Heroku for Apex Developers:
Data Integration & Analysis for Small Teams

Melissa Prcic, Senior Developer

mprcic@stand.org
@MelissaPrcic

Rich Nevin, Director of Technology Development

rnevin@stand.org

STAND for children
LEADERSHIP CENTER

salesforce

Forward-Looking Statements

Statement under the Private Securities Litigation Reform Act of 1995:

This presentation may contain forward-looking statements that involve risks, uncertainties, and assumptions. If any such uncertainties materialize or if any of the assumptions proves incorrect, the results of salesforce.com, inc. could differ materially from the results expressed or implied by the forward-looking statements we make. All statements other than statements of historical fact could be deemed forward-looking, including any projections of product or service availability, subscriber growth, earnings, revenues, or other financial items and any statements regarding strategies or plans of management for future operations, statements of belief, any statements concerning new, planned, or upgraded services or technology developments and customer contracts or use of our services.

The risks and uncertainties referred to above include – but are not limited to – risks associated with developing and delivering new functionality for our service, new products and services, our new business model, our past operating losses, possible fluctuations in our operating results and rate of growth, interruptions or delays in our Web hosting, breach of our security measures, the outcome of any litigation, risks associated with completed and any possible mergers and acquisitions, the immature market in which we operate, our relatively limited operating history, our ability to expand, retain, and motivate our employees and manage our growth, new releases of our service and successful customer deployment, our limited history reselling non-salesforce.com products, and utilization and selling to larger enterprise customers. Further information on potential factors that could affect the financial results of salesforce.com, inc. is included in our annual report on Form 10-K for the most recent fiscal year and in our quarterly report on Form 10-Q for the most recent fiscal quarter. These documents and others containing important disclosures are available on the SEC Filings section of the Investor Information section of our Web site.

Any unreleased services or features referenced in this or other presentations, press releases or public statements are not currently available and may not be delivered on time or at all. Customers who purchase our services should make the purchase decisions based upon features that are currently available. Salesforce.com, inc. assumes no obligation and does not intend to update these forward-looking statements.

> "Ensure all children, regardless of their background, graduate from high school prepared for, and with access to, college or career training."
>
> - Stand for Children's Mission

STAND for children

salesforce.org

salesforce heroku enterprise

Leveraging Salesforce and Heroku to make a difference for ALL kids

Educating & empowering parents and teachers

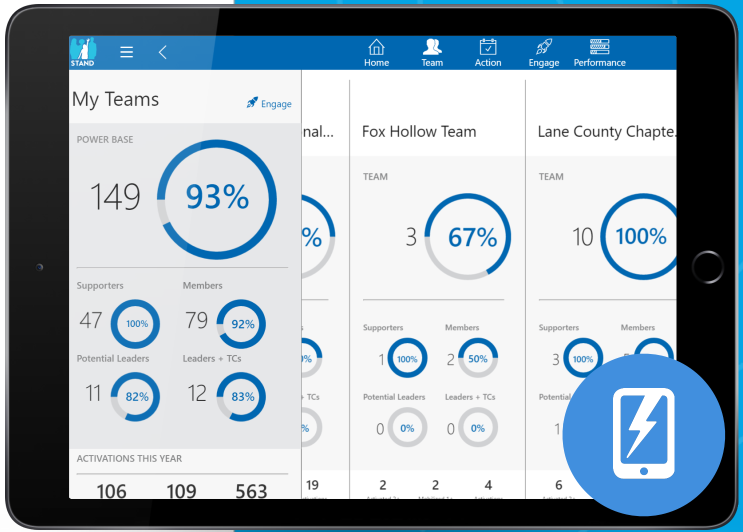
Advocating for effective local policies in 11 states and 18 districts across the country

Stand for Children Leadership Center is a 501 (c)(3)

Why Heroku for Stand?

Problem Statement and Use Case

WE HAVE:	WE NEED:
Transactional data from custom HTML5 app on Salesforce	Who are our rising, engaged volunteers?
Digital, social media and 3rd party data	How can we connect them to advocacy opportunities?



SOLUTION:

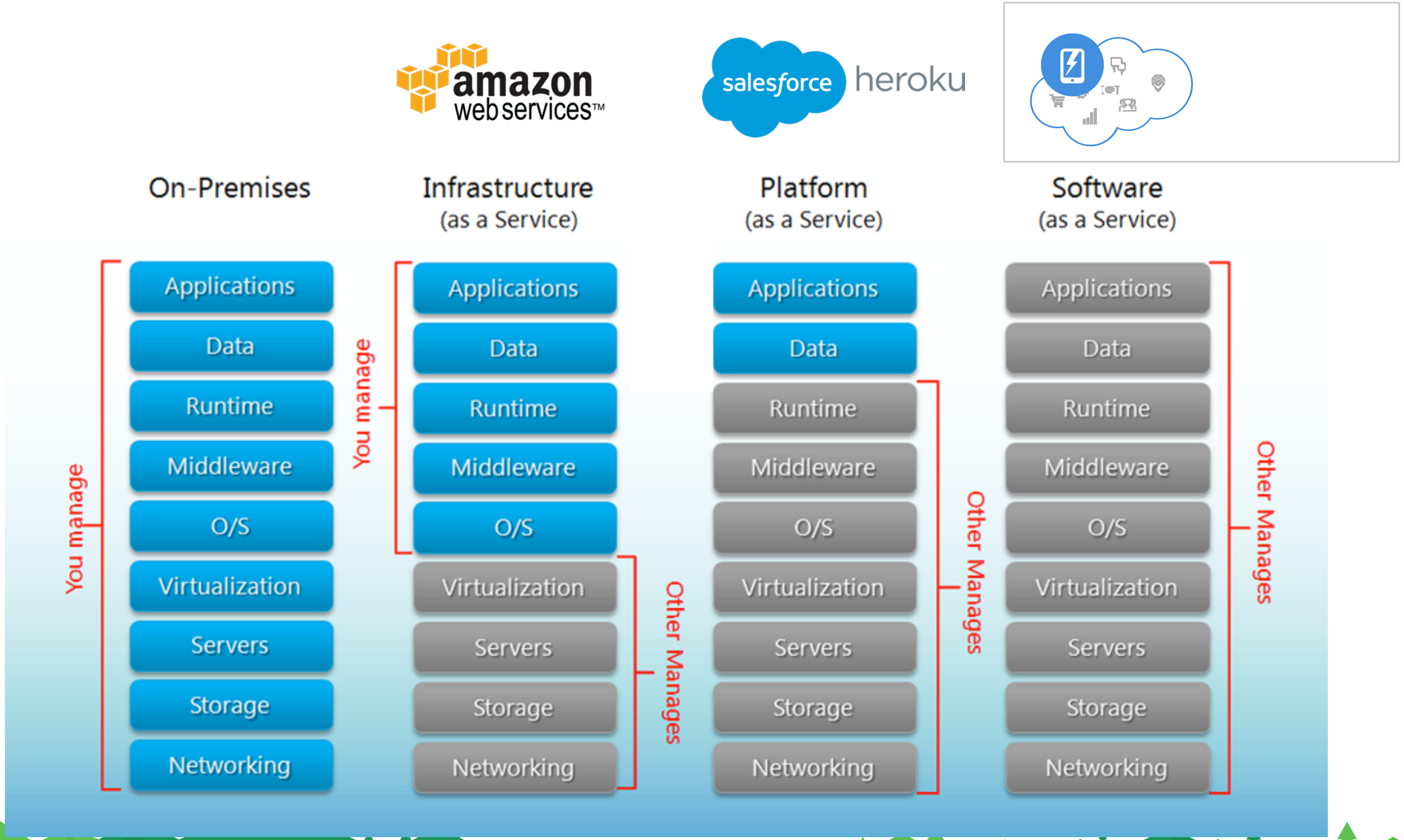
An Heroku app to mash up and analyze data in a PostgreSQL database

Round trip connection to Salesforce for actionable insight in the field

ACTIONABLE INSIGHT

Why Heroku?

And why we couldn’t PaaS this up



On-Premises

You manage: Applications, Data, Runtime, Middleware, O/S, Virtualization, Servers, Storage, Networking

Infrastructure (as a Service)

You manage: Applications, Data, Runtime, Middleware, O/S

Other Manages: Virtualization, Servers, Storage, Networking

Platform (as a Service)

Applications, Data

Other Manages: Runtime, Middleware, O/S, Virtualization, Servers, Storage, Networking

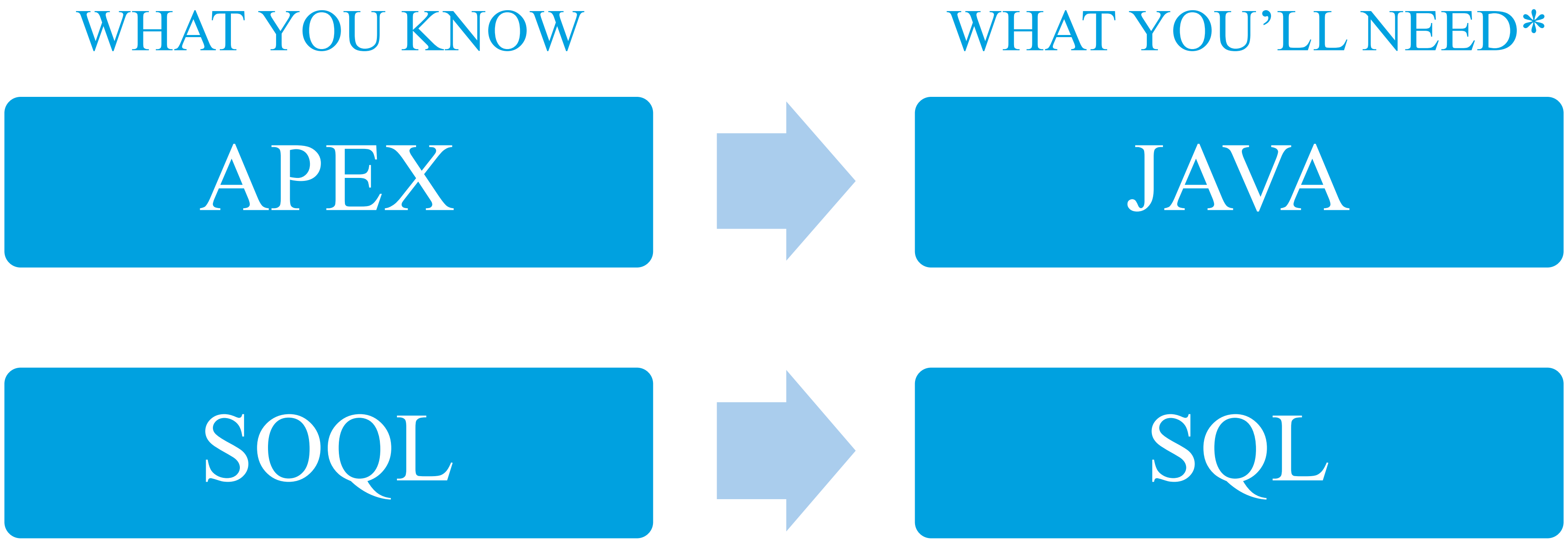
Software (as a Service)

