The Oil Exploration Game

Dr. Kevin Brewer, Department of Physical Sciences
Dr. Larry Vail, Department of Computer Sciences
Olivet Nazarene University

Introduction/Overview

The Oil Exploration Game (OEG) from Olivet Nazarene University is a Java-based set of programs designed to handle the "mechanics" of running a traditional paper-based oil and gas exploration game. This turned-based game pits teams (or individuals) in a quest to find oil and gas production. You will explore using seismic line requests, bid (against the other teams) to acquire land leases, and drill, baby, drill. All that, of course, if you have the money – so keep an eye on your bank balance! And with all the mechanics of the game computerized, you will be able to focus on your game strategy (the fun part!), not on the mechanics of playing.

How to Setup and Run

Create a Simulation

The instructor will create a simulation that consists of an XML file that describes the simulation parameters, and a series of CSV files that contain the gridded data (stratigraphy, oil, and gas). The instructor may use the OEGCreator Java program to create a fairly realistic domain and associated CSV files, but using any method to create any valid domain is fine.

The instructor will inform each team of the basic domain (maybe provide a basic map with surface elevation and surface rock type per cell?), other basic game information (number of turns/rounds, length of turns, starting bank balance and costs for various activities, etc.), minimum interaction requirements (if any), and criteria for "winning".

Start the Director

The instructor will start the OEGDirector Java program and load the simulation. The instructor will then communicate to each team the IP address of the OEGDirector, which the teams will use to connect.

> *Note: The game time starts as soon as the simulation is loaded. The OEGDirector can be run on any Java-capable, internet connected computer, but the OEGDirector program must always be "on" during the entire game. Choose an appropriate computer to host the OEGDirector program for your game.*

Teams Connect and Play

Each team will start the OEGServer Java applet and enter the IP address and team name. The OEGServer program will inform the teams how much time is left in the current turn, facilitates team requests (land

lease bids, seismic data requests, and well drilling), and provides information about team status (primarily their current bank balance). Teams will continue to interact with the OEGServer program until the end of the game.

Currently, teams can exit the OEGServer program at any time and reconnect with the IP address and their team name at a later time (from any computer) without any loss of data/status.

> *Note: There currently is no "security" verification for team connection. Anyone with the team name (and game IP address) can connect to the Director and have full control of team actions.*

Instructor Monitoring

The instructor will monitor the progress of the game with the OEGDirector program. At the end of the game, the instructor can download a history log of game activity and exit the OEGDirector program.

How to Play

After choosing a team name and receiving the IP address from the instructor, each team will start the OEGServer applet via a provided html and jar file. Teams should also have been given a basic map of the game domain, and other basic game information.

At the beginning of each turn, the previous turn's winning lease bids will be awarded, seismic data returned, drilling results returned, and any income from previously successful oil and/or gas wells will be credited to the team's bank balance.

For each turn, each team may:

- Bid for leases for drilling rights for individual cells. (The amount of the bids will be frozen in the team's bank account – any unsuccessful bid amounts will be unfrozen at the beginning of the next turn.)
- Request seismic data (either E-W lines or N-S lines) (and deduct the appropriate amount from the team's bank account). The stratigraphy data (i.e., tops of layers) returned will be "fuzzy", with lower layers less likely to be seen.
- Drill on cells that the team has leased in a previous turn (and deduct the appropriate amount from the team's bank account). Each drilled well will return accurate stratigraphy data and production (oil or gas) information (or if it is a dry hole.

Teams will not be allowed to exceed their bank balance at any time. Teams may rescind any bid, seismic, or well request prior to the end of the turn. Some information is public and will be available to all other teams (who owns leases, well and production information).

