Lecture Notes on
From Inference to Logical Connectives

15-836: Substructural Logics
Frank Pfenning

Lecture 2
August 31, 2023

1 Introduction

Whenever we investigate logic we have to investigate proofs. They have many roles, but in the context of the (sub)structural inference from the first lecture they justify the truth of the propositions in a state. In different examples, this may give rise to different concrete interpretations of proofs: in graph reachability, proofs corresponded to paths, in coin exchange to a sequence of exchange actions, in parentheses matching to a kind of parse tree, in binary increment to a trace of the computation, and in blocks world to a plan to achieve a goal state. While in structural inference it seemed convenient to represent proofs as terms, this did not work so well with substructural inference, at least in part because of rules with multiple (or zero) conclusions. We start this lecture by presenting (but not pursuing in full rigorous detail) an idea proposed by C. B. Aberlé during the first lecture because it is elegant and has some useful properties we can take of advantage later in the course. Will call this proof representation CBA diagrams.

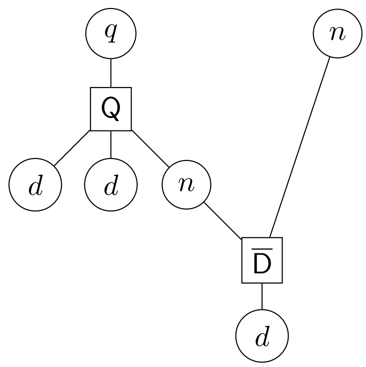
After this, we reflect back on what (substructural) inference can and cannot achieve. One limitation is that we cannot, inside the logic, ask questions such as "*If we start with edges from a to b, from b to c and from b to d, is there a path from a to d?*". Instead, we can only ask this looking at states "from the outside". Asking if-then questions, though, is central to logic so we start our path towards being able to express richer statements. For this, we need logical connectives such as conjunction, implication, disjunction, etc. There will be some surprises along the way, because connectives in substructural logics have some unusual properties.

2 CBA Diagrams for Substructural Proofs

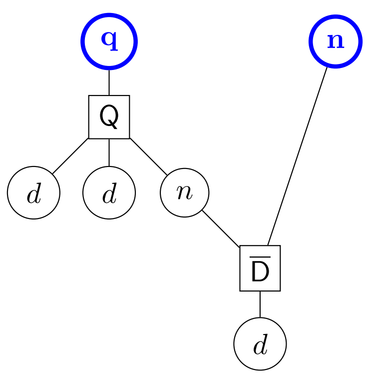
We start with linear inference and the coin exchange example

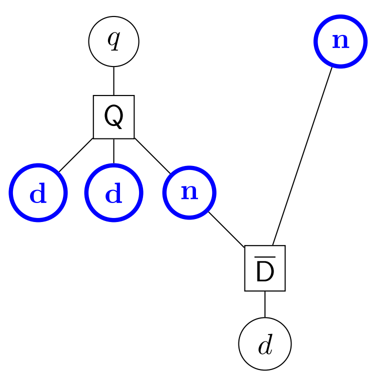
$$\frac{q}{d \quad d \quad n}\ \mathsf{Q} \qquad \frac{d \quad d \quad n}{q}\ \overline{\mathsf{Q}} \qquad \frac{d}{n \quad n}\ \mathsf{D} \qquad \frac{n \quad n}{d}\ \overline{\mathsf{D}}$$

The idea is that propositions are nodes in a diagram and inference rule applications are boxes connecting premises to conclusions. We build up the diagram here step by step, each inference adding new propositions.

