Timebase Pty Ltd v. Thomson Corporation, The Doc. 247 At

Appendix F – Part 13

Defendants' Supplemental Prior Art Statement
'228 Patent
(TC1584-TC1612)

to

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

Dockets.Justi

Prior Art Analysis

the content index to filter versions on time," at *xix*.

- **Arnold-Moore 1994-2:**

Arnold-Moore 1994-2 discloses "a method for electronically searching text-based data encoded with a mark-up language." Specifically, Arnold-Moore 1994-2 discloses storing text-based data encoded with SGML. Arnold-Moore 1994-2 further discloses a method for searching the stored text-based data either by keyword, by attribute, or by a combination. For example:

- *See, e.g.*, "A data model and query language for accessing structured documents expressed in SGML is presented," at THOM00196608.
- *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.*, "We thus see that a database system to support databases of large structured documents need a query language that allows retrieval: by exact matching Boolean combinations of words and phrases; by ranking by similarity to a given text; using hypertext links; by attribute," at THOM00106609.
- *See, e.g.*, "Standard Generalized Markup Language (SGML), now provides a grammar for describing document structure which is widely used for document exchange," at THOM00196609.
- *See, e.g.,* Representative Queries at THOM00196609-10.
- *See, e.g.*, "The relational model extended to support content queries can support a whole range of queries including mixed content and structure, pure structure, and attribute queries," at THOM00196610.
- *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:**

Prior Art Analysis
Arnold-Moore 1997 discloses "a method for electronically searching text-based data encoded with a mark-up language." Specifically, Arnold-Moore 1997 discloses storing text-based data encoded with SGML. Arnold-Moore 1997 further discloses a method for searching the stored text-based data by keywords and by attributes such as effective date. For example: • *See, e.g.*, "The Themis system manages a library of legislation which is encoded in the Structured Generalized Markup Language (SGML)," at 58. • *See, e.g.*, "The drafter can view any Act or search the whole database using Boolean or ranking queries at any time point for which a valid version is stored on the system," at 59. • **Arnold-Moore 1997-2:** Arnold-Moore 1997-2 discloses "a method for electronically searching text-based data encoded with a mark-up language." Specifically, Arnold-Moore 1997-2 discloses storing text-based legislative documents encoded with SGML. Arnold-Moore 1997-2 further discloses a method for filtering legislative material based on effective dates. For example: • *See, e.g.*, "Themis uses SGML to store legislation," at 175. • *See, e.g.*, "Despite improved performance of ranking (or natural language) queries, lawyers who have used digital legal libraries will be most familiar with the Boolean query approach," at 177. • *See, e.g.*, "The SIM system supports both Boolean (with proximity operators) and ranking queries," at 177. • *See, e.g.*, "Law librarians have identified the need for a standard coding of structure in legal texts for ease of distribution and reformatting, particularly identifying SGML as an appropriate scheme," at 177. • *See, e.g.*, "Logical structure is identified by tags which appear interspersed with the text in an SGML document," at 177. • *See, e.g.*, "A standard inverted-file word index can be used to extract a list of matching documents on content (or an enhanced index to allow queries on structure also), and then this list can be filtered by a time constraint using a time-index," at 179. • *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

Prior Art Analysis

defaulting to the current date (See Figure 1)," at 179.

- **Horne 1997:**

  Horne 1997 discloses "a method for electronically searching text-based data encoded with a mark-up language." Specifically, Horne 1997 discloses storing text-based legislative data encoded with SGML. Horne 1997 further discloses a method for searching the stored legislative data by attributes such as effective date. For example:

  - *See, e.g.*, "Since the mid 1980s every statute and statutory instrument has been coded using SGML (the Standard Generalized Markup Language)," at 2.
  - *See, e.g.*, "But markup could go further. It could give the dates on which the amendments were made, the dates on which they took effect, and the names of the Acts or SI which had made them, and the user's program could use this markup to display a statute as it was on a particular date chosen by the user and could offer hypertext cross-references to the amending legislation," at 3.
  - *See, e.g.*, "There is a 'front-end' program which can be used to search the database and to display the results in HTML. So the user should be able to use the program to discover easily what the law is on a particular point on a particular date," at 3.

- **Kim 1996:**

  Kim 1996 discloses "a method for electronically searching text-based data encoded with a mark-up language." Specifically, Kim 1996 discloses storing text-based data encoded with SGML. Kim 1996 further discloses a method for searching the stored text-based data either by keyword or attribute. For example:

  - *See, e.g.*, "In this paper, we design a new hypermedia markup language using SGML," at 496.
  - *See, e.g.*, "[W]e propose a markup language for hypermedia using SGML (Standard Generalized Markup Language)," at 496.
  - *See, e.g.*, "First, since our hypermedia markup language is designed using SGML, the language can … support content-based and structure-based retrieval," at 496.
  - *See, e.g.*, "Therefore, it is necessary to support content-based and structure-based retrieval as well as database mechanisms for hypermedia documents," at 496.
  - *See, e.g.*, "From now one, we will call our markup language HOML (Hypermedia Object Modeling Language). HOML is an SGML application," at 497.
  - *See, e.g.*, "Therefore, it is necessary to support an efficient information retrieval, which provides content and structure-based retrieval, and database query mechanism.

