MICHIGAN STATE UNIVERSITY

Project Plan

Boeing O-Show

The Capstone Experience

Team Boeing

Bryan Askins
Scott Buffa
Matt France

Department of Computer Science and Engineering
Michigan State University

Fall 2010

From Students…
…to Professionals

Project Overview

- Boeing Optimize and Show (O-Show)
- Core Problem:
  - Given a fixed set of maximum resources to serve a fixed set of demands, find the optimal way to organize and task the resources to serve those demands.
- Find the minimum number of stadium evaluators required to evaluate all of the university stadiums in Missouri within a limited amount of time.

Functional Specifications

- **Input:**
  - stadium evaluators
  - the maximum distance each one can travel
  - locations of the stadiums
  - starting point
  - time limit
- **Output:**
  - minimum number of evaluators necessary
  - routes which they took.

- The software shall visualize the output of lp_solve in a meaningful way, such as showing, over time, the stadium evaluators traveling along their optimized routes.

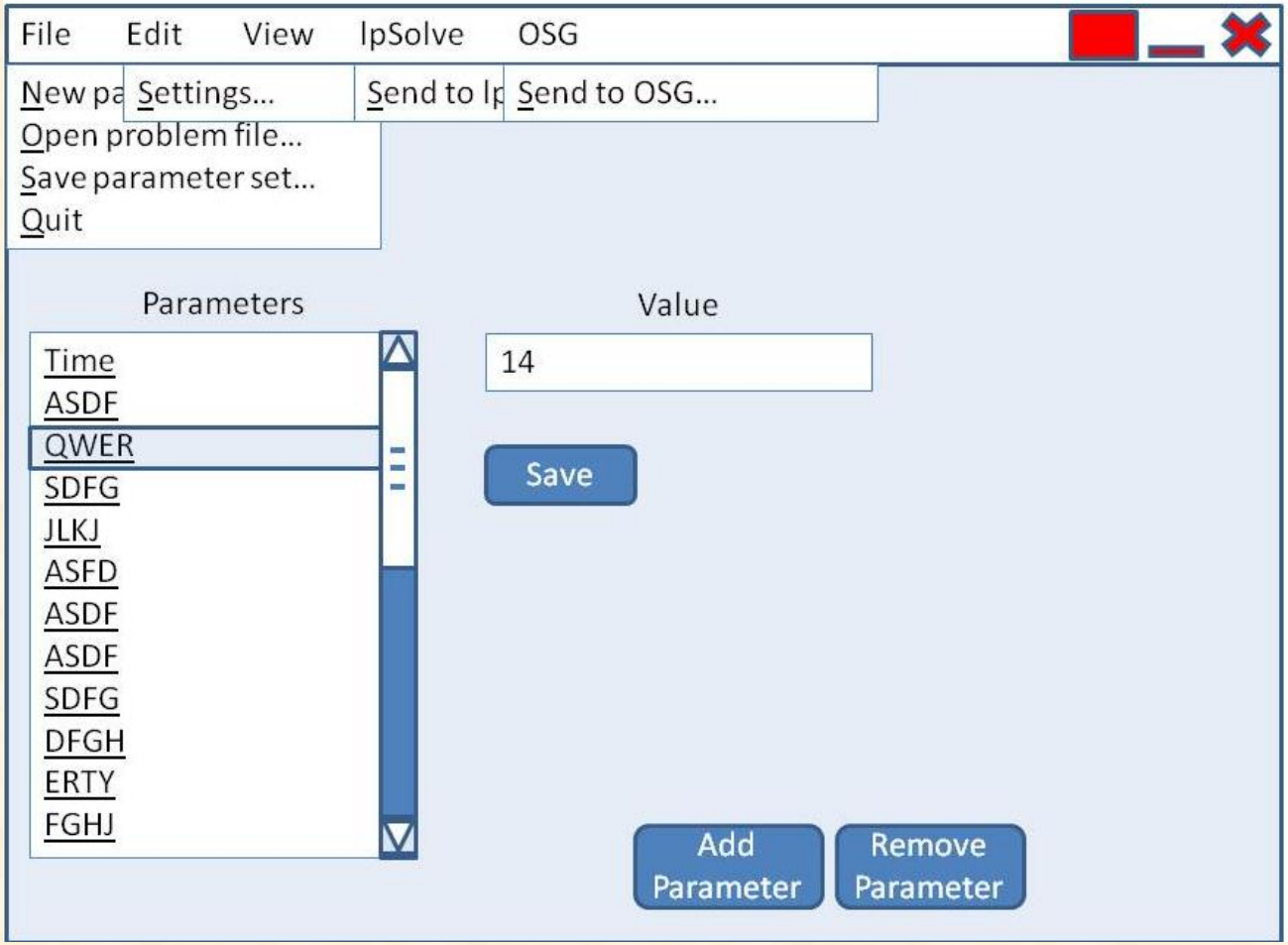
- The project should include a user interface which allows the tool to be used more easily and effectively.

- The user interface shall allow the user to input a file, edit the input parameters, and start the simulation process. Once the user has started the process, the manager shall handle starting the simulation and visualization.

Design Specifications

- The project shall have three main pieces:
  - User interface and manager software
  - Simulation software
  - Visualization software
- The program is started from a launcher which is written in Python.
- There will be entry fields for all of the parameters, such as the number of evaluators, a time limit, a starting point, and an input file with a list of destinations.
- The visualization software will include a user interface that will allow the user to control the visualization time and also allow them to change camera location/orientation and view information about a selected object.

