DOCUMENT ARCHITECTURE STANDARDS EVOLUTION

Debra Ansen

***Debra Ansen** is a member of technical staff in the Applications Standards Planning department of AT&T Bell Laboratories in Holmdel, New Jersey. Ms. Ansen has been with the company since 1985 and is responsible for document communication architecture and standards, with special attention to document interchange and imaging. She has a B.A. from the University of Chicago and an M.S. in computer science from the City College of New York.*

The *Office Document Architecture* (ODA) is a set of standards for structuring and encoding documents so that they can be interchanged between dissimilar systems, viewed as intended by the document originator, and processed by other users.[1] ODA is a joint effort of the International Organization for Standardization (ISO) and the International Telegraph and Telephone Consultative Committee (CCITT). The standards describe documents in terms of their structure and content. Structures are hierarchical and sequential. Content is separated and processed separately from document structures. Three types of content are permitted: text, raster graphics, and geometric graphics. Rules for maintaining relationships between document elements are also defined. There are plans to extend the ODA document processing model and to develop standards for new content types such as voice and spreadsheets.

Introduction

Information exchange in office environments is often accomplished through the use of electronic documents. Many offices have a variety of document creation systems and text formatters manufactured by different vendors and using different document representation schemes. In addition, documents can include a variety of contents such as:

- Text
- Bit-mapped images
- Vector graphics
- Voice
- Spreadsheets.

Documents with such a variety are known as *compound documents.*

The variety and complexity of documents can impede document transfer. To address this problem, ISO and Study Group VIII of CCITT have jointly defined a set of standards for document inter-

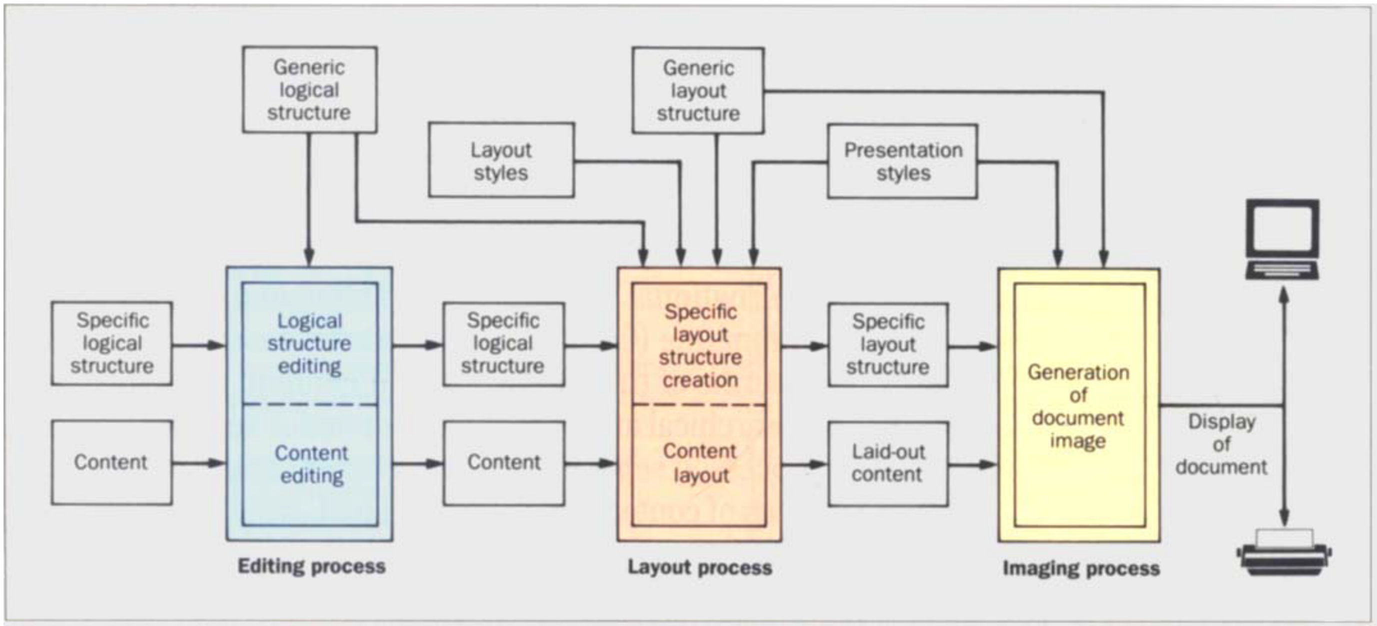

Generic logical structure
Generic layout structure
Layout styles
Presentation styles
Specific logical structure
Content
Logical structure editing
Content editing
Specific logical structure
Content
Specific layout structure creation
Content layout
Specific layout structure
Laid-out content
Generation of document image
Display of document
Editing process
Layout process
Imaging process

Figure 1. Document processing based on the ODA model.

change known in ISO as *Office Document Architecture* and in CCITT as *Open Document Architecture*.[1,2] Experts from both organizations worldwide have spent more than five years developing the standards.

