Druid.io - technologia zmieniająca firmy

Przedstawienie

Piotr Guzik - kim jestem?

- Założyciel firmy Datumo
- Wieloletni architekt i programista technologii Big Data (Spark, Kafka, Druid)
- Trener
- Wykładowca

O czym Państwu opowiem

1. Typowa sytuacja w analityce danych w firmach
   a. Typowe wyzwania
   b. Typowe popełniane błędy
   c. Najlepsze praktyki
2. Use-case 1 - iTaxi
3. Use-case 2 - GoldenLine
4. Druid - technologia przyszłości? Czy sama technologia wystarcza?

Stan zastany analityki w firmach

4 etapy zaawansowania firm:

1. Króluje Excel i raporty ad-hoc
    a. Analitycy nie weryfikują hipotez
    b. Wykonują odtwórczą i czasochłonną pracę
    c. Raporty z różnych źródeł mogą dawać różne wyniki - są “rozjazdy”
    d. Masa pracy generowanej na koniec miesiąca / kwartału
    e. Brak centralizacji danych
    f. Praca na plikach - brak współdzielenia

Excel i ad-hoc

Typowe rozwiązanie:

- Centralizacja danych
- Uporządkowanie danych
- Dostarczenie systemu informatycznego, który zastąpi Excela

Typowe błędy:

- Tableaupodobny system rozwiąże wszystkie nasze problemy
- Analitycy nauczą się SQL i od teraz wszystko będzie pół-automatyczne

Baza danych i SQL

Typowe wyzwania:

- Każdy raport musi być zlecony analitykom / IT
- Działa analiz lub IT to “wąskie gardło”
- Każda zmiana w raportach wymaga zmian w ich przygotowaniu (nie da się uzyskać dowolnego przekroju przez dane)
- Raporty generują się na koniec dnia - nie wiemy jak idzie nasz biznes tu i teraz
- Dane nie są dostępne łatwo i nie dla każdego

Baza danych i SQL

Typowe rozwiązanie:

- Zakup / stworzenie systemu BI
- Budowa hurtowni danych
- Automatyzacja generacji typowych raportów

Typowe błędy:

- Zakup komercyjnego i bardzo drogiego rozwiązanie - gdzie wystarczyłoby coś prostego
- Przekonanie, że dobry BI rozwiąże problem analizy ad-hoc

Hurtownia danych

Typowe wyzwania:

- Rozrasta się infrastruktura (sprzęt) i wraz z nią dział IT do obsługi
- Wraz ze wzrostem ilości danych - potrzeba coraz to droższych rozwiązań (skalowanie wertykalne vs horyzontalne)
- Chcielibyśmy zastosować techniki Data Science i Machine Learning - ale sama hurtownia tego nie daje - co robić?

Hurtownia danych

Typowe rozwiązanie:

- Inwestycje w skalowaną hurtownię bez komercyjnych licencji
- Poszukiwania wygodnego UI do pracy z danymi
- Zakup/budowa modułu DS/ML do pracy z danymi

Typowe błędy:

- XXI wiek - inwestycja w Hadoopa
- Wiara, że produkt “z pudełka” zrobi nam AI

AI all the thing!

Typowe wyzwania:

- Skąd wezmę ludzi, którzy znają się na AI/ML?
- Garbage in = Garbage out
- Brak przemyślenia jakie są automatyczne sposoby mierzenia uzyskanych wyników
- Brak przemyślenia w jaki sposób będziemy aktualizować modele na produkcji
- Uprodukcyjnienie modeli ML

AI all the thing!