Support Contact

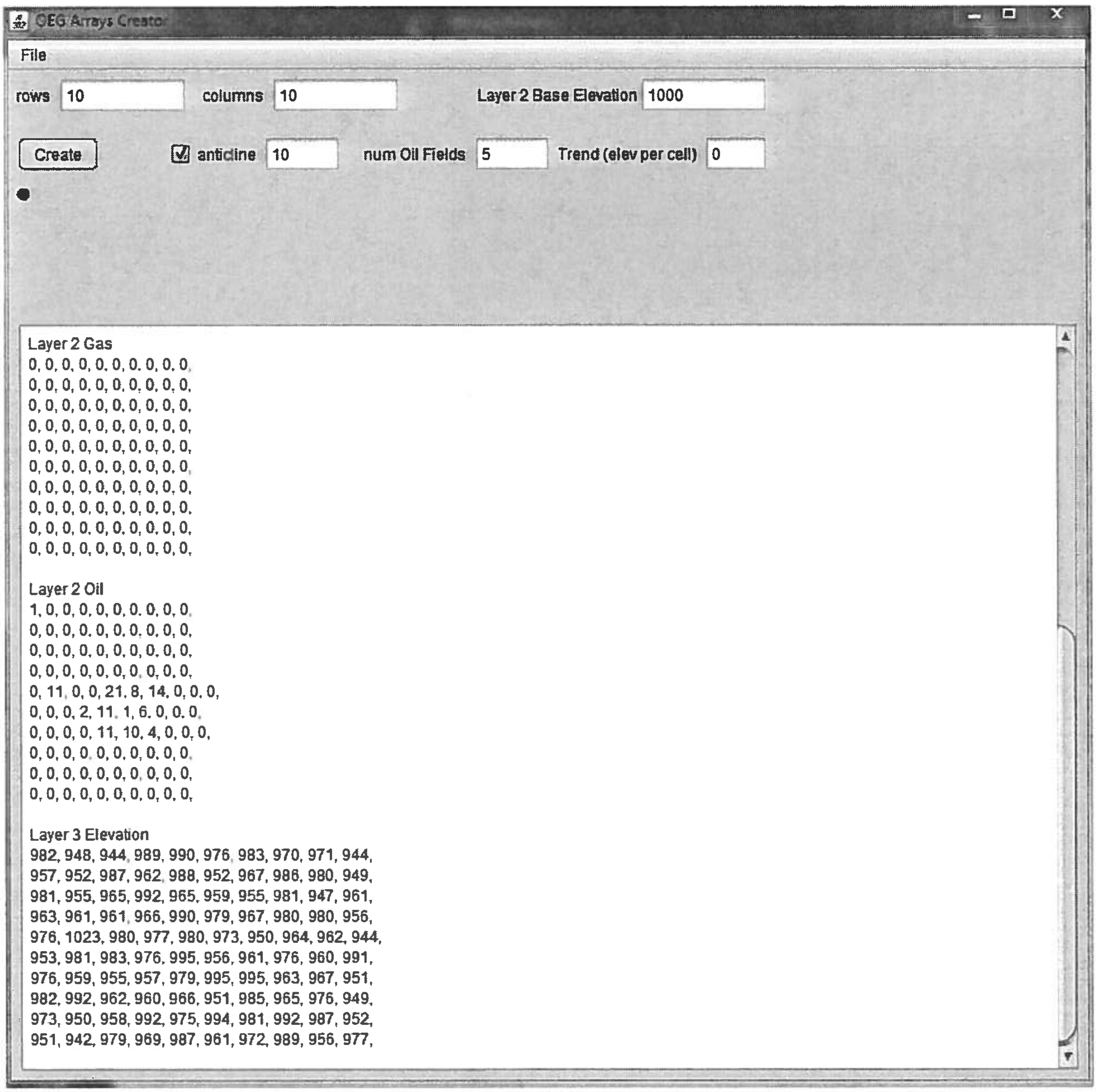
Please contact Dr. Kevin Brewer at Olivet Nazarene University at kbrewer@olivet.edu with any questions.

