# Is Mining of Knowledge Possible?

# Islands of Knowledge


Linear text has a problem with islands of knowledge - it may attempt a general sweep, then circle back around each one, may need to jump to the level of generality when a new phenomenon is to be described - we can’t assume an orderly linear structure

# Structure

What we can assume is that technical text paints a structure, which will eventually be filled in by the reader’s knowledge as well as by the writer - here a broad brush stroke, there a pointillist detail
The structure may be neither consistent nor coherent while it is incomplete - a strong guide to its completion

# Not Free Text

We have been concentrating on contracts

They are a long way from free text - they contain a great deal of structure, they are meant to be unambiguous, they should be consistent and coherent

But they are not meant to be easy

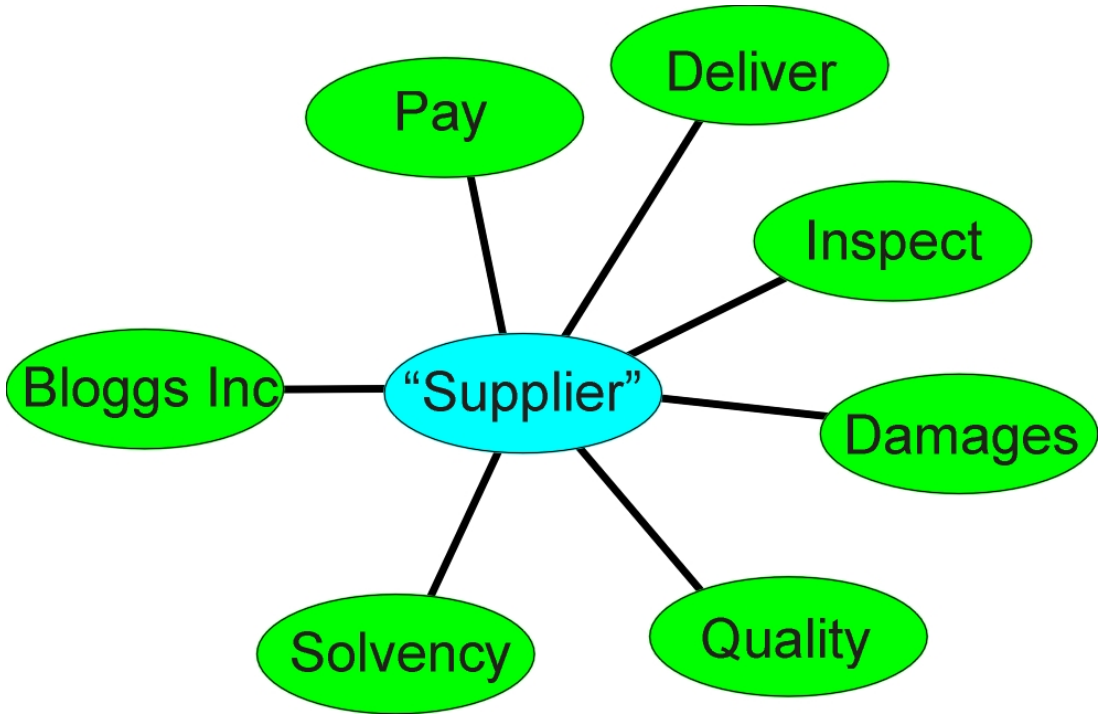
# Linear Text

Here are some problems with the text mining assumption that lack of adjacency implies lack of relevance:

- Defined Terms
- Structural References
- Anaphora
- Cutting Parentheticals
- Implied Symbols
- Embedded lists

These things make crude methods useless

# Defined Terms




There may be a thousand connections in the text to Supplier, and just one to Bloggs Inc., but it is rather important, because Bloggs Inc. *is* the Supplier

We need to read the definitions and capture their meaning

They aren’t ordered to suit - they are where we find them

# Structural References

"the options in Section 3.(a)(ii)"
"equation 21 on page 16"
"in what follows"

A technical document will make significant use of structural references - pointing to some part of the document (or another document) and expecting the reader to extract the meaning from the reference

It is no different to "mine the high grade ore in stope 45"

It means we need to keep a document structure map up to date and close by as we extract the meaning structure