Typowe rozwiązania:

- Utworzenie zespołu Data Science
- Testy i próba pracy z danymi pod kątem ML
- Osobny zespół data science vs przydzielenie data scientistów do zespołów

Typowe błędy:

- Brak wiedzy jak wdrażać ML na produkcję
- Tworzenie modeli PoC które nie dadzą się zastosować na produkcji

Najlepsze praktyki

- Wytypowanie “krytycznych” zbiorów danych w pierwszej kolejności (często to promil danych)
- Sparowanie analityków i ludzi IT
- Odpowiedź na pytanie - czy poradzimy sobie sami? Czy mamy sprzęt oraz zasoby, które stworzą nam Big Data?
- Rozważenie wykorzystania chmury
- 3x zastanowić się zanim kupimy “rozwiązanie z pudełka” - bez customizacji

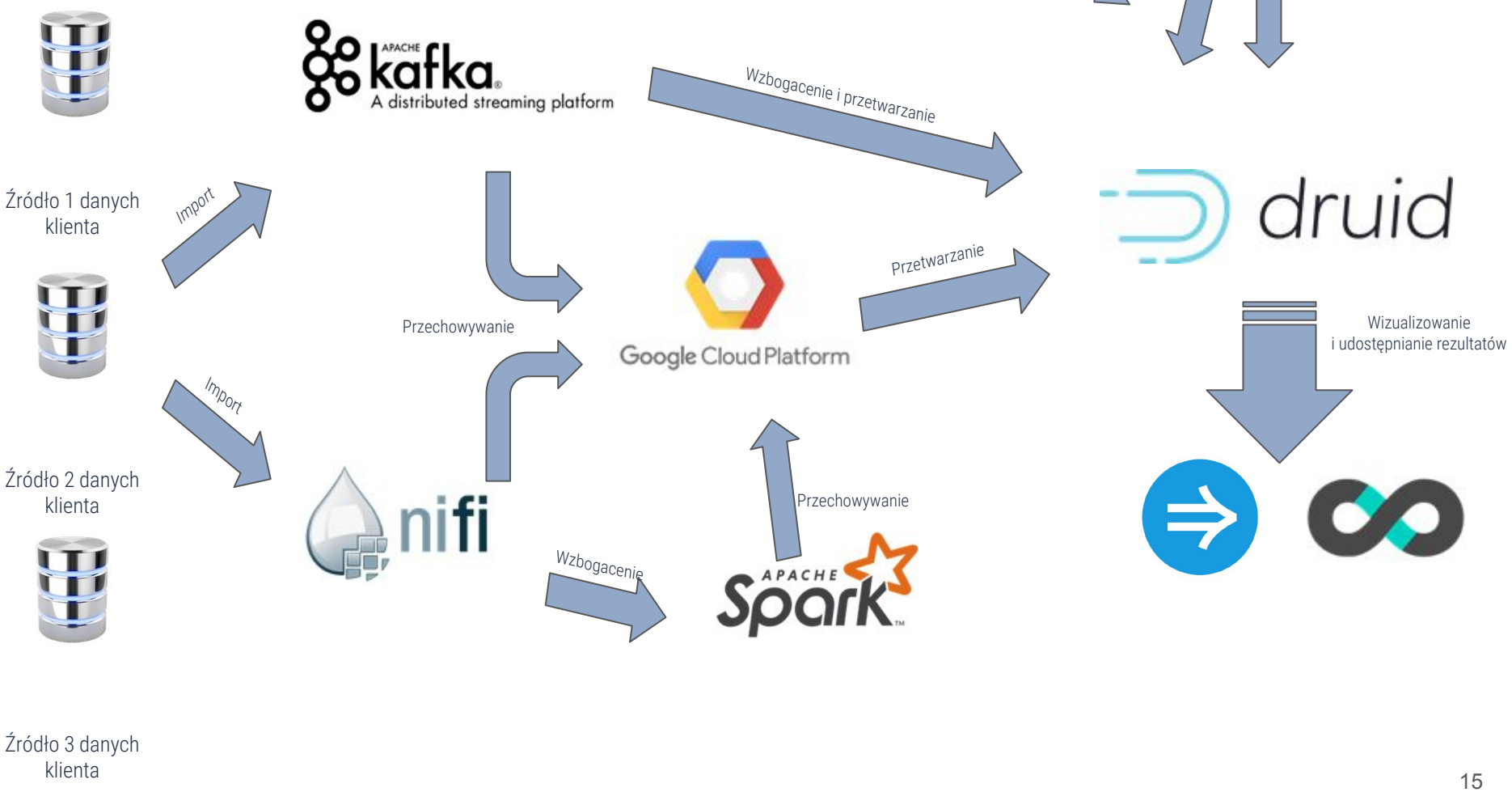
iTaxi

- Młoda, dynamiczna firma (brak dużego IT)
- Przestała się mieścić w Excelu - raporting był procesem żmudnym i pół-manualnym
- Biznes przewozów jest typowym biznesem real-time -> dane powinny być dostępne z krótkim opóźnieniem (sekundy, pojedyncze minuty - NIE godziny)
- Wiele rodzajów pod-biznesów (kursy, flota, marketing, kohorty klientów)

iTaxi

- Potrzeba demokratyzacji danych - wszyscy biznesowi pracownicy mogą obejrzeć stosowne raporty
- Duży potencjał Data Science - od działania algorytmów zależy np. czas dojazdu
- Potrzeba spojrzenia na dane z wielu różnych perspektyw i przekrojów
- Preferowany model płatności - pay as you go (idealny pod chmurę)

Architektura



Nie kupujcie kota w worku

- Zawsze domagajcie się krótkiego PoC
- W ramach PoC
  - dostarczamy dane w formie “zjadliwej” np. CSV, Excel
  - nie tracimy czasu na integrację (“trup w szafie”)
