ActiveState®

MLOps: Machine Learning Operationalization

ActiveState Webinar

Panelists

- **Nisha Talagala**, Co-Founder, CTO & VP Engineering, *ParallelM*
- **Boris Tvaroska**, Global Artificial Intelligence Solutions Lead, *Lenovo*

Housekeeping

- Webinar recording and slides will be available shortly
- Share questions with panelists using the Question panel
- Q&A session following presentations

MLOps: Machine Learning Operationalization

Capital One · Bank of America · American National · HSBC · Tesco Bank · Credit Suisse

Bombardier · GE Aviation · Northrop Grumman · NASA · Siemens · Cisco

intel · ca technologies · IBM · Microsoft · Hewlett Packard Enterprise · verizon

Track-record: 97% of Fortune 1000, 20+ years open source

Polyglot: 5 languages - Python, Perl, Tcl, Go, Ruby

Runtime Focus: concept to development to production

ActiveState®

Machine Learning Operationalization

Nisha Talaga, ParallelM

ParallelM

MLOps: Machine Learning Operationalization

Nisha Talagala

Co-Founder, CTO & VP Engineering
ParallelM

nisha.talagala@parallelm.com

ActiveState®

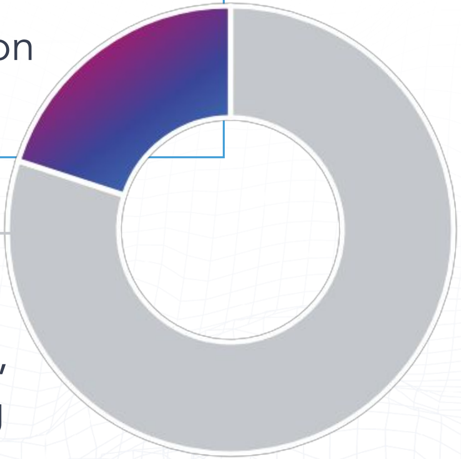
MLOps: The Last Mile

From Data Science to Business ROI

NISHA TALAGALA

CTO, ParallelM

Growing AI Investments; Few Deployed at Scale



20%

AI in Production

80%

Developing, Experimenting, Contemplating

Survey of 3073 AI-aware C-level Executives

Out of 160 reviewed AI use cases:

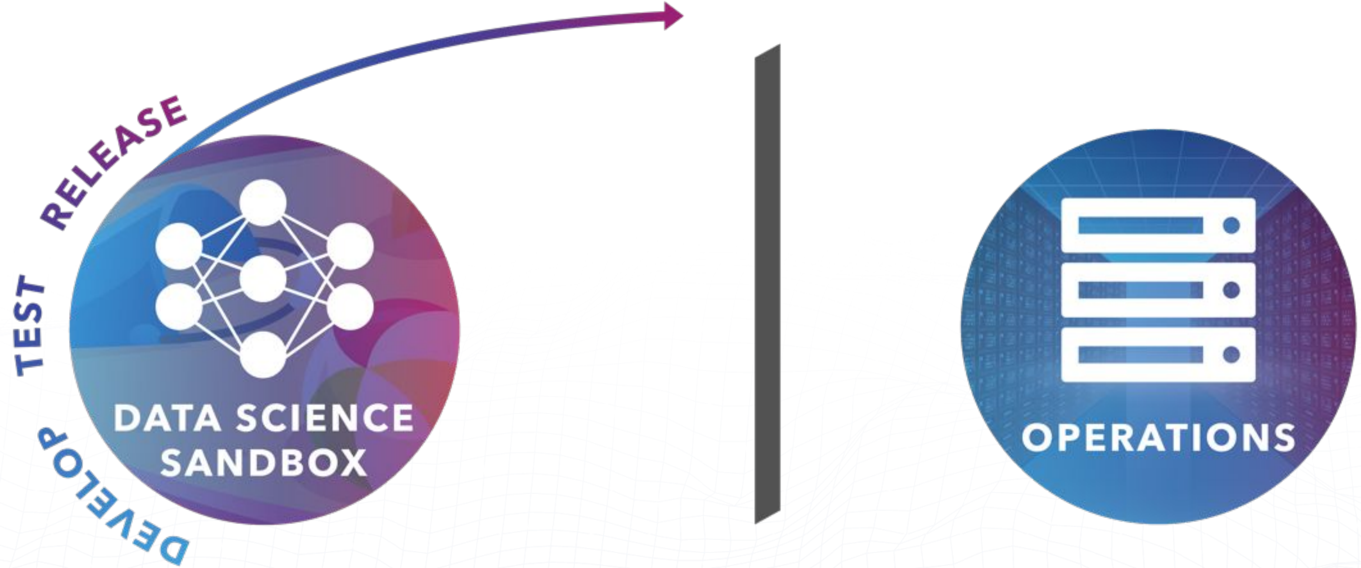
88% did not progress beyond the experimental stage

But successful early AI adopters report:

Profit margins **3–15%** higher than industry average

Source: "Artificial Intelligence: The Next Digital Frontier?", McKinsey Global Institute, June 2017

The ML Development and Deployment Cycle



Bulk of effort today is in the left side of this process (development)

- Many tools, libraries, etc.
- Democratization of Data Science
- Auto-ML

What makes ML uniquely challenging in production?