Prior Art Analysis
Besides, the content-based retrieval method searches every object, which consists of the hypermedia network, and the object contents. On the other hand, the structure-based retrieval searches the logical and hypermedia network structures," at 498. • *See, e.g.*, "Since the data about document structure and attribute values can be stored as instances in the database, Postgres can directly process the structure-based retrieval. The other is a content-based retrieval. When a content-based retrieval query is given, the information retrieval manager performs full-text retrieval against the hypermedia document database," at 500. • *See, e.g.*, "For this reason, we design a new query language which supports both an information retrieval mechanism and a database query mechanism for handling structure hypermedia documents," at 500. • *See, e.g.*, "Element attribute search," at 501. • **Larson 1988:** Larson 1988 discloses "a method for electronically searching text-based data encoded with a mark-up language." Specifically, Larson 1988 discloses searching text-based data. For example: • *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 1996:** Lo 1996 discloses "A method for electronically searching text-based data encoded with a mark-up language, the method comprising." Specifically, Lo 1996 discloses searching for and presenting 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

Prior Art Analysis

can be explicitly defined by marking them with these tags," at 339.

- *See generally* pages 36–37.
- *See, e.g.,* page 113: "All the attributes in both database are indexed by the SIM DBS and are thus searchable."

- **Promenschenkel 1995:**

Promenschenkel 1995 discloses "a method for electronically searching text-based data encoded with a mark-up language." Specifically, Promenschenkel 1995 discloses storing text-based documents encoded with SGML. Promenschenkel 1995 further discloses a method for searching the stored text-based data. For example:

- *See, e.g.*, "It will encompass capture and conversion of the article, Standard Generalized Markup Language (SGML) editing," at 1.
- *See, e.g.*, "IDI's storage manager built to accept, query, retrieve and manipulate SGML document components as separate objects," at 1.

- **Sacks-Davis 1994:**

Sacks-Davis 1994 discloses "a method for electronically searching text-based data encoded with a mark-up language." Specifically, Sacks-Davis 1994 discloses storing text-based documents encoded with SGML. Sacks-Davis 1994 further discloses a method for searching the stored text-based data based on attribute. For example:

- *See, e.g.*, "The ISO Standard Generalized Markup Language (SGML) was designed to support document interchange and fulfils this requirement," at THOM00198835.
- *See, e.g.*, "There is also sufficient information in the DTD to allow the application level to provide information to the user about the elements and attributes which are available for query for each different type of document in the database," at THOM00198836.
- *See, e.g.*, "SGML's power can be utilized to create additional types of query over the whole database on the structural characteristics of the documents," at THOM00198839.
- *See, e.g.*, "We also want to be able to query on SGML attributes, for instance: Query 7.1 Find <corres>s with attribute confidential = yes," at THOM00198839.

- **Sacks-Davis 1995:**

Sacks-Davis 1995 discloses "a method for electronically searching text-based data encoded with a mark-up language." Specifically, Sacks-Davis 1995 discloses storing

Prior Art Analysis

text-based documents encoded with SGML. Sacks-Davis 1995 further discloses a method for searching the stored text-based data based on attribute. For example:

- *See, e.g.*, "Text retrieval systems should have the ability to manipulate structured information and attributes, as do conventional database systems. Thus a text system should, for example, be able to refer to a paragraph within a section, a date of publication, or to one of a list of authors," at 454.

- *See, e.g.*, "SGML is a widely used standard for the representation and interchange of documents. SGML defines a language that can be used to specify abstract grammars consisting of tags that are interspersed throughout the text of documents," at 464.

- **Travis & Waldt:**

  Travis & Waldt discloses "a method for electronically searching text-based data encoded with a mark-up language." Specifically, Travis & Waldt discloses storing text-based data encoded with SGML. Travis & Waldt further discloses a method for searching the stored text-based data either by keyword or by attribute, or both. For example:

  - *Passim*, especially Chapter 10.

  - *See, e.g.*, "SGML browsers offer context-sensitive searching capabilities so that the user can quickly access the required information . . . . For example, a search can be defined to allow a user to search for a part number, but only if it is contained in a chapter that was updated after a certain date. Or, a user can have the browser return a list of all sections containing a particular phrase, but only if the phrase is contained in a note. These are examples of context-sensitive searches," at 52-53.

  - *See, e.g.*, pages 194–95 (and figure 61), 198 (and figure 64).

  - *See, e.g.*, "The loader also makes available to the database parameterized information that can be used later to search and retrieve the appropriate objects. Such parameter information is object identifiers, author names, creation and modification dates, and perhaps some keywords. Most of this information can be obtained by querying the attributes on the element tags in the content of the document object," at 204.

- **Wilkinson 1998:**

  Wilkinson 1998 discloses "a method for electronically searching legislation encoded with a markup language." Specifically, Wilkinson 1998 discloses a case study about a document management system for legislation (EnAct) which has documents encoded in SGML. For example:

  - *See, e.g.* Chapter 9, starting on page 161 (entitled "Case Study: Managing Legislation").

Prior Art Analysis

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

  The Westlaw/Westmate system contained a method for searching text-based data encoded with a mark-up language. For example:

  - *See, e.g.,* DataBasics 1993, ("United States Code Annotated"): Disclosing mechanisms for searching using attributes, at doc no. 79858-59.

  - *See, e.g.,* Westlaw DB 1991, (disclosing the mechanism for searching the Westlaw databases using attributes), at 2–5, 14–15.

  - *See, e.g.,* Wren 1994, "you can use a field-restricted search request in LEXIS or WESTLAW to conduct highly specialized searches that would be nearly impossible—or at least tremendously tedious and time-consuming—to perform without a computer," at 75.

  - The Essential Guide 1996, at 3: "You can use WESTLAW to retrieve information from primary sources, such as cases and statutes from all 50 states and the District of Columbia, and from secondary sources, such as law reviews and treatises. You can seamlessly access Dow Jones News/Retrieval sources, including *The Wall Street Journal*, the same-day *New York Times* News Service and over 2,000 other sources. In addition, WESTLAW contains hundreds of databases from DIALOG, the world's largest online source of factual information. Subjects covered include business, current events, intellectual property, medicine, science and technology, and much more."