  - określcie typowe zbiory raportów, które są niezbędne
  - poproście o dostęp do systemu na okres testów - około tygodnia

iTaxi - początek

- Darmowy PoC
- Zajął 2 tygodnie
- Wytypowano krytyczny zbiór danych o kursach (oczywiście NDA)
- Dostarczono zbiór oczekiwanych raportów
- Oddano system do testów dla analityków na tydzień
- Podpisano umowę time and material

iTaxi - wymogi do współpracy

- Potrzeba integracji z bazą danych
- DEMO - apache NiFi (model Pull)
- Oddano nam do wsparcia analityka, który znał dane
- Analityk zbierał wymagania po stronie iTaxi - był decyzyjny (Project Manager)
- Co tydzień pokazywaliśmy postęp prac i ustalaliśmy co robimy dalej

iTaxi - rozwiązanie

- Dane zrzucane co 5 minut na Kafkę (real-time)
- W nocy przeliczenie danych za cały dzień (dokładniejsze wyniki)
- Kod biznesowy - “wzbogacenia” danych zawarty w Sparku
- Automatyzacja z użyciem Airflow - DEMO
- Druid jako hurtownia danych
- Chmura jako miejsce wdrożenia (dyskusja o RODO)

iTaxi - efekty prac

- Po 2 miesiącach mieliśmy gotowe raporty dotyczące core danych
- W ciągu kolejnych 2 miesięcy dodaliśmy raporty dotyczące danych pobocznych
- Wdrożono analitykę real-time (5 minutowe snapshoty danych)
- Pełna automatyzacja raportów
- Wdrożono 2 systemy UI (Otwarte DEMO):
    - Drill-down dla analityków
    - BI dla managerów i ludzi biznesu

Działania prewencyjne i detekcja anomalii w czasie rzeczywistym

Opis sytuacji

Zamówienia są odrzucane ze względu na brak dostępnych taksówek w najbliższej okolicy.
Firma mogłaby wykonać więcej kursów, jeżeli wiedziałaby o problemie braku samochodów w danym rejonie odpowiednio wcześniej.

