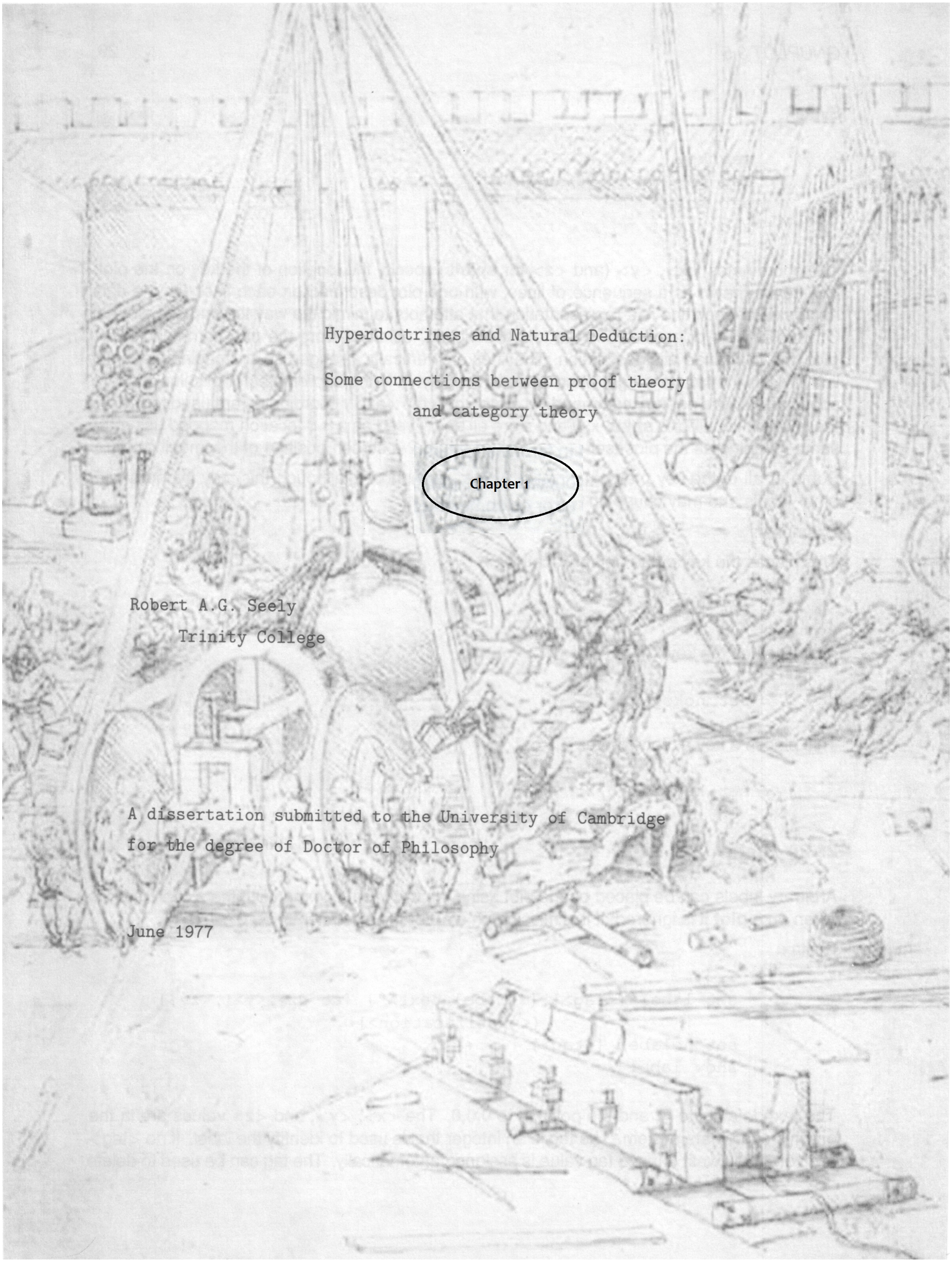


Hyperdoctrines and Natural Deduction:

Some connections between proof theory and category theory

Chapter 1

Robert A.G. Seely
Trinity College

A dissertation submitted to the University of Cambridge for the degree of Doctor of Philosophy

June 1977

Chapter I

§0 Introduction

In this chapter we will analyse the 2-categorical structure of the proof theory of the intuitionistic lower predicate calculus (LPC). We could treat ordinary first order logic, either classical or intuitionistic, in this way, but instead we will discuss a modification of the usual logic which will be of more use in Chapter II, as follows.

It is by now well known (at least among "categorical logicians") that the normal treatment of logic must be modified if we wish to apply it to structures with domains not necessarily non-empty. (See, for example, Ouellet [33], Joyal [18], Coste [11], Boileau [05], Lawvere [26], Fourman [13], etc.) The usual modification is along lines rather like the following: instead of considering sequents (or entailments): $\phi \longrightarrow \phi'$, we consider sequents (entailments) "at level U": $\phi \underset{U}{\longrightarrow} \phi'$, where U is a sequence (or set) of types, so that the free variables of ϕ and ϕ' correspond to types occurring in U. Then the rules of inference and the axioms are modified accordingly, bearing in mind that a type need not be non-empty in an interpretation. This is very informal, so perhaps an example is in order. Suppose $\phi(x)$ is a formula with the free variable x of type X. Then we do not want to be able to derive $\forall x\phi \longrightarrow \exists x\phi$ at the level of sentences ("at level 1"), for this would be false if X was empty. On the other hand, $\forall x\phi \longrightarrow \phi$ and $\phi \longrightarrow \exists x\phi$ should hold at level X; modus ponens then only yields $\forall x\phi \longrightarrow \exists x\phi$ at level X, which is unobjectionable (or at least less objectionable), as it is vacuous when X is empty.

In many, (if not most) text-books on logic, the notation $\phi(x,y,\ldots,z)$ is used to mean that the free variables of ϕ occur among $x,y,\ldots,z$. In fact, we can restrict this to mean the free variables of ϕ are exactly $x,y,\ldots,z$ without any change in practice if we introduce projections as terms. So, for example, if x doesn't really occur in ϕ, then ϕ can be replaced by $\phi(\pi(x,y,\ldots,z))$, for π the projection term $\pi(x,y,\ldots,z)=(y,\ldots,z)$. (Of course, x does appear in $\phi(\pi)$, if only vacuously.) Conversely, we can use this process of substituting a projection term to add dummy free variables, (i.e., ones that don't really appear.) All this becomes much simpler formally if we use the pair operation <- , -> to enable us to consider all terms and formulae as having (at most) one free variable. (So $\phi(x,y)$ becomes $\phi(\langle x,y\rangle)$, also written $\phi(x,y)$.)

These considerations come together when we consider an entailment of the following form:

$$E(x,x') \wedge E(x',x'') \longrightarrow E(x,x'') \qquad (*)$$

where E is a formula with two free variables each of type X. In such an expression we are actually considering E as a formula in three free variables, although we omit the third, second, and first in each successive appearance of E. So if π_{12}, π_{23}, and π_{13} are the three evident projections $X \times X \times X \to X \times X$,

(*) is merely shorthand for:

$$E(\pi_{12}(x,x',x'')) \wedge E(\pi_{23}(x,x',x'')) \longrightarrow E(\pi_{13}(x,x',x''))$$

However, having said all this, for the purposes of this chapter, these modifications can be ignored, and the reader may suppose we are dealing with first order logic as presented in Prawitz [35] or [36], without going too far astray.

§1 LPC

1.0 We suppose our language L to have:

sort symbols $X,Y,Z,\ldots$

free variables of each sort $x,x',\ldots,y,\ldots,z,\ldots$

bound variables of each sort $\xi,\xi',\ldots,\eta,\ldots,\zeta,\ldots$

sorted function symbols $f,g,\ldots$

sorted predicate symbols $P,\Omega,\ldots$

For logical connectives we take $\top,\bot,\wedge,\vee,\supset,\ \exists\ ,\forall$.

We wish to be able to suppose that our function and predicate symbols are unary, and we want terms and formulae to be labelled with the sorts of their free variables, and terms to have a sort themselves. Furthermore, we want to be able to add dummy free variables as discussed in the introduction. To this end, we introduce the metamathematical notion of "type" — a type is a finite sequence of sorts, and will generally be written using product notation, (e.g. X,Y sorts ⟹ X×Y is a type.) Also, "a free variable of type X×Y" will be understood to mean a sequence $\langle x,y\rangle$ of free variables of sorts X,Y respectively. (Similarly for bound variables.)

Note that every function and predicate symbol is typed—a function has a type giving the sorts of its arguments, and a type which is the sort of its value; a predicate has the type of the sorts of its arguments. Also, with this metamathematical notion we can think of a function (respectively predicate) symbol as having precisely one argument - namely an argument of the appropriate type, (which might be the empty type $1 =_{df} \langle\ \rangle$.)

We now add new function symbols to L to form L':

(i) if X is a sort, Y a type in which X appears, then for each occurrence of X in Y, we have a new symbol π_X^Y (or simply π_X) with argument of type Y, value of sort X. (Unfortunately, the notation doesn't distinguish between various occurrences of X, contrary to our intention, but never mind!)

(ii) if Y is a type, we've the new symbol π_1^Y (or simply $!_Y$) with argument of type Y, value of type 1, the empty sequence.

We use L to define terms in the usual manner, except that we note that any term is typed: we say a term t has domain Y if Y is the type giving the sorts of the free variables occurring in t, and codomain X if X is the sort of (the value of) t, and use the notation $t: Y \to X$. Also, the new symbols of $L'\backslash L$ are terms with evident domains and codomains - we do not include these in the inductive definition of terms, but will include them in that of formulae, to enable us to add dummy free variables of the evident sorts. We will write $\pi_X^{X\times Y}$, say, also as $\pi_X^{X\times Y}(x,y)$ to indicate the free variables we suppose it to have. Of course, its value is of sort X.

So, for example, if x is a free variable of sort X, then x is a term with domain and codomain X. Although intuitively $\pi_X^{X\times Y}(x,y)$ is just x, this term has domain given by the sequence $X\times Y$ (using product notation.) (See the appendix for a more precise formulation of the structure imposed on the set of terms. Although irrelevant to the logic, this structure will be used when we come to do some category theory later in Chapter II.)

We define formulae in almost the usual way, except that we note that each formula will be typed (with the type giving the sorts of its free variables), and we require that ϕ, ϕ' have the same type if $\phi\vee\phi'$, $\phi\wedge\phi'$, $\phi\supset\phi'$ are to be wffs. (The following are synonymous: "ϕ has type X", "ϕ is over X", "the free variables of ϕ have sorts as given by the type X". So "sentence" = "formula over 1".)