Version Information

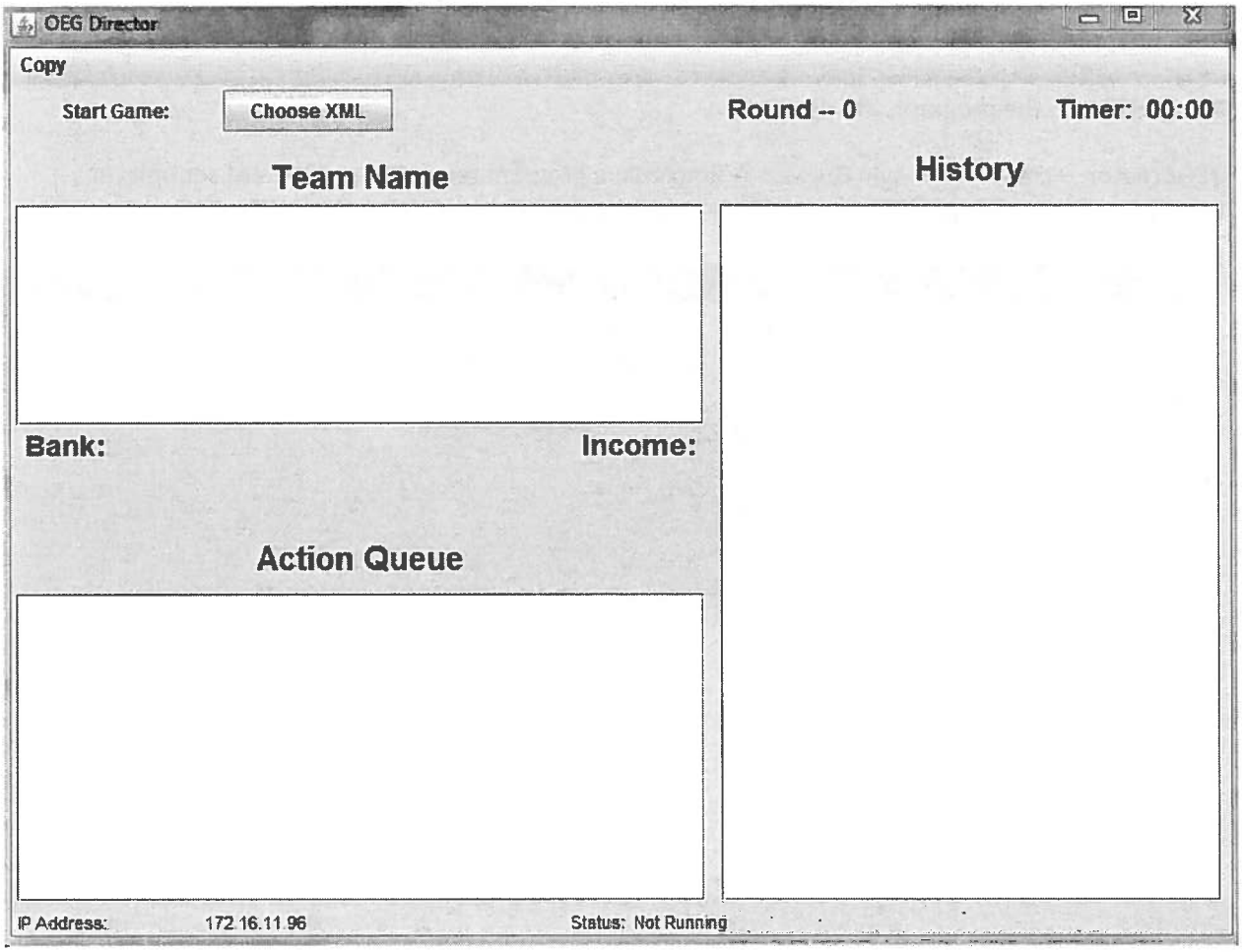
The current version of the OEG is 1.0. The interface is minimal (i.e., there is room for significant improvement to the graphical user interface) and team security is minimal (i.e., other teams can easily "hijack" another team's connection). Improvements to OEG are expected in Fall2013.