ODA is designed to facilitate document interchange in an *open systems interconnection* (OSI) environment.[3,4] Basically, ODA allows a document to be presented as intended by its originator and permits processing such as editing and reformatting. The standard describes the following:

- A document architecture—a content-independent set of rules for defining document structure.[5]
- Content architectures—rules applicable to the presentation of different content types.[6,7,8]
- An interchange format for document representation (known as ODIF, for *Office Document Interchange Format*).[9]

Structures. A document has a hierarchical and sequential order and can be divided structurally into different components. These components can be further subdivided based on logical ordering or on the way in which the components are laid out. Structures based on logical order are known as *logical structures*; those based on what they physically represent in the document are known as *layout structures*.

Logical structure associates the content of a document with a hierarchy of logical objects; paragraphs, chapters, and figures are represented in logical structures. Layout structure associates the same content with a hierarchy of layout objects; pages, frames, and blocks are represented in layout structures.

Both structures are independent of the types of content in a document. ODA provides a general description of the properties of these structures so that they can be used to represent specific documents.

Objects. The components of a document are referred to as *objects*. Two types of objects are defined: *composite* and *basic*. Composite objects can be further

Panel 1. Key Terms in This Paper

Compound document A document that contains multiple content types. A memo that contains text, graphics, and tables is a compound document.

Object The components of a document; objects can be either composite or basic.

Content The information, other than structural information, that is conveyed by a document and is intended for human perception.

Editing process Creating and modifying a document; involves logical structure editing and content editing.

logical structures—structures based on logical order; can be either generic or specific.

structure editing—creating or deleting an object, changing its position in the structure, and modifying attributes associated with that object.

content editing—adding or deleting text and graphic elements and control functions.

Layout process Generating a layout structure and laying out the content; includes nested and overlaying frames and blocks on sets of pages; allocates the content of a document to such blocks.

layout structures—structures based on what they physically represent in a document; can be either generic or specific.

content layout—positioning the text and graphic elements within blocks.

layout styles—rules that guide the layout of objects and not their content.

presentation styles—rules that affect the layout and imaging of the contents associated with basic logical and layout objects.

Imaging process Creating an image of a document on a presentation medium.

subdivided into other objects; basic objects can not. With this approach, a document can be represented in a tree structure. Basic logical and layout objects are leaf nodes that act as containers for document content and can contain only one type of content.

Document Processing Model

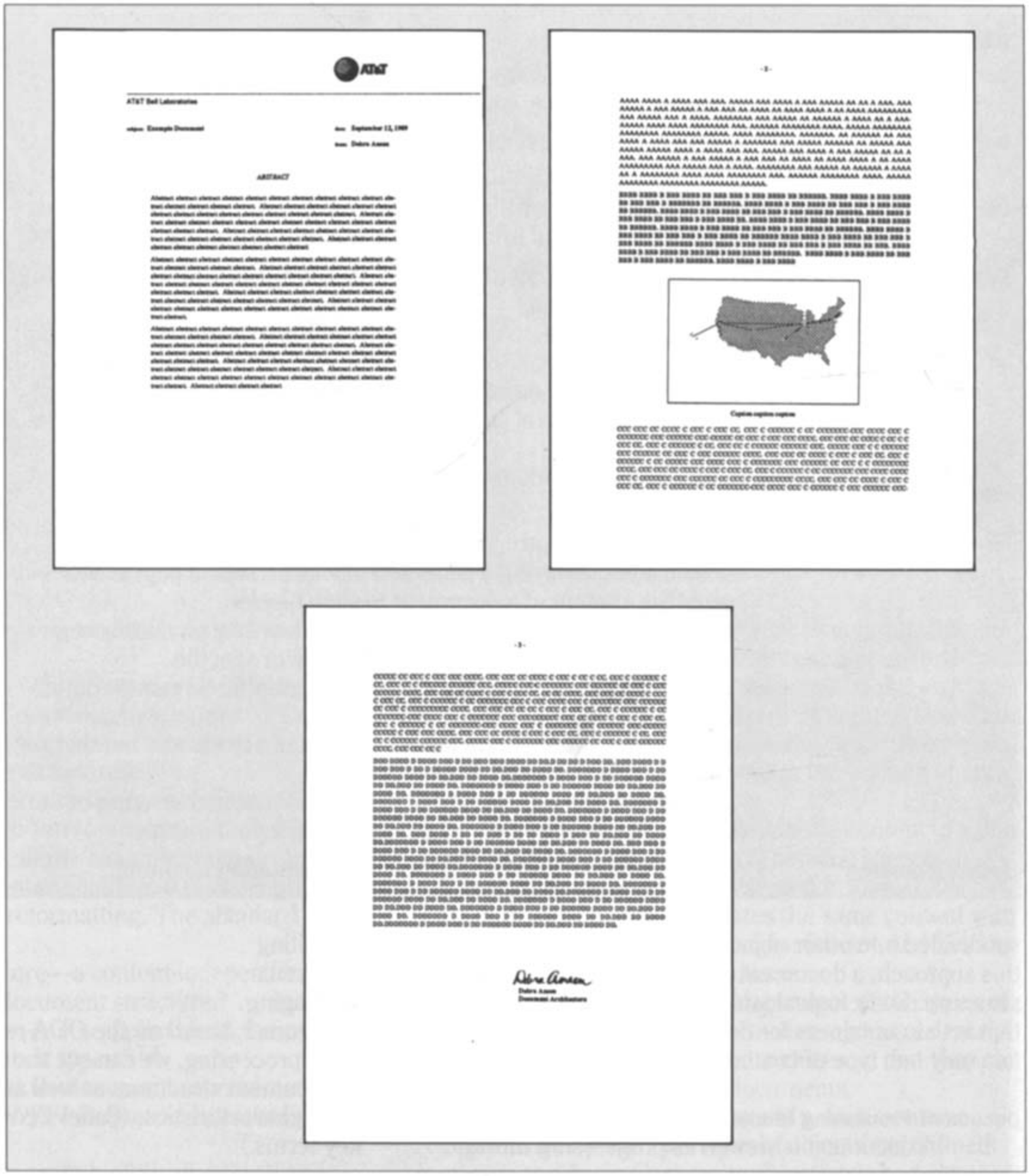
A document is viewed as progressing through three phases of processing:

- Editing
- Layout
- Imaging.

In Figure 1, based on the ODA reference model for document processing, we can see the function and application of document structures as well as the semantics of document characteristics. (Panel 1 contains definitions of key terms.)

Generally, different people are involved at differ-

Figure 2. Sample document.



AT&T Bell Laboratories

subject: Example Document

date: September 12, 1989

from: Debra Aronsen

ABSTRACT

Caption caption caption

Debra Aronsen
Document Architecture

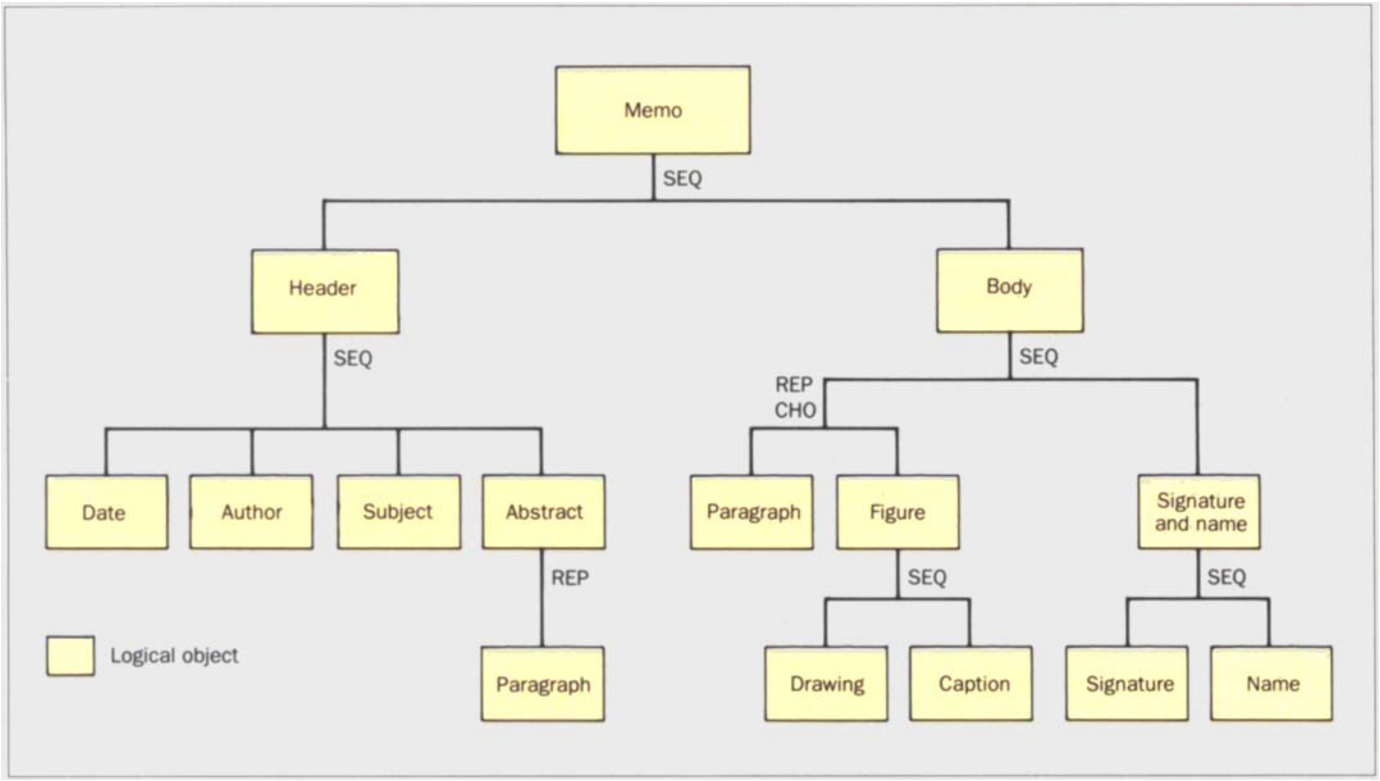


Figure 3. Generic logical structure for the sample memo shown in Figure 2. SEQ = sequence of objects; REP = repetition of objects; CHO = choice of objects.

