An Approach to Integrated Office Document Processing & Management

NELSON M. MATTOS & BERNHARD MITSCHANG
University of Kaiserslautern, CS Department,
Erwin-Schrödinger-Straße, D-6750 Kaiserslautern, Fed. Rep. of Germany, Mail Address: mattos@uklirb.uucp.

ANDREAS DENGEL & RAINER BLEISINGER
German Research Center for AI (DFKI)
P. O. Box 20 80, Erwin-Schrödinger-Straße, D-6750 Kaiserslautern, Fed. Rep. of Germany

ABSTRACT

We propose an approach towards an integrated document processing and management system that has the intention to capture essentially freely structured documents, like those typically used in the office domain. The document analysis system ANASTASIL is capable to reveal the structure as well as the contents of complex paper documents. Moreover, it facilitates the handling of the containing information. Analyzed documents are stored in the management system KRISYS that is connected to several different subsequent services. The described system can be considered as an ideal extension of the human clerk, making his tasks in information processing easier. The symbolic representation of the analysis results allow an easy transformation in a given international standard, e.g., ODA/ODIF or SGML, and to interchange it via global network.

1 INTRODUCTION

All activities in an organisation require or produce information. Therefore, a document is not only the main information carrier but also a central aid for the integration of office functions /1/. As a part of communication, documents play the central role in todays office domain. The continuing dependence on paper documents as an important information medium and the simultaneous thrust in direction of electronic media require systems, which allow information (structural, content-based) to be exchanged between paper and electronic media. As a result, it will become possible to manage both electronic and paper documents by using a common electronic archive. A given document is characterized by its contents and its internal organization, where the organization is defined by a logical and a layout structure. The elements of the logical structure of a document are constituents like *receiver*, *sender*, *date* or *signature*. Layout objects are titles, text blocks, words or single characters. The framework of the document processing and management system which we are developing, has the following characteristics:

- *Reception of printed information.* Using pattern recognition methods, it is possible to automatically transform printed information into a symbolic representation.
- *Document evaluation.* Applying AI techniques, a document interpretation procedure is initiated. It attempts to identify several layout objects of a document at hand by their logical meaning, thus creating a conceptual structure. Moreover, it provides a restricted context for further content-based analysis /5/. Consequently, an OCR-system is used for partial recognition of textual information within the logical objects. The resulting ASCII-Code is employed to initiate a full text search with keywords in connection with morphological analysis. As a result of the reception and evaluation phases, we obtain different perceptions of a document, namely a layout, a conceptual, and a semantic view.
- *Document management.* To support the processes of reception and evaluation and their corresponding hybrid document representations in an appropriate manner, knowledge representation concepts will be applied. Furtheron, a persistent and efficient document management is a prerequisite to the integration of subsequent services, e.g., document archiving and retrieval, document manipulation (i.e., DTP), and mailing.

In order to accomplish these objectives, our architectural approach towards an integrated document management system (reflected in Fig.1) is consequently based on a *knowledge base management system (KBMS)* /13/ responsible for effective document management and service integration, as well as on an *analysis system* /4/ capable of document reception and evaluation.
Due to the KBMS´s flexibility in document representation, this layered architecture allows for easy extensibility of further services (e.g., archive browser, response generator) under a user-friendly interface.

A Document Analysis System (ANASTASIL) /4/ being developed at the University of Stuttgart and DFKI uses a hybrid knowledge base to support structural and conceptual analysis of paper documents. The task of the underlying document management component comprises efficient and reliable management of the various information structures used or generated by ANASTASIL, or other services. The compliance with these constraints is crucial to the overall system behavior and

© 1990 ACM 089791-358-2/90/0004/0118 $1.50

Permission to copy without fee all or part of this material is granted provided that the copies are not made or distributed for direct commercial advantage, the ACM copyright notice and the title of the publication and its date appear, and notice is given that copying is by permission of the Association for Computing Machinery. To copy otherwise, or to republish, requires a fee and/or specific permission.

efficiency. Therefore, we rely on the KBMS KRISYS /3/ (see Fig. 1) developed at the University of Kaiserslautern. KRISYS offers a rich set of powerful and flexible constructs for object modeling and manipulation /6/, as well as object management.

