In this document, I summarize the 3 major advances in AI planning described in the article referenced, and discuss their implications and influence over the AI planning field.

References:

[1] - AI Planning Historical Developments, Shrott 2017

1) STRIPS (1971) [1]

In 1971, Fikes and Nilsson came up with a framework called STRIPS that allows to formulate any planning problem. A search problem in STRIPS is described as an initial world model, a set of operators that can transform a world model into another one, and a goal world model. The goal of the solver is to find the sequence of operators, ie a plan, to get from the initial to the goal world model.
Available operators are put together in schemata, and are defined by preconditions regarding a prior state of the world, and effects on the given state of the world. [1]

STRIPS influenced AI planning in that it allowed to define any kind of planning problem with a common language. This was a first step towards creating general problem solving algorithms and strategies, and porting concepts used to solve one specific problem into solving different problems. A common language for AI planning was born.

2) Planning graphs (1997) [1]

Blum and Furst built upon concepts introduced by STRIPS to tackle AI planning by the means of Planning Graphs. They introduced GraphPlan for this purpose. The concept of GraphPlan is to use constraints to reduce the extent to which the search space needs to be explored. But as the search space is exponential [1], it is still very expensive to build. GraphPlan is guaranteed to find the optimal plan in terms of length of the plan. An essential concept used by GraphPlan to reduce the search space is the use of mutex relationships between actions. Two actions are mutually exclusive if no valid plan could make both simultaneously true [1]. Thanks to the detection of mutexes, GraphPlan reduces the amount of nodes needed to be traversed to find a valid plan.

The introduction on GraphPlan was influential in the area of AI planning as it allowed to make parallels with knowledge in graph theory, e.g. using known exploration strategies as breadth-first or depth-first.

3) Heuristic Search Planner (HSP) (1998) [1]

HSP uses the idea of heuristic search and applies it to planning problems. Heuristics are estimates of distance to the goal in every state of the problem world. It is important for heuristics to be optimistic. A common way to define good heuristics is by relaxing the problem, but often this makes the heuristic MP-hard [1].

HSP transforms the problem into a heuristic search problems thanks to the STRIPS definition, and works iteratively towards the goal state by estimating the distance to the goal with every step it takes.

HSP was influential to AI planning as it allowed to find heuristics automatically, without the need of a human understanding the problem deeply and coming up with good heuristics.