

The MLOps guide

The MLOps guide

Mihai Criveti, Dominik Kreuzberger

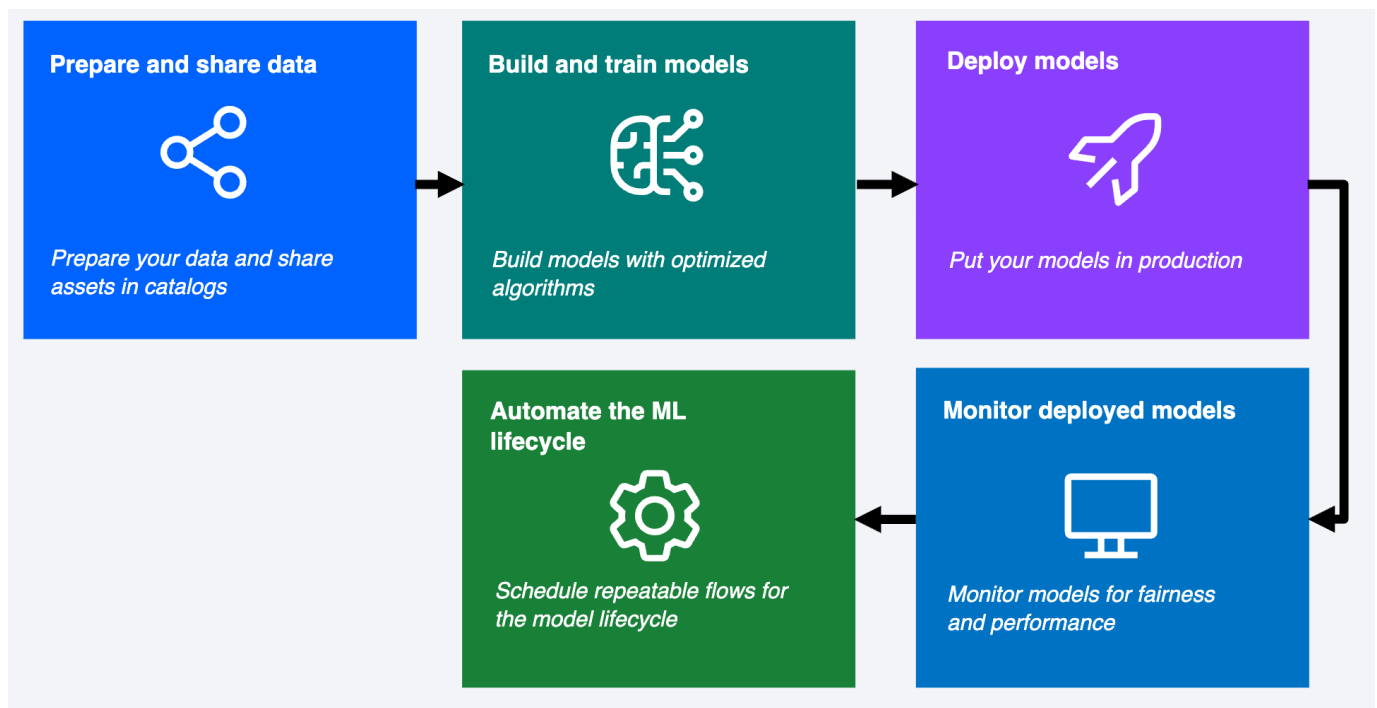
Copyright © IBM Open Innovation Community

Contents

1. What is MLOps?	3
2. MLOps with IBM Cloud Pak for Data	4
3. About this Repository	4
4. What would be a example definition of MLOps?	4
5. Products	5
5.1 Extend your AI capabilities with watsonx	5
5.2 Cloud Pak for Data	7
6. Realizing MLOps	8
6.1 Overview - Realizing MLOps	8
6.2 Data Stage and Data Refinery	11
6.3 Watson Studio	15
6.4 Watson Machine Learning	16
6.5 AutoAI (no-code)*	18
6.6 SPSS Modeler (low-code)*	20
6.7 Watson Pipelines	21
6.8 Watson OpenScale	24
6.9 AI Governance	28
7. watsonx	30
7.1 Introducing watsonx	30
8. MLOps Example Templates	31
8.1 Examples	31
9. Contributors	33
9.1 Contributors	33

1. What is MLOps?

Machine Learning Operations (MLOps) is a practice that builds on DevOps to create an automated and streamlined workflow specifically tailored for developing, deploying, and managing machine learning models and all associated assets. Machine learning models require extra consideration for monitoring models in production and retraining if the performance declines. Planning for MLOps and implementing a strategy is a step in the AI journey if a company wants to derive greater benefit from machine learning models and manage risk. Follow us as we provide a roadmap to becoming an AI-driven company.



The engineering practice of MLOps leverages three contributing disciplines: machine learning, software engineering (especially DevOps), and data engineering. The goal of MLOps is to bridge the gap between development (Dev) and operations (Ops) and create a repeatable process for training, deploying, monitoring, and updating machine learning models.

MLOps improves the collaboration between the different stakeholders in a data science project. An AI project might include data scientists, software engineers, and subject matter experts, among others. MLOps provides tools and processes so that collaborators can contribute their part to achieve a seamless flow from model request to deploying models to solve business problems.

IBM is here to help with your MLOps journey and support you in gaining a better understanding of how MLOps works and how IBM products can be used to build a solid MLOps ecosystem for your organization. IBM's Data and AI portfolio includes the tools and capabilities you need to facilitate successful MLOps. Explore the information on this website to learn more about how to plan, execute, and customize an MLOps process that works for your needs.

2. MLOps with IBM Cloud Pak for Data

What steps and services to consider when realizing MLOps with IBM's Cloud Pak for Data?

- Follow the [realizing MLOps steps in this Guide](#)
- Follow the [official documentation](#).

3. About this Repository

This GitHub repository has been created by several MLOps experts with expertise in IBM Cloud Pak for Data. Use this guide to learn how to design an MLOps strategy based on Cloud Pak for Data services.

4. What would be a example definition of MLOps?

MLOps (Machine Learning Operations) is a paradigm aimed at bridging the gap between development (Dev) and operations (Ops) to manage and automate the AI lifecycle. A full definition of MLOps can be found e.g. here [MLOps Paper](#).