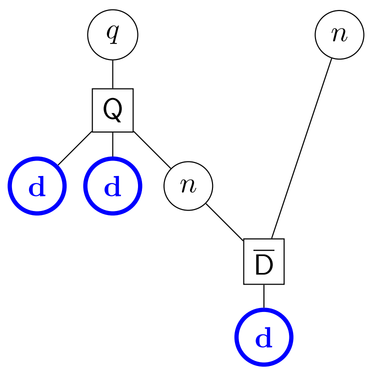


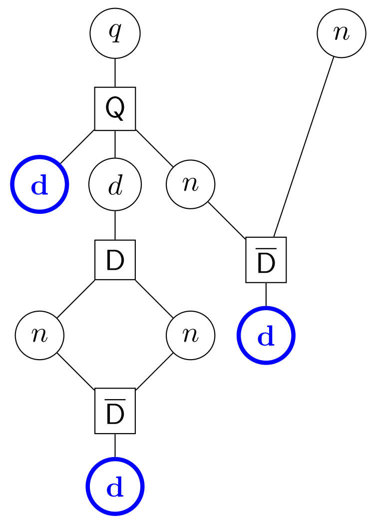
We can now visualize a reachable state as a horizontal slice through the CBA diagram. For example, the initial state would include the top two nodes. On he right, we show the slice after the Q inference. The nodes in the slides are in bold and in blue.





The final state in this example will be the slice just containing the three dimes, shown on the left below. Mathematically, it would be convenient to define the graph and possible slices simultaneously by allowing an inference rule to be applied to an existing slice and moving it by replacing the premises in it by the conclusions. We may still have to account for the fact that identical propositions are indistinguishable. For example, the version of the second diagram where the line from the left n crosses the one from the right n should be identified. The BCA diagram here succinctly represents a number proofs in one diagram. But we can also change the diagram and it would yield a different proof. For example, we could decide to exchange one of the dimes for two nickels and then back to a dime, as shown on the right where the final slice also contains three dimes.



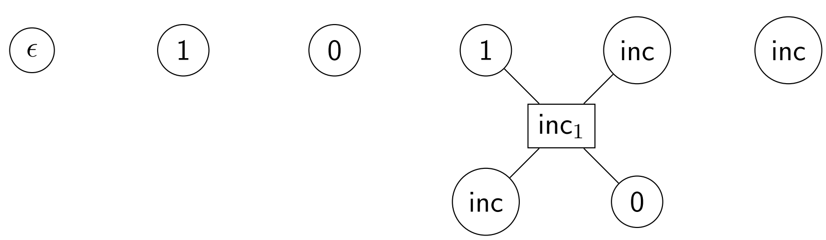


3 CBA Diagrams and True Concurrency

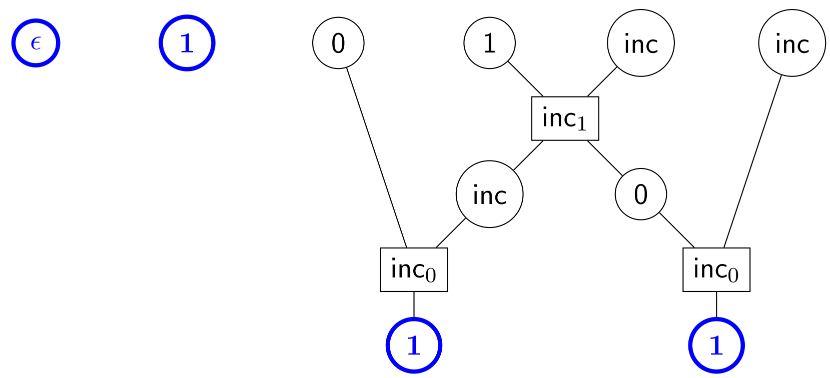
As a next example we considered CBA diagrams for ordered inference, using the example of binary increment.

$$\frac{\mathsf{0}\quad\mathsf{inc}}{\mathsf{1}}\;\mathsf{inc}_0 \qquad\qquad \frac{\mathsf{1}\quad\mathsf{inc}}{\mathsf{inc}\quad\mathsf{0}}\;\mathsf{inc}_1 \qquad\qquad \frac{\epsilon\quad\mathsf{inc}}{\epsilon\quad\mathsf{1}}\;\mathsf{inc}_\epsilon$$

We use the example from the first lecture, incrementing the number 5 twice to arrive at 7. The first step is forced.



At this point, either of the two remaining increments can interact with the 0 to its left. Since they are independent, let's do both.