# Anaphora

Technical text uses pronouns, and it is also a heavy user of things like

Said, So, Same & Such

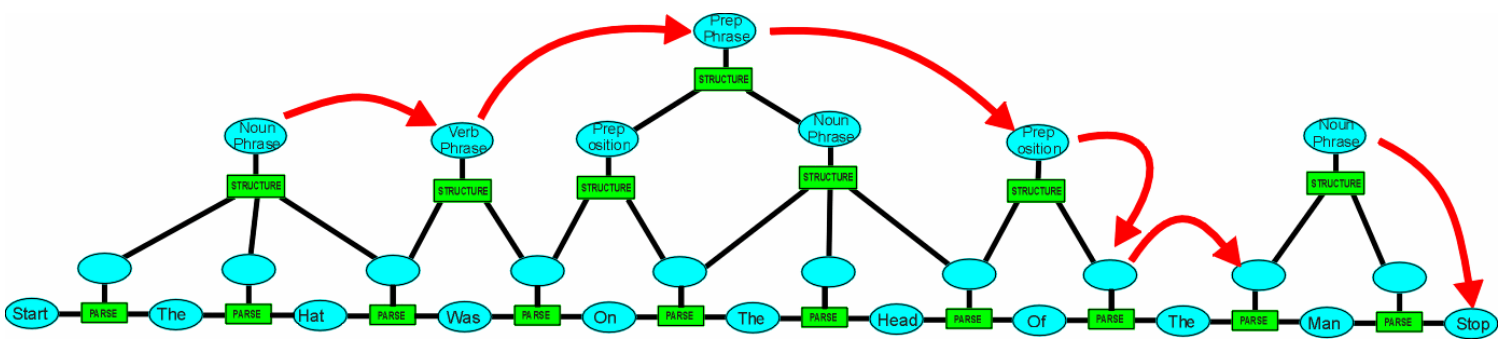
which can refer back (“Use of such methods is...”, “If the Supplier desires same, then...”)
or forward (“It is the same as...”)

Something has to make the connections - only then can the meaning be found

# Cutting and Healing

Parentheticals (like this one) require us to cut out structure, implied objects require us to insert objects, and then restart building in the local area

After surgery, the grammatical structure must continue to support basic operations on its topology - like finding the next symbol

