SKEW MONOIDALES in SPAN

Jim Andrianopoulos

March 29, 2016

Abstract

This article consists of an interesting characterisation of a skew monoidal in the monoidal bicategory \( \text{Span} \). After discussing the shift or decalage functor on simplicial sets we characterise these skew monoidales as categories \( C \) together with a functor \( R: \text{Dec}(C) \to C \) which satisfies two conditions and give an example where the unit of the skew monoidal is not of a restricted type.

1 Introduction

A general classification of skew monoidales in a monoidal bicategory in terms of simplicial maps from the Catalan simplicial set into the nerve of the monoidal bicategory is shown in [3].

We will consider a skew monoidal in the particular monoidal bicategory \( \text{Span} \) to examine a question arising from [8].

Since their introduction by Bénabou in [2], \( \text{Span} \) and the \( \text{Span} \) construction are ubiquitous in higher category theory. This is mainly due to the fact that a category can be regarded as a monad in the bicategory of spans \( \text{Span} \), and various generalisations. However, what interests us is \( \text{Span} \) not just as a bicategory but as a monoidal bicategory made monoidal using the cartesian product of sets. Skew monoidales (= skew pseudomonoids) were defined by Lack and Street in [8], where they also show that quantum categories are skew monoidal objects, with a certain unit, in an appropriate monoidal bicategory. This contains as a special case the fact that categories are equivalently skew monoidales \( C \) in the monoidal bicategory \( \text{Span} \) with tensor product given by

\[
C \times C \xrightarrow{(s,t)} E \xrightarrow{1} C
\]

for some set \( E \), and where the unit is assumed to be of the form

\[
I \xleftarrow{1} C \xrightarrow{1} C;
\]
where \( I \) is a terminal object in \( \text{Set} \). We characterise skew monoidales in \( \text{Span} \) without any restrictions on the unit of the skew monoidale. This means that the tensor product for the skew monoidale \( C \) is given by
\[
C \times C \xrightarrow{(s,r)} E \xrightarrow{t} C
\]
for some set \( E \), and where the unit has the form
\[
I \xleftarrow{U} \xrightarrow{j} C.
\]

This characterisation follows some lengthy but not difficult calculations, which are made easier using the concrete form a pullback takes in \( \text{Set} \). We recover the fact in [8], that categories are equivalently skew monoidales in \( \text{Span} \) with a unit of a certain restricted type. Section 5.2 collects the extra structure obtained from a skew monoidale in the form of a functor \( R \) with some interesting properties. We finish the article with a simple example of a skew monoidale (actually just a monoidale) in \( \text{Span} \) whose unit is not of the restricted type previously considered.

\section{Skew Monoidales}

Instead of defining a skew monoidale in a general monoidal bicategory we define them in a Gray monoid. So in what follows we write as if \( B \) is a 2-category.

Let \( B \) be a Gray monoid; see [4] for an explicit definition. Note that for 1-cells \( f : A \rightarrow A' \) and \( g : B \rightarrow B' \) in a Gray monoid, the only structural 2-cells are the invertible 2-cells of the form
\[
\begin{array}{ccc}
A \otimes B & \xrightarrow{1 \otimes g} & A \otimes B' \\
\downarrow{f \otimes 1} & \cong & \downarrow{f \otimes 1} \\
A' \otimes B & \xrightarrow{1 \otimes g} & A' \otimes B'
\end{array}
\]
or tensors and composites thereof. In this section we denote them with the symbol \( \cong \) as above. These 2-cells satisfy some axioms which we do not list. We write \( I \) for the unit object of the Gray monoid.

A \textit{skew monoidal} structure on an object \( A \) in \( B \) consists of morphisms \( p : A \otimes A \rightarrow A \) and \( j : I \rightarrow A \) in \( B \), respectively called the \textit{tensor product} and \textit{unit}, equipped with the following 2-cells, denoted by \( \alpha \), \( \lambda \) and \( \rho \), respectively called the \textit{associativity}, \textit{left unit} and \textit{right unit constraints}
\[
\begin{array}{ccc}
A \otimes A \otimes A & \xrightarrow{1 \otimes p} & A \otimes A \\
p \otimes 1 & \xrightarrow{\alpha} & p \\
A \otimes A & \xrightarrow{p} & A
\end{array}
\]
subject to the following five axioms

An object $A$ of $B$ equipped with such a skew monoidal structure is called a skew monoidal object in $B$.
A skew monoidale in the cartesian monoidal 2-category $\textbf{Cat}$ of categories, functors and natural transformations is a skew monoidal category.

## 3 Span as a Monoidal Bicategory

In this chapter we are interested in the case where $\mathcal{B}$ is $\text{Span}$. We first remind the reader of some details of $\text{Span}$.

The objects of $\text{Span}$ are those of $\textbf{Set}$; so $A, B, C \ldots$ are sets. We denote the terminal object in $\textbf{Set}$ by $1$ and the unique arrow into it by $!$.

An arrow $r: A \rightarrow B$ is a span $r = (f, R, g)$ in $\text{Set}$, as in (a), where composition of these arrows is by pullback (pullback along $g$ and $h$), as in (b), and the identity arrow is the span (c) below.

A 2-cell from $(w, R, x)$ to $(f, S, g)$ is a map $\tau: R \rightarrow S$ in $\textbf{Set}$ such that the following commutes.

As $\textbf{Set}$ is a category with finite products as well as pullbacks (in the presence of a terminal object, finite products can be obtained as a special case of pullbacks) then the bicategory $\text{Span}$ has a monoidal product on it induced by the cartesian product of sets.

To calculate a left whiskering, such as in the following diagram (on the left), we first form the pullbacks of $f$ and $w$ along $v$, then we use the fact that $f\tau = w$ and $v1 = v$ to construct
4 Notation and Calculations

The motivation for this section is from [8] where skew monoidales in Span with a unit of the form $(!, C, 1): 1 \rightarrow C$ are shown to be equivalent to categories. Here we give a characterisation of a general skew monoidale in Span.

Consider a skew monoidale in Span with underlying object the set $C$.

**The 1-cells of a Skew Monoidal:**

The tensor $p: C \times C \rightarrow C$ has the form

$$E_{(s,r)} \xrightarrow{\tau} C \times C \xrightarrow{\tau} C$$

So for $f \in E$ we will record this data as $s(f) \xrightarrow{f} t(f)$ and $r(f) \in C$.

The unit $j: 1 \rightarrow C$ has the form

$$U \xleftarrow{!} 1 \xrightarrow{j} C$$

So for $u \in U$ we will record this data as $j(u) \in C$.

