

Appendix F – Part 2

Defendants' Supplemental Prior Art Statement
'228 Patent
(TC1270-TC1297)

to

TimeBase's Memorandum in Support of Its Motion
for Summary Judgment of No Invalidity

Prior Art Analysis

• **Arnold-Moore 1994:**

Arnold-Moore 1994 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Arnold-Moore 1994 discusses the fact that legislation can be frequently amended and that one option for storing this information is to store consolidations. A consolidation consist of the original (or base) statute together with all of the changes made by the amending act. For example:

- *See, e.g.*, “The other important characteristic of a legislation is that its content can change with the passage of time. Sections (or indeed larger units) can be added, removed, or altered. A principal Act is created when a new body of law is reduced to legislation creating a new Act where no other existed, or where a large scale restructuring of existing legislation is made creating a new Act (or group of Acts) which completely replaces a previous Act or group of Acts. In between amending Acts are passed which make alterations to the principal Acts, sometimes changing the wording of one or two section, at other times replacing the whole sections or removing or inserting whole parts or chapters,” at 2.
- *See, e.g.*, “The simplest solution to this problem is to apply the amending Acts (either by hand or automatically) once to produce each consolidation, and to store each consolidation as well as the principal Act (and each of the amending Acts).”
- *See generally* 11-12.
- *See, e.g.*, “each unchanged atom will be represented only once . . . ; where an atom is altered a new atom is inserted in the database with the alterations performed,” at 12.

• **Arnold-Moore 1994-2:**

Arnold-Moore 1994-2 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Arnold-Moore 1994-2 discloses storing multiple versions of legislative material. For example:

- *See, e.g.*, “This will, for example, allow a software engineering document that is the right version to be retrieved,” at THOM00196608.

• **Arnold-Moore 1995:**

Arnold-Moore 1995 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Arnold-Moore 1995 discloses storing consolidations of original legislation with subsequent amendments. For example:

- *See, e.g.*, “The second and more important for the purposes of this paper is that its

Prior Art Analysis

content can change with the passage of time. Sections (or indeed larger and smaller elements) can be added, removed or altered,” at 297.

- *See, e.g.*, “The Australian legislators ... have adopted the textual style of amendment where amendments require words to be omitted and others inserted,” at 298.
- *See, e.g.*, “The ideal would be for law libraries to have a copy of the relevant consolidation in which the appropriate amendments are pasted for every different version of an Act,” at 298.
- *See, e.g.*, “There is great potential for CALR systems not only to present legislation in a format familiar to lawyers (like that of the paper consolidation) but to present it as it would have appeared at any arbitrary point in time with annotations available with the text. The problems of how to store these various versions in electronic databases are discussed at length elsewhere,” at 298.
- *See, e.g.*, “a text processing module which produces new consolidations from the structured representation of actions and existing versions of the Principal Act,” at 299.
- **Arnold-Moore 1997:**

Arnold-Moore 1997 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Arnold-Moore 1997 discloses storing consolidated versions of legislative amendments, which constitute modified versions of stored principal acts. For example:

 - *See, e.g.*, “A consolidation (or reprint) of a Principal Act is that Act as amended at a particular time,” at 56.
 - *See, e.g.*, “An ideal drafting tool would present the drafter with the appropriate consolidation of the target of the amendment (see Figure 1) and allow him or her to mark amendments on that consolidation (see Figure 2),” at 57.
 - *See, e.g.*, “There are 6 basic types of amendment, omitting text, inserting text, replacing text, omitting a whole element, inserting a whole element, or replacing a whole element,” at 58.
 - *See, e.g.*, “To do this, and RTF to SGML converter ... is then applied to the document to produce two separate SGML representations. The strike-through and underline is used to generate two parallel trees representing the structure of the document, one before all of the changes and one after the changes have been applied,” at 59.
- **Arnold-Moore 1997-2:**

Prior Art Analysis

Arnold-Moore 1997-2 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Arnold-Moore 1997-2 discusses the fact that legislation can be frequently amended and that one option for storing this information is to store point-in-time versions of the legislation. For example:

- *See, e.g.*, “Themis also allows access to different versions of legislation by specifying a point-in-time at which the law is required,” at 175.
- *See, e.g.*, “In particular, the complex structure of legislation and different versions of a particular piece of legislation can be better supported,” at 175.
- *See, e.g.*, “The other important distinguishing characteristic of legislation is that its content can change with the passage of time,” at 176.
- *See, e.g.*, “Ideally a library, paper or digital, would provide ever possible consolidation of every piece of legislation,” 176.
- *See, e.g.*, “The following section 3 introduces a system which addresses these needs... The problem of handling multiple versions of legislation is addressed in section 6,” at 176.
- *See, e.g.*, “The simplest solution is to apply the amending Statutes (either manually or automatically) once to produce each consolidation, and to store each consolidation as well as the principal Statute (and each of the amending Statutes),” at 179.
- *See, e.g.*, “That means that when a section is modified, only the fragment containing that section needs to be updated, not the whole document,” at 179.

• **Bachman 1973:**

Bachman 1973 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Bachman 1973 discloses storing modified and updated versions of the stored text-based data. For example:

- *See, e.g.*, “It involves all aspects of storing, retrieving, modifying, and deleting data in the files on personnel and production, airline reservations, or laboratory experiments – data which is used repeatedly and updated as new information becomes available,” at 654.
- *See, e.g.*, “The second activity of database management is to update, which includes the original storage of data, its repeated modification as things change, and ultimately, its deletion from the system when the data is no longer needed,” at 655.
- *See, e.g.*, “The hiring of a new employee would cause a new record to be stored.

Prior Art Analysis

Reducing available stock would cause an inventory record to be modified.
Cancelling an airline reservation would cause a record to be deleted. All of these are recorded and updated in anticipation of future inquiries,” at 655.

• **Bentley 1979:**

Bentley 1979 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Bentley 1979 discloses dynamic storage structures which can be used to store updated versions of the stored data. For example:

- *See, e.g.*, “In many applications one may desire various utility operations on data structures, such as insertion and deletion,” at 398.
- *See, e.g.*, “Many applications, however, require dynamic structures, in which insertion and deletions can be made,” at 407.

