

HDDL

The IPC 2020 uses the HDDL language as its input language. The details of the language are described in the following paper:

Daniel Höller, Gregor Behnke, Pascal Bercher, Susanne Biundo,

Humbert Fiorino, Damien Pellier and Ron Alford.

HDDL - A Language to Describe Hierarchical Planning Problems.

In Proceedings of the 34th AAAI Conference on Artificial Intelligence (AAAI 2020), pp. 6-17. 2020.

A PDF version of the paper is available here: <http://gki.informatik.uni-freiburg.de/papers/hoeller-etal-aaai20.pdf>

HDDL - Addendum

In addition to the language as defined in the paper, we make the following additions

1. Forall

Preconditions and effects may contain the forall-quantor as used in PDDL. We guarantee that no negation will act on the forall-quantor, i.e. any negation will be only inside the quantor. Note that forall-quantors can contain conjunctions inside of them. Further, forall-quantors may only occur as preconditions and not in effects.

2. Type names

Note that the type name `Object` is a valid type name that carries no special meaning, i.e. it is not necessarily the root of the type hierarchy. `Object` may be declared to be the sub-type of another type. Declaring constants to be members of a type `Object` without declaring the type `object` is also illegal.

3. Requirements

Parsing and processing requirements is strictly optional. We will however guarantee for the IPC domains that appropriate requirement tags are present in every domain. We will use the following requirements

- `:negative-preconditions` – that the domain may contain a negative precondition
- `:hierarchy` – that the domain contains an HTN structure (present in all domains)
- `:typing` – that types are declared in the domain
- `:equality` – that `=` may be used as a predicate in the domain
- `:universal-preconditions` – that `forall` may be used in a precondition
- `:method-preconditions` – indicating that methods may contain method preconditions

4. Requirements not occurring in the competition

We further define the following two requirements for the future, which however do not occur in the competition:

- `:existential-preconditions` – that `exists` may be used in a precondition
- `:universal-effects` – that `forall` may be used in an effect