1.1. The deduction rules and axioms are based on the standard natural deduction formulation of intuitionistic logic, as given, *e.g.*, in Prawitz [35],[36]. Explicitly, LPC has the following rules:

$$(\wedge I)\ \frac{\phi\quad\phi'}{\phi\wedge\phi'} \qquad (\wedge E)_L\ \frac{\phi\wedge\phi'}{\phi} \qquad (\wedge E)_R\ \frac{\phi\wedge\phi'}{\phi'}$$

$$(\vee I)_L\ \frac{\phi}{\phi\vee\phi'} \qquad (\vee I)_R\ \frac{\phi'}{\phi\vee\phi'} \qquad (\vee E)\ \frac{\phi\vee\phi'\quad \begin{matrix}[\phi]\\ \gamma\end{matrix}\quad \begin{matrix}[\phi']\\ \gamma\end{matrix}}{\gamma}$$

$$(\supset I)\ \frac{\begin{matrix}[\phi]\\ \phi'\end{matrix}}{\phi\supset\phi'} \qquad (\supset E)\ \frac{\phi\supset\phi'\quad\phi}{\phi'}$$

$$(\forall I)\ \frac{\phi(x)}{\forall\xi\phi(\xi)} \qquad (\forall E)\ \frac{\forall\xi\phi(\xi)}{\phi(t)}$$

$$(\exists I)\ \frac{\phi(t)}{\exists\xi\phi(\xi)} \qquad (\exists E)\ \frac{\exists\xi\phi(\xi)\quad \begin{matrix}[\phi(x)]\\ \phi'\end{matrix}}{\phi'}$$

$$(!)\ \frac{\phi}{\top_X} \qquad (\bot)\ \frac{\bot_X}{\phi}$$

(In (!), (respectively ($\bot$)) ϕ is an atomic formula over X, different from $\top_X$, (respectively $\bot_X$), where $\top_X$, (respectively $\bot_X$) is $\top$ (respectively $\bot$) with a dummy free variable x of type X. Also in the above, [ ] indicate a discharged assumption.)

There are the following restrictions on these rules: In the rules ($\forall$E), ($\exists$I) the premises and conclusion must be formulae over the same type (i.e. with the same free variables.) In the rule ($\forall$I), x must not occur in any assumption on which $\phi(x)$ depends, except possibly as a dummy free variable, in which case the dummy occurrences of x may be discharged.

In the rule ($\exists$E), x must not occur in $\exists\xi\phi$, in ϕ', or in any assumption other than ϕ on which the upper occurrence of ϕ' depends, except possibly as a dummy variable in the upper occurrence of ϕ', and the assumptions on which that occurrence depends, in which case the dummy occurrences of x may be discharged.

Remarks: Note that there are some "hidden" restrictions, due to our formation rules for wff's; e.g. we can only apply modus ponens ($\supset$E) if ϕ and ϕ' have the same free variables (for otherwise $\phi\supset\phi'$ isn't a wff.) The point of these "homogeneity" restrictions is to prevent us from "accidently" introducing a type, about which we may know nothing. (For instance, it may not be non-empty.) The similar restrictions on the ($\forall$E), ($\exists$I) rules have been mentioned briefly in the introduction - e.g. in the ($\forall$E) case, without some restriction, the premise can hold vacuously, and yet the conclusion makes a positive assertion which need not hold. That such restrictions are not needed for the ($\forall$I), ($\exists$E) rules is seen by thinking of them as follows:

($\forall$I): If we can deduce $\phi(x)$ without referring to x, then we can easily deduce $\forall\xi\phi$; (i.e. "we could deduce $\phi(\xi)$ for any (other) ξ.")

($\exists$E): If we can deduce ϕ' from $\phi(x)$, using x only in ϕ, then we can deduce ϕ' from $\exists\xi\phi$; (i.e. "we could deduce ϕ' from the knowledge that there was some ξ for which $\phi\xi$.") These do not depend on the (interpretaion of the) type X being non-empty.

Note that for any derivation of our system, assumptions and conclusions are formulae over the same type (i.e. with the same free variables.) We generally follow Prawitz' notation in [36], when discussing derivations, (with obvious changes in typographical conventions.) Also we adopt the convention that T (or its variants T_X) is not considered an assumption: so e.g. $T \vdash \phi$ is equivalent to $\vdash \phi$.

A word about notation: we will frequently write formulae without explicit mention of dummy free variables that must be present, if the formation and deduction rules are to be satisfied (as we have already done above). It must be checked each time we do this that the necessary dummy variables can in fact be added, to make sense of what we've written, in the context of the given rules. We will say a derivation (P): $\Gamma \vdash$ is over X if the formulae in $\Gamma\cup\{\phi\}$ are over X. (So every derivation is over some type X.) Finally, we denote by (id) the "rule" $\frac{\phi}{\phi}$ (rewriting ϕ a second time), which should be understood as being merely the top occurrence of ϕ. (So, e.g. $\dfrac{\dfrac{\phi}{\phi\vee\phi'}\,(\vee I)}{\phi\vee\phi'}\,(id)$ and $\dfrac{\dfrac{\phi}{\phi}\,(id)}{\phi\vee\phi'}\,(\vee I)$ are both just $(\vee I)_L$.)

1.2 We will be interested in a number of operations on derivations in LPC. These fall into three main groups: reductions, expansions, and permutations. For the most part they are taken directly from Prawitz [35] or [36]; as I wish to think of them as 2-cells, they will be written here with arrows to indicate domains and codomains.

Reductions

$$(\wedge\text{ Red})_L\quad \dfrac{\dfrac{\begin{array}{c}P_o\\ \phi\end{array}\quad \begin{array}{c}P_1\\ \psi\end{array}}{\phi\wedge\psi}}{\phi} \Longrightarrow \begin{array}{c}P_o\\ \phi\end{array}$$

$$(\wedge\text{ Red})_R\quad \dfrac{\dfrac{\begin{array}{c}P_o\\ \phi\end{array}\quad \begin{array}{c}P_1\\ \psi\end{array}}{\phi\wedge\psi}}{\psi} \Longrightarrow \begin{array}{c}P_1\\ \psi\end{array}$$

$$(\vee\text{ Red})_L\quad \dfrac{\dfrac{\begin{array}{c}P_o\\ \phi\end{array}}{\phi\vee\psi}\quad \begin{array}{c}[\phi]\\ P_1\\ \theta\end{array}\quad \begin{array}{c}[\psi]\\ P_2\\ \theta\end{array}}{\theta} \Longrightarrow \begin{array}{c}P_o\\ [\phi]\\ P_1\\ \theta\end{array}$$

$$(\vee\text{ Red})_R\quad \dfrac{\dfrac{\begin{array}{c}P_o\\ \psi\end{array}}{\phi\vee\psi}\quad \begin{array}{c}[\phi]\\ P_1\\ \theta\end{array}\quad \begin{array}{c}[\psi]\\ P_2\\ \theta\end{array}}{\theta} \Longrightarrow \begin{array}{c}P_o\\ [\psi]\\ P_2\\ \theta\end{array}$$

$$(\supset\text{ Red})\quad \dfrac{\dfrac{\begin{array}{c}[\phi]\\ P_1\\ \psi\end{array}}{\phi\supset\psi}\quad \begin{array}{c}P_o\\ \phi\end{array}}{\psi} \Longrightarrow \begin{array}{c}P_o\\ [\phi]\\ P_1\\ \psi\end{array}$$

$$(\forall\text{ Red})\quad \dfrac{\dfrac{\begin{array}{c}P(x)\\ \phi(x)\end{array}}{\forall\xi\phi(\xi)}}{\phi(t)} \Longrightarrow \begin{array}{c}P(t)\\ \phi(t)\end{array}$$

$$(\exists\text{ Red})\quad \dfrac{\dfrac{\begin{array}{c}P_o(t)\\ \phi(t)\end{array}}{\exists\xi\phi(\xi)}\quad \begin{array}{c}[\phi(x)]\\ P_1(x)\\ \psi\end{array}}{\psi} \Longrightarrow \begin{array}{c}P_o(t)\\ \phi(t)\\ P_1(t)\\ \psi\end{array}$$

$$(!\text{ Red})\quad \begin{array}{c}\phi\\ P\\ T_X\end{array}. \Longrightarrow \dfrac{\phi}{T_X}\ (!) \quad (\phi \text{ the sole assumption of } P)$$

$$(\bot\text{ Red})\quad \begin{array}{c}\bot_X\\ P\\ \phi\end{array} \Longrightarrow \dfrac{\bot_X}{\phi}\ (\bot) \quad (\bot_X \text{ the sole assumption of } P)$$

(Strictly, (! Red) and ($\bot$ Red) apply only when ϕ is atomic and different from T_X, (respectively $\bot_X$). If ϕ is T_X (respectively $\bot_X$) these reductions will be understood to be the identity derivation. Also, if ϕ is not atomic, the reductions will be to the appropriate derived rule.)