Nasze rozwiązanie

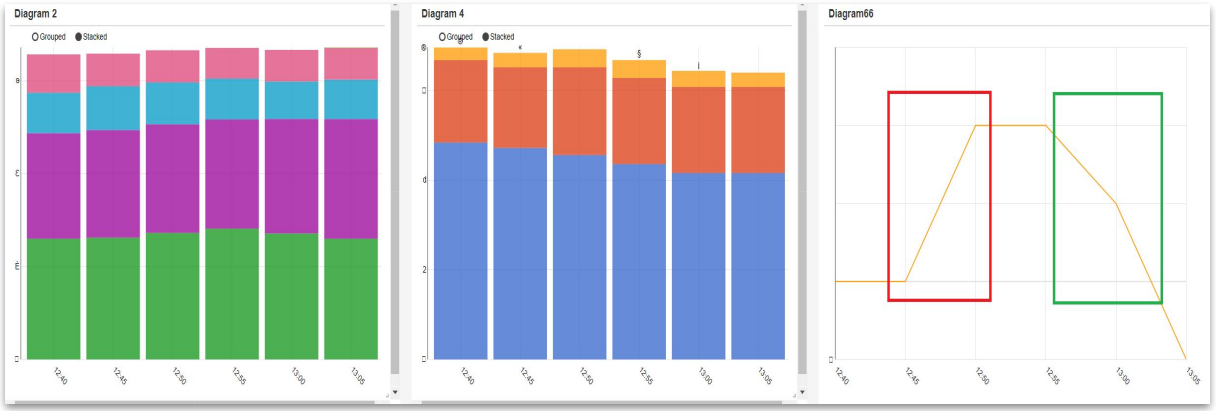
Zastosowanie ciągłego monitoringu (real-time):
liczby zamówień odrzuconych ze względu na brak taksówek.

Wykrycie anomalii (czerwony prostokąt):
wysłanie powiadomienia zawierającego współrzędne zjawiska.

Reakcja firmy (zielony prostokąt):
skierowanie taksówek do wskazanego obszaru.

Co zostało ulepszone?

Błyskawiczne przetwarzanie oraz aktualizowanie danych z maksymalnie jednominutowym opóźnieniem umożliwia niemal natychmiastową detekcję anomalii.
Ciągły przepływ danych daje więcej czasu na odpowiednią reakcję na zaistniałe wydarzenia.



Poprawna interpretacja danych

Opis sytuacji

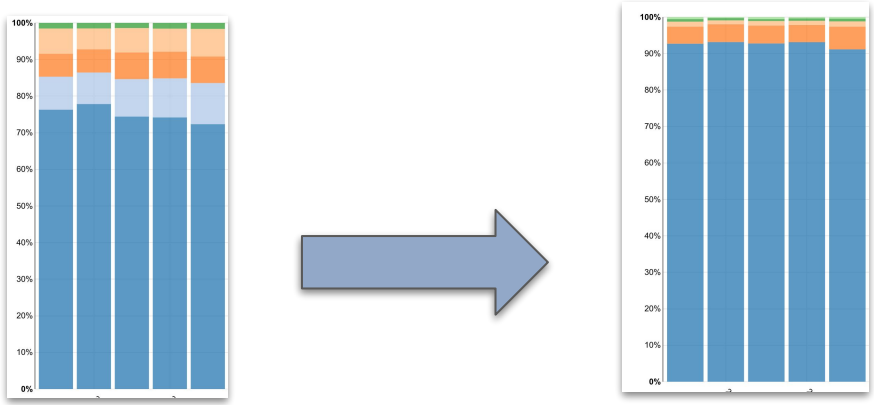
Firma zmagała się ze zbyt dużą liczbą odrzuconych zamówień. Każde kolejne odrzucenie w aplikacji mobilnej było liczone osobno. Prowadziło to do niepoprawnej interpretacji danych, wskazującej na nieintuicyjny interfejs w aplikacji mobilnej.

Nasze rozwiązanie

Wykonaliśmy narzędzie oparte na algorytmie *"Window Functions"*. Każda próba zamówienia tego samego użytkownika w małym odstępie czasu jest agregowana. Dzięki temu jeden z kluczowych KPI stał się wiarygodny i aktualnie stanowi istotną podstawę oceny rozwoju biznesu.

Co zostało ulepszone?

Agregacja jest wykonywana przed wgraniem danych do systemów wizualizacji. Czasochłonna praca analityków w celu przygotowania danych nie jest już konieczna.



