Scio Lease Manager

The "Swing Space"

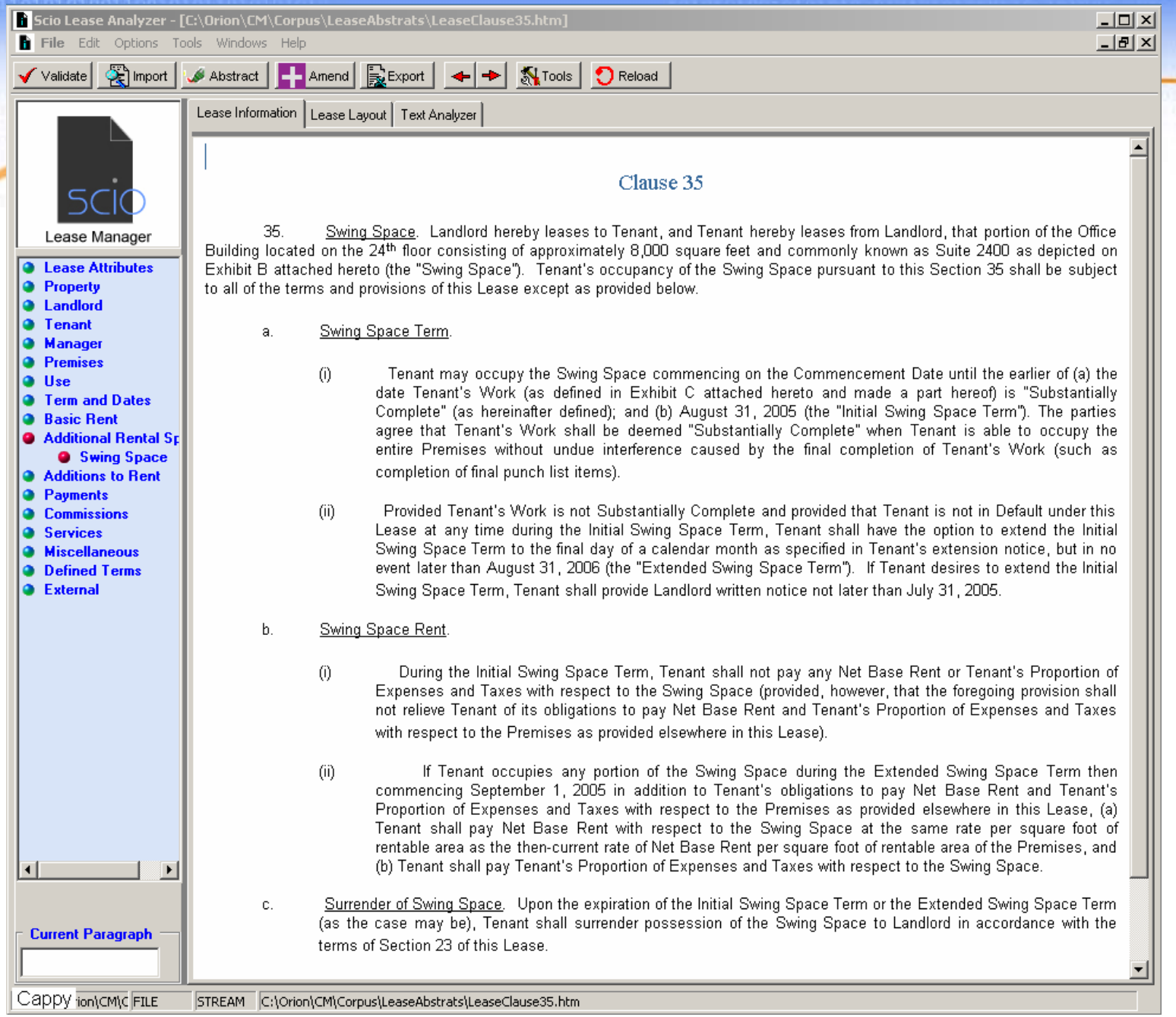
Understanding the meaning of a complex clause

Here is the original text as displayed in Scio's Lease Manager.

It is a Swing Space clause taken from an office lease provided by one of Scio's clients.

After clicking Import, the system automatically deeply analyzes the text, utilizing grammatical and semantic knowledge bases, as well as external lexical resources, and generates persistent active structures that fully represent the text's meaning in the business sense.

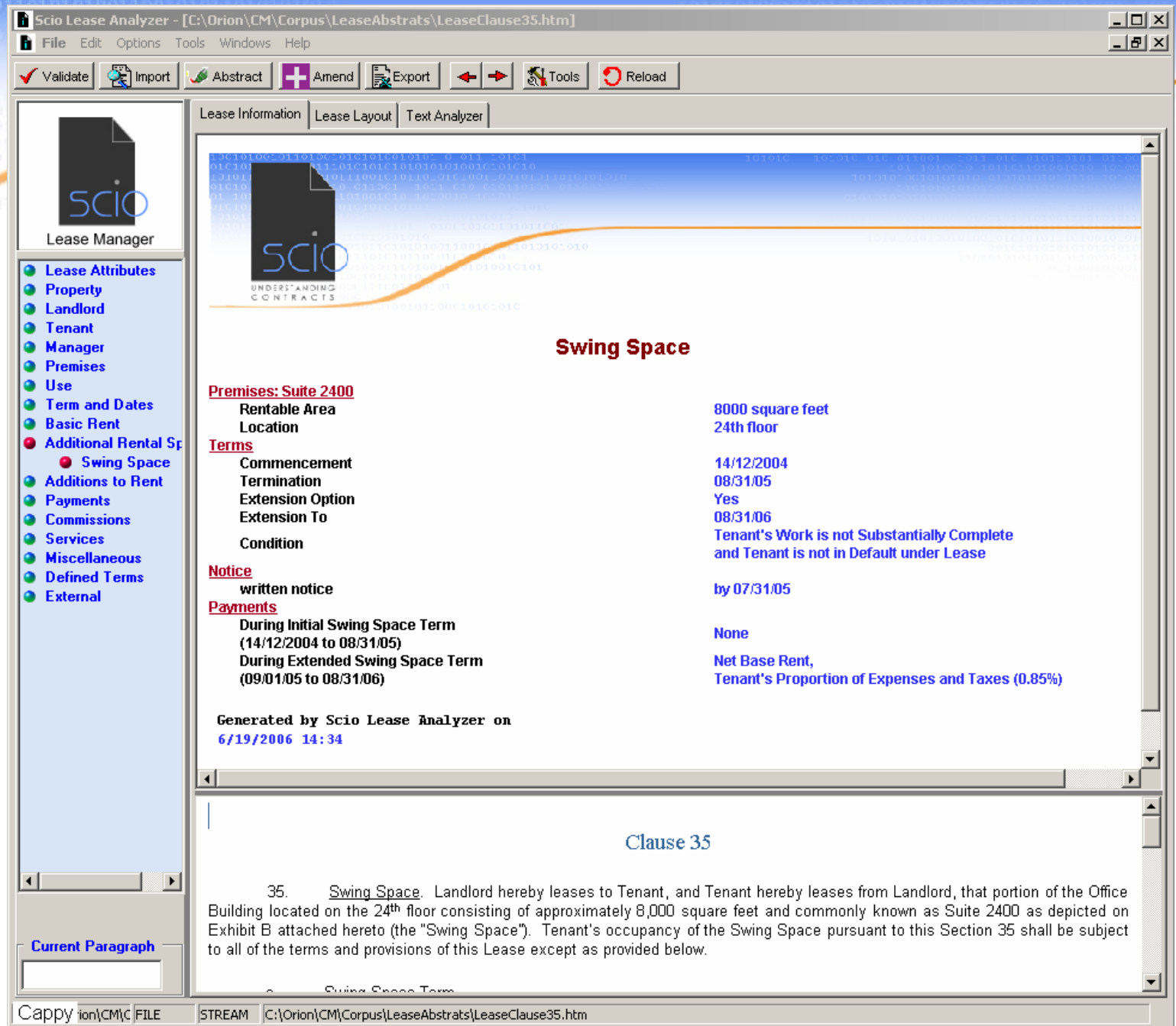
After the lease is imported, clicking Abstract will generate a Lease Abstract in the form and format used by Commercial Real Estate Management Services companies – please see next slide.



