AN INTEGRATED SYSTEM FOR PRINTING AND PUBLISHING APPLICATIONS

S. CAVALIERE, M. FANTINI, A.TURTUR
IBM Italy, Rome Scientific Centre

ABSTRACT

This paper describes the architecture of a system, providing an integrated approach to the printing process, starting from the input of the basic components (images, texts, and graphics) up to the definition and modification of page layout and finally to driving high resolution output devices. The system allows an interactive management of the documents to be printed as well as of their elementary components, i.e. the pages, and at a lower level, the boxes, containing texts, images and graphics. The main peculiarities of the system architecture are interactivity, modularity and non-procedurality discussed in more details and since of its functions are illustrated by means of some system outputs.

1. INTRODUCTION

The publishing process, from page layout definition to printing plate production, is usually broken into a number of activities unevenly automated and very seldom completely integrated.
Composition of texts, for instance, is nowadays almost fully automated, while image manipulation is carried out independently and is often based on manual operations.
The definition and composition of a page is managed by the editor, who, on the basis on his own experience, gives the draftsman the page requirements needed to prepare the fool proof of the page to be printed.
Several trials are usually needed and as many proof-readings, before the press-proof is ready. The whole process is therefore long and expensive.

The present paper describes a system, a prototype of which is being developed by the authors at IBM Rome Scientific Centre, aiming at achieving an extensive automation and a complete integration of the above process as far as allowed by available hardware and software technology.
In the next section, a short overview of the system functions is presented in order to give the reader an idea of the provided features. The description of the system architecture follows with a presentation of the hardware and software tools on which the system is based. The image editor is there described in much more details by means of some examples illustrating the available functions. In section 5 the logical data scheme is synthetically reported in order to show how a document and its elements are stored into the database; finally some examples reporting some pages obtained by using the available system functions conclude the paper.

2. THE SYSTEM FUNCTIONS: AN OVERVIEW

The system manages a set of documents, which are collections of numbered pages identified by name and characterised by date and comment.
A page is a rectangle of an arbitrary size, where one or more boxes are located. Boxes are characterised by a type specifying the nature of their content, i.e. images, graphics, or texts.

Documents, pages and boxes are defined and manipulated by means of a number of editors operating in independent functional states. These editors have been described in detail (Cavaliere et alt, 1984). The present paper will present a longer discussion of the image editor in a later section.
The user can interactively define an initial page layout, put in some contents and progressively modify them until he gets the desired result.

Besides operating on the page components, the system creates, manages and updates an integrated database by means of a Data Base Management System (DBMS) (Date, 1977).
The database contains not only image,text and graphics descriptions, but also information concerning the operations performed on them during the several phases of manipulation. This avoids storing intermediate products as digital images, thus allowing a considerable reduction of the required memory.

When a page has reached a satisfactory aspect, two more operations must be performed before plate preparation: first, changes applied to low-resolution page components are to be transferred to high resolution ones in order to fit output device characteristics; second, the image files, generally representing bands in additive chromatic decomposition (Red, Green, Blue), must be converted into the band files required for subtractive synthesis (Yellow, Cyan, Magenta) (Pearson, 1975).

These operations do not need the user's intervention but require a large memory and considerable CPU time. Therefore they are achieved off-line by the system which, moreover, issues messages indicating correct completion of the different steps.

3. SYSTEM ARCHITECTURE

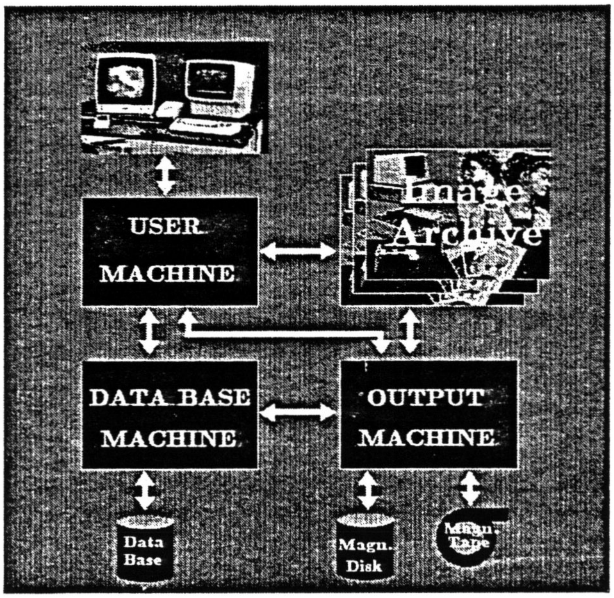

Fig. 1-System Architecture.