Other Manages: Applications, Data, Runtime, Middleware, O/S, Virtualization, Servers, Storage, Networking

Hiker, meet trail

A guide to help you climb Mt. Heroku
… and a demo to prove you can

Good News – You Have Transferrable Skills!



WHAT YOU KNOW

WHAT YOU'LL NEED*

APEX → JAVA

SOQL → SQL

* Heroku supports any number of development languages, such as Ruby, PHP, Python, and others. Since we're Apex developers, we'll focus on Java.

New Stuff to Learn

- Working with the Command Line

  Heroku Command Line Interface (CLI)

- Local Development

  PgAdmin

- Connecting to Postgres from Java

  & Redis from Java

- Force.com Web Services Connector

  WSDLs!

Use the Partner WSDL for WSC; Like Dynamic Apex it won't depend on a specific data model

Apex to Heroku Learning Path

Commence switchbacks

SETUP ENVIRONMENT

Open FREE Heroku Account

Install Heroku Toolbelt

Complete Local Installs

- Apache Maven (Build Automation)
- Java 8

TAKE WALK-THROUGHS

Heroku: Getting Started with Java

Heroku: Connecting to Databases

Force.com: Intro to the WSC

Salesforce: Streaming API Java Client

Connecting Your Data: Option 1 Using Connect™

Connect is great and not free

CONNECT: THE HEROKU ENTERPRISE PRODUCT

- Fast
- Powerful
- Costs money

salesforce

Connecting Your Data: Option 2 Build Custom

It’s more work but flexible and doesn’t cost nearly as much money

CUSTOM CONNECTION

- Same APIs as Connect
- Inexpensive
- Flexible synch options
  - One-Way or Bi-Directional
  - Scheduled or Streaming

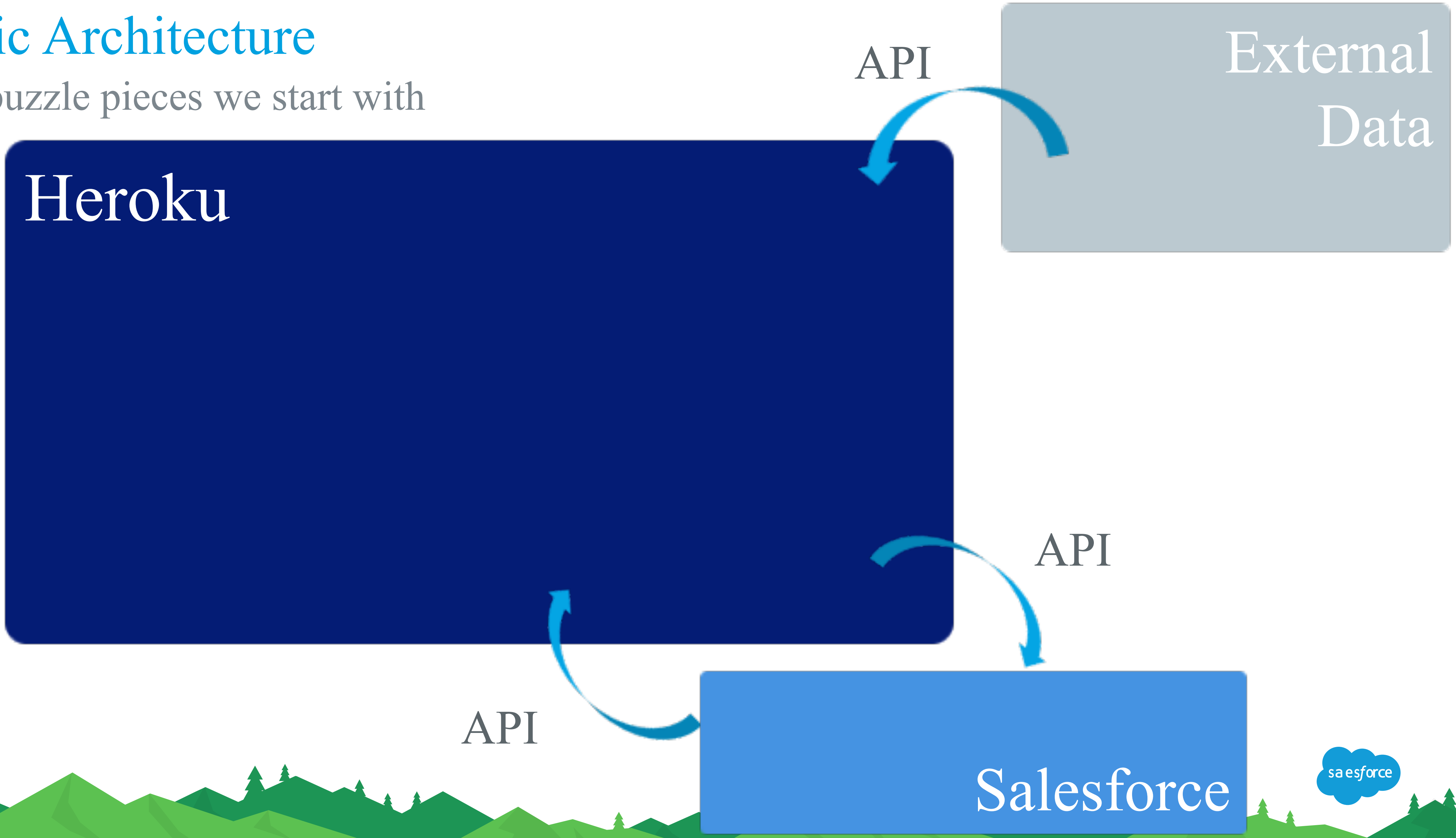
Start with the Heroku Connect Demo version. This will also create Postgres Tables matching your SF tables that you can clone later!

