sky BETTING & GAMING

Data Juggling at Sky Betting and Gaming

A brief look inside the Data Science toolbox

Intro to SB&G

- 100% online sports betting and gaming operator predominantly serving the UK however actively building out propositions in Italy and Germany
- High frequency, so a data rich business
- Market leaders in the UK online market (we have the most online customers across last 12M).
- Very mobile focused (80%+ on SkyBet)
- Highly regulated (PCI, UKGC), leads to key data and operational requirements
- Circa 1,200 employees
- Head office in Leeds, with other offices in Sheffield, Guernsey, Rome, and Munich
- Sunday Times top 100 company to work for in 2016

CVC
Capital Partners

sky

sky BETTING
& GAMING

Our products

Who we are

Darrell Taylor (Principal Data Engineer)

- Software engineer
- Background – Electrical Engineer, Telecoms, eCommerce, Big Data

James Waterhouse (Head of Data Science)

- Joined SBG&G in 2010
- Held numerious roles across analytics, insight and strategy
- Graduated in 2007 BSc Maths & Physics from University of Leeds

Data Journey at SB&G

Oracle – pre 2013

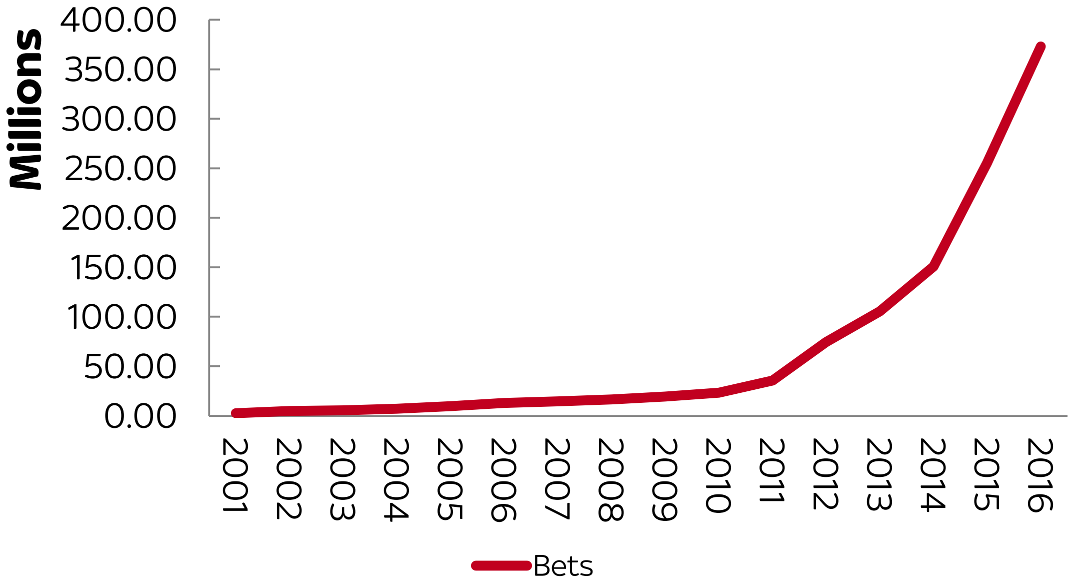
- Data team of one
- Shared Oracle data warehouse with Sky Group
- Daily Batch – 24 hour lag
- Often exceeded platform capacity