- **The Pre-1997 Premise System**:

  The Premise system contained a method for searching text-based data encoded with a mark-up language. For example:

  - *See, e.g.,* Premise Software & Statutes: Select "Search/Search Book…" and then change the "Search Using" field to "Fields Template," which will then display a mechanism for searching the portions and amended portions using attributes.

- **The Astoria System (pre-1997):**

  The Astoria System was used to search documents encoded with a markup language. For example:

  - *See, e.g.,* Astoria 1997-1: "Astoria lets users navigate through the document depository and view documents down to the individual components that comprise them." at THOM00211907.

  - *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,

Prior Art Analysis

whether this be at the volume or book level, or at some finer granular point, such as paragraph or list." (THOM00198652)

- *See, e.g.,* Astoria 1997-1: "Astoria provides a multilingual engine that lets users search on document content, structure, attributes, and version information," at THOM00211909.

- *See, e.g.,* Screen shot, at THOM00211908.

- *See, e.g.,* XSoft Astoria: "Astoria has complete version control capabilities. . . . The result is a version control module that is consistent with an SGML environment," at THOM00198652.

- *See, e.g.,* XSoft Premiers Astoria: "Astoria is fully integrated with the XSoft InContext SGML Editor," at THOM00211913. "Astoria is particularly well-suited to working with structured documents based on the Standardized Generalized Markup Language (SGML), making the management of those documents significantly easier and more productive," at THOM00211914. "[I]t builds on the value of SGML with document management capabilities such as version control, revision tracking and component re-use," at THOM00211914.

- *See, e.g.,* XSoft: "Astoria is an object-oriented document production component management system that enable users to easily find, use, share and manage SGML documents and their components, as well as unstructured documents. . . .Because Astoria works directly with SGML elements using an object-oriented database, it can provide unprecedented control over SGML documents as well as unstructured information by allowing fine-grained access and version control," THOM00198647.

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

  The EnAct system has a method for searching its text-based data encoded with a mark-up language. For example:

  - *See, e.g.,* Arnold-Moore 1997-2, figure 1 (showing ability to search the information with various attributes), and ("The SIM client/server architecture is built around Z39.50 [a standard for searching using attributes],") at 178, 177.

- **The SCALEplus System:**

  The SCALEplus system has a method for searching text-based data encoded with a markup language. For example:

  - *See, e.g.,* Kerr 2000, Figures accompanying ¶ 180 (page 6-4), ¶ 187 (page 6-6), ¶ 429 (page 11-3), ¶ 491 (page 11-19), ¶ 172 (page 6-2).

  - SCALEplus Secrets, at 2: "SCALEplus has lots of information that is huge,

Prior Art Analysis
particularly legislation. SCALEplus data is formatted in HTML which is common to all World Wide Web applications but is ideally suited for one or a few pages—to view a document you have to wait for the browser to load it (often over a modem). Because of this the decision was made to turn each piece of legislation into a number of HTML files, each file being a section of that Legislation. When a results list is returned from SCALEplus what you see are the HTML files that have been found that match your search. For Legislation this will be a section of an Act; for Caselaw an individual case." • SCALEplus UM 2: "Search Results Page" screen shot, and the text describing this screen shot (THOM00221679) • SCALEplus UM 2: "Advanced Search Screen" at (THOM00221692) and text describing the features on that screen, including the "Date Search Options." • **The Documentum/Interleaf System:** The Documentum/Interleaf system allows searching of text-based data encoded with a markup language. *See, e.g.*: • *See, e.g.,* Ovum Interleaf 1996, "It is possible to search on attribute combinations and combine these using Boolean operators," at 262 (and figure H2.6). • *See, e.g.,* Ovum Documentum 1996, "Documentum exposes just about all a document's attributes to querying," at 220 (and figure H1.10). • **The Core Materials on Legal Ethics System:** The Core Materials on Legal Ethics system involves a method of using a system which has a means for searching its text-based information encoded with markup language. • **The Federal Rules of Civil Procedure System:** The Federal Rules of Civil Procedure system a method of using a system which has a means for searching its text-based information encoded with markup language. • **The Law Desk NY System:** The Law Desk NY system a method of using a system which has a means for searching its text-based information encoded with markup language. • **The Law Desk USCS System:** The Law Desk USCS system a method of using a system which has a means for searching its text-based information encoded with markup language.

Prior Art Analysis
• **The New Mexico Law System:** The New Mexico Law on Legal Ethics system a method of using a system which has a means for searching its text-based information encoded with markup language. • **The NY Official Reports System:** The NY Official Reports system a method of using a system which has a means for searching its text-based information encoded with markup language. • **The NY CLS Beta System:** The NY CLS Beta system a method of using a system which has a means for searching its text-based information encoded with markup language. • **The OnPoint System:** The OnPoint system a method of using a system which has a means for searching its text-based information encoded with markup language. • **The Social Security Plus System:** The Social Security Plus system a method of using a system which has a means for searching its text-based information encoded with markup language. • **The UCC System:** The UCC system a method of using a system which has a means for searching its text-based information encoded with markup language.
(b) allowing a user to select a version date as a primary attribute and to input at least one search request; • **Arnold-Moore 1994:** Arnold-Moore 1994 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Arnold-Moore 1994 discloses allowing a user to search while filtering search results based on time. For example: • *See, e.g.*, "Queries can use these indices to return a subset of the database which can then be filtered on time constraints," at *xvi.* • *See, e.g.*, "A time index on the version skeleton can be maintained independently of the content index to filter versions on time," at *xix.*

