RIDDLE Project
Results of the Feasibility Study

The Public Version of the Final Deliverable
of the RIDDLE Project

Undertaken by
Centrum voor Wiskunde en Informatica[1]
Cartermill International Ltd[2]
and
Rutherford Appleton Laboratory[3]

Sponsored by the Libraries Programme of the Commission of the European Communities' TELEMATICS Programme

April 1995

[1]P.O.Box 94079, 1090 GB Amsterdam, The Netherlands
[2]St Andrews, Fife KY16 9EA, Scotland
[3]Chilton, Didcot, Oxon OX11 0QX, England

RIDDLE Project
Results of the Feasibility Study

The Public Version of the Final Deliverable
of the RIDDLE Project

Undertaken by
Centrum voor Wiskunde en Informatica[1]
Cartermill International Ltd[2]
and
Rutherford Appleton Laboratory[3]

Sponsored by the Libraries Programme of the Commission of the European Communities' TELEMATICS Programme

April 1995

[1]P.O.Box 94079, 1090 GB Amsterdam, The Netherlands
[2]St Andrews, Fife KY16 9EA, Scotland
[3]Chilton, Didcot, Oxon OX11 0QX, England

Executive Summary

This deliverable deals with the results of the feasibility study carried out by the RIDDLE project. This project studies the feasibility of scanning the contents pages of scientific journals, with a view to including information on the individual articles in each journal issue in a number of remote on-line library catalogues. A RIDDLE system therefore provides an automatic cataloguing service. The project has designed and developed a pilot system capable of performing this task as automatically as possible.

The requirements were identified in the deliverable concerned with User and Technical Requirements, and the details of the feasibility study have already been published in the following documents:

Scanning Technology Assessment

Scanned Image to Text Technologies Assessment

Translation of Contents Pages Text to On-line Library Catalogue Format

Communication and Transmission Issues Assessment

This report provides a synthesis of the satisfaction of the requirements laid out in the Requirements deliverable mentioned above, thus bringing together the whole technical scope of the project with the overall conclusion as to the technical feasibility and viability of such a system.

The project has shown that it is possible to build such an automatic cataloguing system, using existing technology (possibly already available within libraries), and as a low cost, timely and labour efficient alternative to existing commercial services.

Glossary

A3:	DIN paper size (297mm x 420mm)
A4:	DIN paper size (210mm x 297mm)
ADF:	Automatic Document Feeder
ANSI:	American National Standards Institute
ASCII:	American Standard Codes for Information Interchange
bitmap:	image format in its simplest form with pixels of 1 bit only
CI:	Cartermill International (formerly Longman Cartermill)
CWI:	Centrum voor Wiskunde en Informatica
cps:	characters per second
DCA:	Document Compound Architecture
DIN:	Deutsche Industrie Norm
DOS:	Disk Operating System for IBM compatible computers
dpi:	dots per inch
DTD:	Document Type Definition
EC:	European Community
ECU:	European Currency Unit
GIF:	Graphics Interchange Format
GPIB:	General Purpose Instrument Bus (or IEEE-488 bus)
HPIB:	Hewlett Packard Instrument Bus

IEEE:	Institute for Electronic and Electrical Engineers
IP:	Internet Protocol
ISDN:	Integrated Services Digital Network
ISO:	International Standards Organisation
ISSN:	International Standard Serial Number
Kbps:	Kilobits per second
logo:	an image used to identify an object. In RIDDLE, an image fragment which can help identify a journal
Mb:	Megabyte
OCR:	Optical Character Recognition
OLC:	On-Line Catalogue
omnifont:	(of OCR software) able to handle any non-stylised font
PC:	IBM compatible personal computer
PICT:	QuickDraw picture format for Macintosh computers
pixel:	single element of an image, consisting of 1 or more bits
PSDN:	Packet Switched Data Network
RAL:	Rutherford Appleton Laboratory
RAM:	Random Access Memory
RIDDLE:	Rapid Information Display and Dissemination in a Library Environment
RFT:	Revisable Form Text

RS232C:	Alternative name for V.24 international standard communications interface
RTF:	Rich Text Format
SCSI:	Small Computer Systems Interface; ANSI standard (ANSI X3.131-1986) for the connection of peripherals to computers
SGML:	Standard Generalised Markup Language, ISO Standard 8879
TCP:	Transmission Control Protocol
TIFF:	Tagged Image File Format
UKP:	United Kingdom Pounds
WIMP:	Windows, Icons, Mouse and Pointer. Used to describe a modern computing user interface
zone:	user-definable area of an image, used to assist the OCR software to identify those parts which are important

Contents

1 Introduction

The RIDDLE project studied the feasibility of scanning the contents pages of scientific and technical journals, with the view of extracting the bibliographic information on the individual articles in each issue, for insertion into On-line Library Catalogues (OLC's). The RIDDLE Consortium consisted of Cartermill International (CI, formerly Longman Cartermill), Centrum voor Wiskunde en Informatica (CWI), and Rutherford Appleton Laboratory (RAL).

The project investigated the current state of the art in the areas of scanning technology, Optical Character Recognition (OCR), and automatic markup/translation. International, industry and formal standards such as the Standard Generalized Markup Language (SGML, ISO 8879) were also examined in this context. Having analysed the current technical possibilities for image capture, OCR, and automatic markup, the integration of such data into the OLC was explored.

The feasibility study was carried out under the following headings:

User and Technical Requirements

Scanning Technology Assessment

Scanned Image to Text Technologies Assessment

Translation of Contents Pages Text to On-line Library Catalogue Format

Communication and Transmission Issues Assessment

Each of these modules produced a deliverable in the form of a report, which has already been published. The User and Technical Requirements deliverable assessed the problem and created a set of requirements, thus producing a foundation for the remainder of the technical work to be carried out by the other four modules. The report was structured in such a way that relevant chapters could be used as input to the other technical areas.

The structure of this report follows the same structure, and contains the following chapters:

Chapter 3 Common RIDDLE Requirements. This chapter assesses the requirements identified that were common to all modules of the RIDDLE system.

Chapter 4 Scanning System Requirements. This chapter assesses the requirements identified for the Scanning module.

Chapter 5 Image to Text Conversion Requirements. This chapter assesses the requirements identified for the Image to Text Conversion module.

Chapter 6 Text to OLC Conversion Requirements. This chapter assesses the requirements identified for the Text to OLC Conversion module.

Chapter 7 Communication Systems Requirements. This chapter assesses the requirements identified for the Communications module.

This report, therefore, brings together all of the Requirements Satisfaction chapters of the deliverables, providing a consolidated view of the initial system requirements and their satisfaction. The approach taken is to assess whether it would be possible to satisfy each requirement in the general sense, although, where appropriate, this is enhanced by a description of how existing commercial packages (as used in the RIDDLE pilot) cope with these requirements.

A RIDDLE requirement is represented by a unique identifier of the form **R x.y Z**, where:

R. This is a constant indicating that a requirement follows - this differentiates the identifier from the chapter and section headers.

x. This is a variable indicating the chapter in which the requirement is defined in deliverable D1.

y. This is a variable indicating a sequential number for the requirement relative to the chapter. The first requirement in a chapter has the value 1 (one).

Z. This variable has the value E (for essential) or D (for desirable). A requirement flagged as being essential is considered fundamental to the RIDDLE project meeting its objectives. A requirement specified as being desirable is not considered central.

Prior to these Requirements Satisfaction chapters, chapter 2 contains an overview of the RIDDLE system design. This chapter contains necessary information to set the scene for the succeeding chapters.

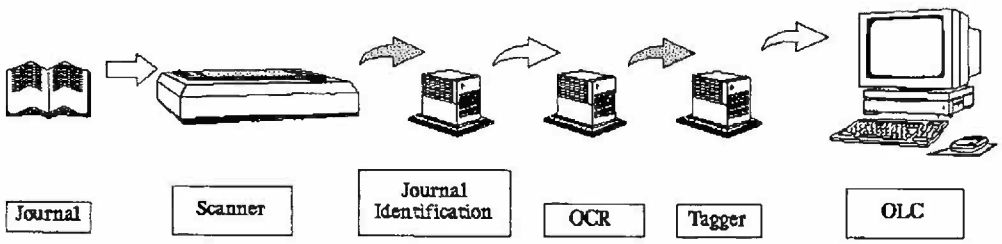
The final chapter provides overall conclusions as to the results of the study and the feasibility of developing such a service in libraries.

2 RIDDLE System Overview

This chapter provides an overview of the RIDDLE system, describing the components and the interfaces between them.

2.1 System Design

RIDDLE explored the feasibility of automating the cataloguing of scientific journal articles. It was the intention to try and automate the process as much as possible, thereby reducing the human intervention (or interaction) required. To allow the feasibility study to take place it was necessary to adopt a model for the system. The system architecture is described graphically in the following figure.



The architecture is basically of modular design consisting of the following components:

Scanning module. This allows the capture of an image of the journal contents pages (and possibly title page if it contains relevant journal data, or it is required for automatic journal identification). The images will either be stored on disk to be picked up by the succeeding module, or passed directly to that module.