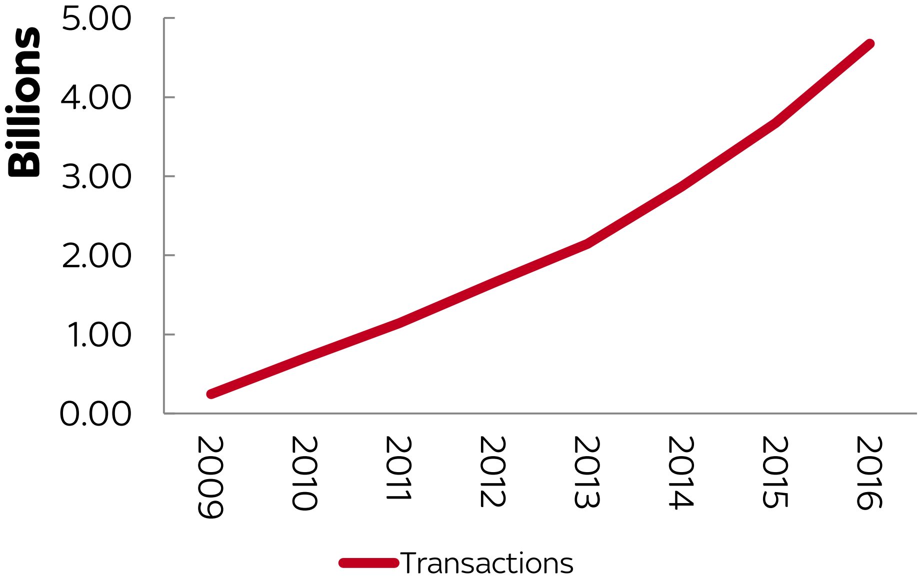
Hadoop – 2013 to present

- Closer to real time data
- Ingest more information sources
- Enable Data Discovery
- Data Driven