Given a skew monoidale, with its unit having the restricted form $(!, C, 1): 1 \rightarrow C$, it will become evident when dealing with the general case below, that this forces the first span to be of the form $C \times C \xleftarrow{(s,t)} E \xrightarrow{\tau} C$, and that it defines a category with $E$ as its set of arrows. Conversely, given a category $C = (C_1, C_0, 1, s, t, \circ)$, we construct the following two spans: $C_0 \times C_0 \xleftarrow{(s,t)} C_1 \xrightarrow{\tau} C_0$, and $1 \xleftarrow{!} C_0 \xrightarrow{1} C_0$. The 2-cell structure making this category into a skew monoidale comes from the composition and identity arrows of the category, with the skewness arising from the non-symmetric nature of the first span.
The 2-cells of a Skew Monoidale:

What is now required is a long and often repetitive calculation with, when we include the equations between the 2-cells, sixteen pullback constructions in \textbf{Set}; so we will present enough of it to introduce and justify the supporting notation that will form our input for a further characterisation.

For the 2-cell \( \lambda: p(j \times 1) \Rightarrow 1 \) we need to consider the following composite

\[
\begin{array}{c}
U \times C \\
\downarrow \downarrow \\
C \times C \\
\downarrow \downarrow \\
C
\end{array}
\xrightarrow{(s,r)}
\begin{array}{c}
E \\
\downarrow \downarrow \\
C
\end{array}
\]

First we need to form the following pullback

\[
\begin{array}{ccc}
P & \xrightarrow{q} & E \\
\downarrow p & & \downarrow s \\
U & \xrightarrow{j} & C
\end{array}
\]

then the required composite is

\[
\begin{array}{c}
P \\
\downarrow \downarrow \\
C
\end{array}
\xrightarrow{(p,rq)}
\begin{array}{c}
U \times C \\
\downarrow \downarrow \\
C \times C \\
\downarrow \downarrow \\
C
\end{array}
\xrightarrow{(s,r)}
\begin{array}{c}
E \\
\downarrow \downarrow \\
C
\end{array}
\]

so we finally have for the 2-cell \( \lambda \), a function which we also denote by \( \lambda \), such that the following diagram commutes,

\[
\begin{array}{c}
P \\
\downarrow rq \\
C
\end{array}
\xrightarrow{tq}
\begin{array}{c}
C
\end{array}
\]

it can only exist if \( rq = tq \) and is then given as a morphism in \textbf{Set} by the common value

\[
\text{rq} = \text{tq}. \tag{7}
\]

As we are in \textbf{Set} we can write \( P \) as \( P = \{(u, f)| u \in U, f \in E, \quad j(u) = s(f)\} \) with \( p(u, f) = u \) and \( q(u, f) = f \) as the projections. With our notation, the elements in \( P \) look like \( j(u) \xrightarrow{f} y \).
We can now record the effect of $\lambda$ as: $j(u) \xrightarrow{f} y \overset{\lambda}{\longrightarrow} y = r(f)$. Thus the existence of $\lambda$ implies that if $j(u) \xrightarrow{f} y$ then $y = r(f)$, and the map itself sends $(u, j(u) \xrightarrow{f} y)$ to $y$.

In the case of a category (that is, the case where $U$ is $C$ and the unit is of the form $1 \xleftarrow{1} C \rightarrow C$) then $P = E = C_1$ and $j = 1$ forces $r = t$, so $\lambda$ is just $t$.

**For the 2-cell $\rho: 1 \implies p(1 \times j)$** we first need to construct the following pullback

\[
\begin{array}{ccc}
B & \xrightarrow{k} & E \\
\downarrow{m} & & \downarrow{r} \\
U & \xrightarrow{j} & C
\end{array}
\]

In the diagram below

\[
\begin{array}{ccc}
C & \xrightarrow{1} & B \\
\downarrow{(1,\psi)} & & \downarrow{\phi} \\
C \times U & \xrightarrow{(s, m)} & E \\
\downarrow{\text{proj}_1} & & \downarrow{(s, r)} \\
C & \xrightarrow{1} & C
\end{array}
\]

the square is the pullback involved in the composite $p(1 \times j)$, so to give $\rho: 1 \implies p(1 \times j)$ is equivalently to give $\phi: C \rightarrow E$ and $\psi: C \rightarrow U$ satisfying $t\phi = 1$, $s\phi = 1$, and $r\phi = j\psi$. We record for later use that

\[r\phi = j\psi.\]

As we are in $\textbf{Set}$, $B = \{(u, f) | u \in U, f \in E, j(u) = r(f)\}$ with $m(u, f) = u$ and $k(u, f) = f$ as the projections. With respect to our notation, the elements in $B$ look like $(j(u) = r(f), x \xrightarrow{f} y)$ so we record the effect of $\phi$ as

\[x \in C \overset{\phi}{\longrightarrow} (x \xrightarrow{\phi_x} x)\]

then $\psi_x \in U$ satisfies $j(\psi_x) = r(\phi_x)$.

Note that in the case of a category then $B = E = C_1$ and so $\rho$ is just the identity.

**For the 2-cell $\alpha: p(p \times 1) \implies p(1 \times p)$** we need the following two pullbacks

\[
\begin{array}{ccc}
X & \xrightarrow{l} & E \\
\downarrow{h} & & \downarrow{s} \\
E & \xrightarrow{t} & C
\end{array}
\quad
\begin{array}{ccc}
Y & \xrightarrow{y} & E \\
\downarrow{e} & & \downarrow{r} \\
E & \xrightarrow{t} & C
\end{array}
\]
The objects $X$ and $Y$ will appear as the vertex of the spans $p(p \times 1)$ and $p(1 \times p)$, respectively.

In the diagram below

\[ \begin{array}{ccc}
  & X & \\
  \alpha & \downarrow \delta & \downarrow t \gamma \delta \\
  C \times C \times C & Y & E \\
  \downarrow 1 \times t \gamma & \downarrow y & \downarrow t \gamma \\
  C \times C & C & C \\
 \end{array} \]

the square is the pullback involved in the composite $p(1 \times p)$, so to give $\alpha: p(p \times 1) \to p(1 \times p)$ is equivalently to give $\tau: X \to E$ and $\delta: X \to E$ satisfying $t \delta = tl$, $s \delta = sh$, $s \tau = rh$, $r \tau = rl$, and $r \delta = t \tau$.

