

PUBLIC

albertoruibal / carballo

★ Star12

Fork2

Code

Network

Pull Requests0

Issues0

Graphs

Carballo Chess Engine

ZIP

HTTP

SSH

Git Read-Only

https://github.com/albertoruibal/carballo.git

Read-Only access

branch: master

Files


Commits

Branches1

Tags

carballo

79 commits

Reformat		
	albertoruibal authored 16 days ago	latest commit 2544aa3682
applet	25 days ago	Dependencies are now to projects, not via local maven repo. Moved som... [albertoruibal]
core	16 days ago	Reformat [albertoruibal]
csharp	24 days ago	Added/removed files to c# project [albertoruibal]
gwt	20 days ago	IntelliJ optimizations [albertoruibal]
jse	20 days ago	IntelliJ optimizations [albertoruibal]
testresults	25 days ago	Updated WinatChess tests results [albertoruibal]
webtvchess	20 days ago	IntelliJ optimizations [albertoruibal]

 build.gradle	25 days ago	Dependencies are now to projects, not via local maven repo. Moved som... [albertoruibal]
 carballo-uci.tgz	a year ago	uci binaries [albertoruibal]
 carballo.png	a year ago	Initial GIT conversion [albertoruibal]
 license.txt	a year ago	Initial GIT conversion [albertoruibal]
 readme.md	20 days ago	IntelliJ optimizations [albertoruibal]
 settings.gradle	25 days ago	Added 'applet' to projects [albertoruibal]

readme.md

Description

Carballo (the galician word for Oak, well it's all about search trees) is an Open Source Java and C# chess engine with Applet and GWT (HTML5) interfaces.

It's organized into modules:

- Core: the chess engine
- JSE: the Java Standard Edition version with an UCI interface and unit tests
- Applet: the applet code, depends of Core and Jse
- GWT: an HTML5 interface, depends of Core
- CSharp: this is a conversion of the Core source code to C# using Sharpen

Links:

- Home: <http://www.alonsoruibal.com/chess>
- Applet interface: <http://www.mobialia.com/webchess>
- GWT interface: <http://www.mobialia.com/webtvchess>
- Source code: <https://github.com/albertoruibal/carballo>
- UCI binary: <https://github.com/albertoruibal/carballo/raw/master/carballo-uci.tgz>

It is licensed under GPLv3, and you are free to use, distribute or modify the code, we ask for a mention to the original authors and/or a link to our pages.

Features

- Simple and clear code
- Cute drag and drop Java Applet GUI, to fit in web sites
- Includes also a great GWT interface by Lukas Laag
- Gadle source code organization
- JUnit used for testing, multiple test suites provided (Perft, BT2630, LCTII, WAC, etc.)
- Based on Bitboards (not so complicated as other people say)
- State-of-the-art magic bitboard move generator (doubles the basic move generator speed!), also code for magic number generation
- PVS searcher
- Iterative deepening
- Aspiration window, moves only one border of the window if falls out
- Transposition Table (TT) with Zobrist Keys (two zobrist keys per board, to avoid collisions) and multiprobe/two tier
- Quiescent search with only good captures (according to SEE) and limited check generation
- Move sorting: two killer move slots, SEE, MVV/LVA and history heuristic

- Also Internal Iterative Deepening to improve sorting
- Fractional Extensions: check, pawn push and passed pawns, mate threat, recapture (2 = 1PLY)
- Reductions: Late Move Reductions (LMR)
- Pruning: Null Move Pruning, Static Null Move Pruning, Futility Pruning and Aggressive Futility Pruning
- Polyglot Opening Book support; in the code I include Fruit's Small Book
- FEN notation import/export support, also EPD support for testing
- Pluggable evaluator function, distinct functions provided: the Simplified Evaluator Function, other Complete and other Experimental
- Parametrizable evaluator (only for the complete & experimental evaluators)
- Contempt factor
- UCI interface with lots of UCI options (for chess GUIs like Arena)
- The core of the chess engine was converted to C# using Sharpen

It scores 2415 ELO points at BT2630 tests in my Core2 Duo 2.2GHz. Also solves 280 positions of the 300 WinAtChess test (5 seconds for each). His real strength is about 2100 ELO points.

