Framework Domain

- ➤ Domain: Event-based data for visual display with countable keywords and optional priority
- > Example data plugins:
 - Facebook posts with location/emoji
 - Webscripting a site such as TicketMaster with concert and song listings
 - Basic CSV file
- ➤ Example visualization plugins:
 - Frequency display
 - Roadmap of events with optimal routes



Framework Details

This framework will be made to be reasonably general to handle most event based data in a particular region.

>Key abstraction:

- Any event will have a time, place, and some description keywords.
- Visualizing events on a map is the most natural presentation and is flexible.

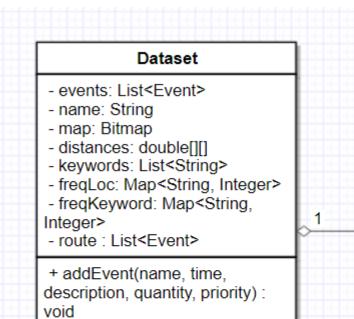
There is tons of flexibility in plugins:

- Data would only require the bare minimum: time, place, and some description at minimum.
- For visualizing, there are many modes of representing data on maps (e.g. frequency, path scheduling).

Project structure

- framework
 - core
 - (I) MapperListener
 - (C) MapperFramework
 - (I) DataPlugin
 - (I) VisualPlugin
 - datapoint
 - (C) DataSet
 - (C) Data

- plugin
 - data
 - (C) TicketMasterPlugin
 - (C) SpreadsheetPlugin
 - visual
 - (C) MapRoutePlugin
 - (C) KeywordFrequencyPlugin



+ getDistances(List<Event>):

+ getTimes: List<Time>

+ combine(List<Dataset>):

double[][]

List<Event>

+ process(): void

Event + subject: String + time: Time + location: String 0..* + latitude: double + longitude: double + keywords: List<String> + quantity: int + priority: int + compareTo(event): int