As we are in $\textbf{Set}$, $X = \{(f, g) | f, g \in E; t(f) = s(g)\}$ with $l(f, g) = g$ and $h(f, g) = f$ as the projections. Similarly, $Y = \{(f, g) | f, g \in E; t(f) = r(g)\}$ with $y(f, g) = g$ and $e(f, g) = f$ as its projections. So with respect to our notation, elements of $X$ look like $x \overset{f}{\to} y \overset{g}{\to} z$ and elements of $Y$ look like $(x \overset{f}{\to} r(g), y \overset{g}{\to} z)$ with $r(f), r(g) \in C$ and we record the effect of $\delta$ as

\[ x \overset{f}{\to} y \overset{g}{\to} z \overset{\delta}{\to} x \overset{gf}{\to} z \]

and $\tau$ as

\[ x \overset{f}{\to} y \overset{g}{\to} z \overset{\tau}{\to} r(f) \overset{gf}{\to} r(gf) \]

with $r(gf) = r(g)$ in $C$.

Note that $\delta$ gives us a map from $x$ to $z$ which we have called $gf$. We want to interpret the set $E$ as a set of arrows and $gf$ as a composite (with $\phi_x$ as an identity), indeed, that this is the composite in a category will be shown below. The map $\tau$ gives us a map from $r(f)$ to $r(gf)$ which we have called $gf$. This map will form the basis of our characterisation for the resulting “extra” structure given on the category.

We now consider the equations between the 2-cells and just do one calculation to give the reader an indication of how the final relations are obtained. Consider the left-hand side of
equation (1) and the whiskering

\[
C \times C \times C \times C \xrightarrow{p \times 1 \times 1} C \times C \times C \xrightarrow{p \times 1} C \times C
\]

\[
1 \times p \quad \quad \alpha \quad \quad p \quad \quad p
\]

\[
C \times C \quad \quad C \times C
\]

For this we need to compose

First form the pullbacks

\[
Q \xrightarrow{\mu} Y
\]

\[
W \xrightarrow{w} X
\]

\[
x \xrightarrow{i} E
\]

\[
x \xrightarrow{i} E
\]

then we have the following pullbacks

\[
Q \xrightarrow{\mu} Y
\]

\[
W \xrightarrow{w} X
\]

\[
E \times C \times C \xrightarrow{t \times 1 \times 1} C \times C \times C
\]

\[
E \times C \times C \xrightarrow{t \times 1 \times 1} C \times C \times C
\]

and so form

\[
E \times C \times C \xrightarrow{t \times 1 \times 1} C \times C \times C
\]

\[
E \times C \times C \xrightarrow{t \times 1 \times 1} C \times C \times C
\]

As before, to give the map

\[
p(p \times 1)/(p \times 1 \times 1) \xrightarrow{\alpha(p \times 1 \times 1)} p(1 \times p)/(p \times 1 \times 1)
\]
lently to give $\gamma: W \rightarrow E$ and $\epsilon: W \rightarrow Y$ as in the diagram below.

```
\begin{center}
\begin{tikzpicture}
  \node (W) at (0,0) {$W$};
  \node (X) at (4,0) {$X$};
  \node (Y) at (2,-2) {$Y$};
  \node (E) at (0,-2) {$E$};
  \node (C) at (-2,-2) {$C$};
  \draw[->] (W) to node {$\gamma=hh'$} (X);
  \draw[->] (W) to node {$\epsilon$} (Y);
  \draw[->] (W) to node {$\tau=\tau_{w}\epsilon$} (E);
  \draw[->] (W) to node {$\tau_{w}=\epsilon\epsilon$} (C);
  \draw[->] (X) to node {$d$} (Y);
  \draw[->] (X) to node {$\tau_{w}$} (E);
  \draw[->] (X) to node {$\tau_{w}$} (C);
  \draw[->] (Y) to node {$e$} (E);
  \draw[->] (Y) to node {$\epsilon$} (C);
  \draw[->] (E) to node {$h$} (C);
  \draw[->] (E) to node {$l$} (Y);
  \draw[->] (E) to node {$s$} (X);
  \draw[->] (E) to node {$r$} (W);
  \draw[->] (C) to node {$t$} (W);
  \draw[->] (C) to node {$a$} (X);
  \draw[->] (C) to node {$b$} (Y);
  \draw[->] (C) to node {$c$} (E);
  \draw[->] (C) to node {$d$} (C);
\end{tikzpicture}
\end{center}
```

From this diagram we have $\gamma = hh'$ and $\epsilon = (\tau, \delta)w = (\tau_{w}, \delta_{w})$. Now writing these as functions into just the set $E$ we recall the previous pullbacks we had constructed and consider the following diagram

```
\begin{center}
\begin{tikzpicture}
  \node (W) at (0,0) {$W$};
  \node (X) at (4,0) {$X$};
  \node (Y) at (2,-2) {$Y$};
  \node (E) at (0,-2) {$E$};
  \node (C) at (-2,-2) {$C$};
  \draw[->] (W) to node {$\gamma=hh'$} (X);
  \draw[->] (W) to node {$\epsilon$} (Y);
  \draw[->] (W) to node {$\tau_{w}=\epsilon\epsilon$} (E);
  \draw[->] (W) to node {$\tau_{w}$} (C);
  \draw[->] (X) to node {$d$} (Y);
  \draw[->] (X) to node {$\tau_{w}$} (E);
  \draw[->] (X) to node {$\tau_{w}$} (C);
  \draw[->] (Y) to node {$e$} (E);
  \draw[->] (Y) to node {$\epsilon$} (C);
  \draw[->] (E) to node {$h$} (C);
  \draw[->] (E) to node {$l$} (Y);
  \draw[->] (E) to node {$s$} (X);
  \draw[->] (E) to node {$r$} (W);
  \draw[->] (C) to node {$t$} (W);
  \draw[->] (C) to node {$a$} (X);
  \draw[->] (C) to node {$b$} (Y);
  \draw[->] (C) to node {$c$} (E);
  \draw[->] (C) to node {$d$} (C);
\end{tikzpicture}
\end{center}
```

From this diagram we have $\gamma = hh'$, $ye = y(\tau_{w}, \delta_{w}) = \delta_{w}$ and $\epsilon = e(\tau_{w}, \delta_{w}) = \tau_{w}$.

