BRICS NS-98-7 J. Power: 2-Categories

Basic Research in Computer Science

2-Categories

John Power

BRICS Notes Series **NS-98-7**

ISSN 0909-3206 **August 1998**

See back inner page for a list of recent BRICS Notes Series publications. Copies may be obtained by contacting:

BRICS
Department of Computer Science
University of Aarhus
Ny Munkegade, building 540
DK–8000 Aarhus C
Denmark

Telephone: +45 8942 3360
Telefax: +45 8942 3255
Internet: BRICS@brics.dk

BRICS publications are in general accessible through the World Wide Web and anonymous FTP through these URLs:

`http://www.brics.dk`
`ftp://ftp.brics.dk`
This document in subdirectory `NS/98/7/`

2-Categories

John Power
LFCS, University of Edinburgh
Scotland, UK

Contents

These notes constitute lecture notes to accompany a course on 2-categories at BRICS in the Computer Science Department of the University of Aarhus in March 1998. Each section corresponds to one lecture.

1 Why 2-categories?

Consider an idealized programming language such as the simply typed λ-calculus with some constants and a call-by-name operational semantics. One can take models in a cartesian closed category. Examples of such cartesian closed categories that have been studied extensively are

1.1 Examples

1. *Set*

2. the category of ω-cpo's with least element and functions that preserve colimits of ω-chains but need not preserve the least element
3. the fully abstract model
4. models given by axiomatic domain theory, and
5. models given by synthetic domain theory.

Thus, one is interested not just in one cartesian closed category but in the class of all cartesian closed categories. One is also interested in the maps between them because a structure preserving functor $J : C \longrightarrow D$ sends one model of the language to another.

1.2 Example In O'Hearn and Tennent's work, extending the λ-calculus to an Algol-like language, they use the fact that the Yoneda embedding $Y : C \longrightarrow [C^{op}, Set]$ preserves cartesian closed structure, to extend a model for the simply typed λ-calculus to another category in which one can incorporate an account of state.

Consequently, one seeks a study not just of the class of all cartesian closed categories but in the category of small cartesian closed categories and structure preserving functors.

These considerations are not special to the simply typed λ-calculus. They apply equally to a simple imperative programming language, for which one might use a symmetric monoidal structure on the category of sets and partial functions to model contexts, finite coproducts to model conditionals, and countable products to model states.

This leads us to ask, for a general class of structure on categories, can we give an account of the category of small structured categories and structure preserving functors?

We need to ask exactly what we mean by a structure preserving functor here. Fundamental to category theory are results like

1.3 Theorem If a functor $U : C \longrightarrow D$ has a left adjoint $F : D \longrightarrow C$, then F preserves whatever colimits exist in D.

That theorem is only true if the notion of preservation of colimits means their preservation up to coherent isomorphism. For instance, left adjoints are only defined up to isomorphism, so if F is any left adjoint to U, and F' is isomorphic to F, then F' is also a left adjoint. So it is impossible in general for both F and F' to preserve colimits strictly.

For another example, let M denote the Lawvere theory for a monoid. Then, we have

1.4 Theorem The category of monoids is equivalent to the category of finite product preserving functors from M into Set and natural transformations between them.

This theorem only holds if, by preservation of finite products, we mean preservation up to coherent isomorphism. The reason is that, in any Lawvere theory, products are strictly associative with strict left and right unit; but in Set, with the Kuratowski definition, finite products do not have strict left and right unit. So if $H : M \longrightarrow Set$ strictly preserves finite products, then $H(1) \times H(X) = H(1 \times X) = H(X)$, but that equality is not true for any nonempty set $H(X)$, yet a monoid must have a unit element, thus be nonempty, a contradiction.

Thus, when we say we want to study the category of small structured categories and functors that preserve the structure, we mean that the functors preserve the structure up to coherent isomorphism.

Returning to our leading example, that of $CartClosed$, the category of small cartesian closed categories and functors that preserve cartesian closed structure, an immediate question arises:

1.5 Question Does the forgetful functor $U : CartClosed \longrightarrow Cat$ have a left adjoint?

The answer is No! If it did, $CartClosed$ would have an initial object, but it does not!

