Architecture Design

Crop Production Analysis

Written By	Juli Kumari
Document Version	0.3
Last Revised Date	08-02-2023

DOCUMENT CONTROL

Change Record:

VERSION	DATE	AUTHOR	COMMENTS
0.1	081- Feb - 2023	Juli Kumari	Introduction and architecture defined
0.2	06- Feb - 2023	Juli Kumari	Architecture & Architecture description appended and updated.
0.3	08- Feb - 2023	Juli Kumari	End to End project and documentation updated

Reviews:

VERSION	DATE	REVIEWER	COMMENTS
0.2	08- Feb - 2023	Juli Kumari	Project Reviews

Approval Status:

VERSION	REVIEW DATE	REVIEWED BY		APPROVED BY	COMMENTS

Contents

1. Introduction

1.1 What is Architecture design document?

Any software needs the architectural design to represent the design of software. IEEE defines architectural design as “the process of defining a collection of hardware and software components and their interfaces to establish the framework for the development of a computer system.” The software that is built for computer-based systems can exhibit one of these many architectures.

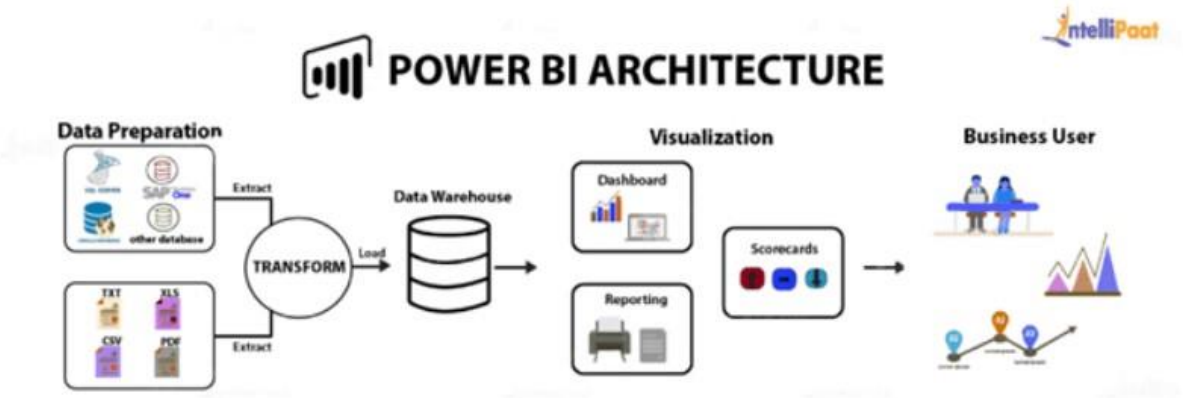
Each style will describe a system category that consists of:

- A set of components (eg: a database, computational modules) that will perform a function required by the system.
- The set of connectors will help in coordination, communication, and cooperation between the components.
- Conditions that how components can be integrated to form the system.
- Semantic models that help the designer to understand the overall properties of the system.

1.2 Scope

Architecture Design Document (ADD) is an architecture design process that follows a step-by-step refinement process. The process can be used for designing data structures, required software architecture, source code and ultimately, performance algorithms. Overall, the design principles may be defined during requirement analysis and then refined during architectural design work.