Expansions

$$(\wedge\text{ Exp})\quad \begin{array}{c}P\\ \phi\wedge\psi\end{array} \Longrightarrow \dfrac{\dfrac{\begin{array}{c}P\\ \phi\wedge\psi\end{array}}{\phi}\quad \dfrac{\begin{array}{c}P\\ \phi\wedge\psi\end{array}}{\psi}}{\phi\wedge\psi}$$

$$(\vee\text{ Exp})\quad \begin{array}{c}P_o\\ [\phi\vee\psi]\\ P_1\\ \theta\end{array} \Longrightarrow \dfrac{\begin{array}{c}P_o\\ \phi\vee\psi\end{array}\quad \begin{array}{c}\dfrac{[\phi]}{\phi\vee\psi}\\ P_1\\ \theta\end{array}\quad \begin{array}{c}\dfrac{[\psi]}{\phi\vee\psi}\\ P_1\\ \theta\end{array}}{\theta}$$

(provided the RHS is a derivation)

$$(\supset \text{Exp}) \quad \begin{array}{c} P \\ \phi\supset\psi \end{array} \implies \dfrac{\dfrac{\begin{array}{c} P \\ \phi\supset\psi \end{array} \quad [\phi]}{\psi}}{\phi\supset\psi}$$

$$(\forall\text{Exp}) \quad \begin{array}{c} P \\ \forall\xi\phi(\xi) \end{array} \implies \dfrac{\dfrac{\begin{array}{c} P \\ \forall\xi\phi(\xi) \end{array}}{\phi(x)}}{\forall\xi\phi(\xi)} \quad (x \text{ not free in } P \text{ or in } \forall\xi\phi)$$

$$(\exists\text{ Exp}) \quad \begin{array}{c} P_o \\ [\exists\xi\phi(\xi)] \\ P_1 \\ \theta \end{array} \implies \dfrac{\begin{array}{c} P_o \\ \exists\xi\phi(\xi) \end{array} \quad \begin{array}{c} \dfrac{[\phi(x)]}{[\exists\xi\phi(\xi)]} \\ P_1 \\ \theta \end{array}}{\theta}$$

(provided the RHS is a derivation)

1.3 Permutations

($\vee$ Exp) and ($\exists$ Exp), in the forms above, are more general than the versions in Prawitz [36]; he gives these operations only in the case P_1 is the identity derivation, (so θ is $\phi\vee\psi$, respectively $\exists\xi\phi(\xi)$.) A consequence of these more general forms is the following operations:

$$(\vee\text{ Perm}) \quad \begin{array}{c} \dfrac{\begin{array}{c} P \\ \phi\vee\psi \end{array} \quad \begin{array}{c} [\phi] \\ \Omega \\ \theta \end{array} \quad \begin{array}{c} [\psi] \\ R \\ \theta \end{array}}{[\theta]} \\ S \\ \gamma \end{array} \implies \dfrac{\begin{array}{c} P \\ \phi\vee\psi \end{array} \quad \begin{array}{c} [\phi] \\ \Omega \\ [\theta] \\ S \\ \gamma \end{array} \quad \begin{array}{c} [\psi] \\ R \\ [\theta] \\ S \\ \gamma \end{array}}{\gamma}$$

$$(\exists\text{ Perm}) \quad \begin{array}{c} \dfrac{\begin{array}{c} P \\ \exists\xi\phi(\xi) \end{array} \quad \begin{array}{c} [\phi(x)] \\ \Omega \\ \theta \end{array}}{[\theta]} \\ S \\ \gamma \end{array} \implies \dfrac{\begin{array}{c} P \\ \exists\xi\phi(\xi) \end{array} \quad \begin{array}{c} [\phi(x)] \\ \Omega \\ [\theta] \\ S \\ \gamma \end{array}}{\gamma}$$

(for each: provided the RHS is a derivation)

(Prawitz' $\vee$E reduction and $\exists$E reductions are special cases of these.)

A proof of this claim: (for $\exists$; $\vee$ is similar.)

$$\begin{array}{c} \dfrac{\begin{array}{c} P \\ \exists\xi\phi(\xi) \end{array} \quad \begin{array}{c} [\phi(x)]^{①} \\ \Omega \\ \theta \end{array}}{[\theta]} ① \\ S \\ \gamma \end{array} \implies \dfrac{\begin{array}{c} P \\ \exists\xi\phi(\xi) \end{array} \quad \begin{array}{c} \dfrac{\dfrac{[\phi(x)]^{②}}{\exists\xi\phi} \quad \begin{array}{c} [\phi(x)]^{①} \\ \Omega \\ \theta \end{array}}{[\theta]} ① \\ S \\ \gamma \end{array}}{\gamma} ② \quad \ldots$$

$$\text{by } (\exists\text{ Exp}): P_1 \text{ is } \begin{array}{c} \dfrac{\exists\xi\phi \quad \begin{array}{c} [\phi(x)] \\ \Omega \\ \theta \end{array}}{[\theta]} \\ S \\ \gamma \end{array}$$

$$\ldots \implies \dfrac{\begin{array}{c} P \\ \exists\xi\phi(\xi) \end{array} \quad \begin{array}{c} [\phi(x)]^{②} \\ \Omega \\ [\theta] \\ S \\ \gamma \end{array}}{\gamma} ② \quad \text{by } (\exists\text{ Red}) \text{ applied to}$$

$$\dfrac{\dfrac{\phi(x)}{\exists\xi\phi} \quad \begin{array}{c} [\phi(x)] \\ \Omega \\ \theta \end{array}}{\theta}$$

Remark: This proof tacitly uses the definition of composition of 2-cells for the 2-category LPC, in §2.

If we wished, we could suppose ($\vee$ Exp) and ($\exists$ Exp) in the form given by Prawitz; then we would have to suppose ($\vee$ Perm) and ($\exists$ Perm) as given above, from which we can derive the forms of ($\vee$ Exp), ($\exists$ Exp) given above. (It will be seen why $\vee$E reduction, $\exists$E reduction are not enough - note, for the moment, that these special cases do not give the more general cases. This can be seen by considering the following derivations, both in normal form:

$$\dfrac{\exists\xi\phi \qquad \dfrac{\dfrac{\phi(x)}{\exists\xi\phi}\,(\exists I)}{(\exists\xi\phi)\vee\psi}\,(\vee I)}{(\exists\xi\phi)\vee\psi}\,(\exists E) \quad \text{and} \quad \dfrac{\exists\xi\phi}{(\exists\xi\phi)\vee\psi}\,(\vee I)\ .)$$

Proof of claim: given Prawitz' $\vee$ expansion and ($\vee$ Perm) we can derive ($\vee$ Exp):

$$\begin{array}{c} P_0 \\ \phi\vee\psi \\ P_1 \\ \theta \end{array} \Longrightarrow \begin{array}{c} \dfrac{\begin{array}{c}P_0\\ \phi\vee\psi\end{array} \quad \dfrac{[\phi]}{\phi\vee\psi} \quad \dfrac{[\psi]}{\phi\vee\psi}}{[\phi\vee\psi]} \\ P_1 \\ \theta \end{array} \quad \text{by Prawitz' } \vee \text{ expansion}$$

$$\Longrightarrow \dfrac{\begin{array}{c}P_0\\ \phi\vee\psi\end{array} \quad \begin{array}{c}\dfrac{[\phi]}{[\phi\vee\psi]}\\ P_1\\ \theta\end{array} \quad \begin{array}{c}\dfrac{[\psi]}{[\phi\vee\psi]}\\ P_1\\ \theta\end{array}}{\theta} \quad \text{by } (\vee \text{ Perm}).$$

<u>Remark</u>: Again we have tacitly used the definition of composition of 2-cells in LPC.

<u>Remark</u>: There is no need for permutation operations for the other logical symbols, because of the nature of the rules of natural deduction. (And so one might be tempted to treat ($\vee$ Perm) and ($\exists$ Perm) as equivalences - an idea we shall return to again later.) For instance, the analagous ($\wedge$ Perm) would be

$$\dfrac{\begin{array}{c}\gamma\\ S\\ [\theta]\\ \Omega\\ \phi\end{array} \quad \begin{array}{c}\gamma\\ S\\ [\theta]\\ R\\ \psi\end{array}}{\phi\wedge\psi} \Longrightarrow \dfrac{\begin{array}{c}\gamma\\ S\\ [\theta]\\ \Omega\\ \phi\end{array} \quad \begin{array}{c}\gamma\\ S\\ [\theta]\\ R\\ \psi\end{array}}{\phi\wedge\psi}$$

This, of course, is the identity operation.

It is more or less a matter of personal taste which operations one chooses as basic and which are derived. For the rest of this chapter we shall be treating all the above operations as basic, (with the evident relationships as given in the proofs of the claims.)

§2 LPC as a 2-category.

It will be evident by now that we may think of LPC as a 2-category: the objects are formulae, the morphisms are derivations (with at most one assumption; a derivation with no assumptions is regarded as a morphism from T), and the 2-cells are operations on derivations, as given in §1. This construction may be carried out for any theory T in LPC, of course. (In this context, a theory is viewed as having not only (non-logical) rules of inference, but also (non-logical) operations on derivations.)

(In fact, LPC has the structure of a fibred 2-category, similar to the construction of Chapter II. The base category is comprised of types and terms, as it is there, and the fibres are the 2-categories as given above, restricting the formulae to those over the given type. However, this structure will not be used in this chapter.)

Composition of morphisms is defined in the obvious way; (it may be described as "concatenation of trees":)

given $\begin{array}{c}\phi\\ P\\ \psi\end{array}$ and $\begin{array}{c}\psi\\ \Omega\\ \theta\end{array}$, ΩP is the derivation $\begin{array}{c}\phi\\ P\\ [\psi]\\ \Omega\\ \theta\end{array}$ - that is,

"write out Ω, and replace each top occurrence of ψ by P."

Identity morphisms have already been defined (in §1.1): the identity for ϕ is $\frac{\phi}{\phi}$ (or just ϕ). (This explains the convention that $\frac{\phi}{\phi}$ is to mean only the top occurrence of ϕ, so that this really is an identity for the above composition.)

Composition of 2-cells is defined in the evident way, as composition of operations:

given P, Ω, R : $\phi \vdash \psi$, S, T : $\psi \vdash \theta$, and 2-cells (operations) a :P $\Rightarrow\Omega$, b :$\Omega \Rightarrow$R, c :S $\Rightarrow$T, then b.a : P $\Rightarrow$ R is the operation b.a ("first perform a, then b"), and ca:SP $\Rightarrow$ TΩ is the operation defined piece-wise:

$$\begin{array}{c}\phi\\ P\\ [\psi]\\ S\\ \theta\end{array} \overset{}{\underset{cP}{\Rightarrow}} \begin{array}{c}\phi\\ P\\ [\psi]\\ T\\ \theta\end{array} \underset{Ta}{\Rightarrow} \begin{array}{c}\phi\\ \Omega\\ [\psi]\\ T\\ \theta\end{array} \quad \text{(or equivalently } \begin{array}{c}\phi\\ P\\ [\psi]\\ S\\ \theta\end{array} \underset{Sa}{\Rightarrow} \begin{array}{c}\phi\\ \Omega\\ [\psi]\\ S\\ \theta\end{array} \underset{c\Omega}{\Rightarrow} \begin{array}{c}\phi\\ \Omega\\ [\psi]\\ T\\ \theta\end{array}$$

(This definition makes sense because all the operations we have considered (and will allow to be considered) have operated only on a part of a derivation: they have allowed us to leave unchanged the top of a derivation, and (tacitly) the bottom of a derivation. This is illustrated in the proofs in §1.3.)

Identity 2-cells are just the identity operations.

It is easy to verify that these definitions do indeed define a 2-category: the proof of the interchange law for horizontal and vertical composition of 2-cells is implicit in the definition given above.