Image to Text Conversion module (Journal Identification and OCR). This comprises 2 sub-modules: automatic journal identification, and contents page capture. The first attempts to automatically identify the journal being processed, and thereby configure the system parameters optimally for that particular journal. This sub-module could employ various techniques to identify the journal, including OCR to find the text of the title, or image processing techniques to recognise a logo or identifying graphic present on the page. It is possible therefore that a colour scan of the pages may be required.

The second sub-module is the OCR engine which produces the text of the contents pages for analysis by the tagging system. A monochrome scan will be sufficient for the OCR

process to extract the text. At this stage the journal will be known so the scanning and OCR components can be set with the correct parameters to give the best results for the journal being processed.

Text to OLC Conversion module (Tagging and Translation). This module is also divided into 2 parts. The tagging process takes as input the text produced by OCR, and applies visual recognition or pattern matching techniques to try and identify the various data elements embodied in the text. The output of this stage is an SGML marked-up document which conforms to the RIDDLE Document Type Definition (DTD). The translation system is implementation dependent in that it has to produce load records for the particular OLC at a particular site. It will be possible though to write one translation procedure which will be applicable to all journals to be processed, as the input will always be consistently formatted ie. SGML text.

Communications system. In the case when one or more of the above components is located physically remote from the others, some form of communication system will be required to facilitate transport of the information from one component to the next.

Detailed descriptions of all of these components can be found in the other RIDDLE deliverables.

An implementation of this architecture, or pilot, has been developed to prove the viability of the concept.

To build such a pilot system it has been necessary to use existing "off-the-shelf" technology as far as possible, and perform a system integration exercise to integrate these components and facilitate communication between them. It is not the intention of the project to propose any "best buy" products for use in an implementation, but rather to show that existing technology is such that the components required to fulfill the necessary functionality can be obtained.

The approach taken towards the development of the pilot system is that of flexibility towards the specific nature of the components of the system. The findings of the project have therefore been specified as an overall system design, together with a set of "profiles". Each of these profiles specifies the features that the specific component should exhibit to be able to be used by the system.

Thus if any customer (ie. librarian) desired to implement a pilot RIDDLE system of their own, based on the findings of this feasibility study, it would be possible to use the system design and component profiles to select existing products and perform the integration activity. This would then allow the library the possibility of making use of components or technology already present in the installation environment, as part of the RIDDLE system.

Of course if a commercial product was being developed to implement this system, a different approach would be taken. Rather than interfacing other commercially available products to

prove the viability of the system, a bespoke solution would be developed to perform the operations seamlessly and effectively.

Having adopted a modular system architecture, this allows for future changes to work practices to be incorporated into the system, by allowing modules of the system to be removed (or added) as necessary. For example, if the bibliographic journal article information could be supplied by the publisher directly in electronic format, then the Scanning and Image to Text Conversion modules could be decoupled, and the electronic text input directly to the tagger. In fact, if the publisher could supply SGML marked-up electronic data, it may also be possible to decouple the Tagging sub-module and feed the information directly into the Translation sub-module.

The particular solutions adopted for the RIDDLE pilot system are described below.

2.2 Pilot Implementation

CWI was chosen to be the site of the pilot, since it was the only one of the partners which possessed all the necessary hardware required. However, the choice of operating system (Unix) allowed software to be developed at all of the sites. CWI also possessed the resources necessary for loading catalogue entries generated by RIDDLE into the OLC, which was deemed to be an essential part of the pilot demonstration.

This section covers the design work for the pilot, and describes two early systems which were constructed to help in understanding the problems.

2.2.1 Basic Design

Having decided on the site for the pilot, this determined the third-party software packages which should be used. The main programming work is then the construction of the interfaces between these packages, the design of any intermediate files required, and the design of a suitable user interface to the whole system.

The packages and equipment used are:

Agfa Scanner The scanner is connected to a SUN workstation, running a version of Unix.

Pixel!FX This is the scanner software, running on the SUN, which produces images in TIFF format. It is, necessarily, interactive, and provides its own window-based user interface.

ScanWorX The OCR software from Xerox, also running on the SUN, can take images in TIFF format and produce output in ASCII text, "intelligent" ASCII, WordPerfect, FrameMaker and Interleaf. It can run in non-interactive mode, driven by command files.

FastTag The tagging software also runs on a SUN. Special thanks are due to Interleaf for making it possible to run the system at CWI in order that the pilot might be demonstrated.

Siemens OLC The catalogue system at CWI is based on Siemens software. It can be loaded via a batch file, so the intention is that the pilot will generate a suitable file for loading. No attempt will be made to make a direct connection to the Siemens software, since this would not be the way the catalogue is currently managed.

It was not possible to obtain a copy of suitable image processing software which might be used to identify a logo. However, the Scanned Image to Text Technologies Assessment deliverable reported independent tests carried out on the Enigma software (from the University of Amsterdam) which provided sufficient evidence that this concept is viable.

There was no need to introduce any communications features into the pilot, since none of the networking solutions proposed required any proof of concept.

Interface requirements were also considered in the feasibility study. The main investigative work involved the various methods of journal identification, possibilities of error handling, and the user interface. Two "alpha" pilots were constructed to provide information on these aspects. This preliminary work, including some of the code written, forms part of the final demonstrator system.

2.2.2 First Alpha Pilot

The first alpha pilot (AP1) was written at RAL during the first quarter of 1994, using Unix Bourne shell commands. A simple command-line interface was provided, and the functions of the scanner and OCR packages were simulated by providing previously prepared files (which are selected at the appropriate moment and used as the output from these packages). AP1 proceeded to the point where output is available to the FastTag package, since work beyond this point had still to be completed. There was no difficulty at all in porting the script to CWI.

AP1 therefore covered the journal identification and contents page capture stage. Error handling was included, as was the display of the "OCR output" in an editor for "correction" prior to being submitted to the autotagger. In addition, AP1 allowed details of a new journal to be added to the system. Simple file formats were devised to contain the search criteria and the basic journal data, and this was extracted by shell commands (this was a cumbersome activity, involving the use of "awk" and "read", which is better done by program in the final pilot). Journal identification was achieved by the use of Unix commands, and here the Unix system commands proved valuable. "head" and "tail" were used to implement the "locality" feature (see Scanned Images to Text Technologies Assessment deliverable) and "sed" was used to look for the text strings. "grep" was used to check for the existence of information in the various

files. The user interface was implemented using "echo" and "read", and this gave valuable information on the order of commands and the most appropriate dialogue.

AP1 was run by members of the library community at both RAL and CWI, and the comments received led to improvements, and also helped in the production of the next alpha pilot.

2.2.3 Second Alpha Pilot

While it is hardly necessary to "prove" that WIMP systems work and are considered beneficial, nevertheless it was considered important to provide as good a user interface as possible, given the limited resource available. A RIDDLE system is not confined to one type of interface, but most of the scanner software appears to assume some form of Window system. It was therefore decided to investigate the possibility by developing a second alpha pilot, AP2.

Fortunately, RAL have access to XDesigner, a user interface generator which helps produce complex window system based on the X11 toolkit and Motif widget set. The output from this system is a set of C routines which do not require any special XDesigner runtime library. The generated code provides for the layout of each window, the detection of an event associated with a particular widget (such as a button press or data typed into a text widget), and the generation of routine stubs where the application-specific code can be inserted. Links between windows can be provided, and there are mechanisms for features such as pop-up windows (for help messages). Since the generated code did not have any proprietary software in it, it was possible to port the resulting software to CWI, where it executed with only one minor change (caused by a difference in implementation of one of the X11 library routines). Thus, although the generator is available only at RAL, it was decided to make use of this for the user interface experiment.

Discussions were held with library personnel to discover what design criteria should be used. In addition, input was received from the RAL human factors group, giving information about the best way of laying out a window. As a result, a small set of widgets was selected (including buttons, scrolled lists and text items) and a standard layout adopted.

17 windows were designed and the code generated (6000 lines produced by XDesigner, together with an additional 1000 lines to handle the various events). This represented that part of the pilot which deals with the journal recognition and data capture. Sufficient code was supplied to link the windows together, and to give the illusion that scanning is taking place. It was therefore possible to sit a library staff member in front of the system and give them a feel for what the final pilot would be like.

It is noted that a similar window-based alpha pilot could have been written on a PC, using a product such as Visual Basic (which is available at both RAL and CI). Since Motif is very

similar in appearance to Microsoft Windows, it would be possible to produce an almost-identical looking system this way. This could be mounted on a portable PC, and taken to different library sites to collect user feedback. Unfortunately, there were not sufficient resources to carry out such an implementation.

2.2.4 Full Pilot

The windows used in AP2, together with the feedback obtained from the users, formed the bases of the full system. The pilot was extended by the inclusion of two methods of journal detection (text strings and ISSN) and implementation of the "add journal" feature. The final system consisted of 33 screens, with 8500 lines of generated C code and 2500 lines of additional routines. This included genuine links to the scanner and OCR packages, and suitable translators to produce output for both CWI and RAL library catalogues. Some changes were required to allow for the fact the scanner had to be run interactively, thus displaying its own windows interface alongside that of RIDDLE.

The original design assumed that journal detection would only be invoked if the number of journals known to the pilot exceeded a particular limit. In order to be able to demonstrate the journal detection system separately from the journal processing, an additional option was added to the first screen, which had no other operational purpose. Therefore it is possible for the operator to bypass the automatic journal detection, and opt to select the journal being processed from a list known to the system. The other major change was the incorporation of an additional screen in the "add journal" section to allow the operator to specify whether a serif font was used by that particular journal, since it was found by experience that ScanWorX produced much better results if it knew beforehand what type of font was being read.