• **Campbell 1988:**

Campbell 1988 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Campbell 1988 discloses storing modified and updated versions of the stored data, or nodes. For example:

- *See, e.g.*, “A node contains arbitrary data that can be stored as text or as fixed-length binary blocks. A node can be classified as archived, nonarchived, or append-only. When an archived node is updated, a new version of the node is created using the new contents. Previous versions of an archived node can be retrieved. When a nonarchived node is updated, the previous contents are replaced by the new contents. When an append-only node is updated, the new contents are appended to the previous contents,” at 856-57.
- *See, e.g.*, “The version history for a HAM object is updated each time that object is modified. Because each access to an object contains a version time, previous versions of objects can be viewed,” at 857.
- *See, e.g.*, “Change operations modify data associated with an existing object. A change operation takes an object index, a version time, and object-dependent data and returns a version time. The object index specifies the unique identifier for the object being modified. The returned version time represents the time the object was modified,” at 857.

• **Fay 1996:**

Fay 1996 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Fay 1996

Prior Art Analysis

discloses that users frequently modify and store portions of text-based data. For example:

- *See, e.g.*, “Each element usually is represented by a number of dated versions, and as the document is updated, the latest version is added to the list of versions for that element, or node,” at 2:14–16.

- **Haake 1992:**

Haake 1992 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Haake 1992 discloses storing versions of text-based data. For example:

- *See, e.g.*, “The states of a versioned object are called versions and are represented by individual nodes, links, or composites,” and “The freeze operation is an explicit operation that saves the state of the version. The state of an atomic node comprises its content and attributes, the state of a link is defined by two references to other hypertext objects and the link attributes Thus, versions referenced by links and composites also belong to the state of the link or composite,” at 46.

- **Horne 1997:**

Horne 1997 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Horne 1997 discloses storing modified and updated versions of the stored statutes. For example:

- *See, e.g.*, “But both the printed and internet versions of the Acts are static, that is to say, they do not show any amendments that have been made by later legislation,” at 1.
- *See, e.g.*, “In the past this problem was to some extent dealt with by the official printed series called Statutes in Force. This series of volumes suffered from the disadvantage that it took a great deal of time for each particular title to be produced,” at 1.
- *See, e.g.*, “HMSO have a program called ‘the Statute Law Database’. This is an electronic version of Statutes in Force. It contains in SGML form the law as it was on a particular date in the 1980s together with all acts and statutory instruments which have come into force since that time,” at 3.

- **Kim 1996:**

Kim 1996 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Kim 1996 discloses storing multiple versions of the stored multimedia. For example:

Prior Art Analysis

- *See, e.g.*, “Finally, our hypermedia system can provide database management system functions such as ... version control of structured document components as separate objects,” at 496.
- *See, e.g.*, “Fourthly, we design a version control mechanism that distinguishes versionable objects and non-versionable objects,” at 496.
- *See, e.g.*, “Thirdly, an object is classified into versionable object or a non-versionable object,” at 497.
- *See, e.g.*, “Versioning is essential to hypermedia system in order to keep track of changes to the hypermedia network,” at 498.

- **Larson 1988:**

Larson 1988 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Larson 1988 discloses that the stored nodes can be changed and versions will be stored. For example:

- *See, e.g.*, “Neptune (like the Xanadu system) retains all versions of a document and permits links to be formed between any two points in a document or between documents,” at 196.
- *See, e.g.*, “The indexes supported in Telesophy include a keyword index (providing Boolean and proximity searching), a ‘temporal index’ that permits selection by the time an IU was created, and a ‘spatial index’ that ‘places items in an N-dimensional space based on their attributes, then allows the space to be searched,” at 197.

- **Lo 1995:**

Lo 1995 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Lo 1995 discloses storing multiple versions of the stored text-based data. For example:

- *See, e.g.*, “Versioning is essential to history-keeping of a document. It allows evolutionary information and states of this document to be captured so that future references are possible,” at 339.
- *See, e.g.*, “The document database may only carry out modification to a committed document’s content by creating a new version of it,” at 340.

- **Lo 1996:**

Lo 1996 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Lo 1996

Prior Art Analysis

discloses versioning, including the storage of modified portions. For example:

- *See, e.g.*, page 10, section 1.2.2 (Managing Functions): “Another important aspect in the basic service layer is versioning. . . . Supporting versioning, however, is not as simple as just storing several versions of the same document in the database.”
- *See, e.g.*, page 9, section 1.3: “one of the major problems tackled in this thesis is to decide to which version a link would connect should the destination exist in more than one version.”
- *See, e.g.*, page 27, under the heading “Committing Versions of Documents”: “Any changes made to a committed document will introduce a new version of the document Once this new version is approved and thereby committed, any further changes will similarly cause the creation of further versions of documents.”
- *See generally, e.g.*, page 31, text following headings “Multiple-Versions-Multiple-Trees” and “Multiple-Versions-Single-Tree.”
- *See, e.g.*, page 32, text following heading “The Fragmentation Model”: “Since structure can be recognized in documents, components such as chapters or sections can be recognized and hence be manipulated individually. The fragmentation model makes use of this advantage brought by the described structure to store only the modified components of a document when a new version of it is created.”
- *See, e.g.*, page 33: “Each time a committed document is modified and a new version created, the system distinguishes and stores only the fragments that are being modified.”
- *See generally, e.g.*, section 2.3.4, starting on page 44, entitled “SGML Document Fragmentation.”
- *See generally, e.g.* text following heading “Hyperlinks,” beginning on page 53.
- **Osterbye 1992:**

Osterbye 1992 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Osterbye 1992 discloses that multiple versions of the stored text-based data can be stored. For example:

 - *See, e.g.*, “This paper discusses issues related to providing version control in hypertext systems,” at 33.
 - *See, e.g.*, “In such systems the very nature of production makes the hypertext evolve over time,” at 33.