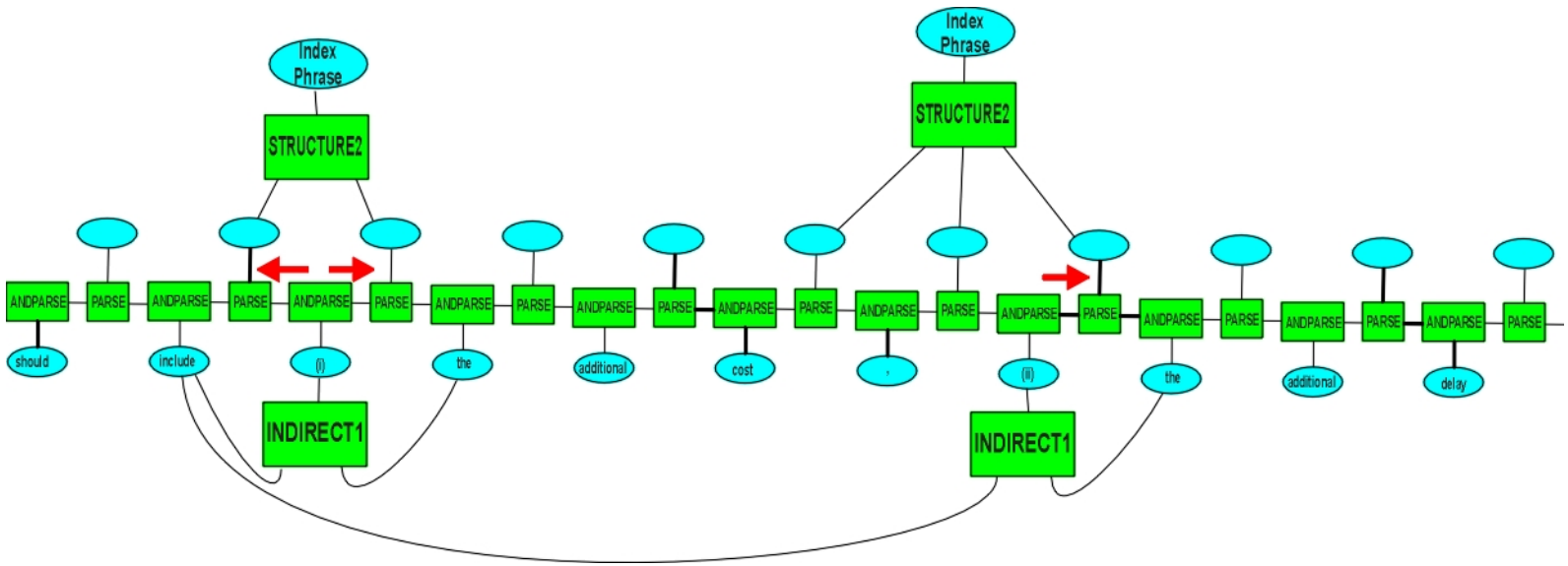


# Implied Symbols

Large documents will usually employ a very clipped syntax, to prevent them becoming even larger

# Embedded Lists

The report should include (i) the additional cost, (ii) the additional delay compared with the work schedule in the report prepared on 21st June and approved by the committee at its meeting on the 29th June, and (iii) any changes to the EIS.



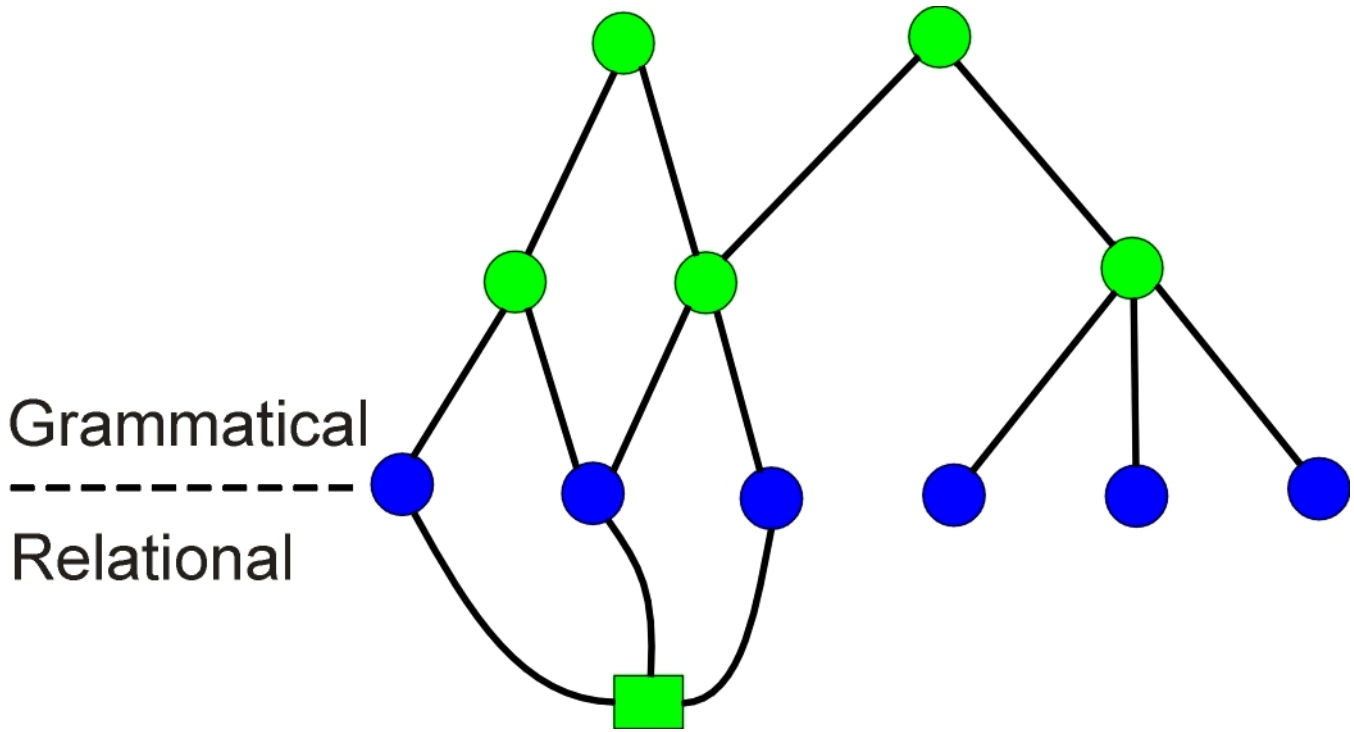

Adjacency may need to be synthesized - a word needs to see what its real neighbor is.

