3. TYPES

3.1 Arithmetic types

The SIMULA 67 Common Base has two basic arithmetic types:

integer and real

3.2 The type Boolean

A variable of type Boolean may only assume truth values, i.e. true or false.

The internal representation and packing of Boolean quantities may be decided by the implementor. It is recommended that Boolean arrays be packed to ensure storage efficiency.

3.3 The type ref

A variable of type ref may denote ("be a name on", "refer to") an instance of a class declaration.

The qualification of a ref variable limits the range of values that this variable may assume. If the qualification of a ref variable is C, this variable may only denote instances of the class C or a subclass of C. An exception is the anonymous subclass of C formed by using C as a block prefix. The rules for the use of the local reference "this" prohibits generation of references to a prefixed block.

3.4 The type character

The type character is introduced to handle one-character alphanumeric information.

3.5 The type text

The type text is introduced to handle sequences of alphanumeric information.

Each declared text has a built-in sequential facility that make its use as an input and output buffer fairly easy.

The type text is the only type within the Common Base which has local procedures.

A procedure of type text is permitted both on the left and on the right hand side of an assignment statement.

3.6 Arrays

It is understood that arrays of any of the six types mentioned may be formed. It is recommended that Boolean and character arrays are packed.

3.7 Type procedures

Each of the six types may be used to form type procedures.

3.8 Initial values

The SIMULA 67 Common Base requires that the following initial values are given to local variables when a block, procedure or class is entered:

integer	zero
real	zero
Boolean	false
ref	none
character	implementation defined
text	notext

The same initial value shall also be associated with type arrays and the value associated with the name of a type procedure.