Prior Art Analysis

- *See, e.g.*, “Version control is the discipline of controlling and tracking the evolution of a product over time,” at 33.
- *See, e.g.*, “If an element is versioned, a specific version represents a state in the development,” at 33.
- *See, e.g.*, “Each module can exist in several versions, and all the versions of a module is often referred to as a version group,” at 34.
- *See, e.g.*, “The node is the basic entity for storing contents. We require all nodes to have an attribute for contents and a name. Nodes are versioned,” at 38.
- *See, e.g.*, “While it is possible to explicitly create new versions of individual nodes, a new-version command is available at the context level,” at 39.

- **Peltonen 1993:**

Peltonen 1993 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Peltonen 1993 discloses that the stored nodes can be changed and versions will be stored. For example:

- *See, e.g.*, “*Documents* represent any data for design tools: drawings, manuals, bitmap images, etc. Originally we only used the concepts of a document and document version. However, our industrial partner turned out to require a considerably richer document structure. A document therefore includes a number of subdocuments, each subdocument has a number of subdocument versions, and each subdocument version has a number of representations,” at 6.
- *See, e.g.*, Figure 2.
- *See, e.g.*, “When a new document version is created, the version is made to include the same subdocument versions as the parent document version. Only when a user modifies a particular subdocument for the first time in the child document version, the system creates a new subdocument version, and includes it in the child document version,” at 8.

- **Promenschenkel 1995:**

Promenschenkel 1995 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Promenschenkel 1995 discloses that multiple versions of the text-based data can be stored. For example:

- *See, e.g.*, “The STEPS systems is designed to take a document through the

Prior Art Analysis

publishing process from author's draft to finished print version," at 1.

• **Sacks-Davis 1995:**

Sacks-Davis 1995 discloses "obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data." Specifically, Sacks-Davis 1995 discloses that multiple versions of the text-based data can be stored. For example:

- *See, e.g.*, "For data modification requests, it checks that the new record satisfies all validation constraints defined on the table, such as key constraints," at 460.
- *See, e.g.*, "The index manager can be supplied with a new record to insert, an old and new version of a record that has been updated, or an existing record to delete," at 460.

• **Sciore 1991:**

Sciore 1991 discloses "obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data." Specifically, Sciore 1991 discloses storing multiple versions of the text-based data. For example:

- *See, e.g.*, "We often call the generic object the design object, and the associated version object its versions," at 358.
- *See, e.g.*, "For example, one version might be the result of a bug fix. Creating a revision of a previous version. Another might be the result of an alternative design decision," at 359.

• **Sciore 1994:**

Sciore 1994 discloses "obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data." Specifically, Sciore 1994 discloses storing new versions of existing text based data as they are created. For example:

- *See, e.g.*, "Each time a versioned attribute changes, a new version is created corresponding to the new state," at 87.
- *See, e.g.*, "Many database applications require the storage and manipulation of different versions of data objects," at Abstract.

• **Stonebraker 1990:**

Stonebraker 1990 discloses "obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data." Specifically, Stonebraker 1990 discloses the no-overwrite nature of POSTGRES, which stores a new

Prior Art Analysis

copy of the portion each time the portion is modified. For example:

- *See generally, e.g.*, section IV.A, 137: discussing the no-overwrite feature of POSTGRES, and saying “the old record remains in the database whenever an update occurs” and “[t]he second benefit of a no-overwrite storage manager is the possibility of *time travel*” and “[t]his storage manager should be contrasted with a conventional one where the previous record is overwritten with a new one.”

- **Taylor 1994:**

Taylor 1994 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Taylor 1994 discloses storing, querying, and displaying portions of text-based data as they evolve over time. For example:

- *See, e.g.*, “The geographical schema models four editions of the Ordnance Survey maps between 1880 to 1994,” at 239.
- *See, e.g.*, “The temporal schema ranges from 1755 . . . to 1994,” at 239.

- **Travis & Waldt:**

Travis & Waldt discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Travis & Waldt disclose the need to store versions of text-based data that is periodically updated. For example:

- *See, e.g.*, “When a version is created, all deltas that have been applied to the original document are applied, and a complete document is stored as the new version,” at 187.
- *See, e.g.*, Case Studies, especially pages 379 and 395.
- *See, e.g.*, “Many of these documents must be updated periodically through the life of the aircraft to reflect improvements implemented after leaving the aircraft (DAC still provides supporting documentation for its DC-3, a fifty-nine year old aircraft). In addition, DAC must retain a highly accurate and comprehensive history of original and revised publications to support internal and FAA (Federal Aviation Administration) requirements,” at 395.

- **Wilkinson 1998:**

Wilkinson 1998 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Wilkinson 1998 discloses dividing documents into portions and storing the portions and

Prior Art Analysis

amended portions. For example:

- *See, e.g.*, “When multiple versions of a document are stored in a system, it is clear that there is great potential for redundancy. . . . The granularity of the redundancy can be reduced by fragmenting the documents. This can be done by breaking the document into lines or by using more sophisticated structural encoding to identify logical units in the document. This approach is typically used in hypertext systems, where larger documents are usually fragmented for viewing purposes anyway. If whole versions or version histories are required, these can be reconstructed from the fragment versions,” at 103.
- *See, e.g.*, “Alternatively, documents can be regarded as sets of fragments that have properties including type, parents, and children,” at 103.

• **Wilson 1988:**

Wilson 1988 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Wilson 1988 discloses that multiple versions of the text-based data can be stored. For example:

- *See, e.g.*, “This list can be inverted so that later legislation that amends or clarifies an earlier act may be accessed directly from the act,” at 27.
- *See generally* 28.
- *See, e.g.*, “Hypertext resolves the difficulties of a single printed version; it can maintain many versions of a single subsection,” at 35.
- *See, e.g.*, “By selecting the button (BEFORE 5 JULY 1973), we can see the text of Section 167(2)(a) before it was amended: figure 9,” at 35.

• **Wilson 1992:**

Wilson 1992 discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, Wilson 1992 discusses the fact that legislation can be frequently amended and that the Guide system can accommodate storing multiple versions of legislative material. For example:

- *See, e.g.*, “Lawyers often need multiple versions of the same text. Some statutes are subject to frequent amendment. Earlier versions may still be needed for cases brought under early legislation, for legal research or for a proper understanding of contemporary cases. . . . Local buttons are an ideal mechanism for multiple versions. An electronic system makes it easier to store the name of the amending author and the date of the amendment where these are required. Figure 12 shows a section of the Industrial Relations Act 1971 with local buttons for an earlier version. Figure 13

Prior Art Analysis

shows the button expanded,” at 179-180.

