❖ *English Language, Formal Language* ❖

2.2. Formal Language, Informally

The proposed formal test of validity requires that for any argument we do the following.

> **1. Get the form of the argument.**
> **2. Test that form for validity.**

Here we begin mastering the first task: isolating an argument's logical skeleton, beneath the irrelevant flesh of subject matter.

1. Logical Form: Four Examples. Our 'logical x-ray' for isolating form is a special-purpose language which speaks only about logical form – what we'll call "the formal language".[1] If an English argument with both logical form and subject matter is translated into this formal language, only the form of the argument will survive the translation. Since the formal language has no words for anything but form, the subject matter will be 'lost in translation'.

So we take our first task – **Get the form** – and break it into two smaller tasks.

> **1. Get the form of the argument.**
> > **1a. Build a language of pure logical form.**
> > **1b. Translate from English into the formal language.**
>
> ***(And then:***
> ***2. Test the form.)***

We've already seen examples of the sort of logical form such a language would discuss. For instance, we stripped the subject matter words from the following argument.

---

[1] In fact we'll be looking at series of ever-larger formal languages through Chapter Six (not to mention a proliferation of variant languages at the end of Chapter Three). But unless it's ambiguous which language is being discussed, the term "the formal language" will refer to the formal language of that chapter.

1. Either Barbie will scale the cliff or Jack will scale the cliff.
2. Barbie will not scale the cliff.

∴ Jack will scale the cliff.

That left us with this logical skeleton.

1. Either ● or ▲ .
2. Not ● .

∴ ▲ .

(Once again, "●" and "▲" are just blanks marking spots where the subject matter sentences go.) Though all the subject matter words are gone here, the words "either… or" and "not" remain – suggesting that they are the rare sort of English words discussing logical form, not subject matter.

We likewise stripped away the subject matter words from this argument.

1. Jake's asleep and Jezebel's asleep.

∴ Jake's asleep.

But the word "and" remains – providing another example of logical form.

1. ● and ▲ .

∴ ● .

We build a list of examples of logical form – the sorts of things a language of form should talk about.

Four Examples of Logical Form:

1.
2. Not
3. And
4. Either… or

A blank space remains in this list for one further example of logical form: marking when different spots in the argument are occupied by the **same subject matter sentence**, and when by **different** subject matter sentences.

To see why that's essential to logical form, compare two different arguments. The first is a valid English argument, with a sketch of its familiar logical form.

Either we're having tacos for dinner, or we're having chicken for dinner.	Either ● or ▲ .
We are not having tacos for dinner.	Not ● .
(So,) We're having chicken for dinner.	∴ ▲ .

As the underlining emphasizes, the same subject matter sentence "We're having chicken for dinner" appears as both the right half of the first premise, and as the conclusion. We recognized this in the logical form by putting the same symbol, "▲," in both spots.

The next argument, by contrast, is clearly **invalid** – its conclusion does **not** follow from its premises.[2]

Either we're having tacos for dinner, or we're having chicken for dinner.	Either ● or ▲ .
We aren't having tacos for dinner.	Not ● .
———————	———
(So,) It's snowing.	∴ ★ .

But the only difference between this argument and its earlier, valid cousin is that here the same subject matter sentence does **not** appear as both the right half of the first premise and the conclusion. (We recognize this in the logical form by putting one symbol, "▲," in the right half of the first premise, and another, "★," as the conclusion.)

Now since these arguments differ in validity, and we assume only logical form affects an argument's validity, these arguments must differ in logical form. But the only difference between them is whether or not the same subject matter sentence appears in both spots. So: **when it's the same subject matter sentence in both spots, and when it's not, makes a difference to the logical form**.

(Note: the subject matter itself doesn't affect the validity of the argument – only when it's the **same** subject matter sentence in different spots, whatever that subject matter may be. So in the valid argument form, having "▲" in two different spots showed that the same subject matter sentence appeared in both places. But "▲" says nothing about chicken – since what the subject matter is in those two spots is **not** a matter of logical form.)

[2] In 2.19.1 we'll consider mutant exceptions where an argument manages to be valid despite fitting the form labeled here as invalid. Still, we can say that this second logical form is not **guaranteed** to yield a valid argument when its blanks are filled in – unlike the earlier, valid form.

Adding this missing example of form completes our list.

