US 20120233701A1

(19) United States

(12) Patent Application Publication

(10) Pub. No.: US 2012/0233701 A1

(43) Pub. Date: Sep. 13, 2012

Kidron

(54) **CONTENT LICENSE ACQUISITION PLATFORM APPARATUSES, METHODS AND SYSTEMS**

(76) Inventor: **Adam Kidron**, Bronx, NY (US)

(21) Appl. No.: **13/248,017**

(22) Filed: **Sep. 28, 2011**

Related U.S. Application Data

(60) Provisional application No. 61/526,210, filed on Aug. 22, 2011, provisional application No. 61/496,512, filed on Jun. 13, 2011, provisional application No. 61/387,450, filed on Sep. 28, 2010, provisional application No. 61/387,453, filed on Sep. 28, 2010.

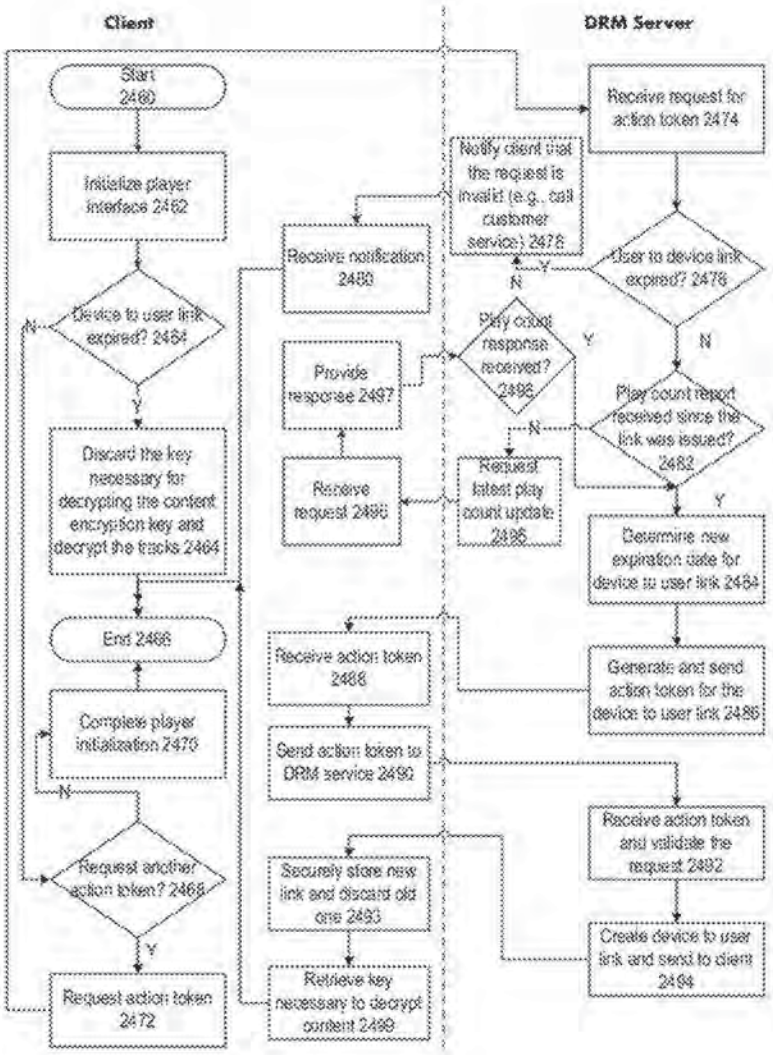
Publication Classification

(51) **Int. Cl.**
G06F 21/00 (2006.01)

(52) **U.S. Cl.** **726/26**

(57) **ABSTRACT**

The CONTENT LICENSE ACQUISITION PLATFORM APPARATUSES, METHODS AND SYSTEMS ("CLAP") transform content seed selections and recommendations via CLAP components such as discovery and social influence into events and discovery of other contents for users and revenue for right-holders. The CLAP may identify an unlicensed content item and uniquely resolve it within a universally resolvable media content ("URMC") service. The CLAP may obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time and an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time. The aggregate URMC service user engagement metrics may be evaluated using at least one URMC license request threshold rule. A target for a license request for the uniquely resolved content item may be identified and the license request may be sent to the identified target.



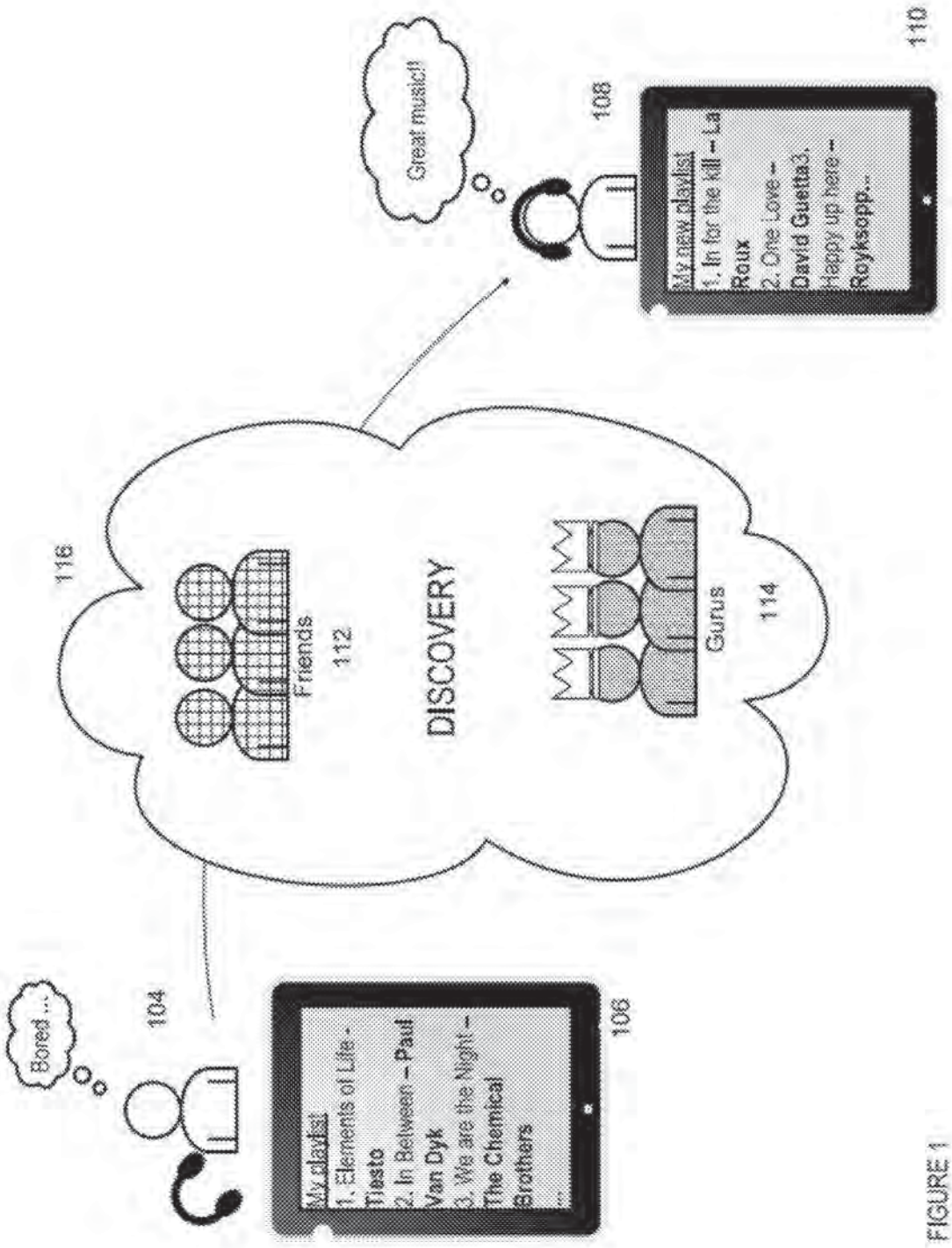

Bored...
104
My playlist
1. Elements of Life - Tiesto
2. In Between – Paul Van Dyk
3. We are the Night – The Chemical Brothers
...
106
DISCOVERY
Friends
112
Gurus
114
116
Great music!!
108
My new playlist
1. In for the kill – La Roux
2. One Love – David Guetta3. Happy up here – Royksopp...
110

FIGURE 1

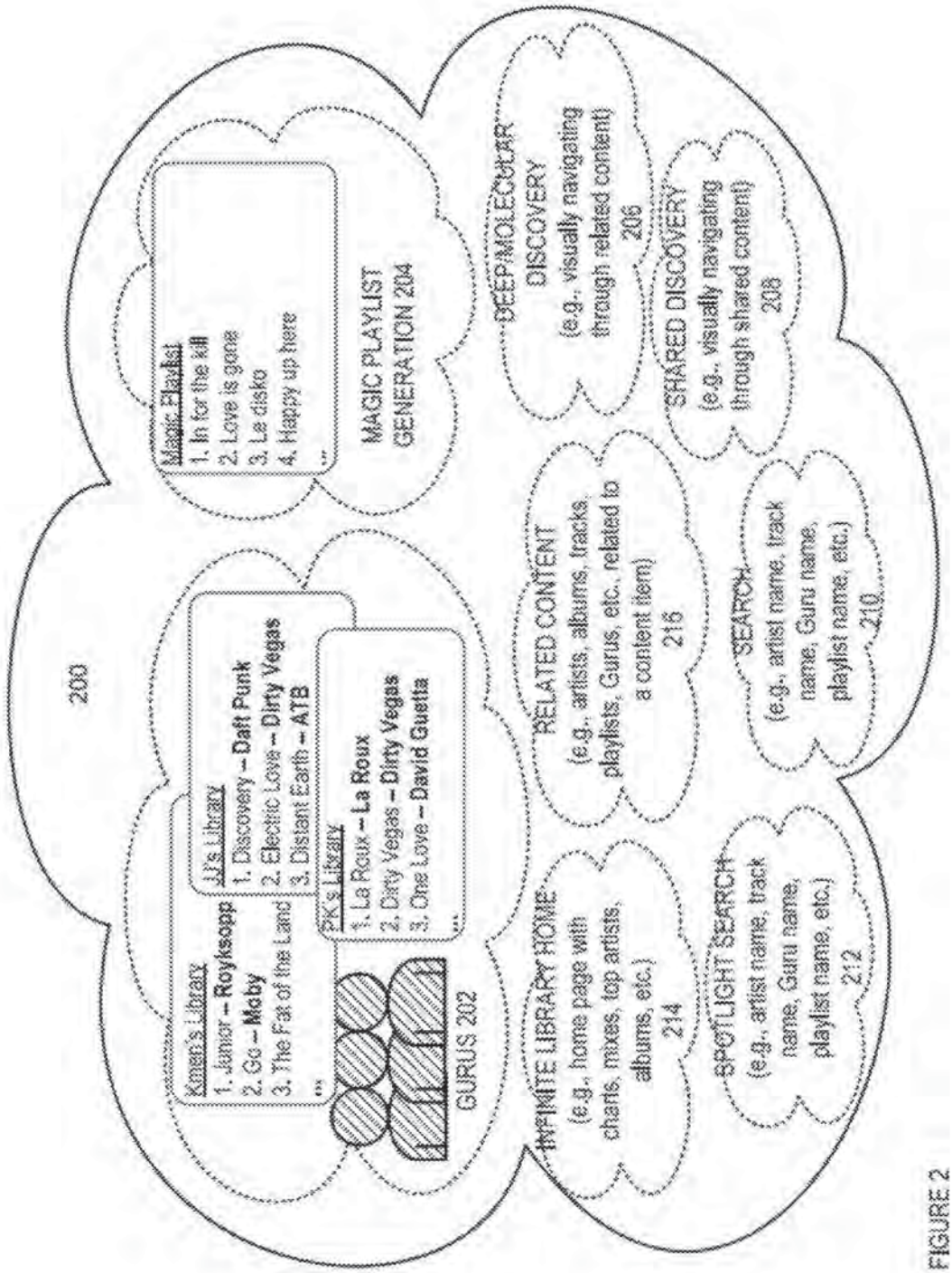

200
Kmen's Library
1. Junior -- Royksopp
2. Go -- Moby
3. The Fat of the Land
...
JJ's Library
1. Discovery -- Daft Punk
2. Electric Love -- Dirty Vegas
3. Distant Earth -- ATB
...
PK's Library
1. La Roux -- La Roux
2. Dirty Vegas -- Dirty Vegas
3. One Love -- David Guetta
...
GURUS 202
Magic Playlist
1. In for the kill
2. Love is gone
3. La disko
4. Happy up here
...
MAGIC PLAYLIST GENERATION 204
DEEP/MOLECULAR DISCOVERY (e.g., visually navigating through related content) 206
SHARED DISCOVERY (e.g., visually navigating through shared content) 208
RELATED CONTENT (e.g., artists, albums, tracks, playlists, Gurus, etc., related to a content item) 216
SEARCH (e.g., artist name, track name, Guru name, playlist name, etc.) 210
INFINITE LIBRARY HOME (e.g., home page with charts, mixes, top artists, albums, etc.) 214
SPOTLIGHT SEARCH (e.g., artist name, track name, Guru name, playlist name, etc.) 212

FIGURE 2

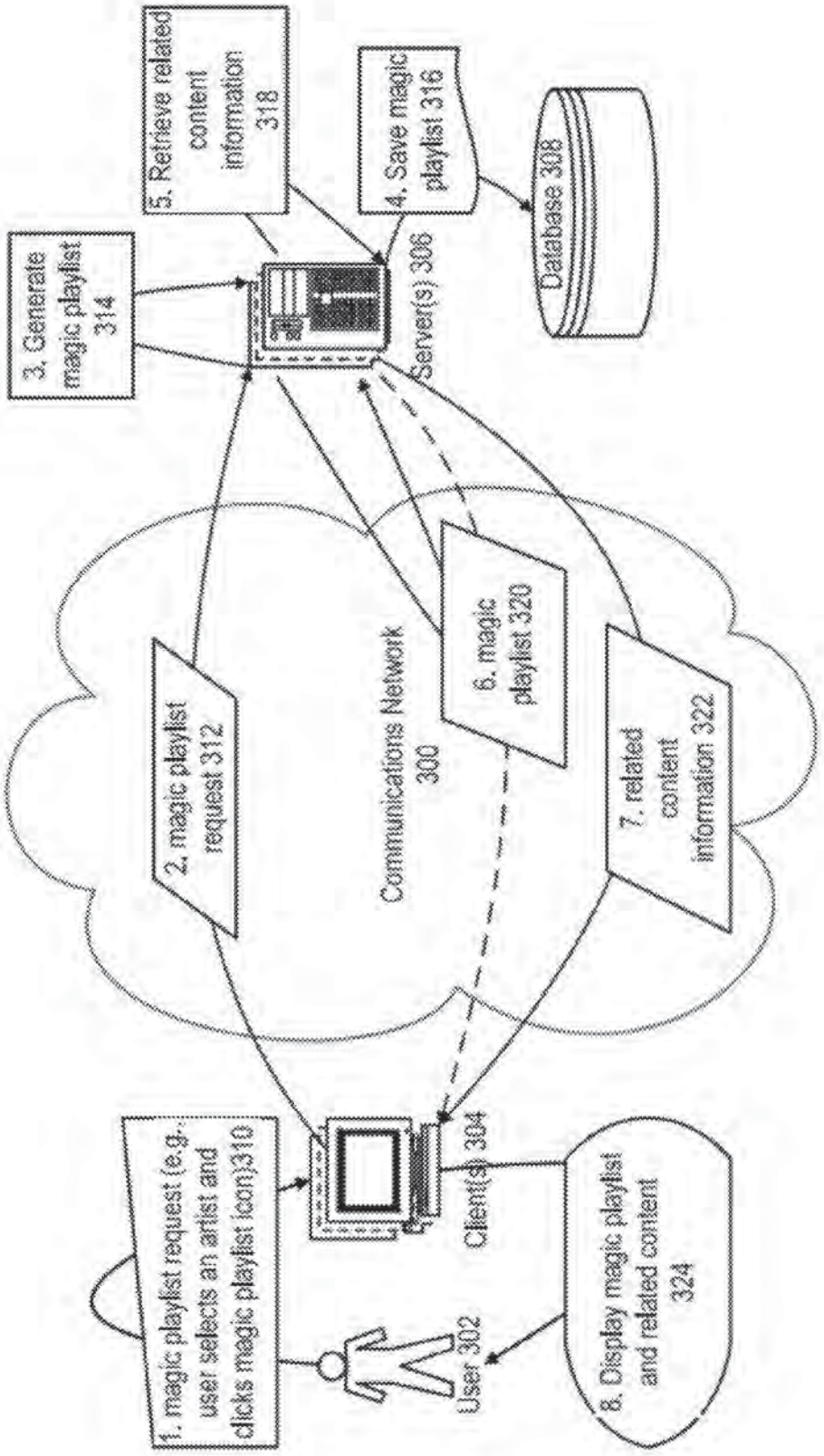

1. magic playlist request (e.g., user selects an artist and clicks magic playlist icon)310
User 302
Client(s) 304
2. magic playlist request 312
3. Generate magic playlist 314
Server(s) 306
4. Save magic playlist 316
5. Retrieve related content information 318
Database 308
Communications Network 300
6. magic playlist 320
7. related content information 322
8. Display magic playlist and related content 324

FIGURE 3

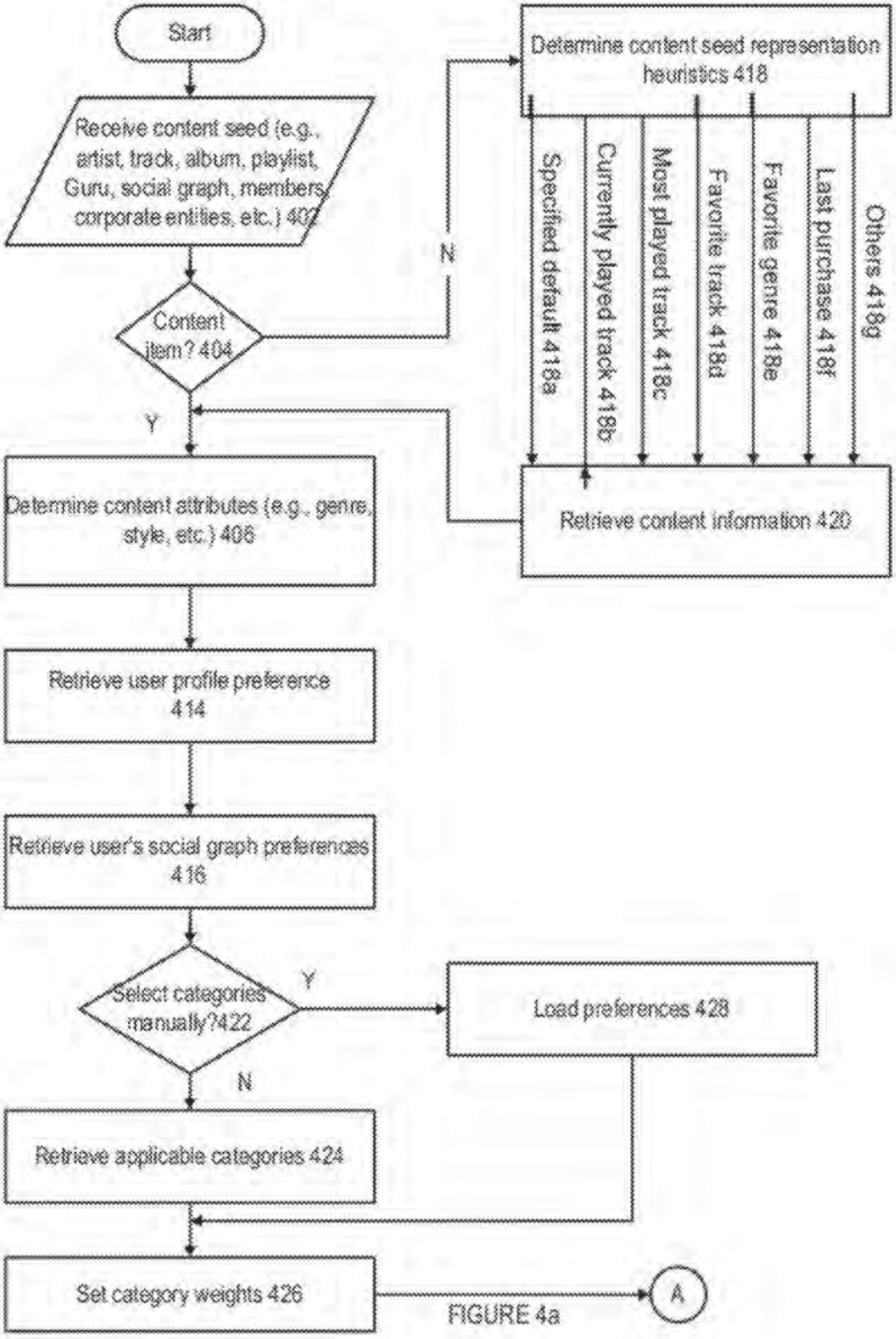

Start
Receive content seed (e.g., artist, track, album, playlist, Guru, social graph, members, corporate entities, etc.) 402
Content item? 404
N
Y
Determine content seed representation heuristics 418
Specified default 418a
Currently played track 418b
Most played track 418c
Favorite track 418d
Favorite genre 418e
Last purchase 418f
Others 418g
Retrieve content information 420
Determine content attributes (e.g., genre, style, etc.) 406
Retrieve user profile preference 414
Retrieve user's social graph preferences 416
Select categories manually?422
Y
N
Load preferences 428
Retrieve applicable categories 424
Set category weights 426
A

FIGURE 4a

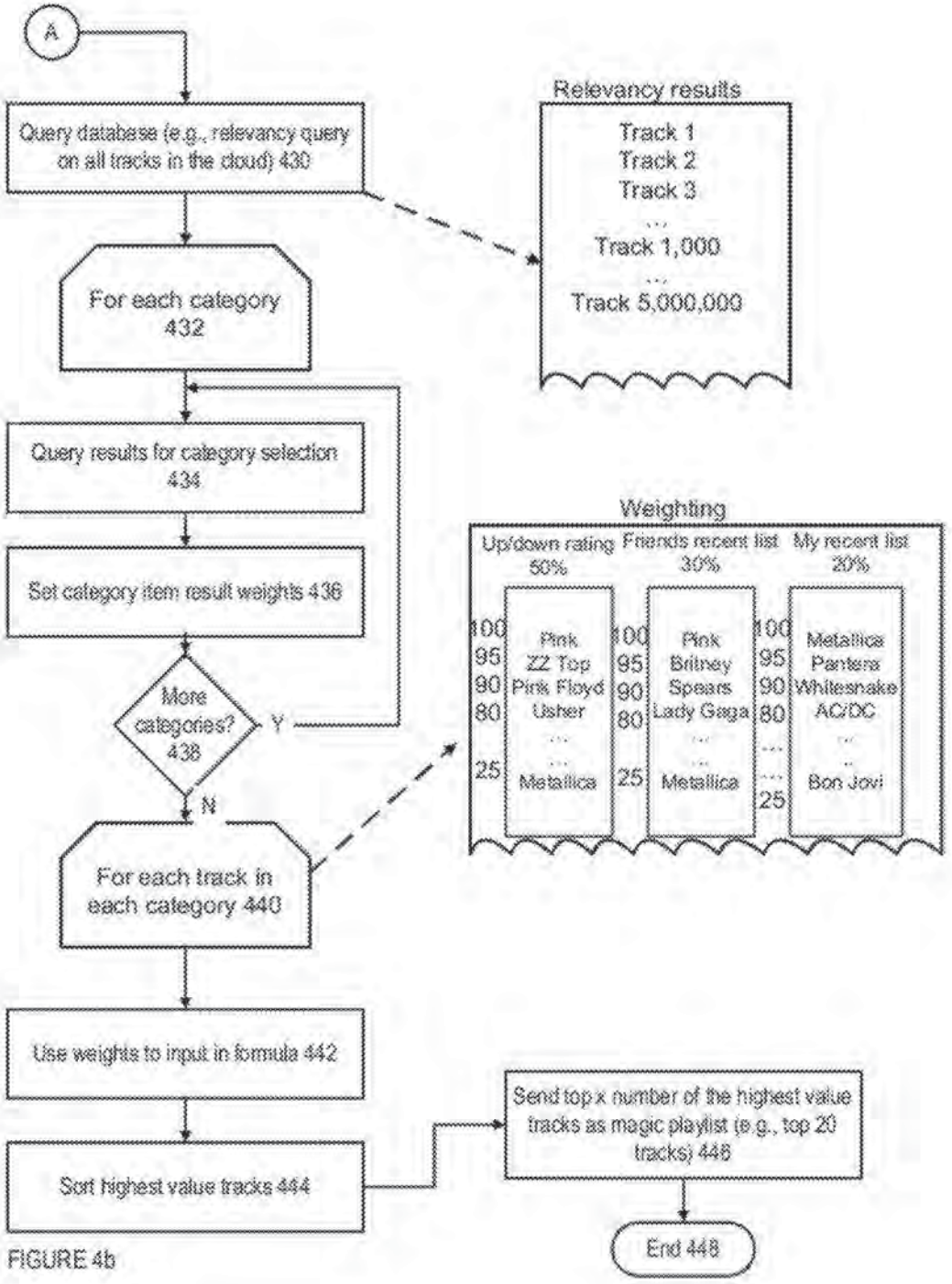

A
Query database (e.g., relevancy query on all tracks in the cloud) 430
Relevancy results
Track 1
Track 2
Track 3
...
Track 1,000
...
Track 5,000,000
For each category 432
Query results for category selection 434
Set category item result weights 436
More categories? 438
Y
N
Weighting
Up/down rating 50%
Friends recent list 30%
My recent list 20%
100
95
90
80
25
Pink
ZZ Top
Pink Floyd
Usher
Metallica
Pink
Britney Spears
Lady Gaga
Metallica
Metallica
Pantera
Whitesnake
AC/DC
Bon Jovi
For each track in each category 440
Use weights to input in formula 442
Sort highest value tracks 444
Send top x number of the highest value tracks as magic playlist (e.g., top 20 tracks) 446
End 448

FIGURE 4b

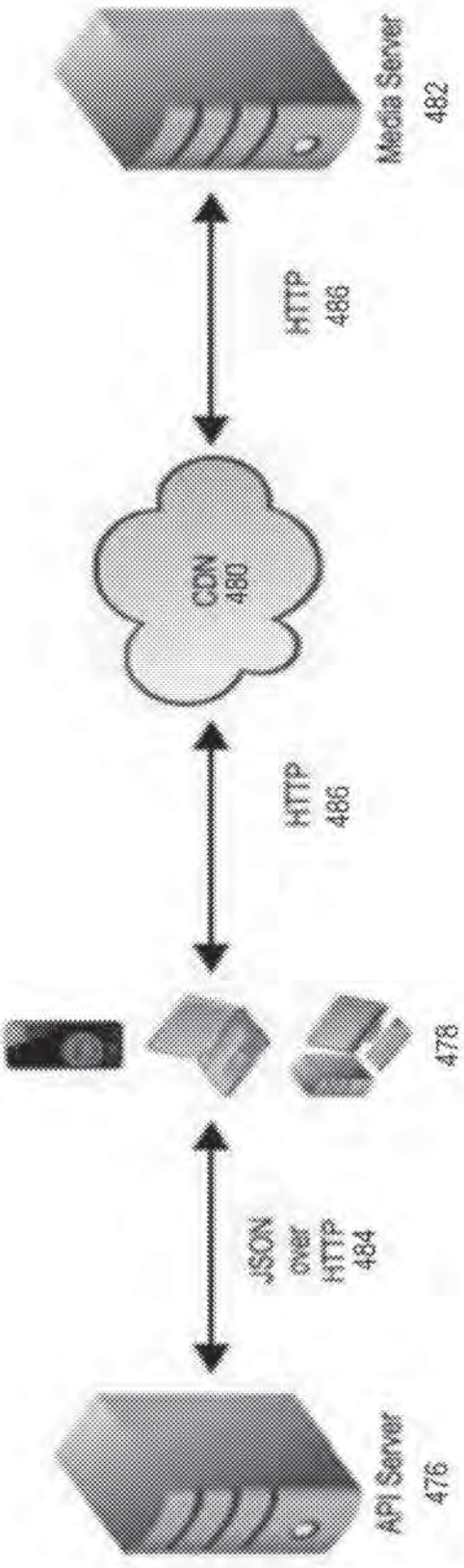

Media Server
482
HTTP
486
CDN
480
HTTP
486
478
JSON
over
HTTP
484
API Server
476

FIGURE 4c

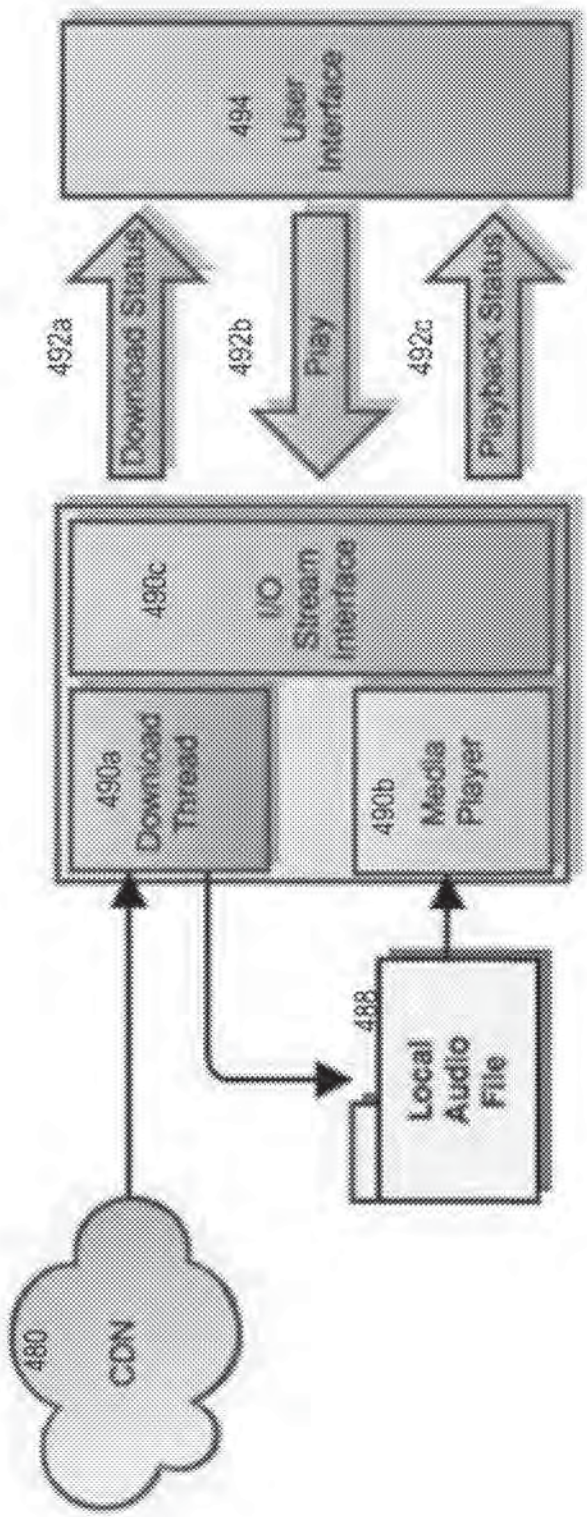

494
User Interface
492a
Download Status
492b
Play
492c
Playback Status
490c
I/O Stream Interface
490a
Download Thread
490b
Media Player
488
Local Audio File
480
CDN

FIGURE 4d

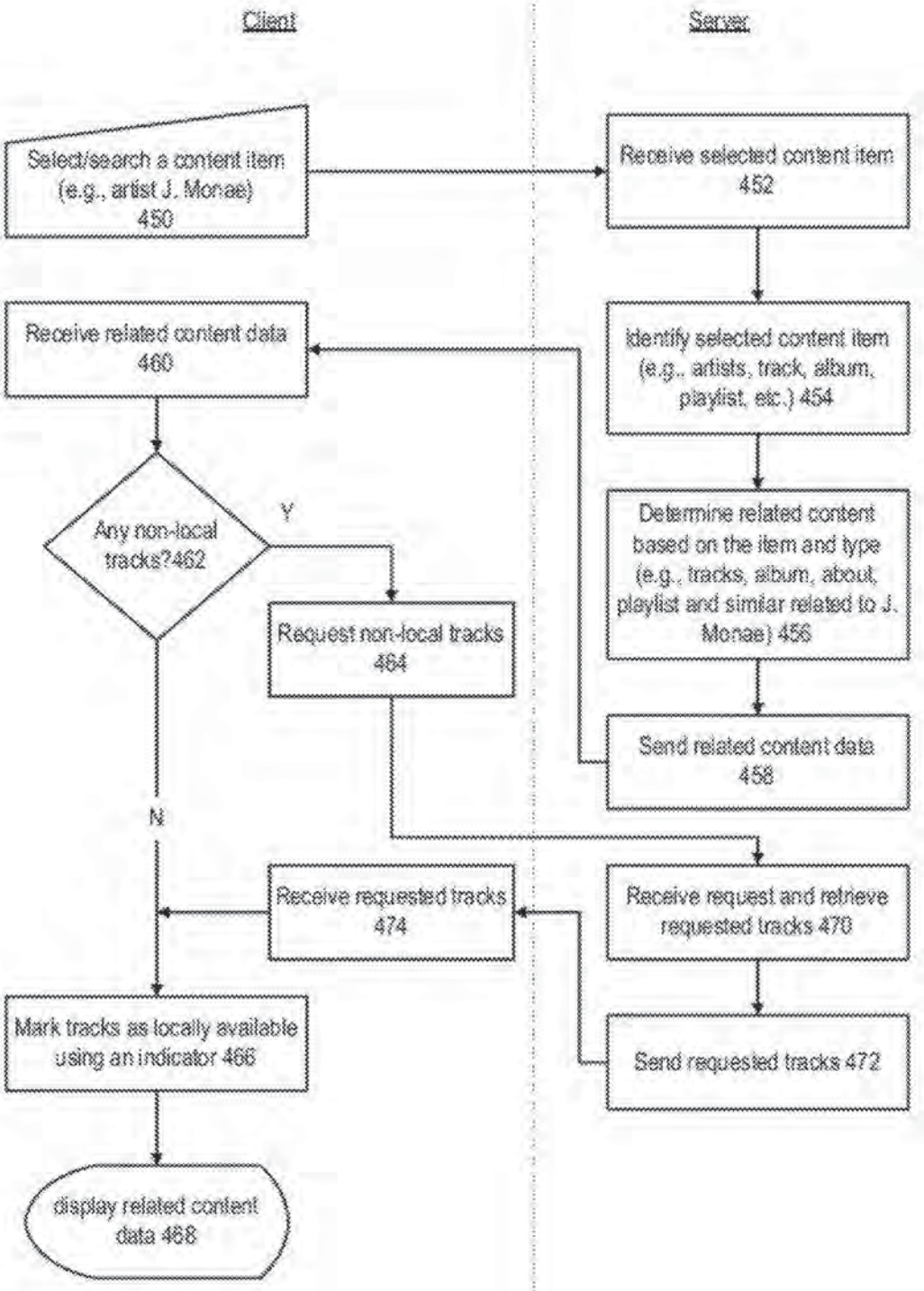

Client
Server
Select/search a content item (e.g., artist J. Monae) 450
Receive selected content item 452
Identify selected content item (e.g., artists, track, album, playlist, etc.) 454
Determine related content based on the item and type (e.g., tracks, album, about, playlist and similar related to J. Monae) 456
Send related content data 458
Receive related content data 460
Any non-local tracks?462
Y
N
Request non-local tracks 464
Receive request and retrieve requested tracks 470
Send requested tracks 472
Receive requested tracks 474
Mark tracks as locally available using an indicator 466
display related content data 468

FIGURE 4e

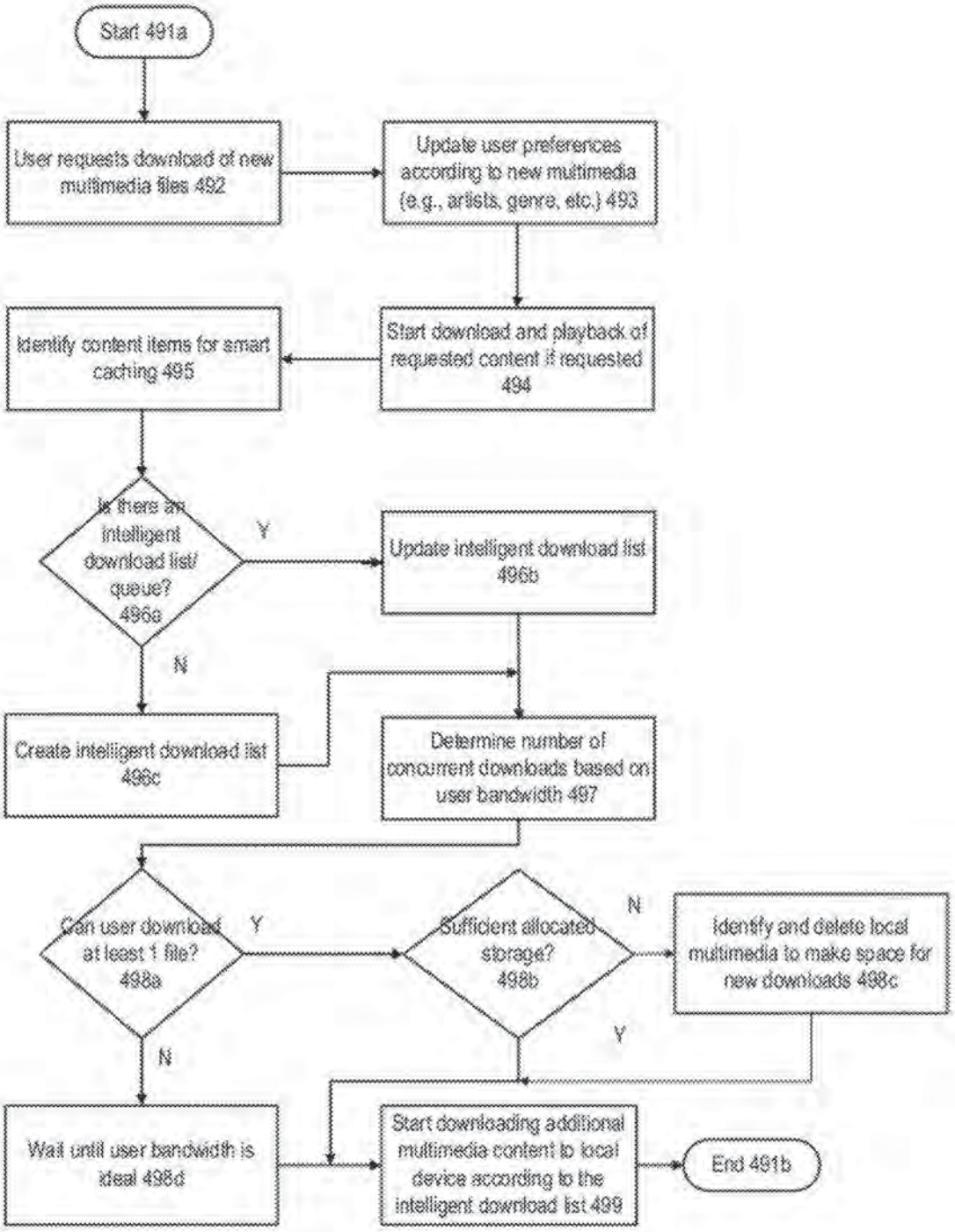

Start 491a
User requests download of new multimedia files 492
Update user preferences according to new multimedia (e.g., artists, genre, etc.) 493
Start download and playback of requested content if requested 494
Identify content items for smart caching 495
Is there an intelligent download list/ queue? 496a
Y
Update intelligent download list 496b
N
Create intelligent download list 496c
Determine number of concurrent downloads based on user bandwidth 497
Can user download at least 1 file? 498a
Y
Sufficient allocated storage? 498b
N
Identify and delete local multimedia to make space for new downloads 498c
Y
N
Wait until user bandwidth is ideal 498d
Start downloading additional multimedia content to local device according to the intelligent download list 499
End 491b

FIGURE 4f

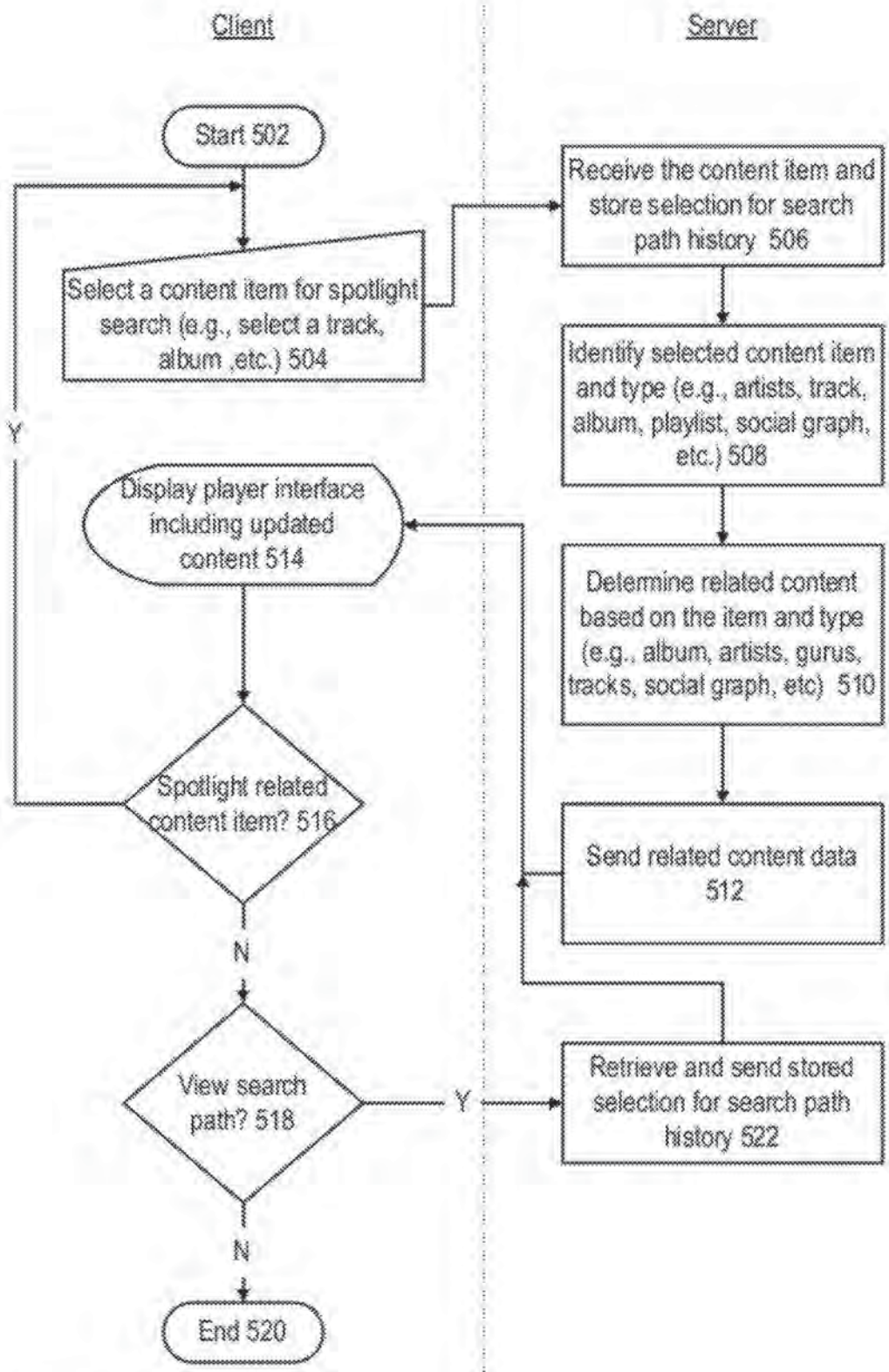

Client
Server
Start 502
Select a content item for spotlight search (e.g., select a track, album ,etc.) 504
Receive the content item and store selection for search path history 506
Identify selected content item and type (e.g., artists, track, album, playlist, social graph, etc.) 508
Determine related content based on the item and type (e.g., album, artists, gurus, tracks, social graph, etc) 510
Send related content data 512
Display player interface including updated content 514
Spotlight related content item? 516
Y
N
View search path? 518
Y
Retrieve and send stored selection for search path history 522
N
End 520