Part I : Dataset dependency

- ML ‘black box’ into which many inputs (algorithmic, human, dataset etc.) go to provide output.
- Difficult to have reproducible, deterministically ‘correct’ result as input data changes
- ML in production may behave differently than in developer sandbox because live data ≠ training data

What makes ML uniquely challenging in production?

Part II : Simple to Complex Practical Topologies

- Multiple loosely coupled pipelines running possibly in parallel, with dependencies and human interactions
- Feature engineering pipelines must match for Training and Inference (CodeGen Pipelines can help here)
- Control pipelines, Canaries, A/B Tests etc.
- Further complexity if ensembles, federated learning etc are used

What makes ML uniquely challenging in production?

Part III : Heterogeneity and Scale

- Possibly differing engines (Spark, TensorFlow, Caffe, PyTorch, Sci-kit Learn, etc.)
- Different languages (Python, Java, Scala, R ..)
- Inference vs Training engines
  - Training can be frequently batch
  - Inference (Prediction, Model Serving) can be REST endpoint/custom code, streaming engine, micro-batch, etc.
  - Feature manipulation done at training needs to be replicated (or factored in) at inference
- Each engine presents its own scale opportunities/issues

What makes ML uniquely challenging in production?

Part IV : Compliance, Regulations…

- Established: Example: Model Risk Management in Financial Services
  - https://www.federalreserve.gov/supervisionreg/srletters/sr1107a1.pdf
- Emerging: Example GDPR on Reproducing and Explaining ML Decisions
  - https://iapp.org/news/a/is-there-a-right-to-explanation-for-machine-learning-in-the-gdpr/
- Emerging: New York City Algorithm Fairness Monitoring
  - https://techcrunch.com/2017/12/12/new-york-city-moves-to-establish-algorithm-monitoring-task-force/

What makes ML uniquely challenging in production?