It was not possible to obtain a copy of the image reading software, so no code relating to that detection method was generated.

A set of CWI journals was selected for the final demonstration and suitable FastTag programs written for them. In addition, tag programs were written for some of the RAL journals. A version of the pilot which retained the scanner and OCR simulation was prepared and run at RAL. This version produced output designed for the RAL OLC, and made use of the packages available at RAL ("ten" for editing and "Mark-IT" for translation).

3 Common RIDDLE Requirements

This chapter provides a synthesis of the Requirements Satisfaction chapters from the individual technical deliverables, for the requirements common to all modules of the system. Therefore material is taken from: Scanning Technology Assessment, Scanned Image to Text Technologies Assessment, Translation of Contents Pages Text to On-line Library Catalogue Format, and Communication and Transmission Issues Assessment.

Further information can be found in these source documents.

3.1 Capabilities

R 2.1 E *A RIDDLE system should be able to process scientific journals contents pages that are written in any of the Roman alphabet languages.*

The term "Roman alphabet language" includes those languages which have accents. Most commercial OCR packages are supplied with the pre-programmed ability to read Roman alphabet based Western European languages. However, there are some Easter European languages which are Roman alphabet based also, but which have accents (usually on consonants) which are not yet supported by any of the commercial packages.

By careful choice of OCR package, it is possible to obtain a greater selection of pre-programmed alphabets. An alternative approach is to select trainable OCR software and teach the package to recognise the required additional characters, for example accented characters. Using this latter technique, the supported alphabets could in fact be expanded to recognise other languages including Scandinavian and those East European languages mentioned above. This can also be used to allow the OCR software to recognise scientific and mathematical characters. Care should be taken however when defining new recognition symbols, to ensure that the output character code or string of codes is unique.

Regarding scanning, most scanners are able to process A4 page size, which is the predominant paper size for scientific journals. To ensure that images are produced that will be adequate for the OCR process, a scanner should be chosen that does not exhibit the phenomenon of "colour drop".

The Text to OLC Conversion module is invoked at the end of the processing stage of a journal. All the problems associated with character sets have already been handled by this point.

Subject to the above constraint, this requirement is capable of satisfaction.

3.2 Constraints

3.2.1 Human-Computer Interaction

R 2.2 E *Operator interaction should be kept to a minimum.*

The objective of the RIDDLE project as a whole is to attempt to automate as much as possible the complete journal article cataloguing process. Mechanisms have been investigated relating to the feasibility of automating the journal identification and contents page capture processes.

Due to the mechanical nature of the scanning operation, some human action will always be required, even if it is only the operator being asked to place the page on the scanner. More interaction is likely to be required to correct errors in either journal identification (where the operator may have to provide information as to which journal is being processed) or contents page capture (where an opportunity will be provided to allow the operator to correct OCR errors if required). Both scanner and OCR packages exist which allow control data to be set by program, and these should be selected in preference to avoid unnecessary operator activity.

The main part of the system where operator interaction will be required, will be during the "set up phase", where new journals are added to the system. This process allows the operator to set up new journal descriptions, including the definition of the recognition features for the journal, and the optimum scanning and OCR settings for processing.

An autotagger such as FastTag performs automatic generation of the SGML intermediate document, which in turn lends itself to a variety of automatic conversion methods to create the OLC load file. Operator intervention is therefore confined to error correction at this stage of the process.

When a communications system is required, the use of command files (see R 6.5 E) and standard filenames will allow automatic transmission with no operator intervention. This requirement may conflict with the need for security (see R 6.14 D). The effect can be minimised by batching transfers together, when the password will only be required once.

This requirement can therefore be satisfied.

R 2.3 E *The human-computer interface should be easy to use by normal library staff.*

It is the intention of the project to supply the operator with one coherent interface. The nature of the system is such that this interface could be command-driven, menu-driven (with the operator selecting the item from the keyboard) or a full WIMP (Windows, Icons, Mouse and Pointer) system. One of these interfaces will be familiar to any library staff member who uses a computer. "Off-the-shelf" components (eg. OCR package) must be selected to coexist with the chosen interface.

Those packages which can run non-interactively will have no difficulty in meeting this require-

ment. For example, the packages used in both sub-modules of the Text to OLC Conversion module are run as batch items, and so provide no direct human-computer interaction. These programs will be run from within a command file envelope, whose design will conform with this requirement.

In the case of the scanner (which by its nature requires some operator interaction), it will be necessary to ensure that software which interacts with the scanner has an interface which does not conflict with the rest of the system (eg. requires a different window manager). Suitable packages are available which meet this requirement.

Keeping the operator interaction to a minimum is also supported by saving the scanner and OCR settings to file (see R 3.9 E).

Thus it should be possible to provide the operator with one simple clear interface where information and instructions are displayed in the most meaningful manner. The pilot system has been designed along such principles.

R 2.4 D *The human-computer interface should follow the "look and feel" of similar packages in use by library staff.*

As stated above, it is possible to construct several different types of interface, providing different "look and feel". From the information reported in the User and Technical Requirements deliverable, WIMP systems are consistent with the modern library environment. It is possible to select packages which can integrate with the more common windowing systems.

The RIDDLE system will have some form of control software around the individual "off-the-shelf" components, so that the human-computer interface will be determined rather by this than the commercial packages.

3.2.2 Adaptability

R 2.5 E *A RIDDLE system should be able to support the addition of new scientific journals as appropriate.*

The system supports the addition of new journals by allowing the operator to supply information describing these journals, including the optimum scanner and OCR settings for processing. It is also possible to update existing journal definitions.

If the scanning software is able to cope with any additional settings required by the new journal, no difficulties are anticipated. Also in the case of a changing journal layout, the RIDDLE system will be able to adapt the scanning settings. To be sure that a scanning system satisfies this requirement, a scanning system with "colour drop" should be avoided.

A new journal will require the provision of an appropriate program for the selected autotagger.

No change will be needed to the conversion system. The inclusion of these files can be handled by the operational procedures outlined in the Scanned Image to Text Technologies Assessment deliverable.

This requirement is therefore satisfied.

3.2.3 Availability

R 2.6 E *A RIDDLE system should be available as required to fit in with normal library working procedures.*

The RIDDLE system will be able to fit in with the library procedures in the normal manner applicable to computer equipment. There are no special features which would make this difficult to achieve.

There are no scanner characteristics which will prevent one fitting into the normal library working procedures. Heat output of scanners do not require special air conditioning. All scanners are desk-top and do not require a large space. Noise levels are comparable with noise levels of photocopiers.

Where a communications system is required, it is normal practice to leave communications software running at all times, at least on multi-tasking machines. Therefore no difficulty is anticipated.

3.2.4 Portability

R 2.7 D *The system should use software which allows the modules to be moved to different hardware.*

It is entirely possible to develop an application that is portable across the most common computing platforms in use today. If a RIDDLE system was going to be realised as a commercial product, this is the development approach that would be adopted. However, the incorporation of third-party products into the system may cause difficulty, since there are no examples of relevant packages which are available on all the different target platforms.

No difficulties will be anticipated, on the other hand, by staying within a particular line of hardware (PC, Mac etc.).

As has been shown in the development of the pilot, it is possible to construct a system such that the interfaces to the packages are well-defined. Replacing one product by another would therefore be feasible. However, such replacement will inevitably lead to a rewrite of all the associated programs. Use of standards where applicable, and a system-independent format

for the interface files, will aid this work. Very little additional software is envisaged, beyond that purchased from suppliers. Portability is achieved by moving the function rather than the software.

3.2.5 Security

R 2.8 E *A RIDDLE system should be capable of operating within the normal library security procedures.*

Regarding the Communications module, this requirement will most likely conflict with the need for "operator-less" operation (see R 6.9 D). One of the most usual security requirements is for authorisation before allowing file transfer to proceed, using a password. It is considered best practice to require such passwords to be provided by the operator (not built in to a file).

Other than this a RIDDLE system will be able to fit in with whatever security procedures are in use in a particular library.

R 2.9 E *It should be possible to keep records of the work if the organisation so requires.*

It will be possible for an audit trail to be kept to identify the work that has been done and the pages that have been rejected whilst processing. This can be included in the command file envelope.

When a communications system is necessary, file transfer logs can be kept by most systems, although these may be central to the transmission system rather than specific to a particular user. Hence it may require system administration effort to access the data, and special programs to process it.

3.2.6 Safety

R 2.10 E *Appropriate (EC) regulations (such as those covering emission levels) should be followed.*

There are no special regulations relating to the work carried out by the system, and therefore no difficulty is anticipated.

3.2.7 Standards

R 2.11 D *International, national, local and de facto standards should be used where appropriate.*

Standards will be used as applicable. Suitable standards exist and more are being developed to keep pace with the improved technology.

With respect to the image storage format, the TIFF format might be considered a de facto standard. An ISO version of TIFF has been defined and currently has draft status, and is expected to be announced as a full standard later in 1995.

For hardware interfacing, the market seems to converge to SCSI. The greater part of the investigated scanning systems can handle the DIN standard for paper size. They also handle US standards for legal papers, etc.