1.6 Proposition $CartClosed$ does not have an initial object.

Proof Suppose I was an initial object in $CartClosed$. The category Iso containing two objects and an isomorphism between them is cartesian closed. The category I must have at least one object, as it has a terminal object. So consider the two functors from I to Iso given by the two constant functors. Both must preserve cartesian closed structure, but they are not equal as they differ on the terminal object of I.

Despite this example, $CartClosed$ does have an object, namely 1, that behaves like an initial object. Moreover, the forgetful functor $U : CartClosed \longrightarrow Cat$ does have a construction that behaves like a left adjoint. So we seek to make the sense in which $CartClosed$ has an initial object and the sense in which $U : CartClosed \longrightarrow Cat$ has a left adjoint precise. That, and similar considerations, such as the non-existence of equalizers but the existence of some weakened sort of equalizer, leads us to the study of 2-categories, with which one can resolve these questions.

2 Calculus in a 2-category

2.1 Definition [6] A small 2-*category* is a small Cat-category. So a small 2-category C has a small set obC of objects; for all objects X and Y, a small category $C(X,Y)$; and composition functors; subject to three axioms expressing associativity of composition and left and right unit laws.

One can draw an elegant picture, treating objects, also known as 0-cells, as labelled points in space; with objects of the homs, also known as arrows, or 1-cells, as labelled lines in space from domain to codomain; and with arrows of the homs, also known as 2-cells, as labelled faces in the plane determined by domains and codomains (see [6]).

2.2 Examples Leading examples of 2-categories are

1. the 2-category Cat of small categories, functors, and natural transformations.

2. the sub-2-category Cat_g of Cat with the same 0-cells and 1-cells as Cat, but with 2-cells given by natural isomorphisms.

3. the 2-category $V-Cat$ of small V-categories for symmetric monoidal V, together with V-functors, and V-natural transformations.

4. the 2-category $Cat(E)$ of internal categories in any category E with finite limits, together with internal functors and internal natural transformations. If $E = Set$, this gives the usual category Cat. If $E = Cat$, it gives the 2-category of small double categories. If $E = Group$, it gives the 2-category of crossed modules.

5. the 2-category $MonCat$ of small monoidal categories, monoidal functors, and monoidal natural transformations. This is an instance of a class of 2-categories of the form $T - Alg_l$ for a 2-monad T on Cat (see [2]).

6. the 2-category Fib/E of fibrations over E, cartesian functors, and cartesian natural transformations.

2.3 Definition A 2-*functor* $U : C \longrightarrow D$ is a Cat-functor, i.e., it has an object function $obU : obC \longrightarrow obD$, and for each pair of objects, a functor $U : C(X, Y) \longrightarrow D(UX, UY)$, subject to two axioms to the effect that U respects composition and identities. A 2-*natural transformation* is a Cat-natural transformation, i.e., a natural transformation between the underlying ordinary functors that also respects 2-cells.

2.4 Definition An *adjunction* in a 2-category consists of 0-cells X and Y, 1-cells $u : X \longrightarrow Y$ and $f : Y \longrightarrow X$, and 2-cells $\eta : id_Y \Rightarrow uf$ and $\epsilon : fu \Rightarrow id_X$ subject to the usual triangle equations, i.e., $(u\epsilon)(\eta u) = id_u$ and $(\epsilon f)(f\eta) = id_f$.

One can draw delightful pictures representing this: see [6].

2.5 Examples

1. An adjunction in Cat is an adjunction in the usual sense.

2. An adjunction in Cat_g is an (adjoint) equivalence.

3. An adjunction in $V - Cat$ is a V-adjunction in the usual sense. In fact, that is how V-adjunctions are defined in the canonical reference [4].

4. An adjunction in $Cat(E)$ is an internal adjunction.

5. An adjunction in $MonCat$ is a monoidal adjunction (see [2]).

6. An adjunction in Fib/E is a fibred adjunction.

In general, if anyone claims to have a new notion of adjunction, it is well worthwhile to try to find a 2-category such that the new notion of adjunction is an adjunction in that 2-category. That attempt was a considerable help in refining the notion of local adjunction, and it offered insight into a notion of free higher dimensional category. Often, the attempt helps to get the axioms right.

