POSEIDON

for RESTful Streaming

TIBCO®

AGENDA

AGENDA

RECAP

UPDATES

AGENDA

RECAP

- Who are we?

UPDATES

AGENDA

RECAP

- Who are we?
- Original Goals

UPDATES

AGENDA

RECAP

- Who are we?
- Original Goals
- Our take on the product

UPDATES

AGENDA

RECAP

- Who are we?
- Original Goals
- Our take on the product
- Technology Stack

UPDATES

AGENDA

RECAP

- Who are we?
- Original Goals
- Our take on the product
- Technology Stack
- Progress from Last Time

UPDATES

AGENDA

RECAP

- Who are we?
- Original Goals
- Our take on the product
- Technology Stack
- Progress from Last Time

UPDATES

- Finished Product

AGENDA

RECAP

- Who are we?
- Original Goals
- Our take on the product
- Technology Stack
- Progress from Last Time

UPDATES

- Finished Product
- Demo

AGENDA

RECAP

- Who are we?
- Original Goals
- Our take on the product
- Technology Stack
- Progress from Last Time

UPDATES

- Finished Product
- Demo
- Architecture

AGENDA

RECAP

- Who are we?
- Original Goals
- Our take on the product
- Technology Stack
- Progress from Last Time

UPDATES

- Finished Product
- Demo
- Architecture
- Reflection

AGENDA

RECAP

- Who are we?
- Original Goals
- Our take on the product
- Technology Stack
- Progress from Last Time

UPDATES

- Finished Product
- Demo
- Architecture
- Reflection
- Questions/Feedback

WHO ARE WE?

- Sharat, Eden, Yahui, Artem - Students at Brandeis

- Pito Salas - Professor and mentor

GOALS

GOALS

- Understand capabilities of Streambase system

GOALS

- Understand capabilities of Streambase system
- Brainstorm new or novel ways to interact with it

GOALS

- Understand capabilities of Streambase system
- Brainstorm new or novel ways to interact with it
- Hopefully bring useful ideas to the table

GOALS

- Understand capabilities of Streambase system
- Brainstorm new or novel ways to interact with it
- Hopefully bring useful ideas to the table
- Potentially influence Streambase's development cycle

STREAMBASE STUDIO

- Extremely powerful tool
- User friendly for the most part
- Cross platform
- High performance
- No programming experience needed

STREAMBASE STUDIO

- Installation required
- Desktop only
- Streaming requires both endpoints to implement protocol (sender + receiver)*
- Compile, run workflow
- No flow through data

OUR TAKE…

- Web browser based

OUR TAKE…

- Web browser based
- RESTful API

OUR TAKE…

- Web browser based
- RESTful API
- Live stream view

OUR TAKE…

- Web browser based
- RESTful API
- Live stream view
- JSON Format

OUR TAKE…

- Web browser based
- RESTful API
- Live stream view
- JSON Format
- Real Time manipulation

STACK

- Vaadin

  - Java based

    - Asynchronous

    - Generates Javascript, CSS from Java Code

      - Google Web Toolkit based

LAST TIME

- UI - simple and compelling

PROGRESS

- UI - simple and compelling
- Connects to REST sources

PROGRESS

- UI - simple and compelling
- Connects to REST sources
- Maps JSON

PROGRESS

- UI - simple and compelling
- Connects to REST sources
- Maps JSON
- Operators Implemented

PROGRESS

- UI - simple and compelling
- Connects to REST sources
- Maps JSON
- Operators Implemented
- Data live view