 The latest version can always be found here: <https://ibm.github.io/MLOps/>

Download the latest (PDF)

Download the latest (DOCX)

Last update: April 8, 2024

Authors: [Dominik Kreuzberger](#), [Julianne Forgo](#), [Mihai Criveti](#), [Przemek Czuba](#), [Yvette Machowski](#)

5. Products

5.1 Extend your AI capabilities with watsonx

IBM watsonx.ai is a studio of integrated tools for working with generative AI capabilities that are powered by foundation models and for building machine learning models. IBM watsonx.ai provides a secure and collaborative environment where you can access your organization's trusted data, automate AI processes, and deliver AI in your applications.

You can accomplish the following goals with IBM watsonx.ai:

- **Build machine learning models** by using open source frameworks and code-based, automated, or visual data science tools.
- **Experiment with foundation models** using prompts to pre-trained foundation models to generate, classify, summarize, or extract content from your input text. Choose from IBM models or open source models from Hugging Face.
- **Manage the AI lifecycle** and automate the full AI model lifecycle with all the integrated tools and runtimes to train, validate, and deploy AI models.

5.1.1 Relationship with Cloud Pak for Data

IBM watsonx as a Service and Cloud Pak for Data as a Service have similar platform functionality and are compatible in many ways. The watsonx platform provides a subset of the tools and services that are provided by Cloud Pak for Data as a Service. However, watsonx.ai on watsonx extends the functionality of the common toolset to enable working with foundation models and generative AI.

Both platforms provide services for data science and MLOps use cases:

Watson Studio
Watson Machine Learning
AI governance

However, these services for watsonx.ai on the watsonx platform include features for working with foundation models and generative AI that are not included in these services on Cloud Pak for Data as a Service.

Cloud Pak for Data as a Service also provides services for these use cases:

Data integration
Data governance

5.1.2 Data science and AI tools

Both platforms provide a common set of data science and AI tools. However, on watsonx, you can also perform foundation model inferencing with the Prompt Lab tool or with a Python library in notebooks. Foundation model inferencing and the Prompt Lab tool are not available on Cloud Pak for Data.

The following table shows which data science and AI tools are available on each platform.

Tool	On watsonx	On Cloud Pak for Data
Prompt Lab	✓	
Synthetic Data Generator	✓	
Prompt lab	✓	
Synthetic Data Generator	✓	
Data Refinery	✓	✓
Visualizations	✓	✓
Jupyter notebooks	✓	✓
Federated Learning	✓	✓
RStudio IDE	✓	✓
SPSS Modeler	✓	✓
Decision Optimization	✓	✓
AutoAI	✓	✓
Watson Pipelines	✓	✓

Which one should you use?

The best platform for you will depend on your specific needs and requirements. If you are looking for a platform that can help you to multiply the impact of AI across your organization, then watsonx is a good option. If you are looking for a platform that can help you to manage your data and build AI applications, then Cloud Pak for Data is a good option.

For details on watsonx, including how you can try this technology, see [watsonx](#).

Ultimately, the best way to decide which platform is right for you is to evaluate your needs and requirements and then speak with an IBM representative.

Last update: November 6, 2023

Authors: [Julianne Forgo](#), [MaxJ](#)

5.2 Cloud Pak for Data

IBM Cloud Pak for Data is a cloud-native platform that helps you to build, deploy, and manage machine learning models at scale. It provides a unified set of tools and services for the entire MLOps lifecycle, from data preparation to model deployment and monitoring.

5.2.1 Features

- **Centralized data catalog:** Makes it easy to find and share data.
- **Self-service data preparation tool:** Helps you to clean and prepare your data for analysis.
- **Machine learning platform:** Makes it easy to build and deploy machine learning models.
- **Model registry:** Tracks the lineage and performance of your models.
- **Model monitoring tool:** Helps you to track the performance of your models in production.

5.2.2 Benefits

- **Reduces the time it takes to get your models into production.**
- **Improves the accuracy and performance of your models.**
- **Helps you to make better decisions based on your models.**
- **Saves money on your ML infrastructure costs.**

5.2.3 Conclusion

IBM Cloud Pak for Data is a great option for organizations that are looking to accelerate their MLOps journey. It is a powerful and flexible platform that can help you to build, deploy, and manage machine learning models at scale.

Last update: July 14, 2023

Authors: [MaxJ](#)

6. Realizing MLOps

`gitops` `cicd` `continuous-delivery` `git`

6.1 Overview - Realizing MLOps

IBM Cloud Pak for Data provides a full stack of tools to assist you in realizing your MLOps strategy. The services work together to provide all of the capabilities you need to work with data, create AI assets, deploy the assets for productive use, and monitor and manage deployed assets.

Note: You do not have to use the full list of Cloud Pak for Data services listed here. You can choose the services you need to realize your MLOps solution.

Key Cloud Pak for Data services for implementing MLOps include:

- **Data Stage** and **Data Refinery** for integrating and preparing data to train machine learning models.
- **Watson Studio** for organizing and creating assets, including data, models, and notebooks. "+ experiment tracking in Factsheets after each run."
- **Watson Machine Learning** for organizing the resources, including environments, for deploying and managing machine learning models, scripts, and functions. Watson Machine Learning includes also:
- **AutoAI (no-code)** for automating training of machine learning models (for example, data scientists can use AutoAI to rapidly prototype a solution.)
- **SPSS Modeler (low-code)** build an ML model by dragging and dropping operators and assets on a canvas and running a model as a flow.
- **Watson Pipelines** for automating a repeatable flow to manage the assets in your MLOps pipeline.

Key Cloud Pak for Data services for implementing the AI Governance part of your MLOps strategy include:

- **Watson OpenScale** to evaluate and monitor deployments to ensure they perform to expectations for such dimensions as fairness, quality, and drift.
- **OpenPages** manage your workflows by meeting governance regulations, policies, regulations for your assets (models) and build trust in your solution.
- **AI Factsheets** captures model metadata, in all stages, from request to production.

The MLOps suite of services all provide rich user interfaces in Cloud Pak for Data for building and managing assets. For example, monitor all of your deployed assets from a Deployments dashboard in Watson Machine Learning, or assemble and run your MLOps flow from a Watson Pipelines canvas.

