

Speaker Bio





Teradata

- Over 12 years of experience building Data Warehouse Solutions.
- 9+ years of experience working with Teradata based Warehouse Applications

2016

- Engineering Foundational Big Data Applications for PayPal's Data Lake
- Apache Spark Based Ingestion Framework, Data Quality Framework, Dedup & Snapshot processes for Transactional Datasets.

2017

- · Big Data Platform Engineering: to build PayPal's Gimel Data Platform
- Gimel powers Data Lake vision & many other Big Data Engineering Use-cases in PayPal Inc.



Agenda



- PayPal at Scale
 - Business
 - Technology
- PayPal Datalake

- Compute
 - Challenges
 - Solution

<u>PayPal Gimel</u> - Bigger Picture

THE EDGE OF NEXT

- Data Access
 - Challenges
 - Solution

PayPal Gimel – Next Steps

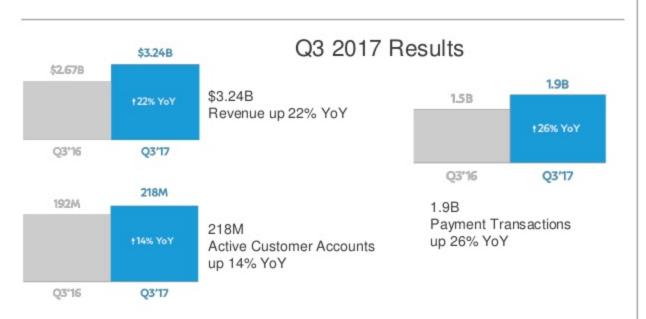
PayPal at Scale

Business & Technology

PayPal Scale | Business



- One of the world's largest internet payment companies
- PayPal platform includes Braintree, Venmo, Paydiant, PP Credit and Xoom



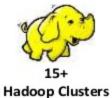
2016 Full Year Results





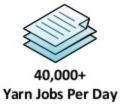
PayPal Scale | Technology







95+ PB Data





6+ compute Supported - Hive,Pig, MR, Spark, Beam, Presto Others - Flink



PayPal - Datalake

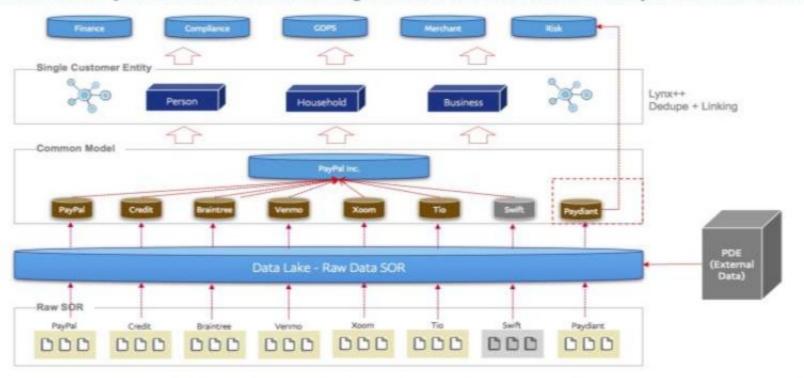
Bigger Picture Complexities

Data Lake | Bigger Picture



Cross Property Data Lake - Conceptual View

Single source for all of PayPal and its subsidiaries data at the most granular level to serve the various needs across PayPal as well as the subsidiaries.





Data Lake | Recap | Complexities



Cross Property Integration

- PayPal, Braintree, Venmo, Xoom, Credit, Tio, Paydient, More to Come in future
- Each Property Unique set of infrastructure & Storages
- Integration & Data Pipelines involves Multiple Storages such as Oracle, MySQL, Kafka, Cloud Bases systems.

Compute / Processing Capabilities

- Multiple Clusters Each has a purpose
- Large Scale / Complex Data Systems
- · Variety of Use Cases Stream, Batch, Interactive, ML
- Multiple Compute Options Hive, Spark, Flink, Beam, Presto, Map-Reduce
- Data & Operational Security!



Data Lake | Recap | Complexities



Data Engineering

- Multiple Storages Cloud, No-SQL, RDBMS, Messaging, In-Memory
- 100's of APIs to access different storages SQL & Programmatic (Python, Scala, Java...)
- Complexity Many-fold if one considers different compute engines spark, hive, flink
- Time to develop and Time to Market goes through "learning curves"

Data Catalog

- Puzzle
- Awareness in what form & where data is available?
- "Storage-specific" dataset creation
- Dataset onboarding various ways
- Data assets Lacking "Centralized" monitoring capabilities



2016

Beginning of the Data Lake Services

Data Lake Services |2016





Simple, Governed, Unified.

