

I *Solver*

- libraries: Libraries
 - flags: FlagStore
 - staticKb: Theory
 - dynamicKb: Theory
 - operators: OperatorSet
 - inputChannels: InputStore
 - outputChannels: OutputStore
-
- standardInput: InputChannel<String>
 - standardOutput: OutputChannel<String>
 - standardError: OutputChannel<String>
 - warnings: OutputChannel<PrologWarning>
-
- solve(goal: Struct, options: SolveOptions): Sequence<Solution>
 - solveList(goal: Struct, options: SolveOptions): List<Solution>
 - solveOnce(goal: Struct, options: SolveOptions): Solution
-
- clone(): Solver

I *MutableSolver*

- loadLibrary(library: AliasedLibrary)
 - unloadLibrary(library: AliasedLibrary)
 - setLibraries(libraries: Libraries)
-
- loadStaticKb(theory: Theory)
 - loadStaticClauses(clauses: Iterable<Clause>)
 - appendStaticKb(theory: Theory)
 - resetStaticKb()
-
- loadDynamicKb(theory: Theory)
 - loadDynamicClauses(clauses: Iterable<Clause>)
 - appendDynamicKb(theory: Theory)
 - resetDynamicKb()
-
- assertA(clause: Clause)
 - assertZ(clause: Clause)
-
- retract(clause: Clause): RetractResult<Theory>
 - retractAll(clause: Clause): RetractResult<Theory>
-
- setFlag(name: String, value: Term)