ent times in the production of a document. The author of a document is primarily responsible for its content and, in part, its organization. This stage of document handling is known as the *editing process* (shown in blue) and is used for creation and modification of a document. It involves logical structure editing and content editing. Structure editing includes creating or deleting an object, changing its position in the structure, and modifying attributes associated with that object. Content editing can include adding or deleting text as well as adding or deleting graphic elements and control functions. The results of the editing steps are the input for the layout process.

The appearance of a document is usually determined by typesetters and graphic artists, and often is based on a set of criteria established for a particular type of document. This phase of document processing is known as the *layout process* (shown in red) and involves generating a layout structure and laying out the content. A layout structure has nested and overlaying frames and blocks on sets of pages and allocates the content of the document to such blocks. The content layout process positions the text and graphic elements within these blocks. The result of the layout process is the input for the imaging process.

During the *imaging process* (shown in yellow), an image of the document is created on a presentation medium (e.g., on paper or on a screen). This process is not defined in the ODA standard and is regarded as a locally defined process that depends on the presentation device used.

Document Architecture

All ODA documents consist of a document profile and a document body.

Document Profile. A document profile is a set of attributes associated with the document as a whole.[10] It has reference information about the document, such as the author's name and the document creation date, and also defines global information about the structures and content types in the document. A document profile can be sent to other users without sending the actual document. This may be beneficial for a recipient who is uncertain whether there is a need to receive a document or who wants to determine whether a certain system can provide an image of a document.

Document Body. The document body consists of structures and styles. Both logical and layout structures are of two types: generic and specific (see Figure 1). Generic structures are analogous to context-free grammars that define the syntactic structure of languages and provide structure for program translation into object code. ODA generic structures define rules for the generation of logical and layout structures that are specific to a particular type of document. They define common objects in a specific class of documents such as a letter, memo, or report.

A Sample Document. Figure 2 is a sample business memo. Figures 3 and 4 show possible *generic logical* and *generic layout* structures for that memo. The generic logical structure (Figure 3) represents the organization of the document around a conceptual sequence of classes of elements with similar characteristics. Paragraphs occur repeatedly in the document; however, the number of paragraphs is not specified in the generic logical structure. The generic layout structure (Figure 4) is organized on the basis of sequences (shown as SEQ) and repetitions (shown as REP) of page classes, frames, and blocks.

Basic objects in generic structures often have content associated directly with them. The logo for the sample memo always has the same layout characteristics. Thus, its content is associated with the generic layout structure logo block (shown in blue in Figure 4).

Specific structures describe instances of documents. They are particular to a given document and can be thought of as parse trees that are constructed by a compiler. The hierarchical syntactic structure of a document that is implied by its generic structure is made explicit in the specific structure. Figures 5 and 6 show the *specific logical* and *specific layout* structures for the sample business memo. Every component of the document is denoted in a specific structure and basic objects of specific structures always have content associated with them. The boxes shown in blue in Figures 5 and 6 represent the actual content associated with each basic object.

Generic and Specific Structures. Figure 1 shows the relationship of generic and specific structures to each phase of document processing. Generic and specific logical structures are input for the editing process and specific logical structures are the output.

The layout process takes generic and specific logical structures and generic layout structures as input and produces a specific layout structure that can be used in the imaging process. If the specific logical structure of a document is all that is interchanged, a document can be displayed, but not edited.

Layout Styles and Presentation Styles. Document styles specify sets of attributes and are divided into two types: layout style and presentation style. Layout style specifies relationships between the logical and layout structure that are interdependent and cannot be changed. Examples include the requirement that an illustration and its caption must appear on the same page or that a chapter must begin on a new page. Presentation styles