Text output formats will either be in ASCII or in one of the forms required by a major word-processing package (such as WordPerfect or Microsoft Word).

SGML is an international standard for document markup, and is most appropriate for use in this system. The output from the system will conform to the local OLC library standard.

3.2.8 Resources

R 2.12 D *A RIDDLE system should use hardware and software already available on site.*

A commercial RIDDLE system will likely be a bespoke application, possibly not utilising any third party products. This therefore may make it difficult to meet the above requirement. In reality though, it is possible that existing hardware may be able to be used ie. computer and scanner.

The RIDDLE system along the lines of the pilot, on the other hand, will consist of a set of software packages (eg. an OCR package) together with software to handle the interfaces between these. If suitable packages already exist, these can be incorporated, since a RIDDLE system is not restricted to one particular product or one particular operating platform.

For this type of system, no difficulties are anticipated as far as PC solutions are concerned. For Unix and Macintosh platforms there are also applicable products available. Difficulties might arise where libraries operate on mainframe or turnkey systems. In that case purchase of a low cost scanning system might be considered in order to capture the contents information. In a later stage this data could be transferred to the library systems by means of a communications system.

R 2.13 E *The maximum spending on hardware for a RIDDLE installation should not exceed 17K ECU (15K UK pounds sterling).*

If a commercial RIDDLE system was developed incorporating both hardware and software, then this requirement should be taken into consideration along with other marketing decisions. On the other hand, a pilot RIDDLE system should be able to use a variety of hardware, and it should be possible for a library to find equipment at a suitable price.

A RIDDLE system requires a computer to be available to process the journals. A PC, with a

typical price of 2K ECU, would be sufficient for this purpose.

Apart from the computer, the main piece of hardware required for a RIDDLE system is a scanner. Scanner hardware covers a large range of prices. No problem is anticipated however, in being able to find a suitable scanner to fall within this budget.

There was one OCR package that was investigated, which requires a specific processor to provide the computational power for the recognition engine. This product is aimed at providing high throughput. The cost of this additional hardware starts at around 6K ECU, which, combined with the cost of a scanner, falls within the hardware budget specified for the project.

Extra hardware may be required if provision is made for the visually impaired. The cost of this falls outside the normal RIDDLE requirements.

R 2.14 E *The maximum spending on software for a RIDDLE installation should not exceed 17K ECU (15K UK pounds sterling).*

In a similar manner to the previous requirement, if a commercial RIDDLE system was to be developed this should be taken into account.

It is assumed that suitable operating system software will be available on the computer used to host RIDDLE.

The majority of the commercial OCR software studied falls below 1K ECU. Even if more than one product had to be purchased to cover the required functionality, this total cost would still fall below the maximum expenditure. The cost of commercial image detection software, which may be required for journal identification, should not be more than 2K ECU. The price of a typical autotagger is approximately 3K ECU. SGML translators tend to be less than 1K ECU, and most computing systems will already have tools installed which can perform this task.

It is expected that communications software will already be in place on most machines if required. If this is not the case, then a variety of options exist at different price levels, and therefore no difficulty is anticipated.

R 2.15 E *The maximum amount of training a member of the normal library staff should require in the use of any one of the individual RIDDLE packages should not exceed 1 working day.*

The following information is based on using commercial software packages as part of a pilot RIDDLE system.

Regarding the use of scanners, no hard information is available here regarding the training time required. On being asked, suppliers and manufacturers estimated that for the purpose of RIDDLE, not more than 1 working day would be required. This certainly does not imply that all aspects of scanning, and especially not the high quality colour scanning that can be needed in the desktop publishing community, would be grasped in 1 working day.

Experience with the three OCR packages available to the consortium suggest that training in use of an OCR package will take much less than a day. The most complicated activity will be the set up process, where a new journal is added to the system. While this may require special skills, it is confidently anticipated that the total training required will fit this timescale.

The work of the Image to Text Conversion module is almost completely batch, and so requires no training in its use. Providing the autotagger programs, however, is a specialised task which is outside the scope of this requirement, since it is not expected to be undertaken by a normal member of staff.

R 2.16 E *The maximum amount of training a member of the normal library staff should require in the use of a RIDDLE installation should not exceed 5 working days.*

Based on the information outlined in R 2.15 E, it is not anticipated that the training time for the complete RIDDLE system will exceed 5 normal working days.

R 2.17 E *A RIDDLE system should not need more than 1 operator at any time.*

The design of the RIDDLE system assumes that only one operator will be required.

3.2.9 Timescales

R 2.18 D *Solutions should be sought from existing software and hardware, but advances planned for the next three years should be taken into account.*

Regarding the Scanning module and image capture, no difficulty is anticipated as long as standard interfaces are used and one allows for sufficient disk space and RAM size.

Details of future plans for OCR software are unknown, but history shows that rapid strides in technology and cost/performance have been made over the last few years, and this trend is expected to continue. The investigations have considered the existing market, and have noted that all features required by a RIDDLE system exist somewhere today, even though there is no one system which exhibits every feature. If the current rate of progress is continued, it is confidently expected that this shortcoming will be rectified in the timescale. Progress is also expected in the area of large font size handling.

It has been shown that existing software can handle the needs of the Text to OLC Conversion module. Future improvements in facilities, and reduction in software price, will help.

When analysing possible communication systems requirements, investigations have looked at the future, and included details of both ISDN and Broadband plans. Solutions already exist to meet the requirement.

R 2.19 E *Development time for the pilot system should not exceed the length of the project.*

The pilot system has been developed within the timescale of the project.

4 Scanning System Requirements

This chapter assesses the satisfaction of the requirements identified in chapter 3 of the User and Technical Requirements deliverable. Further background information on scanning can be found in the Scanning Technology Assessment deliverable.

4.1 Capabilities

Most scanning systems are capable of controlling the results of a particular scan by particular settings. These settings include brightness, contrast, resolution, threshold, gamma correction, edge correction (sharpness), speed, zoom, image inversion, scaling, zoning.

An important feature in the framework of the RIDDLE project is the capability of setting zones which can partition a single page into a number of separate areas. Zones can be set either by the scanner software, when the resulting image of the page will only contain selected areas, or by the OCR software, which uses the zones to concentrate on particular areas of the full image presented to it from the scanner. Initially, information was gathered on zoning capability of both pieces of software, but experience led to performing all the zoning needed for RIDDLE in the OCR software and this has been described in the Scanned Image to Text Technologies Assessment deliverable.

An important feature is also the control of the quality of the resulting image from the scanning process. The more the possibilities to tune the resulting image are, the more the flexibility there is to produce an image that is good enough for the OCR process. These capabilities include setting of brightness, contrast, speed, resolution. Setting of brightness is considered essential for the purpose of the OCR process within the framework of the RIDDLE project.

4.1.1 Capacity, Scan Speed, Accuracy

R 3.1 E *The module should be capable of producing a machine-readable image of the contents page of a scientific journal.*

Capacity: Should be capable of handling up to 75 contents page sets a day.

Speed: Should be comparable to that of a standard library photocopier (15 seconds per page) when operated manually.

Accuracy: Should be acceptable to the Image to Text Conversion module.

Regarding capacity (handling of up to 75 contents page sets a day), practically all scanning systems meet this requirement. However using the speed requirement (15 seconds per page)

which stems from the fact that library staff very likely will compare the scanning operation with the operation of a photocopier, considerably less scanners are able to meet this requirement. In fact, from the information available, only one scanner seemed able to meet this requirement for the scanning of a colour page! The colour timings however are not seen as prohibitive since work described in the Scanned Image to Text Technologies deliverable has determined that colour scanning is not normally necessary for the RIDDLE project (ie. it is not required for the extraction of text, but it may be necessary to aid in the journal identification stage).

It must be noted that not all scanners have colour scanning capability, so these do not meet this requirement for a colour page. If one looks at the scanning speed for monochrome images, there are somewhat more scanners that meet the requirement. One must be aware of the fact that the speed figures obtained from information given by suppliers were not established in objective tests. Furthermore these speed figures are raw scanning speed figures used as an indication. If the resulting scan is of poor quality, the time saved by a fast scan will be wasted if time needs to be spent enhancing the resulting image.

With respect to the accuracy requirement it is experienced that in some cases a repeated scan of the original does not produce the same digitized image in terms of amount of pixels. This means that in principle the subsequent OCR process might produce different results, which should be avoided as much as possible. Repeatability depends on the stability of the light source and the amount of noise (signals without information content).

For example, one supplier stated that the accuracy would be 95% to 98% when measuring the results of two consecutive OCR passes using a particular OCR package.

4.1.2 Page Format

R 3.2 E *The scanner should be capable of processing, as a minimum, an A4 page bound in a journal.*

R 3.3 D *The scanner should be capable of processing images up to A3 size.*

Journal issues can have a wide range of paper format sizes. The actual article title information does not cover however an area in excess of A4, as was found out through sampling the journal collections from both the RAL and CWI libraries. There will be no problem finding a scanner that meets this paper size requirement, since all scanners investigated were able to process A4 paper. Some scanners also meet the A3 requirement, although these scanners tend to be expensive. It seems that the price jump corresponds to the A3 capability.

4.1.3 Image

R 3.4 E *The scanner should be able to produce a scanned image suitable for use as input to the Image to Text Conversion module, retaining as much information as required to ensure the accuracy of the conversion.*

Scanner resolution information is given below in section 4.1.4.