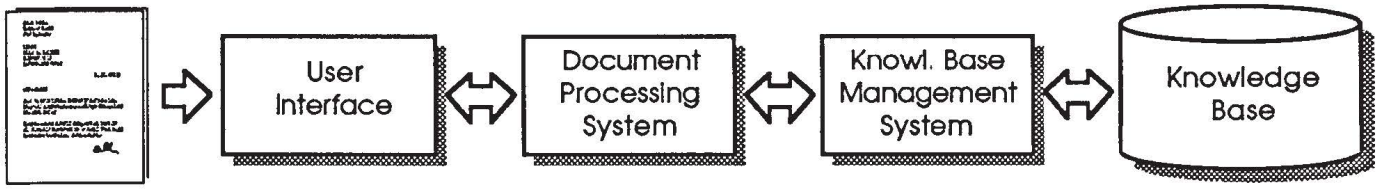

Figure 1: Overall System Architecture

The system architecture of KRISYS is divided into three hierarchically ordered layers which control the stepwise abstraction process and the realization of corresponding tasks within each layer. In Section 2, we point out some features of the knowledge model KOBRA. The application interface of KRISYS is achieved by the query language KOALA, which supports flexible and powerful operations for document retrieval and processing. The lowest layer's goal is to efficiently cope with storage of knowledge structures of KOBRA and its supply to other layers. At this level, most of the issues are related to traditional DB problems applied to large KB, possibly shared by multiple users: storage structures, access techniques, efficiency, integrity features, transaction support, etc. Therefore, this layer is realized by a non-standard database system which seems to be quite advantageous in a KBMS architecture for a number of reasons /13/.

Due to space limitations of the paper's size we concentrate on processing and representation of documents. Moreover, it is not possible to discuss aspects of the NDBS kernel and mapping of documents. The kernel for KRISYS, named PRIMA /14/, offers powerful mechanisms for managing the KB efficiently; among them are storage techniques for a variety of object sizes, flexible representation and access techniques, basic integrity features, locking and recovery mechanisms.

2 DOCUMENT PROCESSING

The task of paper document processing requires the scanning of a given document and the examination of the resulting binary image. The symbolic representation of a document has to capture information about contents as well as about logical and layout structure. After scanning a paper document, a filtering and binarization procedure of the internal document image is initiated. Subsequently, segmentation is performed to establish a document layout representation. The resulting representation of the document page is the input for a highlevel control structure, that attempts to classify the different layout objects as logical objects, like the *subject* and the *date* of a letter, or a specific *company logo*.

2.1 DOCUMENT LAYOUT EXTRACTION

To automatically extract the layout structure of a given document, different phases have to be passed through. They are mainly based on methods of pattern recognition and are more or less supported by knowledge and AI techniques. The phases contain the classification of textual and graphical information, its segmentation in basic and composite layout objects, and their mapping into a data structure, which represents the appropriate layout of the given document.

Layout objects which are text and graphics elements, e.g., characters, words, text lines, text blocks, business graphs, diagrams, company logos are hierarchically nested. The different objects can be described by rectangular regions of text or graphics information. Different results of the preprocessing have to be stored with each object:

- Position and size of layout objects resulting from segmentation processes /8/.
- An ASCII code provided by an optical character recognition procedure /9/.
- Recognition results obtained by the analysis of graphical objects /10/.

2.2 REPRESENTATION OF LAYOUT OBJECTS

So far, we have described how documents are manipulated by ANASTASIL. But how can the knowledge base management system KRISYS be employed to model such documents?

KOBRA, the knowledge model of KRISYS, provides an object-centered representation of the real world /7/. That is, every entity of the application domain is expressed as an object of the KOBRA model, a so-called schema, in which descriptive, operational, and organizational aspects of the real world are integrated. Thus, a schema (not to be confused with a DB-schema) is the symbolic representation of a real world entity (roughly analogous to *frame* or *unit* in other representation systems). It is

always identifiable by a unique schema name and is composed of a set of attributes. The attributes may again be further described by aspects in order to characterize an object in more detail. Attributes are of different kinds. A schema may possess declarative attributes (slots) describing descriptive aspects of an object, procedural attributes (methods) describing operational aspects, and structural attributes (abstraction relationships) used for expressing organizational relationships of the domain.

Layout objects are complex structures composed by other layout objects. For example, a document page is composed by several text blocks, which, in turn, contain several text lines. The latter ones are built of words that are composed of characters (c.f. Fig. 2).