FIGURE 5

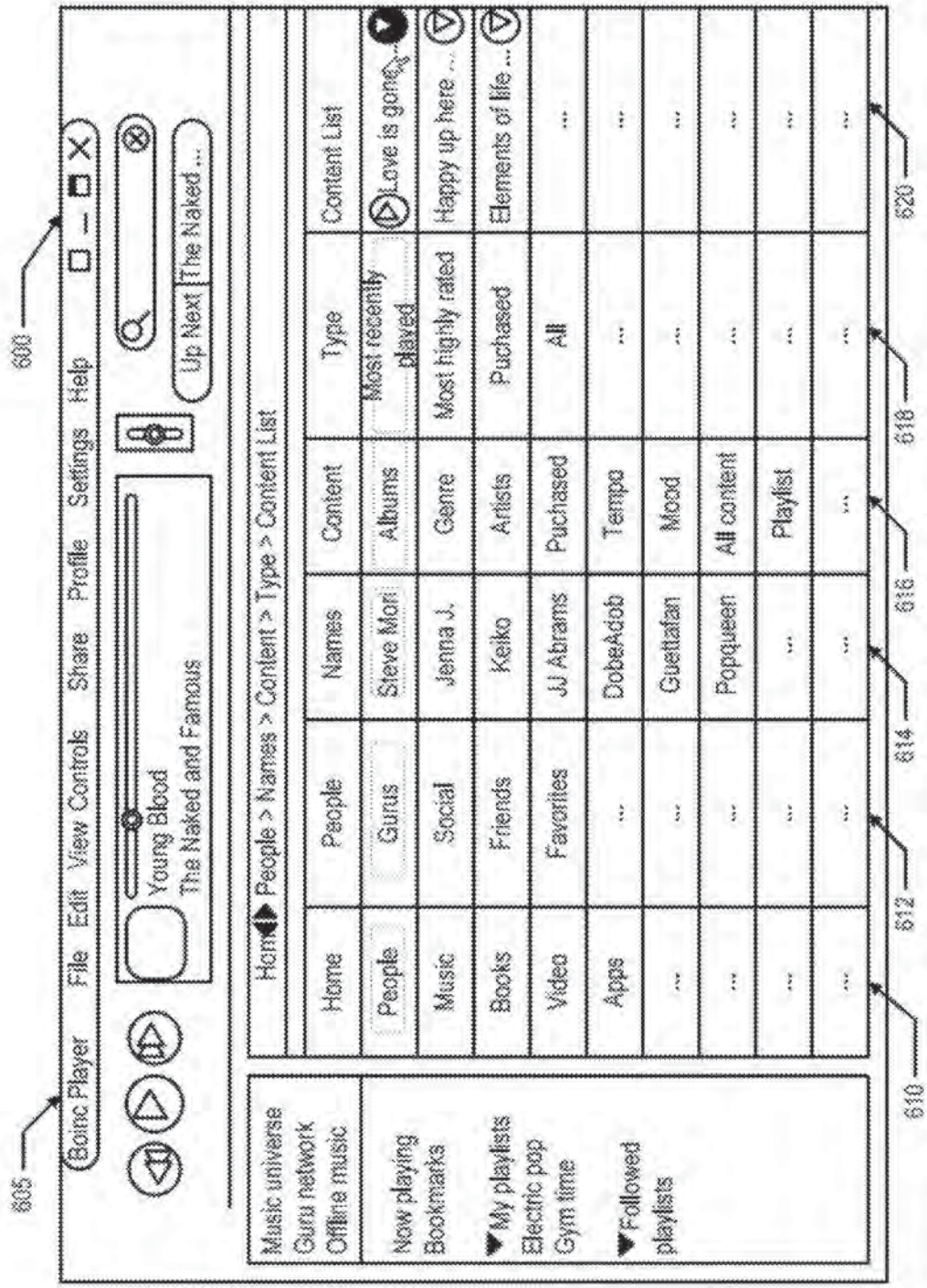

605
600
Boinc Player File Edit View Controls Share Profile Settings Help
Young Blood
The Naked and Famous
Up Next The Naked ...
Music universe
Guru network
Offline music
Now playing
Bookmarks
My playlists
Electric pop
Gym time
Followed playlists
Hom People > Names > Content > Type > Content List
Home
People
Music
Books
Video
Apps
Home
People
Social
Friends
Favorites
Gurus
Steve Mori
Jenna J.
Keiko
JJ Abrams
DobeAdob
Guettafan
Popqueen
Names
Content
Albums
Genre
Artists
Purchased
Tempo
Mood
All content
Playlist
Type
Most recently played
Most highly rated
Purchased
All
Content List
Love is gone
Happy up here ...
Elements of life ...
610
612
614
616
618
620

FIGURE 6a

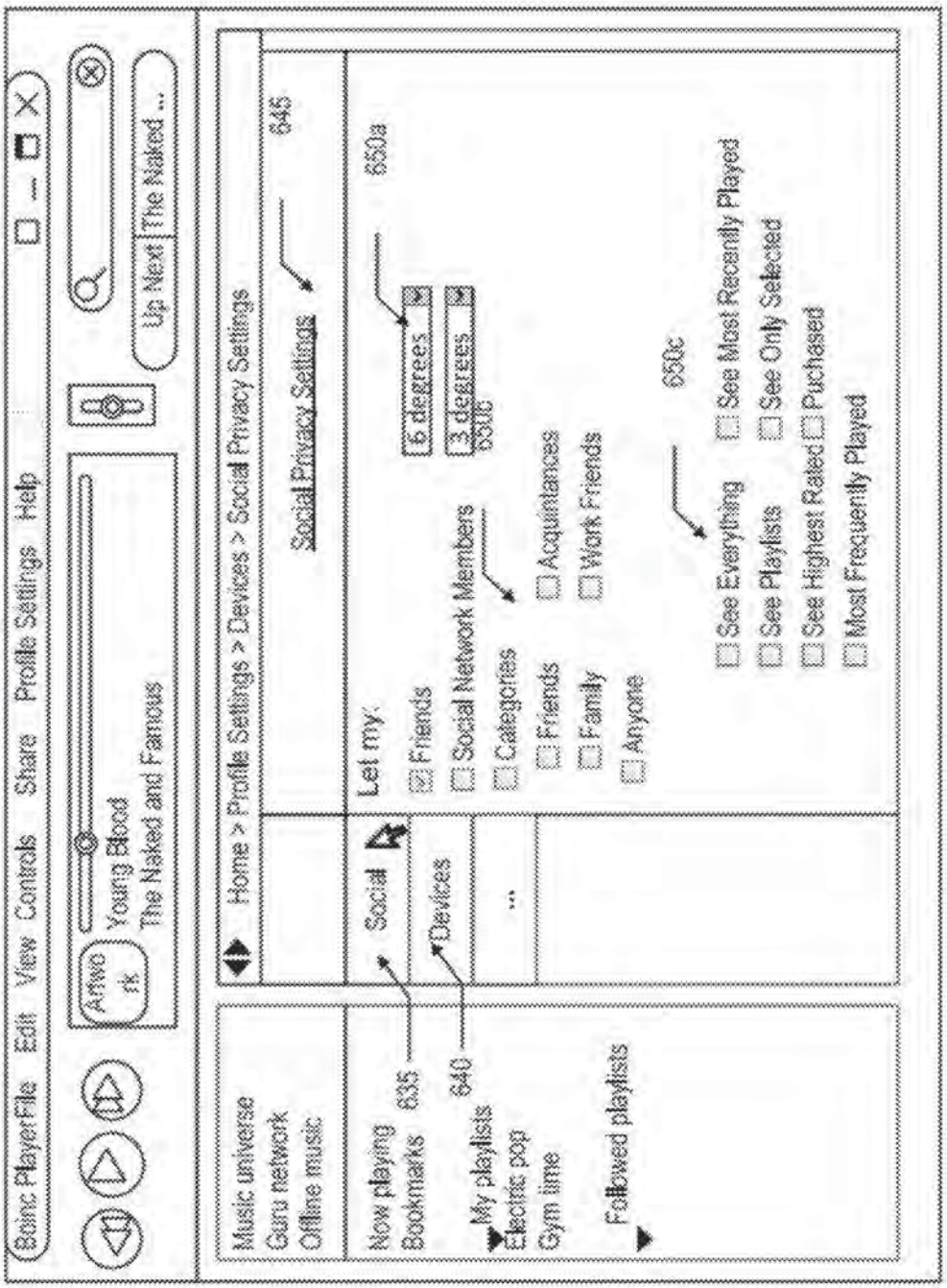

Boinc Player File Edit View Controls Share Profile Settings Help
Young Blood
The Naked and Famous
Up Next The Naked ...
Music universe
Guru network
Offline music
Now playing
Bookmarks
635
My playlists
Electric pop
Gym time
640
Followed playlists
Home > Profile Settings > Devices > Social Privacy Settings
Social Privacy Settings
645
Social
Devices
...
Let my:
650a
Friends
Social Network Members
6 degrees
3 degrees
650b
Categories
Friends
Family
Anyone
Acquintances
Work Friends
650c
See Everything
See Playlists
See Highest Rated
Most Frequently Played
See Most Recently Played
See Only Selected
Purchased

FIGURE 6b

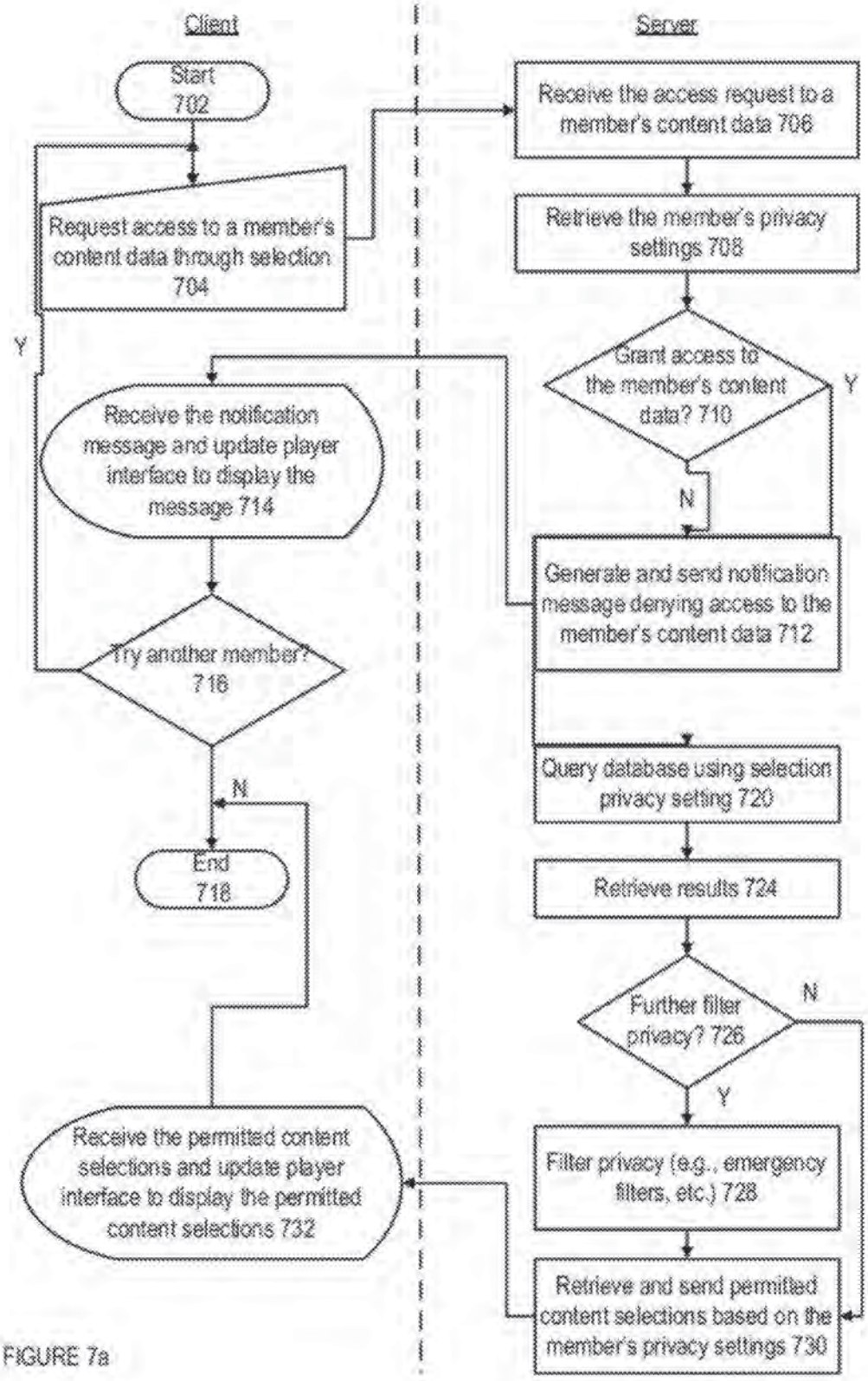

Client
Server
Start 702
Request access to a member's content data through selection 704
Receive the access request to a member's content data 706
Retrieve the member's privacy settings 708
Grant access to the member's content data? 710
Y
N
Generate and send notification message denying access to the member's content data 712
Receive the notification message and update player interface to display the message 714
Try another member? 716
Y
N
End 718
Query database using selection privacy setting 720
Retrieve results 724
Further filter privacy? 726
N
Y
Filter privacy (e.g., emergency filters, etc.) 728
Retrieve and send permitted content selections based on the member's privacy settings 730
Receive the permitted content selections and update player interface to display the permitted content selections 732

FIGURE 7a

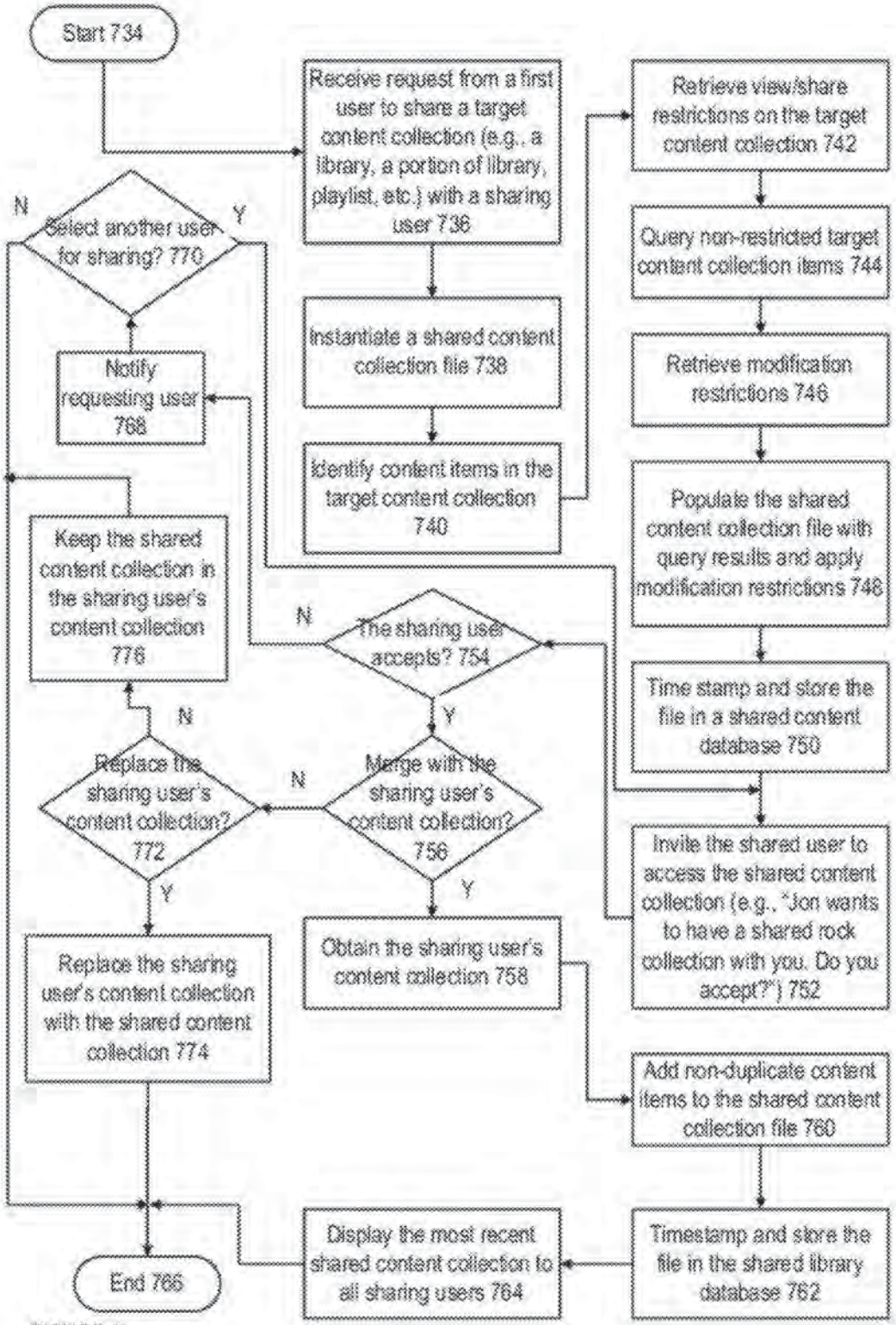

Start 734
Receive request from a first user to share a target content collection (e.g., a library, a portion of library, playlist, etc.) with a sharing user 736
Instantiate a shared content collection file 738
Identify content items in the target content collection 740
Retrieve view/share restrictions on the target content collection 742
Query non-restricted target content collection items 744
Retrieve modification restrictions 746
Populate the shared content collection file with query results and apply modification restrictions 748
Time stamp and store the file in a shared content database 750
Invite the shared user to access the shared content collection (e.g., "Jon wants to have a shared rock collection with you. Do you accept?") 752
The sharing user accepts? 754
N
Y
Merge with the sharing user's content collection? 756
N
Y
Obtain the sharing user's content collection 758
Add non-duplicate content items to the shared content collection file 760
Timestamp and store the file in the shared library database 762
Display the most recent shared content collection to all sharing users 764
End 766
Replace the sharing user's content collection? 772
N
Y
Replace the sharing user's content collection with the shared content collection 774
Keep the shared content collection in the sharing user's content collection 776
Select another user for sharing? 770
N
Y
Notify requesting user 768

FIGURE 7b

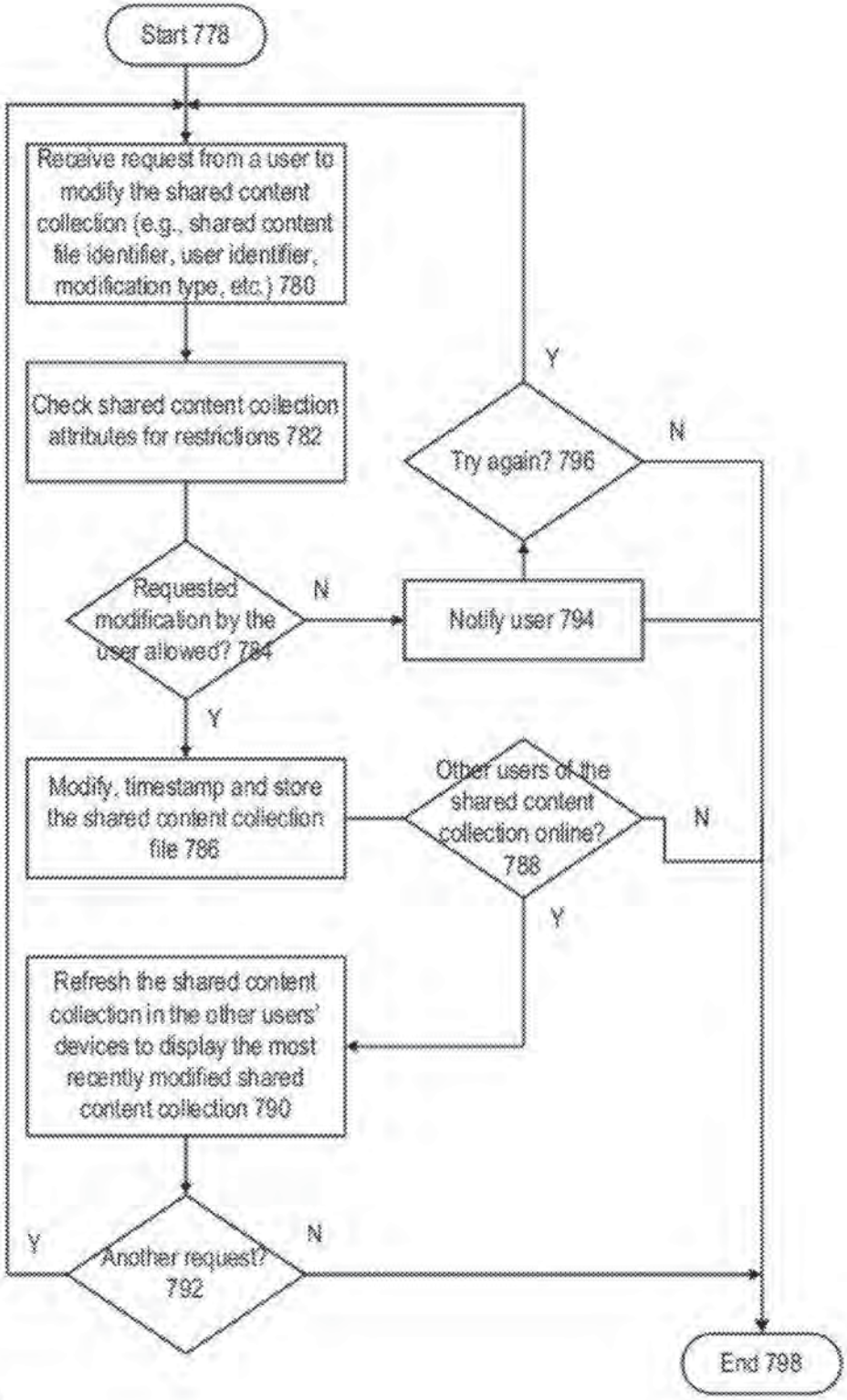

Start 778
Receive request from a user to modify the shared content collection (e.g., shared content file identifier, user identifier, modification type, etc.) 780
Check shared content collection attributes for restrictions 782
Requested modification by the user allowed? 784
N
Notify user 794
Try again? 796
Y
N
Y
Modify, timestamp and store the shared content collection file 786
Other users of the shared content collection online? 788
N
Y
Refresh the shared content collection in the other users' devices to display the most recently modified shared content collection 790
Another request? 792
Y
N
End 798

FIGURE 7c

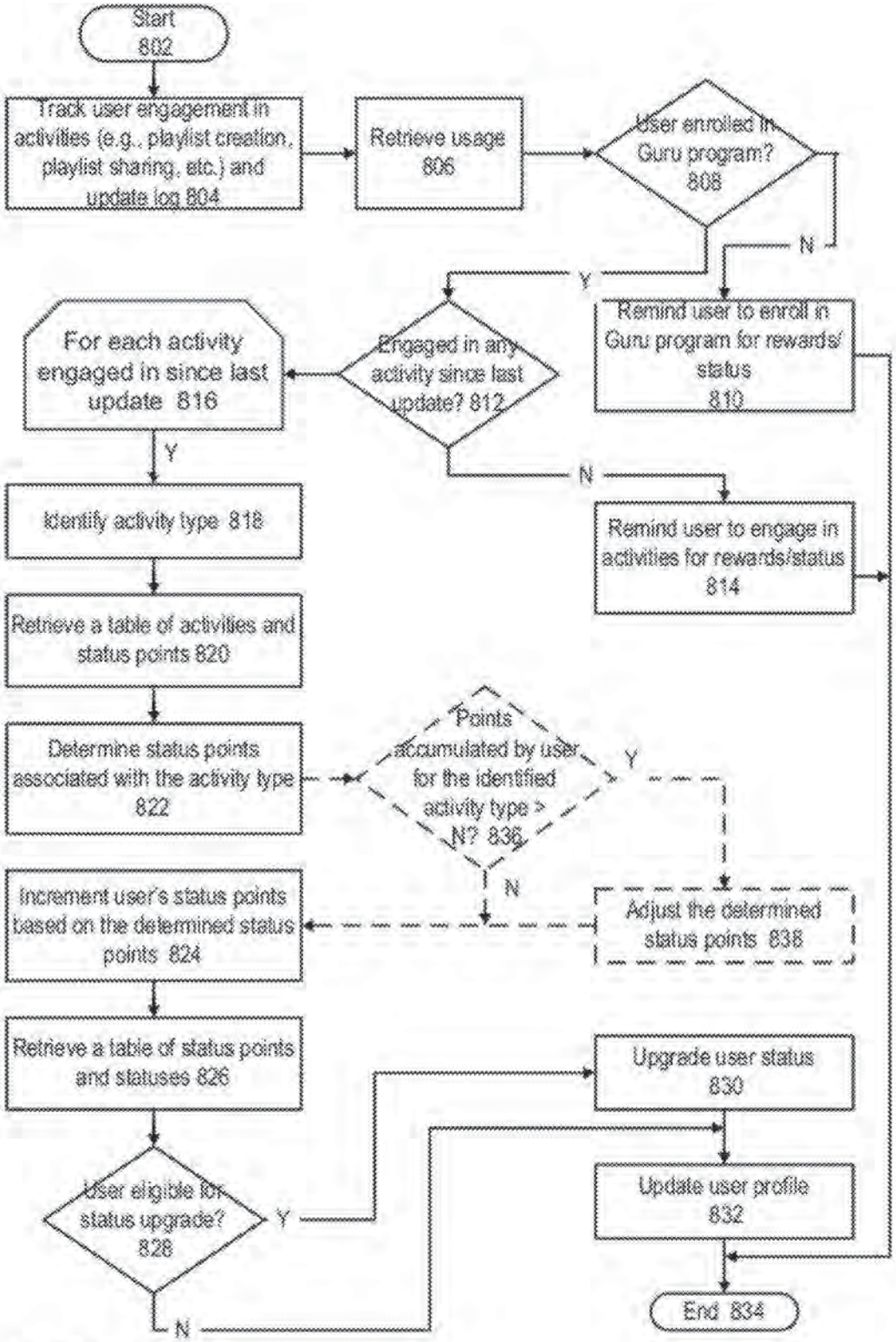

Start 802
Track user engagement in activities (e.g., playlist creation, playlist sharing, etc.) and update log 804
Retrieve usage 806
User enrolled in Guru program? 808
N
Y
Remind user to enroll in Guru program for rewards/ status 810
Engaged in any activity since last update? 812
For each activity engaged in since last update 816
N
Y
Remind user to engage in activities for rewards/status 814
Identify activity type 818
Retrieve a table of activities and status points 820
Determine status points associated with the activity type 822
Points accumulated by user for the identified activity type > N? 836
Y
N
Adjust the determined status points 838
Increment user's status points based on the determined status points 824
Retrieve a table of status points and statuses 826
Upgrade user status 830
User eligible for status upgrade? 828
Y
N
Update user profile 832
End 834

FIGURE 8a

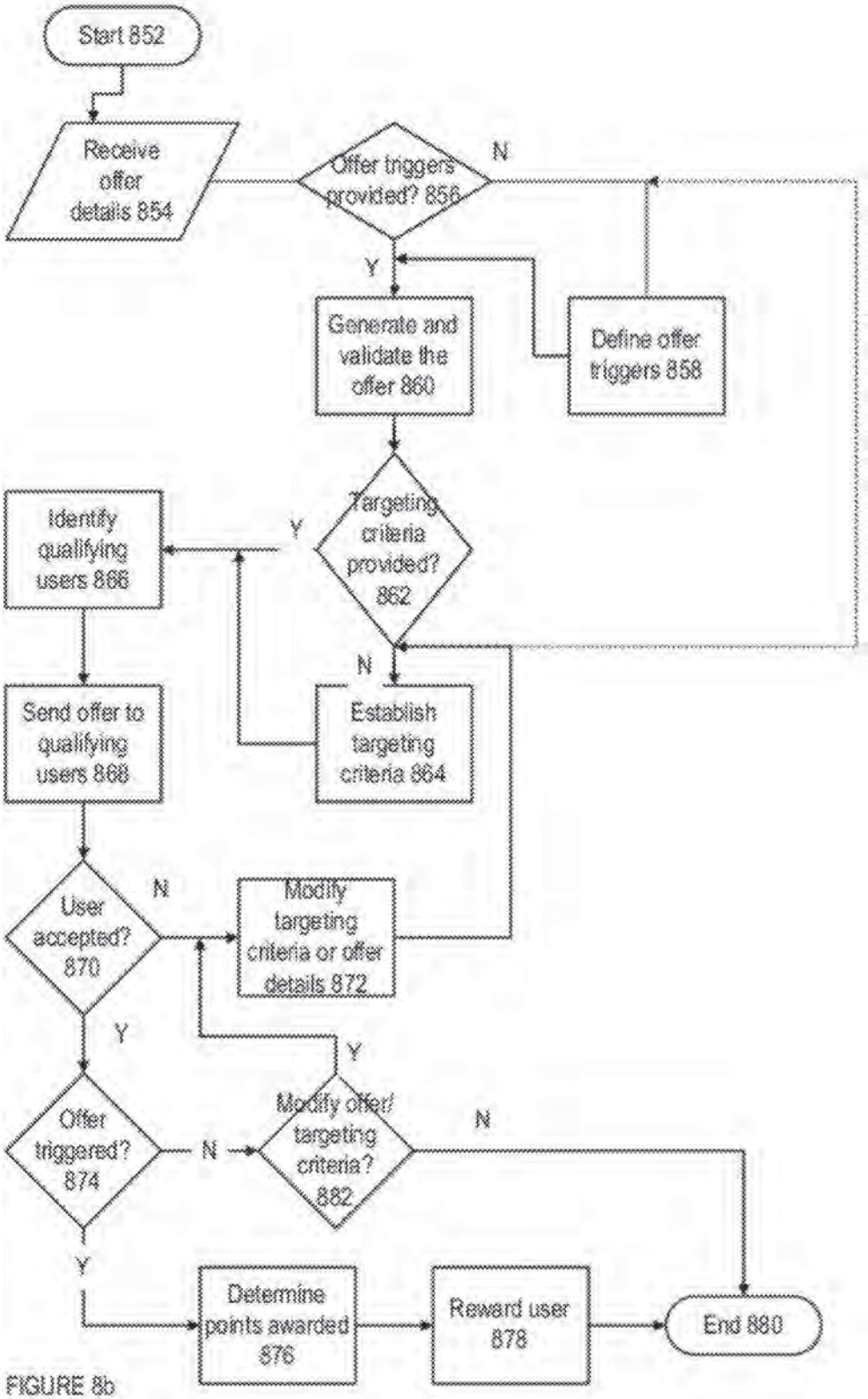

Start 852
Receive offer details 854
Offer triggers provided? 856
N
Y
Define offer triggers 858
Generate and validate the offer 860
Targeting criteria provided? 862
Y
N
Identify qualifying users 866
Establish targeting criteria 864
Send offer to qualifying users 868
User accepted? 870
N
Modify targeting criteria or offer details 872
Y
Y
Offer triggered? 874
N
Modify offer/ targeting criteria? 882
N
Y
Determine points awarded 876
Reward user 878
End 880

FIGURE 8b


900
905h
905i
925
925e
905e
905f
905g
905d
905b
910
920a
920
920b
920c
920d
920e
920f
905a
905
915
915a
915b
915c
915d
915e
915f
915g
915h