Alternatively, you can use programming interfaces to code MLOps assets and processes. Watson Machine Learning provides these programming interfaces:

- Use the [Python client library](#) to work with all of your Watson Machine Learning assets in a notebook.
- Use the [REST API](#) to call methods from the base URLs for the Watson Machine Learning API endpoints.

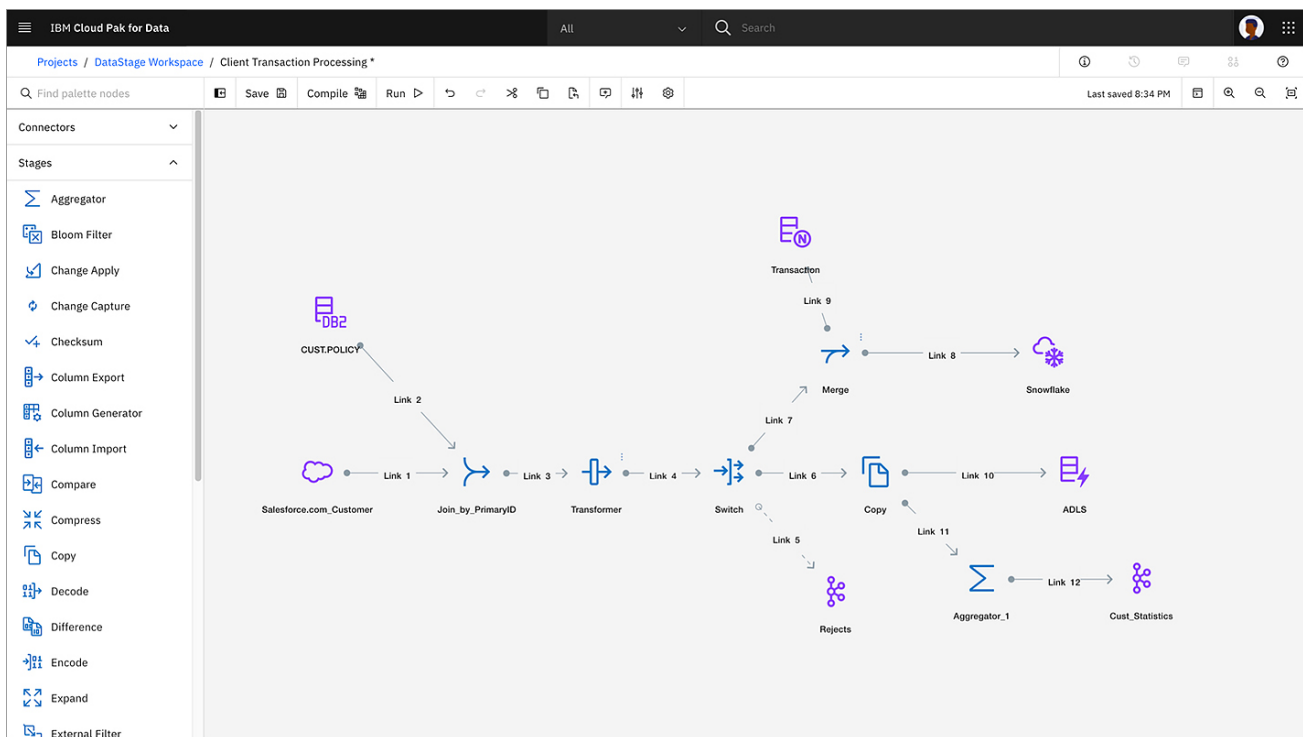
Last update: November 1, 2023

Authors: [Dominik Kreuzberger](#), [Julianne Forgo](#)

gitops ckd continuous-delivery git

6.2 Data Stage and Data Refinery

With the DataStage data integration tool, data engineers can design and run complex ETL data pipelines that move and transform data between operational, transactional, and analytical target systems. Data integration specialists use DataStage to develop flows that process and transform data. DataStage has capabilities that allow connecting directly to enterprise applications, cloud data sources, relational and NoSQL systems, REST endpoints, and more. You can administer, manage, deploy, and reuse these flows to integrate data across many systems throughout your organization.



For more information check out the [official documentation](#).

Data Refinery involves transforming raw data into clean, structured information for analysis and decision-making. It includes data collection, cleansing, integration, enrichment, and validation. Data is gathered from various sources, cleaned to remove errors, and integrated into a unified format. Enrichment adds additional context and attributes. Validation ensures data quality. Data refinery unlocks the value of data, enabling informed decisions and actionable insights.

Steps (3)

×

Data Source

mydata.csv

1. Convert column type

AUTOMATIC

Automatically converted one or more columns to inferred data types. Strings that are converted to decimal use a dot (.) for the decimal symbol.

2. Sort ascending

Sorted rows by ID

3. Replace missing values

JUST ADDED

Replaced missing values in city with Bonn into city

Use a code template to add a step

Data

Profile

Visualizations

	Animal String	city String	ID Integer	Phone Integer
1	horse	Tucson	3	NA
2	cat	Minneapolis	225	444
3	bird	San Francisco	25253	664
4	dog	Bonn	NA	866

- 13/34 -

Copyright © IBM Open Innovation Community

Last update: October 18, 2023

Authors: [Dominik Kreuzberger](#), [moritzscheele](#)

gitops cicd continuous-delivery git

6.3 Watson Studio

Watson Studio is part of Cloud Pak for Data and provides the data science capabilities of the data fabric architecture. Watson Studio provides the environment and tools for you to collaboratively work on data to solve your business problems. You can choose the tools you need to analyze and visualize data, to cleanse and shape data, to ingest streaming data, or to create and train machine learning models.

