Machine Learning Airport Surface Model

Jeremy Coupe: william.j.coupe@nasa.gov
Lead for Sustainable Flight National Partnership CDDR Demo
Digital Avionics and Systems Conference (DASC)

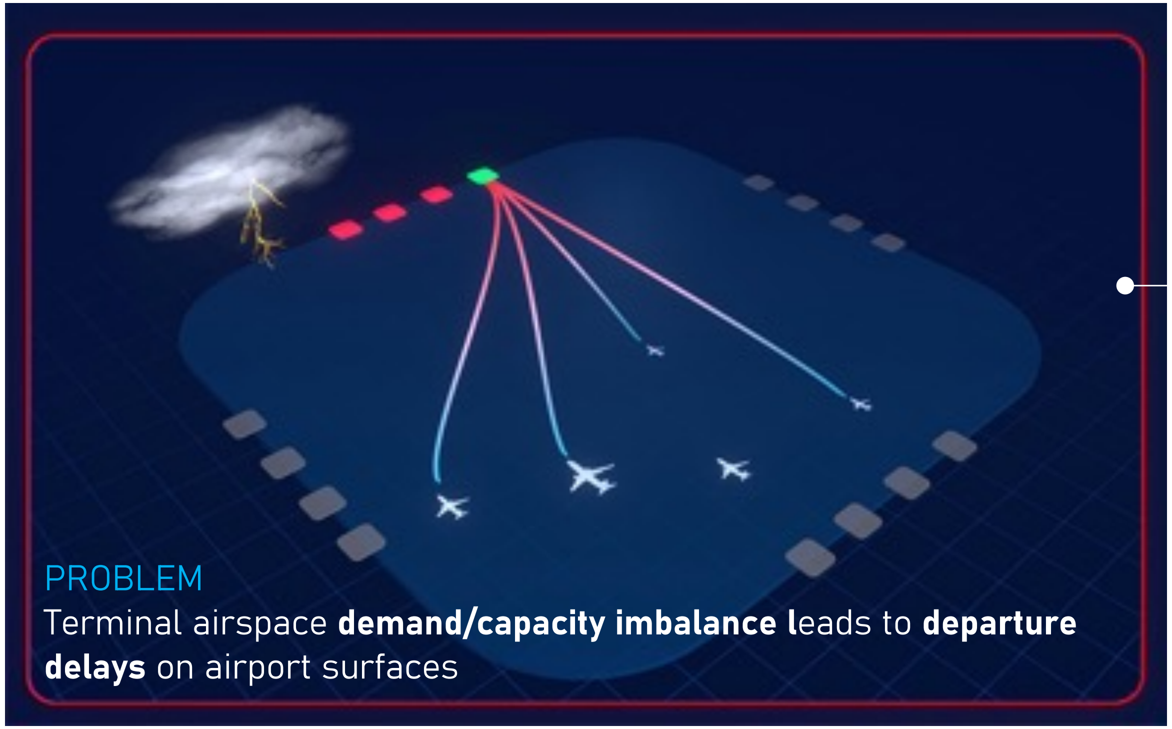
Outline

- Background on Machine Learning (ML) airport surface model
- Machine Learning Operations (MLOps)
- Field evaluation of ML airport surface model
- Conclusion

