

# Knowledge Organiser 2

Knowledge Organiser II (KO2) is a substantially enhanced and RISC OS compliant development of Knowledge Organiser. The latter has been available in various incarnations since the era of the Arthur OS, but the current version is the first in which the software fulfils its designer's original intentions in terms of the integration of features, the overall functionality and the ease of use. It has evolved into that rare (and perhaps even endangered) species, the genuinely innovative software package. In sympathetic hands, KO2 is a powerful tool of unrivalled utility to anyone who needs to store, retrieve, select and cross-reference miscellaneous items of textual information. With the explosion of data becoming available via CD-ROM and online services such as the Internet, there is an ever-increasing requirement for such facilities.

KO2 is not a database but a textbase, or more precisely, a textbase manager. Also termed free-text retrieval programs, this type of software was restricted to multi-user platforms until personal computers were powerful enough to cope with them.

KO2 is designed to assist in the organisation of any kind of textual information including books, academic journal articles and abstracts, bibliographic references, statistical tables, output from online database searches, extracts from newspapers and CD ROMs, interview transcripts, correspondence, addresses, quotations, notes and cross-references. The volume and variety of this information can frequently be

an obstacle rather than an aid to understanding. There is often a need to store the information acquired even though it is in variable-sized chunks gleaned from disparate sources dealing with different aspects of a topic. This leads to the problem of how to classify items before clear ideas about their relevance and the boundaries of the topic have emerged.

KO2 offers the user an open-ended way of tackling these tasks. With a conventional database, the sequence of operations involved in defining the fields, entering the data, carrying out searches and

Kenneth Levine reviews a package which will help all authors to organise their reference material, whatever their field of study

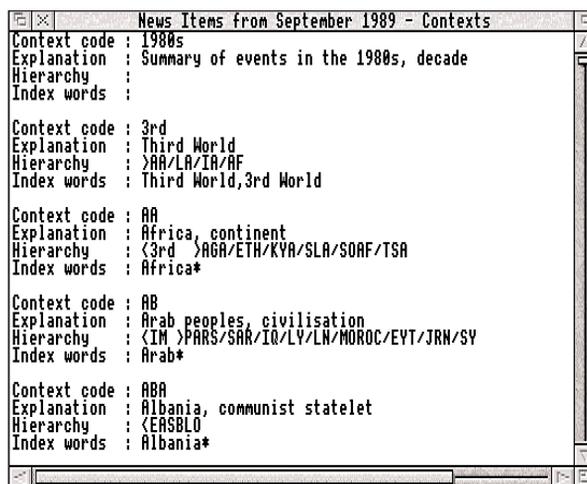


Figure 1. Setting up context

producing a report is more or less constant. The mode of use for KO2, on the other hand, varies greatly according to the types of text being used and the sophistication of the organising ideas that the user has available at the outset. You can use KO2 in a very basic way as a string search utility, or you can build in an

elaborate set of codes in order to tackle a sophisticated project. The Package

The package consists of a single floppy containing the application itself, a sample file based on entries from a dictionary of quotations, a utility to convert textbases created with previous

releases and a read-only version so that people without KO2 can examine your textbases.

The 92 page ring-bound manual is well written and has an index. It gives a particularly good sense of the range of possible uses, while the Simple Search Tutorial is very helpful on how to search already indexed textbases. New users wanting to create a major textbase from scratch should first read the manual as a whole as an aid to planning, and to ensure they exploit the available facilities effectively.

Texts, Fields and Codes

A KO2 textbase is built up from a series of existing files, the source texts, which can be imported into a textbase simply by dragging and dropping them into the appropriate KO2 window. Using the built-in editor, any extraneous material can be eliminated, but KO2 appears happy to accept almost anything textual, so DOS and word processor files can be imported directly (after their filetype has been changed to Text). When a completely new textbase is being created, the necessary disc space is reserved for it, calculated from the user's completion of a File Initialisation window. Then a record card window is designed, by defining field size and type and using the mouse to drag the display boxes into position. The source texts can then be added one by one into the textbase and coded by assigning appropriate entries to all the relevant fields.

There are two basic types of field in KO2 and they are the key to this software. First, Linker fields deal with fixed characteristics of the source texts; they could, for example, record a reference code for each text item, its country of origin and its date of acquisition. Codes for

linker fields can take text, numeric or date formats. Additional linker fields can be created up to a limit set by the pre-selected size of the record card.

