US 20060200443A1

(19) **United States**

(12) **Patent Application Publication** (10) **Pub. No.: US 2006/0200443 A1**

Kahn et al. (43) **Pub. Date: Sep. 7, 2006**

(54) **BOOKMARKS AND SUBSCRIPTIONS FOR FEEDS**

(76) Inventors: **Jessica Kahn**, San Francisco, CA (US); **Jens Alfke**, San Jose, CA (US); **Sarah Anne Wilkin**, Menlo Park, CA (US); **Albert Riley Howard JR.**, Sunnyvale, CA (US); **Steven P. Jobs**, Palo Alto, CA (US); **Scott James Forstall**, Mountain View, CA (US); **Gregory N. Christie**, San Jose, CA (US); **Stephen O. Lemay**, San Francisco, CA (US); **Donald Dale Melton**, San Carlos, CA (US); **Wayne Russell Loofbourrow**, San Jose, CA (US)

Correspondence Address:
FENWICK & WEST LLP
SILICON VALLEY CENTER
801 CALIFORNIA STREET
MOUNTAIN VIEW, CA 94041 (US)

(21) Appl. No.: **11/371,153**

(22) Filed: **Mar. 7, 2006**

Related U.S. Application Data

(63) Continuation of application No. 11/105,637, filed on Apr. 13, 2005.

(60) Provisional application No. 60/582,868, filed on Jun. 25, 2004.

Publication Classification

(51) **Int. Cl.**
G06F 17/30 (2006.01)

(52) **U.S. Cl.** **707/1**

(57) **ABSTRACT**

Techniques for detecting, managing, and presenting syndication XML (feeds) are disclosed. In one embodiment, a web browser automatically determines that a web site is publishing feeds and notifies the user, who can then access the feed easily. In another embodiment, a browser determines that a web page or feed is advertising relationship XML, and displays information about the people identified in the relationship XML. In yet another embodiment, a browser determines that a file contains a feed and enables the user to view it in a user-friendly way. In yet another embodiment, feed state information is stored in a repository that is accessible by applications that are used to view the feed. In yet another embodiment, if a feed's state changes, an application notifies the repository, and the state is updated. In yet another embodiment, a feed is parsed and stored in a structured way.

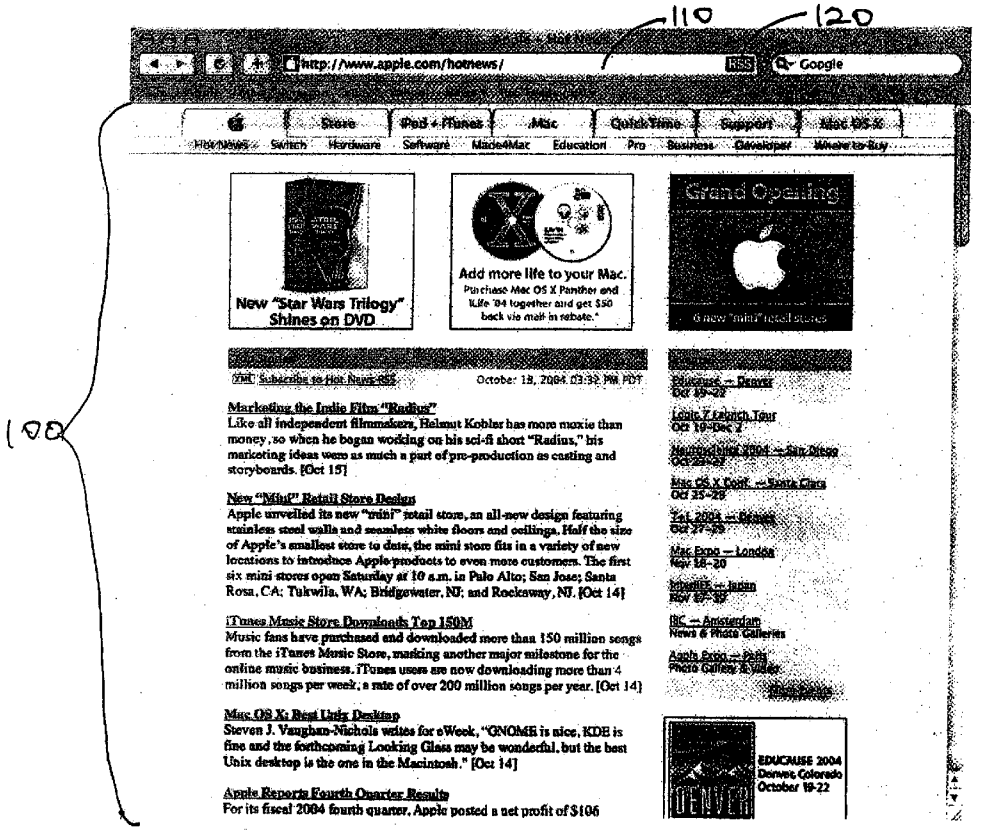
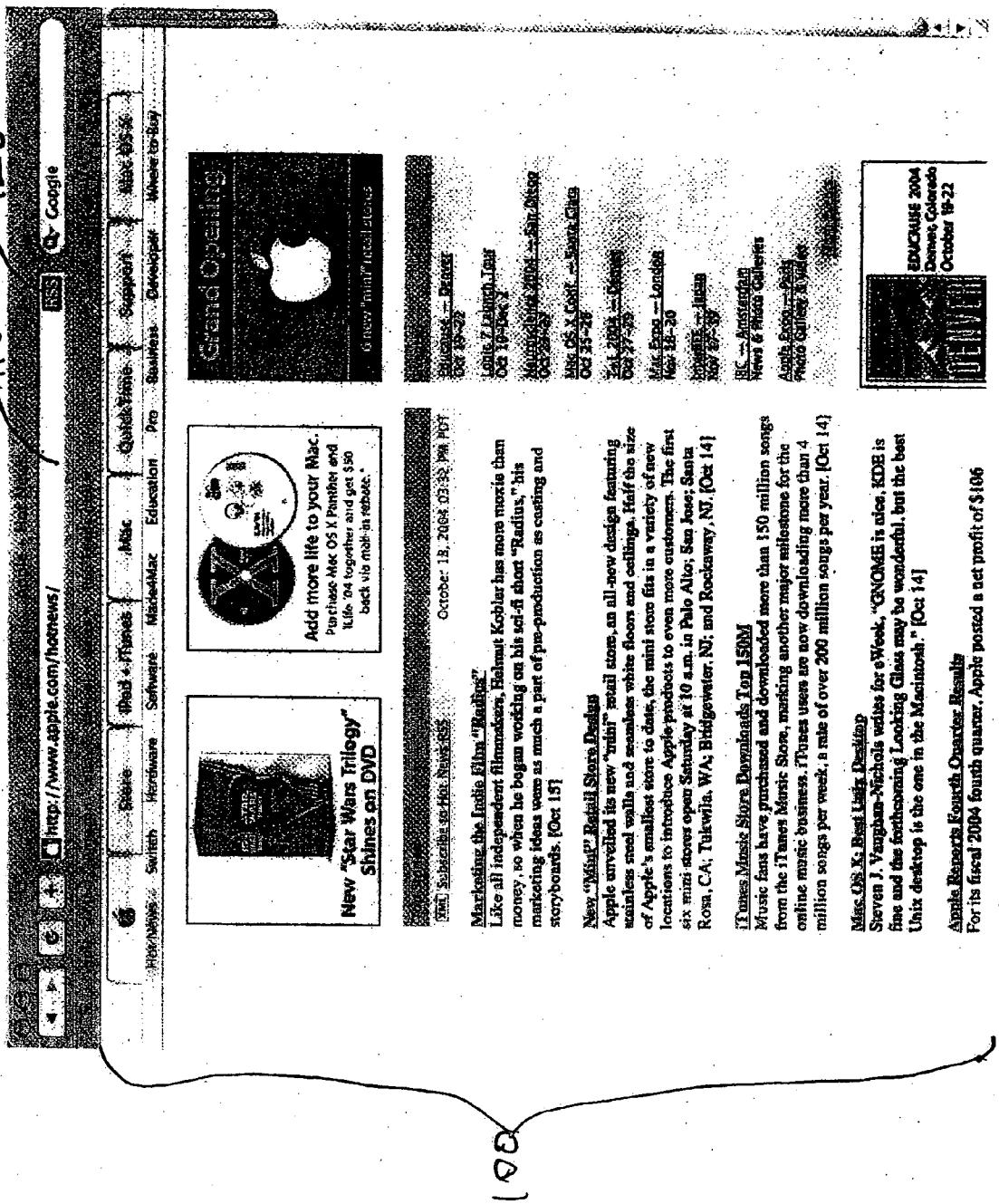


FIG. 1A


120
110
100
http://www.apple.com/hotnews/
Google
Store
iPod + iTunes
Mac
QuickTime
Support
Mac OS X
Hot News
Switch
Hardware
Software
Macie4Mac
Education
Pro
Business
Developer
Where to Buy
New "Star Wars Trilogy" Shines on DVD
Add more life to your Mac.
Purchase Mac OS X Panther and iLife '04 together and get $50 back via mail-in rebate.*
Grand Opening
Marketing the Indie Film "Radius"
Like all independent filmmakers, Helmut Kobler has more moxie than money, so when he began working on his sci-fi short "Radius," his marketing ideas were as much a part of pre-production as casting and storyboards. [Oct 15]
New "Mini" Retail Store Design
Apple unveiled its new "mini" retail store, an all-new design featuring stainless steel walls and seamless white floors and ceilings. Half the size of Apple's smallest store to date, the mini store fits in a variety of new locations to introduce Apple products to even more customers. The first six mini stores open Saturday at 10 a.m. in Palo Alto; San Jose; Santa Rosa, CA; Tukwila, WA; Bridgewater, NJ; and Rockaway, NJ. [Oct 14]
iTunes Music Store Downloads Top 150M
Music fans have purchased and downloaded more than 150 million songs from the iTunes Music Store, marking another major milestone for the online music business. iTunes users are now downloading more than 4 million songs per week, a rate of over 200 million songs per year. [Oct 14]
Mac OS X: Best Unix Desktop
Steven J. Vaughan-Nichols writes for eWeek, "GNOME is nice, KDE is fine and the forthcoming Looking Glass may be wonderful, but the best Unix desktop is the one in the Macintosh." [Oct 14]
Apple Reports Fourth Quarter Results
For its fiscal 2004 fourth quarter, Apple posted a net profit of $106
EDUCAUSE 2004
Denver, Colorado
October 19-22

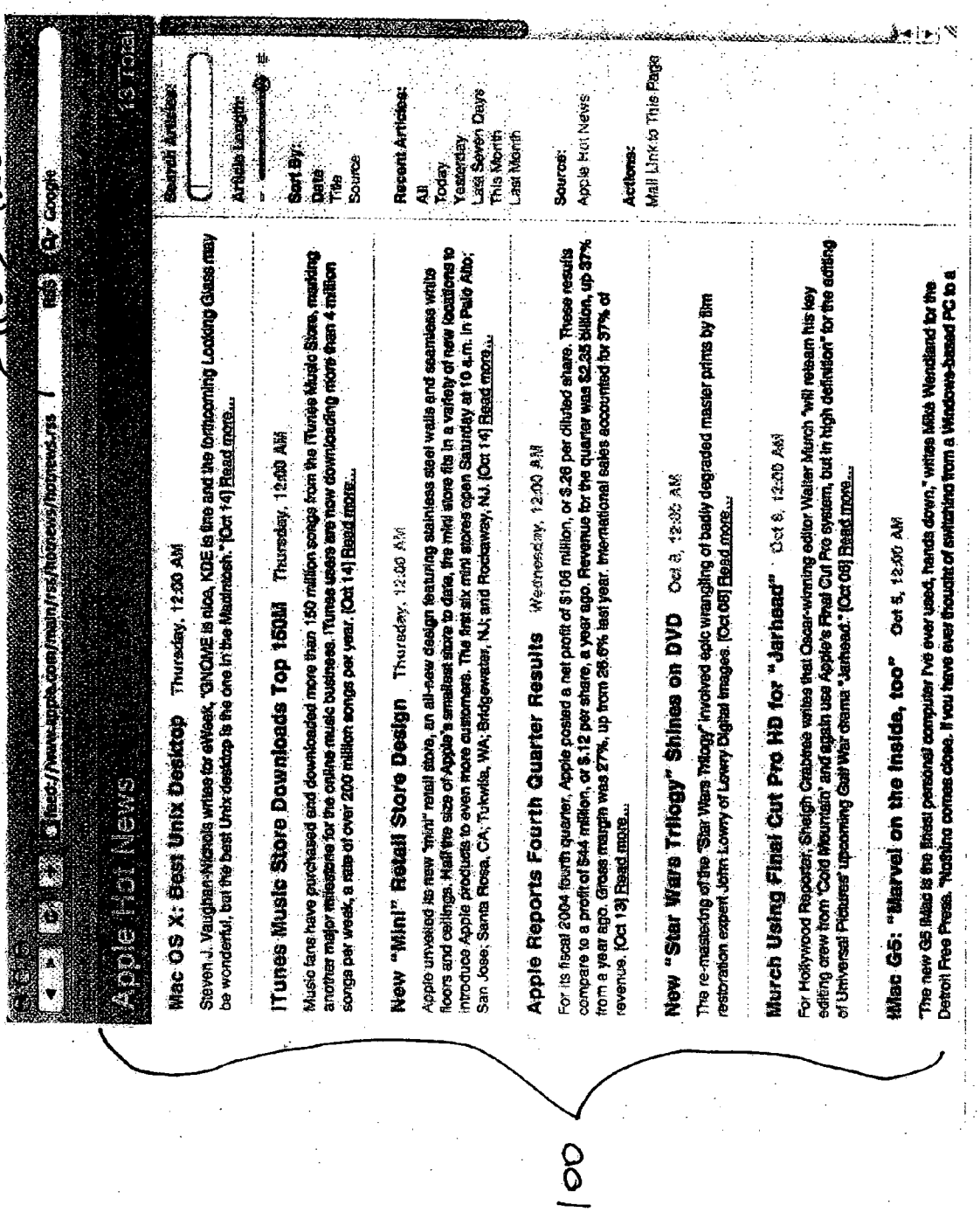

110
120
100
Apple Hot News
13 Total
Search Articles:
Article Length:
Sort By:
Date
Title
Source
Recent Articles:
All
Today
Yesterday
Last Seven Days
This Month
Last Month
Source:
Apple Hot News
Actions:
Mail Link to This Page
Mac OS X: Best Unix Desktop Thursday, 12:00 AM
Steven J. Vaughan-Nichols writes for eWeek, "GNOME is nice, KDE is fine and the forthcoming Looking Glass may be wonderful, but the best Unix desktop is the one in the Mackintosh." [Oct 14] Read more...
iTunes Music Store Downloads Top 150M Thursday, 12:00 AM
Music fans have purchased and downloaded more than 150 million songs from the iTunes Music Store, marking another major milestone for the online music business. iTunes users are now downloading more than 4 million songs per week, a rate of over 200 million songs per year. [Oct 14] Read more...
New "Mini" Retail Store Design Thursday, 12:00 AM
Apple unveiled its new "mini" retail store, an all-new design featuring stainless steel walls and seamless white floors and ceilings. Half the size of Apple's smallest store to date, the mini store fits in a variety of new locations to introduce Apple products to even more customers. The first six mini stores open Saturday at 10 a.m. in Palo Alto; San Jose; Santa Rosa, CA; Tukwila, WA; Bridgewater, NJ; and Rockaway, NJ. [Oct 14] Read more...
Apple Reports Fourth Quarter Results Wednesday, 12:00 AM
For its fiscal 2004 fourth quarter, Apple posted a net profit of $106 million, or $.26 per diluted share. These results compare to a profit of $44 million, or $.12 per share, a year ago. Revenue for the quarter was $2.35 billion, up 37% from a year ago. Gross margin was 27%, up from 26.6% last year. International sales accounted for 37% of revenue. [Oct 13] Read more...
New "Star Wars Trilogy" Shines on DVD Oct 8, 12:00 AM
The re-mastering of the "Star Wars Trilogy" involved epic wrangling of badly degraded master prints by film restoration expert John Lowry of Lowry Digital Images. [Oct 08] Read more...
Murch Using Final Cut Pro HD for "Jarhead" Oct 8, 12:00 AM
For Hollywood Reporter, Sheigh Crabtree writes that Oscar-winning editor Walter Murch "will retrain his key editing crew from 'Cold Mountain' and again use Apple's Final Cut Pro system, but in high definition" for the editing of Universal Pictures' upcoming Gulf War drama "Jarhead." [Oct 08] Read more...
iMac G5: "Marvel on the inside, too" Oct 5, 12:00 AM
"The new G5 iMac is the finest personal computer I've ever used, hands down," writes Mike Wendland for the Detroit Free Press. "Nothing comes close. If you have ever thought of switching from a Windows-based PC to a