FIGURE 9

FIGURE 10

FIGURE 11a

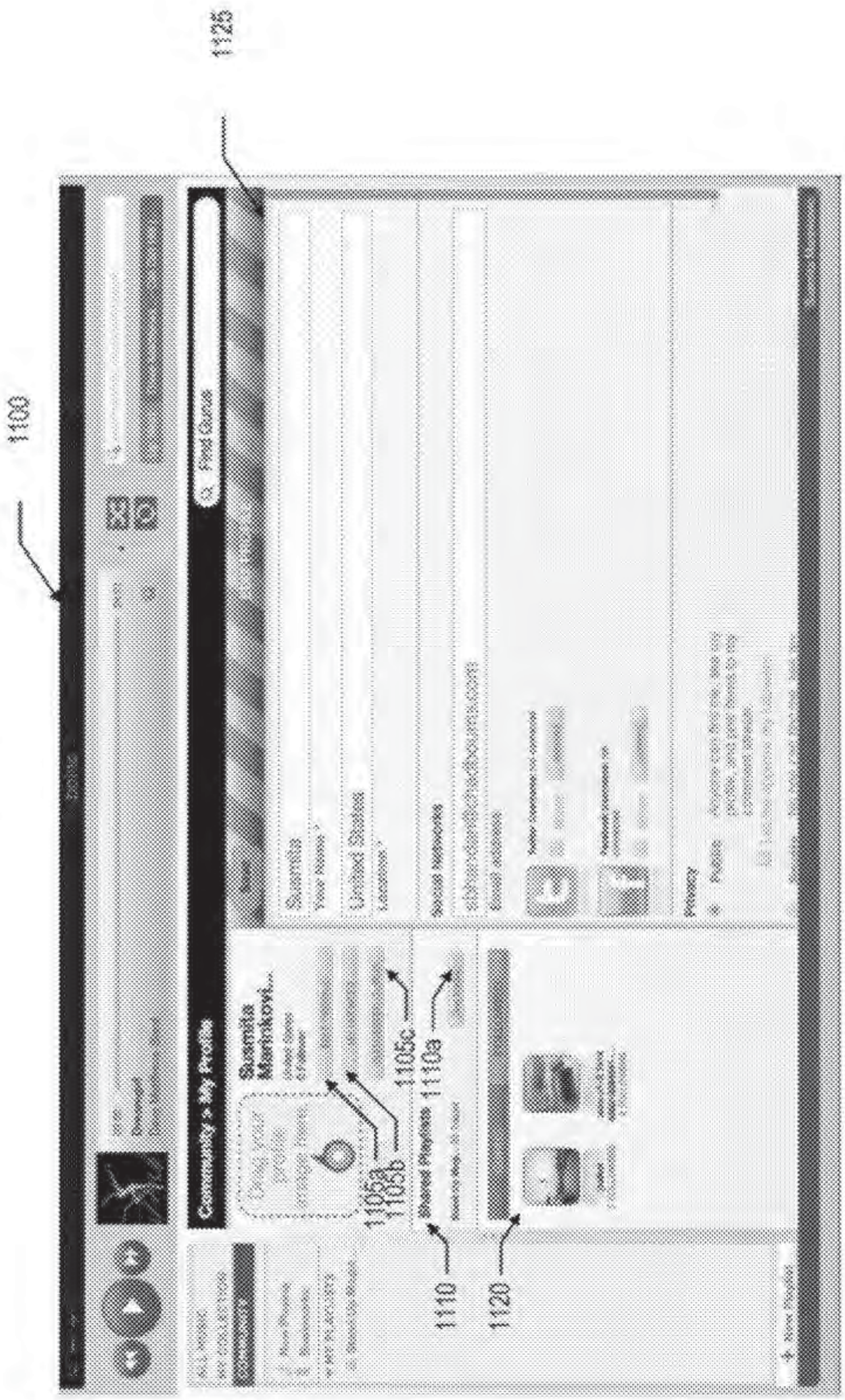


FIGURE 11b

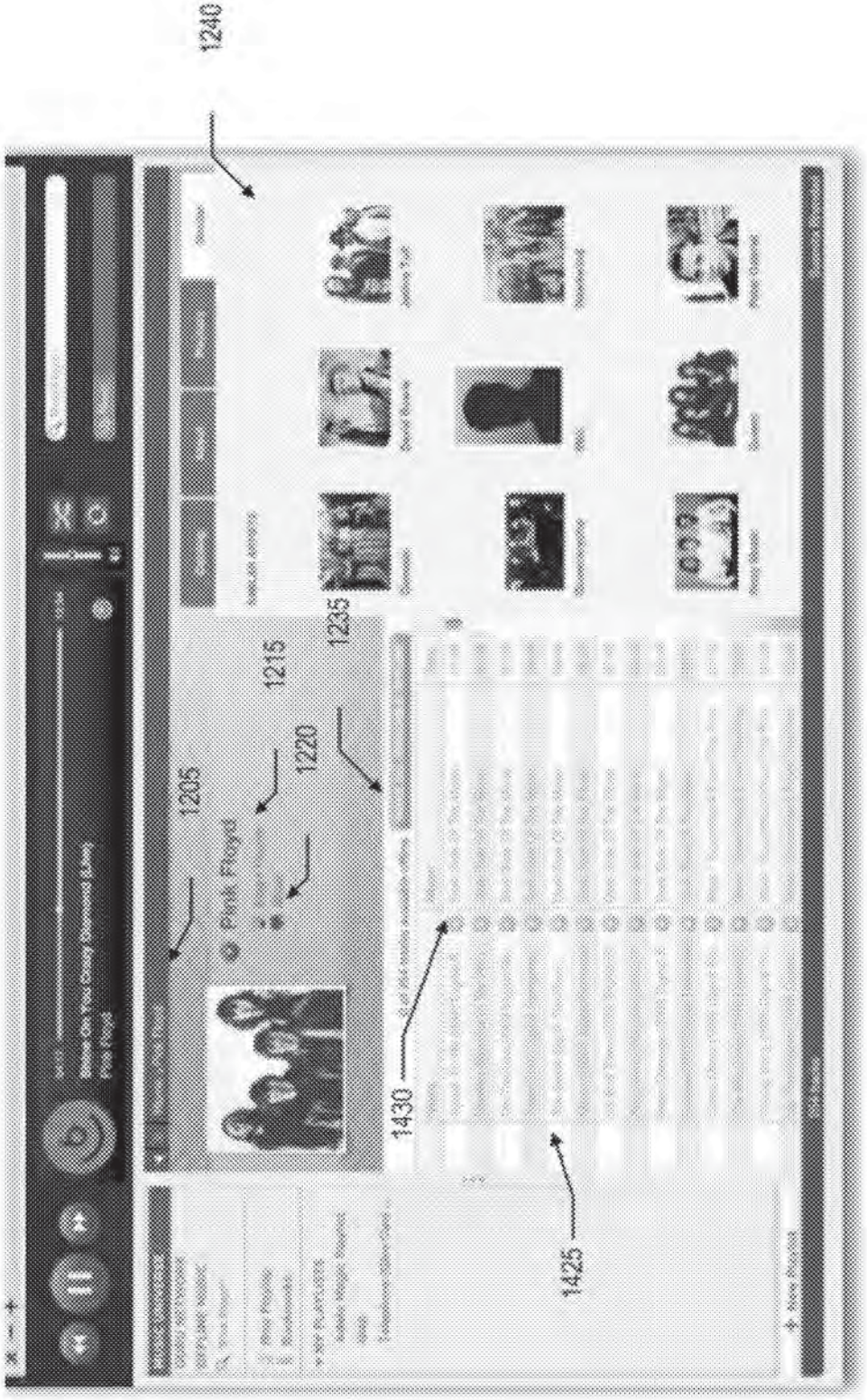


FIGURE 12a

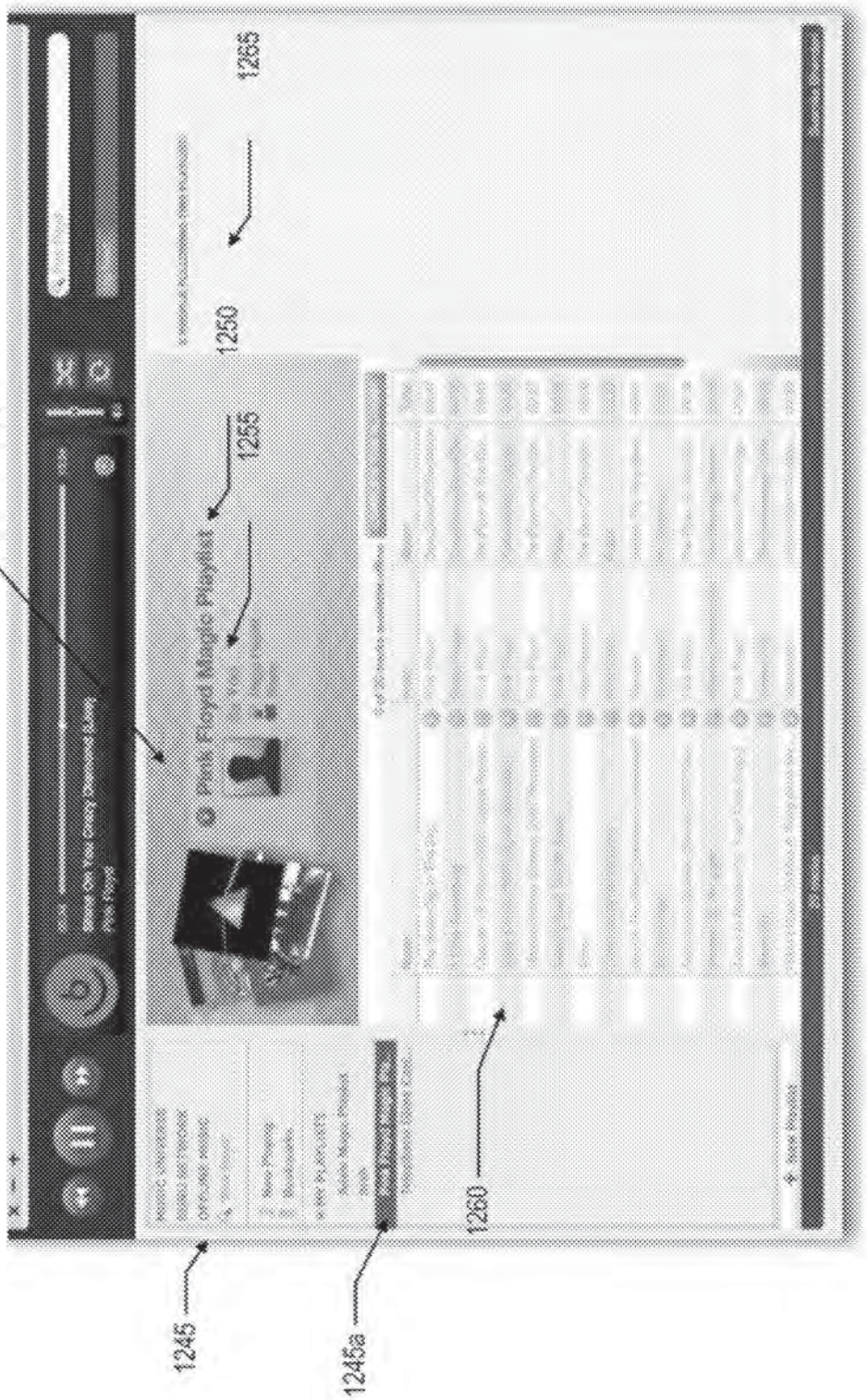


FIGURE 12b

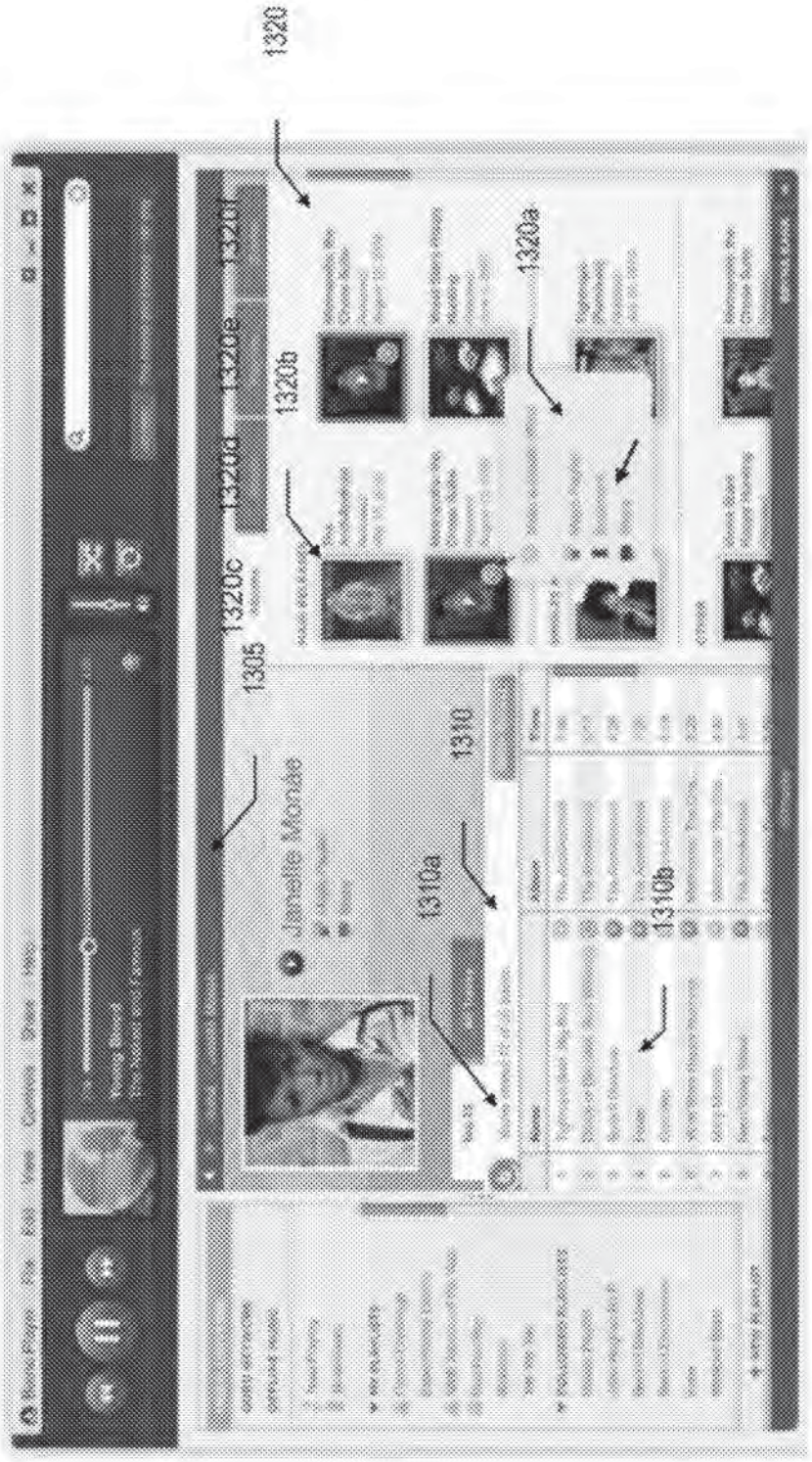

1320
1320a
1320b
1320c
1320d
1320e
1320f
1305
1310
1310a
1310b
Janelle Monáe

FIGURE 13a

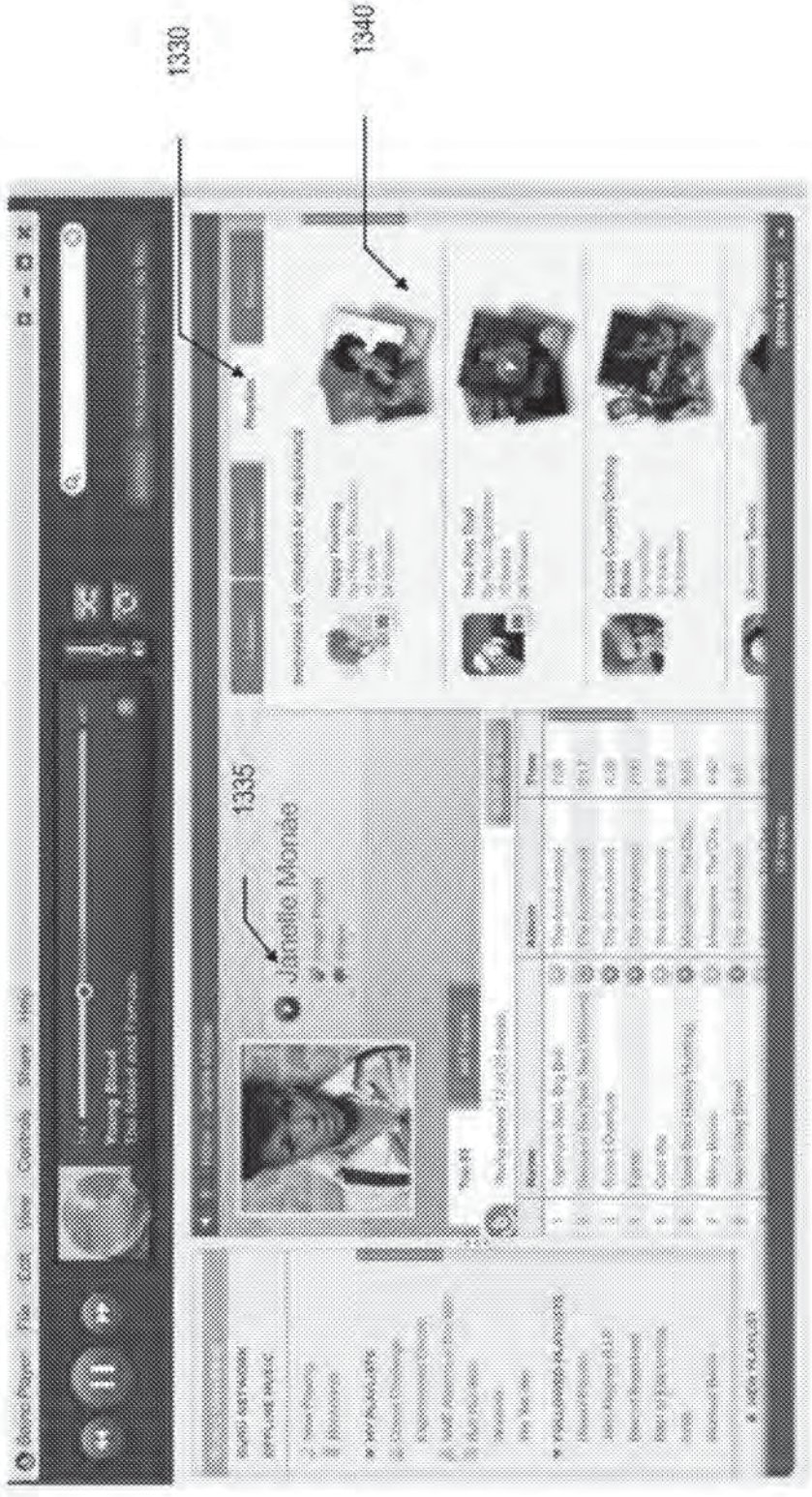


FIGURE 13b

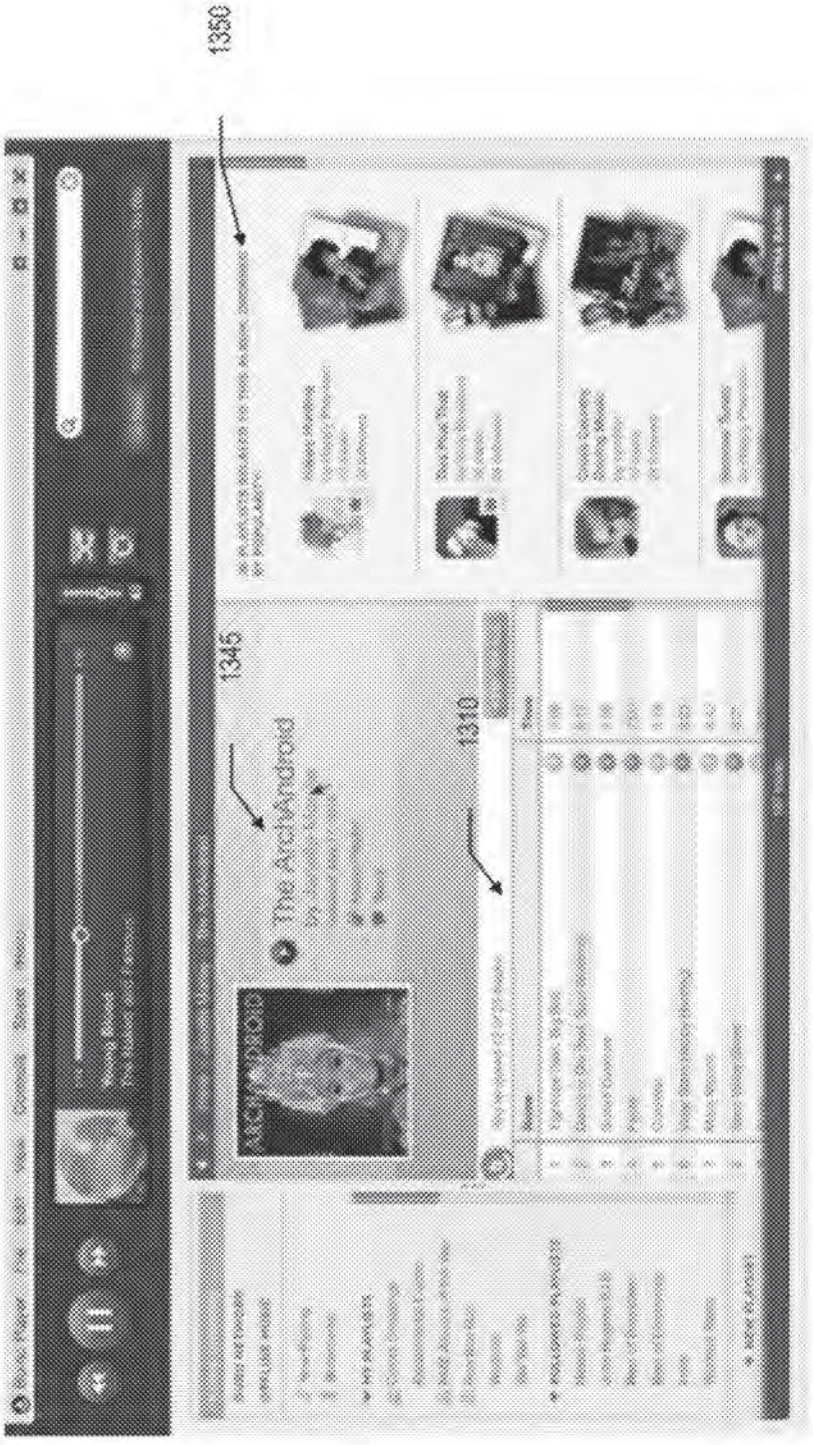


FIGURE 13c

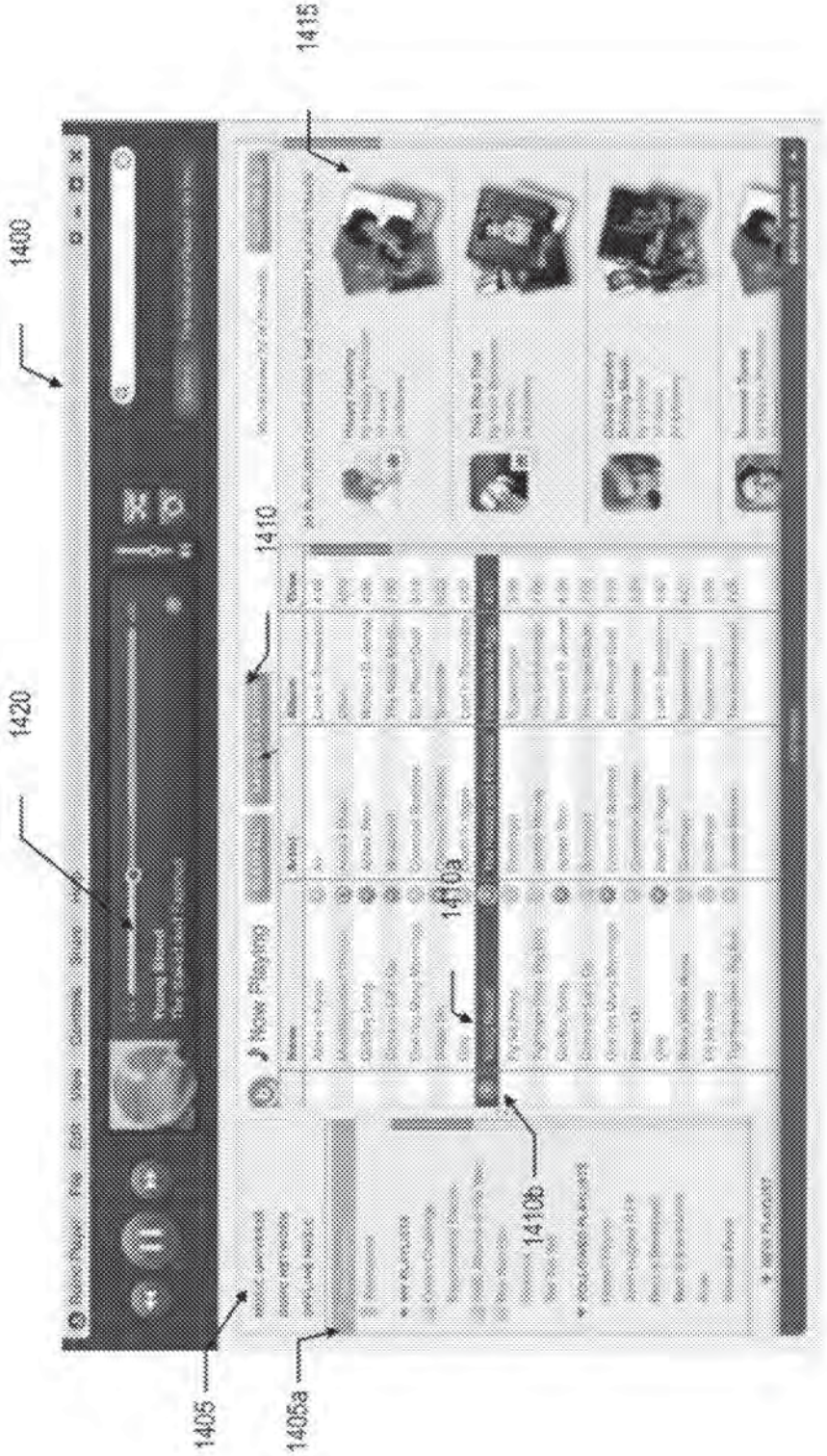


FIGURE 14

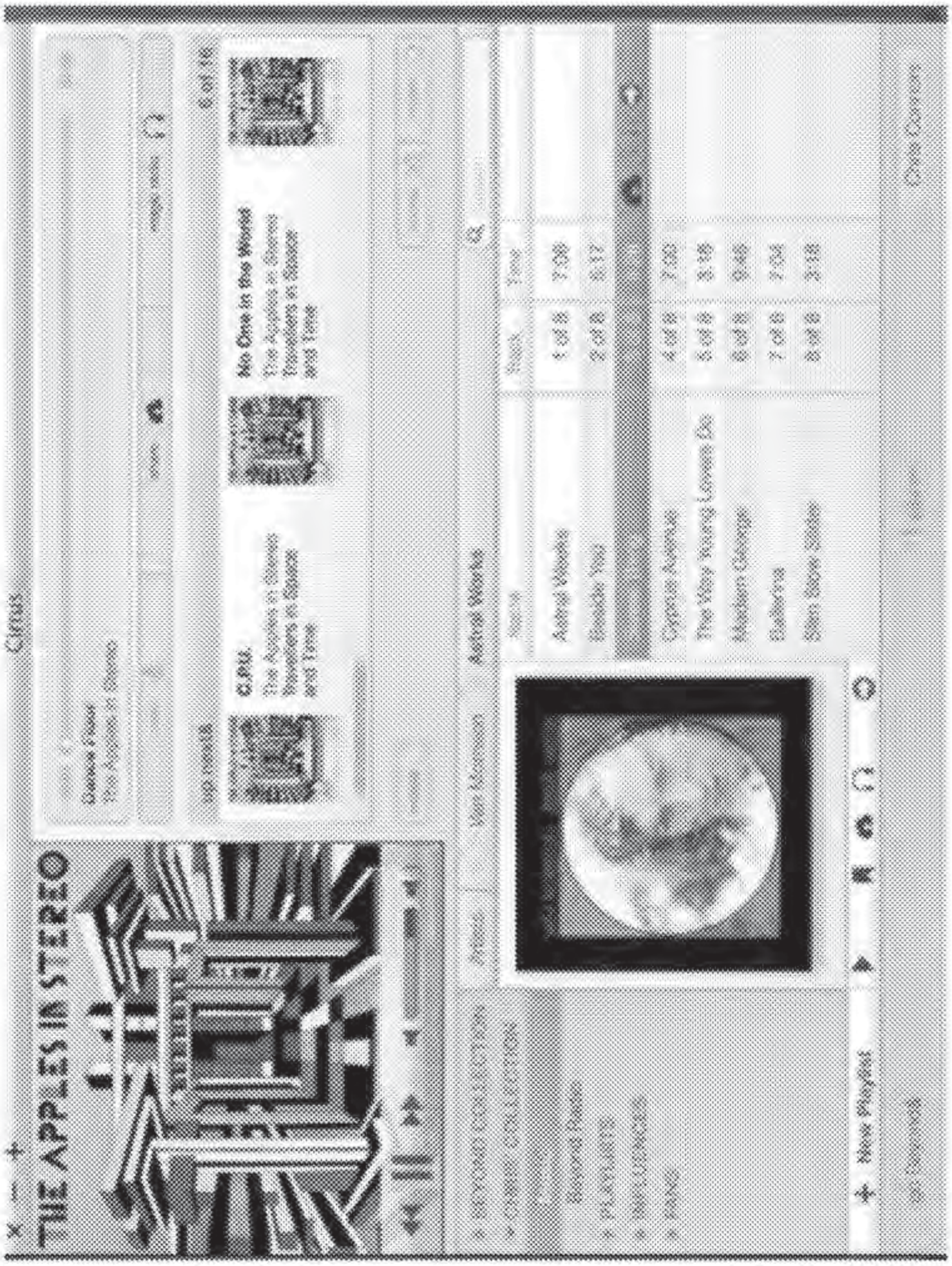


FIGURE 15a

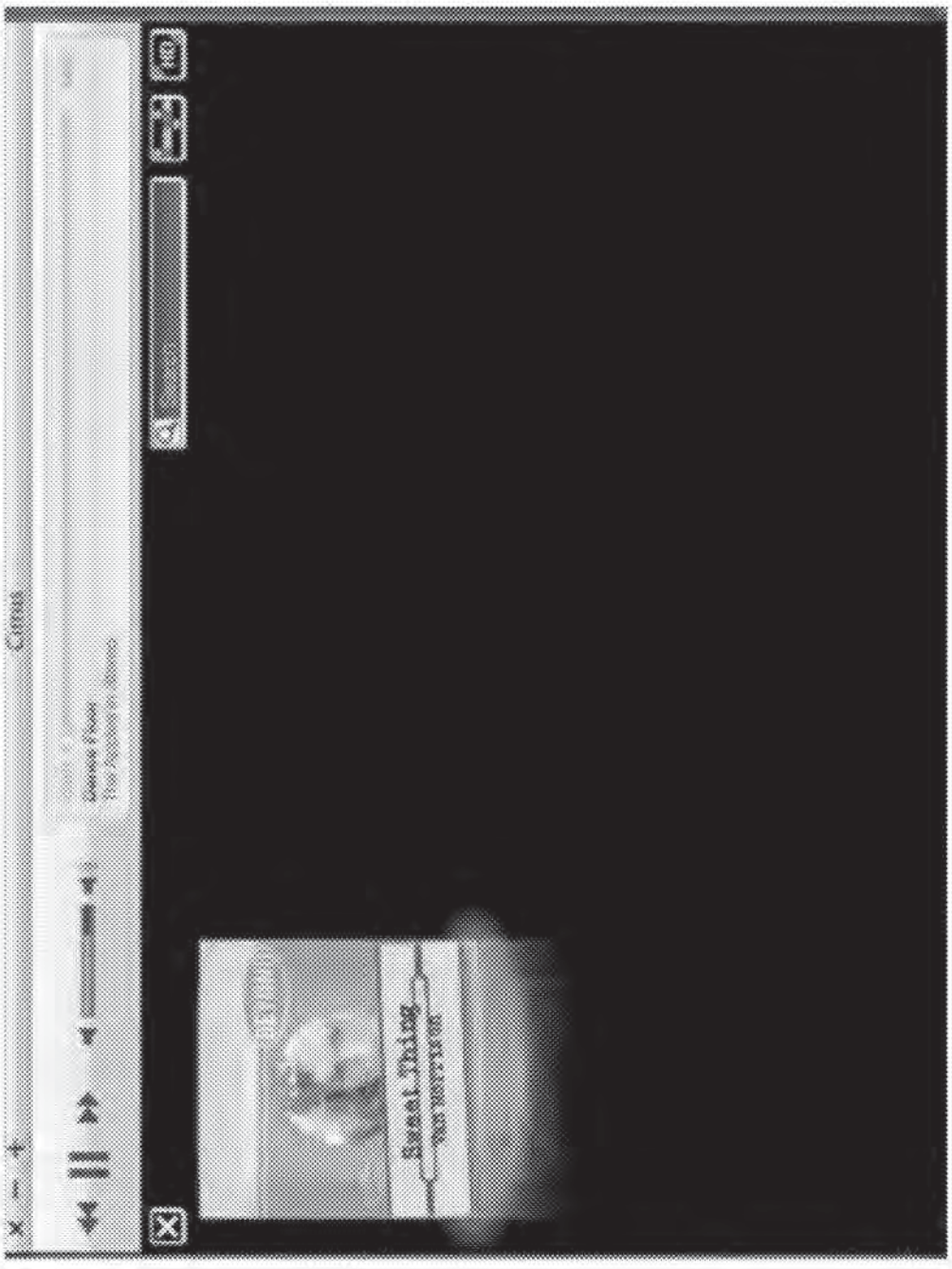


FIGURE 15b

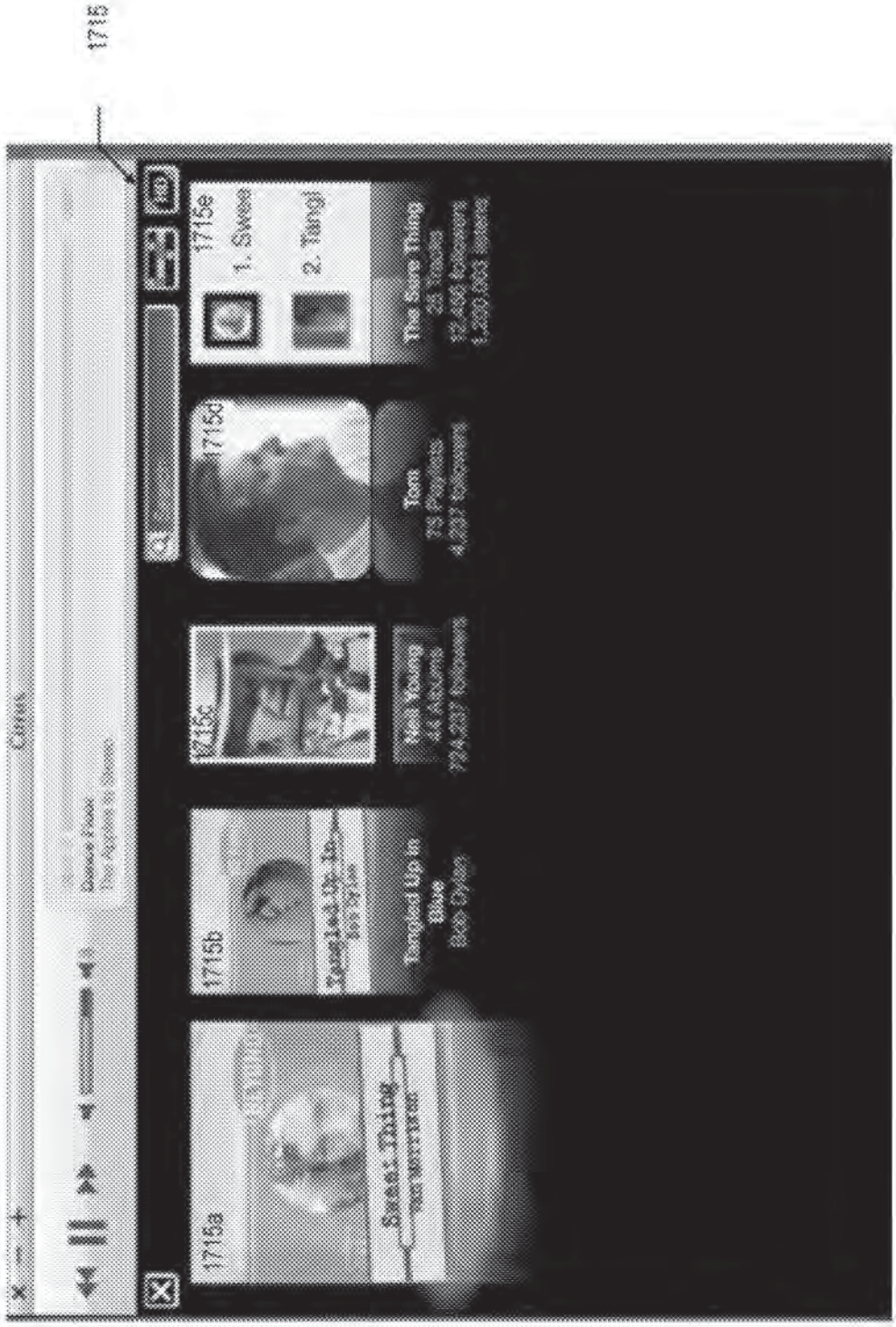

1715
1715a
1715b
1715c
1715d
1715e
1. Sweet
2. Tangl
Tangled Up in Blue
Bob Dylan
Neil Young
Tom
The Same Thing

FIGURE 15c


1720
1720a
1720b
1720c
1720d
1720e
1720f
Sweet Thing
Tangled Up in Blue
Astral Weeks
Best of Van Morrison
Appears on:

FIGURE 15d


1730
1730f
1730e
1730b
1730a
Speaking in Tongues
Talking Heads
Tracks:
Burning Down the House
Making Flippy Floppy
Girlfriend Is Better
Slippery People
I Get Wild/Wild Gravity
Swamp
Moon Rocks
Pull Up the Roots
This Must Be The Place
(Naive Melody)

FIGURE 15e


Still Feel Gone
Brian Eno
Michael
The Bare Thing
1. Swee
2. Tangl
Kicking Television: Live in Chicago
Tracks:
1. Misunderstood
2. Company In My Back
3. The Late Greats
4. Hell Is Chrome
5. Handshake Drugs
6. I Am Trying To Break Your Heart
7. Stuck In The Arms
8. At Least That's What You Said

