**Logical Module**

Adam InTae Gerard[[1]](#footnote-1)

1. **Introduction**
2. **Definitions**

We assume a logical module or grammatical fragment suitable for first and higher-order implementations. More precisely, I define a module m per the following:   
  
(1) A negation operator {¬}.   
(2) A set of conceptual variables {λ1, …, λn}.   
(3) A belief operator {●}.   
(4) A set of temporal operators {t1, …, tn}.   
(5) A time indexing operator {@}.   
(6) The following grammatical rules:   
  
(where wff is any well-formed formula in the language of implementation)   
  
(i) ¬wff is a well-formed formula.   
(iia) ●( λa , λb ) is a well-formed formula   
(iib) ●( λa ) is a well-formed formula   
(iic) where λa , λb range over conceptual variables.   
(iii)  wff @ ta is a well-formed formula where ta ranges over temporal operators.   
  
A module m is implemented by a language *L* whenever:   
  
(7) The syntactic marks (symbols) in (1) – (5) are in *L*’s vocabulary.   
(8) The grammatical rules in (6) are consistent with and part of *L*’s grammar.   
  
I define plug and play with respect to a language *L* as an attribute of a module whenever it, the module, can satisfy conditions (7) and (8) with respect to *L*.   
  
Tertiary bits (no pun): a module m interfaces whenever it is implemented by two languages *L*1, *L*2. Usually we think of this as a morphism (a kind of relation) between two languages. I’d prefer to focus on the module here.   
**2.0 Potential Applications**  
  
Carnap's Linguistic Frameworks   
Hegel's Dialectical Method and Science of Logic   
  
A conceptual variable denotes a logic.

1. **Revision 0.0.2** - **3.18.18** - <https://www.linkedin.com/in/adamintaegerard/> [↑](#footnote-ref-1)