The specifics of the programs are as follows:

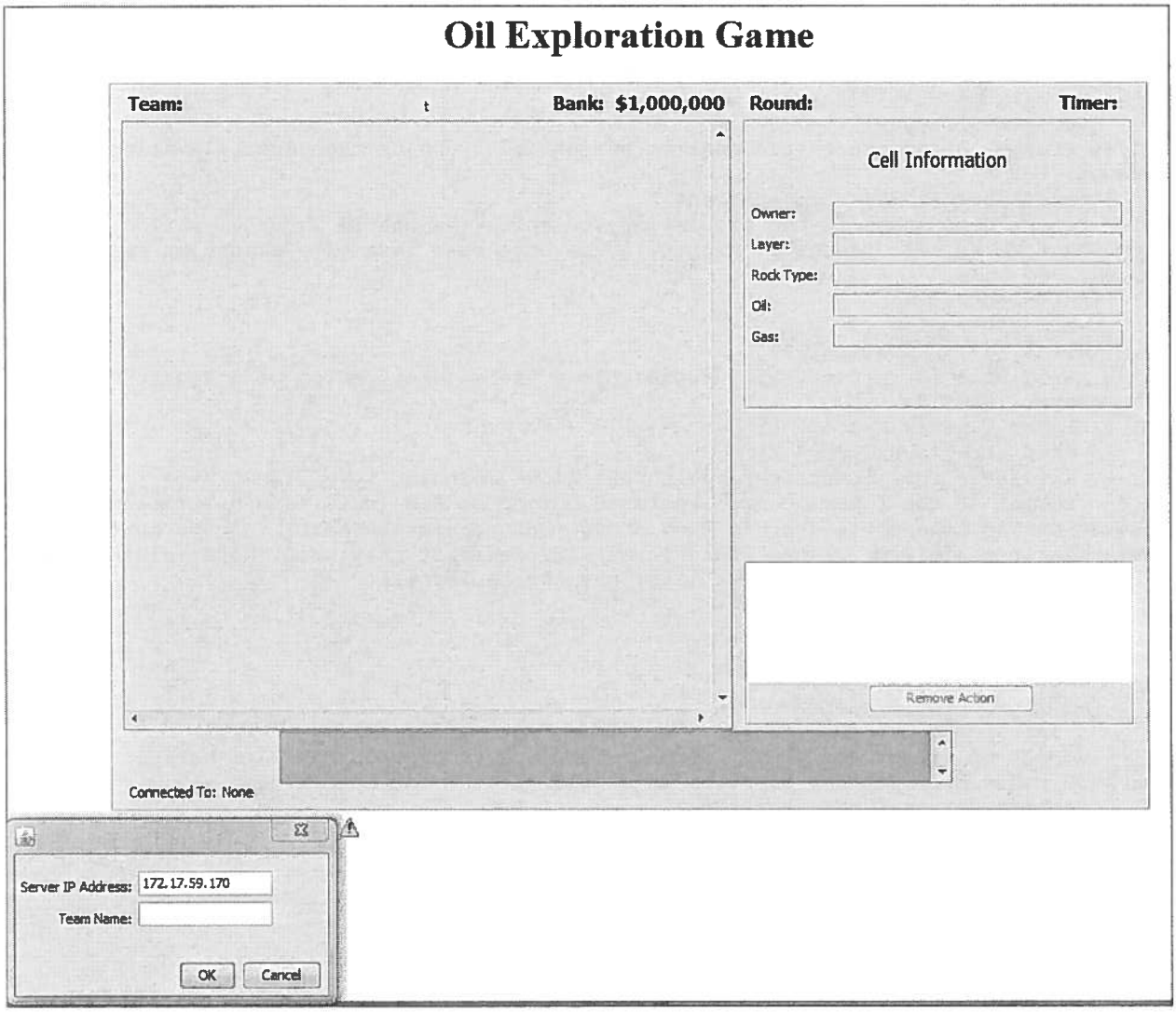
OEGCreator – created by Kevin Brewer. A simple Java program to create a consistent set of layer elevation, oil content, and gas content CSV formatted arrays.



