DocMining: A Document Analysis System Builder

Sébastien Adam[1], Maurizio Rigamonti[2], Eric Clavier[3], Éric Trupin[1], Jean-Marc Ogier[4], Karl Tombre[5], and Joël Gardes[3]

[1] Laboratoire PSI – CNRS FRE 2645, Université de Rouen, Place Emile Blondel, 76821 Mont Saint Aignan CEDEX, France
{Sebastien.Adam,Eric.Trupin}@univ-rouen.fr
[2] DIVA Group, DIUF, Université de Fribourg, Ch. Du Musée 3 1700 Fribourg, Switzerland
Maurizio.Rigamonti@unifr.ch
[3] France Telecom R&D, 2 Avenue Pierre Marzin, 22307 Lannion CEDEX, France
{rntl.ball001,Joel.Gardes}@rd.francetelecom.com
[4] Laboratoire L3i, Université de la Rochelle, Avenue Michel Crépeau 17042 La Rochelle CEDEX, France
Jean-Marc.Ogier@univ-rouen.fr
[5] LORIA, INRIA, B.P.239 54506 Vandoeuvre-lès-Nancy CEDEX, France
Karl.Tombre@loria.fr

Abstract. In this paper, we present DocMining, a general framework that allows the construction of scenarios dedicated to document image processing. The framework is the result of the collaboration between four academic partners and one industrial partner. The main issues of DocMining are the description and the execution of document analysis scenarios. The explicit declaration of scenarios and the plug-ins oriented approach of the framework allow to integrate easily new Document Processing Units and to create new application prototypes. Moreover, this paper highlights the interest of the platform to solve the problem of performance evaluation.

1 Introduction

The design of flexible and adaptable document analysis systems is a very complex task implying the sequential ordering and the tuning of parameters for many software components and algorithms. These components correspond to the classical processing steps of a document understanding process, going from low level processing (filtering, physical segmentation, …) to high level understanding techniques (layout analysis, meta-data extraction,…).

Even if some of these techniques may be considered as mature from a functional point of view, their integration in the context of a generic and flexible approach is a difficult task. The heterogeneous representation of the information and the various objectives of document understanding processes make the sequential ordering of these techniques and the tuning of their parameters a cumbersome problem. In fact, the intention of the users can be diverse, going from classical segmentation to performance evaluation, through content-based image retrieval.

S. Marinai and A. Dengel (Eds.): DAS 2004, LNCS 3163, pp. 472–483, 2004.
© Springer-Verlag Berlin Heidelberg 2004

Many domain specific systems have been described in literature [2], and more precisely many application specific systems. Even when the applied strategies are designed to be as generic as possible, the illustrations given for the system are limited to a single category of documents and, moreover, do not develop any quantitative evaluation of significant document databases.

Actually, to the best of our knowledge, no such complete and generic interpretation system exists because of the necessity to have an excellent know-how in the implementation of an interpretation framework. This expertise derives from both the document analysis technique point of view and the domain specific expertise, in order to adapt the analysis scenario to the intention of the document user [2, 3].

These points highlight that many knowledge categories are involved in such a document analysis system, and that a good manner to let the interpretation system be as flexible as possible is to model this knowledge and to separate it from the source code, which contributes to the global flexibility of the approach [3, 5].

Another way to tackle the difficult problem of heterogeneous document analysis is to have a pragmatic approach, consisting in interacting with the user for the definition of all these knowledge and analysis scenarios. This may appear as less ambitious than the approaches that try to implement generic and fully automatic systems, but it appears as more realistic when dealing with a large category of documents, on which a user can be interested in a large number of analysis scenarios, using many processing toolboxes, according to specific intentions.

This paper presents DocMining, a software platform that aims at providing a general and interactive environment for building such document analysis systems. The project is supported by the DocMining consortium, including four academic and one industrial partners: PSI Lab (Rouen, France), the Qgar team at LORIA (Nancy, France), L3I Lab (La Rochelle, France), DIVA Group (University of Fribourg, Switzerland) and GRI Lab from France Telecom R&D (Lannion, France), with the partial funding of the French Ministry of Research, under the auspices of RNTL (*Réseau National des Technologies Logicielles*). On the basis of a set of heterogeneous components, this platform allows us to solve numerous categories of document analysis problems, through a set of strategic aspects that are described in the different parts of this paper.