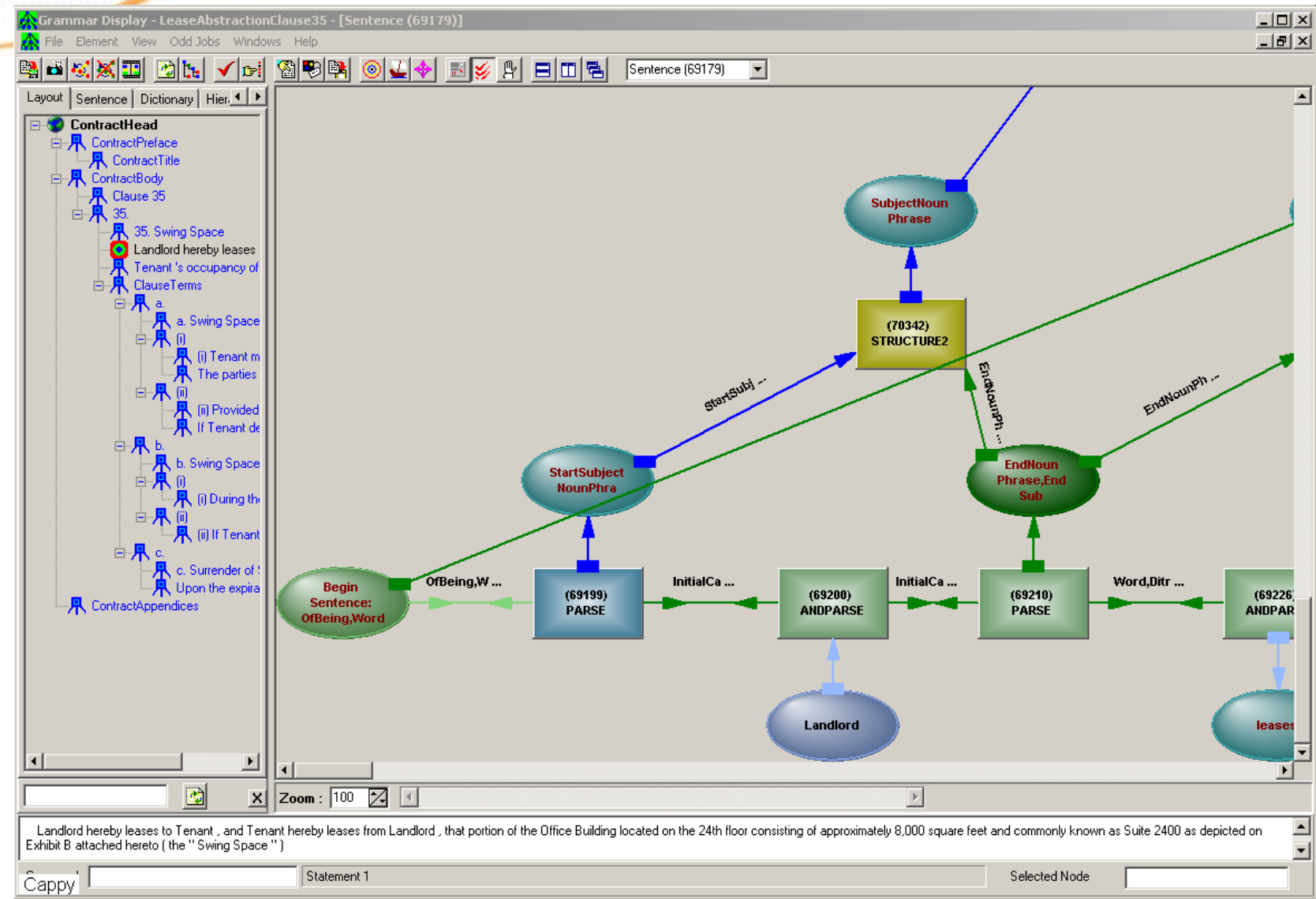
This is the Lease Abstract generated for the Swing Space clause. It is also generated as a separate file.

In this presentation we will go over some of the active structures generated for the clause.

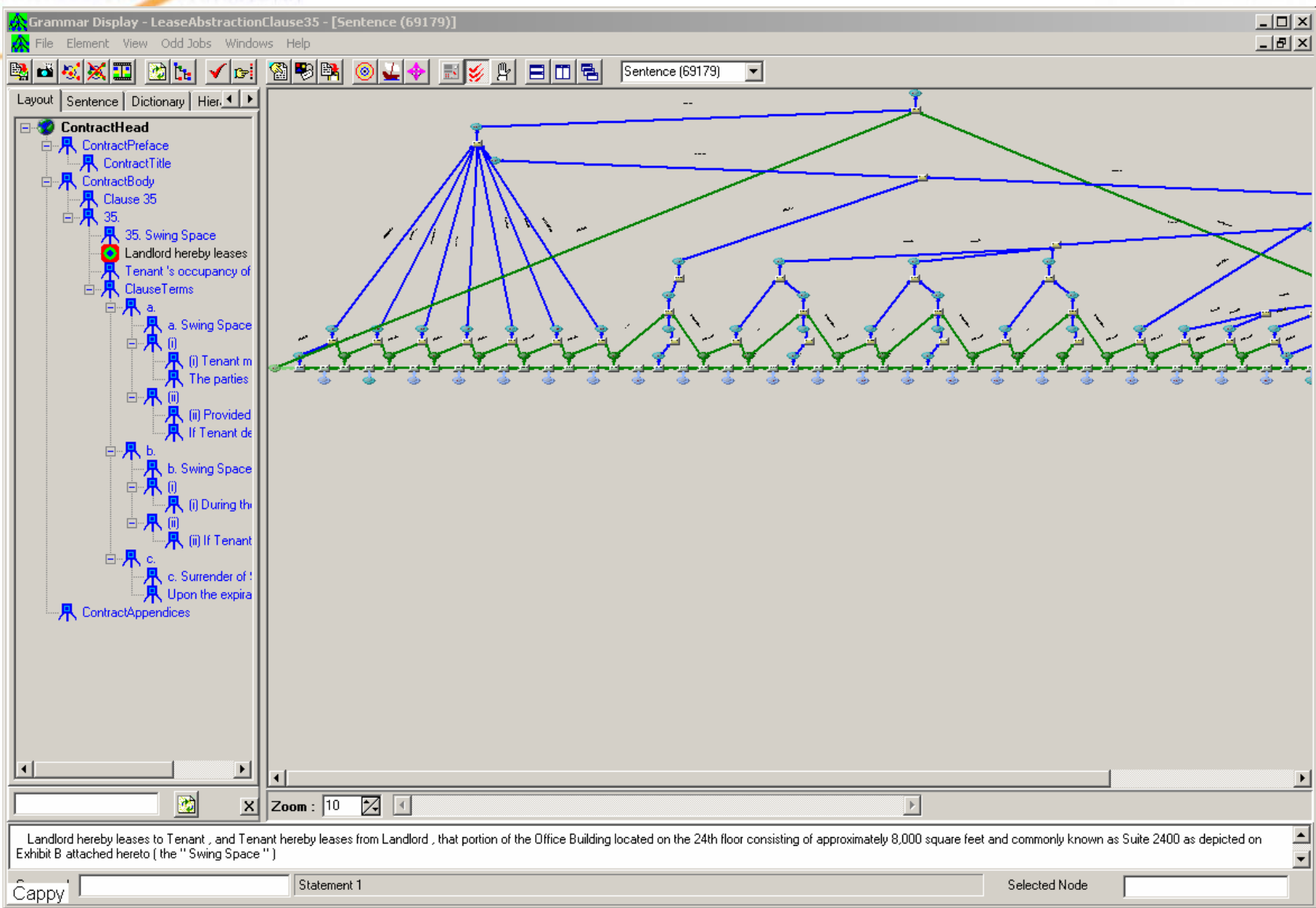
Trying to illustrate multi dimensional logical space on a limited two dimensional monitor, we will show only some of the main structures that were created.



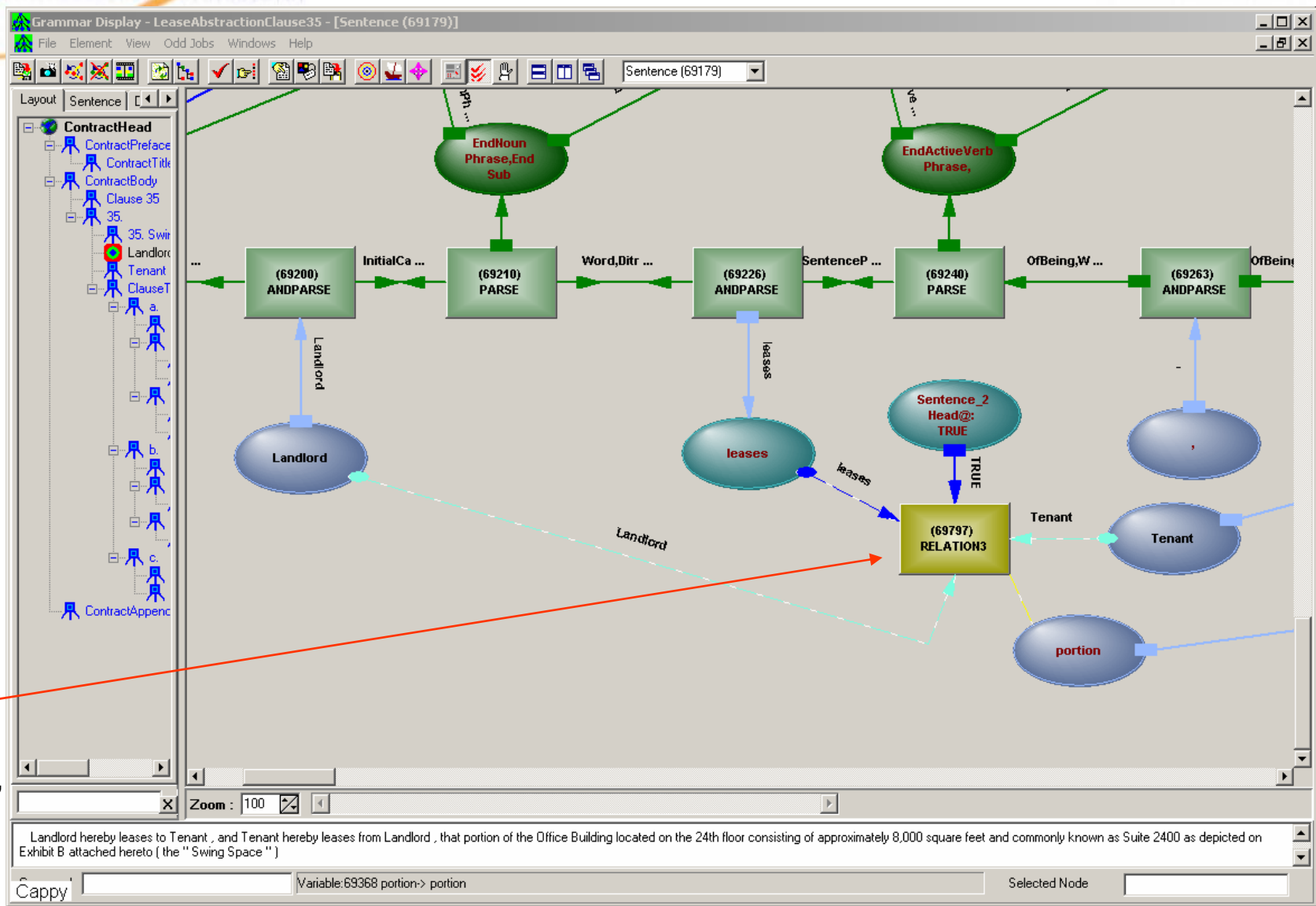
Here is the grammatical structure created for the first sentence, as shown in the Grammar Display tool. On the left, we can see the contextual hierarchy under which the structures are created. The hierarchy is used both for references from other parts of the contract and for providing context for new terms defined in the text. The base of the tree are the tokens, on top of which the system builds a multi level tree with increasing level of abstraction entities.



