technical detail of a Data Lakehouse to write a good data strategy. You need to know they exist, roughly what they solve, and that "we should get a Lakehouse" isn't a strategy on its own, it's a technology choice you make after everything else in this guide.

FURTHER READING

WORTH YOUR TIME

Five things I'd actually point you towards, not a generic reading list.

The Data & AI Ecosystem · thedataecosystem.substack.com

A solid, ongoing commentary on where data platforms and practice are actually heading.

AWS on Data Strategy · aws.amazon.com/what-is/data-strategy

Vendor-flavoured, sure, but a genuinely reasonable starting point on the fundamentals.

Databricks on the Lakehouse · databricks.com/glossary/data-lakehouse

The definition straight from the source. Worth reading once, critically.

Fundamentals of Data Engineering · Housley & Reis, O'Reilly Media, 2022 · https://link.amazon/B0hndFpCC

If you only read one book off this list, make it this one.

POPIT, more broadly · a long-standing business analysis framework · knowledgetrain.co.uk/popit-model

Not a Greyskull invention. It's a well-established way of structuring change, borrowed here and applied specifically to data.

None of these will hand you a finished strategy. They're starting points for better questions, which, if you've made it this far, is roughly the whole idea of this guide.

WHERE TO GO FROM HERE

IF YOU'RE READING THIS, YOU'RE ALREADY AHEAD OF MOST BUSINESSES

None of this is complicated. It's just easy to skip, because the technology is more exciting than the groundwork. This year, that technology comes with a chatbot bolted on and calls itself AI-ready.

I help businesses work through exactly this: People, Organisation, Process, then Technology, before a single pound gets spent on a platform. Get that order right and "AI-ready" stops being a marketing checkbox. It starts being true.

Call it the Greyskull Method if you want a name for it. Fundamentals over fads, every time.

Click To Get In Touch →

Scan to find more from Greyskull Analytics

Johnny Winter

Chief Data Nerd, Greyskull Analytics