FIGURE 15f

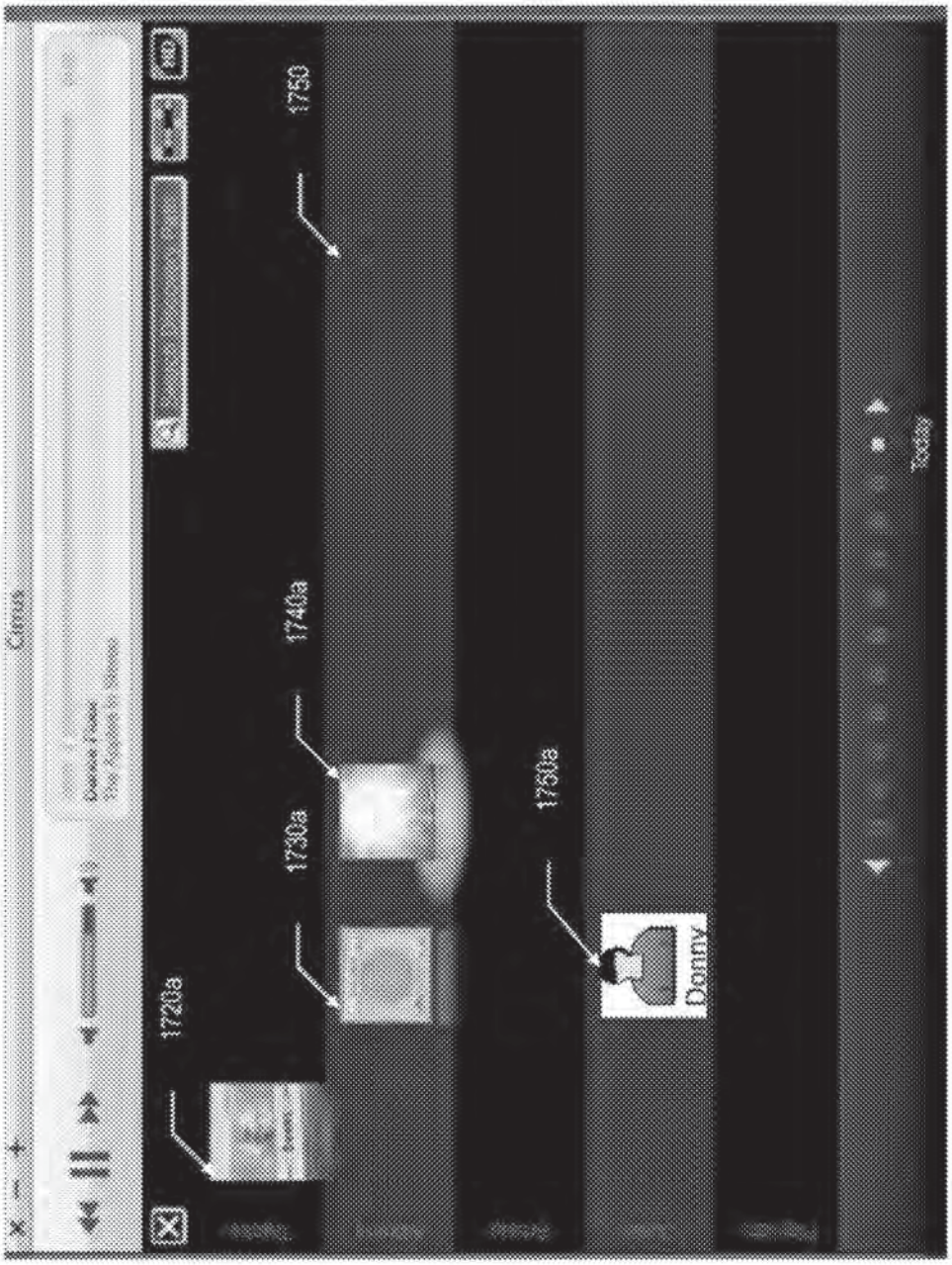


FIGURE 15g

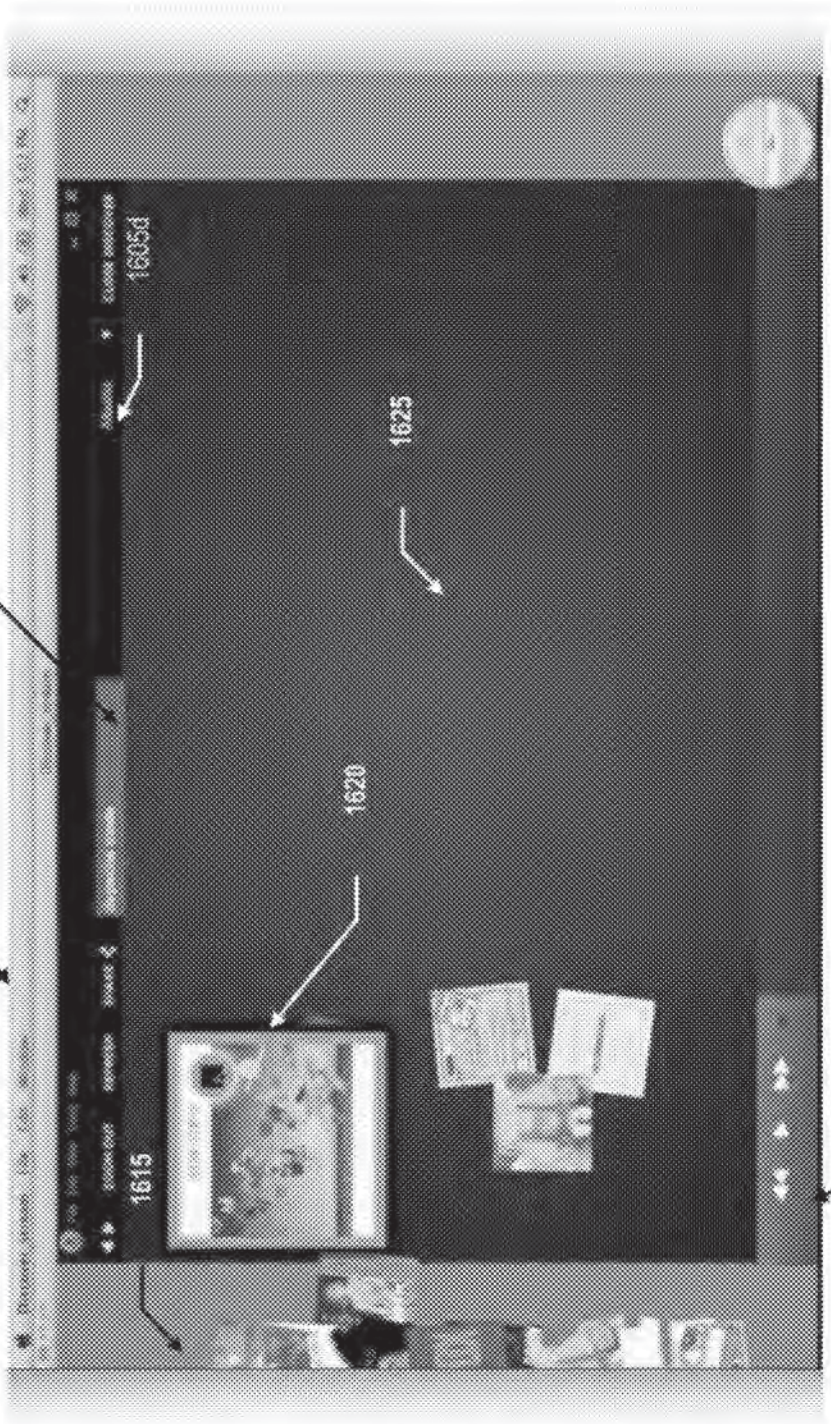


FIGURE 16a

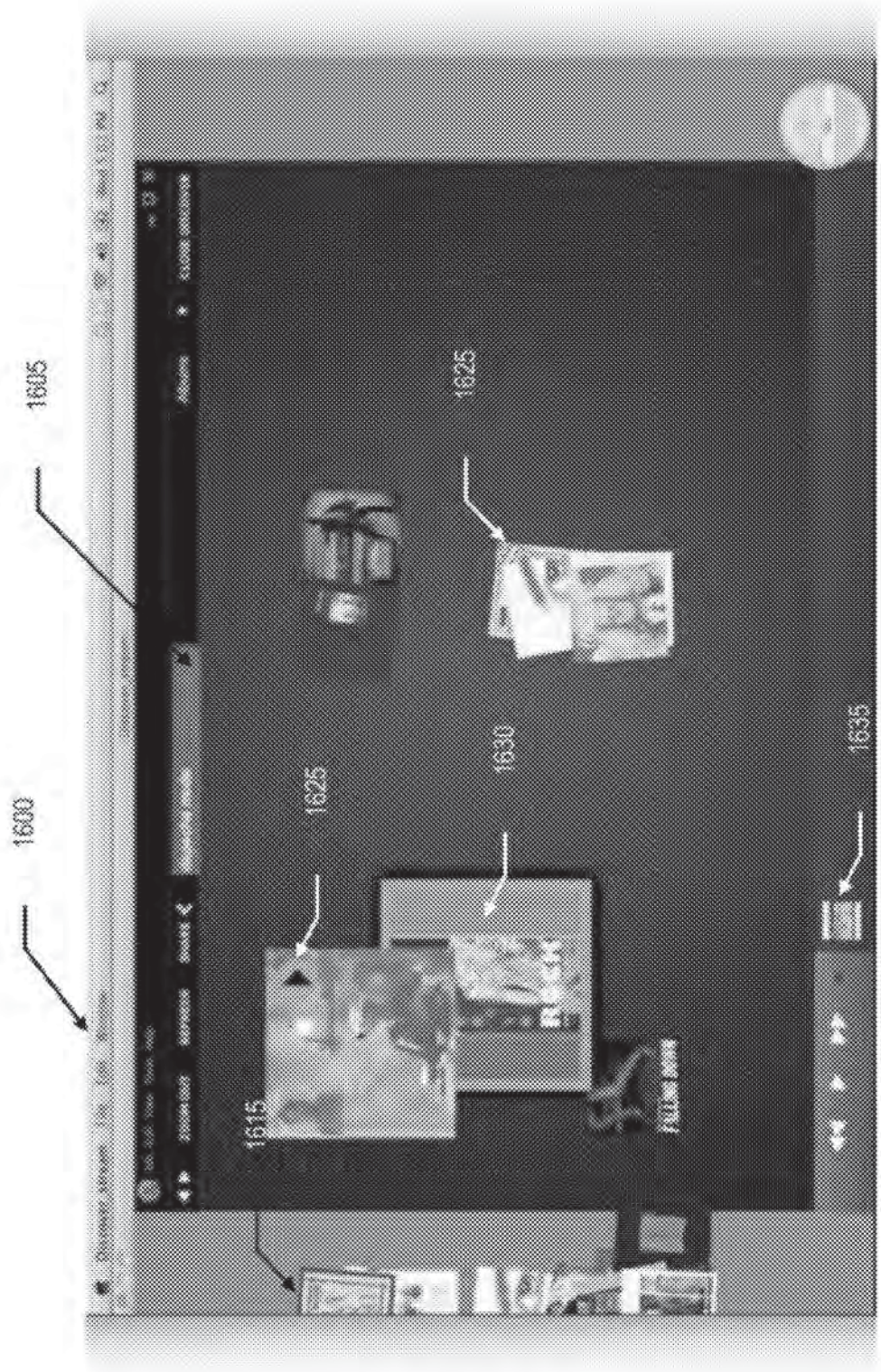


FIGURE 16b

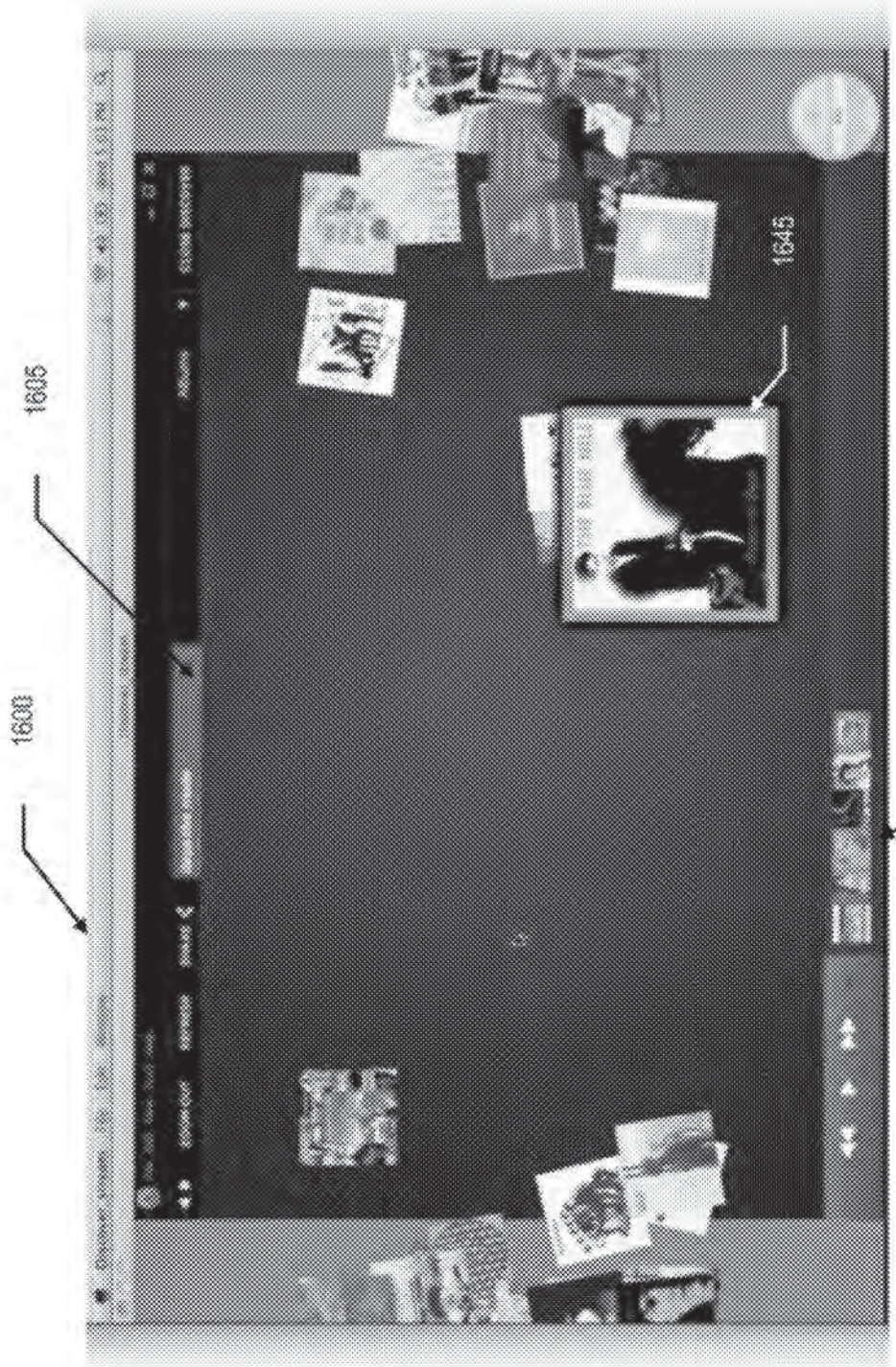


FIGURE 16c

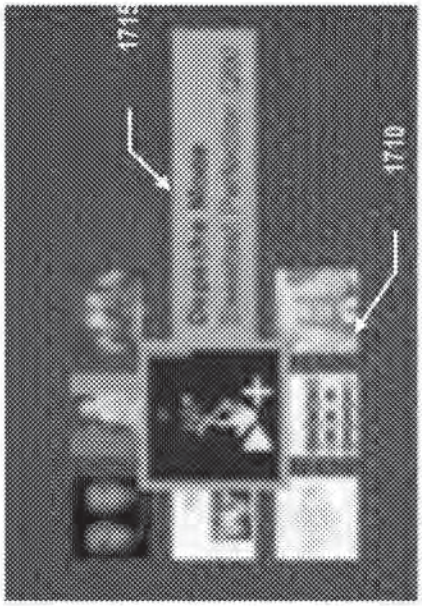


FIGURE 17a

1710a

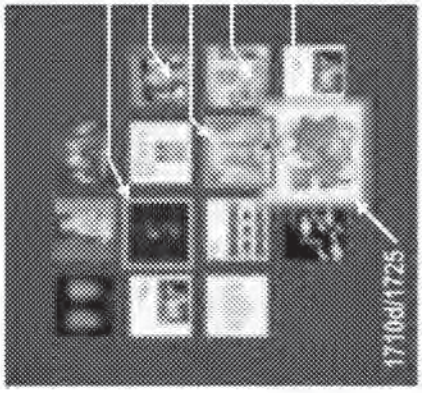


FIGURE 17b

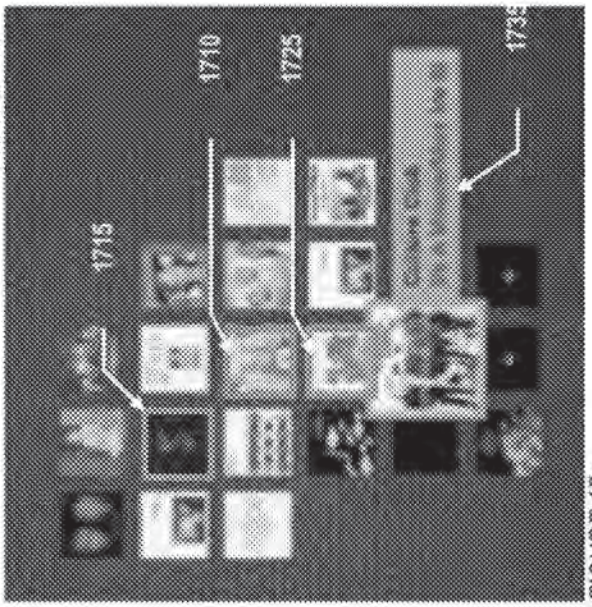


FIGURE 17c

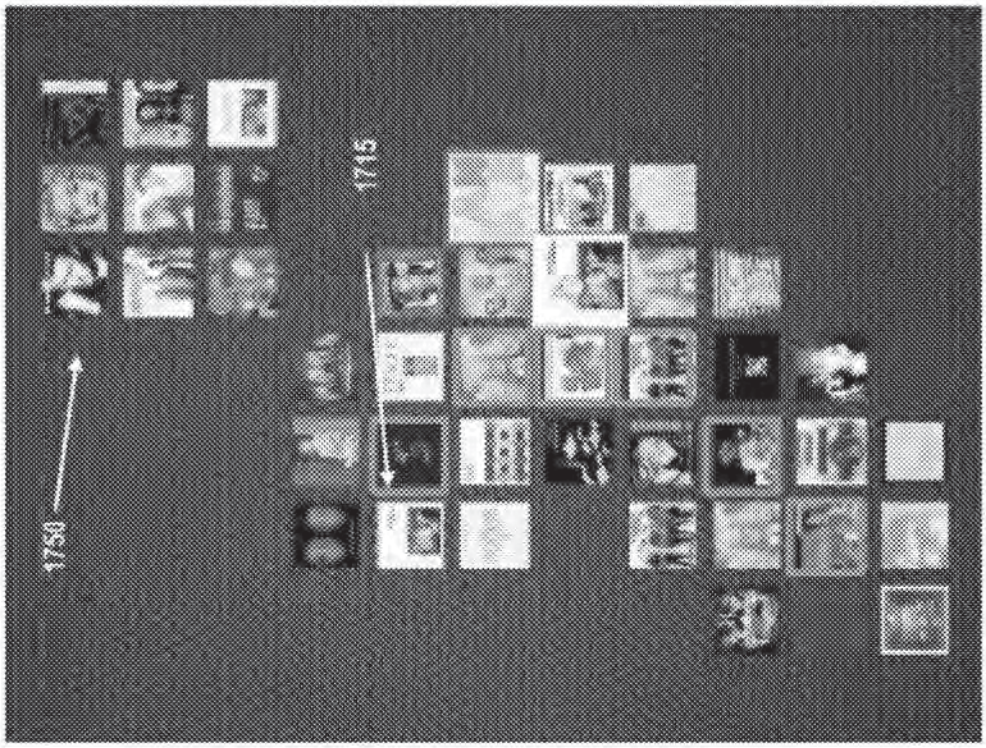


FIGURE 17e

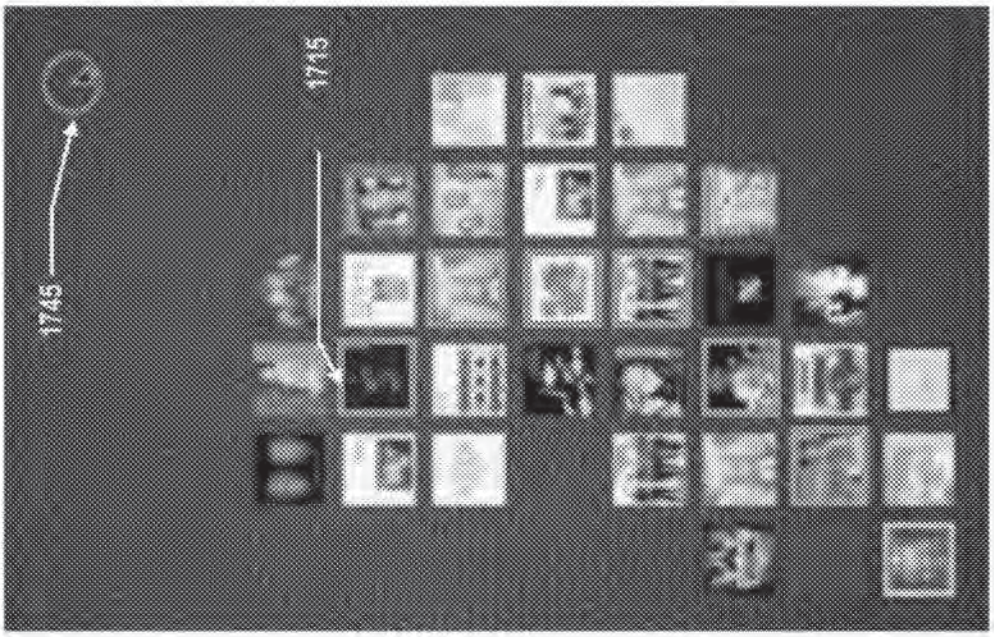


FIGURE 17d

FIGURE 17f

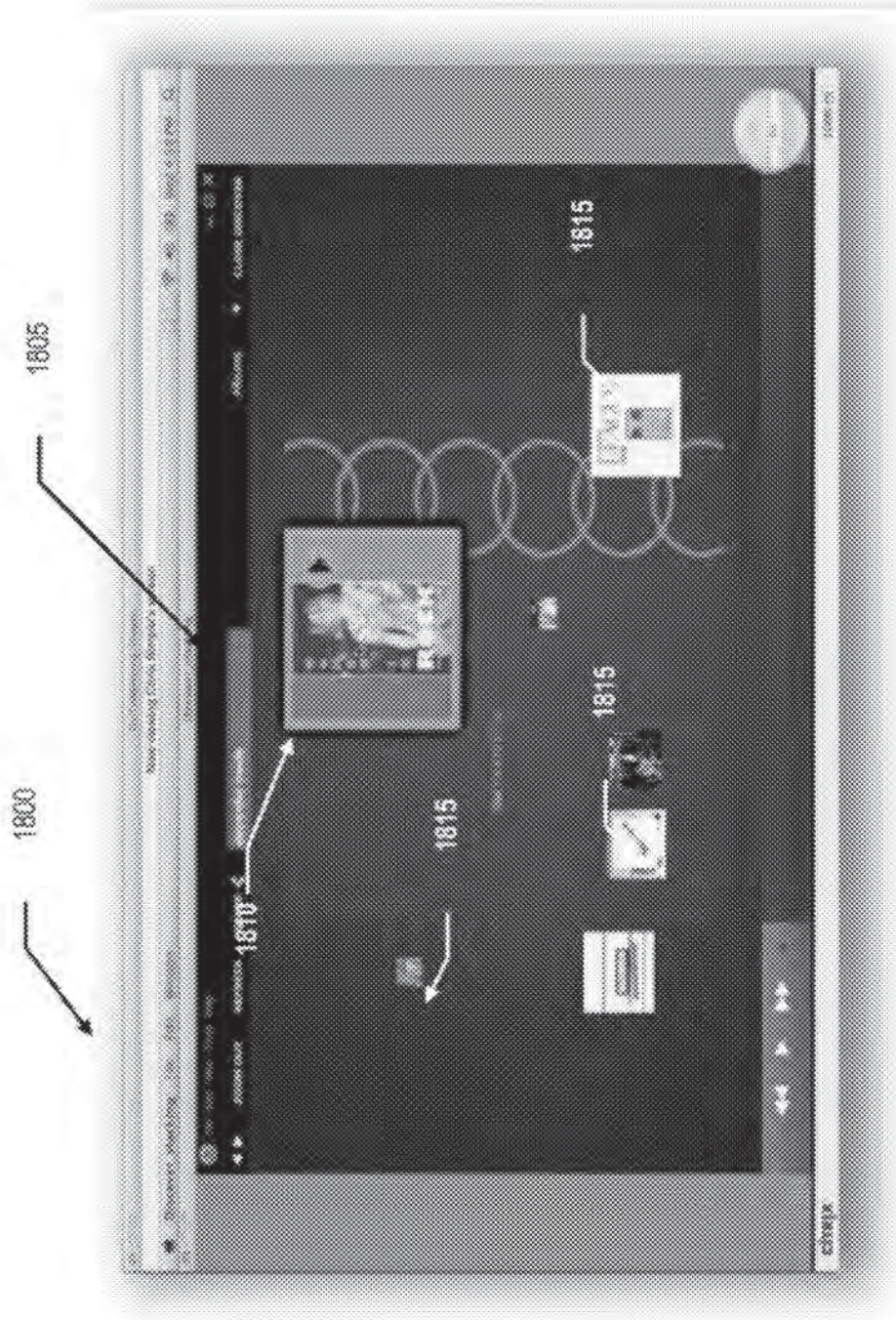


FIGURE 18a

FIGURE 18b

FIGURE 18c

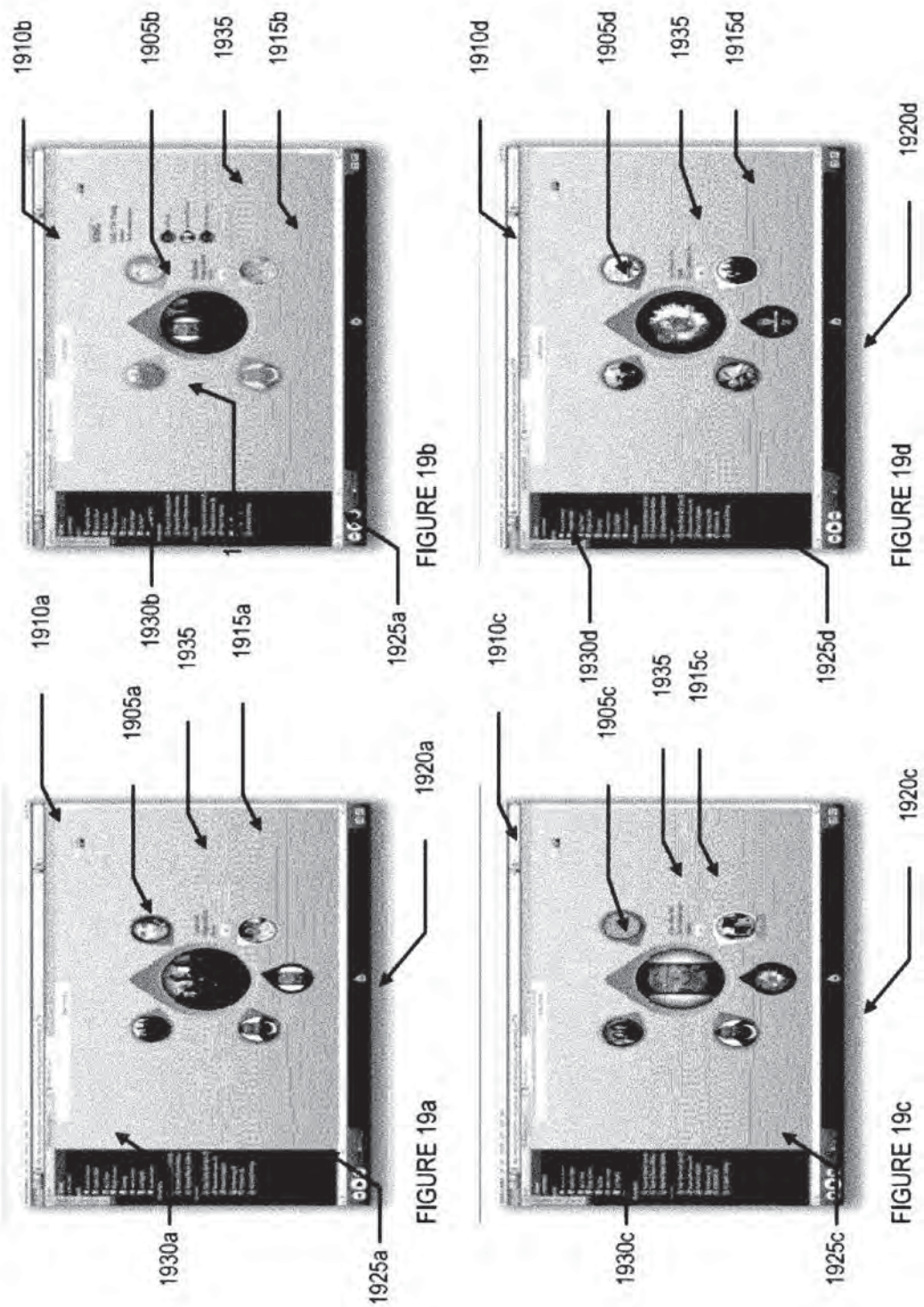


FIGURE 19b

FIGURE 19a

FIGURE 19d

FIGURE 19c

FIGURE 19f

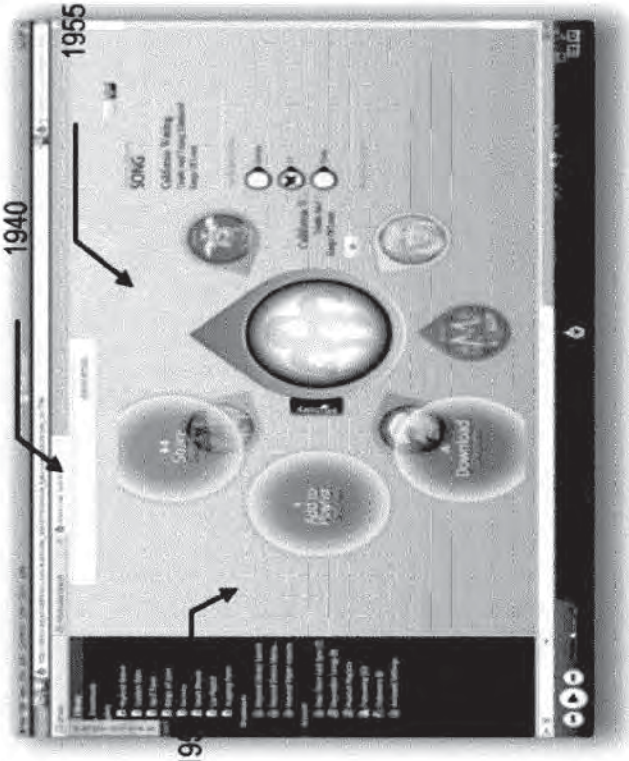


FIGURE 19e

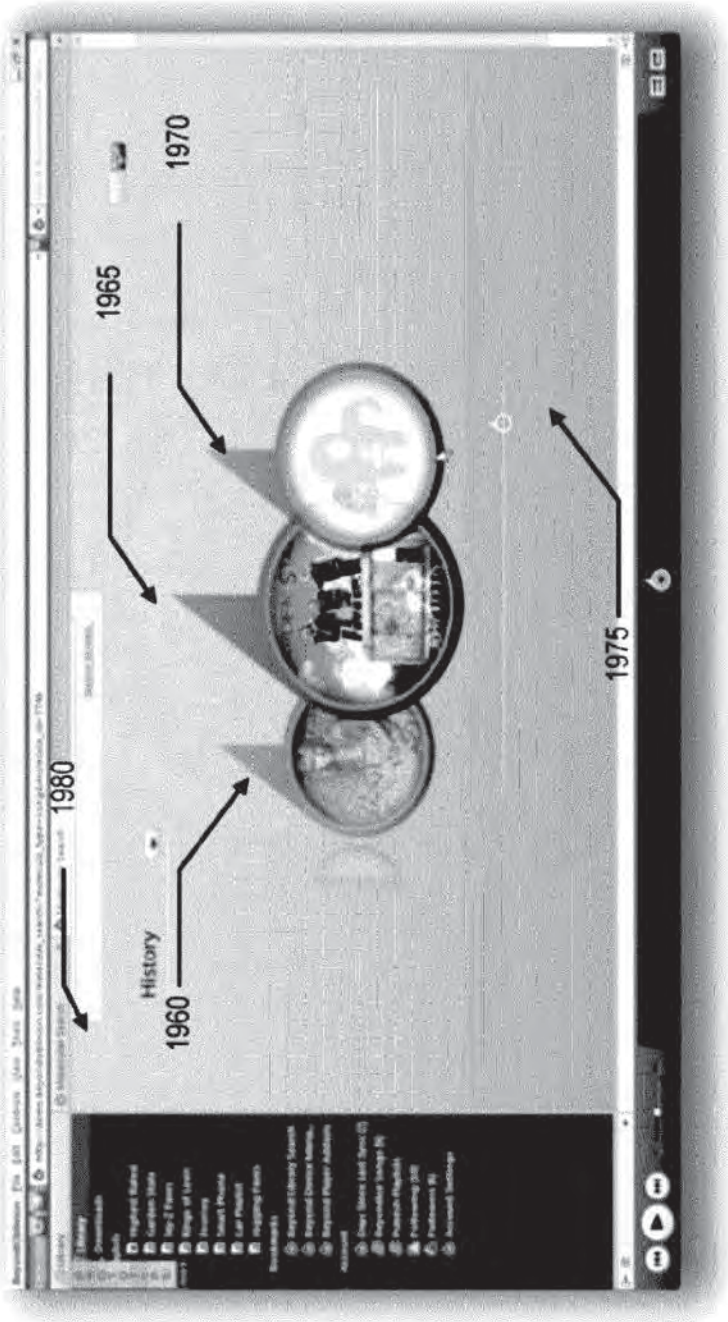


FIGURE 19g

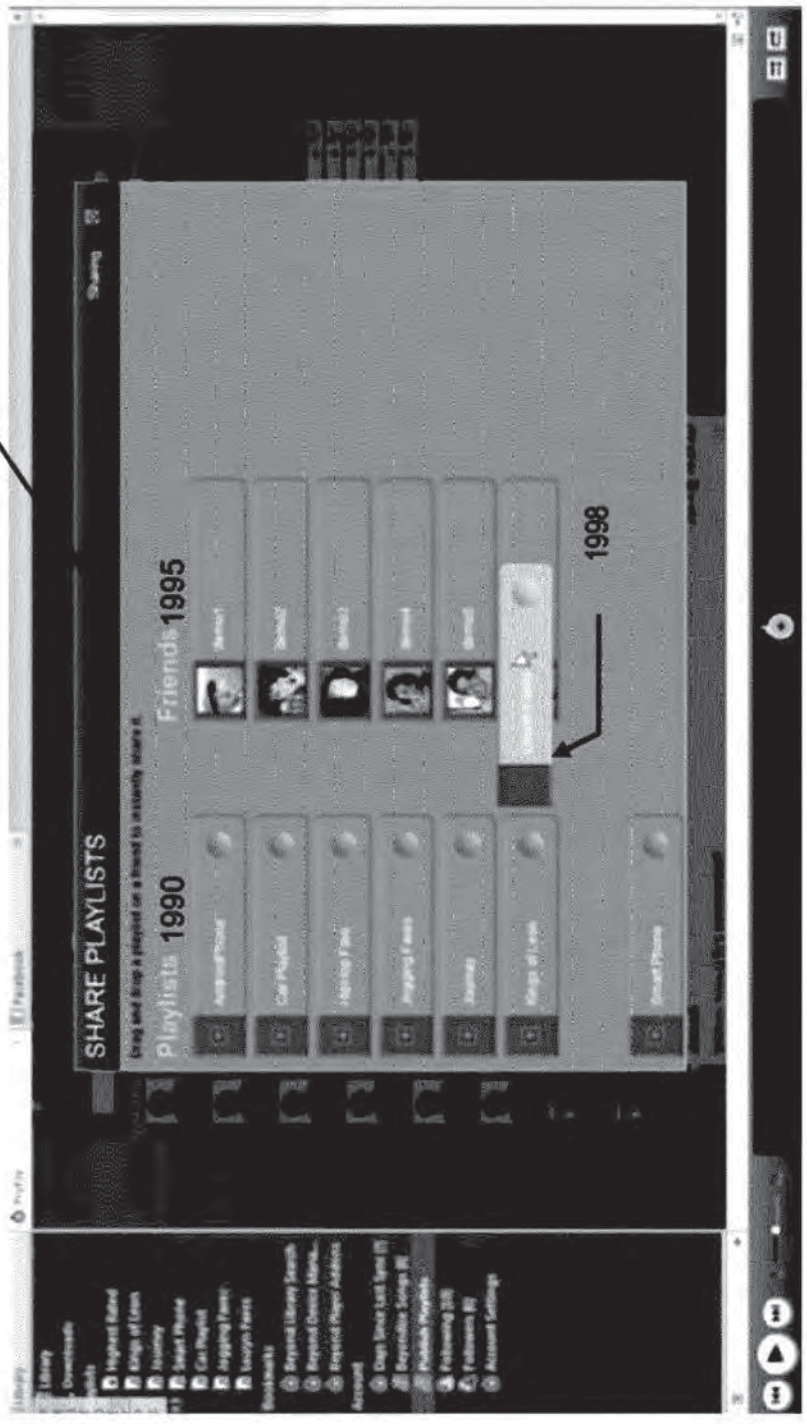


FIGURE 19h

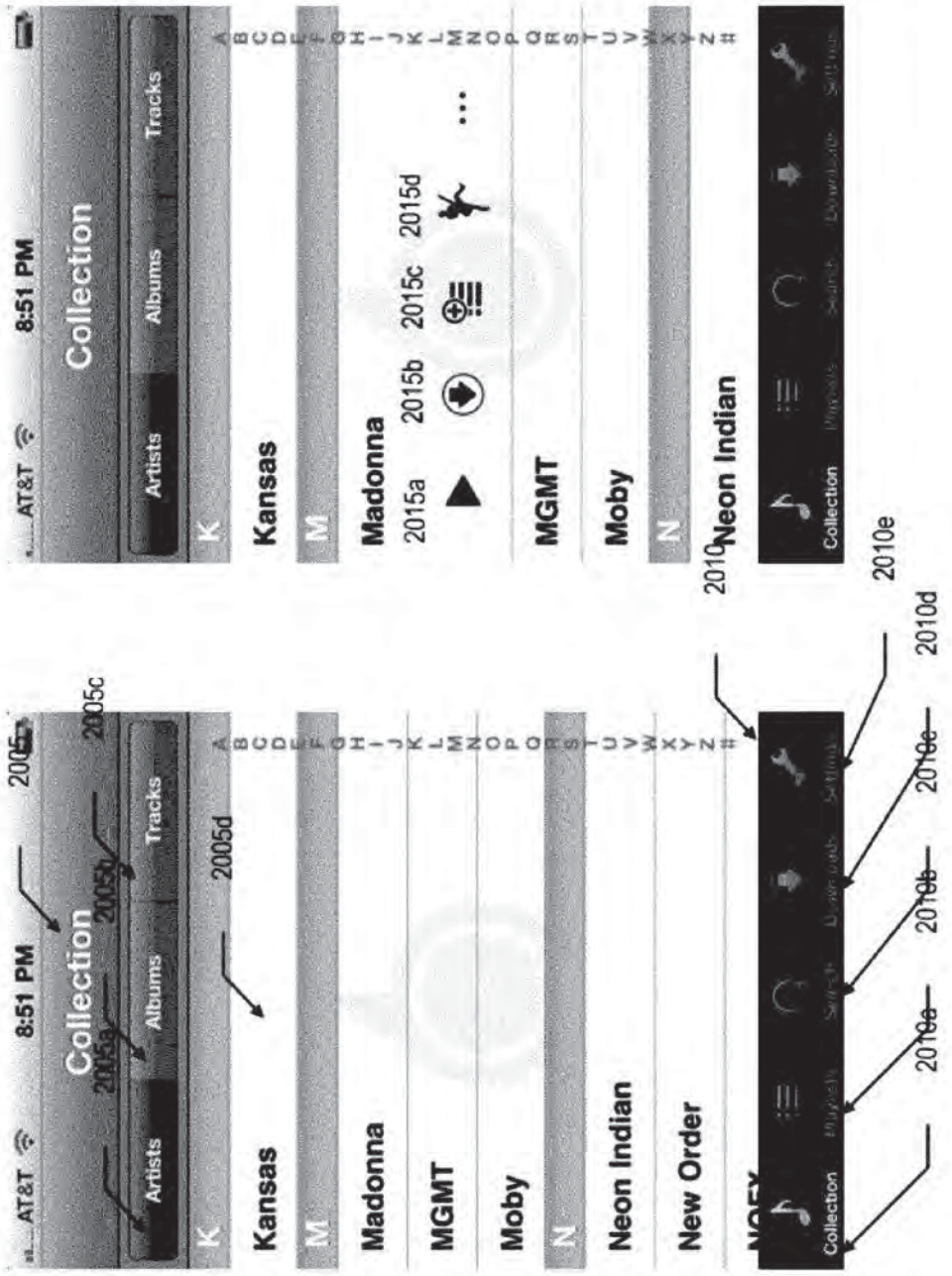


FIGURE 20a

FIGURE 20b

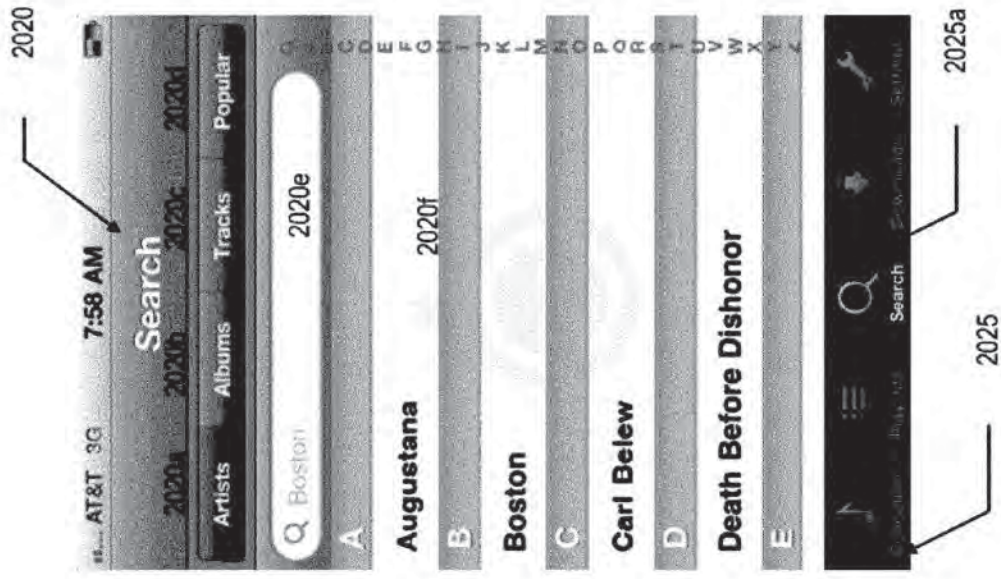


FIGURE 20c

FIGURE 20d

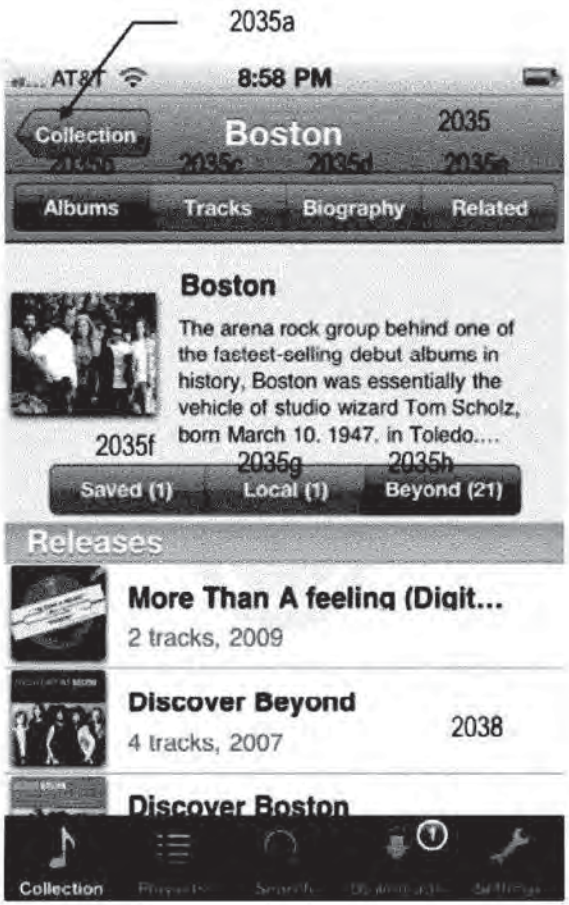


FIGURE 20e

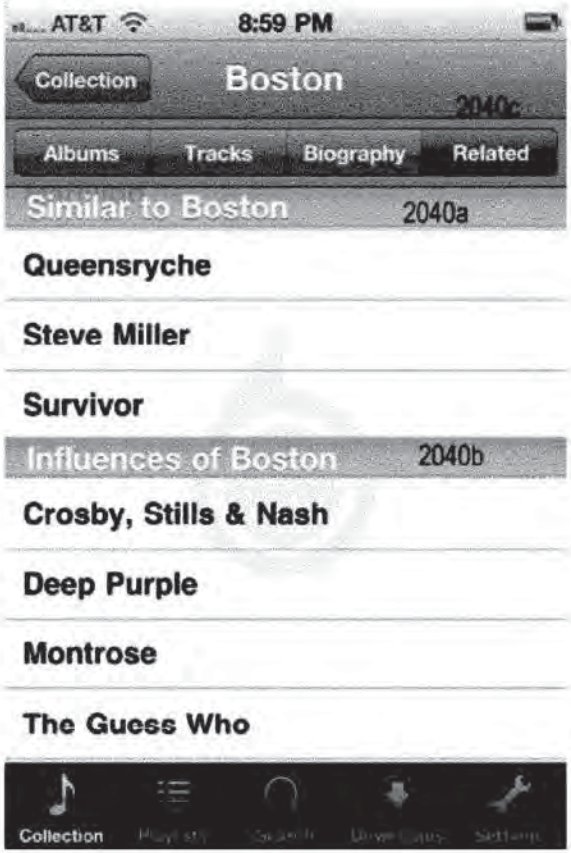


FIGURE 20f

FIGURE 20g

FIGURE 20h

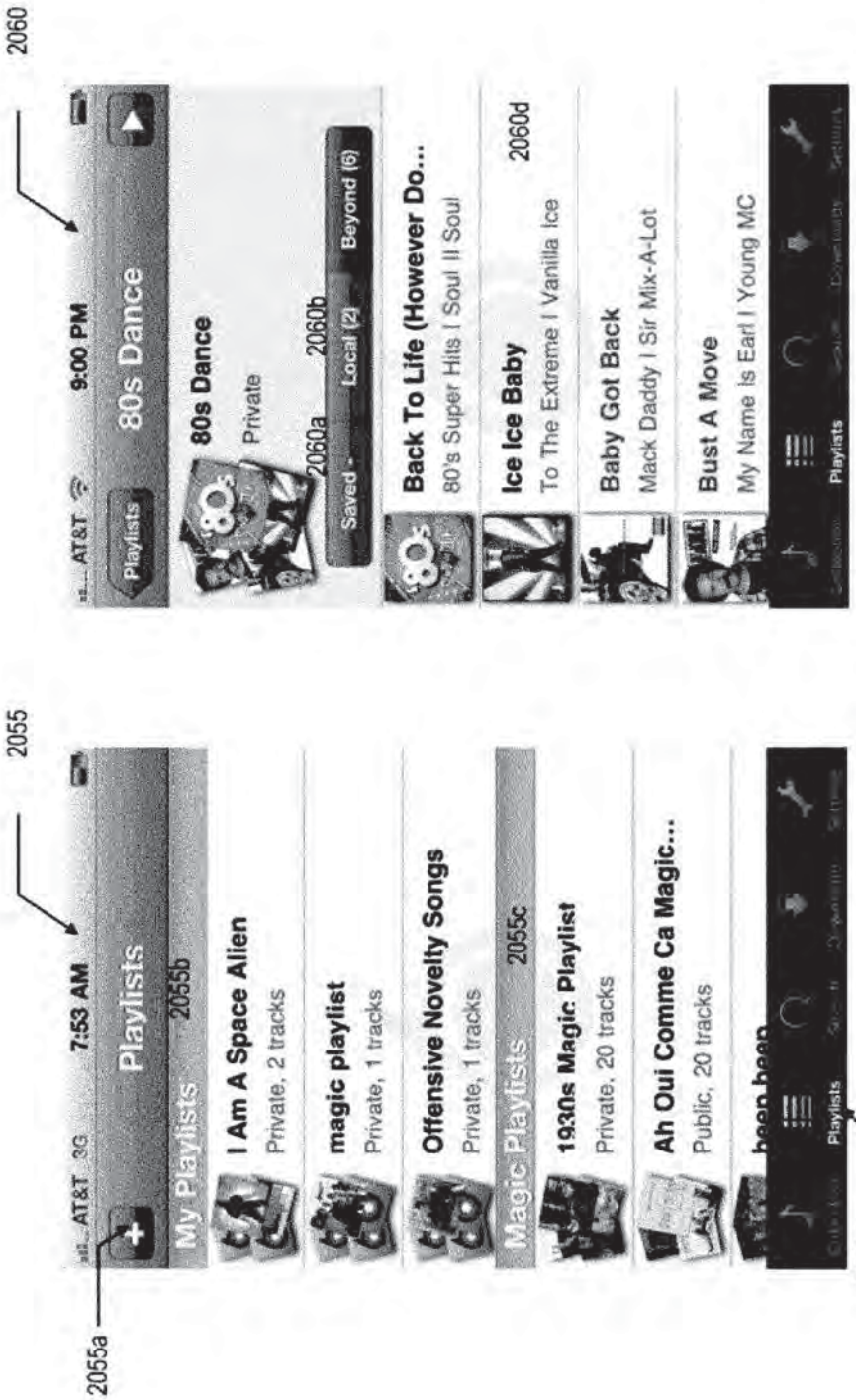


FIGURE 20i

FIGURE 20j

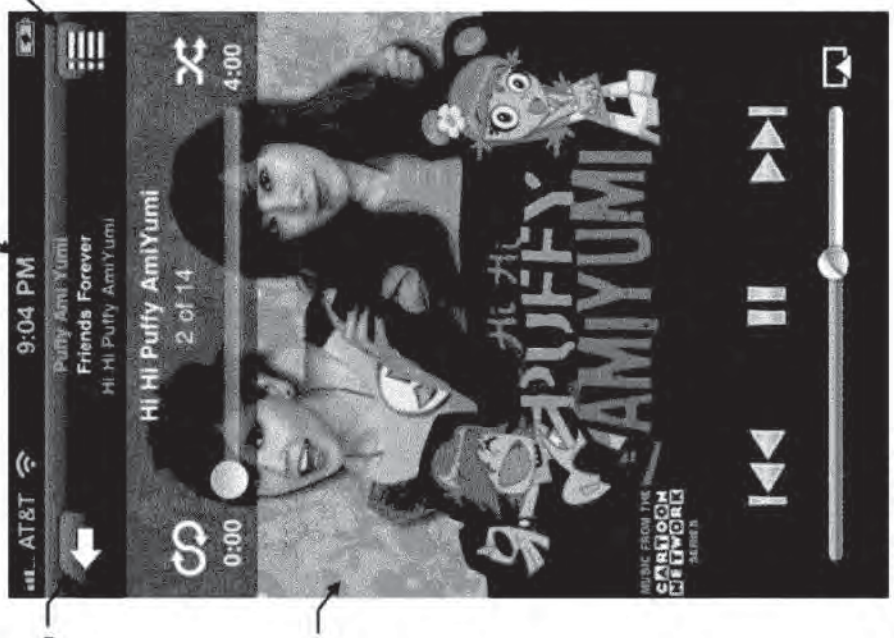


FIGURE 20k

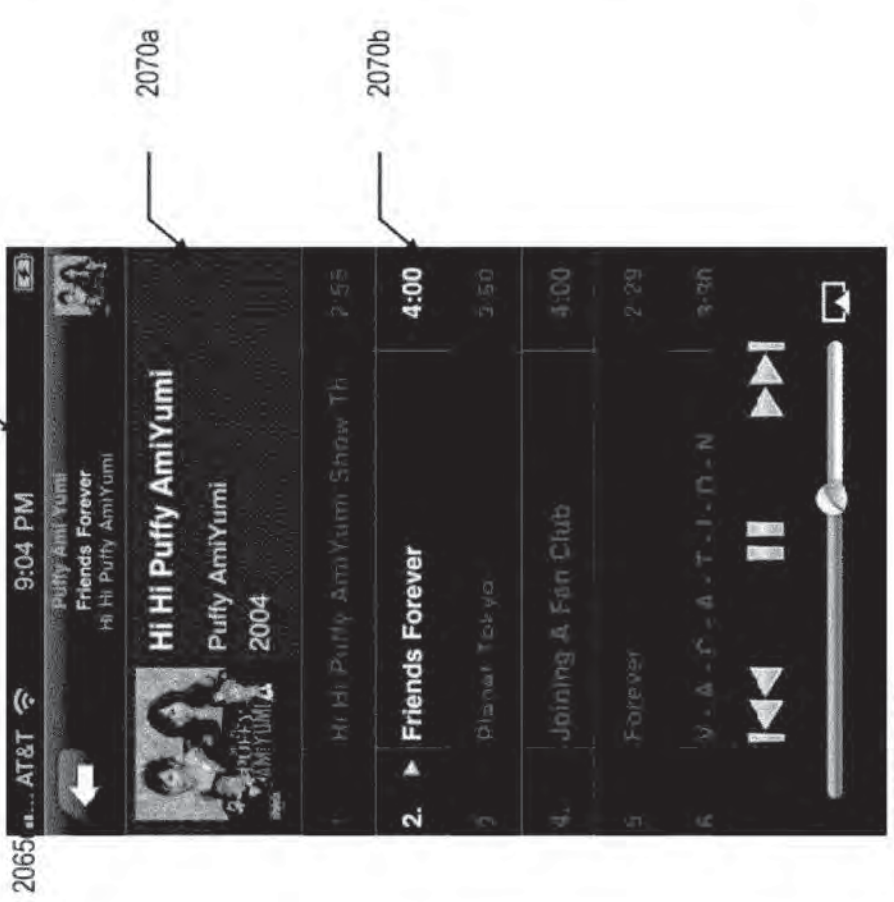
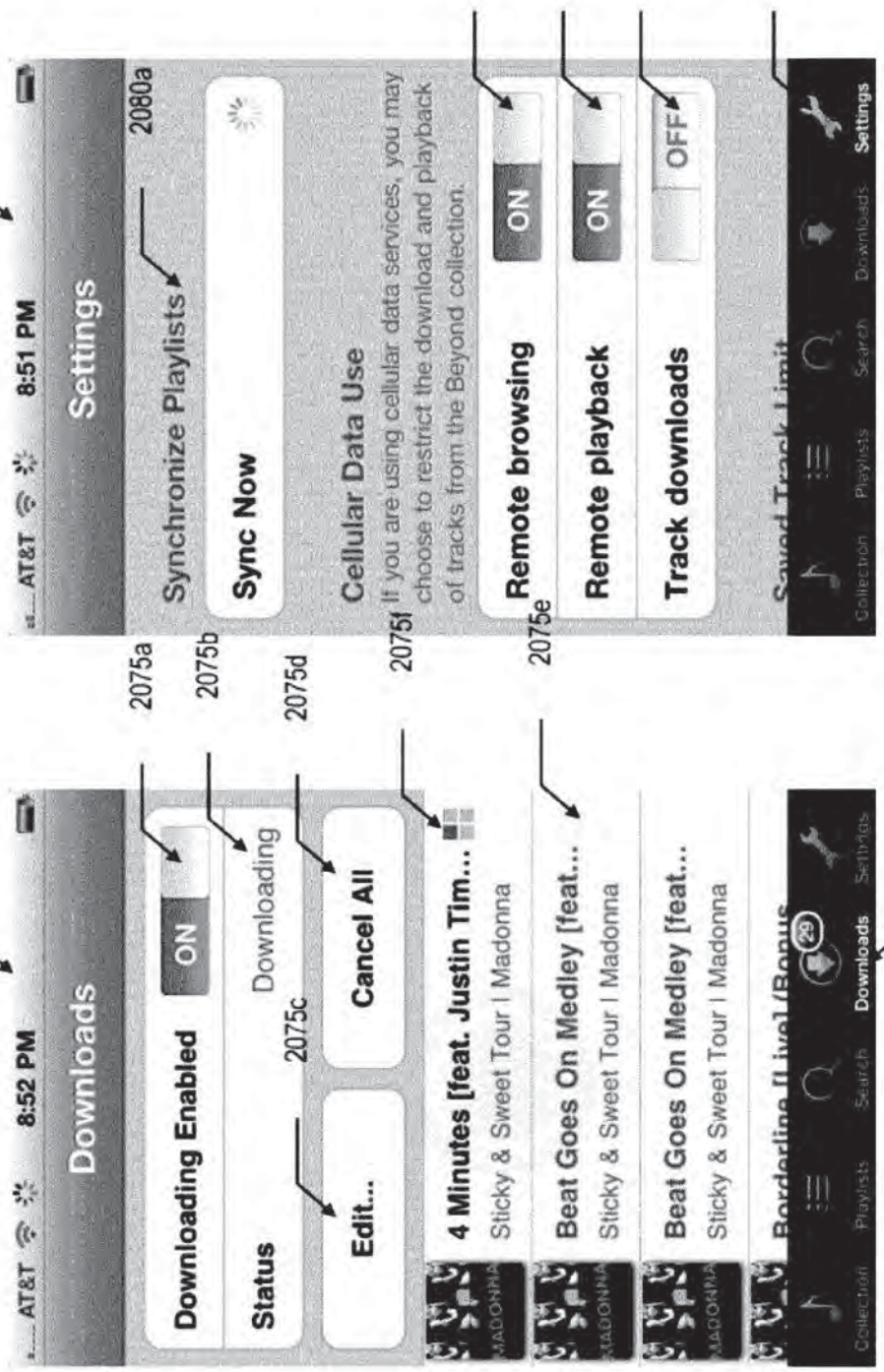