Part V : Collaboration, Process

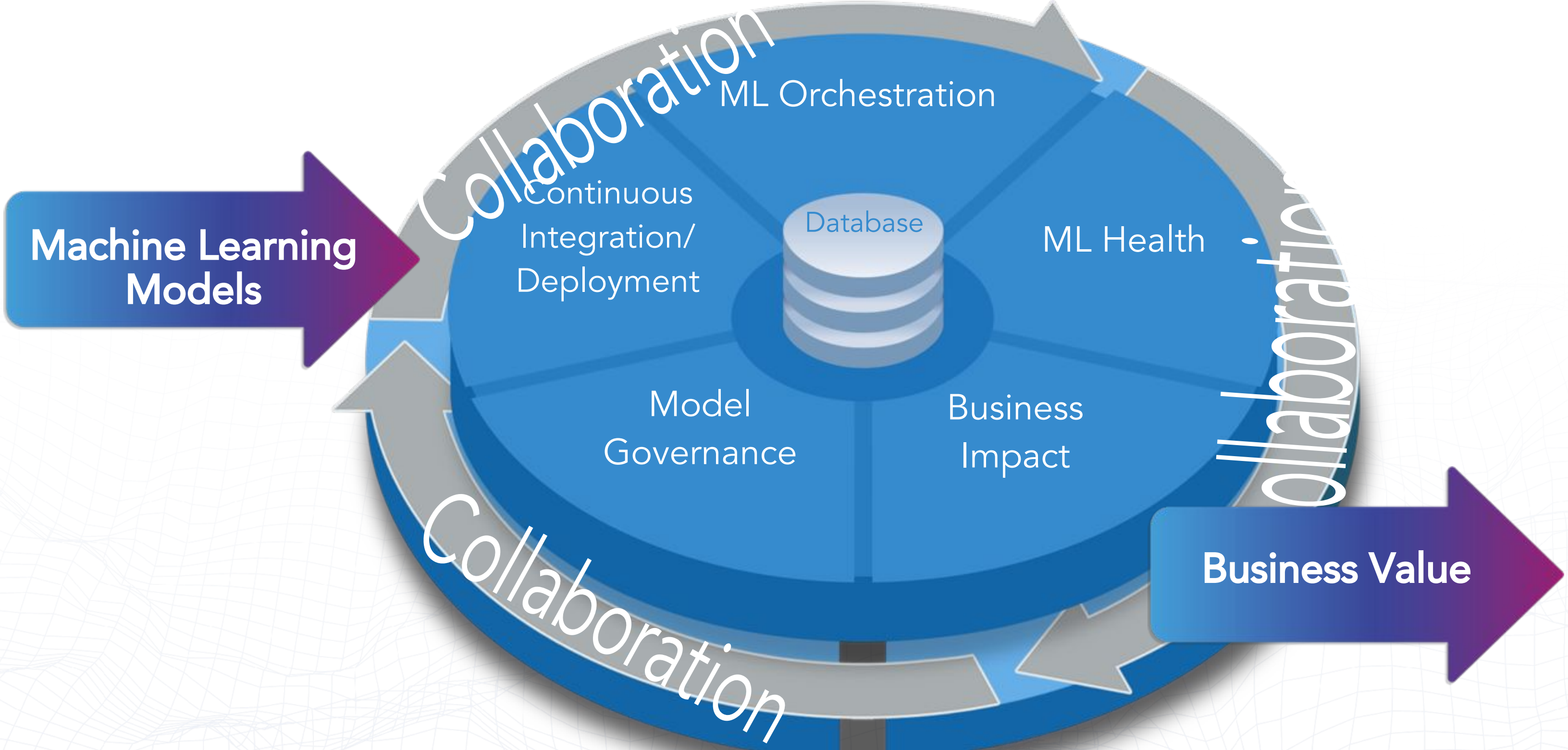
COLLABORATION

- Expertise mismatch between Data Science & Ops complicates handoff and continuous management and optimization

PROCESS

- Many objects to be tracked and managed (algorithms, models, pipelines, versions etc.)
- ML pipelines are code. Some approach them as code, some not
- Some ML objects (like Models and Human approvals) are not best handled in source control repositories

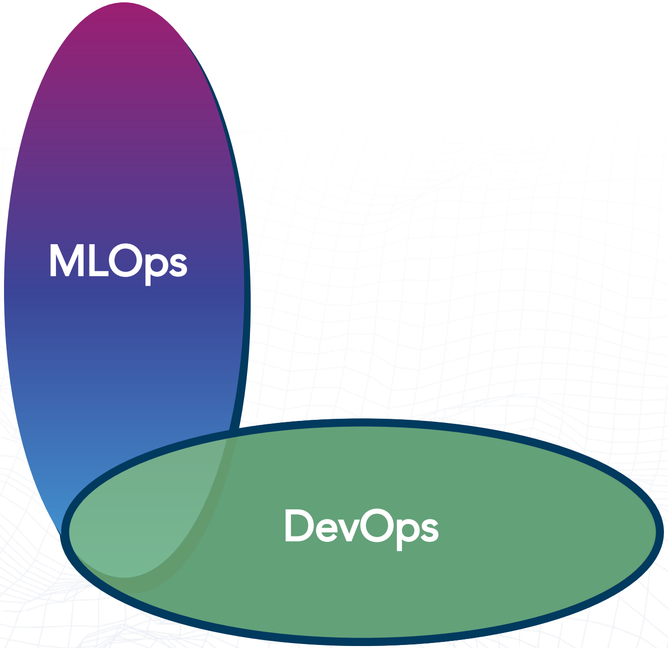
MLOps – Automating the Production ML Lifecycle

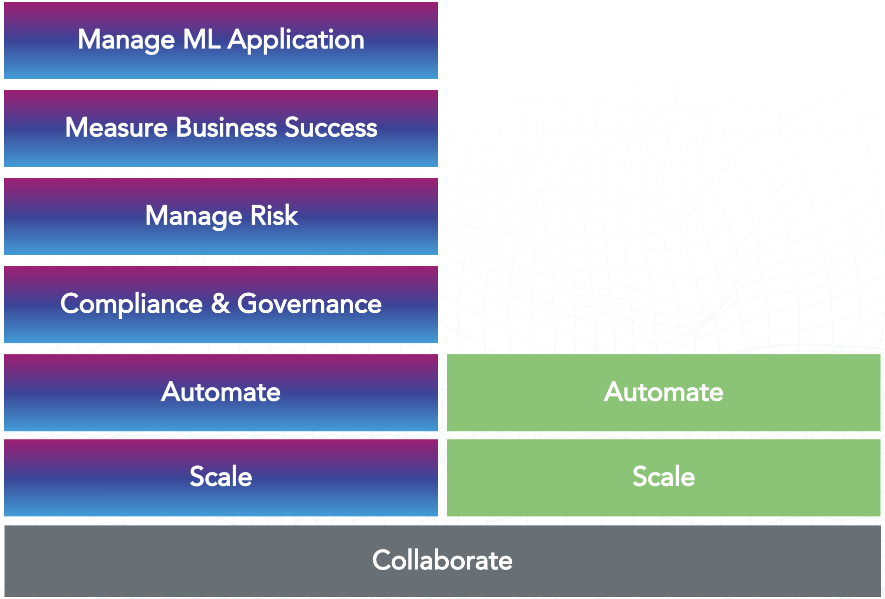


ParallelM

MLOps, DevOps and SDLC

- Integrate with SDLC (Source control repositories, etc.) for code
- Integrate with DevOps for Automation, Scale and Collaboration