Prior Art Analysis
• **Arnold-Moore 1994-2:** Arnold-Moore 1994-2 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Arnold-Moore 1994-2 discloses allowing a user to search based on both content and attributes, such as date. For example: • *See, e.g.*, "A data model and query language for accessing structured documents expressed in SGML is presented," at THOM00196608. • *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.*, "We thus see that a database system to support databases of large structured documents need a query language that allows retrieval: by exact matching Boolean combinations of words and phrases; by ranking by similarity to a given text; using hypertext links; by attribute," at THOM00106609. • *See, e.g.*, "Standard Generalized Markup Language (SGML), now provides a grammar for describing document structure which is widely used for document exchange," at THOM00196609. • *See, e.g.,* Representative Queries at THOM00196609-10. • *See, e.g.*, "The relational model extended to support content queries can support a whole range of queries including mixed content and structure, pure structure, and attribute queries," at THOM00196610. • *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:** Arnold-Moore 1997 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Arnold-Moore 1997

Prior Art Analysis

discloses allowing a user to search on both content and date. For example:

- *See, e.g.*, "The drafter can view any Act or search the whole database using Boolean or ranking queries at any time point for which a valid version is stored on the system," at 59.

- **Arnold-Moore 1997-2:**

  Arnold-Moore 1997-2 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Arnold-Moore 1997-2 discloses allowing a user to search both on content and date. For example:

  - *See, e.g.*, "Despite improved performance of ranking (or natural language) queries, lawyers who have used digital legal libraries will be most familiar with the Boolean query approach," at 177.

  - *See, e.g.*, "The SIM system supports both Boolean (with proximity operators) and ranking queries," at 177.

  - *See, e.g.*, "Law librarians have identified the need for a standard coding of structure in legal texts for ease of distribution and reformatting, particularly identifying SGML as an appropriate scheme," at 177.

  - *See, e.g.*, "Logical structure is identified by tags which appear interspersed with the text in an SGML document," at 177.

  - *See, e.g.*, "A standard inverted-file word index can be used to extract a list of matching documents on content (or an enhanced index to allow queries on structure also), and then this list can be filtered by a time constraint using a time-index," at 179.

  - *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.

- **Bachman 1973:**

  Bachman 1973 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Bachman 1973 discloses searching by attributes including date. For example:

  - *See, e.g.*, "In sequential file technology, search techniques are well established. 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

Prior Art Analysis

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.*, "Database management has two main functions. First is the inquiry or retrieval activity that reaccesses previously stored data in order to determine the recorded status of some real world entity or relationship," at 654.

- *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.

- **Bentley 1979:**

  Bentley 1979 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Bentley 1979 discloses six alternative methods for searching in the disclosed k-dimensional space using attributes. For example:

  - *See, e.g.*, "In this section we investigate a number of search methods for range searching," at 398.

  - *See generally* 398-405.

- **Campbell 1988:**

  Campbell 1988 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Campbell 1988 discloses searching the stored data by attributes, including date. For example:

  - *See, e.g.*, "The HAM maintains history for these objects, allows selective access through a filtering mechanism…," at 856.

  - *See, e.g.*, "Attributes are also used in the predicates that are part of the HAM filters,"

Prior Art Analysis

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 to 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. The version time is a time range for the data retrieval," at 858.

- *See, e.g.*, "*Filter* (and linearize) *operations* selectively retrieve information from a graph. A filter operation takes a predicate, a version time, and a list of attributes. These operations return a list of objects that satisfy the predicate and a list of requested attributes attached to each object. The version time specifies the time at which the filter is to search for the information. Each filter operation also has unique parameters in addition to those already specified," at 858.

- **Elmasri 1990:**

Elmasri 1990 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request ." Specifically Elmasri 1990 discloses searching information by ranges of effective dates, and therefore discloses this method. For example:

- *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.

- *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.*, "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.

- **Horne 1997:**

Horne 1997 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Horne 1997 discloses allowing a user

Prior Art Analysis

to search on attributes, including date. For example:

- *See, e.g.*, "But markup could go further. It could give the dates on which the amendments were made, the dates on which they took effect, and the names of the Acts or SI which had made them, and the user's program could use this markup to display a statute as it was on a particular date chosen by the user and could offer hypertext cross-references to the amending legislation," at 3.

- *See, e.g.*, "There is a 'front-end' program which can be used to search the database and to display the results in HTML. So the user should be able to use the program to discover easily what the law is on a particular point on a particular date," at 3.

- **Kim 1996:**

Kim 1996 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Kim 1996 discloses allowing a user to search the stored text-based data based on attributes, such as date. For example:

- *See, e.g.*, "First, since our hypermedia markup language is designed using SGML, the language can … support content-based and structure-based retrieval," at 496.

- *See, e.g.*, "Therefore, it is necessary to support content-based and structure-based retrieval as well as database mechanisms for hypermedia documents," at 496.

- *See, e.g.*, "Therefore, it is necessary to support an efficient information retrieval, which provides content and structure-based retrieval, and database query mechanism. Besides, the content-based retrieval method searches every object, which consists of the hypermedia network, and the object contents. On the other hand, the structure-based retrieval searches the logical and hypermedia network structures," at 498.

- *See, e.g.*, "Since the data about document structure and attribute values can be stored as instances in the database, Postgres can directly process the structure-based retrieval. The other is a content-based retrieval. When a content-based retrieval query is given, the information retrieval manager performs full-text retrieval against the hypermedia document database," at 500.

- *See, e.g.*, "For this reason, we design a new query language which supports both an information retrieval mechanism and a database query mechanism for handling structure hypermedia documents," at 500.

- *See, e.g.*, "Element attribute search," at 501.

- **Larson 1988:**

Larson 1988 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Larson 1988 discloses allowing a user

Prior Art Analysis

to search while filtering search results based on attributes. For example:

- *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.