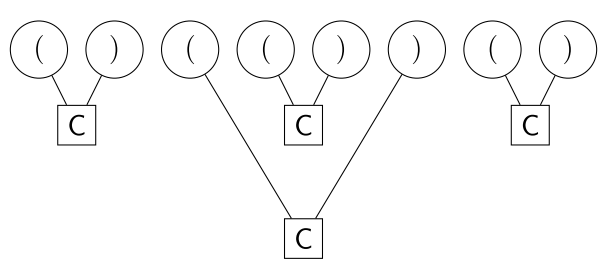


The final slice ϵ 1 1 1 is quiescent. Inference here proceeded with don't care nondeterminism and the last two inferences using inc_0 are *independent* in the sense that neither consumes or produces a proposition that the other needs. Therefore, the order between these two inferences is irrelevant. A nice property of the CBA diagrams here is that you end up with the same diagram (and therefore the same proof) no matter which of the independent actions are taken first. This phenomenon is known as *true concurrency*: we cannot observe the order of independent events. This will be useful later when we specify and reason about parallel and concurrent programming languages.

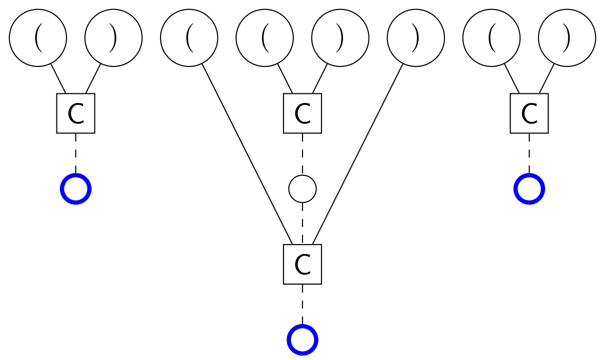
What is the difference between linear and ordered BCA diagrams? One observation made in lecture is that the lines in order diagrams cannot cross the way they can in linear diagrams. But that's not quite sufficient: the premises of a rule application need to be adjacent. That's not always obvious. As a last substructural example, let's consider matching parentheses.

$$\frac{(\quad)}{\cdot}\ \mathsf{Cancel}$$

We draw the complete BCA diagram right away for the example from lecture.



Since the rule of cancellation has no conclusions, there are no outgoing edges from the corresponding boxes. Then we see there is essentially only one proof of the empty slice, because all the actions appear to be independent. However, that's not quite the case: between the two applications that are pictured on top of each other, the one higher up needs to be done first so that the two premises for the lower cancellation are adjacent. We can indicate that, for example, with a dashed line or empty circle indicating zero conclusion, but allowing us to express an otherwise implicit dependency.



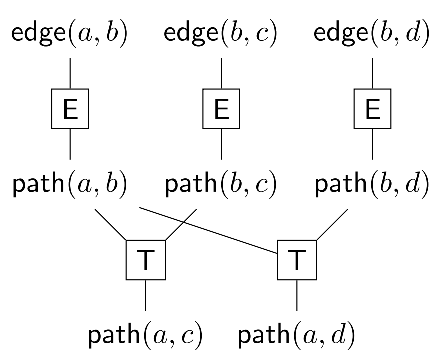
The final slice (shown above in bold blue) is empty.

4 CBA Diagrams for Structural Inference

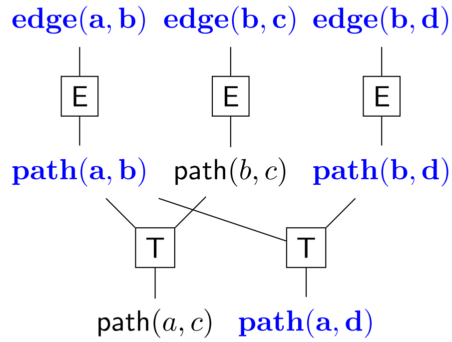
We can apply the idea of CBA diagrams to structural inference, but slices are somewhat different because of the monotonic nature of inference. We just show the example of graph reachability from last lecture.

$$\frac{\mathsf{edge}(x,y)}{\mathsf{path}(x,y)}\ \mathsf{Edge} \qquad\qquad \frac{\mathsf{path}(x,y) \quad \mathsf{path}(y,z)}{\mathsf{path}(x,z)}\ \mathsf{Trans}$$

We go directly to the diagram at the point of saturation.



A slice now has to be "upwards closed" to capture the fact that inference is monotonic: we only add new facts to the slice. In the diagram below we colored a slice containing $\mathsf{path}(a, d)$ in bold blue.