2.6 Definition A *monad* in a 2-category consists of a 0-cell X, a 1-cell $t : X \longrightarrow X$, and 2-cells $\mu : t^2 \Rightarrow t$ and $\eta : id_X \Rightarrow t$, subject to the evident three axioms.

Again, it is possible to draw elegant pictures to depict the axioms [6].

2.7 Examples

1. A monad in Cat is a monad as usual.
2. A monad in $V - Cat$ is a V-monad, and again, that is how the notion of V-monad may be defined.
3. A monad in $Cat(E)$ is an internal monad in E.
4. A monad in $SymMonCat$ is a symmetric monoidal monad, which is equivalent to a commutative monad, i.e., a monad with a commutative strength.
5. A monad in Fib/E is a fibred monad.

It is routine to verify

2.8 Proposition Every adjunction in a 2-category gives rise to a monad.

Proof Just copy the usual construction: given f left adjoint to u, define $t = uf$, and η to be the unit of the adjunction, and define μ using the counit as usual. It is routine to verify the axioms. ■

If a 2-category has some finite limits (to be discussed in a later section), there is a construction in the other direction, i.e., a construction that given a monad, yields an adjunction. In the case of Cat, it is the usual Eilenberg Moore construction. In the presence of finite colimits, then one may deduce, by considering the 2-category K^{op} for a 2-category K, that there is a dual, yielding the Kleisli construction. This work is ultimately by Ross Street [12].

It gives universal properties of the Eilenberg Moore and Kleisli constructions that are stronger than one might imagine. This is best described with pictures: see [6].

We have mentioned many pictures in the above analysis. We have not drawn them owing to my mediocre latex skills. However, they are well worth using, at least on paper and on blackboards; and the 2-category literature is full of them, especially in older papers for which secretaries did the typing, and when journals accepted hand-drawn figures. They are returning now as typesetting is becoming more accessible to mathematicians. See [9] for an account of the pictures. The condition therein has been improved a little by Alex Simpson, so here we give his version, which alas has not been published.

The central question is which figures in the plane (or on paper or on a blackboard) may be drawn to represent precisely one composite of 2-cells in a 2-category. So we make that precise now.

By a *graph*, we mean a (non-empty) connected finite directed graph. A *path* in a graph is an alternating sequence $v_0 e_1 \cdots v_n$ of vertices v_i and edges e_i in the graph such that the endpoints of each e_i are v_i and v_{i+1}, and such that all the v_i's are distinct. A path is *directed* if each e_i goes from v_i to v_{i+1}. A *plane graph* is a graph together with an embedding into the (oriented) plane: for practical purposes, this means a graph written on a blackboard or a sheet of paper, with no crossings of edges. Note that there may be many topologically different embeddings.

A plane graph divides the rest of the plane into one exterior region and a finite number of interior regions. These are called faces. Given an interior face, consider the boundary as an alternating sequence of vertices and edges, moving clockwise around the face.

2.9 Definition A *plane graph with source and sink* is a plane graph with vertices s and t in the exterior face such that

- s only has edges out of it,
- t only has edges into it, and
- every other vertex has edges both in and out of it.

2.10 Definition A *pasting scheme* is a plane graph with source and sink such that for every interior face F, there exist distinct vertices $s(F)$ and $t(F)$ and directed paths $\sigma(F)$ and $\tau(F)$ from $s(F)$ to $t(F)$ such that the boundary of F is given by $\sigma(F)(\tau(F))^*$.

We call $\sigma(F)$ the *domain* of F and $\tau(F)$ the *codomain* of F.

It follows from the definition, and this is the heart of the proof we need, that

2.11 Proposition A pasting scheme has no directed loops.

Proof (Sketch) Suppose a pasting scheme had a directed loop. Take a loop containing the smallest number of faces. Take an interior face of the loop with an edge on the boundary of the loop. With some effort, one can construct another loop that is inside the given loop but does not contain that face, a contradiction. ■

2.12 Definition A *labelling* of a pasting scheme in a 2-category is a labelling of each vertex by a 0-cell, each edge by a 1-cell, and each face by a 2-cell, respecting domains and codomains.