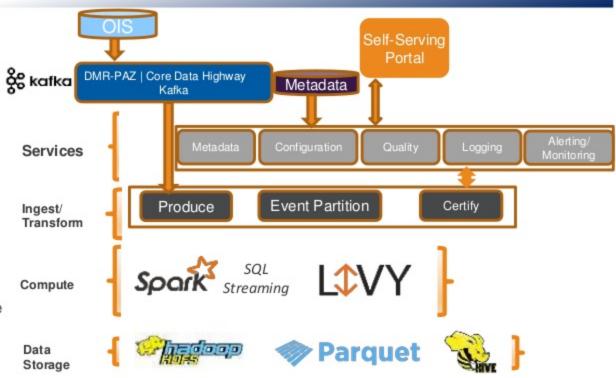
Quick Recap - Architecture

Key Features

- Configuration/Metadata driven apps
- Spark based framework
- Self-Service Portal
- Data Quality Monitoring
- Metrics Collection
- Operational Dashboards

Benefits

- Scalability
- User Experience
- High Speed Data Ingestion
- Converged Data Sources (PayPal, Braintree etc.)
- Secure and Governed Data





Data Lake Services |2016 | Recap Challenges



Data Governance

- Security
- Metadata
- Data Quality & Metrics

Use cases

- Rapid growth rate of new business use cases
- Onboarding

Data Lineage

Most common Challenge!

Data Models & Best Practices

- Data Layout
- Data Models

Data across Property - Integration

 PayPal, Credit, Braintree, Venmo, Paydiant, TIO, Xoom

Technology & Skillset

- Emerging/New technologies. Example Spark, Flink, Beam, Kylo, ...
- Talent & Learning Curve

Best Practices & Common Frameworks

- Logging
- Monitoring
- Alerting
- Auditing
- Operational Dashboards

Administration

- Hosts Management
- Software Management

Users Experience

- Data Engineers
- Analysts
- Scientists



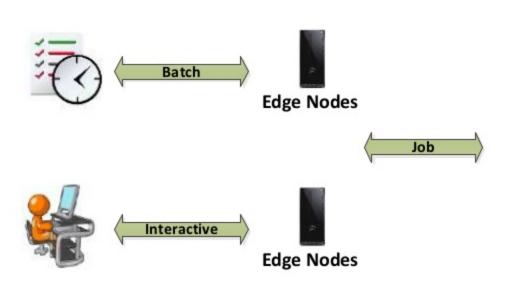
Challenges Faced with...

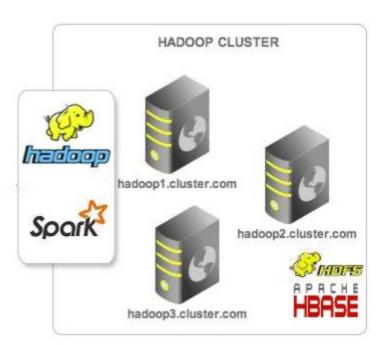
Compute

Spark on Yarn | Typically



Deployment

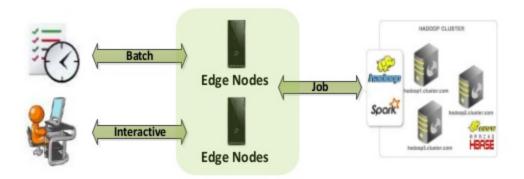




Data Lake | Challenges | Administrators



Administrators



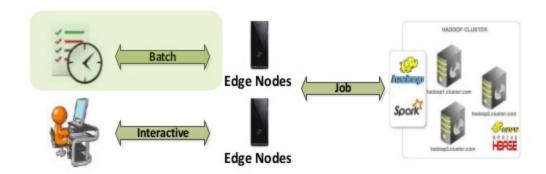
- Need extensive support and maintenance for CLI
- Need to deploy entire stack of software
- Need to sync configurations across systems
- Need extensive testing of jobs before any upgrade



Data Lake | Challenges | Developers



Developers



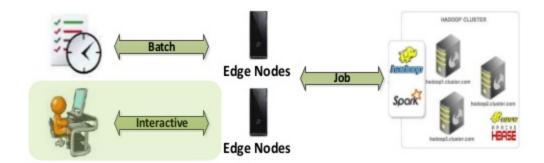
- No REST-friendly API
- No low-latency/sub-seconds execution
- No cache sharing across jobs
- No modularity and easy-restartability



Data Lake | Challenges | Interactive Usecases



Analysts/Scientists



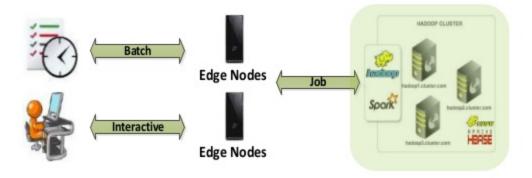
- No easy way of interactive applications
- No multi-tenancy support and private workspace
- No direct spark sql execution
- No Kerberos integration



Data Lake | Challenges | Ops & Security



Operations/Security



- Different ways of jobs execution and coding standards
- No uniform logging, monitoring and alerting
- Limited audit and control
- No statement level history or metrics