While we can intuitively construct such slices that are closed under an ancestor relation, we won't attempt to give a formal definition in this lecture. In particular the fact that there may be multiple proofs of some propositions requires some decisions regarding such a definition.

5 Hypothetical Judgments

Using inference rules we can specify the meaning of basic propositions and reason about them with (sub)structural inference. We now pose several questions in the examples we have considered. We can answer these questions via inference, but, strangely, we cannot even asked them *within* logic because we have no logical connectives!

- If we start with edges from a to b, from b to c, and from b to d, is there a path from a to d?
- Can we exchange a quarter and a nickel for three dimes?
- Is ( ) (( )) ( ) a word with matching parentheses?
- Is ϵ 1 1 1 the result of incrementing ϵ 1 0 1 twice?
- Starting from an initial state where the robot hand is empty, and we have a stack of a on b, with b on the table, and a free spot on the table, can we reach a state where b is on a?

These questions use forms of conjunction and implication, so we have to consider what the meaning of such connectives is and how we can reason with them.

Let's look at the question in the middle: "*Can we exchange a quarter and a nickel for three dimes?*" We are asking if the state with three dimes is reachable from the state with three dimes. We visualize this question as

$$\begin{array}{c} q, n \\ \vdots \\ d, d, d \end{array}$$

So we not only have an initial state, but also a desired final state. This is a form of a *linear hypothetical judgment*: if we had a quarter and a nickel, could we (by linear inference) reach the state where we have three dimes. To express this *within* the logic, we need to figure out how to *internalize* the components of this hypothetical judgment as logical propositions. For linear logic, it will turn out that A, B (two separate propositions in a state) is expressed as $A \otimes B$. This allows us to combine the initial and final states into a single proposition. Then the vertical dots are expressed as a *linear implication*, that is,

$$\begin{array}{c} A \\ \vdots \\ B \end{array}$$

becomes the proposition $A \multimap B$. So the original situation, as a single propositions, is written as $q \otimes n \multimap d \otimes d \otimes d$.

An inference rule is also an example of a hypothetical judgment. For example,

$$\frac{q}{d\ d\ n}\ \mathsf{Q}$$

expresses that if we *had* a quarter, we *could* exchange it for two dimes and a nickel. So it would be internalized as

$$q \multimap d \otimes d \otimes n$$

There is one caveat, though: an inference rule can be used as many times as we wish, even if the process of inference itself is linear. We say the rule is *persistent* while propositions in the state like q or d are *ephemeral*. In order to express inference rules within the logic we therefore will need to model persistence. We will return to this point later in the lecture.

Focusing on the hypothetical judgment for now, we write

$$\Delta \longrightarrow \Sigma \qquad \text{for} \qquad \begin{array}{c} \Delta \\ \vdots \\ \Sigma \end{array}$$

primarily because it is easier to typeset. This form of hypothetical judgment has some nice properties. For example, it is reflexive and transitive. Furthermore, it affords is the option of reasoning "in two directions". We can either perform an inference starting with Δ, using the inference rules as we have done so far, or we can conjecture how we might prove Σ and use an inference bottom-up. For example, we might reduce

$$\Delta \longrightarrow d, d, d$$

to proving

$$\Delta \longrightarrow d, d, n, n$$

instead.

Unfortunately this attempt at explaining hypothetical judgments as reachability between states runs into serious problem when we consider implication. How would we prove $A \multimap B$ (remembering that this means "*if we had an A we could deduce B*")? The obvious answer is that we would add A to the state and then attempt to deduce B. That is:

$$\frac{\Delta, A \vdash B, \Sigma}{\Delta \vdash A \multimap B, \Sigma} \; ??$$

Unfortunately, this brings Σ into the scope of A, which is incorrect! For example, the following purported proof is clearly wrong because the hypothesis A is supposed to be available only for the proof of B and not A.

$$\frac{\dfrac{}{A, B \vdash B, A}\;\mathsf{id}}{B \vdash A \multimap B, A} \; ??$$

In order to extract ourselves from such incorrect reasoning we limit the conclusion to be a single formula and write

$$\Delta \vdash A$$

for linear logic, with corresponding judgments for ordered ($\Omega \vdash A$) and structural ($\Gamma \vdash A$) logics. This structure is called a *sequent*, with the state to the left consisting of the *antecedents* and the proposition to the right being the *succedent*.

