SYNTAX OF THE TECTON LANGUAGE

D. Kapur, D. R. Musser, A. A. Stepanov
General Electric Research & Development Center

1. Introduction

This document describes the syntax of the Tecton language. It is neither complete, nor self-contained, nor consistent; its main purpose is to indicate the scope of the problems and to be a reminder of what syntactic elements should be considered.

2. Sentences

There are three types of sentences: propositions, imperative statements and questions.

- Propositions describe relationship between different kinds of objects and properties of objects. In particular, they are used in defining objects. A proposition does not have any effect; it instead states a property of objects which has a truth value.

- Imperative statements describe an action. The semantics of an imperative statement is that it causes an action to happen.

- Questions are a special kind of imperative statements which order (or request) an action. They have different syntax from imperative statements, but can be represented as imperative statements (for example, "what is 2 + 2?" means the same as "give the value of 2 + 2")

(In this draft only propoposositions are discussed.)

2.1 Propositions

There are six different types of propositions: categorical, modal, meta-deductive, verbal, lexical and compound. Compound propositions are formed by combining propositions using propositional conjuncts. Categorical propositions describe relations between different objects. Modal propositions describe a logical status of propositions. Meta-deductive propositions state relationships between propositions and objects that can perform deduction. Verbal propositions describe relationship between inputs/outputs and senders and receivers. These propositions are different from others because they describe relations between objects and linguistic objects (but not with the meaning of linguistic objects). They are not different semantically from categorical propositions, but are different syntactically. Lexical propositions assign meaning to sentences and other linguistic objects and are used for definitions. Each of these proposition types are discussed in more detail below.

2.1.1 Categorical propositions

A categorical proposition has three parts: subject, copula, predicate.

- The subject of a categorical proposition is an object type or class.
- The predicate of a categorical proposition is a predicate type or class.
- The copula of a categorical proposition tells about three types of attributes: tense, verbal type and mode, which affect the use of predicate.

Categorical propositions are the only kind of propositions that have the property of referential transparency .

2.1.1.1 Tenses

In this document, a rather narrow and abstract view of tenses would be taken. Only three tenses are considered: past tense, present tense and future tense. More general and a refined view would be to introduce time as an explicit parameter using which abstractions such as past, present and future tenses could be defined.

The subject of a proposition is evaluated in present tense even in the case of past or future tense propositions. So, "A was B" does not mean that at some point of time in the past it was true that "A is B." Instead it means that at some point of time in the past, when A was A', it was true that "A' is B."

For example, "every old man was a boy" does not imply that at any time in the past the proposition "every old man is a boy" was true. If it is desired to evaluate the subject in the same time as the predicate it can be done with the help of "at the same time" construct. For example, "every old man was at the same time a boy" is false. The word "always" is used to state that the proposition is tense invariant, for example "A is always B."

2.1.1.1.1 Tense Logic   It is assumed for the sake of discussion that past, present and future tenses are represented as "was, is, and will."

Axiom: "A was B" means that "There exists a past in which A was A', such that A' is necessarily B."

Axiom: "A will B" means that "there exists future in which A will be A', such that A' is necessarily B."

"Every boy will be a old man."

Propositions about the future are true when and only when they are necessary. Propositions about the future are false when and only when they are impossible. All other propositions about the future do not have truth values, but may have many other modalities.

2.1.1.2 Verbal types   There are four verbal types: descriptive, possessive, active and passive. The structure of the different verbal types in the copula can be represented with the help of following picture:

```
                  type of object
                      |
                      |
                      is (descriptive verb)
                      |
                      |
activity-----does-----object-----is done by-----operation
         (active verb)  |    (passive verb)
                      |
                      |
                      has (possessive verb)
                      |
                      |
                  attribute
```

Activities, attributes, operations, and object types will be discussed in later parts of this manual.

2.1.1.3 _Mode_ The mode of a copula is either "affirmative" or "negative."

2.1.2 _Modal propositions_ Every proposition has attributes called modalities. A proposition which describes a modality of some other proposition is called a modal proposition. Examples of modalities are: true, false, contrary, necessary, contingent, possible, impossible, deterministic, absurd, and meaningful. A proposition may have more than one modality. For example, if "X" is true then "X" is possible.

There are two types of modal propositions: real and nominal. A real proposition has the same form as a categorical proposition, except that it has a modal adverb associated with the copula. The meaning of a real modal proposition is that all given instances of the subject are predicated with given predicate with given modality.