Convention: As far as possible in this chapter we shall follow the convention that identity morphisms will be written $1_{(.)}$ and identity 2-cells as $id_{(.)}$. Also, generally objects of $\underline{A}$ and $\underline{B}$ will be written A,B,C,..., morphisms f,g,h..., and 2-cells a,b,c,..., the 2-category they belong to being understood from the context.

§3 Structure of connectives and quantifiers: Concrete adjointness.

3.0 We now come to the main point of this chapter: what is the structure of the connectives and quantifiers? The form of the deduction rules suggests they satisfy adjointness properties more or less analagous to the categorical notions of product, coproduct, exponentiation, image, and dual (or universal) image, and in fact this analogy is given more credence by the operations we have mentioned in §1, which suggest such structure as units and co-units for some adjunctions. However, it is reasonably clear that strict adjointness cannot be expected, so we must see just how the notion has been weakened.

There are two main ways adjointness may be expressed: (i) in terms of hom-sets ("concrete", in Butler's terminology [06],[07]), and (ii) in terms of units and co-units ("formal", in the same terminology); these give rise to two methods of weakening the adjointness, by replacing various equalities of morphisms with "comparison 2-cells". We shall look mainly at the first method, with a few words about the second later.

3.1 The general context will be the following:

We are given 2-categories $\underline{A}$ and $\underline{B}$ and a pair of lax 2-functors $F: \underline{A} \to \underline{B}$ and $G: \underline{B} \to \underline{A}$. (We intend a lax $F \dashv G$.) We shall see later in what ways the word "lax" is to be understood, but in the case of these 2-functors, it will always be the case that it is to mean the following:

F and G are functions from objects to objects, morphisms to morphisms, and 2-cells to 2-cells. They are strictly

functorial at the level of 2-cells:

$F(ba) = F(b)F(a)$, $G(b'a') = G(b')G(a')$, (similarly for vertical composition)

$F(id_f) = id_{Ff}$, $G(id_g) = id_{Gg}$.

For any morphisms $A \xrightarrow[f]{} B \xrightarrow[g]{} C$ of $\underline{A}$, $A' \xrightarrow[f']{} B' \xrightarrow[g']{} C'$ of $\underline{B}$, there are 2-cells

$$\gamma^F_{g,f}: F(g)F(f) \Longrightarrow F(gf),\quad \gamma^G_{g',f'}: G(g')G(f') \Longrightarrow G(g'f')$$

$$\iota^F_A: 1_{FA} \Longrightarrow F(1_A),\quad \iota^G_{A'}: 1_{GA'} \Longrightarrow G(1_{A'}).$$

We shall discuss coherence conditions for γ and ι later.

3.2 There are lax 2-natural transformations

$$\kappa : (F-, -) \longrightarrow (-, G-) \text{ and } \lambda : (-, G-) \longrightarrow (F-, -).$$

That is, for objects A of $\underline{A}$, B of $\underline{B}$, there are functors (i.e. morphisms in Cat):

$$\kappa_{A,B}: \mathrm{Hom}_{\underline{B}}(FA,B) \to \mathrm{Hom}_{\underline{A}}(A,GB) \text{ and } \lambda_{A,B}: \mathrm{Hom}_{\underline{A}}(A,GB) \to \mathrm{Hom}_{\underline{B}}(FA,B),$$

together with other conditions to be discussed later. (We intend that κ and λ be adjoint: for $\vee$ and $\exists$, $\kappa \dashv \lambda$; for $\wedge$ and $\forall$, $\lambda \dashv \kappa$. Unfortunately, $\supset$ will fit poorly into this scheme.)

We shall look at $\vee$ and $\wedge$ fairly closely; $\exists$ and $\forall$ can be treated similarly as will be evident from considering the deduction rules and operations. We shall then see that there are two different notions of weak adjoint suitable for $\vee, \exists$ on the one hand and for $\wedge, \forall$ on the other, as expressed by the two relations $\kappa \dashv \lambda$ and $\lambda \dashv \kappa$. It will not be surprising, then, to find that $\supset$ behaves somewhat as a mutant.

3.3 For the cases $\vee$ and $\wedge$, we have the following dictionary:

	$\underline{\vee}$	$\underline{\wedge}$
$\underline{A}$:	LPC × LPC	LPC
$\underline{B}$:	LPC	LPC × LPC
F :	$\vee$	diagonal
G :	diagonal	$\wedge$

That is, for $\vee$, given formulae ϕ, ψ, $F^{\vee}(\phi,\psi) = \phi\vee\psi$ and $G^{\vee}(\phi) = (\phi,\phi)$. For $\wedge$, $F^{\wedge}(\phi) = (\phi,\phi)$ and $G^{\wedge}(\phi,\psi) = \phi\wedge\psi$.

Given derivations $\begin{matrix}\phi\\ P\\ \phi'\end{matrix}$, $\begin{matrix}\psi\\ \Omega\\ \psi'\end{matrix}$, $F^{\vee}(P,\Omega) = \dfrac{\phi\vee\psi \quad \dfrac{\begin{matrix}[\phi]\\ P\\ \phi'\end{matrix}}{\phi'\vee\psi'} \quad \dfrac{\begin{matrix}[\psi]\\ \Omega\\ \psi'\end{matrix}}{\phi'\vee\psi'}}{\phi'\vee\psi'}$, and

$$G^{\wedge}(P,\Omega) = \dfrac{\begin{matrix}\dfrac{\phi\wedge\psi}{\phi}\\ P\\ \phi'\end{matrix} \quad \begin{matrix}\dfrac{\phi\wedge\psi}{\psi}\\ \Omega\\ \psi'\end{matrix}}{\phi'\wedge\psi'} .$$

The remaining definitions (of $F^{\vee}$ and $G^{\wedge}$ on 2-cells, and of the diagonal 2-functors $G^{\vee}$ and $F^{\wedge}$) are obvious.

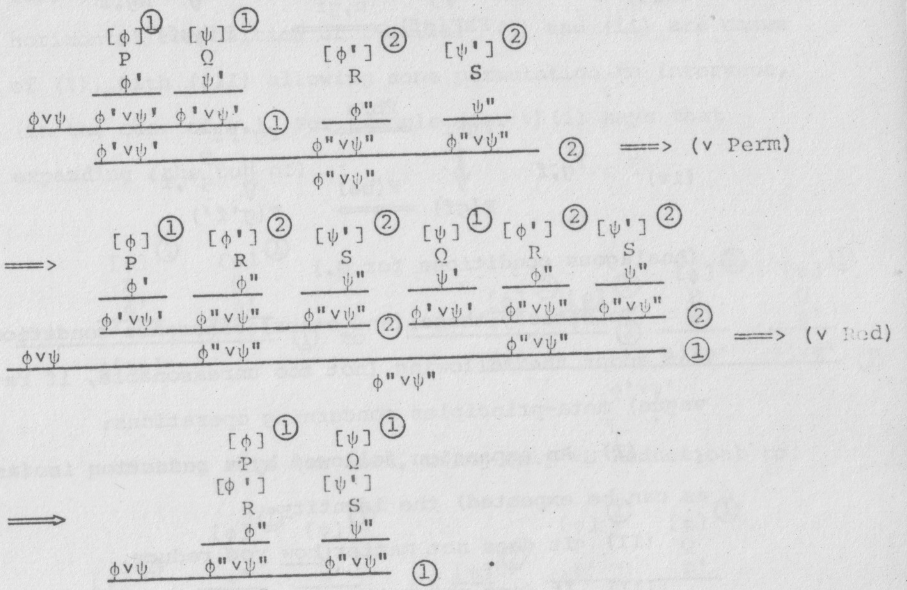
It is not too difficult to see that the dictionary can be extended:

	$\underline{\vee}$	$\underline{\wedge}$
ι^F:	($\vee$ Exp)	identity
ι^G:	identity	($\wedge$ Exp)
γ^F:	($\vee$ Perm)+($\vee$ Red)	identity
γ^G:	identity	($\wedge$ Red)

For example, consider ι^G for $\wedge$: let ϕ, ψ be formulae; then $\iota^G_{<\phi,\psi>}$ is the operation:

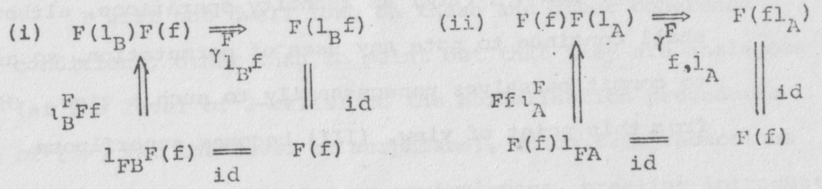
$$\frac{\phi\wedge\psi}{\phi\wedge\psi} \ \text{(id)} \Longrightarrow \dfrac{\dfrac{\phi\wedge\psi}{\phi} \quad \dfrac{\phi\wedge\psi}{\psi}}{\phi\wedge\psi} \quad (\wedge\ \text{Exp})$$

Similarly, consider γ^F for $\vee$: let $\begin{matrix}\phi\\ P\\ \phi'\end{matrix}$, $\begin{matrix}\phi'\\ R\\ \phi''\end{matrix}$, $\begin{matrix}\psi\\ \Omega\\ \psi'\end{matrix}$, $\begin{matrix}\psi'\\ S\\ \psi''\end{matrix}$ be derivations; then $\gamma^F_{(R,S),(P,\Omega)}$ is the composite operation:



3.4 Given such data, it is natural to ask whether the following coherence conditions hold:

given morphisms $A \underset{f'}{\overset{f}{\Longrightarrow}} B \underset{g'}{\overset{g}{\Longrightarrow}} C \xrightarrow{h} D$ and 2-cells $f \underset{a}{\Longrightarrow} f'$, $g \underset{b}{\Longrightarrow} g'$ in $\underline{A}$, the following diagrams commute:



$$\text{(iii)}\qquad \begin{array}{ccc} FhFgFf & \xrightarrow{\ \gamma^F_{h,g}Ff\ } & F(hg)Ff \\ {\scriptstyle Fh\gamma^F_{g,f}}\Big\Downarrow & & \Big\Downarrow{\scriptstyle \gamma^F_{hg,f}} \\ FhF(gf) & \xrightarrow{\ \gamma^F_{h,gf}\ } & F(hgf) \end{array}$$

$$\text{(iv)}\qquad \begin{array}{ccc} FgFf & \xrightarrow{\ FbFa\ } & Fg'Ff' \\ {\scriptstyle \gamma^F_{g,f}}\Big\Downarrow & & \Big\Downarrow{\scriptstyle \gamma^F_{g',f'}} \\ F(gf) & \xrightarrow{\ F(ba)\ } & F(g'f') \end{array}$$

(Analagous conditions for G.)