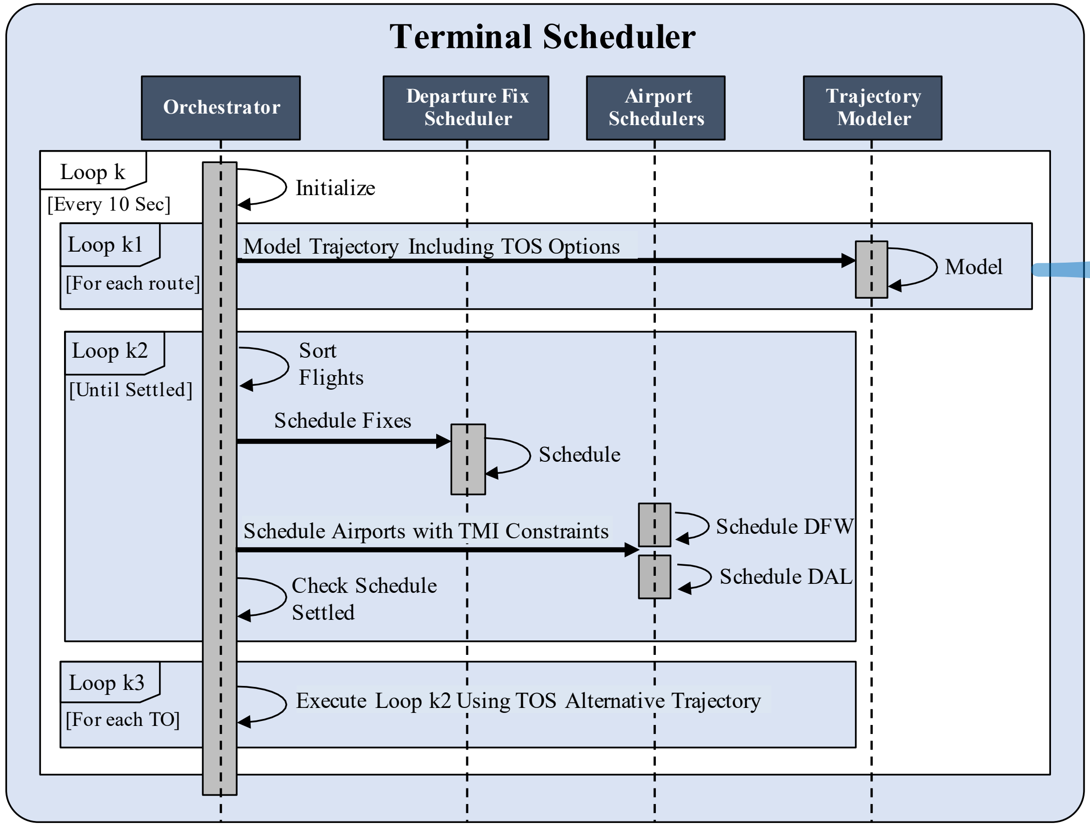
Outline

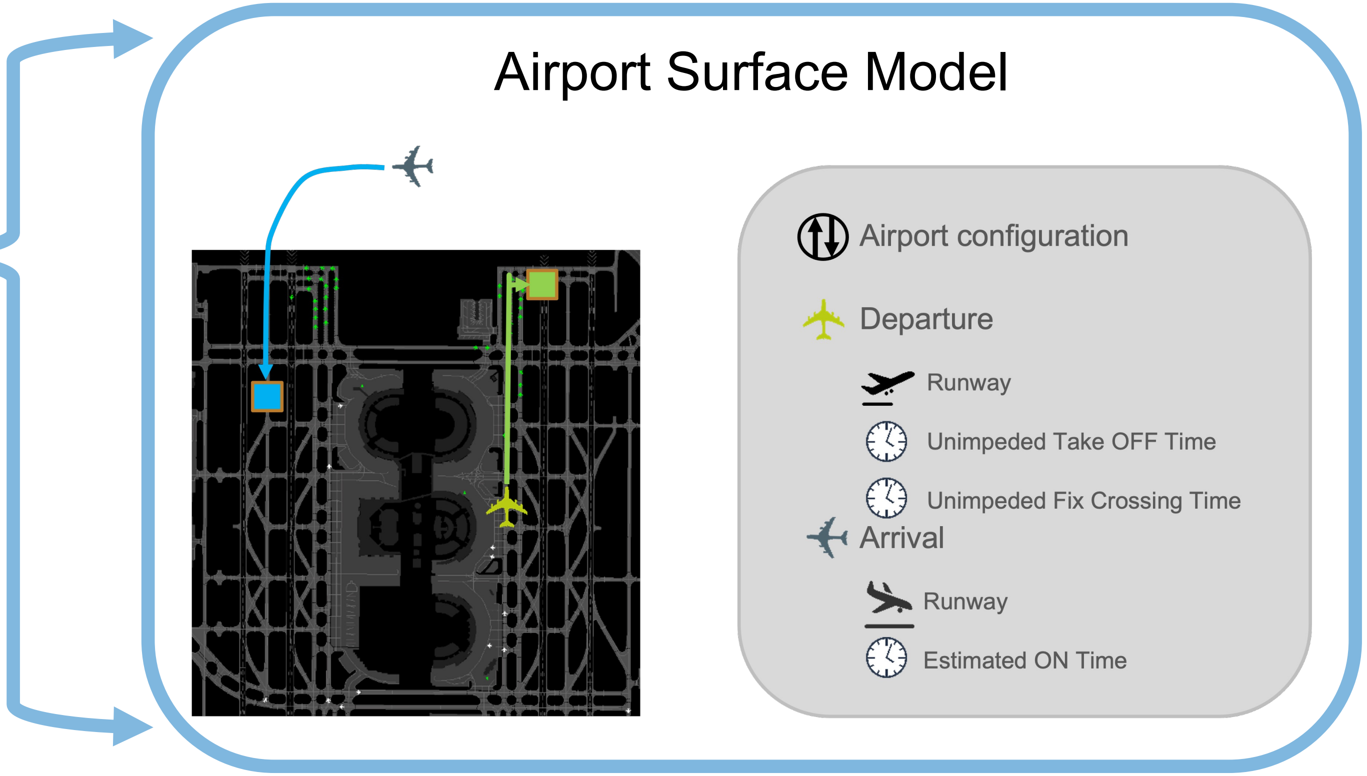
- Background on Machine Learning (ML) airport surface model
- Machine Learning Operations (MLOps)
- Field evaluation of ML airport surface model
- Conclusion

