ML Data Pipelines for Real-Time Fraud Prevention

Mikhail Kourjanski, Principal Architect

QCON New York, June 2018

Service with Velocity and Scale

~237MM
active users
Q1'18

~$1.25 Billion
TPV/day

~250MM
Fast Decisions/Day

~25 Billion
Computations/day

~60 Billion
Queries/day

~150 PB Data

PayPal venmo xoom Braintree Paydiant

Facts and numbers:

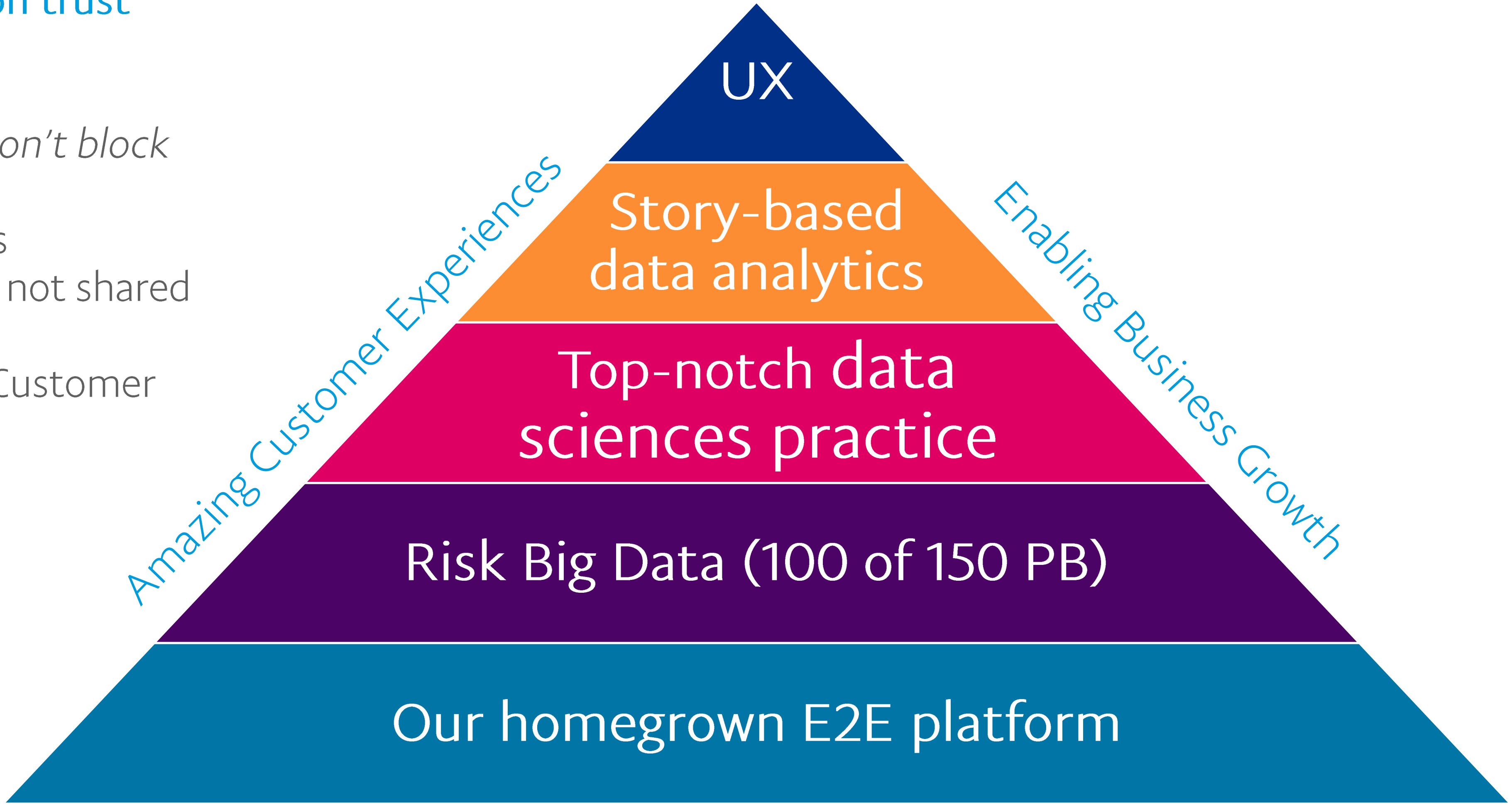
- PayPal - in more than 200 countries and regions.
- Secure Payments: $451 Billion global transaction volume in 2017
- Significant incoming fraud pressure
- Sophistication of the modern day hacker attacks: distributed; high-velocity
- Compliance and Privacy: AML, Prevention of prohibited activities, KYC, PII protection