Here is an overview of the tree – the previous slide showed its lower left part.



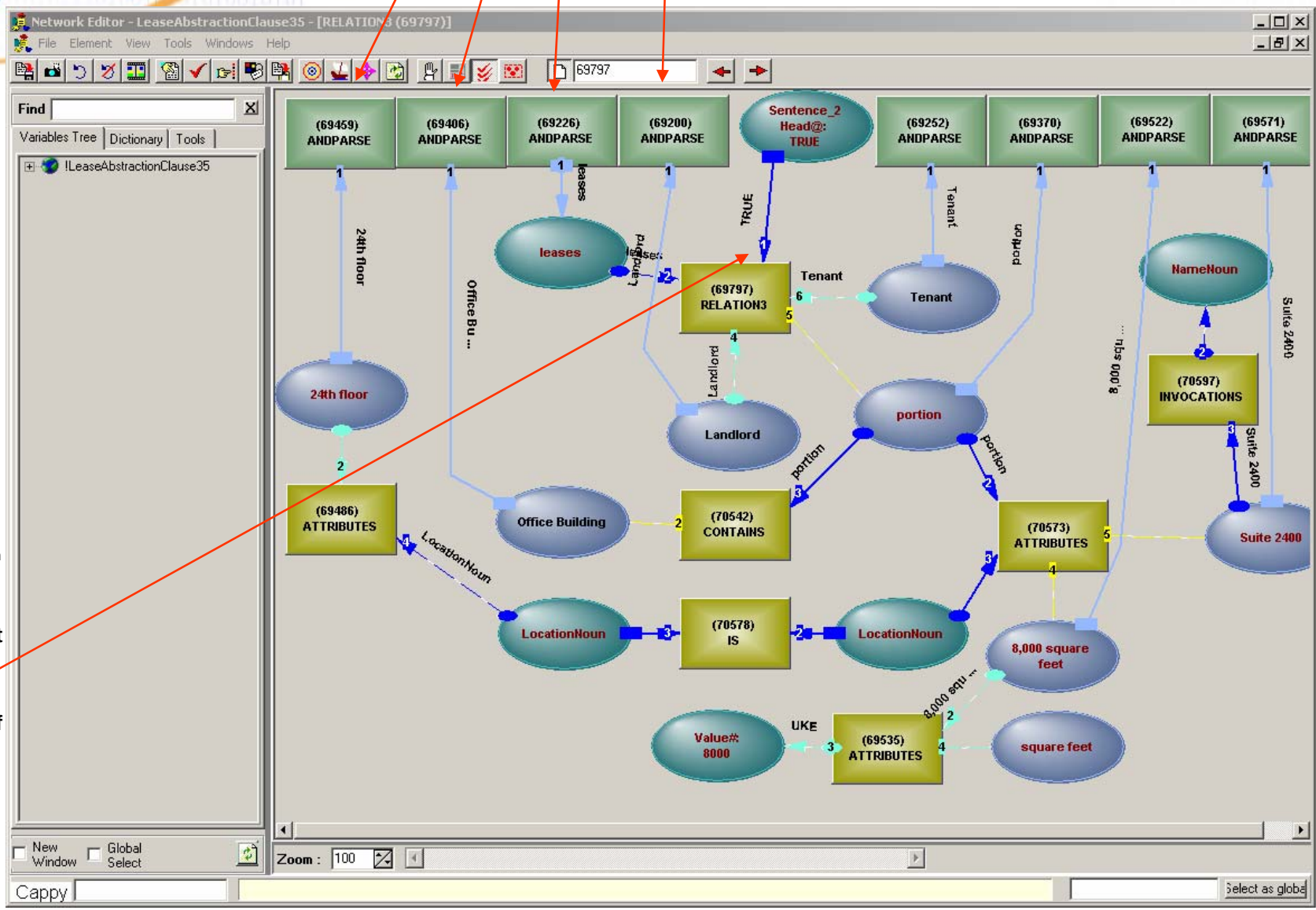
Here is a closer look. Throughout processing, and in the resulting structure, grammar is combined with semantics – the same word can be a noun, or a verb, but also an object, an active agent, an action, etc. In this example, we can see how the first 2 tokens, although part of the tree, are also linked semantically. The structure represents the ditransitive sense of "leases" in a RELATION3 operator (representing actions or states), with Landlord as subject, portion as object, Tenant as second object.



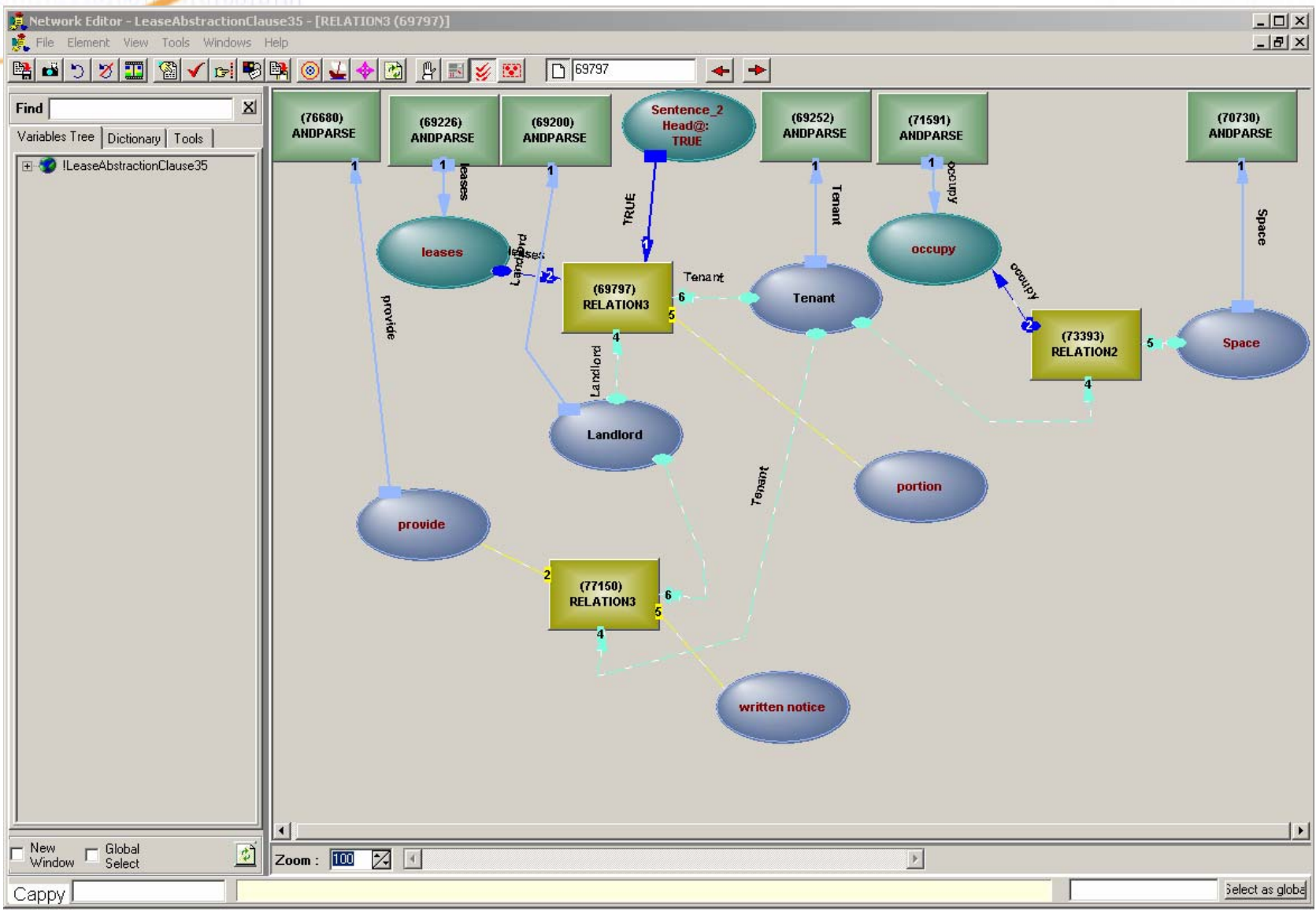
The objects linked to ANDPARSE operators are original tokens.