Before describing in details the system architecture, we review the concepts underlying the chosen data organisation. Two kinds of data must be managed: image and text files, on one hand, and the formal description of documents, pages, boxes and descriptions of the operations on the other.
The former is characterised by large size and unstructured form, the latter has small size and structured form.

Therefore image and text files my be stored on several different devices, while the document components description is stored in a database, containing also information about the physical devices on which images and texts are stored.

When the user requires to load an image into a box, the system must retrieve from the database the information related both to the box and to the physical allocation of the image. This information is then supplied to a module able to load the image and put it into a box.

On the basis of the above concepts the system architecture (Fig. 1) is built on three functional environments, which, in our case, are three virtual machines (IBM VM/370, 1980): the USER MACHINE, interacting with the user through the IBM 7350 Image Processing System, the DATA BASE MACHINE, and the OUTPUT MACHINE.

3.1 THE USER MACHINE

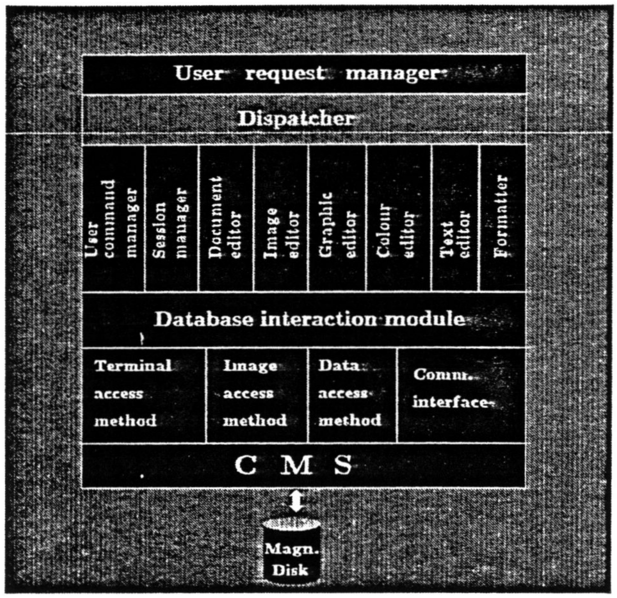

Fig. 2-User Machine.

The USER MACHINE (fig. 2) includes the following modules and subsystems.

MODULES:

User Request Manager

This module provides the full screen management of the alphanumeric display of the work-station IBM 7350 and performs an initial analysis of user information correctness; if some errors are detected, a user intervention is required; when correctness has been checked this module gives the control to the dispatcher.

Dispatcher

Gives the control to the proper subsystem, according to the user's request.

SUBSYSTEMS:

Document Editor

Includes functions for document, page and box management. It simulates on the IBM 7350 color monitor the "cut and paste" operations.
By using its commands the user can, in fact, create or delete boxes, move them on the page, align them, cut and/or expand them, copy them in other positions, etc.

Image Editor

Allows to load images into a box and provides also many important functions such as image windowing, rectangular and generalised cropping, retouching, filtering. Moreover it provides also a photomontage feature among several images. A more detailed description of the image editor functions is reported in Section 4.

Text Editor

Allows the user to edit the text and place it into a box. Moreover changes can be made on selected substrings, in terms of font, size, colour, angle of rotation etc.

Graphic Editor

Provides a set of functions to draw simple shapes, give them a name for later usage and store them into the database.

Colour Editor

Allows to give a name to a colour chosen out of a palette, so that the colour can be referred to simply by name, whenever a colour attribute is required, e.g. for text, box and page background.

User Commands Manager

Invokes the module related to the command added by the user and manages also a simple interaction between the system and the user.

Formatter

Sends commands with proper parameters to the OUTPUT MACHINE to enqueue an output request for a given page.

