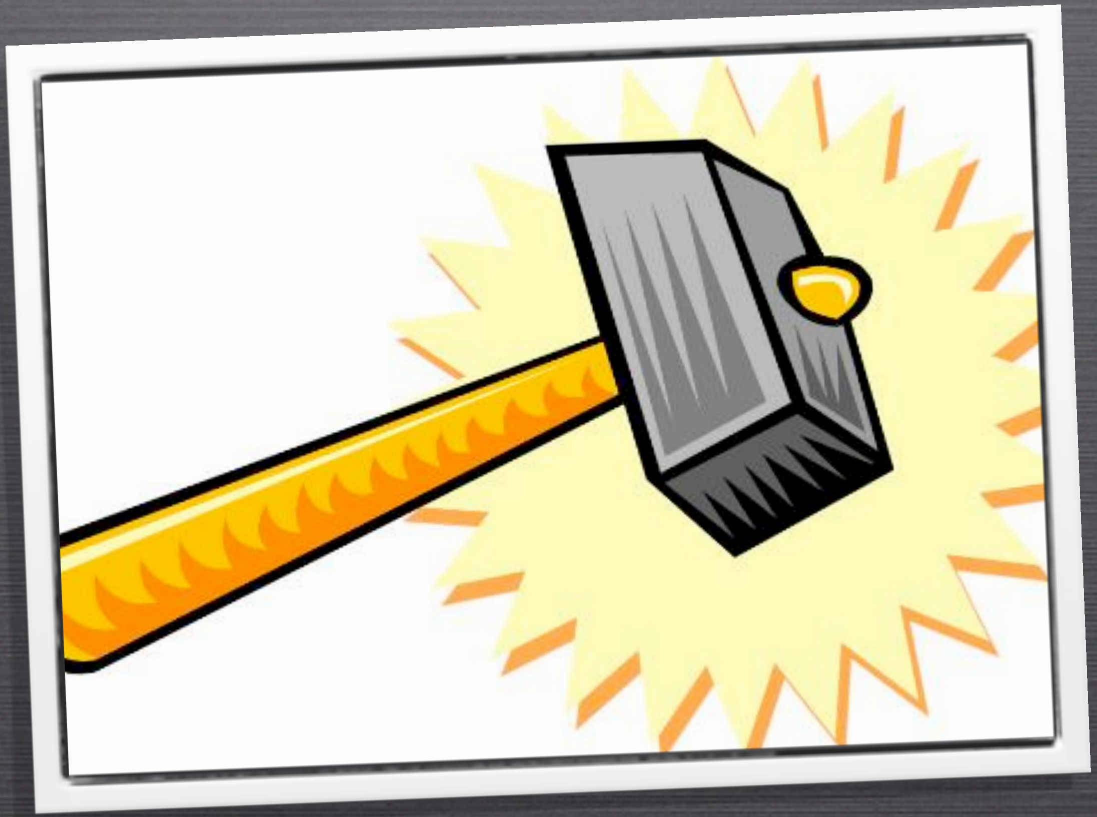
Sledgehammer
A Link between Interactive and Automatic Theorem Provers



Jasmin C. Blanchette
Technische Universität München

Larry Paulson

Jia Meng

Kong Susanto

Claire Quigley

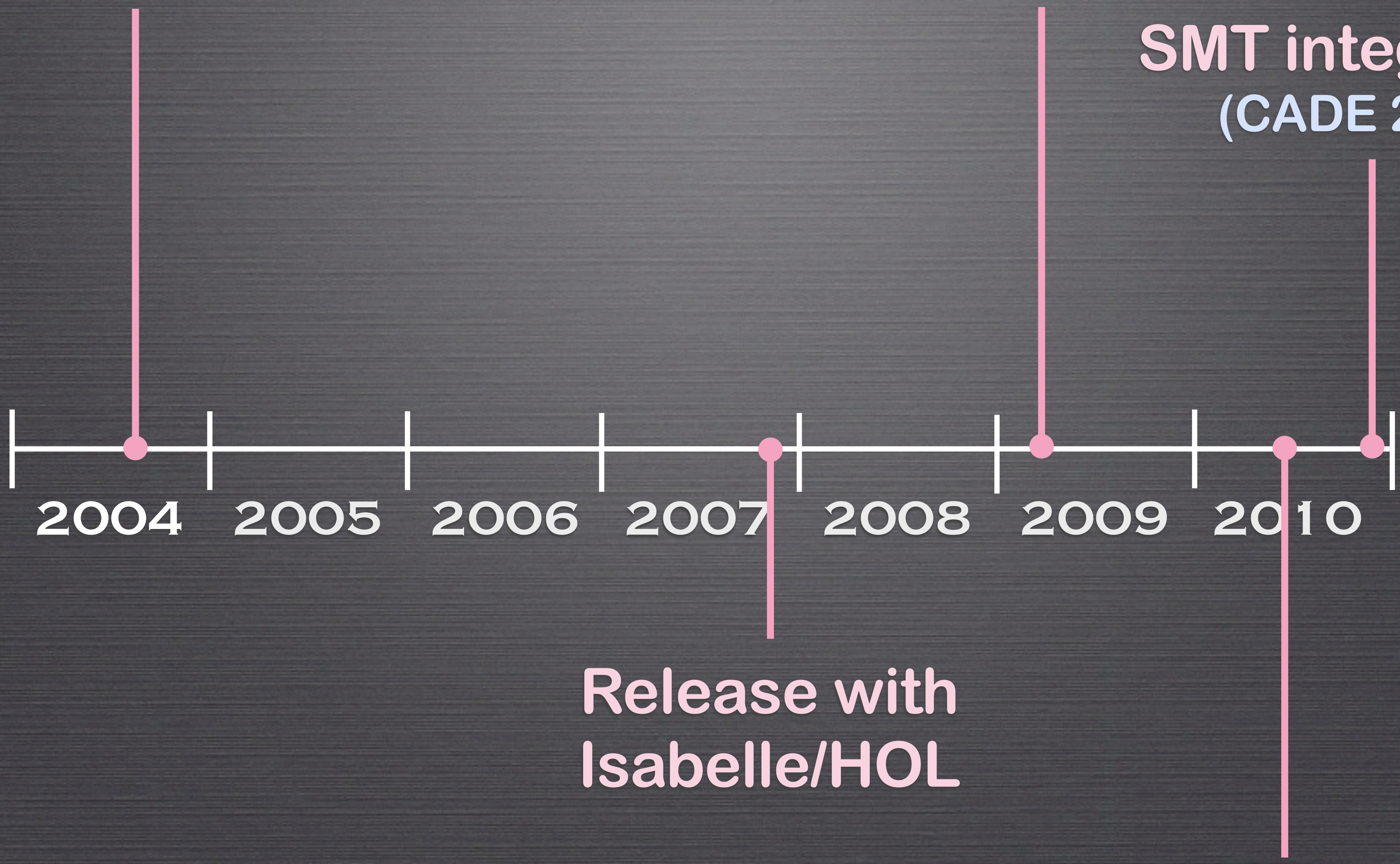
Markus Wenzel

Fabian Immler

Philipp Meyer

Sascha Böhme

2004 2005 2006 2007 2008 2009 2010



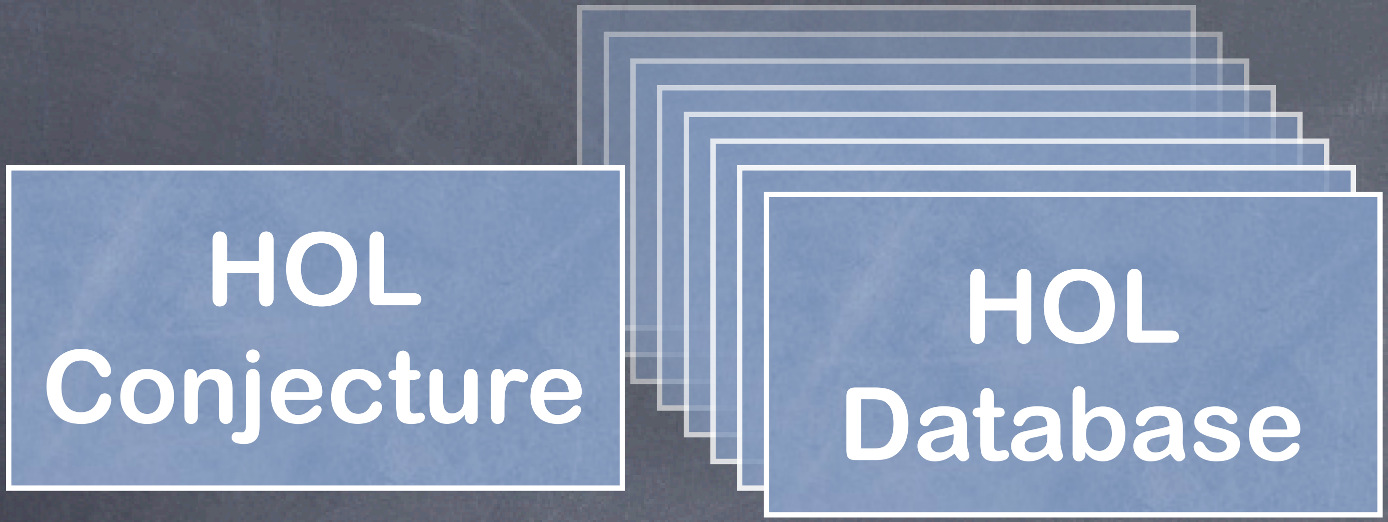
First experiments
(IJCAR 2004)