Pre-departure Trajectory Option Set Reroutes



Predictive Engine = Surface Model + Scheduler

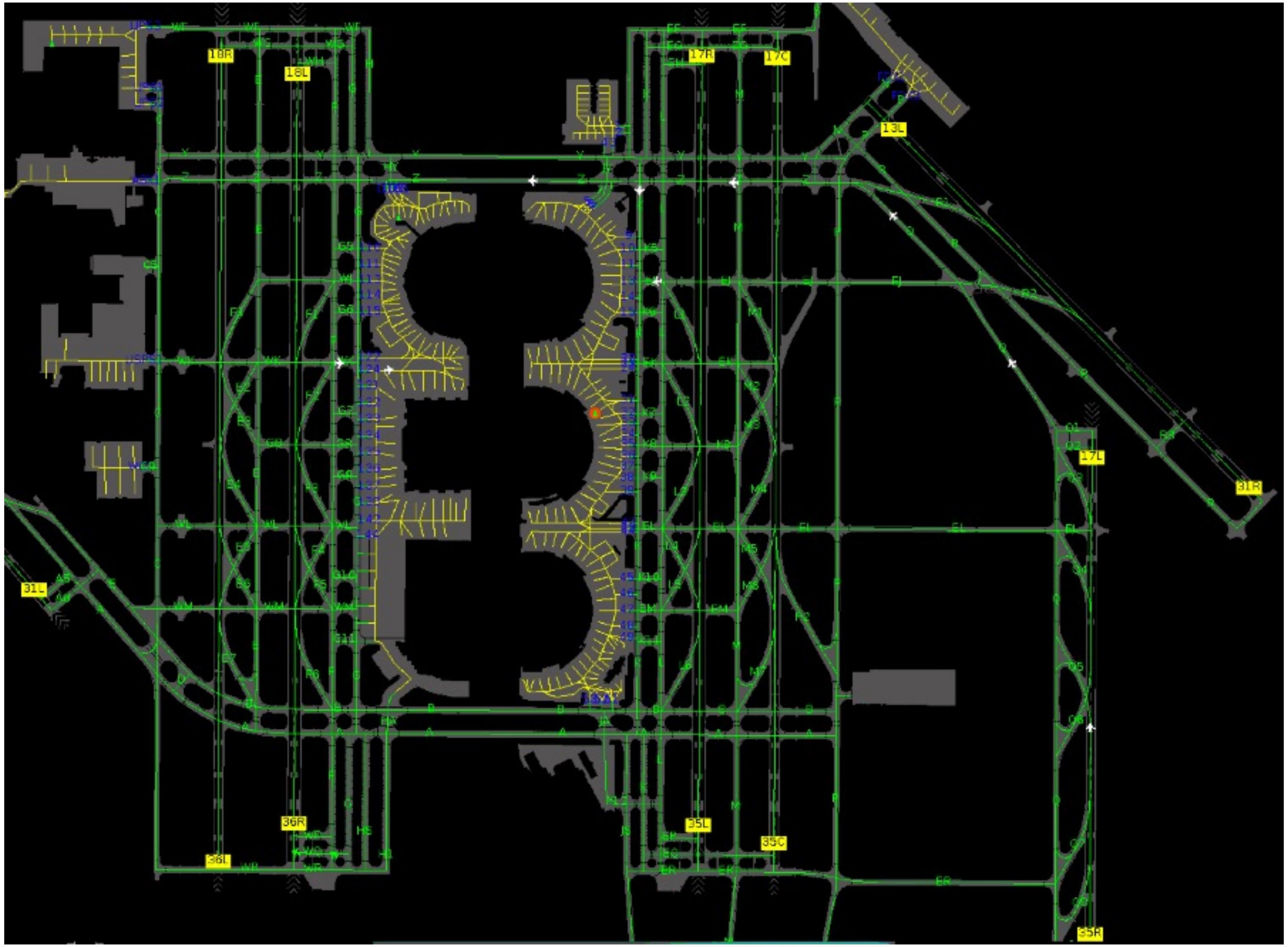




- **Airport Surface Model:** generate unimpeded trajectory used as input by the scheduler
- **Scheduler:** generate OFF prediction while enforcing constraints at terminal boundary and each airport surface

Modeling unimpeded trajectory requires significant adaptation and *was* bottleneck to scalability

KDFW Adaptation Example



- Detailed link node network defines the airport surface structure including gate locations, runway locations, and taxi routes
- Adaptation goes beyond physical structure to include SME knowledge encoded in decision trees (for example the fix to runway mappings)
- Requires significant time and effort to build and maintain for each airport

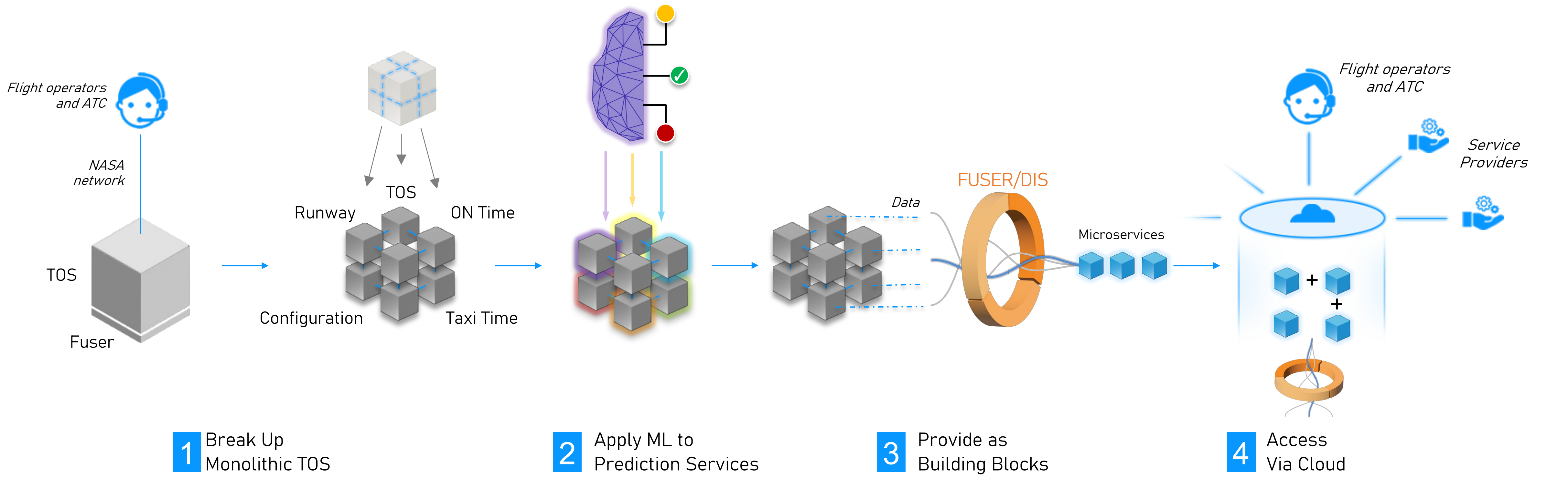
Machine Learning Airport Surface Model: Digital Transformation of Adaptation/Physics Based System to Machine Learning