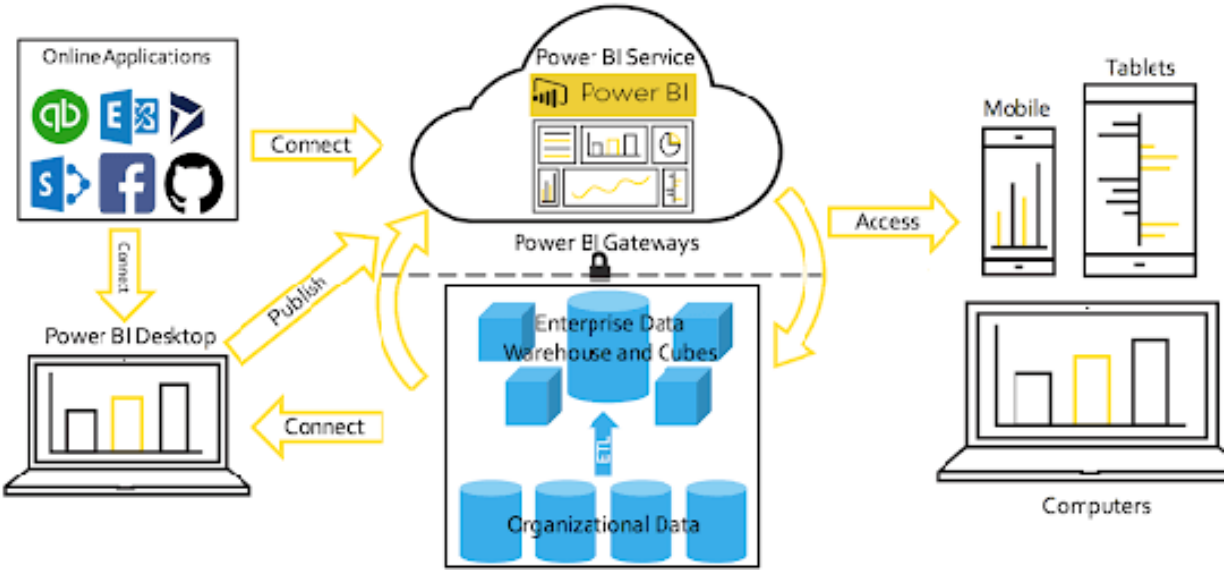
2. Architecture

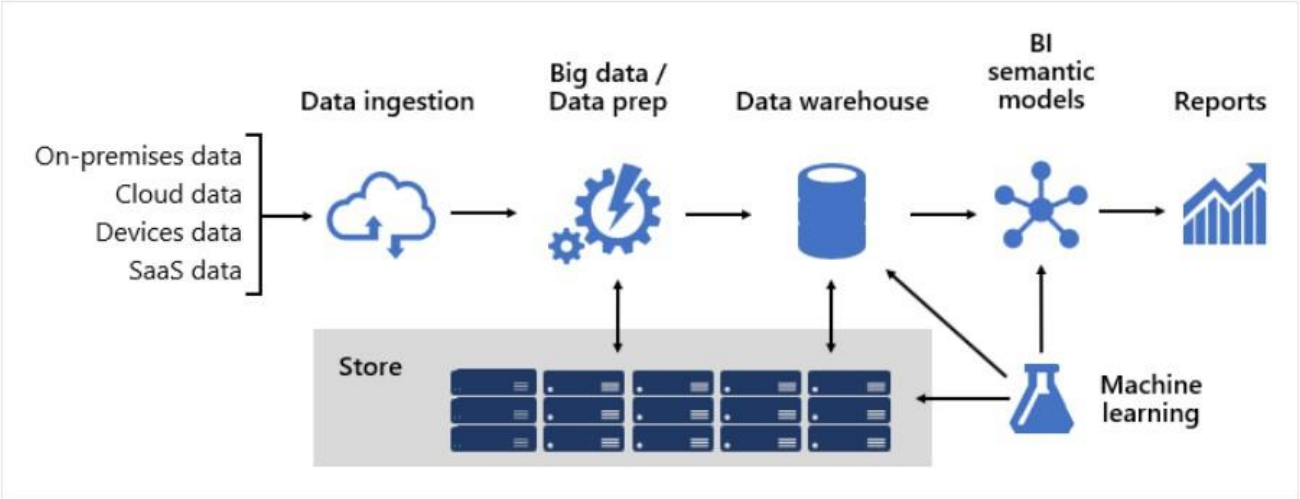


2.1 Power BI Architecture

MS Power BI architecture consists of four major steps that explain the whole process from data sourcing to the creation of reports and dashboards. Various technologies and processes work together to get the required results with extreme precision. Let's see those steps further.

The following diagram shows Power bi Server's architecture:





Power Bi Server is internally managed by the multiple server processes.

