Real-world patterns for continuously deployed advanced analytics
Why?

1. React to market conditions faster, shrink time to production

2. Improve systems quality by failing fast with early stage, more frequent, more thorough feedback

3. Increase the rate at which models are promoted from the laboratory to production

4. Reduce the risk of platform upgrades and application deployments failing

5. Empower teams, break down people silos, data science, engineering and operations
What?

1. Low impedance, frequent workload, model and platform deployments and upgrades

2. Mandatory, multi-stage testing: unit, dependency, system, user, regression, back, and non-functional testing

3. Automated, immutable, idempotent, rollback-able, source controlled deployments

4. Flexible Blue/Green platform, A/B application release processes

5. Optionally leverage the elasticity of the Cloud
Logical Pilot Delivery Pipeline
Logical Nascent Delivery Pipeline

- **Data Engineers**
  - 0% Issues
  - Monthly

- **Production**
  - User Acceptance Test
  - Backup Data
  - Production Workload
  - Data Science

- **Development**
  - DS, Analysts, Apps
  - 100% Issues
  - Monthly - Yearly

- **Operations**
  - Monitor
  - Provision
  - Automate

- **Governance**
  - Audit
  - Security
  - Lineage
Logical Staged Delivery Pipeline

Data Engineers
- 0% Issues
- Weekly
- Workstation
- Development

Dev Ops
- 10% Issues
- Weekly
- SLA
- System Smoke Test

DS, Analysts, Apps
- 90% Issues
- Monthly - Yearly
- Production
- User Acceptance Test
- Backup Data
- Production Workload
- Data Science

Operations
- Monitor
- Provision
- Automate

Governance
- Audit
- Security
- Lineage
Machine Learning Demonstration

1. **Data Analyst**: Examine energy production metrics using the ANode UI in Chrome, running on a Raspberry Pi

2. **Data Scientist**: Select-algo, re-train and cross-validate the AModel energy production forecast model using CDSW and Altus on EC2/S3

3. **Data Engineer**: Validate unit-test and system-test from ANode module in IntelliJ, single node CDH

4. **Dev Ops**: System test all ASYSTEM modules, release new model using GitHub, Jenkins on EC2, multi node CDH

5. **Data Analyst**: A/B accuracy testing of new energy production forecast model using the ANode UI
Analytics Demonstration

A new set of modules providing a data streaming, preparation and analytics pipeline, built from the ground up from a data engineer's laptop, automated via Jenkins.

1. **Data Engineer**: Develop a streaming module and set of unit tests in IntelliJ

2. **Data Engineer**: Develop a data preparation module, with streaming module driven unit tests in IntelliJ

3. **Data Analyst**: Query the data preparation module derived dataset via Hue, embed successful SQL logs into new analytics modules, via Jenkins and Optimizer

4. **Data Engineer**: Upgrade Cloudera platform, run tests

5. **Data Engineer**: Develop non-backwards compatible version of the preparation module, with failing inter-module dependency tests via Jenkins
How?

1. Take inspiration from the apps dev community and apply to your data practices

2. Partner with an organisation that has an opinion and the experience to fast track your journey

3. Choose a machine learning and analytics platform that codifies the best practices

4. Consider the Cloud, if not for today, avoid precluding it for the future

5. Start now, take a phased approach, iterate and refine over time
Resources

- Cloudera Framework Example
  - https://github.com/ggear/cloudera-framework

- Cloudera Parcel Maven Plugin
  - https://github.com/ggear/cloudera-parcel

- Cloudera Manager API
  - https://cloudera.github.io/cm_api/apidocs/v17/index.html

- Cloudera Navigator API
  - http://cloudera.github.io/navigator/apidocs/v3

- Cloudera Altus

- Cloudera Director

- Cloudera Data Science Workbench

- Cloudera Optimizer
  - https://optimizer.cloudera.com
Thank you, questions?

Graham Gear, graham@cloudera.com