There is a head which can be far away, and there may be a tail

# Elements of the Solution

- Read every word and wring meaning from it
- Bind grammar and semantics synergistically
- Build a structure from the sentence
- Use all the structure, including the new structure, in reading the next sentence (sometimes the structure has to update the same sentence)
- Make the structure self-extensible

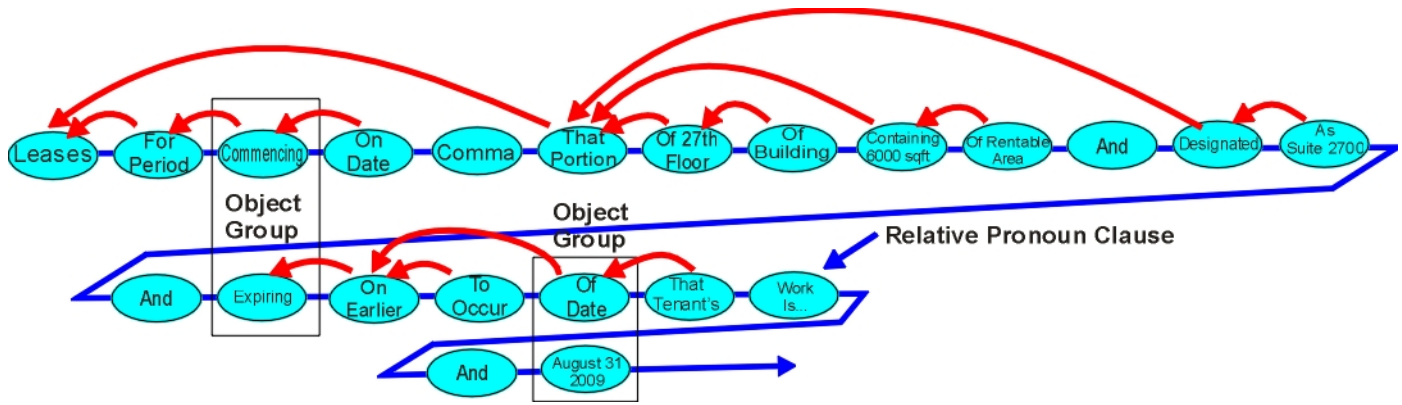
# Building Together




The two structures are built together - grammatical pattern structures can take advantage of the new properties that the objects acquire through relations being built as other grammatical patterns succeed in matching - we don’t want a string of symbols, we want a structure, as only a structure can represent the complexity of free text

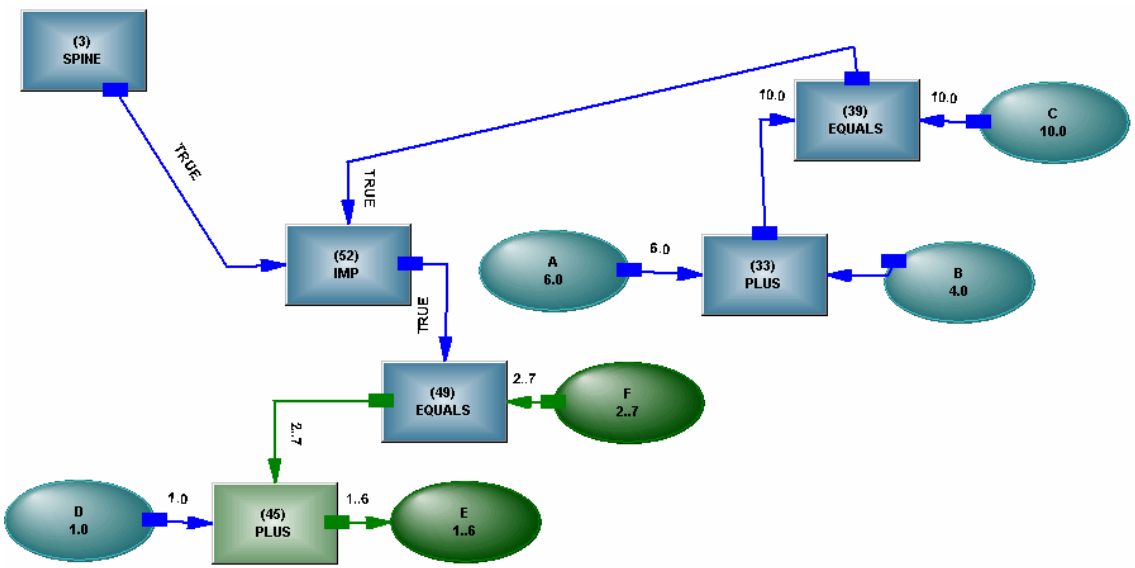
# Purity of Form

For anyone thinking that grammar and relations should be kept separate, we suggest you try to turn a long prepositional chain into its network form - you will need everything you have, all at once