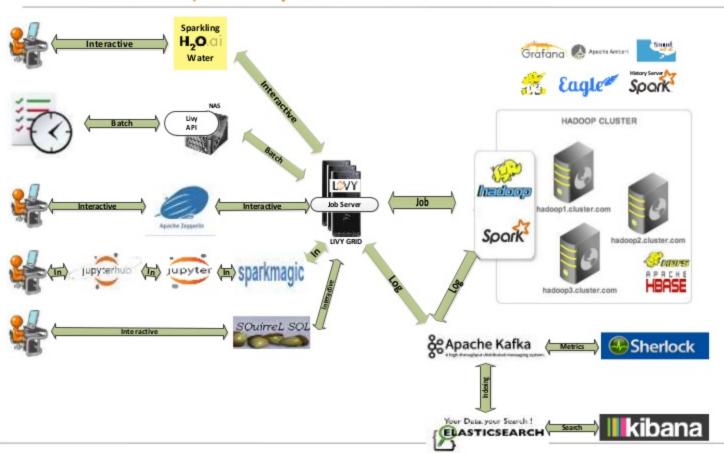


Solution!

Gimel Compute Platform

Data Lake | Compute Platform





Administrators

- ✓ Less maintenance on CLI
- ✓ Deploy software stack only on Job Server
- Configurations at one place
- Easy platform/software upgrade

Developers

- ✓ REST-friendly and Docker-friendly
- Low-latency/sub-seconds execution
- ✓ Sharing cache across jobs
- ✓ Modularity and easy restartability

Analysts/Scientists

- User friendly interactive applications
- Multi-tenancy and Private workspace
- ✓ Direct spark sql execution
- ✓ Kerberos Support

Operations/Security

- Standardized coding and unified execution
- Uniformed logging, monitoring and alerting
- Fine-grained audit
- Complete statement level history and metric



Challenges Faced with...

Data Access

TERADATA PARTNERS

Data Platform | Data Access | Current State

Spark Read From Hbase

```
import org.apache.hadoop.hbase.client.{HBaseAdmin, Result}
import org.apache.hadoop.hbase.{ HBaseConfiguration, HTableDescriptor }
import org.apache.hadoop.hbase.mapreduce.TableInputFormat
import org.apache.hadoop.hbase.io.ImmutableBytesWritable
import org.apache.spark._
object HBaseRead {
 def main(args: Array[String]) {
   val sparkConf = new SparkConf().setAppName("HBaseRead").setMaster("local[2]")
   val sc = new SparkContext(sparkConf)
   val conf = HBaseConfiguration.create()
   val tableName = "table1"
   System.setProperty("user.name", "hdfs")
   System.setProperty("HADOOP_USER_NAME", "hdfs")
   conf.set("hbase.master", "localhost:60000")
   conf.setInt("timeout", 120000)
   conf.set("hbase.zookeeper.quorum", "localhost")
   conf.set("zookeeper.znode.parent", "/hbase-unsecure")
   conf.set(TableInputFormat.INPUT_TABLE, tableName)
   val admin - new HBaseAdmin(conf)
   if (!admin.isTableAvailable(tableName)) {
     val tableDesc = new HTableDescriptor(tableName)
     admin.createTable(tableDesc)
   val hBaseRDD = sc.newAPIHadoopRDD(conf, classOf[TableInputFormat], classOf[ImmutableBytesWritable], classOf[Result])
   println("Number of Records found : " + hBaseRDD.count())
   sc.stop()
```

TERADATA PARTNERS

Data Platform | Data Access | Current State

Spark Read From Elastic Search

```
import org.apache.spark.SparkContext
import org.apache.spark.sql.SQLContext
import org.apache.spark.SparkConf
import org.elasticsearch.spark.sql._
object ElasticSearchRead {
  def main(args: Array (link is external)[String (link is external)]) {
        val conf = new SparkConf().setAppName("ReadFromES")
        conf.set("spark.es.nodes", "10.111.222.333")
        conf.set("es.index.auto.create", "true")
        conf.set("spark.es.nodes.client.only", "true")
        conf.set("es.query", "?q=*")
        conf.set("es.resource", "index/type")
        val sc = new SparkContext(conf)
        val sqlContext = new org.apache.spark.sql.SQLContext(sc)
        val es_df=sqlContext.read.format("org.elasticsearch.spark.sql").load("index/type")
        es_df.registerTempTable("things")
        val filtered_things = sqlContext.sql("select * from things where col = '1000'")
```

Data Platform | Data Access | Current State

Spark Read From Hbase

```
Name of the agencies Andrew Money of Lord (Managadate), Result)
Impart org. speaks hadosp. Name. ( Messtarfiguration, #fabladescriptor )
Import org. specks hadrog Mono supreduce Table Input Formal
Comment only appeals basing Money to Sepurate of physical Chables
Depart org. seaths, spark ....
skinct History (
  ind main(argo: Mrray(String[) {
    unl tearfiled = non Sparkford(), ortRayName("WassRoad").tetMester("local(2)")
    usi us = new igark(artest(spark(sef))
usi usef = Hisselanfiguration.orests()
     System set/freperty/("corr.name", "hefts")
     System setProperty("18050F_SSER_BOR", "hefy")
    comf.set("Mone.mester", "Incolless::0000")
comf.set("timeout", 120001)
    coef.set("Mose zoekesper secrum", "local host")
coef.set("zoekesper zeck server", "/Mess-unsecure")
     coef.com/YubinDeputformon.DMPUT_FMBLE, tobleNome)
     vol. advis - rase MassAdvis (conf)
     of (habite infablished lable(tablebase)) (
       nel tableCone - new WishleConstrictor(tableCone)
       admin.organis/labile(tableSess)
     uni Miccoliii - sc.maaffikoloopiiiiiCcorf. cloopiifTable[aputformot], classif[]amutable[ates#ritable], classif[Disaulti])
     printing Number of Records Found : " + HeaseROR:count())
    Acceptant()
```