As we are in \textbf{Set}, $W = \{(x_1, x_2)\mid x_1, x_2 \in X; l(x_1) = h(x_2)\}$ with projections $h'(x_1, x_2) = x_1$ and $w(x_1, x_2) = x_2$. Similarly, $Q = \{(x, z)\mid x \in X, z \in Y; l(x) = y(z)\}$ with projections $y'(x, z) = x$ and $l'(x, z) = z$. So with respect to our notation, the elements of the set $W$ look like $a \xrightarrow{f} b \xrightarrow{g} c \xrightarrow{k} d$ with $r(f)$, $r(g)$, $r(k) \in C$ and the elements of $Q$ look like $(a \xrightarrow{f} b \xrightarrow{g} c \xrightarrow{k} r(g))$ with $r(f)$, $r(g)$ and $r(k) \in C$. So the 2-cell under consideration gives for $W \rightarrow Q$ that

\[
\begin{aligned}
  a \xrightarrow{f} b &\xrightarrow{g} c \xrightarrow{k} d \\
  (a \xrightarrow{f} b \xrightarrow{k} r(k), b \xrightarrow{k} d) &\xrightarrow{r(g)} (r(g), r(k) \xrightarrow{k} r(k))
\end{aligned}
\]
Now observe that the two sides of the cube 1 act as in the diagram below,

\[
\begin{array}{c}
\xymatrix{\text{\(p(\alpha \otimes 1)\)}} & (gf, g^f, h) \ar[rr]^-{\alpha(1 \otimes p \otimes 1)} \ar[dr]_p & & (gf, h g^f, h(gf)) \\
(f, g, h) \ar[ur]^{\alpha(p \otimes 1 \otimes 1)} \ar[d]_{\alpha(p \otimes 1 \otimes 1)} & & ((h g^f)^g f, h g^f, h(gf)) \\
(f, h^g, h g) \ar[ur]^{\alpha(1 \otimes 1 \otimes p)} & & (h^g, (h g)^f, (h g) f) \\
\end{array}
\]

so the cube commutes if and only if the two expressions in the lower right corner agree; in other words, if the following equations, \(h^g = (h g^f)^g f\), \((h g)^f = h g^f\), and \((h g) f = h(g f)\) hold, for a composable triple of arrows. The remaining four equations are analyzed similarly, and the results summarized below.

**Summary:** We summarize all the calculations with respect to a skew monoidale into the notation introduced earlier to get:

For the 1-cell \(p: C \times C \longrightarrow C\) with vertex \(E\): For \(f \in E\), \(x \xrightarrow{f} y\) for \(x, y \in C\) and \(r(f) \in C\).

For the 1-cell \(j: 1 \longrightarrow C\) with vertex \(U\): For \(u \in U\) that \(j(u) \in C\).

For the 2-cell \(\lambda\): if \(j(u) \xrightarrow{f} y\) then \(y = r(f)\) in \(C\).

For the 2-cell \(\rho\): for \(x \in C\) we have \(x \xrightarrow{\phi_x} x\) in \(E\) and \(\psi_x \in U\) with \(j(\psi_x) = r(\phi_x)\).

For the 2-cell \(\alpha\): if \(x \xrightarrow{f} y \xrightarrow{g} z\) then \(r(f) \xrightarrow{g^f} r(g f)\) and \(x \xrightarrow{g^f} z\) are both in \(E\) with \(r(g^f) = r(g)\).

For the equation between the 2-cells involving \((\lambda, \rho)\): For \(j(u) \in C\) we have \(\psi j(u) = u\), that is, \(\psi j = 1\).

For the equation between the 2-cells involving \((\rho, \alpha)\): For \(x \xrightarrow{f} y\) we have \(\psi y = \psi r(f)\), \(\phi_y f = f\), and \(\phi_y = \phi_r\).

For the equation between the 2-cells involving \((\lambda, \alpha)\): For \(j(u) \xrightarrow{f} y \xrightarrow{g} z\) we have \(g^f = g\).

For the equation between the 2-cells involving \((\rho, \alpha, \lambda)\): For \(x \xrightarrow{f} y\) we have \(f \phi_x = f\).

For the equation between the 2-cells involving \((\alpha, \alpha)\) (the pentagon):

For \(x \xrightarrow{f} y \xrightarrow{g} z \xrightarrow{h} a\) we have \((h g) f = h(g f)\), \((h g)^f = h g f\), and \(h^g = (h g^f)^g f\).

We conclude that we can now safely rename \(\phi_x\) as \(1_x\) and change our notation for \(x \xrightarrow{f} y\) to an arrow \(x \xrightarrow{f} y\) and with the condition that \((h g) f = h(g f)\) obtain a category with some
extra structure consisting of:

(a) for each morphism $f$ an object $r(f)$.

(b) a set $U$ with a function $j$ from $U$ to the set of objects.

(c) for each composable pair $x \xrightarrow{f} y \xrightarrow{g} z$ a map $r(f) \xrightarrow{g^f} r(gf)$ with $r(gf) = r(g)$.

(d) for each object $c$ an element $\psi_c \in U$.

satisfying the following

\begin{align*}
\text{For } u \in U & \quad \text{that } \psi_j(u) = u. \quad (10) \\
\text{For } x \in C & \quad \text{that } r(1_x) = j\psi. \quad (11) \\
\text{For } j(u) \xrightarrow{f} y \xrightarrow{g} z & \quad \text{that } g^f = g. \quad (12) \\
\text{For } x \xrightarrow{f} y, r(f) \in C & \quad \text{that } 1^f_y = 1_{r(f)}. \quad (13) \\
\text{For } x \xrightarrow{f} y, r(f) \in C & \quad \text{that } \psi_y = \psi_{r(f)}. \quad (14) \\
\text{For } x \xrightarrow{f} y \xrightarrow{g} z \xrightarrow{h} a & \quad \text{that } (hg)^f = h^g g^f. \quad (15) \\
\text{For } x \xrightarrow{f} y \xrightarrow{g} z \xrightarrow{h} a & \quad \text{that } h^g = (h^g) g^f. \quad (16)
\end{align*}

Before we consider these equations again, we notice that from (10) $j$ is already injective.

**Lemma 4.1.** If $j$ is surjective then $r = t$.

**Proof.** If $j$ is surjective then by (6) $q$ is also surjective. Since $rq = tq$ by (7), we can conclude that $r = t$. \qed

So with the assumption that $j$ is surjective we see that a skew monoidal in Span is precisely a category. The extra structure given by $\tau$ and the map $g^f$ reduces to $g^f = g$ for all $f, g \in E$ by (12). This recovers the result in [8] where the skew monoidal in Span assumed the unit was of the form

\[
\begin{array}{c}
1 \\
\downarrow \quad \downarrow \\
1 & \rightarrow \quad \rightarrow \\
\downarrow & \\
\rightarrow C & \\
\end{array}
\]

5 A Characterisation

5.1 Coslice Category

In this subsection we use the notation of [9] to denote the coslice category or undercategory of a category, which we now define.
Let $C$ be a category and $x$ an object of $C$, then the coslice category denoted by $(x \downarrow C)$ has objects the arrows of $C$ with source $x$, that is, $x \xrightarrow{f} y$ which we sometimes denote by the pairs $(f, y)$; and arrows those $g: (f, y) \to (f', z)$ where $y \xrightarrow{g} z$ is an arrow of $C$ such that $f' = gf$, which we usually denote as $(f, y) \xrightarrow{g} (gf, z)$. It is useful sometimes to write these arrows as the following triangles

\[
\begin{array}{ccc}
x & \xrightarrow{f} & y \\
\downarrow{g} & & \downarrow{g} \\
gf & \xrightarrow{gf} & z
\end{array}
\]

There is an evident functor $\text{Cod}_x: (x \downarrow C) \to C$ defined on objects by $x \xrightarrow{y} y$ and on arrows by $(f, y) \xrightarrow{g} (gf, z) \mapsto y \xrightarrow{g} z$.

