

# fix[IMAGE[x]]

Johan G. F. Belinfante  
2010 July 12

```
In[1]:= SetDirectory["1:"]; << goedel.10jul11a; << tools.m

:Package Title: goedel.10jul11a          2010 July 11 at 8:30 p.m.

It is now: 2010 Jul 12 at 23:35

Loading Simplification Rules

TOOLS.M                                Revised 2010 February 26

weightlimit = 40
```

---

## summary

New rewrite rules for the relation of being a final segment are derived in this notebook. Recall that this relation is:

```
In[2]:= reify[x, fix[IMAGE[x]]]

Out[2]= fix[composite[inverse[SECOND], IMG]]
```

The following membership rule holds.

```
In[3]:= member[pair[x, y], fix[composite[inverse[SECOND], IMG]]]

Out[3]= and[equal[y, image[x, y]], member[x, V]]
```

---

## derivation

Theorem.

```
In[4]:= domain[fix[composite[inverse[SECOND], IMG]]] // Renormality

Out[4]= domain[fix[composite[inverse[SECOND], IMG]]] == V

In[5]:= domain[fix[composite[inverse[SECOND], IMG]]] := V
```

Theorem.

```
In[6]:= composite[UCLOSURE, VERTSECT[fix[composite[inverse[SECOND], IMG]]]] // FastReifNormalit

Out[6]= composite[UCLOSURE, VERTSECT[fix[composite[inverse[SECOND], IMG]]]] ==
        VERTSECT[fix[composite[inverse[SECOND], IMG]]]
```

```
In[7]:= composite[UCLOSURE, VERTSECT[fix[composite[inverse[SECOND], IMG]]]] :=
      VERTSECT[fix[composite[inverse[SECOND], IMG]]]
```

Corollary.

```
In[8]:= Map[subclass[#, fix[UCLOSURE]] &,
      ImageComp[UCLOSURE, VERTSECT[fix[composite[inverse[SECOND], IMG]]], V]]
```

```
Out[8]= subclass[range[VERTSECT[fix[composite[inverse[SECOND], IMG]]], fix[UCLOSURE]] = True
```

```
In[9]:= subclass[range[VERTSECT[fix[composite[inverse[SECOND], IMG]]], fix[UCLOSURE]] := True
```

## projections rule

In this section a variable-free formulation for the following fact is derived.

```
In[10]:= implies[and[FUNCTION[x], idempotent[x]], equal[fix[IMAGE[x]], P[fix[x]]]]
```

```
Out[10]= True
```

Lemma.

```
In[11]:= Map[composite[Id, complement[#]] &,
      complement[dif[composite[VERTSECT[fix[composite[inverse[SECOND], IMG]]], id[PROJ]],
        composite[POWER, IMAGE[inverse[DUP]]]]] // FastReifNormality
```

```
Out[11]= composite[intersection[composite[Di, POWER, IMAGE[inverse[DUP]]],
      VERTSECT[fix[composite[inverse[SECOND], IMG]]], id[PROJ]] = 0
```

```
In[12]:= % /. Equal → SetDelayed
```

Lemma.

```
In[13]:= SubstTest[empty, dif[u, v],
      {u -> composite[VERTSECT[fix[composite[inverse[SECOND], IMG]]], id[PROJ]],
      v -> composite[POWER, IMAGE[inverse[DUP]]]}]
```

```
Out[13]= subclass[composite[VERTSECT[fix[composite[inverse[SECOND], IMG]]], id[PROJ]],
      composite[POWER, IMAGE[inverse[DUP]]]] = True
```

```
In[14]:= % /. Equal → SetDelayed
```

Theorem.

```
In[15]:= SubstTest[implies, and[subclass[u, v], FUNCTION[v]],
      equal[u, composite[v, id[domain[u]]],
      {u -> composite[VERTSECT[fix[composite[inverse[SECOND], IMG]]], id[PROJ]],
      v -> composite[POWER, IMAGE[inverse[DUP]]]}] // Reverse
```

```
Out[15]= equal[composite[VERTSECT[fix[composite[inverse[SECOND], IMG]]], id[PROJ]],
      composite[POWER, IMAGE[inverse[DUP]], id[PROJ]]] = True
```

```
In[16]:= composite[VERTSECT[fix[composite[inverse[SECOND], IMG]]], id[PROJ]] :=
  composite[POWER, IMAGE[inverse[DUP]], id[PROJ]]
```

---

## complete lattice rule

In this section a variable-free formulation for the following fact is derived.

```
In[17]:= member[intersection[S, cartsq[fix[IMAGE[setpart[x]]]]], CL]
```

```
Out[17]= True
```

Theorem.

```
In[18]:= Map[subclass[range[VERTSECT[fix[composite[inverse[SECOND], IMG]]]],
  image[VERTSECT[fix[composite[inverse[SECOND], IMG]]], #]] &,
  image[inverse[VERTSECT[fix[composite[inverse[SECOND], IMG]]]],
  fix[image[inverse[CART], image[inverse[IMAGE[id[S]]], CL]]] // Normality]
```

```
Out[18]= subclass[range[VERTSECT[fix[composite[inverse[SECOND], IMG]]]],
  fix[image[inverse[CART], image[inverse[IMAGE[id[S]]], CL]]] = True
```

```
In[19]:= subclass[range[VERTSECT[fix[composite[inverse[SECOND], IMG]]]],
  fix[image[inverse[CART], image[inverse[IMAGE[id[S]]], CL]]] := True
```