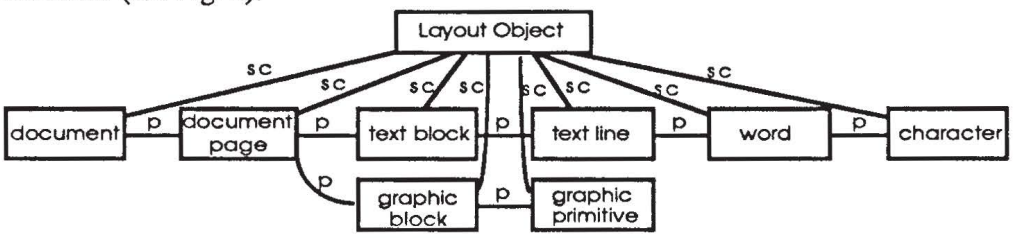

Figure 2: Hierarchy of Layout Objects.

Therefore, all layout objects hold the same information, however, with partially different semantics, i.e., all of them are complex objects built of distinct components. This is represented by means of the KOBRA model as shown in Figure 3. The class *Layout Object* describes the aspects which every character, word, text line, etc. has in common. That is:

x-origin, y-Origin position of left-upper corner of circumscribing rectangle
width, height size of circumscribing rectangle

The different semantics of their components are then specified in the corresponding subclass by means of particular attributes and the aspects *possible-values* and *cardinality*. In addition, instances of object class *Character* and *Graphic-Block* have additional slot variables

ASCII ASCII code as resulting of the OCR-procedure
chain-code internal representation of the original binary image /5/.

```
Layout Objects
has-subclasses (document, document page, ... , character)
                                    terminal ownslot
x-origin                            terminal instanceslot
        possible-values (integer)
        cardinality [ 1 1 ]
y-origin                            terminal instanceslot
        possible-values (integer)
        cardinality [ 1 1 ]
width                               terminal instanceslot
        possible-values (integer)
        cardinality [ 1 1 ]
height                              terminal instanceslot
        possible-values (integer)
        cardinality [ 1 1 ]
```

```
document page
subclass-of (layout-objects)        terminal ownslot
in-document                         nonterminal instanceslot
        possible-values (instance-of (document))
        cardinality [ 1 1 ]
has-text-block                      nonterminal instanceslot
        possible-values (instance-of (text block))
        cardinality [ 1 ∞ ]
has-graphic-block                   nonterminal instanceslot
        possible-values (instance-of (graphic block))
        cardinality [ 0 4 ]
```

Figure 3: Representation as KOBRA Schemas.

It is important to point out the different types of attributes supported by KOBRA:

- Ownattributes (i.e., ownslots and ownmethods), as subclass-of in Figure 3, are used to describe properties of the object itself, and as such may have values.
- Instanceslots and instancemethods, on the other hand, describe properties of the object's instances, and have, therefore, no values (e.g., x-origin, y-origin, width, and heigth).
- Ownslots and instanceslots are further classified in nonterminals or terminals. Nonterminal slots indicate part-of properties (i.e., the components) of objects since their values correspond to other objects of the knowledge base (e.g., in-document, has-text-block, and has-graphic-block). Terminal slots, on the other hand, describe either characteristics of the objects themselves (terminal ownslots) or of their instances (terminal instanceslots).

Therefore, the abstraction concept of aggregation (/7/, /11/) is represented in KRISYS by means of user-defined attributes, allowing for the specification of several kinds of relationships, each of with very fine semantics (observe, for example, the distinct integrity constraint associated to has-text-block, has-graphic-block, which can not be expressed by systems supporting aggregation by means of one single part-of relationship).

2.3 REPRESENTATION OF DOCUMENTS

Documents are represented in the knowledge base as instantiations of the structure previously described. During document preprocessing, its different layout objects are extracted and represented as instances of corresponding classes (see Figure 4). By means of generalization (i.e., class/subclass) and classification (i.e., class/instance) relationships, inheritance is automatically applied by KRISYS, exactly defining the properties with associated integrity constraints of every recognized layout object.

