The analysis of the beginning would thus yield the notion of the unity of being and not-being—or, in a more reflected form, the unity of differentiatedness and non-differentiatedness, or the identity of identity and non-identity.

Hegel
The Science of Logic

8.1 Qualitative and Numerical Identity

The logicians’ sense of ‘identical’

In English, we use the words ‘identity’/‘identical’ in a number of different ways.

Wider uses of ‘identity’/‘identical’

1. Mancunians have a strong sense of cultural identity.
2. Dr. Jekyll has multiple identities.
3. Jedward are almost completely identical.

The sense of ‘identity’ used in (3) is sometimes called ‘qualitative identity’.

(3) says that John and Edward are almost exactly similar in every respect.

None of these uses of ‘identical’ is the logicians’ use.

In logic, we always use ‘identical’ in the following strict sense

A is identical to B iff A is the very same thing as B
i.e. A and B are one and the same thing.

This is sometimes called ‘numerical identity’

(Unless otherwise stated ‘identity’/‘identical’ henceforth mean numerical identity/numerically identical.)

Examples

- George Orwell is identical to Eric Arthur Blair
- Dr. Jekyll is identical to Mr. Hyde
- John is not identical to Edward
A third formal language

The new language makes a single addition to $L_2$.

The language $L_=$

The language $L_=$ of predicate logic with identity adds a single binary predicate letter to the language of predicate logic $L_2$.

- $L_=$ adds the identity predicate $=$ to $L_2$.

$=$ differs from the other predicate letters in several ways.

- $P$, $R^2$, etc., are non-logical expressions.
- Different $L_2$-structures interpret them differently.
- $=$ is treated as a logical expression.
- It always has the same interpretation in any structure.
- Minor difference: we write $a = b$ (rather than $=ab$).

Syntax

We make a slight change to the definition of atomic formula.

Definition (atomic formulae of $L_=$)

All atomic formulae of $L_2$ are atomic formulae of $L_=$. Furthermore, if $s$ and $t$ are variables or constants, then $s = t$ is an atomic formula of $L_=$.

The definition of formula and sentence is otherwise just like the definition for $L_2$.

Examples

- Atomic $L_=$-formulae: $c = a$, $x = y_3$, $x = a$, $R^2 ax$.
- Complex $L_=$-formulae: $\neg x = y$, $\forall x (Rxy_2 \rightarrow y_2 = x)$.

Semantics

The definition of structure is just the same as before.

Definition: $L_=$-structure

An $L_=$-structure is simply an $L_2$-structure.

Why no change?

- Structures interpret non-logical expressions like $P$ and $a$.
- Structures do not interpret logical expressions like $\neg$ and $\forall x$.
- The fixed interpretation of logical expressions is specified in the definition of satisfaction.
  - e.g. $|\neg \phi |_A = T$ iff $|\phi |_A = F$
  - Similarly $=$ is treated as a logical expression, which is not assigned a semantic value by the structure.
  - The fixed interpretation of $=$ is specified in the definition of satisfaction.

Let $A$ be an $L_=$-structure (i.e. an $L_2$-structure). Truth in $A$ is defined just as before with one addition:

Definition: satisfaction of identity statements

(ix) $|s = t |_A = T$ if and only if $|s |_A = |t |_A$.

Note: $=$ is used in both $L_=$ and the metalanguage.

The other definitions from Chapter 5 carry over directly to $L_=$.

- Valid
- Logical truth
- Contradiction
- Logically equivalent
- Semantically consistent

These are defined just as before replacing ‘$L_2$’ with ‘$L_=$’.
Worked example

∀x∀yx = y isn’t logically true.

Counterexample: let A be an Ln-structure with domain {1, 2}.

Proof. Let α be an assignment over A.
Sufficient to prove (STP:) ∀x∀yx = y is false in A under α.

Now: ∀x∀yx = y |α_A = T iff ∀y x = y |β_A = T for every β differing from α at most in x. ∀x∀yx = y |α_A = F iff ∀y x = y |β_A = F for some β differing from α at most in x.

STP: ∀y x = y |β_A = F for some assignment β differing from α at most in x.

But: ∀y x = y |β_A = T iff x = y |γ_A = T for every γ differing from β at most in y. ∀y x = y |β_A = F iff x = y |γ_A = F for some γ differing from β at most in y.

STP: |x = y |γ_A = F for some γ differing from β at most y, differing from α in at most x and y.

So: Let γ assign x to 1 and y to 2 (otherwise agreeing with α).

=Intro

Any assumption of the form t = t where t is a constant can and must be discharged.

A proof with an application of =Intro looks like this:

[|t = t|]

Example: prove ⊢ ∀z (z = z)

=Elim

If s and t are constants, the result of appending φ[t/v] to a proof of φ[s/v] and a proof of s = t or t = s is a proof of φ[t/v].

\[
\begin{array}{c}
\vdots \\
\phi[s/v] \\
\vdots \\
\phi[t/v] \\
\end{array}
\]

=Elim

\[
\begin{array}{c}
\vdots \\
\phi[s/v] \\
\vdots \\
\phi[t/v] \\
\end{array} = \text{Elim}
\]

Worked example: prove the following.