FIGURE 20l


2075
AT&T 8:52 PM
Downloads
Downloading Enabled
ON
Status
Downloading
Edit...
Cancel All
4 Minutes [feat. Justin Tim...
Sticky & Sweet Tour | Madonna
Beat Goes On Medley [feat...
Sticky & Sweet Tour | Madonna
Beat Goes On Medley [feat...
Sticky & Sweet Tour | Madonna
Collection
Playlists
Search
Downloads
Settings
2075a
2075b
2075c
2075d
2075g
FIGURE 20m
2080
AT&T 8:51 PM
Settings
Synchronize Playlists
Sync Now
Cellular Data Use
If you are using cellular data services, you may choose to restrict the download and playback of tracks from the Beyond collection.
Remote browsing
ON
Remote playback
ON
Track downloads
OFF
Collection
Playlists
Search
Downloads
Settings
2080a
2075f
2075e
2080b
2080c
2080d
2080e
FIGURE 20n

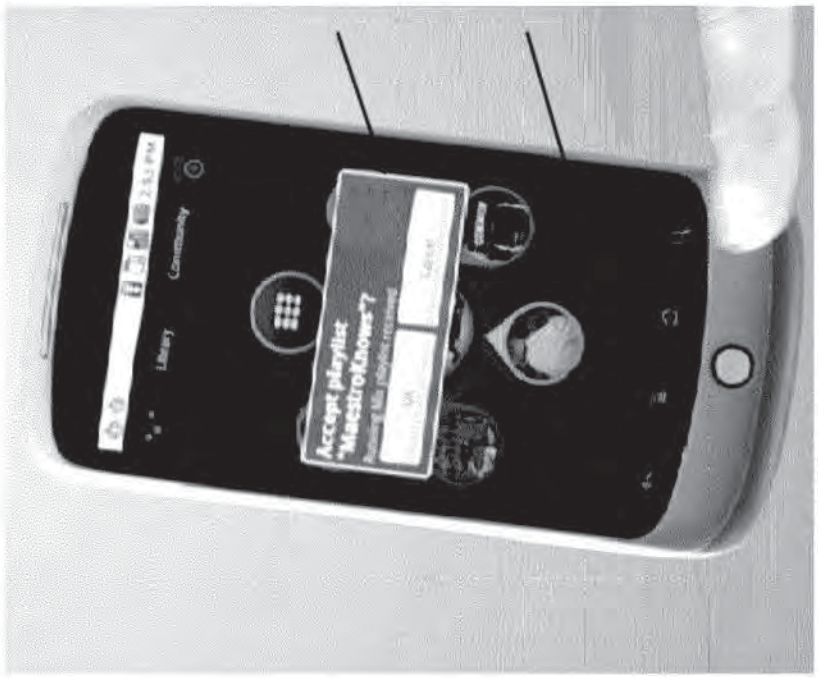


FIGURE 21b

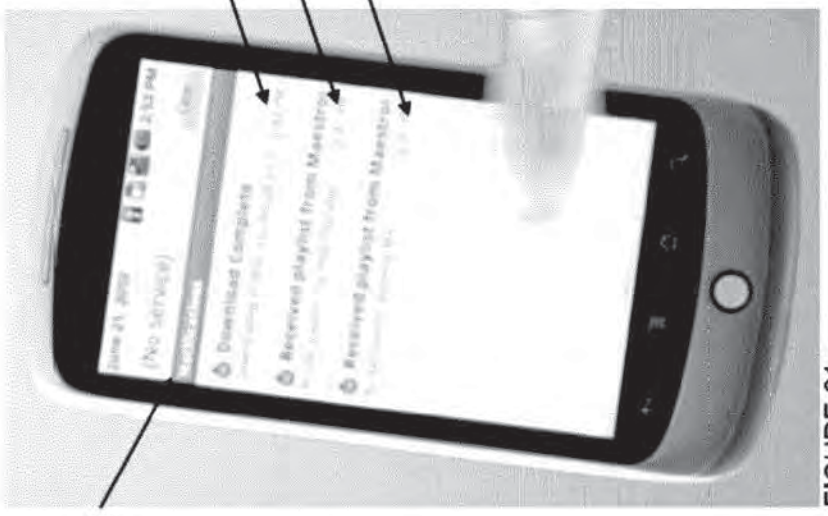


FIGURE 21a

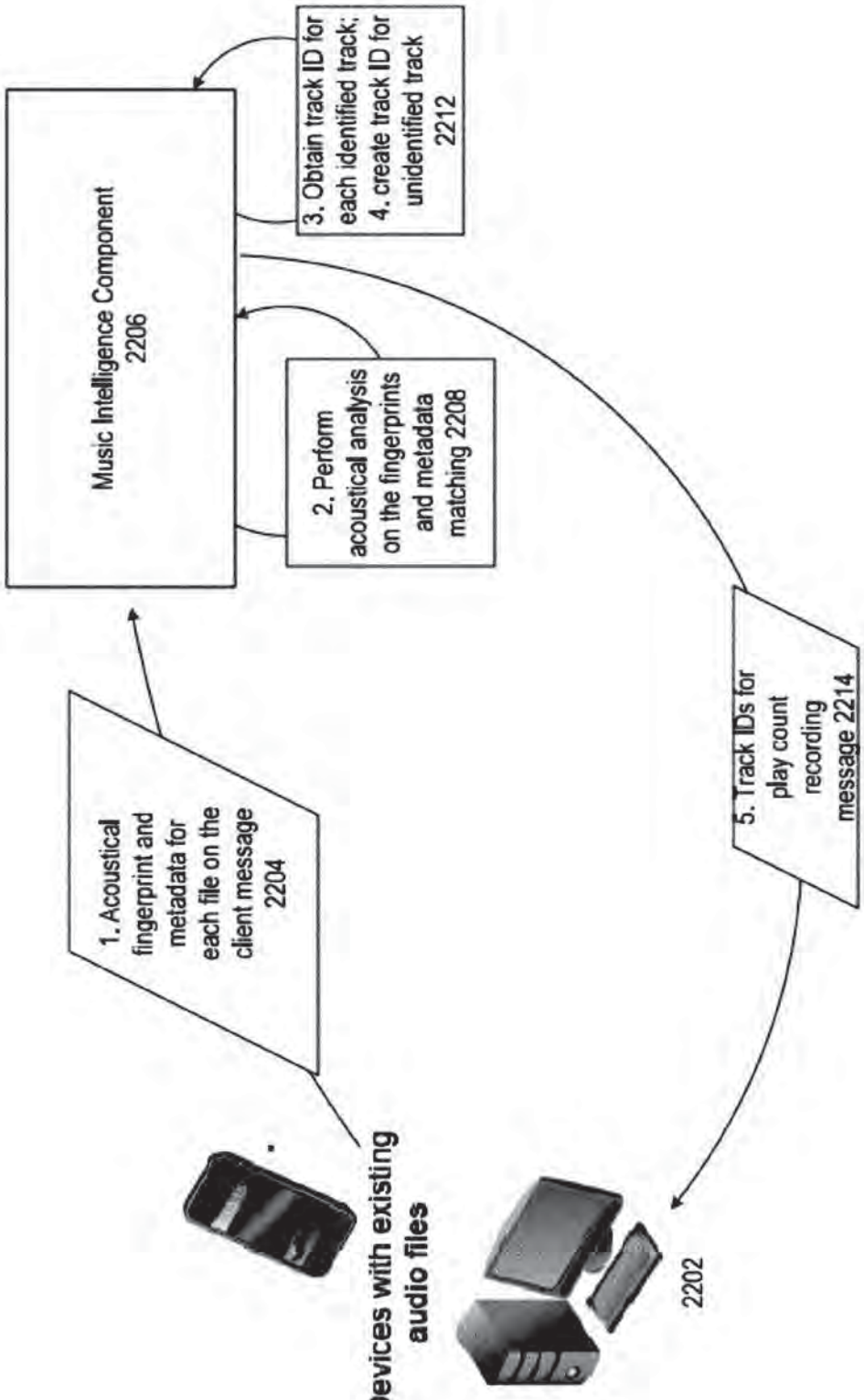

Music Intelligence Component
2206
3. Obtain track ID for each identified track; 4. create track ID for unidentified track
2212
2. Perform acoustical analysis on the fingerprints and metadata matching 2208
1. Acoustical fingerprint and metadata for each file on the client message
2204
5. Track IDs for play count recording message 2214
Devices with existing audio files
2202

FIGURE 22a

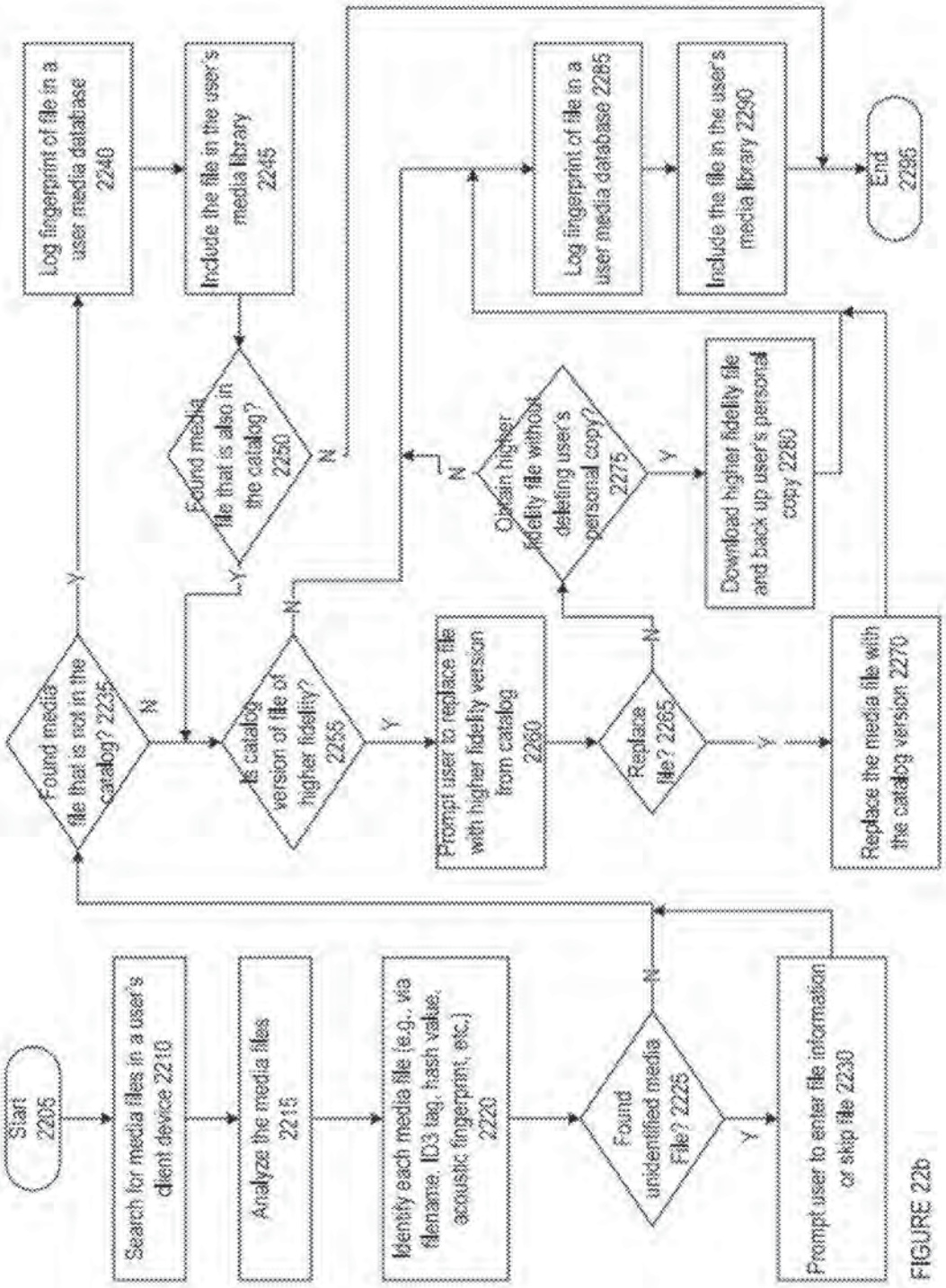

Start 2205
Search for media files in a user's client device 2210
Analyze the media files 2215
Identify each media file (e.g., via filename, ID3 tag, hash value, acoustic fingerprint, etc.) 2220
Found unidentified media File? 2225
Prompt user to enter file information or skip file 2230
Found media file that is not in the catalog? 2235
Is catalog version of file of higher fidelity? 2255
Prompt user to replace file with higher fidelity version from catalog 2260
Replace file? 2265
Replace the media file with the catalog version 2270
Obtain higher fidelity file without deleting user's personal copy? 2275
Download higher fidelity file and back up user's personal copy 2280
Log fingerprint of file in a user media database 2285
Include the file in the user's media library 2290
Found media file that is also in the catalog? 2250
Log fingerprint of file in a user media database 2240
Include the file in the user's media library 2245
End 2295
Y
N

FIGURE 22b

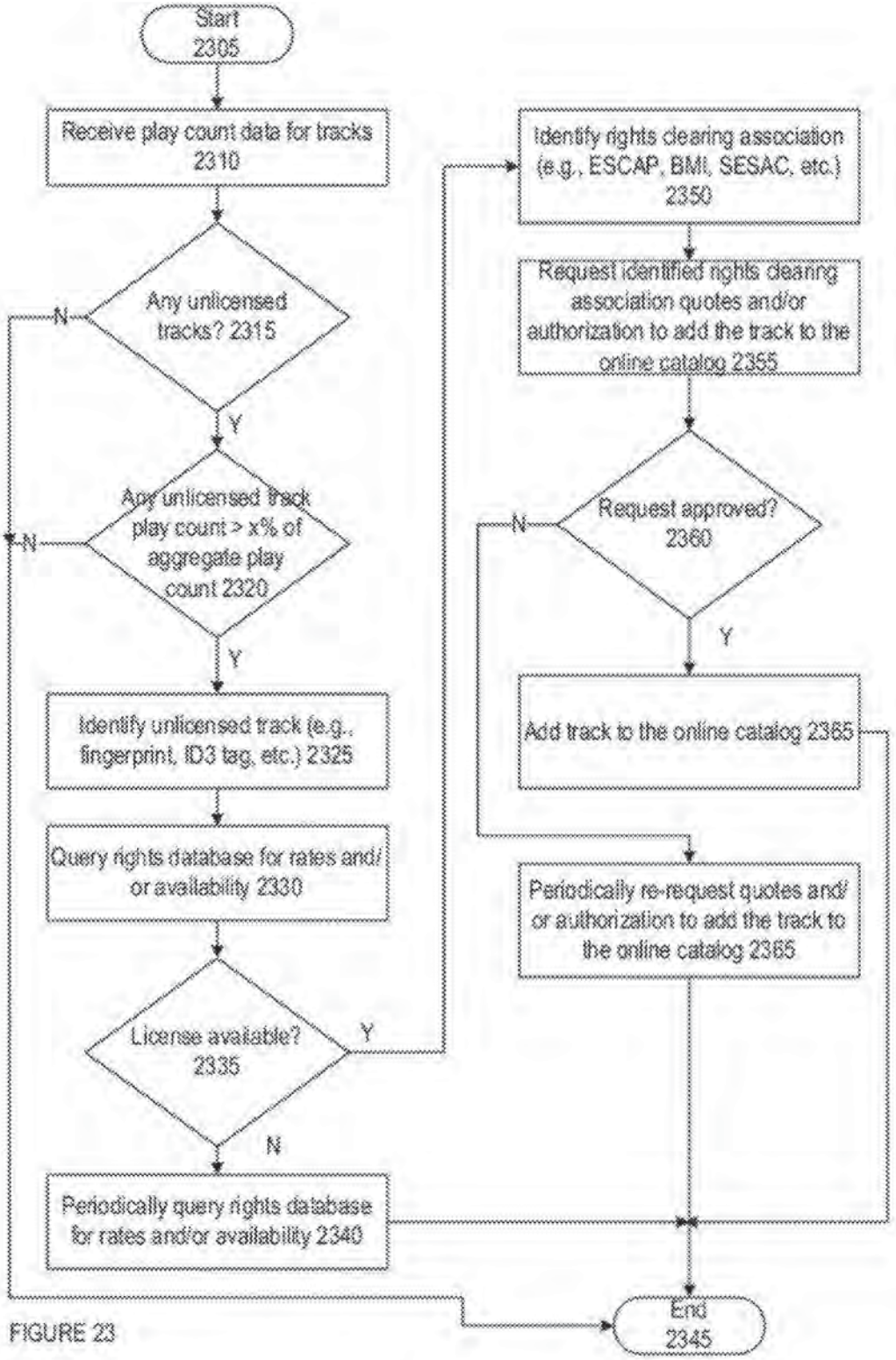

Start 2305
Receive play count data for tracks 2310
Any unlicensed tracks? 2315
N
Y
Any unlicensed track play count > x% of aggregate play count 2320
N
Y
Identify unlicensed track (e.g., fingerprint, ID3 tag, etc.) 2325
Query rights database for rates and/or availability 2330
License available? 2335
Y
N
Periodically query rights database for rates and/or availability 2340
Identify rights clearing association (e.g., ESCAP, BMI, SESAC, etc.) 2350
Request identified rights clearing association quotes and/or authorization to add the track to the online catalog 2355
Request approved? 2360
N
Y
Add track to the online catalog 2365
Periodically re-request quotes and/or authorization to add the track to the online catalog 2365
End 2345

FIGURE 23

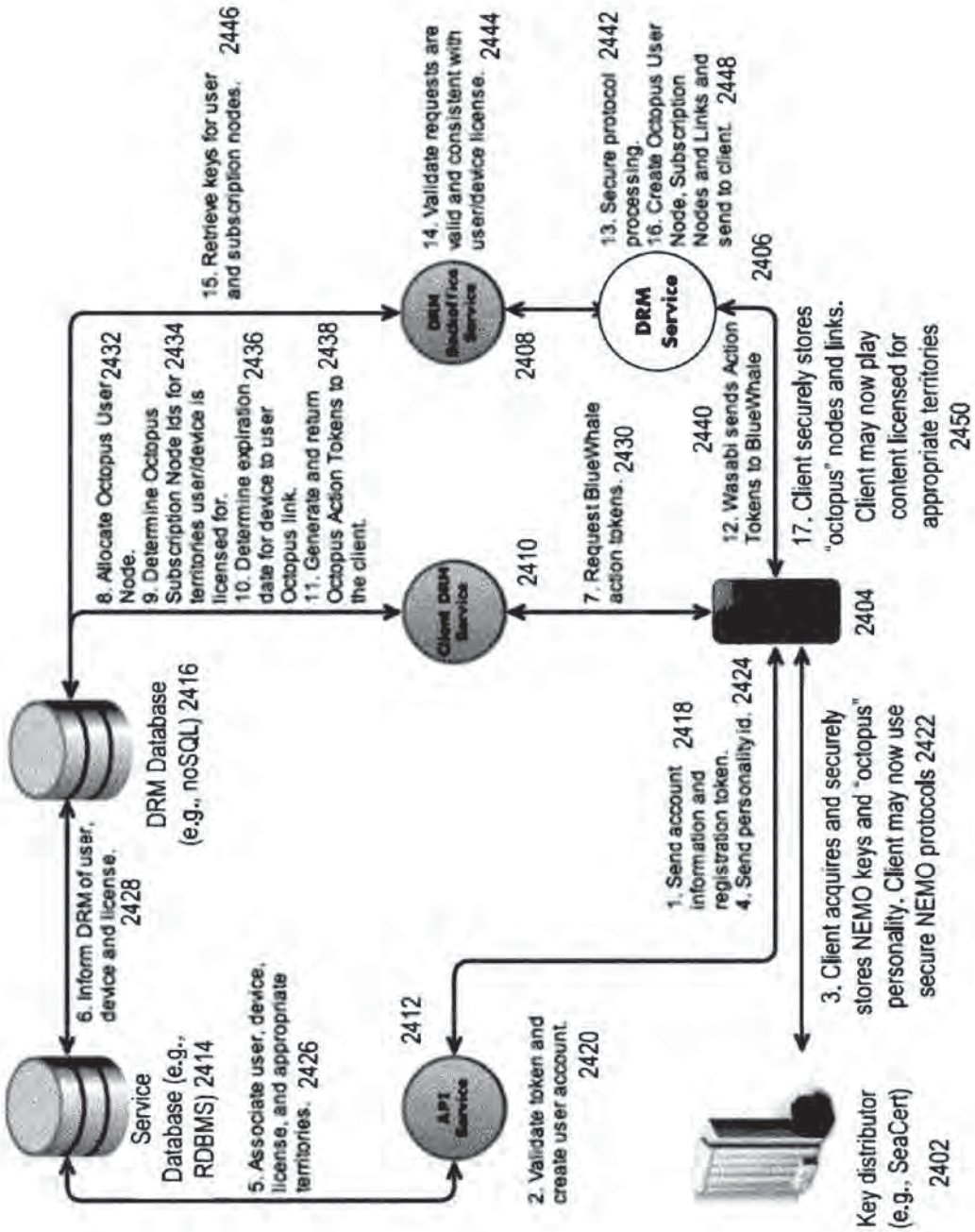

Service Database (e.g., RDBMS) 2414
DRM Database (e.g., noSQL) 2416
Key distributor (e.g., SeaCert) 2402
API Service
2412
Client DRM Service
2410
DRM Backoffice Service
2408
DRM Service
2406
2404
6. Inform DRM of user, device and license. 2428
5. Associate user, device, license, and appropriate territories. 2426
2. Validate token and create user account. 2420
1. Send account information and registration token. 2418
4. Send personality id. 2424
3. Client acquires and securely stores NEMO keys and "octopus" personality. Client may now use secure NEMO protocols 2422
8. Allocate Octopus User 2432 Node.
9. Determine Octopus Subscription Node Ids for 2434 territories user/device is licensed for.
10. Determine expiration 2436 date for device to user Octopus link.
11. Generate and return Octopus Action Tokens to 2438 the client.
15. Retrieve keys for user and subscription nodes. 2446
14. Validate requests are valid and consistent with user/device license. 2444
7. Request BlueWhale action tokens. 2430
2440
12. Wasabi sends Action Tokens to BlueWhale
13. Secure protocol 2442 processing.
16. Create Octopus User Node, Subscription Nodes and Links and send to client. 2448
17. Client securely stores "octopus" nodes and links. Client may now play content licensed for appropriate territories 2450

FIGURE 24a

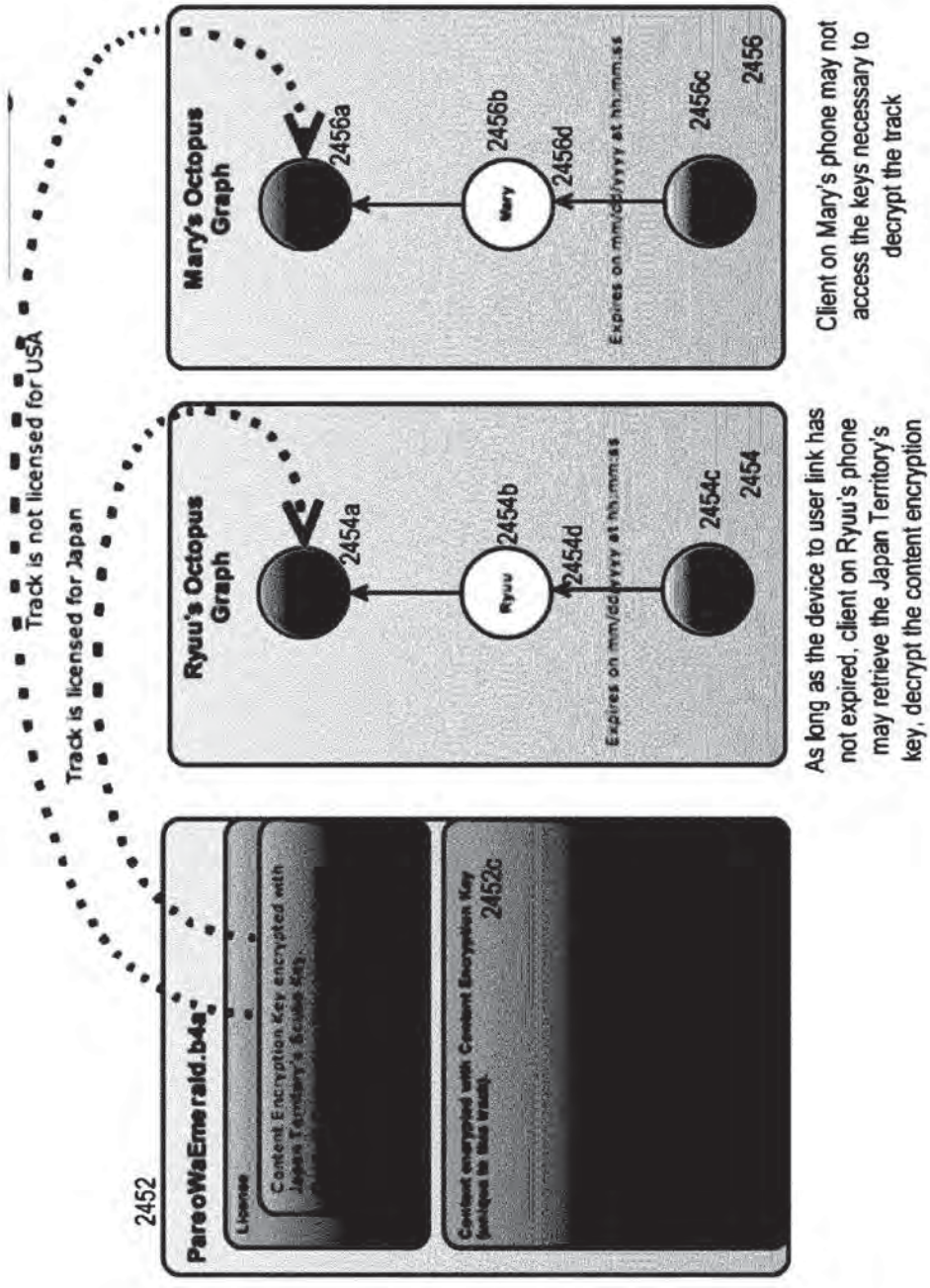

2452
ParooWaEmerald.b4g
License
Content Encryption Key encrypted with
Content encrypted with Content Encryption Key
2452c
Track is licensed for Japan
Track is not licensed for USA
Ryuu's Octopus Graph
2454a
Ryuu
2454b
2454d
2454c
2454
Mary's Octopus Graph
2456a
Mary
2456b
2456d
2456c
2456
As long as the device to user link has not expired, client on Ryuu's phone may retrieve the Japan Territory's key, decrypt the content encryption key, and decrypt the track
Client on Mary's phone may not access the keys necessary to decrypt the track

FIGURE 24b

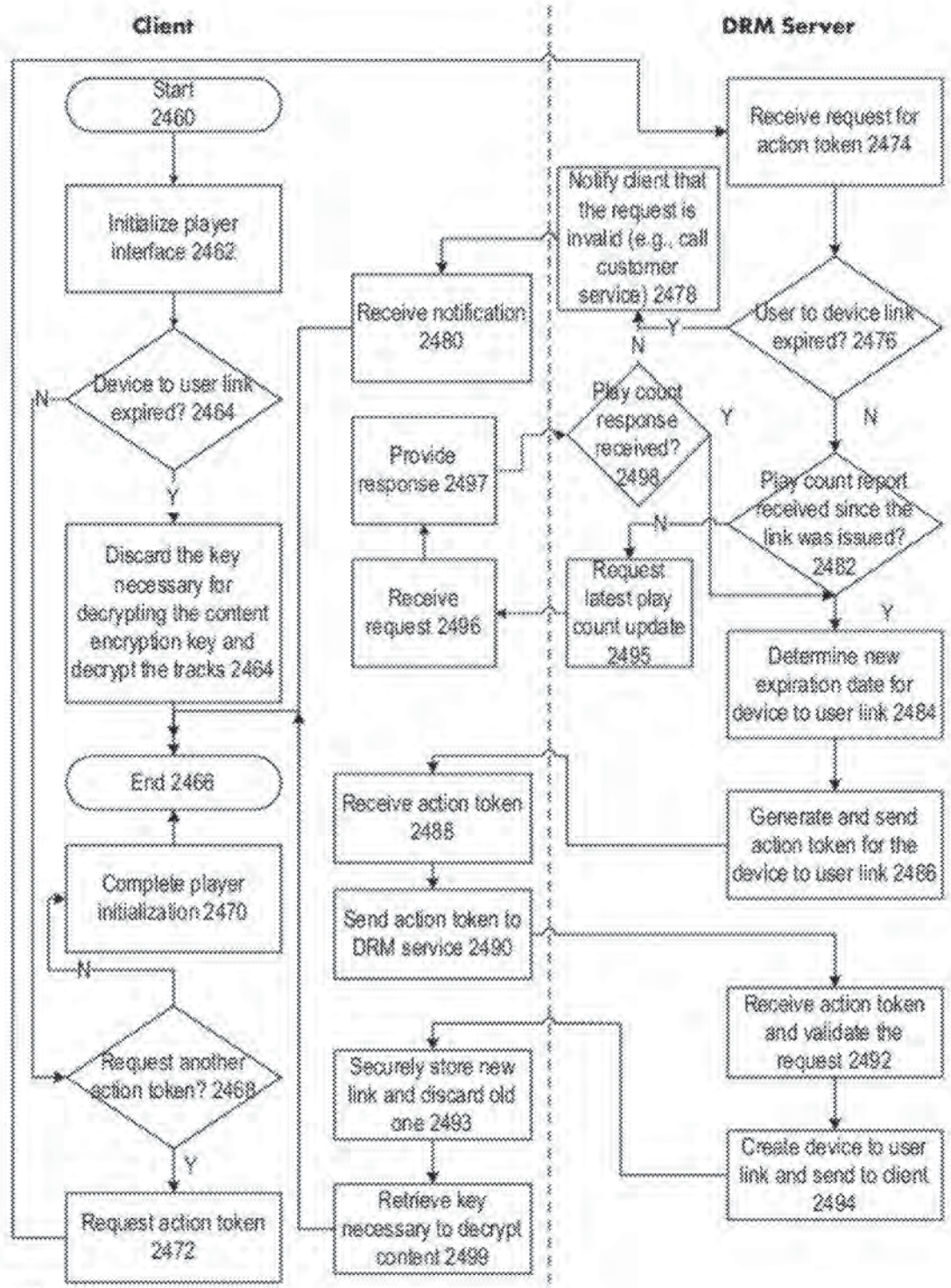

Client
DRM Server
Start 2460
Initialize player interface 2462
Device to user link expired? 2464
N
Y
Discard the key necessary for decrypting the content encryption key and decrypt the tracks 2464
End 2466
Complete player initialization 2470
N
Request another action token? 2468
Y
Request action token 2472
Receive request for action token 2474
Notify client that the request is invalid (e.g., call customer service) 2478
User to device link expired? 2476
Y
N
Receive notification 2480
Play count report received since the link was issued? 2482
Y
N
Request latest play count update 2495
Play count response received? 2498
Y
N
Provide response 2497
Receive request 2496
Determine new expiration date for device to user link 2484
Generate and send action token for the device to user link 2486
Receive action token 2488
Send action token to DRM service 2490
Receive action token and validate the request 2492
Create device to user link and send to client 2494
Securely store new link and discard old one 2493
Retrieve key necessary to decrypt content 2499

FIGURE 24c

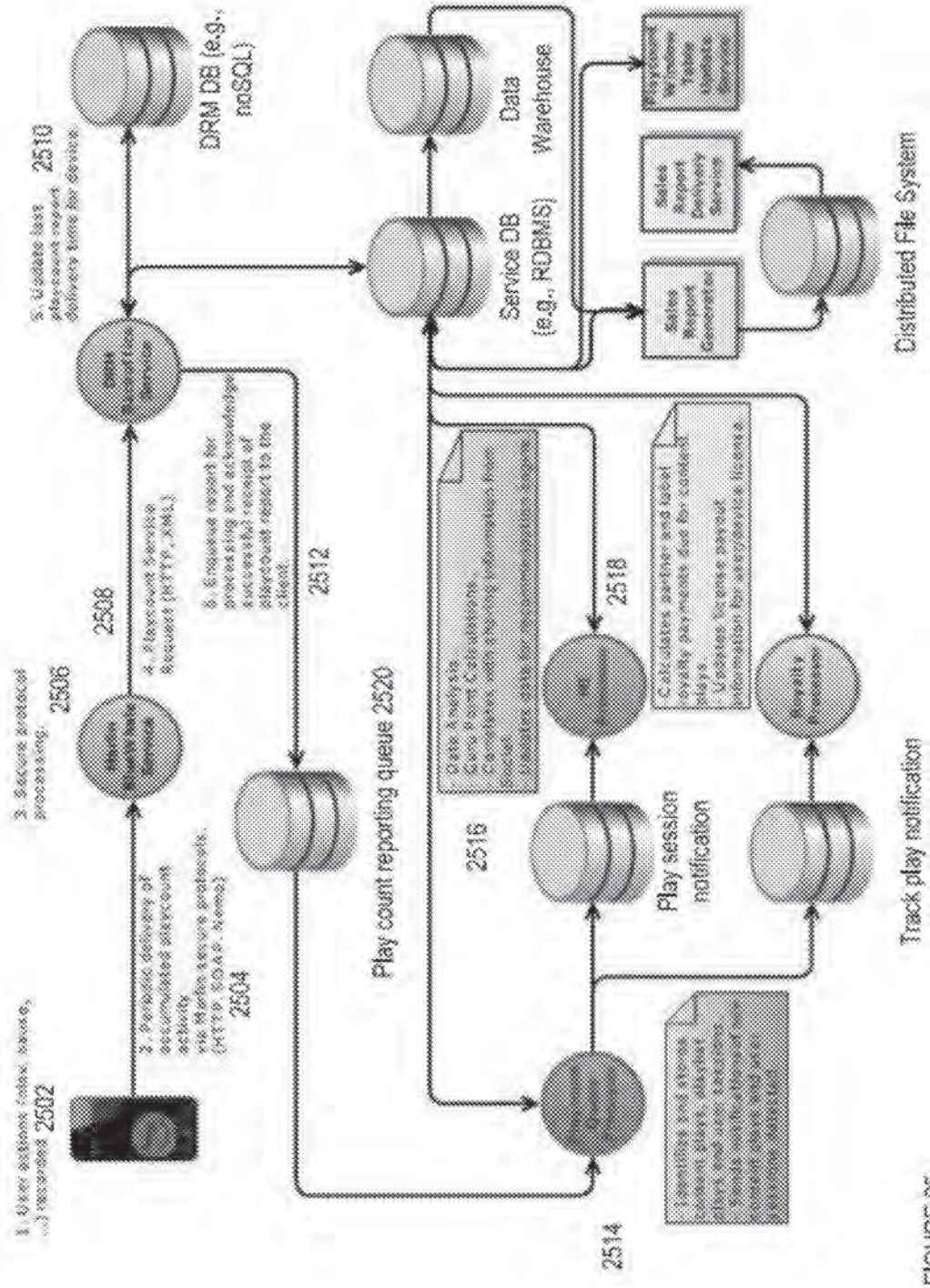

2502
2504
2506
2508
2510
2512
2514
2516
2518
2520
DRM DB (e.g., noSQL)
Service DB (e.g., RDBMS)
Data Warehouse
Distributed File System
Play count reporting queue 2520
Play session notification
Track play notification

FIGURE 25

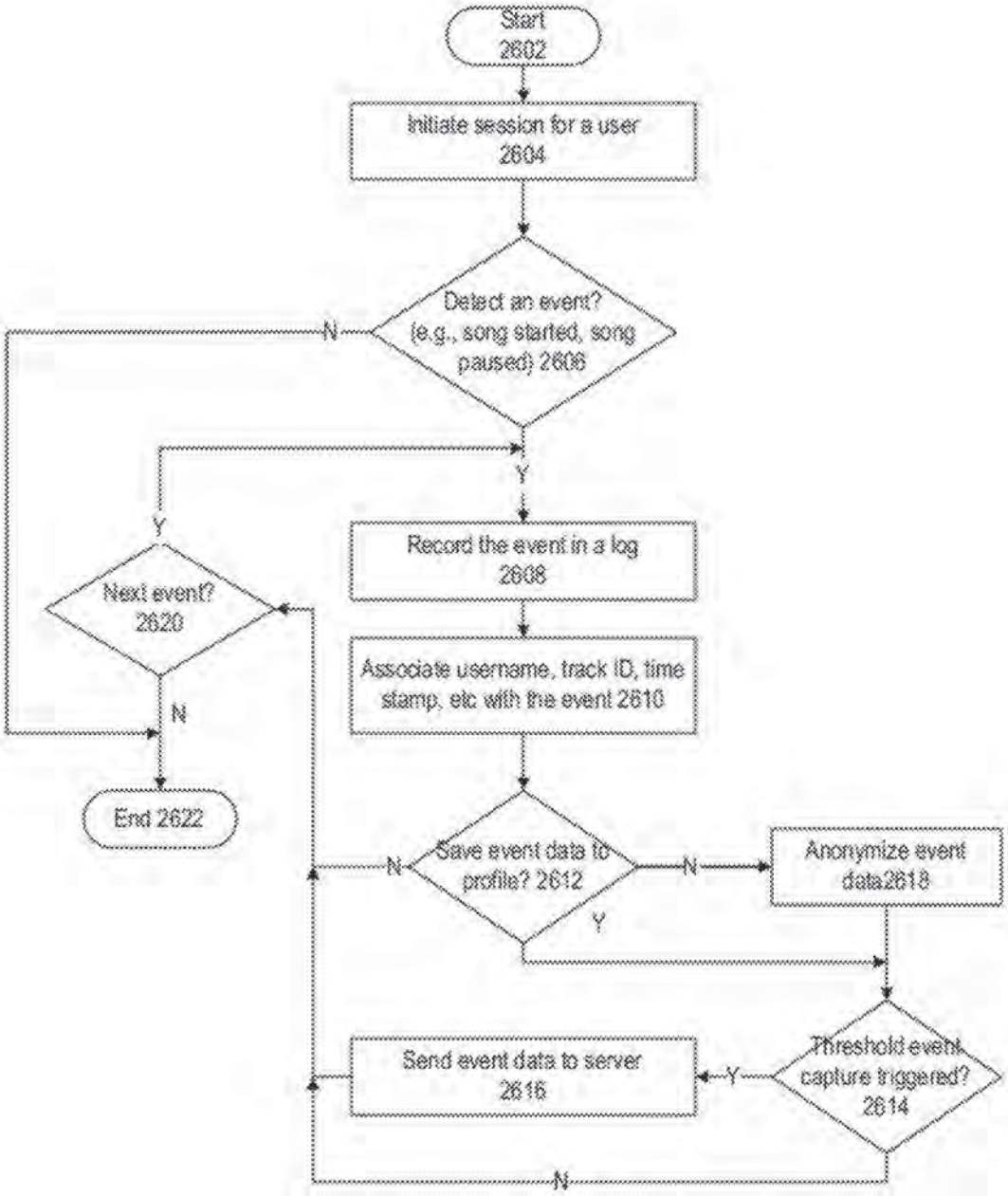

Start 2602
Initiate session for a user 2604
Detect an event? (e.g., song started, song paused) 2606
N
Y
Record the event in a log 2608
Associate username, track ID, time stamp, etc with the event 2610
Save event data to profile? 2612
N
Y
Anonymize event data2618
Threshold event capture triggered? 2614
Y
Send event data to server 2616
N
Next event? 2620
Y
N
End 2622

FIGURE 26a

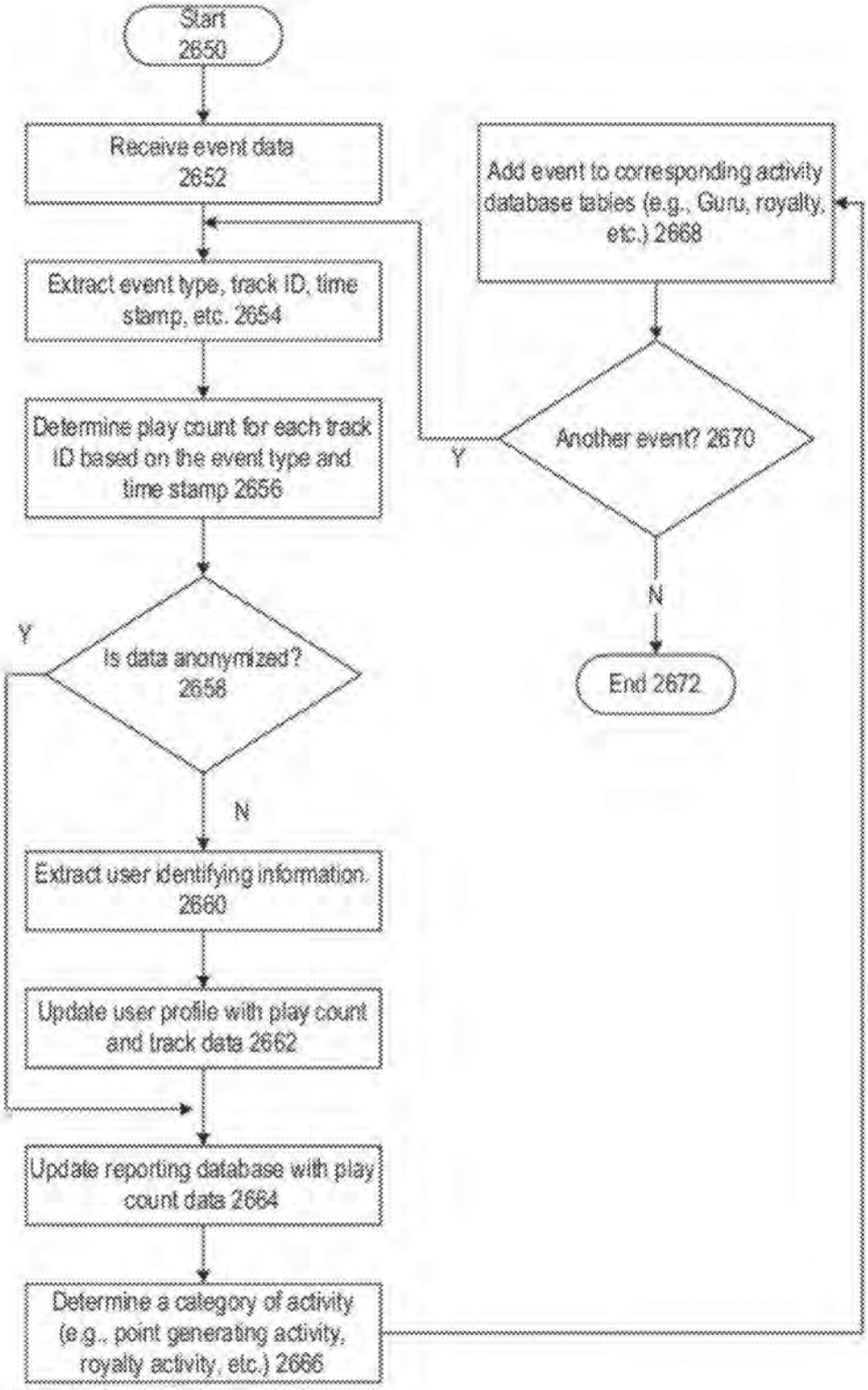

Start 2650
Receive event data 2652
Extract event type, track ID, time stamp, etc. 2654
Determine play count for each track ID based on the event type and time stamp 2656
Is data anonymized? 2658
Y
N
Extract user identifying information. 2660
Update user profile with play count and track data 2662
Update reporting database with play count data 2664
Determine a category of activity (e.g., point generating activity, royalty activity, etc.) 2666
Add event to corresponding activity database tables (e.g., Guru, royalty, etc.) 2668
Another event? 2670
Y
N
End 2672