Architecture

Let’s draw some pictures.

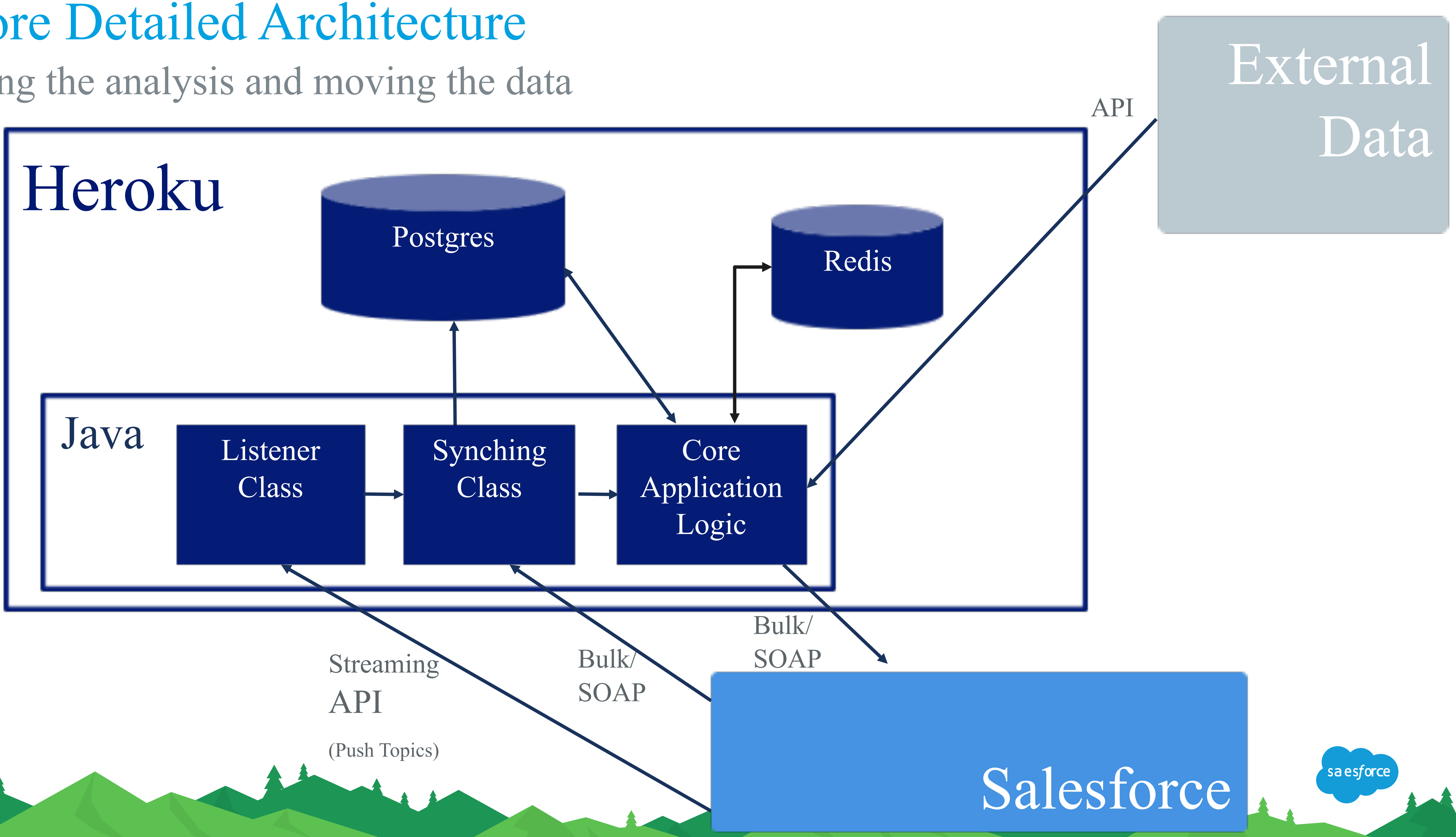
Basic Architecture

The puzzle pieces we start with