Now we have

2.13 Theorem Every labelling of a pasting scheme has a unique composite.

Proof Induction on the number of faces of the pasting scheme. The unicity is easy; it is the existence that requires a little work. Essentially, you need to prove that a pasting scheme has a topmost face, i.e., that there exists an interior face whose domain lies entirely on the exterior face of the graph. So one proceeds by induction, starting at s, and using heavily the fact that a pasting scheme has no directed loops. Either s is $s(F)$ for some face, or it is not. If it is not, pass to the unique vertex to which there is an edge from s and continue inductively. If it is, then consider the topmost such F and see whether its domain lies entirely on the exterior. If so, we are done. If not, then using no directed loops, it follows that there is another point on the exterior face and on the domain of F that is itself of the form $s(F')$ for some F' with domain commencing along the exterior. Proceed inductively. ■.

See [9] for more detail of pasting.

3 The calculus of 2-categories

In the previous section, we studied calculus in a 2-category. We now study the calculus of 2-categories. By that, I mean that we study the relationship between several 2-categories, rather than restricting our attention to an individual one.

The basic notion here is that of a 2-categorical version of adjunction. Typically, the underlying ordinary functor of a 2-functor does not have a left adjoint, although in principle it should. For instance, the forgetful functor $U : CartClosed \longrightarrow Cat$ from the category of small cartesian closed categories and structure preserving functors into Cat does not have a left adjoint, although there is clearly some sort of free construction of a cartesian closed category on any small category. This leads to the notion of a biadjunction. It is a horrible word, due to Ross Street [13].

It may have come time now to replace the "bi" notation, possibly by a consistent use of "pseudo". The problem with that has been that "pseudo" has been used for a different meaning, so there would have been a clash. However, the clash may be disappearing now as the other use, in connection with limits, is proving to be a false direction.

3.1 Definition A left *biadjoint* to a 2-functor $U : C \longrightarrow D$ is given by, for each object X of D, a 1-cell $\eta_X : X \longrightarrow UFX$ in D such that composition with η_X induces an equivalence of categories from $C(FX, Y)$ to $D(X, UY)$.

For a reasonably easy example of this

3.2 Example Consider, given a 2-category C, what it means for the unique 2-functor $t : C \longrightarrow 1$ to have a left biadjoint. That amounts to the statement that C has a *bi-initial* object. In elementary terms, it means that there is an object 0 such that for every object X, there is a 1-cell from 0 to X, and such that, for every pair of 1-cells $f, g : 0 \longrightarrow X$, there is a unique 2-cell from f to g.

So, to check that a 2-functor has a left biadjoint, one needs to check two conditions: that application of U followed by composition with η_X is essentially surjective, and that it is fully faithful. Many people forget the latter point, and that can lead to considerable error.

The existence of a left biadjoint does not imply that F can be extended to a 2-functor and η to a 2-natural transformation: if we had demanded an isomorphism of categories in the definition, it would have done so; but we only demanded an equivalence of categories, and that weaker condition is all that is true of the leading examples. So we need to extend the notions of 2-functor and 2-natural transformation. The following definitions, stated in the mildly more general setting of bicategories, ultimately came from Benabou's [1].

3.3 Definition A *pseudo-functor* or *homomorphism* of 2-categories consists of

- an object function $obF : obD \longrightarrow obC$,
- functors $F : D(A, B) \longrightarrow C(FA, FB)$, and
- natural isomorphisms to replace the equalities in the definition of 2-functor,

subject to three coherence axioms, representing associativity and left and right unit laws.

3.4 Definition A *pseudo-natural transformation* or a *strong transformation* from F to G consists of,

- for each object X of D, a 1-cell $\alpha_X : FX \longrightarrow GX$, and
- for each 1-cell $f : X \longrightarrow Y$, an isomorphism in what would be the commutative square for a 2-natural transformation,

subject to three coherence conditions making the latter isomorphisms respect composition and identities in D and respect 2-cells in D.