There is a philosophy here - of using all the knowledge you have all the time - the alternative is wearing hobbles and failing with a pure heart

# Undirected Structure




Here is a simple implication. If we make it undirected and direct it dynamically, we can get every possible use out of it As the structure grows more complicated, this becomes more important

# Structure Building

A person reads a book and converts it into internal structure.
What they have read helps them to read further
- the structure builds on itself.

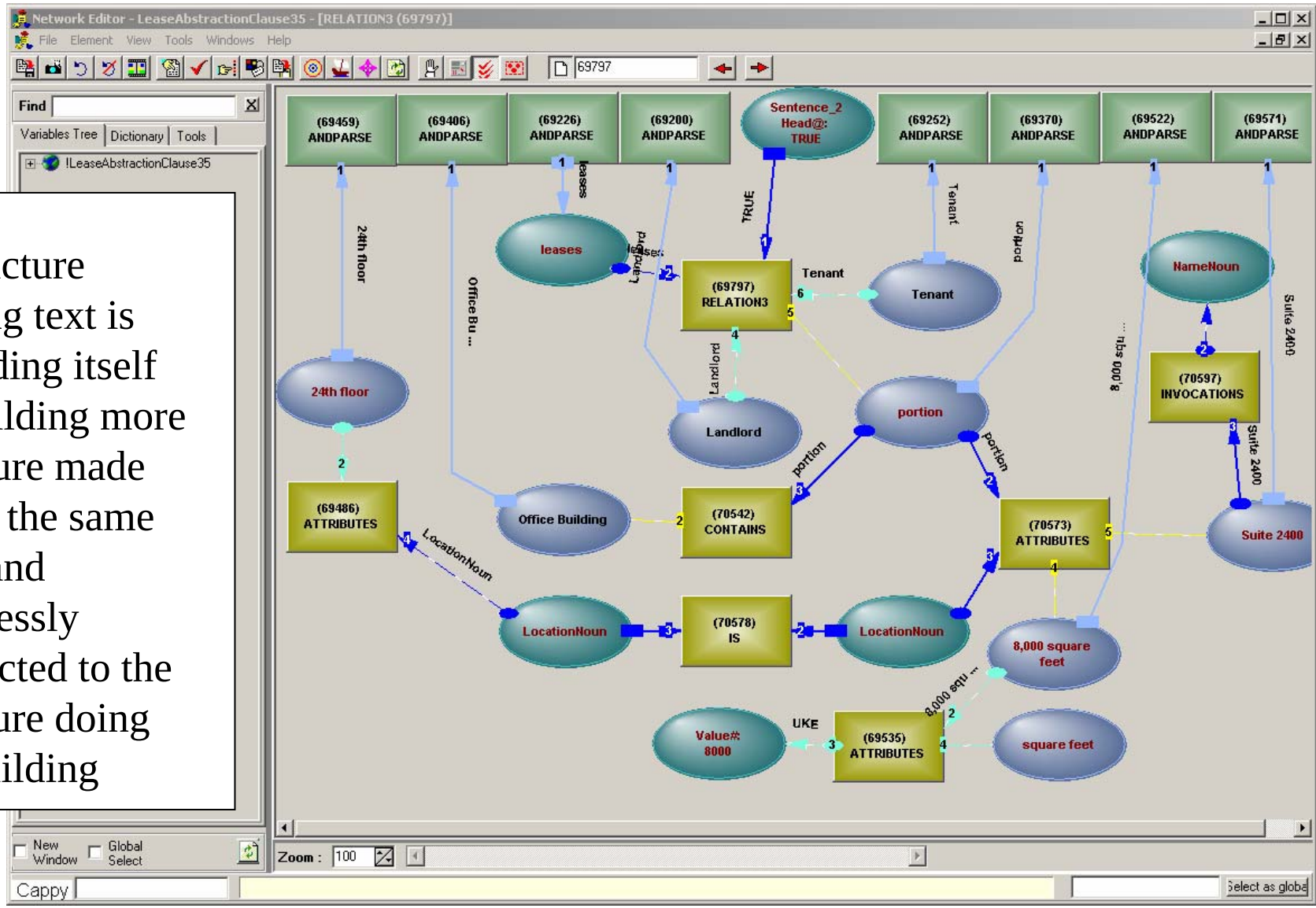
# Consistent and Coherent

A structure which can extend itself has to be made out of the same stuff everywhere - it can’t have a database, a stored procedure, a tree structure, some rules - it can’t be a mess of pottage, because that won’t extend.