Database Interaction Module

All the above listed subsystems interact with the DATA BASE INTERACTION MODULE which, on one hand, formulates data base transactions to satisfy the user's requests and, on the other one, processes the results of the previous transactions.

Finally at a lower level the access methods are located to drive the devices used by the system. The access methods include HBUS/VM (HBUS, 1983) for IBM 7350 image processing work-station, GDDM, for full screen management (GDDM, 1982) and a proper image access method to manage the digital images (Cavaliere et al., 1984).

3.2 THE DATABASE MACHINE

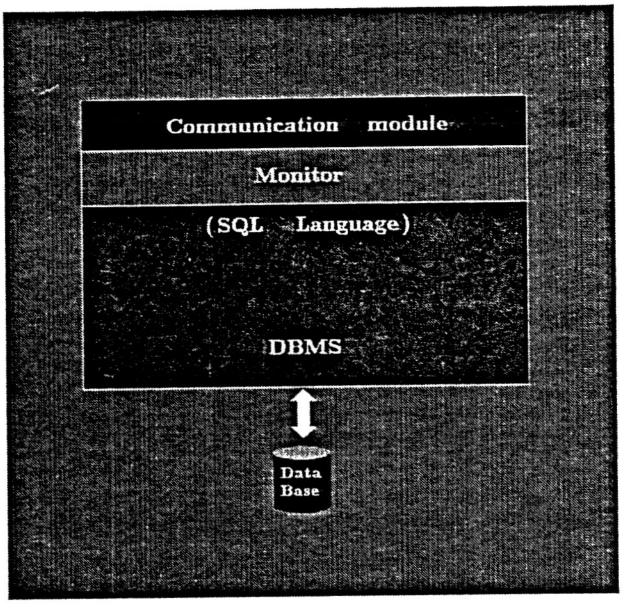

Fig. 3-Database Machine.

The DATA BASE MACHINE (see Fig. 3) contains at the top level the communication module, which can exchange data with the USER MACHINE. At the second level a monitor is placed able to receive the Database statements from the communication module, submit them to the DBMS module (Astrahan et al., 1976). and finally pass the responses back to the communication module The database transactions are formulated in the SQL language (Chamberlin et al., 1976).

3.3 THE OUTPUT MACHINE

The OUTPUT MACHINE architecture (see Fig. 4) contains, in addition to the communication module, the formatter, interacting with the DATA BASE MACHINE in order to retrieve the formal description of the components of the page to be printed (e.g. images, texts, boxes). On the basis of this information the formatter produces the output files for high resolution output devices, e.g. an output scanner.

A final aspect of the system architecture to be underlined is the possibility of including new storage devices to store large amounts of data (images for instance), by adding new ACCESS MODULES.

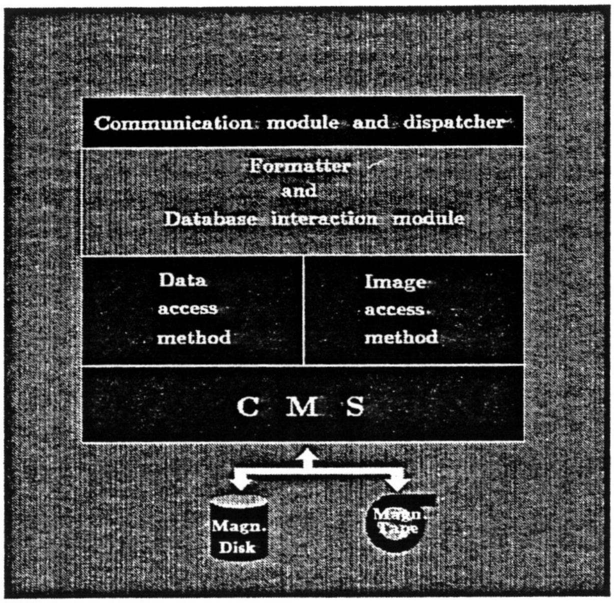

Fig. 4-Output Machine.

3.3 THE IMAGE PROCESSING SYSTEM

The IBM 7350 Image Processing System (P. Franchi, 1983; IBM 7350, 1983) is an intelligent work-station that can be attached via channel to a host system such as IBM 370/xxx, IBM 30xx or IBM 43xx.
It is structured in four subsystems:

The controller
It is able to manage the communication protocol between IBM 7350 and the host computer, and supervises the operations performed by the other subsystems.

