**Epipog**

**Specification**

**Index Class Family**

**Dec. 23, 2016**

# 1. Index Interface

The index object is the interface definition for specifying an index. An index is used to index records in a data store. An index consists of mapped values (e.g., hash) of one or more columns in a record and a location in the data store corresponding to the record. Conceptually an index is:

mapped-values location-in-data-store  
 mapped-values location-in-data-store  
 …

An index may be unique (no duplicate entries) or non-unique (duplicate entries permitted).

## Default Methods

The following are default methods defined in the interface. If not overwritten in an implementation, the default method is used in the implementation of the interface.

public default long[] Hash( Data value );

This method generates a pair of hash codes for a data value. If the data value is numeric, both hash codes are the numeric value. In the case of floating point numbers, the value is rounded. For characters, the value is the numeric code for the character, and for boolean it is 1 and 0, respectively. For strings, the first hash value is the java hashCode() function and the second hash value is generated by an internal hash function that generates a different value.

While the hash code generated by hashCode() results in low collisions, the generation of a second independent hash code is used to significantly reduce the likelihood of a collision.

## 1.3 Abstract Methods

The following abstract methods are declared in the interface.

public long Add( long hash, long pos, long data );

public long Find( long hash, long data );

public long Remove( long hash, long data );

public long Pos( int nth );

# 2. IndexLinkedList Implementation