Release with
Isabelle/HOL

SystemOnTPTP
support

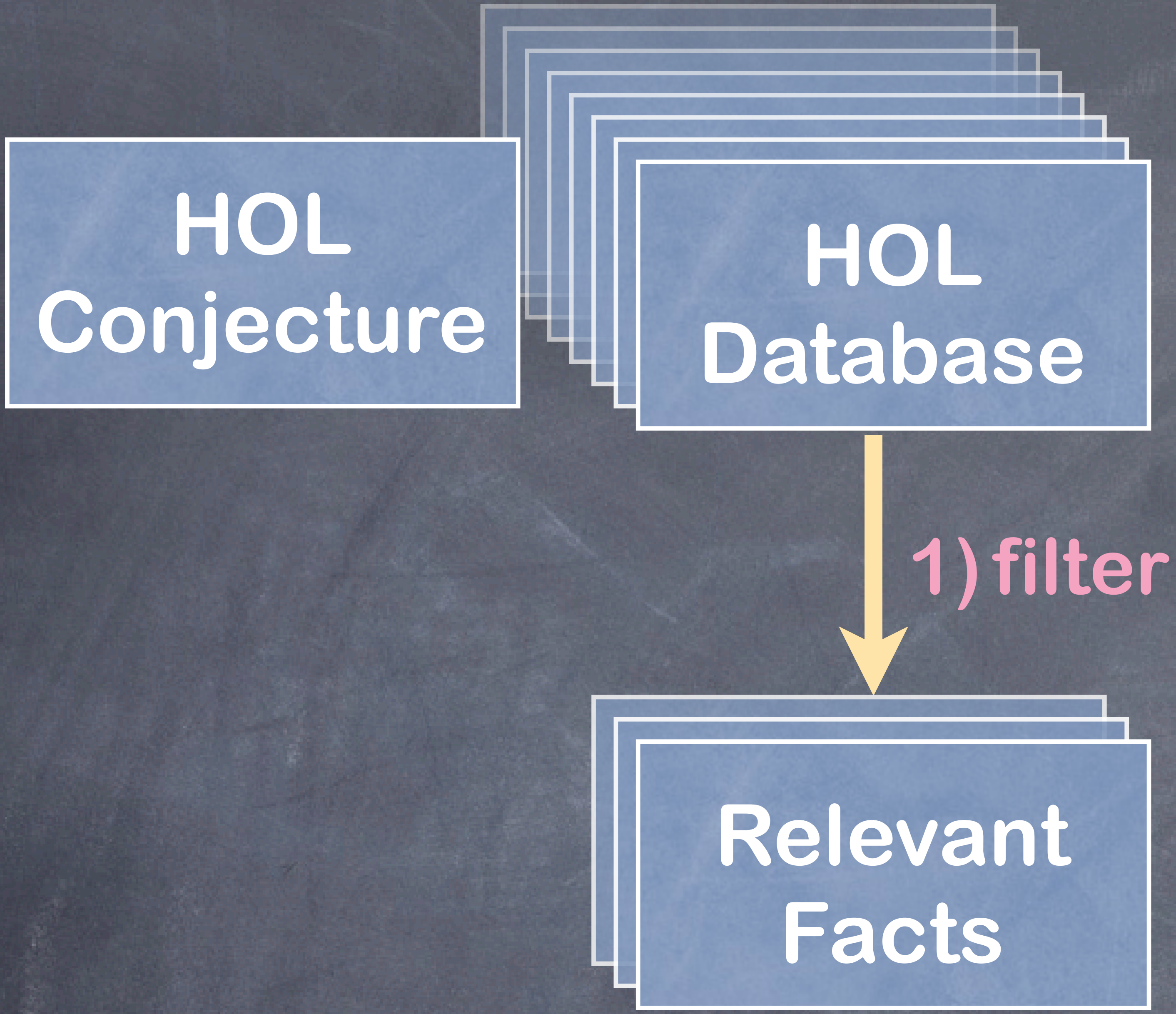
"Judgement Day"
(IJCAR 2010)

SMT integration
(CADE 2011)