The form of a nominal modal proposition is "It is M that P," where M is a modality and P is a proposition, or simply "M, P" where M is modal adverb. The meaning of a nominal modal proposition is that the subject is predicated with given predicate with given modality.

For instance, "people in this room normally don't smoke" is a real normal proposition, while "normally, people in this room don't smoke" is a nominal normal proposition.

The following relations between particular modalities hold:

- A proposition is true if and only if it is not false.

- A proposition is false if and only if it is not true.

- A proposition is contrary if it is false for all instances.

- A proposition is necessary if it is derivable and its negation is not derivable.

- A proposition is contingent if and only if it is not necessary and its negation is not necessary.

- A proposition is possible if and only if its negation is not necessary.

- A proposition is impossible if and only if its negation is necessary.

- A proposition is deterministic if and only if it is necessary or its negation is necessary.

- A proposition is absurd if it implies its negation and is implied by its negation.

- A proposition is meaningful if it is not absurd.

- A proposition A is normal if and only if "possible that A" implies "true that A".

- A proposition is possible if it is true.

- A proposition is true if it is necessary.

For all non-modal propositions there is one general law of abstraction: if a proposition is true for a subject X, then it is true for any concretization of X. However it is not necessarily true for modal propositions. For example, the proposition "it is possible that a scientist becomes a manager" does not entail the proposition "it is possible that I become a manager".

If a nominal modal proposition implies a real modal proposition with the same modality, subject and predicate, then the converse is true for the dual modality. If a nominal necessary proposition is true, then a real necessary proposition with the same subject and predicate is true also.

2.1.3 Meta-Deductive Propositions Meta-deductive propositions describe relations between objects and the meanings of propositions. They are used to describe properties of objects that have deductive capabilities.

The form of a meta-deductive proposition is "subject meta-verb that meta-clause" or "subject meta-verb abstraction". A meta-clause is a proposition.

Meta-verbs express meta-actions, actions which affect descriptions (or propositions) and not objects themselves. Meta-propositions deal with meanings of descriptions, and not with their linguistic form (that is done by verbal propositions).

Examples of meta-verbs are: know, deduce, think, assume.

Rules of inference that are valid on categorical propositions are not valid for meta-propositions. A meta-clause is not referentially transparent.

If X is A and Q knows that every A is B, it is not necessary that Q knows that X is B.

For example:

Alex works for GE.

Jack knows that everybody who works for GE has dental insurance.

It is not necessary that Jack knows that Alex has dental insurance.

One of the most important classes of meta-propositions is the class of consistent meta-propositions, with following rule of inference:

> If "a-imply" is a consistent meta-verb, then
> (X a-implies that A and X a-implies that A=>B) implies
> it is possible that X a-implies that B.

Examples of consistent meta-verb are: derive, know.

An example of an inconsistent meta-verb is: forget.

2.1.4 Verbal propositions As said earlier, verbal propositions are to state properties about input and output of objects. Verbal predicates express verbal actions - actions which deal with words and not their meanings. Examples are "say", "write", "transmit".

The form of a verbal proposition is "subject copula verbal-predicate: quoted-string."

2.1.5 Lexical propositions A lexical proposition is used for definitions. There are two types of lexical propositions: lexical descriptions and lexical type descriptions. A lexical proposition often introduces a new word into the vocabulary.

3. Noun constructs

A noun construct denotes either a type of objects or a set of objects.

A noun construct is one of the following:
1) proper name;
2) name of type;
3) two noun constructs connected with a conjunct;
4) qualified noun construct;
5) quantified noun construct;
6) exclusive noun construct;
7) individualized noun construct;

8) pronoun.

3.1 Proper names

Proper names denote sets of actual objects. It seems convenient to start them with capital letters.

3.1.1 Individual proper names The only peculiar thing to know about individual proper names as noun constructs is their somewhat irregular way of interacting with qualifiers. Qualified individual proper name denotes an empty set of objects (improperly described) if it is known that given qualifier is not predicated to an object which is denoted by this name. Otherwise a proposition which predicates given qualifier to the object is implicitly inserted into the text. For example, if we consider a proposition "Stupid Jack made a stupid speech", then it is equivalent to an empty proposition if it is necessary that Jack is not stupid; otherwise it is equivalent to a conjunctive proposition "Jack is stupid and Jack made a stupid speech."