The display subsystem
It consists of a colour monitor, a scroll-zoom logic, a cursor logic, a colour look-up table and a series of display controls. The monitor displays 1024x1024 pixels. The colour look-up table has 4096 15-bit words (5 bits for each primary colour component), so 32768 colours can be displayed, although only 4096 different ones can be active at the same time.

The Image Processing subsystem
It is composed of up to 8 1-Mbytes band buffers, an image processing unit, an x/y processor, an interpolator and a histogrammer; it can perform locally several image processing operations at a very high speed.

The Graphics subsystem
It provides the conversion from vector to raster representation, the polygon filling and the character generator features.

4. IMAGE EDITOR

A prerequisite to any digital image processing is image acquisition, that is achieved by using special devices, such as optical scanners. The acquisition process must be usually carried out at high resolution so that in the printing process a high quality reproduction of the image can be obtained by suitable output devices. This generates two problems: first a large file is produced and must be handled, second the digital image cannot usually be displayed on commercially available monitors since their resolution is considerably lower than that of the output equipment.
In order to solve the above problems, the system manages two copies of the same image: a coarse resolution copy is used for interactive manipulation and display, the fine resolution copy is used only when all the composition process has been completed. This implies that the system must store into the database information on the modifications performed on the images. This information is retrieved and modifications are actually performed on the fine resolution copy only when the final output is prepared.

A brief description of the main functions of the image editor is reported in the following paragraph.

4.1 FUNCTIONS

The image editor functions can be divided in three major classes:

A) functions for loading and positioning;
B) functions for colour editing and image composing;
C) functions for image saving;

CLASS A.

This class includes a set of functions allowing the user to place an image into a box in the way he considers the most satisfactory. The user can select a window in the box where to put the image; moreover he can scroll the window on the image, rotate the image, magnify/reduce it according to the scale factor he chooses.
Another interesting point is the possibility of loading several levels of images in the same box. The different levels can be interchanged or suppressed and new levels can be added. Finally, images can be moved or copied from one place to another.

Figure 5 shows some results obtained by using functions belonging to class A. In fact the image of a Madonna by Beato Angelico has been rectangularly cropped at different scale factors. The same operation has been done with the Primavera by Botticelli and San Girolamo by Leonardo da Vinci. Different loading points have been checked.

CLASS B.

The system provides several functions operating on the image colour components: they modify the contrast and brightness of images and balance the colour chromaticity of different images thus performing the some tasks of the usual photographic filtering.

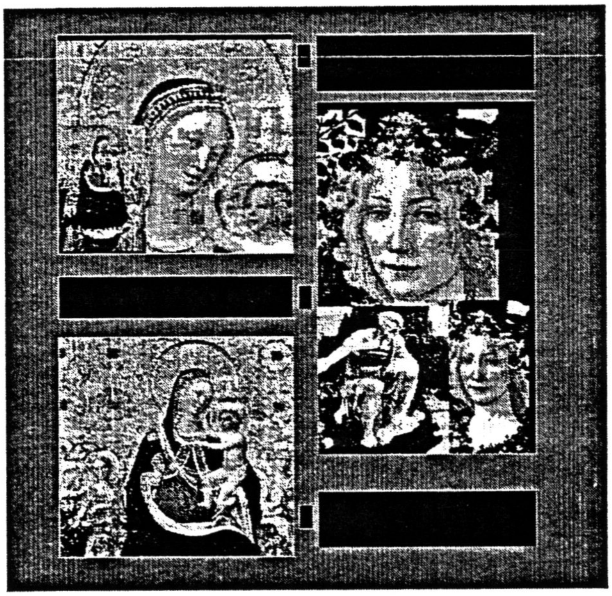

Fig. 5-Class A functions.

Other functions allow the user to make image compositions by using a generalised cropping function. This can be obtained by using two types of contour following mechanism: the first one is semi-automatic, the second one is manual. On a chosen detail, the user can however apply all colour editing functions. Other functions, belonging to this class, give the user the possibility to perform an electronic make-up to modify some details by simulating a painting process. Special effects can be achieved by means of mirroring and shading functions.