Data Journey at SB&G

Data Growth





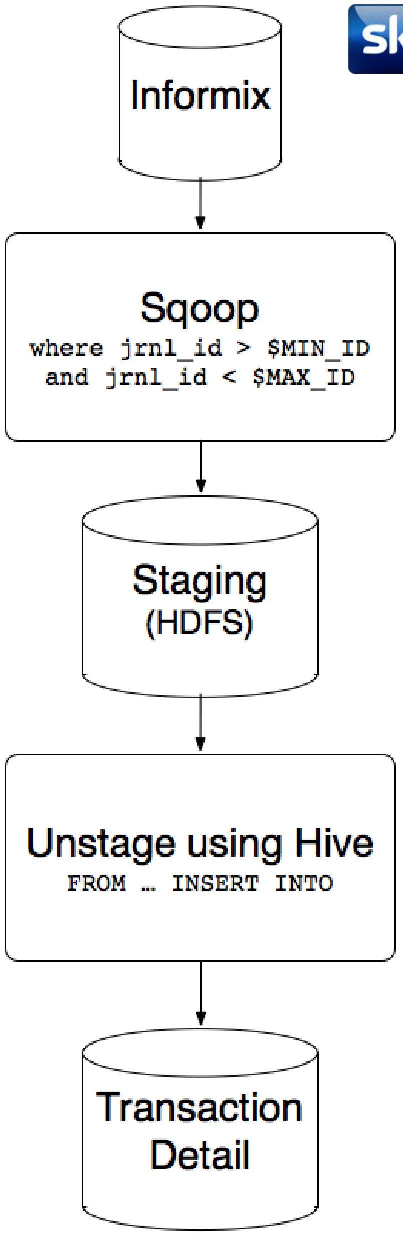
Ingest Overview

Sqoop 'new' data from Informix into a staging area

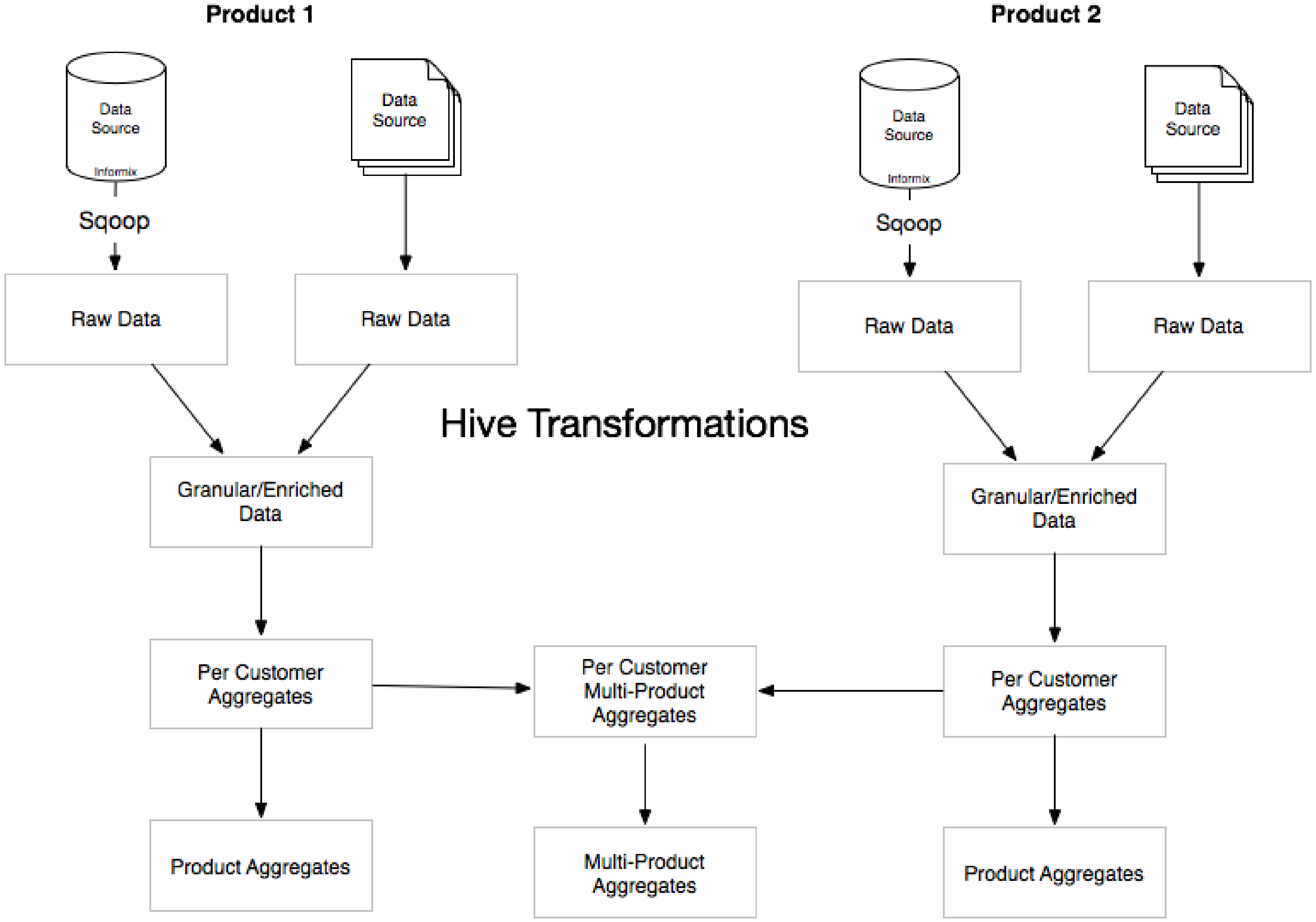
- Definition of new depends on pipeline, examples are increasing primary key id, date ranges (creation/modification)

Copy and transform staged data into a destination 'detail' table

- Business logic and data cleansing
- Determine new threshold values for next import



What do we do with it all?

