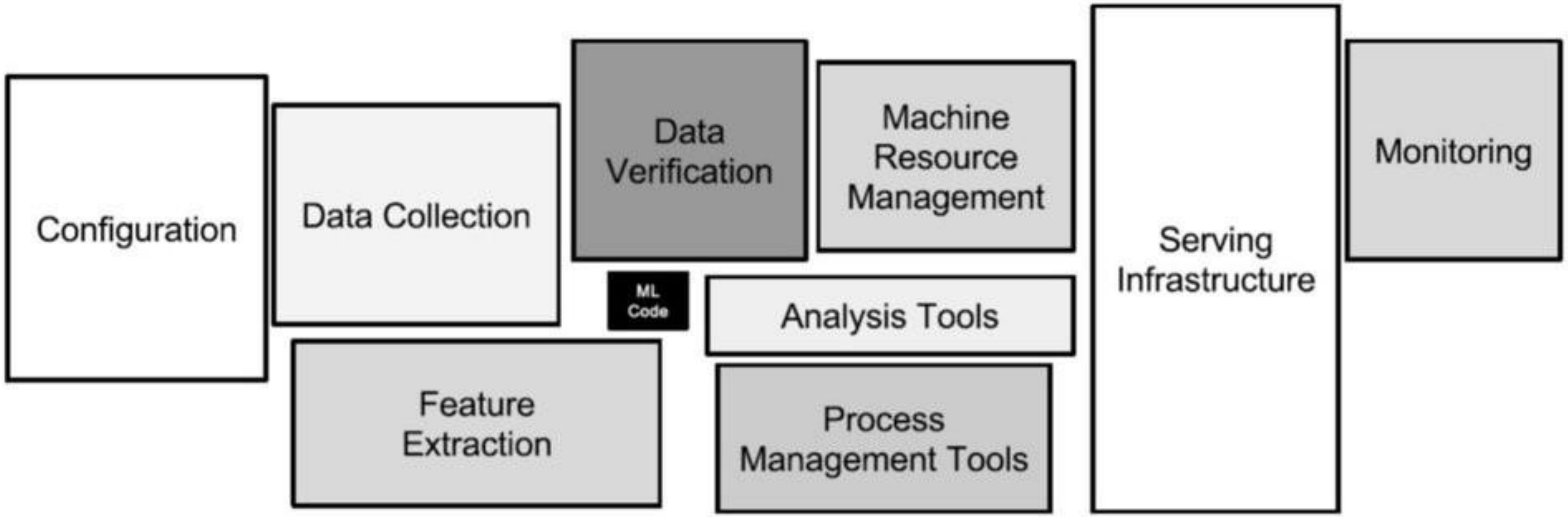
PACE - ML

Mphasis MLOps Framework & Methodology

Machine Learning Project Complexities



Source: Hidden Technical Debt in Machine Learning Systems, Sculley, Holt et al.

Machine Learning Projects are different because….

1. **Skills**: The team consists of data scientists, data architects, data engineers and business analysts who focus on data analysis, model development, experimentation & visualisation.
2. **Model Training & Development**: ML development is experimental and is different from traditional programming. Teams work on model features, algorithms, and configurations iteratively and model auditing, data & model versioning and reproducibility are critical. Teams should be able to create a training pipeline that runs across several machines and can be reused by others. Automated model training is a key need.
3. **Collaboration**: Developing a successful system requires collaboration across multiple groups in an iterative manner. Developers should be able to explore past knowledge, results of experiments across versions, peer-to-peer sharing and branch out new variants of experiments. Teams often use basic office tools resulting in lost productivity.
4. **Testing**: Testing an ML system involves input data validation, model quality, model performance, model validation, explainability, infrastructure testing, pipeline integration testing, API testing, data drift testing. Model reuse is different than software reuse, as models must be tuned based on input data / scenario.
5. **Deployment**: ML systems may need a multi-step pipeline to automatically retrain and deploy models. Teams should be able to create deployment pipelines that runs across several machines and can be reused by others. It should support model portability across a variety of platforms and should be able to monitor & know when to retrain given scenarios such as data drift.
6. **Production**: ML models can have reduced performance not only due to suboptimal coding, but also due to constantly evolving data profiles. In other words, models can decay in more ways than conventional software systems, and you need to consider this degradation. Therefore, you need to track summary statistics of your data and monitor the online performance of your model to send notifications or roll back when values deviate from your expectations.

