

## Input File Formats for Knowledge Discovery Toolbox (KDT) and Combinatorial BLAS (CombBLAS)

### 1. Standard binary header

Each input file used in KDT/CombBLAS typically has to have a binary header that has the following fields and lengths. It is

'HKDT': four 8-bit characters describing the beginning of header

Followed by six unsigned 64-bit integers:

- version number
- object size (including the row and column ids)
- format (0: binary, 1: ascii)
- number of rows
- number of columns
- number of nonzeros (nnz)

If format is 'binary', this is followed by nnz entries, each of which are of size "object size" and parsed by the `HANDLER.binaryfill()` function supplied by the user. The general signature of the function is:

```
void binaryfill(FILE * rFile, IT & row, IT & col, NT & val)
```

IT is the index template parameter, and NT is the object template parameter. Below is an example:

```
template <class IT>
class TwitterReadSaveHandler
{
    void binaryfill(FILE * rFile, IT & row, IT & col, TwitterEdge & val)
    {
        TwitterInteraction twi;
        fread (&twi, sizeof(TwitterInteraction), 1, rFile);
        row = twi.from - 1;
        col = twi.to - 1;
        val = TwitterEdge(twi.retweets, twi.follow, twi.twtime);
    }
}
```

As seen, `binaryfill` reads indices as well. Please note that the file uses 1-based indices while C/C++ indices are zero based (hence the -1). In general, the number of bits used in the indices by the file should match the number of bits used by the program. If the program's bits should be larger/smaller; then a cast after the original object creation can be employed. Here is an example to read a file with 64-bit integer indices into 32-bit local -per processor- indices (given that they fit):

```
typedef SpParMat < int64_t, bool, SpDCCols<int64_t, bool> > PSpMat;
typedef SpParMat < int64_t, bool, SpDCCols<int32_t, bool> > PSpMat_s32;
PSpMat A;
A.ReadDistribute(string(argv[2]), 0);
PSpMat_s32 Aeff = PSpMat_s32(A);
```

**Important:** Ascii format with the binary header is currently not supported.

## 2. Ascii text file (without header information):

For backwards compatibility, KDT/CombBLAS allows ascii-only files without headers. In this case, the file doesn't start with 'H', instead it has (optional) comments lines that start with '%', followed by the first uncommented line that is:  
*#rows #cols #nonzeros*

This is followed by #nonzeros lines, each of which are of the form:  
*rowid colid parsable\_object*

The file row and column id's are 1-indexed

An example follows:

```
% Edges with retweets: 7
% Edges with follows: 10
9      9      13
1      2      1      0
1      3      1      2      2009-06-09 00:42:46
1      4      0      3      2009-06-03 20:13:40
2      4      1      0
2      8      0      1      2009-06-01 13:45:23
3      4      1      0
3      5      1      1      2009-08-21 00:45:10
4      6      1      0
4      8      0      2      2009-06-02 15:00:03
6      7      1      0
7      9      1      0
8      7      1      4      2009-08-31 23:32:11
8      9      1      1      2009-08-10 14:56:19
```