Risk Decisioning is a Competitive Advantage for PayPal

Key Differentiating Capabilities

➢ User Experience is based on trust

- Block fraud…
- …with low False Positives *(don't block good folks!)*
- Buyer and Seller Protections
- Full customer financial data not shared with merchants
- Regulatory Compliance => Customer Safety



Use Cases:
ML for Fraud Prevention

Fraud has many different forms

Illustrative scenarios (*including, but not limited to…*)

Stolen accounts

- Existing account with linked financial instruments and balance is taken over
  - Change of shipping address and contact info (email, phone)
  - Attempts to purchase expensive goods; or transfer money out (e.g. P2P send money)

Identity fraud

- Account opening under stolen identity
  - Credit risks
  - Usage in a chain of account to account transfers in attempt to exit stolen money
  - High velocity in attempting to open multiple accounts linked in some way (e.g. from same IP)
  - Or, "grooming" of the account to build positive history – later to be used in a burst of bad activity

Collusion

- Fraudulent Merchant account along with multiple fraudulent consumer accounts
  - Using stolen credit cards to "pay for goods" – actually funneling out stolen money via Merchant

Carrying Risk of Transactions: Decisions at Checkpoints

Each payment transaction is a customer's story

Explore	Enroll – New Acct	Manage via Self-Service	Transact	Resolve

Reporting, and Analytics - Research

Explore	Enroll – New Acct	Manage via Self-Service	Transact	Resolve
➢ Behaviour ➢ Offers ➢ Guest checkout	➢ Do we know you? ➢ Validations ➢ Credit Risk	➢ Login / Auth ➢ Wallet ➢ Profile	➢ Purchase <Buyer, Seller> ➢ Transfer ➢ Send Money ➢ Withdrawal	➢ Complaints ➢ Chargebacks ➢ Recover NSF ➢ Investigations

Velocity

Linked Objects & Activities

What Data Do We Process?

Types of data affect choice of modeling methods and frameworks

Explore → Enroll – New Acct → Manage via Self-Service → Transact → Resolve

Structured data…

- *Numbers*
- *Strings*
- *Dates*
- *Geo, …*

Features

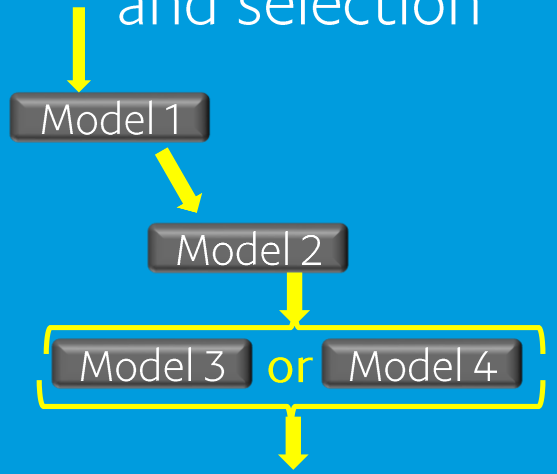
… + Unstructured data

- *Text – emails, customer interaction records*
- *ChatBot*
- *Voice - IVR*
- *Images*
- *Social media*

