

KnightCap is an open source computer chess engine. It's primary author is Andrew Tridgell and it was created circa 1996. Major contributions have also been made by Jon Baxter and probably minor contributions by a few others.

In most ways, KnightCap is a fairly typical modern program. It uses bitboard data structures that are slightly different to those that were well known in 1996, but obvious enough and probably well known now. There is backward pruning using mtd-f (a method approximately equivalent to Alpha-beta_pruning but slightly more efficient in some settings). There is Null-move_heuristic. There is a fairly complex end-node evaluation process that considers similar features to other programs.

In addition, KnightCap has support for multi-processor computers, in particular the now obsolete Fujitsu CAP computer research machines.

The most original feature of KnightCap, introduced in the late 1990s, was an experiment in temporal difference learning as applied to chess. This technique allowed KnightCap to automatically tune the weights applied to the various features in its evaluation function based on the games it played.

For a while in the 1990s, KnightCap was quite active on chess servers on the Internet, but it is now semi-retired and rarely seen. It's strength is below that of the strongest programs, but still quite good.[citation needed]