The screenshot displays the IBM Watson Studio interface. The top navigation bar includes the IBM logo, a search bar, and a 'Buy' button. The breadcrumb trail shows 'Projects / Customer data / Precipitation data analysis'. The main workspace contains a Jupyter Notebook with the following code:

```
In [1]:
import os, types
import pandas as pd
from botocore.client import Config
import boto3

def __iter__(self): return 0

#@hidden_cell
# The following code accesses a file in your IBM Cloud Object Storage. It includes your credentials.
# You might want to remove those credentials before you share the notebook.
client_2d0ac51a1f3647bfb66c445dd2b711f = boto3.client(service_name='s3',
    iam_api_key_id='Adu0JBP17He3L3dvyh141xLxw66zE5QgMOQCE1qTODff',
    iam_auth_endpoint='https://iam.cloud.ibm.com/oidc/token',
    config=Config(signature_version='oauth'),
    endpoint_url='https://s3.private.us.cloud-object-storage.appdomain.cloud')

body = client_2d0ac51a1f3647bfb66c445dd2b711f.get_object(Bucket='customerdata-donotdelete-pr-qfb0qd6xytllx',Key='precipitation.csv')['Body']
# add missing __iter__ method, so pandas accepts body as file-like object
if not hasattr(body, "__iter__"): body.__iter__ = types.MethodType(__iter__, body)

df_data_1 = pd.read_csv(body)
df_data_1.head()
```

The output of the code is a table showing precipitation data for various countries and areas from 1990 to 2004. The table has 13 columns: Country or Area, 1990, 1995, 1996, 1997, 1998, 1999, 2000, 2001, 2002, 2003, 2004, and a final column with values like 4284, 51, 12, 37.

On the right side of the interface, the 'Environment' tab is selected, showing the following configuration:

- General: Environment
- Environment definition: Runtime 22.1 on Python 3.9 X
- Language: Python 3.9
- Hardware configuration: 2 vCPU 8 GB RAM
- Software configuration: View details
- Runtime status: Running

For more information check out the [official documentation](#).

Creating Jupyter Notebooks in Watson Studio [official documentation](#).

Last update: October 27, 2023

Authors: [Dominik Kreuzberger](#)

ml model_training model_development

6.4 Watson Machine Learning

[Watson Machine Learning](#) (WML) provides a range of tools and services for you to build, train, test, and deploy models in Cloud Pak for Data.

Depending on what is installed and configured for your deployment, you can use Watson Machine Learning to:

- Build, train, and deploy models from notebooks by using the Watson Machine Learning Python client library or the Watson Machine Learning API.
- Create AutoAI experiments. AutoAI automatically preprocesses your structured data, selects the best estimator for the data, and then generates model candidate pipelines for you to review and compare. Deploy the best-performing pipeline as a machine learning model.
- Run experiments to train complex Deep Learning models in Experiment Builder.
- Deploy your models so that you can score the models and generate predictions.
- Evaluate online deployments (requires Watson OpenScale service).

6.4.1 Training models with Watson Machine Learning

Watson Machine Learning supports training machine learning models with tools that provide automation or autonomy matching your needs. Build a custom model using popular Machine Learning frameworks such as [PyTorch](#) or [TensorFlow*](#)), or fully automate model creation and training with [AutoAI](#).

AutoAI enables you to build and deploy machine learning models without coding. Use it as a rapid prototyping tool or as part of your end-to-end machine learning solutions. You can rely on the auto-detection features that analyze the training data, select a model type, and apply algorithms and tuning features, or you can customize the configuration to exercise finer control. Save the best model-candidate pipeline as a deployable model, or save the model code as a notebook so you can review the code and customize as needed. AutoAI can be a powerful part of your machine learning solution.

6.4.2 Deploying models and other assets with Watson Machine Learning

Using IBM Watson Machine Learning, you can collect and organize all of the dependencies required to bring a model, script, function or web app from training to production. For example, create a pre-production collaborative space where you can upload the deployable asset and testing data for validating a deployment before moving it to a production space and putting it to work predicting outcomes based on real-world input data.

[Managing Deployments - Official documentation](#)

6.4.3 Extending Watson Machine Learning to govern machine learning assets

As part of your ModelOps strategy, you can extend Watson Machine Learning to include governance features, provided with the Watson OpenScale service. Once your tools and services are connected, you can monitor and evaluate online deployments for dimensions such as fairness, quality, and drift. Machine learning models require vigilance to make sure they do not stray from their intended mission. Part of the MLOps pipeline is to measure the machine learning outcomes and retrain and update the model and deployment when performance or outcomes fall below the thresholds you establish.

Watson OpenScale also provides tools for creating what-if scenarios to better understand how a model is performing or to find better approaches to solving the business problem. Finally, track and collect the metadata for assets as they move through the AI lifecycle to ensure compliance with governance standard. The combination of Watson Machine Learning and Watson OpenScale gives you the power to create AI solutions that are effective, responsible, and trustworthy.

Last update: October 31, 2023

Authors: [Dominik Kreuzberger](#), [Ilias Ennmouri](#), [Julianne Forgo](#), [Przemek Czuba](#)