ML Models – Inferencing in Production Ecosystem

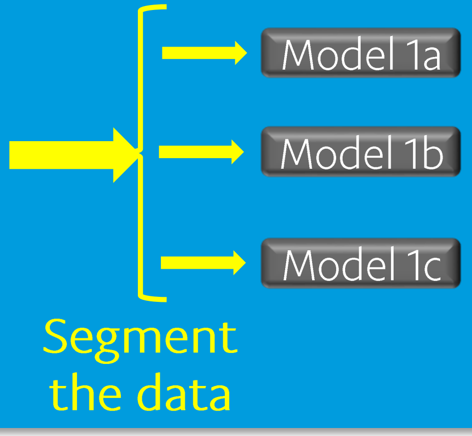
Model Composition

- Model sequencing and selection



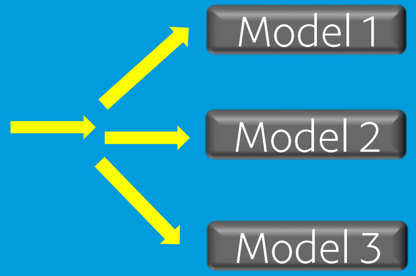
Segment Model

- Models on different subsets of data



Model Ensemble

- Different models on same data

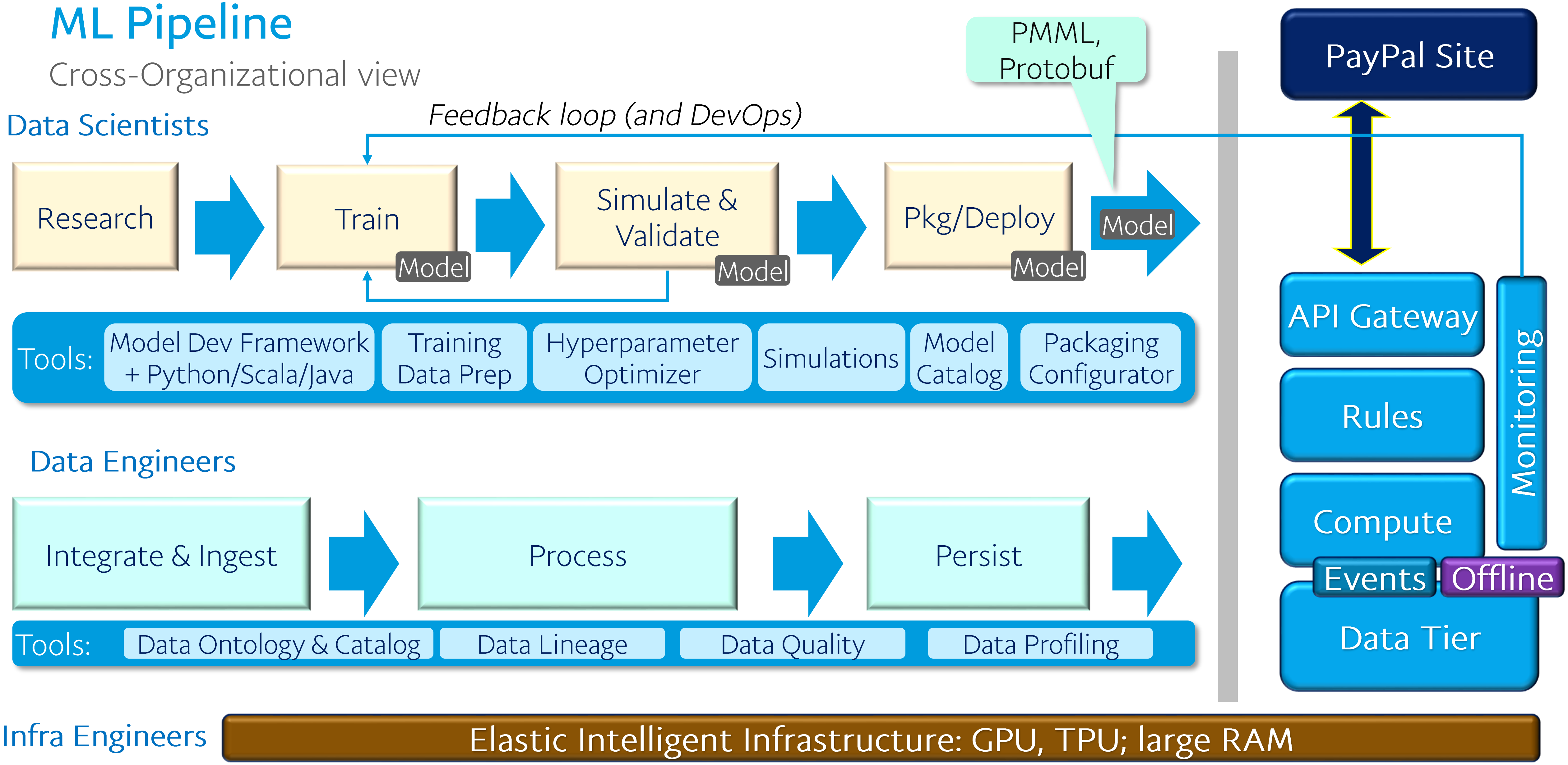


- ➢ Multiple models at checkpoint (Acct Takeover; Card Auth; Linkage…)
- ➢ Analysis of models' performance (sample group; champion-challenger…)