Spark Read From AeroSpike

```
legant are marks upon.
Separt you serceptive spark out --
import you serospile climat Aeraspile(line)
impart con acrospiles client file
Separation, appropriate climat Nov.
import non-acrespiles climes. Value
union birequiledned (
 set spinlarger hereplikining[] [
   on appricant a rem Sporkfort() anthonous "Roundoor") anthonia's "cort(23")
   call as - one SpartCartest(spartCart)
   rell aglicement - new Salicantentical)
    nall thingsEF - sqlCarbest_read.
     flammetic?com pre-sourcher sprore and "I
     options 'on may be insuffrant', "121 ft b. 1"5.
     option("sermpthe part", "1988")
     option, "service in revenues,", remission?
      agricus, "sweep he are", "relf-east").
    change of the family of the party
    us). flinered things - salfartant.iglf orlink * from blings attorn one - 18").
    printled water of Boores found : " a Pilitare/Prings.court())
   DWW.
```

Spark Read From Elastic Search

```
Import org.apoche.apork.SparkCovtext
Import org. spoche. spork. sql. SQLContext
Import ang. apoche, spork, SparkCanf.
import org.elesticsearch.spark.sel ...
object [losticSearchRead [
 def main(args: Arroy (link is external)(String (link is external))) {
       vol coef - new SparisCoef().setAppNome("Read"roof5")
       conf.set('spork.ss.nodes', '18.111.222.333')
       conf. astC'ss. index.outs.creots", "true")
       coef.set("spork.es.nodes.cliest.sely", "true")
       conf.set("es.query", "7q-4")
       conf.set("es.resource", "index/type")
       vol sc - new SparkContextCoorf)
       vol sqlContext - new erg.spocke.spork.sql.SQLContext(sc)
       vol es_df-sqlCartest.read.forest("org.elasticsearch.spork.sql").load("index/type")
       es_df.registerTempTobleC"tWings")
       vol filtered_things - sqlContext.sqlC'select * from things where cal - "1888""]
```

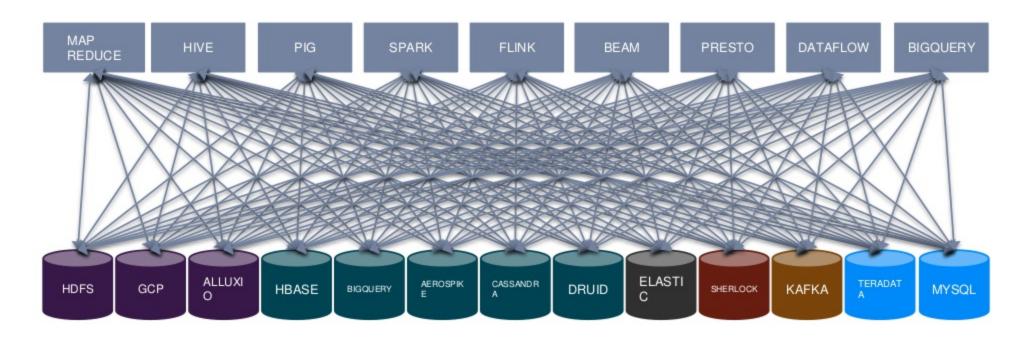
Spark Read From Druid

```
teach on scale teach.
Apart Southfreel )
  det saintarge: Breathtringth 1
          oil markfard a new Specificant's artifactions "Specificant's artifactors" constitution
          yel as a sim Spartification (spartifical)
          rel undistant o new ballarismit(sel-
         applications of the contract that price indicates in particular
                 UKBS org. sport lineders should
                 OFTERS Commissional Sevent and restrictions before
                 timelitiment and all are "1, shi solder".
                 stratisfichensorum "hook"
                 About Minery "Transactions b"
                 siresiffeet "$182",
                 reliablement (" " . "Lauret to" : "san, Lauret to", "an, and lary" : "san, or, out total " ) . . . .
               | Continue 
                 Cody Same Soll Same Special Section (Special Soll)
          out Alberta Things is spillers and a spillers and the form and the destination (special contraction).
         printle Charge of Secretar Stand 1 To Silliers (Stongs, count (S))
          so sing?
```