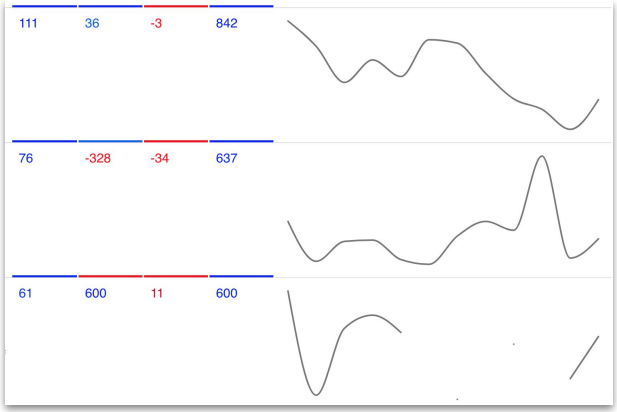
Monitorowanie zadowolenia klienta

Opis sytuacji

Firma powinna monitorować zamówienia stałych klientów. Powinna być świadoma sytuacji, w której klient składa mniej zamówień niż dotychczas – pozwala to na szybką reakcję i poprawia relacje.

Nasze rozwiązanie

Wykonaliśmy narzędzie, które przypisuje klientów do trzech różnych kategorii: przeciętna, powyżej przeciętnej i poniżej przeciętnej z dodatkową klasyfikacją wydajności: stała, rosnąca, malejąca. Kiedy użytkownik zmienia status, zostaje to odpowiednio odnotowane.



Co zostało ulepszone?

Opracowanie takich raportów wymaga filtrowania i grupowania bardzo dużych ilości danych. Nasze rozwiązanie pozwala na wykonywanie zapytań różnicowych, które przekładają się na efektywność oraz dają możliwość przeprowadzania ciągłych aktualizacji.

iTaxi - support

- Kanał Slack dla biznesu
- Kanał Slack dla IT
- Slack Bot - automatycznie informuje o wystąpieniu awarii
- Pełna transparentność kosztów Cloud
- Wspólne ustalanie rozwojów oraz ich wycena

GoldenLine

- Firma posiadająca duży wolumen danych (3 MLN użytkowników wraz z ich aktywnością)
- Niemożność ręcznej pracy i dopasowania ofert pracy do kandydatów
- Potrzeba wdrożenia AI/ML aby grupować podobnych kandydatów i precyzyjnie dopasować oferty pracy
- Potrzeba automatyzacji raportów

GoldenLine

- Spory i bardzo kompetentny dział IT - ale zajęty swoimi zadaniami
- Brak zespołu Big Data
- Kokpity BI (licencjonowane oprogramowanie)
- Dział analiz “wąskim gardłem” raportowania

GoldenLine - start

- Darmowy PoC
- 3 tygodnie pracy
- Efektem model ML klastrujący użytkowników w określone grupy zawodowe
- Wykorzystanie algorytmu FastText
- Dostarczenie narzędzia graficznego do analizy dokładności algorytmu
- Po analizie - algorytm o 30% lepszy niż eksperci domenowi - ale 20% false positive

GoldenLine - integracja

- Solidny dział IT - preferuje model Push (nie NiFI - wady i zalety)
- Dostępy do konta techicznego do uploadu danych
- Monitoring dostępności danych
- Możliwość zmiany dostawcy rozwiązania i większa elastyczność
- Wspólna nauka schematów AVRO
- Masa pytań i komunikacji IT na Slacku

GoldenLine - jak testować model

- Przygotowanie danych testowych przez dział analiz wraz z labelkami
- Po każdej zmianie w kodzie modelu - możemy zobaczyć, jakie przyniosły one skutek - BRAK RĘCZNEGO TESTOWANIA!
- Automatyczny program, który uruchamia się na danych testowych i wylicza:
    - Accuracy
    - Recall

GoldenLine - jak często przeliczać model?

- Dane użytkowników są zmienne, ale zmiana pracy lub profilu kandydata nie jest codziennością
- Wystarczy 1 raz na tydzień
- Nadal potrzeba pełnej automatyzacji
- Kod napisany w Python - jak go wdrożyć na produkcję z użyciem klastra Apache Spark?
- Dlaczego nie pisać kodu ML w R?