Figure 6 shows results obtained by using the above functions; texts, images and graphics have been used. The images have been loaded and scaled, then corrected and balanced chromatically. After cropping, two of them, representing fishes, have been composed with graphics drawn by means of graphic features such as the one generating the background.

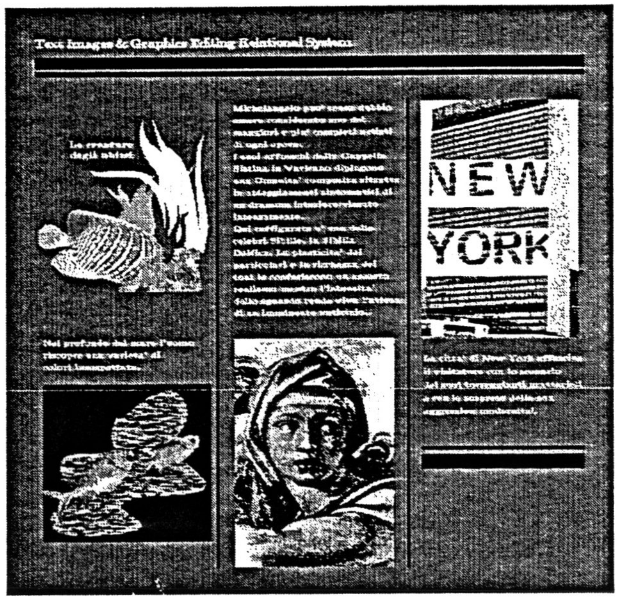

Fig. 6-Class B functions plus graphics and texts.

CLASS C.

The system is equipped with two different functions to save images onto disks. The first one is used to save intermediate versions during an interactive session. It operates on the coarse resolution image and saves a compressed form of it (16 levels per band). The second one operates on the high resolution copy and produces the final version (256 levels per band), that can be used to prepare the printing proofs of the page and, when no further modification is required, the printing plate.

5. DATABASE LOGICAL SCHEME

This paragraph reports a short description of the logical scheme adopted in the design of the database.
All the entities managed by the system and their functional dependencies are organised logically according to the extended relational model (Codd, 1970; Codd, 1979), representing the data scheme in terms of tables or relations.
Figure 5 shows a simplified representation of the above logical scheme: the boxes represent the relations and the lines represent the links among relations.

The "document" is the entity at highest hierarchical level: it is described in the table DOCUMENT. This relation reports for every document its unique identifier, its name, and other information describing the document itself.

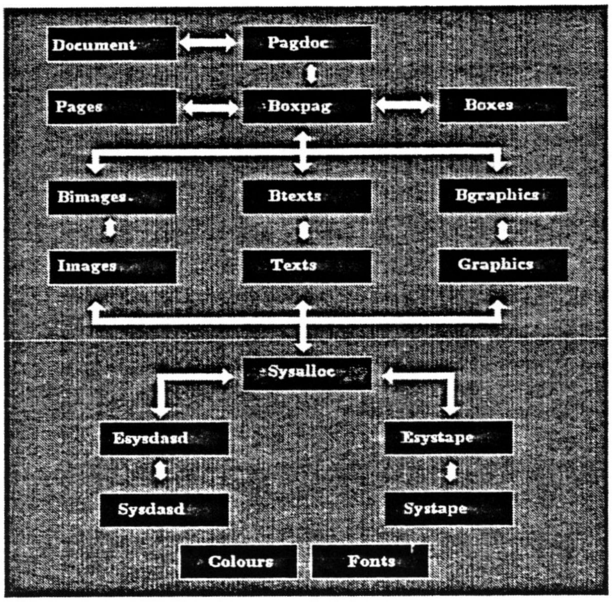

Fig. 7-Database logical scheme.