To keep the indexes small, KO2 encourages the use of abbreviations in all textual fields so that the source field for a book should contain an abbreviation rather than the full citation, but a separate scrolling window is provided where expansions can be listed. This feature is duplicated for context codes and the index of keywords; full search, edit and

Context codes are used to capture the content of each source text and to reflect the new themes and dimensions of the topic thrown up in the course of analysis. A vital part of KO2 coding is that the user may associate up to ten keywords with any single context code; keywords may end in a wildcard asterisk to allow for plurals and roots with variable endings (figure 1). KO2 will, on request, automatically index a source text by searching for all keywords and placing the appropriate context codes in the Context field for that item. The capacity for the user to set up connections between

keywords which occur in the text and codes which do not (but which stand for important abstractions and classifying ideas) is a very powerful facility.

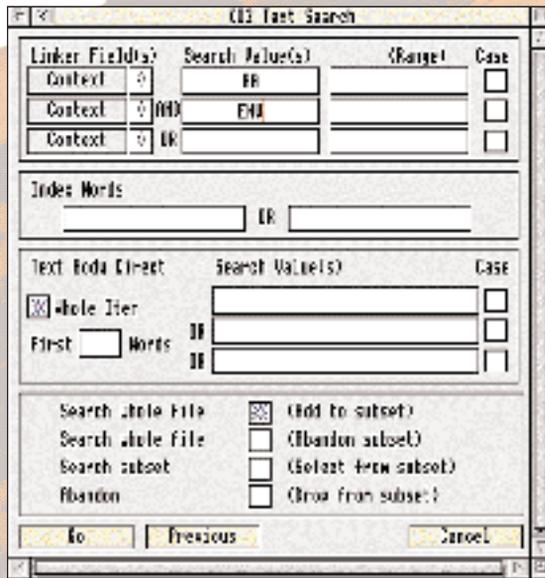


Figure 2. Searching for context codes

rename facilities are available within all these windows. New linker codes (but not linker fields) may also be created at any time.

The second type of field is a Context field in which users can create a series of mnemonic or arbitrary codes each of up to six characters in length. They may be assigned hierarchical relations with each other (that is, one code may include, or be included by, another).

A practical example

Consider a hypothetical journalist working on a feature on environmental problems in Africa who uses KO2 to analyse news items (250-500 words) downloaded from the online Observer Dialcom world news service. The journalist could create context codes like ENV for the environment (associated with the keyword environment\*), and at a lower level in the code hierarchy, EN for energy (associated with the keywords energy, oil, fuel and electricit\*), PO for pollution (associated with the keywords pollut\* and contamin\*). A hierarchy of country codes could also be set up: an AA context code for Africa could include ETH, KYA, SOAF (for Ethiopia, Kenya, South Africa, etc).

The journalist is then able to select out the subset of items which contain both environment codes and AA from the mass of irrelevant news (figure 1). Later, she can search the subset by country codes to look for themes. Later still, she may decide to work up the links between development and pollution problems, so she can then add further context codes (eg. GAT for the General Agreement on Trades and Tariffs, IMF for the International Monetary Fund and so on) and re-search.

### Searching

In one pass through the data, the user may search for up to three context or linker codes and simultaneously search the textbase directly for index keywords or any other strings (although it is quicker to produce a subset containing the nominated codes and then to search this subset for strings). The limit of three codes is not restrictive since the search of the whole file can be repeated and the hits added to (or removed



from) the existing subset.

A successful search results in the record card and the source text being displayed for the first member of the subset (figure 3 shows a search for context codes mentioned above). Searches for multiple codes can be logically ANDed or ORed and hierarchical relations specified. You may browse the source texts and their associated record cards forward or backwards, jump to a specific text, or tag texts manually and make the tagged items the current subset or exclude them from it. You can keep four windows open simultaneously on different texts or different parts of the same text. Subsets can be sorted on any linker field (text, numeric or date) and may be merged into other textbases. A full log of how a subset has been produced is available for reference. Practically everything created within KO2 can be exported as ASCII text to wordprocessors.

### Capacities and Performance

Apart from the 6-character limit on context codes and 10 keywords per context code, nearly all the capacity parameters in KO2 are

### Product: Knowledge Organiser 2

Supplier	Clares Micro Supplies 98 Middlewich Road Rudheath, Northwich Cheshire CW9 7DA Tel. 01606 48511 Fax 01606 48512
Price	£79.95 inc. VAT

flexible and adjustable by the user to achieve suitable trade-offs within the storage space available. Once initialised, the total size of the textbase is fixed, but the contents can be exported to a newly defined textbase with different internal allocations within a larger or smaller overall size. Search speed for either codes or strings is excellent.

### Summary

KO2 already has a lot of bells and whistles but one desirable enhancement would be the automated entry and