FIG. 1B

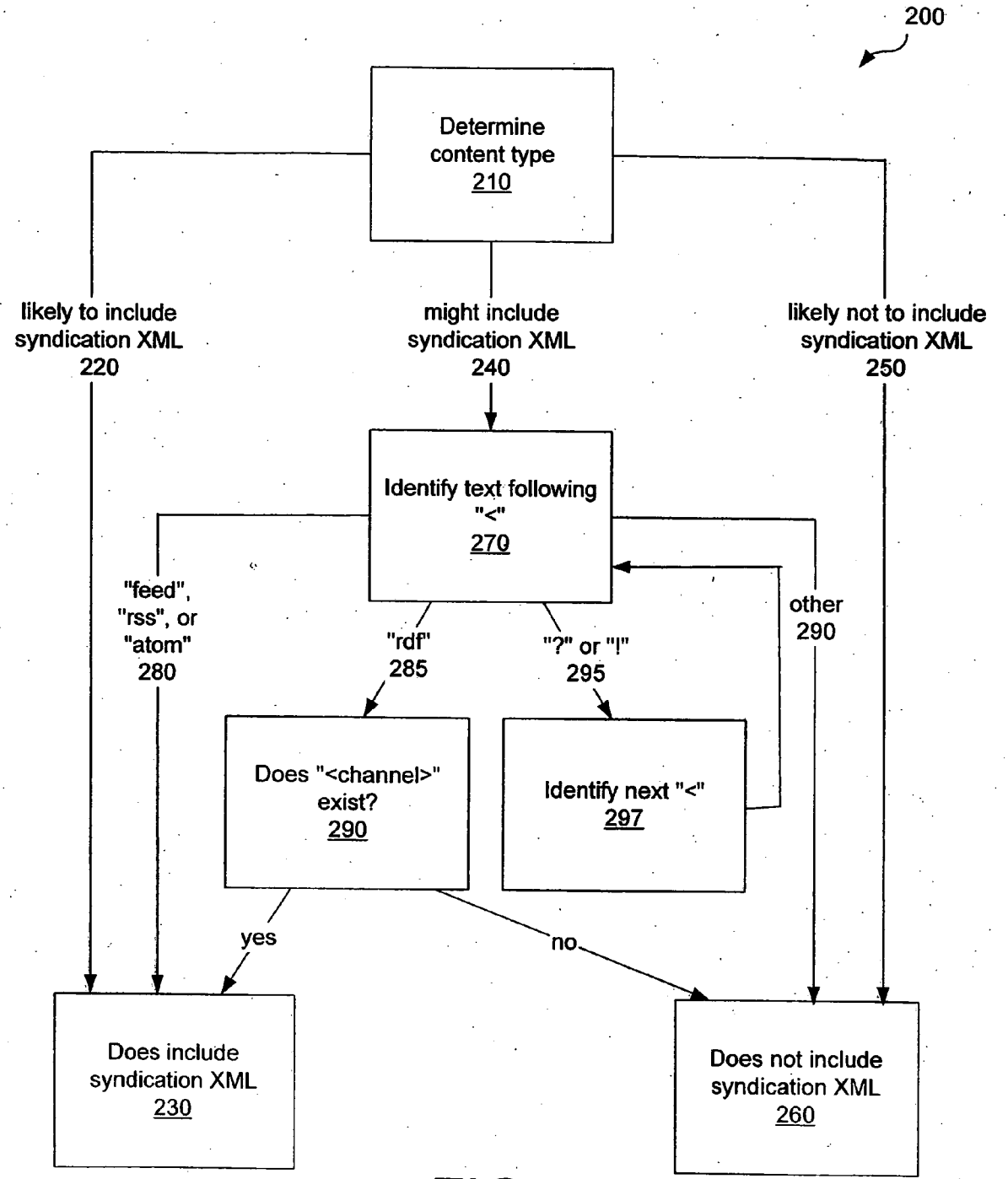

200
Determine content type 210
likely to include syndication XML 220
might include syndication XML 240
likely not to include syndication XML 250
Identify text following "<" 270
"feed", "rss", or "atom" 280
"rdf" 285
"?" or "!" 295
other 290
Does "<channel>" exist? 290
Identify next "<" 297
yes
no
Does include syndication XML 230
Does not include syndication XML 260

FIG. 2

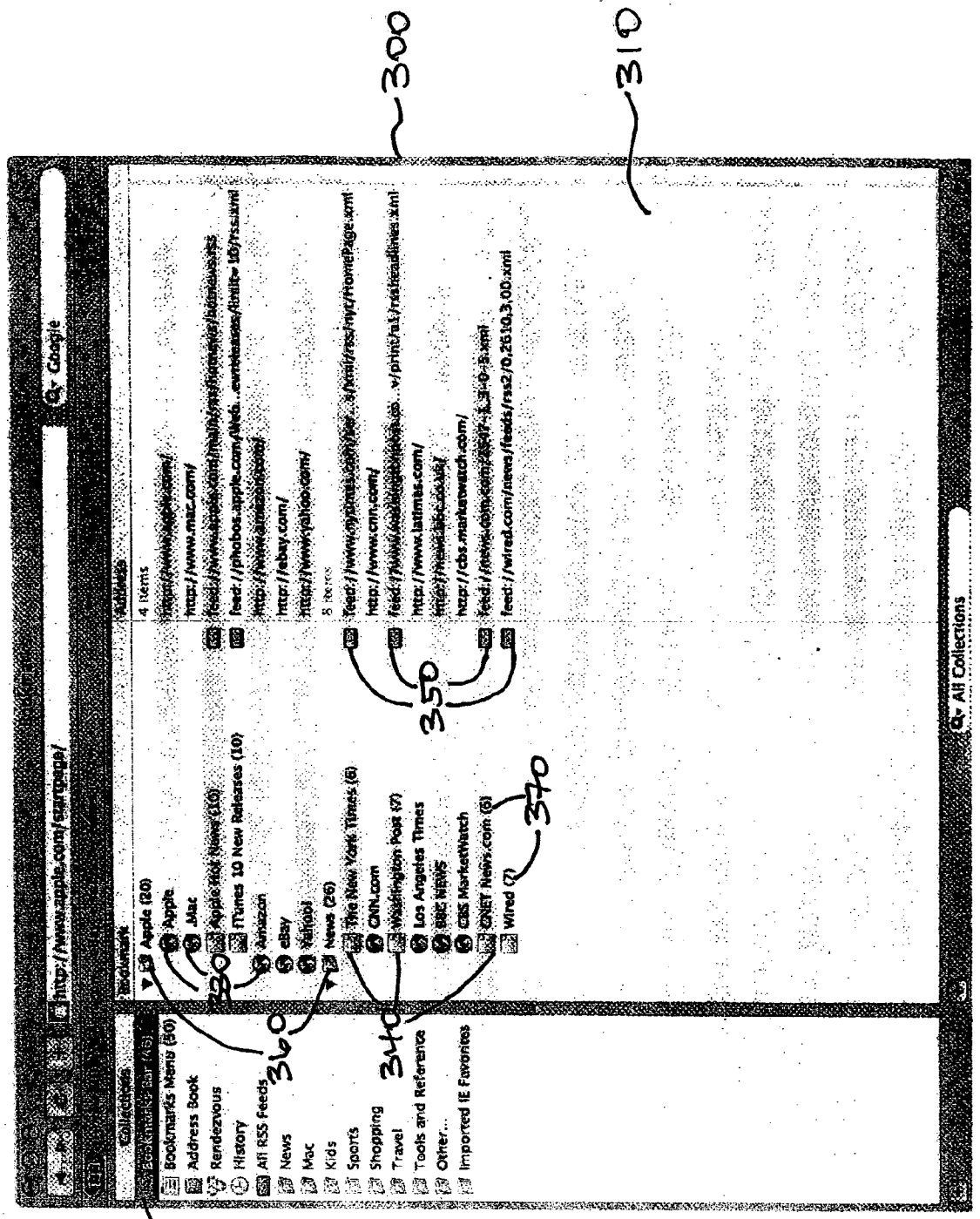

300
310
320
330
340
350
360
370
Collections
Bookmarks Bar (50)
Bookmarks Menu (50)
Address Book
Rendezvous
History
All RSS Feeds
News
Mac
Kids
Sports
Shopping
Travel
Tools and Reference
Other...
Imported IE Favorites
Bookmark
Apple (20)
Apple
Mac
Apple Hot News (10)
iTunes 10 New Releases (10)
Amazon
eBay
Yahoo!
News (26)
The New York Times (6)
CNN.com
Washington Post (7)
Los Angeles Times
BBC NEWS
CBS MarketWatch
CNET News.com (6)
Wired (7)
Address
4 items
8 items
All Collections

FIG. 3A

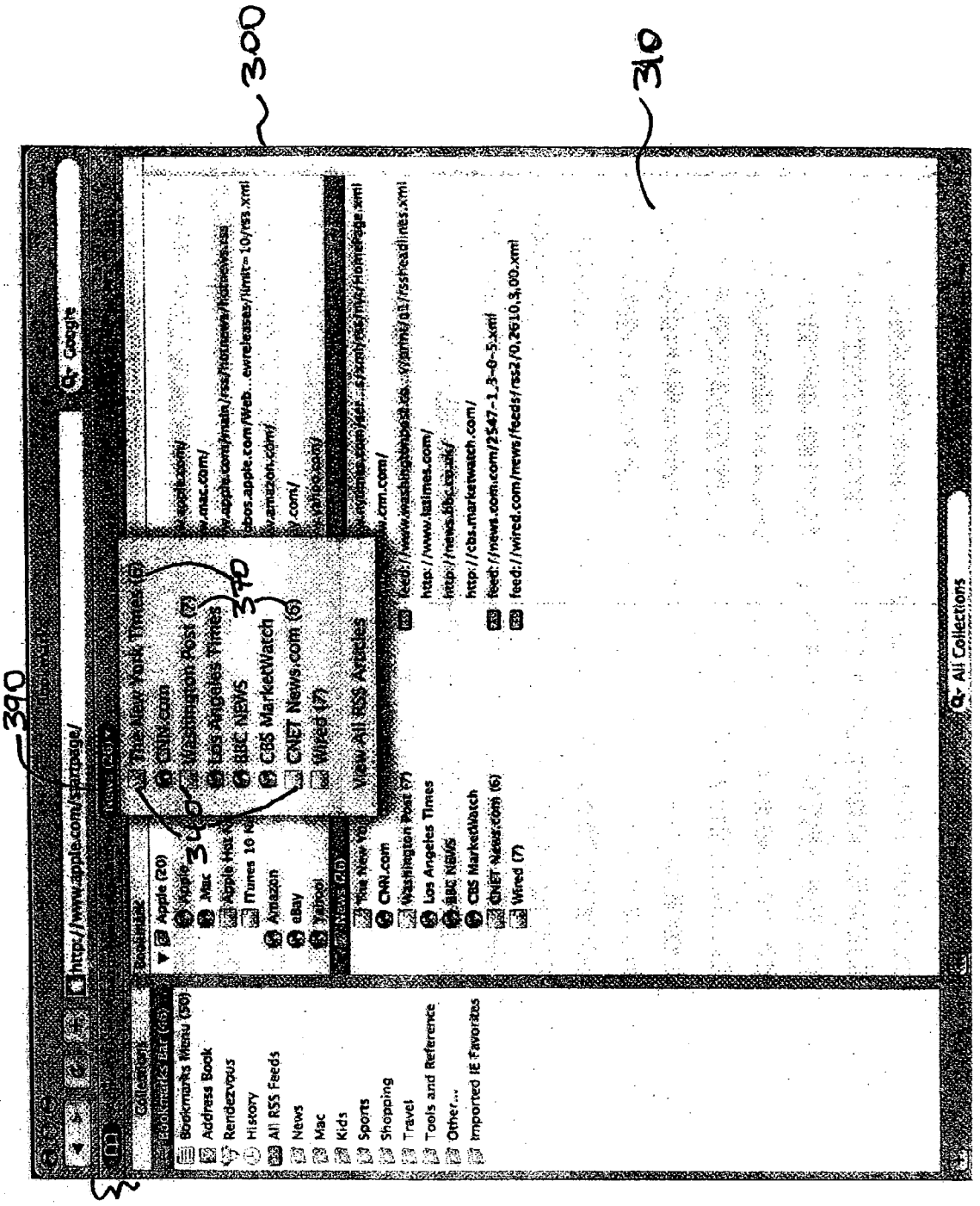

300
310
390
380
340
370
http://www.apple.com/startpage/
Google
Collections
Bookmarks Bar (66)
Bookmarks Menu (50)
Address Book
Rendezvous
History
All RSS Feeds
News
Mac
Kids
Sports
Shopping
Travel
Tools and Reference
Other...
Imported IE Favorites
Apple (20)
Apple
Mac
Amazon
eBay
Yahoo!
The New York Times (6)
CNN.com
Washington Post (7)
Los Angeles Times
BBC NEWS
CBS MarketWatch
CNET News.com (6)
Wired (7)
View All RSS Articles
feed://www.washingtonpost.com/...
http://www.latimes.com/
http://news.bbc.co.uk/
http://cbs.marketwatch.com/
feed://news.com.com/2547-1_3-0-5.xml
feed://wired.com/news/feeds/rss2/0,2610,3,00.xml
All Collections