Introducing Mphasis PACE-ML

PACE - ML is Mphasis Framework for machine learning development and deployment

PACE - ML is a combination of Mphasis proprietary tools and methodologies along with best in-class third-party as well as open-source tools

PACE – ML empowers data scientists and app developers to help bring ML models to production *faster* and at *scale*.

PACE – ML improves *efficiency* and *streamline* the management of model selection, reproducibility, versioning, auditability, explainability, packaging, re-usability, validation, deployment & monitoring.

PACE-ML OFFERINGS

Standardized frameworks and checklists for
- ML Assessment questionnaires
- MLOps project set-up guidelines

Collaboration frameworks & Infrastructure for
- Versioning for data, code, and models
- Experiment tracking
- Standardized notebooks for collaboration
- Model hub

Automated Gitactions Pipelines for
- Code checking
- Unit testing
- Containerization
- Model deployments
- Model retraining pipelines

Modules for automated Data Preparation
- Data imputation, missing values, normalization
- Data encoding
- Data profiling
- Data augmentation

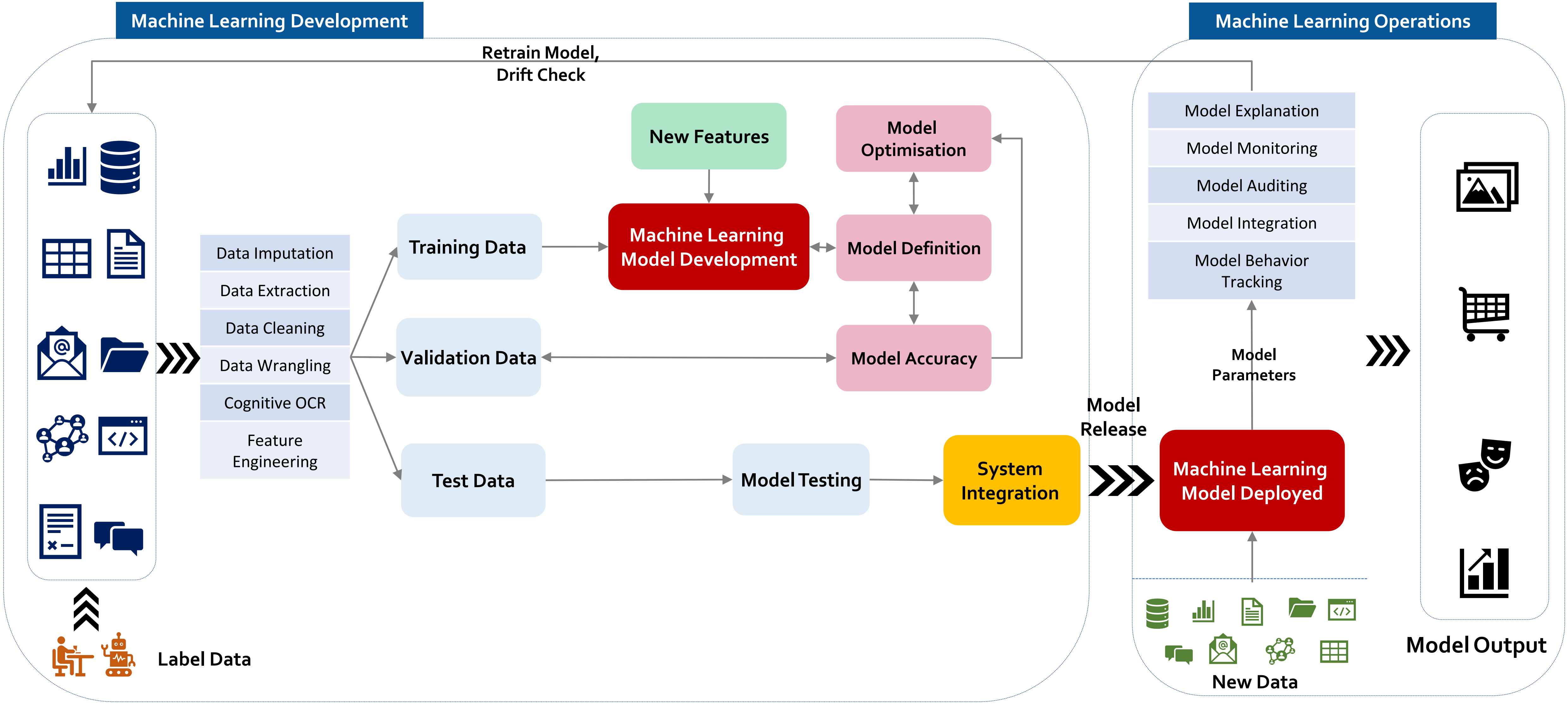
Modules for Feature engineering
- Feature interactions
- Polynomial features
- Grouping and binning features