Data Access | Bigger Picture | Today







Data Platform | Today



Data Application Lifecycle - Current State

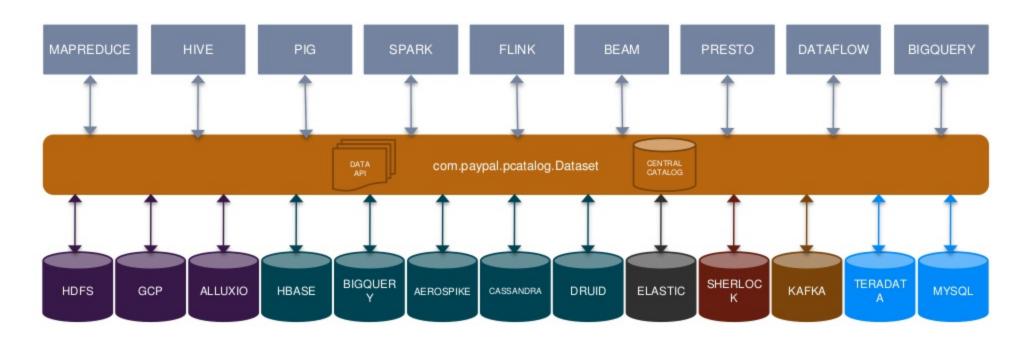


Solution!

Gimel Data Platform

Data Platform | Data Access | With Data API







Data Platform | Objective



Data API

- Unified Data Access API
- Access from any Compute Engine
- Access any Storage

Data Catalog

- Global Catalog Explore at one place
- Onboard at one place
- Dataset Management- Track/Approve Onboard Request
- Dataset Stats



Data Platform | Data Access | Data API



Spark Read From Hbase

```
Import org. speaks, hadeny Abens, chiest, $40cceldrin, Brackt)
import org speaks hodesp Abous ( Hissafonfiguration, Highletbeartplor )
taport org.specks; hodeop. Noise. to . Danutobliefly testiritable
Import org.specks.speck...
object Househood C
  out melecange: Array(String)) 4
wol spark(out o man Spark(out)) satAppHane("Microfrod") setMester("Inca)[2]")
     ust canf - MisseConfiguration.presset)
     upl tabletone - "bobles"
     System ant Property Change, none", "helfu"):
     System art Property ("NABOOF, DOCK, MAKE", "Nafe")
    conf. netf"those maker", "local hoar: 60000")
    mont.set('bhose.conbesper.queres', 'localbest')
     conf. act(Table logor) formet. (MPLF_TMILE, toble fame)
     unit mining a new Whomakini of currier
     of (Ladetin, tyPathLakestLattis(LobisSane)) |
      vel toblefenc - new Wloblefenoriptor(toblefene)
      admin. crasts lob/isCtob/isBeac)
     uni Microfile - sc.nes#Filedosp#E0(comf. class#fThd/sEnguriumet), closs#fIlesutobleBytes#ritable], class#f
    Charle-16
```

Spark Read From AeroSpike

```
import one specks speck.
Layort you serception years out -
least for deringthe citiest Arrangibellions
immert com servapiles allert for
Seport ine aimpelio. client Key
impart our arrespitor client. Vylus
stictact Asymptophise C.
 set seinlarger Army (Nirring)) 1
    on) sporkfoof a rear Sporkfoot() antiquitiess("Househoof") antifoctor("house(22")
   not as a one (gard(school(spork(self))
    out nettrement a new Solf-sensettuch
    call things!! - tellimined read.
      Farters Cross person by appropriately
      particular formation interest to 1927 A.A. 173
      option "or expite part", "1998").
      particle "mercentile remaining", passes appell
      agenced "overspite on", "roll-teat").
    strongly employed baseful of "things").
    sal fillteredThings - salCartout.igl('belook + from things others one - 88').
    printle("Name" of Asserts Food : " + FilteredPrings.count())
   Dec. NUMBER
```

Spark Read From Elastic Search

```
Import org.apoche.spark.SparkContext
Import org.spochs.spark.sql.5QLContext
import org.spoche.spork.SporkCost
import org.elosticsearch.spark.sol.,
object ClosticSearchBook (
 def main(args: Array (link is external)[String (link is external)]) (
        vol conf = mem SporkConf().setAgoNome("ReadFrontS")
       conf.set("spork.es.wodes", "18.111.222.333")
       conf.set("es.index.suto.cresto", "true")
       conf.set("spark.es.wodes.client.ordy", "true")
       conf.set("es.query", "?g-4")
        cord.set("es.resource", "index/type")
       vol at - new SparkCantestCoarf)
        val aglContext - new org.opeche.aperk.agl.5QlContext(ac)
        es_df.registerTempTable("things")
        vol filtered_things = sqlCostest.sql("select * from things where cal = "SBBP"")
```

