

#BullhornEngage

# ENGAGE

## Automated Data Synchronization: Data Loader, Data Mirror, & Beyond

Nathan Dickerson  
Adam Crowe  
Ryan McDole



**Nathan Dickerson**  
Senior Developer,  
Bullhorn



**Adam Crowe**  
Senior Developer,  
Bullhorn



**Ryan McDole**  
Senior Developer,  
Bullhorn

# ENGAGE

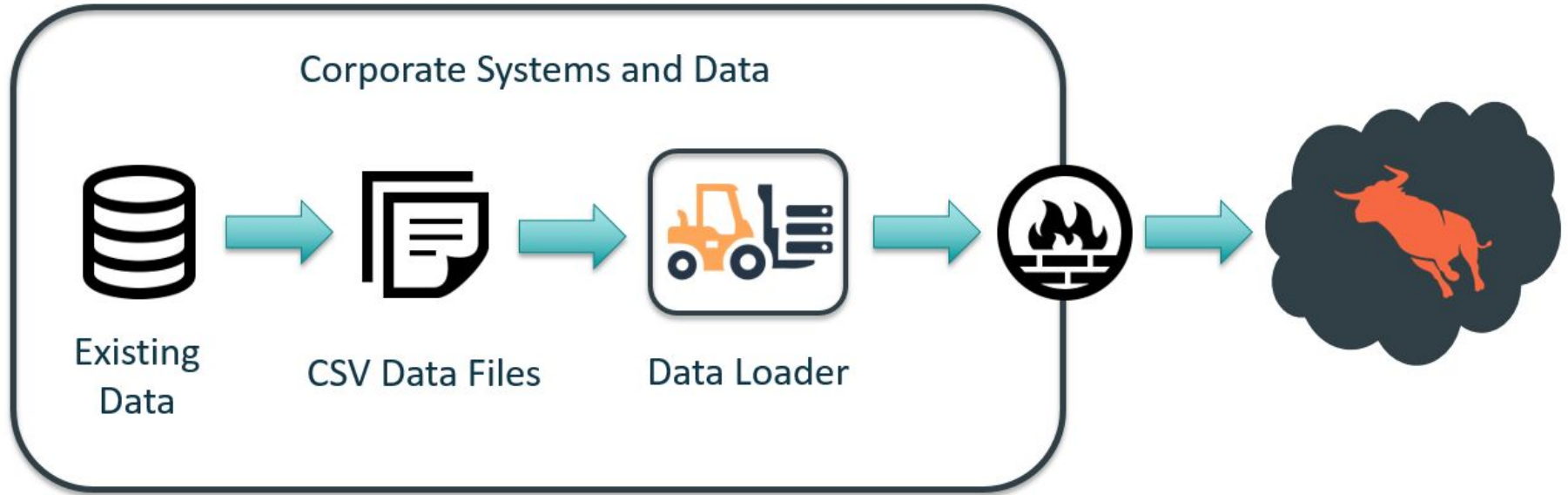
#BullhornEngage

ENGAGE

# Data Loader

Nathan Dickerson

# Data Loader: Mass Import/Update

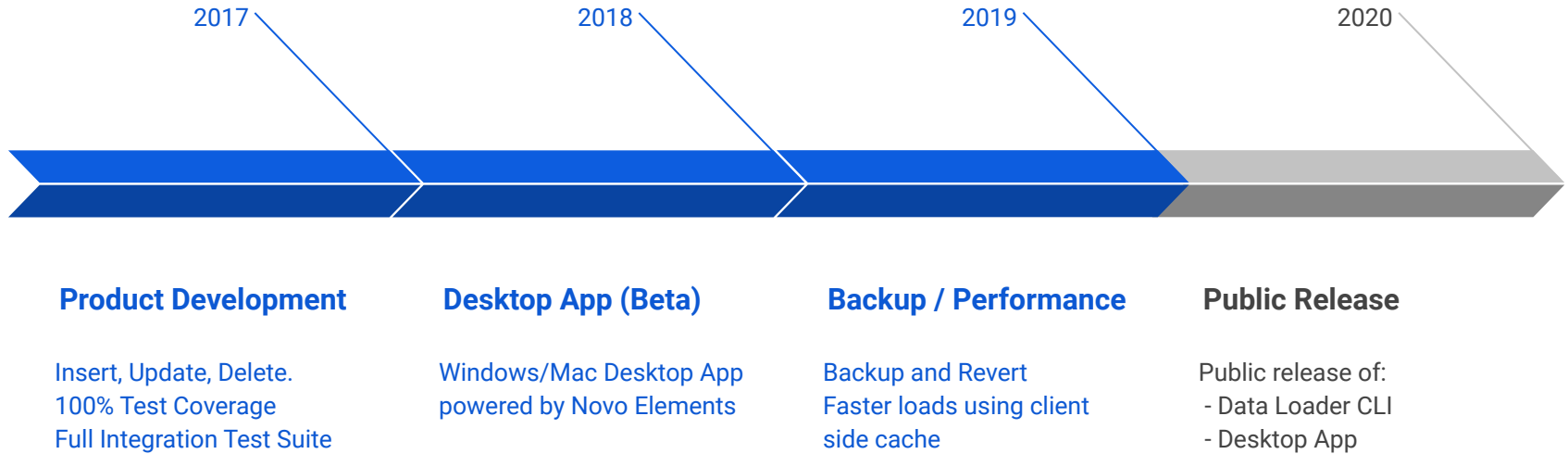


Migrate Your Existing Data Into Bullhorn

# Data Loader by the Numbers

- **166** - Corporations loaded since 2016
- **13M** - Average records per corporation migration
- **1.6M** - Average daily records loaded
- **91** - Resolved GitHub issues
- **50+** - Internal users (support, professional services...)

# Development Timeline



# Performance

- Performance Today:
  - Client-side cache all lookup calls
- Future:
  - Bundle lookup calls across rows
  - Bundle associations with insert/update

# Backup and Revert

- Reverting Today:
  - Backup one record per row before load
- Future:
  - Backup multiple records per row
  - Ad-hoc multi-field backup for update



# Public Release

- Today:
  - Server side rate limiting in version 4.4.0
- Future:
  - Testing/refining of rate limiting
  - Public release of Data Loader CLI
  - Desktop app in Windows/Mac stores



# Desktop App

- Today:
  - Loading and history
- Future:
  - Backup checkbox on load screen
  - Field names from field maps
  - Column mapping from any CSV file

