Applet based 3D Chess
Pallavi Arora
# Table of Contents

Overview.................................................................................................................................................. 2  
General rules of sandlot chess.................................................................................................................. 2  
Core Architecture of the 3D Chess Applet Implementation................................................................. 3  
Pre-requisites for testing the 3D Chess Applet program ................................................................. 9  
Testing the 3D Chess Applet Program.................................................................................................... 9  
  Test the 3D display orientation of the chessboard.............................................................................. 9  
  Test promoting the chess pieces ........................................................................................................... 9  
  Test taking out the other player’s chess piece ..................................................................................... 10  
  Test the checkmate situation............................................................................................................... 10
Overview

The Java 3D API provides a set of object-oriented interfaces that support a simple, high-level programming model one can use to build, render, and control the behavior of 3D objects and visual environments. With the Java 3D API, one can incorporate high-quality, scalable, platform-independent 3D graphics into applications and applets based on Java technology.

The purpose of this project is to demonstrate application of the Java 3D API in building an applet based Chess game. This chess game allows two players to play against each other from a single computer. The following features have been built into the applet program –

- Allow the players to customize the 3 dimensional view of the chessboard
- General rules of sandlot chess have been implemented
- Validate all moves by the players
- Display the last move by the player
- Display message when checkmate occurs

General rules of sandlot chess

The ultimate aim in the game of chess is to win by trapping your opponent's king. (This is called checkmate)

White is always first to move and players take turns alternately moving one piece at a time. Movement is required.

Each type of piece has its own method of movement. A piece may be moved to another position or may capture an opponent's piece. This is done by landing on the appropriate square with the moving piece and removing the defending piece from play.

With the exception of the knight, a piece may not move over or through any of the other pieces.

For the complete set of rules implemented in this program, please refer to the following website http://www.conservativebookstore.com/chess/grules.htm
Core Architecture of the 3D Chess Applet Implementation

The primary java package that holds the 3D Chess implementation is “pallavi.chess”. This package holds all the main classes of the 3D chess implementation along with the resource files. The sub-package “pallavi.chess.util” holds the utility classes. Please refer to the javadoc for listing of all classes in these packages.

The class “Preload” is the main Applet class that extends the “java.applet.Applet”. This class loads the J3DChess object, which in turn does all the initialization, creates the chessboard and starts the game. The J3Dclass also implements the “java.awt.event.ActionListener” to respond to events other than moving chess piece like changing the 3 dimensional view of the chessboard etc.
The J3DChess object initializes the objects, creates the chessboard and waits for the users actions. It instantiates and utilizes the LoadPiece object to load all the chess pieces from the 3ds resource files. The J3DChess object then creates the MovePiece object. This object becomes the main listener for all moves made by the players. This class extends the Behavior class provided in Java 3D API. The Behavior leaf node provides a framework for adding user-defined actions into the scene graph. The processStimulus method of the MovePiece object receives and processes a behavior's ongoing messages. The Java 3D behavior scheduler invokes a Behavior node's processStimulus method when an active ViewPlatform's activation volume intersects a Behavior object's scheduling region and all of that behavior's wakeup criteria are satisfied.
As previously mentioned, the MoviePiece object extends the behavior class provided in the Java 3D API. It also implements the CBHistory interface to validate if the correct player has moved, if there is a checkmate situation etc.
The player’s actions lead to the Java 3D Behavior object invoking the processStimulus method of the MovePiece object. The MovePiece object in turn utilizes the CBValidator object to validate the player’s move. After successful validation, the promoter object promotes the player’s chess piece to its new chessboard location. If the move resulted in taking out the other player’s chess piece, then that is handled by the RemovePiece Object.
The applet program also utilizes some utility classes. The LoadPiece object utilizes the Inspector3DS utility class to load the 3ds files. Inspector3DS is a wrapper around the Loader3DS class, which creates the chess pieces from the 3ds files. The Loader3DS class extends the com.sun.j3d.loaders.LoaderBase class.
Another important utility class utilized by the Chess Applet program is the SimpleTransformGroup. This class extends the TransformGroup class provided as a java media extension in the Java 3D API. The TransformGroup node specifies a single spatial transformation, via a Transform3D object, that can position, orient, and scale all of its children. A generalized transform object represented internally as a 4x4 double-precision floating point matrix which is used to perform translations, rotations, and scaling and shear effects. SimpleTransformGroup objects are used to encompass the different chess pieces on the board.
Pre-requisites for testing the 3D Chess Applet program

