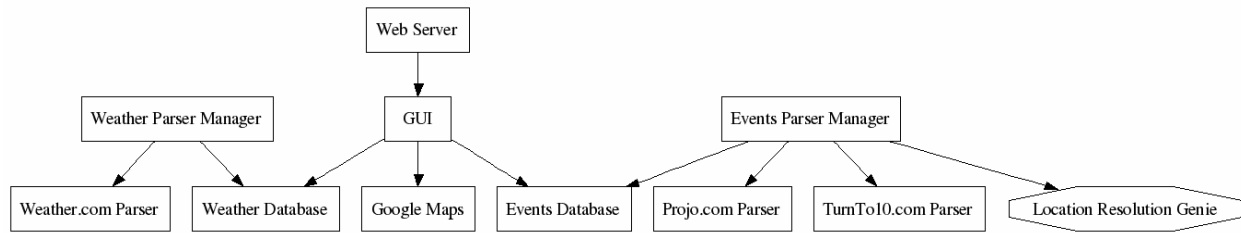


GeoEvents

Top-Level Design

Sam Cates
24 February 2006

1. Component Diagram



- a. Web Server (2): This is the HTTP server that users will connect to via their web browser. This server will host the GUI code that generates HTML pages in response to user requests. The database(s) might also be stored on the same machine.
- b. GUI: (1) The GUI is responsible for formulating responses to user requests. This means that the GUI not only has to be able to take the input and decipher some meaning from it, but it also has to decide what to query for from the database(s) based on that meaning and then it has to use Google Maps to display the info it gets from the database(s). The GUI is basically the glue of the whole project ties all the separate pieces together.
- c. Weather Parser Manager (1): The Weather Parser Manager is in control of deciding when to gather new information about weather. When the time comes, the manager uses one or more parsers to collect weather data and update the Weather Database.
- d. Weather.com Parser (0): The Weather.com Parser, when called upon, parses the weather.com website, collecting data about local weather.
- e. Weather Database (0): The Weather Database stores information about weather for about the next two weeks.
- f. Events Parser Manager (1): The Events Parser Manager is in control of deciding when to gather new information about events. When the time comes, the manager uses one or more parsers to collect events data and update the Events Database.
- g. Projo.com Parser (0): The Projo.com parser, when called upon, parses the projo.com website, collecting data about local events.
- h. TurnTo10.com Parser (0): The TurnTo10.com parser, when called upon, parses the turno10.com website, collecting data about local events.
- i. Events Database (0): The Events Database stores the most important information, the actual events. Locations should already be resolved and latitude/longitude coordinates already computed and stored in the database.

- j. Location Resolution Genie (0): The Location Resolution Genie is the component that takes a “location” and returns both an address and a pair of latitude/longitude coordinates corresponding to that address. The first version of this should be trivial. Any location that is passed in that isn’t just an address that can be geocoded should just return null which will cause the Events Parser Manager to drop the event instead of add it to the database. Once this simple functionality is in place, if there are sufficient resources to do something more clever, then a new strategy may be devised to replace this first one.

2. External Dependencies

- a. Google Maps
- b. Weather.com
- c. Projo.com
- d. TurnTo10.com
- e. LocalSearchMaps.com GeoCoder

3. Task Breakdown

- a. Design the GUI
- b. Write the interface and documentation for each database
- c. Write the Events Parser Interface and Documentation
- d. Write the Weather Parser Interface and Documentation
- e. Write the Location Resolution Interface and Documentation
- f. Pseudocode the database
- g. Pseudocode the parsers
- h. Pseudocode the Location Resolution Genie
- i. Pseudocode the parser managers
- j. Pseudocode the GUI
- k. Revise the interfaces and documentations
- l. Setup the Web Server
- m. Design component tests
- n. Implement the databases
- o. Test the databases
- p. Implement the parsers
- q. Test the parsers
- r. Implement the Location Resolution Genie
- s. Test the Location Resolution Genie
- t. Implement the parser managers
- u. Test the parser managers
- v. Implement the GUI
- w. Test the GUI
- x. Test component integration
- y. User-test

4. Group Organization

- a. Sheriff (Sam): This project is small enough for one person to know every line of code. The sheriff is responsible for upholding conceptual integrity across components written by different people by reading all code written and revising code where he sees fit. The sheriff enforces coding conventions and acts as a mediator in design debates because he should typically understand the needs of all the different components. The sheriff's final judgment is law.
- b. Database Engineer (Toby): One person will be responsible for designing, implementing and documenting both databases. The reasoning is that the databases, while storing very different data, will require the same resources, very similar code and will both be used in the same context in the GUI, so should have very similar interfaces.
- c. Parser Engineers (Kaveh, Andy, Amy): One person will be responsible for designing, implementing and documenting each parser. The parser engineers should design their parsers together to increase conceptual integrity and then implement them in parallel. The sheriff should be present at their design meeting.
- d. Manager Engineer (Yotsowan): One person should be responsible for implementing and documenting both Parser Managers. This person also designs the database interfaces with the help of the Database Engineer and the two Parser Interfaces with the help of the Parser Engineers.
- e. Location Resolution Engineers (Amy, Toby): Two people should be in charge of first implementing a more-or-less trivial Location Resolution Genie and then designing and implementing a more sophisticated one.
- f. Test Engineers (Andy, Peter, Kaveh): Three people should be in charge of designing component tests for the databases and the parsers.
- g. GUI Designer (Haley): One person should be in charge of the original GUI design and then any decisions that need to be made about changes to the original GUI design as problems arise.
- h. GUI Engineers (Yotsowan, Peter): Two people should be responsible for implementing the GUI.
- i. Google Maps Expert (Sam): One person should be responsible for knowing just about everything about what one can do with the Google Maps API and how to do it. Depending on the design for the GUI, this person may be asked by the GUI Engineer to write a wrapper for the Google Maps interface.
- j. Web Server Administrator (Sam): One person should be responsible for setting up and maintaining the web server.
- k. Deputy Doc (Haley): One person should be responsible for revising and integrating all the documentation. This person is basically the documentation equivalent of the sheriff.
- l. Secretary (Amy): One person should be in charge of organizing and scheduling meetings. This person should also keep track of the overall project schedule and keep people aware of their deadlines. The secretary is not, however, a taskmazer.

5. Schedule
 - a. 3/3 Design the GUI
 - b. 3/8 Write the interface and documentation for each database
 - c. 3/8 Write the Events Parser Interface and Documentation
 - d. 3/8 Write the Weather Parser Interface and Documentation
 - e. 3/6 Write the Location Resolution Interface and Documentation
 - f. 3/13 Pseudocode the database
 - g. 3/13 Pseudocode the parsers
 - h. 3/10 Pseudocode the Location Resolution Genie
 - i. 3/13 Pseudocode the parser managers
 - j. 3/13 Pseudocode the GUI
 - k. 3/17 Revise the interfaces and documentation
 - l. 3/10 Setup the Web Server
 - m. 3/20 Design component tests
 - n. 3/27 Implement the databases
 - o. 3/29 Test the databases
 - p. 3/27 Implement the parsers
 - q. 3/29 Test the parsers
 - r. 3/20 Implement the Location Resolution Genie
 - s. 3/22 Test the Location Resolution Genie
 - t. 3/29 Implement the parser managers
 - u. 3/31 Test the parser managers
 - v. 3/29 Implement the GUI
 - w. 3/31 Test the GUI
 - x. 4/5 Test component integration
 - y. 4/12 User-test

6. Specification Assumptions
 - a. That we will come up with a way to resolve locations reliably