FIGURE 26b

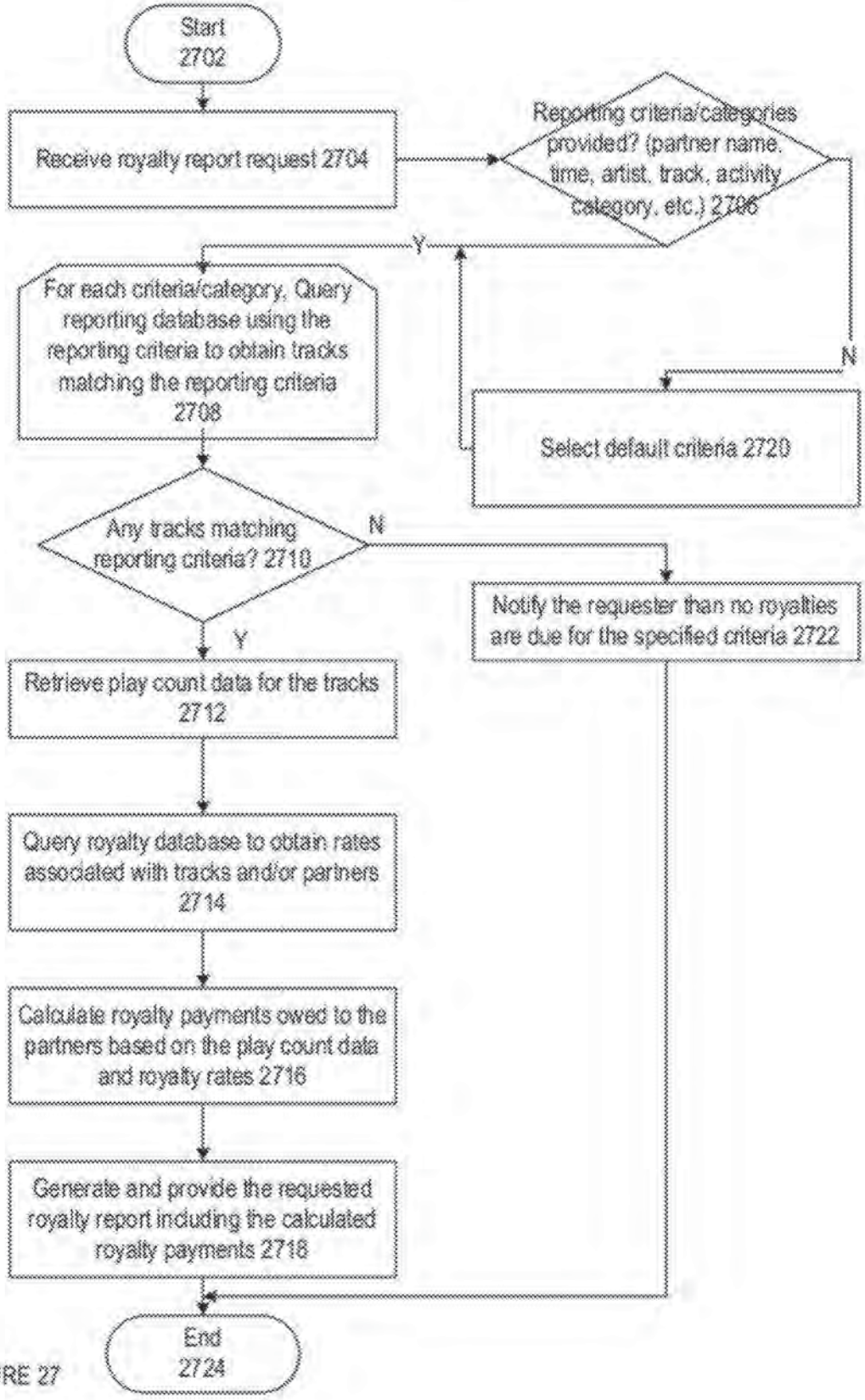

Start 2702
Receive royalty report request 2704
Reporting criteria/categories provided? (partner name, time, artist, track, activity category, etc.) 2706
Y
N
For each criteria/category, Query reporting database using the reporting criteria to obtain tracks matching the reporting criteria 2708
Select default criteria 2720
Any tracks matching reporting criteria? 2710
N
Y
Notify the requester than no royalties are due for the specified criteria 2722
Retrieve play count data for the tracks 2712
Query royalty database to obtain rates associated with tracks and/or partners 2714
Calculate royalty payments owed to the partners based on the play count data and royalty rates 2716
Generate and provide the requested royalty report including the calculated royalty payments 2718
End 2724

FIGURE 27

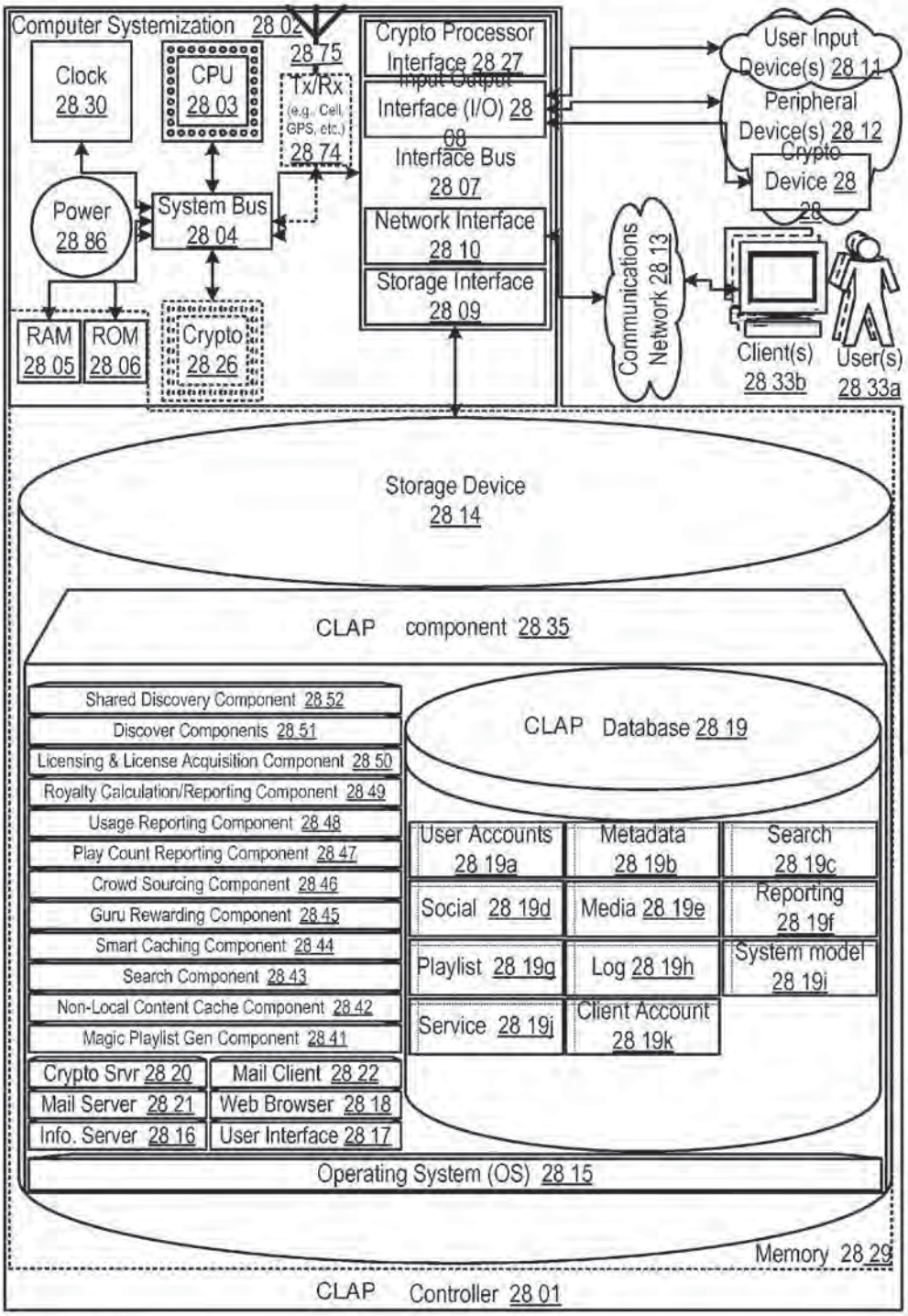

Computer Systemization 28 02
Clock 28 30
CPU 28 03
28 75
Tx/Rx (e.g., Cell, GPS, etc.) 28 74
Power 28 86
System Bus 28 04
RAM 28 05
ROM 28 06
Crypto 28 26
Crypto Processor Interface 28 27
Input Output Interface (I/O) 28 08
Interface Bus 28 07
Network Interface 28 10
Storage Interface 28 09
User Input Device(s) 28 11
Peripheral Device(s) 28 12
Crypto Device 28
Communications Network 28 13
Client(s) 28 33b
User(s) 28 33a
Storage Device 28 14
CLAP component 28 35
Shared Discovery Component 28 52
Discover Components 28 51
Licensing & License Acquisition Component 28 50
Royalty Calculation/Reporting Component 28 49
Usage Reporting Component 28 48
Play Count Reporting Component 28 47
Crowd Sourcing Component 28 46
Guru Rewarding Component 28 45
Smart Caching Component 28 44
Search Component 28 43
Non-Local Content Cache Component 28 42
Magic Playlist Gen Component 28 41
Crypto Srvr 28 20
Mail Client 28 22
Mail Server 28 21
Web Browser 28 18
Info. Server 28 16
User Interface 28 17
CLAP Database 28 19
User Accounts 28 19a
Metadata 28 19b
Search 28 19c
Social 28 19d
Media 28 19e
Reporting 28 19f
Playlist 28 19g
Log 28 19h
System model 28 19i
Service 28 19j
Client Account 28 19k
Operating System (OS) 28 15
Memory 28 29
CLAP Controller 28 01

FIGURE 28

CONTENT LICENSE ACQUISITION PLATFORM APPARATUSES, METHODS AND SYSTEMS

CLAIM FOR PRIORITY

[0001] This application claims priority to and benefit from International Application Number PCT/US2011/53780, filed 28 Sep. 2011, and titled CONTENT DISCOVERY AND DELIVERY PLATFORM APPARATUSES, METHODS AND SYSTEMS (Attorney Docket No. 20676-012PC); U.S. Provisional Application Ser. No. 61/526,210 filed on Aug. 22, 2011 titled CONTENT DISCOVERY AND DELIVERY PLATFORM APPARATUSES, METHODS AND SYSTEMS (Attorney Docket No. 20676-012PV1); U.S. Provisional Application Ser. No. 61/496,512 filed on Jun. 13, 2011 titled CONTENT DISCOVERY AND DELIVERY PLATFORM APPARATUSES, METHODS AND SYSTEMS (Attorney Docket No. 20676-012PV); U.S. Provisional Application Ser. No. 61/387,450 filed on Sep. 28, 2010, titled APPARATUSES, METHODS AND SYSTEMS FOR A MOLECULAR MULTIMEDIA SEARCH PLATFORM (Attorney Docket No. 20676-003PV); and U.S. Provisional Application Ser. No. 61/387,453 filed on Sep. 28, 2010 titled APPARATUSES, METHODS AND SYSTEMS FOR A MULTIMEDIA PIVOT SEARCH PLATFORM (Attorney Docket No. 20676-004M.

[0002] The entire contents of the aforementioned applications are herein expressly incorporated by reference.

[0003] This patent application disclosure document (hereinafter "description" and/or "descriptions") describes inventive aspects directed at various novel innovations (hereinafter "innovation," "innovations," and/or "innovation(s)") and contains material that is subject to copyright, mask work, and/or other intellectual property protection. The respective owners of such intellectual property have no objection to the facsimile reproduction of the patent disclosure document by anyone as it appears in published Patent Office file/records, but otherwise reserve all rights.

APPLICATIONS OF INTEREST

[0004] Applications of interest include: International Application Number PCT/US2009/061296, filed 20 Oct. 2009, and entitled A METHOD AND SYSTEM FOR ACCOUNTING FOR DOWNLOAD TRANSACTIONS AND SOCIAL NETWORK INTERACTION; International Application Number PCT/US2009/061307, filed 20 Oct. 2009, and entitled A METHOD AND SYSTEM FOR ACCOUNTING FOR DOWNLOAD TRANSACTIONS AND SOCIAL NETWORK INTERACTION; and International Application Number PCT/US2009/061309, filed 20 Oct. 2009, and entitled A METHOD AND SYSTEM FOR ACCOUNTING FOR DOWNLOAD TRANSACTIONS AND SOCIAL NETWORK INTERACTION.

[0005] The entire contents of the aforementioned applications are herein expressly incorporated by reference.

FIELD

[0006] The present innovations are directed generally to apparatuses, methods, and systems for multimedia applications, and more particularly, to CONTENT LICENSE ACQUISITION PLATFORM APPARATUSES, METHODS AND SYSTEMS.

BACKGROUND

[0007] Consumers of music and other multimedia have several options to purchase music that they like. They may go to music retail stores such as BEST BUY, TARGET or other independent stores and purchase a copy of their desired album packaged in the ubiquitous compact disc (CD) format. Consumers may also purchase digital copies of music from online music stores such as ITUNES and AMAZON, and subscription based services from service providers such as NAPSTER.

BRIEF DESCRIPTION OF THE DRAWINGS

[0008] The accompanying appendices and/or drawings illustrate various non-limiting, example, innovative aspects in accordance with the present descriptions:

[0009] FIG. **1** shows a diagram illustrating discovery aspects in one embodiment of the CLAP;

[0010] FIG. **2** shows a diagram illustrating example modes of discovery in some embodiments of the CLAP;

[0011] FIG. **3** shows an example data flow illustrating generation of a magic playlist in some embodiments of the CLAP;

[0012] FIGS. **4***a-b* are logic flow diagrams illustrating magic playlist generation component in one embodiment of the CLAP;

[0013] FIG. **4***c-d* are data flow diagrams illustrating remote client play in some embodiments of the CLAP;

[0014] FIG. **4***e* is a logic flow diagram illustrating non-local content cache component in some embodiments of the CLAP;

[0015] FIG. **4***f* is a logic flow diagram illustrating smart caching component in some embodiments of the CLAP;

[0016] FIG. **5** shows a logic flow diagram illustrating stagelight/spotlight search in some embodiment of the CLAP;

[0017] FIG. **6***a* shows an example shared discovery in some embodiments of the CLAP;

[0018] FIG. **6***b* shows an example shared discovery settings configuration in some embodiments of the CLAP;

[0019] FIGS. **7***a-c* show logic flow diagrams illustrating shared discovery components in some embodiments of the CLAP;

[0020] FIG. **8***a* shows a logic flow diagram illustrating the Gurus rewarding component in some embodiments of the CLAP;

[0021] FIG. **8***b* shows a logic flow diagram illustrating the Gurus offer delivery and redemption component in some embodiments of the CLAP;

[0022] FIGS. **9-14** are schematic views of the example CLAP application interface in some embodiments of the CLAP;

[0023] FIGS. **15***a-g* are schematic views of the example stagelight/spotlight interface in some embodiments of the CLAP;

[0024] FIGS. **16***a-c* are schematic views of the discover stream component in some embodiments of the CLAP;

[0025] FIGS. **17***a-f* are schematic views of the discover lens component in some embodiments of the CLAP;

[0026] FIGS. **18***a-c* are schematic views of the discover stacking component in some embodiments of the CLAP;

[0027] FIGS. **19***a-h* are schematic views of the molecular discovery component in some embodiments of the CLAP;

[0028] FIGS. **20***a-n* are schematic views of the example mobile application interfaces in some embodiments of the CLAP;

[0029] FIGS. **21***a-b* are schematic views of the example mobile application molecular interfaces in some embodiments of the CLAP;

[0030] FIG. **22***a* is a data flow diagram of an example content identification component in some embodiments of the CLAP;

[0031] FIG. **22***b* is a logic flow diagram illustrating an example content identification in some embodiments of the CLAP;

[0032] FIG. **23** is a logic flow diagram illustrating an example license acquisition component in some embodiments of the CLAP;

[0033] FIGS. **24***a-b* are data flow diagrams illustrating example licensing component in some embodiments of the CLAP;

[0034] FIG. **24***c* is a logic flow diagram illustrating an example license acquisition component in some embodiments of the CLAP;

[0035] FIG. **25** is a data flow diagram illustrating an example usage reporting component in some embodiments of the CLAP;

[0036] FIGS. **26***a-b* are logic flow diagrams illustrating example play count reporting components in some embodiments of the CLAP;

[0037] FIG. **27** is a logic flow diagram illustrating an example royalty reporting component in some embodiments of the CLAP; and

[0038] FIG. **28** shows a block diagram illustrating embodiments of a CLAP controller.

[0039] The leading number of each reference number within the drawings indicates the figure in which that reference number is introduced and/or detailed. As such, a detailed discussion of reference number **101** would be found and/or introduced in FIG. **1**. Reference number **201** is introduced in FIG. **2**, etc.

DETAILED DESCRIPTION

Clap

[0040] In one embodiment, the CLAP allows unlimited, lifetime-of-device and lifetime-of-ownership access to a comprehensive catalog of music ("Infinite Music Library," "universal music library, "CLAP catalog") on licensed devices (LDs). Such LDs support digital rights management (DRM) and adhere to the CLAP specification for reporting instances of digital downloads and/or plays ("play count"). LDs include a licensed CLAP client ("application"), a full-features music management application providing access to a comprehensive catalog of music via catalog browsing, music discovery, social networking, track download and playback, and/or the like.

[0041] In some embodiments, the CLAP may include facilities for music downloads, music playback controls, library management, playlist management and sharing, license activation, account registration, artist, album or Guru browsing, automatic disc-space management features, music search, a variety of discovery interfaces include molecular, lens, streaming and stacking discovery interfaces, automatic playlist synchronization to the universal music library, and across one or more user devices, user library reconciliation ("beyondization"), side-loaded device management, direct social interaction through the CLAP community and extended social interaction with existing online communities such as FACEBOOK, MYSPACE, TWITTER and LINKEDIN, player add-ons that extend the capabilities of the CLAP application in search, recommendation, sharing and discovery, and/or the like.

[0042] In some embodiments, the CLAP may include a web-based browser interface with search, discovery, cloud-based content sync, social music features and other components. For example, a CLAP user ("user," "consumer" or "customer") may download tracks, albums or playlists, may obtain song previews or listen to full length tracks, learn more about artists and their music, and discover related artists and their music.

[0043] In some other embodiments, the CLAP may include a portal or an application client for web-based partner (B2B) account management and partner interface functioning as the face of the one or more databases and/or tables. Some components of the CLAP may account for and pay copyright owners royalties, in a way that ASCAP, Harry Fox, SESAC or BMI may do, for each and every play of a digital music file on an LD. Copyright owners, including record labels, publishers, and other entities associated with content rights are referred to as partners.

[0044] The CLAP catalog includes a large number of content items that have been cleared for legal distribution and sharing among the CLAP users. Each content item in the CLAP catalog is uniquely identified. Some embodiments of the CLAP may facilitate assigning of unique identifier for licensed and distributable music. Other embodiments of the CLAP may further facilitate assigning of unique identifiers for unlicensed music on a local client device, thereby uniquely and universally resolving each content item.

[0045] The CLAP's user interfaces facilitate several modes of content discovery. Content discovery may allow users to discover new music, new social connections, new information, and/or the like. The CLAP may further facilitate users to expand and/or refine their own music preferences and knowledge base from others users' usage history, playlists, favorites, etc. In some embodiments, discovery may be driven by driven by a variety of factors including Gurus, personal usage patterns and/or content meta data. In other embodiments, discovery may be facilitated by a variety of visually engaging and interactive user interfaces.

[0046] As such, aspects of the CLAP facilitate discovery, sharing, playback, and secure usage data aggregation and reporting, and/or the like.

[0047] The CLAP application may run on a variety of operating system platforms including Windows, Macintosh, iOS (Apple iPhone, iPod Touch, etc.), Android, Symbian, Java/J2ME, Samsung Bada, Windows Mobile 7, RIM Blackberry, HP Palm WebOS, Google Chrome, set top boxes (Linux), automotive devices, other mobile handsets, portable media players and consumer electronic devices. In some implementations, manufacturers may install a copy of the application in devices during manufacturing. The CLAP application may also be installed on devices, post-manufacture, by device resellers (e.g., mobile carriers), by technicians at a point-of-sale location, or by consumers themselves (e.g., from a website, an application store for mobile devices, web stores such as Chrome web store, etc.).

[0048] FIG. **1** shows a diagram illustrating discovery aspects in one embodiment of the CLAP Platform. As illustrated in the diagram, a user **104/108** may be uninterested in his or her current selection of music on his or her LD. He or she may desire to know not just what is currently popular, or on the top 40 lists, but also what his or her friends are listening to, which tracks are being recommended, and/or the like. The discovery interface **116** facilitates the discovery of music which the user may enjoy using LDs **106/110**.

[0049] In one implementation, the discovery interface **116** may include a network of friends **112** and/or subject experts known as "Gurus" **114** through whom the user **104/108** may discover music. Gurus are users who have opted in to the Guru program which is discussed in detail with respect to the Guru program component.

[0050] FIG. **2** shows a diagram illustrating example modes of discovery in some embodiments of the CLAP. Examples modes of discovery **200** include, for example, Gurus **202**, magic playlist **204**, deep/molecular discovery **206**, shared discovery **208**, search **210**, stagelight/spotlight search **212**, infinite library home **214**, and/or the like. Each of these modes of discovery **200** is discussed in greater detail below.

Magic Playlist Generation (MPG) Component

[0051] Magic playlists are automatically generated playlist of content related to a "seed" item such as an artist, an album, a track, a playlist, another user or a combination thereof. In one embodiment, a magic playlist generation algorithm may utilize historical (e.g., listening/usage history), social (e.g., what friends are listening to), usage, profile, Gurus, track rating, crowd sourcing and other data to create a magic playlist. In one implementation of the MPG algorithm, these factors may be weighted using one or more weighting criteria. For example, in one implementation, a "social" weighting criterion may be selected. Such "social" criteria may result in more weight being accorded to social factors such as friends' listening history, Guru recommendations, crowd sourcing, etc. Similarly, other criteria may include personal usage (e.g., emphasis on the listener's usage history), profile (e.g., emphasis on the listener provided profile information), promotional (e.g., emphasis on music promoted by advertisers or other third parties), music traits (e.g., emphasis on music trait analysis), and/or the like. In some implementations, a combination of one or more of these criteria may be utilized to identify tracks that a user is likely to enjoy and/or share with other users.

[0052] FIG. **3** shows an example data flow illustrating generation of a magic playlist in some embodiments of the CLAP. As shown in FIG. **3**, in one implementation, a user **302** may request a magic playlist at **310** by selecting, for example an artist, as a content seed and clicking a "magic playlist" icon. In one implementation, the magic playlist request may be packaged as an HTTP(S) POST message for transmission to the servers **306**. An example magic playlist request message **312** formatted in the eXtensible Markup Language (XML) form may be as follows:

```
POST /magicplaylist.php HTTP/1.1
Host: www.websitename.com
Content-Type: Application/XML
Content-Length: 1306
<?XML version = "1.0" encoding = "UTF-8"?>
```

-continued

```
<MagicPlaylist>
    <Username>dingdong555@gmail.com</Username>
    <Timestamp>2011-02-22 15:22:43</Timestamp>
    <ClientDetails>
        <ClientIP>192.168.23.126</ClientIP>
        <ClientType>smartphone</ClientType>
        <ClientModel>HTC Hero</ClientModel>
        <OS>Android 2.2</OS>
        <AppInstalledFlag>true</AppInstalledFlag>
    </ClientDetails>
    <EventDetails>
        <Event>create magic playlist</Event>
        <SeedType>artist</SeedType>
        <SeedName>pink floyd</SeedName>
    </EventDetails>
</MagicPlaylist>
```

[0053] The magic playlist request message **312** may be transmitted to the servers **306** over the communication network **300** for further processing. In one implementation, the request **312** may be routed to those servers that are geographically proximal to the location of the user's client device **304**. At **314**, the servers may initiate generation of the requested magic playlist that may include a list of tracks selected based on the seed item and/or other factors. In one implementation, the servers **306** may execute the magic playlist generation algorithm to obtain a list of tracks for the requested magic playlist. Details of the magic playlist generation algorithm(s) are discussed with respect to FIGS. **4***a* and **4***b*.

[0054] At **316**, the created magic playlist may be saved in a playlist database **308**. The playlists in the playlist database **308** may be available to all users or may be subject to restrictions imposed by the respective creators. At **318**, content related to the magic playlist tracks may be retrieved. For example, if the magic playlist includes a track by the artist "Eagles," the related content may include, for example, a list of playlists including the same track that are created and shared by other users. Similarly, information such as similar artists, albums corresponding to the tracks, etc., may also be retrieved. Identification of related content information is further discussed in detail with respect to the Smart Caching component.

[0055] At **320**, the generated magic playlist may be sent by the servers **306** over the communications network **300** to the client device **304**. At **322**, the related content information may also be sent over the communications network **300** to the client device **304**. In one implementation, the response message **320** and/or **322** from the servers **306** may include track IDs and thumb nail images corresponding to the tracks in the magic playlist. In a further implementation, the response message **320** and/or **322** may also include information on related tracks, artists, albums, playlists including the magic playlist tracks, Gurus associated with the magic playlist tracks, and/or the like. The response message may specifically include track names, track IDs, album names, album IDs, artist names, artist IDs associated with the related content. An example response message **320** and/or **322** may be in XML format substantially in the following form:

```
<XML>
  <MagicPlaylist>
    <PlaylistInfo>
```

-continued

```
        <Username>dingdong555@gmail.com</Username>
        <Timestamp>2011-02-22 15:23:00</Timestamp>
        <PlaylistName>Pink Floyd Magic Playlist</PlayListName>
        <PlaylistImg>DM2345.png</PlaylistImg>
        <SeedType>artist</SeedType>
        <SeedName>pink floyd</SeedName>
      </PlayListInfo>
      <TrackIDInfo>
        <TrackID1>6786865</TrackID1>
        .
        .
        .
        <TrackID20>342554</TrackID20>
      </TrackIDInfo>
      <TrackImgInfo>
        <TrackImg1>DM3243.png</TrackImg1>
        .
        .
        .
      </TrackImgInfo>
      <RelatedArtistInfo>
        <RelatedArtist1>Genesis</RelatedArtist1>
        .
        .
        .
        </RelatedArtist9>Peter Gabriel</RelatedArtist9>
      </RelatedArtistInfo>
      <RelatedPlaylistInfo>
        <Playlist1>Jeb loves Roger Waters</Playlist1>
        <Playlist2>Jenna's Wall</Playlist2>
        .
        .
        .
      </RelatedPlaylistInfo>
    </MagicPlaylist>
  </XML>
```

[0056] In one implementation, the CLAP client may receive the message **320** and/or **322** and may parse the received message to extract the track IDs. The CLAP client may also use the extracted track IDs to retrieve and display track information such as track name, track length, location of the track (e.g., local library, cloud library or connected device library), images, and/or the like.

[0057] In some implementations, the CLAP client may keep a log of various events associated with playlists which may include magic playlists. For example, a user may remove a track that was part of a playlist, and may create a new version of the playlist without the removed track. As another example, a user may reorder the tracks within a playlist in a different sequence. Example playlist events logged by the CLAP client may include, but is not limited to: tracks removed from the playlist, name of artist upon which magic playlist was created, name of track upon which magic playlist was created, name of tracks in playlist, name of playlist upon which magic playlist was created, name of album upon which magic playlist was created, total times a playlist was renamed (track composition may or may not have changed), total times playlist was reordered, and/or the like. The log may be periodically uploaded to the CLAP server(s).

[0058] FIG. **4***a* is a logic flow diagram illustrating MPG component in one embodiment of the CLAP. In one implementation, the CLAP may receive seed item information such as an artist, a track, an album, a playlist, a Guru, and/or the like at **402**, which is provided as an input to the MPG algorithm. The seed item may be universally resolvable. A user may provide this information by selecting or entering the seed item. If the seed item is a content item as determined at **404**, one or more content attributes that define the seed item may be extracted at **404**. Examples of content attributes may include, for example, track meta data information such as album, artist, comment, copyright, title, track, performers, genre, label, cover art, song length, a globally unique identifier (GUID), and/or the like. The content attributes may further include stylistic traits such as female vocal, electric guitar, tonality, tempo, bass, etc. and/or the like.

[0059] If the user provided content seed is not a content item, the content seed may be analyzed using several heuristics. Examples of non-content items, i.e., items that are not content and do not have a corresponding media file, may include, for example, a friend name, a Gurus name, mood, tempo, comment and/or the like. At **418**, content seed representation heuristics may be determined. Examples of the content seed representation heuristics may include a specified default or predetermined item **418***a*, currently played track **418***b*, most played track **418***c*, favorite track **418***d*, favorite genre **418***e*, last purchase **418***f*, others **418***g* and/or the like. These representative heuristics may also be derived from aggregated correlation among entire community of users, circle of friends or entity graph.

[0060] In some implementations, entity graph may include social graph and interest graph. Social graph may include categories of friends and/or acquaintances from social networks such as FACEBOOK, MYSPACE, TWITTER, GOOGLE+, and/or the like. Social graph may also include categories of friends and acquaintances from the CLAP community and/or other communication media such as instant messengers, contacts from address books, and/or the like. Interest graph may include categories of friends and acquaintances from any of the above social networks and the CLAP community who have an interest profile similar to a user. Additionally, an interest graph of a user is more expansive and may include people that do not have a social relationship with the user, but with who the user may share similar interests, usage pattern, listening history, and/or the like. In addition to people, entities may also include organizations and companies reachable through any of the mentioned social networks, CLAP community network and other communication media.

[0061] Referring to FIG. **4***a*, content information pertaining to one or more of these heuristics may be retrieved at **420**. For example, for the last purchase heuristic **418***f*, an album ID or a track ID corresponding to the last purchase made by the user may be obtained. Now that the non-content item seed is transformed using heuristics to a content item seed, the attributes of the content item seed may be determined at **406**.

[0062] At **414**, user profile preference information may be retrieved. The user profile preference information may include user provided information such as favorite artist, favorite genre, album, etc. Such information may be provided by the user during registration, profile creation or profile update. The user profile preference may also be derived from the user's CLAP profile. The CLAP profile may be an "in house" profile organically built over time using information learned from the user, such as listening history. The CLAP profile may include not only information such as the user's preferred album, tracks, artists, genre and other attributes, but also a breakdown of the user's preferences according to location, time, mood (e.g., mood indicated by the user's mood status) and/or the like. For example, the CLAP may track and analyze the user's listening history over time to surmise that the user is likely to listen to ambient music late at night, rock music in early morning, and instrumental music at midday,

etc. Furthermore, the user profile preference information may facilitate the CLAP's efforts to pre-fetch tracks in anticipation of the user's desire to listen to such pre-fetched tracks.

[0063] Further at **416**, the user's social graph information may be retrieved. Social graph information may include friends, family, acquaintances, corporate entities, and/or the like. Social graph connections may exist within the CLAP network as well as other social network sites such as FACEBOOK, TWITTER and LINKEDIN. Using the social graph information, the CLAP may obtain information such as music his or her friends are interested, their listening history, playlists, currently played, most played, favorite, last purchased tracks, artists or albums, and/or the like. In a further implementation, the degree of separation and/or degree of friendship between the user and members of his or her social network may be taken into consideration. The degree of friendship may be established using information provided by the user and/or information derived from the frequency of communication exchanged between two users, existence of similar relationship in multiple social networks, and/or the like. The degree of separation may be established using the social graphs of various users in the network.

[0064] In one implementation, one or more weighting categories may be established in order to determined a magic playlist that embodies not just the musical attributes, but also one or more social and personal preferences. These categories may be selected manually at **422**, which may trigger a dialog box requesting, for example, the user to select or input one or more social/personal categories that should be considered. The user selected category preferences may then be loaded at **428**. On the other hand, these categories may be predefined parameters in the algorithm, in which case, applicable categories may be retrieved at **424**. These categories may include, for example, the user's listening history, thumbs up/down rating of tracks, Guru rating of tracks, the user's friends' listening history, most recently played, most frequently played, most shared, most purchased, recently released, day/time preferences, and/or the like. Further, each category may be assigned its category weight. In one implementation, these category weights may be predetermined and as such, set by the CLAP at **426**. In an alternate implementation, these category weights may be included in the loaded preferences specified by the user at **428**. The user, when asked for category selections, may also be requested to specify how important that category is for the user. (For e.g., How important is social influence—select one: (a) very important, (b) not so important or (c) indifferent).

[0065] Referring now to FIG. **4***b*, the extracted content attributes, along with the retrieved user profile preferences and social graph information, may be utilized to form a query for searching a list of content items that have matching or similar content attributes. Such a search may be executed on one or more databases and/or tables of the CLAP at **430**. Further, such a search may be executed for relevancy, popularity, aggregated popularity, and/or the like. Relevancy, for example, may be established in various ways. Consider, for example, an artist ATB as a seed item. ATB may have attributes such as trance roots, dance feel and female vocal. A relevancy search based on these attributes may find other artists/tracks having these attributes. Further, a three attribute match may be considered more relevant than a two-attribute match. In some implementations, relevancy search may be based on fingerprinting technologies provided by third party services such as Gracenote.

[0066] At **432**, for each heuristic category, the magic playlist algorithm may be used to identify content items for the magic playlist. A query based on each content criterion may be constructed and run on one or more databases and/or tables at **432**. Although each query may result a small or large number of results, only the first n number of results having high similarity metric may be selected. At **436**, the category weight may be assigned to each category items obtained from the query. The process may continue until all the category item results have been assigned category weights. When all the categories have been exhausted at **438**, each content item result for each category may be assigned a position factor or ranking based on relevance at **440**. In one implementation, the algorithm may determine "relevancy" by calculating a relevancy score for each identified track. Table 1 below illustrates example results from a query based on four criteria and the calculation of the relevancy score for each unique result.

	CATEGORY (C)			
POSITION (P)	1	2	3	4
10	Track A	Track B	Track C	Track D
9	Track X	Track A	Track Y	Track A
8	Track B	Track P	Track Q	Track E
7	Track C	Track R	Track F	Track J
...	...	...	...	...
1	Track Q	Track J	Track J	Track A

[0067] For each unique track t, based on the category weight (C) and the position weight (P), the relevancy score may be calculated as below in one implementation:

$$S_{t=1 \text{ to } x} = \sum_{j=1}^{M} \sum_{k=1}^{N} C_j \times P_{jk}$$

[0068] Where, S=relevancy score, x=track, j=category and k=position in the list. In this example, M=4 and N=10.

[0069] At **442**, the weights (category and position) may be used to input in the formula above to obtain the relevancy score for each unique track identified. The highest value tracks i.e., the unique tracks having the highest relevancy scores S may be sorted at **444**. At **446**, top x number of the highest value tracks (e.g., 20 tracks) may be selected for inclusion in the requested magic playlist. In some implementations, other methods of calculating relevancy based on weighted criteria may be utilized.

[0070] Once the content items for the magic playlist have been identified, the magic playlist may be sent to the requesting user's client device. The user may then select a content item to play. FIGS. **4***c-d* show data flow diagrams illustrating remote client play in some embodiments of the CLAP. In FIG. **4***c*, a client device **478** may make an application programming interface (API) request to an API server **476** that returns metadata information for the selected content item. If the content item is licensed for the authenticated user, download universal resource locators (URLs) for standard/mobile audio files may be included. The API request **484** may in one implementation be a JSON over HTTP POST request. Another HTTP post request **486** may then be made to a content distribution network (CDN) **480** for downloading an audio/video file for the requested content item using the appropriate

download URL obtained from the API server **476**. In one implementation, the CDN may be utilized for facilitating faster downloads. The CDN may be in communication with a media server **482** over HTTP for **486** for obtaining copies of the audio/video files. The downloaded audio/video file may then be played back by the user.

[0071] Referring now to FIG. **4***d*, data flow between the CDN **480**, the CLAP client and the user interface **494** is discussed in more detail. In one implementation, the CLAP client may include a download thread **490***a*, a media player **490***b* and an Input/Output (I/O) stream interface **490***c*. In one implementation, the download thread **490***a* and/or the I/O stream interface **490***c* may be substantially implemented using C++. In some implementations, the media player **490** may be implemented using a software development kit (SDK) that are digital rights management (DRM) compatible. The download thread **490***a* may track the download status of a content item (e.g., an audio file). For example, the item being downloaded may be saved locally as a local audio file **490**. The download thread **490***a* may track the download status and may issue download status events to indicate state transitions for the local file. Examples of state transitions may include "downloading" to "playable" to "downloaded."

[0072] When the user interface **494** receives the download status event **496***a* from the download thread **490***a* indicating that the local audio file is playable, it can start playback. When the user selects a play control from the user interface **494**, the play event **492***b* may be sent to the media player **490***b* to begin playback of the requested local audio file **490**. The media player **490***b* may then issue playback events that are used in the generation of playback status events **492***c* for the user interface, and in playcount event recording. In one implementation, the audio file may become playable after only a small portion of the file is downloaded. In one implementation, the small portion may include only the license, the MP4 atoms, and/or the portion of the encrypted content that contains the first sample of the unencrypted content. For a track that is 3 minutes and 30 seconds long, it may be playable after approximately 30 kB (e.g., mobile quality file size), and approximately 50 kB (e.g., standard quality file) have been downloaded.

Content Caching Component

[0073] In some embodiments of the CLAP, one or more local client devices may be provided with facilities for media content caching. In one implementation, caching may be limited to those content items that are not available locally in the client device. In another implementation, caching may be for those content items that match the client device owner specified preferences. Caching may be carried out in the background without any active user intervention in some implementations.

[0074] FIG. **4***c* shows a logic flow diagram illustrating the content caching component in some embodiments of the CLAP. In one implementation, a user may select or input a content item at **450**. For example, a user may select an artist as a content seed. The selected content item may be received by the server at **452**. The server may then identify the selected content item at **454** and determine related content based on the item at **456**. For example, if the user selected artist "Janelle Monae", the CLAP platform may identify tracks related to the artist, albums related to the artist, information about the artist, playlists including tracks by the artist, artists similar to Janelle Monae, and/or the like. The determination may include creating a content query based on the received input and/or other input identifying information and querying one or more content databases using the created content query. The related content data determined by the CLAP may be sent to the user's client device at **458** and may be received by the user's client device at **460**.

[0075] In one implementation, at **462**, a determination as to whether any of the tracks returned by the search are non-local tracks. In some implementations, non-local tracks are tracks that are not available locally in the client device. For example, non-local tracks may be present in the universal music library in the cloud, but are absent in the user's local device, either in one or more media folders, or cache. This determination may be achieved, for example, by comparing the track IDs returned from the search with the track IDs of the tracks saved locally. If there are no non-local tracks, i.e., all tracks from the search result are local, those tracks may be marked as locally available using an indicator at **466** to distinguish them from non-local tracks. The related content may then be displayed on the client device at **468**.

[0076] On the other hand, if one or more tracks are non-local tracks, a local cache request may be generated for requesting the non-local tracks from the server. In one implementation, the local cache request for the non-local tracks may be in XML format substantially in the following form:

```
<XML>
    <ClientDetails>
        <ClientIP>192.168.22.111</ClientIP>
        <ClientType>smartphone</ClientType>
        <ClientModel>HTC Hero</ClientModel>
        <OS>Android 2.2</OS>
    </ClientDetails>
    <UserName>JaneSmith@gmail.com</UserName>
    <NonLocalTrackDetails>
        <Track1ID>238348</Track1ID>
        <Track2ID>338458</Track2ID>
        .
        .
        .
        <Track18ID>245788</Track18ID>
    </NonLocalTrackDetails>
</XML>
```

[0077] The server may receive the request for non-local track and may retrieve the requested tracks from one or more media content databases using the universally resolvable track ID in the XML request at **470**. The retrieved tracks may then be sent to the requesting user's client device at **472**. The requested tracks may be received by the client device at **474** for temporary storage in the client device cache. The received tracks may be encrypted, and may become playable after only a small portion of the content file is downloaded.

[0078] In one implementation, the content items stored in the cache may be made permanent by the user at any time as long as the item has not been cleared from the cache. The CLAP client may mark the previously local tracks as locally available tracks at **466** and may update the CLAP client interface to display the received and/or identified related content data at **468**.

[0079] In another embodiment of the CLAP, the content caching component may facilitate caching of non-local items in a media content collection explored, created or copied by a user. For example, a user may create a playlist (or request a magic playlist) which may include a list of tracks. The CLAP

may then automatically determine whether any of the tracks in the list are non-local, and if so, may obtain the non-local tracks for temporary storage in the local cache. Such automatic caching for non-local items may enhance user experience, whether the user is online or offline, with seamless delivery of music.

[0080] In some implementations, the content caching component may include a cache manager component. The cache manager component may facilitate management of content data stored temporarily in the cache. When cache memory is filled up, there may not be enough space in the cache to store new content items. In order to make room for new content items, older content items may have to be deleted. The cache manager may determine which content items to delete based on various factors such as priority, play count, last play date and time, size of content items, date and time of storage, and/or the like. In one implementation, priority may be determined based on user preference. In another implementation, priority may be determined based on track play attributes. For example, if a content item is a recommendation engine ranked song or of a ranked album, the content item may have a higher priority and thus may not be the first track to be deleted.