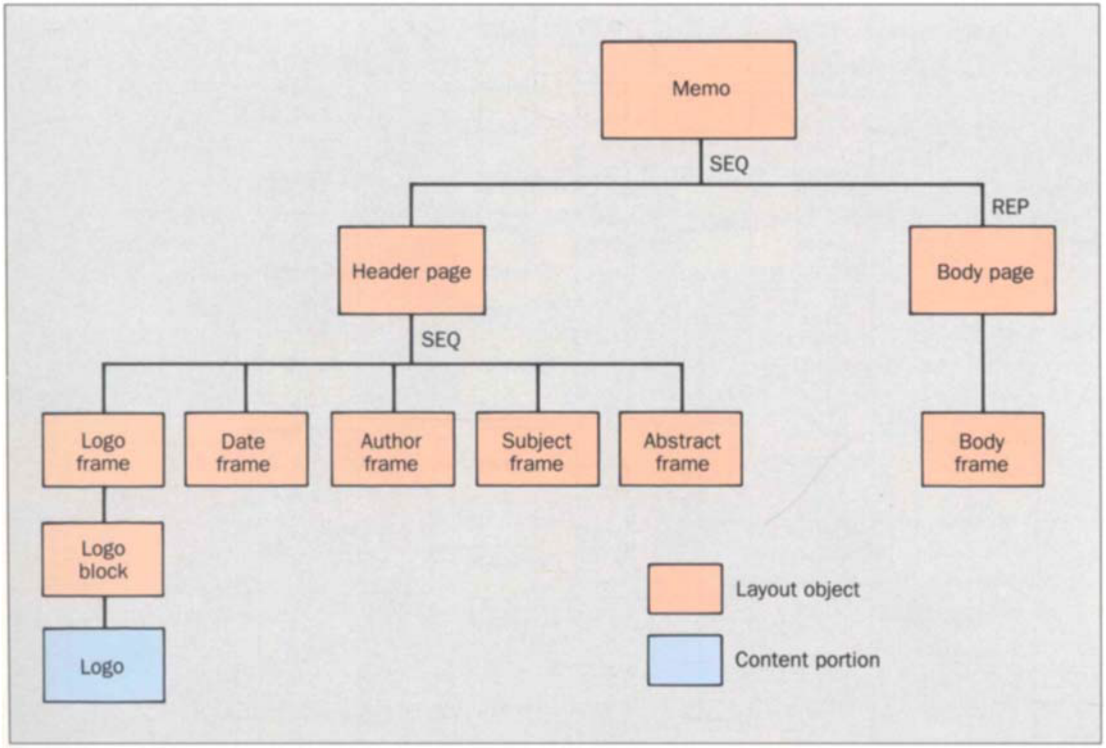


Figure 4. Generic layout structure for the sample memo shown in Figure 2. SEQ = sequence of objects; REP = repetition of objects.

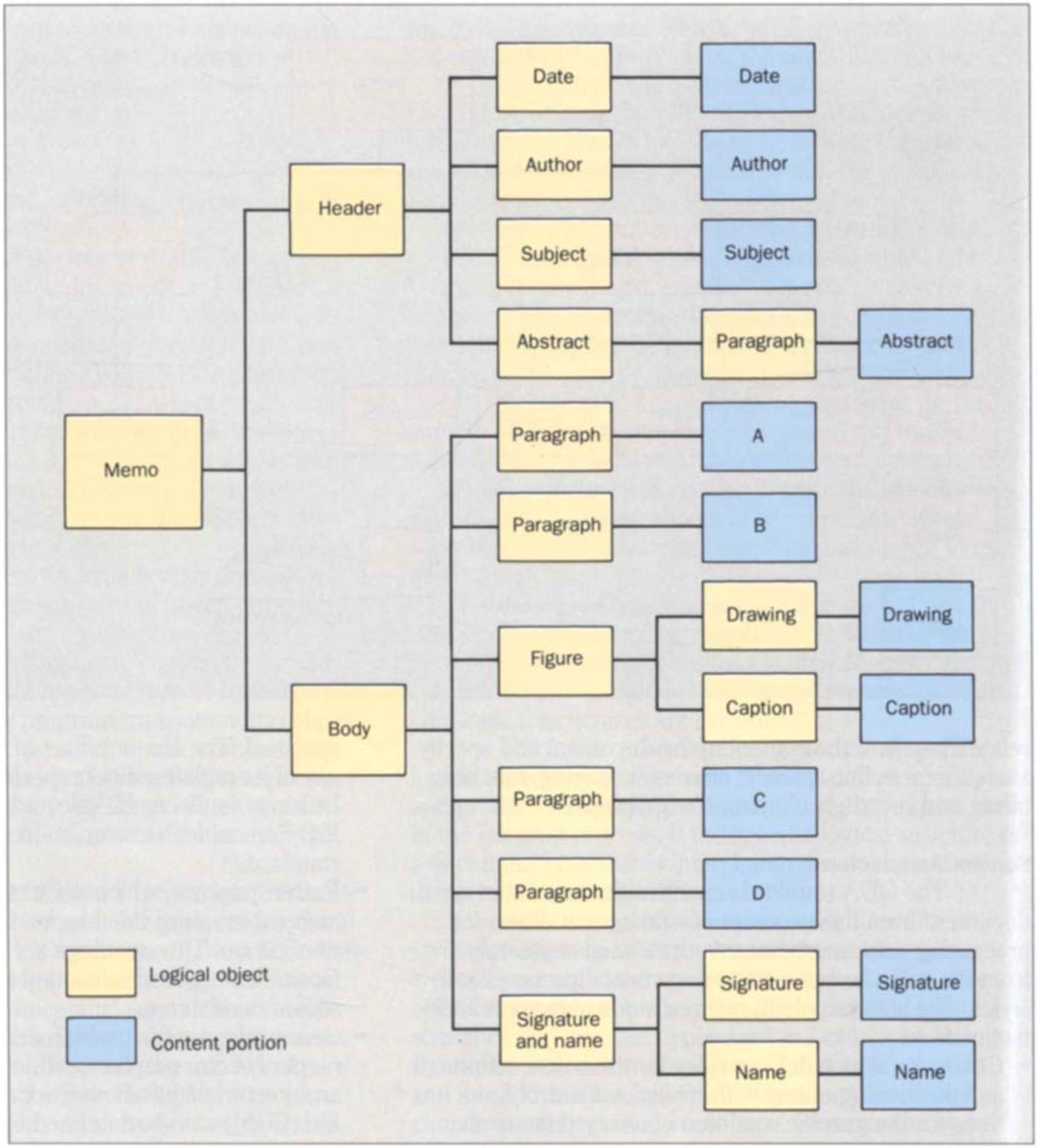
relate directly to the content of the document and specify such things as line spacing, character spacing, font size, offset, and rectangle clipping (for graphics).