Model Development Process and Roles

ML Pipeline

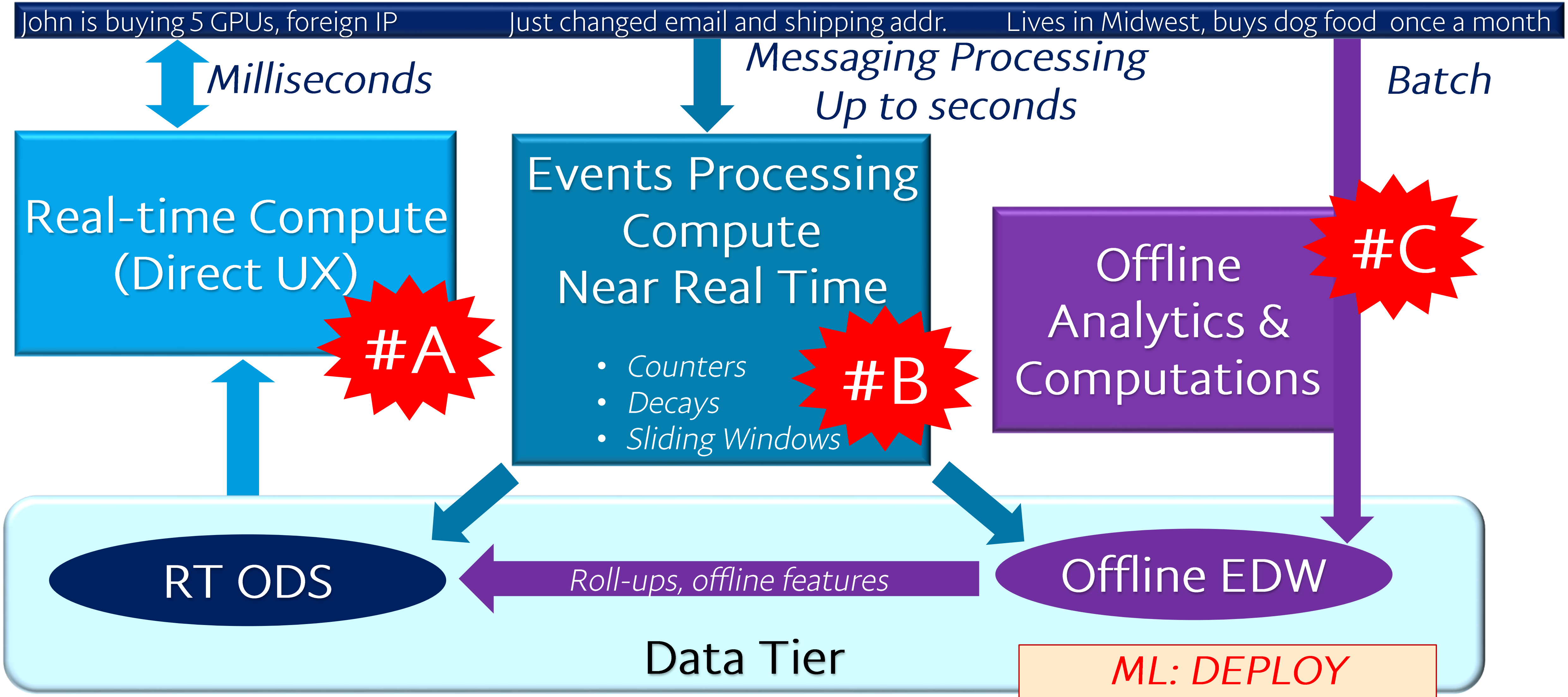
Cross-Organizational view



Production Platform: Real-Time Inference At Scale