2.2. Data Sources: -

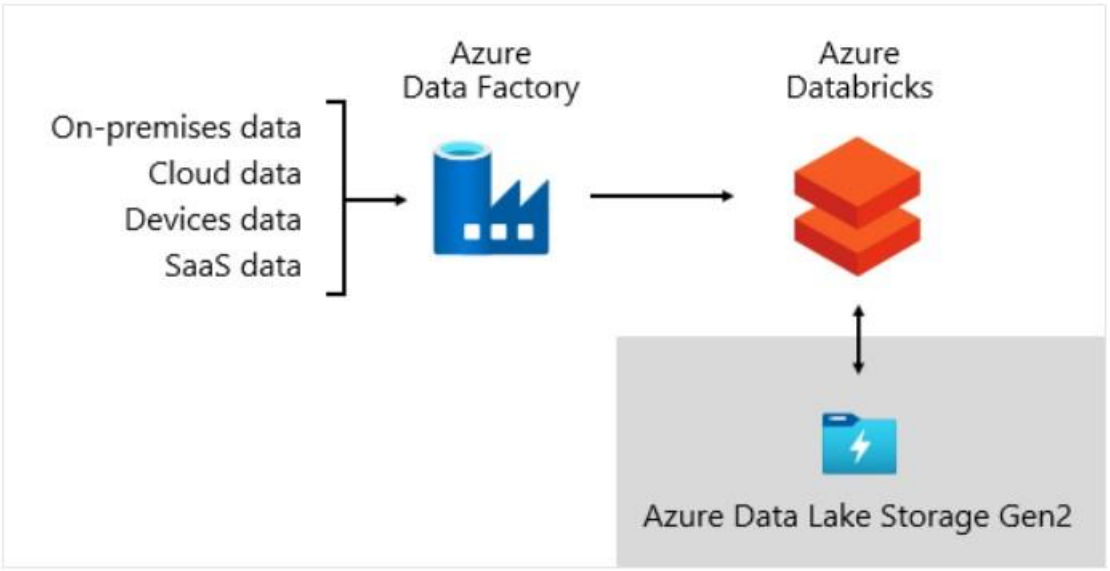
A data warehouse can consolidate data from practically any data source. It's mostly built over LOB data sources, which are commonly relational databases storing subject-specific data for sales, marketing, finance, etc. These databases can be cloud-hosted, or they can reside on-premises. Other data sources can be file-based, especially web logs or IOT data sourced from devices. What's more, data can be sourced from Software-as-a-Service (SaaS) vendors.

At Microsoft, some of our internal systems output operational data direct to ADLS Gen2 using raw file formats. In addition to our data lake, other source systems comprise relational LOB applications, Excel

workbooks, other file-based sources, and Master Data Management (MDM) and custom data repositories. MDM repositories allow us to manage our master data to ensure authoritative, standardized, and validated versions of data.

2.3 Data ingestion: -

On a periodic basis, and according to the rhythms of the business, data is ingested from source systems and loaded into the data warehouse. It could be once a day or at more frequent intervals. Data ingestion is concerned with extracting, transforming, and loading data. Or, perhaps the other way round: extracting, loading, and then transforming data. The difference comes down to where the transformation takes place. Transformations are applied to cleanse, conform, integrate, and standardize data. For more information, see Extract, transform, and load (ETL).



2.4 Big data / data preparation: -

The services is used to schedule and orchestrate data validations, transformations, and bulk loads from external source systems into our data lake. It's managed by custom frameworks to process data in parallel and at scale. In addition, comprehensive logging is undertaken to support troubleshooting, performance monitoring, and to trigger alert notifications when specific conditions are met.

2.5 Data warehouse: -

Power BI integrates out of the box with MySQL. MySQL is one of the most popular open-source relational databases and is employed by many companies. An in-built connector allows for easy integration, visualization and analysis of MySQL data. However, it works with the desktop version of Power BI only.

2.6 BI semantic models: -

BI semantic models represent a semantic layer over enterprise models. They're built and maintained by BI developers and business users. BI developers create core BI semantic models that source data from enterprise models. Business users can create smaller-scale, independent models—or, they can extend core BI semantic models with departmental or external sources. BI semantic models commonly focus on a single subject area and are often widely shared.

Business capabilities are enabled not by data alone, but by BI semantic models that describe concepts, relationships, rules, and standards. This way, they represent intuitive and easy-to-understand structures that define data relationships and encapsulate business rules as calculations. They can also enforce fine-grained data permissions, ensuring the right people have access to the right data. Importantly, they accelerate query performance, providing extremely responsive interactive analytics—even over terabytes of data. Like enterprise models, BI semantic models adopt naming conventions ensuring consistency.

2.7 Reports: -

A Power BI report is a multi-perspective view into a dataset, with visuals that represent findings and insights from that dataset. A report can have a single visual or many pages full of visuals. Depending on your job role, you might be someone who designs reports, or you might be a business user who consumes reports. This article is for business users.

2.8 Data consumption: -

At the reporting layer, business services consume enterprise data sourced from the data warehouse. They also access data directly in the data lake for ad hoc analysis or data science tasks.