DEMO

- Data sources
  - MBTA
  - Twitter

DEMO

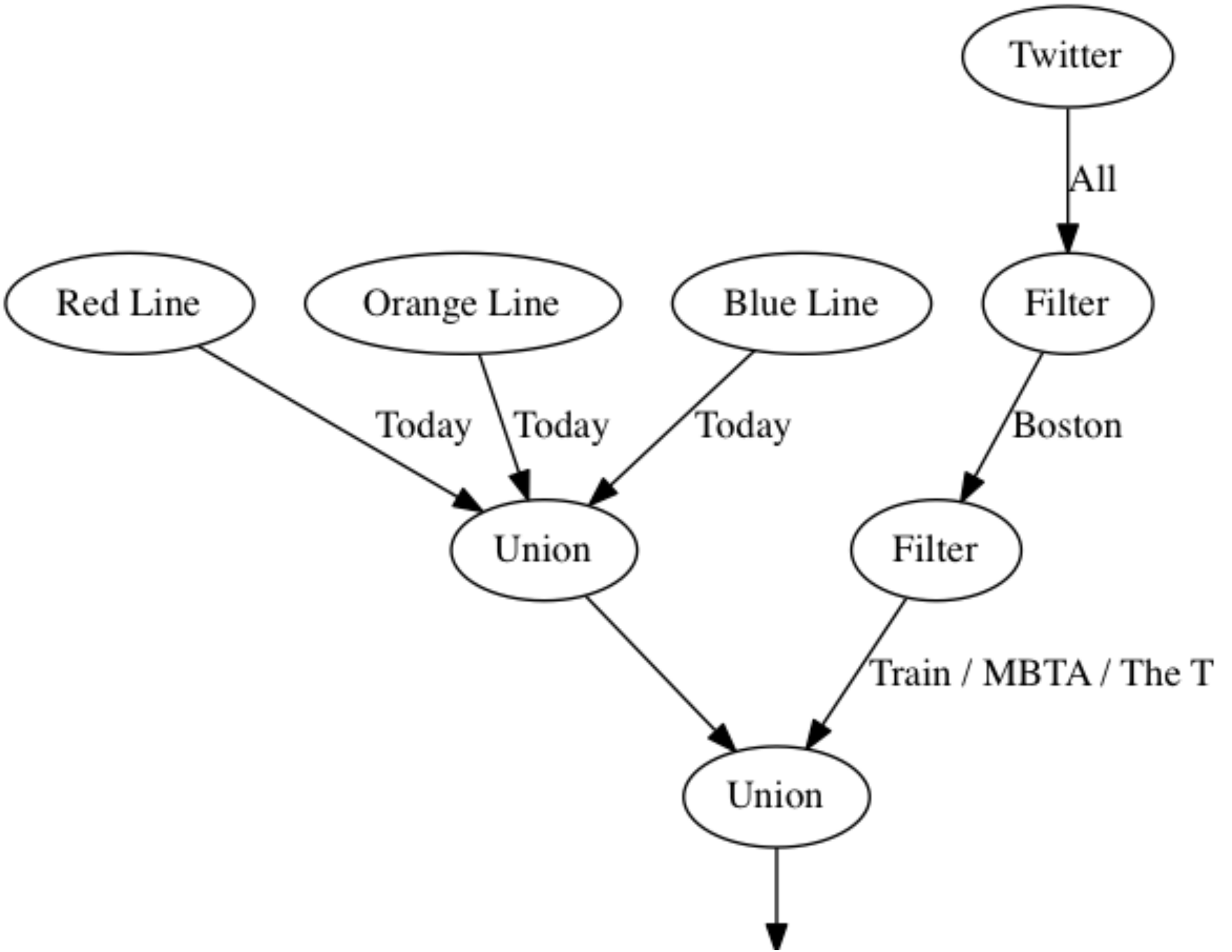
- Data sources
- Answer a question

DEMO

- Data sources
- Answer a question
  - "If I live in Harvard but work at State St, how can I minimize waiting time?"

DEMO

- Data sources
- Answer a question
- Visually assemble a query


Twitter
All
Red Line
Orange Line
Blue Line
Filter
Today
Today
Today
Boston
Union
Filter
Train / MBTA / The T
Union

DEMO

http://edenzik.github.io/StreambaseBrandeis

http://poseidon-streambase.herokuapp.com

ARCHITECTURE

- Operators
  - Independent "units"
  - Have a buffer (Queue)
  - Each operator processes least recent element
  - When done, forwards to next operator's buffer

ARCHITECTURE

- Operators
- Threads
  - Each operator = Thread
  - Runs continuously
  - Agnostic of other operators

ARCHITECTURE: PROBLEMS

- Real time push
  - Data pours through as soon as connection is made
- Join "Synchronization"
- Bottleneck of system is at slowest node

ARCHITECTURE

- Operators
- Threads
- Vaadin

ARCHITECTURE

- Operators
- Threads
- Vaadin
    - Continuous survey of operators

ARCHITECTURE

- Operators
- Threads
- Vaadin
  - Continuous survey of operators
  - Push to front end

REFLECTION:VAADIN

- Difficult to debug
- Steep learning curve that never ends
- Expensive Asynchronous UI Pushes
- Low plugin support compared to JS plugins
- Recompile time is frustrating
- Clunky in development and usage

REFLECTION: IF REDONE

- Front-end javascript framework using plugins would have been easier and faster

- Node.js for server for low overhead and speed

QUESTIONS?