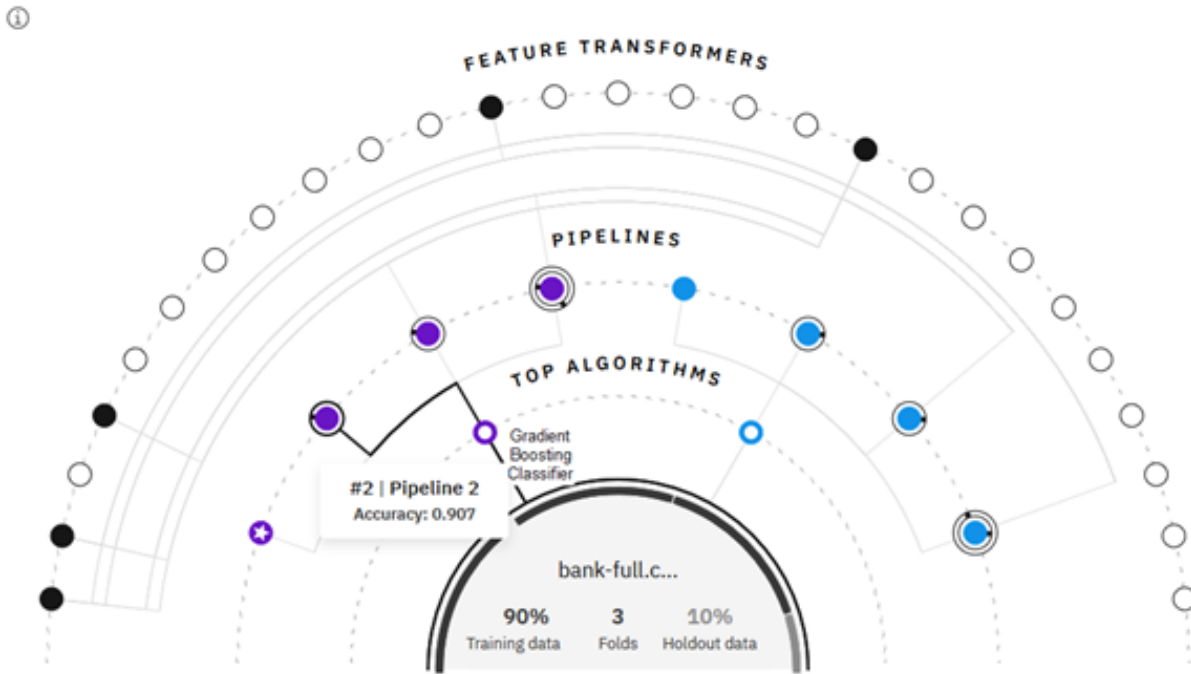
autoai model_training cicd

6.5 AutoAI (no-code)*

**Use of AutoAI remains optional for your MLOps workflow. Depending on your scenario it may increase efficiency and model quality.*

With AutoAI, you can train and save machine learning models without coding. The AutoAI tool in CP4D does most of the work for you. AutoAI uses sophisticated training capabilities to build models from your structured data sets. AutoAI automatically runs all major tasks that are part of building and ranking candidate model pipelines:

- **Data pre-processing:** AutoAI analyzes, cleans, and prepares your raw data for machine learning by applying various algorithms, or estimators. Unlike standard machine learning algorithms, it can work with various data formats and is able to handle missing values with data imputation methods you configure.
- **Automated model selection:** AutoAI analyzes the training data, then suggests the model type(binary classification, multiclass classification, or regression) that best matches the data. If you are working with sequenced data/time data, you can also configure the model selection to create a time-series model.
- **Automated feature engineering:** AutoAI explores various feature construction choices while progressively improving model accuracy with reinforcement learning to transform the data into the combination of features that best represents the problem for the most accurate prediction.
- **Hyperparameter optimization:** The AutoAI hyper-parameter optimization uses a novel hyperparameter optimization algorithm to rapidly find and apply optimizations for the model-candidate pipelines.



Integrating AutoAI in your MLOps workflow

The way you use AutoAI can vary depending on your use case. For example, you can use AutoAI to fully train models for your scenario. You can also use AutoAI to complement or accelerate your data science, by using it for specific tasks, such as model selection or hyperparameter optimization. As part of your MLOps planning, consider one of these approaches:

If one of the above is true, or a similar scenario occurs, you are faced with two options.

- **Full Reliance:** You can implement AutoAI within the Watson Studio Pipeline of your end-to-end MLOps workflow as sole Model Selector, Model Trainer *etc.*
- **Semi Reliance:** Alternatively, you can run AutoAI in parallel to your custom `train_model.ipynb` notebook. In that case, your model training notebook will train your custom-built model while AutoAI trains all pre-developed models. You can then proceed to either deploy your custom-built model, or the model selected by AutoAI, depending on performance metrics you established earlier.

For more information check out the [official documentation](#).

Examples [official documentation](#).

Last update: October 31, 2023

Authors: [Dominik Kreuzberger](#), [Ilias Ennmouri](#), [Przemek Czuba](#)

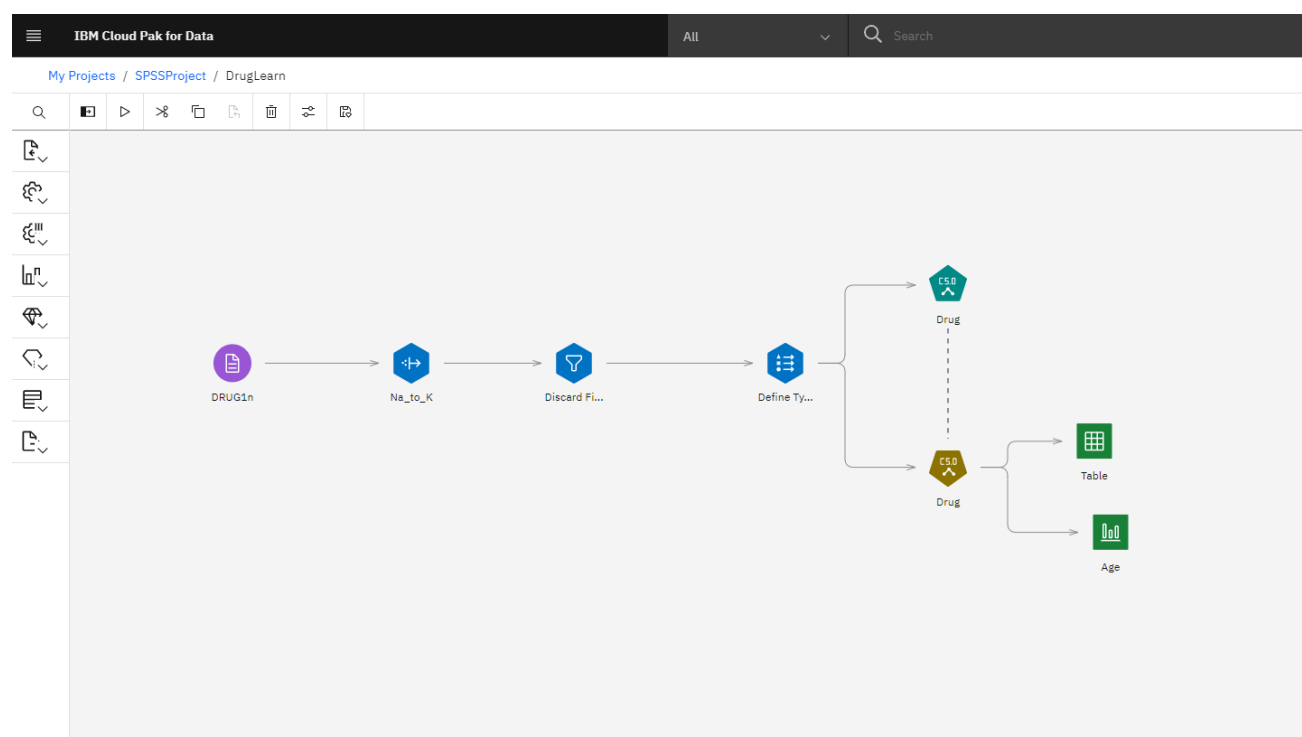
autoai model_training cicd

6.6 SPSS Modeler (low-code)*

**Use of SPSS Modeler remains optional for your MLOps workflow. Depending on your scenario it may increase efficiency and model quality.*

You can use SPSS Modeler flows to build machine learning pipelines that you can use to iterate rapidly during the model building process. Whether you're trying to find the best algorithm or experimenting with different ways of preparing your data, you can create reproducible research that's easily understood by any member of your team.