Section 2 presents the objectives and the foundation of DocMining; section 3 describes in detail the architecture of the DocMining platform; section 4 illustrates three practical examples of scenarios; finally, the last section concludes the paper and presents future perspectives of using the platform.

2 Objectives and Foundation

From the final user's point of view, DocMining appears to be a Document Analysis System Builder, allowing to define multiple document analysis scenarios, on the basis of heterogeneous components, without any specific constraints concerning the components with a great flexibility. From the point of view of the DocMining consortium itself, the platform has allowed us to interface all the processing units coming from the

libraries of each partner in order to assess very different analysis scenarios, such as page segmentation, classification problems, performance evaluation…

The platform relies on three specific ”concepts”:

- Dynamic scenario construction: The principle is to allow the user to interact with the system for the development of a problem adapted scenario, consisting in the ordering of Document Processing Units (DPU is the equivalent acronym in the rest of this paper); DPUs are linked as a function of the user's intentions. Furthermore, running a scenario collects user experience, which becomes part of the scenario itself. The scenario may then be transformed into a new DPU corresponding to a higher-level granularity. Here, we can find similarities with electronic circuit construction devices, such as VHDL for instance.
- Document centered approach: All the knowledge – data structure and DPU parameters – that is extracted during scenario execution is “archived” in the document. Its structure evolves as a function of the intentions and interactions with the users… Thus, the document actually appears as a communication channel between DPUs that are run by the users. Such an approach avoids the problems of data scattering usually met in classical document processing chains.
- A plug-in oriented architecture: The principle is to permit the integration of heterogeneous software components. As a consequence, developers can conveniently add new processing units, thus making the platform easily upgradeable. Document visualization and manipulation tools are also designed according to this approach, so that a user is able to fully customize the interactions with the document structure.

As a consequence, DocMining is a platform that allows to quickly and easily solve any new problem, on the basis of a set of heterogeneous processing tools. But the advantages of this approach exceed the mere simplicity of implementing an image processing chain, since the document centered approach enables the user to share knowledge about parameter tuning, results and analysis scenarios. This advantage may appear very important in the context of collaboration between different research teams, such as in the DocMining Consortium. Two practical examples are that the DocMining platform offers features allowing to compare easily different strategies of document understanding or to share the know-how concerning the utilization of a particular processing tool.

Actually, DocMining does not aim at representing the different categories of knowledge that are implicitly involved in a document understanding process, but offers a pragmatic environment helping the user to interactively and explicitly integrate this knowledge:

- Document specific knowledge, including the representation of the objects, structural/syntactic information, threshold concerning image processing techniques, …
- Strategic knowledge, which deals with the processing chain (scenario), built when solving a particular problem.
- Processing knowledge with parameter tuning, which remains a very difficult problem when considering a new category of document, in a specific context.

The DocMining platform has been experimented in the context of various problems (document segmentation, document vectorization, pattern recognition, performance

evaluation, …) and with different documents (structured documents, graphic documents, databases, …). It uses various software components stemming from the respective processing libraries of each partner in the DocMining Consortium (DIUF's xmillum framework, Qgar library, PSI library, …). Some of these experiments will be presented in section 4. In the following section, we will present the architecture of the DocMining platform.

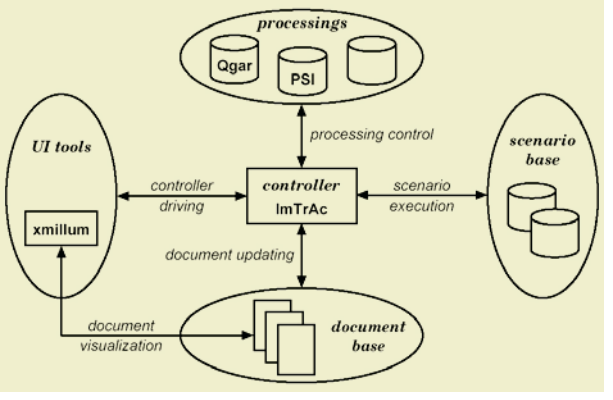

Fig. 1. Overview of the DocMining platform architecture.