HOL Conjecture

HOL Database

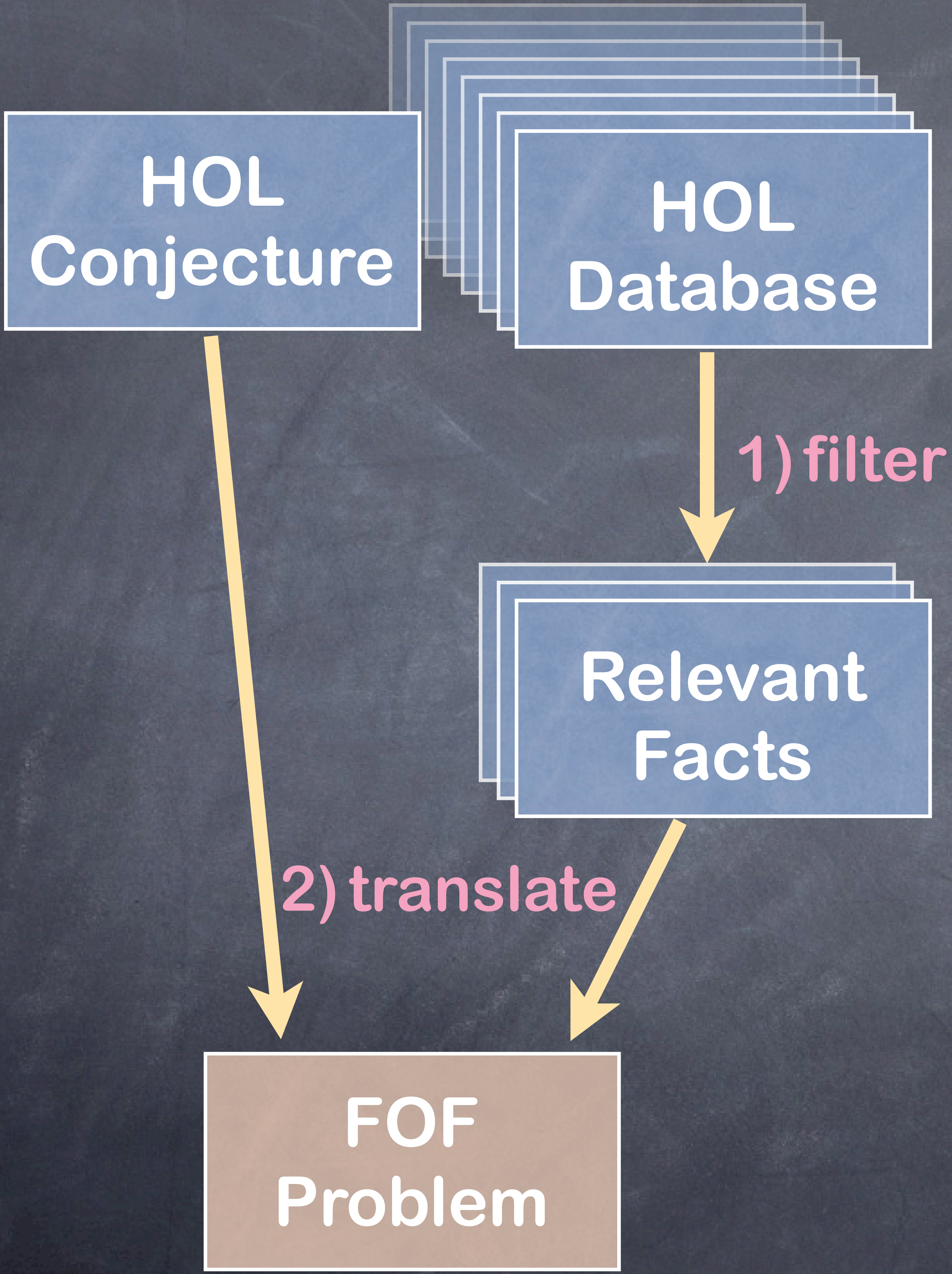


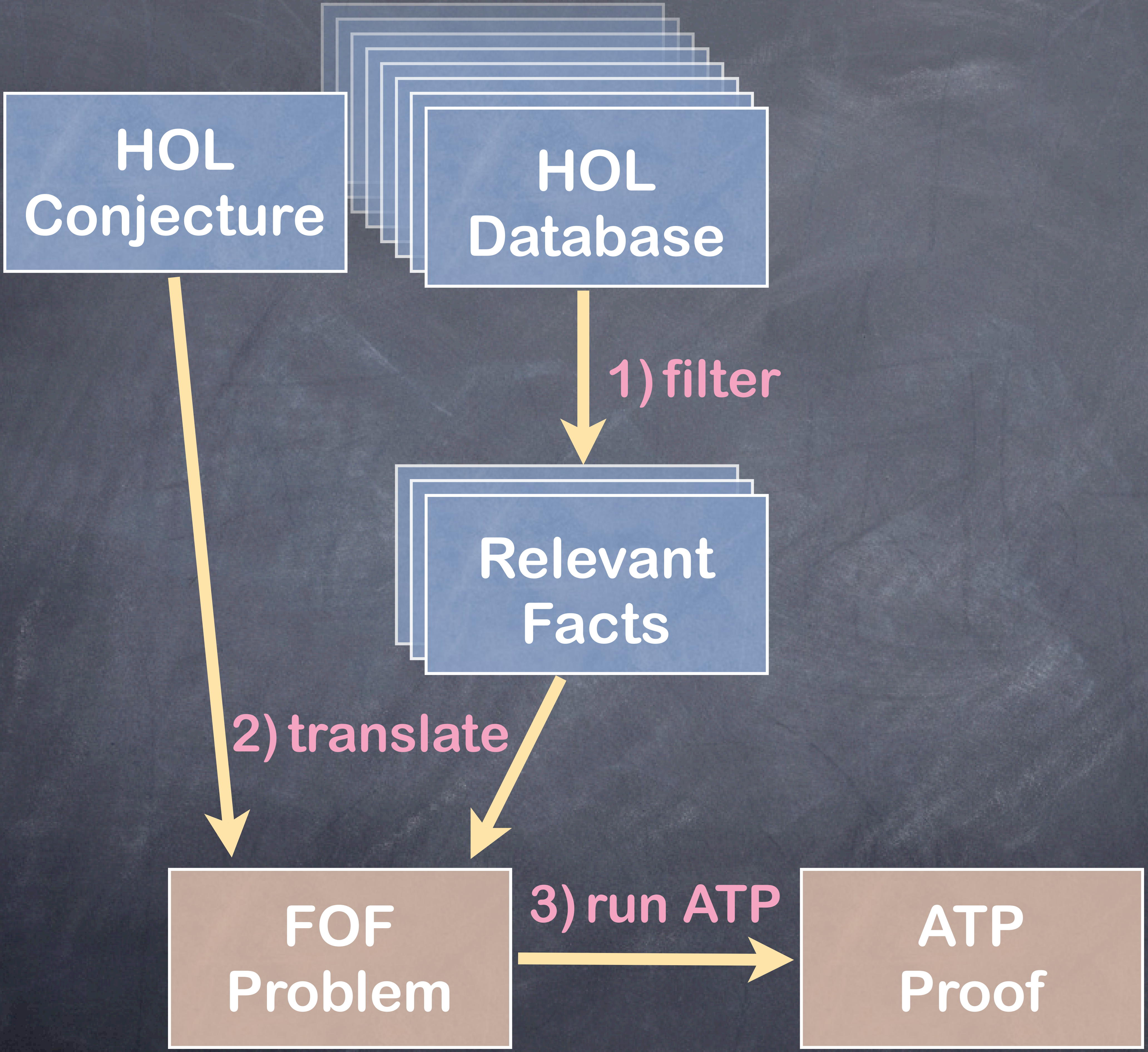
HOL Conjecture

HOL Database

1) filter

Relevant Facts


HOL Conjecture
HOL Database
1) filter
Relevant Facts
2) translate
FOF Problem



HOL Conjecture

HOL Database

1) filter

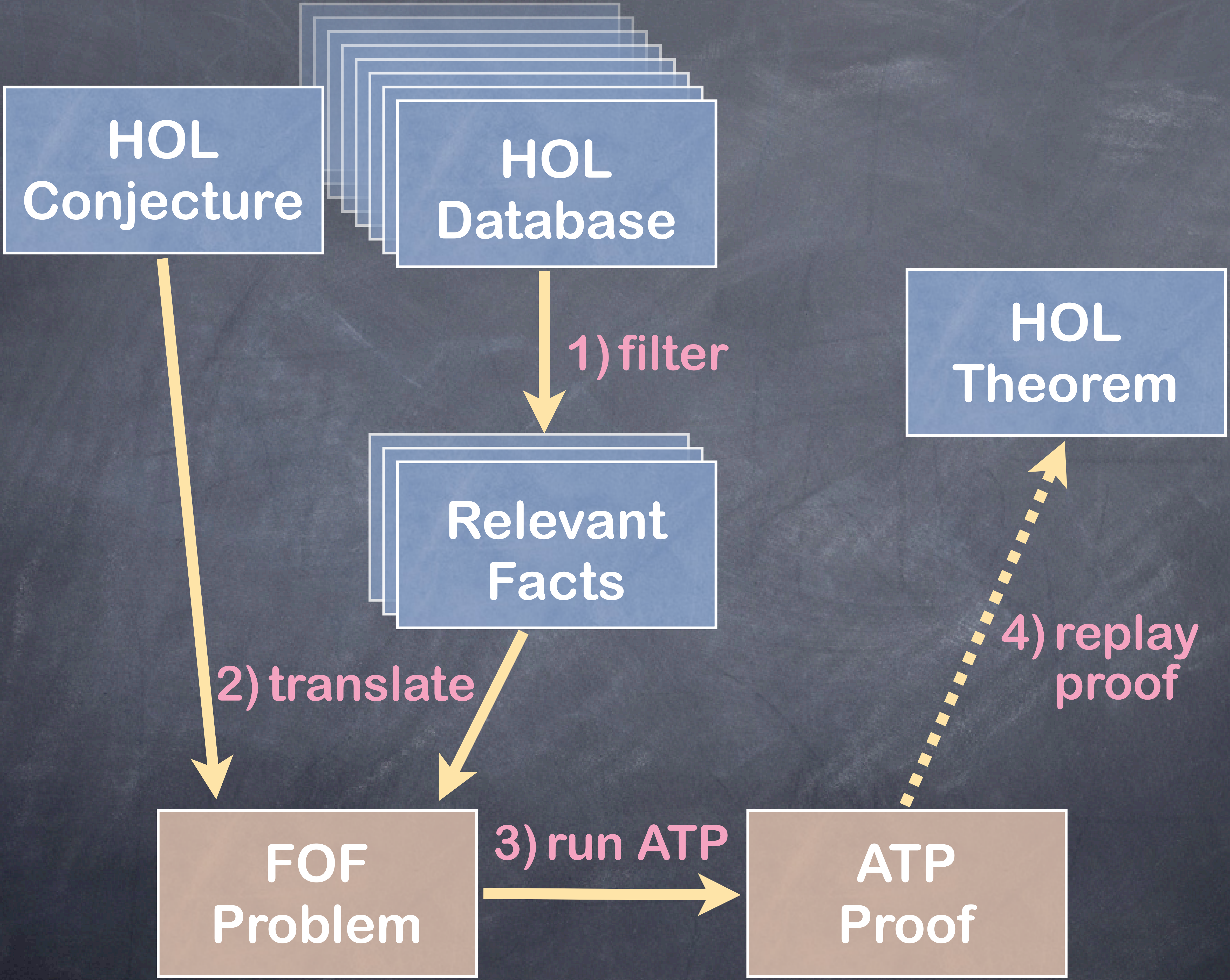
Relevant Facts

2) translate

FOF Problem

3) run ATP

ATP Proof


HOL Conjecture

HOL Database

1) filter

Relevant Facts

2) translate

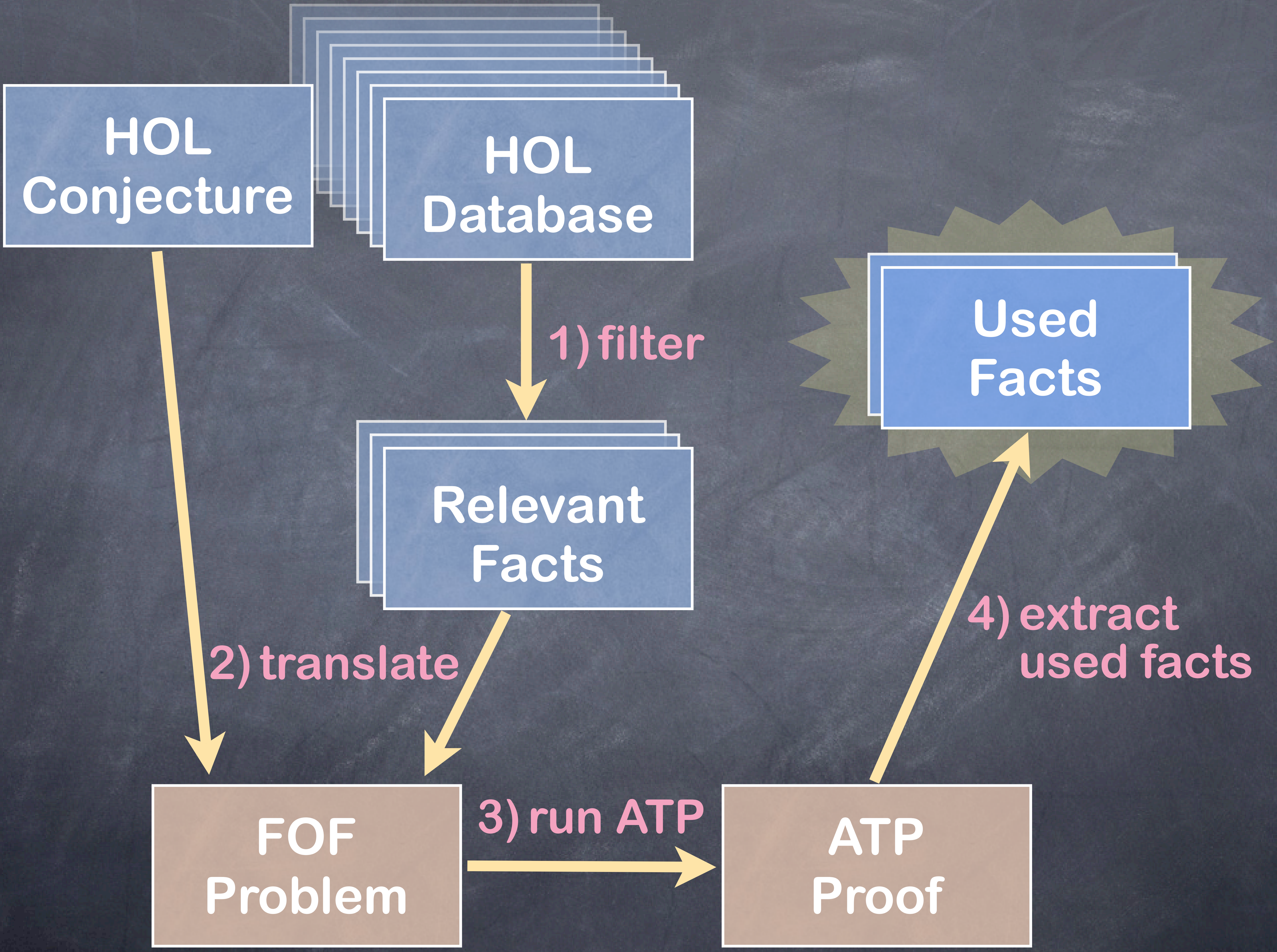
FOF Problem

3) run ATP

ATP Proof

4) replay proof

HOL Theorem



HOL Conjecture

HOL Database

1) filter

Relevant Facts

2) translate

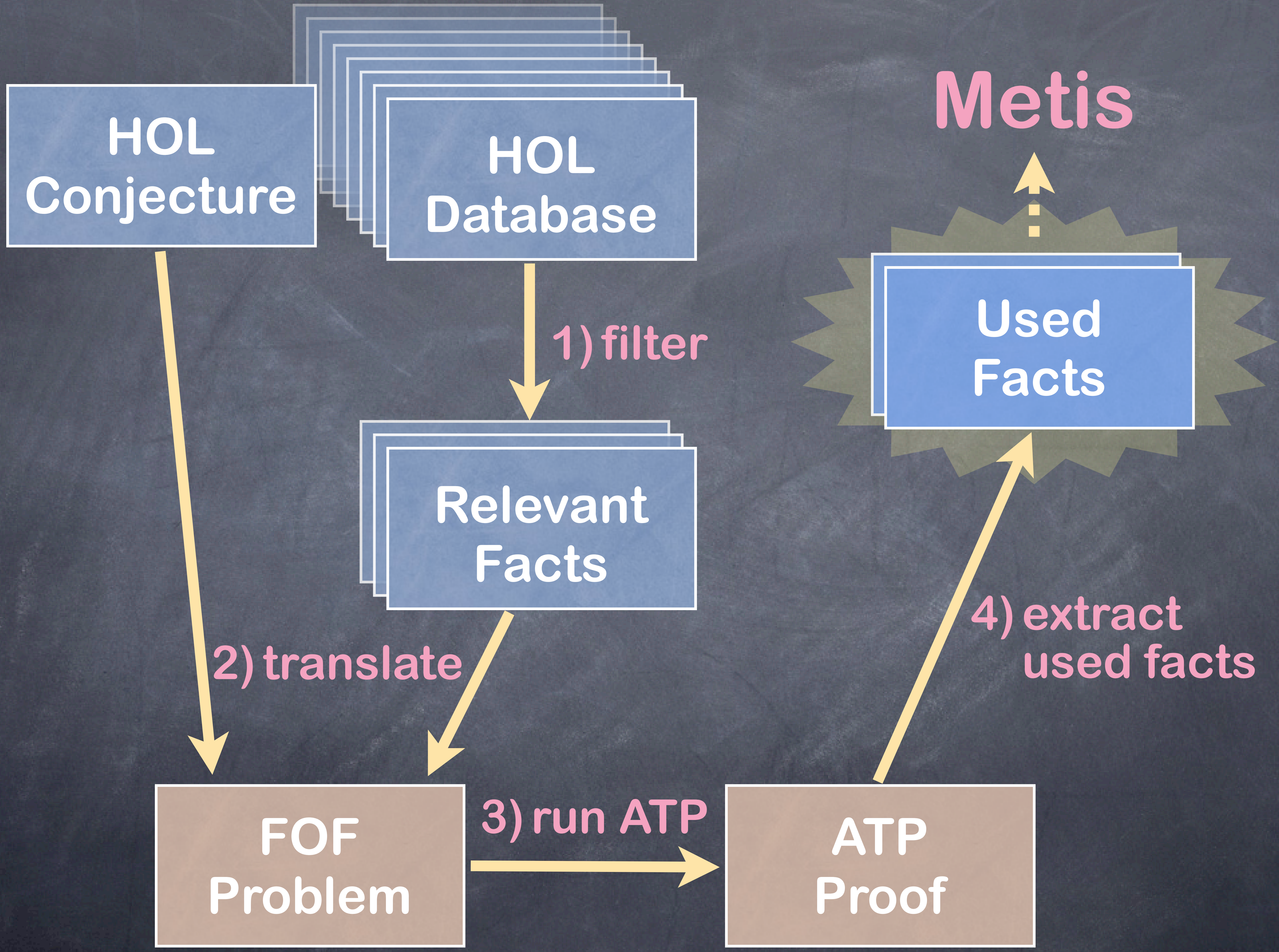
FOF Problem

3) run ATP

ATP Proof

4) extract used facts

Used Facts



HOL Conjecture

HOL Database

1) filter

Relevant Facts

2) translate

FOF Problem

3) run ATP

ATP Proof