At an intermediate hierarchical level there are the "pages", listed in the table PAGES, where the page identifier, the page format and other page attributes are reported.
The relationship between pages and documents is represented in the table PAGDOC reporting for each page identifier the corresponding identifier of the document to which the page belongs. Similarly, since a page is defined as a set of boxes with different types, the table BOXES reports the identifier, the type, and all the information describing the box attributes, such as box background, colour identifier, frame thickness, frame colour identifier, etc. The table called BOXPAG specifies the relationship between boxes and pages by reporting for every box identifier the identifier of the page to which it belongs as well as its position in the page.
Moreover since the boxes, as before mentioned, are characterised by their contents (image, text or graphics), three more relations are needed to report the information about such contents. This is accomplished by the tables BIMAGES, BTEXTS and BGRAPHICS reporting respectively: the description of the properties of "image type" box (image identifier, description of image window, description of operations performed on that window, such as retouching, cropping, photo-montage, etc), of "text type" box (text identifier, font identifier, size, text, colour identifier, etc.), and of "graphic type" box (graphics identifier and other information).

The navigation among the BOXES table and the above three tables is managed by using an UGI (Unconditional General Inclusion) type table (Codd, 1979).
Whenever a transaction on a box is to be executed, the system retrieves from the BOXES table the box type, and from the UGI table the relation name, according to the box type, that will be accessed successively.

Finally the contents of the boxes (texts, images and graphics) are described in the proper tables called TEXTS, IMAGES and GRAPHICS respectively, reporting their identifier, name, creation date, comments etc.

UGI

SUB	SUP	PER
BTEXTS	BOXES	"TEXTS"
BIMAGES	BOXES	"IMAGES"
BGRAPHICS	BOXES	"GRAPHICS"
ESYSDASD	SYSALLOC	"DASD"
ESYSTAPE	SYSALLOC	"TAPE"

Information about the physical allocation of the entities is supplied by the relation "SYSALLOC", which reports the identifier of the physical device on which the entity has been stored for each entity identifier known by the system.
Storage devices has been divided into several classes according to the device type. Therefore, on the basis of the information retrieved from the table SYSALLOC, the system finds in the tables ESYSDASD (if the object to be retrieved is on disk) or ESYSTAPE (if it is on tape) the description of the related files, while information on the physical device is reported by the tables SYSDASD and SYSTAPE, respectively. Using another storage device would require the addition of two more tables, one describing the files and the other describing the physical device.

The tables COLOUR and FONTS are dedicated to the description of colour and font entities, respectively.

This kind of data scheme implies more simple formalisation of the functions performed by the system with respect to others types of schemes. In fact all functions can be expressed in terms of data base transactions. For instance the page creation implies the addition of a new tuples to PAGES and PAGDOC tables. The modification of the page background colour implies the update of the related tuple in the PAGES table.

6. EXAMPLES

Figure 8 reports in the left upper corner an image of Toscana taken by Landsat 3. The image is composed of some bands related to different frequency ranges. Three of them have been assumed as RGB bands and displayed. Two details have been enlarged. The scale factors are written on the arrows. Some lines of text have been placed and the attributes of selected strings have been modified.

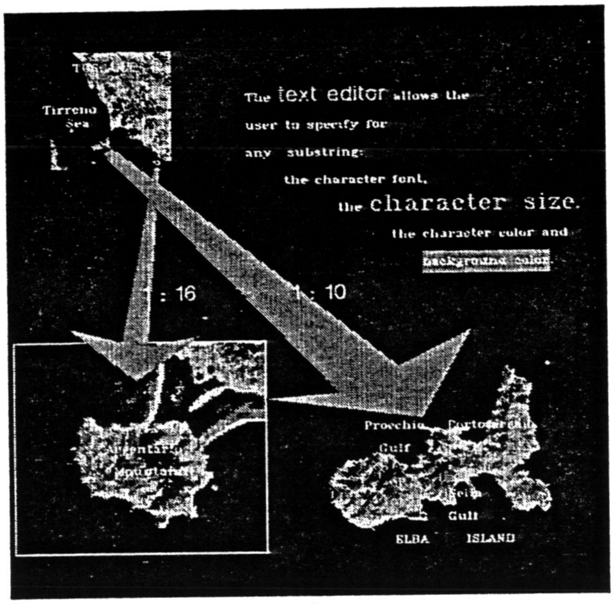

fig. 8-Example 1'.

