category unit laws

Johan G. F. Belinfante
2009 January 6

```
In[1]:= SetDirectory["l:"]; << goedel.09jan05a;<< tools.m

    :Package Title: goedel.09jan05a        2009 January 5 at 4:00 p.m.

    It is now:  2009 Jan 6 at 15:35

    Loading Simplification Rules

    TOOLS.M                                    Revised 2008 December 26

    weightlimit = 40
```

summary

For MacLane, an **abstract objects-and-arrows category** consists of five classes, subject to various axioms. MacLane's five classes are a class of morphisms, a subclass of the class of morphisms called the class of identity morphisms (identified with objects), an associative function called the product (or composition) law, and two functions called domain and codomain. In the **GOEDEL** program only the product law is considered basic, and the other four classes are defined in terms of it. We use the **cat[x]** wrapper to denote the **product law**. The class of **morphisms** is defined to be **range[cat[x]]**, the class of identity morphisms is **ids[cat[x]]**, and the domain and codomain functions are **dom[cat[x]]** and **cod[cat[x]]**, respectively.

```
In[2]:=  "Saunders MacLane, Categories for the
         Working Mathematician, Springer-Verlag, New York, 1971.";
```

In the comments only, the following abbreviations will be used: **mor = range[cat[x]]**, **ids = ids[cat[x]]**, **prod = cat[x]**, **dom = dom[cat[x]]** and **cod = cod[cat[x]]**. MacLane's notation **f: a → b** means:

and[member[f, mor], member[pair[f, a], dom], member[pair[f, b], cod]].

Among MacLane's objects-and-arrows axioms for category theory are what he calls the **identity** and **unit laws**. In this notebook rewrite rules are derived that correspond to these laws.

MacLane's identity law

MacLane takes as an axiom the Identity Law: Every identity is a morphism with the same domain and codomain {see MacLane, page 7): $\mathbf{1_a}$**: a → a**. This assumption is here translated into the two statements **fix[dom] = ids** and **fix[cod] = ids**. These equations are derived here from the definitions of **dom** and **cod**.

Theorem. The fixed-point class of the **dom** function is equal to the class of identity morphisms.

```
In[3]:= SubstTest[fix, composite[id[u], v], {u → ids[cat[x]], v → domain[cat[x]]}] // Reverse

Out[3]= fix[dom[cat[x]]] == ids[cat[x]]

In[4]:= fix[dom[cat[x_]]] := ids[cat[x]]
```

Corollary. The fixed-point class of the **cod** function is equal to the class of identity morphisms.

```
In[5]:= SubstTest[fix, dom[cat[t]], t → flip[cat[x]]] // Reverse

Out[5]= fix[cod[cat[x]]] == ids[cat[x]]

In[6]:= fix[cod[cat[x_]]] := ids[cat[x]]
```

MacLane’s unit laws

Lemma.

```
In[7]:= SubstTest[empty, composite[Id, dif[u, v]],
          {u -> composite[FIRST, id[dom[cat[x]]]], v -> composite[cat[x], id[dom[cat[x]]]]}]

Out[7]= subclass[composite[FIRST, id[dom[cat[x]]]], cat[x]] == True

In[8]:= (% /. x → x_) /. Equal → SetDelayed
```

Theorem. Equational form of the second unit law.

```
In[9]:= SubstTest[implies, and[subclass[u, v], FUNCTION[v]],
          equal[u, composite[v, id[domain[u]]]],
          {u -> composite[FIRST, id[dom[cat[x]]]], v -> cat[x]}] // Reverse

Out[9]= equal[composite[FIRST, id[dom[cat[x]]]], composite[cat[x], id[dom[cat[x]]]]] == True

In[10]:= composite[cat[x_], id[dom[cat[x_]]]] := composite[FIRST, id[dom[cat[x]]]]
```

Corollary. Equational form of the first unit law.

```
In[11]:= Map[flip, SubstTest[composite, cat[t], id[dom[cat[t]]], t → flip[cat[x]]]] // Reverse

Out[11]= composite[cat[x], id[inverse[cod[cat[x]]]]] ==
          composite[SECOND, id[inverse[cod[cat[x]]]]]

In[12]:= composite[cat[x_], id[inverse[cod[cat[x_]]]]] :=
          composite[SECOND, id[inverse[cod[cat[x]]]]]
```

MacLane takes as an axiom the Second Unit law (see MacLane, page 8): if $\mathbf{g: b \to c}$, then $\mathbf{g \cdot 1_b = g}$.

Theorem. The composite of a morphism **u** with its dom-identity morphism is **u**.

```
In[13]:= Map[A, ImageComp[cat[x], id[dom[cat[x]]], set[PAIR[u, APPLY[dom[cat[x]], u]]]]] //
          Reverse

Out[13]= APPLY[cat[x], PAIR[u, APPLY[dom[cat[x]], u]]] ==
          union[u, complement[image[V, intersection[range[cat[x]], set[u]]]]]

In[14]:= APPLY[cat[x_], PAIR[u_, APPLY[dom[cat[x_]], u_]]] :=
          union[u, complement[image[V, intersection[range[cat[x]], set[u]]]]]
```

MacLane takes as an axiom the First Unit law (see MacLane, page 8): if **f: a → b**, then $1_b \cdot f = f$.

Corollary. A similar result for the cod-identity morphism obtained by using duality.

```
In[15]:= SubstTest[APPLY, cat[t], PAIR[u, APPLY[dom[cat[t]], u]], t → flip[cat[x]]] // Reverse

Out[15]= APPLY[cat[x], PAIR[APPLY[cod[cat[x]], u], u]] ==
          union[u, complement[image[V, intersection[range[cat[x]], set[u]]]]]

In[16]:= APPLY[cat[x_], PAIR[APPLY[cod[cat[x_]], u_], u_]] :=
          union[u, complement[image[V, intersection[range[cat[x]], set[u]]]]]
```