With SPSS Modeler flows, you can quickly develop predictive models using business expertise to improve decision making. The flows interface is based on the long-established SPSS Modeler client software and uses industry-standard CRISP-DM methodology. The SPSS Modeler service supports the entire data mining process, from data exploration all the way to better business results.



For more information check out the [official documentation](#).

Examples [official documentation](#).

Last update: October 31, 2023

Authors: [Dominik Kreuzberger](#)

gitops cicd continuous-delivery git

6.7 Watson Pipelines

With the Watson™ Pipelines service, you can create a pipeline to automate the end-to-end flow of various assets, whether it's training a model or running script-based assets from the time they are created through their deployment. Automating the end-to-end flow of these assets with a pipeline makes it simpler to build, run, and evaluate models, which speeds up the flow and reduces the overall time investment.

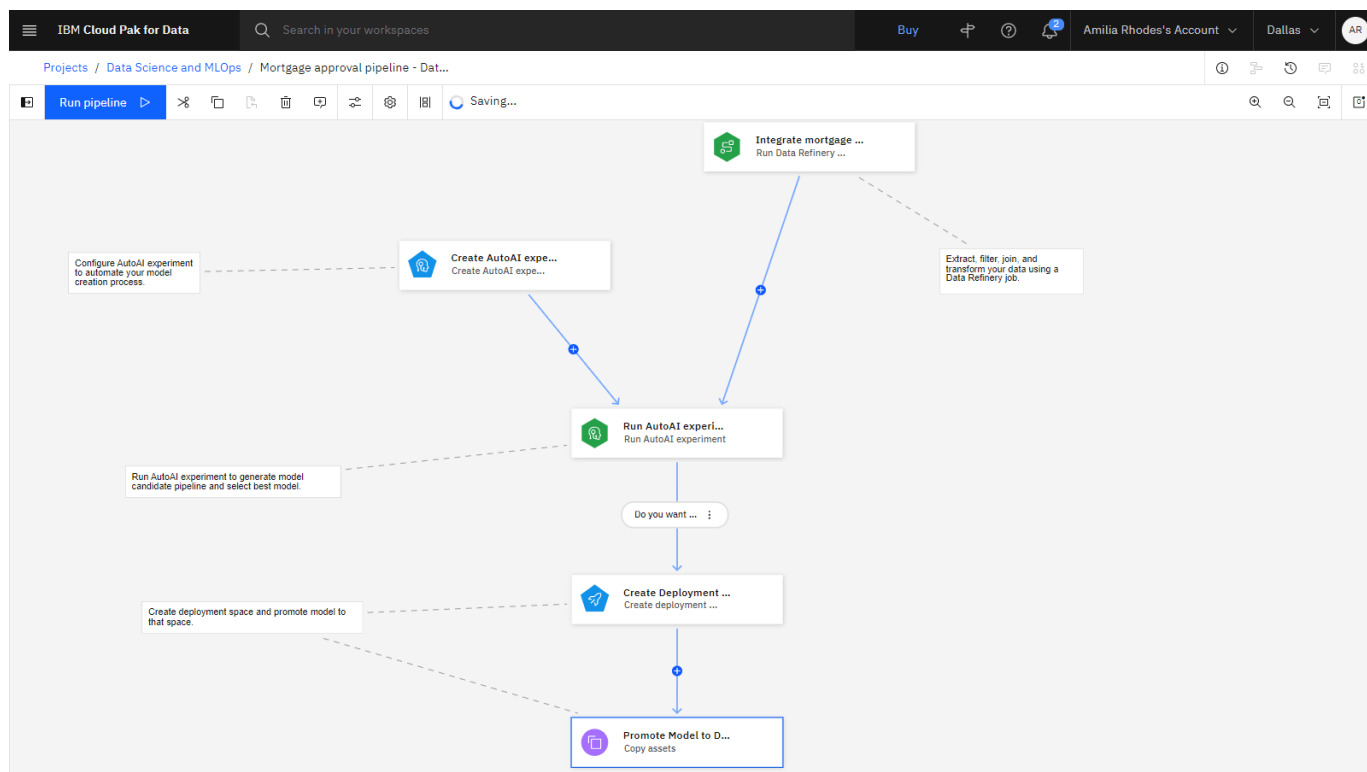
You use a pipelines editor canvas to assemble and configure a pipeline that creates, trains, deploys, and updates machine learning models and Python scripts. To design a pipeline, you drag nodes onto the canvas, specify objects and parameters, then run and monitor the pipeline. Note that you must have the required service for any asset you include in your pipeline. For example, if you are cleaning data with DataStage, the DataStage service must be installed or provisioned for your Cloud Pak suite.

Your team can collaborate across roles in the pipelines editor. For example, a data scientist can create a flow to train a model in the editor, and then a ModelOps engineer can add the steps to the flow to automate the process of training, deploying, and evaluating the model to a production environment.

After you assemble the pipeline, you can rapidly update and test modifications with the Pipelines editor canvas, which provides tools to visualize the pipeline, customize it at run time with pipeline parameter variables, and then run it as a trial job or on a schedule.

These tools are available with the Watson Pipelines service:

- Create a flow to collect data, run scripts, train models, store results, and more.
- Customize a unique pipelines component that can run a user-written function.
- Schedule jobs to run flows and enhance automation by adding node conditions.



For more information check out the [official documentation](#).

Example Use [official documentation] (<https://www.ibm.com/docs/en/cloud-paks/cp-data/4.7.x?topic=assets-watson-pipelines>).