The 3D Chess Applet uses the classes provided in the Java 3D API. Therefore, it is required that the Java 3D API classes are installed in the Java runtime environment of the machine where this Applet is being tested. If the Java 3D software is not installed on the local machine, then the Applet will not load.

You can download the Java 3D 1.3.1 software from the Sun java website. [http://java.sun.com/products/java-media/3D/download.html](http://java.sun.com/products/java-media/3D/download.html)

A copy of the Java 3D software has also been kept on the server in my home directory [http://www.ece.uic.edu/~pnavlakh/java3d_exe/java3d-1_3_1-windows-i586-directx-sdk.exe](http://www.ece.uic.edu/~pnavlakh/java3d_exe/java3d-1_3_1-windows-i586-directx-sdk.exe)

Testing the 3D Chess Applet Program

The following test cases can be used as a guide to test the Chess Applet program.

<table>
<thead>
<tr>
<th>S.No</th>
<th>Test Case</th>
<th>Expected Result</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Go to <a href="http://www.ece.uic.edu/~pnavlakh/project/chess.html">http://www.ece.uic.edu/~pnavlakh/project/chess.html</a></td>
<td>Applet should get loaded and you should see the chessboard laid out</td>
</tr>
<tr>
<td><strong>Test the 3D display orientation of the chessboard</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Click on the following buttons to change the 3 dimensional orientation of the chessboard&lt;br&gt;• Up&lt;br&gt;• Down&lt;br&gt;• &lt;&lt;&lt;&lt;&lt;br&gt;• &gt;&gt;&gt;&gt;</td>
<td>The 3 dimensional view of the chessboard will change as you click on the buttons in the test case</td>
</tr>
<tr>
<td><strong>Test promoting the chess pieces</strong></td>
<td></td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>Select any white pawn by clicking on the chessboard square on which the pawn stands. (Say select the square C-2)</td>
<td>A transparent box will encompass the selected pawn indicating the selection was successful</td>
</tr>
<tr>
<td>3</td>
<td>Now, click on the square to which you want the pawn to move (Say C-4). In the first move, the pawn is allowed to move one or 2 steps ahead.</td>
<td>The transparent box will disappear and the pawn will be moved to the new chessboard square location. The message display below the chessboard will show to move: Black Below that it will show the most recent move i.e. c2-c4</td>
</tr>
<tr>
<td>4</td>
<td>Now, click on the black pawn square (D-7). Then click on the square D-5</td>
<td>The message display below the chessboard will</td>
</tr>
</tbody>
</table>
|   |   | show to move: White  
Below that it will show the most recent move  
i.e.d7-d5 |
|---|---|---|
| **Test taking out the other player’s chess piece** | 5 | Now click on the chessboard square c4. A transparent box will encompass the pawn on this square. Then, click on the square d5.  
The black pawn will disappear and the white pawn will take its place  
The message display below the chessboard will show to move: Black  
Below that it will show the most recent move  
i.e.c4xd5 |
|   |   |   |
| **Test the checkmate situation** | 6 | Select the pawn e7 by selecting the square e7 on which the pawn stands. Click on the square e6  
The message display below the chessboard will show to move: White  
Below that it will show the most recent move  
i.e.e7-e6 |
|   |   |   |
|   | 7 | Select the white queen by selecting the square d1. Now click the square a4 to move the queen to that location  
The message display below the chessboard will show to move: Black  
Below that it will show the most recent move  
i.e.Qd1-a4  
The message Check will also be displayed to the right of the to move block. |