• **The Pre-1997 Westlaw/Westmate System:**

The Westlaw/Westmate system discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.”

Specifically, the Westlaw/Westmate system contained modified predefined portions of text-based data. *See, e.g.:*

- DataBasics 1993, at doc no. 79858–59 (“United States Code Annotated”): Disclosing Westlaw’s USCAYY databases: “United States Code Annotated 19YY* (YY is the last two digits of a year, e.g., 90 for 1990”).
- www.westlaw.com, California Statutes Annotated Database from 1996 (CA-STAN96), CA BUS & PROF § 28 (two versions: one active and one as amended).
- The Essential Guide 1996, at 131: “WESTLAW contains the current version and historical versions of the *United States Code Annotated* and the current version and historical versions of statutes for all 50 states. To display the current version of a cited statute while viewing its Shepard’s result, type **fi**.”

• **The Pre-1997 Premise System:**

The Premise system discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, the Premise System contained modified predefined portions of text-based data. *See, e.g.:*

- “Browse” the “Document List” in the CA-STAT-AN1 database within the Premise software to view the modified portions of legislation associated with the Premise system, including Bus. & Prof. Code § 2003 to see both a predefined portion and a modified predefined portion of the same statutory section.

• **The Astoria System (pre-1997):**

The Astoria System contained modified predefined portions of text-based data. Use of this system therefore constitutes a method for obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data. For example:

- *See, e.g.*, XSoft Astoria: “Astoria deals with the concept of ‘document components.’ A document component is a piece that is designed to be maintained as a unit, whether this be at the volume or book level, or at some finer granular point, such as paragraph or list,” at THOM00198652.
- *See, e.g.*, Astoria 1997-1: “Astoria can apply revision information to only the components that change during an editing session. Astoria detects and maintains

Prior Art Analysis

revision history at the component level, not just at the document level. . . Astoria stores versioning information in an efficient format, and past versions are always available for republishing or for providing an audit trail,” at THOM00211908.

- *See, e.g.*, XSoft: “Because of its sophisticated integration with SGML editors, Astoria maintains revision information on individual elements, and past versions are always available,” at THOM00198648.

- **The EnAct System** (previously known as Themis):

The EnAct system discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, the EnAct System stores modified portions of text-based data. *See, e.g.*:

- Arnold-Moore 1997-2, at 179 (text following the heading: “Managing Versions in *Themis*”), and in particular: “Since *Themis* already fragments large documents for other reasons, the obvious solution was to move the versioning granularity [sic] down to the fragment level. The *Themis* systems stores [sic] a table of contents directly and each fragment separately, indexing each fragment. That means that when a section is modified, only the fragment containing that section needs to be updated, not the whole document. . . . For fast query response, *Themis* does not make use of deltas either in the index or in the storage of the documents.”
- <http://web.archive.org/web/19990430002036/www.thelaw.tas.gov.au/background.html>: “Each fragment contains the dates for which that piece of legislation is in force. When legislation is amended, the system automatically builds new versions of fragments which are affected by amendments and keeps the old ones for historical reference. Consolidations are generated by joining together the fragments relevant at a particular point-in-time.”

- **The SCALEplus System:**

The SCALEplus system discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, the SCALEplus system stores modified portions of text-based data. *See, e.g.*:

- Kerr 2000, at 6-4, ¶ 178: “The *Historical* database[] contain[s] copies of legislation as they appeared at certain date[s].” *See also* 6-8, 196.

- **The Documentum/Interleaf System:**

The Documentum/Interleaf system discloses “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data.” Specifically, the Documentum/Interleaf System allows users to modify portions of documents and then store the modified portions. *See, e.g.*:

Prior Art Analysis

- Interleaf provides versioning capabilities when users modify documents. *See* Ovum Interleaf 1996, at 260–61.
- Documentum provides versioning capabilities when users modify documents. *See* Ovum Documentum 1996, at 218–219.
- **The Federal Rules of Civil Procedure System:**

The Federal Rules of Civil Procedure system contained modified predefined portions of text-based data.
- **The Law Desk NY System:**

The Law Desk NY system contained modified predefined portions of text-based data.
- **The Law Desk USCS System:**

The Law Desk USCS system contained modified predefined portions of text-based data.
- **The New Mexico Law System:**

The New Mexico Law on Legal Ethics system contained modified predefined portions of text-based data.
- **The NY CLS Beta System:**

The NY CLS Beta system contained modified predefined portions of text-based data.
- **The OnPoint System:**

The OnPoint system contained modified predefined portions of text-based data.

(d) storing each of the plurality of portions of text-based data;

Each of the references discussed above with regard to claim 1(b) discloses both “dividing the text-based data into a plurality of portions of text-based data” and then subsequently “storing each of the plurality of portions of text-based data.” The supporting quotations for this assertion are provided above in connection with claim 1(b).

(e) storing the amended portion of text-based data;

Each of the references discussed above with regard to claim 1(c) discloses both “obtaining an amended portion of text-based data that is amended relative to one of the plurality of portions of text-based data” and then subsequently “storing the amended portion of text-based data.” The supporting quotations for this assertion are provided above in connection with claim 1(c).

Prior Art Analysis

(f) providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space;

• **Agosti 1991:**

Agosti 1991 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.”