GoldenLine - wyniki

- Automatyzacja raportowania
- Wdrożenie Machine Learning - automat
- Analitycy weryfikują hipotezy zamiast tworzyć raporty
- UI dla analityków do wygodnej pracy (żegnajcie licencje)
- UI dla managerów do śledzenia KPI
- Support na Slack
- Automatyczne notyfikacje o błędach

Druid

Druid.io

- Sprawdzona w boju platforma drill-down
  - Stworzona przez firmę Metamarkets, która targetuje reklamy real-time
  - Narzędzie napędzające ich biznes i niezbędne do zarabiania
  - Wypuszczone jako Open Source (obecnie Apache Foundation)

Druid.io

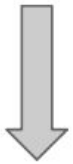
- Zalety
  - Wsparcie dla danych real-time oraz historycznych
  - Super szybka hurtownia danych
  - Stabilna - bardzo rzadkie awarie
  - Elastyczna konfiguracja kostek OLAP - poprzez pliki JSON
- Wady
  - Całość konfiguracji kostek musi być znana przed załadunkiem danych

Druid - główne założenia

Druid - rollup

timestamp	publisher	advertiser	gender	country	click	price
2011-01-01T01:01:35Z	bieberfever.com	google.com	Male	USA	0	0.65
2011-01-01T01:03:63Z	bieberfever.com	google.com	Male	USA	0	0.62
2011-01-01T01:04:51Z	bieberfever.com	google.com	Male	USA	1	0.45
2011-01-01T01:00:00Z	ultratrimfast.com	google.com	Female	UK	0	0.87
2011-01-01T02:00:00Z	ultratrimfast.com	google.com	Female	UK	0	0.99
2011-01-01T02:00:00Z	ultratrimfast.com	google.com	Female	UK	1	1.53

```
GROUP BY timestamp, publisher, advertiser, gender, country
  :: impressions = COUNT(1),  clicks = SUM(click),  revenue = SUM(price)
```



100x/1000x less records

timestamp	publisher	advertiser	gender	country	impressions	clicks	revenue
2011-01-01T01:00:00Z	ultratrimfast.com	google.com	Male	USA	1800	25	15.70
2011-01-01T01:00:00Z	bieberfever.com	google.com	Male	USA	2912	42	29.18
2011-01-01T02:00:00Z	ultratrimfast.com	google.com	Male	UK	1953	17	17.31
2011-01-01T02:00:00Z	bieberfever.com	google.com	Male	UK	3194	170	34.01

Druid c.d.