Here is the main structure created for the first sentence (Blue text indicates a reference to structures). A ditransitive relation meaning "Landlord leases portion to Tenant". Portion is contained in Office Building, its Location attribute is the Location of the 24th floor (meaning it is located in the 24th floor), its area attribute is 8000 square feet, and it has Suite 2400 as a name attribute.

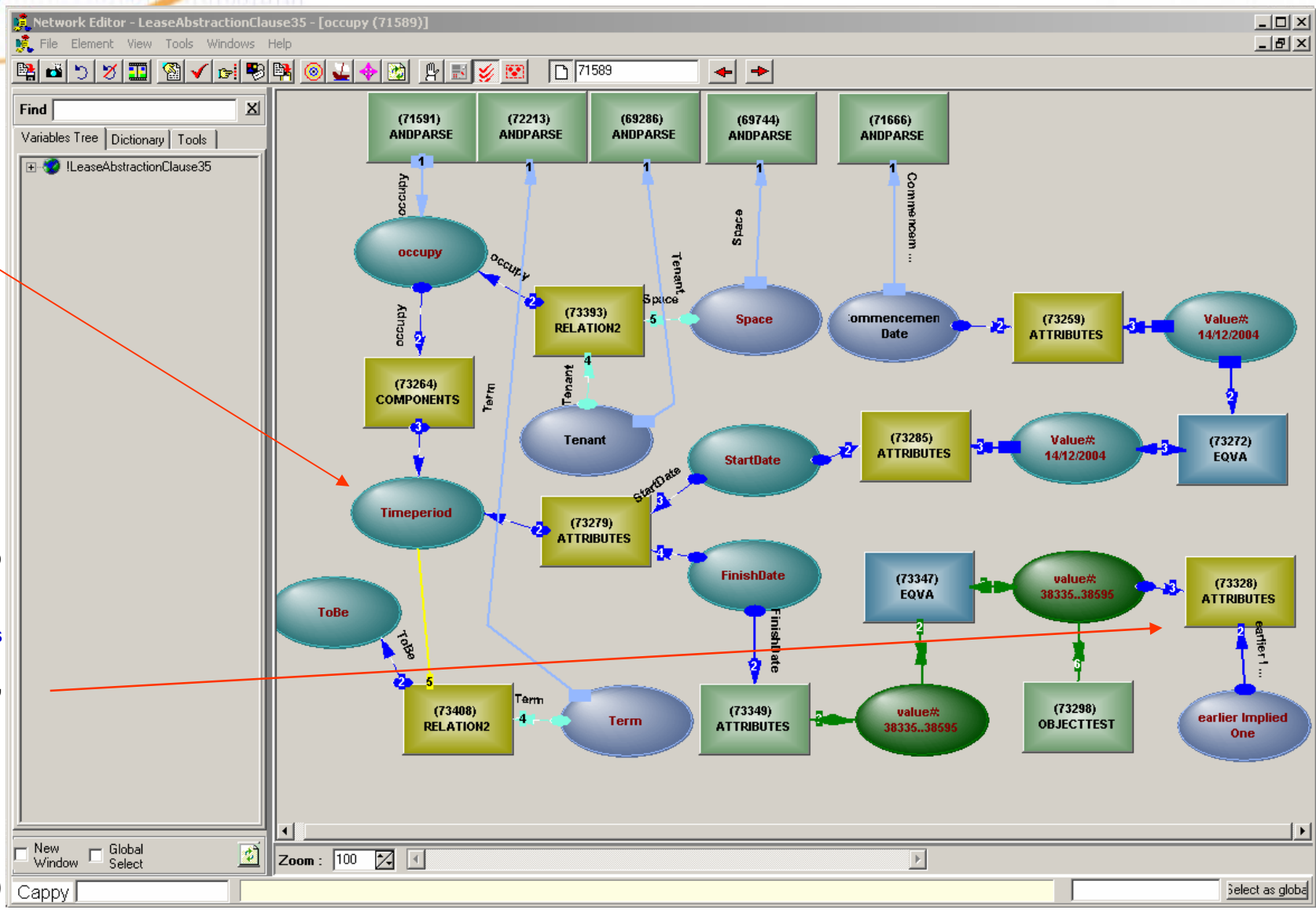
In general, pin1 of a relation controls its validity. The "leases" relation is controlled by the validity of the sentence.



Note that some objects are shared among other sentences, clauses, or contracts. Here are some structures attached to Tenant and Landlord, generated for other sentences.

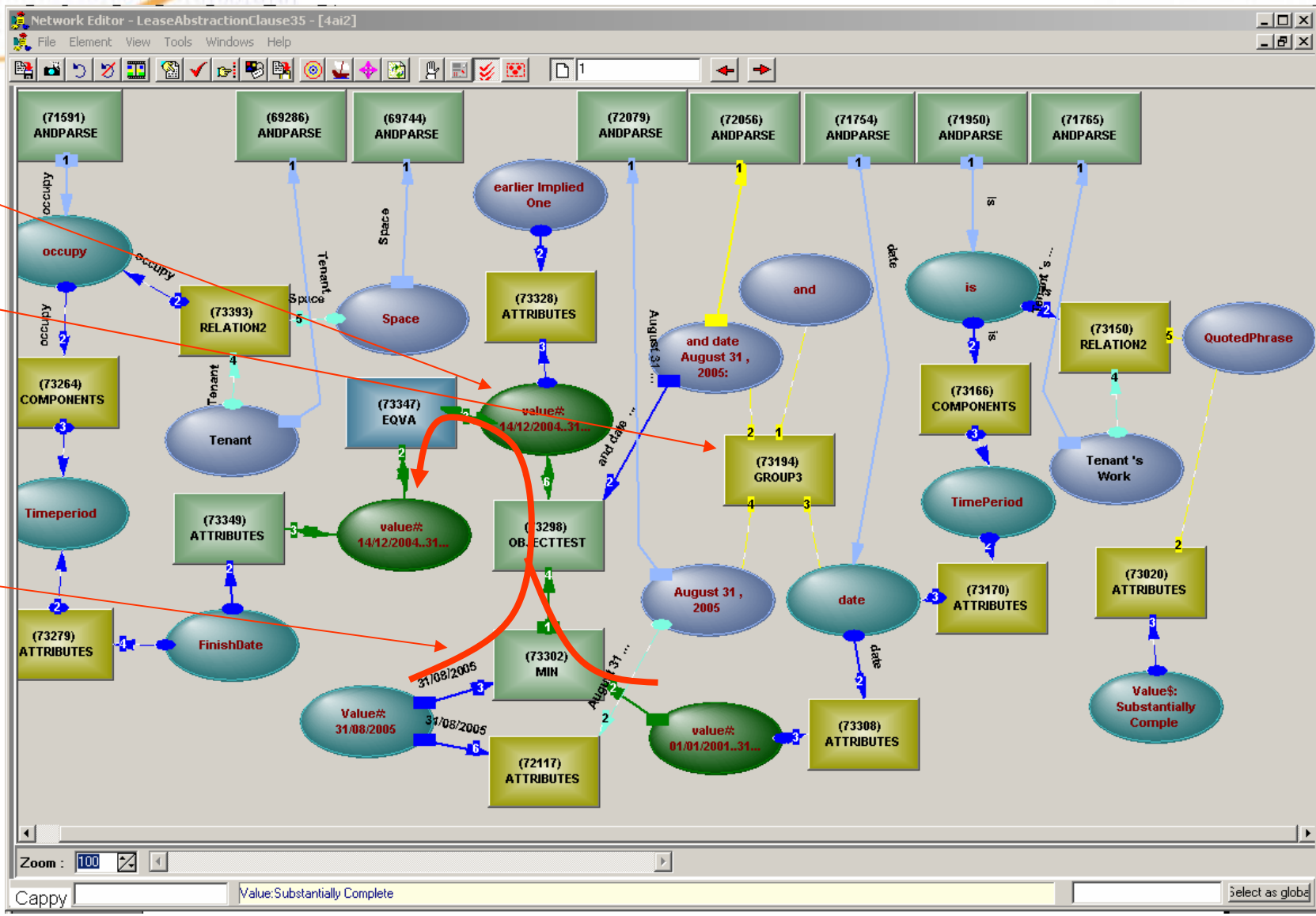


Here is the structure for section a(i). A **relation** – Tenant occupy space – which has a **TimePeriod component** (every relation has a TimePeriod comprising StartDate, FinishDate and Duration, which are mutually synchronized automatically). Its **StartDate** is the **StartDate** of CommencementDate (14/12/2004), and its **FinishDate equals** that of EarlerImpliedOne, an implied object, created by the system when processing "…the earlier of…" (becoming "… the earlier ImpliedOne of…")