Specifically, Agosti 1991 discloses a hyperconcept made up of auxiliary data about the stored text-based documents which can be organized, displayed and linked to the stored text-based data. For example:

- *See, e.g.*, “EXPLICIT is based on a two-level architecture which holds the two main parts of the informative resource managed by an information retrieval tool: the collection of documents and the indexing term structure. The term structure is managed as a schema of concepts which can be used by the final user as a frame of reference in the query formulation process,” at 316.
- *See, e.g.*, “The EXPLICIT hypertext retrieval model is based on a two-level architecture, which holds the two main parts of a database managed by an information retrieval system: the collection of documents, and the auxiliary data. By the term auxiliary data we mean the data describing the document information contents,” at 317.
- *See, e.g.*, “a set of structured data which represents the different deterministic properties of the object (e.g., date of publication, title, list of authors, etc.),” at 318.
- *See, e.g.*, “Objects of the second level result from the application of the classification abstraction mechanism to the objects of the first level; they denote concepts which are variously interrelated, for example through a classification hierarchy... Objects and links of this level form the ‘hyperconcept,’” at 319.
- *See, e.g.*, “Each of the two levels of the system’s architecture represents a distinct network of nodes and links,” at 319.
- *See, e.g.*, “The model supports navigation between the two levels by means of the navigability function. In this way it is at all times possible to pass from the hyperdocument to the hyperconcept and back again,” at 320.
- *See, e.g.*, “When the object is inserted in the network it becomes a node of the structure. The data which the object contains are modeled as property values of the object and become, when inserted, actual node attributes. Some node attributes can be, for example, name, node type (e.g. legal authority documents, law documents, auxiliary data items), or the link type,” at 320.

Prior Art Analysis

- See generally 323 (Figures 2-6).

- **Anwar 1996:**

Anwar 1996 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.”

Specifically, Anwar discloses computer system for querying, displaying, and analyzing multi-dimensional data using attributes. For example:

- See, e.g., “This invention provides a computer including . . . at least one display device The multi-dimensional display and manipulation system includes an [sic] user interface and a multi-dimensional data object subsystem where the subsystem includes means for creating, populating, updating, manipulating, storing, retrieving, modifying and analyzing multi-dimensional data objects and the user interface includes means for displaying and graphically manipulating the multi-dimensional data objects in a window defined on a display device. The multi-dimensional data objects include multi-sided geometrical objects having a side for each dimensionality of the data to be represented by the data object.” at Summary of the Invention.
- See, e.g., “The present invention also provides a user interface and data management system that allow a user to more efficiently visualize, display, manipulate, and analyze multidimensional data.” Summary of the Invention.
- See, e.g., “Multi-dimensional is any set of information that has associated with it many attributes which require considerably more than [sic] three dimensions to display, analyze and manipulate the data, i.e., each country, each state within a country, each city within each state within each country, etc. [i.e. text-based data],” at 4:60–67.
- See 7:26–30, indicating that dimensions are attributes (*i.e.*, the number of dimensions is the number of attributes).
- See 5:58–6:3, indicating that the attributes define the manner in which the data is displayed in the multi-dimensional space.

- **Arnold-Moore 1994:**

Arnold-Moore 1994 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.” Specifically, Arnold-Moore 1994 discloses the attribute of effective data, as well as other keys relating to the content of the text based data. Arnold-Moore 1994 explicitly discloses that these attributes can be used as part of a “multi-dimensional

Prior Art Analysis

access structure[].” For example:

- *See, e.g.*, “Alternatively multi-dimensional access structures can be used to index simultaneously on time and other keys (in this case content),” at 11.
- *See, e.g.*, “Content and structural indexing can be done on each version,” at 9.

• **Arnold-Moore 1994-2:**

Arnold-Moore 1994-2 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.” Specifically, Arnold-Moore 1994-2 discloses encoding stored legislative material with attributes and then using attributes to access the stored documents. For example:

- *See, e.g.*, “The language, Structured Generalized Query Language (SGQL), allows efficient access to the content, structure and attributes of documents at any level within their structure,” at THOM00196608.
- *See, e.g.*, “Queries might specify that certain attributes have particular values, but have concentrated on matching against the content of the document,” at THOM00196608.
- *See, e.g.*, “These databases will need to be searched by attribute,” at THOM00196608.
- *See, e.g.*, “SGML describes a tagging scheme and a metagrammar for describing the structure of documents,” at THOM00196609.
- *See, e.g.*, “One can associate typed information with particular SGML elements by using attributes which appear in the text within the begin tag,” at THOM00196609.
- *See, e.g.*, “Thus, we may have associated with each document, or element of a document, a set of attributes that we shall call features, to distinguish them from the attributes defined by the grammar that describes the SGML document class,” at THOM00196609.
- *See, e.g.*, “In order to construct a conceptual model of the database system we consider the database to be a list of ELF’s (ELEMENTS with FEATURES) where an ELF is: a complete SGML element ... a list of features associated with that element,” at THOM00196611.

• **Arnold-Moore 1997-2:**

Arnold-Moore 1997-2 discloses “providing a plurality of attributes, wherein the attributes

Prior Art Analysis

define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.” Specifically, Arnold-Moore 1997-2 discloses encoding stored legislation with attributes such as effective dates. Further, Arnold-Moore 1997-2 also discloses searching based on attributes, which results in a list of links to stored text-based documents. For example:

- *See, e.g.*, “Logical structure is identified by tags which appear interspersed with the text in an SGML document,” at 177.
- *See generally* 178 (Figures 1 and 2).
- *See, e.g.*, “Each fragment or table of contents has a valid start and end time associated with it. We then use an inverted-file index to give access to the fragments by content. As described for whole documents, we then filter results using the time information to collect just the fragments which are valid at the specified time defaulting to the current date (See Figure 1),” at 179.
- *See, e.g.*, “Schemes which rely only on the structure of the document and do not make use of the attributes of the SGML elements produce references that must be changed when the structure of the target document changes,” at 181.
- *See generally* 180 (Figure 4).

- **Bachman 1973:**

Bachman 1973 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.” Specifically, Bachman 1973 discloses encoding stored text-based data with attributes to create an *n*-dimensional space. Bachman 1973 further discloses accessing the stored text-based data using these attributes. For example:

- *See, e.g.*, “Start with the value of the primary data key, of the record of interest, and pass each record in the file through core memory until the desired record, or one with a higher key, is found. (A primary data key is a field within a record which makes that a record unique within the file.) Social security numbers, purchase order numbers, insurance policy numbers, bank account numbers are all primary data keys. Almost without exception, they are synthetic attributes specifically designed and created for the purpose of uniqueness. Natural attributes, e.g. names of people and places, dates, time, and quantities, are not assuredly unique and thus cannot be used,” at 654.
- *See, e.g.*, “From this point, I want to begin the programmer’s training as a full fledged navigator in an *n*-dimensional data space,” at 654.