Spark Read From Druid

```
based and stocky speck.
officer Bouldfreet 1
            per sanctarge, arrestitiving!
                                or) workford a non-law-blood() antiquipme("Baseless") antiferror("seet(2)")
                                val no c con Spartifactions (spartiface)
                                ral addisons - no fallerand(sal
                                   spillersmith "Thirt median like order malipelanticaption
                                                UKBB over report Constant street.
                                            CPTINE Committee from "anders confine floor" reflect
                                            time bismed and all are "I, also wishes".
                                            strate offendament and "base"
                                            sire officer. "Invalidant"
                                                single Plant "$182"
                                            Schmidtsgebeg (* 15. pages) og * 1. hand generop*, "pages hat hat the seagen product by the face of the seagen product by the sea
                                                      "bold" : "s.mee"; "orb!" : "s.manageon"; "tges" : "e-E");
                                            Plant 2" - School Could' - School Strate - Special Strate
                                   rel Pilleredfiles I selferent applied to the entrainmentalist ( selfential to the entrainmental selfential s
                                printing better of Security Second 1.7. a Silliared Storage country;
                                so allow?
```

```
vol es_df-sqlComtext.read.formst("org.elosticsearch.spark.sql").lead("index/type")
```

With Data API

```
val df = Dataset.Read("dataset_from_hbase")
val df = Dataset.Read("dataset_from_elastic")
val df = Dataset.Read("dataset_from_aerospike")
val df = Dataset.Read("dataset_from_druid")
val df = Dataset.Read("dataset_from_hdfs")
val df = Dataset.Read("dataset_from_kafka")
val df = Dataset.Read("dataset_from_sherlock")
val df = Dataset.Read("dataset_from_cassandra")
val df = Dataset.Read("dataset_from_solr")
val df = Dataset.Read("dataset_from_mysql")
val df = Dataset.Read("dataset_from_teradata")
val df = Dataset.Read("dataset_from_oracle")
val df = Dataset.Read("dataset_from_bigguery")
val df = Dataset.Read("dataset_from_bigtable")
val df = Dataset.Read("dataset_from_alluxio")
```





Data Platform | Data Access | Gimel SQL



Spark Read From Hbase

```
Import org. speaks, hadeap /bhass.s/isst. (Massidein, Messit)
Import org. speaks, hadeap /bhass.( | MassionFiguration, | MakinDassriptor ]
taport org.specks; hodeop. Noise. to . Danutobliefly testiritable
Import org.specks.speck...
object Househood C
  def me(scarge: Array(String)) {
    set spork(orf) = min Spork(orf().setAppHane('Mosefund').netMester('local(Z)')
     upl ac - new Sportclantzxt(apprison())
upl canf - MisseConfiguration.create()
     upl tabletone - "bobles"
     System terProperty("ster.ness", "bdfs")
System terProperty("st8000_utilit.sem;", "bdfs")
     conf. netf"those maker", "local hoar: 60000")
     mont.set('bhose.conbesper.queres', 'localbest')
     conf. act(Table logor) formet. (MPLF_TMILE, toble fame)
     unit mining a new Whomakini of currier
     of (Ladetin, tyPathLakestLattis(LobisSane)) |
        vel toblefenc - new Wloblefenoriptor(toblefene)
        admin. crusts/lob/is/tob/isfless)
     uni Microfile - sc.nes#Filedosp#E0(comf. class#fThd/sEnguriumet), closs#fIlesutobleBytes#ritable], class#f
     Charle-16
```

Spark Read From AeroSpike

```
import one specks speck.
Layort you serception years out -
least for deringthe citiest Arrangibellions
immert com.sertopile.c3test.60-
Seport ine aimpelio. client Key
impart our arrespitor client. Vylus
sitted hirospinshad (
 set seinlarger Army (Nirring)) 1
    on) sporkfoof a rear Sporkfoot() antiquitiess("Househoof") antifoctor("house(22")
   not as a one (gard(school(spork(self))
    out nettrement a new Solf-sensettuch
    call things!! - tellimined read.
      Farmer, Time person by appropriately
      particular formation interest to 1927 A.A. 173
      option "or expite part", "1998").
      Concessors of Constitute of Springer Sections
      agencies, "correspond and", "red and "b.
    strongly register function of "things").
    sal fillteredThings - salCartout.igl('belook + from things others one - 88').
    printle("Name" of Asserts Food : " + FilteredPrings.count())
   Distributed To
```

Spark Read From Elastic Search

```
Import org.apoche.spark.SparkContext
Import org.spochs.spark.sql.5QLContext
import org.spoche.spork.SporkCost
import org.elosticsearch.spark.sol.,
object ClosticSearchBook (
 def main(args: Array (link is external)[String (link is external)]) (
       vol conf = mem SporkConf().setAppHome("Brodfront5")
       conf.set("spork.es.wodes", "18.111.222.333")
       conf.set("cs.index.sutc.create", "true")
       conf.set["spark.es.modes.clitent.only", "true"]
       conf.set('es.query', "?g-4")
       cord.set("es.resource", "index/type")
       vol at - new SparkCantestCoarf)
       val aglicontext - new org.opeche.aperk.agl.5@iContext(ac)
        vol es_df-sqlComtext.read.formst("org.elosticsearch.spark.sql").lead("index/type")
        es_df.registerTempTable("things")
        vol filtered_things = sqlCostest.sql("select * from things where cal = "SBBP"")
```