We can complete a hypothetical proof when a hypothesis (antecedent) matches the conclusion. In the sequent calculus, this is called the rule of *identity*. In the linear and ordered case, this must be exact; in the structural case we can silently ignore some antecedents. This is also a structural property, but it cannot be presented as an equational property. Instead, it should be thought of as a relation between states, $\Gamma \supseteq \Gamma'$. We could either have a general rule of weakening (from $\Gamma \vdash C$ infer $\Gamma, A \vdash C$ for any A) or we can build it into the initial sequents, that is, sequents without premises. We illustrate here the latter.

structural	linear	ordered
$\dfrac{}{\Gamma, A \vdash A}\;\mathsf{id}$	$\dfrac{}{A \vdash A}\;\mathsf{id}$	$\dfrac{}{A \vdash A}\;\mathsf{id}$

6 Internalizing State Formation as Conjunction

The first connective we consider is the one that expresses the state former, written as a comma in structural and linear logic and juxtaposition in ordered logic. We have the following rules, adhering to our convention that Γ is a structural state, Δ is a linear state, and Ω is an ordered state. The rules below are *left rules* because they apply to the proposition among the *antecedents*, that is, to the left of the turnstile '$\vdash$'. Because we internalize state formation, it makes sense to first consider the proposition with the connective to be among the antecedents.

structural	linear	ordered
$\dfrac{\Gamma, A, B \vdash C}{\Gamma, A \wedge B \vdash C}\ \wedge L$	$\dfrac{\Delta, A, B \vdash C}{\Delta, A \otimes B \vdash C}\ \otimes L$	$\dfrac{\Omega_L\ A\ B\ \Omega_R \vdash C}{\Omega_L\ (A \bullet B)\ \Omega_R \vdash C}\ \bullet L$

We see that the notation for the different forms is different. We also see that in the structural and linear cases we write the conjunction in the rightmost position, which is always possible due to the law of exchange. In the ordered case the conjunction can be anywhere in the state, with Ω_L to its left and Ω_R to its right.

According to the structural properties we would expect the following laws to hold or not hold in general. We write $A \dashv\vdash B$ for $A \vdash B$ and $B \vdash A$.

structural	linear	ordered
$A \wedge (B \wedge C) \dashv\vdash (A \wedge B) \wedge C$	$A \otimes (B \otimes C) \dashv\vdash (A \otimes B) \otimes C$	$A \bullet (B \bullet C) \vdash (A \bullet B) \bullet C$
$A \wedge B \dashv\vdash B \wedge A$	$A \otimes B \dashv\vdash B \otimes A$	$P \bullet Q \not\dashv\vdash Q \bullet P$
$A \wedge A \dashv\vdash A$	$P \otimes P \not\dashv\vdash P$	$P \bullet P \not\dashv\vdash P$

In the cases where the entailments do not hold (and neither direction is correct), we use atomic propositions P and Q for our counterexamples because there may be some specific propositions A and B for which such a law might hold.

In order decompose connectives when they appear as a succedent we use *right rules*.

structural	linear	ordered
$\dfrac{\Gamma \vdash A \quad \Gamma \vdash B}{\Gamma \vdash A \wedge B}\ \wedge R$	$\dfrac{\Delta_1 \vdash A \quad \Delta_2 \vdash B}{\Delta_1, \Delta_2 \vdash A \otimes B}\ \otimes R$	$\dfrac{\Omega_1 \vdash A \quad \Omega_2 \vdash B}{\Omega_1\ \Omega_2 \vdash A \bullet B}\ \bullet R$

In the structural rule, all antecedents are available in both premises, so there is only one way to apply this rule. This suggests we are reading the rule bottom-up, which is true for our formulations of the sequent calculus. In the linear rule, we have to find a way to split the antecedents among the two premises. Because the antecedents satisfy exchange, any submultiset Δ_1 can be used to prove A, with the remaining antecedents Δ_2 going to the proof of B. So there are 2^n possible ways to apply this rule when there are n antecedents. In the ordered rule, we have to split the ordered context somewhere, and everything to the left of the split has to prove A, while everything to the right has to prove B. So there are $n+1$ ways to possibly apply this rule when there are n antecedents.