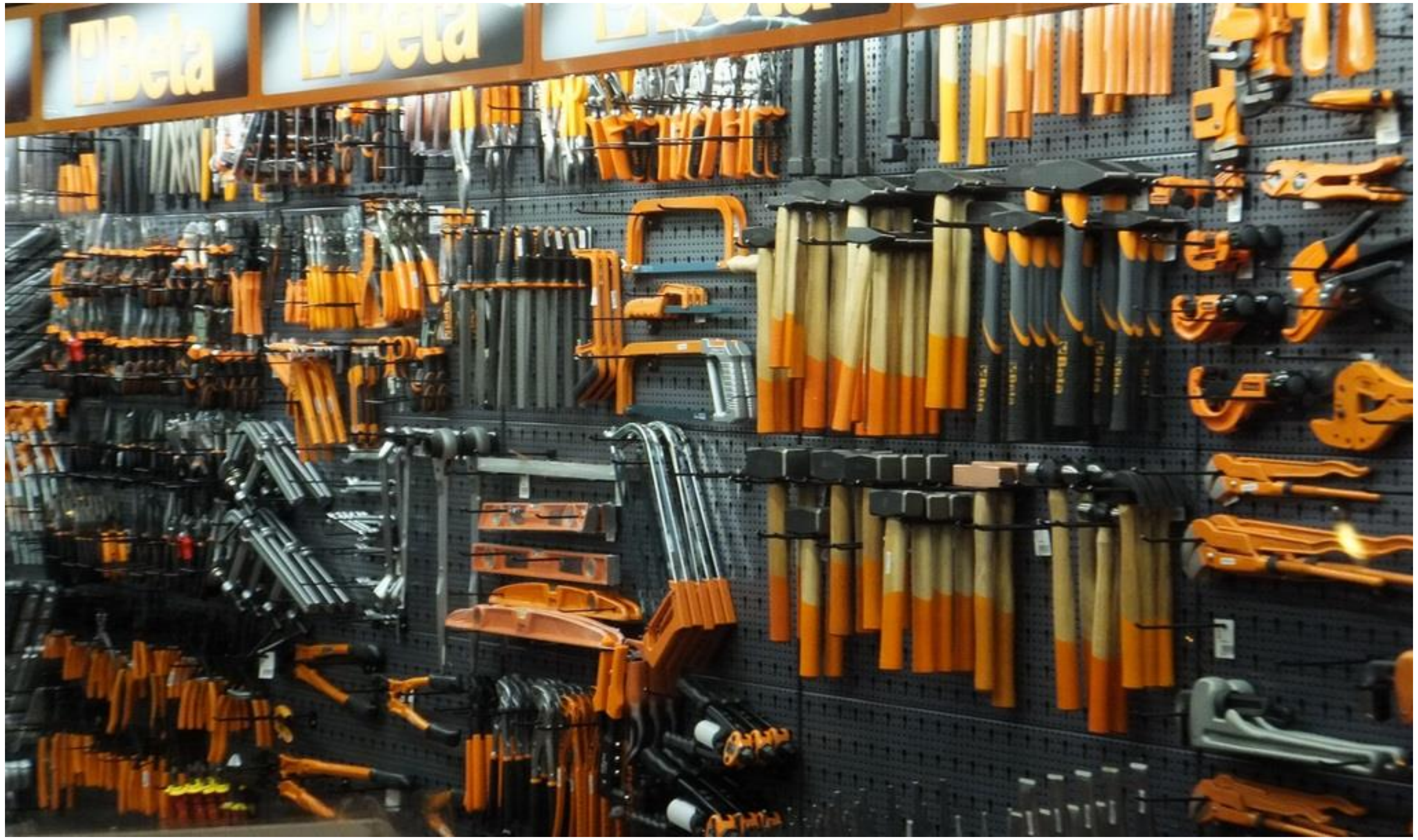


SQL Sledgehammer

- Most of our analytics teams use SQL
- Familiar and easy to work with
- Most data ends up in Excel
- Impala allows for analysis of much bigger datasets, previously too large to work with in Oracle
- Even with increased scale and speed, we need to combine with something that's more refined to enable our data science

Pick the right tools for the job

- Lots of tools and new technologies in a space that is constantly evolving.
- Important to make the right choices at the right times.
- Must be prepared to test and fail quickly.



Keep it simple

- The predictive models at the top of our build list used relatively small datasets (1-2M rows)
- No requirement to continually retrain the models
- Only necessary to score customers on a daily basis
- Made the decision to run R locally, with Impala doing most of the data processing work up front
- Allowed for easy local model development in a familiar environment
- Removed the headache of problems associated with distribution

How do we use R in Production?