R 3.5 E *If required by the journal identification process, the scanner should preserve information on colour.*

For the purpose of the OCR process, the results of the study show that line-art capability is required, which does not of course include colour.

For the purpose of the automatic journal identification, text-based methods also do not require colour. For those journals which require detection by image-based techniques, it is anticipated that line-art images will be sufficient for most. If the additional colour information is required, then it may be sufficient to save it in the form of a gray scale image.

In order to cover all aspects, the study considered both gray scale and colour image capabilities. It is noted that the scanner may well be required for additional purposes outside the context of RIDDLE. In this case, a colour scanner will probably be the better choice.

4.1.4 Resolution

R 3.6 D *The scanner should be able to produce images in a variety of different forms, retaining as much information as possible.*

R 3.7 E *The scanner should be able to produce an image of sufficient quality to be acceptable to the Image to Text module. A minimum resolution equivalent to that produced by Group 3 fax (200dpi) is required.*

All scanners except one satisfied the resolution requirement. Almost all scanners are capable of 300dpi, with some capable of 400dpi. Some scanners use different resolutions for x- and y-axes. For the purposes of RIDDLE, there was found to be no difference in the results between using scanners that used different resolutions for the x- and y-axes, and those that used the same resolution.

4.1.5 Colour Drop

R 3.8 E *The scanner should be able to generate an image suitable for input to the Image to Text Conversion module from a coloured original.*

It is important for the OCR process that the text is readable by this module in good contrast with the background. Samples from both the RAL and CWI libraries showed a large variety of coloured type fonts on a large variety of coloured backgrounds on the contents pages of scientific journals. Due to a phenomenon known as "colour drop", it is possible that a scanner in a certain instance may not "see" a type font because of its colour, and therefore produces a white image of the font. If the background is also white, the type font will in that case not be reproduced on the resulting image. Hence the OCR package will not be able to distinguish this type font and important information may be lost.

Scanners of which it is known that they have no "colour drop" are the preferred choice for the RIDDLE system.

4.1.6 Settings, Separation and Flat-bed

R 3.9 E *If it is necessary to change control settings on the scanner in order to scan a particular journal, it should be possible to save these settings so that they can be applied automatically whenever that journal is scanned.*

As previously indicated, zoning and brightness are important scanning features for RIDDLE. However, zoning requirements will in fact be met by the OCR package. In order to ensure an acceptable working procedure the brightness settings should be saved for each different journal contents page. In that case an operator does not have to optimize the settings every time a new journal is processed.

The majority of scanners possessed software capable of saving the settings for brightness and zoning.

R 3.10 E *The module should pass images to the Image to Text Conversion module in such a manner that the receiving system can associate the complete contents information (possibly spread across several page images) with a particular journal.*

Since every scan is resulting in a file, the question is whether these files can be recognised as belonging to a particular journal. This can be done by giving the files unique identifiers or by associating unique file names to the files.

R 3.11 E *The scanner should be flat-bed.*

Only flat-bed scanners have been considered for the scanning module of the RIDDLE system. There is a wide range of these scanners available nowadays.

4.1.7 Sheet Feeding

R 3.12 D *The scanner should include some feature to help prevent misalignment of the page being scanned.*

R 3.13 D *The scanner should incorporate a sheet feed system.*

In a typical library environment a book support device might ease the scanning operation. This device supports that part of a book or a bound volume which falls over the edge of the desk-top and hangs down the side of the scanner. This ensures that the bindings will not be damaged. This requirement is however desirable rather than essential. Some feature to prevent misalignment is also desirable. Skewed images might be problematic to the OCR process. It is also necessary that there is some form of guide to show where the scan will start on the glass plate. Most scanners have some form of indication to help prevent misalignment. Some scanners that meet the alignment requirement have, for instance, a ruler with paper sizes and alignment indicators in a similar vein to photocopiers.

In case one wants to scan photocopies of contents pages a sheet feeder might be useful. There are libraries that already operate a contents page service based on photocopies. The sheet feed capability may be an additional helpful feature here. However, quality of photocopy output makes it unlikely that a library would choose to use photocopies in place of the original journal.

4.2 Constraints

4.2.1 Hardware Interfaces

R 3.14 D *The scanner should be capable of connection to existing library computer hardware.*

Most are capable of connecting to a PC or Macintosh. Some scanners are intended only for Unix systems, and some of those with PC connection capability can also connect to OS/2. Most Unix connection is effected via SCSI interfaces (see below).

R 3.15 E *The scanner should be connected by commonly used (standard) interfaces (eg SCSI-interface, RS232C-interface).*

The SCSI interface is predominant among the considered scanners. So a scanner with this interface should be the preferred one to choose. Recently SCSI-2 is emerging, fixing a lot of

the compatibility problems with SCSI-1. A minority of vendors use non-standard SCSI.

RS232 is a serial interface probably less common among scanner products, because of the lower throughput stemming from the serial nature.

GPIB is the acronym for General Purpose Instrument Bus. It is also known as HPIB (Hewlett Packard Instrument Bus) or IEEE-488 bus. It is supplied as an additional interface option on some systems.

Some products use only a so called "kofax" interface.

4.2.2 Software Interfaces

R 3.16 E *The scanner system must produce image files suitable for use by the Image to Text Conversion module.*

The TIFF image format is predominant among the considered scanners.

Aldus and Microsoft originally created TIFF in an effort to have an all-encompassing standard among the many de facto standards. It was defined primarily with desktop publishing and related applications in mind. TIFF was foreseen to be used for scanning or painting software and intended for inserting images into documents or publications. An ISO standard version of TIFF should be available later in 1995.

The GIF and PICT formats are also widely used. PICT is closely related to the Macintosh computers.

A "bitmap" file contains image data in a simple format with pixels of 1 bit only.

Besides the above mentioned common formats, there are numerous others which are supported by scanner manufacturers. Therefore finding an image format acceptable to both the scanner and Image to Text Conversion module, is not considered a problem (see R 4.1 E in chapter 5).

4.2.3 Human-Computer Interaction

R 3.17 D *The scanning system should be no more complex to handle than a photocopier.*

It is likely that operating a scanner will be compared by library staff with operating a photocopier, for at first sight the activities have several points in common. Since a photocopying machine can be complex to handle, it was decided to use this as the reference for assessing the complexity of operating a scanner.

In general one could state that operating a scanner by using a keyboard is more complex than operating a photocopier. Having manual switches as well to deal with, would lead to an even

higher complexity. Thus it was decided to keep the operational activities to a minimum and to set parameters where possible by program, from predefined files.

All scanner software appears to have some form of graphical interface, which may follow the "look and feel" of the underlying operating system, but equally may use different techniques. This adds to the complexity.

Graphical interfaces may not yet be common in the library community, but undoubtedly will be predominant in the future. Familiarity with such systems should therefore be considered the norm.

The overall conclusion is that, strictly, this requirement cannot be met, but the additional complexity is not sufficient to give rise to problems in operation.

4.2.4 Availability

R 3.18 D *The scanning system should coexist with existing software.*

The meeting of this requirement was investigated by looking at the size of the scanning software. The results are somewhat ambiguous since "size" could refer to the size of the internal memory, size of the scan control software and size of other software (eg. scanner drivers, interface software). It appeared that this information was not easy to get or to interpret. Unix systems need far more disk space for instance than PC or Macintosh systems. This is mainly caused by the fact that on the Unix systems only the larger packages are implemented (see also 4.2.1). The internal memory must therefore also be greater. It is recommended to ask the supplier what disk space and what internal memory is needed. (PC: min. 2Mb RAM, min. 40Mb disk; Unix: min. 8Mb RAM, > 40Mb disk).

4.2.5 Standards

No particular standards or regulations have been found with respect to scanning systems. It is expected that scanning will fit into existing library procedures without very much difficulty.

4.2.6 Resources

R 3.19 D *It should be possible for members of the normal library staff to correct any failures of the scanning process.*

Attempts were made to find out how much training would be needed for members of the normal library staff to operate a scanner and to correct any failures.

Some suppliers give training, others do not, or do not consider this necessary.

To help the operator correct any failures a diagnostics system would be helpful. Some scanners indicate failures by messages via the software. Some other scanners produce messages and/or error codes on the scanner itself (much like a photocopier).

4.3 Conclusion

The investigations have shown:

a) Scanners exist that meet the essential scanning systems requirements and also meet the common RIDDLE requirements.

b) The speed requirement is satisfied by the monochrome and gray scale scanners. The colour scanners did not satisfy the speed requirement, except one. It depends very much on the journal set of a particular library, as to whether the scanner will be required to preserve possibly needed colour information.

c) It depends very much on the user friendliness of the software whether this can be used by normal library staff. During this study there was no product encountered that determined its optimal settings automatically.

d) As a rule of thumb one should select a scanner which couples the highest speed with the highest image quality (resolution, repeatability etc.) at the lowest price.

e) Most scanners can be afforded by libraries and fall within the budget. Prices tend to be decreasing keeping the same quality or even better.

The current state of scanning technology makes the use of scanners for the purpose of the RIDDLE project feasible.