Agregacja “w locie” na poziomie 100-1000x

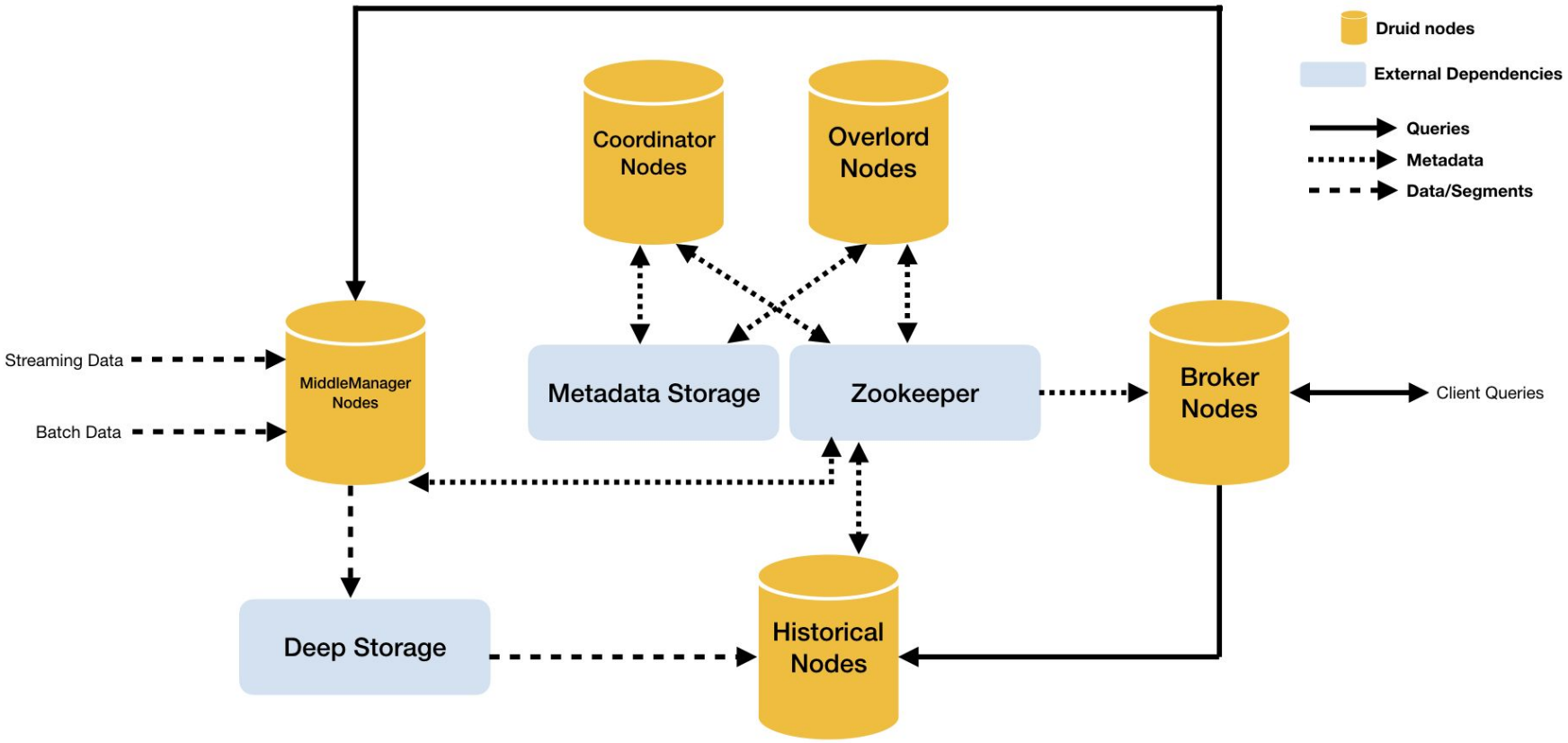
Druid.io

- Generalna zasada - im mniej danych (większy poziom agregacji) - tym lepiej
  - Nie ma sensu agregować wartości typu userId
- Dane “surowe” są tracone w trakcie załadunku danych do hurtowni - Druid przechowuje tylko agregaty:
  - Dzięki temu jest tak szybko
  - Nie ma możliwości odtworzenia rekordów (trzeba ponownie załadować dane)

Druid.io

- Bardzo krótka listów typów zapytań:
  - GroupBy - najbardziej ogólne - najwolniejsze
  - TopN:
    - Bardzo szybkie i zoptymalizowane
    - Najbardziej typowe zapytanie dla ludzi biznesu
    - Ograniczone do N pierwszych rekordów
  - Timeseries - najszybsze, zwraca jedną lub wiele agregacji - bez sortowania i poukładania
- NIE wspiera JOIN (znanego z SQL):
  - jeżeli chcemy robić JOIN-y, to musimy przygotować odpowiednio kostki w procesie załadunku danych

Druid - architektura



Druid - architektura

- Bardzo złożona
- Wiele różnych typów węzłów:
  - Każdy ma jedną odpowiedzialność (świetny design)
- Bardzo duża odporność na awarie
- W razie awarii konkretnych typów węzłów:
  - Pozostałe operują normalnie
  - Dla przykładu można czytać dane, ale nie można wstawiać nowych (brak widocznej awarii)

Druid - architektura c.d.

- Bardzo ciężko wdrożyć Druida
- Niewiele firm to robi:
  - Imply
  - Datumo
- Wymaga sporej infrastruktury (Cloud znakomicie obniża koszta)
- Jest używany przez coraz więcej firm

Dziękuję!

Q & A

piotr.guzik@datumo.pl