

Nicholas A. Woodfield  
7 Briarwood Drive, Sherman, CT 06784  
(860) 350-2846  
[nicholas.woodfield@gmail.com](mailto:nicholas.woodfield@gmail.com)  
<http://www.nicholaswoodfield.com/>

## Education

---

*University of Connecticut, Storrs, CT*

**B.S.E** in Computer Science & Engineering with Honors, May 2010

Honors Thesis Title: "Development of a 3D Game Engine"<sup>1,2</sup>

Thesis Advisor: Dr. Thomas J. Peters

**Minor** in Mathematics

**GPA:** 3.528 / 4.0

## Technical Skills and Experience

---

### Languages

- Skilled in C#, Java
- Has experience in C/C++, HLSL/Cg, XML file formats

### API/Frameworks

- Microsoft .NET, XNA, DirectX
- Java SE, Swing, SlimDX
- Nvidia CUDA

### Platforms

- Windows
- Some experience with Linux

### Software

- Visual Studio 2008/2010, Eclipse, Netbeans
- Subversion (Tortoise, Subclipse, Ankhsvn)
- Photoshop 7, DeleD 3D Editor, MS Office, OpenOffice

## Work Experience

---

### Associate Software Engineer (12/2011 - Present)

Bentley Systems Watertown

- Part of Bentley's Hydraulics and Hydrology team which involves the implementation and testing of water/wastewater network design and analysis software.

### Student Researcher (06/2009 - 08/2009)

University of Connecticut, Research Experience for Undergraduates (REU) Biogrid Program

- NSF-sponsored summer research in general-purpose computing on GPU's.
- Developed algorithms in C and CUDA to detect doubly normal line segments and a global minimum separation distance on Bezier Curves and Surfaces. Observed a 200% speed increase using CUDA for curves.

---

1 [http://digitalcommons.uconn.edu/srhonors\\_theses/134/](http://digitalcommons.uconn.edu/srhonors_theses/134/)

2 <http://code.google.com/p/spark-engine/>

### **Student Programmer (01/2009 - 03/2009)**

University of Connecticut, Computer Science and Engineering Department

- Developed a phylogenetic tree drawing Java web applet for use in bioinformatics research.
- Created parsing tools for specialized implementation of a newick tree format.

## **Open Source Experience**

---

### **Tesla Graphics Engine (04/2010 – Present)**

*Website:* <http://www.tesla-engine.net>

*Sources:* <http://code.google.com/p/tesla-engine/>

Responsible for:

- Core product design and development.
- Creating auxiliary tools for content management and creation.
- Documentation writing.
- Maintaining the project's website.

Software Highlights:

- Designed an abstraction layer to support multiple graphics API's (DirectX, OpenGL, XNA) that promotes code reuse and cross-platform capabilities.
- Implemented a Vector, Quaternion, and Matrix math library.
- Software provides easy-to-use input, windowing, content management, picking/collision, and advanced rendering systems.
- Written in C#.

## **Awards and Accomplishments**

---

- Engineering Dean's List **2007-2008**
- Engineering Merit Scholarship **2006-2007**
- Roland & Carol Pampel Endowed Scholarship **2007-2009**
- Allan Brian Glickstein Scholarship **2009-2010**
- Senior Design Project: "Development of a 3D Game Using Spark Engine"
  - Worked in a team of three to develop a game that used the software developed for the honors thesis.

## Extracurricular Projects

---

### **DXS Model Converter** (August 2007)

*Website:* <http://code.google.com/p/deledimporter/>

- Format converter for the .dxf XML model format that parsed 3D data into an intermediate representation that could be outputted to different engine formats or other model formats.
- Included a plugin system that allowed users to define and parse custom tags.
- Written in Java.

### **jMonkey Model Importer** (Summer 2007)

*Website:* <http://www.nicholaswoodfield.com/projects/java-projects/jmonkeyimporter/>

- Application tool with a GUI that aided users in converting 3D models from a variety of formats (.3ds, .obj, .dae, etc) into a binary format readable for the jMonkeyEngine platform.
- Allowed for single or batch file conversions to streamline the process of importing geometry without requiring custom code or coding experience.
- GUI included a real time 3D window to allow viewing of geometry once it is converted.
- Written in Java.

## Courses

---

- CSE 123C: Computing (C/C++)
- CSE 133: OOP & Design (Java)
- CSE 134: Data Structures & Algorithms (Java)
- CSE 210W: Digital Logic Design
- CSE 2102: Software Engineering
- CSE 350: Algorithms and Complexity
- CSE 3666/4302: Computer Architecture
- CSE 3300: Computer Networks and Data
- CSE 3502: Theory of Computation
- CSE 4300: Operating Systems
- CSE 4904/4939W: Senior Project
- CSE 4100: Compilers
- CSE 470: Computer Graphics
- ECE 3101: Signals and Systems