Manage ML Application

Measure Business Success

Manage Risk

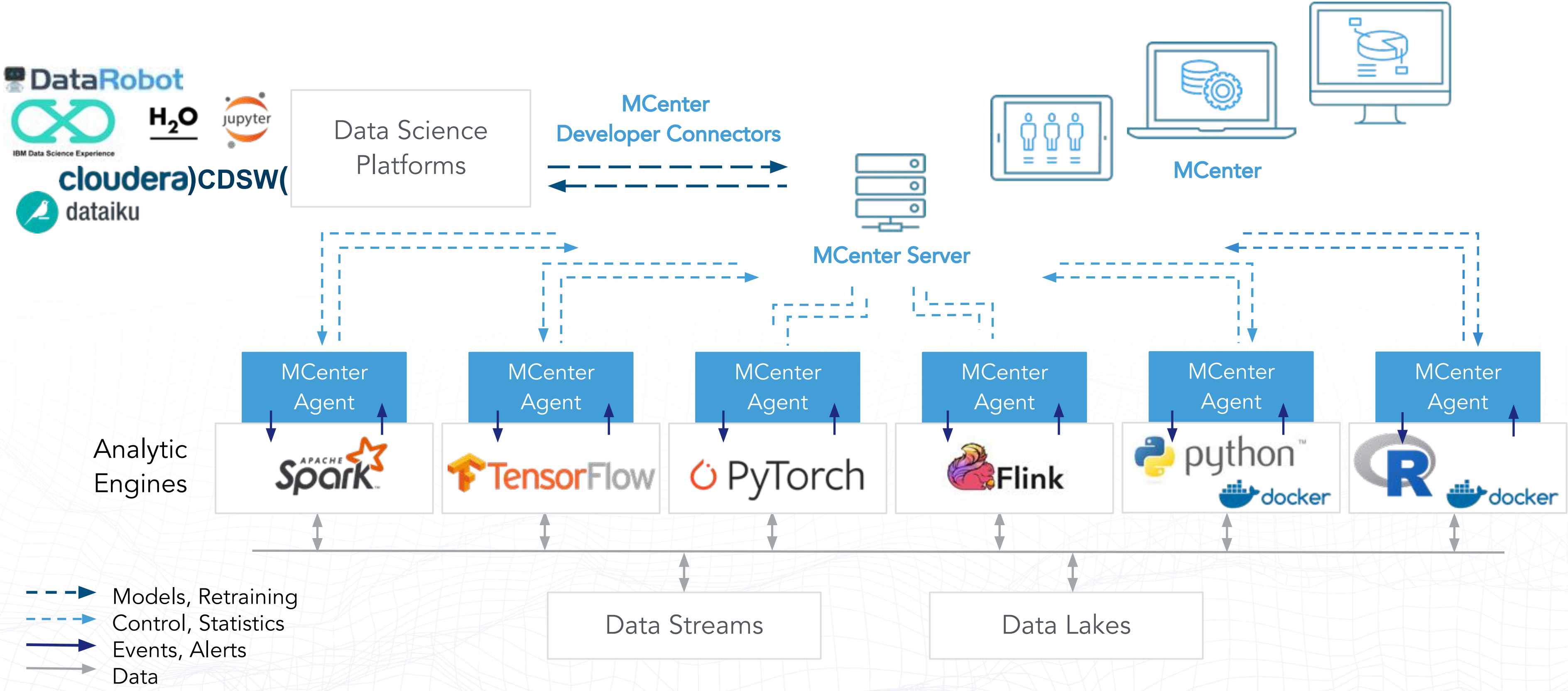
Compliance & Governance

Automate | Automate

Scale | Scale

Collaborate

How it Works – MCenter Architecture



ParallelM

Summary

- We are at the beginnings of ML Operationalization
- Much like databases (backbone of production applications) need DBAs and software needs DevOps, ML needs MLOps (specialized operationalization practices, tools and training)
- For more information
  - https://www.mlops.org for MLOps resources
  - https://www.parallelm.com