Previous: ATD-2 monolithic physics/adaptation based

Monolithic service for single application, using adaptation-based algorithms to generate trajectory predictions as input to terminal scheduler; requiring site-to-site deployment

Current: DIP service-oriented leveraging machine learning

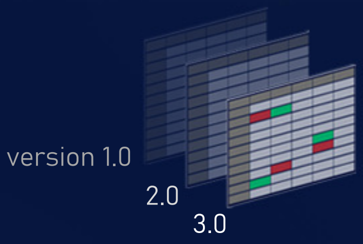
Transformed into service-oriented architecture of highly reusable digital services accessible on the platform to support many advanced applications; upgraded to machine learning-based algorithms for predictions to enable NAS-wide scalability



Digital transformation aligned with FAA vision for Info-Centric NAS:
Learning, adaptable, and lightweight interacting systems

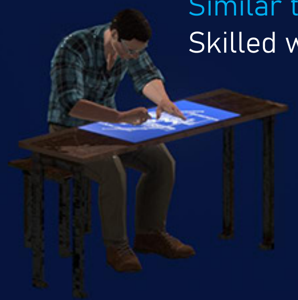
Previous:

ATD-2 physics/adaptation based airport surface model



Requires **continuous** maintenance

Similar to:
Skilled worker



Mental models encoded
into **complex decision trees**

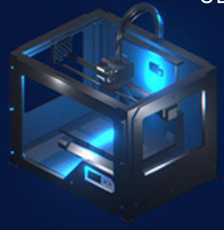
Good at **specific** tasks
Requires detail knowledge

Current:

Scalable DIP Machine Learning airport surface model

Semi-automated maintenance

Similar to:
3D printer



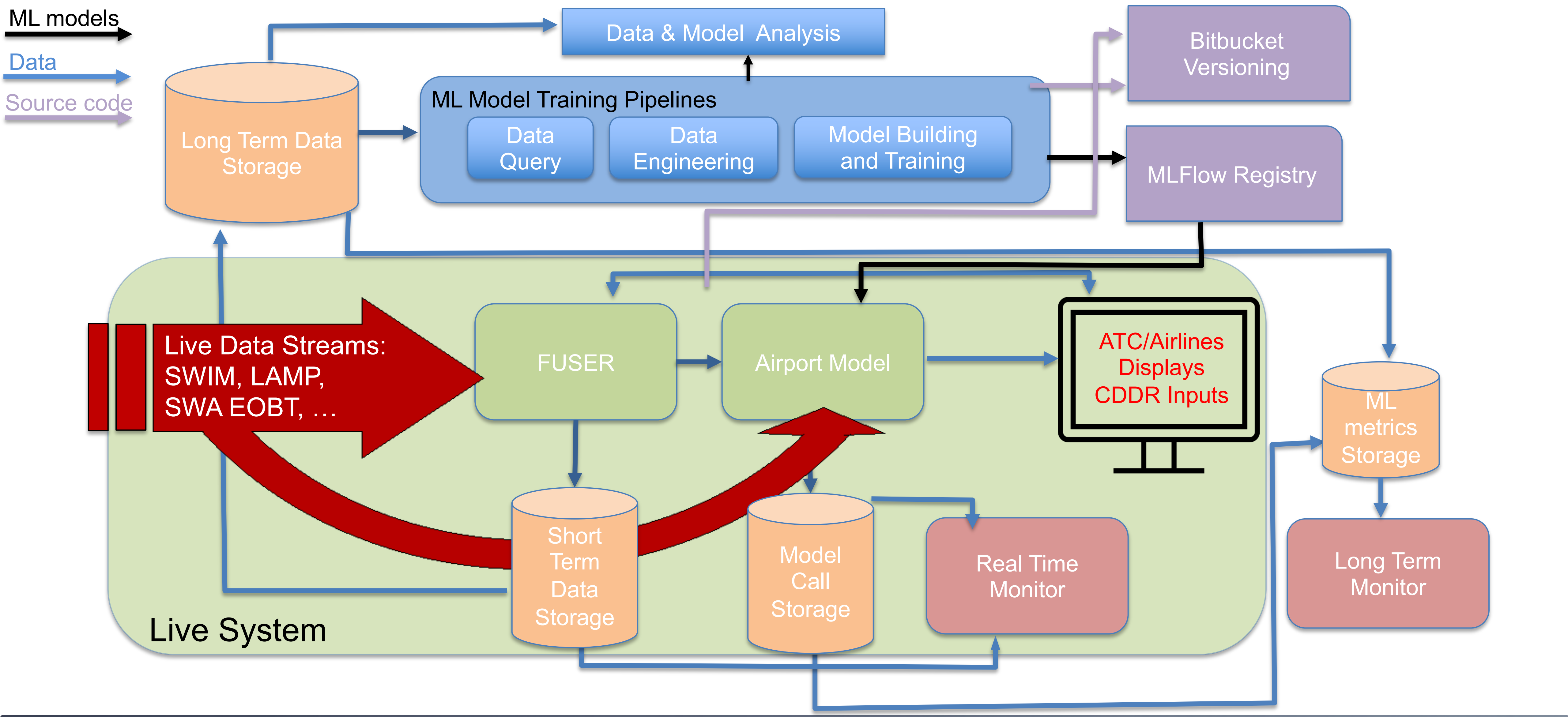
Absorbs information from
multiple existing data sources