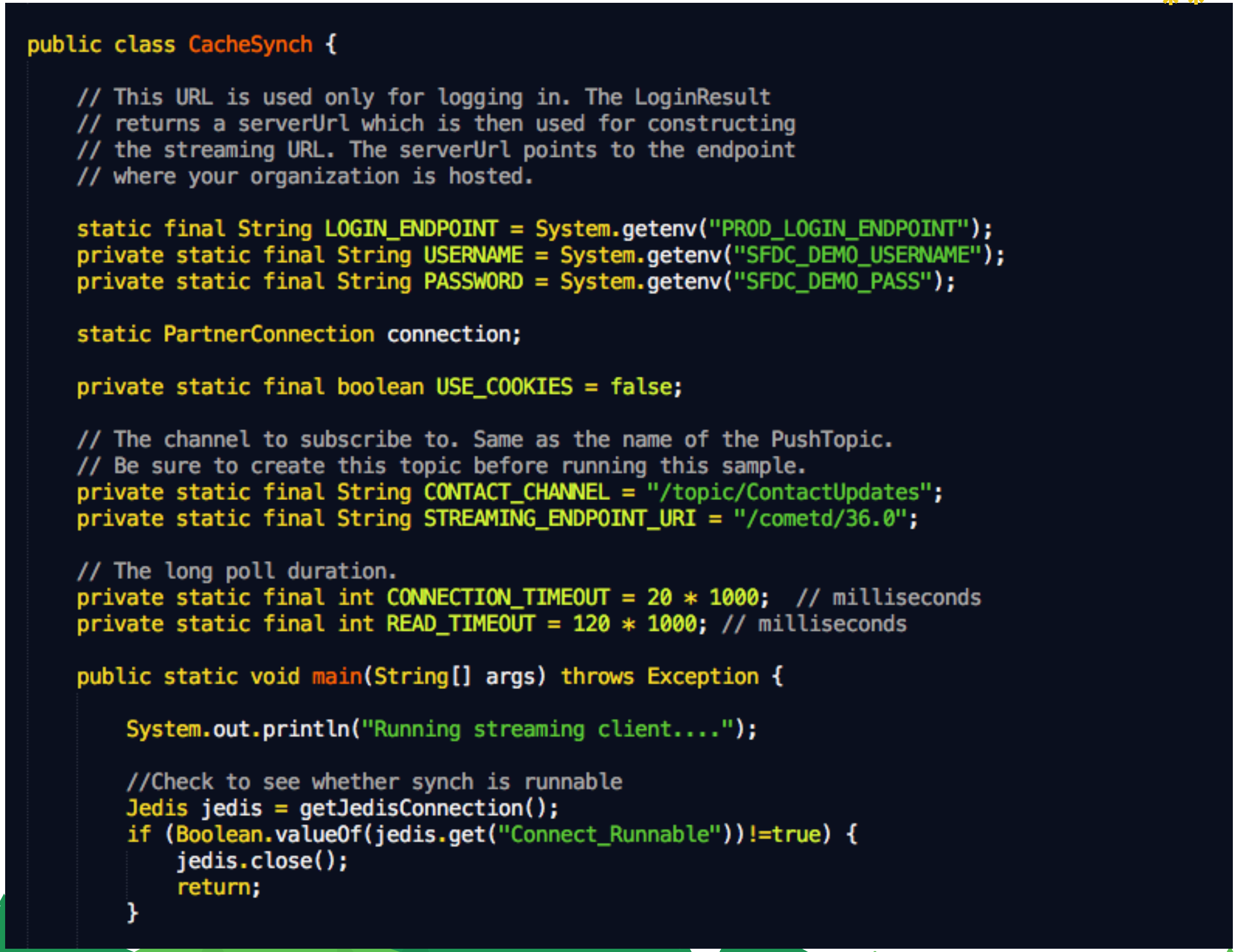
More Detailed Architecture

Doing the analysis and moving the data



Demo!!!