Content Architecture

The ODA standard separates the content of a document from the structure of a document. Rules for processing document content are defined separately from the rules for processing document structures and styles. The standard defines three kinds of content architectures:

- Character, which defines rules for the construction and positioning of text within blocks. Control functions for the graphic rendition of characters are also specified. The character set of ISO standard 6937 and any of its registered subrepertoires are permitted.[11] Different fonts can be selected in accordance with the ISO Font and Character Information Interchange standard.[12]
- Raster graphics, which defines rules for placing bit-mapped graphics into blocks. CCITT Recommendation T.4 and T.6 encodings are permitted.[13,14] Most facsimile images are encoded according to these recommendations.
- Geometric graphics, which defines rules for the placement of vector graphics within blocks. This content architecture is based on the Computer Graphics Metafile (CGM) standard defined by ISO.[15]

Figure 5. Specific logical structure for the sample memo shown in Figure 2.



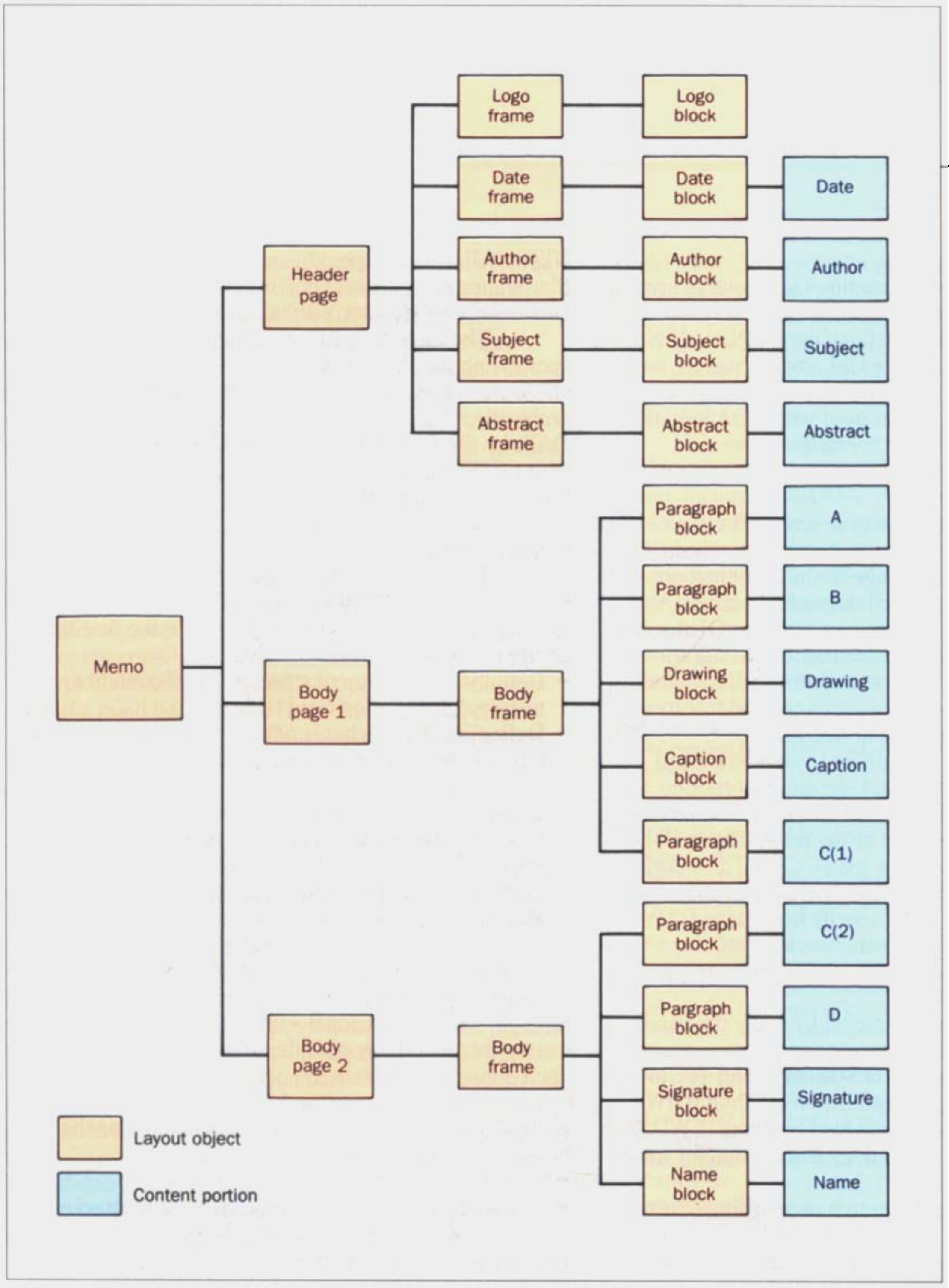


Figure 6. Specific layout structure for the sample memo shown in Figure 2.

Document Interchange Format

All ODA document structures and content architectures are encoded for data interchange in a standard notation, Abstract Syntax Notation One (ASN.1). ASN.1 has been defined for use in the OSI environment to facilitate network- and hardware-independent transfers of data.[16,17] The particular format used with ODA is ODIF (*Office Document Interchange Format*).

Implementation Agreements

The recent completion of several ODA implementation agreements, known as *document application profiles* (DAPs), has set the stage for implementations that embody the ODA standard. Essentially, an ODA DAP is a functional standard that specifies an ODA subset suitable for a particular application. These agreements are necessary because the ODA standard allows great flexibility in the choice of attributes and features selected to represent a particular document.

CCITT plans to provide for the encoding and transfer of ODA documents and plans to incorporate ODA into many of its telematic services, such as facsimile and videotex, as evidenced by the publication of ODA DAPs for these services in the 1988 CCITT T.500 Series Recommendations.[18,19,20] Other application profiles under development in CCITT specify how these DAPs are to be used in CCITT telematic services and the equipment characteristics and communication requirements for these services.

DAPs have been completed by many user and/or vendor forums, including:

- The U.S. National Institute of Standards and Technology (NIST) Open Implementation Workshop (OIW).
- The European Workshop on Open Systems (EWOS).
- The Interoperability Technology Association for Information Processing (INTAP), Japan.