**Note:** Let $A$ and $B$ be categories and $x$ an object of $A$. For a functor $T: A \to B$ there is an induced functor $(x \downarrow A) \xrightarrow{(x \downarrow T)} (Tx \downarrow B)$ sending an object $x \xrightarrow{y} y$ to $Tx \xrightarrow{Ty} Ty$ and an arrow

\[
\begin{array}{ccc}
x & \xrightarrow{f} & y \\
\downarrow{g} & & \downarrow{g} \\
gf & \xrightarrow{gf} & z
\end{array} \mapsto \begin{array}{ccc}
Tx & \xrightarrow{Tf} & Ty \\
\downarrow{Tg} & & \downarrow{Tg} \\
T(gf) & \xrightarrow{T(gf)} & Tz
\end{array}
\]

Let $C$ be a category and $f: x \to y$ be an object of $(x \downarrow C)$; we remind the reader of the coslice category $(f \downarrow (x \downarrow C))$. This category has as its objects the morphisms in $(x \downarrow C)$ starting at $f$ denoted by $f \xrightarrow{g} gf$ and as its morphisms the commuting triangles between its objects which we denote by

\[
\begin{array}{ccc}
f & \xrightarrow{hg} & hgf \\
g & \xrightarrow{g} & hgf \\
gf & \xrightarrow{hgf} & hgf
\end{array}
\]

we sometimes denote them by $g \xrightarrow{h} hg$.

The functor $(f \downarrow \text{Cod}_x): (f \downarrow (x \downarrow C)) \to (y \downarrow C)$ is invertible; it sends an object $f \xrightarrow{g} gf$ to $g$ and a morphism

\[
\begin{array}{ccc}
f & \xrightarrow{hg} & hgf \\
g & \xrightarrow{g} & hgf \\
gf & \xrightarrow{hgf} & hgf
\end{array} \mapsto \begin{array}{ccc}
\text{Cod}(f) & \xrightarrow{hg} & \text{Cod}(hgf) \\
g & \xrightarrow{g} & \text{Cod}(hgf) \\
\text{Cod}(gf) & \xrightarrow{hgf} & \text{Cod}(hgf)
\end{array}
\]
5.2 The Functor $R_x$

From the previous sections we have seen that a skew monoidale $C$ in $\text{Span}$ gives rise to a category $\mathcal{C}$ with some extra structure via the function $g^f$ and equations (10) - (16). In this section we use some of these equations to obtain a functor from a coslice category of $\mathcal{C}$ to $\mathcal{C}$ and relate the remaining equations to this functor.

For $x \in \mathcal{C}$ we use equations (13) and (15) to define a functor $R_x : (x \downarrow \mathcal{C}) \rightarrow \mathcal{C}$ sending an object $x \xrightarrow{f} y$ to $r(f)$ and an arrow $(f, y) \xrightarrow{g} (gf, z)$ to $r(f) \xrightarrow{g^f} r(gf)$. When it is clear in context we write that on the objects $R_x(f) = r(f)$ and on the arrows $R_x(g) = g^f$.

We check that we do have a functor.

We have by definition that $R_x(hg) = (hg)^f$ and $R_x(h)R_x(g) = h^{gf}g^f$ and by (15) these agree so that $R_x$ preserves composition. Similarly by (13), $R_x$ preserves identities and so is a functor.

We now express equation (16) in terms of the functor $R_x$. However for the benefit of the reader we will explicitly describe the functor $(f \downarrow R_x f) : (f \downarrow (x \downarrow \mathcal{C})) \rightarrow (R_x f \downarrow \mathcal{C})$ which is defined on objects by $f \xrightarrow{g} gf \mapsto r(f) \xrightarrow{g^f} r(gf)$ and on arrows by

$$
\begin{array}{ccc}
\begin{array}{ccc}
f & \xrightarrow{g} & gf \\
\downarrow{g} & & \downarrow{h} \\
hg & & hg \ \\
\end{array}
& \mapsto & \begin{array}{ccc}
r(f) & \xrightarrow{g^f} & r(gf) \\
\downarrow{g^f} & & \downarrow{hg^f} = (hg)^f \\
\end{array}
\end{array}
$$

The above remark allows us to conclude that equation (16) asserts that the following diagram commutes (it agrees on objects since $r(g^f) = r(g)$).

$$
\begin{array}{ccc}
(y \downarrow \mathcal{C}) & \xrightarrow{R_y} & \mathcal{C} \\
(f \downarrow \text{Cod}_x) & \downarrow{R_{R_x f}} & \downarrow{R_{R_x f}} \\
(f \downarrow (x \downarrow \mathcal{C})) & \xrightarrow{(f \downarrow R_x f)} & (R_x f \downarrow \mathcal{C})
\end{array}
$$

In the following section we consider the remaining structure involving $U$, $j$, and $\psi$. 

14
5.3 The Function $E$

We define a function $E$ on the set of objects of the category $\mathcal{C}$ by $E(x) = r(1_x)$. Using (13) and $r(g') = r(g)$ (for a composable pair of morphisms), we note that if $x \xrightarrow{f} y$ then $E(r(f)) = E(y)$. Taking $f = 1_x$ we find that $E(E(x)) = E(r(1_x)) = E(x)$, so $E$ is idempotent.

From equation (10), $\psi j = 1$, and equation (11), $j \psi_x = r(1_x)$, we can define $U$, $j$, and $\psi$ as a splitting of $E$. So in terms of the functor $R_x$ we have $E(x) = R_x(1_x)$ for each object $x$ in the category $\mathcal{C}$. With this notation, equation (14) then asserts that the following diagram commutes on the objects of the respective categories:

\[
\begin{array}{ccc}
\text{Ob}(x \downarrow \mathcal{C}) & \xrightarrow{R_x} & \text{Ob}(\mathcal{C}) \\
\text{Cod} & \downarrow & \\
\text{Ob}(\mathcal{C}) & \xrightarrow{E} & \text{Ob}(\mathcal{C})
\end{array}
\]

Following an object $x \xrightarrow{f} y$ of $(x \downarrow \mathcal{C})$ around (18) then asserts in terms of the functor $R_x$ that $R_y(1_y) = R_{R_x f}(1_{R_x f})$ and as $R_x$ is a functor we also have $R_{R_x f}(1_{R_x f}) = R_{R_x f}(R_x(1_y))$.