- **Sacks-Davis 1994:**

  Sacks-Davis 1994 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Sacks-Davis 1994 discloses allowing a user to search the stored text-based documents on attributes, such as date. For example:

  - *See, e.g.*, "SGML's power can be utilized to create additional types of query over the whole database on the structural characteristics of the documents," at THOM00198839.
  - *See, e.g.*, "We also want to be able to query on SGML attributes, for instance: Query 7.1 Find <corres>s with attribute confidential = yes," at THOM00198839.

- **Sacks-Davis 1995:**

  Sacks-Davis 1995 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Sacks-Davis 1995 discloses allowing a user to search the stored text-based documents on attributes, such as date. For example:

  - *See, e.g.*, "Text retrieval systems should have the ability to manipulate structured information and attributes, as do conventional database systems. Thus a text system should, for example, be able to refer to a paragraph within a section, a date of publication, or to one of a list of authors," at 454.
  - *See, e.g.*, "SGML is a widely used standard for the representation and interchange of documents. SGML defines a language that can be used to specify abstract grammars consisting of tags that are interspersed throughout the text of documents," at 464.

- **Sciore 1991:**

  Sciore 1991 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Sciore 1991 discloses allowing a user to search based on attributes such as "occuredAt" which records the date of a change.

Prior Art Analysis

For example:

- *See, e.g.*, "At the conceptual level, we show how versions can be chosen from a version set based on the values of their attributes," at 356-57.
- *See, e.g.*, "The attribute occuredAT records the time at which the change took place. This attribute can be thought of as defining a one-dimensional time line, and allows the version set to be viewed as a function from times to versions," at 364.
- *See, e.g.*, "In general, the attributes chosen as dimensions should form a key of the version set, so that at most one version is associated with any coordinate in the version space," at 366.
- *See, e.g.*, "Our framework provides the means by which a database designer can specify a multi-dimensional logical structure to the version set. This logical structure can then be used to choose versions easily and conveniently," at 367.

- **Sciore 1994:**

  Sciore 1994 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Sciore 1994 discloses allowing a user to search on multiple attributes, including date. For example:

  - *See generally* 81-83.

- **Stonebraker 1990:**

  Stonebraker 1990 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Stonebraker 1990 discloses that users may query the POSTGRES database using attributes, such as date. For example:

  - *See generally* section II.B, starting on page 127 (demonstrating numerous examples of queries using attributes).

- **Taylor 1994:**

  Taylor 1994 discloses "a plurality of attributes, each attribute being a point on an axis of a multidimensional space for organizing and displaying." Specifically, Taylor 1994 discloses the ability to search on the time dimension as well as other dimensions. For example:

  - "SHIC provides for multiple classifications of an item which is crucial for classifying media items such as photographs where one scene could be classified in several different ways," at 240.

Prior Art Analysis

- "The temporal classification schema . . . stores information relating artefacts to some existence in time. The schema consists of year time point identifiers that constitute the relevant temporal span," at 240.

- *See* figure 1.

- "The combination of the conceptual, temporal and geographical classification spaces and their respective operators means that information requests do not need to be limited to one particular dimension. They can be multi-dimensional. An example of such a query would be: Show me Costume from Pontypridd during the Victorian era. The temporal dimension provides a method by which a temporal walk through a concept can be obtained. A temporal walk consists of showing the development of an historical concept or geographic area over time. In practical terms this requires the ordering of a set of media items according to the information stored in the temporal schema," at 242.

- **Travis & Waldt:**

  Travis & Waldt discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Travis & Waldt discloses allowing users to search on multiple attributes including date. For example:

  - *See, e.g.*, "SGML browsers offer context-sensitive searching capabilities so that the user can quickly access the required information . . . . For example, a search can be defined to allow a user to search for a part number, but only if it is contained in a chapter that was updated after a certain date. Or, a user can have the browser return a list of all sections containing a particular phrase, but only if the phrase is contained in a note. These are examples of context-sensitive searches," at 52-53.

  - *See, e.g.*, pages 194–95 (and figure 61), 198 (and figure 64).

  - *See, e.g.*, "The loader also makes available to the database parameterized information that can be used later to search and retrieve the appropriate objects. Such parameter information is object identifiers, author names, creation and modification dates, and perhaps some keywords. Most of this information can be obtained by querying the attributes on the element tags in the content of the document object," at 204.

- **Wilkinson 1998:**

  Wilkinson 1998 discloses "allowing a user to select a version date as a primary attribute of a multidimensional space and to input at least one search request." Specifically, Wilkinson 1998 discloses a case study about a document management system for legislation (EnAct) which allows searching by time. For example:

  - *See, e.g.* "The major motivation of the project was to develop a system to produce and manage an electronic repository of legislation to track and record legislation as it

Prior Art Analysis

changes with time, allowing access to the legislation both as it is now and also as it was at any time in the past," at 162

- *See, e.g.* "The most important differentiating characteristic of EnAct is its support of temporal attributes," at 169.
- *See, e.g.* "This time point defaults to the current date but can be set in the search screens, after which every query or link traversal uses that new date until another is provided," at 169.

- **Wilson 1990:**

Wilson 1990 discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Wilson 1990 discloses allowing a user to search either by keyword or attributes, such as date. For example:

- *See, e.g.*, "When the user enters the Justus running under Guide he is given a choice of three methods of access: direct access, index of pre-defined terms, and boolean query," at 125.
- *See, e.g.*, "it is easy to provide a structured index composed of the terms appearing in the catchwords section," at 125.

- **Wilson 1992:**

Wilson discloses "allowing a user to select a version date as a primary attribute and to input at least one search request." Specifically, Wilson discloses examples of searches that users can conduct by author name, key word, and date. For example:

- *See generally* 183.

