complement[SELECT] is a set <=> axch

Johan G. F. Belinfante
2004 November 17

```
In[1]:= SetDirectory["i:"]; << goedel63.17a; << tools.m

:Package Title: goedel63.17a        2004 November 17 at 9:15 a.m.

It is now:  2004 Nov 17 at 10:25

Loading Simplification Rules

TOOLS.M                      Revised 2004 November 17

weightlimit = 40
```

summary

It is shown in this notebook that the set version of the axiom of choice is equivalent to the statement that the class of sets without cross-sections is a set. If the axiom of choice holds, then there are no sets without cross-sections, and the empty set is of course a set. If the axiom of choice fails, then there must be lots of relations without cross-sections, because then there is a relation **x** without a cross-section, and this implies that every set for which **x** is a restriction also fails to have a cross-section. That is, the class **SELECT** of sets with cross-sections is invariant under **RESTRICT**. Perhaps it would be better to look at it this way: the class **complement[SELECT]** is invariant under **inverse[RESTRICT]**. From this it follows that when **axch** fails, **complement[SELECT]** is a proper class. That is, **axch** is equivalent to the statement that the **complement[SELECT]** is a set.

the easy direction

The easy half of the theorem is that axch implies **complement[SELECT]** is a set, namely the empty set:

```
In[2]:= SubstTest[implies, equal[0, x], member[x, V], x → complement[SELECT]]

Out[2]= or[member[complement[SELECT], V], not[axch]] == True
```

```
In[3]:= or[member[complement[SELECT], V], not[axch]] := True
```

---

the reverse implication

The main idea is to use the fact that **complement[SELECT]** is invariant under **inverse[-RESTRICT]**:

```
In[4]:= SubstTest[implies, and[subclass[u, v], member[v, V]], member[u, V],
          {u -> image[inverse[RESTRICT], complement[SELECT]], v → complement[SELECT]}]

Out[4]= or[member[complement[lb[image[inverse[COMPOSE], SELECT], P[Id]]], V],
          not[member[complement[SELECT], V]]] == True

In[5]:= % /. Equal → SetDelayed
```

The final step is to use the fact that **domain[VERTSECT[inverse[RESRICT]]]** is **complement[P[cart[V,V]]]**.

```
In[6]:= Map[implies[member[complement[SELECT], V], #] &, SubstTest[member, x,
          domain[IMAGE[y]], {x → complement[SELECT], y → inverse[RESTRICT]}]]

Out[6]= or[axch, not[member[complement[SELECT], V]]] == True

In[7]:= or[axch, not[member[complement[SELECT], V]]] := True
```

Putting together these two implications yields this logical equivalence:

```
In[8]:= equiv[member[complement[SELECT], V], axch]

Out[8]= True

In[9]:= member[complement[SELECT], V] := axch
```