7 Internalizing Hypothetical Judgments as Implication

Among the statements we wanted to express as a logical proposition were if-then statements, such as "*If we had a quarter and a nickel, we could exchange them for three dimes.*" For this, we need implication, which renders the turnstile $\vdash$ as a logical connective. We'll consider this for ordered logic in the next lecture and just focus on structural and linear logic.

Intuitively, $A \supset B$ should be true if B is true under the assumption A. When our logic is structural, A can be used arbitrarily many times in the proof of B. In linear logic, A becomes part of the linear state and will be consumed when inference rules are applied, so we have $A \multimap B$ as a different notation. In this case, we write out the right rules first, because they most naturally relate the meaning of the connective to the hypothetical judgment.

structural	linear
$\dfrac{\Gamma, A \vdash B}{\Gamma \vdash A \supset B} \supset\! R$	$\dfrac{\Delta, A \vdash B}{\Delta \vdash A \multimap B} \multimap\! R$

What are the corresponding left rules? An assumption $A \supset B$ licenses us to assume B if we can prove A. That is:

structural	linear
$\dfrac{\Gamma \vdash A \quad \Gamma, B \vdash C}{\Gamma, A \supset B \vdash C} \supset\! L$	$\dfrac{\Delta_1 \vdash A \quad \Delta_2, B \vdash C}{\Delta_1, \Delta_2, A \multimap B \vdash C} \multimap\! L$

These, like all rules in our sequent calculi, should be read from the bottom upwards. The slightly subtle point in $\supset\! L$ is that because the antecedents form a set (and so comma is a form of union), the implication itself still remains in both premises. By contrast, in the linear case we need to split up the antecedents between the two premises and we also remove the implication itself.

As mentioned before, inference rules themselves also form a hypothetical judgment. Below are two examples from the coin exchange:

rule	proposition
$\dfrac{q}{d\ d\ n}\ Q$	$q \multimap d \otimes d \otimes n$
$\dfrac{n\ n}{d}\ \overline{D}$	$n \otimes n \multimap d$

Here is the proof that we can exchange a quarter and a nickel for three dimes. The first few steps are easy. We have offset the inference rules to make them visually easier to read.

$$\dfrac{\dfrac{\begin{array}{c}\vdots\\ q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d, \quad q, n \vdash d \otimes d \otimes d\end{array}}{q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d, \quad q \otimes n \vdash d \otimes d \otimes d}\otimes L}{q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d \quad \vdash q \otimes n \multimap d \otimes d \otimes d}\multimap R$$

At this point we need to decide which implication left rule to use. Since we have q as an antecedent, it makes sense to use the first.

$$\dfrac{\dfrac{\dfrac{\dfrac{}{q \vdash q}\,\mathsf{id} \quad \begin{array}{c}\vdots\\ d \otimes d \otimes n, \quad n \otimes n \multimap d, \quad n \vdash d \otimes d \otimes d\end{array}}{q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d, \quad q, n \vdash d \otimes d \otimes d}\multimap L}{q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d, \quad q \otimes n \vdash d \otimes d \otimes d}\otimes L}{q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d \quad \vdash q \otimes n \multimap d \otimes d \otimes d}\multimap R$$

Now we can break up $d \otimes d \otimes n$ and then apply implication left again. A key aspect of the implication left rule is how we split the antecedents, so the two nickels go to the first premises and the two dimes to the second (to be joined by the succedent of the implication, which is the third dime).

$$\dfrac{\dfrac{\dfrac{\dfrac{}{q \vdash q}\,\mathsf{id} \quad \dfrac{\dfrac{\begin{array}{c}\vdots\\ n, n \vdash n \otimes n\end{array} \quad \begin{array}{c}\vdots\\ d, d, d \vdash d \otimes d \otimes d\end{array}}{d, d, n, \quad n \otimes n \multimap d, \quad n \vdash d \otimes d \otimes d}\multimap L}{d \otimes d \otimes n, \quad n \otimes n \multimap d, \quad n \vdash d \otimes d \otimes d}\otimes L^2}{q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d, \quad q, n \vdash d \otimes d \otimes d}\multimap L}{q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d, \quad q \otimes n \vdash d \otimes d \otimes d}\otimes L}{q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d \quad \vdash q \otimes n \multimap d \otimes d \otimes d}\multimap R$$