FIG. 3B

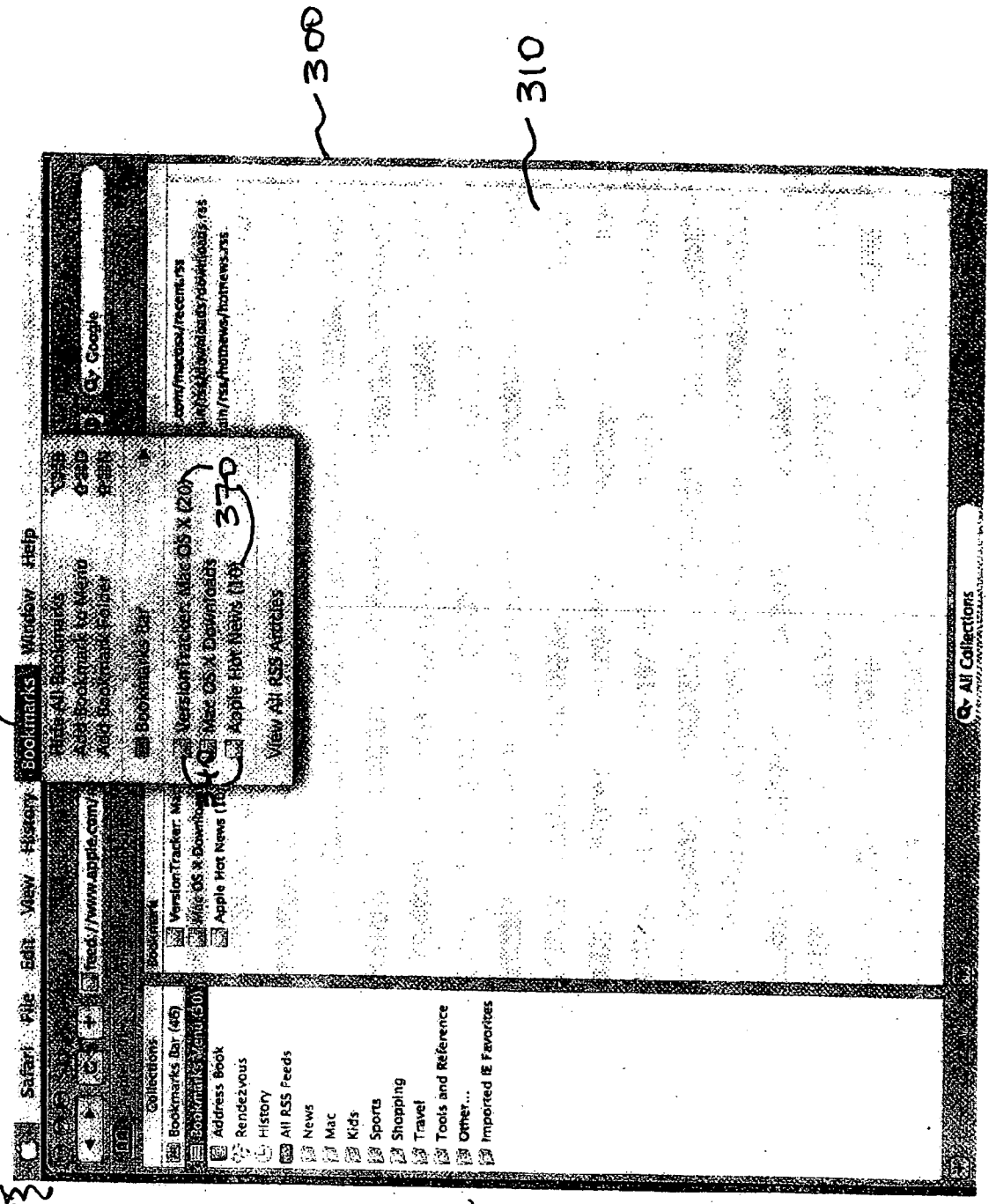

395
397
300
310
370
Safari File Edit View History Bookmarks Window Help
Collections
Bookmarks Bar (46)
Address Book
Rendezvous
History
All RSS Feeds
News
Mac
Kids
Sports
Shopping
Travel
Tools and Reference
Other...
Imported IE Favorites
All Collections

FIG. 3C

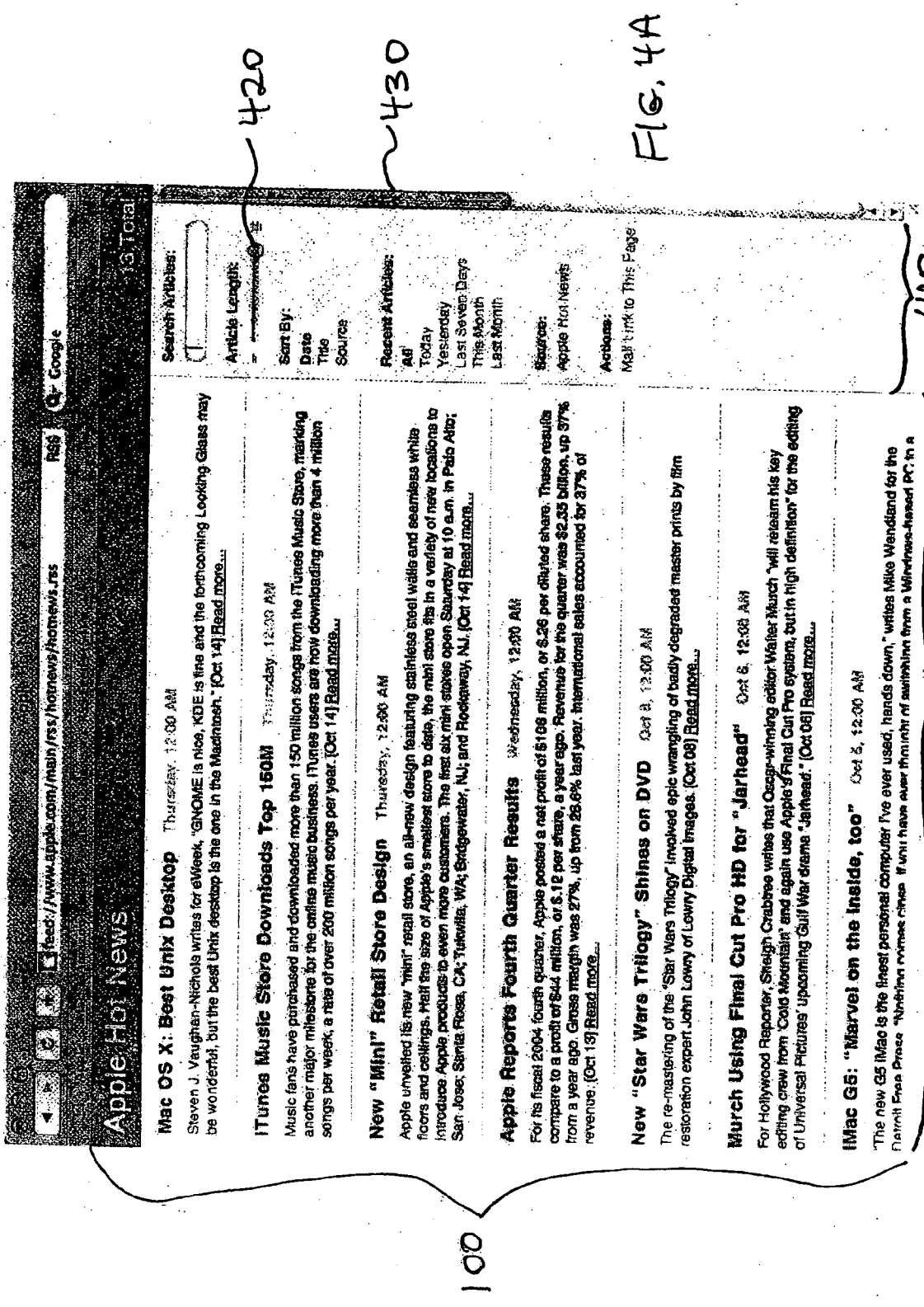

Apple Hot News
Mac OS X: Best Unix Desktop
iTunes Music Store Downloads Top 150M
New "Mini" Retail Store Design
Apple Reports Fourth Quarter Results
New "Star Wars Trilogy" Shines on DVD
Murch Using Final Cut Pro HD for "Jarhead"
iMac G5: "Marvel on the Inside, too"
Search Articles:
Article Length:
Sort By:
Date
Title
Source
Recent Articles:
All
Today
Yesterday
Last Seven Days
This Month
Last Month
Sources:
Apple Hot News
Actions:
Mail Link to This Page
100
400
410
420
430

FIG. 4A

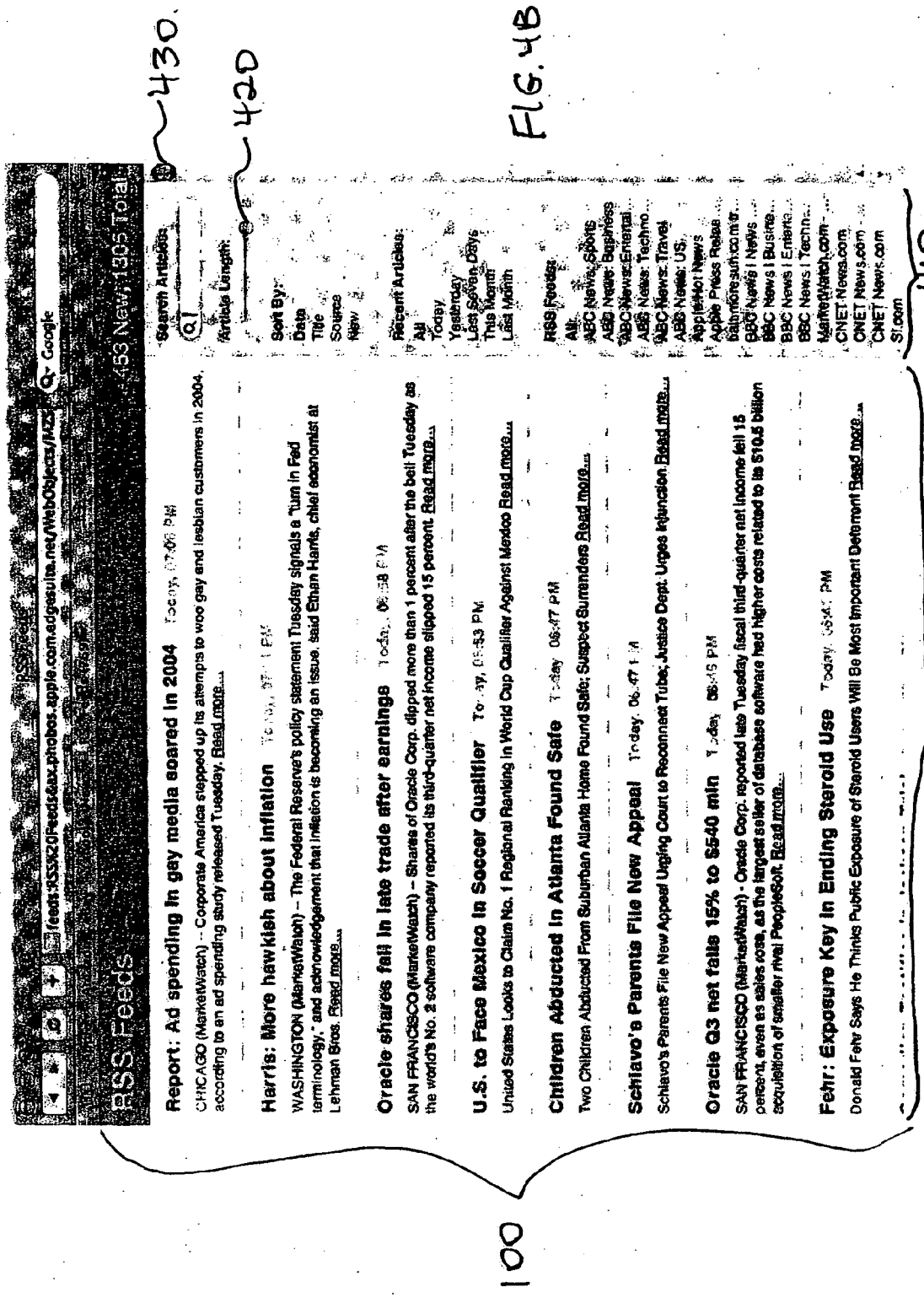

RSS Feeds
453 New, 1305 Total
Report: Ad spending in gay media soared in 2004
CHICAGO (MarketWatch) -- Corporate America stepped up its attempts to woo gay and lesbian customers in 2004, according to an ad spending study released Tuesday. Read more...
Harris: More hawkish about inflation
WASHINGTON (MarketWatch) -- The Federal Reserve's policy statement Tuesday signals a "turn in Fed terminology," and acknowledgement that inflation is becoming an issue, said Ethan Harris, chief economist at Lehman Bros. Read more...
Oracle shares fall in late trade after earnings
SAN FRANCISCO (MarketWatch) -- Shares of Oracle Corp. dipped more than 1 percent after the bell Tuesday as the world's No. 2 software company reported its third-quarter net income slipped 15 percent. Read more...
U.S. to Face Mexico in Soccer Qualifier
United States Looks to Claim No. 1 Regional Ranking in World Cup Qualifier Against Mexico Read more...
Children Abducted in Atlanta Found Safe
Two Children Abducted From Suburban Atlanta Home Found Safe; Suspect Surrenders Read more...
Schiavo's Parents File New Appeal
Schiavo's Parents File New Appeal Urging Court to Reconnect Tube; Justice Dept. Urges Injunction Read more...
Oracle Q3 net falls 15% to $540 mln
SAN FRANCISCO (MarketWatch) - Oracle Corp. reported late Tuesday fiscal third-quarter net income fell 15 percent, even as sales rose, as the largest seller of database software had higher costs related to its $10.8 billion acquisition of smaller rival PeopleSoft. Read more...
Fehr: Exposure Key In Ending Steroid Use
Donald Fehr Says He Thinks Public Exposure of Steroid Users Will Be Most Important Deterrent Read more...
Search Articles
Article Length
Sort By:
Date
Title
Source
New
Recent Articles:
All
Today
Yesterday
Last Seven Days
This Month
Last Month
RSS Feeds:
All
ABC News: Sports
ABC News: Business
ABC News: Entertai...
ABC News: Techno...
ABC News: Travel
ABC News: US
AppleInsider News
Apple Press Relea...
BBC News | News ...
BBC News | Busine...
BBC News | Enterta...
BBC News | Techn...
MarketWatch.com ...
CNET News.com
CNET News.com
SI.com
100
400
410
420
430