4) extract used facts

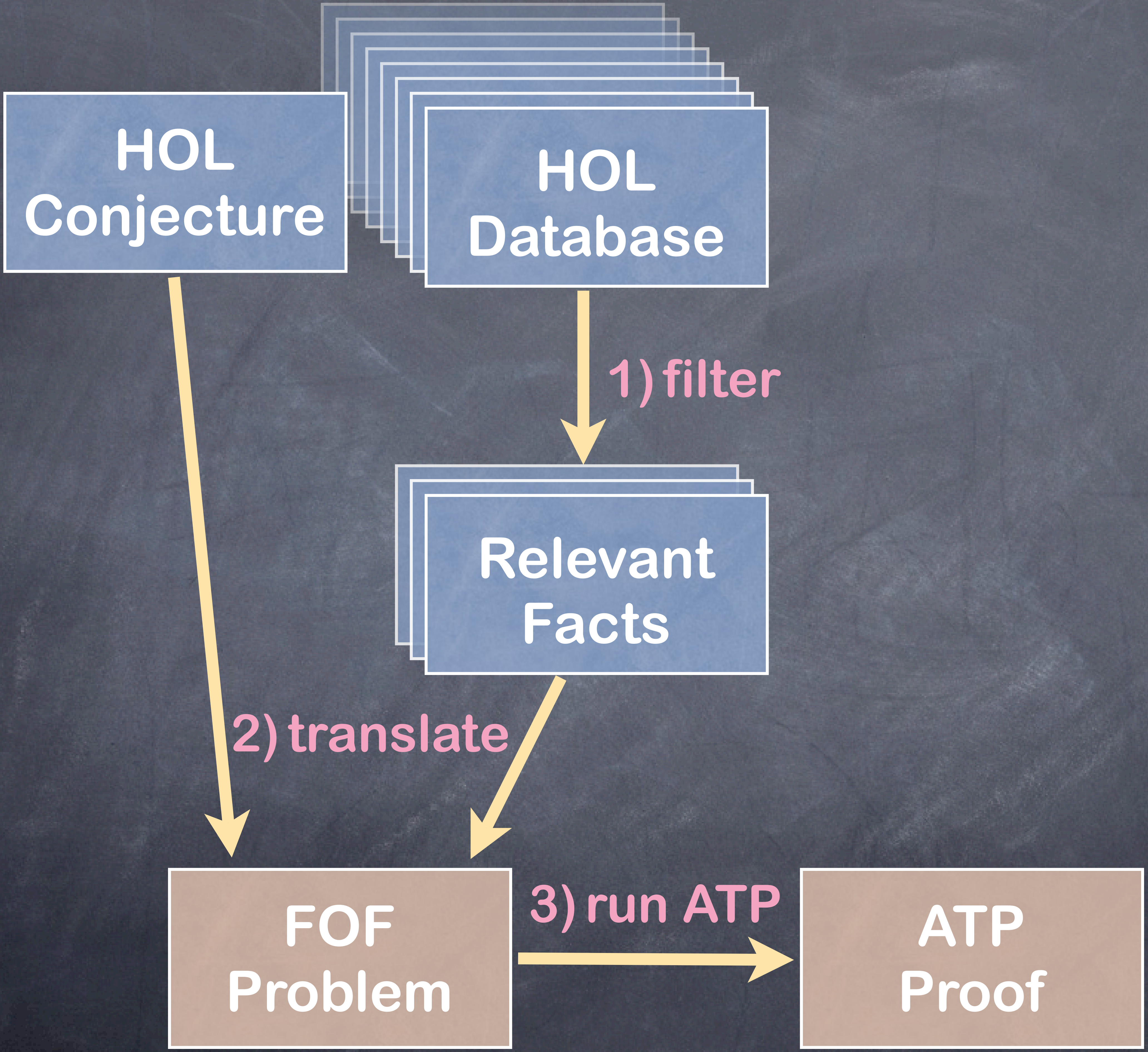
Used Facts

Metis

rev [a, b] = [b, a]

rev [a, b] = [b, a]

```
by (metis Cons_eq_appendI
          eq_Nil_appendI
          rev.simps(2)
          rev_singleton_conv)
```



HOL Conjecture

HOL Database

1) filter

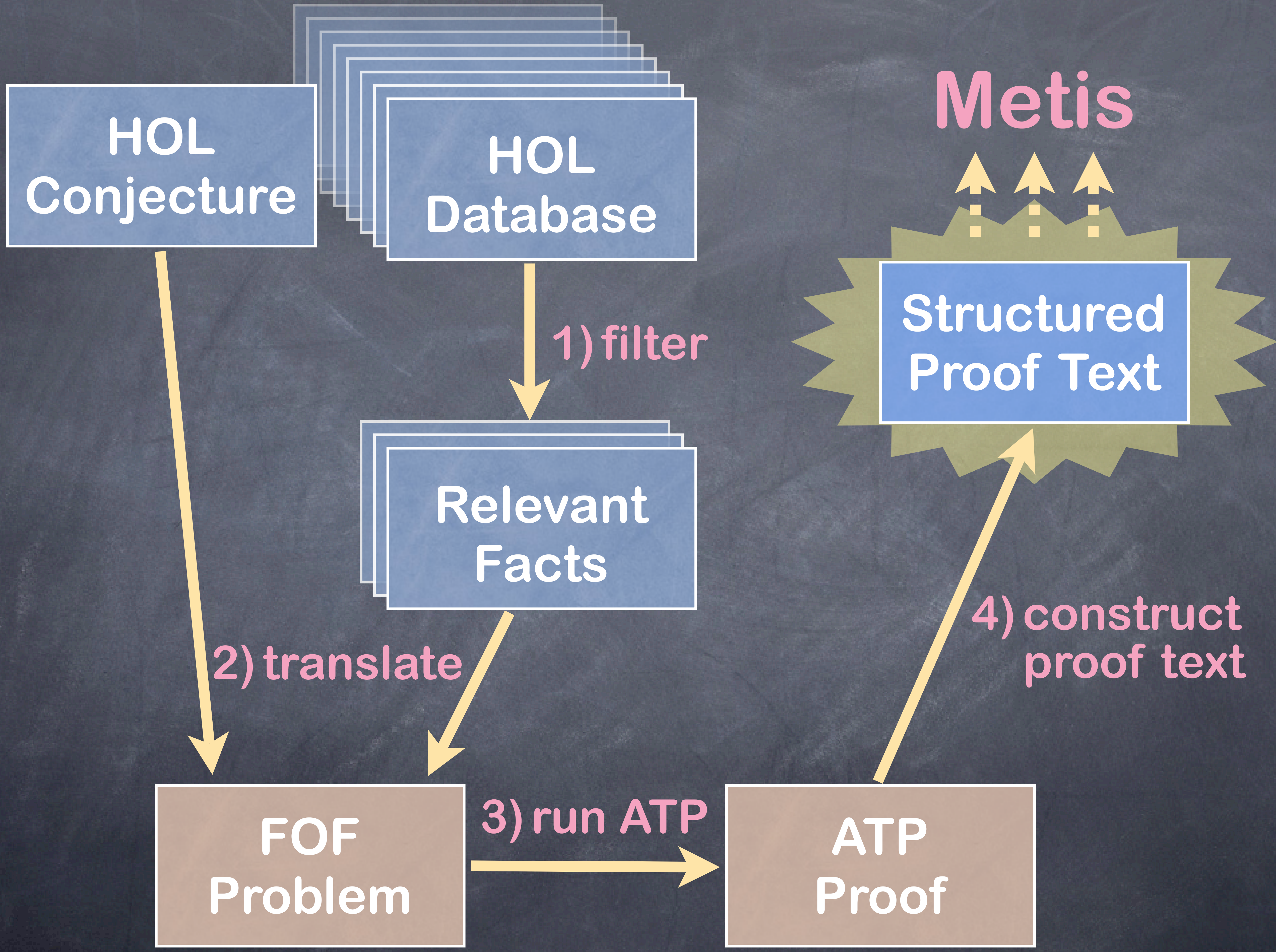
Relevant Facts

2) translate

FOF Problem

3) run ATP

ATP Proof


Metis
HOL Conjecture
HOL Database
1) filter
Relevant Facts
2) translate
Structured Proof Text
4) construct proof text
FOF Problem
3) run ATP
ATP Proof

rev [a, b] = [b, a]

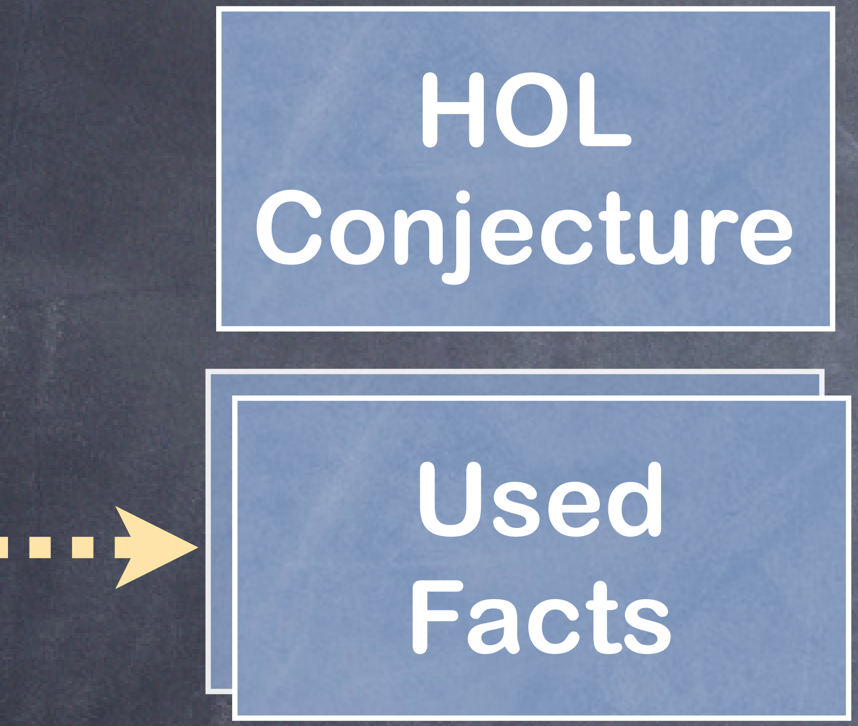
```
proof -
 have $\forall x_3\ x_2.\ [x_2]$ @ $[x_3]$ = rev $[x_3, x_2]$
   by (metis rev.simps(2) rev_singleton_conv)
 hence $\forall x_3\ x_2.\ [x_2, x_3]$ = rev $[x_3, x_2]$
   by (metis Cons_eq_appendI eq_Nil_appendI)
 thus rev [a, b] = [b, a]
   by metis
qed
```