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

The Westlaw/Westmate system discloses searching its text-based information by date, and so using the system would entail performing this method. *See, e.g.*:

- Westlaw DB 1991, generally (describing how the search process within Westlaw used attributes, text, pointers, and database files; the search files are coupled to the text-based data files using a linking means)
- DataBasics 1993, ("United States Code Annotated"): Disclosing mechanisms for searching using words or phrases within portions and amended portions of statutes at doc no 79858-59.
- DataBasics 1993, generally: Disclosing annual databases for statutory codes.

Prior Art Analysis
Selecting an annual database entails selecting a version date. • Wren 1994: 114–24, 141–42, listing the searchable fields, including, among others, the citation, date, and title fields • Password 1991: 5 (limiting your search by date) • Westlaw Ref 1993: 109–13 (describing date searching in Westlaw) • The Essential Guide 1996, at 29: "Some documents, such as archival statutes, are not retrievable with Find*, when you have a citation to one of these documents, access the appropriate database and retrieve it by restricting your search to the citation field (ci)." • The Essential Guide 1996, at 30: "To retrieve a specific section of the old tax code, e.g., 26 U.S.C.A. § 1201, restrict your search to the citation field with the following query . . . ." • The Essential Guide 1996, at 36: "**res** Restrictions—displays a screen allowing you to add restrictions to your WIN description, such as date or court" • The Essential Guide 1996, at 47: "You can use restrictions to limit your WIN search. For example, in a case law database, you can restrict your search by court, date, judge, attorney or added date." • The Essential Guide 1996, at 58: "To restrict your search to documents decided or issued on, before or after a certain date, or between a range of dates, add a date restriction (da) to your query." • **The Pre-1997 Premise System**: The Premise system discloses searching its text-based information using a date attribute, and so using the system would entail performing this method. *See, e.g.*: • Premise Research: Page 109 (describing how to search documents using the date field) • **The Astoria System (pre-1997):** The Astoria System allowed a user to select a version as an attribute used for a means of organizing a document or portion of a document. For example: • *See, e.g.,* Astoria 1997-1: "Astoria detects and maintains 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

Prior Art Analysis

republishing or for providing an audit trail," at THOM00211908.

- *See, e.g.,* XSoft: "REVISION TRACKING: Because of its sophisticated integration with SGML editors, Astoria maintains revision information on individual elements, and past versions are always available," at THOM00198648.
- *See, e.g.,* Screen shot at THOM00211908.
- *See, e.g.,* Astoria 1997-1: "Astoria provides a mechanism for associating arbitrary, user-definable attributes with Astoria objects. Custom Attributes provide a means for Astoria users to store relevant information directly with any object, providing a robust foundation for object status tracking, and the search and assembly of individual document components." "Astoria users specify a value for the custom attribute and then can search, retrieve, and assemble new documents based on custom attribute values," at THOM00211911.

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

  The EnAct system discloses searching its text-based information using a date attribute, and so using the system would entail performing this method. *See, e.g.*:

  - Arnold-Moore 1997-2, at 178, figure 1 (showing ability to search the text-based information using the date and other search requests)

- **The SCALEplus System:**

  The SCALEplus system discloses searching its text-based information using a date attribute, and so using the system would entail performing this method. *See, e.g.*:

  - Kerr 2000: ¶ 429 (page 11-3), ¶¶ 493–99 (pages 11-14–11-15)
  - SCALEplus UM 2: "Advanced Search Screen" at THOM00221692 and text describing the features on that screen, including the "Date Search Options."

- **The Documentum/Interleaf System:**

  The Documentum/Interleaf system discloses searching its text-based information using a date attribute, and so using the system would entail performing this method. *See, e.g.*:

  - Ovum Interleaf 1996, at 262 (and figure H2.6) (demonstrating the ability to search on date)

- **The Law Desk NY System:**

  The Law Desk NY system a method of using a system which has a means for searching

Prior Art Analysis
its text-based information by date. • **The NY Official Reports System:** The NY Official Reports system a method of using a system which has a means for searching its text-based information by date.
(c) producing results based on the text of the text-based data; • **Agosti 1991:** Agosti 1991 discloses "producing results based on the text of the text-based data." Specifically, Agosti 1991 discloses a user interface that displays search results. For example: • *See generally* 322-324 (Figures 2-8) • *See, e.g.*, "Figure 7. An example of a node: the representation of a Legal Authority document," at 324. • **Anwar 1996:** Anwar 1996 discloses "producing results based on the text of the text-based data." Specifically, Anwar 1996 discloses a method for querying text data. For example: • *See, e.g.*, "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 . . . retrieving. . . 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," at Summary of the Invention. • *See, e.*g., Analysis Scenario 1, starting at 14:64–16:39, describing ways in which a user can formulate multi-dimensional queries using the text data. • **Arnold-Moore 1994:** Arnold-Moore 1994 discloses "producing results based on the text of the text-based data." Specifically, Arnold-Moore 1994 discusses displaying units of text on the screen in response to a search. For example: • *See, e.g.*, "In this context hypertext would allow the note to be visible to the user only after they have selected (usually by pointing and clicking a mouse) a 'button' which is displayed with the text on the screen. This button could be positioned where one might expect an annotation to appear in a paper service. Each separate

Prior Art Analysis

unit of text which is presented on the screen is termed a node," at 3.

- *See, e.g.*, "The querying needs of typical full text databases should be supported including the ability to: select Acts from the database using boolean combinations of words and phrases in the Act…rank Acts according to a measure of similarity to a list of words or passage of text (and select the top ten say)," at 4.

- **Arnold-Moore 1994-2:**

Arnold-Moore 1994-2 discloses "producing results based on the text of the text-based data." Specifically, Arnold-Moore 1994-2 discloses using the described storage system together with a graphical user interface that will display the stored text-based data in response to a search. For example:

- *See, e.g.*, "These databases will need to be searched by attribute. This will, for example, allow a software engineering document that is the right version to be retrieved," at THOM00196608.