Some of these DAPs permit interchange of simple text-only documents; while other profiles, such as the NIST DAP, describe ODA structures that can be used to interchange documents containing text and graphics. The NIST DAP has attempted to specify document structures that accurately represent documents created with current and anticipated desktop publishing systems.

The three forums mentioned above are jointly sponsoring the PAGODA (Profile Alignment Group on ODA) meetings, where the goal will be to produce a hierarchically related, internationally harmonized set of DAPs by the first quarter of 1990. There is close liaison with CCITT and it is expected that CCITT Recommendation T.502 will form the basis of the lowest level profile.

Future Extensions

There is currently a joint effort in CCITT and ISO to extend the capabilities of ODA. Color specifications (for both paper and video display) will be the first major additions. Other extensions include:

- Definition of document structures and content architectures for equations and tables.
- Definition of a mechanism for the association of different levels of security within different parts of a document.
- Definition of a tiled raster graphics content architecture. The tiled raster graphic work, which is progressing rapidly, provides the capability of partitioning a graphic laid-out page into individual segments (tiles) that can be individually manipulated.

To put the various possible extensions of the ODA model in context, a superset of the current document processing model has been proposed. This extended model is multidimensional—that is, several different processes can potentially operate on the document during each phase of processing.

It also allows support functions for two distinct methods of providing necessary information for the processing of a document:

- Direct user input
- Automatic processes that require no user intervention.

In the current proposals, the processing of a document from its initial form is described as a sequence of steps, each of which is an *evaluation*, *resolution*, or *manipulation* of the document. Evaluation involves processing within a document that is determined by information within the document itself. Resolution refers to processing within a document that is determined by using external references, i.e., information located outside of the document. Manipulation refers to the capability of accessing and/or modifying all or part of a document at any stage of processing. All automatic processes are envisioned as combinations of the resolution and evaluation steps.

Much of the current discussion on ODA extensions centers on whether new content architectures need to be created or whether different types of content can be handled by extending some of the document structures already available. Table layout and formulae processing are consistently mentioned in connection with this topic. The UNIX® system `tbl` preprocessor is considered a good basic model for table layout and can be described using existing ODA constructs. Thus, tables could be handled in logical structures by creating a table object type.

There is a different approach for the processing of formulae. A formula content architecture based on `eqn`,[21] the UNIX system equation typesetter preprocessor to `troff`,[22] has been proposed.