Hogwarts needs a new Sorting Hat

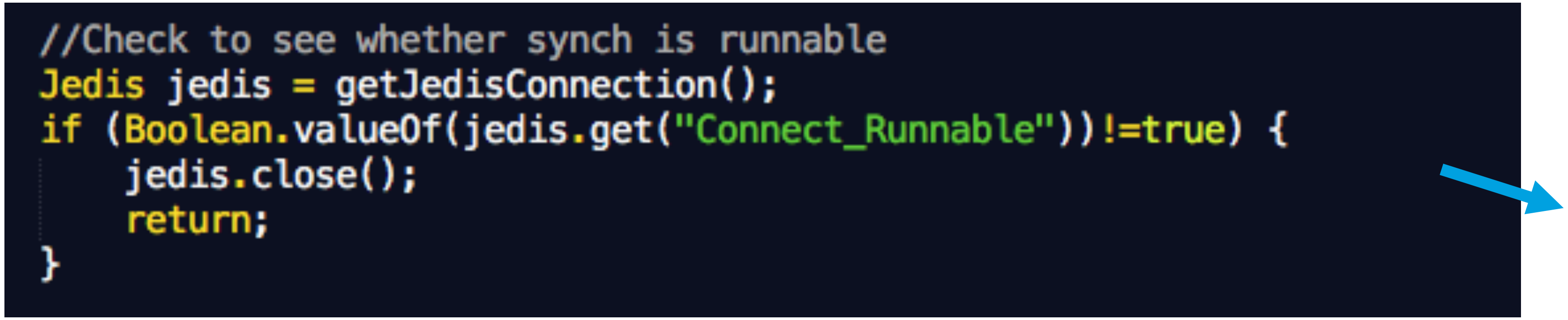


```java
public class CacheSynch {

    // This URL is used only for logging in. The LoginResult
    // returns a serverUrl which is then used for constructing
    // the streaming URL. The serverUrl points to the endpoint
    // where your organization is hosted.

    static final String LOGIN_ENDPOINT = System.getenv("PROD_LOGIN_ENDPOINT");
    private static final String USERNAME = System.getenv("SFDC_DEMO_USERNAME");
    private static final String PASSWORD = System.getenv("SFDC_DEMO_PASS");

    static PartnerConnection connection;

    private static final boolean USE_COOKIES = false;

    // The channel to subscribe to. Same as the name of the PushTopic.
    // Be sure to create this topic before running this sample.
    private static final String CONTACT_CHANNEL = "/topic/ContactUpdates";
    private static final String STREAMING_ENDPOINT_URI = "/cometd/36.0";

    // The long poll duration.
    private static final int CONNECTION_TIMEOUT = 20 * 1000;  // milliseconds
    private static final int READ_TIMEOUT = 120 * 1000; // milliseconds

    public static void main(String[] args) throws Exception {

        System.out.println("Running streaming client....");

        //Check to see whether synch is runnable
        Jedis jedis = getJedisConnection();
        if (Boolean.valueOf(jedis.get("Connect_Runnable"))!=true) {
            jedis.close();
            return;
        }
```

Code Highlights-Redis

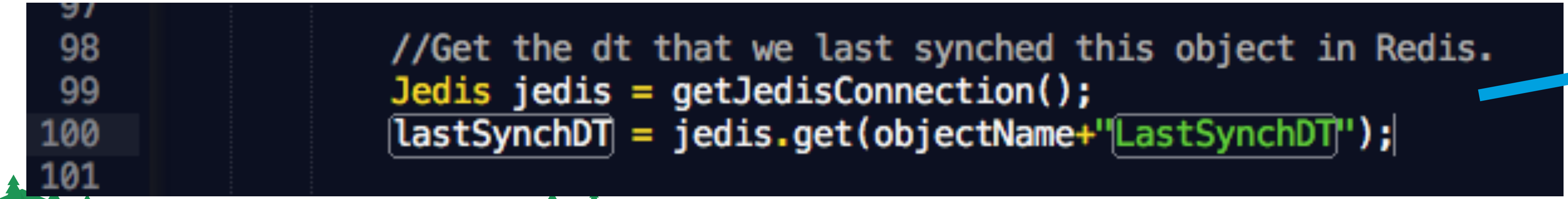
Key/Value Store!

Jedis is a Java Redis Client



```
//Check to see whether synch is runnable
Jedis jedis = getJedisConnection();
if (Boolean.valueOf(jedis.get("Connect_Runnable"))!=true) {
    jedis.close();
    return;
}
```