OEGDirector – created by ONU Computer Science. The Java program that will run on the instructor's computer, which communicates with each team's Java program, and which handles the game dynamics (data requests, land lease bidding, etc.).



OEGServer – created by ONU Computer Science. The Java program (applet) that runs on each team's computer, which communicates with the OEGDirector program. It includes a rudimentary graphical representation of the simulation grid.



XML File for Example Simulation

```
<?xml version="1.0" ?>
<!--  This is a sample general simulation data input file    -->
<!--  Created by Kevin Brewer 10/14/11                       -->
<simdata version="OEG01">
  <simname>Sample Simulation 01</simname>
  <simdomain layers="3">
    <fudge>10</fudge>
<!--   Fudge percentage is the maximum percentage by which the layer elevation will be
skewed. -->
    <removepercent>10</removepercent>
<!-- Remove percentage is the probability by which the bottom layer elevation will be
removed from seismic requests. Higher layers have even less of a chance of being
moved. -->
    <xdim>100</xdim>
    <ydim>100</ydim>
    <gridsize>1320</gridsize>
<!--  Grid size is in feet. 1320 represents a 4x4 discretization of a square mile. -->
    <layer lay="1">
<!--  Layer 1 never has oil or gas -->
      <top src="layer1.csv" />
<!--  all layer top elevations are in feet above mean sea level (AMSL) -->
<!--  Format of top files: Comma separated. Oriented E-W (rows) and N-S (columns).
Elevations of tops of layers, in feet above mean sea level (AMSL) (always integers,
negative or positive). Blanks indicate missing layer at that grid. Note, since layers
are about geologic types, surface layer may not be layer 1! -->
      <rocktype>Shale</rocktype>
    </layer>
    <layer lay="2">
      <top src="layer2.csv" />
      <oil src="oil2.csv" />
      <gas src="gas2.csv" />
<!-- Format of oil or gas files: Comma separated. Production rates in barrels per day
or 1000 cubic feet per day for each grid cell. Zero means no oil/gas. -->
      <rocktype>Sandstone</rocktype>
    </layer>
    <layer lay="3">
<!--  Bottom layer never has oil or gas -->
      <top src="layer3.csv" />
      <rocktype>Basalt</rocktype>
    </layer>
    <bottom>1150</bottom>
<!--  Bottom is the flat bottom elevation of the lowest layer. -->
  </simdomain>
  <simtimesteps simsteptime="30">
<!--  simsteptime is the number of days each step represents in the simulation -->
    <numsteps>20</numsteps>
    <steptime>2</steptime>
<!--  step time units are minutes and represent how long students have for each time
step -->
  </simtimesteps>
  <simmoney denom="USdollar">
    <seismicsetup>50000</seismicsetup>
<!--  this is setup/mobilization cost to do survey. -->
    <seismiclinear>1500</seismiclinear>
<!--  this is additional cost based on length of survey line (either N-S or E-W) per
grid cell -->
    <startcash>1000000</startcash>
    <drillcost>50000</drillcost>
    <minleasecost>500</minleasecost>
```

```
<!--  this is minimum lease bid per grid cell -->
    <oilincome>100</oilincome>
<!--  units are per barrel -->
    <gasincome>50</gasincome>
<!--  units are per 1000 cubic feet -->
<!-- Oil and gas revenue per time step determined by multiplying oil or gas rate given
in "oil" and "gas" files/tags above, with the simsteptime, with these income prices --
>
  </simmoney>
</simdata>
```