3.5 Definition A *modification* between pseudo-natural transformations with the same domain and codomain consists of an obD-indexed family of 2-cells $\gamma_X : \alpha_X \Rightarrow \beta_X$ subject to coherence with respect to 1-cells in D.

3.6 Proposition Given a left biadjoint (FX, η_X) to U, the construction F extends to a pseudo-functor and η to a pseudo-natural transformation.

Proof This is a routine generalisation of the usual situation. ∎

As usual, we could define the notion of biadjunction in terms of a pair of pseudo-functors. One can routinely extend the notion of left biadjoint from being that of a 2-functor to that of a pseudo-functor, upon which F is left biadjoint of U if and only if U is right biadjoint of F, with the notion of right biadjoint defined by duality. Moreover, a left biadjoint is unique up to coherent pseudo-natural equivalence. Putting some of this together, we have

3.7 Proposition Given a pseudo-functor $U : C \longrightarrow D$, the following are equivalent:

- to give a left biadjoint to U,
- to give, for each X and Y, an equivalence of categories between $C(FX, Y)$ and $D(X, UY)$ subject to coherence laws, and
- to give a pseudo-functor $F : D \longrightarrow C$ and pseudo-natural transformations η and ϵ, and isomorphic modifications where the usual triangle identities hold, subject to coherence axioms.

Again, a proof is routine; see [13] for more detail. Observe also, that it follows from the above definitions, that

3.8 Proposition and Definition Given 2-categories C and D, the structure given by

- 0-cells are pseudo-functors from C to D
- 1-cells are pseudo-natural transformations
- 2-cells are modifications

with composition determined pointwise by that in D, forms a 2-category.

Again, this follows by routine calculation.

We attempt to follow the usual development of category theory here. In ordinary category theory, if a functor has a left adjoint, it preserves limits. We know that is not true here because, in the dual situation, recall that we have

3.9 Example *CartClosed* does not have an initial object. So the left biadjoint F to the forgetful 2-functor $U : CartClosed \longrightarrow Cat_g$ cannot preserve the initial object 0 of Cat_g.

Obviously however, such pseudo-functors as F here do preserve some colimiting constructions in some weakened sense, as we know that $F0$ is in some sense initial in *CartClosed*. In order to make that precise, we need to generalise the notion of limit a little, just as we had to generalise the notion of adjunction. This leads to the notion of bilimit, perhaps now better called a pseudo-limit despite a clash with old terminology as for instance in [2] or [3].

The most natural general notion here is that of weighted bilimit [13], but I specifically want to avoid the notion of weight, as it adds complexity that I think, although elegant and valuable, would distract from the main exposition. So I shall use (and outline the definitions of the notions in)

3.10 Theorem [13, 14] A 2-category has all weighted bilimits if and only if it has all biproducts, biequalizers, and bicotensors. A pseudo-functor preserves all bilimits if and only if it preserves each of the above classes of bilimits.

3.11 Theorem If $U : C \longrightarrow D$ has a left biadjoint, then U preserves all bilimits.

For the definitions used here (see [13] for more detail)

3.12 Definition A 2-category C has *biproducts* if for every small set X, the diagonal 2-functor $\Delta : C \longrightarrow Bicat(X, C)$ has a right biadjoint.

3.13 Definition A 2-category C has *biequalizers* if the diagonal 2-functor $\Delta : C \longrightarrow Bicat(Pair, C)$ has a right biadjoint, where $Pair$ is the category with two objects and a pair of 1-cells from one to the other.

3.14 Definition A 2-category C has *bicotensors* if for every object X and every small category c, there is an object X^c such that for every object Y, there is an equivalence of categories between $Cat(c, C(Y, X))$ and $C(Y, X^c)$, pseudo-naturally in X and Y.

Bicotensor generalises the usual notion of cotensor in a V-category [4]. One does not see the notion of cotensor explicitly in ordinary categories because it is subsumed by the notion of product: a cotensor of an object A by a set s in an ordinary category is just the s-fold product of copies of A. Often, bicotensors are easy to describe. For instance, in *Cat*, a cotensor, which is necessarily a bicotensor, amounts exactly to a functor category. The same is true in *CartClosed* and in many other categories of small categories with structure. The dual, bitensors, are often more difficult to describe explicitly, for much the same reason as coproducts tend to be difficult to describe.