Last update: October 31, 2023

Authors: [Dominik Kreuzberger](#)

gitops ccd continuous-delivery git

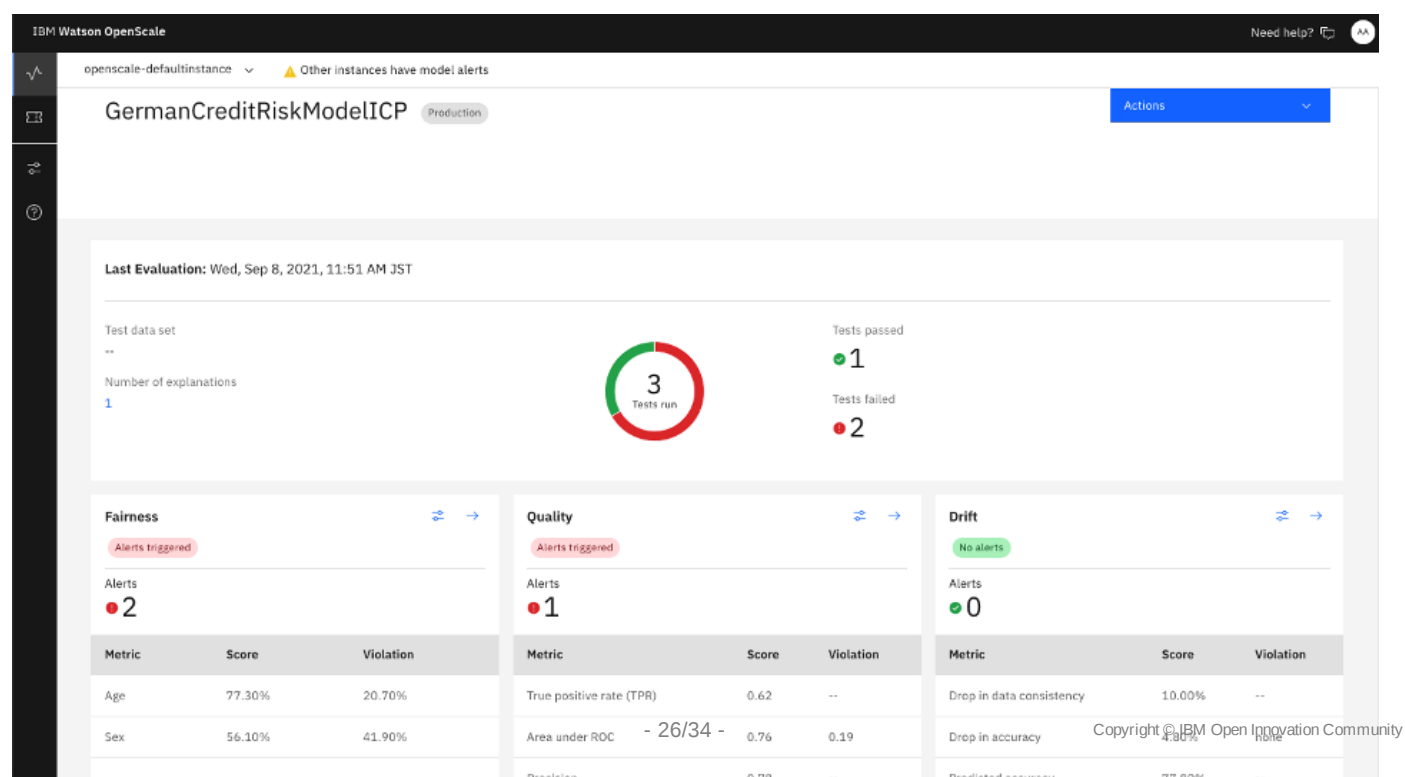
6.8 Watson OpenScale

After you deploy a machine learning model, the work doesn't stop. To guarantee that the model is functioning in production as expected, you must have a plan for monitoring the deployed model and updating it as needed. As part of your end-to-end MLOps process, consider IBM Watson OpenScale to evaluate model deployment to make sure they are fair, accurate, and performing to your standards. •

When OpenScale is installed or provisioned as part of your Cloud Pak suite, you can provide the details for a deployment, then run scheduled evaluations that measure dimensions you configure for thresholds you set. For example, if you want to test whether predicted outcomes are fair across various age groups, you can configure the Fairness monitor to evaluate the outcomes for a monitored group, such as young adults, and compare the results to the age group most likely to get favorable results. If the results deviate more than a threshold you specify, you will get an alert that results require attention. The dimensions you can test are:

- **Fairness:** Configure a monitor for fairness to check if your model produces biased results for different groups, like gender or race. Set thresholds to measure predictions for a monitored group compared to a reference group.
- **Quality:** Configure a monitor for quality to assess your model's performance based on labeled test data. Set quality thresholds to track when a metric value falls outside an acceptable range.
- **Drift:** Configure a monitor for drift to ensure your deployments are up-to-date and consistent. Use feature importance to determine the impact of feature drift on your model.
- **Explainability:** Configure explainability settings to understand which features influence your model's predictions. Different methods like SHAP and LIME are available to suit your needs.

All of the evaluation results can be reviewed and monitored in a single dashboard, as shown in this example:



Last update: October 27, 2023

Authors: [Dominik Kreuzberger](#), [Julianne Forgo](#), [Przemek Czuba](#), [moritzscheele](#)

`gitops` `cicd` `continuous-delivery` `git`

6.9 AI Governance

AI governance is about ensuring greater transparency across the AI lifecycle and the model itself. IBM recently announced `watsonx.governance`, a next generation enterprise toolkit which is designed to automate and accelerate workloads **across the AI lifecycle** while providing **risk management** and facilitating **regulatory compliance**. Use IBM AI Governance services to accelerate responsible, transparent, and explainable AI workflows with an AI governance solution that provides end-to-end monitoring for machine learning. Monitor machine learning assets from request to production. Collect facts about models that are built with IBM tools or third-party providers in a single dashboard to aid in meeting compliance and governance goals. Implement an approval workflow to meet compliance goals.

These Cloud Pak for data services can be used individually or together as part of your governance and MLOps plan.