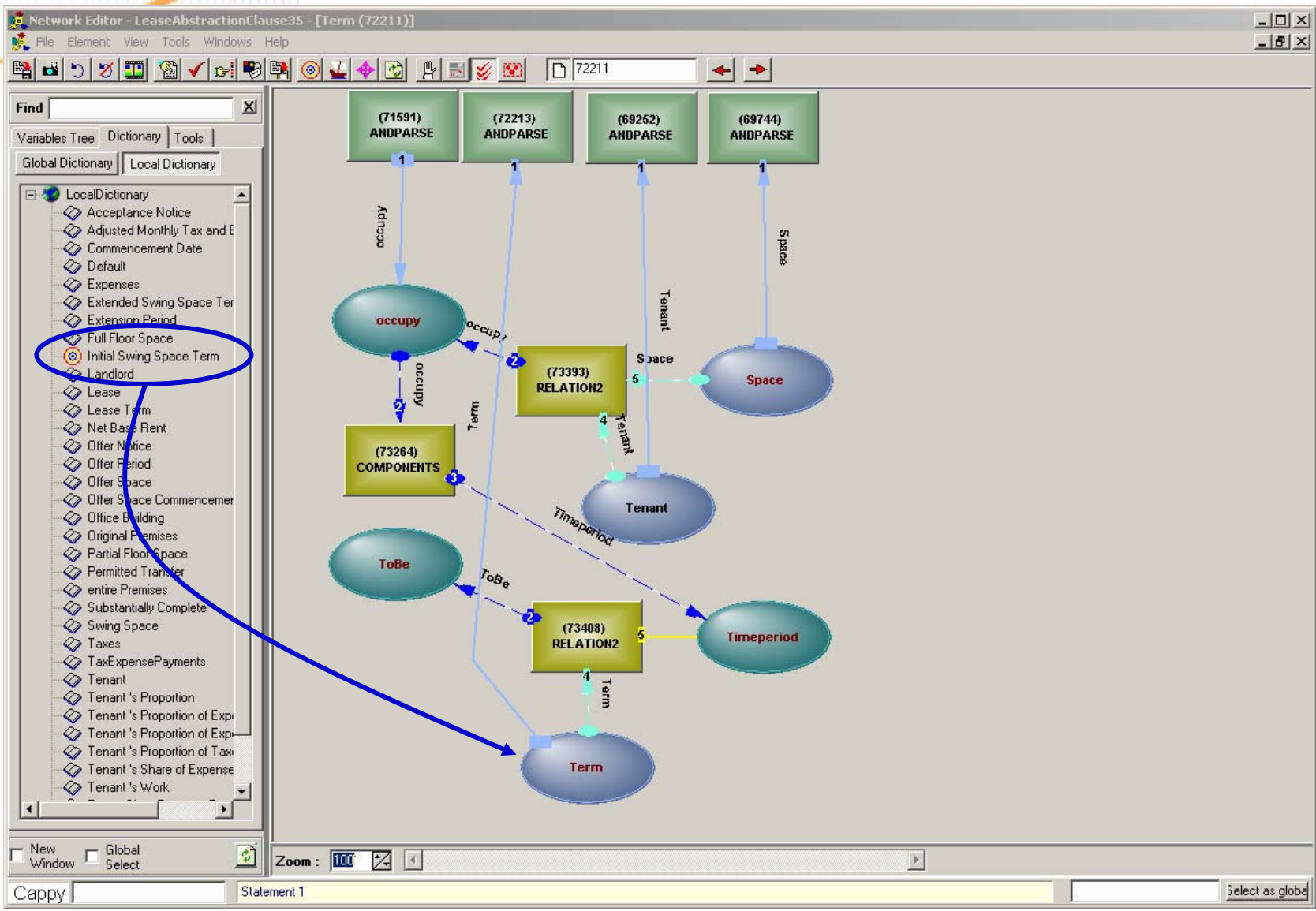


SCIO
UNDERSTANDING CONTRACTS

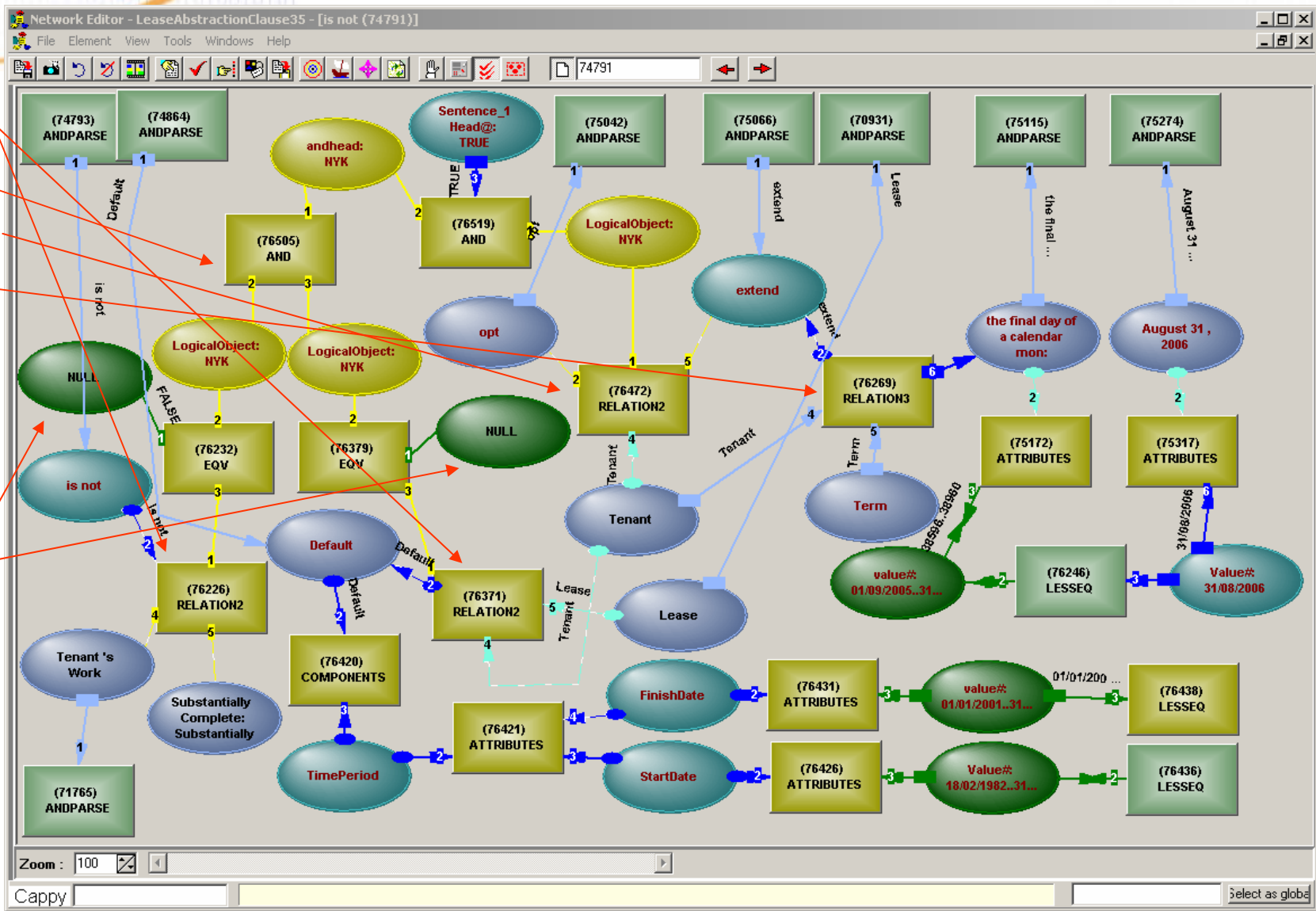
A closer look at the finishDate of the Occupy relation. The date value of the ImpliedOne is linked both to an "and" objectgroup comprising August 31, 2005, and the date which is the StartDate of the IS relation of "Tenant's Work *is* Substantially complete". The OBJECTTEST operator created a MIN operator linking the three date values. Note that since the date is not known yet, its value is a range (green), which propagates all the way to the occupy relation.