- *See, e.g.*, "[I]t is anticipated that SGQL will be primarily used as an API to text and graphical user interfaces rather than used directly by the user. It is presumed that these interfaces will have access to the appropriate DTD's and output specifications so that users will be able to avoid knowing the exact generic identifiers required for every query," at THOM00196615.

- **Arnold-Moore 1997:**

Arnold-Moore 1997 discloses "producing results based on the text of the text-based data." Specifically, Arnold-Moore 1997 discloses providing the legislative drafter with a version of the Act or Regulation to be amended in response to a search. For example:

- *See, e.g.*, "Themis provides the legislative drafter with a version of the Act or Regulation to be amended on which the drafter marks the amendments directly," 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.*, "The drafter can view any Act or search the whole database using Boolean or ranking queries at any time point for which a valid version is stored on the system. This allows a drafter to check out the Principal Act to be amended as it was or will be at a given time," at 59.

- **Arnold-Moore 1997-2:**

Arnold-Moore 1997-2 discloses "producing results based on the text of the text-based

Prior Art Analysis

data." Specifically, Arnold-Moore 1997-2 discusses and shows a user interface that shows the stored nodes of text-based data in response to a search. For example:

- *See generally* 178, 180-181 (Figures 1-5)
- *See, e.g.*, "While most queries are executed through graphical user interfaces," at 177.
- *See, e.g.*, "Using a dual display with a table of contents on one side and the actual provision on the other provides an appropriate compromise (see Figure 3 which shows one of the results from Figure 2). By using SGML to store the Statutes, we can automate the process of fragmenting large documents and only present to the user the parts of the document that the user requests," at 179.

- **Campbell 1988:**

Campbell 1988 discloses "producing results based on the text of the text-based data." Specifically, Campbell 1998 discloses displaying text-based data to the user in response to a search. For example:

- *See, e.g.*, "A filter operation takes a predicate, a version time, and a list of attributes. These operations return a list of objects that satisfy the predicate and a list of requested attributes attached to each object," at 858.
- *See, e.g.*, "Replacement buttons replace the button icon displayed on the screen with the information associated with that button," at 858.

- **Caplinger 1986:**

Caplinger 1986 discloses "producing results based on the text of the text-based data." Specifically, Caplinger 1986 discloses a computer system for that allows a user to search using text-based information. For example:

See, e.g., "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

- **Horne 1997:**

Horne 1997 discloses "producing results based on the text of the text-based data." Specifically, Horne 1997 discloses the display of text-based data to the user in response to a search. For example:

- *See, e.g.*, "SGML is not concerned with how that paragraph is formatted by the appropriate program on the user's computer," at 2.

Prior Art Analysis

- *See, e.g.*, "The program on the user's computer could ignore the repealed text and display the inserted text. But the markup could go further. It could give the dates on which the amendments were made, the dates on which they took effect, and the names of the Acts or Sis which had made them, and the user's program could use this markup to display a statute as it was on a particular date chosen by the user and could offer hypertext cross-references to the amending legislation," at 3.

- **Lo 1996:**

Lo 1996 discloses "producing results based on the text of the legislation." Specifically, Lo 1996 discloses a document management system, including legislation, which queries on the text. For example:

- *See, e.g.*, "The main purpose of indexing is to provide a means through which the data items can be quickly searched and retrieved without the need to compare every item against a query," at 8-9, section 1.2.2.

- *See, e.g.*, "Notes supports full-text searches that allow users to index and search Notes documents based on user queries. An indexed database can be searched for words, phrases, numbers, and dates," at 65.

- *See, e.g.*, "Basic document management features are provided, such as keyword search or full text retrieval," at 66-67.

- *See* "Furthermore, the vector space information retrieval paradigm of the SIM DBS supports querying of the documents based on content," at 113.

- *See* page 27, under the heading "Types of Documents": "In the example of legal databases, the role of auxiliary documents is played by Amendment Acts . . . ."

- **Sacks-Davis 1994:**

Sacks-Davis 1994 discloses "producing results based on the text of the text-based data." Specifically, Sacks-Davis 1994 discloses providing users with access to the stored text-base data through queries. For example:

- *See, e.g.*, "A query language for accessing collections of structured documents, in particular SGML documents, requires support for several classes of query," at THOM00198845.

- **Sacks-Davis 1995:**

Sacks-Davis 1995 discloses "producing results based on the text of the text-based data." Specifically, Sacks-Davis 1995 discusses and shows a user interface that shows the stored nodes of text-based data through queries. For example:

Prior Art Analysis

- *See, e.g.*, "Children can be found with TQL queries on the parent information," at 465.

- *See, e.g.*, "The application is implemented under X windows. An example screen from this application is shown in Fig. 9, showing a person's details including three photographs, some notes, and names of some immediate relatives. At the right is a query window, which is used to find names using soundex or ranking," at 466.

- **Sciore 1991:**

  Sciore 1991 discloses "producing results based on the text of the text-based data." Specifically, Sciore 1991 discloses providing users with access to the stored versioned data through searches. For example:

  - *See, e.g.*, "Our framework provides the means by which a database designer can specify a multi-dimensional logical structure to the version set. This logical structure can then be used to choose versions easily and conveniently," at 367.

- **Sciore 1994:**

  Sciore 1994 discloses "producing results based on the text of the text-based data." Specifically, Sciore 1994 discloses providing users with access to the stored versioned data through searches. For example:

  - *See, e.g.*, "The constructs are high-level, allowing users to access versioned data in exactly the same non-procedural way as unversioned data," at 103.

- **Taylor 1994:**

  Taylor 1994 discloses "producing results based on the text of the text-based data." Specifically, Taylor 1994 discloses a user interface that displays the information to a user. For example:

  - *See, e.g.*, figures 3 & 4, showing display of information, including text.