Flexible to new tasks
Learns detailed knowledge

Outline

- Background on Machine Learning (ML) airport surface model
- Machine Learning Operations (MLOps)
- Field evaluation of ML airport surface model
- Conclusion

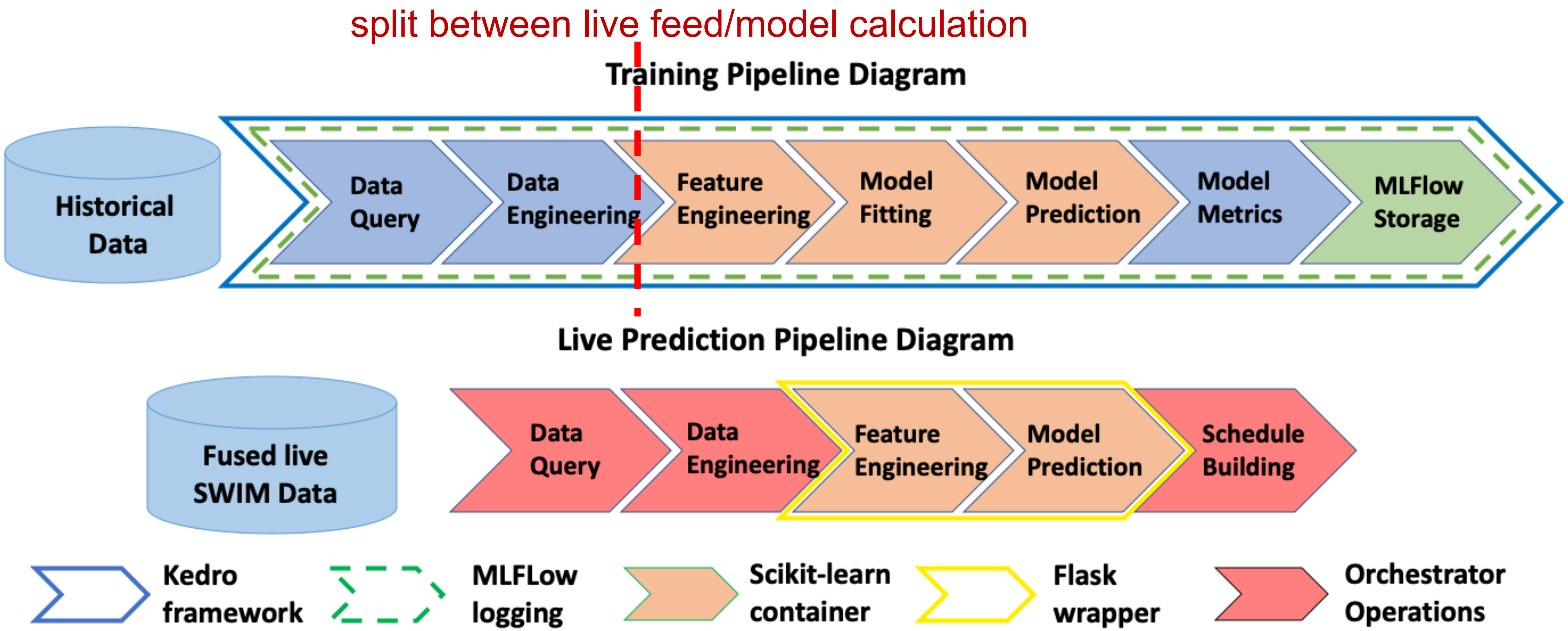
MLOps Architecture Supporting Real-time Deployment



ML pipelines

We developed our training pipelines in Python around mainly three Python libraries :

- Kedro : pipeline structures defined by DAGs, help to abstract inputs and to define a common design for the team
- Scikit-learn : used for the Pipeline class that allows to store some feature engineering into the model
- MLflow : keeps track of the models, their artifacts, performances, tags the latest models



MLflow log example : parameters

Each run in MLflow contains :

- Run date
- Git Hash : allows to identify the source code used for the run
- Parameters of the training : along with the Git hash allows to reproduce the results of any run, and compare run performance

mlflow Experiments Models GitHub Do

dep_rwy > new_wrapper

Date : 2021-08-02 11:24:04
Source : transition_models.py
Git Commit : 03a0c46dea15b03ed93fd828ee7e9a4f3c3df78d

User : iads
Duration : 2.5s
Status : FINISHED

Notes

None

Parameters

Name	Value
baseline	False
core_features	['lookahead', 'departure_fix_source_data', 'airport_configuration_name']
default_response	17R
end_time	2021-06-30
features	['lookahead', 'departure_fix_source_data', 'filed_flight', 'airport_configuration_name', 'aircraft_engine_class', 'wake_turbulence_category']
known_runways	['17L', '17C', '17R', '18L', '18R', '36L', '36R', '35L', '35C', '35R', '13L', '13R', '31L', '31R']
model	XGBClassifier
model_params	{'random_state': 42, 'objective': 'multi:softmax'}
start_time	2021-01-01

MLflow log example : metrics & artifacts

For each run, MLflow stores a set of performance metrics and some artifacts. Artifacts are the pickle file containing the model, the model environment definition, some visualizations of the performance/data property (ie feature value distribution)