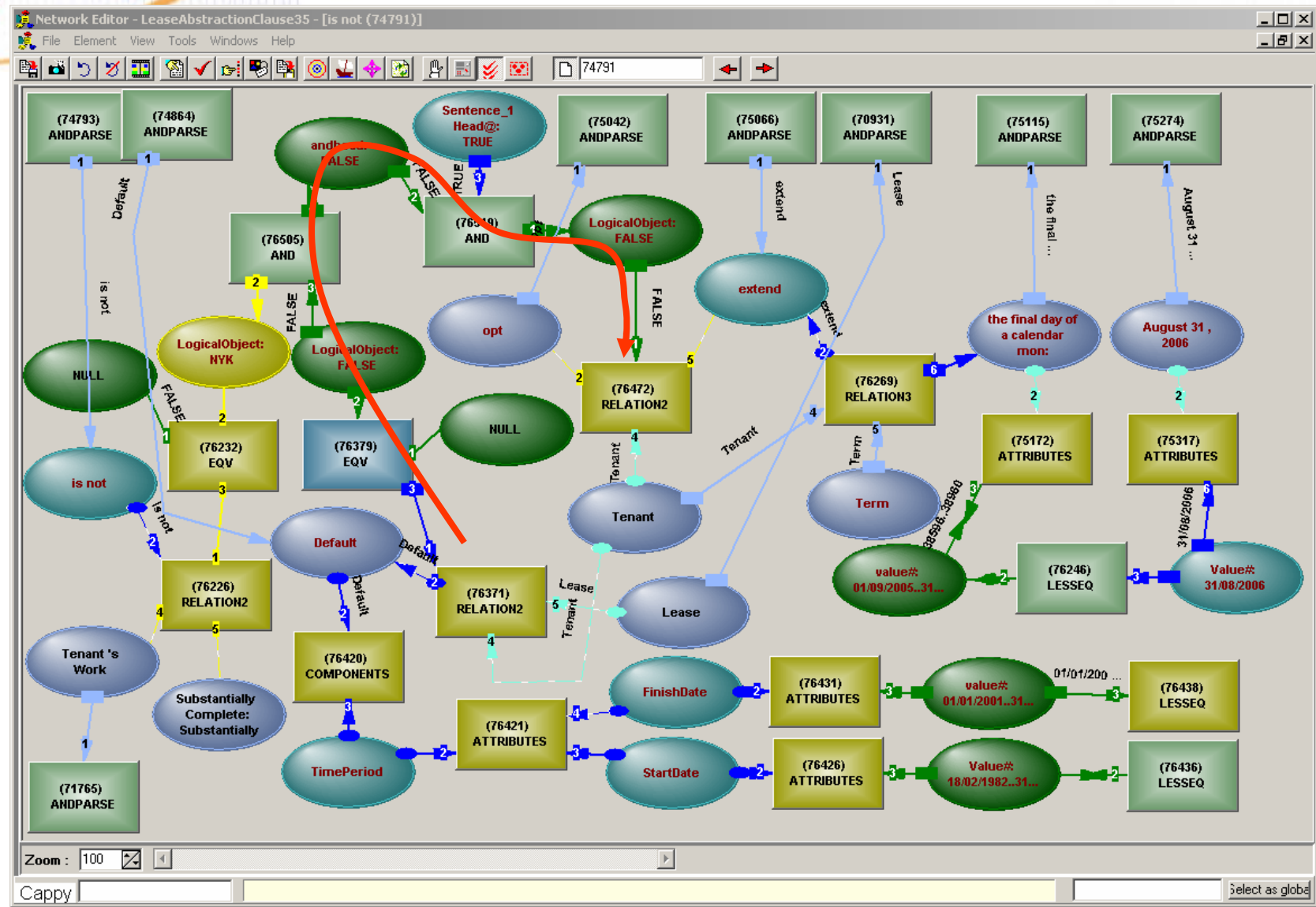
The system maintains a real time, multi-level dictionary for newly defined terms. The local dictionary terms are shown listed on the left. Initial Swing Space Term is defined in a(i) and is used several times later in the clause. The system creates an entry in the local dictionary, pointing to the Term token, linked through a **ToBe** (is) **relation** to the **TimePeriod** of occupy, which is the actual meaning of the newly defined term. Further references to Initial Swing Space Term will refer to the **TimePeriod** of Occupy.



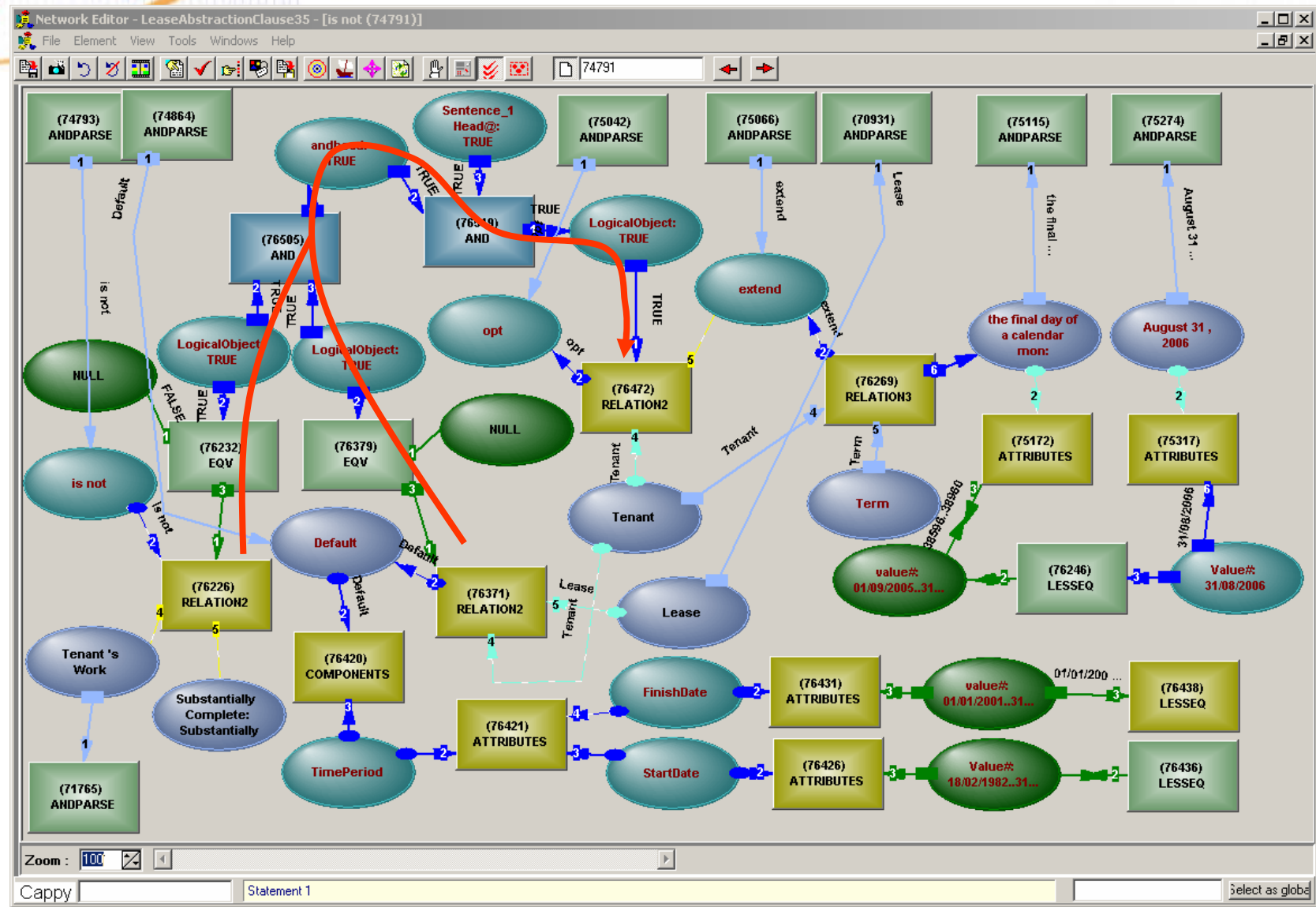
Section a(ii). Two relations, is not and Default, which logical control is AND'ed and controls an opt relation which object is an extend relation. The object of the extend is Term which is the Initial Swing Space Term (described in the previous slide). Note that the not's are represented by linking a False to the relation's logical control. This logical control structure implements IF..THEN. As the system is inherently simulative (What-If), next, we'll show how this works.



Say the **Default relation** is True (Tenant *is* in Default under the Lease). A **True** flowing from the relation is turned into **False, which** flows through the **AND**, all the way to the **opt relation**, meaning Tenant now does not have the option to extend the Initial Swing Space Term. (A closer look at the dates follows).