FIG. 4B

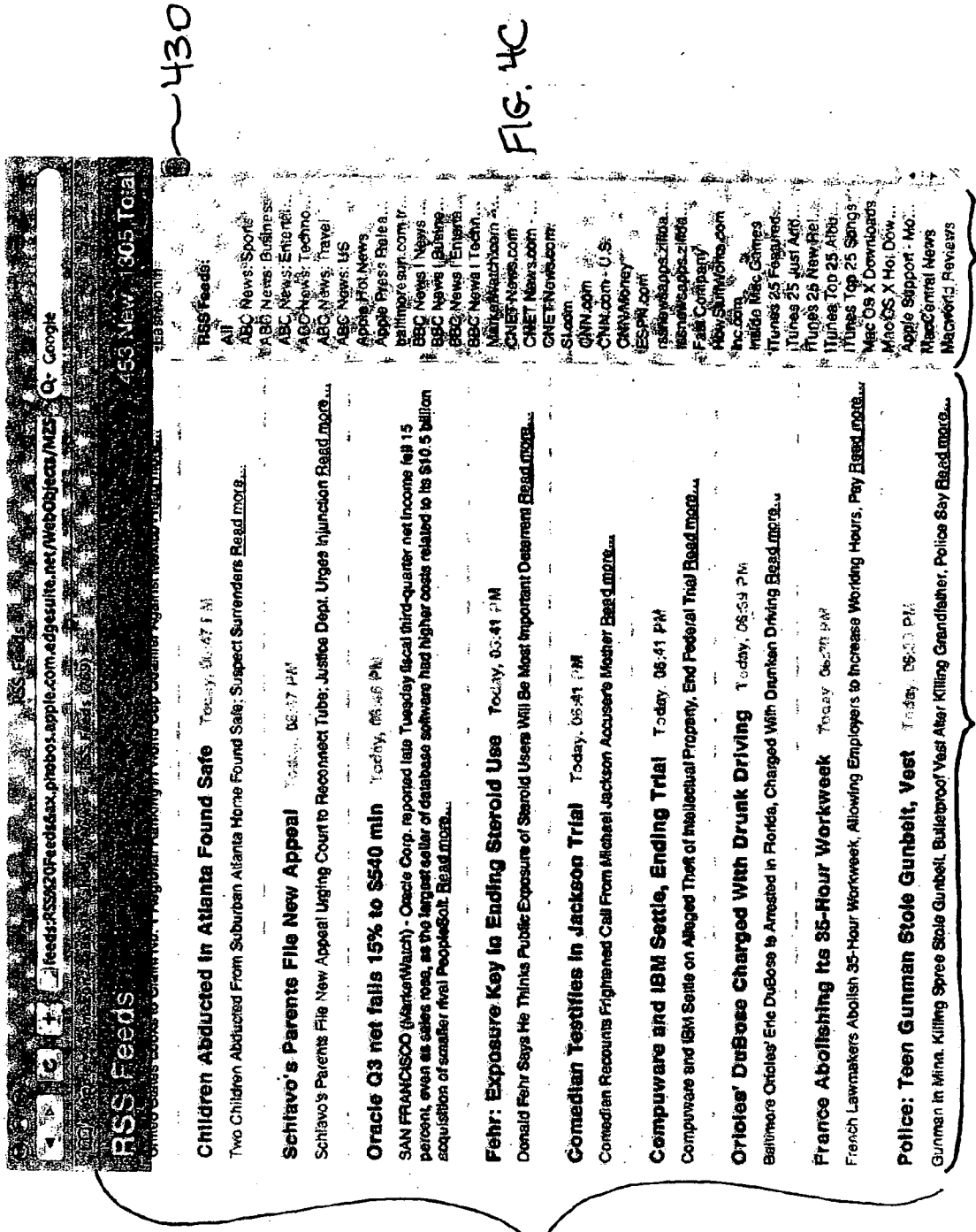

RSS Feeds
453 New, 1305 Total
Children Abducted In Atlanta Found Safe
Two Children Abducted From Suburban Atlanta Home Found Safe; Suspect Surrenders Read more...
Schiavo's Parents File New Appeal
Schiavo's Parents File New Appeal Urging Court to Reconnect Tube; Justice Dept. Urges Injunction Read more...
Oracle Q3 net falls 15% to $540 mln
SAN FRANCISCO (MarketWatch) - Oracle Corp. reported late Tuesday fiscal third-quarter net income fell 15 percent, even as sales rose, as the largest seller of database software had higher costs related to its $10.5 billion acquisition of smaller rival PeopleSoft Read more...
Fehr: Exposure Key In Ending Steroid Use
Donald Fehr Says He Thinks Public Exposure of Steroid Users Will Be Most Important Deterrent Read more...
Comedian Testifies In Jackson Trial
Comedian Recounts Frightened Call From Michael Jackson Accusers Mother Read more...
Compuware and IBM Settle, Ending Trial
Compuware and IBM Settle on Alleged Theft of Intellectual Property, End Federal Trial Read more...
Orioles' DuBose Charged With Drunk Driving
Baltimore Orioles' Eric DuBose Is Arrested in Florida, Charged With Drunken Driving Read more...
France Abolishing Its 35-Hour Workweek
French Lawmakers Abolish 35-Hour Workweek, Allowing Employers to Increase Working Hours, Pay Read more...
Police: Teen Gunman Stole Gunbelt, Vest
Gunman in Mine, Killing Spree Stole Gunbelt, Bulletproof Vest After Killing Grandfather, Police Say Read more...
RSS Feeds:
All
ABC News: Sports
ABC News: Business
ABC News: Entertai...
ABC News: Techno...
ABC News: Travel
ABC News: US
Apple Hot News
Apple Press Relea...
baltimoresun.com:tr...
BBC News | News ...
BBC News | Busine...
BBC News | Enterta...
BBC News | Techn...
MarketWatch.com: ...
CNET News.com
CNET News.com
Slashdot
CNN.com
CNN.com - U.S.
CNN/Money
ESPN.com
rssnewsapps.zfeed...
rssnewsapps.zfeed...
Fast Company
How Stuff Works.com
Inc.com
Inside Mac Games
iTunes 25 Featured...
iTunes 25 Just Add...
iTunes 25 New Rel...
iTunes Top 25 Albu...
iTunes Top 25 Songs
Mac Os X Downloads
MacOS X Hot Dow...
Apple Support - Mo...
MacCentral News
Macworld Reviews
100
400
410
430

FIG. 4C

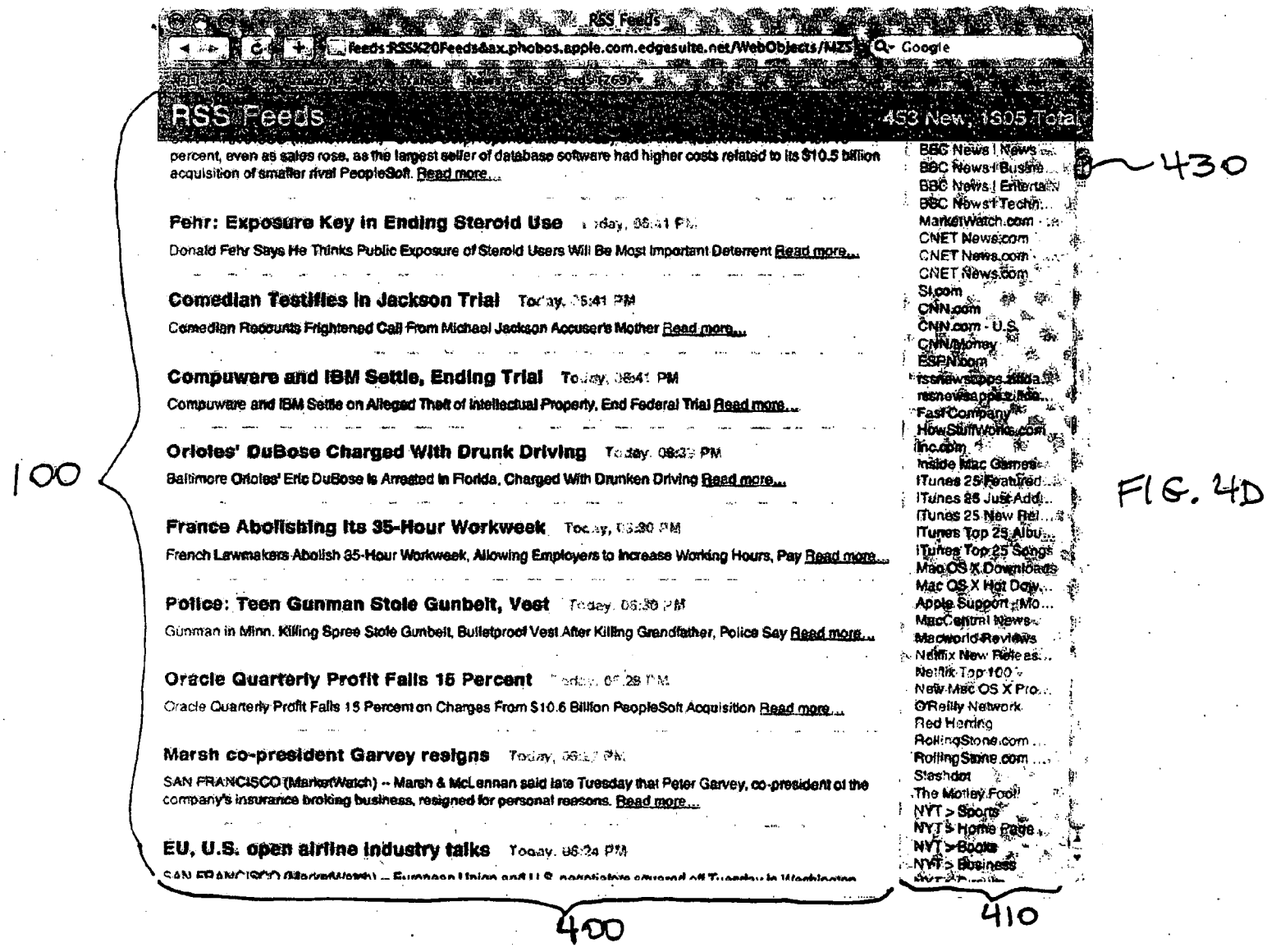

RSS Feeds
feeds:RSS%20Feeds&ax.phobos.apple.com.edgesuite.net/WebObjects/MZS
Google
RSS Feeds
453 New, 1305 Total
percent, even as sales rose, as the largest seller of database software had higher costs related to its $10.5 billion acquisition of smaller rival PeopleSoft. Read more...
Fehr: Exposure Key in Ending Steroid Use
Donald Fehr Says He Thinks Public Exposure of Steroid Users Will Be Most Important Deterrent Read more...
Comedian Testifies in Jackson Trial
Comedian Recounts Frightened Call From Michael Jackson Accuser's Mother Read more...
Compuware and IBM Settle, Ending Trial
Compuware and IBM Settle on Alleged Theft of Intellectual Property, End Federal Trial Read more...
Orioles' DuBose Charged With Drunk Driving
Baltimore Orioles' Eric DuBose Is Arrested in Florida, Charged With Drunken Driving Read more...
France Abolishing Its 35-Hour Workweek
French Lawmakers Abolish 35-Hour Workweek, Allowing Employers to Increase Working Hours, Pay Read more...
Police: Teen Gunman Stole Gunbelt, Vest
Gunman in Minn. Killing Spree Stole Gunbelt, Bulletproof Vest After Killing Grandfather, Police Say Read more...
Oracle Quarterly Profit Falls 15 Percent
Oracle Quarterly Profit Falls 15 Percent on Charges From $10.6 Billion PeopleSoft Acquisition Read more...
Marsh co-president Garvey resigns
SAN FRANCISCO (MarketWatch) -- Marsh & McLennan said late Tuesday that Peter Garvey, co-president of the company's insurance broking business, resigned for personal reasons. Read more...
EU, U.S. open airline industry talks
BBC News | News
BBC News | Busine...
BBC News | Enterta...
BBC News | Techn...
MarketWatch.com
CNET News.com
CNET News.com
CNET News.com
SI.com
CNN.com
CNN.com - U.S.
CNN/Money
ESPN.com
Fast Company
HowStuffWorks.com
Inc.com
Inside Mac Games
iTunes 25 Featured...
iTunes 25 Just Add...
iTunes 25 New Rel...
iTunes Top 25 Albu...
iTunes Top 25 Songs
Mac OS X Downloads
Mac OS X Hot Dow...
Apple Support (Mo...
MacCentral News
Macworld Reviews
Netflix New Releas...
Netflix Top 100
New Mac OS X Pro...
O'Reilly Network
Red Herring
RollingStone.com
RollingStone.com
Slashdot
The Motley Fool
NYT > Sports
NYT > Home Page
NYT > Books
NYT > Business
100
400
410
430

FIG. 4D

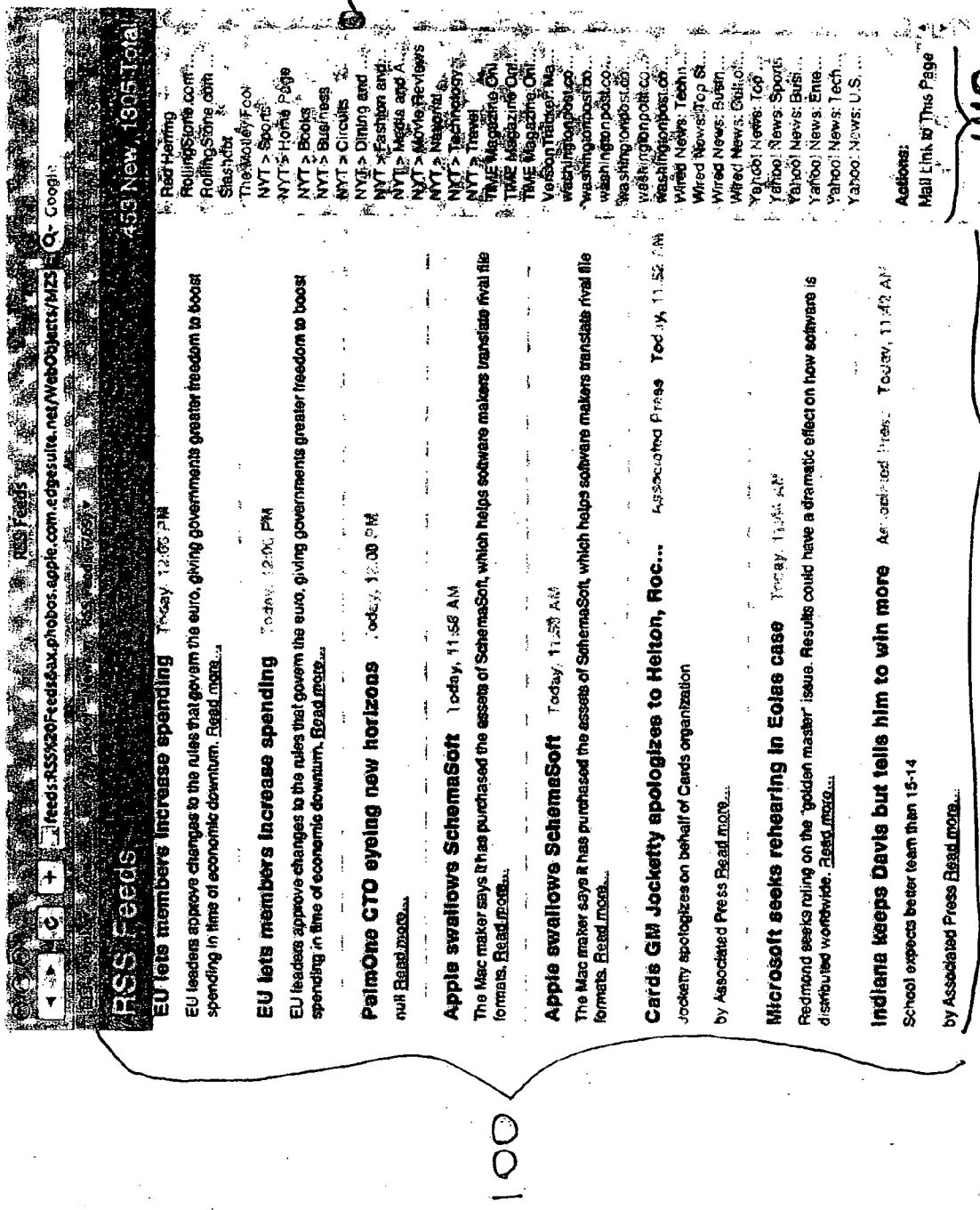

RSS Feeds
453 New, 1905 Total
EU lets members increase spending
EU leaders approve changes to the rules that govern the euro, giving governments greater freedom to boost spending in time of economic downturn. Read more...
PalmOne CTO eyeing new horizons
null Read more...
Apple swallowe SchemaSoft
The Mac maker says it has purchased the assets of SchemaSoft, which helps software makers translate rival file formats. Read more...
Cards GM Jocketty apologizes to Helton, Roc...
Jocketty apologizes on behalf of Cards organization
by Associated Press Read more...
Microsoft seeks rehearing in Eolas case
Redmond seeks ruling on the 'golden master' issue. Results could have a dramatic effect on how software is distributed worldwide. Read more...
Indiana keeps Davis but tells him to win more
School expects better team than 15-14
by Associated Press Read more...
Actions:
Mail Link to This Page
100
400
410
430