Spark Read From Druid

```
based and stocky speck.
officer Bouldfreet 1
         per sanctarge, arrestitiving!
                       or) workford a non-law-blood() and solven ("Bandered") and former ("cont(2)")
                         yel no c non Spartifactions (spartiface)
                       rall and financian is now facility-tomb (and
                       satheranals." There's merchant that a security destroys the
                                   UKBS one sport lineders street
                                CPTING Commission from "aniest confine floor" refine"
                                time bismed and all are "I, also wishes".
                                stratiglishmanarum "bash".
                                sire direct "localitati".
                                   single Plant "$182"
                                orthodography (* 15. pared by * 1 fam.), generally (* 15. pared by * 1 fam.), property (* 15. pared by
                                       "bold" : "s.mee"; "orb!" : "s.manageon"; "tges" : "e-E");
                                Plants' - N. James' - No. 12" - N. James and J. Nagari' - No. 17",
Plants' - No. James' - No. 12" - N. James and J. Nagari' - No. 17",
Plants' - No. James and J. No. 12" - No. James and J. Nagari' - No. 17", S.
                         rel Pilleredfiles I selferent applied to the entrainmentalist ( selfential to the entrainmental selfential s
                       printing better of Security Second 1.7. a Silliared Storage country;
                       so allow?
```



With Gimel SQL

SELECT * FROM dataset_from_hbase

SELECT * FROM dataset_from_elastic

SELECT * FROM dataset_from_aerospike

SELECT * FROM dataset_from_druid

SELECT * FROM dataset_from_hdfs

SELECT * FROM dataset_from_kafka

SELECT * FROM dataset_from_sherlock

SELECT * FROM dataset_from_cassandra

SELECT * FROM dataset_from_solr

SELECT * FROM dataset_from_mysql

SELECT * FROM dataset_from_teradata

SELECT * FROM dataset_from_oracle

SELECT * FROM dataset_from_bigguery

SELECT * FROM dataset_from_bigtable

SELECT * FROM dataset_from_alluxio



Data Platform | Express everything as Gimel SQL



All Kind of Hive SQLs Supported!

Batch SQL



Streaming SQL



%%gimel-batch
insert into pcatalog.hdfs_dataset
partition(yyyy,mm,dd,hh,mi)
select kafka_ds.*,gimel_load_id
,substr(commit_timestamp,1,4) as yyyy
,substr(commit_timestamp,6,2) as mm
,substr(commit_timestamp,9,2) as dd
,substr(commit_timestamp,12,2) as hh
,case when cast(substr(commit_timestamp,15,2) as INT) <= 30 then "00" else
"30" end as mi
from pcatalog.kafka_dataset kafka_ds;

%%gimel-stream
insert into pcatalog.hdfs_dataset
partition(yyyy,mm,dd,hh,mi)
select kafka_ds.*,gimel_load_id
,substr(commit_timestamp,1,4) as yyyy
,substr(commit_timestamp,6,2) as mm
,substr(commit_timestamp,9,2) as dd
,substr(commit_timestamp,12,2) as hh
,case when cast(substr(commit_timestamp,15,2) as INT) <= 30 then "00" else
"30" end as mi
from pcatalog.kafka_dataset kafka_ds;</pre>



Data Platform | Data Application Lifecycle | With Data API



()	Onboarding Big Data Apps	Leam	>	Code	\geq	Optimize	>	Build	>	Deploy	>	Run	
(1)	Compute Engine Changed										\geq	Run	
(L)	Compute Version Upgraded										\geq	Run	
(1)	Storage API Changed											Run	
(1)	Storage Connector Upgraded										\geq	Run	
(1)	Storage Hosts Migrated										\geq	Run	
()	Storage Changed										\geq	Run	
(1)												Run	

Data Platform | Data API | Unlocking Additional Benefits



Explorer

- · View sample data
- · Add capacity reservations
- Approval and Tracking

Discovery

 Auto-discovery of datasets and importing into PCatalog

Dashboard

- Operation dashboard: Metrics, statistics, trends, refresh status
- Admin dashboard: Capacity/growth/space statistics and action

Alert

- User alerts to notify user datasets refresh delay and quality issue
- Admin alerts to notify datasets capacity and access violation

Query/BI Integration

Integrate catalog with Jupyter, Zeppelin and query any dataset

Out-of-the-box Benefits

- Connector dependencies maintained by BDPE
- Change component version at one place. E.g. Hbase version
- Easy cluster upgrade
- Abstract storage from user. Just Prod, QA and Dev
- Add new compute engine to stack easily
- · Add new storage system to stack easily
- · Time to Develop
- Time to Market



