1. Background

The rough idea of model theoretic semantics was present in Frege. But formal development of semantics as a rigorous mathematical theory about a logical system doesn’t really get started until Carnap and Tarski. The program is to characterize a notion of semantic value – assigned to each of the primitive expressions of the language – and to formulate rules that tell you what the semantic value of a complex expression is as a function of the semantic value of its components. A model is an assignment of values to the primitive expressions. You have the notion of 'truth in a model'. Some sentences are true in all models. These are the valid sentences. A sort of semantics for extensional propositional logic is implicit in Wittgenstein's Tractatus.

But at the time of Russell’s Principia, and of C. I. Lewis's initial formulation of Modal Logic, the formal theory is entirely syntactic. It was all about giving axioms and mechanical rules to derive theorems from them. The 'meanings' of the connectives were explained informally, using natural language.

2. Quine on Modal Logic

Quine has shown consistent hostility to Modal Logic. He said Modal Logic was conceived in sin: the sin of confusing use and mention. Quine is responsible for the fact that the use-mention distinction became a sort of fetish in philosophy. But it is true that the confusion between use and mention has led to serious philosophical errors.

The relevant distinction is roughly that between using an expression to talk about something (use) and talking about the expression itself (mention). Mention occurs most explicitly when expressions are within quotation marks.

Quine was a student of C.I. Lewis, who first developed Modal Logic as a calculus based on the meaning of an "implication" connective. On the intended meaning of the strict implication hook, "⇒ "p ⇒ q" should be synonymous with "q is deducible from p". The use mention issue here is the question: are we talking about sentences of a logical language or are we using those sentences to express something that is a function of the values of the component sentences? Deducibility is a relation between (forms of) sentences, not between values of sentences. One might think then that q is never deducible from p, because both "p" and "q" are primitive sentence letters. So Lewis explained a connective of his object language – namely "⇒" – in terms of a metalogical relation between sentences. And in that case, Quine was right that he was confusing the use and the mention of sentences.

At this time, there is no explicit notion of a value of a sentence. There were axioms and rules, and there was an informal explanation of what the connectives and
sentence letters mean. Sentences were thought to express "propositions" – whatever they are – and a connective such as "∧" was supposed to get us the proposition that \( p \) and \( q \) as a function of the proposition \( p \) and proposition \( q \). Now we take the value of a sentence to be its truth value, and the meaning of "∧" is a function of a pair of truth values to a truth values. So we don’t have to worry about what propositions are when we are doing extensional logic.

Quine recognized that the sin in conception does not necessarily carry itself over to the object which has been conceived: that the use-mention confusion is not essential to Modal Logic. But he saw clearly what commitments you have to make in order to eliminate that confusion. These he laid out in his paper "Three Grades of Modal Involvement".

3. Lewis’s S1 and S2

Lewis formulated five systems of modal logic of increasing strength: S1 to S5. Only the last two are "normal" systems, in a sense to be defined, but the weaker systems are of some interest. His idea was not to build modal logic on top of the standard extensional propositional logic, but to give axiom system from the ground up. He thought that Russell’s analysis of implication as material implication was unsatisfactory, and should be replaced by a different account. The standard account of negation, conjunction and disjunction were okay, in his view, and of course the material conditional is definable in terms of these other truth functional connectives, but Lewis thought of his strict implication as an alternative to the material conditional.

The Lewis systems were first formulated in the early years of modern symbolic logic, not long after the publication of Russell and Whitehead’s *Principia Mathematica*. The method was purely syntactic. Systems of axioms and rules were rigorously formulated, but the semantic interpretation was entirely informal.

Here are the two systems S1 and S2 formulated by Lewis. The primitive symbols are propositional variables ‘\( p \)’, ‘\( q \)’, ‘\( r \)’, etc., parentheses, the three connectives ‘\( \neg \)’ for negation, ‘\( \land \)’ for conjunction (Lewis used a dot), and ‘\( \Diamond \)’ for possibility. The axioms use a defined connective, ‘\( \Rightarrow \)’, for strict implication, which has the following definition: \( (\varphi \Rightarrow \psi) =_{df} \neg \Diamond (\varphi \land \neg \psi) \). (Lewis used a hook symbol).

The system S1
- Axioms
  1. \( (p \land q) \Rightarrow (q \land p) \)
  2. \( (p \land q) \Rightarrow p \)
  3. \( p \Rightarrow (p \land p) \)
  4. \( ((p \land q) \land r) \Rightarrow (p \land (q \land r)) \)
  5. \( p \Rightarrow \neg \neg p \)
  6. \( ((p \Rightarrow q) \land (q \Rightarrow r)) \Rightarrow (p \Rightarrow r) \)
  7. \( (p \land (p \Rightarrow q)) \Rightarrow q \)
- Rules
  1. Substitution of provably necessary equivalents
  2. Uniform substitution of sentences for sentence letters
3. Adjunction (if |- \( \varphi \) and |- \( \psi \), then |- (\( \varphi \land \psi \))
4. Strict detachment (if |- \( \varphi \) and |- (\( \varphi \Rightarrow \psi \)), then |- \( \psi \))

S2 has all of the axioms and rules of S1, plus one additional axiom.
8. \((p \Rightarrow \neg p) \Rightarrow (p \Rightarrow q)\)

Here is an alternative and equivalent formulation of S2.
- Axiom schemata
  - A1. All tautologous sentences
  - A2. \( \Box \varphi \supset \varphi \)
  - A3. \( \Box(\varphi \supset \psi) \supset (\Box \varphi \supset \Box \psi) \)
- Rules
  - R1. If |- (\( \varphi \supset \psi \)) and |- \( \varphi \), then |- \( \psi \)
  - R2. If \( \varphi \) is an axiom, then |- \( \Box \varphi \)
  - R3. If |- \( \Box(\varphi \supset \psi) \), then |- \( \Box(\Box \varphi \supset \Box \psi) \)