Model pickle and environment definition

▾ Metrics

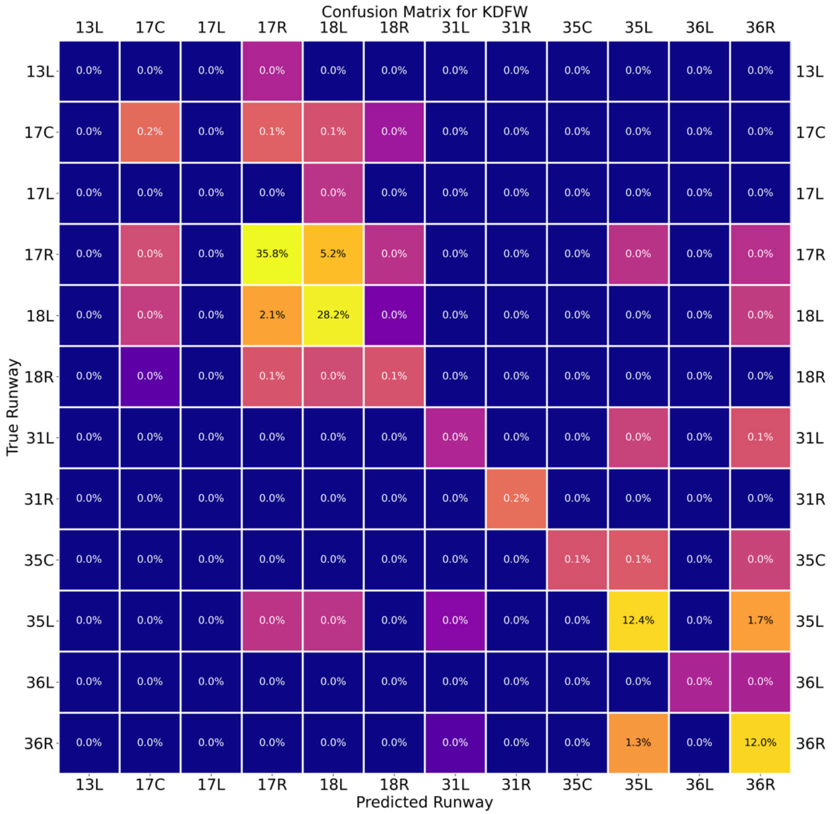
Name	Value
accuracy_test	0.89
accuracy_train	0.89
auc_test	0.965
auc_train	0.982
drop_fraction_test	0.029
drop_fraction_train	0.029
fraction_in_config_actual_test	0.996
fraction_in_config_actual_train	0.996
fraction_in_config_pred_test	1
fraction_in_config_pred_train	1
misclass_to_parallel_runway_frac_test	0.109
misclass_to_parallel_runway_frac_train	0.108
num_testing_samples	2780617
num_training_samples	2367730
precision_test	0.892
precision_train	0.893
recall_test	0.89
recall_train	0.89

▾ Artifacts

- ▾ model
  - MLmodel
  - conda.yaml
  - model.pkl
  - requirements.txt
- cm_kdfw_test.csv
- cm_kdfw_test.png
- cm_kdfw_train.csv
- cm_kdfw_train.png
- config_rwy_pairs_kdfw_act_test.html
- config_rwy_pairs_kdfw_act_train.html
- config_rwy_pairs_kdfw_pred_test.html
- config_rwy_pairs_kdfw_pred_train.html
- doa_kdfw_test.png
- doa_kdfw_training.png
- frac_in_config_kdfw_act_test.html
- frac_in_config_kdfw_act_train.html
- frac_in_config_kdfw_pred_test.html
- frac_in_config_kdfw_pred_train.html
- ts_kdfw_test_all.png
- ts_kdfw_test_ml.png
- ts_kdfw_training_all.png
- ts_kdfw_training_ml.png