To satisfy these (and other) coherence conditions, we can adopt the following (not too unreasonable, if rather vague) meta-principles concerning operations:

(I) An expansion followed by a reduction is (as close as can be expected) the identity.

(II) It does not matter how you reduce.

(III) It does not matter how, or (to some extent) even whether you permute.

(Equivalently, we could examine each coherence condition as it appears, and postulate it as a condition on our operations. This would amount to a case-by-case description of the meta-principles above.)

In keeping with principle (I), we are tempted to regard ($\vee$ Perm) and ($\exists$ Perm) as identity operations, although we shall continue to note any uses of permutation, so as not to commit ourselves unnecessarily to such a view. Of course, from this point of view, (III) becomes superfluous.

So let us see how these principles give us the coherence conditions: (iv) is satisfied in any case (without need of (I), (II), or (III)), in view of the piece-wise definition of horizontal composition of 2-cells. (i) and (ii) are cases of (I), with (III) allowing some permutation to intervene, (in the case of $\vee$.) For example (for $\vee$) (i) says that expanding (the top of)

$$\dfrac{\phi\vee\psi \qquad \dfrac{\begin{array}{c}[\phi]^{(1)}\\ P\\ \phi'\end{array}}{\phi'\vee\psi'} \qquad \dfrac{\begin{array}{c}[\psi]^{(1)}\\ \Omega\\ \psi'\end{array}}{\phi'\vee\psi'}}{\phi'\vee\psi'}\,(1) \quad\text{to}\quad \dfrac{\dfrac{\phi\vee\psi \qquad \dfrac{[\phi]^{(2)}}{\phi\vee\psi} \qquad \dfrac{[\psi]^{(2)}}{\phi\vee\psi}}{\phi\vee\psi}\,(2) \qquad \dfrac{\begin{array}{c}[\phi]^{(1)}\\ P\\ \phi'\end{array}}{\phi'\vee\psi'} \qquad \dfrac{\begin{array}{c}[\psi]^{(1)}\\ \Omega\\ \psi'\end{array}}{\phi'\vee\psi'}}{\phi'\vee\psi'}\,(1)$$

then permuting this (in fact, using only $\vee$E Reduction) to

$$\dfrac{\phi\vee\psi \qquad \dfrac{\dfrac{[\phi]^{(2)}}{\phi\vee\psi} \qquad \dfrac{\begin{array}{c}[\phi]^{(1)}\\ P\\ \phi'\end{array}}{\phi'\vee\psi'} \qquad \dfrac{\begin{array}{c}[\psi]^{(1)}\\ \Omega\\ \psi'\end{array}}{\phi'\vee\psi'}}{\phi'\vee\psi'}\,(1) \qquad \dfrac{\dfrac{[\psi]^{(2)}}{\phi\vee\psi} \qquad \dfrac{\begin{array}{c}[\phi]^{(1)}\\ P\\ \phi'\end{array}}{\phi'\vee\psi'} \qquad \dfrac{\begin{array}{c}[\psi]^{(1)}\\ \Omega\\ \psi'\end{array}}{\phi'\vee\psi'}}{\phi'\vee\psi'}\,(1)}{\phi'\vee\psi'}\,(2)$$

and finally reducing the evident subderivations to arrive again at the original derivation, is to be the identity operation. (iii) is similarly a case of (II), with some (III), in the $\vee$ case.

I will not dwell long on these and other coherence conditions, other than to point out that they are analagous (at the level of 2-cells) to the normalisation procedures of [36] (at the level of morphisms), which take reductions and certain permutations as equivalences, treating introductio and elimination rules as "reductions" and "expansions" of formulae. They are more complicated when set out in full of course, partly because of the permutations for $\vee$ and $\exists$, but also because derivations are more complicated structures than formulae. One is at liberty to impose coherence conditions or not, as one pleases; the corresponding 2-categorical structure will share such properties or the lack thereof.

3.5 We come now to the lax 2-natural transformations

$$(F-,-) \underset{\lambda}{\overset{\kappa}{\rightleftarrows}} (-,G-),$$

and extend our dictionary:

	$\underline{\vee}$	$\underline{\wedge}$
κ:	($\vee$I)	($\wedge$I)
λ:	($\vee$E)	($\wedge$E)

These are defined as follows.

For $\vee$: let $\begin{array}{c}\phi\vee\psi\\ P\\ \theta\end{array}$, $\begin{array}{c}\phi\\ \Omega\\ \theta\end{array}$, $\begin{array}{c}\psi\\ R\\ \theta\end{array}$ be derivations. Then ($\kappa = \kappa^{\vee}$)

$$\kappa_{(\phi,\psi),\theta}(P) = \left\langle \begin{array}{c}\dfrac{\phi}{\phi\vee\psi}\\ P\\ \theta\end{array} , \begin{array}{c}\dfrac{\psi}{\phi\vee\psi}\\ P\\ \theta\end{array} \right\rangle \quad \text{and} \quad (\lambda = \lambda^{\vee})$$

$$\lambda_{(\phi,\psi),\theta}(\Omega,R) = \begin{array}{c} \dfrac{\phi\vee\psi \quad \begin{array}{c}\phi\\ \Omega\\ \theta\end{array} \quad \begin{array}{c}\psi\\ R\\ \theta\end{array}}{\theta} \end{array} .$$

For $\wedge$: let $\begin{array}{c}\theta\\ P\\ \phi\wedge\psi\end{array}$, $\begin{array}{c}\theta\\ \Omega\\ \phi\end{array}$, $\begin{array}{c}\theta\\ R\\ \psi\end{array}$ be derivations. Then ($\kappa = \kappa^{\wedge}$)

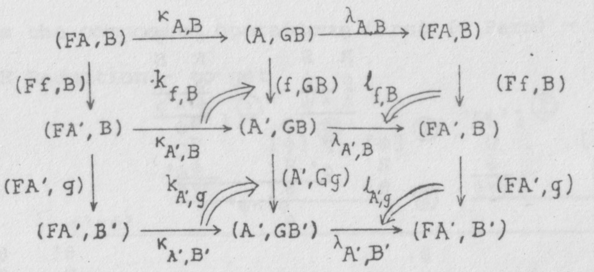
$$\kappa_{\theta,(\phi,\psi)}(\Omega,R) = \begin{array}{c}\theta \;\; \theta\\ \Omega \;\; R\\ \dfrac{\phi \;\; \psi}{\phi\wedge\psi}\end{array} \quad \text{and} \quad (\lambda = \lambda^{\wedge})$$

$$\lambda_{\theta,(\phi,\psi)}(P) = \left\langle \begin{array}{c}\theta\\ P\\ \dfrac{\phi\wedge\psi}{\phi}\end{array} , \begin{array}{c}\theta\\ P\\ \dfrac{\phi\wedge\psi}{\psi}\end{array} \right\rangle .$$

(The actions of $\kappa^{\vee}$, $\lambda^{\vee}$, $\kappa^{\wedge}$, $\lambda^{\wedge}$ on 2-cells (operations) are obvious.)

3.6 κ and λ are lax in the sense that strict naturality is replaced by the existence of 2-cells as indicated in the following diagrams:

(For $\vee$ and $\exists$): Given morphisms $A' \xrightarrow{f} A$ in $\underline{A}$, $B \xrightarrow{g} B'$ in $\underline{B}$, there are natural transformations (2-cells in $\underline{Cat}$)



(i)

(Definitions: $k_{f,g} = k_{A',g}k_{f,B}$; $\ell_{f,g} = \ell_{A',g}\ell_{f,B}$.)

We shall say $\kappa^{\vee}$ is lower and $\lambda^{\vee}$ is upper, following Butler [07].

For $\wedge$ (and $\forall$), the directions of the natural transformations k, ℓ are reversed, so the $\kappa^{\wedge}$ is upper and $\lambda^{\wedge}$ is lower.

This gives another section of the dictionary:

	$\underline{\vee}$	$\underline{\wedge}$
$k_{A',g}$:	identity	($\wedge$ Red)
$\ell_{f,B}$:	($\vee$ Perm)+($\vee$ Red)	identity
$\ell_{A',g}$:	($\vee$ Perm)	($\wedge$ Red)
$k_{f,B}$:	($\vee$ Red)	identity

For example, consider $k_{A',g}$, for $\wedge$. Suppose $\begin{matrix}\theta'\\R\\\phi\end{matrix}$, $\begin{matrix}\theta'\\S\\\psi\end{matrix}$ are derivations, (<u>i.e</u>. an object h of (FA',B),) and that $\begin{matrix}\phi\\P\\\phi'\end{matrix}$, $\begin{matrix}\psi\\Q\\\psi'\end{matrix}$ are derivations, (<u>i.e</u>. a morphism g of <u>B</u>.) Then

(A',Gg) $\kappa_{A',B}$(h) is the derivation

```
θ'  θ'        θ'  θ'
R   S         R   S
φ   ψ         φ   ψ
-------       -------
 φ ∧ ψ         φ ∧ ψ
-------       -------
  φ              ψ
  P              Q
  φ'             ψ'
  ----------------
      φ'∧ψ'
```

and $\kappa_{A',B'}$(FA',g)(h) is the derivation

```
θ'     θ'
R      S
[φ]   [ψ]
P      Q
φ'     ψ'
----------
  φ'∧ψ'
```
;

$k_{A;g}$ is the operation ($\wedge$ Red) applied to the subderivations above ϕ' and above ψ' in the first derivation, yielding the second.