However if we follow the object $y \xrightarrow{1_y} y$ of $(y \downarrow \mathcal{C})$ around (17) (really we follow $f \xrightarrow{1_y} 1_y f$ of $(f \downarrow (x \downarrow \mathcal{C}))$ around (17)) we get that $R_y(1_y) = R_{R_x f}(R_x(1_y))$. So we have shown:

**Lemma 5.1.** If (17) holds then so does (18).

We now consider the remaining equation (12) in terms of the functor $R_x$. It is the statement that if for $j(u) \xrightarrow{f} y \xrightarrow{g} z$ then $g' = g$.

As $\psi j = 1$ it can be shown that $x = j \psi x$ if and only if there exist a $u$ such that $x = j u$. So for the $u$ where $x = j u$ then $x = E(x) = R_x(1_x)$ (We could now define $U$ to be those $x$ for which $x = R_x(1_x)$). So we conclude that (12) is the statement that if $x = R_x(1_x)$ then $R_x = \text{Cod}_x$.

**Conclusion:** A skew monoidale $C$ in Span amounts to a category $\mathcal{C}$ with

(a) a functor $R_x : (x \downarrow \mathcal{C}) \rightarrow \mathcal{C}$ for each $x$ in $\mathcal{C}$.

(b) if $x = R_x(1_x)$ then $R_x = \text{Cod}_x$.

(c) $R_x$ satisfies (17), that is, for an arrow $x \xrightarrow{f} y$ in $\mathcal{C}$ the following commutes

\[
\begin{array}{ccc}
(y \downarrow \mathcal{C}) & \xrightarrow{R_y} & \mathcal{C} \\
(f \downarrow \text{Cod}_x) & \xrightarrow{R_{R_x f}} & (R_x f \downarrow \mathcal{C})
\end{array}
\]
Note: For each $x \in C$, the case when $j = 1$ (equivalently, $j$ is surjective) corresponds to $R_x = \text{Cod}_x$.

5.4 The Simplicial category and the Decalage Functor

We recall some standard facts about the simplex category $\Delta$, before using it in our characterisation. There are many references for this section we use [9] and [5].

The simplicial category $\Delta$ has as objects the finite ordinals $n = \{0, 1, \ldots, n - 1\}$ and morphisms the order-preserving functions $\xi : m \to n$ with composition that of functions; the composite of order preserving functions is again order preserving. We note that the ordinal numbers $0$ and $1$ are respectively, initial and terminal objects in $\Delta$.

If $0 \leq i \leq n$, we write $\delta_i : n \to n+1$ for the injective order-preserving function where $\delta_i(k)$ is equal to $k$ if $k < i$ and $k + 1$ otherwise (thus its image omits $i$). Similarly, if $0 \leq i \leq n - 1$, we write $\sigma_i : n+1 \to n$ for the order-preserving surjective function where $\sigma_i(k)$ is equal to $k$ if $k \leq i$ and $k - 1$ otherwise (thus $\sigma_i(i) = \sigma_i(i+1)$, that is, it repeats $i$). We call these maps coface and codegeneracy maps respectively and they satisfy the well known simplicial identities which allow for a presentation of $\Delta$ with the $\delta_i$ and $\sigma_i$ as its generators and the simplicial identities as its relations. Moreover, $\Delta$ has a strict (non-symmetric) monoidal structure $(\Delta, +, 0)$ given by ordinal addition $+: \Delta \times \Delta \to \Delta$, defined on ordinals as the ordered sum and on arrows by placing them "side by side". So in terms of the presentation we have that $1_m + \delta_i = \delta_{m+i}$, $1_m + \sigma_i = \sigma_{m+i}$, $\delta_i + 1_m = \delta_i$, and $\sigma_i + 1_m = \sigma_i$ where $1_m$ denotes the identity on $m$; see [9].

A simplicial set is a contravariant functor from $\Delta$ to $\text{Set}$. The category $\text{Simp}$ of simplicial sets and simplicial maps between them is defined to be the functor category $[\Delta^{\text{op}}, \text{Set}]$. For a functor $S : \Delta^{\text{op}} \to \text{Set}$ we write $S_n$ for $S(n)$. It can be shown that the data for a simplicial set can be specified by the sets $S_n$ and maps $d_i : S_n \to S_{n-1}$ and $s_i : S_n \to S_{n+1}$ where for $0 \leq i \leq n$ we define $d_i$ as $S\delta_i$ and $s_i$ as $S\sigma_i$. We call these face and degeneracy maps and they satisfy relations dual to those in $\Delta$.

For a simplicial set $S$ we consider the shift or (left) decalage functor $\text{Dec} : \text{Simp} \to \text{Simp}$ which removes the 0-th face and degeneracy maps, shifts dimension so that $(\text{Dec}(S))_n = S_{n+1}$ and shifts indices on the remaining face and degeneracy maps down by 1 so that $d_i : (\text{Dec}(S))_n \to (\text{Dec}(S))_{n-1}$ is $d_{i+1} : S_{n+1} \to S_n$ and $s_i : (\text{Dec}(S))_n \to (\text{Dec}(S))_{n+1}$ is $s_{i+1} : S_{n+1} \to S_{n+2}$. 

16
Given a simplicial set $S$ as in the diagram

\[ S: \quad \cdots S_n \longrightarrow S_{n-1} \longrightarrow \cdots S_1 \longrightarrow S_0 \]

The decalage $\text{Dec}(S)$ of $S$ is the simplicial set

\[ \text{Dec}(S): \quad \cdots S_{n+1} \longrightarrow S_n \longrightarrow \cdots S_3 \longrightarrow S_2 \longrightarrow S_1 \]

There is a simplicial map $\text{Dec}(S) \to S$ given (in degree $n$) by the original face map that was discarded $d_0: S_{n+1} \to S_n$; we write this map as $d_0: \text{Dec}(S) \to S$. The above explicit description of the left decalage construction has a right version and left and right versions for the case of augmented simplicial sets.

There is a comonad underlying the decalage functor which we briefly describe. Since $1$ is terminal in $\Delta$, the arrows $\delta_0: 0 \to 1$ and $\sigma_0: 2 \to 1$ form the "universal" monoid $(1, \sigma_0, \delta_0)$ in $\Delta$. Moreover, there is a monad on $\Delta$ with endofunctor part $- + 1: \Delta \to \Delta$, multiplication $- + \sigma_0$, and unit $- + \delta_0$. Now, by reversing the order of each ordinal, $\Delta^{op}$ contains the universal comonoid $1$, and as a result we can form a comonad $- + 1: \Delta^{op} \to \Delta^{op}$ dual to the previous construction. This induces by precomposition with the above comonad a comonad on $\text{Simp}$ whose underlying endofunctor is $\text{Dec}: \text{Simp} \to \text{Simp}$. The counit of the comonad $\text{Dec}$ is $d_0: S_{n+1} \to S_n$.