Thank You

nisha.talagala@parallelm.com

ActiveState®

Machine Learning Operationalization

Boris Tvaroska, Lenovo

Lenovo

MLOps: Machine Learning Operationalization

Boris Tvaroska
Global Artificial Intelligence Solutions,
Lenovo

btvaroska@lenovo.com

ActiveState

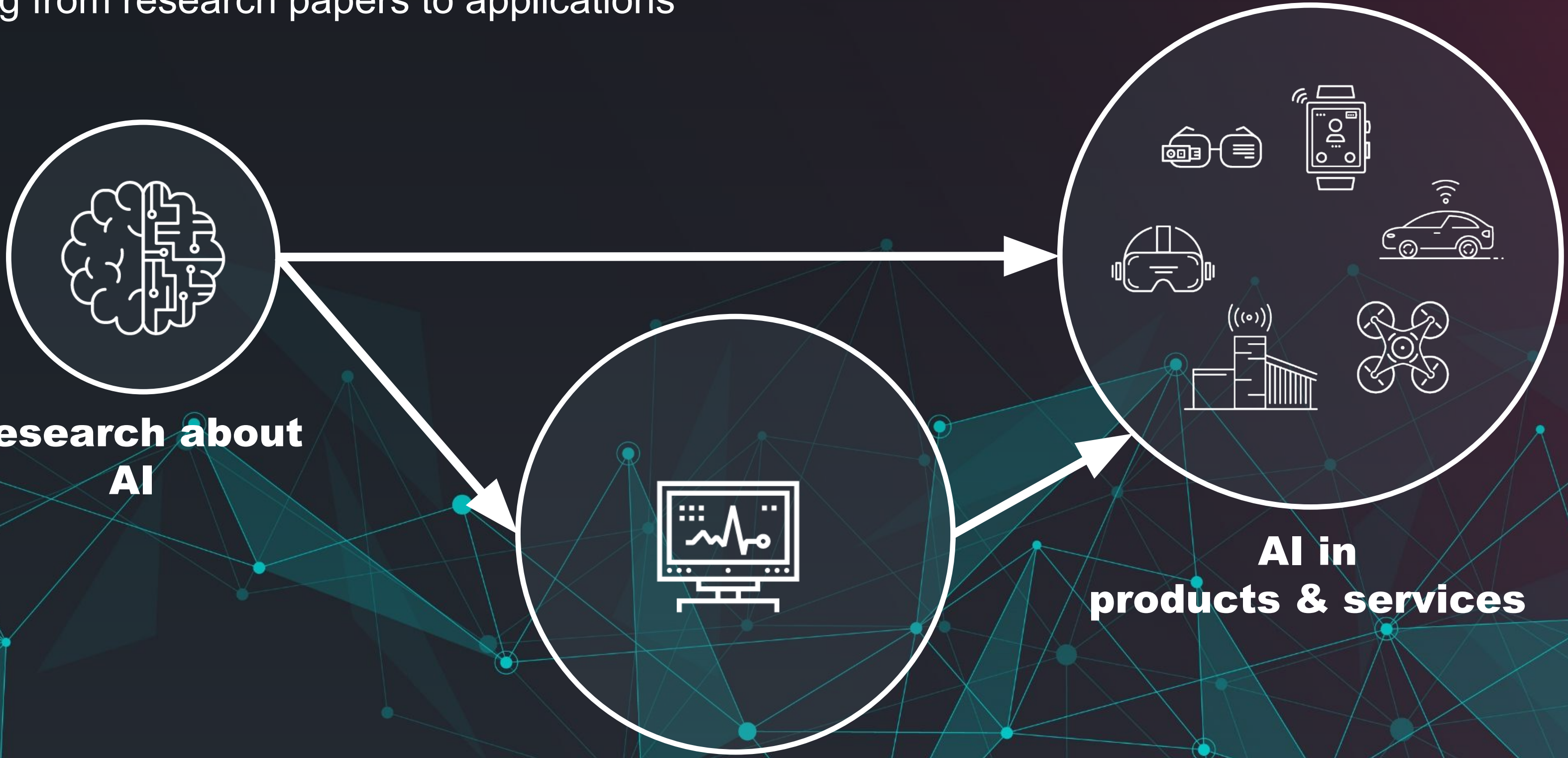
Integrating data science into SDLC

Boris Tvaroska

September 2018

Evolution of AI

Moving from research papers to applications



Research about AI

Reports using ML/DL

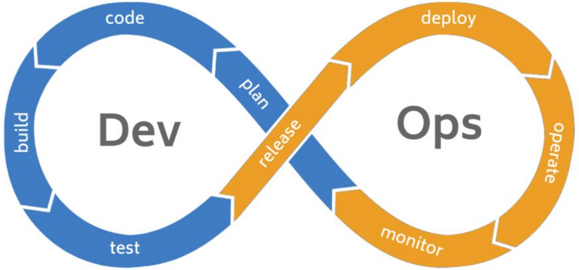
AI in products & services

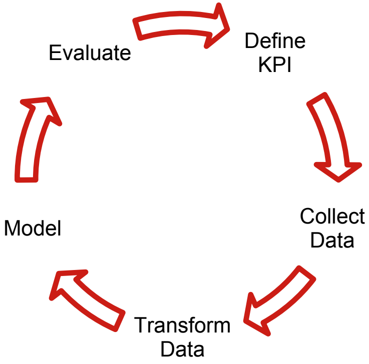
What can happen?

I did not change a single line of code.

Junior Software Engineer after breaking the build

Different lifecycles





- Starts with change in code
- Established practice
- Iterations in days / weeks

- Starts which change in code, data or metrics
- Emerging practice
- Iterations as fast as possible, several times per day

Main challenges

Test

- The wrong result is acceptable
- Need to test for False Positives
- Need to test for False Negatives

- Longer test times
- More test cases needed

Build & Deploy

- More artifacts to work with
- Frequent changes
- Versioning of artifacts and source data