  - *See, e.g.*, section 5, entitled "Navigation Aids & Clustering," starting on page 242, and discussing the design considerations for the user interface.

- **Travis & Waldt:**

  Travis & Waldt discloses "producing results based on the text of the text-based data." Specifically, Wilson 1988 discusses and shows user interfaces which display the stored text-based data. For example:

  - *See, e.g.*, "Hidden beneath the formatted view of information prepared in tools that we commonly call WYSIWYG (What You See Is What You Get), or rich text, is

Prior Art Analysis

data with buried coding that drives that same formatting…The following formatted view is what an author may wee while editing in a WYSIWYG environment," at 22.

- *See generally* 23 (Figure 4).
- *See, e.g.*, "SGML browsers offer context-sensitive searching capabilities so that the user can quickly access the required information…For example, a search can be defined to allow a user to search for a part number, but only if it is contained in a chapter that was updated after a certain date. Or, a user can have the browser return a list of all sections containing a particular phrase, but only if the phrase is contained in a note. These are examples of context-sensitive searches," at 52-53.
- *See, e.g.*, "It is more common to see a graphical front-end for systems that have traditionally been command-line oriented. Version control systems are no exception. Microsoft SourceSafe has a native graphical front-end in the Windows, Windows NT, and Macintosh versions. This graphical front-end makes it easy to see the structure of a project or group of text files, and to view the current status," at 191.
- *See generally* 191 (Figure 59).
- *See, e.g.*, pages 194–95 (and figure 61), 198 (and figure 64).
- *See, e.g.*, "The loader also makes available to the database parameterized information that can be used later to search and retrieve the appropriate objects. Such parameter information is object identifiers, author names, creation and modification dates, and perhaps some keywords. Most of this information can be obtained by querying the attributes on the element tags in the content of the document object," at 204.

- **Wilkinson 1998:**

Wilkinson 1998 discloses "producing results based on the text of the legislation." Specifically, Wilkinson 1998 discloses searching on a document's contents. For example:

- *See generally* section 5.5, beginning on page 87 (entitled "Access by Content").
- *See, e.g.,* "Characteristic of many document storage systems is the ability to query against the content of stored documents. This important task is enabled by indexing documents' text content. The most prevalent form of text index is the *inverted file*," at 100.
- *See, e.g.,* "To satisfy a query, the query engine uses the inverted index to identify those documents that match the query terms and generates an answer list," at 102.

- **Wilson 1990:**

Prior Art Analysis

Wilson 1990 discloses "producing results based on the text of the text-based data." Specifically, Wilson 1990 discusses and shows user interfaces which display the stored text-based data in response to a search. For example:

- *See, e.g.*, "How the text is displayed varies from hypertext system to hypertext system…In Guide the conventional mode of display for any text is a single linear window. Whenever a definition button or usage button is selected, the button is expanded in place and the display window is reformatted to accommodate the replacement text for the button," at 123.
- *See generally* 124-126 (Figures 1-7).
- *See, e.g.*, "When the user enters the Justus running under Guide he is given a choice of three methods of access: direct access, index of pre-defined terms, and boolean query," at 125.

- **Wilson 1992:**

Wilson discloses "producing results based on the text of the text-based data." Specifically, Wilson 1992 discusses and shows user interfaces which display the stored text-based data. For example:

- *See, e.g.*, "This label can be defined as a node icon or, in the Guide hypertext system, a definition button. The replacement text for this definition button is the actual words of the paragraph; for paragraph 6(2)(a)," at 161.
- *See generally* 163-164, 169-174, 178-182 (Figures 1-15).
- *See generally* 183.

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

The Westlaw/Westmate system displays text-based legislation in response to a search, and so using the system would entail performing this method. *See, e.g.*:

- Wren 1994: 141–42 (displaying a statutory sections in response to a search)
- Essential Guide: 139–40 (displaying a statutory sections in response to a search)
- Johnson 1991: *generally* and 84–92 (including figure 4.11)
- The Essential Guide 1996, at 43: "WESTLAW processes your description and displays the 20 documents most closely matching the concepts in your description . . ."
- The Essential Guide 1996, at 43: "You can review the documents you retrieve using

Prior Art Analysis

standard WESTLAW browsing commands. When you browse documents retrieved by a Natural Language description in term mode, the five portions of each document that most closely match your description are displayed. To view the portion of each document most closely matching your description, type **best** or **b** to browse your documents in best mode."

- *See generally* The Essential Guide 1996, at Chapter "4.6 Using Terms and Connectors"

- **The Pre-1997 Premise System**:

The Premise system displays text-based legislation in response to a search, and so using the system would entail performing this method. *See, e.g.*:

- Premise Software & Statutes: Select "Search/Search Book…" menu item, and conduct a search of the Statutes to see the text of the document retrieved by the search.

- Premise Research: Chapter 7 *generally* ("Retrieving Documents Using Descriptive Words")

- **The Astoria System (pre-1997):**

The Astoria System contained a means for searching portions and amended portions of text-based data using words or phrases. Use of this system therefore constitutes a method for producing results based on the text of the text-based data. For example:

- *See, e.g.,* Astoria 1997-1: "Astoria provides a multilingual engine that lets users search on document content, structure, attributes, and version information," at THOM00211909.

- *See, e.g.,* XSoft Astoria: "The search engine allows context-sensitive searching. For example, it can find the phrase "lower taxes," at THOM00198652.

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

The legislation within the EnAct system displays text-based legislation in response to a search, and so using the system would entail performing this method. *See, e.g.*:

- Arnold-Moore 1997-2, at 178, 180, figures 2 & 3 (and p. 179 saying "(see Figure 3 which shows one of the results from Figure 2)").

- **The SCALEplus System:**

The SCALEplus system displays text-based legislation in response to a search, and so