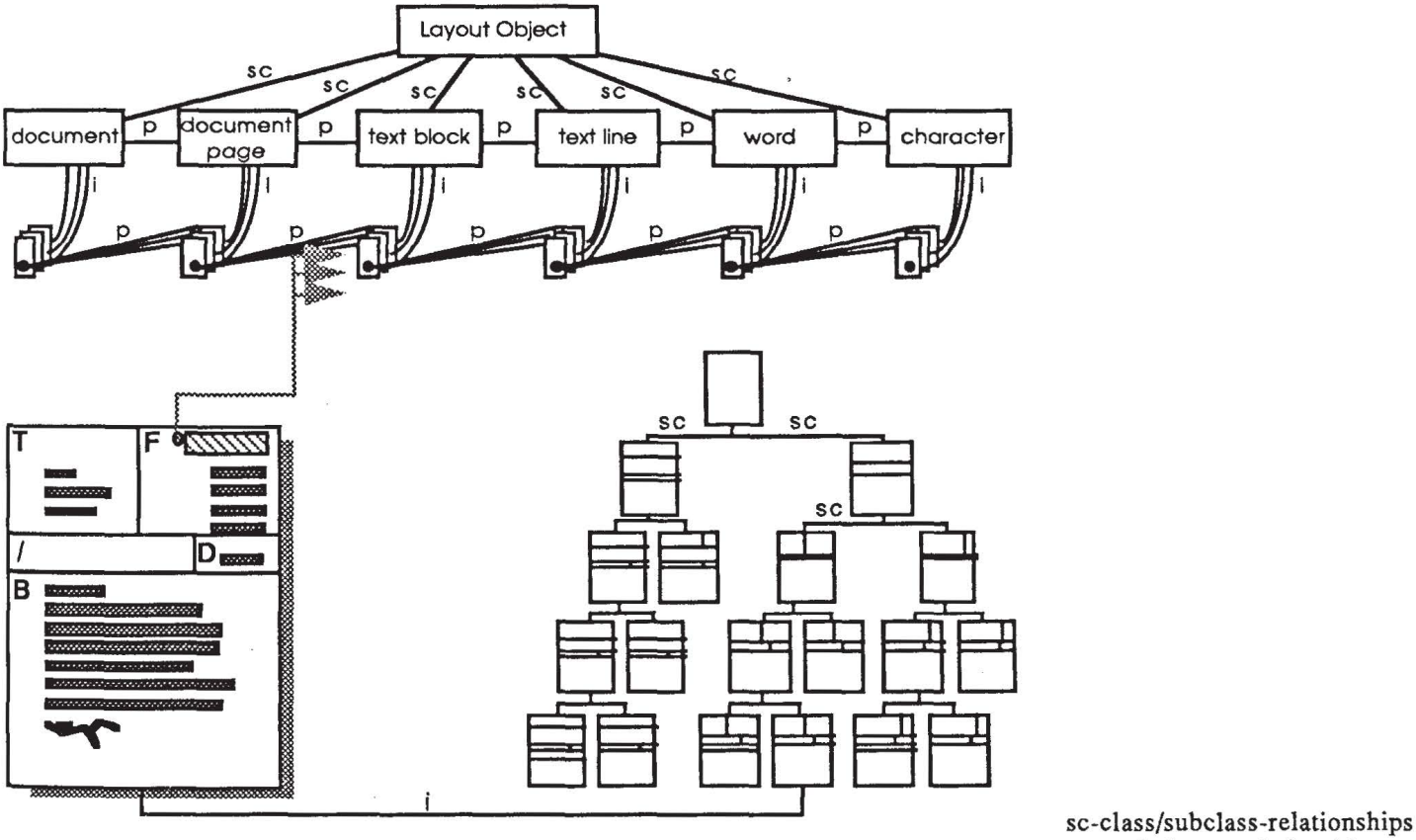

Figure 4: Representation of Layout & Logical Objects and Geometric Tree.

Since the classification process is based on hierarchical analysis of the document structure, it also provides the existing relationships between the several layout objects which are expressed in the part-of attributes of each introduced instance. The KOBRA model provides several built-in reasoning facilities on specified abstraction relationships between objects /6/. Inheritance, as mentioned above, is the reasoning as to the structure of an object applied on generalization/classification hierarchies. Aggregation-relationships are the basis for reasoning with so-called implied predicates (/7/, /12/). For example, the width and height of layout objects must grow upwards. Therefore, the knowledge about the size of a text line may be used by KRISYS either to infer minimum sizes for text blocks, pages, etc., or to control whether specified widths and heigths of layout objects are in accordance to each other.

2.4 DOCUMENT LAYOUT CLASSIFICATION

The goal of the classification phase comprises the assignment of semantics to parts of the layout structure such that essential logical objects of the document, like the *sender*, the *receiver*, or the *footnote* are determined. A starting point for this process comprises formal attributes. Therefore, knowledge as to the possible layout and the composition of information in the document is used. Existing knowledge about document classes, which has been obtained by experience and from empirical tests,

serves as a basis for the execution of this rough analysis. ANASTASIL is based on a tree search /2/. The fundamental tree structure represents knowledge at different layout abstraction levels. The tree is called *geometric tree* /4/ (see Figure 4).

As a result of the classification phase, we obtain a document image in which all important constituents have assigned a logical label. The label indicates a common logical meaning for one or more layout objects grouped together.