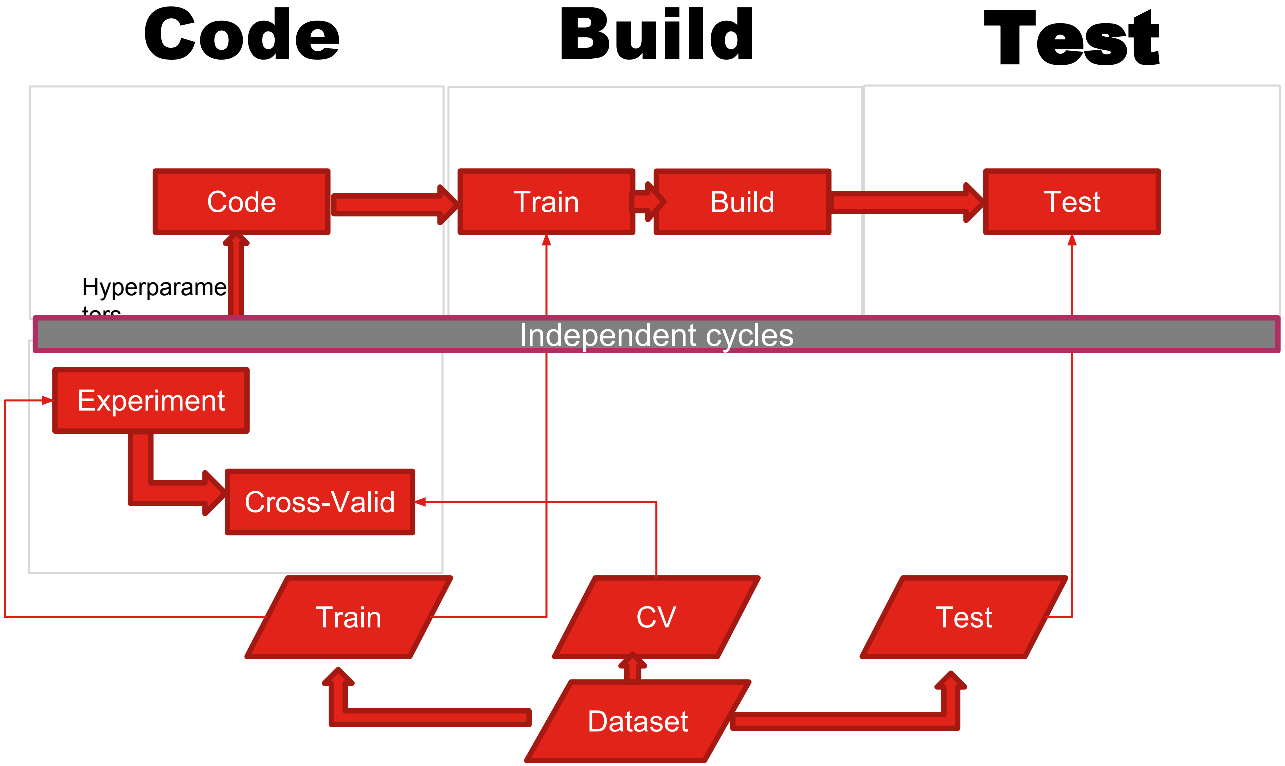
Training in test/build cycle

Possible for simple models with small amount of data

Existing toolset

Risks:

- Slow CI/CD cycle
- More failing builds



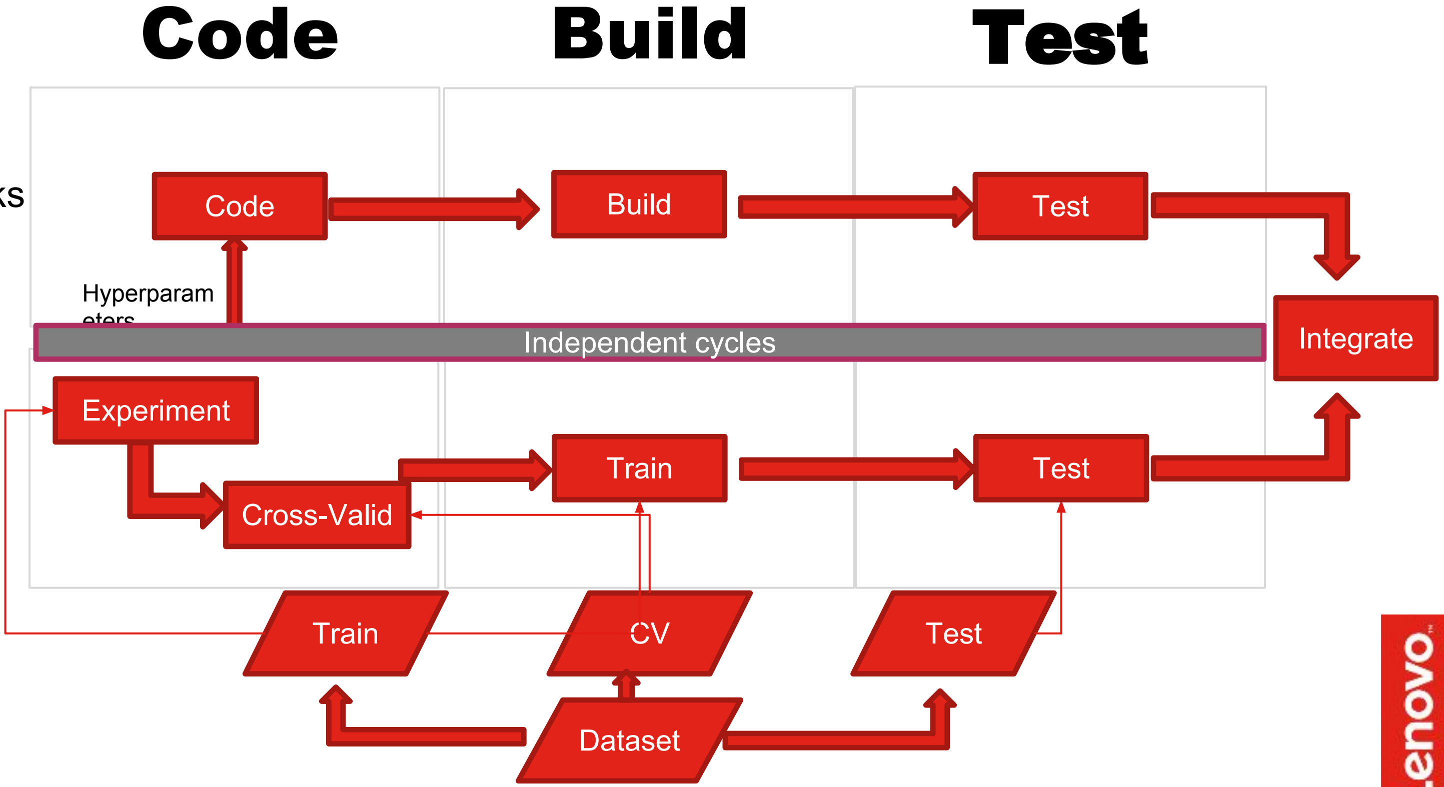
Model as a service

Model is independent

Fit languages/frameworks

Risks:

- Interface is vector
- Pre-mature service boundaries
- Multi-step application

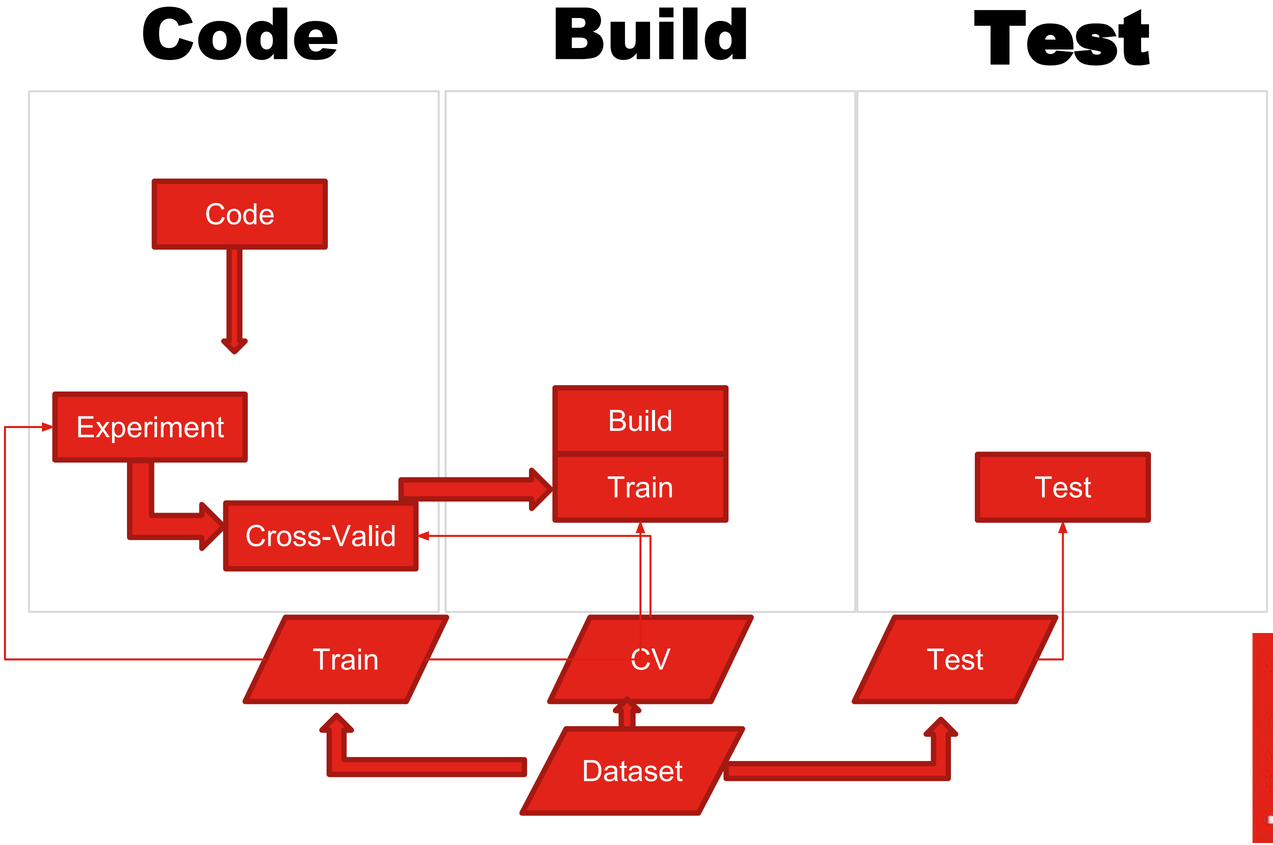


SW emerged in Data Science

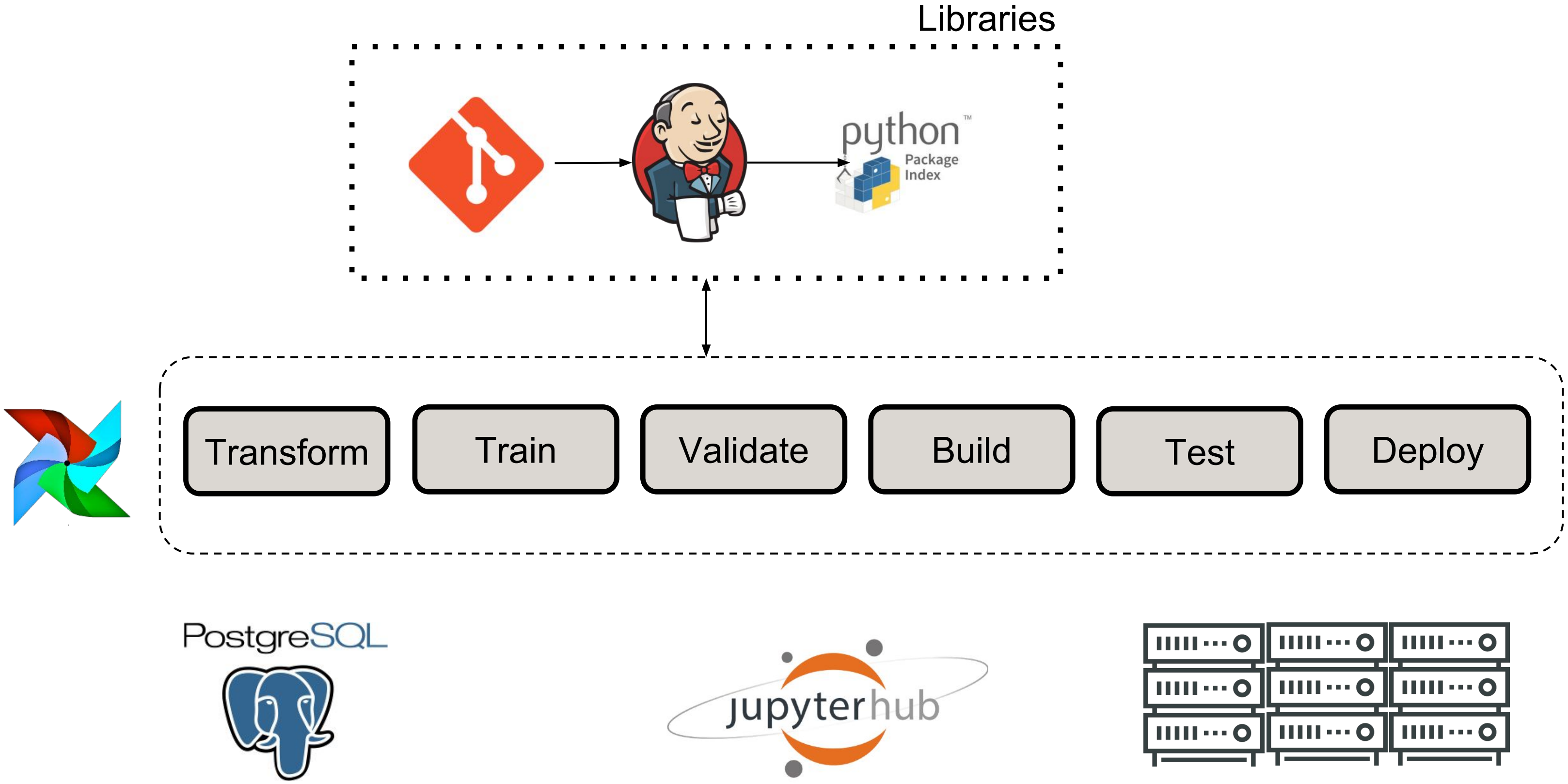
Clearly defined service

Data Science toolset
Data Science framework

Risks:
- Culture clash



Practical example



Boris Tvaroska

Global Solution Lead for Lenovo

AI Innovation Centers

20 years of experience running engineering teams across Europe, North and South America, Middle East, India

Email: btvaroska@lenovo.com
boris@tvaroska.sk

Linkedin: www.linkedin.com/in/boristvaroska

Twitter: @btvaroska

Q & A

ActiveState®

Thank you to our panelists

- **Nisha Talagala**, Co-Founder, CTO & VP Engineering, *ParallelM*
- **Boris Tvaroska**, Global Artificial Intelligence Solutions Lead, *Lenovo*

What’s Next

- Learn more about our Platform: https://www.activestate.com/platform

- Watch a demo: https://www.youtube.com/watch?v=c5AIxN9ehrI

- Contact platform@activestate.com for more information.

ActiveState®

Where to find us

Tel: **1.866.631.4581**

Website: **www.activestate.com**

Twitter: **@activestate**

Facebook: **/activestatesoftware**

ActiveState®