

Cap over Mill - Widelands

For Build 20

SirVer:

I already implemented zoom. I want to fix the small bugs around this - mostly automatic moving using Lua which is not zoom aware. I still want to work on the proxy feature for the metasever. And I would love to look into trading to finish the features for Widelands 1.0. Also I am interested in a tablet port.

Unfortunately, I will not have too much time for Widelands in the future, so it is unclear what I can contribute going forward. All these features are therefore up for grabs if somebody else wants to work on them.

For Build 19

SirVer:

I want to merge the better lua serialization - this will break all savegames. I then want to clean up the code base - remove useless options, fix some style inconsistencies, remove compatibility code for savegames, rework the cmake build files. I also want to revisit the [stacked animations](#) idea and finish up this code. And I want to look into adding a proxy to the metasever, so that no open ports are required on the client side anymore. Oh and I have some plans for a better messages window... but that all sounds like so much already :).

Generally speaking I believe we are moving towards a 1.0 version with build 19 or at the latest build 20.

My biggest project though will be to explore the idea of a Widelands Foundation - essentially a way to sell Widelands on AppStores and collect the money for non-profit charities like Child's play or WWF or something. I hope that Widelands can do a little more good for the World than just being an awesome time sink. I started talking to lawyers if there is a good organizational structure for such an idea.

Shevonar:

1.) Create a GUI class that is responsible for handling the GUI and is called from the WlApplication main loop to do so. The Panel::run() method is removed and its purpose is split into the WlApplication main loop and the GUI class. These changes reduce the size of the WlApplication file by several hundred lines. I already have some proof of concept code on my HDD which I will push to launchpad soon™ to get some feedback. However this might conflict with the SDL2 changes in some places.

2.) The GUI class also adds the possibility to create GUI elements from XML files instead of hard coding the GUI. This makes UI adjustments for small screens and touch devices possible!

The idea is to have the whole GUI written in XML+Lua which also makes it easier to create special elements and dialog for scenarios. I already have successfully built the main menu from an XML file as proof of concept. However it is really necessary to have the GUI class in place first.

3.) To further clean up the WLApplication class and continue the work that I wanted to have done for build 18, I will try to figure out a proper way to handle command line arguments, the config file, default values and option settings.

That is everything I have planned so far. The main problem I see right is the interference with other big changes to the Widelands code, like the SDL2 migration or the ingame help (we should not create a lot of LUA help files if we decide to create a XML+LUA GUI).

Teppo Mäenpää:

- Write another AI. I would probably aim for a CPU-light version, which does not perform worse than the current one.

- Make it possible to run an entire game from command line, without UI. The purpose of this would be to characterize map starting points using AI only players and repeated matches, plus playing AIs against each other. This could also be used to find maps where the AI stalls, to make it easier to find AI weaknesses.

- Small improvements here and there. I would like to add a way to change the working triggers for various production sites to economy settings, for example.

- Add features to the editor to make symmetric maps more easy to do; either copy&paste/paste_rotatedinverted region, or through a possibility to multi-apply each user action. I am a bit unsure on how this should be done.

It is 100% certain that I will not do all the above. The most likely outcome is that I do none of the above; there is also a fair chance that I manage to get some of the above done. Feedback on which of the above would be improvements from your points of view too is appreciated.

I do not intend to reserve any topics. If you are interested, feel free to improve the odds of a working solution by doing it yourself! An e-mail reducing odds of duplicate work would be nice.

Nasenbaer:

* I would really like to work on the proxy feature with Holger to finally remove the port forwarding, firewall opening and ipv4 barriers our current internet gaming engine has. This is more interesting than ever to me, as I am currently connected only via IPv6 and Widelands does not support pure IPv6 for opening a game.

* Further seafaring needs some love (or better to say war) to implement a solution for island take over.

* And finally I changed my mind concerning the auto tree spreading feature. I am still a big fan of the terrain affinity that gives a lot to the look and makes foresters less efficient on e.g. mountain terrain. However I must agree with those of you who argued that auto spreading trees lead to bigger problems than the improvements to the game. So I would be glad to disable this feature again if the majority of you agree on this point.

(Small disclaimer: This however does not change the need for a "tree removing feature" of expeditions, as else players could shut up their islands by just placing trees over the whole beach.

For Build 18

SirVer:

I want to continue my work on the atlanteans campaign, the editor and the stacked animations feature I was thinking about for a while. I will be unemployed starting from June till September - in this timeframe I plan to invest a lot of time into widelands and report my daily/weekly status on my blog. I hope I will be able to make this happen - I have other obligations in this time as well - moving to Munich and getting married beside others. I will announce this as news post when I start.

Thoughts after release: I did not work much on the stuff I set out to do. But I added a test suite, fixed tons of bugs, added a new text renderer, implemented a better Lua integration. I also wrote a new metaserver and changed how the website is deployed (and moved the server).

Chuck:

- the continued support of the ever-evolving seafaring facet
- completion of the 3D models for those elements currently without them (buildings, critters, etc.)

- speaking of 3D, a concerted effort to migrate from Blender 2.49
- providing and upgrading animations for working buildings and hopefully getting more workers visible onsite
- the conversion to stacked animations
- more work on terrain and trees (including stumps perhaps) ;)
- a host of other things that I have probably forgotten or have yet to discover, and...
- maybe a surprise here and there, though they likely will not be as big as moving my household or getting married (been there, done that) You have my best wishes, SirVer!
:D

Nasenbaer:

Anyways I hope to get the seafaring expedition and colonization feature in Widelands step by step.

Shevonar:

My plans for build18 can be summarized as "Improving the overall user experience". What do I mean by that? Well currently Widelands has many nice features but for new players it is quite hard to get started. It is even harder to get involved in the development. Although there are a lot of inconsistencies which naturally arise in such a long development. But what do I want to do in concrete:

- * First I want to complete the work on the new website design, which is clearer and more structured IMO. [1]
- * In a second step I will try to improve the wiki. Currently it is hard to find the information you are looking for or you are flooded with too much information.
- * Add new features to the website and fix bugs which were ignored for a long time.

So far it doesn't really belong to the game itself but I think this is also very important because the website is the first thing a possible player sees.

On Widelands game I want to do tackle the following issues:

- * The OpenGL rendering requires a GLSL implementation to work properly and fast on modern systems [2]
- * The refactoring of campaigns (I started this a while ago) [3]
- * The refactoring of options and option menus [4]
- * Improve the new ingame help and try to join it with the website help

* Bugfixes

By writing down all this stuff that I want to do I realized that it is really much. Some of that might have to wait for build19...