Analagously, consider $\ell_{f,B}$ for $\vee$. Suppose $\begin{matrix}\phi\\R\\\theta\end{matrix}$, $\begin{matrix}\psi\\S\\\theta\end{matrix}$ are derivations, (<u>i.e</u>. an object h of (A,GB),) and that $\begin{matrix}\phi'\\P\\\phi\end{matrix}$, $\begin{matrix}\psi'\\\Omega\\\psi\end{matrix}$ are derivations, (<u>i.e</u>. a morphism f of <u>A</u>.) Then

(Ff,B)$\lambda_{A,B}$(h) is the derivation

```
             (2)         (2)
        [φ']        [ψ']
         P           Ω
         φ           ψ            (1)       (1)
        ----        ----      [φ]       [ψ]
φ'vψ'   φvψ         φvψ        R         S
----------------------- (2)
          φvψ                  θ         θ
          ------------------------------------ (1)
                         θ
```

$\lambda_{A;B}$(f,GB)(h) is the derivation

```
          (1)       (1)
      [φ']      [ψ']
       P         Ω
      [φ]       [ψ]
       R         S
φ'vψ'  θ         θ
------------------------ (1)
       θ
```

$\ell_{f,B}$ is the composite operation: first ($\vee$ Perm) - in fact only $\vee$E Reduction - to get

```
        (1)                              (1)
    [φ']       (2)       (2)        [ψ']       (2)       (2)
     P      [φ]       [ψ]            Ω      [φ]       [ψ]
     φ       R         S             ψ       R         S
    ----                            ----
    φvψ      θ         θ            φvψ      θ         θ
    ------------------------ (2)    ------------------------ (2)
φ'vψ'        θ                               θ
--------------------------------------------------------- (1)
                         θ
```

and then ($\vee$ Red) to get $\lambda_{A;B}$(f,GB)(h).

Finally, consider $\ell_{A;g}$ for $\vee$. Suppose $\begin{matrix}\phi'\\R\\\theta\end{matrix}$, $\begin{matrix}\psi'\\S\\\theta\end{matrix}$ are derivations, (<u>i.e</u>. an object h of (A',GB),) and $\begin{matrix}\theta\\P\\\theta'\end{matrix}$ a derivation, (<u>i.e</u>. a morphism f of <u>A</u>.) Then (FA',g)$\lambda_{A;B}$(h) is the derivation

```
       [φ'] [ψ']
        R    S
φ'vψ'   θ    θ
------------------ ,
       [θ]
        P
        θ'
```

$\lambda_{A;B'}(A',Gg)(h)$ the derivation

$$\begin{array}{ccc} & [\phi'] & [\psi'] \\ & R & S \\ & [\theta] & [\theta] \\ & P & P \\ \phi'\vee\psi' & \theta' & \theta' \\ \hline & \theta' & \end{array}\ ,$$

and $\ell_{A;g}$ is precisely ($\vee$ Perm).

The other natural transformations k,ℓ are treated similarly.

3.7 We could impose coherence conditions on k,ℓ in precisely the same way we did on ι and γ; for example, for $k_{f,B}$ (for $\vee$), we would require that the following diagrams commute:

(i)
$$\begin{array}{ccc} \kappa_{A,B}(1_{FA},B) & \xRightarrow{\ \kappa_{A,B}(\iota_A,B)\ } & \kappa_{A,B}(F1_A,B) \\ \mathrm{id}\ \Big\| & & \Big\Downarrow\ k_{1_A,B} \\ \kappa_{A,B} & \overset{}{\underset{\mathrm{id}}{=\!=\!=\!=\!=}} & (1_A,GB)\kappa_{A,B} \end{array}$$

(ii)
$$\begin{array}{ccc} \kappa_{A'',B}(Ff',B)(Ff,B) & \xRightarrow{\ \kappa_{A''B}(\gamma_{f',f},B)\ } & \kappa_{A'',B}(F(f'f),B) \\ k_{f;B}(Ff,B)\ \Big\Downarrow & & \Big\Downarrow\ k_{f'f,B} \\ (f',GB)\kappa_{A',B}(Ff,B) & \xRightarrow[\ (f',GB)k_{f,B}\ ]{} & (f',f,GB)\kappa_{A,B} \end{array}$$

(for morphisms $A'' \xrightarrow{f'} A' \xrightarrow{f} A$ in $\underline{A}$.)

(iii)
$$\begin{array}{ccc} \kappa_{A',B}(Ff,B) & \xRightarrow{\ k_{f,B}\ } & (f,GB)\kappa_{A,B} \\ \kappa_{A;B}(Fa,B)\ \Big\Downarrow & & \Big\Downarrow\ (a,GB)\kappa_{A,B} \\ \kappa_{A',B}(Ff',B) & \xRightarrow{\ k_{f',B}\ } & (f',GB)\kappa_{A,B} \end{array}$$

(for a 2-cell $A' \underset{f'}{\overset{f}{\Downarrow a}} A$ in $\underline{A}$.)

Such conditions are consequences of our meta-principles of §3.4, just as similar conditions were for ι and γ. So, for example, for the conditions above for $k_{f,B}$ (for $\vee$), (iii) is true in any event, (ii) follows from (II) (with a little (III)), and (i) follows from (I), as is easily seen by writing out the corresponding derivations and operations.

3.8 The adjointness of κ and λ is expressed in terms of a unit and a counit:

(For $\vee$) There are modifications

$$1_{(F-,-)} \xRightarrow{\ \alpha\ } \lambda\kappa \text{ and } \kappa\lambda \xRightarrow{\ \beta\ } 1_{(-,G-)}$$

(unit and counit respectively); that is, for objects A of $\underline{A}$, B of $\underline{B}$ there are 2-cells

$$\alpha_{A,B}: 1_{(FA,B)} \Longrightarrow \lambda_{A,B}\kappa_{A,B} \text{ and } \beta_{A,B}: \kappa_{A,B}\lambda_{A,B} \Longrightarrow 1_{(A,GB)}\ .$$

For $\wedge$, the directions of α and β are reversed, so that α is the counit, β the unit.

These are given by the new entries in the dictionary:

	$\underline{\vee}$	$\underline{\wedge}$
α:	($\vee$ Exp)	($\wedge$ Red)
β:	($\vee$ Red)	($\wedge$ Exp)

In fact, for $\vee$, given a derivation $\begin{array}{c}\phi\vee\psi\\ P\\ \theta\end{array}$, $\alpha_{(\phi,\psi),\theta}(P)$ is the operation

$$\begin{matrix}\phi\vee\psi\\ P\\ \theta\end{matrix} \quad \xRightarrow[(\vee\ \mathrm{Exp})]{} \quad \dfrac{\phi\vee\psi \quad \begin{matrix}\underline{[\phi]}\\ \phi\vee\psi\\ P\\ \theta\end{matrix} \quad \begin{matrix}\underline{[\psi]}\\ \phi\vee\psi\\ P\\ \theta\end{matrix}}{\theta}\ , \quad \text{and}$$

given derivations $\begin{matrix}\phi\\ P\\ \theta\end{matrix}$, $\begin{matrix}\psi\\ \Omega\\ \theta\end{matrix}$, $\beta_{(\phi,\psi),\theta}$ is the operation

$$\dfrac{\dfrac{\phi}{\phi\vee\psi} \quad \begin{matrix}[\phi]\\ P\\ \theta\end{matrix} \quad \begin{matrix}[\psi]\\ \Omega\\ \theta\end{matrix}}{\theta} \quad \xRightarrow[(\vee\ \mathrm{Red})]{} \quad \begin{matrix}\phi\\ P\\ \theta\end{matrix}\ , \qquad \dfrac{\dfrac{\psi}{\phi\vee\psi} \quad \begin{matrix}[\phi]\\ P\\ \theta\end{matrix} \quad \begin{matrix}[\psi]\\ \Omega\\ \theta\end{matrix}}{\theta} \quad \xRightarrow[(\vee\ \mathrm{Red})]{} \quad \begin{matrix}\psi\\ \Omega\\ \theta\end{matrix}\ .$$

(The case $\wedge$ is similar.)

3.9 We cannot expect α and β to be modifications in the usual sense, because κ and λ are lax in different senses, one being upper, the other lower. However, with the meta-principles of §3.4, they are as close to modifications as can be expected - i.e. they satisfy the following coherence conditions:

Given morphisms $A' \xrightarrow[f]{} A$ in $\underline{A}$, $B \xrightarrow[g]{} B'$ in $\underline{B}$, the following commute:

(For $\vee$)

$$\begin{array}{ccc}
(Ff,g) & \xRightarrow{(Ff,g)\alpha_{A,B}} & (Ff,g)\lambda_{A,B}\kappa_{A,B} \\
\alpha_{A;B'}(Ff,g) \Big\Downarrow & & \Big\Downarrow \ell_{f,g}\kappa_{A,B} \\
\lambda_{A;B'}\kappa_{A;B'}(Ff,g) & \xRightarrow{\lambda_{A;B'}k_{f,g}} & \lambda_{A;B'}(f,Gg)\kappa_{A,B}
\end{array}$$

$$\begin{array}{ccc}
\kappa_{A;B'}(Ff,g)\lambda_{A,B} & \xRightarrow{k_{f,g}\lambda_{A,B}} & (f,Gg)\kappa_{A,B}\lambda_{A,B} \\
\kappa_{A;B'}\ell_{f,g} \Big\Downarrow & & \Big\Downarrow (f,Gg)\beta_{A,B} \\
\kappa_{A;B'}\lambda_{A;B'}(F,Gg) & \xRightarrow[\beta_{A',B'}(f,Gg)]{} & (f,Gg)
\end{array}$$

(For $\wedge$, reverse all arrows.)

(All three principles are used - for $\vee$ at least - as may be seen by writing out the corresponding derivations and comparing the operations used.)

3.10 All that remain to make $\kappa \dashv \lambda$ (or $\lambda \dashv \kappa$, as appropriate) are the triangle equalities; viz. the following should commute (for any A, B):

$$(\beta\kappa)(\kappa\alpha) = \kappa: \quad \begin{array}{ccc}
\kappa_{A,B} & \xRightarrow{\kappa_{AB}\alpha_{AB}} & \kappa_{A,B}\lambda_{A,B}\kappa_{A,B} \\
 & {\scriptstyle \mathrm{id}} \searrow\!\!\searrow & \Big\Downarrow \beta_{AB}\kappa_{AB} \\
 & & \kappa_{A,B}
\end{array} \quad \text{and}$$

$$(\lambda\beta)(\alpha\lambda) = \lambda: \quad \begin{array}{ccc}
\lambda_{A,B} & \xRightarrow{\alpha_{AB}\lambda_{AB}} & \lambda_{A,B}\kappa_{A,B}\lambda_{A,B} \\
 & {\scriptstyle \mathrm{id}} \searrow\!\!\searrow & \Big\Downarrow \lambda_{AB}\beta_{AB} \\
 & & \lambda_{A,B}
\end{array}$$

Claim: These two conditions together express the strict form of principle (I):

(Is) An expansion of an occurrence of a logical symbol, followed by a reduction of the same occurrence, is (provided the composite is an endo-operation) the identity.