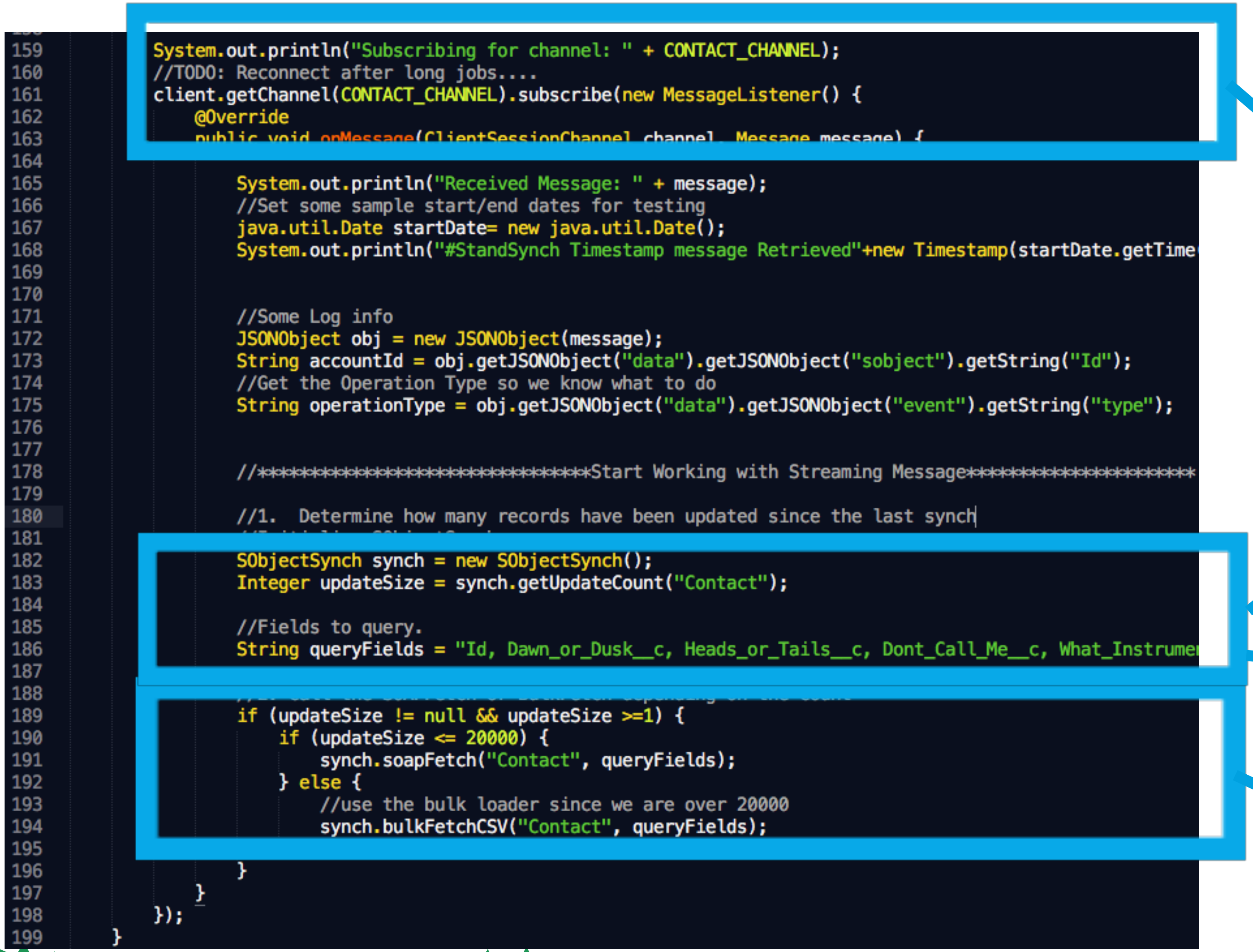
Redis stores app-wide settings, including whether Synch should be run at all



```
        //Get the dt that we last synched this object in Redis.
        Jedis jedis = getJedisConnection();
        lastSynchDT = jedis.get(objectName+"LastSynchDT");
```

Redis stores the DT each object was last synched

Code Highlights-ListenerClass



```
System.out.println("Subscribing for channel: " + CONTACT_CHANNEL);
//TODO: Reconnect after long jobs....
client.getChannel(CONTACT_CHANNEL).subscribe(new MessageListener() {
    @Override
    public void onMessage(ClientSessionChannel channel, Message message) {

        System.out.println("Received Message: " + message);
        //Set some sample start/end dates for testing
        java.util.Date startDate= new java.util.Date();
        System.out.println("#StandSynch Timestamp message Retrieved"+new Timestamp(startDate.getTime


        //Some Log info
        JSONObject obj = new JSONObject(message);
        String accountId = obj.getJSONObject("data").getJSONObject("sobject").getString("Id");
        //Get the Operation Type so we know what to do
        String operationType = obj.getJSONObject("data").getJSONObject("event").getString("type");


        //*******************************Start Working with Streaming Message*********************

        //1.  Determine how many records have been updated since the last synch
        //Initialize SObjectSynch
        SObjectSynch synch = new SObjectSynch();
        Integer updateSize = synch.getUpdateCount("Contact");

        //Fields to query.
        String queryFields = "Id, Dawn_or_Dusk__c, Heads_or_Tails__c, Dont_Call_Me__c, What_Instrumen

        //2. Call the SOAPFetch or BulkFetch depending on the count
        if (updateSize != null && updateSize >=1) {
            if (updateSize <= 20000) {
                synch.soapFetch("Contact", queryFields);
            } else {
                //use the bulk loader since we are over 20000
                synch.bulkFetchCSV("Contact", queryFields);
            }
        }
    }
});
}
```

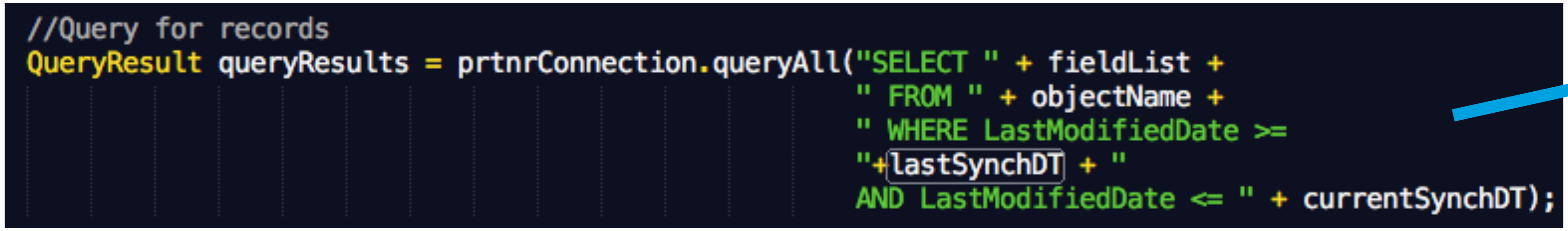
A method for each SObject to be synched

Calls the SObjectSynch class which is dynamic (Sobject agnostic)