FIG. 4E

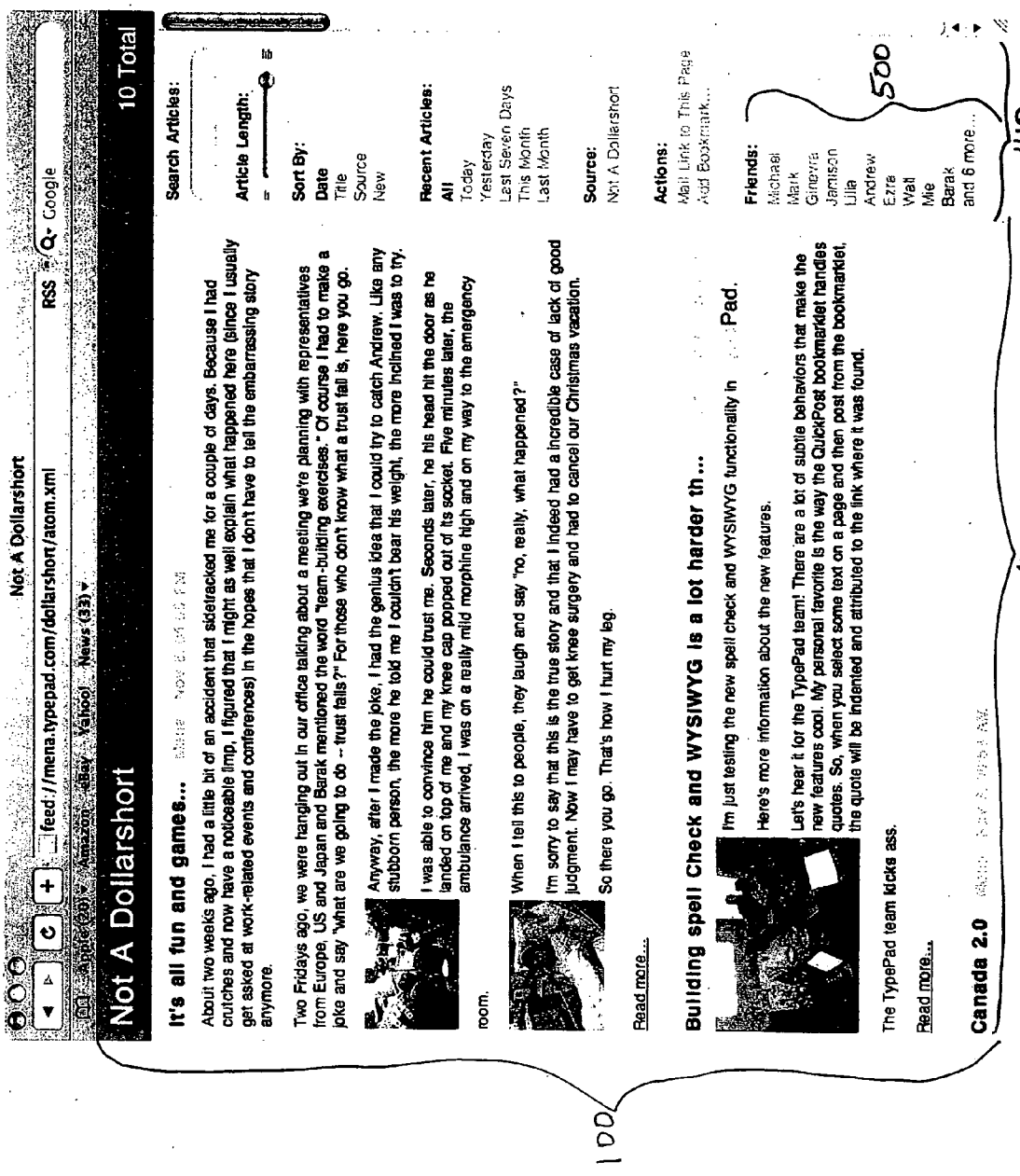

Not A Dollarshort
feed://mena.typepad.com/dollarshort/atom.xml
RSS
Google
Not A Dollarshort
10 Total
It's all fun and games...
About two weeks ago, I had a little bit of an accident that sidetracked me for a couple of days. Because I had crutches and now have a noticeable limp, I figured that I might as well explain what happened here (since I usually get asked at work-related events and conferences) in the hopes that I don't have to tell the embarrassing story anymore.
Two Fridays ago, we were hanging out in our office talking about a meeting we're planning with representatives from Europe, US and Japan and Barak mentioned the word "team-building exercises." Of course I had to make a joke and say "what are we going to do -- trust falls?" For those who don't know what a trust fall is, here you go.
Anyway, after I made the joke, I had the genius idea that I could try to catch Andrew. Like any stubborn person, the more he told me I couldn't bear his weight, the more inclined I was to try.
I was able to convince him he could trust me. Seconds later, he his head hit the door as he landed on top of me and my knee cap popped out of its socket. Five minutes later, the ambulance arrived, I was on a really mild morphine high and on my way to the emergency room.
When I tell this to people, they laugh and say "no, really, what happened?"
I'm sorry to say that this is the true story and that I indeed had a incredible case of lack of good judgment. Now I may have to get knee surgery and had to cancel our Christmas vacation.
So there you go. That's how I hurt my leg.
Read more...
Building spell Check and WYSIWYG is a lot harder th...
I'm just testing the new spell check and WYSIWYG functionality in Pad.
Here's more information about the new features.
Let's hear it for the TypePad team! There are a lot of subtle behaviors that make the new features cool. My personal favorite is the way the QuickPost bookmarklet handles quotes. So, when you select some text on a page and then post from the bookmarklet, the quote will be indented and attributed to the link where it was found.
The TypePad team kicks ass.
Read more...
Canada 2.0
Search Articles:
Article Length:
Sort By:
Date
Title
Source
New
Recent Articles:
All
Today
Yesterday
Last Seven Days
This Month
Last Month
Source:
Not A Dollarshort
Actions:
Mail Link to This Page
Add Bookmark...
Friends:
Michael
Mark
Ginevra
Jamison
Lilia
Andrew
Ezra
Walt
Mie
Barak
and 6 more...
100
400
500
410

FIG. 5

BOOKMARKS AND SUBSCRIPTIONS FOR FEEDS

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application is a continuation of the following patent application, which is hereby incorporated by reference: U.S. application Ser. No. 11/105,637, filed on Apr. 13, 2005, entitled "News Feed Browser." This patent application is related to the following patent applications, which are hereby incorporated by reference: U.S. patent application Ser. No. 09/467,073, filed on Dec. 20, 1999, entitled "Secondary Windows Generated as Document-Modal Overlays"; U.S. patent application Ser. No. 10/951,915, filed on Sep. 27, 2004, entitled "Live Content Resizing"; U.S. patent application Ser. No. 11/105,771, filed on Apr. 13, 2005, entitled "Multiple-Panel Scrolling"; U.S. patent application Ser. No. 11/105,810, filed on Apr. 13, 2005, entitled "News Feed Viewer"; U.S. patent application Ser. No. _____, filed on Mar. 7, 2006, entitled "MIME Type Detection for Feeds", attorney docket #18602-11356 (P3678C1); and U.S. patent application Ser. No. _____, filed on Mar. 7, 2006, entitled "Platform for Feeds", attorney docket #18602-11358 (P3678C3).

FIELD OF THE INVENTION

[0002] The present invention relates generally to software tools for accessing a news feed, and more particularly to software tools for detecting, displaying, and managing a news feed.

BACKGROUND OF THE INVENTION

[0003] Many users turn to the Internet as a source of news. Millions of web pages are available as a source of news content. These web pages take many forms, including, for example, articles, blurbs, and web logs ("blogs"), among others. In general, news content is published by inserting it into a web page (e.g., coding it into a HyperText Markup Language (HTML) file), which can then be viewed using a web browser.

[0004] However, there are drawbacks to this approach. Users who wish to see content from a variety of sources must typically visit many different web sites. In addition, if the user later wants to obtain the most current content from all the sources, she must visit all the different web sites again. This is very time consuming, especially if the user is interested in many different types of content and/or sources.

[0005] One possible solution is a software application that automatically visits a number of web sites that provide news content. The application analyzes the content found at each of the web sites, determines whether the content has changed, and notifies the user as to such changes. The user can then visit the web sites having newly-updated content. Alternatively, the application can automatically extract the updated content and present directly to the user. Unfortunately, in many cases it is very difficult (if not impossible) to identify the news content of a web page without a priori knowledge of the structure of the web page.

[0006] In response, "feeds" have been developed as a way to publish news content in a structured format that facilitates identification and analysis. A feed (also known as a "news feed" or "blog feed") can contain articles (content items), meta-data about the articles (e.g., title, date, and author), and/or meta-data about the feed itself (e.g., title, URL, and last-updated timestamp). Some feeds are described by Resource Description Framework (RDF) statements and/or encoded using extended Markup Language (XML) (such as, e.g., the XML syntax RDF/XML). These features of feeds make it easier for applications to automatically collect and evaluate news content for presentation to the user.

[0007] The term "syndication XML" refers to XML that has been developed for feeds, such as Atom and RSS ("Really Simple Syndication,""Rich Site Summary," or "RDF Site Summary"). Since the general structure of a feed is known, it is possible to identify the news content within it. Software applications have been developed that work with content published using syndication XML.

[0008] One feature of such applications enables a user to view the content of a feed in a way that is more user-friendly than a raw XML (text) file. (Syndication XML identifies content, without requiring that the content be displayed in any particular way.) Applications and/or application handlers, commonly known as "feed viewers,""feed readers," or "feed aggregators," are capable of reading the syndication XML and presenting it to the user. These applications, which can be implemented as (for example) stand-alone applications or web browser plug-ins, often contain useful user interface controls for facilitating navigation, searching, and the like with respect to news feeds. Another feature of many such feed viewers is "aggregation," which enables a user to specify multiple feeds and integrate their content into a single feed.

[0009] While feeds have made it easier to access news content, existing techniques for detecting, managing, and presenting news feeds have some disadvantages. One problem is that users may be unable to determine whether a server is publishing a feed. Then, once a feed has been found and navigated to, the web browser will often prompt the user to save the feed to disk or display it as raw text, rather than display it in a user-friendly way. Another problem is that even if a feed can be displayed in a user-friendly way, this display often cannot be modified based on user preferences. Yet another problem is that if a user views a feed using multiple feed viewers (for example, on different computers), the state of the user's interaction with the feed (for example, which articles have been read) is inconsistent across the viewers. One viewer is not aware of the fact that the user has already viewed a particular article in another viewer. What is needed is a software application and/or software architecture that solves these problems.

SUMMARY OF THE INVENTION

[0010] The present invention provides techniques for detecting, managing, and presenting syndication XML (news feeds). In one embodiment, a web browser automatically determines that a web site is publishing syndication XML and notifies the user. The user is then able to access the feed easily. The user can be notified, for example, by displaying a badge in the address bar of the browser window. The user can access the feed, for example, by using the badge to toggle between viewing the web page and viewing the feed.

[0011] In another embodiment, a web browser automatically determines that a web page or feed is advertising

relationship XML. The web browser then determines that the relationship XML identifies a feed or web page. The user is then able to access the identified feed or web page easily. In one embodiment, when relationship XML is found, information about the people identified in the relationship XML is displayed. For example, a person's name can be displayed and can link to her syndication XML or to her homepage.

[0012] In yet another embodiment, a web browser automatically determines whether a file contains syndication XML. If it does, the web browser enables the user to view the feed in a user-friendly way. For example, the web browser can open the feed file in an application handler to display the feed. As another example, the web browser can display the feed in its content window. In one embodiment, the web browser converts the feed to HTML and then displays the HTML.

[0013] In yet another embodiment, a user can bookmark feeds and organize the bookmarks using folders. In one embodiment, a user can aggregate multiple feeds by selecting a folder containing multiple feed bookmarks. In one embodiment, a bookmark displays the number of "read" or "unread" items in its associate feed. A user can also subscribe to feeds.

[0014] In yet another embodiment, a user can modify how a feed is displayed. For example, the user can specify which content is to be displayed. In one embodiment, the user can control the order in which articles are displayed. In another embodiment, the user can specify which articles are to be displayed. In yet another embodiment, the user can control the amount of each article that is displayed. As another example, a user can modify the format in which a feed is displayed. A format can specify which information is displayed (and in what order) and how the displayed information should be formatted. In one embodiment, a modification regarding how a feed is to be displayed is stored so that it can be used again at a later time.

[0015] In yet another embodiment, feed state information is stored in a repository that is accessible by applications that might be used to view the feed. In one embodiment, if the state of a feed changes, an application notifies the repository, and the repository updates the state accordingly. In another embodiment, a feed is parsed and stored in a structured way.

[0016] In yet another embodiment, a user can create a custom feed through aggregation and/or filtering of existing feeds. Aggregation includes, for example, merging the articles of multiple feeds to form a new feed. Filtering includes, for example, selecting a subset of articles of a feed based on whether they satisfy a search query. Aggregation and/or filtering can be performed server-side or client-side.

[0017] 5In yet another embodiment, a user can find articles from feeds that she has not bookmarked or subscribed to. The user can enter a search query into a search engine that searches feeds, which will identify one or more feed articles that satisfy the query. Once the articles that satisfy the query have been identified, the search engine returns the results. In one embodiment, the results are presented to the user in the form of a web page containing one or more links to feed articles. In another embodiment, the articles that satisfy the query are assembled into a feed, which is then presented to the user.

BRIEF DESCRIPTION OF THE DRAWINGS

[0018] The accompanying drawings illustrate several embodiments of the invention and, together with the description, serve to explain the principles of the invention.

[0019] **FIG. 1A** is a screen shot depicting an example of a browser window, where the browser window is displaying a web page and the URL address bar includes a badge.

[0020] **FIG. 1B** is a screen shot depicting an example of a browser window, where the browser window is displaying a feed and the URL address bar includes an RSS badge.

[0021] **FIG. 2** is a flowchart depicting a method for determining whether a file includes syndication XML.

[0022] **FIG. 3A** is a screen shot depicting an example of a browser window, where the browser window is displaying a bookmark management view.