3 DocMining Platform Architecture

3.1 Architecture Overview

The concepts presented in section 2 have led to the design of a platform for building and applying document analysis scenarios. This platform, called DocMining, relies on the architecture illustrated in figure 1. The platform contains five main components:

- The document itself, comparable to a blackboard containing the current extracted data.
- An extensible set of Document Processing Units, which works such as the "specialists" in the blackboard paradigm.
- A scenario, containing both DPUs and fine grained components designed for various purposes, such as data adaptation between the document and DPU, parameters observation, DPUs repetition…
- A scenario engine called ImTrAc that interprets specific elements of a scenario, controls DPUs execution and document updates.
- A set of user interfaces, assisting the construction of new scenarios and visualizing intermediate and final results.

3.2 The Document

Our approach is focused on the document, which can be considered in a first approximation as a set of graphical objects. The document centralizes the major part of the

information involved in the processing and is progressively enriched by the DPUs. In order to store all this information, documents are represented by an XML tree that describes the graphical objects. The XML format guarantees the user the possibility of extending the set of already-defined objects. The DocMining consortium defined an XML tree structure standardizing the graphical objects. The tree is specified in an XML schema (figure 2a), which defines some primitive elements that can appear in the document (GreyImage, BinaryImage, Bloc, TextLine...), their attributes (position in the image for example), and their source. The schema is not exhaustive and can evolve with new data types, according to user needs. It is associated to a JAVA API, proposed in order to manage the objects of the structure. This library enables us to load, create and save object instances.

The next sub-section shows how to define a DPU, the main issue for analyzing and enriching a document.

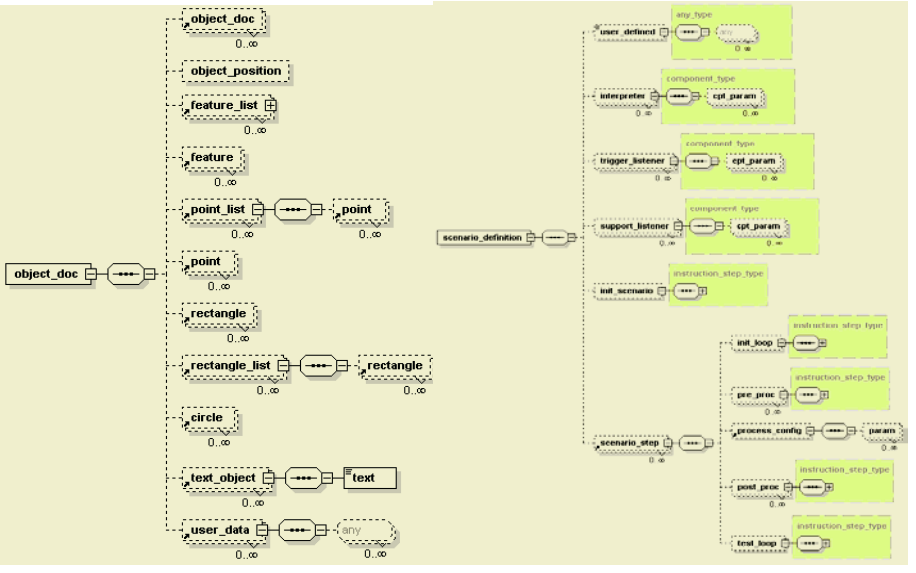

Fig. 2. XML schemas defining the structure of a document(a) and the scenario structure (b).

3.3 Document Processing Unit

One objective of the DocMining consortium is to enable interoperability between the different processing libraries proposed by research teams. In such a context, five main processing libraries have already been included in the platform: The Qgar library, the PSI library, the JAVA Advanced Imaging, a DocMining classification library and a DocMining feature extraction library. These libraries contain various components, such as image processing tools, classification tools, structural/syntactical operators, and so on. The goal of this sub-section is not to describe the libraries already included in the platform [2], but to illustrate how to integrate new DPUs.