3.2 Names of types

Names of type denote possible objects (unless actuality is included in the definition of the type). They normally do not start with capital letters. The way classes are defined will be introduced later in "Definitions". Changing denotation from possible to actual objects is done by quantifiers and by actualizing qualifiers.

3.3 Qualifiers

Qualifiers specify properties which objects in the class satisfy. Different kinds of qualifiers can be all reduced to propositional qualifiers.

3.3.1 Propositional qualifiers Propositional qualifiers are added to noun constructs with the help of conjunct "such that" and are propositions in which pronouns are bound over the qualified noun construct. For example, "programs, such that any verifier cannot verify them".

3.3.2 Noun modifiers Noun modifiers are abbreviations for most common types of propositional qualifiers. They have two parts: preposition, which determines a type of modifier; and noun construct which is a parameter to a propositional qualifier which defines a meaning for a given type of noun modifier.

3.3.2.1 Local modifiers There are local noun modifiers corresponding to each type of local proposition. For every type of a local proposition, <proposition> <local preposition> <locality description>, there is a type of a local noun modifier, <noun construct> <local preposition> <locality description>, where corresponding proposition which defines the meaning of a local modifier is "they are <local preposition> <locality description>". For example, "cars in the garage" is the same as "cars, such that they are in the garage".

3.3.2.2 Temporal modifiers There are temporal noun modifiers corresponding to each type of temporal propositions. For a type of a temporal proposition, <proposition> <temporal preposition> <temporal description>, there is a type of a temporal noun modifier, <noun construct> <temporal preposition> <temporal description>, where corresponding proposition which defines the meaning of a temporal modifier is "they exist <temporal preposition> <temporal description>". For example, "computers in 1950" is the same as "computers, such that they existed in 1950".

3.3.2.3 Structural modifiers Structural modifiers are introduced with the help of preposition "with". The meaning of structural modifiers is determined by a proposition "they have <noun construct>". For example, "people with blue cars" means the same as "people, such that they have blue cars".

3.3.2.4 Attributive modifiers Attributive modifiers are introduced with the help of preposition "of". If A is an attribute of B, then A of B is a noun if A is a noun.

3.3.3 Adjectives Adjectives are abbriviations of propositional qualifiers which stand in front of qualified noun. For example: "blue cheese".

3.3.4 Compound qualifiers Qualifiers may be connected with propositional conjuncts. The meaning of compound qualifiers is the same as the meaning as the meaning of a propositional qualifier which has as its proposition a compound proposition, which is built out of corresponding propositional qualifiers and has the same propositional form.

3.4 Quantifiers

Normally, quantifiers construct sets of objects out of types (or other sets). In some special cases which we are not going to discuss in this section they construct types of sets. (It happens when quantifiers appear in future tense propositions which are not necessary.) Some quantifiers (like negative quantifiers) are really propositional constructs. Quantified noun constructs cannot be qualified or quantified. Quantified noun constructs have two parts: quantifier, which determines the type of quantification and unquantified noun construct. They specify existing objects of given type unless possibility is explicitly predicated to a noun construct in which case possible objects of a given type are meant. Time of construction is determined by a proposition in which a quantified noun construct appear, unless absolute time is specified in a noun construct.

3.4.1 Universal quantifiers Universal quantified noun constructs are introduced by quantifiers "all", "every", "any". There is a difference between "all" on one side and "every" and "any" onther. "All" gives a set, while "every" and "any" give any element from the set. For example, "all members of CSB ate 25 hamburgers" is quite different from "every member of CSB ate 25 hamburgers". They specify all existing objects of given type unless possibility is explicitly predicated to a noun construct in which case all possible objects of a given type are meant. For example: "all brown cats", "all possible brown cats".

3.4.2 Negative quantifiers Negative quantified noun constructs are introduced by a quantifier "no". Their meaning can be illustrated by following examples: "He likes no computer scientist" means the same as "he does not like any existing computer scientist"; "no computer scientist likes him" means the same as "any existing computer scientist does not like him".

3.4.3 Existential quantifiers Existential quantified noun constructs are introduced by quantifiers "some", "a", and "an". (Note that indefinite article is not equivalent to "any").

3.4.3.1 Singular existential quantifiers Singular existential quantified noun constructs are introduced by quantifiers "a", "an", and "one". They specify one object of a given type. For example, "a man", "one big computer".

3.4.3.2 Plural existential quantifiers Plural existential quantified noun constructs are introduced by quantifiers "some", and "some of". They specify one noneempty subset of objects of a given type. For example, "some natural numbers".