For let us look at this condition carefully to see just what it means:

(For $\vee$) To be able to expand an occurrence of $\vee$ in a derivation means that the derivation has the form

$\begin{array}{c} P_o \\ [\phi\vee\psi] \\ P_1 \\ \theta \end{array}$, where the occurrence of $\vee$ shown is the one to be expanded. After the expansion, we have the derivation

$$\dfrac{\begin{array}{c} P_o \\ \phi\vee\psi \end{array} \quad \begin{array}{c} [\phi] \\ \hline [\phi\vee\psi] \\ P_1 \\ \theta \end{array} \quad \begin{array}{c} [\psi] \\ \hline [\phi\vee\psi] \\ P_1 \\ \theta \end{array}}{\theta}$$

Now, this is supposed to admit a reduction of $\vee$ at one of the occurrences of $\vee$ that correspond to the original one - viz. to either that occurrence at the bottom of P_o (case i) or that at the top of P_1 (case ii). In case i, P_o must then be of the form $\dfrac{\begin{array}{c}\Omega_o\\ \phi\end{array}}{\phi\vee\psi}$ (or of the form $\dfrac{\begin{array}{c}\Omega_o\\ \psi\end{array}}{\phi\vee\psi}$.)

Thus (Is) says the composite operation

$$\begin{array}{c} \Omega_o \\ \phi \\ \hline [\phi\vee\psi] \\ P_1 \\ \theta \end{array} \xrightarrow{(\vee\ \mathrm{Exp})} \dfrac{\begin{array}{c} \Omega_o \\ \phi \\ \hline \phi\vee\psi \end{array} \quad \begin{array}{c} [\phi] \\ \hline [\phi\vee\psi] \\ P_1 \\ \theta \end{array} \quad \begin{array}{c} [\psi] \\ \hline [\phi\vee\psi] \\ P_1 \\ \theta \end{array}}{\theta} \xrightarrow{(\vee\ \mathrm{Red})} \begin{array}{c} \Omega_o \\ \phi \\ \hline [\phi\vee\psi] \\ P_1 \\ \theta \end{array}$$

(or similarly, mutatis mutandis, for ψ) is the identity. This is precisely the equation $(\beta\kappa)(\kappa\alpha) = \kappa$, (although there Ω_o need not be explicitly mentioned.)

In case ii, P_1 must have the form $\dfrac{\phi\vee\psi \quad \begin{array}{c}[\phi]\\ \Omega_1\\ \theta\end{array} \quad \begin{array}{c}[\psi]\\ \Omega_2\\ \theta\end{array}}{\theta}$,

if we are to be able to apply ($\vee$ Red) to the occurrences of $\vee$ in the top formula of P_1. (This is not quite true - P_1 could also have the form

$$\begin{array}{c} \dfrac{\phi\vee\psi \quad \begin{array}{c}[\phi]\\ \Omega_1'\\ \theta'\end{array} \quad \begin{array}{c}[\psi]\\ \Omega_2'\\ \theta'\end{array}}{\theta'} \\ \Omega_3 \\ \theta \end{array}$$

but this can be ruled out on other grounds - viz. first expanding, then reducing gives the derivation

$$\dfrac{\begin{array}{c} P_o \\ \phi\vee\psi \end{array} \quad \begin{array}{c}[\phi]\\ \Omega_1'\\ \theta'\\ \Omega_3\\ \theta\end{array} \quad \begin{array}{c}[\psi]\\ \Omega_2'\\ \theta'\\ \Omega_3\\ \theta\end{array}}{\theta}$$

which equals the original derivation P_1P_o only if Ω_3 is the identity; i.e. we do not have an endo-operation.)

Then (Is) says the composite operation

$$\dfrac{\begin{array}{c} P_o \\ \phi\vee\psi \end{array} \quad \begin{array}{c}[\phi]\\ \Omega_1\\ \theta\end{array} \quad \begin{array}{c}[\psi]\\ \Omega_2\\ \theta\end{array}}{\theta} \xrightarrow{(\vee\ \mathrm{Exp})} \dfrac{\begin{array}{c} P_o \\ \phi\vee\psi \end{array} \quad \dfrac{\dfrac{[\phi]}{\phi\vee\psi} \quad \begin{array}{c}[\phi]\\ \Omega_1\\ \theta\end{array} \quad \begin{array}{c}[\psi]\\ \Omega_2\\ \theta\end{array}}{\theta} \quad \dfrac{\dfrac{[\psi]}{\phi\vee\psi} \quad \begin{array}{c}[\phi]\\ \Omega_1\\ \theta\end{array} \quad \begin{array}{c}[\psi]\\ \Omega_2\\ \theta\end{array}}{\theta}}{\theta} \Longrightarrow$$

$$\xrightarrow{(\vee\ \mathrm{Red})} \dfrac{\begin{array}{c} P_o \\ \phi\vee\psi \end{array} \quad \begin{array}{c}[\phi]\\ \Omega_1\\ \theta\end{array} \quad \begin{array}{c}[\psi]\\ \Omega_2\\ \theta\end{array}}{\theta}$$

is the identity. This is precisely the equation $(\lambda\beta)(\alpha\lambda) = \lambda$ (although there P_o need not be explicitly mentioned.) (And so we have proved the claim.)

Of course, $\wedge$ is similar: $(\kappa\alpha)(\beta\kappa) = \kappa$ says that

$$\begin{array}{c}\theta\ \ \theta\\ P\ \ \Omega\\ \underline{\phi\ \ \ \psi}\\ \phi\wedge\psi\end{array} \xRightarrow{(\wedge\ \mathrm{Exp})} \begin{array}{c}\theta\ \theta\ \ \ \theta\ \theta\\ P\ \Omega\ \ \ P\ \Omega\\ \underline{\phi\ \psi}\ \ \ \underline{\phi\ \psi}\\ \underline{\phi\wedge\psi}\ \ \underline{\phi\wedge\psi}\\ \underline{\phi\ \ \ \ \ \ \ \ \psi}\\ \phi\wedge\psi\end{array} \xRightarrow{(\wedge\ \mathrm{Red})} \begin{array}{c}\theta\ \ \theta\\ P\ \ \Omega\\ \underline{\phi\ \ \ \psi}\\ \phi\wedge\psi\end{array}$$

is the identity; $(\alpha\lambda)(\lambda\beta) = \lambda$ says that

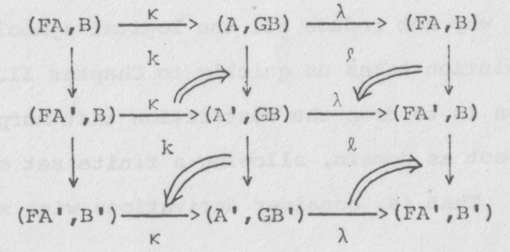
$$\begin{array}{c}\theta\\ P\\ \underline{\phi\wedge\psi}\\ \phi\end{array} \xRightarrow{(\wedge\ \mathrm{Exp})} \begin{array}{c}\theta\ \ \ \ \theta\\ P\ \ \ \ P\\ \underline{\phi\wedge\psi}\ \ \underline{\phi\wedge\psi}\\ \underline{\phi\ \ \ \ \ \ \ \ \psi}\\ \underline{\phi\wedge\psi}\\ \phi\end{array} \xRightarrow{(\wedge\ \mathrm{Red})} \begin{array}{c}\theta\\ P\\ \underline{\phi\wedge\psi}\\ \phi\end{array}$$

is the identity (and similarly for ψ, mutatis mutandis). These together give (Is) for $\wedge$.

3.11 This completes the description of the weak hom-set (or concrete) adjunctions suitable for $\vee$ and $\exists$ on one hand, and for $\wedge$ and $\forall$ on the other. Not surprisingly, $\supset$ does not have so nice a description, as it shows some of the characteristics of each, as shown by the following table:

	$\underline{\vee}$	$\underline{\wedge}$	$\underline{\supset}$
$\underline{A}$:	LPC × LPC	LPC	LPC
$\underline{B}$:	LPC	LPC × LPC	LPC
Γ:	$\vee$	diagonal	$(-)\wedge\phi$ (fixed ϕ)
G:	diagonal	$\wedge$	$\phi\supset(-)$ (fixed ϕ)
ι^F:	($\vee$ Exp)	identity	($\wedge$ Exp)
ι^G:	identity	($\wedge$ Exp)	($\supset$ Exp)
γ^F:	($\vee$ Perm)+($\vee$ Red)	identity	($\wedge$ Red)
γ^G:	identity	($\wedge$ Red)	($\supset$ Red)
κ:	($\vee$ I)	($\wedge$ I)	($\wedge$ I)+($\supset$ I)
λ:	($\vee$ E)	($\wedge$ E)	($\wedge$ E)+($\supset$ E)
$k_{A!g}$:	identity	($\wedge$ Red)	($\supset$ Red)
sense of $k_{A!g}$:	lower (or strict)	upper	upper
$\ell_{f,B}$:	($\vee$ Perm)+($\vee$ Red)	identity	($\wedge$ Red)
sense of $\ell_{f,B}$:	upper	lower (or strict)	upper
$\ell_{A!g}$:	($\vee$ Perm)	($\wedge$ Red)	($\supset$ Red)
sense of $\ell_{A!g}$:	upper	lower	lower
$k_{f,B}$:	($\vee$ Red)	identity	($\wedge$ Red)
sense of $k_{f,B}$:	lower	upper (or strict)	lower

These last entries mean that the diagram 3.6(i) (for $\vee$) and the co-diagram (for $\wedge$ - reverse the 2-cells only) are replaced (for $\supset$) by



(i)

Even worse, we no longer have the modifications α,β and so do not have the unit and counit of a desired adjunction. The best we can do is the existence of modifications

$$\alpha':\lambda\kappa \longrightarrow (F(1\_),-) \text{ and } \beta':\kappa\lambda \longrightarrow (-,G(1\_)).$$

(These are given by ($\supset$ Red) and ($\wedge$ Red) respectively.) While α' is almost a counit, β' is nothing of the sort - what should be the unit is really

$$(-,\iota^G_-): 1_{(-,G-)} \longrightarrow (-,G(1\_)),$$

given by ($\supset$ Exp). (Compare with $\vee$ and $\wedge$.) The meta-principles of §3.4 give us all the coherence we could want, but there is no way to write out the triangle equalities (which should express the strict principle (Is) of §3.10.)