2.9 Power Bi Power BI Key Components

Let us have a glimpse of these components:

Power Query

Power Pivot

Power View

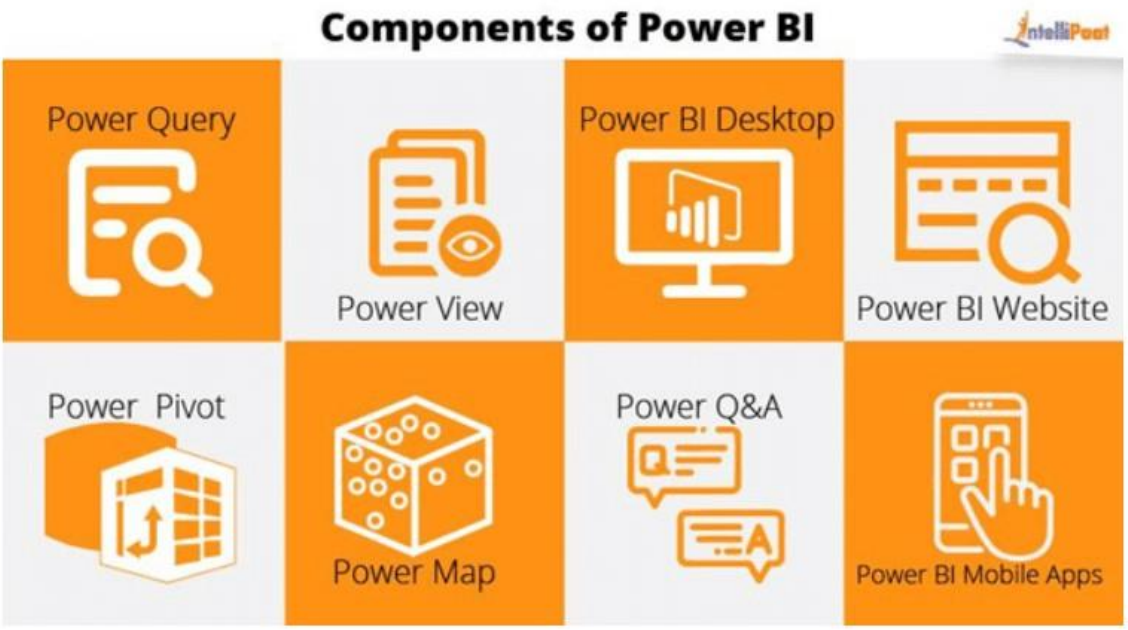
Power Map

Power Q&A

Power BI Desktop

Power BI Website

Power BI Mobile Apps

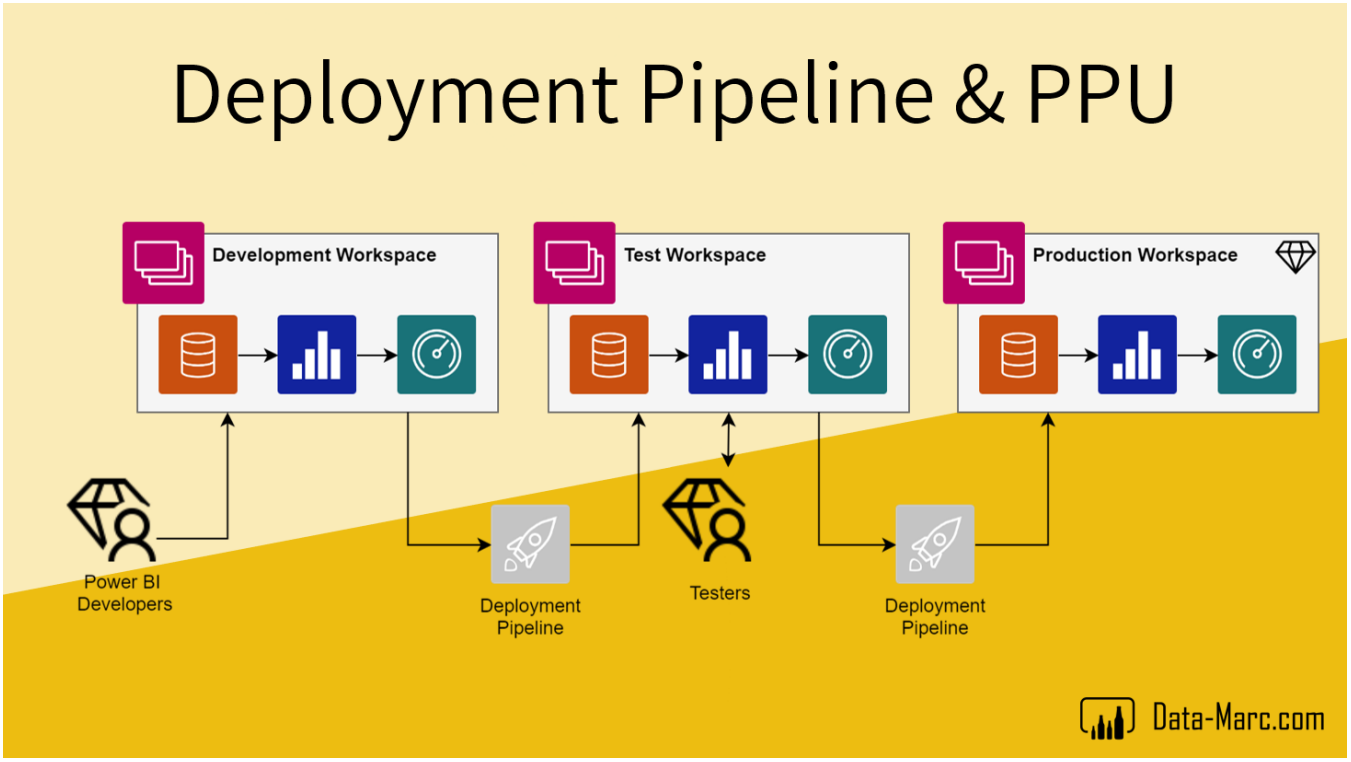


ARCHITECTURE

3. Deployment Description

3.1 Deployment Pipeline in Power BI

The deployment pipelines tool enables BI creators to manage the lifecycle of organizational content. It's an efficient and reusable tool for creators in an enterprise with Premium capacity. Deployment pipelines enable creators to develop and test Power BI content in the Power BI service, before the content is consumed by users. The content types include reports, paginated reports, dashboards, datasets and dataflows.:



3.2 Power BI Pipeline structure

Deployment pipelines is designed as a pipeline with three stages:

- **Development**

  This stage is used to design, build, and upload new content with fellow creators. This is the first stage in deployment pipelines.