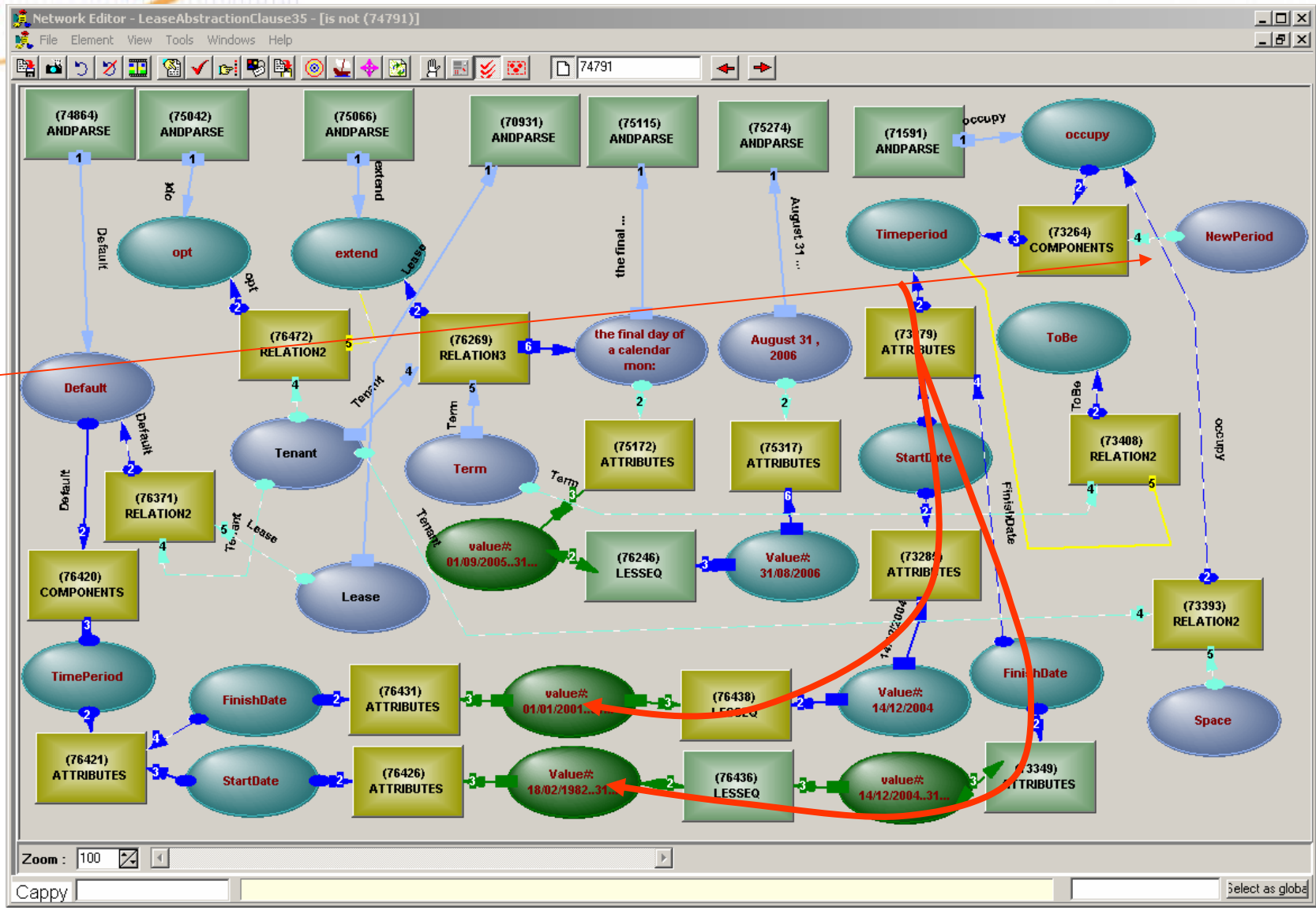


Say Tenant is not in Default, and Tenant's Work is not Substantially Complete. A False flows out of both relations' logical controls, turned by the logic into a True which flows into the opt relation, meaning Tenant does have the option to extend the Initial Swing Space Term.



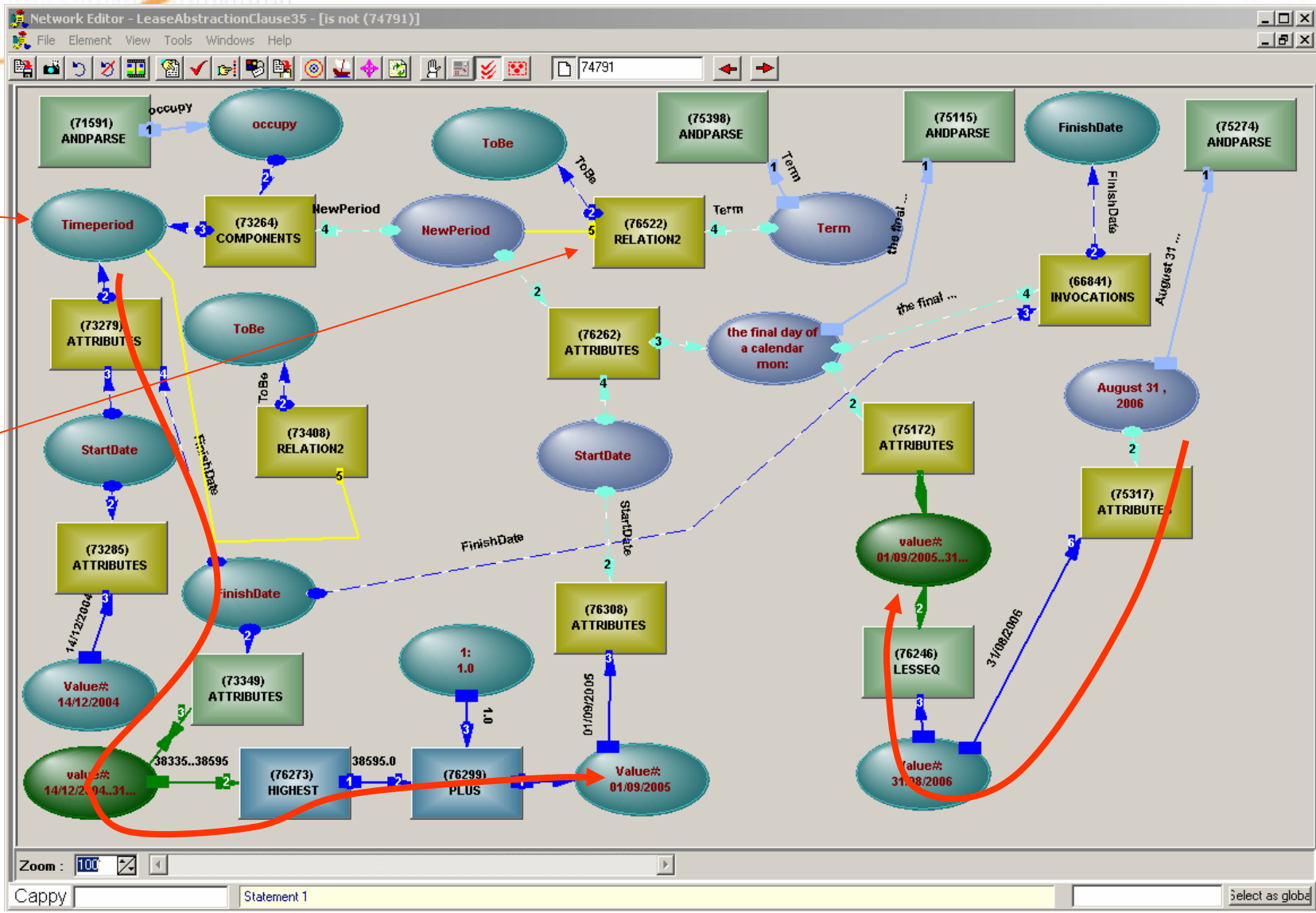
A closer look at the dates. The TimePeriod of the Default relation gets its values from that of the Initial Swing Space Term (the TimePeriod of the Occupy relation).

The system creates a new TimePeriod, with Existence state of MayExist (as it is still optional), unlike the original TimePeriod, which has Existence state of DoesExist (the Initial Swing Space Term). The new TimePeriod is linked to a new local dictionary defined term – Extended Swing Space Term. More in the next slide.

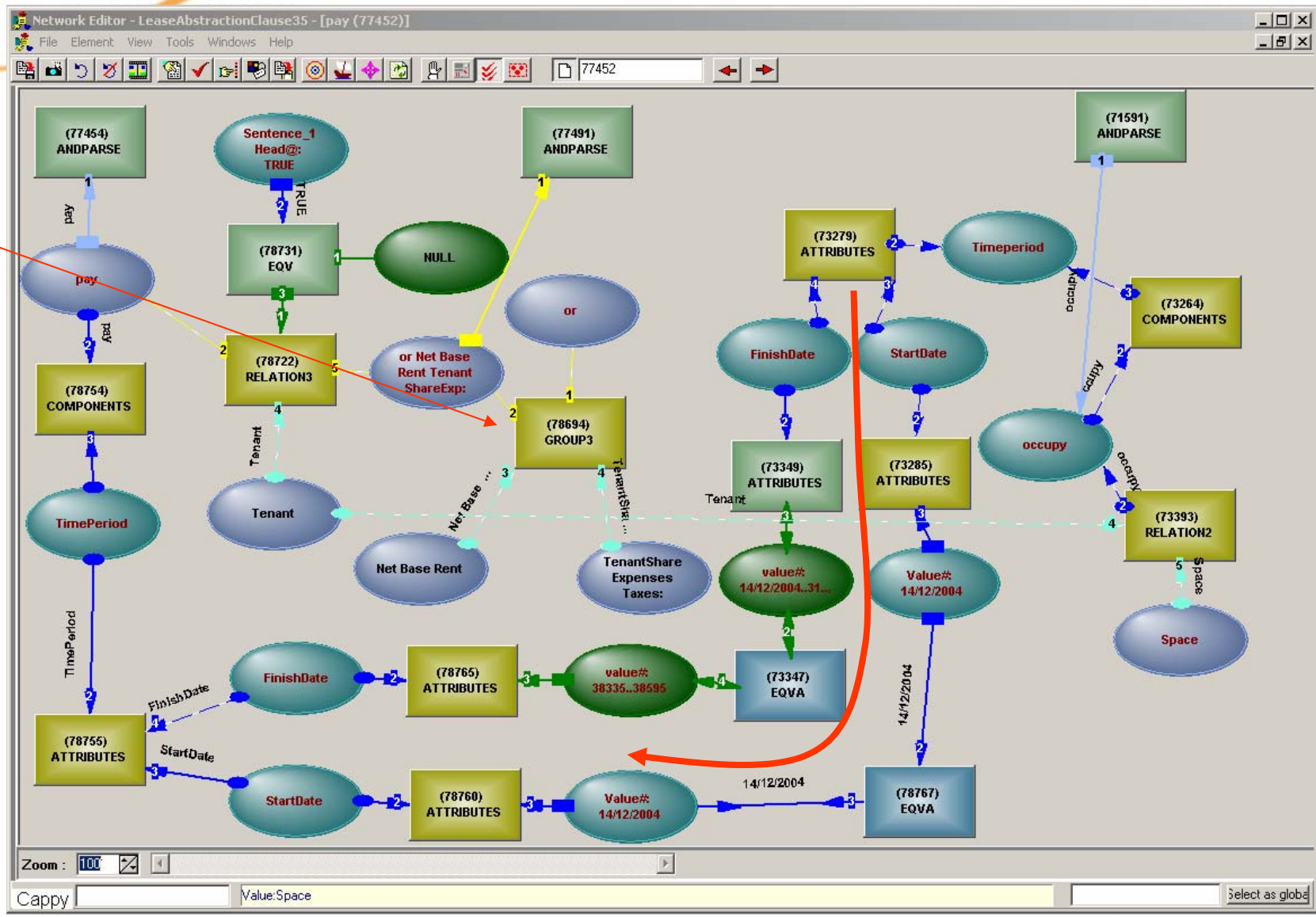


The new **period**, the **Extended Swing Space Period**, starts a day after the end of the **Initial Swing Space Term**, and ends at a **final day of a month** which is no later than **August 31, 2006**.