Given a category $C$ we can form the nerve $N(C)$ of $C$, it is the well known simplicial set where the face and degeneracy maps are those given in [9] and where

$N(C)_0 = \text{set of objects in } C$

$N(C)_1 = \text{set of morphisms in } C$

$N(C)_2 = \text{set of composable pairs of morphisms in } C$

$\vdots$

$N(C)_n = \text{set of composable } n\text{-tuples of morphisms in } C$.

With the above discussion in mind we see that if $C$ is a category then so is $\text{Dec}(C)$ where
\[ \text{Dec}(C)_0 = \text{set of morphisms in } C \]
\[ \text{Dec}(C)_1 = \text{set of composable pairs of morphisms in } C \]
\[ \vdots \]
\[ \text{Dec}(C)_n = \text{set of composable } (n+1)\text{-tuples of morphisms in } C. \]

Recall that in the category \textbf{Cat} of small categories and functors, the coproduct of a family of categories is their disjoint union. For \( I \) a set and \((C_i)_{i \in I}\) a family of objects in \textbf{Cat} we write \( \coprod_{i \in I} C_i \) for the coproduct of the family \((C_i)_{i \in I}\). Now with this notation and from the functors \( \text{Cod} \) we can form a functor from \( \coprod_{x \in C} (x \downarrow C) \) to \( C \), which we denote by \( \text{Cod} \).

Having described above what the functor \( \text{Dec} \) does on objects of \textbf{Cat} we notice for a category \( C \), that \( \text{Dec}(C) = \coprod_{x \in C} (x \downarrow C) \). So to complete this (brief) description of \( \text{Dec} \) as an endofunctor from \textbf{Cat} we need to describe what it does on arrows of \textbf{Cat}.

Let \( F : X \to C \) be a functor where \( X \) and \( C \) are categories. As we need a functor from a coproduct in \textbf{Cat}, it is sufficient, for each \( x \in X \), to specify a functor from \( (x \downarrow X) \) to \( \text{Dec}(C) \) where \( \text{Dec}(C) = \coprod_{c \in C} (c \downarrow C) \). We define the functor \( \text{Dec}(F)_x : (x \downarrow X) \to \text{Dec}(C) \) by the following composite

\[ (x \downarrow X) \xrightarrow{(x \downarrow F)} (F(x) \downarrow C) \xrightarrow{\text{inclusion}} \text{Dec}(C) \]

Thus we have a functor \( \text{Dec}(F) : \text{Dec}(X) \to \text{Dec}(C) \).

We complete this description of \( \text{Dec} \) as an endofunctor on \textbf{Cat} with the observation that \( \text{Cod} : \text{Dec}(C) \to C \) is the map \( d_0 : \text{Dec}(C) \to C \) and note that any coslice category can be extracted from \( \text{Dec}(C) \) using this \( d_0 \).

Using these constructions we can rewrite the previous description of a skew monoidale in \textbf{Span} as:

**Conclusion:** A skew monoidale \( C \) in \textbf{Span} amounts to a category \( \mathcal{C} \) with

(a) a functor \( R : \text{Dec}(\mathcal{C}) \to \mathcal{C} \), where

(b) \( R \) makes the following diagram commute

\[ \begin{array}{ccc}
\text{Dec}(\mathcal{C}) & \xrightarrow{R} & \mathcal{C} \\
\text{Dec(Cod)} \downarrow & & \downarrow R \\
\text{Dec(Dec(\mathcal{C}))} & \xrightarrow{\text{Dec}(R)} & \text{Dec(\mathcal{C})}
\end{array} \]

(c) such that, if \( x = R(1_x) \) then \( R_x = \text{Cod}_x \).

Note that when starting with just a category then \( R = \text{Cod} \).
6 An Example

In this section we denote by \((M, \mu, \eta)\) or just \(M\) a monoid in the monoidal category \((\text{Set}, \times, 1)\) where the tensor product is the cartesian product \(\times\) and \(1 = \{\star\}\) denotes a one point set as its unit. Here the two arrows \(\mu\) and \(\eta\) in \(\text{Set}\) satisfy the usual equations (see [9]). For \(\mu: M \times M \to M\) and for \(a, b \in M\) we write \(\mu(a, b) = a \cdot b\) and write for \(\eta(\{\star\}) = 1_M\), we sometimes just write \(\eta(\{\star\}) = 1\) where it should be clear in context what \(1\) represents.

We recall the embedding \((-)_*: \text{Set} \to \text{Span}\) which is the identity on objects and assigns to the morphism \(f: A \to B\) the following span

\[
\begin{array}{ccc}
A & \xrightarrow{f} & B \\
\downarrow{1_A} & \nearrow{\pi_2} \\
A & \to & M \\
\end{array}
\]

In fact, this is a strong monoidal pseudofunctor and as a consequence sends monoids in \(\text{Set}\) to monoidales in \(\text{Span}\). We can therefore consider a monoid \((M, \mu, \eta)\) in \(\text{Set}\) as a (skew) monoidale in \(\text{Span}\).

The 1-cell \(p: C \times C \to C\) for a skew monoidale in \(\text{Span}\) is given by

\[
\begin{array}{ccc}
M \times M & \xrightarrow{\mu} & M \\
\downarrow{(\pi_1, \pi_2)} & \nearrow{\mu} \\
M \times M & \to & M \\
\end{array}
\]

where \(\pi_i: M \times M \to M\) is defined by \(\pi_i(m_1, m_2) = m_i\) for \(i=1,2\) and \(m_1, m_2 \in M\).

The 1-cell \(j: 1 \to C\) for a skew monoidale in \(\text{Span}\) is given by

\[
\begin{array}{ccc}
1 & \xrightarrow{1} & \eta \\
\downarrow{1} & \nearrow{\eta} \\
1 & \to & M \\
\end{array}
\]

With these choices for \(p\) and \(j\), the 2-cell \(\rho: 1 \Rightarrow p(1 \times j)\) for this skew monoidale is given by
the following diagram

and the 2-cell $\alpha: p(p \times 1) \Rightarrow p(1 \times p)$ is given by

where $\pi_{23}: M \times M \times M \rightarrow M \times M$ is defined as $\pi_{23}(m_1, m_2, m_3) = (m_2, m_3)$. We will now describe the resulting monoidale in terms of the characterisation of skew monoidales in Span given in the previous sections.