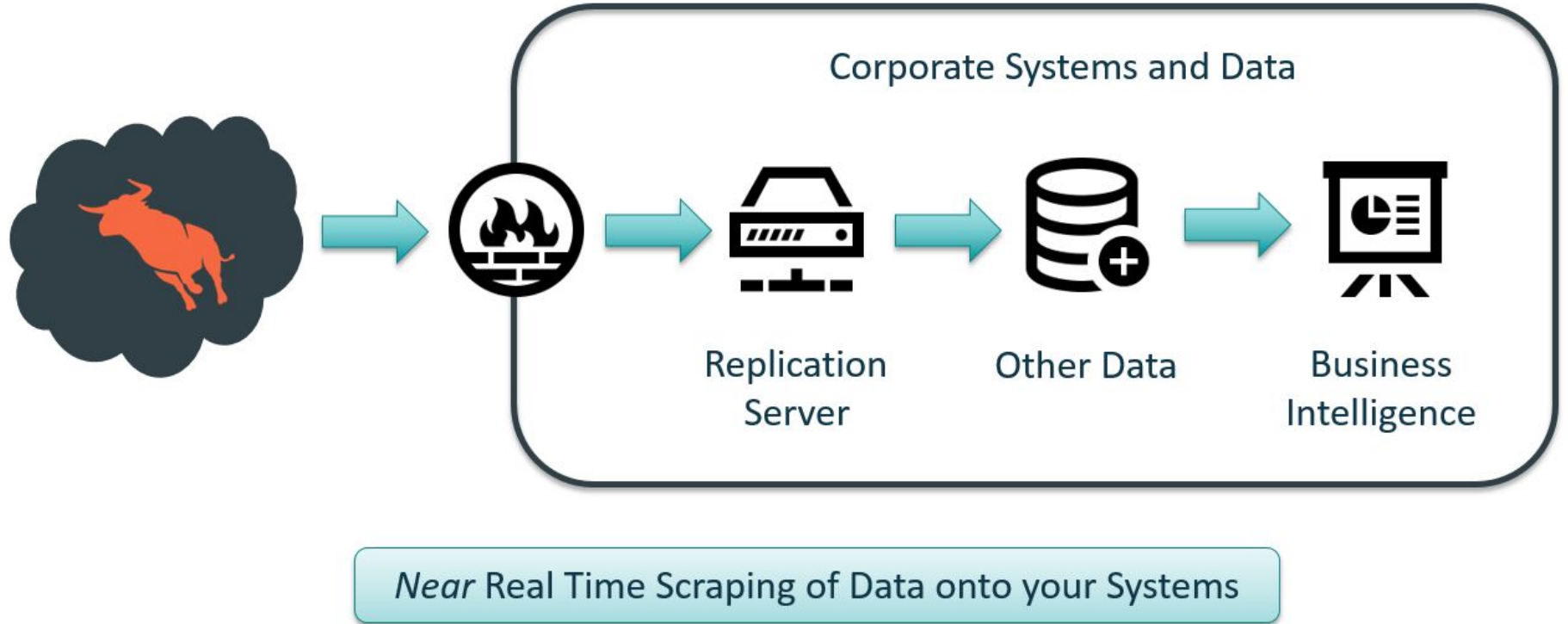
#BullhornEngage

ENGAGE

# Data Mirror

Adam Crowe

# Data Mirror: Your Data. Your Database.



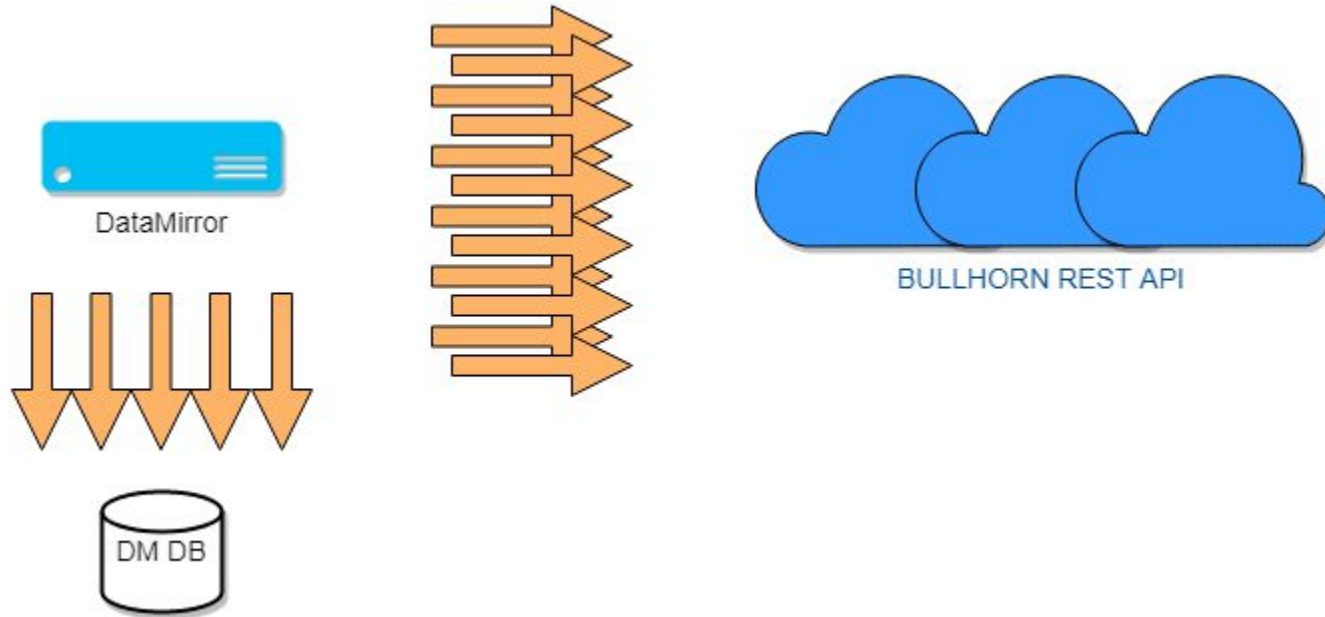
# Data Mirror Features

- Clone all of your data
- Nightly Status Reports
- You provide the server, we provide the software
- Updates are as simple as replacing a single file
- Support handles any issues

# Past Year Improvements

- Replication Optimizations
  - Edit History
  - Custom Object
  - General Replication improvements

# Data Mirror Workflow





# Data Mirror Limitations

- REST API Rate limiting
- 7 day enablement window
- Not as performant as it could be

#BullhornEngage

ENGAGE

Beyond...