⊢ ∀x∀y (Rxy → (x = y → Ryx))
Adequacy

Soundness and Completeness still hold.

Let $\Gamma$ be a set of $L_=$-sentences and $\phi$ an $L_=$-sentence.

**Theorem (adequacy)**

$\Gamma \vdash \phi$ if and only if $\Gamma \models \phi$.

Formalisation with identity

Using $=$ one can formalise ‘is [identical to]’ in English.

**Formalise:**

William II is Wilhelm II.

**Formalisation:** $a = b$.

**Dictionary:** $a$: William II. $b$: Wilhelm II.

**Note:** don’t confuse the ‘is’ of identity with the ‘is’ of predication.

**Formalise:**

Wilhelm II is an emperor.

**Formalisation:** $Ea$.

**Dictionary:** $a$: Wilhelm. $E$: ... is an emperor.

Here ‘is’ forms part of the predicate ‘is an emperor.’

Definite descriptions

**Examples of definite descriptions:**

- ‘the Queen’
- ‘Bellerophon’s winged horse’
- ‘the author of Ulysses’

In $L_2$: the best we can do is to formalise definite descriptions as constants.

But this isn’t perfect...
8.4 Uses of identity

Example

Bellerophon’s winged horse isn’t real; so there is something that is Bellerophon’s winged horse.

The obvious formalisation with constants is valid.

Formalisation: premiss:  \neg Rb. Conclusion: \exists x(x = b).
Dictionary: \( R \): \ldots is real. \( b \): Bellerophon’s winged horse.

(In fact: the conclusion is a logical truth.)

Source of the trouble:

- \( L \)-constants always refer to an object in a \( L \)-structure.
- definite descriptions may fail to pick out a unique object.

Russell’s theory of descriptions.

There’s a better way to formalise definite descriptions in \( L \).

Formalise:

The author of Ulysses wrote Dubliners.

Russell analyses this as the conjunction of two claims.

(i) There is exactly one author of Ulysses
(ii) and it wrote Dubliners.

Dictionary: \( A \): \ldots is an author of Ulysses.
\( W \): \ldots wrote Dubliners.

Formalisation: \( \exists x(Ax \land \forall y(Ay \rightarrow y = x) \land Wx) \)

Example

Bellerophon’s winged horse isn’t real; so there is something that is Bellerophon’s winged horse.

We can capture its non-validity by using the second formalisation of the premiss.

Dictionary: \( R \): \ldots is real.
\( B \): \ldots is a winged horse belonging to Bellerophon.

Formalisation: \( \neg \exists x(Bx \land \forall y(By \rightarrow y = x) \land \neg Rx) \).

Dubious: this is true only if there are non-real things.

Paraphrase 1: (i) there is exactly one winged horse belonging to Bellerophon and (ii) it is not real.

Formalisation 1: \( \exists x(Bx \land \forall y(By \rightarrow y = x) \land \neg Rx) \).

Paraphrase 2: It’s not the case that (i) there is exactly one winged horse belonging to Bellerophon and (ii) it is real.

Formalisation 2: \( \neg \exists x(Bx \land \forall y(By \rightarrow y = x) \land Rx) \).

The structure \( A \) is a counterexample to this argument.

\( D_A = \{ x : x \text{ is a horse} \}; |B|_A = \emptyset \).
(It doesn’t matter what the extension of \( R \) is here.)
Multiple descriptions

We deal with these much like multiple quantifiers.

**Formalise**

The author of Ulysses likes the author of the Odyssey

Dictionary: U: ... is an author of Ulysses
O: ... is an author of the Odyssey. L: ... likes ...

It’s helpful to break this into two steps.

**Partial formalisation:**

$$\exists x_1 (Ux_1 \land \forall y_1 (Uy_1 \rightarrow y_1 = x_1) \land x_1 \text{ likes the author of the Odyssey})$$

It remains to formalise ‘$x_1$ likes the author of the Odyssey’.

Logical constants

$\neg, \land, \lor, \rightarrow, \leftrightarrow, \forall, \exists$ and = are our **only** logical expressions. 45

This raises two questions:

Q1 What’s special about these expressions?

A1 Alfred Tarski proposes to analyse topic neutrality in terms of ‘permutation invariance’

- Roughly: logical expressions are the ones whose meaning is insensitive to which object is which.
- See Tarski ‘What are Logical Notions?’ *History and Philosophy of Logic* 7, 143–154.

Q2 What happens if we add more logical constants?

A2 This is the business of philosophical logic.

<table>
<thead>
<tr>
<th>Extension of $\mathcal{L}_2$</th>
<th>New logical expressions</th>
</tr>
</thead>
<tbody>
<tr>
<td>Generalised quantifiers</td>
<td>more than half</td>
</tr>
<tr>
<td></td>
<td>infinitely many, etc.</td>
</tr>
<tr>
<td>Modal logic</td>
<td>It is necessarily the case that</td>
</tr>
<tr>
<td></td>
<td>It is possibly the case that</td>
</tr>
<tr>
<td>Deontic logic</td>
<td>It is obligatory that</td>
</tr>
<tr>
<td></td>
<td>It is permissible that</td>
</tr>
</tbody>
</table>

See the finals paper 127: Philosophical Logic.