Presenting

PayPal Gimel

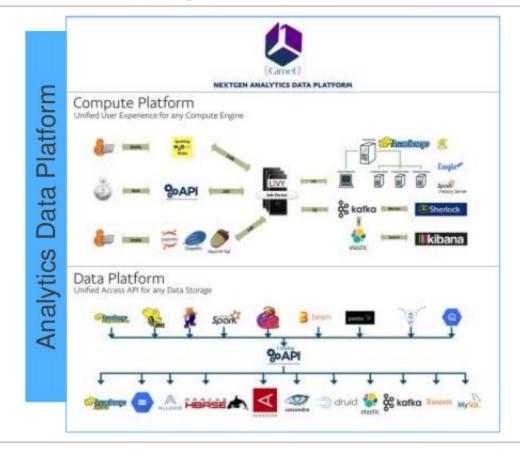
Data Platform

+

Compute Platform

TERADATA PARTNERS CONFERENCE

PayPal GIMEL





Data Platform: Plan



2017 =======

2018 ========

Open Source - Data API

















POC

- Analysis Evaluation
- Show & Tell

- Elastic Search
- ✓ Kafka
- ✓ Aerospike
- ✓ Sherlock

Catalog:

- ✓ Catalog API

Phase-1

Compute Engine:

- Spark
- Hive
- ✓ Presto

Storage:

- ✓ TERADATA
- ✓ HDFS
- √ Hhase

- ✓ Web UI
- √ 1Time Onboard

Phase-2

Compute Engine:

- Spark
- Hive
- ✓ Presto

Storage:

- ✓ Druid
- ✓ Cassandra

Catalog:

- ✓ Web UI
- ✓ Auto-Onboard
- ✓ Catalog API

Phase-3

Compute Engine:

- ✓ Pig/MR Beam
- ✓ Flink

Storage:

- ✓ Alluxio
- ✓ Oracle
- √ Mysql

Catalog:

- ✓ Metrics/Trend/Status
- ✓ Discovery/Alerts
- Onboard Approval Jira Integration
- Rally Integration
- Storage Integration

Phase-4

Compute Engine: ✓ Big Query

Storage:

- √ GSFS
- ✓ Big Query
- ✓ Big Table

Catalog:

- ✓ Enhancements
- ✓ Improvements
- ✓ Deployment

Phase-5

- ✓ Cont. Onboarding
- ✓ Post-Prod Support
- ✓ Maintenance



Thank You!

Rate This Session # 0348

with the PARTNERS Mobile App

Questions/Comments

Email: dmohanakumarchan@paypal.com

LinkedIn: https://www.linkedin.com/in/deepakmc/

Appendix

References



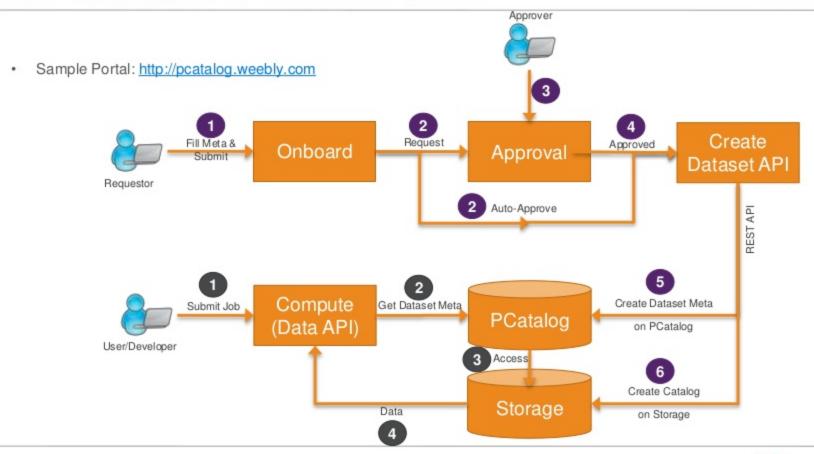
Approver

Google Search Links Used for Logos of open source technologies : https://www.google.com/search?q=big+data+stack+icons&source=lnms&tbm=isch&sa=X&ved=0ahUKEwiwnevLztzWAhVHsFQKHZp7CukQAUICigB&biw=1369&bih=739

PayPal Q3 2017 & 2016 Full Year Results: https://www.paypal.com/us/webapps/mpp/about

Data Platform: Process Flow

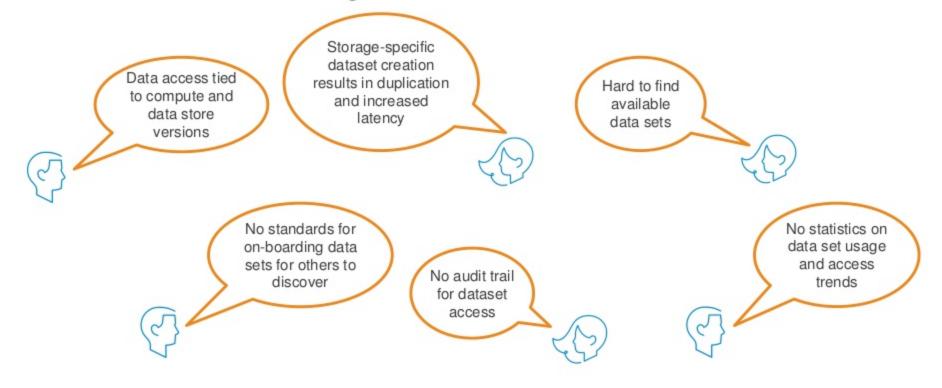




Data Platform | Today



Data Access – Other challenges



Data Platform: Plan Proposal