Prior Art Analysis

- *See, e.g.*, “In addition to a record’s primary key, it is frequently desirable to be able to retrieve records on the basis of the value of some other fields. For example, it may be desirable, in planning ten-year awards, to select all the employee records with the ‘year-of-hire’ field value equal to 1964. Such access is retrieval by secondary data key. The actual number of records to be retrieved by a secondary key is unpredictable and may vary from zero to possibly include the entire file. By contract, a primary data key will retrieve a maximum of one record,” at 655.
- *See, e.g.*, “With the advent of retrieval on secondary data keys, the previously one-dimensional data space received additional dimensions equal to the number of fields in the record,” at 655.
- *See, e.g.*, “My proposition today is that it is time for the application programmer to abandon the memory-centered view, and to accept the challenge and opportunity of navigation with an n-dimensional data space. The software systems needed to support such capabilities exist today and are becoming increasingly available,” at 657-58.
- *See, e.g.*, “[L]et us enumerate his opportunities for record access...He can enter the database with a secondary data key value and sequentially access all records having that particular data value for the field,” at 656.
- **Bentley 1979:**

Bentley 1979 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.” Specifically, Bentley 1979 discloses encoding multiple attributes into the stored data to create a linked k-dimensional space. For example:

 - *See, e.g.*, “This subject area, which is often called ‘multikey searching,’ ‘multidimensional searching,’ or ‘multiple attribute retrieval,’ has been the focus of a great deal of research in the past few years,” at 397.
 - *See, e.g.*, “In database terminology a file is a collection of records, each containing several attributes or keys,” at 397.
 - *See, e.g.*, “This problem can also be cast in geometric terms by regarding the record attributes as coordinates and the k values for each record as representing a point in a k-dimensional coordinate space. The file of records then becomes a point set in k-space,” at 397.
 - *See, e.g.*, “The cell technique can support insertions and deletions by merely keeping a linked list of the points in each cell and inserting or deleting the new or old record in the appropriate list,” at 407.

Prior Art Analysis

• **Campbell 1988:**

Campbell 1988 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.”

Specifically, Campbell 1988 discloses encoding stored data with multiple attributes that are then used to manage and link to the stored text-based data. For example:

- *See, e.g.*, “The HAM storage model is based on five objects: graphs, contexts, nodes, links, and attributes. The HAM maintains history for these objects, allows selective access through a filtering mechanism, and can allow for access restrictions through a data security mechanism,” at 856.
- *See, e.g.*, “Attributes can be attached to the contexts, nodes, or links. Attribute values can be strings, integers, floating-point numbers, or user-defined types. Attribute/value pairs give semantics to HAM objects. They can represent application-specific properties of objects or contain information that further describes an object. Attributes are also used in the predicates that are part of the HAM filters,” at 857.
- *See, e.g.*, “The HAM provides a filtering mechanism that allows subsets of HAM objects to be extracted from large graphs. Filters allow the user to specify visibility predicates, which are expressions relating attributes and their values. HAM filters only return objects that satisfy the predicates. Filters also allow the user to specify a version time so that earlier versions of a graph can be examined,” at 857.
- *See, e.g.*, “Get operations retrieve data from existing objects. A get operation takes an object index and a version time, and returns the data that existed at the specified time. The object index specifies a unique identifier for the object from which data is being retrieved,” at 858.

• **Caplinger 1986:**

Caplinger 1986 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.”

Specifically, Caplinger 1986 discloses a computer system for querying, displaying, and linking multi-dimensional data using attributes. For example:

- *See, e.g.*, “The best known application of SDMS was a database of naval ships. At the topmost level, the SDMS display consisted of a world map, with the locations of task forces displayed on it at their actual physical locations. A user could ask the system to zoom in on a particular group of ships, which were then shown as icons based on ship type, along with additional textual information like the ships’ names and nationalities,” at 114–15.

Prior Art Analysis

- See, e.g., “If we can continue to differentiate entries along a dimension no matter how small a range we view, then we call the dimension *continuous*. If on the other hand we reach a point where we must map multiple entries to the same point, then we call the dimension *quantized*, since there are only a finite number of possible locations in a given range,” at 115.
- See, e.g., “Part of the information space consists of object locations. Additional information may be conveyed by the *attributes* of the objects in the space. One alternative is to give the objects no attributes or features at all, but simply display them as featureless points. We call this a *point space*. All of the information contained in a point space is held in object location. However, there may be characteristics of objects that make little sense as dimensions, particularly if we cannot think of an ordering for them. In such cases, we may want to associate those characteristics with object attributes. For example, we might use geometric shapes as objects, and put information in their sizes, shapes, or colors,” at 116.
- See, e.g., “Once we have represented the database entries in the space themselves, we still have to add the link information. There are two ways of doing so. The first is simply to draw the links as arcs connecting the objects corresponding to linked entries. The problem with this is that the links are not visually useful if there are many of them—in fact, they will generate so much visual clutter that the actual objects will be obscured. A much preferable way of representing links is implicitly, in the dimensions. In this scheme, linked entries are simply close to one another in space. We assign an integer to each entry based on the path length from some root entry to that node,” at 116.
- See, e.g., “In general, dimensions and attributes mix well together. Moving the viewpoint acts as a coarse query,” at 116.
- **Elmasri 1990:**

Elmasri 1990 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.” Specifically Elmasri 1990 discloses many attributes that organize information, retrieve information, and link information in a multidimensional space, and therefore discloses this method. For example:

 - See, e.g., “Although the interval-based search problem is similar in many respects to the k-dimensional spatial search problem, the various methods proposed for the k-dimensional spatial search . . . are not suitable for the time dimension.” Background of the Invention.
 - See, e.g., “In addition to the regular record attributes, A_i , each record will have an interval attribute, *valid_time*, consisting of two subattributes, t_s (valid start time), and

Prior Art Analysis

t_e (valid end time).” Summary of the Invention.