In order to be included in the platform, a unit has to respect two contracts. The first one is a declarative contract that describes four elements:

- The offered service, describing effects of the DPU on the XML tree (creation, modification or fusion of data).
- The target data of the DPU. This parameter defines the conditions (i.e. the XML elements to be present in the document structure) that allow the DPU to be triggered.
- The DPU parameters characterized by their name, their type, their support, their possible values and their default value.
- The resulting objects.

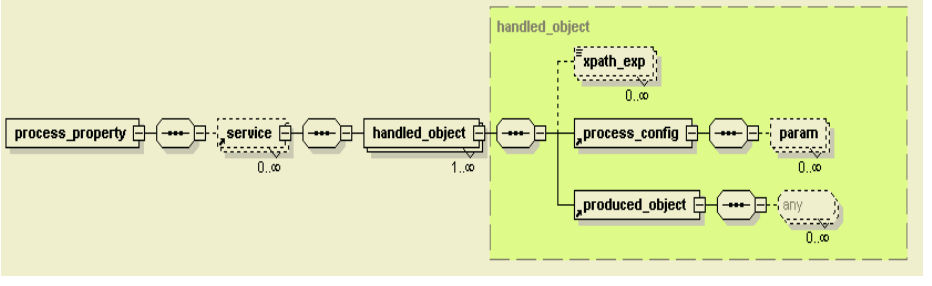

Fig. 3. XML schema for the contract definition.

The contracts must respect the XML schema in figure 3. Thus, the knowledge concerning DPUs can be made explicit. Such knowledge encapsulates both the behavioral and operational aspects. According to us, such a model of processing is oriented towards explicit knowledge in document analysis and library sharing.

The second contract is software oriented; it consists of a set of Java interfaces that direct the user in the encapsulation of algorithms into a DPU.

A sequence of DPUs is involved in the construction of the scenario; its structure is presented in the next sub-section.

3.4 The Scenario Structure

The notion of scenario is another main issue of the DocMining platform. A scenario consists of a set of elementary actions that are run sequentially on part of the objects in the document. The DPUs presented in the previous sub-section are not the only authorized actions: The scenario includes much functionality, allowing us to integrate heterogeneous software components and to offer a “high” programming level to final users:

- Triggering. It is run automatically at each step of the scenario. For instance, the trigger allows to isolate some intermediary information related with the running DPU: Time processing measures, global variable evolution following, …
- Support adapter. It provides the capability of adapting a particular dataset to a DPU. The support adapter works as an interface between DPU and dataset.

- Instruction. This notion corresponds to code that is interpreted, thus offering the user simple scripting capabilities…
- Variables, allowing the exchange of data between the other components provided by scenario.

The XML schema illustrated by figure 2b declares the concepts, which define a valid scenario. The scenario engine, presented in the next sub-section, interprets these concepts.

3.5 The Scenario Engine

The scenario engine, called ImTrAc, interprets and coordinates a well-defined scenario, through the sequential ordering of available DPUs. It acts the role of a mediator between the document and DPUs. It parses the scenario and schedules the involved units. Firstly, ImTrAc loads the document and the scenario; then, it checks their validity with respect to the XML schemas. After the variable and components instantiation, it analyses each scenario step: This starts detection of the activation nodes into the XML tree. Thus, a sequence of instructions consisting of pre-processing, DPU, post-processing and triggers is executed. This sequence may be repeated many times in the same step. Finally, ImTrAc searches for the next step defined in the scenario and repeats the operations previously described. The fully execution of the scenario produces new data, thus enriching the document. The next sub-section presents the interfaces involved in the creation and validation of scenarios.

3.6 User Interface

Two User Interfaces have been integrated in the DocMining platform (figure 4). The first one, called SCI (Scenario Construction Interface), aims at providing a user-friendly assistance in order to build new scenarios. The second one, called xmillum, is a framework for visualizing and interacting with the produced data. Although the two user interfaces can be used separately, they communicate together, allowing the user to watch, correct and tune intermediary results of the scenario construction process.

3.6.1 The Scenario Construction Interface

The SCI tool allows the user to build new scenarios. For a given element of the document structure, the SCI tool is able to supply the list of DPU that may be applied. After the user chooses a unit, the SCI tool supplies its parameter list so as to be able to launch the corresponding process. The SCI tool has two modes for DPU execution:

- The emulation mode that checks the contract, in order to determine which object is produced by a DPU. The document is updated according to the provided service and the produced object declared in the contract.
- The execution mode actually applies the DPU to the selected elements by calling the ImTrAc engine.