3.4.4 Numerical quantifiers Numerical quantifiers are a refinment of existential quantifiers, namely, for any numerical quantifier A and any type T A(T)=>some(T). Numerical quantifiers specify a non-empty subset of certain cardinality of objects of given type.

3.4.4.1 Exact numerical quantifiers Exact numerical quantifiers specify cardinality. They are introduced by a cardinal or by a construct "as many <type description> as <set description>. If in the second case a set is empty, then the construct is equivalent to negative quantified noun construct. For example, "3 men", "as many hamburgers as people in CSB".

3.4.4.2 Non-exact numerical quantifiers Non-exact numerical quantifiers specify not cardinality of a set, but certain predicate on cardinality.

3.4.4.2.1 Relational numerical quantifiers Relational numerical quantified noun constructs are introduced by syntactic constructs "<comparator> <type description> then <set description>" or "<comparator> then <cardinal numeral> <type description>", where comparators are "more", "less", "more or equal" and so on. For example, "less then 5 men", "less hamburgers then people in CSB".

3.4.4.2.2 Fuzzy quantifiers Fuzzy quantified noun constructs are introduced by fuzzy quantifiers or by syntactic construct "approximately <exact quantified noun construct>. Fuzzy quantifiers are "many", "most", "few" and so on. For example, "most cats", approximatly 5 people".

3.4.5 Compound quantifiers

3.4.5.1 Negation of quantified noun constructs Negation of quantified noun constructs is introduced by syntactic construct "not <quantified noun construct>". For example, "not less then 5 people".

3.4.5.2 Disjunctive quantifiers Disjunctively quantified noun construct is introduced by disjunction of quantifiers. It is equivalent to disjunction of noun constructs. For example, "3 or 5 hamburgers" means the same as "3 hamburgers or 5 hamburgers".

3.4.5.3 Conjunctive quantifiers Conjunctively quantified noun construct is introduced by conjunction of quantifiers. It is equivalent to conjunction of noun constructs. For example, "3 and 5 hamburgers" means the same as "3 hamburgers and 5 hamburgers".

3.5 Exclusions

3.5.1 Strict exclusions

3.5.2 Exeptions

3.5.3 Compound exeptions

3.6 Compound noun constructs

3.6.1 Noun conjuncts

3.6.2 Noun disjuncts

3.7 Precedence in noun constructs

3.8 Pronouns

4. Predicative constructs

```
<imperative statement>::=<categorical imperative>|
                         <hypothetical imperative>|
                         <program>
```

4.0.1 Hypothetical predicates

<hypothetical predicate>::=<conjunctive predicate>|
<disjunctive predicate>|
<conditional predicate>|
<temporal predicate>|
<local predicate>|

Hypothetical predicates combine different predicates and propositions together and serve the same purpose as control structures in programming languages.

4.0.1.1 Conjunctive predicates

<conjunctive predicate>::=<unordered conjunctive predicate>
<ordered conjunctive predicate>

<unordered conjunctive predicate>::= <predicate> and <predicate>

<ordered conjunctive predicate>::= <predicate> and then <predicate>

In case of unordered conjunctive predicate both parts of it may be done in any order. In case of ordered conjunctive predicate the first part should be done before second.

4.0.1.2 Disjunctive predicate

<disjunctive predicate>::=<regular disjunctive predicate>|
<exeptional disjunctive predicate>

<regular disjunctive predicate>::= <predicate> or <predicate>

<exeptional disjunctive predicate>::=
either <predicate> or <predicate>

In case of regular disjunctive predicate any of the parts or both of them must be done. In case of exceptional disjunctive predicate only one part should be done.

4.0.1.3 Conditional predicate

<conditional predicate>::=<necessary predicate>|
<sufficient predicate>|
<necessary and sufficient predicate>|

<necessary predicate>::=<predicate> if <proposition>

<sufficient predicate>::=<predicate> only if <proposition>

<necessary and sufficient predicate>::=
<predicate> if and only if <proposition>

The necessary predicate means that the statement should be done when the proposition is true. The sufficient predicate means that the statement may be done only when the proposition is true. The necessary and sufficient imperative means that the statement should be done when the proposition is true and may not be done when the proposition is not true.

4.0.1.4 Temporal predicates when
until
while
after
before
during
sequentially
simultaniously

4.0.2 Local predicates

4.0.3 Programs