Smart Caching (SC) Component

[0081] Some embodiments of the CLAP may include an SC component that may facilitate smart caching of content items that are likely to be consumed by a user. The SC component may include a recommendation engine that generates music recommendations based on, in one implementation, users' implicit interests. In a further implementation, the SC may also include a cache manager component that manages caching queue. Together, the recommendation engine and the cache manager may identify tracks that are likely to be of interest to the user, and download the identified tracks to the user's client cache. In one implementation, a user's interests may be derived from the behavioral and use data collected from the user's client and/or website. For example, length of play, i.e., whether the user plays a track for 10 seconds or the full length, may be a strong indicator of the user's interest in that particular track, genre or artist. Similarly, if the user adds a track to a playlist, creates a magic playlist with a track or shares a track, such activities may also be a strong indicator of the perceived value of the track to the user. As yet another example, tracks that are played in proximity in terms of timing and/or session may also suggest common clusters for recommendation.

[0082] In some implementations, the collected play data may include, but are not limited to play data such as track play detail, track added to playlist, track shared, rating, track plays in close proximity, track ID, track bookmarked, track used to create magic playlist, and/or the like. Track play detail may include information such as genre, data and time of play, partial or full play, and/or the like. In one implementation, a play that is less than 30 seconds long may be classified as partial play, while a play that is longer than or equal to 30 seconds may be classified as full play. In other implementations, the critical play length may be a number different from 30 seconds, or may even be a percentage of the track length (e.g., a track play that is at least 30% of the track length may be considered a full play).

[0083] In some implementations, the collected data may also include personally identifiable information (PII), user ID, and/or the like. In one implementation, PII may include any information (i) that may identify or may be used to identify, contact, or locate the person to whom such information pertains; and (ii) from which identification or contact information of an individual person is derived. In some implementations, PII may include, but is not limited to: name, address, phone number, fax number, email address, financial profiles, medical profile, social security number, credit card information, and/or the like. Additionally, in some other implementations information such as a personal profile, unique identifier, biometric information, IP address, and/or the like associated with PII may be considered PII. In yet other implementations, PII may not include information that is aggregated or collected anonymously (i.e., without identification of the individual user) or demographic information not connected to an identified individual. In some implementations, PII may include third party PII. In some implementations, user ID may be a totally anonymous number series or alphanumeric characters.

[0084] Using the collected data, the recommendation engine may in some implementations, aggregate tracks, artists, albums, Gurus, playlists that are consistently favored or banned by the user. The recommendation engine may also keep track of user ratings (e.g., thumbs up, thumbs down) for content items. In a further implementation, the aggregated tracks may be periodically refreshed to ensure that the recommendation source content items are currently of interest to the user. These recommendation source content items may also be a part of the user profile in some implementations. Table 1 below shows example fields of data collected and maintained by the recommendation engine.

FIELD	DATA TYPE
most_played_artist	char
most_played_album	char
most_played_playlist	char
most_played_genre	char
most_played_track	char
artists_filtered_out_as_thumbs_down	char
playlists_filtered_out_as_thumbs_down	char
albums_filtered_out_as_thumbs_down	char
tracks_filtered_out_as_thumbs_down	char
guru_filtered_out_as_thumbs_down	char
artists_flagged_as_thumbs_up	char
playlists_flagged_as_thumbs_up	char
albums_flagged_as_thumbs_up	char
tracks_flagged_as_thumbs_up	char
guru_flagged_as_thumbs_up	char
tracks_bookmarked	char
most_shared_track	char
most_popular_track_in_interest_graph	char

[0085] In some implementations, the recommendation engine may consider one or more of the listed fields of collected data to identify seed content items for recommendation. For example, the recommendation engine may consider the tracks identified by the fields: most_played_tracks, tracks_filtered_out_as_thumbs_up, most_shared_track and most_popular_track_in_interest_graph. The recommendation engine may then compute a fingerprint for each recommendation seed using a fingerprinting technology. The fingerprinting technology may use digital signal processing algorithms to process actual audio signal of a recording to compute the fingerprint.

[0086] In some implementations, the recommendation engine may package the recommendation seeds and/or other identifying information in an XML format and send to a

third-party service such as Gracenote MusicID Service via an application programming interface (API) for identifying related content. In yet other implementations, recommendations may be generated using crowd sourcing methods such as that of Pandora or collaborative filtering approach of Amazon (e.g., others who bought x, also bought y).

[0087] FIG. **4***f* is a logic flow diagram illustrating smart caching component in another embodiment of the CLAP. At the start **491***a*, the user may request download of a new multimedia file at **492**. Using the attributes of the multi-media file and/or other play detail information, at **493**, the user's preference profile is updated. At **494**, as soon as an engageable portion of the multimedia file is obtained, it is ready for playback if requested by the user. At **495**, the SC component may identify content items for smart caching. The SC component may utilize the recommendation engine to identify the content items for smart caching. At **496***a*, the SC component may determine whether there is an existing intelligent download list/queue. If there is an intelligent download list, the SC component may update the list with the identified content items at **496***b*. If there is no intelligent download list, one may be created at **496***c* using the identified content items. At **497**, the SC component may determine if the user has enough bandwidth to determine the number of concurrent downloads. If the bandwidth is high enough to download at least one file, the SC component may instruct the CLAP cache manager to identify and delete local multimedia in the cache to make space for the new files at **498***b*. If the user bandwidth does not have enough bandwidth at **498***a*, the SC component may hold caching until user bandwidth is ideal at **498***c*. Once the bandwidth is ideal, the SC component may provide the additional multimedia content to local client device according to the intelligent download list at **499**. This may conclude the SC component caching at **491***b*.

Stagelight/Spotlight Search

[0088] FIG. **5** shows a logic flow diagram illustrating stagelight/spotlight search in some embodiments of the CLAP. In one implementation, the stagelight search may begin at **502** with a user selection of a content item for stagelight search at **504**. For example, a user may select a track, artist, album, etc., and initiate a stagelight search. The selected content item may be received by the server at **506**. The server may then initiate user selection tracking. User selections and the sequence in which such selections are made may be employed to generate a search path history.

[0089] At **508**, the selected content item may be identified as an artist, track, album, playlist and/or the like. Using the identified content item, other related content items may be searched and identified at **510**. In some implementation, related content items may include albums, tracks, artists, Gurus, playlists, and/or the like. The related content item data may then be sent to the user's client device at **512**. The sent content item data may be utilized to render an updated client interface at **514**. At this time, the user may continue interacting with content items to discover related content items. For example, the user may select a related content item, which may lead to discovery of additional related content items. If the user wishes to stagelight a related content item displayed on the user interface, the user may select the content item and may click on stagelight option or icon at **504**.

[0090] In another implementation, the stagelight interface may feature a drag and drop interface, where the user may stagelight a content item by dragging and dropping the content item to a stagelight object. At **516**, the user may decide to not stagelight any of the related content items, at which point the user may exercise the option to view the search path traversed so far. At **518**, the user may request to view the search path traversed, and in response, the server may retrieve the stored user selections including indications of the sequence in which they were selected and may send them to the client device at **522**. At **514**, the client device may receive the search path data and may render the graphical interface to display the search path taken by the user as the user traversed through related content items. The process may end at **520** when the user exits the stagelight search interface within the CLAP client.

Shared Discovery Component

[0091] FIG. **6***a* shows an example shared discovery in some embodiments of the CLAP. In one implementation, the CLAP facilitates graphical discovery of shared content items through the network of users of the CLAP and/or the network of users of other social networking sites such as FACEBOOK, MYSPACE, LINKEDIN, etc. For example, using the CLAP GUI **600**, a user may see a list of all other users who are sharing their content (e.g., library, playlists, etc.). The user may navigate further by selecting options such as "home," "people," "names," "content," "type," "content list" and/or the like. When the user selects "home," a list of options **610** may appear under the "home" column for further selection. In one implementation, the "home" option **610** may include an option to select people, music, books, video, apps, and/or the like. As shown in FIG. **6***a*, the user may select "people" from the column **610**. The selection may then populate the "people" column **612** for further selection options. These options may include Gurus, social, friends, favorites, and/or the like. In one implementation, the user may select "Gurus" from the column **612**. The selection of "Gurus" may in turn populate the next column **614** with Guru names for further selection. In one implementation, the user may select a Guru name (e.g., Steve Mori) from the names column **614**. The selected user (i.e., Steve Mori) may have configured for sharing one or more content items such as albums, genre, artists, purchased, tempo, mood all content, playlist, and/or the like as shown in the content column **616**. Further granularity in terms of content type **618** and content list **620** may also be available for selection in some implementations. As shown in FIG. **6***a*, the user may be provided an option to select "albums" from the content column **616**, which in turn may provide a drilled down view of content type **618** for selection. Examples of content type may include most recently played, most highly rated, purchased, all, and/or the like. Any selection of the content type items from the content type column **618** may lead to a display of content list in column **620**. As shown, the content list may display a list of tracks under any of the selected content type (e.g., most recently played). In one implementation, the user may select any of the displayed tracks for playing, downloading or creating magic playlist. In another implementation, the user may generate a playlist including all or selected tracks from Steve Mori's shared library. For example, the user may create a playlist including the most recently played tracks, most highly rated tracks or all tracks.

[0092] In one embodiment, users who wish to share their content with other users may be able to individually configure their social privacy settings to define what items they would like to share and what items they would like to keep private.

FIG. **146***b* shows an example shared discovery settings configuration in some embodiments of the CLAP. In one implementation, a user may select profile settings option from the menu to view the profile settings interface. An example profile settings interface may include options to configure social settings, device settings, publishing settings, and/or the like. In one implementation, the user may select social settings to reveal an interface including options for social privacy settings. In a further implementation, the social privacy settings interface may include options to select who can view what type of contents from the user's library contents. Example options for selecting who can view may include friends, friends of friends, social network members, network members, anyone, and/or the like and any combination thereof. Example options for selecting what can be viewed may include see everything, see playlists, see most recently played, see only selected, and/or the like and any combination thereof. As shown in FIG. **6***b*, a user may select to let his friends see everything. In an alternate implementation, there may be options to configure social privacy settings on a group basis. For example, the user may configure to let his friends see everything, friends of friends see most recently played, anyone see his playlists, and/or the like.

[0093] In some implementations, the privacy settings may include options to specify the people, music, books, video, apps, and/or the like listed in column **610** in FIG. **6***a*. For example, the user may configure people who are gurus named Jenna J., J J Abrams or Guettafan access to albums, purchased and playlist content that are most recently played. In this way, only the specified gurus may be given access to specified content type.

[0094] FIG. **7***a* shows a logic flow diagram illustrating shared discovery in some embodiments of the CLAP. In one implementation, the process may begin at **200** with a user using his or her client device to request access to a member's content data at **702**. The request may involve, for example, selection of the member's name or alias. The request may be received by the server at **704**. The server may then retrieve the member's privacy settings at **706**. In one implementation, the relationship between the member and the requesting user may be determined for applying the member's privacy settings. For example, the member's privacy settings may permit sharing with friends, and the relationship between the member and the requesting user may be determined in order to facilitate the user's request. At **708**, a determination may be made whether the requesting user should be granted access to the member's content data. Based on the privacy settings and/or the member-user relationship, if the user is not granted access to the member's shared content data, a notification message indicating denied access may be generated and sent to the user at **710**. Such a notification message may be received and rendered on the client interface at **712**. At this time at **714**, the user may select another member and request access to the selected member's shared content, or may end the process at **716**.

[0095] If the user is granted access to the member's shared content, the server may retrieve and send to the user's client device permitted content selections at **718**. Permitted content selections may include the contents selected by the member for sharing with users of certain designations such as friends, friends of friends, network members, etc. The permitted content selections may be received by the user's client device for rendering on the client interface at **720**.

Shared Content Management (SCM) Component

[0096] The shared media content management component of the CLAP may, in some embodiments, allow multiple users to collaboratively manage a shared media content collection such as a media library, a portion of a media library, a playlist, a specific collection (e.g., "my jazz collection" or "Smith family library"). Collaborative management may take various forms depending on the type of users involved. For example, a family may seek to create and maintain a single "family library" which can be modified by one or more family members using their individual LDs such that each LD having access to the "family library" may be automatically synchronized to display the most recently modified "family library." As another example, two friends, each having their own LD, may seek to manage a shared "jazz collection." The shared "jazz collection" would then be commonly owned by the two friends and any change one makes to the collection would be immediately reflected in the collection displayed to the other friend if the friend is connected to the internet, or upon connection to the internet.

[0097] In some implementations, LDs in the vicinity of each other may establish communication with each other via blue tooth, infra-red or other near field communication methods when such communication methods are enabled by the respective users. A handshake protocol, where a LD that is offline is authenticated through via another LD that is offline, may be carried out. After successful completion of the handshake protocol, the offline LD may receive content shared by the online LD, including updates to any shared collection commonly owned by the online LD and offline LD.

[0098] FIG. **7***b* shows a logic flow diagram illustrating SCM component set up in an embodiment of the shared discovery of the CLAP. The set up may start at **734**, where a server may receive a request from a client device of first user ("sharing user") to share a target content collection with a second user ("shared user") at **736**. The target content collection may include a library, a portion of a library, playlist, and/or the like. The share request may include information relating to client details such as client IP address, device type and model, operating system, and/or the like. The request may also include identifying information relating to the sharing user and the shared user. User identifying information may include, but is not limited to, an email address, user name, alias or nick name, address, phone number, and/or the like. The request may also identify the content collection being shared, such as the content collection name (e.g., library name, collection name, playlist name, etc.), and in some implementations, the track identifiers (e.g., Track ID) of the tracks in the specified content collection. The share request, in one non-limiting example, may be in XML format substantially in the following form:

```
<XML>
    <ClientDetails>
        <ClientIP>192.168.22.111</ClientIP>
        <ClientType>smartphone</ClientType>
        <ClientModel>HTC Hero</ClientModel>
        <OS>Android 2.2</OS>
    </ClientDetails>
    <UserDetails>
```

-continued

```
            <UserName>JaneSmith@gmail.com</UserName>
            <SharedUser1>JoeKline@gmail.com</SharedUser1>
            <SharedUser2>BillMadd@hotmail.com</SharedUser2>
        </UserDetails>
        <SharedContentDetails>
            <SharedLibraryName>Jane's Jazz
            Collection</SharedLibraryName>
            <Track1ID>238348</Track1ID>
            <Track2ID>338458</Track2ID>
            .
            .
            .
            <Track50ID>245788</Track50ID>
        </SharedContentDetails>
</XML>
```

[0099] At **738**, after receiving the share request from the sharing user, an instance of the specified content collection may be created. Content items may be populated in the instantiated shared content collection file according user specified constraints. At **740**, the SCM component may identify content items in the target content collection. In one implementation, the identification may include extracting Track IDs of content items in the target content collection from the XML share request. At **742**, the SCM component may any retrieve restrictions on the target content collection. These restrictions may be restrictions on who can view and/or share, when the contents may be viewed and/or shared, where the contents may be viewed and/or shared, and/or the like. In some implementations, these restrictions may be specified by the sharing user. For example, a sharing user may want to make the shared collection a private collection between himself or herself and the shared user. In other implementations, these restrictions may be related to license rights. For example, certain tracks in the sharing user's target collection may be licensed for Japan. The tracks licensed for Japan would be decrypted by the sharing user's LD when the sharing user is in Japan. However, when the sharing user shares such tracks licensed only for Japan with a shared user in the United States, the SCM component may detect the licensing restrictions, and apply such restrictions to the appropriate tracks in the target collection to ensure that shared user may view and/or share only those tracks that are licensed for his or her own territory, in this example, the United States.

[0100] At **744**, the SCM component may query one or more media content databases and/or tables for the non-restricted target content collection items, i.e., the target content collection items from which restricted content collection items have been removed. At **746**, the SCM component may then retrieve any modification restrictions to be imposed on the target content collection. The modification restrictions may be specified by the sharing user, and may include rights to modify the shared target content collection. Modifications may include adding, deleting or renaming content items or collection, and/or the like. For example, in some implementations, a sharing user may allow adding content items to the shared target content collection, but restrict the sharing user from deleting any items from the target content collection. Selective restrictions may also be possible. For example, a sharing user may allow deletion of any artists other than, for example, "Norah Jones." These modifications may be displayed in the form of options for users to select. The user may also be allowed to craft and apply one or more restriction rules. For example, when a parent is sharing a content collection with a child, the parent may set up rules that, for example, do not allow any tracks having explicit lyrical content or other parental advisories to be added to the content collection.

[0101] At **748**, the SCM component may then populate the shared content collection file with the results of the query and apply the retrieved modification restrictions. At **750**, the SCM component may time stamp the shared content collection file and store the file in one or more shared content database and/or tables. Time stamping not only ensures that the most recently updated instance of the content collection file is retrieved when requested, but also allows a user to go back in time to retrieve prior versions of the shared content collection. At **752**, the SCM component may invite the shared user to access the shared content collection. For example, an invitation email or message within the CLAP (e.g., "Jon wants to have a shared rock collection with you. Do you accept?") may be sent to obtain the shared user's acceptance. If the shared user accepts at **754**, the SCM component may confirm with the shared user if he or she would like to merge the shared content collection with her or her own content collection. If the user agrees to merge at **756**, the SCM component may obtain the list of content items in the shared user's collection at **758**. The SCM component may then compare the shared user's content collection with the shared content collection to identify non-duplicate content items. The identified non-duplicate content items may be added to the shared content collection file at **760**. Allowing the option to merge would require that there are no restrictions to add content items. Other view/share restrictions and modification restrictions may be applicable in some implementations. The shared content collection file may then be time stamped and stored in the shared content database at **762**. At **764**, the most recent shared content collection would be displayed to all shared users.

[0102] At **756**, in one implementation, the shared user may not agree with the merger, or may be restricted from exercising the merger (e.g., adding to the shared content collection). In this case, the shared user may be provided an option to replace his or her content collection at **772**. If the shared user agrees, the SCM component may then replace his or her content collection with the shared content collection at **774**, concluding the set up process at **766**. However, if the shared user does not agree to the replacement, the shared content collection will be saved as a separate collection in the shared user's CLAP account at **776**. The shared content collection may then be displayed on the user interface of the CLAP application as a separate shared content collection.

[0103] At **754**, the shared user may not accept the sharing user's invitation to access the shared content collection. In this case, the SCM component may notify the requesting user i.e., the sharing user at **768**. At **770**, the sharing user may be given an option to select another user to share the target content collection. If the sharing user decides to select another user, the set up process may move to **752**. On the other hand, if the sharing user does not want to select another user for sharing, the set up process concludes at **766**.

[0104] FIG. **7***c* shows a logic flow diagram illustrating SCM component update in an embodiment of the shared discovery of the CLAP. Once a shared content collection has been established, the SCM component may maintain and update the shared content collection as necessary. The update process may start at **778**. At **780**, a user of the shared content collection may request to modify the shared content collection. The request may be initiated when a user attempts to modify the shared content collection. In one implementation,

the CLAP application interface may include an icon or button for requesting modification. The modification request may include an identifier for the shared content collection (e.g., SC768765, name of collection, etc.), user identifier, modification type (e.g., add, delete, rename, etc.), content items to be modified, and other relevant information. At **782**, the SCM component may check the attributes of the shared content collection for any restrictions, including view/share restrictions and modification restrictions. If the requested modification by the user is not allowed, the requesting user may be notified at **794**. The requesting user may also be provided an option to try again at **796** (e.g., try a different modification or try a different shared content collection). If the requested modification by the user is allowed at **784**, the SCM component may make the modification, and then time stamp the modified version of the file before saving it on the shared content database at **786**. The determination of whether a modification request by a user is allowed or not may involve examining the attributes for restrictions from **782**. Specifically, in some implementations, the SCM component may verify whether the user has the right to modify the content collection and/or whether the modification is acceptable under the restrictions in **782**. At **788**, the SCM component may check to determine if any of the users of the shared content collection are online (i.e., connected to the CLAP). At **788**, when no users of the shared content collection are online, the SCM component update process may end at **798**. However, if at **788**, one or more users of the shared content collection are online, the SCM component may push the updates to the users' client devices to display the most recently modified shared content collection at **790**. In some implementations, the users may be notified of the availability of updates and may be requested to provide approval for pushing the update. At **792**, the SCM component may entertain any additional modification requests available.

Guru Program Component

[0105] In some embodiments, the Guru program component helps identify and recognize and/or reward users for the value, whether intrinsic or extrinsic, they deliver. In some implementations, value may be derived directly or indirectly from the activities of Gurus. Examples of value for the CLAP include driving new and/or repeat license sales, enhancing the experience for other users by influencing their music plays, driving promotions sponsored by partners such as artists, labels, publishers, and/or the like. In some other implementations of the Guru program, driving new and repeat license sales may be deemed a function of enhancing the experience for other users by influencing their music plays. As such, in some implementations, influence may directly correlate with plays.

[0106] In one implementation, influence may be any activity carried on by or caused by a user that is responsible for content items played by another user. For example, when a user publishes a playlist, the user's activity may be an influencing activity if another user plays songs from the published playlist or publish notification, follows the playlist and play songs in it, navigate to the user's profile and play songs from the playlist directly from the profile, and/or the like. Similarly, an influencing activity may also include a user sharing a content item such as an artist, album, song, and/or the like, with which another user may end up interacting. Non-limiting examples of such interaction may include a user playing songs from the share notification, clicking through the share and playing songs from the artist or album page, adding the artist, album or song to his or her personal collection (e.g., "My Collection") and playing songs from there. Other examples of an influencing activity may include a user playing a song, and another user navigating to the user's profile and playing songs from designated collections such as "Top Artist," "Most Recently Played," "Most Frequently Played," "Top Songs," and/or the like directly from the user's profile. The another user may also navigate to the user's profile and add songs from the designated collections to his or her own collection ("My Collection") and play songs from the collection.

[0107] While there are many examples of influence within a social graph, an interest graph or the wider community of user of the CLAP, not all activities of a user that leads to song play by another user may be considered influencing activities. Table 2 below shows some non-limiting examples of non-influencing activities:

ACTIVITY	EXPLANATION
Copying	If User B copies the songs in User A's published playlist into one of his or her own playlists and plays songs from there, User A may not get points for plays from the copied playlist.
Searching	If User B learns about an artist/album/song/playlist from User A, searches for it, and then plays it or adds it to My Collection and plays it from there, User A may not get points for plays from the search.
Multi-User	If User A influences User B who influences User C, User A may not get points for User C plays
Related Content Plays	If User A influences User B who plays a song related to the influence, User A may not get points for related plays

[0108] In other alternate implementations, the non-influencing activities from table 2 may be considered influencing activities.

[0109] Other activities that may lead to rewards and/or recognition include, without limitation, sharing libraries, playlists and/or knowledge, posting to other users', friends', and followers' comment streams, recruiting new followers and/or friends, suggesting and/or recommending music to friends and/or followers, sharing playlists for followers, friends and/or other, increasing music consumption, reviewing contents **6** (artists, albums, playlists) and posting to content comment streams, and/or the like.

[0110] In one embodiment of the Guru program these and other influencing activities may be tracked to recognize and reward users for their advocacy and effort. Table 3 below list some non-limiting examples of use cases and exemplary data tracked in some embodiments of the Guru program.

USE CASES	EXPLANATION	TRACKED DATA
Track plays	Evaluate if and/or how Gurus impact other customers' behavior (e.g., usage volume, range of choice, and/or the like), before and after they begin following: Form an informed profile of Gurus.	Source, track type, access method, track ID, date & time, and/or the like

-continued

USE CASES	EXPLANATION	TRACKED DATA
Playlist	Determine if a Guru delivers value to a follower	Playlist creation and/or publishing, following, and unfollowing history, playback frequency, copying of playlists (i.e., becoming an owner), editing, reviews and/or comments, ratings, and/or the like.
Sharing	Determine if a Guru is proactively sharing content	Sharing channel, consumer response to the sharing, and/or the like.
Purchase	Identify who to attribute a sale of a LD, a CLAP license and/or subscription to.	Purchase history of CLAP licenses, buying process, funnel conversion, and/or the like.
Opt-in	Guru accepts invitation to participate in the Guru program and agrees to share data publicly	Opt-in preference.
Followers	Evaluate relationship between Gurus and their followers.	Personally identifiable information (PII), user ID, following status, and/or the like.

[0111] As shown in Table 3 above, behavioral, product use, account management, registration, and other information relating to Gurus and/or followers may be tracked. In some implementations, data relating to track plays may be obtained for evaluating if and/or how Gurus impact other users' behavior, before and after they began following one or more Gurus. For example, a Guru shares a track via TWITTER and a non-CLAP user clicks on a link within the tweet. The link takes the user to the CLAP hosted landing page where he or she buys a license and plays the track. In this case, the Guru who shared the track via TWITTER gets a credit for play, free trial and/or paid accounts. Various user behavior data including usage volume, range of choices, etc., and whether such behavior occurrences are increasing and/or expanding may be tracked for rewarding Gurus, to get more information on product usage, to create and maintain an informed Guru profile, for music intelligence, customer profiling, product feature evaluation, dashboard reporting, marketing response, fraud monitoring, partner reporting and/or the like.

[0112] As listed in Table 3, track play detail data for Gurus and/or followers may include, but are not limited to, source, track type, access method, track ID, date and time, and/or the like. A source may include the CLAP library, imported content, shared content (e.g., via email, FACEBOOK, TWITTER, links, etc.), and/or the like and details relating to the sharing user. Track type may include information relating to whether a track is downloaded by the user, downloaded in the smart cache, streamed from the cloud, imported, and/or the like. Similarly, access method may include information relating to how and/or from where the track was accessed. For example, access method may indicate where in the user interface (e.g., library, discovery or community) or from which artist, album, track, playlist, published playlist, chart, offline view, bookmarks, etc., a track was accessed from. Further, track ID may include the track number, associated meta data and/or the like. These data may be collected from the client and/or website.

[0113] In some implementations, Guru playlist related data may be collected for determining if a Guru delivers value to one or more followers. Examples of playlist related data collected may include playlist creation and/or publishing, following and/or unfollowing history, playback frequency, copying of playlist (becoming an owner), editing, reviews and comments, ratings, and/or the like. For example, unfollowing may suggest that the follower did not find any value in a Guru's playlist, while interaction with a playlist may suggest value. Similarly, playlist contributions of followers may be a function of Guru influence. These data, in addition to providing an insight into user behavior and/or product use, may also be useful for music intelligence, customer profiling, product feature evaluation, dashboard reporting, marketing response, fraud monitoring, partner reporting and/or the like.

[0114] In some implementations, sharing data may be collected to determine if a Guru is proactively sharing music. In particular, 1:1 sharing where messages make a single loop may be useful in evaluating how recipients are responding to shared content items. The sharing data may also be utilized to assess if a Guru is stimulating more listening, what channels are being used for sharing, which channel is most effective, and/or the like. Examples of tracked sharing data may include sharing channel, consumer response to the sharing, and/or the like. Sharing channel may include sharing via channels such as email, FACEBOOK, TWITTER, within the CLAP, hyperlink on website or other digital channels, and/or the like. Consumer response may include response of users to the sharing in the form of track plays, track downloads, purchase of tracks, licenses, LDs, and/or the like. In addition to behavior and product use information, these data may also have other uses in customer profiling, product feature evaluation, dashboard reporting, marketing response, partner reporting, and/or the like. These data may be collected from the client and/or websites.

[0115] In some implementations, purchases may also be tracked to identify who to attribute a sale of a LD, a CLAP license and/or subscription. They may also facilitate in determining whether a follower buys more licenses, whether a user buys a LD when his or her old device dies, whether license sales is attributable to a Guru activity, whether a Guru is driving new business, and/or the like. The outcome from these determinations may feed into the Gurus' points and rewards discussed in further detail below. Non-limiting examples of tracked data may include purchase history of CLAP licenses, buying process, funnel conversion, and/or the like. Purchase history may include information including type of license, price, device type, and/or the like. Buying process may include channels and source of traffic such as email, FACEBOOK, TWITTER, within the CLAP, hyperlink on website or other digital channels, and/or the like, as well as funnel conversion. These data may uses in customer profiling, product feature evaluation, dashboard reporting, marketing response, partner reporting, and/or the like. Purchase data may be collected from website registration page(s), account management portals, or from any appropriate locations from the client and websites.

[0116] In some implementations, opt-in preference data may be collected from users who accept invitation to participate in the Guru program. The opt-in data may be used for publicly recognizing and rewarding users in the Guru program. In a further implementation, opting in may require the user to agree to share his or her data publicly. Opt-in preference may be collected from the account management portal or relevant web page from where the user may opt in. In one

implementation, the default opt-in value may be "off" and Guru participation may require a user to opt-in in order to change the value to "on."

[0117] In some implementations of the Guru program, data on followers may be collected for evaluating relationship between Gurus and their followers. Such data may include, in one non-limiting example, Personally Identifiable Information (PII), user ID, following status and/or the like. Following status in some implementations may include information that may identify whether a user is following another or is being followed by another. In some implementations, Guru and follower user IDs and/or other information may be attached to play events, sale events, or other events. In some implementations, there may be an option to create an alias or nickname that is public facing. These data may be collected from website registration page and account management portal among others. In addition to the Guru program, these data may be used for music intelligence, customer profiling, product feature evaluation, dashboard reporting, marketing response, partner reporting, and/or the like.

[0118] In some embodiments, the Guru program may be leveraged by the discovery and social aspects of the CLAP. The point-based system for rewarding users may, in some embodiments, sustain user engagement with the facilities of the CLAP and encourage users to actively listen, share and manage music. In some implementations, "Guru" may be the designation bestowed to those users who may have opted in to the Guru program and are eligible to receive rewards and/or recognition. In other implementations of the Guru component of the CLAP, a user opt-in to the Guru program to earn influence points may be optional. As discussed above, influence points may be earned when one user plays a song due to influence from another user. Each play by the influenced user may earn the influencing user one or more points or a fraction thereof. In one implementation, when the influencing user earns his or her first point, it triggers a 12 month timer. In some implementations, a time period other than the 12 months may be selected. During the 12 month period, the influencing user may earn points for all plays by the influenced user according to a schedule. An example schedule may, for example, reward the influencing user 1 point for each play by an influenced user during the first month. The example schedule may further stipulate that for the next 2-3 months, each play by an influenced user may earn the influencing user 0.75 points. Similarly, for the next 4-6, the points earned may be 0.5 points, and for the following 7-12 months, 0.25 points. After the 12 month time period, any play by the influenced user would not earn any points for the influencing user. In one implementation, each user that the influencing user influences to play a given song or other content may initiate a separate timer such that the influencing user may perpetually earn points for a given song or other content as long as new users continue to play it. While one specific example of an earning schedule is discussed herein, other variations in the earning schedule are within the scope of the various embodiments of the CLAP.

[0119] The Guru program, in some embodiments, may comprise of several levels, each of which corresponds to the number of influence points earned by a user. For example, in one implementation, the Guru program may include a base level and levels 1-5. In some implementations, any user may achieve the base level, but in order to ascend to levels 1-5, a user may need to enroll or opt in to the Guru program. The base level may, in one implementation, be achieved by acquiring a follower, and may be maintained, without opting in to the Guru program, so long as the user has at least one follower. User who has opted in to the Guru program, on the other hand, may earn influence points and be eligible for levels 1-5. In one implementation, a Guru program user's current level may be determined based on the total influence points earned during the previous 30 day period. In a further implementation, the current level may be calculated periodically, or in a daily basis. As such, a user's influence level may fluctuate periodically, or daily, depending on the total influenced points amassed. Further, when a user stops earning influence points, his or her influence level may steadily decline until the base level is reached. A user may also revert back to the base level, provided there is at least one follower, when he or she opts out of the Guru program. In some implementations of the Guru program, a user's influence points may not be lost or may not expire so long as the user is a member of the CLAP. In a further implementation, this may be true even for those users who opt out of the Guru program, but remain a member of the CLAP service. An exemplary level and point correspondence is illustrated below in Table 4. Depending on the distribution of users, the influence point ranges may be adjusted to achieve a desired distribution (e.g., a normal distribution, as opposed to a skewed distribution).

LEVEL	INFLUENCE POINTS	PREMIUM STATUS
1	10-39	None
2	40-89	Gem
3	90-159	Silver
4	160-249	Gold
5	250-449	Platinum
6	450+	Executive Platinum

[0120] As discussed above, a user's current influence level may be based on the total influence points earned during the previous 30-day period. In one implementation, the Guru program may also track the user's lifetime influence points. In such an implementation, a user with 10,000 lifetime influence points may be a level 1 Guru if he or she has influenced only between 10-39 plays during the preceding 30-day period. In other implementation, a combination of points earned over a lifetime and over a preceding number of days may be utilized to determine the current influence level of a user. In a further implementation, Gurus may be accorded a premium status based on the accrued influenced points. For example, Gurus may be accorded platinum, gold and silver status based on the accrued influence points.

[0121] As discussed above, Gurus contribute towards a more social environment, generate value, and generally uplift the CLAP experience. In return, Gurus enjoy benefits, recognition and/or rewards. All Gurus in the CLAP community may be recognized by the level they have achieved on their profile page and with a level-based icon on their image. Further, a user, by opting into the Guru program and agreeing to allow CLAP to feature himself or herself as music influencers throughout the CLAP experience, may become eligible to be featured in one or more product areas such as the "Music Trends" page, "Music Genre" page, "Music Artist" page, "Music Album" page and/or the like. In a further implementation, the higher the Guru status, the more likely it may be for the Gurus to be featured in relational search results, which may in turn lead to acquiring more friends, followers and influence. Being featured in various product pages, searches

and/or interfaces of the CLAP may further reinforce the user's status as a Guru and music influencer. It may also increase opportunities for the Guru to influence other users' plays and earn influence points.

[0122] Gurus may also be eligible to earn badges for their influential activities. The badging may be tiered such that it may become progressively more difficult to complete activities and earn badges as a user rises through the tiers. Example badges may include level badges (e.g., level 1 badge, level 2 badge, etc.), follower badges (e.g., 10 followers, 100 followers, 1000 followers, etc.), influenced play badges (e.g., 10 influenced plays, 100 influenced plays, etc.), and/or the like.

[0123] In some implementations, Gurus may receive perks and incentives for their social influence. Through relationships with music labels and other rights holders, Gurus may have opportunities to earn external rewards and recognition related to the content they most heavily influence. Some non-limiting example of rewards include, merchandise, concert tickets, autographed memorabilia, artist meet and greets, and/or the like. Other examples of rewards include invitation to Guru-only offline and online events, such as concerts by artists they support, incentives provided by partners to further promote plays of their repertoires in earning more royalties, monetary incentives such as commissions on a play count basis, and/or the like. Furthermore, in some implementations, the premium status Gurus may be more highly targeted by partners for promotions of artists and songs than others.

[0124] FIG. **8** shows a logic flow diagram illustrating the Gurus rewarding component in some embodiments of the CLAP. In one implementation, for example, the process may begin at **802**, with the tracking of user engagement in various activities relating to social, usage, influence, and/or the like at **804**. Example activities include posting to other users', friends', and followers' comment streams, recruiting new followers and/or friends, suggesting, referring and/or recommending music to friends and/or followers, sharing playlists for followers, friends and/or other, increasing music consumption, reviewing contents (artists, albums, playlists) and posting to content comment streams, using micro-blog "beats" about their music and related interests, including providing referrals, recommendations for new music to friends and/or the community. These posts or beats may be limited to 140 characters and may include a bit of text and a link. Comments about content (e.g., artist, album, track, etc.) found in the universal music library may automatically hyperlink to contextual information about that product and ways to further explore that subject.

[0125] At **806**, a determination as to whether a subject user is enrolled in the Guru program may be made. If the user is not enrolled or has not opted in the Guru program, the user may be reminded at **808** to enroll in the Guru program for rewards and/or status benefits and may conclude the process at **810**. However, if the user is enrolled in the Guru program, a determination may be made at **812** whether the user is also engaged in activity whether social, usage and/or influence. In one implementation, each activity may have a corresponding value in status points. At **816**, type of activity that the user may be engaged in may be identified and the number of status points associated with the activity type may be determined. For example, a user engaged in recommending playlists that are eventually downloaded and/or played by another user may obtain 2 status points.

[0126] In one implementation of the Guru rewards program, the CLAP may increment the user's status points by the determined value at **820**. A determination may then be made, based on the incremented status points, whether the user is eligible for a status upgrade at **822**. For example, the user may have accumulated 24 status points before acquiring 2 more status points in this instance, causing the total status points to exceed, for example, a cut-off of 25 for silver status. In one implementation, if the user is eligible for a status upgrade, the user's status may be upgraded to the next level at **830** and the user's profile may be updated to reflect the change in status at **824**. On the other hand, if the user is determined to be ineligible for a status upgrade, the user profile may be updated at **824**, without any status upgrades. The user profile may include published user profile and/or profile in one or more user databases at the backend that is not published.

[0127] In another implementation of the Guru rewards program, the CLAP may implement a rewards program that encourages users to engage in a variety of activities, and not merely the kind of activities that have previously had some success. For example, at **826**, the CLAP may determine whether the points accumulated for the identified activity exceeds a threshold N. If the threshold is not exceeded, the user's status points may be incremented at **820**. However, if the threshold has exceeded, the determined status points may be adjusted at **1228** before incrementing the user's status points by the value of the adjusted status points. Adjustment may include an adjustment by a factor less than or greater than 1.

CLAP User Interfaces

[0128] FIG. **9** is a schematic view of the CLAP application interface in one embodiment. A user launching the CLAP application may encounter an interface **900** that provides a quick overview of information. The interface **900** may include **905** player control panel, menus **910**, a media explorer panel **915**, a media display panel **920**, information panel **925**, and/or the like. In one embodiment, the menu **910** may include menu items such as file, edit, view, controls, share, help, etc. The player control panel **905** may include various controls such as play controls **905***a*, logos/artwork **905***b*, current track information **905***c*, track position bar **905***d*, volume control bar **905***e*, shuffle button **905***f*, repeat button **905***g*, search bar **905***h*, next track information **905***i*, and/or the like. The media explorer panel **915** may include various selectable items such as music universe **915***a*, Guru network **915***b* and offline music **915***c*. Selection of the music universe item **915***a* may facilitate the display of information relating to media in the CLAP catalog. As shown in FIG. **9**, wherein the music universe **915***a* has been selected, the media display panel **920** shows items such as music trends **920***b*, playlists **920***c*, Gurus **920***d*, decade mixes **920***e*, genre mixes **920***f*, and any other links to organized content in the music universe. On the right of the interface **900** is an information panel **925** which displays additional information based on selections of items such as **920***b-f* in the media display panel **920**. The information display panel **925** may include several tabs, for example tabs **925***a-e*. Information corresponding to the selected tab and the selected item **920***b-f* may be displayed in the information display panel. For example, when music trends items **920***b* is selected, the information panel **925** is updated to display artists in one tab **925***a*, albums in **925***b* tab, tracks in **925***c* tab, new releases in **925***d* tab and just added in **925***e* tab. Similarly, when the Gurus item **920***d* is selected, the information panel **925** is updated to display information relat-

ing to the Guru network. For example, the selection of the Guru network may include a listing of all or some Gurus having the most followers.

[0129] The media explorer **915** also include the item Guru network (or "community") **915***b*. When the Guru network **915***b* is selected, a community (or Guru) display panel **1000** is displayed, as shown in FIG. **10**. The community display panel may include a profile area **1005** that may identifying user information **1005***a*, profile image **1005***b*, and a "my profile" button **1005***c*. The "my profile" button **1005***c* display the user's profile information and is discussed in further detail in FIG. **13**. The community display panel may also include Gurus panel **1010** which may include Gurus suggested by the CLAP, Gurus currently followed, and/or the like. The right of the community display panel **1000** may include one or more panels **1015-1025** displaying activity streams. In addition, the community display panel **1020** may also include a search bar **1020** which may be used to search for Gurus.

[0130] FIG. **11***a* is a schematic view of the CLAP application interface that is displayed in response to the selection of my profile button **1005***c* in one embodiment. The profile area **1105** may include buttons **1105***a-c* to edit profile information, view updates and view suggested Gurus respectively. The profile display panel **1100** may also include shared playlist area **1110** that provides an overview of the playlists that a user has shared with others, or has made available for sharing. Next to each listed shared playlists, the user may click the see tracks button **1110***a* to explore the tracks in the playlist. Also provided in the profile display panel **1100** is the Guru information area **1120** which may list Gurus that the user follows, as well as other users following the user. On the right side of the profile display panel **1100**, an activity stream display panel **1115** may be provided for displaying posts, beats, comments or other messages exchanged between the user and others in the community.

[0131] FIG. **11***b* is a schematic view of the CLAP application interface that is displayed in response to the selection of edit profile button **1105***a* in the profile area of the profile display panel **1100** in one embodiment. The edit profile panel **1125** allows the user to edit and/or update identifying information such as name, location, email address, social networks, etc. The user may also select options to integrate with other social networks such as TWITTER and FACEBOOK. In addition, the user may also set privacy options of his or her profile. For example, the user may make his or her profile public which would let anyone to find the user (e.g., by search), view his or her profile, post items to his or her comment/activity stream, send messages, and/or the like. The user may also specify whether or not to use an approval process to let other users follow him or her. These facilities may no longer apply if the profile is set to be private. Additionally, the interface may also include facilities to set preferences for the activity stream **1125**. Examples of such preference settings may include options to select information displayed in the activity stream **1125** in FIG. **11***a* to those involving friends having a specified degree of separation, Gurus, followers, and/or the like.

[0132] FIG. **12***a* is a schematic view of one embodiment of the CLAP application interface. As shown in FIG. **12***a*, the media display panel **1205** displays information relating to a selected item (e.g., artist Pink Floyd). The media display panel **1205** may display identifying information **1210**, and a listing of tracks **1225** pertaining to the selected item **1210**. The user may have the option to create a magic playlist based on the selected item **1210** by clicking on the magic playlist icon **1215**. In addition, the user may also share the tracks **1225** with other users by clicking on the share icon **1220**.

[0133] Next to each track in the listing **1225**, is provided a track location indicator **1230**. The indicator **1230** may specify using color or other indicia whether or not the track is available in the local/offline library. As shown in the figure, each of the tracks in the listing **1225** may be made available offline by clicking on the make all available offline button **1235**. The CLAP, in response to the request to make all tracks available offline, may identify the tracks that are not present in the local or offline library and download them to the user device. Instead of downloading all of the tracks listed at **1225**, the user may also individually select one or more tracks for downloaded. Of course, the user may also play, pause or skip these tracks, regardless of whether they reside in the local library or on the cloud.

[0134] FIG. **12***b* is a schematic view of one embodiment of the CLAP application interface. When the magic playlist icon **1215** shown in FIG. **12***a* is clicked, the media display panel **128***o* is displayed. Here, the display panel may show the magic playlist **1250** that was created in response to the request. The playlist may identify the creator at **1255** and include options to create another magic playlist or share the playlist with other users. At the media explorer bar **1245**, the listing of playlists may be updated to include the newly generated magic playlist. Similar to FIG. **12***a*, the media display panel may also list the tracks included in the magic playlist at **1260**. In addition, on the right side of the interface, in the information panel **1265**, information such as people following the playlist, people who have downloaded the playlist, those who have shared the playlist with others, and/or the like may be displayed.