Three Velocities of the Data Flows

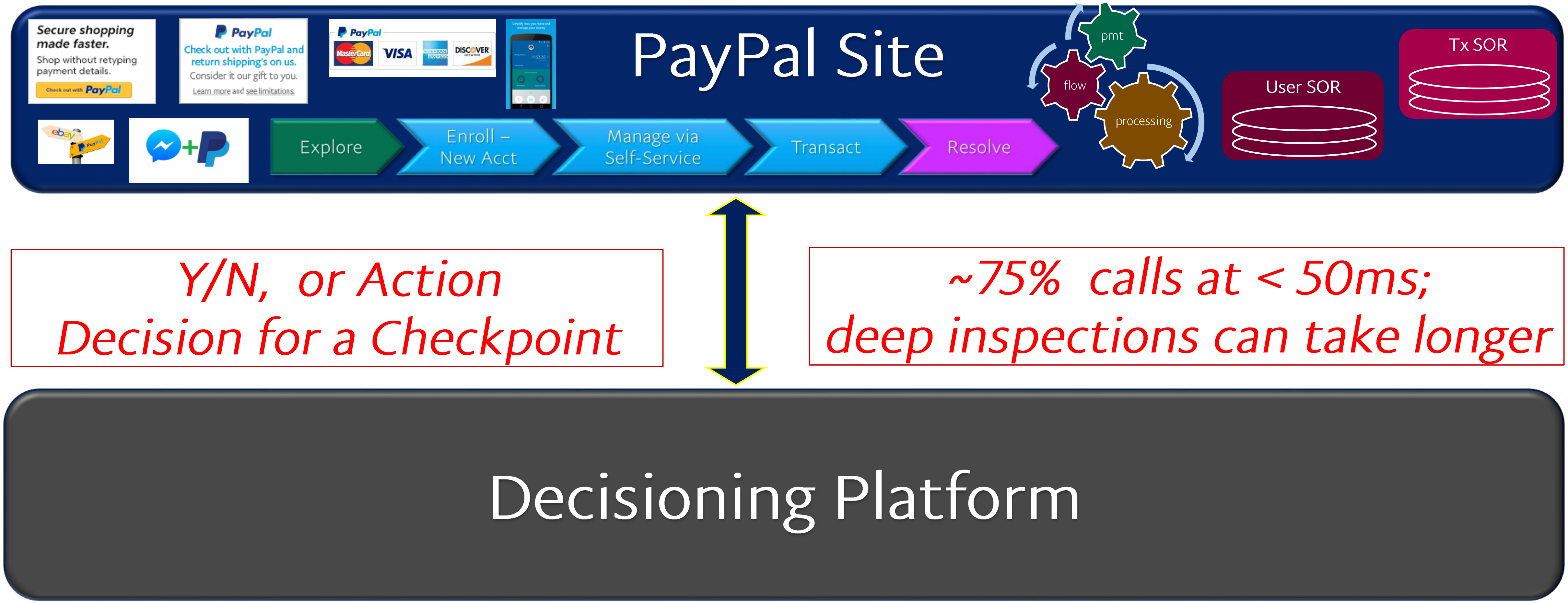
Where to execute ML models (Inference) – in #A, or #B, or #C?



