HICAP (Hierarchical Interactive Case-Based Architecture for Planning)

HICAP, a general-purpose, interactive case-based plan authoring architecture that can be applied to decision support tasks to yield a hierarchical course of action. It integrates a hierarchical task editor with a conversational case-based planner. HICAP maintains both a task hierarchy representing guidelines that constrain the final plan and the hierarchical social organization responsible for these tasks. It is the first tool to combine a doctrine-guided task decomposition process with a case-based reasoning approach to support interactive plan formulation. Thus, it yields a plan that benefits from previous experiences and is sound according to doctrine. Furthermore, HICAP supports experience sharing, which allows planners to exploit knowledge from other planning experts. It also supports bookkeeping, which is crucial for real-world large-scale planning tasks. By selecting tasks corresponding to the hierarchy's leaf nodes, users can activate the conversational case-based planner to interactively refine guideline tasks into a concrete plan. Thus, HICAP can be used to generate context sensitive plans and should be useful for assisting with planning complex tasks such as noncombatant evacuation operations.

STRIPS (Stanford Research Institute Problem Solver)

The Stanford Research Institute Problem Solver (STRIPS) is an automated planning technique that works by executing a domain and problem to find a goal. With STRIPS, you first describe the world. You do this by providing objects, actions, preconditions, and effects. These are all the types of things you can do in the game world. Once the world is described, you then provide a problem set. A problem consists of an initial state and a goal condition. STRIPS can then search all possible states, starting from the initial one, executing various actions, until it reaches the goal. A common language for writing STRIPS domain and problem sets is the Planning Domain Definition Language (PDDL). PDDL lets you write most of the code with English words, so that it can be clearly read and well understood. It’s a relatively easy approach to writing simple AI planning problems.

HTN (Hierarchical Task Network) Planner

In artificial intelligence, the hierarchical task network, or HTN, is an approach to automated planning in which the dependency among actions can be given in the form of networks. In a hierarchical task network (HTN) planner, the planner’s objective is described not as a set of goal states but instead as a collection of tasks to perform. Planning proceeds by decomposing tasks into subtasks, subtasks into sub-subtasks, and so forth in a recursive manner until the planner reaches primitive tasks that can be performed using actions similar to the actions used in a classical planning system. To guide the decomposition process, the planner uses a collection of methods that give ways of decomposing tasks into subtasks. In HTN planning, the world and the basic actions that can be performed are represented in a manner similar to the representations used in STRIPS (Fikes et al. 1971; Chapman 1987). Each “state” of the world is represented as a collection of atoms, and operators are used to associate effects to actions (primitive tasks). The fundamental difference between STRIPS-style planning and HTN planning is the representation of “desired change” in the world.

References:

http://www.aiai.ed.ac.uk/project/plan/ooc/
http://rakaposhi.eas.asu.edu/learn-plan-full.pdf
https://www.aaai.org/ojs/index.php/aimagazine/article/view/833/751
http://www.cs.umd.edu/~nau/papers/munoz-avila1999hicap.pdf

https://link.springer.com/chapter/10.1007/3-540-48508-2_21
https://en.wikipedia.org/wiki/Hierarchical_task_network
http://www.primaryobjects.com/2015/11/06/artificial-intelligence-planning-with-strips-a-gentle-introduction/
http://www.cs.umd.edu/~nau/papers/erol1994umcp.pdf
https://pdfs.semanticscholar.org/7bbb/657db5a542a8a514314eda8b2f9635a2af72.pdf