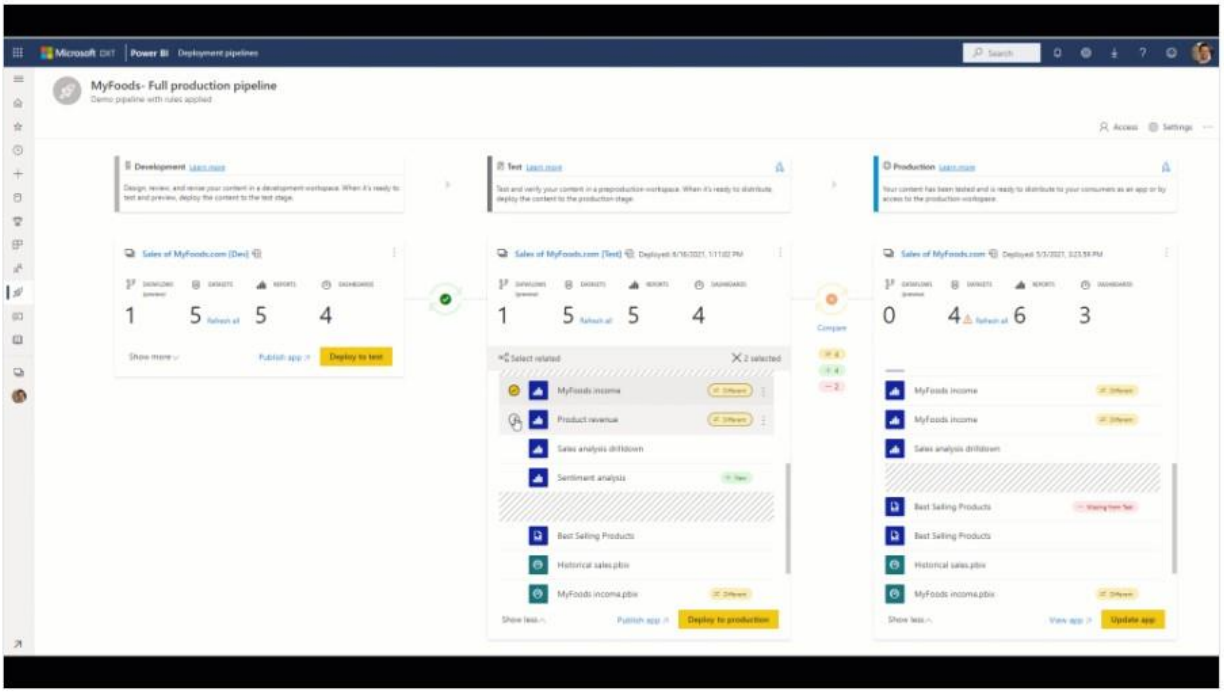
- **Test**

  You're ready to enter the test stage after you've made all the needed changes to your content. You upload the modified content so it can be moved to this test stage. Here are three examples of what can be done in the test environment:

  - **Share content with testers and reviewers.**
  - **Load and run tests with larger volumes of data.**
  - **Test your app to see how it will look for your end users.**

- **Production**

  After testing the content, use the production stage to share the final version of your content with business users across the organization.



3.3 Deployment method

When you deploy content from the source stage to a target stage, the source content will overwrite anything with the same name in the target stage. Content in the target stage that doesn't exist in the source stage remains in the target stage as is. After you select *deploy*, you'll get a warning message listing the items that will be overwritten.

3.4 Accessing deployment pipelines.

- You'll be able to access the deployment pipelines feature, if the following conditions are met:
- You have one of the following Premium licenses:
- You're a Power BI Pro user, and you belong to an organization that has Premium capacity.
- Premium Per User (PPU).
- You're an admin of a Premium workspace.

Step 1 - Create a deployment pipeline.

You can create a pipeline from the deployment pipelines tab, or from a workspace.

After the pipeline is created, you can share it with other users, edit, or delete it. When you share a pipeline with others, those will be given access to the pipeline and will become pipeline admins. Pipeline access enables users to view, share, edit, and delete the pipeline.

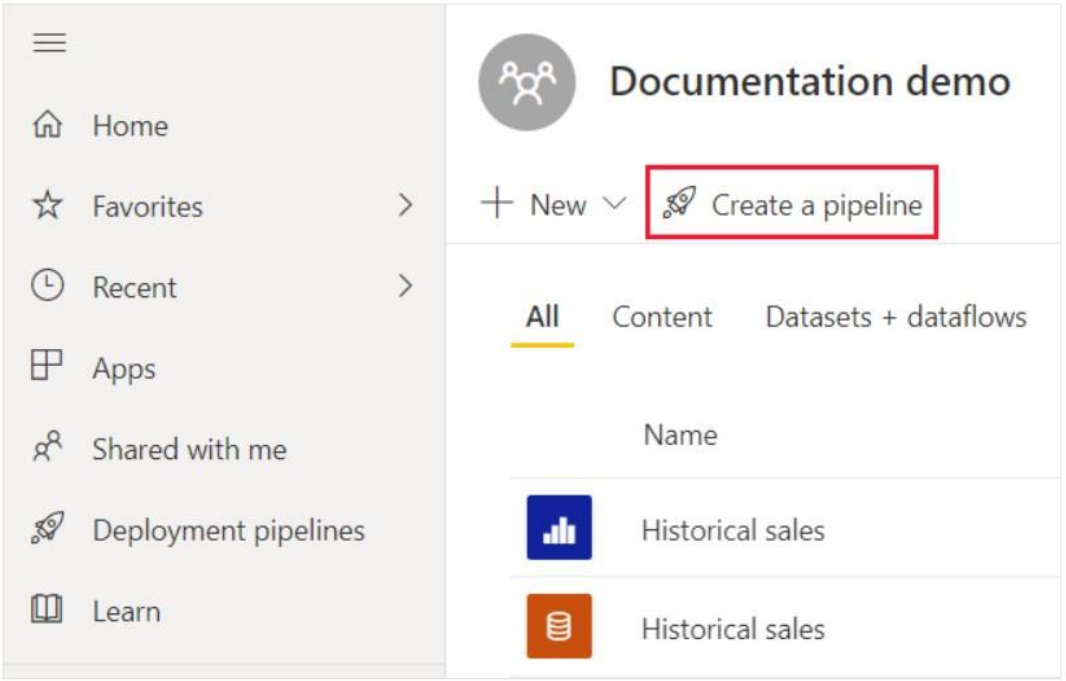
Create a pipeline from the deployment pipelines tab.