How Metis works

How Metis works

Used Facts

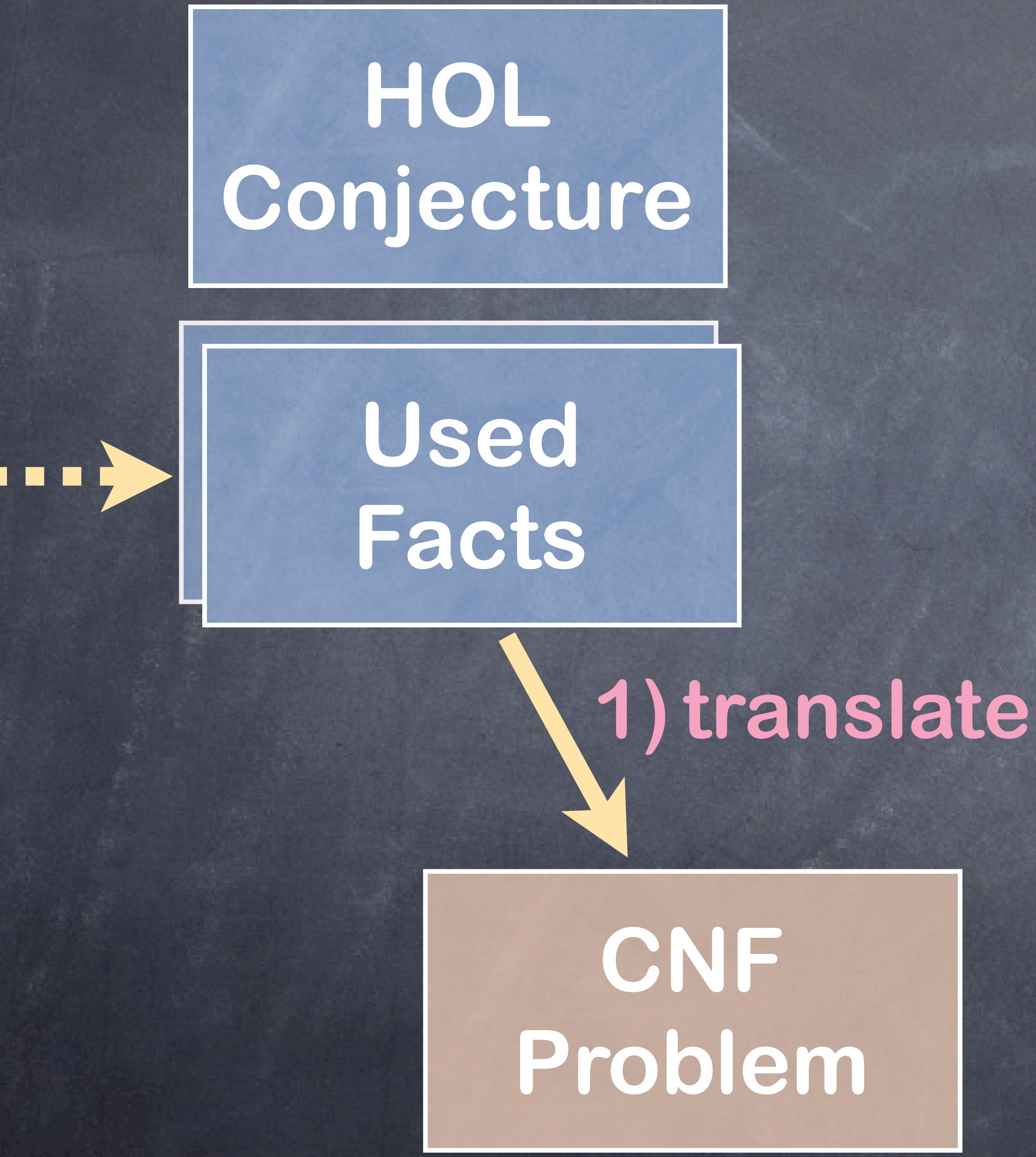
How Metis works



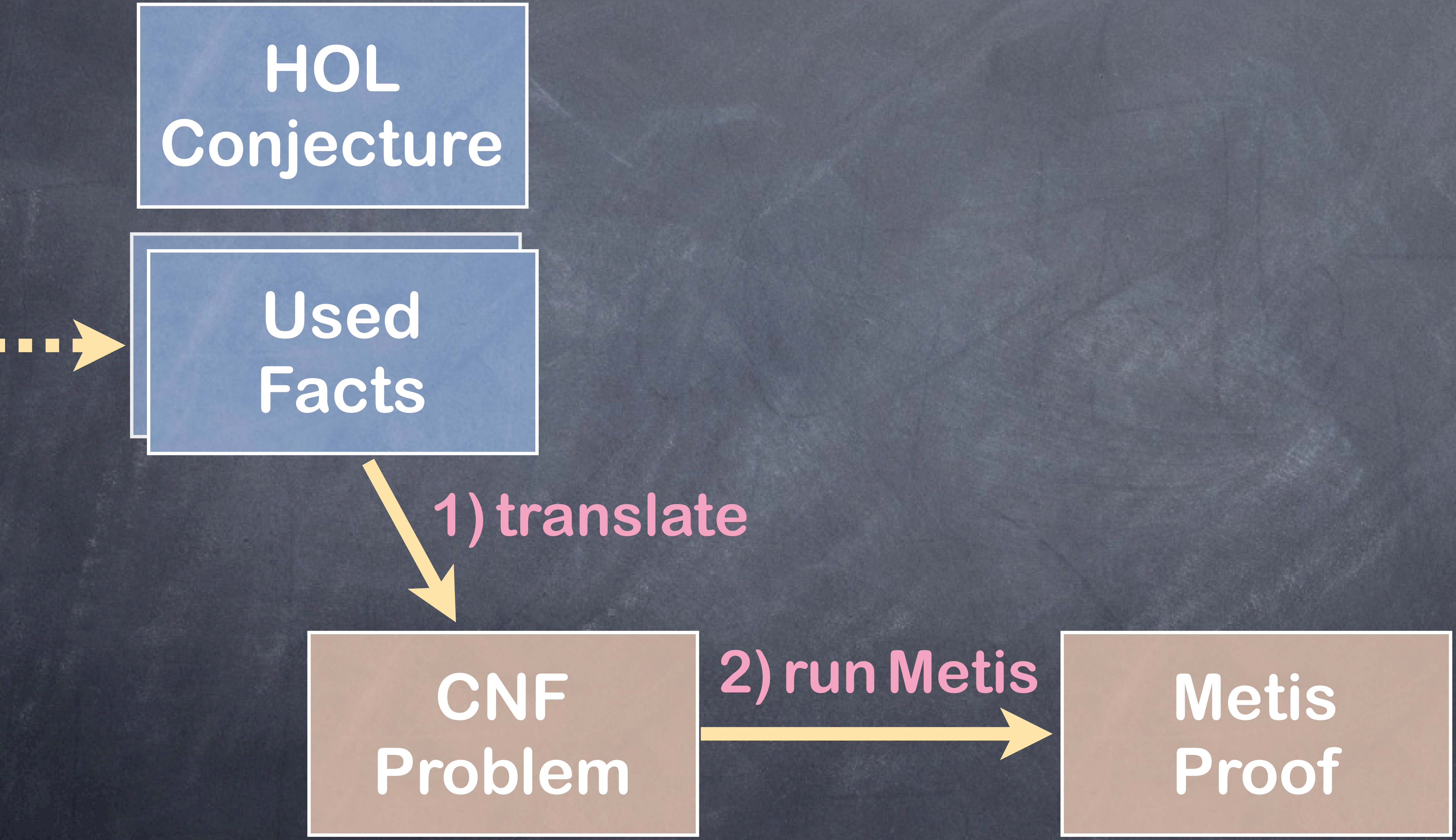
HOL Conjecture

Used Facts

How Metis works



How Metis works



HOL Conjecture

Used Facts

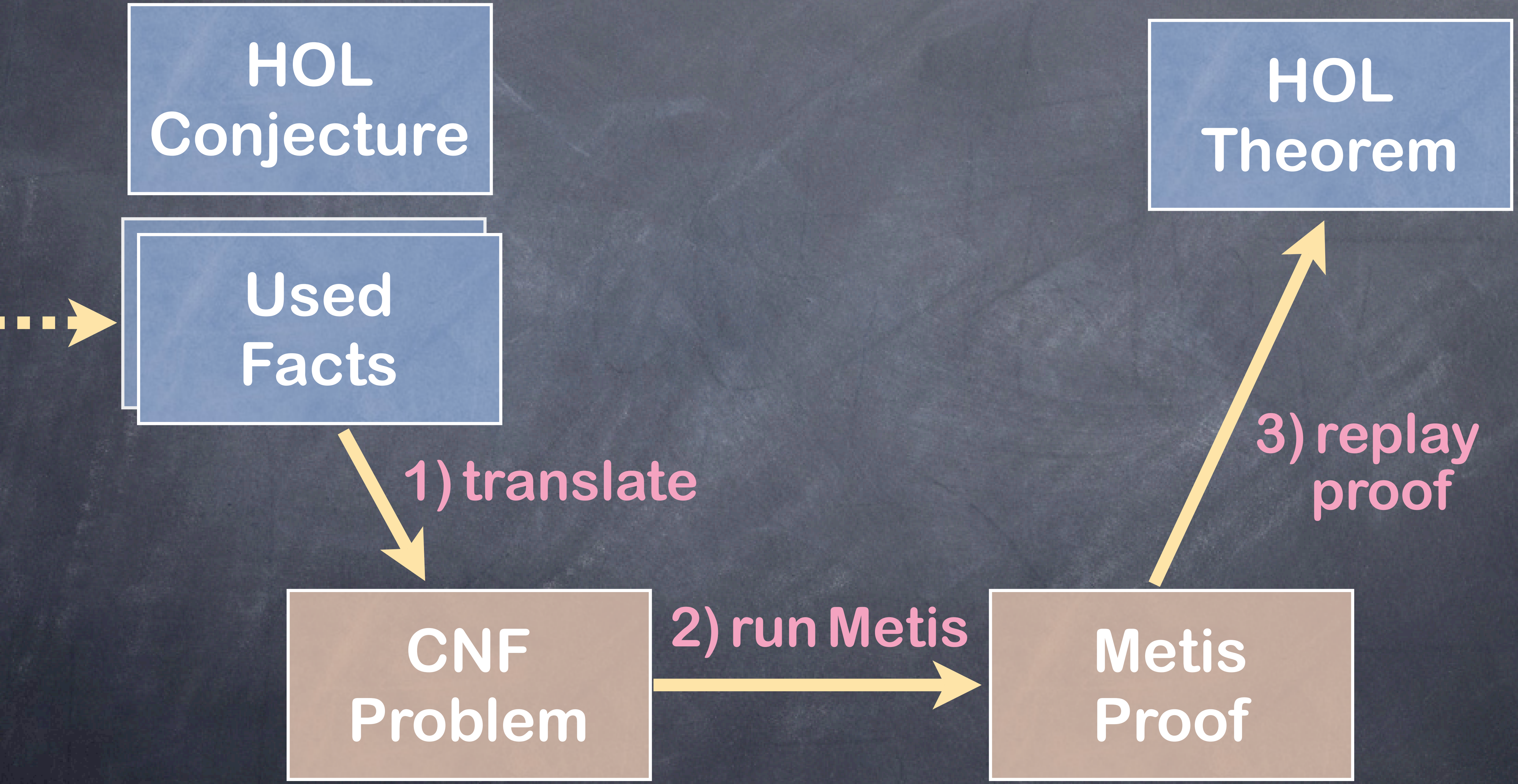
1) translate

CNF Problem

2) run Metis

Metis Proof

How Metis works



HOL Conjecture

Used Facts

1) translate

CNF Problem

2) run Metis

Metis Proof

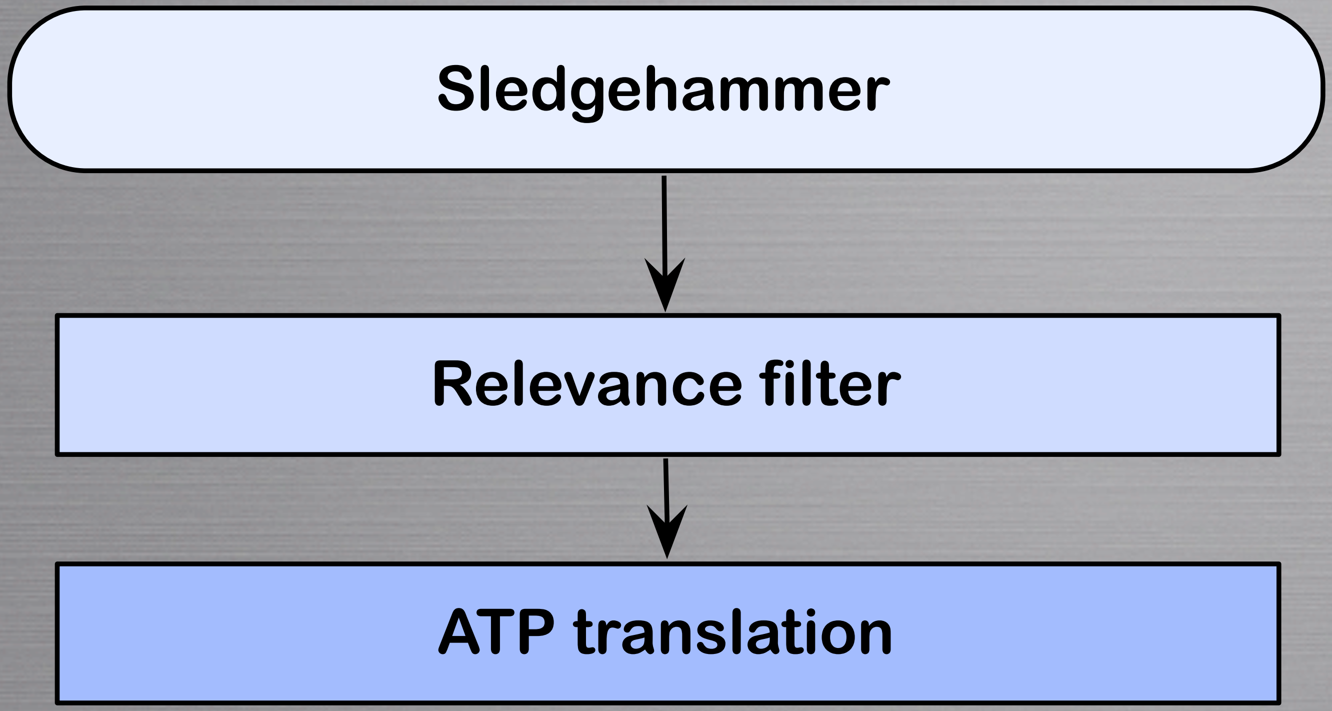
3) replay proof

HOL Theorem

```
proof -
  assume x: x ∈ lam_sys M f
  hence x ⊆ space M
    sorry
  hence space M - (space M - x) = x
    sorry
  thus space M - x ∈ lam_sys M f
    sorry
qed
```

```
proof -
  assume x: x ∈ lam_sys M f
  hence x ⊆ space M