After each processing step, the document structure is updated and the user can interact with the newly created object. During the entire scenario construction, the SCI tool ensures the coherence and the validity of its structure.

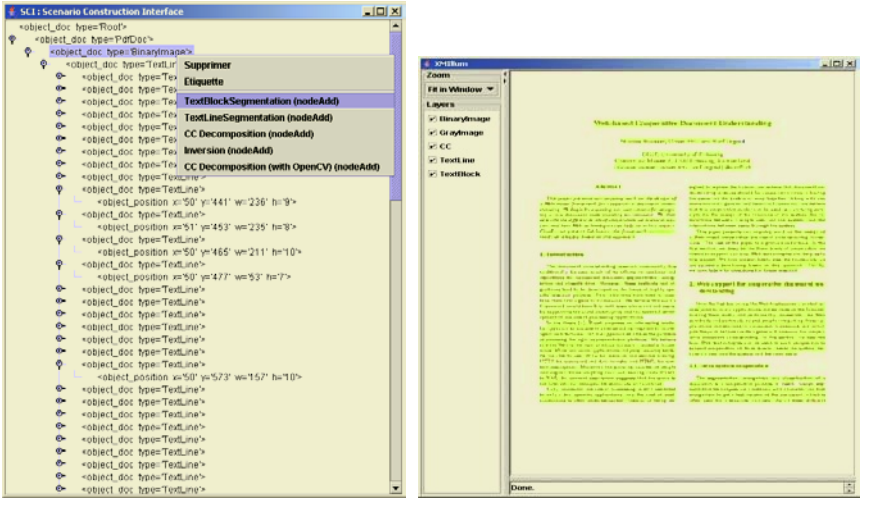

Fig. 4. Scenario construction with SCI and results visualization with xmillum. SCI allows to build new objects and xmillum allows to reorganize the produced objects.

3.6.2 xmillum

xmillum is a framework for cooperative and interactive analysis of documents. It offers an interface that uses external modules to visualize and to interact with a document. The framework has been developed to work with documents represented in XML. The aims of xmillum in the DocMining platform are:

- Visualization of documents. The interface of xmillum uses a background layer to represent the original document and semitransparent layers to visualize the new views, which contain the result of the DPUs.
- Interaction with the document. The user can interact with xmillum in order to manipulate the original document or the extended views of it. Interaction is useful for validating or correcting the results of DPUs. Another issue of the interaction is related to the definition of ground-truthed data, which allows to parameterize DPUs defined in the scenario chain.

  The next section presents examples of interactive analysis of documents.

4 Three Performance Evaluation Use-Cases

The DocMining platform has been experimented in the context of various problems dealing with different documents, using various software components from the respective processing libraries of each of the DocMining consortium partners, or from other contributions. In this section, we describe three practical cases: a scenario of segmentation evaluation, another for pattern recognition evaluation and a last for text-graphics separation.

4.1 Segmentation Evaluation Scenario

Performance evaluation of algorithms has become a major challenge for document analysis systems. In order to choose the most valuable algorithm according to the domain or to tune algorithm parameters, users must have evaluation scenarios at their disposal. But efficient performance evaluation can only be achieved with a representative ground-truth dataset. This application tackles the two aspects of performance evaluation, the construction of a ground-truthing scenario and the definition of a segmentation evaluation procedure.

4.1.1 Ground-Truth Dataset Construction Scenario

PDF documents serve as basis for our ground-truth dataset. Indeed, the PDF format is widely used in many applications (newspaper, advertising, slides, …) and PDF documents can be easily found on the web. Moreover search engines enable the user to refine a search according to the document format. So it is very easy to build a PDF document base where many domains are represented. A ground-truth dataset can also be built with newly created PDF documents based on the transformation of XML documents. With these two approaches, we can build a document base which contains "real life" documents obtained through an internet search and "problem specific" documents built from an XML source. Nevertheless, the PDF format has a major drawback: It is based on a pure display approach, so that structural and logical information is not directly accessible; this information must be computed from the low level objects contained in the PDF document. So we built a scenario that partially constructs the physical structure of a document. The ground-truth dataset is obtained through a three step scenario:

- Select the PDF document.
- Parse the PDF document and partially extract the physical structure. We developed a DPU, based on the PDF parsing API of the Multivalent package [6], that extracts content end location of letters, words and text lines. Additional elements such as images are also extracted.
- Save the generated ground-truth structure. The document structure is saved according to the XML schema we defined.