To create a pipeline from the deployment pipelines tab, do the following:

- In Power BI service, from the navigation pane, select Deployment pipelines and then select Create pipeline.
- In the Create a deployment pipeline dialog box, enter a name and description for the pipeline, and select Create.

Create a pipeline from a workspace.

1. From the workspace, select **Create a pipeline**.



2. In the *Create a deployment pipeline* dialog box, enter a name and description for the pipeline, and select **Create**.

Step 2 - Assign a workspace.

After creating a pipeline, you need to add the content you want to manage to the pipeline. Adding content to the pipeline is done by assigning a workspace to the pipeline stage. You can assign a workspace to any stage.

Step 3 - Deploy to an empty stage.

When you finished working with content in a pipeline stage, you can deploy it to the next stage. Deploying content to another stage is usually done after you've performed some actions in the pipeline. For example, made development changes to your content in the development stage, or tested your content in the test stage. A typical workflow for moving content from stage to stage, is development to test, and then test to production, but you can deploy in any direction. You can learn more about this process, in the deploy content to an existing workspace section.

Deployment pipelines offer three options when it comes to deploying your Power BI content:

- Deploying all content - Deploy all your Power BI content to the target stage.
- Selective deployment - Select which Power BI content to deploy to the target stage.
- Backwards deployment - Deploy your content to a previous stage in the pipeline.

Step 4 - Deploy content from one stage to another.

To deploy content to the next stage in the deployment pipeline, select the deploy button at the bottom of the stage.

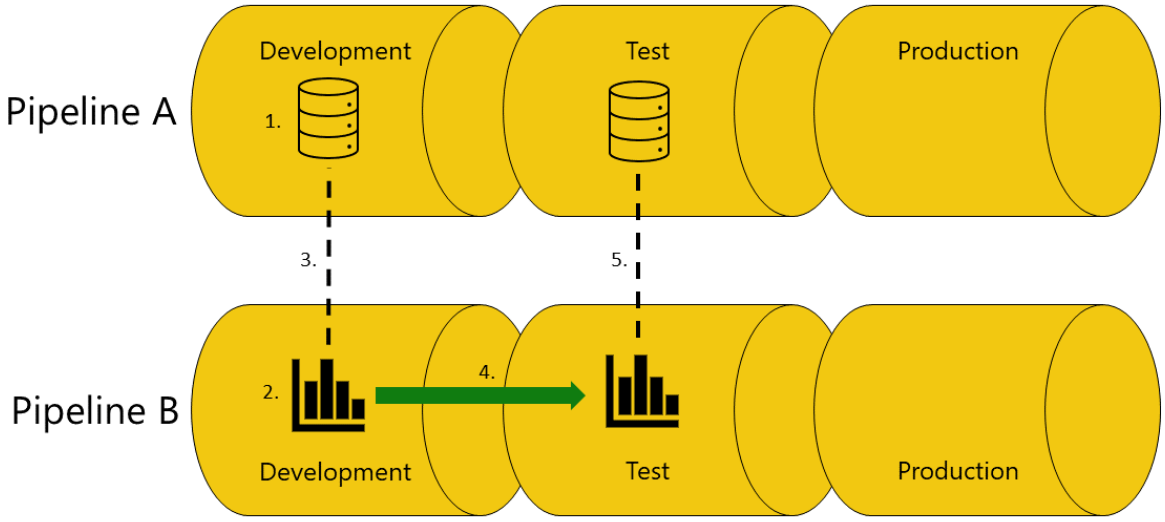
When reviewing the test and production stage cards, you can see the last deployment time. This indicates the last time content was deployed to the stage.

Deployment time is useful for establishing when a stage was last updated. It can also be helpful if you want to track time between test and production deployments.

Step 5 - Create deployment rules (optional)

When you're working in a deployment pipeline, different stages may have different configurations. For example, each stage can have different databases or different query parameters. The development stage might query sample data from the database, while the test and production stages query the entire database.

When you deploy content between pipeline stages, configuring deployment rules enables you to allow changes to content, while keeping some settings intact. For example, if you want a dataset in a production stage to point to a production database, you can define a rule for this. The rule is defined in the production stage, under the appropriate dataset. Once the rule is defined, content deployed from test to production, will inherit the value as defined in the deployment rule, and will always apply as long as the rule is unchanged and valid.


Development
Test
Production
Pipeline A
1.
3.
5.
Pipeline B
2.
4.
Development
Test
Production