- *See, e.g.*, “However, numerous past versions of the object may also exist. These versions of an object are linked to the current version and may be recovered through the use of various known techniques” Summary of the Invention.
- *See, e.g.*, “A time index for temporal databases is provided which enables the retrieval of database object versions that are valid during specified time periods. Unlike prior access and retrieval structures, the present index is based on objects whose search values are time intervals rather than time points.” Abstract.
- *See, e.g.*, “A search for objects that satisfy such a temporal condition combines selection based on a time interval with a selection based on conditions involving attribute values.” Summary of the Invention.
- **Fay 1996:**

Fay 1996 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.” Specifically, Fay 1996 discloses that there are attributes associated with the text-based information. For example:

 - “[T]he imported document instance would contain document, chapter, section, etc., objection having their own attributes and connected according to the structure implied by the descriptive markup.” 3:35–40.
 - “Each element of a document is associated with a data field. For example, one node of a tree may be a chapter, and contain textual data in the form of a chapter heading, a chapter introductory paragraph, a chapter abstract, etc., as well as ‘structural’ data such as the identity of a parent (document), identify of children (sections), and connections to other places in other documents where the same language may also be used. The person who has the document checked out is the only person who can change any of this textual or structural data,” at 1:25–34.
 - “In a document instance, descriptive markup identifies the elements in a document with tags. These tags are interspersed with the textual content of the document. When imported into this document system, the tags and text are separated into various objects. For example, a simple DTD might define a document to consist of chapters, a chapter to consist of sections and a section to be made up of paragraphs. Thus, the imported document instance would contain document, chapter, section, etc., objects having their own attributes and connected according to the structure implied by the descriptive markup,” at 3:29–40.
- **Haake 1992:**

Prior Art Analysis

Haake 1992 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.”

Specifically, Haake 1992 discloses many attributes that organize, display, and link portions and amended portions of information. For example: *See, e.g.*

- *See, e.g.*, “SEPIA’s basic hypertext objects are typed atomic nodes, typed composite nodes, and typed labeled links. Composite nodes contain an ordered set of references to other hypertext objects while atomic nodes contain data like text, graphics etc. All hypertext objects are equipped with attributes holding information like names of nodes or labels of links.” at 43.
- *See, e.g.*, “CHS offers nodes, links, and composites that can be equipped with application-defined attributes. Objects can be accessed by their attribute values using the query language of the underlying database system of CHS,” at 46.
- *See, e.g.*, “CHS maintains object histories. It stores the creation time and the author of each node, link, composite, and attribute and records each update to these objects with time and author information in an update history,” at 46.
- *See, e.g.*, “Next to an internal mob [multi-versioned object] identifier and the version set, CoVer maintains for a mod optional application-defined and application-supplied state-independent attributes (e.g. an elaborated description, project management information such as the name of a responsible person). The attributes carried by the versions represent the state-dependent characteristics of a specific version of a versioned object,” at 46.

• **Hansen 1993:**

Hansen 1993 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.”

Specifically, Hansen 1993 discloses information retrieval with information represented as points along dimensions. For example:

- *See, e.g.*, “a method for information representation and retrieval within a general-purpose digital computer. Information of all simple types is represented as points along dimensions, and compound information types are represented as the intersection of two or more dimensions in a multidimensional data space.” Abstract.
- *See, e.g.*, “In the method of the present invention, all information entities are treated as points along multidimensional lines.” Brief Description of the Invention.

• **Kim 1996:**

Kim 1996 discloses “providing a plurality of attributes, wherein the attributes define a

Prior Art Analysis

manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.”

Specifically, Kim 1996 discloses encoding stored multimedia with attributes and then using attributes to access the stored documents. For example:

- *See, e.g.*, “Second, since we apply an object-oriented paradigm to modeling hypermedia data and links, we can inherit the properties and methods of the object-oriented model,” at 496.
- *See, e.g.*, “Secondly, we apply an object-oriented approach to modeling hypermedia data as objects,” at 496.
- *See, e.g.*, “SGML provides a syntax flexible enough to describe the logical structure of documents,” at 497.
- *See, e.g.*, “From now on, we will call our markup language HOML (Hypermedia Object Modeling Language). HOML is an SGML application and provides facilities for describing the relations between multimedia data and provides methods for describing hypermedia links and synchronization,” at 497.
- *See, e.g.*, “The text element has attribute values for security, version, and duration,” at 497.
- *Seem e.g.*, “On the other hand, the structure-based retrieval searches the logical and hypermedia network structures,” at 498.
- *See, e.g.*, “[W]e design a new query language which supports both an information retrieval mechanism and a database query mechanism for handling structured hypermedia documents,” at 500.
- *See, e.g.*, “Element attribute search,” at 501.

• **Kimball 1996:**

Kimball 1996 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.”

Specifically, Kimball 1996 discloses dimensional modeling that can be visualized as a cube of multiple dimensions. For example:

- “Fundamentally, this is a book about dimensional modeling and how to build a dimensional data warehouse and keep it running. Dimensional modeling is a new name for an old technique for making databases simple and understandable. When a database can be visualized as a ‘cube’ of three, four, or even five or more dimensions, people can imagine slicing and dicing that cube along each of its

Prior Art Analysis

dimensions.” Page xx.

- Text following the heading “The Time Dimension,” Pages 7–8.
- Text following the heading “The Dimensional Model,” Pages 10–11.
- Text following the heading “The Dimensional Tables,” Pages 13–14.
- Text following the heading “Attributes Are the Drivers of the Data Warehouse,” Pages 17–18.

- **Larson 1988:**

Larson 1988 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.” Specifically, Larson 1988 discloses a multidimensional space made up of nodes and links. For example:

- *See, e.g.*, “The basic conceptual model of hypertext is a multidimensional graph (or network) whose vertices (nodes) are information sources and whose edges (or links) represent both directed associative indices to those sources, and operational programs to display or utilize the information they contain. . . . The concept of an ‘information space’ that supports a hypertext network is useful. The world’s store of recorded human knowledge may be viewed as a N-dimensional ‘information space,’ where N is the number of possible attributes that may be used to make up any information source,” at 195.
- *See, e.g.*, “Obviously, not all points in this information space will be occupied. Various dimensions of similarity may be considered to provide a useful clustering of information items within this space. The clustering may be based on any number of entity/attribute relationships, such as storage location (e.g., a library collection), form, topical, historic, author, citations, language, or literary style,” at 196.
- *See, e.g.*, “Neptune also permits any number of descriptive attribute/value pairs to be assigned to any node or link, which may be searched,” at 196.
- *See, e.g.*, “The indexes supported in Telesophy include a keyword index (providing Boolean and proximity searching), a ‘temporal index’ that permits selection by the time an IU was created, and a ‘spatial index’ that ‘places items in an N-dimensional space based on their attributes, then allows the space to be searched,” at 197.