4.1.2 Benchmarking Scenario

The second part of the application consists of evaluating segmentation algorithms for the raw dataset. This benchmarking scenario is composed of three steps:

- Transformation of the PDF document into an image. The DPU we designed encapsulates a ghostscript command.
- Physical structure extraction using a page segmentation processing. At this time, we have two segmentation algorithms, one based on a classical top down approach and the other one based on a hybrid approach [4].
- Segmentation performance evaluation. Segmentation algorithms produce a resulting XML structure, which is matched with the ground-truth dataset to measure the regions overlap ratio. Ground-truth information is extracted from the dataset by using

Xpath expressions, which allow to select the desired corresponding structure. XPath expressions give great flexibility to the user to select exactly what he needs. He can modify these selection expressions by choosing another kind of object (words for example) or by adding constraints (for example small areas may be filtered). Node matching itself is done with Yanikoglu's method based on the ON pixels contained in a zone [7]. In order to ignore insignificant differences between the ground-truth regions and the segmented ones, only the black pixels content of the areas are taken into account.

Although many page segmentation evaluation problems are not yet addressed in this experimentation, DocMining shows its ability to tackle many aspects of ground-truthing and benchmarking. Its modularity can help building such a scenario according to users needs. Moreover it allows users to design their own performance evaluation algorithm. For more information about this application, one can refer to [1].

4.2 Pattern Recognition Evaluation Scenario

This second application deals with recognition on "chicken dataset" composed of 5 classes and issued from the IAPR TC5. This dataset is composed of binary images associated with features corresponding to contour coding. Such as in the first use-case, the goal is not to demonstrate the superiority of an approach, but to show the advantage of using the DocMining platform to compare results of various approaches. In this context, additional tools for features extraction (Fourier-Mellin invariants from the PSIlib, Hu moments from the openCV Library) and classification (API from PSIlib) have been implemented, that demonstrate the interoperability of the platform. In the scenario, three parts have then been constructed:

- The knowledge base construction, which is first constructed with correctly labeled images from the dataset. References to these images are included in the document at the first step of the scenario. Secondly, each symbol of the dataset is associated with its contour coding. This step is done through a specific DPU constructing a structured description of all the symbols and by using XPath expressions.
- The new features addition part, where the symbol description is completed with a new set of features (Fourrier-Mellin invariants and Hu moments). A new specific DPU has therefore to be applied to enlarge the features database describing the symbols. It can address each image referred in the Document and simply add to its description new structured information as an XML fragment. Whereas the features data increase, their manipulation is not more difficult because the document is the only entry for their access. The knowledge base is then complete.
- The classification part, which consists in creating learning and test bases from the dataset, and classifying samples from the test base. The bases are randomly generated and their contents are simply labeled without any physical cut of the knowledge base. The samples of the test base are then classified using A KNN operator, to finally score the recognition process. The data formats are automatically adapted to the treatments thanks to XPath expressions. Performance evaluation consists in comparing results of recognition and input labels for the symbols.

In this application, the whole classification chain has been implemented using an image dataset and some pre-defined features. We have shown in this section that it is quite easy to enlarge the feature set by applying other feature extraction tools, and various specific treatments. This enlargement adds new features, which can be in various representation formats because they can be adapted to the classification-processing tool.

4.3 A Scenario for Mixed Text/Graphics Documents

The third application that is presented here performs the analysis of documents with mixed text/graphics content. Its aim is the separation of the graphical part from the textual part. The steps of the scenario are ordered as follows:

- Binarize the image,
- Perform text-graphics separation,
- Perform segmentation into text blocks on the textual layer image,
- Perform segmentation into connected components on the graphical layer image,
- Perform segmentation into connected components on the parts that have not been classified as graphics or text,
- Correct text-graphic separation errors by a cross-examination of the results of these three segmentations,
- Perform OCR on the text blocks of the resulting textual layer image,
- Perform vectorization on the resulting graphical layer image.

