Caliber stores the data

Biforce is the interface in a way

Caliber 1 - 2017 (Build on Spring and Angular)

performance aggregator

collects info from QC, trainers, other sources

assessments from trainer and QC outlets

Caliber 2

build to replace caliber 1's issues with maintenance

has lots of historical data in it, cannot be lost

schemas are different, but built to be very similar to make transitioning easier

we must come up with a migration plan

Goals:

1. Migrate data from Caliber 1 into Caliber 2 (migration plan)

2. BI on metrics on the building of this (reports);

aggregate metrics to see if this process is feasible for other systems that are old

A large amount of data- but not a very very large amount.

When we present findings on BI side, we will have data, but we will be presenting the system that can get the report, not the actual report.

first goal: infrastructure must be provisioned

EMR cluster

RedShift for reporting needs

will allow us to pick a reporting technology we want

BiForce (name of project)

uses MetaBase- can use again if needed (is open source)

Was a tool used against the Caliber DB, gathered KPIs on trainers, associates, etc.

UI with creation of questions, and would answer to those questions

Milestone 1: spin up what already exists

Milestone 2: at least one migration

basically partitioning the schema, at least one should be done by the end of the week

some of them will be very challenging-

at least ONE must have a migration plan by the end of the week.

Milestone 3: Project completion

\* project showcase (will see one, 11 am Thursday example)

1. everyone must present

2. working demo

3. must gather certain metrics to present

End: QA (extremely fair game; public to world)

\* startup documentation

when setting up, read through it- if it needs changing, we must fix them ourselves

IF WE CHANGE ANYTHING- WE MUST UPDATE IT

\* features on github

just update issues on github, who approved it, here's where we're at

or to say what branch of work it's on-

\* work on features, then merge into dev - at the end of the sprint, we will merge request into staging.

\* If we added code, what did we test the code/how

\* number of story points complete

\* calculate agile velocity

Peter Product owner of BiForce

He's Product Owner of Caliber (And also Project Manager for all project 3s)

@cloud-ops (ping people who are responsible for cloud admin)

assessments will be extremely hard - 1-through relationship dealing with

Mainly:

Sqoop

Spark

MapReduce

will ping @cloud-ops to ask for EMR cluster for this stuff etc

s3 bucket to store this info, ect redshift, etc.

github revaturelabs/biforce

1. migration

2. analytics

was: caliber 1

associate -> batch

is now: caliber 2

associate -> assignment -> batch

1 Migration plan from Caliber 1 to Caliber 2

2 Building out caliber 2, looking to do profit analysis

We are building the system to give a report, not actually doing the report

Business questions

Previously

Tool used to collect KPI’s (key performance indicators) about associates (individual level)

Setup locally for biforce

**Coordinate with cloud-ops by Friday** with biforce setup locally

Sprints:

1. Infostructure must be presented
   1. EMR cluster
   2. Redshift
   3. Have to come up with a monitoring tool?

First week is figuring it out

Second week

1. At LEAST 1 migration
   1. Partitioning the schema should be done by this week

Final week

1. Project showcase
   1. Everyone has to present what they’ve done

@cloud-ops

For pinging, help with infrastructure

When we set the project up, if the documentation is missing something, we have to document it

* Document everything!

Master staging, **dev**, features

* Dev is our main branch

Sprint metrics

* MRUnit tests

Total story points over time

Questions for feature:

* Things marked with user story