[0023] **FIG. 3B** is a screen shot depicting an example of a browser window, where the browser window is displaying a bookmark management view and the contents of the News folder are displayed from the bookmarks bar.

[0024] **FIG. 3C** is a screen shot depicting an example of a browser window, where the browser window is displaying a bookmark management view and the Bookmarks menu is displayed from the menu bar.

[0025] **FIG. 4A** is a screen shot depicting an example of a browser window, where the browser window is displaying a feed and user interface elements for controlling how the feed is displayed.

[0026] **FIG. 4B** is a screen shot depicting an example of a browser window, where the browser window is displaying an aggregate feed and user interface elements for controlling how the feed is displayed.

[0027] **FIG. 4C** is a screen shot depicting the browser window of **FIG. 4B**, where the browser window has been scrolled down to display different articles.

[0028] **FIG. 4D** is a screen shot depicting the browser window of **FIG. 4C**, where the browser window has been scrolled down to display different articles.

[0029] **FIG. 4E** is a screen shot depicting the browser window of **FIG. 4D**, where the browser window has been scrolled down to display different articles.

[0030] **FIG. 5** is a screen shot depicting an example of a browser window, where the browser window is displaying a feed and relationship XML.

[0031] One skilled in the art will recognize that these Figures are merely examples of the operation of the invention according to one embodiment and that other user interface arrangements and modes of operation can be used without departing from the essential characteristics of the invention.

DETAILED DESCRIPTION OF THE EMBODIMENTS

[0032] The present invention is now described more fully with reference to the accompanying Figures, in which several embodiments of the invention are shown. The present invention can be embodied in many different forms and should not be construed as limited to the embodiments set

forth herein. Rather, these embodiments are provided so that this disclosure will be complete and will fully convey the invention to those skilled in the art.

[0033] For illustrative purposes, the invention is described in connection with a mechanism for interacting with news feeds. Various specific details are set forth herein and in the Figures, to aid in understanding the present invention. However, such specific details are intended to be illustrative, and are not intended to restrict in any way the scope of the present invention as claimed herein. In addition, the particular screen layouts, appearance, and terminology as depicted and described herein are intended to be illustrative and exemplary and in no way limit the scope of the invention as claimed.

[0034] In one embodiment, the present invention is implemented in a conventional personal computer system, such as an iMac, Power Mac, or PowerBook (available from Apple Computer, Inc. of Cupertino, Calif.), running an operating system such as Mac OS X (also available from Apple Computer, Inc.). It will be recognized that the invention can be implemented on other devices as well, such as handhelds, personal digital assistants (PDAs), mobile telephones, consumer electronics devices, and the like. The invention can be embodied in software that runs on the personal computer. The invention can be included as add-on software, or it can form part of the operating system itself, or it can be a feature of an application that is bundled with the computer system or sold separately. The various features of the invention as described herein include output presented on a display screen that is connected to the personal computer. In addition, the invention makes use of input provided to the computer system via input devices such as a keyboard, mouse, touchpad, or the like. Such hardware components, including their operation and interactions with one another and with a central processing unit of the personal computer, are well known in the art of computer systems and therefore are not depicted here. Several Figures are presented as screen shots depicting examples of the user interface as it might appear on a display screen or other output device.

[0035] In the following description, the terms "news feed, ""blog feed," and "feed" are used interchangeably to refer to articles (content items), meta-data about articles (e.g., title, date, and author), and meta-data about feeds (e.g., title, URL, and last-updated timestamp). A feed can be described by Resource Description Framework (RDF) statements and/or encoded using extended Markup Language (XML) (such as, e.g., the XML syntax RDF/XML). The term "syndication XML" refers to XML that has been developed for feeds, such as Atom and RSS ("Really Simple Syndication,""Rich Site Summary," or "RDF Site Summary"). The terms "feed viewer" and "feed reader" are used interchangeably to refer to a software component that enables a user to view a feed. A feed viewer can be, for example, a web browser plug-in or a standalone application. For the purposes of this application, the phrase "feed browser" refers to a feed viewer that can also be used to browse the web (e.g., to display files containing HTML). Although a feed viewer is usually located on a user's computer (e.g., as a web browser), it can also be located on a remote computer (e.g., as a remote application accessed using a terminal).

Detecting the Availability of a Feed

[0036] As described above, existing news feed viewers have several disadvantages. One problem is that users are unable to determine, via a conventional feed viewer, whether a server is publishing a feed. Some feeds are published in conjunction with web sites. Web sites that offer their content as feeds (e.g., in addition to HTML) can inform users of the existence of these feeds by, for example, displaying on a web page the feed's address or a link to it. However, not all web sites with feeds advertise them to users. And, even if they do, users may not see the advertisements if the advertisements are scarce and/or buried within large web pages.

[0037] In one embodiment of the present invention, a web browser automatically determines that a web site is publishing a feed and notifies the user. The user is then able to access the feed easily. The user can be notified, for example, by displaying a badge **120** in the Uniform Resource Locator (URL) address bar **110** of the browser window **100**. The user can access the feed, for example, by using the badge **120** to toggle between viewing the web page and viewing the feed. One skilled in the art will recognize that in other embodiments of the present invention, other techniques can be used to notify the user of the existence of a feed and to enable the user to easily access the feed.

[0038] **FIG. 1A** is a screen shot depicting an example of a browser window, where the browser window is displaying a web page and the URL address bar includes a badge. In the illustrated embodiment, the badge **120** includes the letters "RSS". If the user toggles the view (e.g., by clicking on the badge **120**), the feed browser window **100** displays the feed instead of the web page. **FIG. 1B** is a screen shot depicting an example of a browser window, where the browser window is displaying a feed and the URL address bar includes an RSS badge. Note that in the illustrated embodiment, the appearance of the badge **120** has changed. This change in appearance indicates that a feed is being displayed and that an associated web page exists. If the user again toggles the view (e.g., by clicking on the badge **120**), the browser window **100** displays the associated web page instead of the feed.

[0039] In one embodiment, when the feed browser toggles from displaying a web page to displaying a feed, a distinctive visual effect or animation is presented to reinforce the transition between the two modes. For example, in one embodiment, a window-shade effect is used, in which the feed appears to "roll down" over the web page, obscuring it. Similarly, when the browser toggles from displaying a feed to displaying a web page, the feed appears to "roll up" and expose the underlying web page. This behavior is described in the following patent application, which is hereby incorporated by reference: U.S. patent application Ser. No. 09/467,073, filed on Dec. 20, 1999, entitled "Secondary Windows Generated as Document-Modal Overlays."

[0040] In one embodiment of the present invention, the web browser automatically determines whether a website is publishing a feed. Many different techniques can be used to make this determination. For example, the web browser can determine whether a web site is publishing a feed by determining whether the HyperText Markup Language (HTML) code of the web page contains a <link> element that refers to a feed. This <link> element can indicate the URL of the feed and its content type. For example, <link rel="alternate" type="application/rss+xml" href="http://www.apple.com/main/rss/hotnews/hotnews.rss"> indicates that an alternate link for the web page is a feed located at

http://www.apple.com/main/rss/hotnews/hotnews.rss that has a content type of application/rss+xml.

[0041] Other online resources besides websites can also advertise feeds. Feeds can be advertised, for example, by "relationship XML." The term "relationship XML" refers to XML that has been developed to describe people, such as Friend-of-a-Friend (FOAF) and XFN (XHTML Friends Network). Relationship XML uses XML and RDF to publish information about people, the relationships between them, and the things that they create or do. This information can include, for example, name, email address, homepage, syndication XML feed, and acquaintances ("friends"). Using relationship XML to provide a link to a person's feed enables a user to traverse a web of interconnected people who are all publishing feeds.

[0042] Web sites that publish relationship XML can advertise this information to users by, for example, displaying on a web page the relationship XML's address or a link to it. However, not all web sites with relationship XML advertise it to users. And, even if they do, users may not see the advertisements if the advertisements are scarce and/or buried within large web pages.

[0043] In one embodiment, a web browser automatically determines that a web page is publishing relationship XML. The browser determines whether the relationship XML identifies a feed (or web page). If it does, the browser notifies the user. The user is then able to access the identified feed (or web page) easily. The user can be notified, for example, by displaying a link in the content window of the browser window **100**. The user can access the identified feed or web page by, for example, using the link. In one embodiment, such a methodology is enabled by modifying the web page that publishes the relationship XML to include the link.

[0044] Feeds can also advertise relationship XML. In one embodiment, a feed viewer determines that a feed is advertising relationship XML. The viewer determines whether the relationship XML identifies a feed (or web page). If it does, the browser notifies the user, and the user is able to access the identified feed (or web page) easily. The user can be notified, for example, by displaying a link in the feed viewer (in addition to displaying the original feed that advertised the relationship XML). The user can access the identified feed or web page by, for example, using the link. In one embodiment, the feed that advertises the relationship XML is modified to include the link.

[0045] In one embodiment, when relationship XML is found, a feed viewer or feed browser displays the names (e.g., identified in a <foaf:name> block) of people identified as acquaintances in the relationship XML (e.g., identified in a <foaf:knows> block). An acquaintance name can be linked to the acquaintance's syndication XML feed (e.g., identified in a <foaf:weblog> block) or homepage (e.g., identified in a <foaf:homepage> block; for example, if no such feed exists). In one embodiment, an acquaintance's syndication XML feed can be identified by reviewing the acquaintance's relationship XML (e.g., identified in a <foaf:seeAlso> block) or by reviewing the acquaintance's homepage, which might advertise the feed. These acquaintance name links can be used to traverse a web of interconnected people who are publishing feeds and/or web pages.

[0046] **FIG. 5** is a screen shot depicting an example of a browser window, where the browser window is displaying a feed and relationship XML. In the illustrated embodiment, feed browser window **100** includes a feed panel **400** and a user interface element panel **410**. The user interface element panel **410** includes a list **500** of names of people identified as acquaintances in the relationship XML. If a user selects a name from this list **500** (e.g., by clicking on it or issuing a voice command), the browser window **100** displays the feed or homepage (not shown) associated with that name.

[0047] In one embodiment, when relationship XML is found, a feed viewer or feed browser displays information about the owner of the XML. This information is available from the XML and can include, for example, a picture of the person, her name, her contact information, and links to her feed and/or web page. In one embodiment, rather than displaying links to the person's feed and/or web page, the feed viewer or feed browser displays the actual feed and/or web page. In addition, if the feed or web page (or an alternate web page) advertises relationship XML, this relationship XML can also be displayed, as described above.

[0048] Information displayed about the owner of the XML can also include the names of people identified as acquaintances in the XML. In one embodiment, selecting an acquaintance name accesses that acquaintance's relationship XML and displays the same types of information about the acquaintance (possibly including a related feed or web page, if one exists). In this way, a user can drill farther and farther down into a social network of friends (acquaintances) and friends-of-friends.

[0049] A web browser can determine whether a web page or feed is advertising relationship XML by, for example, determining whether the HyperText Markup Language (HTML) code of the web page (or the XML code of the feed) contains a <link> element that refers to relationship XML. This <link> element can indicate the URL of the relationship XML and its content type. For example, <link rel="meta" type="application/rdf+xml" title="FOAF" href="http://janedoe.typepad.com/foaf.rdf" /> indicates that a metadata link for the web page is a file located at http://janedoe.typepad.com/foaf.rdf that has a content type of application/rdf+xml.

[0050] Even if a feed does not advertise relationship XML, it might advertise an alternate (or parent) web page. Such an advertisement could be, for example: <link rel="alternate" type="text/html" href="http://janedoe.typepad.com/nameless/"/>. This alternate or parent web page might, in turn, advertise relationship XML, which can be used as described above.

Detecting that a File Contains a Feed

[0051] Once a user has found a feed and navigated to it, many existing web browsers will prompt her to save the feed to disk or will display it as raw text, rather than display it in a user-friendly way. Some existing browsers do this because they do not recognize that the requested file contains a feed.

[0052] In one embodiment, the web browser of the present invention avoids such undesirable behavior by automatically determining whether a file contains a feed and, if it does, using this information to enable the user to view the feed in a user-friendly way.

[0053] A web browser can determine whether a file contains a feed by, for example, analyzing the file's attributes. These attributes can include, for example, content type (also

known as "Multipurpose Internet Mail Extension (MIME) type"), filename extension, and/or URL scheme, and can indicate that a file contains a feed. For example, a content type of text/rss+xml, application/rss+xml, text/atom+xml, or application/atom+xml, a filename extension of .rss or .atom, and/or a URL scheme of "feed" (so that a URL would begin with "feed://") usually indicate that the file contains a feed.

[0054] Sometimes the content type, filename extension, and/or URL scheme information is missing or inconsistent. For example, a file's content type might be missing, while its filename extension is .rss and its URL scheme is http. Files including feeds frequently have non-syndication XML content types (e.g., text/rdf+xml, application/rdf+xml, text/xml, application/xml, and text/plain), filename extensions (e.g., .rdf, .xml, and .txt), and/or URL schemes (e.g., http). In these situations, it is unclear whether the file includes syndication XML.

[0055] In one embodiment, the web browser of the present invention determines, based on the available information, whether the file includes syndication XML. Alternatively, it gathers more information before making this determination. For example, the web browser can analyze the file's contents (either the entire file or just a portion thereof for syndication XML tags. These tags can include, e.g., <feed>, <rss>, and <atom>.

[0056] **FIG. 2** is a flowchart depicting a method for determining whether a file includes syndication XML. Method **200** begins by determining **210** the file's content type. In one embodiment, this is accomplished using the URL Loading System set of classes and protocols in the Cocoa application environment (e.g., an NSURLResponse object), which is available from Apple Computer, Inc. If the content type indicates that the file is likely to include syndication XML **220** (e.g., if the content type is text/rss+xml, application/rss+xml, text/atom+xml, or application/atom+xml), then the file does include syndication XML **230**. If the content type indicates that the file might include syndication XML **240** (e.g., if the content type is text/rdf+xml, application/rdf+xml, text/xml, application/xml, or text/plain), then the file might include syndication XML, and more information is gathered (step **270**). If the content type indicates that the file is likely not to include syndication XML **250** (e.g., if the content type is something else), then the file does not include syndication XML **260**.

[0057] If the file might include syndication XML **240**, its content is analyzed **270**. In one embodiment, the first "<" character in the file is identified. If "<" is followed by "feed", "rss", or "atom"**280**, then the file includes syndication XML **230**. If "<" is followed by "rdf"**285**, then "<channel>" is searched **290** for. If "<channel>" exists, then the file includes syndication XML **230**. If "<channel>" does not exist, then the file does not include syndication XML **260**.

[0058] If the first "<" character is followed by "?" or "!"**295**, then the next "<" character is identified **297** and the characters following it are considered. If "<" is followed by anything else, then the file does not include syndication XML **260**.

Displaying a Feed

[0059] Once a web browser has determined that a file contains a feed (e.g., syndication XML), in one embodiment, it enables the user to view the feed in a user-friendly way. In one embodiment, this is accomplished by opening the feed file in an application handler (such as a feed viewer) to display the feed. The user then views the feed using the feed viewer. If multiple feed viewers are available, the browser determines which one to use.