ML: DEPLOY ANYWHERE

A Story of a Payment: Serving Decisions at Checkpoints

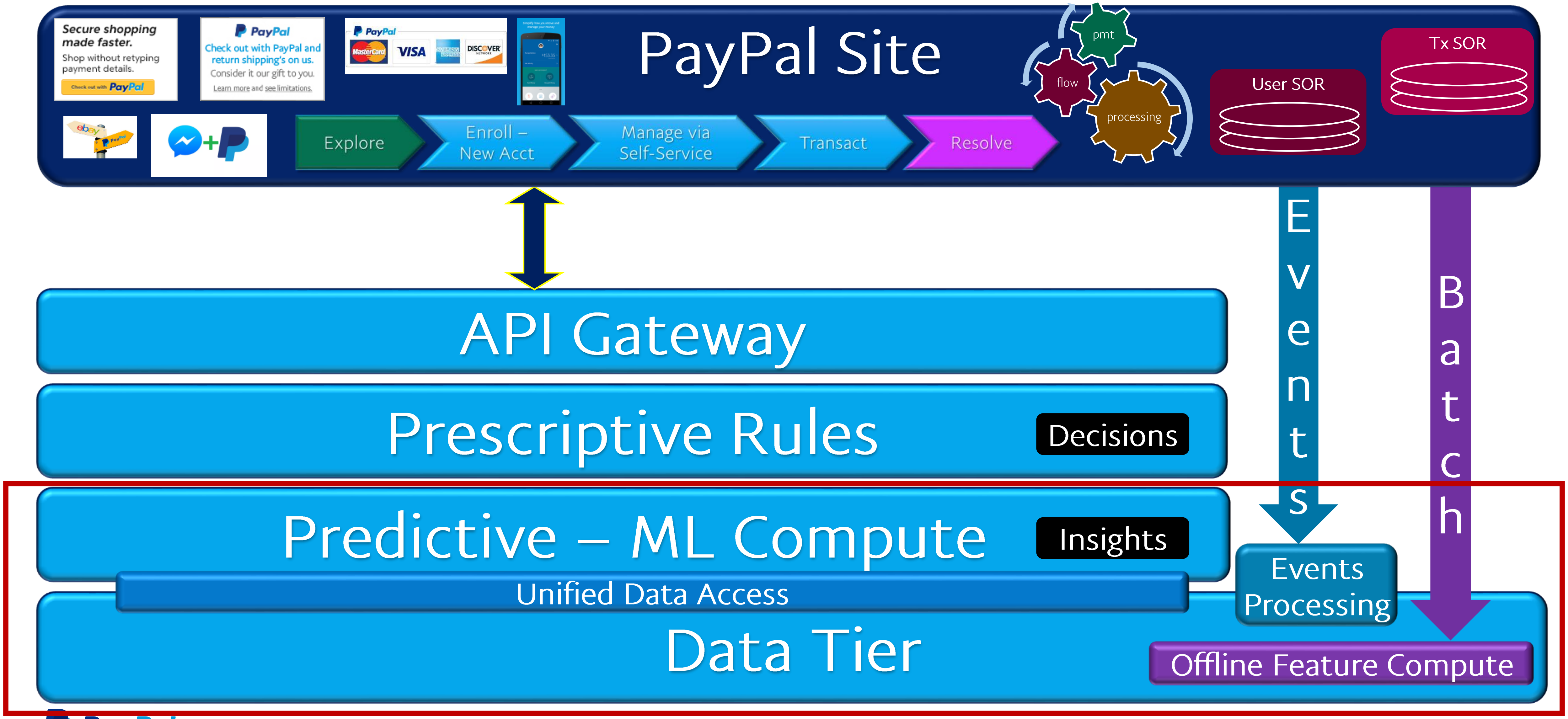
Decisioning flow



Fail-Open or Fail-Close? – ask Biz & Compliance

The Anatomy of Decisioning

Decisioning flow



Model Integration Pattern

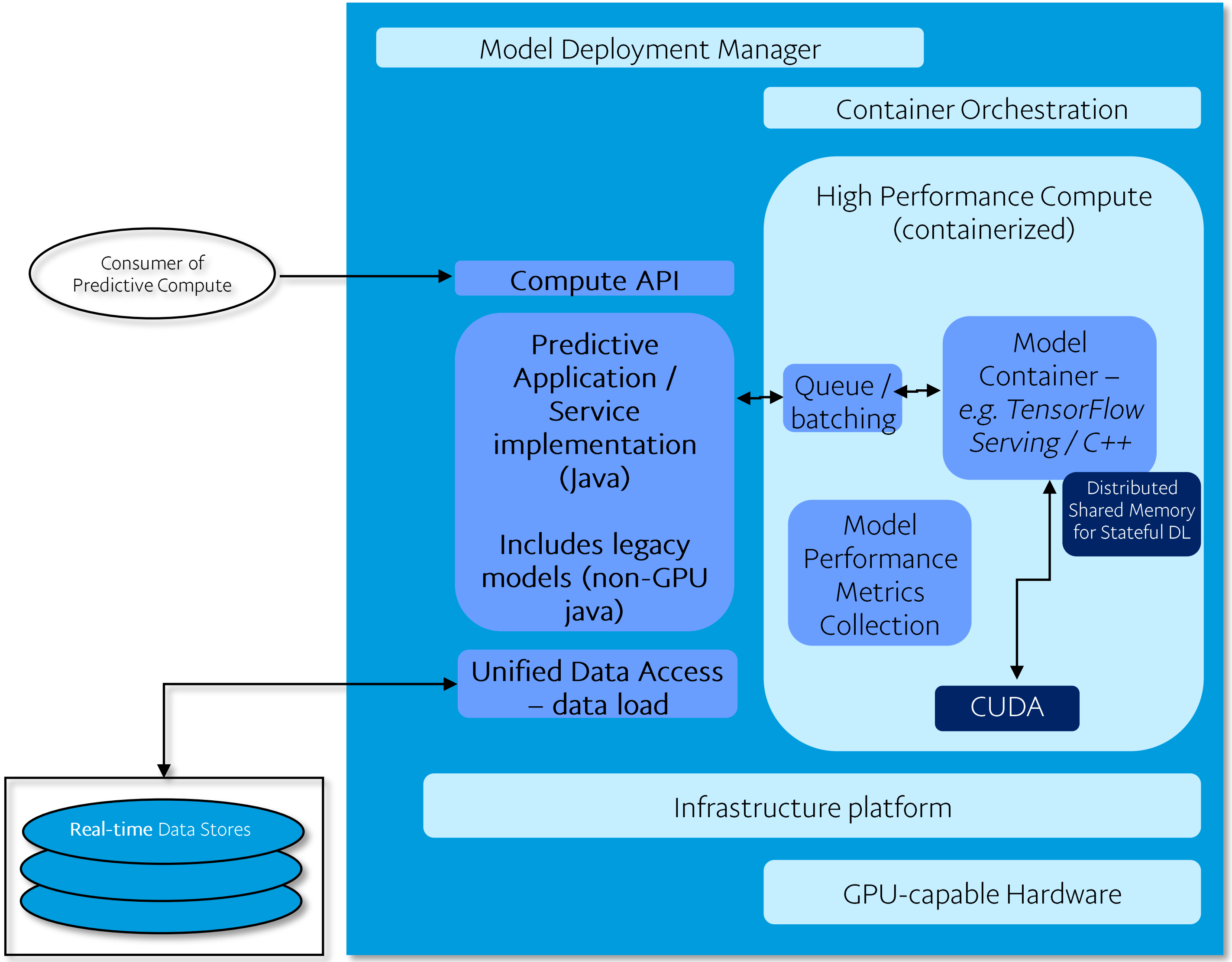
ML inferencing

Requirements

- Framework agnostic
- Support complex co-existing model portfolio: Ensembles, Cascades
- Automated model version deployment w/o production stack downtime
- Reuse of the model deployment pattern across RT / NRT / Offline-analytical
- Unified data access – componentized; supports Production and Simulations