   ~~sorry~~
  hence space M - (space M - x) = x
   ~~sorry~~
  thus space M - x ∈ lam_sys M f
   ~~sorry~~
qed
```

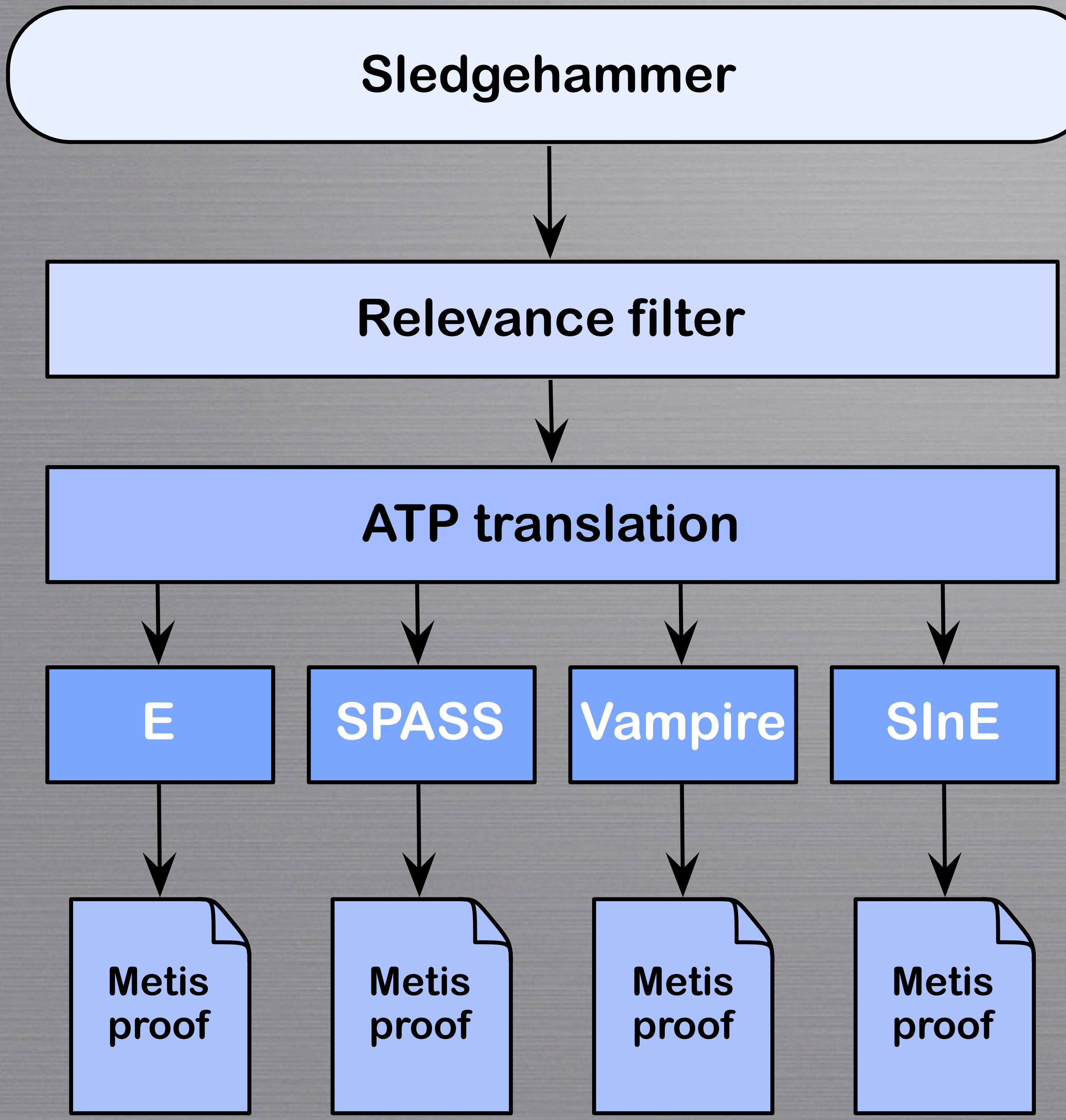
```
proof -
 assume x: x ∈ lam_sys M f
 hence x ⊆ space M
  ~~sorry~~   by (metis sets_into_space lam_sys_sets)
 hence space M - (space M - x) = x
  ~~sorry~~   by (metis double_diff equalityE)
 thus space M - x ∈ lam_sys M f
  ~~sorry~~   using x by (force simp add: lam_sys_def)
qed
```

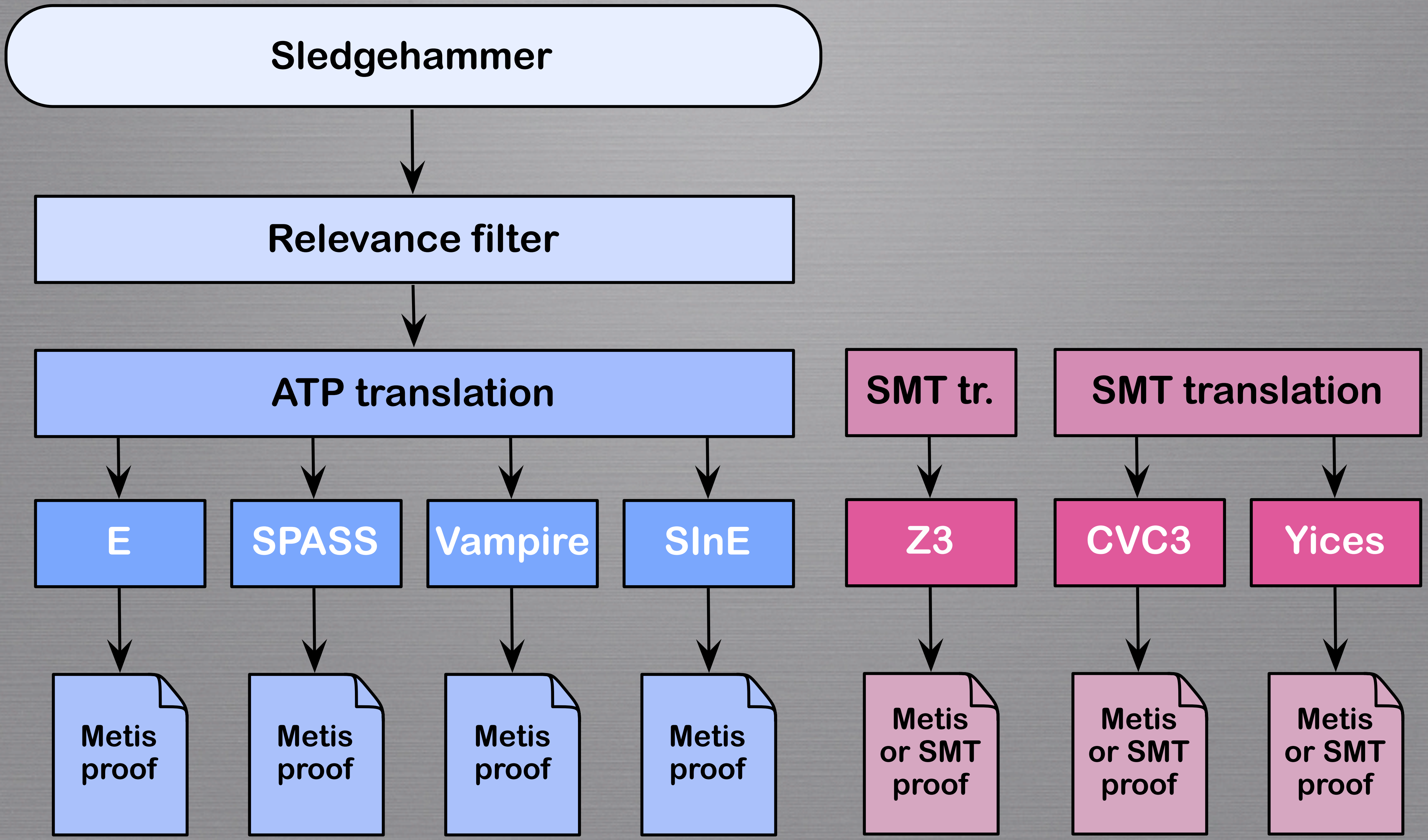


Sledgehammer

Relevance filter

ATP translation


Sledgehammer
Relevance filter
ATP translation
E
SPASS
Vampire
SInE
Metis proof
Metis proof
Metis proof
Metis proof



Sledgehammer

Relevance filter

ATP translation

SMT tr.