3.12 There are two ways out of this problem, which are more or less equivalent, to some extent. First, we could "factor out by $\wedge$"; that is, we could introduce an equivalence relation on derivations which would have the effect of making all the $\wedge$ 2-cells equivalences, in the manner of the next chapter. Of course, doing this trivialises much of what has been said so far, and raises the question - "why reduce only $\wedge$; why not reduce all the logical symbols?" In short, this solution takes us quickly to Chapter II. The second solution is to drop the restriction that morphisms have only one object as domain, allowing a finite set of objects as domain. That is, consider derivations with more than one

assumption as well. In that case we could drop all references to $\wedge$ in the description of $\supset$, and give an analysis that resembled (in spirit, at least) that for $\wedge$ and $\forall$. (The basic outline of such an analysis is already to be seen in the table in §3.11, and in the remarks re unit, counit, and triangle equalities for $\supset$ above.) It is clear how this can be done - however, anyone who has worked with multi-categories will, I hope, forgive me if I do not give a complete analysis of the (lax) multi-2-category structure of LPC here.

§4 Formal adjointness

4.0 It is equally possible to express the proof-theoretical structure in terms of weak formal adjointness; i.e. in terms of units and counits. This works reasonably well for $\wedge, \forall$, and $\supset$, but not so well for $\vee$ and $\exists$; in these latter cases, it is necessary to go outside the usual framework, using some of the notions from the concrete adjointness of §3, to describe the structure. We will give a brief outline of the structures obtained, and compare them with those of §3. It will be clear from this discussion that the weak adjointnesses suitable for these proof-theoretic considerations do not have (natural) equivalent concrete and formal descriptions. (It should be kept in mind that these descriptions are not characterisations.)

4.1 We begin with the same initial data as in §3.1: two 2-categories $\underline{A}, \underline{B}$, and lax (in the sense there) 2-functors

$$\underline{A} \underset{G}{\overset{F}{\rightleftarrows}} \underline{B}.$$

Then, there are supposed to exist lax 2-natural transformations

$$\eta : 1_{\underline{A}} \longrightarrow GF \quad \text{and} \quad \varepsilon : FG \longrightarrow 1_{\underline{B}} .$$

These are lax in the following sense:

for $f: A' \longrightarrow A$ a morphism of $\underline{A}$, there is a 2-cell

$$n_f : GF(f)\eta_{A'} \Longrightarrow \eta_A f ;$$

for $g: B \longrightarrow B'$ a morphism of $\underline{B}$, there is a 2-cell

$$e_g : \varepsilon_{B'} FG(g) \Longrightarrow g\varepsilon_B .$$

(This is not quite right - we shall see we must modify this last condition for $\vee$ and $\exists$.) Naturality at the level of 2-cells will be satisfied.

Furthermore, there are modifications

$$d: (\varepsilon F)(F\eta) \Longrightarrow F(1\_); \quad b: (G\varepsilon)(\eta G) \Longrightarrow G(1\_).$$

We may suppose whatever coherence we wish, as we did in §3.

4.2 Rather than show how the logical structure may be described this way, we shall compare this set-up with that of §3; this will show just why this set-up does and does not work in the various cases, and will point out its inadequacies.

For all connectives and quantifiers, η is given by the introduction rule and ε by the elimination rule: in each case, if A (respectively B) is an object of $\underline{A}$ (respectively $\underline{B}$), then $\eta_A = \kappa_{A,FA}(1_{FA})$ and $\varepsilon_B = \lambda_{GB,B}(1_{GB})$. Using this, we may construct n_f and e_g: first notice that then

$$\eta_A f = (f,GFA)\kappa_{A,FA}(1_{FA}); \; GF(f)\eta_{A'} = (A',GFf)\kappa_{A;FA'}(1_{FA'});$$

$$g\varepsilon_B = (FGB,g)\lambda_{GB,B}(1_{GB}); \; \varepsilon_{B'}FG(g) = (FGg,B')\lambda_{GB;B'}(1_{GB'}).$$

Consider the case of $\supset$: we may take n_f to be the 2-cell

$$GF(f)\eta_{A'} = (A',GFf)\kappa_{A;FA'}(1_{FA'}) \xrightarrow{k_{A;Ff}} \kappa_{A;FA}(FA;Ff)(1_{FA'}) =$$

$$= \kappa_{A;FA}(Ff,FA)(1_{FA}) \xrightarrow{k_{f,FA}} (f,GFA)\kappa_{A,FA}(1_{FA}) = \eta_A f .$$

Similarly, we may take e_g to be the 2-cell

$$\varepsilon_{B'}FG(g) = (FGg,B')\lambda_{GB;B'}(1_{GB'}) \xRightarrow{\ell_{Gg,B'}} \lambda_{GB,B'}(Gg,GB')(1_{GB'})$$

$$= \lambda_{GB,B'}(GB,Gg)(1_{GB}) \xRightarrow{\ell_{GB,g}} (FGB,g)\lambda_{GB,B}(1_{GB}) = g\varepsilon_B \ .$$

Looking at the table in §3.11, we see that this works because $k_{A',g}$ and $k_{f,B}$ (respectively $\ell_{f,B}$ and $\ell_{A',g}$) have opposite senses. The same thing will work for ∧ and ∀ , because $k_{f,B}$ and $l_{f,B'}$ being identities, could have their senses reversed if we so choose. For the same reason we can construct n_f for ∨ (and ∃), but unless we "factor out by (Perm)" (i.e. consider (Perm) as an equivalence) we must replace the existence of e_g by something weaker: there are 2-cells

$$\varepsilon_{B'}FG(g) \xRightarrow{e^0_g} \lambda_{GB,B'}(Gg) \xLeftarrow{e^1_g} g\varepsilon_B \ .$$

(There is no notation for $\lambda_{GB,B'}(Gg)$ in the "formal" set-up - we must use the λ notation from the "concrete" set-up.)

4.3 The modifications d and b are somewhat more complicated when expressed in terms of the notation of §3; they depend on the expression of Ff (respectively Gg) in terms of λ, κ, and f (respectively g). For example, for the case of ∨ (and ∃), we have the equation (for $A' \xrightarrow{f} A$ a morphism of $\underline{A}$)

$$F(f) = \lambda_{A;FA}(f,GFA)\kappa_{A,FA}(1_{FA}) \ .$$

(This is more familiarly written $Ff = \lambda_{A',FA}(\eta_A f)$, or even $Ff = \widehat{\eta_A f}$!) Then for an object A of $\underline{A}$, d_A is the 2-cell

$$\varepsilon_{FA}F\eta_A = (FA,\varepsilon_{FA})\lambda_{A,FGFA}(\eta_A,GFGFA)\kappa_{GFA,FGFA}(1_{FGFA}) \Longrightarrow$$

$$\xRightarrow{\ell_{A',\varepsilon_{FA}}\mathrm{id}} \lambda_{A,FA}(A,G\varepsilon_{FA})(\eta_A,GFGFA)\kappa_{GFA,FGFA}(1_{FGFA}) =$$

$$= \lambda_{A,FA}(\eta_A,GFA)(GFA,G\varepsilon_{FA})\kappa_{GFA,FGFA}(1_{FGFA}) \xRightarrow{\mathrm{id}\ k_{GFA',\varepsilon_{FA}}}$$

$$\Longrightarrow \lambda_{A,FA}(\eta_A,GFA)\kappa_{GFA,FA}(FGFA,\varepsilon_{FA})(1_{FGFA}) =$$

$$= \lambda_{A,FA}(\eta_A,GFA)\kappa_{GFA,FA}\lambda_{GFA,FA}(1_{GFA}) \xRightarrow{\beta_{GFA,FA}}$$

$$\Longrightarrow \lambda_{A,FA}(\eta_A,GFA)(1_{GFA}) = \lambda_{A,FA}\kappa_{A,FA}(1_{FA}) = F(1_A) \ .$$

The other constructions are more or less similar - we omit further details. (One remark, though: in considering ⊃, it is necessary to consider κ and λ not only for ⊃, but also for ∧. This should not be too surprising, however.)

4.4 Remark: With this presentation, there is no clear-cut statement of principle (Is), the "triangle equalities" of §3.10. To be sure, the three meta-principles are to be found in various coherence conditions, just as they were in §3, but we might expect that (Is) be singled out for greater attention, as it was there. Considering Butler's proof of equivalence between (one type of) formal and concrete adjointness [06], one might consider some condition like his condition (𝔸) and (𝔹). (For they correspond to his (𝕂) and (𝕃), which are our triangle equalities.) If one writes out the corresponding diagrams, one quickly realises that this will not do. His (𝔸) is commutativity of the square (in his notation)

$$\begin{array}{ccc} 1_{BA}\cdot\alpha & \xrightarrow{Ba.\alpha} & B\beta A\cdot BA\alpha.\alpha \\ {\scriptstyle id}\Big\| & & \Big\downarrow{\scriptstyle B\beta A\cdot\alpha\alpha} \\ 1_{BA}\cdot\alpha & \xleftarrow{bA\cdot\alpha} & B\beta A\cdot\alpha BA\cdot\alpha \end{array}$$

In our case, this would have to be replaced by the diagram

$$\begin{array}{ccccc} 1_{GF}\cdot\eta & \xrightarrow{1_G\cdot\iota^F\cdot\eta} & 1_G\cdot F(1)\cdot\eta & \xleftarrow{Gd\cdot\eta} & G\varepsilon F\cdot GF\eta\cdot\eta \\ {\scriptstyle id}\Big\| & & & & \Big\downarrow{\scriptstyle G\varepsilon F.n\eta} \\ 1_{GF}\cdot\eta & \xrightarrow{\iota^G.1_F.\eta} & G(1)\cdot 1_F\cdot\eta & \xleftarrow{bF\cdot\eta} & G\varepsilon F\cdot\eta GF\cdot\eta \end{array}$$

To state (Is), it would be necessary to use at least some of the concrete presentation of §3 - in fact, it is not hard to see that the formal set-up does not even have enough notation! This is yet another inadequacy of the formal presentation.

Chapter II

§0 Introduction

We have seen in the preceding chapter that one does not get a natural or satisfying 2-categorical structure by considering ordinary proof theory from a category point of view. However, it will be clear that if we ignored all the 2-cells (treating them as equivalences, and considering any two morphisms with a 2-cell between them as equivalent) we would have a well behaved 1-categorical structure, with the expected properties, that ∧ be a product, ∨ a coproduct, and so on. In this chapter, we shall do precisely that, but to make life a little more interesting (and to arrive at a "smoother" categorical structure), we shall consider not just LPC, but LPCE (that is, LPC with equality.) We shall also take more seriously the restrictions on LPC which made it differ from ordinary first order logic.

Similar results have been obtained independently by Moens, [32] .