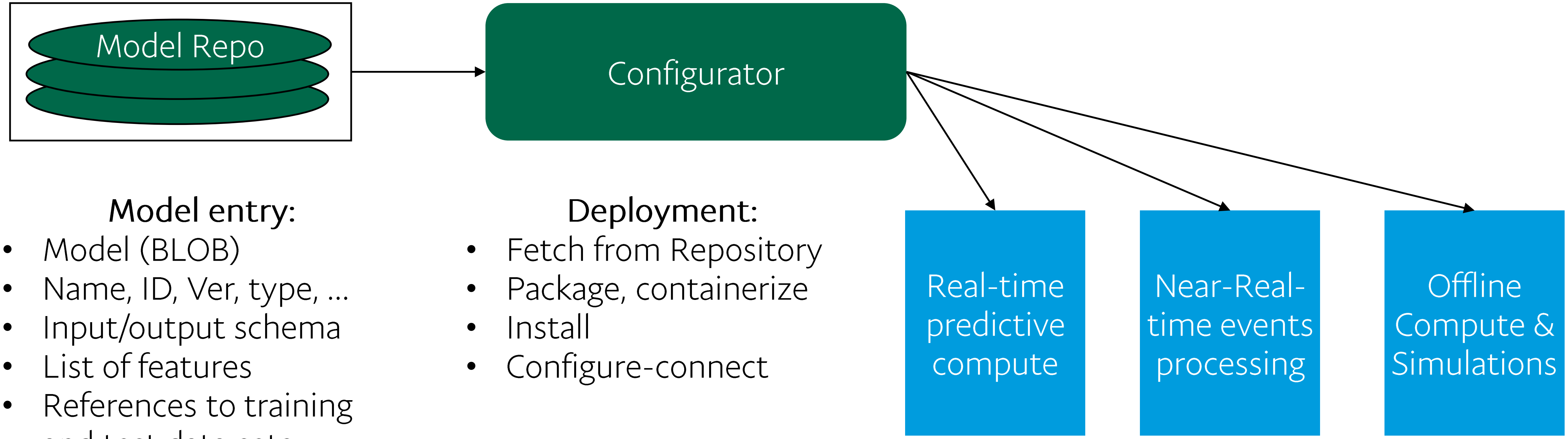
Challenges

- Manage execution digraph & config
- Dynamic updates / zero downtime
- Efficiency of data loads



Model Repository and Deployment

Supporting agile lifecycle for the models in production



Model entry:

- Model (BLOB)
- Name, ID, Ver, type, …
- Input/output schema
- List of features
- References to training and test data sets
- Business metrics
- NFR parameters (sizing, latency,…)
- Lifecycle status

Deployment:

- Fetch from Repository
- Package, containerize
- Install
- Configure-connect

How to Manage Data?