AI Factsheets allows automated collection and documentation of model metadata at all stages, from model idea to production

Model and process metadata is captured in a central meta store. Having all model facts in central place is important both to increase the productivity of the MLOps process (model facts are immediately visible to all parties involved in the lifecycle of an AI model) and to comply with regulatory requirements. Data scientists benefit from assistance and automation of the documentation process. Transparency of model metadata supports audits and brings more clarity to stakeholder or customer requests. Metadata captured in AI factsheets includes model details, training information, metrics, input and output schemas, or details about the models used, such as quality metrics, fairness or drift details.

[AI Factsheets - Official Documentation](#) [AI Governance - Factsheets - Official Documentation](#)

OpenPages: Govern models through the complete AI workflow considering policies and regulations

The next generation governance-toolkit provides a range of capabilities to identify, manage, monitor, and report on risk and compliance. It accelerates the creation of models at scale, from use case idea (model candidates) to production deployment, by incorporating approvals in the workflow-based approach. Full transparency of any type of model (e.g., task specific data science artefacts or foundation models) is ensured and made visible in customisable risk monitoring dashboards. Additionally in Open Pages corporate policies and regulations can be assigned to models, e.g., annual bias review (required for EU AI ACT) to ensure that models are fair, transparent, and compliant [4].

[OpenPages - Official Documentation](#)

OpenScale allows to monitor, explain, and benchmark your model

Model monitoring is an ongoing task to track models and to drive transparency. This includes the monitoring of the general model performance (e.g., accuracy) and more specifically monitoring of fairness or model and data consistency over time (i.e. drift). Open Pages supports threshold definitions for model performance metrics and combines those with automated detection of threshold violations to trigger model retraining. It implements explainability by supporting explanations how the model arrived at certain predictions. Model benchmarking is supported – it is common practice to compare and benchmark a challenger model with a model in production to ensure that the best model is the one in production.

[See OpenScale in this Handbook.](#)

For more information check out the [official documentation](#) or the [example Notebooks](#).

Last update: April 8, 2024

Authors: [Dominik Kreuzberger](#), [Julianne Forgo](#)

7. watsonx

7.1 Introducing watsonx

Machine Learning Operations (MLOps) is the process of moving machine learning models from development and testing to production. MLOps is now extended to support training of foundation models and putting foundation model assets into productive use. Foundational models are the basis of generative AI. Large Language Models (LLMs) are a type of foundation models that support language-based generative AI used for a variety of applications, including advanced chat or summarization. A large language model takes input in the form of a prompt and generates output. Operating large language models in production is MLOps for LLMs called LLMOps.

Overview of watsonx:

The platform
for AI and data

watsonx

Scale and
accelerate the
impact of AI with
trusted data.

watsonx.ai

Build, train, validate, tune and
deploy AI models

A next generation enterprise studio for AI builders to build, train, validate, tune, and deploy both traditional machine learning and new generative AI capabilities powered by foundation models. It enables you to build AI applications in a fraction of the time with a fraction of the data.

watsonx.data

Scale AI workloads, for all
your data, anywhere

Fit-for-purpose data store, built on an open lakehouse architecture, supported by querying, governance and open data formats to access and share data.

watsonx.governance

Accelerate responsible,
transparent and explainable AI
workflows

End-to-end toolkit for AI governance across the entire model lifecycle to accelerate responsible, transparent, and explainable AI workflows

For the latest news on watsonx offerings and capabilities, see <https://www.ibm.com/watsonx>.

Last update: November 10, 2023

Authors: [Dominik Kreuzberger](#), [Julianne Forgo](#)

8. MLOps Example Templates

8.1 Examples

8.1.1 CP4D MLOps Example Templates



For example, the [CP4D MLOps Accelerator](#) is a step-by-step guide to help you set-up your Cloud Pak for Data environment to facilitate efficient MLOps.


The accelerator documents the implementation of a simple end-to-end MLOps workflow that uses services from the Cloud Pak for Data software stack to demonstrate the full AI lifecycle, from training to production. The accelerator is specifically built to demonstrate the creation of a basic workflow, while allowing for rapid customization. For example, you can tailor this flow to include your custom-built PyTorch or TensorFlow models.

Use this accelerator as a guide to building your own MLOps pipeline.

We include:

- full documentation of the CP4D environment set-up.
- sample notebooks that populate the environment with MLOps steps, including `data_validation`, `model_training`, and `model_deployment`.
- a pipeline that connects the standalone notebooks sequentially to create a repeatable, automated end-to-end flow.

8.1.2 Additional Examples

- The sample [mlops-sustainability-oss](#)  demonstrates a self-sustaining end-to-end CP4D MLOps workflow for a flood forecasting model with automated data/model versioning and rollback.

8.1.3 More Examples (coming soon)

- We are continuously improving our assets. More example assets will be added shortly.

Last update: June 3, 2024

Authors: [Dominik Kreuzberger](#), [Ilias Ennmouri](#), [Julianne Forgo](#)

9. Contributors

9.1 Contributors

This MLOps Guide has been created by several different technical communities: Customer Success, Client Engineering, Expert Labs, Documentation Team, Consulting, Development and Infrastructure. Special thanks to our Executive Champions Kate Blair and Hardy Gröger for supporting this initiative!

Name	IBM Unit/Description
Hardy Gröger	Distinguished Engineer & Technical Lead Data and AI DACH
Kate Blair	Director, Product Management, watsonx.ai
Dominik Kreuzberger	Customer Success
Maximilan Jesch	Customer Success
Hasan Özdemir	Customer Success
Carsten Holtmann	Customer Success
Ilias Ennmouri	Customer Success
Yvette Machowski	Customer Success
Alejandro González	Customer Success
Daniel Horn	Customer Success
Andrea Bartoš	Customer Success
Benedikt Bothur	Customer Success
Ivan Iliash	Customer Success
Moritz Scheele	Expert Labs
Lena Eckstein	Expert Labs
Lucas Baier	Client Engineering
Leonard Brucksch	Client Engineering
Michael Vössing	Client Engineering
Ellen Hoeven	Client Engineering
Tim Noah-Schmidt	Client Engineering
Michael Daschner	Development
Mihai Criveti	Consulting
Maximilian Wurzer	Consulting
Sebastian Lehrig	Infrastructure
Julianne Forgo	Documentation Team - Data & AI
Przemek Czuba	Documentation Team - Data & AI

Last update: April 8, 2024

Authors: [Dominik Kreuzberger](#), [Mihai Criveti](#)