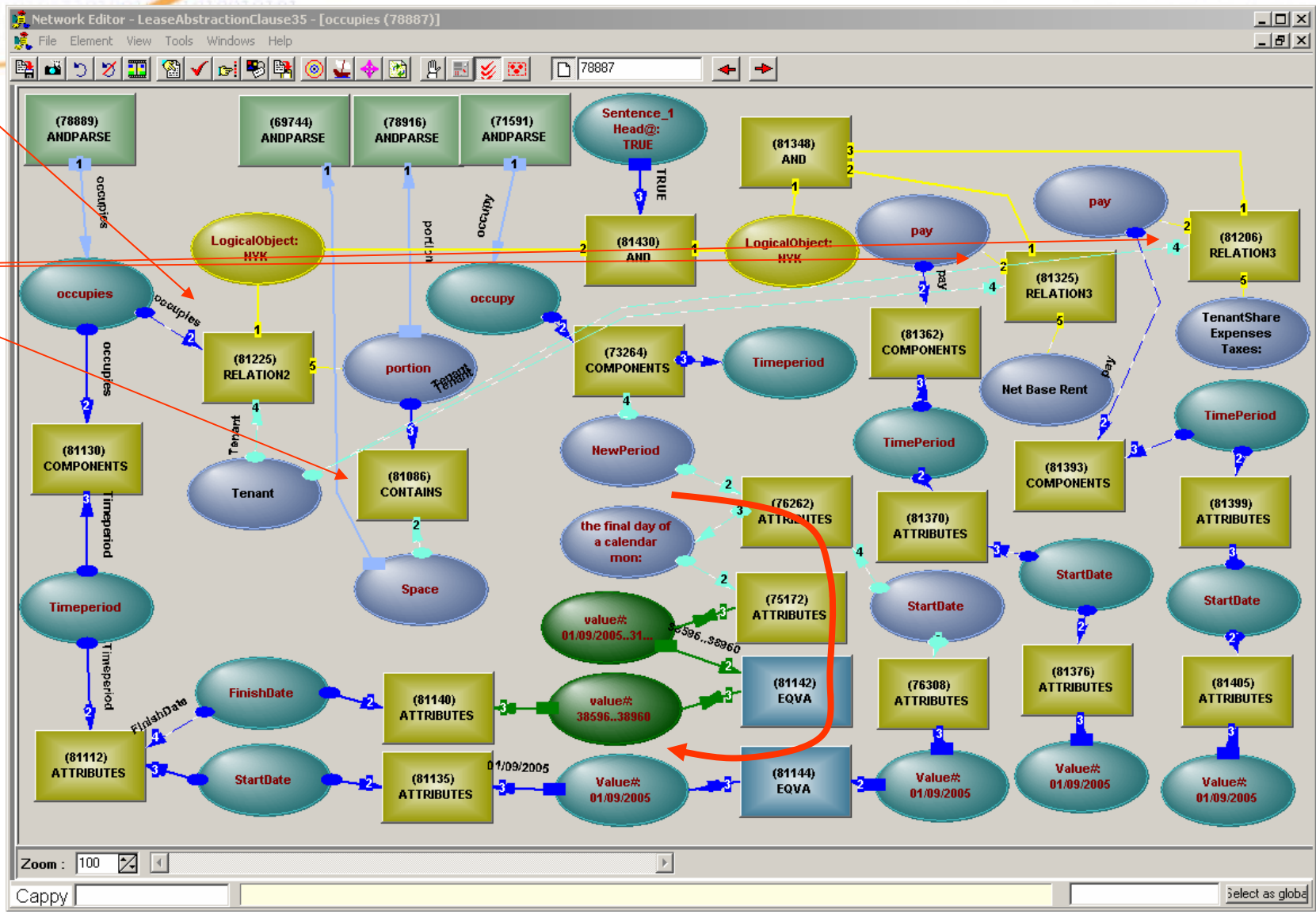
The new defined term is linked through a **ToBe** (is) relation to the new **Term** token that represents it in the local dictionary.



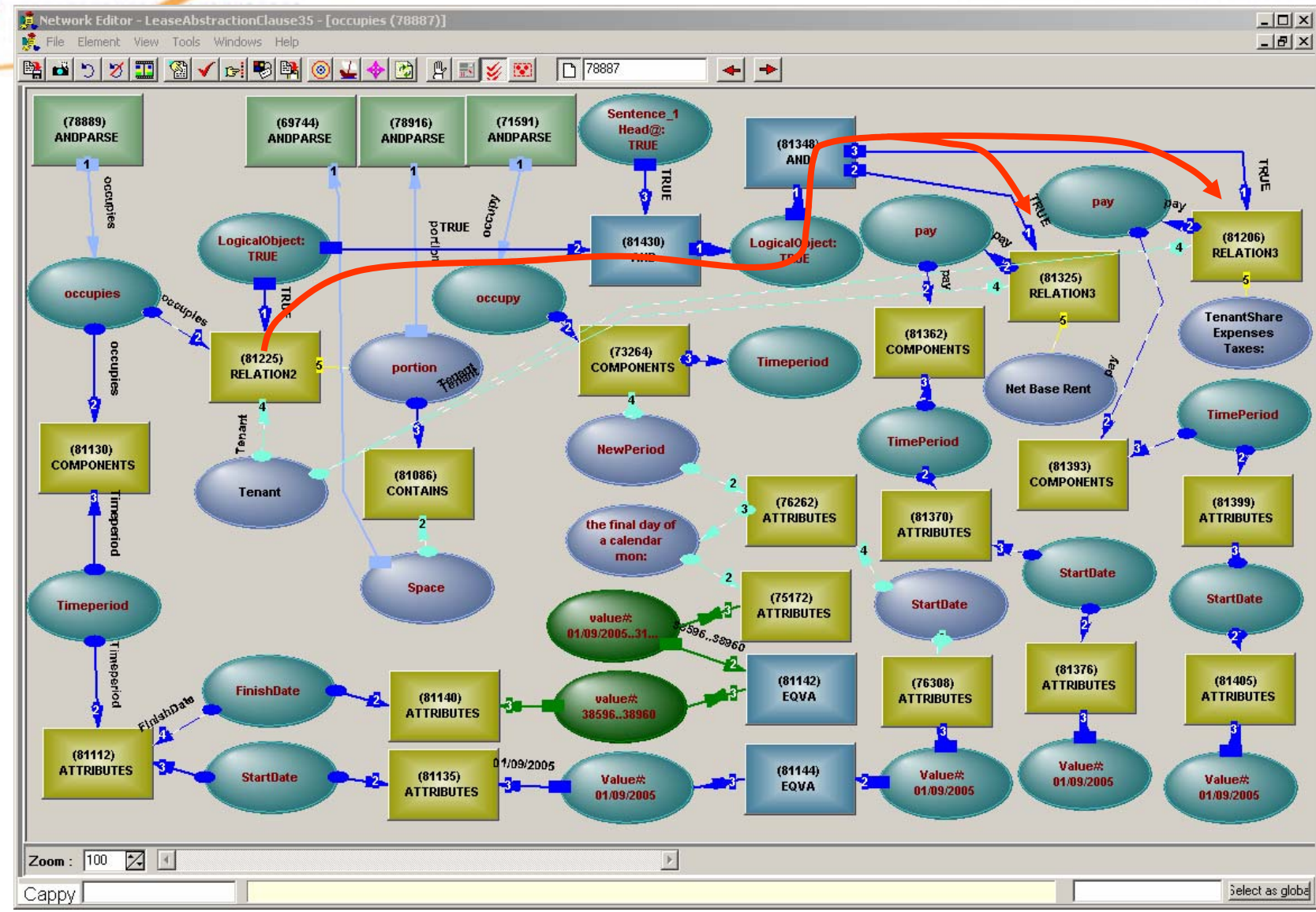
Section b(i). A pay relation with a negated control (meaning not pay), with Tenant as a subject, and an OR objectgroup, comprising the Rent and Taxes objects, as an object. The pay relation gets its TimePeriod from the Initial Swing Space Term (During the Initial Swing Space Term, Tenant shall …).



Section b(ii). An occupies relation, with Tenant as a subject and portion as an object, and a logical control structure controlling 2 pay relations. Portion is part (contained in) Space, which is the local dictionary object for Swing Space. The occupy relation gets its TimePeriod from the Extended Swing Space Term – the new TimePeriod of the occupy relation. The StartDate of both pay relations is 01/09/2005.



Finally, say the occupies relation becomes true (Tenant *does* occupy a portion of the Swing Space during the Extended Swing Space Term). A True flows to both pay relations, meaning Tenant pays both Rent and Taxes commencing 01/09/2005.



The structures shown in previous slides were automatically built from this text.

Changes in the text result in changes to the structures, and changes in the lease abstract.

Reading of the text is faster than an attentive human reader, but here the structure is permanently captured, and can be used for

- **validity checking**
- **answering questions**
- **simulation**

over the life of the lease.

With many leases in a large building, each with its own captured structure, it becomes easy to ask questions like "Does anyone have right of first offer on Suite 2403"