Full Path: /casa/mlruns/13/11f5da4ce4054dc98984c5305852ca01/artifacts/cm_kd

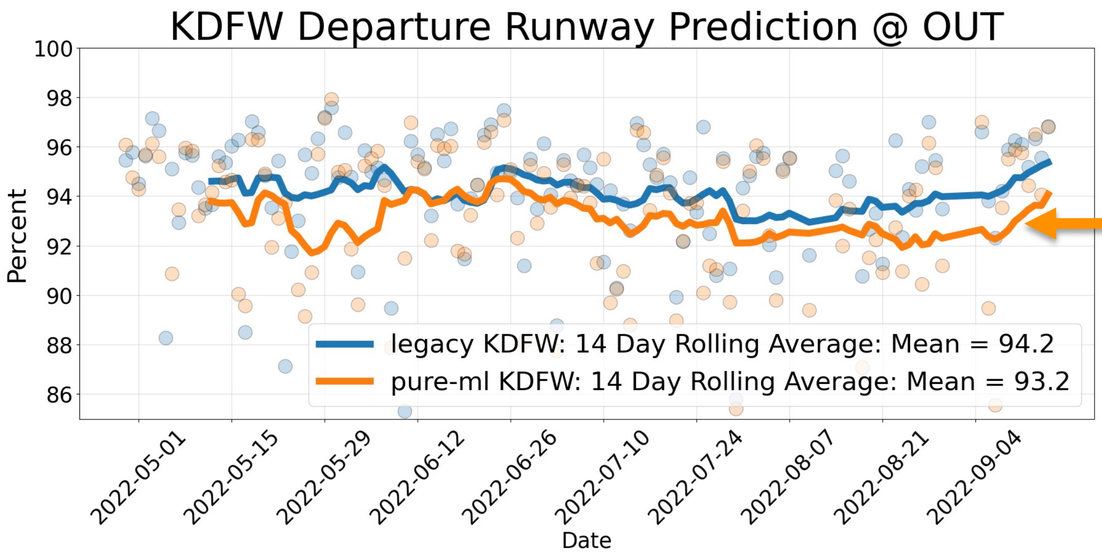
Size: 599.34KB



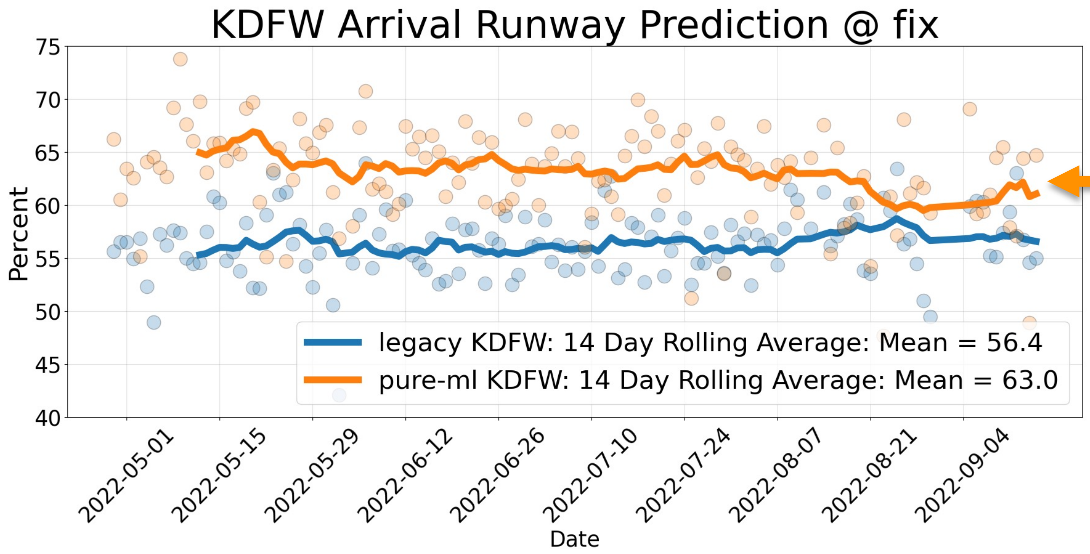
Outline

- Background on Machine Learning (ML) airport surface model
- Machine Learning Operations (MLOps)
- Field evaluation of ML airport surface model
- Conclusion

KDFW Runway Prediction Accuracy

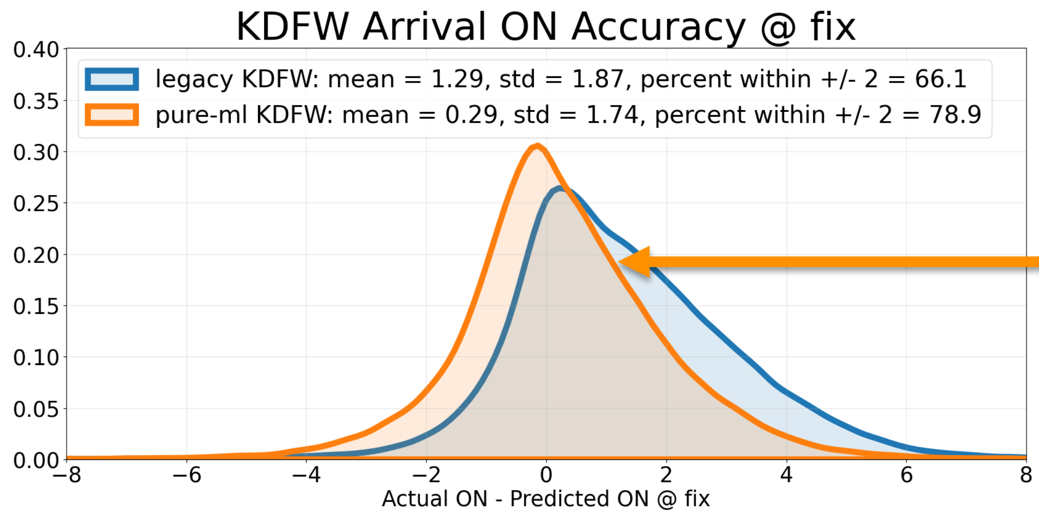


Departure Runway: ML prediction accuracy within 1% of ATC *assigned* runways

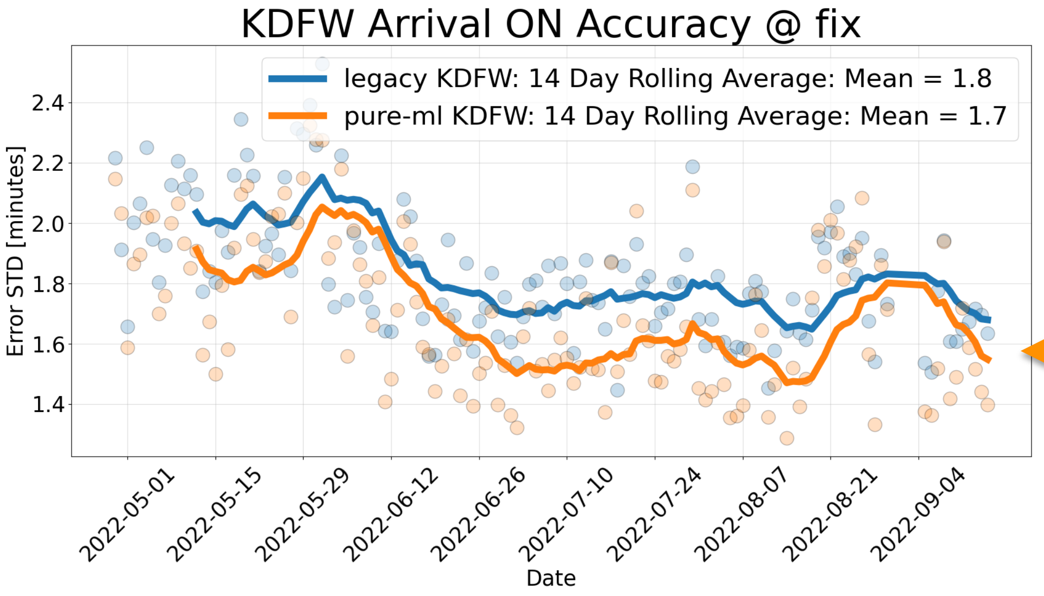


Arrival Runway: ML prediction outperformed legacy FAA Time Based Flow Management (TBFM)