3.15 Definition A pseudo-functor U *preserves* a bilimit if, modulo the coherence isomorphisms in the definition of U, it sends a bilimiting diagram to a bilimiting diagram.

Note that bilimits are only unique up to coherent equivalence, not up to isomorphism as is the case for ordinary limits in ordinary categories. Moreover, there is not a unique comparison map. So one must be much more careful about coherence here.

In fact, there are somewhat simpler but equivalent versions of the above definitions. For instance, for the definitions of biproduct and biequalizer, it is routine to verify that every pseudo-functor from the index category (X and *Pair* respectively) is equivalent to a 2-functor. So one need only verify the biadjointness condition with respect to 2-functors, not all pseudo-functors. This is an easy example of a coherence result. We shall see more of coherence in the next section. Remember that in checking for the existence of bilimits, one must verify both the essential surjectivity condition of an equivalence and the fully faithfulness condition.

These definitions rapidly become spectacularly complicated. Try to spell out the definition of biequalizer, or bipullback. Thus we seek stronger, albeit less natural structures, that a wide class of 2-categories possess, and we seek coherence theorems that state when we may, without loss of generality, replace a complex structure such as those we have described here, by a less complex structure: that is what we did in saying that we could restrict our attention to 2-functors in defining biproducts and biequalizers. We shall study coherence in depth in the next section.

4 Coherence

In this section, we see how the complexity that arises from the definitions of pseudo-functor, pseudo-natural transformation, bilimit, and preservation of bilimits, can be eased to a substantial extent. Much of this simplification relies upon coherence results, which we shall outline here (see [7, 8, 10, 11]). Also, we can see by careful analysis of large classes of 2-categories that we often have a simpler situation than those of full generality of the definitions.

There have been many attempts at defining good notions of limit in 2-categories. These have gone under names such as pseudo-limits, lax limits, oplax limits, flexible limits, and pie limits. In my view, the best of these is the class of pie limits. They include all of the others except flexible limits, which I think are mildly unnatural. For a detailed account of these notions, see [5]. For an idea of pie-limits,

4.1 Definition A *pie limit* is any 2-limit generated by 2-products, inserters, and equifiers.

4.2 Definition A 2-product is a Cat-product.

For example, a binary 2-product in a 2-category C is just a binary product diagram in the underlying ordinary category, for which a two-dimensional property also holds.

4.3 Definition Given parallel 1-cells $f, g : X \longrightarrow Y$ in a 2-category C, an *inserter* of f and g is a universal 2-cell $j : Ins \longrightarrow X$ together with a 2-cell $\alpha : fj \Rightarrow gj$, i.e., for every $(h : Z \longrightarrow X, \beta : fh \Rightarrow gh)$, there exists a unique 2-cell $k : Z \longrightarrow Ins$ such that $jk = h$, and $\alpha k = \beta$, and a two-dimensional property holds.

4.4 Definition Given 2-cells $\alpha, \beta : f \Rightarrow g : X \longrightarrow Y$ in a 2-category, an *equifier* is a universal 1-cell $j : E \longrightarrow X$ such that $\alpha j = \beta j$, universal with respect to both one- and two-dimensional properties.

Note that these are not just bilimits: the universal property asserts the existence of unique 1-cells, with commutation strictly, not just up to isomorphism. This makes life considerably easier. Moreover, plenty of 2-categories have such limits: see [2] for a large class of them. In general, for any 2-monad

T on *Cat*, the 2-category of strict T-algebras and functors that preserve the structure in the usual sense has pie limits given as in *Cat*. For instance, *CartClosed* has pie limits, as have *Cart*, *FL*, and the 2-category of accessible categories. The defining paper is [11]. A central but easy theorem here is

4.5 Theorem All pie limits are bilimits, and if a 2-category has all pie limits, then it has all bilimits, and the latter are given by pie limits.

