Development Status and Prioritized To-Do List for Christoph’s WebGL Application
(rev 2 jbc, 2/27/15)

1. Priority 1: Review status of current version, as it is installed on Christoph’s server; this is the deployed version of the software. Make a list of what is working (and what is not, if there are bugs).
2. Priority 1: Connect with Christoph to make sure he is knows how to run the application and understands what you mean by each item in your list of what features are working (preferably with some simple documentation – (). After he’s tested the features you documented in (1), connect again to ask for any issues. If any major bugs, add them to the priority list. Copy a working version of the code onto this server (as a backup).
3. Priority 1: Documentation exists but will need to be updated as changes are made. Revise documentation to jive with the current deployed version.
4. Priority 2: Set up a development environment (at Willamette?) that will live beyond your tenure at the University. Check with Nik: There should be 3 instantiations of the software running at any point in time (see #6 below):
	1. deployed version on Christoph’s server
	2. deployed version on dev server (I’m not sure we need this – but it seems useful that you could run a demo of working software whenever – without having to go to christoph’s server);
	3. development version (one you are working on) on dev server.
5. Priority 2: Investigate putting Git on the development server for source code control. This will simplify editing/maintaining the application for current and future developers on this project.
6. Priority 2: Move some test files (Christoph’s data) onto the development server, and open up a version of the software that can be run by anyone; this will allow YOU to test the software, and allow for anyone interested in the software to try it out.
7. Priority 2: Complete the new navigation controls for the graphical interface that you are working on now. This will have the effect of making the tool easier to use. Tell Christoph what you are doing (easier to do over a skype call).
8. Priority 2: Allow the user to save settings.
9. Priority 3: Investigate using an additional library, jQuery, for cross-browser compatibility. This should help make the graphical interface more closely uniform over browsers that support WebGL (Chrome/Firefox).
10. Priority 4: Investigate (with Jenny/Judy/Mike/Nik) using Azure or Amazon Cloud in the longer term. The code repository should be on this server.
11. Priority 4: Looking into a strategy for implementing a file converter that reads Christoph's data and outputs a file that is faster for the application to read. This would have the effect of vastly improving the speed of the application. Note that slow file reading caused one of the bugs Kendra could not find; apache timed out the app on a file read. This is a PHP config default. Another issue to watch out for is the browser caching the file – Nik and Kendra added a timestamp that forced a read of the file – not from the cache. There is a better way to accomplish this – use a versioning scheme to change the path so old cached versions won’t load.
12. Priority 4: When software is working well on both Christoph’s server and the development server, put pointers (in the VISTAS web site) to the working version on the dev server.