There is great interest in extending ODA to allow for interactive manipulation of document structures so that data from spreadsheets and data entry forms could be included in ODA documents. ODA would need to be able to provide a structure to which a database access protocol could feed information from a database. It would have to be able to export a description of the database record (number of fields, location, size, date, etc.) and of the logical relationship between records (keys) to the application. A new formatting capability would be required to effectively display data fields as entered by a user. There will have to be decisions on whether ODA will standardize calculation rules and field-value validation techniques.

Other extensions not currently under discussion, but part of the future work plan, are extensions that would allow different content architectures to be concatenated and to overlay each other as well as extensions for the support of three-dimensional (3-D) graphics, audio, video images, remote document editing, document filing and retrieval, and hypertext.

Summary

There is only one internationally recognized standard for the representation and open interchange of compound documents—ODA. The recent completion of ODA DAPs will foster the development and appearance of ODA systems. Extensions to the current standard will provide methods for including multimedia content in a single document. And finally, additional extensions for interactive data entry and remote editing of ODA documents will make ODA the key element in the future integration of office automation systems.

References

1. *Information Processing—Text and Office Systems—Office Document Architecture (ODA) and Interchange Format*, ISO 8613, International Organization for Standardization, 1988.
2. *Open Document Architecture*, CCITT Recommendations T.411—T.418, International Telegraph and Telephone Consultative Committee, 1988.
3. *Information Processing Systems—Open Systems Interconnection—Basic Reference Model*, ISO 7498, International Organization for Standardization, 1984.
4. *Reference Model of Open Systems Interconnection for CCITT Applications*, CCITT Recommendation X.200, International Telegraph and Telephone Consultative Committee, 1984 and 1988.
5. *Information Processing—Text and Office Systems—Office Document Architecture (ODA) and Interchange Format, Part 2: Document Structures*, ISO 8613/2, International Organization for Standardization, 1988 (also known as CCITT Recommendation T.412, 1988).
6. *Information Processing—Text and Office Systems—Office Document Architecture (ODA) and Interchange Format, Part 6: Character Content Architectures*, ISO 8613/6, International Organization for Standardization, 1988 (also known as CCITT Recommendation T.416, 1988).
7. *Information Processing—Text and Office Systems—Office Document Architecture (ODA) and Interchange Format, Part 7: Raster Graphics Content Architectures*, ISO 8613/7, International Organization for Standardization, 1988 (also known as CCITT Recommendation T.417, 1988).

8. *Information Processing—Text and Office Systems—Office Document Architecture (ODA) and Interchange Format, Part 8: Geometric Graphics Content Architectures*, ISO 8613/8, International Organization for Standardization, 1988 (also known as CCITT Recommendation T.418, 1988).
9. *Information Processing—Text and Office Systems—Office Document Architecture (ODA) and Interchange Format, Part 5: Office Document Interchange Format*, ISO 8613/5, International Organization for Standardization, 1988 (also known as CCITT Recommendation T.415, 1988).
10. *Information Processing—Text and Office Systems—Office Document Architecture (ODA) and Interchange Format, Part 4: Document Profile*, ISO 8613/4, International Organization for Standardization, 1988 (also known as CCITT Recommendation T.414, 1988).
11. *Information Processing—Coded Character Sets for Text Communication,* ISO 6937, International Organization for Standardization, 1985.
12. *Information Processing—Font and Character Information Interchange,* ISO 9541, International Organization for Standardization, to be published.
13. *Standardisation of Group 3 Facsimile Apparatus for Document Transmission*, CCITT Recommendation T.4, International Telegraph and Telephone Consultative Committee, 1984.
14. *Facsimile Coding Schemes and Coding Control Functions for Group 4 Facsimile Apparatus*, CCITT Recommendation T.6, International Telegraph and Telephone Consultative Committee, 1988.
15. *Information Processing Systems—Computer Graphics—Metafile for the Storage and Transfer of Picture Description Information (CGM)*, ISO 8632, International Organization for Standardization, 1987.
16. *Information Processing—Open Systems Interconnection—Abstract Syntax Notation One*, ISO 8824 and ISO 8825, International Organization for Standardization, 1988.
17. *Abstract Syntax Notation One*, CCITT Recommendations X.208 and X.209, International Telegraph and Telephone Consultative Committee, 1988.
18. *A Document Application Profile PM1 for the Interchange of Processable Form Documents*, CCITT Recommendation T.502, International Telegraph and Telephone Consultative Committee, 1988.
19. *A Document Application Profile PM1 for the Interchange of Group 4 Facsimile Documents*, CCITT Recommendation T.503, International Telegraph and Telephone Consultative Committee, 1988.
20. *Document Application for Videotex Interworking*, CCITT Recommendation T.504, International Telegraph and Telephone Consultative Committee, 1988.
21. B. W. Kernighan and L. L. Cherry, "A System for Typesetting Mathematics," *Communications of the ACM*, Association of Computing Machinery, Vol. 18, No. 3, March 1975, pp. 151-157.
22. B. W. Kernighan and R. Pike, *The UNIX Programming Environment*, Prentice-Hall, Englewood Cliffs, New Jersey, 1984.

(Manuscript received May 1, 1989)