Test Results

I made a Java Engines Tournament to compare Carballo against other chess engines at tournament time 5 minutes. Here are the results:

Engine	Score	Cu	Ca	Ol	Me	Ca	Fr	Br	Ar	S-B
1: Cuckoo-1.12	34,5/35	1=111	11111	11111	11111	11111	11111	11111	515,75
2: Carballo-0.7.11	23,5/35	0=000	110=0	11111	10101	1111=	01101	11111	318,00
3: OliThink-5.3.2	22,5/35	00000	001=1	00110	11101	11011	11111	11111	274,75
4: Mediocre-0.3.4	17,5/35	00000	00000	11001	11001	01011	11=11	11101	210,50
5: Carballo-0.5	15,0/35	00000	01010	00010	00110	=1100	010=1	11111	184,50

```
6: FrankWalter-1.0.8    14,0/35 00000 0000= 00100 10100 =0011 ..... 11011 10111 158,75
7: Bremboce-0.6.2      8,0/35 00000 10010 00000 00=00 101=0 00100 ..... 00011 117,25
8: ArabianKnight-1.0.9 5,0/35 00000 00000 00000 00010 00000 01000 11100 ..... 55,50
```

140 games played / Tournament is finished

Level: Tournament Game in 5 Minutes

Hardware: Intel(R) Core(TM)2 Duo CPU T7500 @ 2.20GHz 2200 MHz with 752 MB Memory

Authors

- Alberto Alonso Ruibal: <http://www.alonsoruibal.com>
- Lukas Laag, developer of a great GWT SVG library (<http://www.vectomatic.org>) and the Carballo GWT interface

History

Version 0.8:

- Project build system migrated from Maven to Gradle
- Solved some evaluator bugs
- New option to do not use Magic Bitboard Attacks: optimizes start time in HTML5
- Removed specific Bitboard attacks code from GWT, can be simulated setting BitboardAttacks.USE_MAGIC = false
- New ArrayBufferBook for GWT, can process any opening book loaded as a JS ArrayBuffer

Version 0.7: A small leap on the engine performance and a big code clean

- Code moved to github

- Integrated ROOT, PV and NULL nodes search routine
- Activated singular movement extensions and changed default singular extension margin
- Do null move only when the remaining depth is > 3 PLY
- No not overwrite the value in the TT if there is no room
- Converted code to C# using sharpen. At the moment only the core of the engine
- Solved a big bug getting the move from the transposition table
- Also found another bug on the search getting the last captured piece value
- And the complete evaluator had a bug calculating the attacks value

Version 0.6: Source code reorganization, GWT and PGN improvements, no changes on the engine code

- Code splitted in carballo-core, carballo-jse and carballo-applet
- Carballo-core is GWT-friendly
- Integrated SAN notation on Board class
- Improved PGN export with SAN notation
- Added a GWT interface based on the one by Lukas Laag (<http://vectomatic.org>) code

Version 0.5: Improves about 150 ELO points over Carballo 0.4

- PVS searcher: SearchEngine completely changed
- Futility pruning now works!
- New TT algorithm, now also uses TT to store evaluation values
- Bug with draw detection with 3-fold repetition
- Bug with time management on tournament, was using the opponent's time amount
- Bug with history table overflow

Version 0.4: First version integrated with Mobialia Chess

- Parametrizable evaluator

- [Evaluator changes](#)

GitHub

[About us](#)
[Blog](#)
[Contact & support](#)
[GitHub Enterprise](#)
[Site status](#)

Applications

[GitHub for Mac](#)
[GitHub for Windows](#)
[GitHub for Eclipse](#)
[GitHub mobile apps](#)

Services

[Gauges: Web analytics](#)
[Speaker Deck: Presentations](#)
[Gist: Code snippets](#)
[Job board](#)

Documentation

[GitHub Help](#)
[Developer API](#)
[GitHub Flavored Markdown](#)
[GitHub Pages](#)

More

[Training](#)
[Students & teachers](#)
[The Shop](#)
[Plans & pricing](#)
[The Octodex](#)

[Terms of Service](#) [Privacy](#) [Security](#)



© 2013 GitHub, Inc. All rights reserved.