5 Image to Text Conversion Requirements

This chapter assesses the satisfaction of the requirements identified in chapter 4 of the User and Technical Requirements deliverable. This chapter has been adapted from the Scanned Image to Text Technologies Assessment deliverable, and further information can be found in that document.

5.1 Capabilities

R 4.1 E *The module should be capable of accepting as input for processing, the image file format (or formats) produced by the Scanning module.*

In many cases. the OCR software includes components which will control the scanner directly. This means that the OCR software will receive a bitmap directly from the scanner as it is being produced, and there will be no need to save the image in a file.

For the cases where the OCR software is distinct, or the scanning is carried out remotely, a file containing the image will have to be passed between the two packages. This may also be the case when the page being scanned contains white text on a dark background, and the image has to be inverted prior to the OCR operation. If neither the scanner nor OCR software can invert the image, then the image will have to be passed to some image manipulation software to carry out the task.

Most scanners are able to store images in a variety of graphic file formats, and most OCR packages can accept as input a variety of formats, including TIFF, compressed TIFF and other common image file formats (see R 3.16 E in the previous chapter). It is thus possible to match scanner and OCR software packages so that the output of one will be accepted as input to the other.

The use of a standard such as TIFF will enable image processing software (possibly required at the journal identification stage to search for a logo etc.) to accept input from the scanner.

This requirement is therefore satisfied.

R 4.2 E *The module should be capable of identifying the journal being processed.*

After analysing a subset of scientific journals from each of RAL and CWI libraries, it became apparent that there are a number of technical methods that can be used to identify a journal. These include the use of OCR and image matching techniques. For a particular site, a database will be set up to hold recognition information for each journal, and procedures have been proposed to make use of this information in the identification process.

If the system fails to find a match by automatic means, the operator will be prompted to identify

the journal. This fall-back situation ensures that the requirement will be satisfied, at the expense of some additional operator interaction.

R 4.3 E *The OCR software should be capable of recognising text, in the fonts and font sizes used, from an image (or images) of a scientific journal's contents pages.*

Capacity: A library may subscribe to 2000 journals, with a maximum number of issues per year of 20000.

Speed: The module requires to process a maximum of 75 journals per working day.

Accuracy: The accuracy of this process should be sufficient for the Text to OLC Conversion process to recognise the journal article information elements, and sufficient for insertion of the information into the OLC.

Most OCR products utilise omnifont technology and should therefore be able to recognise a range of non-stylised fonts, at a variety of font sizes. Evidence from the example journal set analysis suggests that all the essential information will be displayed on the page using non-stylised fonts.

The recognition speed of these OCR products varies from 40 to 2400 characters per second (cps). Using this as a basis, it is possible to look at various scenarios, taking a typical example of 81 seconds to process a full A4 page (information supplied by an OCR manufacturer). (It is assumed that most of the time taken by the module will be in the third-party products.)

Maximum Case: The maximum number of contents pages in any one journal of the example set was 4. For journal identification, the time taken by the image handling software that was assessed, was similar to that required by the OCR software (around 1 minute 25 seconds). So it is proper to consider the maximum case as requiring a text-based recognition procedure. Such a journal will require the handling of 2 pages for identification and 5 pages for data capture (assuming some of the data is on the title page). Thus the whole process will take approximately 8.5 minutes. Processing 75 journals of this form will therefore take approximately 10 hours.

Minimum Case: Assuming only one contents page, which is identical to the title page, the above time per journal is reduced to approx 2.5 minutes. Processing 75 journals will then take around 3 hours.

An average set of journals will of course produce timings somewhere between these two extremes. Timings can be further reduced if it is decided to avoid duplication of processing when title page and first contents page are identical.

There is also nothing in the system design to prevent more than one person running a RIDDLE system at the same time, assuming that the final OLC update is performed in a way which will avoid clashes. The elapse time can then be reduced.

Taking everything into account, it is considered possible (working full-time) for an operator to process on average at least 75 journals per day.

All products claim to achieve accuracy in the region 95% to 99% working on a "good original". However, 99% accuracy implies one error every 100 characters, or every three lines of text. This is not a high accuracy rate for a typist, for example. OCR accuracy is required for four reasons.

1. If OCR is being used to identify the journal it is essential that the results are accurate enough that a string match operation can be performed.

2. The recognised text and the preservation of the page layout and text formatting characteristics must be accurate enough for the autotagger to identify the data elements in the text stream.

3. The recognition must be accurate enough for the purposes of loading the article data records into the OLC.

4. The essential information relating to articles must be sufficiently accurate to be accepted within the OLC itself (discussions with potential users suggest that, even here, some inaccuracies may be acceptable).

The effects of inaccuracies can be minimised, particularly if the searching/matching functionality of the respective engines is flexible enough to perform "fuzzy" searches. In addition, the proposed operational procedures include the ability to correct errors after the OCR process.

The capacity and speed requirements are certainly satisfied. The accuracy requirement can be considered fully satisfied if manual correction is allowed. Accuracy of one or two errors per full page can be found in some packages now, and future improvements should produce an acceptable level within a three-year period.

R 4.4 E *The OCR software should be able to recognise mathematical and scientific symbols on the contents pages of scientific journals.*

After carrying out tests with specially prepared pages, it is apparent that OCR packages are not, in general, supplied with pre-programmed mathematical or scientific symbols. It is therefore necessary to select an OCR package that is trainable.

R 4.5 E *The module should be capable of identifying areas of the image to perform the OCR conversion on (zones).*

Most OCR products allow the setting of zones. (See also R 3.9 E in the previous chapter.) A RIDDLE system also requires that the zone information, or template, for a particular journal can be saved in a file to be loaded when processing an issue of that journal.

R 4.6 E *The module should be able to process one or more image files for each scientific journal.*

During the journal analysis, it was found that it was not unusual for several contents pages to be present in a single journal issue. The journal may also have a title page that requires processing. The design of the pilot system has therefore taken account of the requirement to process multiple images per journal issue.

R 4.7 E *It should be possible to produce text from OCR and perform error correction faster than hand-typing the text from the original and correcting it.*

The literature search discovered an article in the SIGCHI Bulletin, April 1990, entitled "Usable OCR: what are the minimum performance requirements" by W H Cushman et al. This paper had been presented at a conference, and compared the results of using OCR instead of human typists. Actual experiments were conducted with groups of typists at various skill levels. The number of errors was recorded, and the time taken to correct these errors included in the comparison. The conclusion was that the OCR out-performed manual input methods if it was able to produce output which was more than 98% accurate. This is well within the capabilities of the current products.

To investigate this requirement further, one of the journals was presented to two typists to obtain estimates for how long it would take to type the data and to proof read afterwards. The journal chosen contains a very dense A4 contents page, and includes subscripts and scientific symbols. It is not untypical of the type of contents page found in a scientific library. A typist would expect to take about 12 minutes to type an A4 page of normal text, but the existence of the unusual symbols will lengthen the time. The more experienced of the two typists estimated that the whole task would take between 30 and 45 minutes, including proof reading and correction. The less experienced typist considered that it would take at least an additional half hour. Taking the timing example for a typical OCR package, which can process a full A4 page in 81 seconds, this means that the text generation phase is at least 6 times faster. This allows an additional margin of at least 10 minutes to add to the proof reading stage to correct any errors the OCR will have introduced, before the OCR process becomes slower than the hand-typing option.

The overall conclusion is that this requirement is considered satisfied.

R 4.8 E *It should be possible to process many journals in batch.*

Batch processing can be divided into two categories: scanning the journals in batch, and performing OCR processing on a batch of images. The only way the former can be accommodated is to use an Automatic Document Feeder (ADF) attachment for the scanner. This option has been ruled out as it would mean that the journals would have to be deconstructed to obtain single pages to gather together and bundle into the hopper of the ADF. It would be possible however to process a batch of images that had been previously scanned.

This requirement is therefore satisfied.

R 4.9 D *It should be possible to defer the processing until a suitable time.*

As mentioned above, it would be entirely feasible to consider a system that scanned all of the journal pages first, then at a later time, processed the images using the OCR software. Use of this facility can be left to each individual site.

R 4.10 E *The module should be able to save the recognised text in a file (or files) for passing to the Text to OLC module.*

All OCR products can store the recognised text in files, using a variety of formats (eg. ASCII or some wordprocessor form such as RTF or RFT/DCA).

Investigations carried out in workpackage 4 indicates that autotagger software can accept input files in a variety of formats, including those mentioned above. This requirement is therefore satisfied.

R 4.11 D *The module should provide facilities for error correction.*

Error correction was taken into consideration during the design of the system. In the case of journal identification, error correction involves the operator telling the system which journal is being processed. During contents page capture, the resulting text can be displayed in an editor to allow the operator to make corrections if required. However, it is noted that some of the automatic correcting facilities often found in OCR packages (such as the use of word dictionaries) are likely to cause more harm than good in a RIDDLE context, and must be disabled.

R 4.12 D *It should be possible to display two A4 sized pages on the monitor at the same time.*

This is a feature of the hardware and operating system present, rather than the RIDDLE system itself. There is nothing in the design of the system which would prevent this. It needs a landscape monitor, and a window-based operating system.

5.2 Constraints

5.2.1 Communications Interfaces

R 4.13 D *If the Text to OLC Conversion operation is not performed locally to the Image to Text Conversion process, then it will be necessary to interface to a communications system to transport the textual information.*

The Communication and Transmission Issues Assessment deliverable contains the analysis of the communications requirements for connecting two remote modules of the RIDDLE system, and transferring text files between them.