The Active Structure concept is coherent down to its logical root.

# All the Same Stuff

A structure reading text is extending itself by building more structure made out of the same stuff and seamlessly connected to the structure doing the building



# Repairing Discontinuities

Humans are very good at repairing gaps and other faults in the cognitive structures they build.

We compare it to an octopus, moving to the site of a discontinuity, grasping the loose ends, tying a knot, then moving away. Humans have about the same range - a maximum of six to nine loose ends.

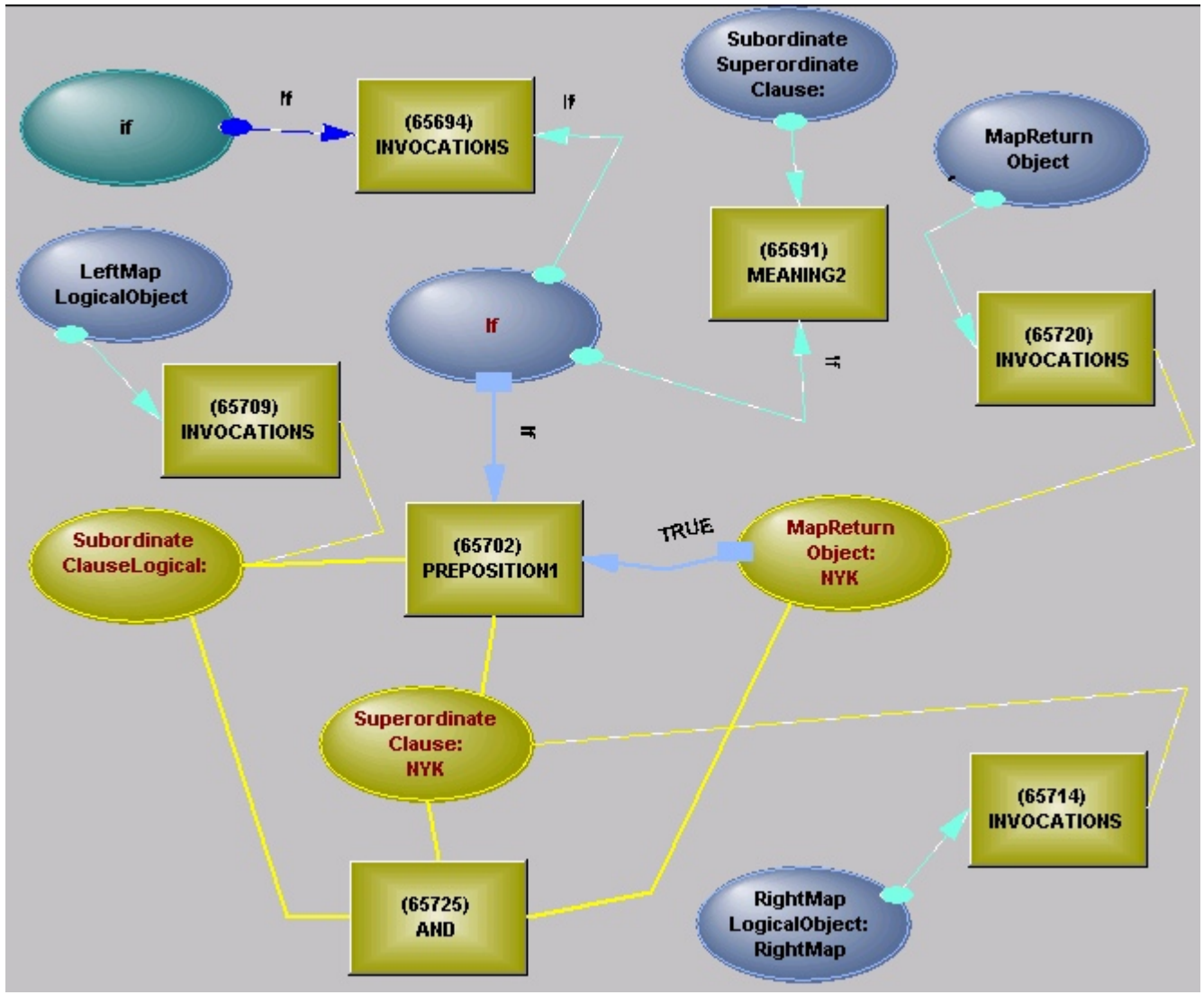
Our analog of this behavior is an active map - a free structure capable of crawling over another structure.