- **Lo 1995:**

Lo 1995 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-

Prior Art Analysis

based data can be organized, displayed and linked in a multidimensional space.” Specifically, Lo 1995 discloses encoding stored text-based documents with attributes using SGML, and linking the stored text-based documents. For example:

- *See, e.g.*, “SGML (Standard Generalised Markup Language) was adopted by ISO as an international standard to describe the structure of electronic documents. The reason for using SGML is its international acceptance as an electronic document markup standard. Furthermore, while the description of a document’s structure is primarily applied in publication, database technology could also make use of this structural knowledge to enhance its management of documents,” at 339.
- *See, e.g.*, “SGML tags are placed in text to denote its structure and such practice is known as descriptive markup. A component in the text such as a title or a paragraph can be explicitly defined by marking them with these tags,” at 339.
- *See, e.g.*, “Each destination is given an identifier unique to the version... Globally unique identifications are then derived by combining the document identifier (DocId) and the version number (VerNo) with the destination identifier (DestID),” at 340.
- *See, e.g.*, “Attributes can be associated to this global link table as well to describe the characteristics of the link objects,” at 343.
- *See, e.g.*, “While the Direct Reference Method can also achieve this by incorporating the attributes in the document (may utilize SGML attributes),” at 344.

- **Lo 1996:**

Lo 1996 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.” Specifically, Lo 1996 discloses multi-dimensional attributes associated with text-based information. For example:

- *See* page 31. For example: “Nodes of this document tree are associated with the information of valid time by two attributes: the start valid time and the end valid time.”
- *See* page 38. For example: “*Attributes* can be embedded into the elements to describe some of their aspects, for example, the element chapter may contain the attribute “id” which gives the identifying number of that element.”
- *See generally* section 2.3.3, starting on page 41, entitled “SGML Support for Versioning,” including figure 2.5.
- *See* page 48, for example. “More precisely, HyTime addresses three aspects: . . . Representing any quantifiable dimension (typically spatial and temporal dimensions)

Prior Art Analysis

as a coordinate space and placing objects within it.”

- See page 61, for example. “Such a facility will allow a user to traverse to any version that contains the destination that he wants.”
- See generally page 112, including: “Documents are stored as records in SIM databases As a result of applying the fragmentation model in the prototype system . . . , two databases are defined: the *basedoc* database and the *fragdoc* database. The former is for storing bases, and the latter for fragments. The schema of the *basedoc* database defines six attributes for a record. The *body* attribute contains a whole piece of text marked-up in SGML. This piece of text is recognized to be a specific version of a specific document by the values of the *version number* (*VerNo*) and the *document identified* (*DocId*) attributes.”

- **Osterbye 1992:**

Osterbye 1992 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.”

Specifically, Osterbye 1992 discloses encoding stored text-based documents with attributes such as author, date, and state, and providing links between documents. For example:

- See, e.g., “If an element is versioned, a specific version represents a state in the development. Will it then be possible to annotate it, or to add new attributes?” at 33.
- See, e.g., “The top of the hierarchy is an entity which allows attributes to be attached to all entities,” at 34.
- See, e.g., “Selection from version groups is based on the attribute values of the modules in the version group. Such attributes can typically contain information about authors, creation date, and release state (tested, experimental, etc.),” at 34.
- See, e.g., “Adding new attributes to a node might also be attractive,” at 35.
- See, e.g., “For both models methods for controlling change and addition of attributes must be found,” at 35.
- See, e.g., “The general entity provides attributes, which are key-value pairs attached to the objects. All information is stored as attributes,” at 38.

- **Peltonen 1993:**

Peltonen 1993 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.”

Prior Art Analysis

Specifically, Peltonen 1993 discloses a multidimensional space made up of nodes and links. For example:

- See, e.g., “Documents and various attributes are stored in a centralized database,” at 2.
- See, e.g., “The goal of the EDMS project was to develop a system which stores the actual documents contents in addition to attribute data,” at 2.
- See, e.g., “All data, i.e., the actual document contents and various attributes, are stored in a commercial relational database,” at 3.
- See, e.g., “Various data about objects are stored as *attributes*. All objects of the same object kind (e.g., all document versions) have the same *system attributes*. These attributes are defined by EDMS and their values can only be modified by the system. The EDMS administrator can define both *common* and *type-specific* attributes. A common attribute is defined for an object kind; for example, all documents can have a particular attribute. A type-specific attribute is defined for an object type; for example, all versions of documents of the type *drawing* can have a particular attribute,” at 9.

• **Sacks-Davis 1994:**

Sacks-Davis 1994 discloses “providing a plurality of attributes, wherein the attributes define a manner in which the plurality of portions of text-based data and the amended portion of text-based data can be organized, displayed and linked in a multidimensional space.” Specifically, Sacks-Davis 1994 discloses encoding stored documents with multiple attributes and using those attributes to organize and link together the stored documents. For example:

- See, e.g., “Typed information can be associated with particular SGML elements by using attributes that appear in the text with the begin tag,” at THOM00198836.
- See, e.g., “A list of attributes can be associated with each element type, and these attributes contain typed data associated with each element instance,” at THOM00198839.
- See, e.g., “While strictly part of the SGML text, data stored as SGML attributes is representative of the data that is typically stored in traditional databases,” at THOM198840.
- See, e.g., “We also want to be able to query on SGML attributes...A list of attributes can be associated with each element type, and these attributes contain typed data associated with each element instance. Access by SGML attributes is a commonly used method for supporting hypertext links,” at THOM00198839.