SMT translation

E

SPASS

Vampire

SInE

Z3

CVC3

Yices

Metis proof

Metis proof

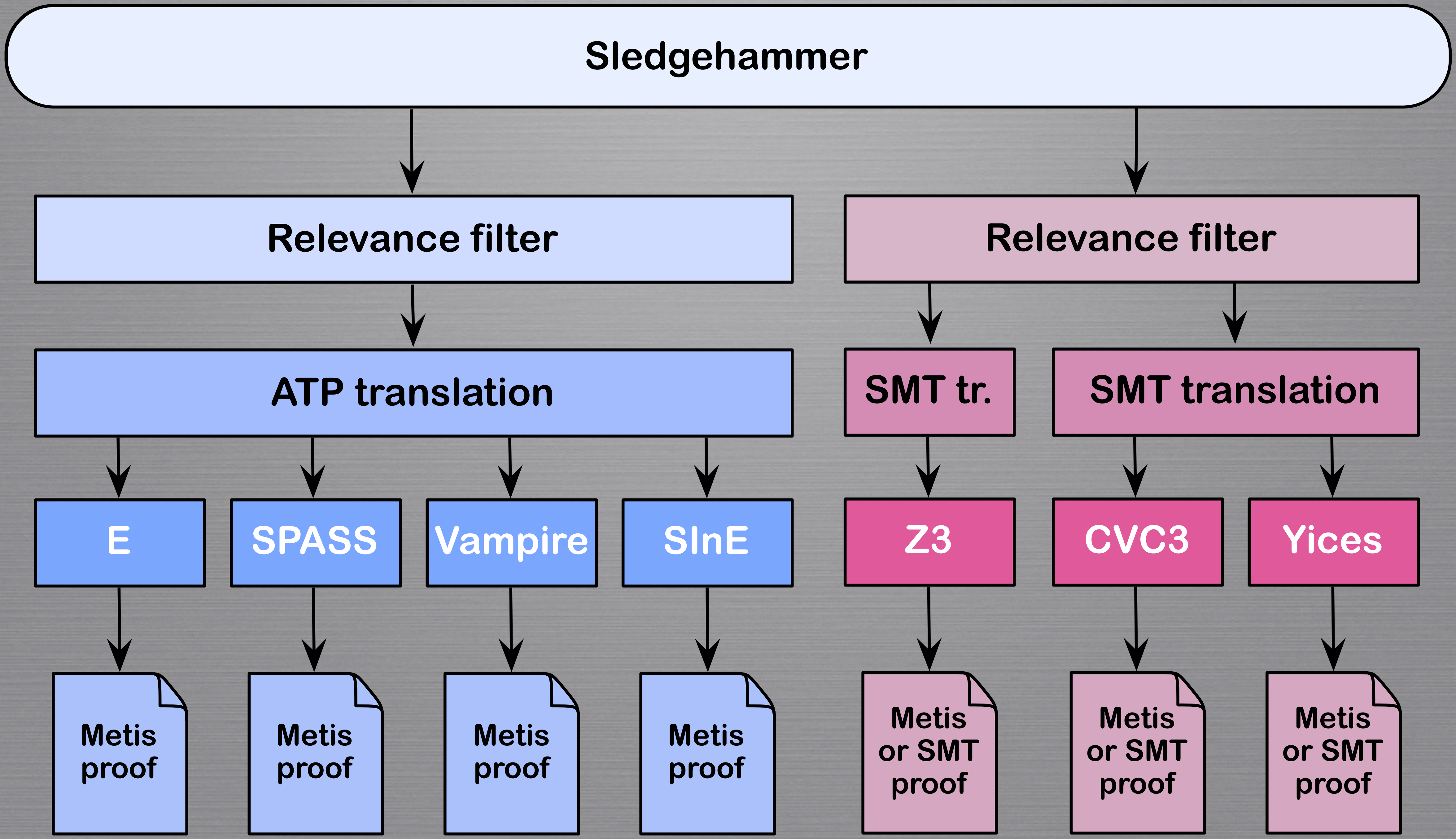
Metis proof

Metis proof

Metis or SMT proof

Metis or SMT proof

Metis or SMT proof



Sledgehammer

Relevance filter

Relevance filter

ATP translation

SMT tr.