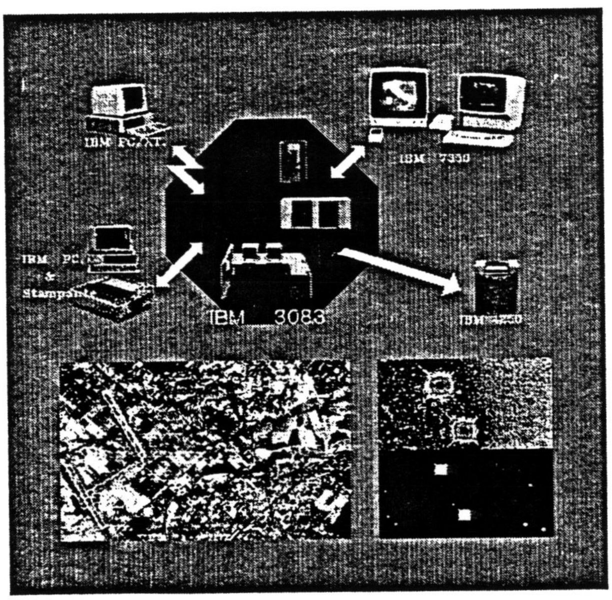

Fig. 9-Example 2'.

Figure 9 contains, on the upper side, a photomontage realized with details cropped out from images of IBM devices. The lower side shows, on the left, a Landast-4 image of the Leonardo da Vinci international airport; on the right, a detail taken from an astronomical plate and displayed in two different ways: direct and false colour.

All the figures of this paper have been prepared by using the system described. They have been halftoned before being printed on a APA printer at resolution of 240 pel/inch.

7. CONCLUSIONS

An interactive system able to automate and integrate the editorial process has been considered. The architecture of the system, characterised by modularity and flexibility, has been described; moreover it has been underlined how the different modules are related to the different tasks of the editorial process.

The logical data scheme, followed to organise the information managed by the system has been reviewed.

The prototype is in the development phase; however the most important functions concerning the document editor, the text editor, and the image editor are available as shown by the reported examples.

8. REFERENCES

1 M. M. Astrahan et alter, "SYSTEM-R: a Relational Approach to Data Base Management" , ACM Trans. on Database Systems, vol.1, n.2,1976.

2 S. Cavaliere, M. Fantini, A. Turtur "Un sistema per la gestione integrata di testi ed immagini" , Proceedings of "Computer graphics, CAD, elaborazione di immagini: sistemi ed applicazioni" (AICOGRAPHICS84), pp. 217-229, Milan (Italy , November, 1984).

3 S. Cavaliere, M. Fantini, A. Turtur "Text, images and graphics management system for editorial environment" , Proceedings of "CAMP 84", pp. 493-496, Berlin (Germany , Sept. 25-28, 1984).

4 S. Cavaliere, M. Fantini, A. Turtur "An Image access method for IBM 7350" , IBM Technical Report, Rome Scientific Centre G513-4053 1984.

5 D. D. Chamberlin, "SEQUEL 2: a Unified Approach to Data Definition, Manipulation and Control" , IBM J. Res. Develop., vol.20, n.6, 1976.

6 E. F. Codd, "A Relational Model of Data for Large Shared Data Banks" , Comm. ACM, vol.13, n.6, June 1970.

7 E. F. Codd, "Extending the Database Relational Model to Capture More Meaning", ACM Transaction on Data Base System, Vol. 4, N. 4, Dec. 1979.

8 P. Franchi et alter, "Design Issues and Architecture of Hacienda, an Experimental Image Processing System", IBM Journal of research and development, Vol. 27, N. 2, March 1983.

9 C. J. Date, "An Introduction to Database Systems", Addison-Wesley Publishing Company, 1977.

10 D. E. Pearson, "Transmission and Display of Pictorial Information", Pentech press, London, 1975.

11 "IBM 7350 Image Processing System System Overview", IBM GA19-5343, available through IBM branch offices, 1983.

12 "IBM Virtual Machine/System Product: Introduction", IBM GC19-6200, available through IBM branch offices, 1980.

13 "Graphical Data Display Manager (GDDM): General Information", IBM SC33-0100, available through IBM branch offices, 1982.

14 "IBM 7350 Host Basic User Subroutines Programming RPQs: Programmer's Guide", IBM SH19-6298, available through IBM branch offices, 1983.