Ryan McDole

# Bigger Picture

- REST API is the only access point for data
  - All of our internal services use it
  - All integrations use it
  - All of our products use it
- Millions of REST calls daily

#BullhornEngage

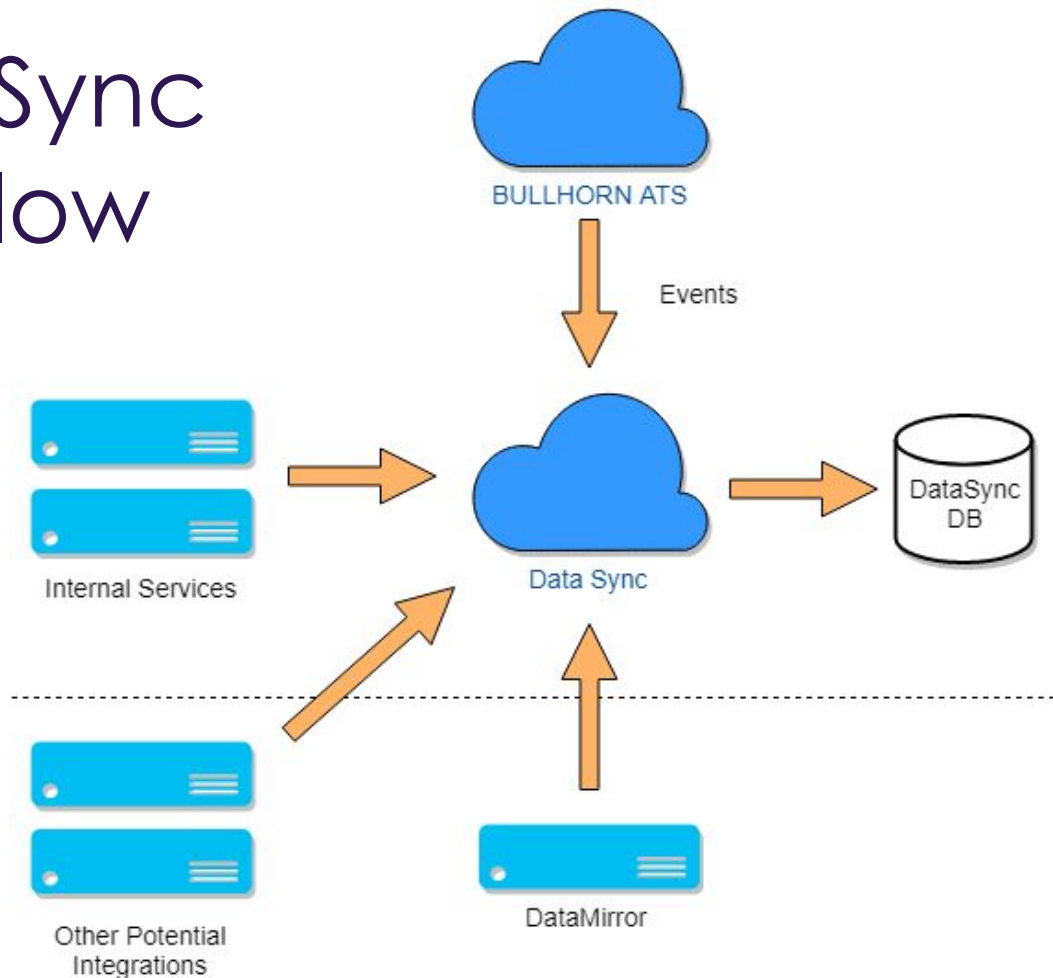
ENGAGE

Data Sync API

# Data Sync API

- In progress
- Independent API separate from our main REST API
- Cached subscription model
  - Retrieves events with relevant data
  - Can get events that were already retrieved

# Data Sync Workflow



# Event Conflation

- Kafka consumer picks up data events
  - firstName of Candidate 123 was updated on corp 44
- Store events in buckets on Redis
  - 44\_Candidate\_123 -> firstName updated

# Data Retrieval

- Event details are used to retrieve actual data from the Bullhorn database to generate a Data Sync object



# Data Sync Object

```
{  
  corpld : 123  
  , id : 345  
  , entityType : 'Candidate'  
  , updated : {  
    firstName : 'Bob'  
  }  
}
```