In KRISYS, the geometric tree is represented by a generalization hierarchy. Every schema in this hierarchy contains slots corresponding to the labels. They indicate different logical objects of a document (such as sender, receiver, subject, date and body in the case of a letter) as well as their respective layout features within the document. Since a node in the tree is a specialization of its parent node (superclass), as well as a generalization of its more specialized children nodes (subclasses), ANASTASIL exploits the inheritance mechanism provided by KRISYS to support the classification of documents.

The document itself is represented as an instance of one terminal node of the layout hierarchy. Each of the document's logical objects (e.g. *receiver, sender, date,* etc.) are represented as a slot expressing a different aggregation relationship between the document and the existing text blocks. In other words, several text blocks have to be aggregated to build the information about one logical object, as illustrated in Figure 4.

3 SUMMARY

This paper gives an overview of an approach to integrated document processing and management. It mainly concentrates on aspects of the internal representation of documents. Certainly after representing the documents, the system may be used for several additional purposes. For example, we can imagine an evaluation by means of a content-based analysis in order to support further processing steps, e.g., postal systems, response generation. Details about such a service are given in /15/.

REFERENCES

/1/ H. Donner, *Normen schaffen Freiheit im Büro, com* Siemens-Magazine for: Computer & Communications, 4 (1985)

/2/ A. Dengel and G. Barth, *High Level Document Analysis Guided by Geometric Aspects*, Internat. Journal on Pattern Recognition and AI, Vol. 2, No. 4, Dec. 1988, pp. 641-656

/3/ N. Mattos, *KRISYS - A Multi-layered Prototype Supporting Knowledge Independence;* Proceedings of the Internatl. Comp. Science Conf.-Artificial Intelligence: Theory and Applications, Hong Kong, Dec. 1988, pp. 31-38

/4/ A. Dengel and G. Barth, *ANASTASIL: Hybrid Knowledge-based System for Document Image Analysis*, Proceedings of the IJCAI'89, Vol. 2, Detroit, MI, Aug. 1989, pp. 1249-1254

/5/ A. Dengel, *Automatische Visuelle Klassifikation von Dokumenten*, Doctoral Thesis, University of Stuttgart, Computer Science Department, Stuttgart 1989

/6/ N. Mattos and M. Michels, *Modeling with KRISYS - The design Process of DB-Applications Reviewed*, Proceedings of the 8th Interntl. Conf. on Entity-Relationship Approach, Toronto, Canada, Oct. 1989, pp. 159-173

/7/ N. Mattos, *Abstraction Concepts - The Basis for Data and Knowledge Modeling*, Proceedings of the 7th Interntl. Conf. on Entity-Relationship Approach, Roma, Italy, Nov. 1988, pp. 331- 350

/8/ E. Schweizer, *Erfassung, Justierung und Segmentierung von Dokumentstrukturen*, Diploma Thesis, CS Department, University of Stuttgart, 1989

/9/ F. Hönes, *Möglichkeiten der visuellen Erkennung von Worten mit Hilfe von geometrischen Eigenschaften der enthaltenen Zusammenhangskomponenten*, B.Sc. Thesis, CS Department, University of Stuttgart, 1988

/10/ P. Kuner and B. Ueberreiter, *Knowledge-Based Pattern Recognition in Disturbed Line Image Using Graph Theory, Optimization, and Predicat Calculus*, Proceedings of the 8th ICPR, Paris 1986, p. 240

/11/ J.M. Smith and P.C.P. Smith, *Database Abstractions: Aggregation and Generalization*, ACM Transcations on Database Systems, Vol. 2, No. 2, June 1977, pp. 105-133

/12/ A. Rosenthal et al, *Query Facilities for Part Hierarchies: Graph Traversal, Spatial Data and Knowledge-based Detail Supression*, Research Report, CCA, Cambridge, MA, 1987

/13/ N. Mattos, *An Approach to Knowledge Base Management - requirements, knowledge representation, and design issues -*, Doctoral Thesis, University of Kaiserslautern, Computer Science Department, Kaiserslautern, 1989.

/14/ T. Härder, K. Meyer-Wegener, B. Mitschang, A. Sikeler, *PRIMA - A DBMS Prototype Supporting Engineering Applications*, in: Proceedings of the 13th Conf. Brighton, UK, 1987, pp. 433-442

/15/ A. Dengel, N. M. Mattos and B. Mitschang, *An Integrated Document Management System*, Proceedings of the AAI VIII, Orlando, FL, April 1990