Data Tier

Types of data stores

Features from DW historical data

Features from Events

New data –stream into DW

Data Stores

Real-time ODS	Near-Real Time Streaming; Big Data NoSQL	Enterprise Data Warehouse
ORACLE, AEROSPIKE	Spark, Apache Kafka, Apache HBase	hadoop, TERADATA

- ~1% data volume (1PB):
  - Service contexts
  - Events history (near-term)
  - Precomputed features – from offline and Events/Near-RT

- Need Big Raw Data in NRT for Deep Learning
- Considerations:
  - *Key space*
  - *Read or Write optimized?*

- ~99+% data volume
  - Historical raw data (available as Point-in-Time)
  - Features

Cloud Appeal, but Beware of Compliance, Privacy.

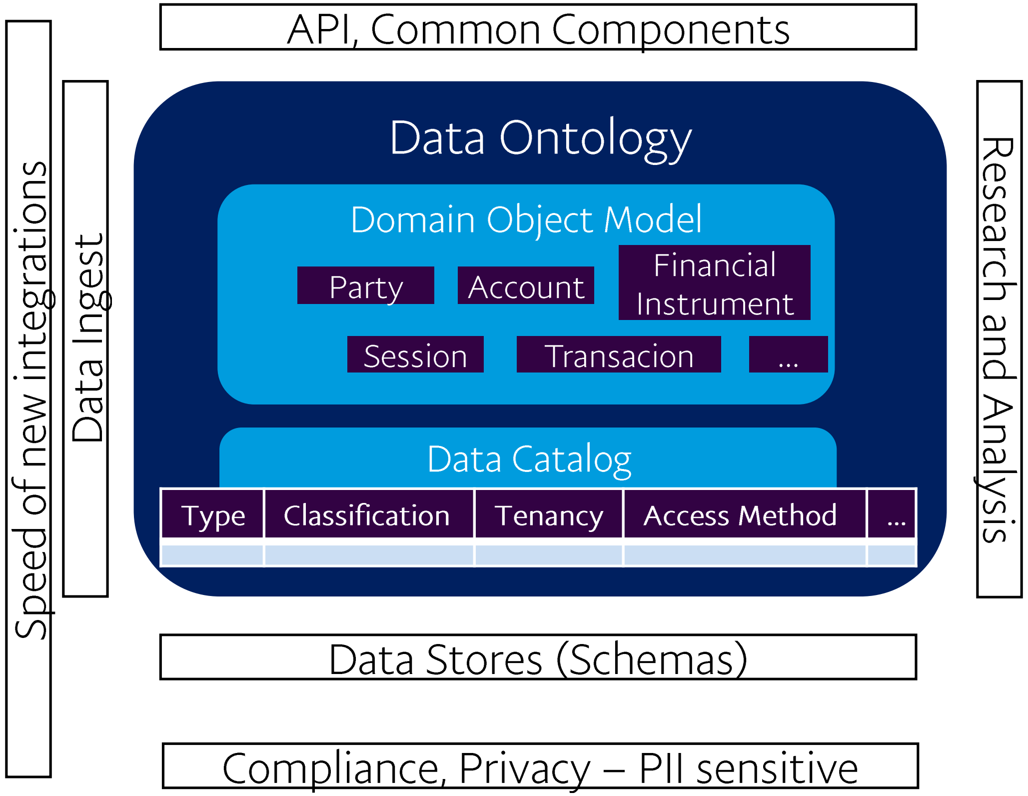
Data Management Discipline

FinTech Rigor for Compliance, Security and Privacy

- Know your data:
  - Raw data
  - Features with lineage to raw data
  - Models (and rules)

Challenges

- ❑ Data Quality; Lineage
- ❑ Privacy & PII
- ❑ Multi-data-center
  - Eventual consistency
  - Geo-distribution and locality



Conclusion

Takeaways

- **Modeling:** Review business performance of DL vs simpler models
- **Model deployment:** Choose Real-time vs Near-RT vs Offline
- **Data:** Have a data store strategy with clearly defined data processing flows, and know your data
- **Infrastructure:** Analyze ROI for GPU inferencing (unlike training)
- **DevOps:** Automated deployment & config mgmt
- **Architecture:**
  - Framework / product agnostic
  - Modular – separating Compute from Data Access

To be continued.....

Thank You!

Mikhail Kourjanski
Principal Architect

Email: mkourjanski@paypal.com

Linkedin: https://www.linkedin.com/in/mikhail-kourjanski-79358/