Screen Mockups

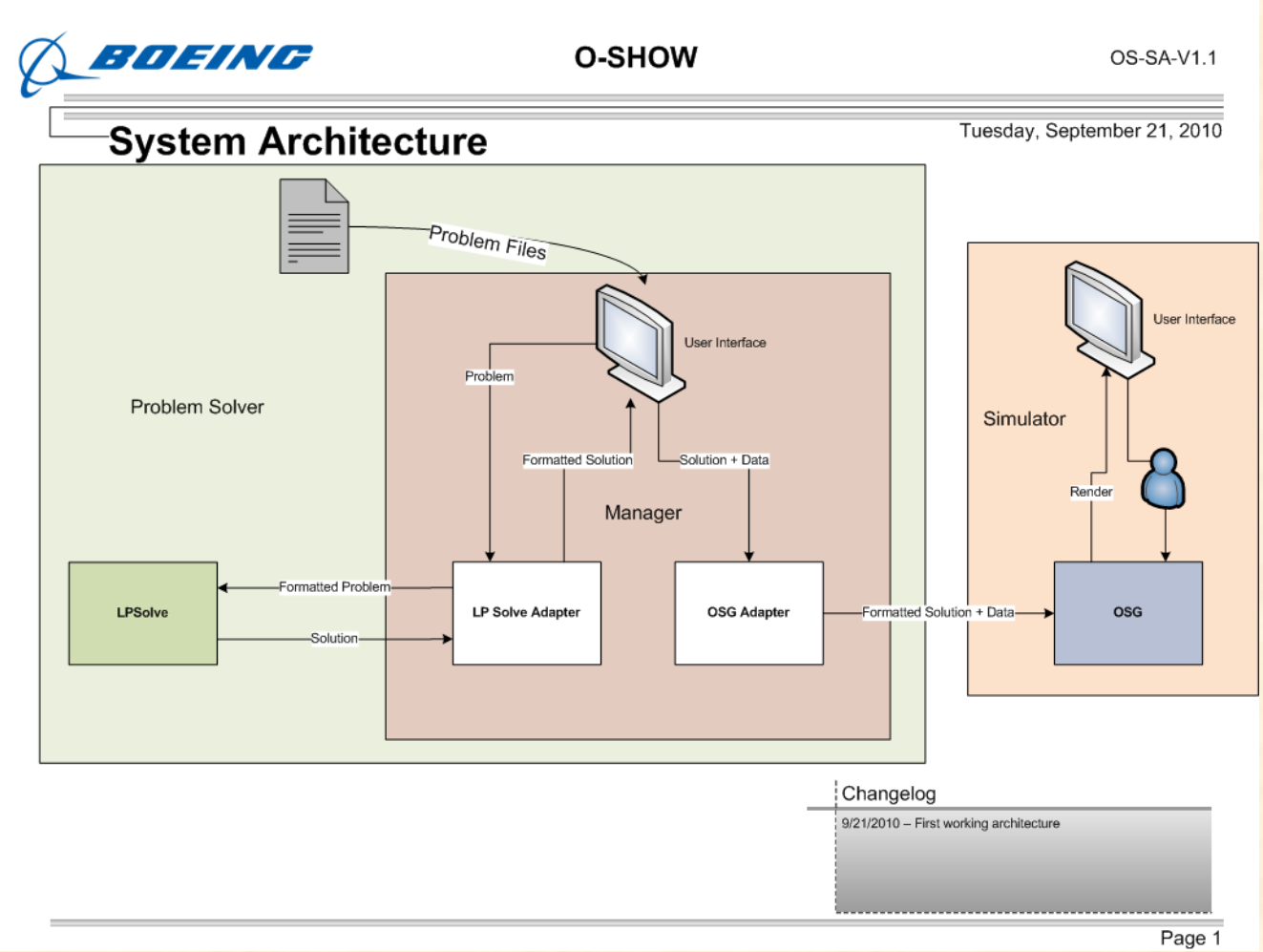


Python GUI Mockup

Technical Specifications

- Manager software:
  - GUI will be written in Python v2.6 using wxPython.
  - Calls the simulation software once an input file has been specified and the user has entered the input parameters
  - The manager will send the data to lp_solve.
  - Handles starting the visualization software which will render the scenario.
- Simulation software:
  - Written in C++.
  - Interface with the linear programming solver lp_solve.
  - The simulation software shall create a mathematical model and pass it to lp_solve.
  - The evaluators, stadiums, routes will be represented using C++ classes.
- Visualization software:
  - Written in OSG.
- A majority of the development will be done using the Eclipse IDE.
- The initial target platform for this project is Linux.

System Architecture



System Components

- Hardware Platforms
  - PC
- Software Platforms / Technologies
  - Ubuntu 10.04
  - Fedora 13
  - OSG
  - Lp_solve
  - Python 2.6
  - wxPython
  - Eclipse with PyDev

Testing

- The input generated for lp_solve will be tested using the standalone lp_solve binary to ensure that the input is being handled as expected.
- The simulation software will be tested at a basic level by testing the classes and class methods, followed by the interaction of the classes with OSG.
- The Python front-end will be tested with test-cases comprising of unexpected input.

Risks

- Formulating a linear programming model for the problem and passing that into lp_solve.
- Interpreting our output data from lp_solve and visualizing it.
- Very little experience with OSG.
- Manipulating time and perspective in the visualization.