5.2.2 Hardware Interfaces

R 4.14 D *A suitable video card and monitor should be chosen to support the display of two A4 sized pages alongside one another.*

As stated above, hardware and software exists outside the RIDDLE system to satisfy this requirement. Whether or not it is provided is up to the discretion of the library running the system.

R 4.15 D *If a communication system is required to transfer the text to the Text to OLC module, then suitable telecommunications hardware should be used.*

The Communication and Transmission Issues Assessment deliverable contains the analysis of the hardware requirements for any necessary communication to the Text to OLC Conversion module.

5.2.3 Software Interfaces

R 4.16 E *The module should be able to accept image files in the format supplied by the Scanning module.*

See R 4.1 E.

R 4.17 E *The module should be able to deliver text files to the Text to OLC Conversion module in the format required.*

See R 4.10 E.

R 4.18 E *The other software components of the module should be able to interface to the OCR package.*

Most OCR products are standalone products, although they generally do provide some way of

being configured or controlled by another process.

5.2.4 Adaptability

R 4.19 E *The module should be able to adapt to new OCR technology as it arises, without affecting the other parts of the system.*

In treating the OCR process as a "black box", it would be entirely feasible to replace one package with another, as long as the interface between the package and the controlling process remained the same. Designing clean interfaces, and the use of standards where applicable, will allow other replacement modules to be included, with only a small amount of reprogramming. None of these changes will affect either the Scanner module or the Text to OLC module.

R 4.20 E *The OCR tool should be able to recognise characters from all Roman alphabet languages.*

All of the products investigated have support for multiple languages. Some products have the ability to use more than one dictionary at a time, and to define new dictionaries.

However, as stated under R 4.1 E, some of the accented characters in Eastern European Roman alphabet based languages cannot be handled, unless the OCR package is trained to recognise the character combination. Such training is feasible, and hence this requirement is satisfied.

5.3 Conclusion

On the basis of the state of current research into OCR techniques, it is possible to satisfy all of the essential requirements, although, at present, there may be difficulty finding a single OCR product which contains all the required features. A RIDDLE system would also benefit from improved OCR accuracy. This does not affect the feasibility study, since the existence of OCR software libraries make it technically possible to construct a suitable package now. It is also expected that these problems will be overcome in the next few years.

6 Text to OLC Conversion Requirements

This chapter, adapted from the Translation of Contents Pages Text to On-line Library Catalogue Format deliverable, assesses the satisfaction of the requirements identified in chapter 5 of the User and Technical Requirements deliverable. Further information can be found in these documents.

6.1 Capabilities

R 5.1 E *The Tagging sub-module should tag each required information item, so that the Translator sub-module can perform the required conversion.*

Capacity: should be able to handle files up to 128Kbits (16Kbytes) in size per journal (equivalent to 4 A4 pages of text).

Speed: should complete its task in under 2 minutes per journal.

Accuracy: should perform its task with minimal errors. Error rates should be less than those expected if a member of the library staff were to type in original text.

The Translation of Contents Pages Text to On-line Library Catalogue Format deliverable describes how such tagging is achieved. There are no specific problems associated with file size. Speed tests carried out on the autotagger available (FastTag) show that it takes between 8 and 11 seconds to process a single page, depending on the complexity of the program. Thus a complete journal will take at most 55 seconds. The accuracy of the tagging will depend to a large extent on the effectiveness of the error correction procedures associated with the previous stage in the RIDDLE process. Tests reported in the deliverable involving "correct" data reinforce the idea that the tagging process can be made almost completely error free if the input file does not contain errors, and can in some instances allow for, and even correct, errors which have been missed before.

R 5.2 D *The option to detect and correct tagging errors on-line should be provided.*

It is possible to consider displaying the tagged text using an editor, but the format does not lend itself to easy reading by a human. It is therefore recommended that error checking in this module is left till the translation stage.

R 5.3 E *The correct selection of template and/or DTD for the current journal should be made with minimum operator intervention.*

The proposed scheme does not require a change in DTD for a particular journal. The autotagger program files can be selected according to the naming convention which will associate the name

with the particular journal. Details of the way in which this can be done are presented in the Scanned Image to Text Technologies Assessment deliverable. This requirement is therefore considered satisfied.

R 5.4 E *The Tagging sub-module should be able to handle non-English characters in such a way that they can be converted to meet the requirements of the OLC.*

It has been assumed that non-English characters have already been converted into a suitable (SGML entity reference) form during the OCR process. Subscripts and superscripts are handled by additional tags. Both can be processed by the translator into a form suitable for a particular OLC (in many cases, the characters are either translated into their English equivalent or are ignored).

R 5.5 E *The Tagging sub-module should be able to handle scientific symbols in such a way that they can be converted to meet the requirements of the OLC.*

These will also have been converted into SGML entity references, and so be in a form which can be handled by the translator.

R 5.6 E *The Translator sub-module should be capable of generating load instructions for an OLC from the output produced by the Tagging sub-module.*

Capacity: should be able to handle files up to 160Kbits (20Kbytes) in size per journal (equivalent to 4 A4 pages of tagged text).

Speed: should complete its task in under 2 minutes per journal.

Accuracy: should perform its task with no errors.

Examples of such translators have been provided in chapter 7 of the deliverable. There is no difficulty in handling files of the size specified. Timing tests show that a typical SGML translator ("Mark-IT") takes 9 seconds to process a journal file, whereas an equivalent program written in a widely available Unix text processing tool ("nawk"), takes only 1.5 seconds. Given that the tagged file is correct, the translation process will not add any further errors.

R 5.7 E *The Translator sub-module should be capable of initiating the OLC update automatically.*

This requirement depends on the capability of the OLC.

R 5.8 E *The Translator sub-module should carry out its function in a manner which complies with the normal procedures for OLC update.*

The output from the translator is a load file for the OLC, adopting whatever format is used at a particular site. Such a file will be processed in the normal way.

R 5.9 E *The output of the Translator sub-module should be capable of being subjected to the same checks as currently used when updating the OLC.*

This requirement is satisfied, since the output format is the same as that used normally at that particular library.

R 5.10 D *Options to perform such checks on-line or from print-out, before and after update, should be provided.*

It is possible to include, as part of the command file envelope, a call to an editor to display the output file in a form which can be changed. Otherwise, the file can be handled in whatever way is usual for that particular library.

6.2 Constraints

6.2.1 Communications Interfaces

R 5.11 E *The module should be able to accept text input from the Image to Text Conversion module. If this module is remote, the system should be able to interface to the Communications system, without operator intervention.*

The Communication and Transmission Issues Assessment deliverable considered the analysis of communications requirements for connecting two remote modules of the RIDDLE system, and transferring files between them.

R 5.12 E *If the work is carried out on a machine other than that running the OLC, it is necessary to provide a connection so that the OLC update can be performed.*

The output of this module is an OLC load file, in text form. Using a communications system, it is possible to transfer this file to a remote machine so that the update can take place. It is also possible, in extreme cases, to consider porting such a file using a floppy disc.

6.2.2 Software Interfaces

R 5.13 E *The module must be able to accept the file formats generated by the module described in deliverable D3.*

Some autotaggers are capable of handling a variety of wordprocessing formats, while others can be made to handle anything which can be read as a stream of characters. This requirement is therefore considered satisfied.

R 5.14 E *The Translator sub-module must be able to generate load commands for the OLC.*

This requirement can be satisfied as long as the particular OLC can be updated by batch file,

or the particular operating system can provide a command language capable of providing file-based input to an otherwise interactive system. If however the OLC can only be loaded by typing at the keyboard, the best that this module can do is to provide a straight-forward listing for the typist.

R 5.15 E *The output from the Tagging sub-module must be acceptable to the Translator sub-module.*

This is ensured by the use of SGML as the intermediate form.

6.2.3 Human-Computer Interaction

R 5.16 E *It should be possible to apply the normal OLC update checks on the entries generated by the Translator sub-module.*

The output from the module will be a batch load file for the OLC. The library can then apply whatever checks are usually done on such modules: either by checking the file itself or by studying the result of the load after execution.

R 5.17 E *It should be possible to provide, if required, an interface to the Translator sub-module to allow manual intervention in the update procedure, including authorisation features if these have not already been provided some other way (such as login password).*

Since this module is called at the end of the journal processing, it is expected that all necessary authorisation will have already been handled. The remaining case concerns the authorisation for the loading of the OLC, and the output from this module will consist of a batch file which can be processed in the same way that existing updates are carried out. This requirement is considered satisfied.

6.2.4 Adaptability

R 5.18 E *It should be possible to add new journals to the system.*

A new journal requires the provision of additional autotagger program files. Adding a new journal to the system is considered in detail in chapter 8 of the Translation of Contents Pages Text to On-line Library Catalogue Format deliverable.

R 5.19 E *The package should be designed so that it is possible for a particular instance of the package to fit in with whatever procedures are currently used in the particular library situation.*

The batch nature of the products mean that there are no specific issues relating to this module which will prevent successful integration with existing library procedures.

R 5.20 E *The package should be designed so that it is possible for a particular instance of the*

package to provide output to whatever OLC system is currently used in the particular library situation.