[0060] In one embodiment, the web browser determines a feed viewer to use by analyzing the file's attributes. This is accomplished, for example, using the Launch Services API in the Carbon set of procedural APIs, which is available from Apple Computer, Inc. For example, the function LSCopyApplicationForMIMEType can be used to determine the feed viewer based on the file's content (MIME) type. Similarly, the function LSGetApplicationForInfo can be used to determine the feed viewer based on the file's filename extension and/or URL scheme.

[0061] In one embodiment, when a web browser has determined that a file includes syndication XML, it changes the file's content type to "application/rss+xml" or "application/atom+xml" (if necessary). In another embodiment, the browser "reroutes" the loading of the file by opening a URL that is similar to the original URL but that has a URL scheme of "feed" instead of "http." Changing a file's content type and/or URL scheme improves the results obtained by using the Launch Services API, since the API determines the appropriate feed viewer based on these characteristics.

[0062] Once the feed viewer has been determined, it is launched and passed the file's URL. In one embodiment, this is accomplished by sending the AppleEvent "GURL" (get URL) to the feed viewer.

[0063] In another embodiment, a browser enables a user to view a feed in a user-friendly way by displaying the feed in its content window. In this embodiment, a user need not be aware that she has requested a file that contains a feed. The browser will simply display the feed in a seamless fashion. For purposes of the description provided herein, a browser that can display a feed in its content window is referred to as a "feed browser," which is a type of feed viewer (since it enables a user to view a feed).

[0064] A browser can display a feed in its content window by, for example, generating a visual representation of the feed. For example, the browser can include code that takes a feed as input and generates a visual representation of it. The code can determine graphical user interface elements to display the feed (similar to code found in feed viewers). Alternatively, the code can convert a feed to HTML and then display the HTML using the browser. For example, the code can assemble an HTML file that includes articles, article meta-data, and feed meta-data. In one embodiment, the feed, the articles in the feed, or the resulting HTML file is checked for validity and well-formedness and is corrected if necessary (e.g., using HTML Tidy, which is available at http://tidy.sourceforge.net/).

Obtaining New Feed Content

[0065] When a user finds an interesting feed, she may want to obtain new feed content as it is published. Conventionally, if the user does not know when publication occurs, she may have to repeatedly return and view the feed periodically to determine whether the content has changed.

[0066] In one embodiment, the present invention provides functionality that helps streamline and/or minimize such

repeated manual viewings of the feed. A user can bookmark a feed. Once a feed has been bookmarked, a user can select the bookmark in order to navigate to the feed. Although the following figures show bookmarks in feed browser windows, they can also be implemented in feed viewers. **FIG. 3A** is a screen shot depicting an example of a browser window, where the browser window is displaying a bookmark management view. In one embodiment, feed browser window **300** displays the bookmark management view **310** in response to a command entered by a user, for example by clicking on the Bookmarks Bar icon **320**, choosing a selection from a menu (not shown), or pressing a key (not shown).

[**0067**] In the illustrated embodiment, the bookmark management view **310** includes several bookmarks, including bookmarks for web pages and bookmarks for feeds. In the illustrated embodiment, a web page bookmark is indicated by a globe Favorites icon ("favicon" or "page icon") **330**, while a feed bookmark is indicated by a newspaper favicon **340** and an RSS badge **350**. In one embodiment, a folder can be used to group and/or organize bookmarks, regardless of whether the bookmarks are for feeds or for web pages. In the illustrated embodiment, the bookmark management view **310** includes two folders **360**, entitled "Apple" and "News." The News folder contains eight bookmarks, of which four are feeds (The New York Times, Washington Post, CNET News.com, and Wired) and four are web pages (CNN.com, Los Angeles Times, BBC NEWS, and CBS MarketWatch). In one embodiment, a folder can contain other folders, in addition to containing bookmarks. A first folder contained within a second folder is referred to as being "nested" within the second folder.

[**0068**] In one embodiment, a user can open all of the bookmarked items in a folder at once by selecting the folder. In one embodiment, if multiple bookmarked feeds exist, each feed opens in a different feed window or under a different tab (e.g., in a tabbed browser). In another embodiment, the bookmarked feeds are aggregated so that their articles form one feed and that one feed is then displayed in one feed window or under one tab (e.g., in a tabbed browser). In one embodiment, a feed contained in a first folder and a feed contained in a second folder (which is nested within the first folder) can also be aggregated, even though they are located in different levels of the folder hierarchy. Aggregation will be further discussed below.

[**0069**] In one embodiment, a bookmark can display the number of "read" and/or "unread" items in its associated feed. In the illustrated embodiment, a feed's "unread count"**370** is located adjacent to the name of the feed. For example, the Wired bookmark has an unread count **370** of 7. This means that there are seven unread articles in the Wired feed. The News folder **360** also has an unread count **370**. Its unread count (**26**) is the sum of the unread counts **370** of the feeds whose bookmarks it contains.

[**0070**] **FIG. 3B** is a screen shot depicting an example of a browser window, where the browser window is displaying a bookmark management view and the contents of the News folder are displayed from the bookmarks bar. In one embodiment, the contents of the News folder are displayed from the bookmarks bar **380** in response to a command entered by a user, for example by clicking on "News"**390** in the bookmarks bar **380**.

[**0071**] The contents of the News folder are displayed in both the bookmark management view **310** and the bookmarks bar **380**. Note that the feed bookmarks in the News folder in the bookmarks bar **380** include favicons **340** and unread counts **370**.

[**0072**] **FIG. 3C** is a screen shot depicting an example of a browser window, where the browser window is displaying a bookmark management view and the Bookmarks menu is displayed from the menu bar. In one embodiment, the contents of the Bookmarks menu are displayed from the menu bar **395** in response to a command entered by a user, for example by clicking on "Bookmarks"**397** in the menu bar **395**. Note that the feed bookmarks in the Bookmarks menu include favicons **340** and unread counts **370**.

[**0073**] In one embodiment, the data associated with a feed bookmark includes a URL. In another embodiment, the data includes modifications regarding how the feed should be displayed. Modifying how a feed is displayed will be further discussed below.

[**0074**] If the user does not know when feed publication occurs, another option is to "subscribe" to the feed. If a feed has been subscribed to, the user can be notified when new articles are added to a feed. For example, the feed URL can be polled periodically for updates. In one embodiment, if the feed has been updated, the new articles are downloaded to the user's feed viewer or an associated server. Subscription information can include, for example, an update frequency or a timestamp signifying the time that a feed URL was last checked for updates.

Modifying How a Feed is Displayed

[**0075**] In one embodiment, the present invention enables a user to modify how a feed is displayed. For example, the user can specify which content is to be displayed and/or the format in which it is to be displayed. Such functionality is particularly useful for feeds that include large numbers of articles, since such as large amount of information can overwhelm users. By providing a user with the ability to control what feed content is displayed and how it is displayed, the present invention makes such large feeds manageable and useful.

[**0076**] For example, the user can control the order in which articles are displayed by selecting a particular article characteristic. This characteristic could be, for example, article date, article title, or article source (e.g., the feed that published the article). Combining articles from different feeds will be discussed below.

[**0077**] In another embodiment, the user can specify which articles are to be displayed. For example, the user can restrict display based on article date (e.g., today, last seven days, or last month) and/or article source. A user can also specify that only articles that satisfy a search query be displayed. In one embodiment, articles are filtered as the user enters the search query (e.g., after each keystroke, or after a slight pause of predetermined length during user entry of the query). A user can also specify that only "unread" or "flagged" articles should be displayed. In one embodiment, if the number of articles to be displayed exceeds a particular threshold (e.g., 50), then the articles are divided into multiple pages, where a page includes a subset of the total articles to be displayed. Page size is selected to be some manageable and easily displayed quantity of articles, and need not correspond to

screen or window size. If a page is large than a window, then a scroll bar can be used for navigating to other portions of the page. A user can navigate from one page to another by, for example, using links on the page that is currently being displayed.

[0078] In yet another embodiment, the user can control the amount of each article that is displayed. In one embodiment, this is achieved by manipulating a slider user interface element **420** (see **FIG. 4A**). For example, dragging the slider to the left shows less of each displayed article, while dragging the slider to the right shows more of each displayed article. In one embodiment, the far-right slider setting causes articles to be displayed in their entirety, while other slider settings cause only a portion of each displayed article to be displayed. In one embodiment, the far-left slider setting causes only the titles of articles to be shown. Alternatively, the far-left slider setting causes only the titles and one line (or sentence) of articles to be shown.

[0079] In one embodiment, a slider **420** setting affects the amount of each displayed article shown in an absolute way (e.g., so that the first 10 lines (or sentences) of an article are shown, regardless of the size of the article). Alternatively, a slider setting affects the amount of each displayed article shown in a relative way (e.g., so that half of the total lines (or sentences) of an article are shown).

[0080] In one embodiment, when an article is not shown in its entirety, the lines (or sentences) that are displayed are determined based on their location within the article (e.g., the first 5 sentences of the article are displayed or the last 5 sentences of the article are displayed). Alternatively, the lines (or sentences) that are displayed are determined based on their importance (e.g., the 5 most important sentences are displayed).

[0081] This is accomplished, for example, using the Search Kit API in the Carbon set of procedural APIS, which is available from Apple Computer, Inc. The function SKSummaryCreateWithString can be used to analyze a body of text (such as an article) and return an SKSummaryRef object. This object can then be used to determine a particular sentence of the original text based on that sentence's position within the text. For example, in order to obtain the third sentence, use "3" as the input to the function SKSummaryCopySentenceAtIndex. In order to obtain the middle sentence, use the function SKSummaryGetSentenceCount to determine the total number of sentences in the original text, then divide that number in half and use the result as the input to the function SKSummaryCopySentenceAtIndex.

[0082] An SKSummaryRef object can also be used to determine a particular sentence of the original text based on that sentence's importance relative to other sentences. The function SKSummaryGetSentenceSummaryInfo returns three arrays of information: 1) the ordinal numbers (e.g., "3" for the third sentence) of the n most important sentences, in order of occurrence in the original text; 2) the rank order number (in terms of relative importance) of each of these sentences; and 3) the ordinal paragraph number of each of these sentences, where n can range from 1 to the total number of sentences in the original text.

[0083] In one embodiment, the slider user interface element **420** affects all displayed articles in a feed simultaneously. In another embodiment, the slider user interface element **420** affects only one displayed article (for example, a currently selected article), or a subset of displayed articles, in a feed.

[0084] If the slider user interface element **420** affects several articles simultaneously, it can be time-consuming to determine, for each article, which sentences should be shown. In one embodiment, in order to decrease the time necessary to make this determination, each article is pre-processed before the slider setting is changed (e.g., when the article is initially displayed). This pre-processing can include creating an SKSummaryRef object, determining the total number of sentences in the article, and calling the SKSummaryGetSentenceSummaryInfo with n equal to the total number of sentences in the article.

[0085] Each sentence can then be tagged based on its rank order, and a Cascading Style Sheet (CSS) can control whether the sentence is displayed (or not) based on the tagged value. Then, when the slider is set to m (e.g., "5" would display 5 sentences per article), any sentence whose rank order was greater than m would not be displayed. As a result, only sentences with rank orders less than or equal to m (i.e., the most important m sentences) would be displayed. If each sentence in each article is tagged this way, then all of the articles will display the m most important sentences at the same time as the slider setting is changed.

[0086] The slider user interface element **420** is further described in the following patent applications, which are hereby incorporated by reference: U.S. Provisional Patent Application Ser. No. 60/582,868, filed on Jun. 25, 2004, entitled "Live Content Resizing" and U.S. patent application Ser. No. 10/951,915, filed on Sep. 27, 2004, entitled "Live Content Resizing."

[0087] In one embodiment, a feed viewer incorporates one or more user interface elements into its display, where the term "user interface element" includes any type of on-screen control or other input element. For example, a feed browser could convert a feed to HTML and generate an HTML page that contained the converted feed in one panel and the user interface elements in another panel. **FIG. 4A** is a screen shot depicting an example of a browser window, where the browser window is displaying a feed and user interface elements for controlling how the feed is displayed. In the illustrated embodiment, browser window **100** includes a feed panel **400** and a user interface element panel **410**, which includes a slider user interface element **420** as described above.

[0088] User interface element panel **410** can also include a user interface element to trigger a particular action, such as emailing a link to the content displayed in the feed panel **400**, bookmarking the URL of the content displayed in the feed panel **400**, or bookmarking a search query (in order to filter the feed displayed in the feed panel **400**). In one embodiment, a user interface element is displayed only if it makes sense based on the user's context. For example, if the URL of the content displayed in the feed panel **400** has already been bookmarked, a user interface element to bookmark the URL will not be shown. In other words, the presence of a user interface element can be context-dependent.

[0089] Scroll bar **430** can be used to scroll the contents displayed in feed browser window **100** so that different

articles of a feed are displayed. In one embodiment, as the position of scroll bar **430** changes, the contents displayed in feed panel **400** and user interface element panel **410** scroll in unison. In this embodiment, if the feed is long, it is possible that scrolling to the end of it will cause user interface element panel **410** to cease displaying any user interface elements. In an alternate embodiment, depending on the position of scroll bar **430** and the contents in panels **400** and **410**, scrolling may or may not be performed in unison. For example, the contents displayed in user interface element panel **410** remain the same while the contents displayed in feed panel **400** scroll. This functionality is beneficial because the user can access user interface elements displayed in user interface element panel **410** even when she has scrolled to the end of feed panel **400**.

[**0090**] **FIG. 4B** is a screen shot depicting an example of a browser window, where the browser window is displaying an aggregate feed and user interface elements for controlling how the feed is displayed. Note that because there are so many articles in the feed, they are not displayed all at once in the window. Similarly, because there are so many user interface elements (here, the names of the source feeds that comprise the aggregate feed), they also are not displayed all at once in the window. The window in **FIG. 4B** is scrolled all the way to the top.

[**0091**] **FIGS. 4C-4E** are screen shots depicting the browser window of **FIG. 4B**, where the browser window has been scrolled down (to varying points) to display different articles. **FIG. 4C** is a screen shot depicting the browser window of **FIG. 4B**, where the browser window has been scrolled down to display different articles. In **FIG. 4C**, both the articles (feed panel **400**) and the user interface elements (user interface element panel **410**) have been scrolled down. Also, there are so many articles and user interface elements that some still have not been displayed. **FIG. 4D** is a screen shot depicting the browser window of **FIG. 4C**, where the browser window has been scrolled down to display different articles. In **FIG. 4D**, both the articles and the user interface elements have been scrolled down, and some articles and user interface elements still have not been displayed.

[**0092**] **FIG. 4E** is a screen shot depicting the browser window of **FIG. 4D**, where the browser window has been scrolled down to display different articles. Note that the articles (feed panel **400**) have been scrolled down farther than the user interface elements (user interface element panel **410**). This is because the user interface element panel **410** has reached the end of its content. This functionality is further described in the following patent application, which is hereby incorporated by reference: U.S. patent application Ser. No. _____ filed on the same date as the present application, entitled "Multiple-Panel Scrolling."

[**0093**] In yet another embodiment, a user can modify the format in which a feed is displayed. In one embodiment, a "theme" determines the format in which a feed is displayed. In one embodiment, a theme is specified using a set of template files and a set of format files. The template files specify which information is displayed (e.g., articles, article meta-data, and feed meta-data) and in what order. The format files specify the formatting of the information, such as fonts and colors. For example, a format file can specify that an "unread" content item is to be displayed differently than a "read" content item. Thus, by modifying a template file and/or a format file, the format in which a feed is displayed can be changed.