The remaining steps are straightforward applications of $\otimes R$ and identities.

$$
\dfrac{\dfrac{}{q \vdash q}\,\mathsf{id} \qquad \dfrac{\dfrac{\dfrac{\dfrac{}{n \vdash n}\,\mathsf{id} \quad \dfrac{}{n \vdash n}\,\mathsf{id}}{n, n \vdash n \otimes n}\,\otimes R \qquad \dfrac{\dfrac{}{d \vdash d}\,\mathsf{id} \quad \dfrac{\dfrac{}{d \vdash d}\,\mathsf{id} \quad \dfrac{}{d \vdash d}\,\mathsf{id}}{d, d \vdash d \otimes d}\,\otimes R}{d, d, d \vdash d \otimes d \otimes d}\,\otimes R}{d, d, n, \quad n \otimes n \multimap d, \quad n \vdash d \otimes d \otimes d}\,\multimap L}{d \otimes d \otimes n, \quad n \otimes n \multimap d, \quad n \vdash d \otimes d \otimes d}\,\otimes L^2}{\dfrac{q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d, \quad q, n \vdash d \otimes d \otimes d}{\dfrac{q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d, \quad q \otimes n \vdash d \otimes d \otimes d}{q \multimap d \otimes d \otimes n, \quad n \otimes n \multimap d \quad \vdash q \otimes n \multimap d \otimes d \otimes d}\,\multimap R}\,\otimes L}\,\multimap L
$$

Even though this was not difficult, it is considerably longer and more elaborate that our earlier proof using linear inference. So we pay some price for expressing all the rules and components of the state in propositional form.

We have also cheated. We put exactly one copy of the two needed inference rules among the antecedents. But even in linear inference, the inference rules themselves can be used arbitrarily often. So, really, the propositions $q \multimap d \otimes d \otimes n$ and $n \otimes n \multimap d$ (and the ones corresponding to the other two rules we omitted) should be *persistent*! There are multiple solutions on how to achieve this, which we discuss in the next section.

8 Persistence as a Modality

We have characterized propositions in a linear state as *ephemeral* because they are consumed as part of linear inference. In contrast, propositions in a structural state are *persistent*: they are never removed, even if they may eventually be ignored. In order to internalize (persistent) rules as propositions into linear logic, the simplest way is to make them persistent. Then we have to kinds of antecedents: persistent and ephemeral ones. It is not difficult to imagine what the rules might then look like.

Another solution goes a little further: it also internalizes the very notion of persistence into linear logic as a modal operator $!A$ (pronounced "*of course* A" and sometimes "*bang* A". The key idea is that the proposition $!A$ itself is also linear (rather than persistent), but we have explicit rules to duplicate and delete such propositions. They are the following:

$$
\dfrac{\Delta, !A, !A \vdash C}{\Delta, !A \vdash C}\,\mathsf{contraction} \qquad \dfrac{\Delta \vdash C}{\Delta, !A \vdash C}\,\mathsf{weakening} \qquad \dfrac{\Delta, A \vdash C}{\Delta, !A \vdash C}\,!L
$$

With these rules we can obtain as many copies of A from $!A$ as we want. The $!L$ rule is also called *dereliction*. But what is the correct right rule? Because we are supposed to be able to generate as many copies of $!A$ as we want, any proof of A can only depend on propositions that can be duplicated and erased themselves. That is:

$$\frac{!\Delta \vdash A}{!\Delta \vdash !A}\ !R$$

where $!\Delta$ means that every antecedent in Δ has the form $!B$. In the next lecture we will see some techniques to explicitly construct counterexamples to wrong rules, such as the one where we do not restrict the context.

Returning to our previous example, the rules now become

$$\Delta_0 = !(q \multimap d \otimes d \otimes n),\ !(d \otimes d \otimes n \multimap d),\ !(d \multimap n \otimes n),\ !(n \otimes n \multimap d)$$

and it should be easy to see how to construct a proof of

$$\Delta_0 \vdash q \otimes n \multimap d \otimes d \otimes d$$

using the new rules following the blueprint of our previous derivation.