SMT translation

E

SPASS

Vampire

SInE

Z3

CVC3

Yices

Metis proof

Metis proof

Metis proof

Metis proof

Metis or SMT proof

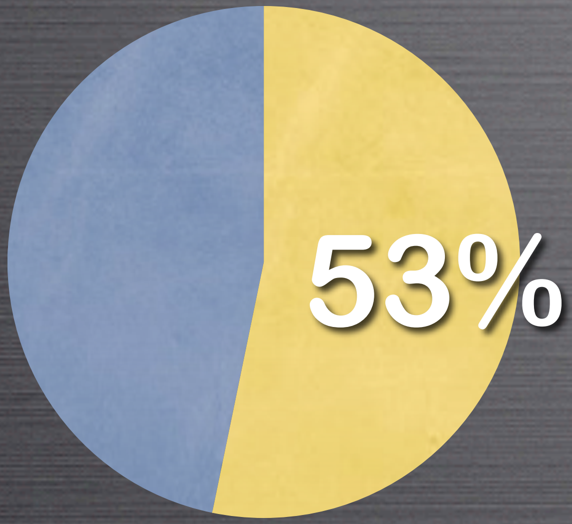
Metis or SMT proof

Metis or SMT proof

Success rate

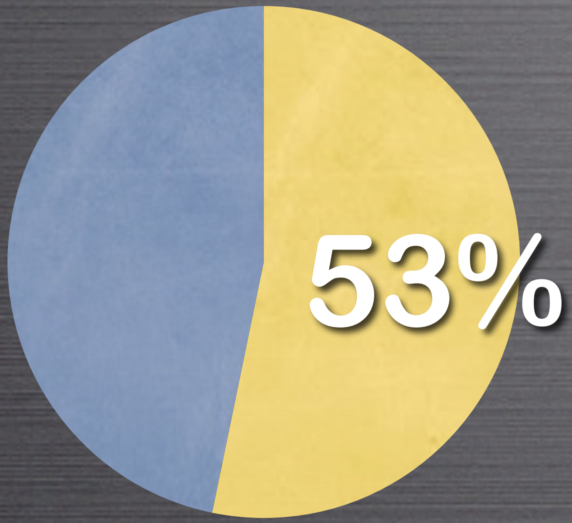
Success rate

4 ATPs x 30s

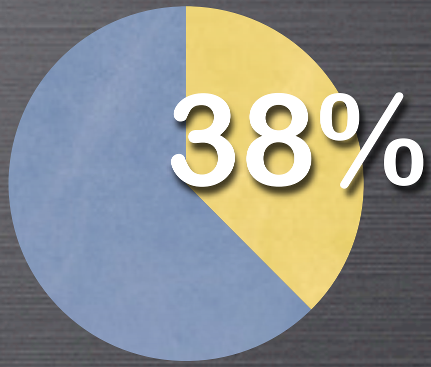


Success rate

4 ATPs x 30s

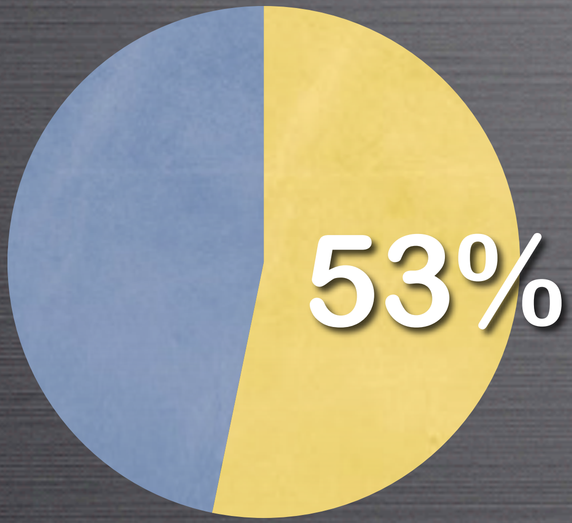


4 ATPs x 30 s
nontrivial goals

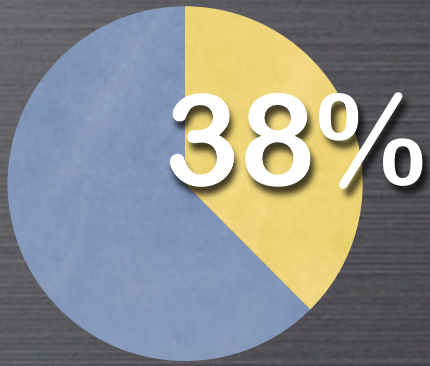


Success rate

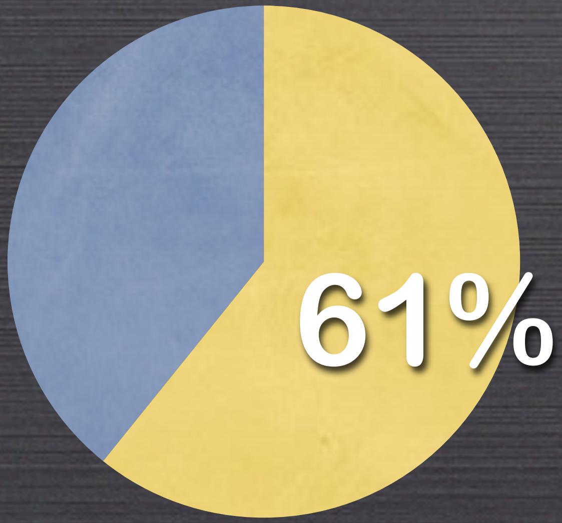
4 ATPs x 30s



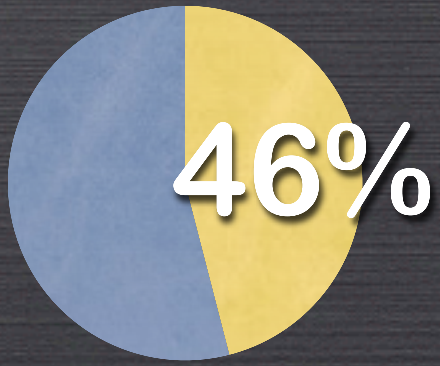
4 ATPs x 30 s
nontrivial goals



+ 3 SMTs x 30s



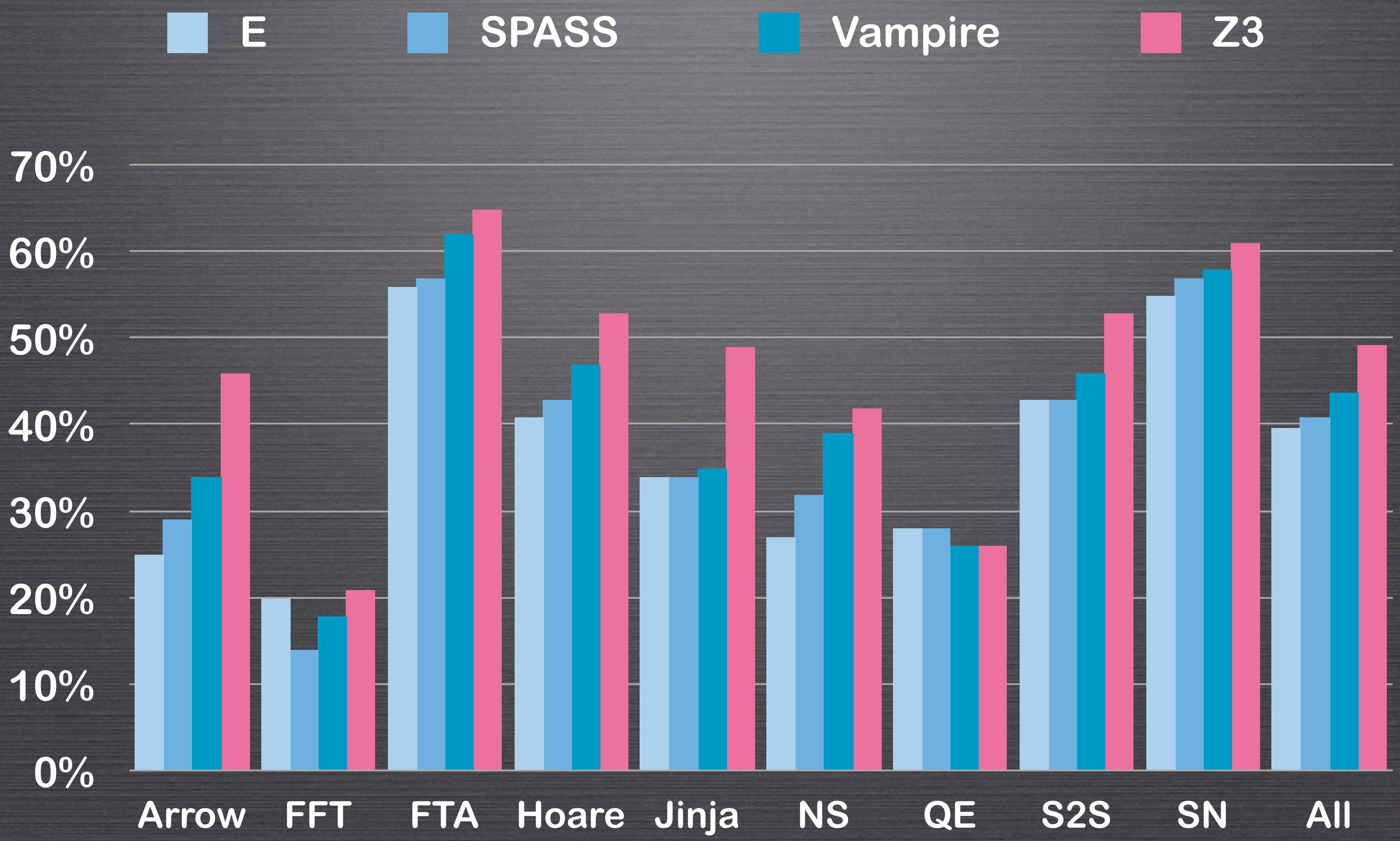
+ 3 SMTs x 30 s
nontrivial goals



Theories and Provers

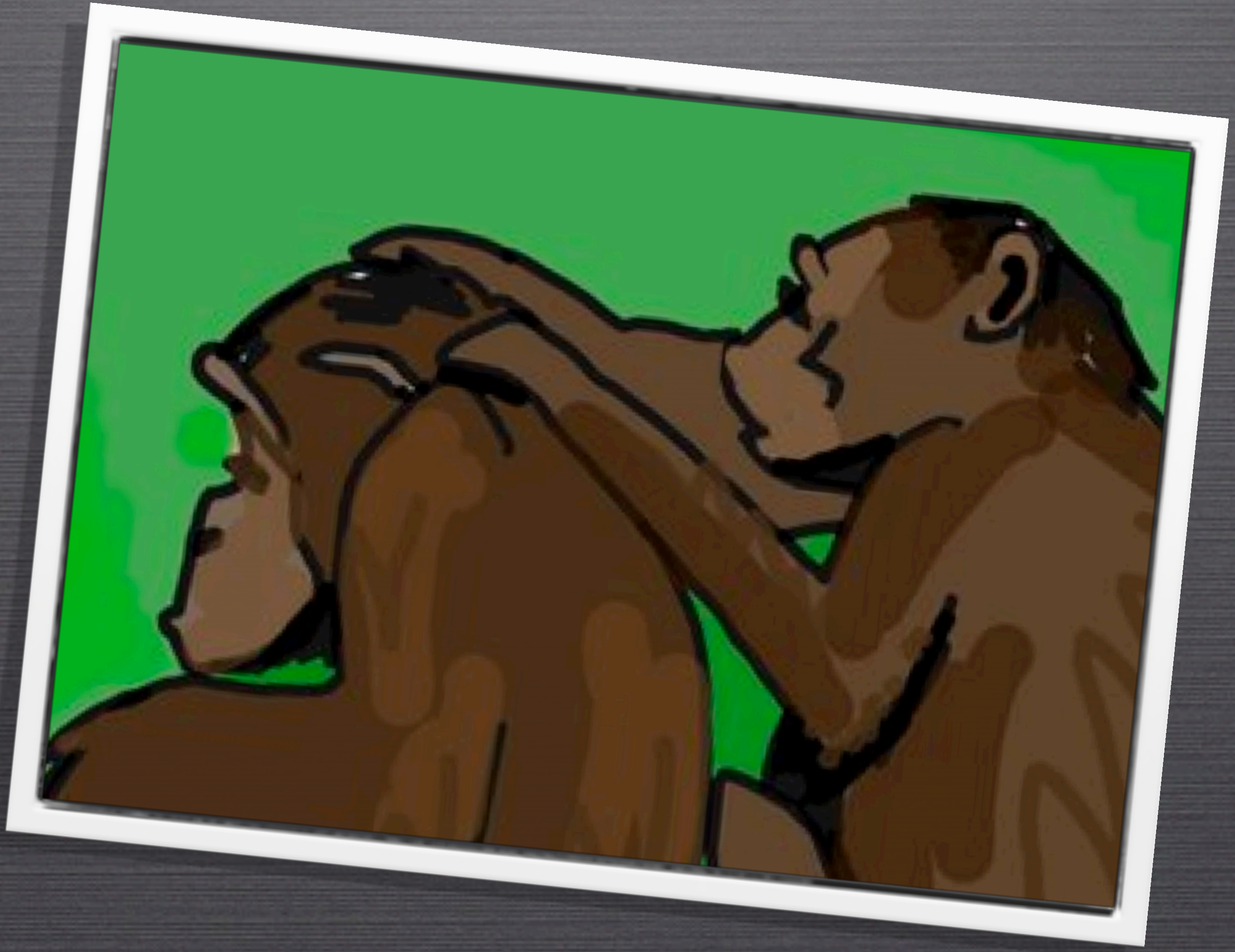
🟦 E 🟦 SPASS 🟦 Vampire 🟪 Z3

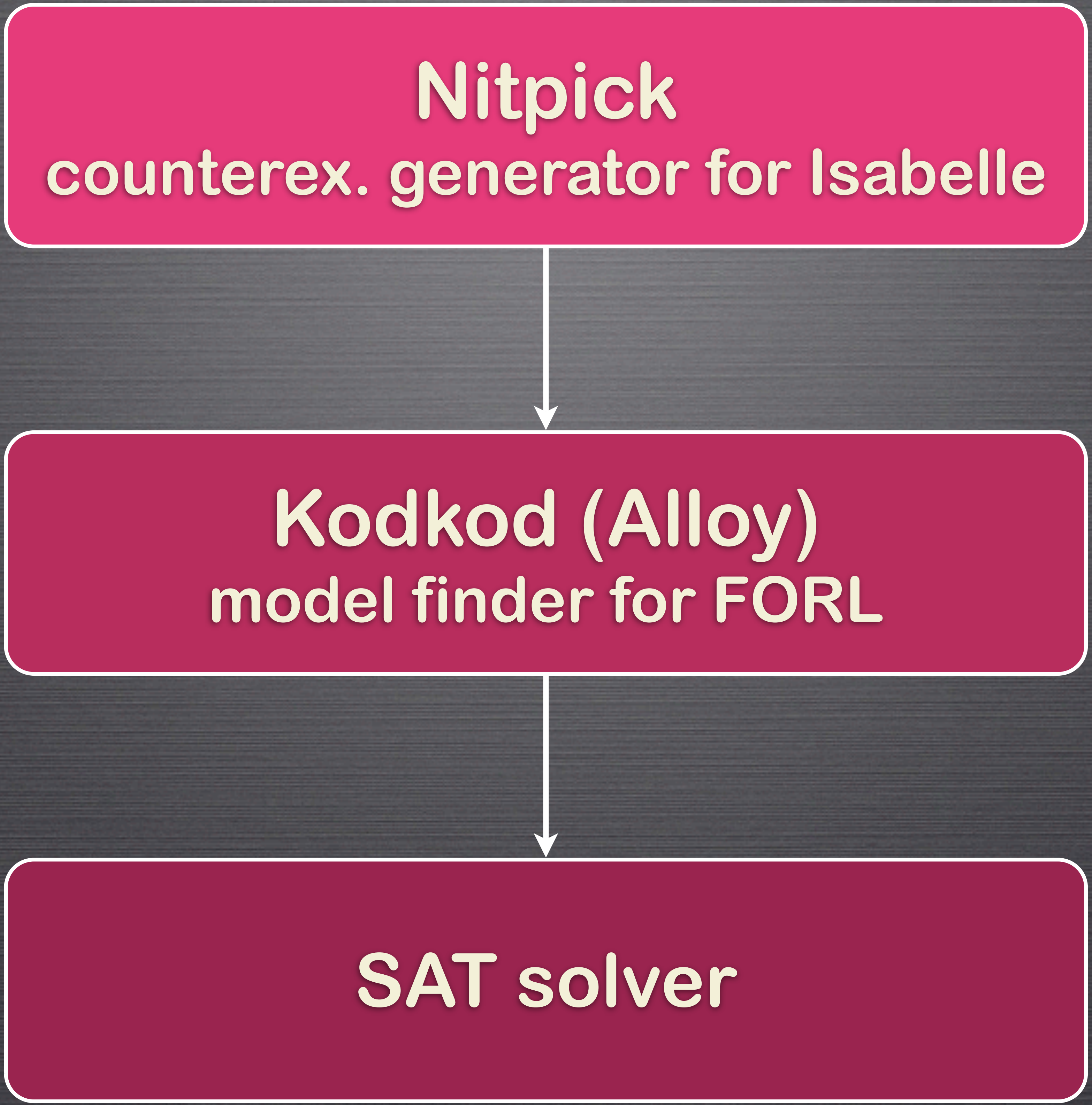
Theories and Provers



Nitpick

"Alloy for Higher-Order Logic"





Nitpick

counterex. generator for Isabelle

Kodkod (Alloy)

model finder for FORL

SAT solver

Conclusion

- Automatic tools help novices and experts
  - save time
  - allow playful exploration
  - ease learning curve

- They scale fairly well

- There is much potential for improvements

Thank You!