AutoML for
- Compare best models for regression, classification, and clustering
- Hyperparameter tuning

Deployment pipelines for
- Containerize
- Orchestration

Monitoring for
- Drift check APIs
- Explainability APIs
- Infrastructure tracking dashboard
- Model monitoring dashboard

PACE-ML - Workflow



PACE-ML – Building Blocks

Machine Learning Development

Data	Model Development	Testing
Data pipeline	Model training pipeline	Testing pipeline
Data Extraction from unstructured sources*	Feature Engineering	Model correctness, performance, relevance, AI explainability*
Data Cleaning & Wrangling*	Automated model selection*	Model efficiency, robustness, fairness, interpretability
Version Control of Code and Data	Automated training	Packaging, infrastructure testing, pipeline integration testing, API testing
Data Tagging & Labeling	Model reproducibility, versioning	Data & model drift testing
	Model Packaging	Automated Testing

Model Release →

← Retrain Model

Machine Learning Operations

Deployment	Production
Deployment Pipeline	Governance
Model portability across different platforms	Monitoring
Automated Deployment	Model behavior tracking
	Model performance*
	Explainability*
	Auditing, Compliance

DeepInsights™ DeepInsights™

Collaboration

Version Control of Model, Code and Data	Jointly build, select and track model versions	Execute experiments in a visual intuitive manner

Collaboration

Standardise Deployment	Model Lifecycle Management

Benefits of PACE - ML

Speed & Time to Market

- Bring ML models to production faster and at scale
- Faster turn-around-times with one touch deployment

Efficiency

- Reduce development time of the Data Science projects with the toolchain and automation.

Explainability

- Proactive testing of biases and issues with global and local explainability

Effectiveness

- Better feature engineering, model debugging, experiment tracking and metrics comparison to increase solution effectiveness.

Trust

- Increase users' confidence in the system with metrics, explainability, auditability

Automation

- Automated model pipeline management reduces manual interventions, decrease time for deployment, enables continuous delivery

Collaboration

- Common notebooks and experiment tracking leading to better collaboration

Scrutability

- Explainability, monitoring and auditability allow users to achieve better transparency and identify anything wrong.

Debugging

- Enhanced collaboration of the project stakeholders leading to lesser rework and quality issues

Monitoring

- Respond to model performance issues faster
- Avoid model performance degradation with drift detection

Cost of Development

- Reduced cost of development due to automation & seamless integration
- Avoid single cloud tie-up with completely open-source stack.

Governance & Compliance

- Improved governance with audit trails, automated quality checks and monitoring dashboards

THANK YOU

About Mphasis

Mphasis (BSE: 526299; NSE: MPHASIS) applies next-generation technology to help enterprises transform businesses globally. Customer centricity is foundational to Mphasis and is reflected in the Mphasis' Front2Back™ Transformation approach. Front2Back™ uses the exponential power of cloud and cognitive to provide hyper-personalized (C=$X2C^2_{TM}$=1) digital experience to clients and their end customers. Mphasis' Service Transformation approach helps 'shrink the core' through the application of digital technologies across legacy environments within an enterprise, enabling businesses to stay ahead in a changing world. Mphasis' core reference architectures and tools, speed and innovation with domain expertise and specialization are key to building strong relationships with marquee clients. Click here to know

Important Confidentiality Notice