Use of SGML as the intermediate file form makes it a straight-forward process to produce output for any particular OLC system.

6.3 Conclusion

The investigations have shown:

a) It is possible to define a generic OLC record which can be used as an intermediate form to retain maximum flexibility and to minimise the influence of the different library systems.

b) There exist a small number of presentation styles (both font and layout) which cover the formats used in most of the journals to display the information relevant to RIDDLE.

c) Software exists to capture the important data from the output of the Image to Text Conversion module, inserting tags to match the generic OLC record format. The success of this software depends on the accuracy of the data produced by the Image to Text Conversion module, particularly those aspects which are used by the tagging software for detection.

d) The generic OLC record format can be translated into at least two different library systems. The systems and the methods used are sufficiently diverse to reinforce the expectation that such translation will always be simple.

It is therefore concluded that, subject to the comment on accuracy above, it is feasible to consider a system which can extract text and generate records suitable for insertion into an OLC.

7 Communication Systems Requirements

The RIDDLE requirements for communications are presented in chapter 6 of the User and Technical Requirements deliverable. This chapter has been adapted from the Communications and Transmission Issues Assessment deliverable, and further information can be found in that document.

7.1 Capabilities

R 6.1 E *There should be a network link between the sites wishing to communicate.*

It should be possible to handle up to 75 files a day. The maximum size of text file will be 16Kbytes, generating up to 1.2Mbytes of data a day. The maximum size of image will be 4Mbytes, generating up to 300Mbytes of data a day.

The transmission speed of the network should be sufficient to allow a "same-day" service to operate. The response time of the transmission package should be "immediate" if possible, with a maximum time per page of 2 minutes. A minimum transfer rate of 24Kbps is required in order to handle the maximum number of uncompressed images.

The results of transmission should be error-free. The system should be capable of transferring text files containing encoded non-English characters without change. The system should be capable of transferring files containing wordprocessing features without change. Any compression techniques used should preserve enough information to allow a human to distinguish text when viewing the image. If the recipient wishes to use the image for OCR, suitable compression techniques should be selected.

The widespread existence of the current Packet Switched Data Network (PSDN), together with the future plans for the provision of higher speed public networks, means that it will be possible for the necessary connections to be installed, even if they are not there already.

The information on capacity and speed has been used to size the problem. The limiting factors are seen to be the individual file transfer rates rather than the overall daily rate. There is no problem with handling text transmissions over the slowest systems. The only technology capable of handling uncompressed images is broadband. Otherwise, image compression is required. Even then, PSDN speeds barely cope. An ISDN (Integrated Services Digital Network) connection would seem to be needed, but the level of traffic on the network used must be considered.

The file transfer systems mentioned are implemented over protocols which take steps to detect and correct errors. The OSI (Open Systems Interconnection) protocol set in particular, takes a

lot of trouble with this. Problems over line breaks are less easy to handle. Almost all network links are now digital, and errors tend to be caused by congestion (and hence packet loss) rather than poor signal quality. The situation is likely to improve as broadband services are installed.

The use of an agreed coding standard by both sites should ensure that there is no problem with character sets. File transfer will not alter the characters. However, if there are basic differences (eg. one of the partners uses an ASCII-based coding scheme and the other uses EBCDIC) then translation software will have to be written. Non-English characters will have been encoded in the file already by the OCR system. As long as standard character sets or escape schemes are used, then no additional problems are posed.

It is recommended that loss-less compression algorithms be used unless there is a real need to reduce costs or overcome severe timing constraints.

7.2 Constraints

7.2.1 Communications Interfaces

R 6.2 E *The transmission software used at both ends of the link should be capable of interworking.*

The wide availability of file transfer over such networks as the IP (Internet Protocol) network, means that it should be possible to install compatible systems at both ends.

R 6.3 E *The system should be capable of sending images over international networks.*

No problem is envisaged regarding the interconnectability of international networks, as long as a standard protocol is used.

7.2.2 Hardware Interfaces

R 6.4 E *Hardware interfaces should exist so that the chosen transmission route can be used by the equipment at both ends.*

The general availability of networks means that it is possible to obtain the necessary hardware interfaces.

7.2.3 Software Interfaces

R 6.5 E *Transmission software should be capable of providing an interface to the OCR output at one end and the tagging software at the other.*

The existence of "command files" in the operating systems allows a user to call several commands as if they were one. It is expected that the output from the OCR package will be written to a file, and that the autotagging software will require its input from file. File transfer systems move files from one place to another, and can be built into command files in order to perform any filename matching. The only place where operator intervention may be required is in the provision of passwords (see R 6.14 D).

R 6.6 E *Transmission software should be capable of providing an interface to the scanner output at one end and the OCR software at the other. Image compression and decompression may be required.*

Command files (see R 6.5 E) provide the necessary mechanism for interfacing between the two packages, both of which can handle information in files.

It is not unusual for the scanning software to be capable of generating compressed output without the need for an additional stage.

R 6.7 E *The software should be capable of setting up the network link without operator intervention, consistent with normal procedures.*

Automatic file transfer setup can be achieved by the use of command files (see R 6.5 E).

R 6.8 E *The transmission software interfaces must conform to any interface standards set by the project.*

This is a standard requirement for a project. No difficulty is anticipated.

7.2.4 Adaptability

R 6.9 D *The transmission system should be capable of adapting to the advances in networking. If a choice of transfer methods exists, the one chosen should be capable of scaling up as technology improves.*

The future plans for public networks include the need to retain compatibility with existing systems. The problem of scaling suggests that, today, TCP/IP is a better choice (but it is possible that further advances in OSI will overcome the problem).

7.2.5 Availability

R 6.10 D *Existing transmission facilities should be used.*

If transmission facilities have already been installed, including file transfer, then these should be adequate to handle text. There may be a need to replace the equipment if the transmission speed is not adequate for images.

R 6.11 D *Use should be made of the public networks.*

The European public carriers have facilities capable of performing the necessary work, and are continuing to improve these.

R 6.12 E *Network software should run on available hardware.*

It appears that text transmission is available on almost all hardware. Faster technology (such as ISDN) is becoming common as well.

7.2.6 Resources

R 6.13 E *The system should be capable of handling the transfer without affecting other applications on the same machine to an unacceptable degree.*

This will depend on the current loading of that machine, and can only be determined by benchmark testing in a particular instance. The tests carried out exhibited no noticeable slowdown in response for other users of the system. It is noted that some systems (such as PC/DOS) do not have the facility to run more than one application at a time.

R 6.14 D *Transmission costs should be kept to a minimum.*

This suggests that the files should be compressed as much as possible, since volume is often a component of the charging algorithm. At present, the quarterly rental charge for PSDN use varies from country to country (it is approximately 700 UKP in the UK), with transmission charges by time and by volume. ISDN rental charges are cheaper (84 UKP per quarter in the UK) and transmission charges are the standard telephone ones, but there is a higher initial cost for equipment. Broadband initial costs appear to be of the order of 5000 UKP, but charges for current systems appear to be based on access capability rather than actual usage.

7.3 Conclusion

The investigations have shown:

a) Text transmission systems are widely available, and are currently adequate to meet the needs of the RIDDLE project.

b) Image transmission requires compression. There are systems capable of providing the performance required, but the general facilities seem adequate only for batch (unattended) use. However, the technology does exist to provide the speeds needed, and it is evident that systems will be available within 5 years for use in libraries.

c) Current software costs for non-mainframe products fall within the budget. New technology interfaces are expensive, but are expected to come down in price.

d) There are still some stand-alone library systems which will not provide communications. It is expected that such systems will disappear from the market.

It is therefore feasible to make use of communications within a RIDDLE system, in the manner required.

8 Overall Conclusions

The overall conclusion is that it is feasible to design and produce a system that can automatically produce catalogue entries for the articles held in scientific and technical journals, using technology available today. It is possible to demonstrate this by linking together several existing software packages to form a pilot demonstration system. However, it would not be sensible to construct a production system in this way, since there are some areas where no one package currently has every required feature, and most packages contain additional unwanted features which lead to a larger, slower result than required.

Sufficient variety of hardware and software exists to allow such a system to be implemented on a variety of different hardware platforms.

Cooperation between libraries is possible because of the use of standards. In particular, the use of SGML means that OLC-independent autotagging programs can be constructed which can be used by any RIDDLE site. The provision of an archive of such programs can be considered a possible additional area of exploitation.

In addition the following conclusions can be drawn:

a) Librarians want a system such as RIDDLE. This was expressed by the respondents to the questionnaire during the requirements gathering process. The integration of article information into the standard OLC adds an important additional facility for the library user, since search information need only be entered once.

b) It is technically feasible to demonstrate a RIDDLE system using existing "off-the-shelf" technology, and to consider basing a product on the underlying architecture. It will also be possible to utilise equipment and technology already available within libraries.

c) A RIDDLE system can be a fast alternative to manually typing the catalogue entries. In addition, it provides a timely alternative to information provided by a commercial subscription agency, and one which is guaranteed to reflect the situation in a particular library.

d) The modular nature of the system design makes it possible to consider integrating contents information from other sources.

e) A RIDDLE system can be cost-effective, by using existing library staff, requiring little training. This can be achieved by using in-house resources, especially if any existing equipment can be utilised to provide the service.

f) There are no serious barriers to the exploitation of such a system.