Ways of work

An introduction to DEPLOY

Ways of work

The EGAD framework

> The EGAD framework provides us with a guide on how to **leverage data to solve problems**. It includes everything from understanding the problem to maintaining the solution.

	Draft	Do	Deliver	Decompress
EXPLAIN	Problem statement	Storytelling	Communication	Feedback
GATHER	Problem landscape	Databases	Data engineering	Insights
ANALYSE	Equation of value	Programming	Solution governance	Performance metrics
DEPLOY	Project management	Version control	Production	Maintenance

Ways of work

DEPLOY

> DEPLOY in the EGAD framework focuses on the **development of a system or solution**, with proper management and version control, that “lives” in the **real world**.

	DRAFT	DO	DELIVER	DECOMPRESS
DEPLOY	**Project management** When we have **processes** in place to **manage** a project,	**Version control** and we can easily **apply** and **track changes**,	**Production** we can **deploy** a **working solution** into the “real world” that **stakeholders can engage with**,	**Maintenance** while ensuring that it is **properly maintained** and in working order.

Ways of work

Solutions in the real world

> Successful data deployment integrates solutions seamlessly into real-world applications. To enable this we need a systematic and structured approach to our development and deployment.

Solutions in the real world refer to how our **deployed data** and **code** perform **in actual environments** that stakeholders interact with.

A solution's efficacy is gauged by its **performance** in these actual environments, emphasising **scalability** and **adaptability** to user feedback.

Due to the **complexity of real-world solutions**, a robust solution is only possible with planning of all project phases, including:

- **Development**
- **Management**
- **Control**
- **Maintenance**

Ways of work

DEPLOY, DRAFT

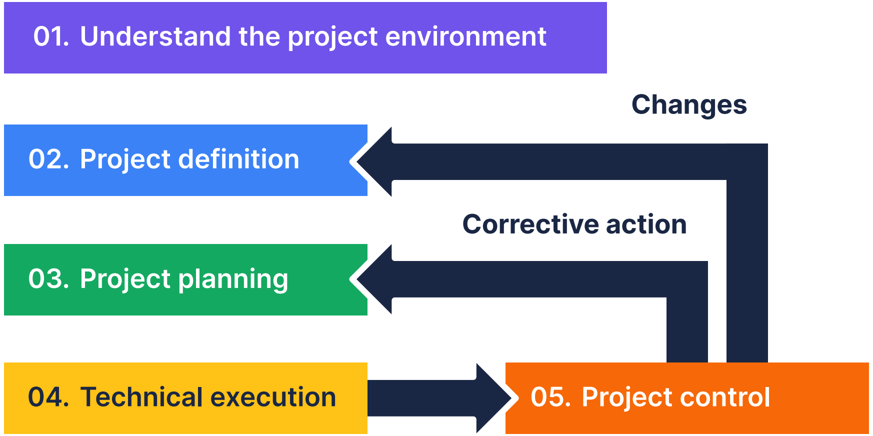
> The first phase of DEPLOY, DRAFT, focuses on the **planning**, **execution**, and **control of a project** to ensure that a specific goal and success criteria are met.

At this point in the project, we already know what the problem is, and we possibly also have a solution, now we need to:

- Define the scope and timelines of the project to **DEPLOY resources** appropriately.
- Plan **how** the resources will **DO** the project.
- Plan **how** the resources will **DELIVER** the project.

Project management and **Agile** approaches are often used to ensure that a solution or project is successfully **deployed**.

A **standard project** typically has the following **phases**:



Ways of work

DEPLOY, DO

> The **DO** phase of **DEPLOY** ensures that we can **track** and **manage changes** to files and code over time.

Version control systems provide a structured way to **manage different versions of files**, **facilitate collaboration**, and **help prevent conflicts** when multiple contributors work on the same project simultaneously.

Version control is essentially saving the **state of a file.** It helps ensure that we can track the **history** of the file's editing.

How do we do version control?

- We use distributed version control systems (DVCSs) such as **Git**.
- Git can show us the history of our files and codebase, which can help us find errors and bugs in our code.
- When we commit (save) a file, we will have to comment in detail on the changes we've implemented in the code to ensure the lineage of the changes.

Ways of work

DEPLOY, DELIVER

> The **DELIVER** phase of **DEPLOY** focuses on the live system or solution that is actively serving its intended purpose. This is called a **production environment**, or simply production or prod.

A model should be seamlessly **integrated** into a **production environment** and be **maintained**:

It is our solution integrated into an environment where it takes an **input** and returns an intended **output**.

We need to develop **solutions** with production in mind.

Modern solutions should be **robust** and **scalable** – this often requires the solution to live in the cloud.

Some of the **key characteristics of production** include:

- **Stability:** Stable and free from major bugs. It has been thoroughly tested.
- **High performance:** Optimised for performance to provide users with a responsive and high-quality experience.
- **Scalability:** Capable of handling increased demand without degrading performance.
- **Security:** Configured with security controls, firewalls, access controls, encryption, and other security measures.

Ways of work

DEPLOY, DECOMPRESS

> The **DECOMPRESS** phase of **DEPLOY** focuses on how a solution or project is **maintained** after it has been deployed.

Once a model, solution, or project has been deployed, **continuous maintenance** and **improvement** are required.

Model performance very often **degrades** over time – this is called model drift. So deployed models need to be **monitored** through **logs**.

As a result, the **DECOMPRESS** phase can **only be successful** when the DRAFT, DO, and DELIVER phases have been robustly implemented:

DRAFT		DO		DELIVER		DECOMPRESS
Project management	+	Version control	+	Production	=	Maintenance

The EGAD framework is used to ensure that the **right solution** to the **right problem** is developed but also that it is sustainable and so **continues to solve the problem**.

Tools for DEPLOY

> Some of the **tools** that are used in DEPLOY include the following:

Project management:
Trello, Jira

- There are many options to manage a team's productivity but all project management tools are designed to **provide a structured** and **collaborative space** to keep track of tasks and issues.
- Trello and Jira are web-based **Kanban-style** applications.

Version control:
Git and GitHub

- Git and GitHub are popular **distributed version control** tools.
- It is a powerful combination to **manage additions** or **changes** to source code.

Production:
AWS, Azure

- Modern, real-world products have to "**live" in the cloud.**
- Both AWS (Amazon Web Services) and Microsoft Azure offer many resources and certifications to learn how to use their cloud services.

Ways of work

Effective DEPLOY techniques

Throughout this course, you'll **learn** various **skills** and **tools** you can apply in the DRAFT, DO, DELIVER, DECOMPRESS phases of many different **projects** to solve real **problems**.

Proper **management** and **version control** ensure **consistency** and **adaptability** throughout the development lifecycle of a project.

Deploying a data solution means **translating theoretical models** into **actionable**, **real-world applications** that drive tangible results.

Leveraging **project management methodologies** and robust **version control** strategies is foundational to seamless and successful deployment.