The existence of pie limits is a remarkably strong condition. For example, any 2-category with pie limits has cotensors, as a cotensor can be given by a equifier of an inserter of a product. There is some analysis of pie limits in the various papers such as [5], showing that they include lax, oplax, and pseudo-limits, and in particular, Eilenberg Moore objects, which are lax limits. This makes precise a remark earlier herein when we said that a 2-category has Eilenberg Moore objects if it has some limits. What pie limits specifically do not include are equalizers.

4.6 Example The 2-category *CartClosed* has all pie limits, but does not have equalizers. Consider the two constant functors from 1 to *Iso*, where the latter is the category consisting of two objects and an isomorphism between them. Both categories are cartesian closed, and both functors preserve the cartesian closed structure. Any equalizer must be empty, as the two functors do not agree on any object, but an equalizer must be cartesian closed, hence contain a terminal object, a contradiction.

Now we consider coherence theorems. The central coherence theorem in this regard is

4.7 Theorem [8] Every bicategory with finite bilimits is biequivalent to a 2-category with finite pie limits.

We have not addressed bicategories at all here, so we shall restrict our attention to 2-categories. In fact, the theorem was stated a little more generally, in that one can prove biequivalence with a 2-category with finite flexible limits, but the latter are a strange class of 2-limits that I think are better left to history.

The proof of this result, once one knows it, is not difficult, but relies upon a few key ideas. First, we generalise the Yoneda lemma to

4.8 Theorem (bicategorical Yoneda) For every pseudo-functor $H : C \longrightarrow Cat$ and every object X of C, there is an equivalence, pseudo-natural in H and X, between $Bicat(C^{op}, Cat)(C(-, X), H)$ and HX.

4.9 Corollary The Yoneda pseudo-functor $Y : C \longrightarrow Bicat(C^{op}, Cat)$ is locally an equivalence, i.e., on each homcategory, it is an equivalence.

Now recall that since Cat is a 2-category, so is $Bicat(C^{op}, Cat)$. Moreover, it is routine to verify that it has pie limits, given pointwise. Just as for the ordinary Yoneda functor, it is also routine to verify

4.10 Theorem The Yoneda pseudo-functor $Y : C \longrightarrow Bicat(C^{op}, Cat)$ preserves whatever bilimits exist in C.

Putting this together gives a proof of our main theorem, by taking the full sub-2-category of $Bicat(C^{op}, Cat)$ given by closing the representables under finite pie limits. This 2-category is biequivalent to C. You need to be just a little careful about size, but it is routine to account for it. See [8] for the small extra amount of detail required.

Coherence for finite bilimit preserving pseudo-functors is considerably more difficult. To say that a pseudo-functor preserves bilimits means that it need only preserve the bilimit up to equivalence, not isomorphism. So, given 2-categories with pie limits, to say that a 2-functor between them preserves bilimits is a weaker statement than saying it preserves pie limits, because preservation of pie limits is preservation in the usual sense, up to isomorphism. It seems not to be the case that every finite bilimit preserving pseudo-functor from a small 2-category C with finite pie limits into Cat need be equivalent to a finite pie limit preserving 2-functor from C into Cat: there is a counterexample for the corresponding statement with a terminal object.

But there is a theorem here. It is quite difficult, and a proof still has not been published, although an outline appears in [10]. But the statement is

4.11 Theorem For every bicategory B with finite bilimits, there is a 2-category C that has finite pie limits and is biequivalent to B, for which the 2-category $FB(B, Cat)$ of finite bilimit preserving pseudo-functors from B to Cat and pseudo-natural transformations is biequivalent to the 2-category $FPie(C, Cat)$ of finite pie limit preserving 2-functors from C to Cat and pseudo-natural transformations.

To prove this result, we would first need to explain two-dimensional monad theory, as in [2], then show how that extends to a weak version of three-dimensional monad theory, then prove a coherence result in three dimensions extending the main result of [2], which is in two dimensions. That seems too much for now!

References

[1] J. Benabou, Introduction to bicategories, Reports of the Midwest Category Seminar, Lecture Notes in Math 47, Springer (1967) 1–77.

[2] R. Blackwell, G.M. Kelly, and A.J. Power, Two-dimensional monad theory, J. Pure Appl. Algebra 59 (1989) 1–41.