[**0094**] In one embodiment, a first template file describes three areas of a feed viewer content window: the top area (including, e.g., the title of the feed), the bottom-right area (including, e.g., user interface element panel **410**), and the bottom-left area (including, e.g., feed panel **400**). A second template file describes the bottom area of the feed viewer content window (including, e.g., pagination for when the number of articles to be displayed exceeds the threshold). A third template file describes how an article should be displayed (e.g., when it is displayed within feed panel **400**). One skilled in the art will recognize that other arrangements, using different numbers or types of template files, can be used.

[**0095**] In one embodiment, the template files are used to generate HTML files. A template file can include, for example, HTML code and specialized tags. The specialized tags can be replaced with information when a feed is actually being processed for display. For example, the specialized tag <<title>> can be replaced with the title of the feed that is being processed. As another example, the specialized tag <<Sort By>> can be replaced with the phrase that means "Sort By," depending on the language preference of the user (e.g., English versus Japanese). (Sort By is used in user interface element panel **410**.) In one embodiment, a format file is a style sheet (such as a Cascading Style Sheet (CSS)) that is used in conjunction with the HTML files generated using the template files.

[**0096**] In one embodiment, a modification regarding how a feed is to be displayed (such as which content is displayed and/or the format in which it is displayed) is stored so that it can be used again at a later time. Then, when the feed is viewed again, it will be displayed according to the stored modification. The stored modification can be associated with, for example, a particular feed or a particular user. If the modification is associated with a particular feed, then different modifications can be stored for different feeds. If the modification is associated with a particular user, then feeds viewed by that user will be displayed according to the stored modification.

Feed State

[**0097**] As described above, a user can specify several different types of information regarding a feed, which together make up the feed's state. This information can include, for example, whether the feed has been bookmarked, whether the feed has been subscribed to, and how the feed should be displayed in terms of content and formatting. Feed state can also include information about a specific content item, such as whether the item has been read (e.g., displayed) and/or whether it has been flagged. In one embodiment, a user can "flag" a content item of a feed. The flagged item can then be displayed differently than or separately from a non-flagged item. For example, a flagged item can be displayed in a different color or typeface or the like. As another example, only flagged items are displayed, while non-flagged items are not displayed.

[**0098**] A user may want to use different feed viewers, and even different computers or machines, to view the same feed. For example, she may want to view the feed from her

laptop using one feed viewer and from her PDA using another feed viewer. While feed state information can be stored on a user's device or on a server, in one embodiment, such information is stored in a location (called a "feed state repository") that is accessible by feed viewers that might be used to view the feed. That way, if the state of the feed changes using one viewer, the change will be available to other viewers. For example, if a user views an article using one viewer, the feed state will describe the article as "read" the next time the user views the feed, even if she is using a different viewer.

[0099] In one embodiment, if the state of a feed changes, the feed viewer notifies the feed state repository, and the feed state repository updates the state accordingly. In another embodiment, when feed state information changes, a timestamp is stored regarding when the change occurred. In yet another embodiment, distributed notifications are used to notify interested processes (such as feed browsers or feed viewers) of the change.

[0100] While a feed can be stored as a flat file, in one embodiment, the feed is parsed and then stored in a structured way. This makes it easier to access information contained in the feed, including articles, article meta-data, and feed meta-data. In one embodiment, feeds are stored in a database, called a "feed database." This database can be, for example, an SQLite database (available from Hwaci—Applied Software Research of Charlotte, N.C.). Similarly, feed state information can be stored, for example, as a flat file or in a database.

[0101] Feed state information is usually specific to a particular user. One way to associate information in a database with a particular user is to store each user's information in a separate database. Another way is to have multiple users share a database and have a user's feed browser or feed viewer identify the user when it accesses the database.

[0102] While feed state information is usually specific to a particular user, the feed itself (e.g., the original syndication XML file) can be shared among multiple users. If the feed is stored in a database accessible by multiple users, then only one copy of each feed needs to be stored, regardless of how many users want to access it. In addition, if multiple users have subscribed to the feed, the feed source can be polled once and the new feed stored in the shared database, thereby decreasing traffic at the feed source's web server.

Building a Custom Feed

[0103] In one embodiment, the present invention enables a user to create a custom feed through aggregation and/or filtering of existing feeds. This custom feed can act as a "personal clipping service," gathering together articles of interest to the user. A custom feed has similar properties to a "simple" feed. For example, it can be bookmarked or subscribed to and its articles can have state, such as "read" or "flagged."

[0104] Aggregation includes, for example, merging the articles of multiple feeds to form a new feed. Aggregation enables a user to read articles from different source feeds without having to switching back and forth between the feeds or to visit many different web pages or other resources. A user can specify which feeds to aggregate. For example, a user can store bookmarks to multiple feeds in a single folder, thereby aggregating the feeds. As another example, a user can specify a feed to aggregate by typing in its URL or by choosing it from a list. In one embodiment, this list includes feeds to which the user has already subscribed. In another embodiment, this list includes feeds that the user has bookmarked (whether or not they have been subscribed to). In yet another embodiment, this list includes feeds that have been identified by others (arranged, for example, in a hierarchical listing based on topic).

[0105] Filtering includes, for example, selecting a subset of articles of a feed based on whether they satisfy a search query. A search query can include, for example, one or more search terms or phrases connected by Boolean operators. In one embodiment, articles are searched using string-based methods, such as those available in JavaScript (e.g., for searching articles displayed in a feed browser) or SQLite (e.g., for searching articles stored in an SQLite database). In one embodiment, any type of feed can be filtered, including both simple feeds and aggregate feeds.

[0106] In one embodiment, a search query can be made "persistent" by storing it in conjunction with a feed identifier. In this embodiment, when that feed is displayed, its articles will be filtered so that only articles that satisfy that search query will be displayed. One way to specify a persistent search is by appending the search query to the feed URL. For example, if the feed URL were "feed://www.apple.com/news.rss" and the search query were "safari," a persistent query could be specified by the alphanumeric string "feed://www.apple.com/news.rss?filter=safari." In this embodiment, the filtered feed can be shared with others by adding the string to a document, such as a web page or email. Entering the string into a feed browser or feed viewer (e.g., by typing it or by double-clicking on it) would then display the filtered feed.

[0107] Aggregation and/or filtering can be performed server-side (e.g., by a feed database or feed server) or client-side (e.g., by a feed browser or feed viewer). Server-side aggregation includes, for example, a feed database storing an aggregate feed and a feed browser accessing this feed. Client-side aggregation includes, for example, a feed database storing two simple feeds, and a feed browser accessing these simple feeds and aggregating them into one feed. Server-side filtering includes, for example, a feed database storing an unfiltered feed, filtering the feed based on a query, and making the filtered feed available to a feed browser. Client-side filtering includes, for example, a feed database storing an unfiltered feed, and a feed browser accessing this feed and filtering it.

Finding Articles

[0108] In one embodiment, a user can find articles from feeds that she has not bookmarked or subscribed to. The user can enter a search query into a search engine that searches feeds, which will identify one or more feed articles that satisfy the query. In one embodiment, the search query is augmented using additional information available to the feed viewer, such as localization information from the IP address or the time zone of the operating system and user interface language information (e.g., English versus Japanese) from the operating system.

[0109] In one embodiment, the search engine searches a database that stores feed articles. The articles in the database

can be found by, for example, crawling a network (such as the World Wide Web) searching for feeds. The database can be located, for example, at a client (e.g., the user's computer) or at a server.

[0110] Once the articles that satisfy the query have been identified, the search engine returns the results. The results can be, for example, in the form of a web page containing one or more links. When the web page is presented and the user clicks on a link, a feed article will be displayed that satisfies the query. As another example, the articles that satisfy the query can be assembled into a feed. This feed can then be presented to the user. This is another type of "custom feed." In this embodiment, since the query result is simply another feed, the user can bookmark it, subscribe to it, assign state to its articles, etc. Regardless of the data format of the search results, they can be presented as-is or reformatted as desired (e.g., into a feed or web page).

[0111] In another embodiment, once the articles that satisfy the query have been identified, the feed identifiers (such as the feed URLs) of their source feeds are returned. The source feeds can then be aggregated into a new (custom) feed. The search query can be performed on the new feed to identify the desired articles.

[0112] In the above description, for purposes of explanation, numerous specific details are set forth in order to provide a thorough understanding of the invention. It will be apparent, however, to one skilled in the art that the invention can be practiced without these specific details. In other instances, structures and devices are shown in block diagram form in order to avoid obscuring the invention.

[0113] Reference in the specification to "one embodiment" or "an embodiment" means that a particular feature, structure, or characteristic described in connection with the embodiment is included in at least one embodiment of the invention. The appearances of the phrase "in one embodiment" in various places in the specification are not necessarily all referring to the same embodiment.

[0114] Some portions of the detailed description are presented in terms of algorithms and symbolic representations of operations on data bits within a computer memory. These algorithmic descriptions and representations are the means used by those skilled in the data processing arts to most effectively convey the substance of their work to others skilled in the art. An algorithm is here, and generally, conceived to be a self-consistent sequence of steps leading to a desired result. The steps are those requiring physical manipulations of physical quantities. Usually, though not necessarily, these quantities take the form of electrical or magnetic signals capable of being stored, transferred, combined, compared, and otherwise manipulated. It has proven convenient at times, principally for reasons of common usage, to refer to these signals as bits, values, elements, symbols, characters, terms, numbers, or the like.

[0115] It should be borne in mind, however, that all of these and similar terms are to be associated with the appropriate physical quantities and are merely convenient labels applied to these quantities. Unless specifically stated otherwise as apparent from the discussion, it is appreciated that throughout the description, discussions utilizing terms such as "processing" or "computing" or "calculating" or "determining" or "displaying" or the like, refer to the action and processes of a computer system, or similar electronic computing device, that manipulates and transforms data represented as physical (electronic) quantities within the computer system's registers and memories into other data similarly represented as physical quantities within the computer system memories or registers or other such information storage, transmission or display devices.

[0116] The present invention also relates to an apparatus for performing the operations herein. This apparatus can be specially constructed for the required purposes, or it can comprise a general-purpose computer selectively activated or reconfigured by a computer program stored in the computer. Such a computer program can be stored in a computer readable storage medium, such as, but is not limited to, any type of disk including floppy disks, optical disks, CD-ROMs, and magnetic-optical disks, read-only memories (ROMs), random access memories (RAMs), EPROMs, EEPROMs, magnetic or optical cards, or any type of media suitable for storing electronic instructions, and each coupled to a computer system bus.

[0117] The algorithms and modules presented herein are not inherently related to any particular computer or other apparatus. Various general-purpose systems can be used with programs in accordance with the teachings herein, or it may prove convenient to construct more specialized apparatuses to perform the method steps. The required structure for a variety of these systems will appear from the description below. In addition, the present invention is not described with reference to any particular programming language. It will be appreciated that a variety of programming languages can be used to implement the teachings of the invention as described herein. Furthermore, as will be apparent to one of ordinary skill in the relevant art, the modules, features, attributes, methodologies, and other aspects of the invention can be implemented as software, hardware, firmware or any combination of the three. Of course, wherever a component of the present invention is implemented as software, the component can be implemented as a standalone program, as part of a larger program, as a plurality of separate programs, as a statically or dynamically linked library, as a kernel loadable module, as a device driver, and/or in every and any other way known now or in the future to those of skill in the art of computer programming. Additionally, the present invention is in no way limited to implementation in any specific operating system or environment.

[0118] It will be understood by those skilled in the relevant art that the above-described implementations are merely exemplary, and many changes can be made without departing from the true spirit and scope of the present invention. Therefore, it is intended by the appended claims to cover all such changes and modifications that come within the true spirit and scope of this invention.

What is claimed is:

1. A method for creating a bookmark to a feed, comprising:

receiving, by a web browser, a first user input indicating the feed;

receiving, by the web browser, a second user input indicating an identifier; and

storing, by the web browser, an association between the feed and the identifier.

2. The method of claim 1, wherein receiving, by the web browser, the first user input indicating the feed comprises receiving, by the web browser, while the feed is displayed, a user input.

3. The method of claim 2, wherein receiving, by the web browser, while the feed is displayed, the user input comprises receiving, by the web browser, while the feed is displayed, a signal indicating activation of a user interface control.

4. The method of claim 1, wherein the received identifier comprises one or more alphanumeric characters.

5. The method of claim 1, wherein storing, by the web browser, the association between the feed and the identifier comprises storing, by the web browser, the association between the feed and the identifier in a data repository configured to automatically store content of a feed and state information regarding the feed, wherein the data repository is communicatively coupled to both a first client-side software application configured to process the feed and a second client-side software application configured to process the feed.

6. The method of claim 1, further comprising receiving, by the web browser, a third user input indicating a folder, and wherein storing, by the web browser, the association between the feed and the identifier comprises storing, by the web browser, with respect to the received folder, the association between the feed and the identifier.

7. The method of claim 6, wherein storing, by the web browser, with respect to the received folder, the association between the feed and the identifier comprises storing, by the web browser, with respect to the received folder, the association between the feed and the identifier in a data repository configured to automatically store content of a feed and state information regarding the feed, wherein the data repository is communicatively coupled to both a first client-side software application configured to process the feed and a second client-side software application configured to process the feed.

8. A method for displaying Internet content, comprising:

receiving, by a web browser, a first user input indicating a name of a feed;

identifying, by the web browser, a feed associated with the received name of the feed; and

displaying, by the web browser, one or more feed items of the identified feed.

9. The method of claim 8, further comprising:

receiving, by the web browser, a second user input indicating a name of a web page;

identifying, by the web browser, a web page associated with the received name of the web page; and

displaying, by the web browser, the identified web page.

10. A computer program product for creating a bookmark to a feed, the computer program product comprising a computer-readable medium, the computer-readable medium comprising:

program code for receiving, by a web browser, a first user input indicating the feed;

program code for receiving, by the web browser, a second user input indicating an identifier; and

program code for storing, by the web browser, an association between the feed and the identifier.

11. The computer program product of claim 10, wherein the computer-readable medium further comprises program code for receiving, by the web browser, a third user input indicating a folder, and wherein the program code for storing, by the web browser, the association between the feed and the identifier comprises program code for storing, by the web browser, with respect to the received folder, the association between the feed and the identifier.

12. A computer program product for displaying Internet content, the computer program product comprising a computer-readable medium, the computer-readable medium comprising:

program code for receiving, by a web browser, a first user input indicating a name of a feed;

program code for identifying, by the web browser, a feed associated with the received name of the feed; and

program code for displaying, by the web browser, one or more feed items of the identified feed.

13. A web browser for creating a bookmark to a feed, comprising:

a first receiving module configured to receive a first user input indicating the feed;

a second receiving module configured to receive a second user input indicating an identifier; and

a storage module configured to store an association between the feed and the identifier.

14. The web browser of claim 13, further comprising a third receiving module configured to receive a third user input indicating a folder, and wherein the storage module is further configured to store, with respect to the received folder, the association between the feed and the identifier.

15. A web browser for displaying Internet content, comprising:

a receiving module configured to receive a first user input indicating a name of a feed;

an identification module configured to identify a feed associated with the received name of the feed; and

a display module configured to display one or more feed items of the identified feed.

* * * * *