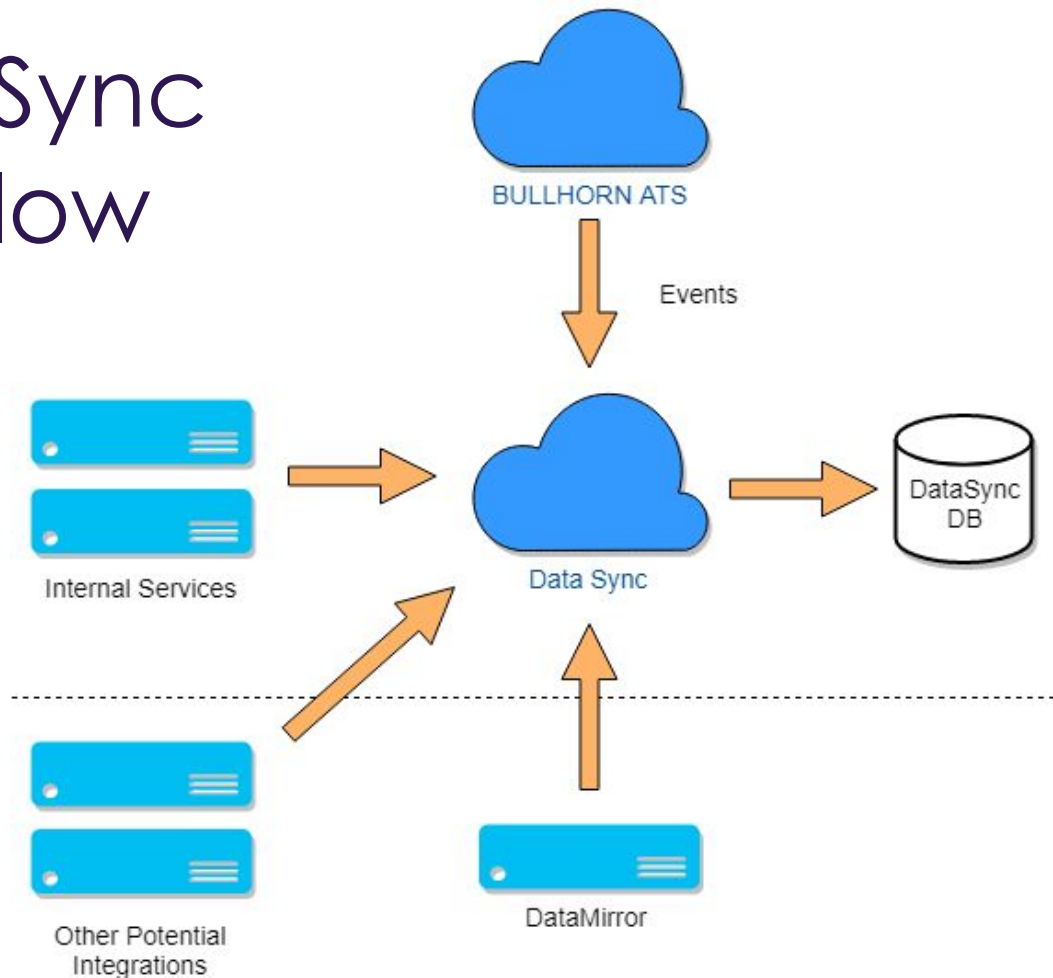
# Data Sync API

- Data Sync object stored in MongoDB.
- When requested, retrieves new objects from MongoDB and returned to caller.

# What about Data Mirror?

- Data Mirror now just asks the Data Sync API for any new events
- Receives everything it needs from that one response to then update it's database.
  - No follow up required
- Won't hit main REST API!
- Independent Rate Limiting

# Data Sync Workflow



# Enhanced Performance

- Significantly faster
  - One REST call to retrieve relevant data for event
  - Updating DB is significantly faster due
- Less resource requirements on your server
- Frees up REST servers to go even faster

# Beyond

- Countless possibilities:
  - Everything



Questions?

ENGAGE

# Data Loader Backup Slides



# Why Data Loader?

- Basic Mechanics
  - Based on Rest SDK project
    - Supports entire Bullhorn data model (and growing)
  - Supports To-One/To-Many fields
    - Ex: `primarySkills.id`, `primarySkills.name`, `candidate.externalID`
  - Supports multiple associations in a single cell
    - Ex: `java;javascript;python`

# Why Data Loader?

- Data Integrity
  - 100% Unit Test Coverage
    - Twice the test code as production code
  - Integration Tests
    - Every command for every field on every entity
    - Over 1,000 records tested for each code change
  - Semver versioning using GitHub releases

# Why Data Loader?

- Powerful Features
  - Supports duplicate checking
    - Ex: externalID (single) or firstName,lastName,email
  - Supports Wildcard searching
    - Ex: java\* => java, javascript
  - Allows up to 20,000 associations per cell
    - Ex: large number of associated business sectors
  - Supports single-byte, dual byte and BOM