# Active Map

An active map is brought to the site of a discontinuity and uses constraint reasoning to check whether it matches the things to be connected. If so, it connects itself, it patches in or rearranges or destroys some structure, then disconnects itself and is put back on the shelf.

It breaks the connection paradigm, even though there is an implicit connection to some of the things it will connect.

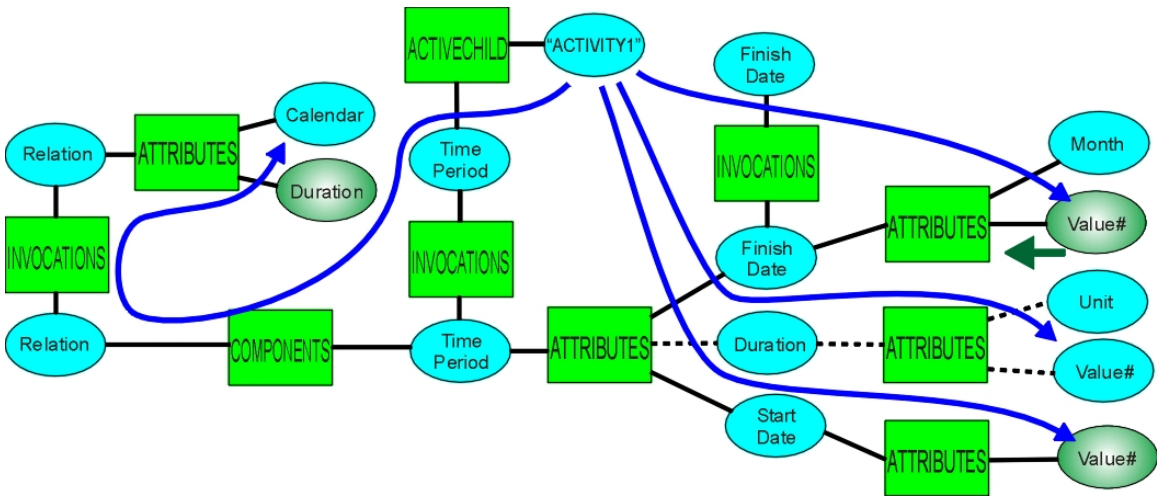
# Active Map for Joining Clauses




Map joins subordinate and superordinate clause

# Diffuse Operators

We shouldn't attempt to build every tiny piece of structure - some structure can be built on demand




Timing operators dynamically gather their inputs from inheritance, from direct assertion, from constraints from other operators, and from current time

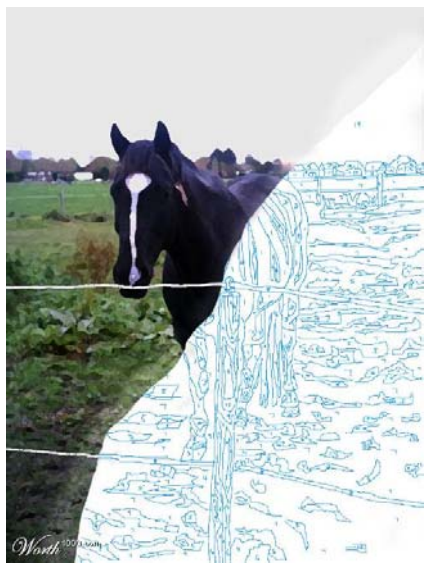
# Self - Extension

The text mining metaphor assumes that the text contains nuggets of knowledge dispersed in dross

A well written technical document has no dross

We need to read it word by word to be sure of what it is saying - anything less is a fudge which requires human labor to clean it up

A picture will emerge, based on the reader's preexisting knowledge and the knowledge built up during reading - a reading system needs dynamic self-extension to be able to paint the rest of the picture



# Mining of Knowledge Isn't Possible

The methods of Text Mining are antithetical to the extraction of knowledge from text

Knowledge requires a complete and detailed structure - Text Mining expects a few nuggets