> **<u>Four Examples of Logical Form</u>:**
>
> **1. When it’s the same subject matter sentence as before, when not.**
> **2. Not**
> **3. And**
> **4. Either… or**

These are the sorts of things which a language of logical form should talk about.

2. Outline of the Formal Language. Running through the list in order, our formal language **first** needs a way of marking when different spots have the same subject matter sentence, when not. Using geometrical shapes for this task, as we’ve done so far, proves impractical, since for arguments with a large number of subject matter sentences we’ll soon run out of distinct, easily-drawn shapes to mark those different sentences.

Instead we use capital letters “P” through “Z”. And though this provides only 11 markers, we can if necessary add numerical subscripts to them (for example: “P_1,” “P_2,” “Z_{256}”) to get infinitely many different markers. For obvious reasons we call these **sentence letters**.

> **Sentence letters**: capital letters “P” through “Z” (with or without numerical subscripts).

As we’ll see, linking sentence letters with subject matter sentences provides the translation bridge from English to the formal language. So an essential first step in such a translation is establishing a ‘translation dictionary’ linking sentence letters with subject matter sentences – as in this example.

> **P**: We’re having tacos for dinner
> **Q**: We’re having chicken for dinner

We call such a dictionary a **translation key** (like the key at the bottom of a map, stating that one inch on the map stands for 100 miles on land, or that a dotted line means railroad tracks).

Second on our list: a way to say "not" in the formal language. For this we introduce the following symbol, called the **tilde**.

~

While in English proper positioning of "not" can be tricky, the formal language keeps matters simple: when we want to deny a formal sentence, we attach a tilde to the **left edge** of that sentence. Using the above translation key, we can then translate the denial "We're *not* having tacos for dinner."

P: We're having tacos for dinner

We're **not** having tacos for dinner

~P

Third on our list: a way of saying "and" in the formal language. For this purpose we introduce the **wedge**.

$\wedge$

Just as the English "and" comes between two English sentences, linking them together, so the wedge comes between and links two formal sentences – as the following example illustrates.

P: We're having tacos for dinner
Q: We're having chicken for dinner

We're having tacos for dinner **and** we're having chicken for dinner

$(P \wedge Q)$

Note that with the wedge we wrap **parentheses** outside the formal sentence: left parenthesis outside the left part, "P," right parenthesis outside the right part, "Q".

$$(P \wedge Q)$$

Parentheses might look boring and irrelevant. But we'll see that they're essential for translating complicated sentences correctly.

We only add parentheses when a sentence has **both a left and a right part**. (By contrast, we didn't add parentheses to the tilde sentence "~P" – since a tilde doesn't link together left and right parts.) Here's a simple memory aid.

Left and right *parts*: left and right *parentheses*

Finally, we need a way of expressing "either… or" in formal language. We do this with the **vel**.

$$\vee$$

"Either… or" links together left and right sentences. The vel does the same, as in this example.

P: We're having tacos for dinner
Q: We're having chicken for dinner

Either We're having tacos for dinner, **or** we're having chicken for dinner

$$(P \vee Q)$$

(Because a vel joins together left and right parts, the vel wraps left and right parentheses outside those parts.)

In the jargon of logic, the symbols "~," "∧," and "∨" are called **connectives**. Connectives are the formal language counterpart to the **form phrases** of English. (Parentheses don't count as connectives; they're just punctuation.)

Though this much understanding of the formal language might seem enough to capture the logical form of English sentences, in fact it will serve only for the simplest cases – more complex cases posing an obstacle for this casual grasp of the formal language. In the sections that follow we improve on this situation through a two-pronged strategy: (i) cataloguing the stylistic complications of English, and (ii) developing a better understanding of the mechanics of the formal language.

Summary: The Formal Language
(Rough Draft)

- **Subject matter sentences** are translated by **sentence letters**: capital letters "P" through "Z" (with or without numerical subscripts).
- "**Not**" is translated by the **tilde**: ~ .
- "**And**" is translated by the **wedge**: $\wedge$.
- "**Either… or**" is translated by the **vel**: $\vee$.

Examples:

P: We're having truffles
Q: We're having grog

We're not having truffles: ~**P**
We're having truffles and we're having grog: **(P $\wedge$ Q)**
Either we're having truffles or we're having grog: **(P $\vee$ Q)**