[3] G.J. Bird, G.M. Kelly, A.J. Power, and R. Street, Flexible limits for 2-categories, J. Pure Appl. Algebra 61 (1989) 1–27.

[4] G.M. Kelly, Basic concepts of enriched category theory, London Math. Soc. Lecture Notes Series 64, Cambridge University Press (1982).

[5] G.M. Kelly, Elementary observations on 2-categorical limits, Bull. Austral. Math. Soc. 39 (1989) 301–317.

[6] G.M. Kelly and R. Street, Review of the elements of 2-categories, Proc. Sydney Category Theory Seminar 1972/73, Lecture Notes in Math. 420, Springer (1974) 75–103.

[7] A.J. Power, A general coherence result, J. Pure Appl. Algebra 57 (1989) 165–173.

[8] A.J. Power, Coherence for bicategories with finite bilimits I, Contemporary Mathematics 92 (1989) 341–347.

[9] A.J. Power, A 2-categorical pasting theorem, J. Algebra 129 (1990) 439–445.

[10] A.J. Power, Why tricategories? Information and Computation 120 (1995) 251–262.

[11] A.J. Power and E.P. Robinson, A characterization of pie limits, Math. Proc. Camb. Phil. Soc. (1991) 33–47.

[12] R. Street, The formal theory of monads, J. Pure Appl. Algebra 2 (1972) 149–168.

[13] R. Street, Fibrations in bicategories, Cahiers de Topologie et Geometrie Differentielle 21 (1980) 111–160.

[14] R. Street, Correction to "Fibrations in bicategories", Cahiers de Topologie et Geometrie Differentielle Categoriques 28 (1987) 53-56.

Recent BRICS Notes Series Publications

NS-98-7 **John Power. *2-Categories*. August 1998. 18 pp.**

NS-98-6 **Carsten Butz, Ulrich Kohlenbach, Søren Riis, and Glynn Winskel, editors. *Abstracts of the Workshop on Proof Theory and Complexity, PTAC '98,* (Aarhus, Denmark, August 3–7, 1998), July 1998. vi+16 pp.**

NS-98-5 **Hans Hüttel and Uwe Nestmann, editors. *Proceedings of the Workshop on Semantics of Objects as Processes, SOAP '98,* (Aalborg, Denmark, July 18, 1998), June 1998. 50 pp.**

NS-98-4 **Tiziana Margaria and Bernhard Steffen, editors. *Proceedings of the International Workshop on Software Tools for Technology Transfer, STTT '98,* (Aalborg, Denmark, July 12–13, 1998), June 1998. 86 pp.**

NS-98-3 **Nils Klarlund and Anders Møller.** MONA ***Version 1.2 — User Manual*. June 1998. 60 pp.**

NS-98-2 **Peter D. Mosses and Uffe H. Engberg, editors. *Proceedings of the Workshop on Applicability of Formal Methods, AFM '98,* (Aarhus, Denmark, June 2, 1998), June 1998. 94 pp.**

NS-98-1 **Olivier Danvy and Peter Dybjer, editors. *Preliminary Proceedings of the 1998 APPSEM Workshop on Normalization by Evaluation, NBE '98,* (Gothenburg, Sweden, May 8–9, 1998), May 1998.**

NS-97-1 **Mogens Nielsen and Wolfgang Thomas, editors. *Preliminary Proceedings of the Annual Conference of the European Association for Computer Science Logic, CSL '97* (Aarhus, Denmark, August 23–29, 1997), August 1997. vi+432 pp.**

NS-96-15 **CoFI. *CASL – The CoFI Algebraic Specification Language; Tentative Design: Language Summary*. December 1996. 34 pp.**

NS-96-14 **Peter D. Mosses. *A Tutorial on Action Semantics*. December 1996. 46 pp. Tutorial notes for FME '94 (Formal Methods Europe, Barcelona, 1994) and FME '96 (Formal Methods Europe, Oxford, 1996).**

NS-96-13 **Olivier Danvy, editor. *Proceedings of the Second ACM SIGPLAN Workshop on Continuations, CW '97* (ENS, Paris, France, 14 January, 1997), December 1996. 166 pp.**