



The Playground: Wesnoth

OSM2011
Kaido Kikkas

The Battle for Wesnoth

- A turn-based, fantasy-themed strategy game
- Founded by David White in 2003, now approaching version 1.10 (we use 1.8 stable)
- Influences: Master of Monsters, Warsong
- Fully free and open-source software under GPL
- Cross-platform (Linux, MS Windows, MacOS X, ...)
- Both single- and multiplayer
- Available in 50 languages (~10 are rather complete)
- Extensive and active community at wesnoth.org

Basics

- Played on a hex map with several dozens of terrain types (influence movement and defense)
- Hundreds of units in various (6 in default era) factions, each with distinct strengths and weaknesses, 2-5 levels (mostly 3), movement...
- Day and night cycle vs the unit alignment
- Economy based on gold, generated by villages
- In campaign mode, surviving units from previous scenarios can be recalled
- Great variety of missions

Why use a game in a serious course?

- Homo ludens – or Playful Cleverness
- Like web design, should provide 3 skillsets:
 - Technology: coding (programming)
 - Design: artwork, maps, overall aesthetics
 - Content: scenarios and campaigns
- Compared to web design:
 - Technology is more prominent – includes event-driven programming in addition to markup (comparable with HTML+Javascript)
 - Content is more affected by the other two (or vice versa)

What can be taught with Wesnoth?

- Storytelling and expression, overall creativity
- Graphical design
- Animation
- Markup (a possible step before moving to XHTML, XML, AJAX etc)
- Event-driven programming
- NB! Easier to 'sell' to non-technical students!
- All these can be mixed and balanced according to the audience

Campaign building

- Write the storyline, design major events and divide them into scenarios
- Choose/build units for main characters
- For each scenario
 - Design (objectives, events)
 - Draw the map (considering terrain and starting points)
 - Choose units and recruitment scheme
 - Code the scenario
- Code the campaign summary
- Test and balance

Storyline

- Wesnoth has its own fictional history
http://www.wesnoth.org/wiki/History_of_Wesnoth
as well as geography:
http://www.wesnoth.org/wiki/Geography_of_Wesnoth
- Etymology: Wesnoth <= 'wes noth' <= 'west north'
- Several interesting attempts have been made to base totally different storylines on the Wesnoth engine
- Eric S. Raymond has written the Wesnoth campaign design How-To:
<http://www.catb.org/~esr/wesnoth/campaign-design-howto.html>

Maps

- Wesnoth has an easy-to-use map editor
- Maps can also be edited by hand (if you know the terrain codes) via any text editor – they are ordinary text files

- Example:

```
_off^_usr , Mm^Vhh , Rd , Rd  
_off^_usr , Gs^Fp , Gs^Fp , Gs^Fp
```

Units

- Several hundreds are available, distributed under different eras (historic sets)
- At first, choosing among ready-made ones is more than enough. The unit list for v1.8 is at <http://www.wesnoth.org/units/1.8/C/mainline.html>
- To create new units, tutorials are available at http://www.wesnoth.org/wiki/Art_Tutorials
- Unit design can be quite complex in newer versions of the game (shadows, animations, etc)

Wesnoth Markup Language

- Rather similar to XML
- Tags written in the form of [tag]
- A typical scenario consists of
 - Various metadata in the beginning
 - Storytelling part explaining the situation
 - Map data (can be included, but is typically linked)
 - Day/night cycle and difficulty specifications
 - Prestart part (definitions of sides, objectives etc)
 - Event-based buildup for the scenario

WML...

- Can be very simple, but allows for really complex operations (changing units and terrain on the fly – e.g. a character can be magically turned into a monster or a cave wall may open under a spell or password; events may also depend on if a character from a previous scenario is alive etc etc)
- Teaches good structure (opening and closing tags, correct use of parameters) as well as 'the big picture' (campaign level)

Final words

- Wesnoth is a versatile tool for teaching a number of things
- Fully open in nature => may be used to develop new things without messing with the 'IP' stuff
- Sometimes people learn better from playing :)



</Wesnoth_intro>