Fields defined here

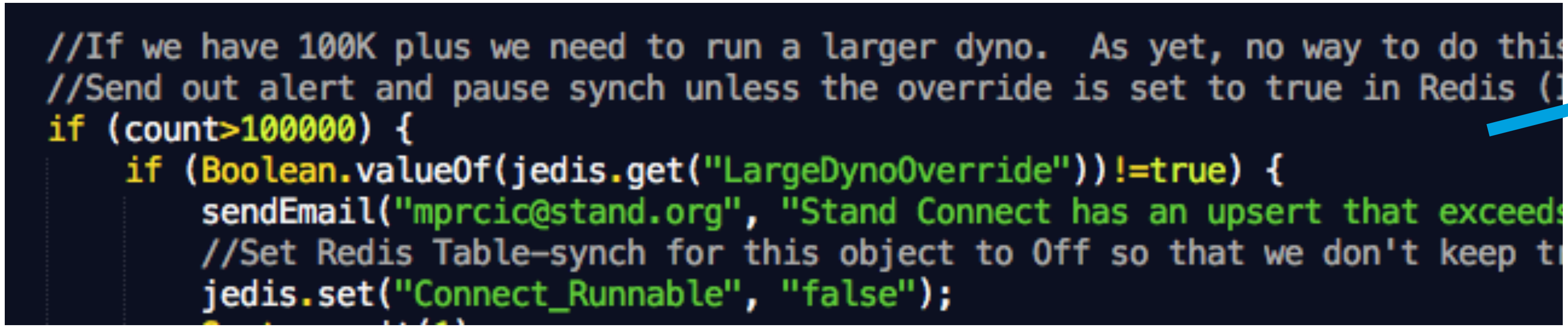
Synch via SOAP if <=20k records, BULK if more

Code Highlights-SObject Synch



```
//Query for records
QueryResult queryResults = prtnrConnection.queryAll("SELECT " + fieldList +
                                                    " FROM " + objectName +
                                                    " WHERE LastModifiedDate >=
                                                    "+lastSynchDT + "
                                                    AND LastModifiedDate <= " + currentSynchDT);
```

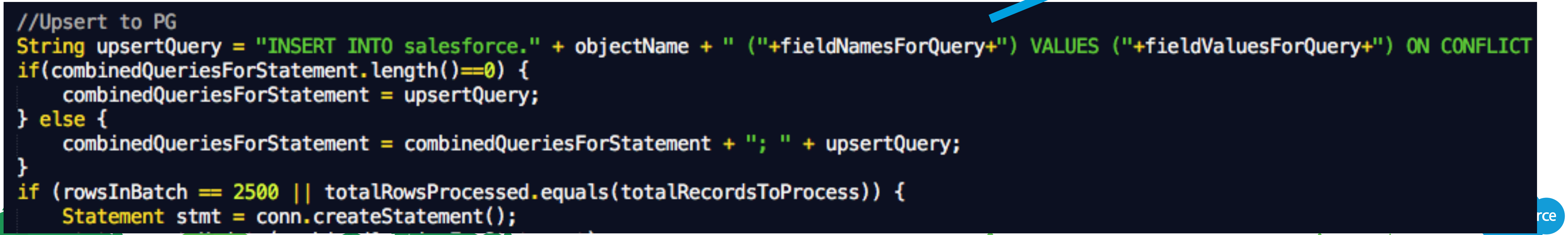
Query for all records updated since the last synch



```
 //If we have 100K plus we need to run a larger dyno.  As yet, no way to do this
 //Send out alert and pause synch unless the override is set to true in Redis (
 if (count>100000) {
     if (Boolean.valueOf(jedis.get("LargeDynoOverride"))!=true) {
         sendEmail("mprcic@stand.org", "Stand Connect has an upsert that exceeds
         //Set Redis Table-synch for this object to Off so that we don't keep t
         jedis.set("Connect_Runnable", "false");
```

Synching 100k+ records need a big dyno. Pause synch & Notify

Batch Upserts to Postgres



```
//Upsert to PG
String upsertQuery = "INSERT INTO salesforce." + objectName + " ("+fieldNamesForQuery+") VALUES ("+fieldValuesForQuery+") ON CONFLICT
if(combinedQueriesForStatement.length()==0) {
    combinedQueriesForStatement = upsertQuery;
} else {
    combinedQueriesForStatement = combinedQueriesForStatement + "; " + upsertQuery;
}
if (rowsInBatch == 2500 || totalRowsProcessed.equals(totalRecordsToProcess)) {
    Statement stmt = conn.createStatement();
```

Tools & Resources

To help you reach the top

PostgreSQL & Redis UIs

- PGAdmin
- Redsmin

Heroku Addons

- Papertrail
- Redis
- Postgres

Code Reference

- Web Services Connector on Github

Dreamforce Session Recordings

- Heroku Postgres
- Integration at Scale

Articles

- Non-Web Dynos
- Heroku Connect Architecture

Maven

- Intro to Repositories
- http://mvnrepository.com/

Thank Y☁u

Code: goo.gl/tgqVih

Slides: Session Chatter Feed & Github