So with these choices for $p$ and $j$, $M$ is a category whose objects are the elements of the set $M$ and whose arrows are the pairs $(a, b) \in M \times M$ with source $\pi_1(a, b) = a$ and target $\mu(a, b) = a.b$ which we represent as $a \xrightarrow{b} a.b$. The composition of arrows in $M$ and the functor $R$: $\text{Dec}(M) \rightarrow M$ are both defined by the 2-cell $\alpha: p(p \times 1) \Rightarrow p(1 \times p)$. The composition of arrows in $M$ is then given by $(a, b, c) \xrightarrow{1 \times \mu} (a, b, c)$ for $(a, b, c) \in M \times M \times M$ and so the composite $a \xrightarrow{b} a.b \xrightarrow{c} (a.b).c$ is given by $a \xrightarrow{b.c} a.(b.c)$.

For the functor $R$: $\text{Dec}(M) \rightarrow M$ and the $p$ and $j$ chosen from $M$ we have on the objects of $\text{Dec}(M)$ that $R((a, b)) = \pi_2(a, b) = b$ or $R( a \xrightarrow{b} a.b ) = b$ and on the arrows of $\text{Dec}(M)$ we have that $R((a, b, c)) = \pi_{23}(a, b, c) = (b, c)$ or $R( a \xrightarrow{b} a.b \xrightarrow{c} (a.b).c ) = b \xrightarrow{c} b.c$.

The identity arrow for the category $M$ exists via the 2-cell $\rho: 1 \Rightarrow p(1 \times j)$ and is represented as $a \xrightarrow{1} a.1 = a$.

**Remark 6.1.** The monoids in $\textbf{Set}$ constitute a category $\textbf{Mon}$ and the above example defines the object part of a functor $T$: $\textbf{Mon} \rightarrow \textbf{Cat}$. For a morphism of monoids $f: (M, \mu, \eta) \rightarrow (M', \mu', \eta')$ the induced functor $TM \rightarrow TM'$ sends an object $m$ to $fm$ and a morphism $(m, n)$ to $(fm, fn)$. 

-
Remark 6.2. Considering a category as a partial monoid and using the notation of [9] we can generalize the above example; we can instead start with a (small) category $C$ where $O$, $A$ and $A \times_O A$ respectively denotes the sets of objects, arrows and composable arrows of $C$.

The tensor for a monoidale in $\text{Span}$ is given by

\[
\begin{array}{ccc}
A \times_O A & \xrightarrow{\text{comp}} & A \\
\xrightarrow{(\pi_1, \pi_2)} A \times A & \xrightarrow{\text{comp}} & A
\end{array}
\]

The unit for that monoidale in $\text{Span}$ is given by

\[
\begin{array}{ccc}
O & \xrightarrow{id} & A \\
\xrightarrow{!} 1 & \xrightarrow{id} & A
\end{array}
\]

Remark 6.3. The following observations are due to Joachim Kock who has noted that for the example starting with a monoid $M$ the above construction of a (skew) monoidale is just the category $\text{Dec}(M)$ and the functor $T : \text{Mon} \rightarrow \text{Cat}$ is the restriction of the functor $\text{Dec} : \text{Cat} \rightarrow \text{Cat}$. Similarly, the example starting with a category $C$, the corresponding construction of a (skew) monoidale is $\text{Dec}(C)$. Now starting with a category $C$ we consider the category $D = \text{Dec}(C)$ so now using our previous notation $D_0 = C_1$, $D_1 = C_2$ and so on. The extra structure required on this category $D$ to be considered as a skew monoidale in $\text{Span}$ is a functor $R : \text{Dec}(D) \rightarrow D$ which amongst other conditions agrees with the counit $d_0$ (that is with $\text{Cod} : \text{Dec}(D) \rightarrow D$) on the objects $U$ in $D$ but now since $D = \text{Dec}(C)$ this $d_0$ from $D$ is $d_1$ from $C$. Now a natural choice for the functor $R$ would be $\text{Dec}(d_0)$ where $d_0$ from $C$ and to define $U$, (which are the objects $x$ in $D$ for which $x = R_x(1_x)$ and since $C_1 = D_0$ we can use $\text{Dec}(s_0)$ where $s_0 : C_0 \rightarrow C_1$ is from $C$, previously omitted by $\text{Dec}$. Hence, for the above examples, the extra face map $R$ is available naturally resulting in the monoidale being non-skew.

Remark 6.4. The following is a non-trivial example given by Stephen Lack at a talk to the Australian Category Seminar [7]. Batanin and Markl in [1] define a strict operadic category as a category $\mathcal{C}$ equipped with a cardinality functor into $\text{sFSet}$, the skeletal category of finite sets, where each connected component of $\mathcal{C}$ has a chosen terminal object. One of the axioms for a strict operadic category requires the existence of a family of functors from a slice category of $\mathcal{C}$ into $\mathcal{C}$, for the chosen terminal object this is required to be the domain functor. Lack has shown that strict operadic categories are equivalent to left normal skew monoidales in $\text{Span}([\mathbb{N}, \text{Set}])$. Here $\mathbb{N}$ denotes the set of natural numbers, seen as a discrete category, and the functor category $[\mathbb{N}, \text{Set}]$ is given a monoidal structure via Day’s convolution.
Acknowledgements:

I wish to thank my supervisor, Stephen Lack, whose patience and guidance made possible the writing of this article and Joachim Kock for his comments.

References

[1] M. Batanin and M. Markl. Operadic categories and duoidal Deligne’s conjecture. arXiv:1404.3886v2 (2014).

[2] J. Bénabou. Introduction to bicategories. Reports of the Midwest Category Seminar. LNM Vol. 47 (1967) 1–77.

[3] Mitchell Buckley, Richard Garner, Stephen Lack and Ross Street. Skew monoidal categories and the Catalan simplicial set. arXiv:1307.0265v1 (2013) preprint.

[4] Brian J. Day and Ross Street. Monoidal bicategories and Hopf algebroids. Advances in Math., 129 (1997) 99–157.

[5] John W. Duskin. Simplicial matrices and the nerves of weak n-Categories I: Nerves of Bicategories. Theory and Applications of Categories 9(10) (2002) 198–308.

[6] Robert Gordon, A. John Power and Ross Street. Coherence for tricategories. Memoirs of the American Mathematical Society, 117 no. 558 (1995) vi+81 pp.

[7] Stephen Lack. Skew aspects of operads. Talk in the Australian Category Seminar. (2 April 2014).

[8] Stephen Lack and Ross Street. skew monoidales, skew warpings and quantum categories. Theory and Applications of Categories, 26:385–402, 2012.

[9] Saunders Mac Lane. Categories for the working mathematician. Graduate Texts in Mathematics 5 (Springer-Verlag, 1971).