KDFW Arrival ON Accuracy Sampled @ Fix

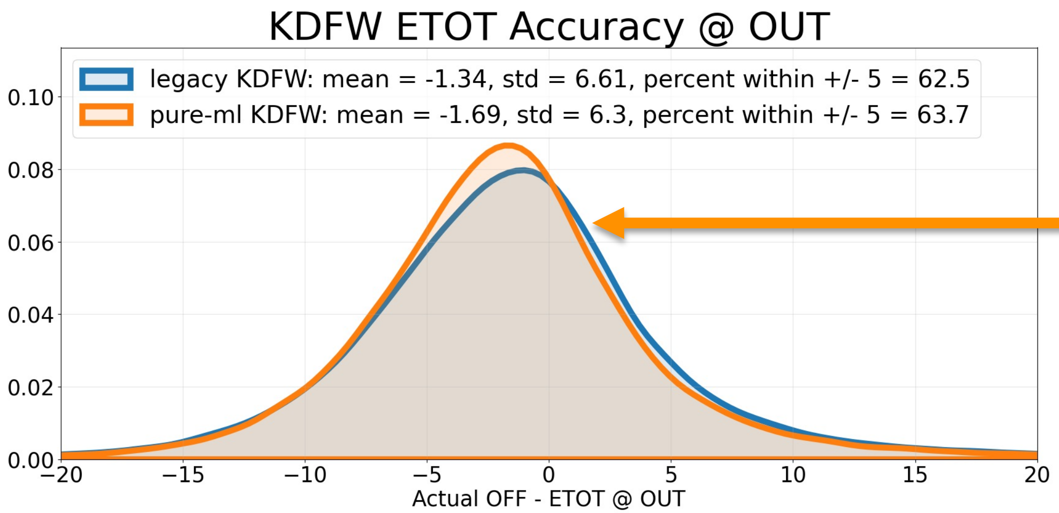


Arrival ON: ML prediction reduced the bias and as a result had higher percentage of flights landing within +/-2 minutes of prediction

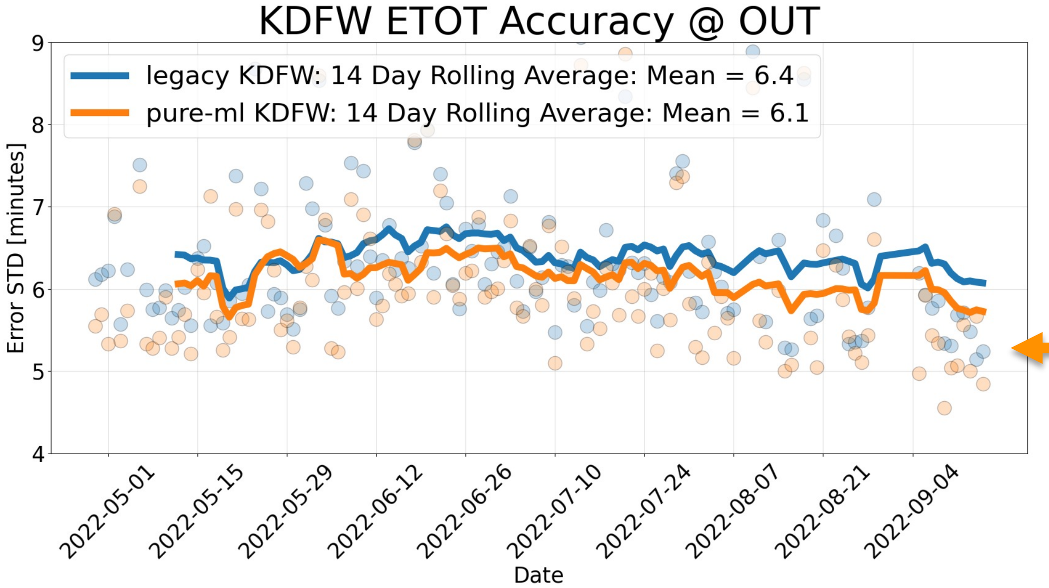


Arrival ON: ML prediction outperformed throughout the lifecycle of the field evaluation in variety of operating conditions

KDFW Estimated Take Off Time (ETOT) Accuracy Sampled @ OUT



Departure Estimated Take Off Time: ML prediction slightly better than legacy adaptation-based methods



Departure Estimated Take Off Time: Performance consistent across lifecycle of field evaluation in variety of operating conditions

Outline

- Background on Machine Learning (ML) airport surface model
- Machine Learning Operations (MLOps)
- Field evaluation of ML airport surface model
- Conclusion

Conclusion

- ML Airport Surface Model a result of digital transformation from legacy system to scalable ML system in alignment with FAA Info-Centric NAS

- Validation results from operational field evaluation in North Texas while running parallel systems (ML vs legacy approach)

- Arrival ML predictions showed improvement over legacy system for both arrival runway and arrival ON time

- Departure Estimated Take Off Time (ETOT) ML predictions slight improvement over legacy system and achieved with more scalable architecture