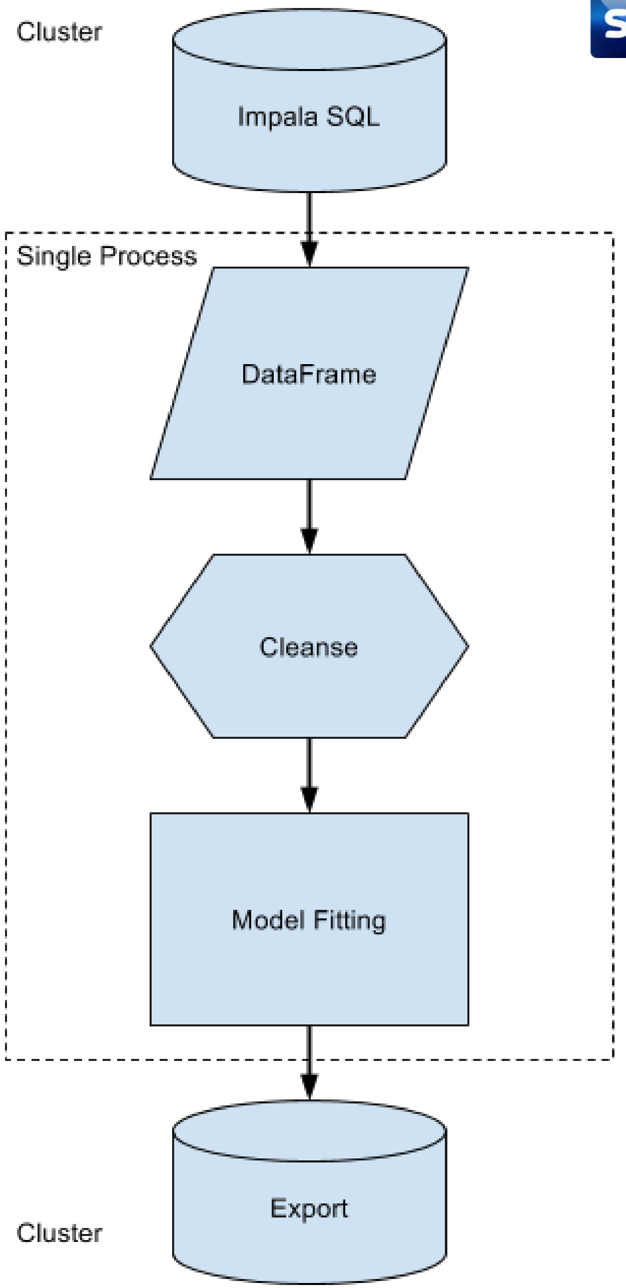
- Impala – SQL Query
- DataFrame
- Cleanse – R functions, data types, NULLs etc.
- Model Fitting – predict()
- Export – CSV > HDFS > Hive

```r
cleanse_data <- function(input_data ) {

  input_data$first_dep_month <- as.factor(input_data$first_dep_month)

  input_data$first_dep_dayofweek <- as.factor(input_data$first_dep_dayofweek)

  dummy <- ifelse(input_data$age_group == "UNKNOWN", 1,  input_data$age_group)
```



Models in production

- We now have 30+ Models running overnight in production.
- Models include:
  - Cross-brand propensity models
  - Churn
  - Early problem gambling identification
  - Customer future value prediction
  - HV value customer identification
- Models exported into an Oracle presentation layer for use in CRM via IBM Campaign
- Various applications of model within our Operations team

Speed to production

- R framework limits us with regards to the models we build
- However it means we have a very quick route to production
- A new model can be designed, built, trained, tested and release into production in less than a week

What can we do better?

- Model Training
  - Currently ad-hoc and semi-repeatable
- Development process
  - CI with R
  - Remove dependency on Impala for dev
- Automated Testing
  - Docker environment to run all tests off a pull request
- Deployment
  - Model versioning
- Dependencies
  - Docker environment again, pre-built with all the correct dependencies
- Data Dictionary
  - Data lineage and relationships. Neo4j

Team structure

- We're more Frankencorn than Unicorn
- Team consists of data scientists, an engineer and test resource
- Importantly plenty of domain knowledge
- The more we work together, the more broad our skillsets become

Future plans

- PySpark
  - Common Python packages
- Notebooks – Jupyter, Zeppelin (TBD)
  - Currently use local Jupyter notebooks with Docker
- Streaming – Near real time
  - Promotions team use Kafka Streams for near real time churn prediction
- CI and Automation
  - More of this

Questions?