The main DPUs available to run the scenario are described in [2] and the construction of this scenario relies on a structural combination of these DPUs. It has been made possible by associating each processing with a contract describing its behavior. This contract includes processing instructions as Xpath expressions, which are interpreted by the ImTrAc engine. ImTrAc extracts the resulting element set from the document and transmits it to the processing. Such a scenario may be used for different purposes.It may become part of a more general document interpretation system.It may also be used to evaluate the robustness of an algorithm in case of noisy input images (leading to text-graphics separation errors). Finally, it may be used as a benchmarking tool: When a developer implements a particular step of the scenario, he may run the different available DPU to evaluate the efficiency of his implementation.

5 Conclusion

DocMining is a multi purpose platform and is characterized by three major aspects. At first, its architecture relies on a document-centered approach. Document processing units (DPUs) communicate through the document itself; such an approach avoids the problems of data scattering usually met in classical document processing chains. Second, the DocMining framework is based on a plug-in oriented architecture. Developers can conveniently add new DPUs, thus making the platform easily upgradeable (for example in order to process color documents). Document visualization and manipula-

tion tools are also designed according to this approach, so that a user is able to fully customize the interactions with the document structure. Third, the platform handles scenario-based operations. Running a scenario collects user experience, which becomes part of the scenario itself. The scenario may then be transformed into a new DPU, corresponding to a higher-level granularity. Thus, the DocMining architecture is really modular because a user can create his own objects, integrate his own DPUs into the platform, design his own interfaces, define and run his own scenarios.

More than a software environment, DocMining must therefore be considered as a general framework for integrating various tools, DPUs and systems. The aim of the framework is to be flexible and general enough for building various kinds of document analysis systems (documents can be black and white, grey or color)– actually, we have shown that it can even be extended to more general pattern recognition systems.

We are especially convinced of the usefulness of the DocMining framework for setting up and running performance evaluation and benchmarking campaigns or contests. As shown in the use-cases presented in this paper, it is fully possible to easily build scenarios including taking into account some ground-truth, computing performances by using some metric, or even matching the recognized entities with the ground-truth – in this case, the matching would simply be another DPU to be integrated in the chain. Work remains to be done on defining a general "roadmap" to build such evaluation campaigns, but we hope to be able to offer the power of our software framework, in the coming months and years, for the benefit of various performance analysis campaigns within the pattern recognition community.

References

1. E.Clavier, P.Héroux, J.Gardes, E.Trupin. "Ground-truth production and benchmarking scenarios creation with DocMining". 3rd International Workshop on Document Layout Interpretation and its application DLIA2003, Edinburgh, Scotland, August 2003.
2. E.Clavier, G.Masini, M.Delalandre, M.Rigamonti, K.Tombre, J.Gardes. "DocMining: A cooperative platform for heterogeneous document interpretation according to user-defined scenarios". International Workshop on Graphic Recognition GREC2003, Barcelona, Spain, July 2003.
3. B.Coüasnon. "DMOS: A generic document recognition method. Application to an automatic generator of musical scorers, mathematical formulae and table structures recognition systems". Proceedings of 6th International Conference on Document Analysis and recognition ICDAR2001, Seattle, USA, 2001.
4. P.Parodi, G.Piccioli. "An efficient pre-processing of mixed-content document images for OCR systems". *13th Int. Conf. On Pattern Recognition*, Vol. 3, pp 778-782, 1996.
5. B.Pasternak. "Adaptierbares Kernsystem zur Interpretation von Zeichnungen". Dissertation zur Erlangung des akademisch Grades eines Doktors der Naturwissenschaften (Dr. rer. nat.), Universität Hamburg, 1996.
6. T.A.Phelps, R.Wilensky. "The multivalent browser: A platform for new ideas". Document Engineering 2001, Atlanta, Georgia, USA, 2001.
7. B.A.Yanikoglu, L.Vincent. "Pink panther: a complete environment for ground-truthing and benchmarking document page segmentation". Pattern Recognition 31, pp 1191-1204, September 1998.