[0135] FIG. **13** is a schematic view of one embodiment of the CLAP application interface. Here, an artist has been selected as shown in the media display panel **1305**. A list of selected tracks (e.g., by popularity) or all tracks related to the artist may be listed in the track listing area **1310**. Further, the listing area may also identify the tracks that are located in the local library, and their total number using the location indicator **1310***b*, and the item **1310***a*. On the right side of the interface, is the information panel **1320**, which may provide additional information on the item displayed on the media display panel **1305**. For example, as shown in FIG. **13***a*, the artist's main releases are shown in the information panel under the albums tab **1320***b*. Any of these albums may be made available offline, used for creating a magic playlist, bookmarked or shared using the options in the popup menu **1320***a*. Information about the artist may also be displayed by selecting the about tab **1320***c*. FIG. **13***b* is a schematic view of one embodiment of the CLAP application interface of FIG. **13***a* after the playlists tab **1320***d* has been selected. The playlist tab **1330** may provide a listing of all or selected playlists **1340** that include or are similar to the selected item (e.g., artist Janelle Monae) **1335** ordered by relevance, popularity, or any other criteria. Similarly, FIG. **13***c* is a schematic view of one embodiment of the CLAP application interface of FIG. **13***a* after the item **1320***f* has been selected. As in the previous figures, the media display panel **1345** displays identifying information on the selected item, as well as a track listing. In addition, the information panel **1350** on the right may display playlists that are related to the selected item, and ordered using or one more criteria such as popularity, published date, relevance, and/or the like.

[0136] FIG. **14** is a schematic view of one embodiment of the CLAP application interface. The interface **1400** shown in FIG. **14** is rendered in response to the selection of the now playing item **1405***a* in the media explorer panel **1405**. The media display panel **1410** may provide a listing of tracks that are in a play queue. Such a queue may include tracks that were previously played, is currently being played and are in queue for playing next. For example the track **1410***a* has been selected, and is the track being played as indicated by the indicator **1410***b* and by the player control panel **1420**. Additionally, the information display panel **1415** on the right is updated in response to the selection of track **1410***a*. The information display panel **1415** may list playlists including the current playing track. Such a listing may be arranged based on one or more criteria such as popularity, preference, recently published, and/or the like.

Discovery User Interfaces

[0137] Referring now to FIGS. **15***a-g*, in one embodiment of the CLAP, there is shown a sequence of player interfaces as the user explores and discovers music related to an item. In general, the user may discover related music by selecting a displayed item, for example, the photo of Van Morrison at the bottom middle of FIG. **15***a*. The CLAP may then generate recommendations of related items based upon the selected item.

[0138] Referring again to FIG. **15***a*, the user may select an artist ("Van Morrison"), an album of the artist ("Astral Works"), and more specifically, a track in the album ("Sweet Thing"). Referring now to FIG. **15***b*, an image representing the track selected may be displayed in a stagelight/spotlight window. Referring now to FIG. **15***c*, images representing selected items **1515***b-e* related to item **1515***a* may be displayed in the stagelight window **1515**. The images displayed in the window **1515** may include an image representing a related track **1515***b*, an image representing an album **1515***c*, an image representing another user **1515***c* and an image representing a playlist **1515***d* that includes tracks **1515***a* and **1515***b*. Referring now to FIG. **15***d*, the user may select the track **1520***a*, in response to which, the stagelight window **1520** is presented. The stagelight window **1520** may display additional information about that selected track **1520***a* and also albums **1520***b* in which the track may appear. Similarly, items **1520***c-f* may also be selected to obtain additional information. In addition, selection of one of the items **15***e-f* may reveal control options to scroll through other images corresponding to other tracks. The user may simply use click through the images.

[0139] Referring now to FIG. **15***e*, the stagelight window **1530** is updated to show the user selection of another item ("Speaking in Tongues") image **1530***a*, causing it to overlay the item **1520***a* image of FIG. **15***d*. Selection herein may include selecting a stagelight button or icon, double click, single click, right click and select stagelight or drag and drop. The user may now be presented with more information about the selected item **1530***a* and the images representing tracks, artists, and playlists related to **1520***c-e* have been replaced with images of an album **1530***b* ("Kicking Television: Live in Chicago"), an artist **1530***c* ("Brian Eno") and another user offering playlists **1530***d* ("Michael") related to item **1530***a*. In addition item **1530***f* may show the tracks in the selected item **1530***a*. Referring now to FIG. **15***f*, the user may select the item **1530***b* ("Kicking Television: Live in Chicago") image from **1530***b*, causing it to replace the item **1530***a* image of FIG. **15***e*. The user may be presented with more information about the stagelit item **1540***a* and one or more updated items including **1540***b*.

[0140] Referring now to FIG. **15***g*, the user may be presented with a time line of their exploration/discovery sequence in the stagelight window **1550**. The top row **1550***a* is for tracks and includes the track item **1520***a* from FIG. **15***d*. The track item **1520***a* was the first selected item in time sequence. The second row **1550***b* is for albums and includes album item **1530***a* from FIG. **15***e*. The album item **1530***a* came second in time sequence, after the track item **1520***a*. Similarly the third item selected in time sequence is the album **1540***a* from FIG. **15***f*. Other rows **1550** *c-e* (in order from top to bottom) are for "artists", "Gurus" (other users like "Tom" and "Michael" with trusted recommendations) and playlists (like "The Sure Thing"). Thus, FIGS. **15***a-g* illustrate an exploration/discovery sequence in which the user is provided facilities, via the CLAP stagelight/spotlight user interface, to discover and play music related to an initial musical selection using recommendations of various sources including related tracks, related albums, related artists and trusted advisors (Gurus).

[0141] FIGS. **16***a-c* are schematic views of the discover stream component in some embodiments of the CLAP. As shown in FIG. **16***a*, the discover stream user interface includes a display window **1600**. The display window further comprises an input text box **1605** where a user may input a content item name such as an artist, album, playlist, track, and/or the like. As shown in the FIGURE, an artist "depeche mode" is input as a content item seed, and an album view **1605***d* is selected. The display area **1625** displays a stream of albums of artists similar to "depeche mode." Instead of an album view, users may select other views such as artists or tracks. When a content item seed is provided to the discover stream component, the recommendation engine may take the provided content item seed and generate other content items that are related to the seed. The discover stream component may then stream the related content items, i.e., albums **1615** from the left side of the screen to the right side. At any time a user may click or select an album **1630** as shown in FIG. **16***b*. The selection may cause the discover stream component to take the selected album **1630** as an input to the recommendation engine to regenerate an album stream related to the selected item **1630**. The selected item may stay stationary while related albums may continuously stream from the left to the right. A user may also hover a curser over an album **1625** to display control icons such as play, stop and pause. In one implementation, the discover stream component may allow the user to play a representative track which may include, without limitation, a track selected at random, the most popular track, a track determined to be most likely of interest to the user by the recommendation engine, the most frequently played track, the most highly recommended track, and/or the like.

[0142] In one implementation, the discover stream interface may also keep track of and display all selections made by a user. The bread crumb panel shown in FIG. **16***b* shows an icon **1635** of the selected album. Further, FIG. **16***c* shows a bread crumb panel **1640** that includes six album selections made by the user. The bread crumb panel establishes a sequential history of the selections made by the user. In one implementation, the user may select any of the album icons and may perform a variety of operations including, but not limited to: play a track, download the album, create a magic

playlist, share the album, download the album to a cache memory, download the album to your local client library, and/or the like.

[0143] FIGS. **17***a-f* are schematic views of the discover lens component in some embodiments of the CLAP. As in the discover stream component of FIGS. **16***a-c*, a user may enter a content item seed as an input (for e.g., "depeche mode"). When a track view is selected, the recommendation engine may generate tracks (e.g., track **1710**) that are similar to the track **1715** by the artist depeche mode. As shown in FIG. **17***a*, the related tracks including track **1710** may be rendered around the seed track **1715**. In one implementation, a user may select the seed track **1715** to play the track or add the track in a playlist, download to the client, share with other users, and/or the like. In some implementations, the seed track may be selected based on the user-provided track input. In other implementations, the seed track may be derived from the user provided content seed such as an artist name, album name or a playlist name. In a further implementation, a representative track may be selected for the track view. The representative track may be a track selected at random, the most popular track, a track determined to be most likely of interest to the user by the recommendation engine, the most frequently played track, the most highly recommended track, and/or the like.

[0144] When the user selects the track **1710** from FIG. **17***a*, the discover lens **19** component interface may be updated to display a collection of tracks as shown in FIG. **17***b*. Here the nine tracks from FIG. **17***a* are displayed, with track **1715** being displayed in a bounding box. Further, track **1710** is also displayed in a bounding box. In one implementation, a track that was previously selected or is currently selected may be displayed in a bounding box. When track **1710** is selected, the recommendation engine generates related tracks **1710***a-e*. The user may then continue to select any of the displayed tracks (e.g., track **1710***d*) and obtain more related tracks. In this way, the discover lens component may facilitate a visual lens like discovery of related content items. Furthermore, as shown in FIG. **17***c*, the discovery lens component may also map out the discovery path traversed by the user. For example, from FIG. **17***c*, it is clear that the user selected track **1715**, track **1710**, track **1725** and track **1735** to generate the tracks displayed. In one implementation, the discovery path and the related tracks displayed may provide an indication of how far the user has traveled from his or her initial selection of track **1715** to the last selected track **1735**. The distance between track **1715** and track **1735** may visually show the degree of relatedness between the tracks.

[0145] In some implementation, the discovery lens component may support clicking and holding the cursor at a location **1745** in the display window farther away from the center (the seed track **1715**), as shown in FIG. **17***d*. The action may then result in a cluster of related tracks **1750** shown in FIG. **17***e*. The cluster **1750** is not only visually farther out from the origin (the seed track **1715**), but also in terms of relatedness with respect to the seed track **1715**.

[0146] In one implementation, the discover lens component interface may include facilities for retrieving and displaying a prior search. FIG. **17***f* displays a back/forward icon **1755** that may be utilized to go back or forward a pre-defined number of past searches, even when the searches were carried out in separate sessions. In one implementation, these search results may be stored as a series of numbers in a grid based format, such that with the location of the grid and identifier of the track (e.g., track ID), the entire display of related tracks may be accurately regenerated and displayed for the user. In another implementation, these search results may be stored in an array, which could be used for reproducing the lens map **1765**. In a further implementation, the discover lens component may include a share icon **1760**. Via the share feature, the user may share the lens map **1765** with other users. In one implementation, the lens map **1765** including the display of related tracks may be shared as an image file (e.g., JPEG, PNG, TIFF, BMP, and/or the like), as a reproducible list, in an email among others. In the case of the reproducible list in an email or in comment stream, the recipient may load the file in the discovery lens component interface and obtain the lens map **1765**. Various other features such as zoom **1770** and maximize **1775** may also be available for the user to customize his or her viewing preferences.

[0147] FIGS. **18***a-c* are schematic views of the discover stacking component in some embodiments of the CLAP. In some implementations, the discover stacking component interface may include a display window **1800**, similar to that of the discover streaming and lens components. A user may enter a content item name in the input entry box **1805**. In this example, an artist name "depeche mode" is entered. The recommendation engine may then take this input and run its algorithm to identify related albums **1815** (or songs, playlists, artists, etc.). Thumb nails of the identified related albums **1815** may be displayed on the window. In some implementations, the thumb nails of the album covers may shrink and grow to gain the user's visual attention. In one implementation when a user selects an album **1810** (shown in FIG. **18***a*), the selected album may be brought to the forefront and the previously selected album **1825** may be added to the album stack (shown in FIG. **1**A). When the album **1810** is selected, related albums **1820** may be identified and displayed as shown in FIG. **1813**. The user may repeatedly select various albums as he or she is discovering music. As the user continues to select albums, the discover stacking component may continue to provide new recommendations as well as add the last selected album continues to the stack **1840**, making the stack **1840** grow as shown in FIG. **18***c*. In a further implementation, the discover stacking component may allow the user to peel away albums from the stack **1840** in a manner similar to flipping through a stack of record. In some implementations, facilities for sharing and publishing may be provided to the user.

[0148] FIGS. **19***a-g* are schematic views of the molecular discovery component in some embodiments of the CLAP. In some implementations, the molecular discovery component interfaces may facilitate content discovery related to their selection. In FIGS. **19***a-d*, the center molecule **1905***a-d* respectively represents the selection. It is also the user's entry point as well as search subject. Surrounding the center molecule are top related results by category for album **1910***a-d*, artist **1915***a-d*, song **1920***a-d*, Guru **1925***a-d*, and genre **1930***a-d*. The user may follow a new path of discovery by dragging any result in the surrounding categories and dropping it to the center, where that center molecule becomes the new subject and is surrounded by top results directly related to the category.

[0149] FIG. **19***b* shows the song molecule **1920***a* of FIG. **19***a* dragged and dropped into the center molecule **1905***a*. As a result of this action, the song molecule **1920***a* becomes the center molecule **1905***c* as shown in FIG. **19***c*. The molecular discovery component then generates top results for the sur-

rounding categories **1910***c*, **1915***c*, **1920***c*, **1925***c* and **1930***c* based on the search subject **1905***c*. FIG. **19***d* shows the resulting display after the song molecule **1920***c* (from FIG. **19***c*) is dragged and dropped into the center molecule **1905***c*. As shown in FIG. **19***d*, the top results for the surrounding categories **1910***d*, **1915***d*, **1920***d*, **1925***d* and **1930***d* are also refreshed to reflect the new search subject **1905***c*.

[0150] In some implementations, the molecular discovery component may include facilities for sharing, adding to playlist, downloading to local client, publishing, creating a magic playlist and/or the like. For example, in FIG. **19***e*, several bubbles for share **1940**, add to playlist **1950** and download **1945** are shown surround the center molecule **1955**. The bubbles may be displayed when the user selects or hovers over the center molecule. The center molecule may then be instantly shared, added to a playlist or downloaded by an action such as drag and drop. FIG. **19***f* shows a schematic view where the center molecule **1955** is being dragged and dropped into the bubble for download **1945**. As a result of this action, the molecular discovery component may download the item in the center molecule in the user's local client. In some implementations, the center molecule and/or surrounding molecule items that are non-local may be automatically cached in the local client.

[0151] In some implementations, the molecular discovery component may track the search subjects in the center molecule over time and display an interactive historical crumb trail as shown in FIG. **19***g*. The historical crumb trail **1980** may include a scroll bar **1975** which may be scrolled forward or backward to view the search subjects **1960**, **1965** and **1970**. In a further implementation, the historical crumb trail may not only show the historical path taken by the user, but also a predictive path forward. The predictive forward path may be determined by the music intelligence component based on the user's past listening history, preferences, interest and/or social graph, Gurus, and/or the like.

[0152] In some other implementations, the molecular discovery component may also include an interface for sharing playlists or other content items with other users. An example interface is illustrated in FIG. **19***h*. The playlist items **1990** are shown on the left, while a list of the user's friends **1995** are shown on the right. In some implementations, the interface may include an option to select other content items such as artists, albums, songs, libraries, collections, and/or the like. Similarly, in some implementations, there may be granularity in selecting friends (e.g., all friends, n degree of friends, family, co-workers, and/or the like). In the implementation shown here, the user may select a playlist item **1998** and may drag and drop the playlist item **1998** to any friends he or she desired to share his or her playlist with.

Mobile Application User Interfaces

[0153] FIGS. **20***a-n* are schematic views of the mobile application user interface in some embodiments of the CLAP. FIG. **20***a* shows artist view of the collection interface **2005**. The interface **2005** includes a navigation bar **2005**, buttons **2005***c-d*, a display panel **2005***e* and a tab bar **2010**. When the collection icon **2010***a* is selected along with one of the artist, album or track button, a table view of the selected artist, album or track is displayed in the display panel **2005***d*. Referring now to FIG. **2013**, when a user performs a swiping action over an artist item (e.g., Madonna), the artist view may expose functionality such as play/pause **2015***a*, download **2015***b*, add to playlist **2015***c*, share **2015***d*, and/or the like.

[0154] Referring to FIG. **20**C, when a search icon **2025***a* in the tab bar **2025** is selected, the search navigation bar **2020**, along with buttons **2020***a-d*, a search bar **2020***e* and a table view **202** of may be rendered. As shown in the figure, when the artist button **2020***a* is selected, and a search item "Boston" is entered in the search bar **2020***e*, the results **202** of may be displayed in table or other views. As shown in FIG. **20***d*, a similar search may be conducted for the album view **2030***a*. For example, an album name "Elvis Presley" may be entered in the search bar **2030***b*, and search results **2030***c* in album view may be displayed in the display pane.

[0155] Referring to FIG. **20***e*, when a user selects an artist ("Boston"), the interface **2035** may be rendered. Options for several views including album **2035***b*, tracks **2035***c*, biography **2035***d* and related items **2035***e* may be provided. As shown in FIG. **20***e*, when the album view **2035***b* is selected, information about the artist may be displayed in area of the display panel **2036**. In addition, options to view saved, local and beyond albums may be available. When the saved view **2035***f* is selected, the display area **2038** displays only those albums that have been saved or bookmarked. When the local view **2035***g* is selected, the listing in **2038** includes only the albums that are available locally, the albums being stored in the memory of the mobile device. Similarly, when the beyond or cloud view **2035***h* is selected, the entire list of albums released by the selected artist ("Boston") and available in the CLAP catalog may be displayed in table view in the display area **2038**.

[0156] FIG. **20***f* is the schematic view of the interface **2035** from FIG. **20***e* after the selection of the related view **2035***e*. This related artists view **2040** may display in table view or other format, artists that are similar to the selected artist ("Boston") in a display panel portion **2040***a*. In addition, influences of the selected artist ("Boston") may also be provided in another display portion **2040***b*.

[0157] Referring to FIG. **20***g*, when a user in an album view swipes an album ("Oracular Spectacular"), the interface **2045** is displayed. The user has the option to go back to the last view by tapping on the last view button **2045***a* (last view was artist view and is labeled by the name of the artist "MGMT"). The user may also be provided with options to select track view **2045***b* which displays identifying information related to the selected album in a display panel portion **2045***h* and a control bar **2045***d* for play/pause, download, add to playlist, share and other functionalities. Additionally, a location status bar **2045***e* may also be provided, from where the user may select saved, local or beyond/cloud views of the tracks in the selected album. For example, as seen from the status bar **2045***e*, 3 tracks from the album have been saved or bookmarked, 3 tracks of the album are available locally, and similarly, 10 tracks of the album are available in the cloud catalog. The tracks corresponding to the status bar selections may be displayed in table view in the display portion **2045***h*. The description view **2045***c* may display information relating to the album and/or artist. Referring now to FIG. **20***h*, the interface **2050** displays an album track list in cloud view similar to FIG. **20***g*. Also shown next to track **4** is a caching icon **2050***a* indicating the track being played/cached.

[0158] Referring now to FIG. **20***i*, as indicated by the navigation bar and the selected playlists icon **2055***d* in the tab bar, a playlists view **2055** is displayed. The interface includes a listing of playlists **2055***b* ("my playlists") and a listing of magic playlists **2055***c*. For each playlist, information such as sharing status (public/shared or private) and the number of

tracks included may be identified. There may also be an option for the user to add or create playlists using the icon **2055***a* on the top left corner of the interface.

[0159] Referring now to FIG. **20***j*, the user may explore any of the playlists displayed in the playlists view of FIG. **20***i*. Here, the playlist "80s Dance" has been selected. In this view, a saved option **2060***a*, a local option **2060***b* and a beyond/cloud option **2060***c* on a location status bar are displayed for selection. In addition, the cloud view of the tracks in the playlist is shown in the display panel portion **2060***d*.

[0160] FIG. **20***k* is a schematic view showing the now playing track in one embodiment. The interface **2065** includes a navigation bar **2065***a*, with a back button **2065***b* to go to the last selected item and a playlist icon **2065***c* which when selected causes the interface shown in FIG. **201** to be displayed. FIG. **201**, as discussed, shows the playlist view with the track **2070***b* that is up next highlighted.

[0161] Referring to FIGS. **20***m-n*, some embodiments of the mobile application platform may include settings for configuring the download manager and user settings. Shown in FIG. **20**M is the download manager interface **2075**. The download manager interface may include options to enable or disable downloading by sliding or tapping button **2075***a*. Additionally, the current download status may also be identified at **2075***b*. Examples of status include downloading, completed, not started, paused, canceled, error etc. The interface may also display a download queue **2075***e* that lists the tracks that are in queue for downloading. As shown in FIG. **20**M, the item that is currently being downloaded is identified by the icon **2075***f*. The download manager interface may also include options to edit the download queue using the edit button **2075***c*. Editing may including changing the order of the tracks in the queue by simple swiping motions, deleting/canceling selected tracks from the queue, etc. Additionally, the cancel all button **2075***d* may be used for canceling the download of all the tracks in the queue.

[0162] FIG. **20***n* is a schematic view showing the user settings management interface **2080**. The settings management interface may be launched from the settings option **2080***e* on the tab bar. Using this interface, a user may configure his or device to manage synchronization, memory usage, network usage, and/or the like. As shown in FIG. **20***n*, the sync now option **2080***a* may be selected to synchronize playlists between the device and the cloud or another device. For example, the user may create a playlist in his or her mobile device. This playlist may be saved locally in the device memory until the user selects the sync now option. As a result of the syncing, the playlist may be saved in his or her beyond/cloud account. Additionally, the playlist now stored in the beyond/cloud account may be downloaded/synced with one or more LDs.

[0163] The settings interface may also be utilized to configure cellular data use. Cellular data plans subscribed by users may vary. While some users may have unlimited data plan, others may have a limited monthly plan (e.g., 2 GB/month) and may like to limit the amount of data downloading. In such a situation, the user may set on or off the remote browsing **2080***b*, remote playback **2080***c* and track downloads **2080***d*.

Content Identification Component

[0164] In one embodiment of the CLAP, facilities may exist for incorporating a user's existing collection of music tracks to the CLAP music library ("beyondization"). The user may opt in to the beyondization service to obtain copies of some or all of his or her existing tracks in one or more formats at a variety of bitrates selectable by the user. The user may also choose whether to hold on to their existing files, back them up, or delete them (e.g., to save disk space). Benefits offered by beyondization may include opportunity to replace low quality tracks with higher fidelity versions and/or replace tracks that have missing metadata. In one implementation, tracks that have been beyondized may be shared or duplicated as only LDs may be able to play tracks obtained from the CLAP library. However, in other implementations, when the user chooses to not beyondize, the CLAP may not support sharing, duplicating, playlist creating, etc., activities utilizing the CLAP.

[0165] FIG. **22***a* is a data flow diagram of an example content identification component in some embodiments of the CLAP. Content identification may be initiated at the devices **2202** with existing audio or other media files. In one implementation, a request to identify content items from the device **2202** may be sent to the music intelligence component **2206**. The request message may packaged as an HTTP(S) POST message and may include information such as acoustical fingerprints and metadata of each file on the client. An example content identification request message **2204** may be in XML format substantially in the following form:

```
POST /beyondization.php HTTP/1.1
Host: www.websitename.com
Content-Type: Application/XML
Content-Length: 1306
<?XML version = "1.0" encoding = "UTF-8"?>
<Beyondization>
        <Username>dingdong555@gmail.com</Username>
        <Timestamp>20xx-02-22 15:22:43</Timestamp>
        <ClientDetails>
            <ClientIP>192.168.23.126</ClientIP>
            <ClientType>smartphone</ClientType>
            <ClientModel>HTC Hero</ClientModel>
            <OS>Android 2.2</OS>
            <AppInstalledFlag>true</AppInstalledFlag>
        </ClientDetails>
        <TrackDetails>
            <FingerprintID1>ABF6789</FingerprintID1>
            <FingerprintID2>ASF6H88</FingerprintID2>
            .
            .
            .
            <FingerprintID78>A7GHK33</FingerprintID78>
        </TrackDetails>
        <MetaData>
            <Track1>
                <genre>blues</genre>
                <artist>Blue man</artist>
            </Track1>
            .
            .
            .
            <Track78>
                <album>Nevermind the Wishkaws</album>
                <songtitle>smells like</songtitle>
            </Track78>
        </MetaData>
</Beyondization>
```

[0166] In one implementation, the music intelligence component **2206** may perform acoustical analysis on the fingerprints and metadata matching of the obtained files at **2208**. At **2212**, for each identified file, the music intelligence component may determine the corresponding track ID. In a further

implementation, for files that are not recognized, or that are not present in the CLAP catalog, a new track ID may be generated. In a further implementation, one or more metadata database and/or tables may be periodically queried for identifying the unidentified files. In one implementation, the generated track IDs may be sent in a HTTP(S) POST response message **2214** to the client **2202** for play count recording and/or other customer usage recording purposes. An example response message **2214** may be in XML format substantially in the following form:

```
POST /beyondization.php HTTP/1.1
<XML>
<Beyondization>
        <Username>dingdong555@gmail.com</Username>
        <Timestamp>20xx-02-22 15:22:43</Timestamp>
        <ClientDetails>
            <ClientIP>192.168.23.126</ClientIP>
            <ClientType>smartphone</ClientType>
            <ClientModel>HTC Hero</ClientModel>
            <OS>Android 2.2</OS>
            <AppInstalledFlag>true</AppInstalledFlag>
        </ClientDetails>
        <TrackInfo>
          <TrackID1>5768689</TrackID1>
          <TrackID2>3457689</TrackID2>
          .
          .
          .
          <TrackID78>6786889</TrackID78>
        <MetaData>
          <Track1>
            <type>audio</type>
            <name>track12.m4a</name>
            <size>64KB</size>
            <genre>blues</genre>
            <album>Taste</album
            <artist>Blue man</artist>
            <length>3 min 22 sec</length>
            <ranking>678</ranking>
            <year>1987</year>
            .
            .
            .
          </Track1>
          .
          .
          .
          <Track78>
            .
            .
            .
          </Track78>
        </MetaData>
</Beyondization>
</XML>
```

[0167] FIG. **22***b* is a logic flow diagram illustrating an example content identification in some embodiments of the CLAP. The beyondization may begin at **2205** with the search for media files in a user's client device at **2210**. In one implementation, the search may be conducted only after verifying that the user has opted in to the beyondization service. The verification may be performed by requiring the user to confirm beyondization (e.g., click to confirm beyondization) or informing the user of the option to beyondize (e.g., want to beyondize your tracks?). At **2215**, the media files may be analyzed. The analysis may include aggregating the media files, examining the files for information for metadata, bitrates, and/or the like. At **2220**, each media file may be identified using information such as filename, ID3 tag or other metadata containers, hash value, acoustic fingerprint, and/or the like. If any media file is unidentified at **2225**, the user may be prompted to enter the file information or skip the file at **2230**. In one implementation, if a media file is not in the catalog as determined at **2235**, the fingerprint and/or other identifying information relating to the file may be logged in one or more user media databases and/or tables at **2240**. The media file may then be included in the user's local or offline media library at **2245**. At **2250**, some media files may also be found in the catalog. Those media files may be analyzed to determine whether the catalog version of the file is of higher fidelity than the local version at **2255**. Similarly, if all media files are also in the catalog as determined at **2235**, a further determination as to whether the catalog version of the file is of higher fidelity than the local version may be made at **2255**. If the catalog version of the file is of higher fidelity, the user may be prompted to confirm the replacement of the local file with the higher fidelity version from the catalog at **226***o*. If the user confirms the replacement at **2265**, the media file may be replaced with the catalog version at **2270** or the local version of the file is determined to be of higher fidelity as determined at **2255**, the fingerprint and/or other identifying information relating to the file may be logged in the user media database and/or tables at **2285** for inclusion in the user's offline or local library at **2290**. In one implementation, the user may decide to obtain a higher fidelity version of the file while keeping his or her personal copy of the file at **2275**. In this case, the higher fidelity file may be downloaded and the user's personal copy may be backed up.

Content License Acquisition Component (CLAC)

[0168] The CLAP in its effort to grow its collection and its user base seeks to improve services and facilities as well as add media that users may want to consume. In one embodiment, the CLAP employs crowd sourcing to automatically detect and initiate acquisition of rights for tracks catering to users' interests.

[0169] FIG. **23** is a logic flow diagram illustrating an example content license acquisition component (CLAC) in some embodiments of the CLAP. The process may start **2305** at **1402**, with the receiving or retrieving of play count data at **2310**. The play count data may be initially received from LDs that periodically or in real time report to the servers play data relating to each played tracks. At **2315**, the play count data may be analyzed to determine any unlicensed tracks. The analysis may include, for example, examination of the tracks including track ID or other identifying information relating to the track. The analysis may further comprise checking the CLAP media databases and/or tables to determine if corresponding records of the tracks are present. Based on the analysis, a determination may be made as to whether any of the tracks being reported are unlicensed tracks. In one implementation, such tracks may be uniquely resolved within the CLAP collection and may be provided a track ID upon identification. If there are no unlicensed tracks identified at **2315**, the process may end at **2345**. However, for the identified unlicensed tracks, the CLAC may determine if the unlicensed tracks are popular with users, and if so, may automatically initiate a rights acquisition process. In one implementation the CLAC may obtain aggregate play count data for each unlicensed track over a period of time (e.g., a week, month, etc.). The play count data may be aggregated from one or more users of the CLAP. In a further implementation, the CLAC may also obtain aggregate play count data for all

licensed tracks from one or more users of the CLAP for the same period of time. The CLAC may also retrieve one or more rights acquisition trigger rules to evaluate the obtained aggregate play count data. In one implementation, rights acquisition may be triggered when the aggregate play count for the licensed track exceeds a threshold (e.g., play count greater than 500. In another implementation, any unlicensed tracks with a non-zero play count may trigger rights acquisition. In yet another implementation, rights acquisition may be triggered when the aggregate unlicensed track play count is greater than a predefined percentage of the aggregate play count data for all licensed tracks. At **2320**, if the rights acquisition is not triggered, the process may end at **2345**. At **2320**, if the rights acquisition is triggered, the CLAC may identify the unlicensed track using, for example, fingerprint, ID3 tag, etc., at **2325**. Using the obtained information, the CLAC may the query a rights database at **2330** to obtain rates, quotes and/or availability for licensing rights. If there is an opportunity for obtaining rights for the unlicensed tracks at **2335**, the CLAC may identify the appropriate rights clearing association such as ESCAP, BMI, SESAC, and/or the like at **2350**. At **2355**, the identified rights clearing association may be requested to provide rates, quotes and/or authorization to add the track to the CLAP catalog. If the request is approved at **236***o*, the added track may be added to the online catalog at **2365** and made available to the users of the CLAP. If at **2360**, the request is not approved, the CLAC may periodically request rates, quotes, and/or authorization to add the track to the CLAP catalog at **2365**. The process may end at **2345**.

[0170] At **2335**, if there is no licensing opportunity available, the CLAC may periodically query rights database for rate and/or availability at **234***o*. In one implementation, the rights database may be constantly updated as more tracks are licensed or are offered for licensing by partners. In one implementation, even if the rights holder cannot be identified, the CLAC may upload the unlicensed tracks to one or more servers. In some implementations, the servers may be selected based on the geographic location where the demand for the track is the highest. For example, if users in the UK are playing an unlicensed track from a music artist, a copy of the track may be uploaded to servers at or near the UK. Royalties and/or license fees for the uploaded but unlicensed track may be collected for payment upon formalization of the licensing arrangement. In this way, licensed tracks may be readily provided to the users of the CLAP, while protecting royalties owed to partners.

License Management Component (LMC)

[0171] The CLAP provides users an unlimited access to a comprehensive catalog of multimedia content to use and share, while protecting the commercial interests of content owners, OEMs and network operators at the same time. This is facilitated by the use of Digital Rights Management (DRM) and other security technologies. The CLAP facilitates protection of the interests of content licensors through secure play count reporting. Through the use of play count reporting, royalties can be calculated and properly attributed to partners. Through DRM, the CLAP ensures that content is only playable in its territory of licensing and protects the interests of OEMs by ensuring that users can only play content on LDs. Within the CLAP ecosystem, the DRM may not inhibit using and sharing content. For example, the DRM may place no limits on duration of a user's music ownership, methods of sharing (including email, portable storage devices, drop box, etc.), the number of licensed devices that a user can own, playback quality, and/or the like.

[0172] In one implementation, the CLAP may leverage DRM technology for supporting a wide range of consumer electronics that can play multimedia content, from minimal "shuffle" type devices all the way up to desktop PCs. In a further implementation, the CLAP DRM may provide portability to a wide variety of operating platforms, including, but not limited to, broadly available OSs (e.g. Windows, Linux, Android) as well as proprietary operating platforms (e.g. Samsung BADA). In some implementation, the CLAP DRM may provide strong support for connected portable devices, to eliminate tethering to PCs through side-loading, enhancing the user experience and making the ecosystem more attractive to mobile operators. In addition, the CLAP DRM may utilize robust, field-recoverable security to eliminate or minimize the impact of hacks or other security concerns.

[0173] In yet another implementation, the CLAP DRM may facilitate flexible domain authentication, so that rights may be configured on groups of devices ranging from all devices owned by a single user up to all devices in a territory of licensing (e.g., country), thereby increasing flexibility of content usage and licensing. In one implementation, CLAP DRM may include DRM technologies developed by third parties (e.g., Marlin DRM). Many modern DRM technologies include a "root of trust" organization that may issue security material such as device certificates and encryption keys. Root-of-trust services also facilitate defining robustness rules for DRM implementations in order to thwart hacks on susceptible devices, and DRM implementers must adhere to these rules as part of their technology licensing agreements. The Marlin Trust Management Organization (MTMO) serves as the root of trust organization for Marlin. Marlin DRM uses a graph theory-based rights model, which consists of links and nodes. Nodes may represent concrete entities such as users, devices, and servers, or abstract entities such as domains (e.g., groups of users) or subscriptions (e.g., rights to content). Links may represent relationships between nodes. In order for a device to get permission to exercise rights to content (e.g., to play it), the device must connect through links to a node corresponding to the user licensed to access that content on that device. The LMC of the CLAP may leverage user nodes to authenticate content according to territories of license (e.g., countries), device personality nodes, corresponding to LDs, subscription links, referring to time-bounded licenses to all content, and/or the like. In one implementation, the CLAP may use user IDs to bind users to devices, to ensure that only the original person who bought the LD (or otherwise obtained it, e.g., got it as a gift) has access to the universal library. All compliant devices in a given geography may access content licensed for that geography. There is no need to limit the number of devices that a user may use, because each device represents a separate fee structure and royalty stream for content owners. The geographic licensing scheme may further facilitate royally reporting component to determine which royally scheme is to be used to calculate payments.

[0174] FIGS. **24***a-b* are data flow diagrams illustrating example license management components in some embodiments of the CLAP. As shown in FIG. **1224***a*, at **2418**, a client device **2404** may send account information obtained from user registration along with a registration token the API service **2412** of the LMC of the CLAP. The API service **2412** may receive and validate the token and create a user account

for the user of the device **2404** at **2412**. At **2422**, the client may also request and obtain from a key distributor component of LMC (e.g., SeaCert) Network Mobility (NEMO) keys and device personality ("octopus" personality). The client may securely store the NEMO keys and may utilize NEMO protocols for obtaining keys. At **2424**, the client may send device personality ID to the API service **2412**. At **2426**, the service database **2414** may associate user and device with license and appropriate territories. At **2428**, the user, device and license information may be sent to the DRM database **2416**. At **2430**, the client device may request an action token from the client DRM service **2410**. At **2432**, the client DRM service **2410** may allocate user node at **2432**, may determine subscription node IDs for territories where user and/or device is licensed for at **2434**. At **2436**, an expiration date for device to user link may be determined. The requested action token may then be generated and returned to the client at **2438**. At **2440**, the client may then send the action token to the DRM service **2406**. The DRM service **2406** may perform secure protocol processing at **2442** At **2444**, the DRM backoffice service **2408** may validate any requests for keys to ensure that the requests are consistent with user and/or device license. At **2446**, the DRM backoffice may obtain keys for user and subscription nodes and provide them to the DRM service which may create user node, subscription node(s) and links at **2448**. The created user node, subscription nodes and links may then be sent to the client at **2448**. The client may receive and store the nodes and links obtained from the DRM service. The nodes and links facilitate the client to play content licensed for appropriate territories at **2450**.

[0175] FIG. **24***b* illustrates example "octopus" graphs **2454** and **2456** for two users Ryuu and Mary respectively. In one implementation, the graph identifies the subscription nodes **2454***a*, **2456***a*, the user nodes **2454***b*, **2456***b*, the user devices **2454***c*, **2456***c* as well as the links **2454***d*, **2456***d* between the user and the device. As shown in the figure, each link may be associated with an expiration date and/or time. Each content item **2452** may include a license portion **2452***a* and a content portion **2452***c* that is encrypted with content encryption key. In this example, the license portion includes a content encryption key encrypted with Japan territory's key and Australia territory's key. From graph **2454**, the user Ryuu has a subscription node for Japan territory, and as long as the device to user link has not expired, the client on Ryuu's device may retrieve the Japan territory's key, decrypt the content encryption key and decrypt the track. However, the graph **2456** shows that the user Mary has subscription node for the USA. As the content item does not include license for USA territory, the client of Mary's device may not access the keys necessary to decrypt the track. In this way, using the subscription node, user node, device node and user-device link, only contents licensed for a user/device and territory may be decrypted.

[0176] FIG. **24***c* is a logic flow diagram illustrating an example license management component in some embodiments of the CLAP. In one implementation, the process may start at **246***o*. The player interface may be initialized at **2462**. At **2464**, the client may determine whether the device to user link is expired. If the device to user link is expired, the key necessary for decrypting the content encryption key and decrypt the tracks may be discarded at **2464**. The process may conclude at **2466**. However, if the device to user link is not expired, the client may request an action token at **2468**. Action tokens may be requested periodically by the client to ensure user, device and subscription nodes are valid. If the client does not need to request the action token, the player initialization may be completed at **2470**. In one implementation, if the client requests an action token at **2472**, the request is received by the DRM server at **2474**. The server may then check whether the user to device link is expired at **2476**. If the user to device link is expired, a notification may be sent to the client indicating that the request is invalid at **2478**. The client may receive the notification at **248***o*, concluding the process. In one implementation, if the user to device link is not expired, the server may determine if a play count report has been received since the link was last issued at **2482**. If the play count report has not been uploaded to the server, a request may be sent by the server to the client for the latest play count update at **2495**. The request may be received by the client at **2496** and the client may provide a response at **2497**. If the client fails to provide a valid play count report at **2498**, the client may be notified of the invalid request at **2478**. However, if the play count response is valid, or if the play count report was received since the issue of the link at **2482**, the server may determine a new expiration date for the device to user link at **2484**. At **2486**, a new action token for the device to user link may be generated and send to the client. The client may receive the action token at **2488** and may send the token to the server at **249***o*. The server may receive the action token and may validate the request at **2492**. The server may then create a user to device link and send the link to the client at **2494**. The client may receive and securely store the link and discard the old one at **2493**. The client may then retrieve the key necessary to decrypt content.

Encryption-Free Content Purchase Component

[0177] In one embodiment of the CLAP, an encryption-free content purchase component (ECPC) may be provided to facilitate legal purchase of DRM free content items. In one implementation, a user of the CLAP may have the option to select an option to purchase a selected track, album or a playlist from the CLAP player. Using information provided by the user (e.g., a user ID and/or password), and/or user provided payment information, a transaction for the DRM free content purchase may be completed. In one implementation, the purchase price of the content item may be discounted based on social influence. For example, if the user has opted in to a Guru program, and has achieved a threshold number of influence points (e.g., 500), the user may be offered a discount on the purchase price of the DRM free content. In another example, if the item to be purchased is in a playlist published by the user, the purchase price may be discounted for every x level or degrees of separation of people that subscribe and/or add the song and/or playlist to their library. In yet another example, when a threshold number of plays of the content item is reached, the discount may be offered. In one implementation, the threshold number of plays may be reached by the user alone. In another implementation, the threshold number of plays is an aggregate number of plays from one or more users. In one implementation, discount may be provided to a user when a threshold number of plays of the content item and/or number of people that play or buy the content DRM free after discovery from people discovering the content from the user's playlists or library is reached.

[0178] FIG. **25** is a data flow diagram illustrating an example usage reporting component in some embodiments of the CLAP. At **2502** various user actions such as play, pause,

etc., are recorded by the client. At **2504**, the accumulated user action activity and play count activity may be periodically delivered via secure protocols such as HTTP, SOAP, NEMO, etc., to the DRM service. At **2506**, secure protocol processing may be performed before sending a play count service request (e.g., HTTP, XML) to a DRM backoffice service at **2508**. The DRM backoffice service may update last play count delivery time for device in the DRM database at **2510**. The obtained activity report from the client may also be placed on a play count reporting queue **2520** at **2512**. An acknowledgement of successful receipt of the play count report may be sent to the client at **2512**. A play count queue processor may then identify various processing components and send reporting data for further processing (e.g., **2516** in music intelligence processor and **2518** in royally processor).

[0179] FIGS. **26***a-b* are logic flow diagrams illustrating example play count reporting components in some embodiments of the CLAP. As shown in FIG. **26***a*, the client side play count reporting may start at **2602**. At **2604**, a session may be launched (e.g., the client may be launched). In one implementation, a log may be created to initiate set up. Set up may include passing identifying information such as username or email address to the server. At **2606**, if an event (e.g., song started, song paused, song resumed, etc.) is detected, the event may be recorded in a log at **2608**. Each event record may be associated with identifying information such as username, track ID, time stamp, etc., at **2610**. At **2612**, the user's profile settings may be retrieved to determine whether event data should be saved to the profile. In one implementation, if the user opted to forego saving event data in his or her profile, the event data may be anonymized at **2618** such that event data may not be used to identify the user. At **2614**, a determination may be made whether threshold event capture has been triggered. Examples of threshold event capture triggers include reaching a threshold size limit of the log, number of play counts, time since last event capture, type of event, and/or the like. If the threshold event capture is triggered, the log in the client device may be synced with the log in the server device by sending event and/or other data to the server at **2616**. If on the other hand, connection to the server is not available, the event data may continue to be logged if there is another occurrence of an event at **262***o*. If there are no events (e.g., application is closed), the log file may be closed and the session may end at **2622**.

[0180] As shown in FIG. **26***b*, the server-side play count reporting may start at **2650** by receiving event data at **265***o*. The event data may include information such as track ID, time at which a song started, ended, was paused or resumed, username, and/or the like. At **2654**, information such as event type, track ID, time stamp, etc., may be extracted from the received event data. Using the extracted information and one or more business rules, play count for each track ID in the event data may be determined. An example business rule may include, for example, a rule which classifies a track that was played for at least x seconds (e.g., 30 seconds, 45 seconds, etc.) as a "play event." Using the time stamp for each event (e.g., song started, song paused, song resumed or song finished) and the business rule, play count for each track may be determined at **2656**. In one implementation, the event data may be anonymized with references to any identifying user information removed. If the event data is anonymized as determined at **2658**, the reporting database is updated with determined play count data at **2664**. On the other hand, if the event data is not anonymized, user identifying information such as username, user ID, social graph, interest graph, etc., may be extracted, obtained and/or derived at **2660**. At **2662**, the user profile may be obtained and updated with play count and track data at **2662**. As in the case of anonymized data, the reporting database may be updated with play count data at **2664**. In one implementation, a category of activity for each event may be determined at **2666**. Example categories may include point generating activity, royalty activity, recommendation engine activity, and/or the like. At **2668**, the event data may be added to one or more databases and/or tables corresponding to the determined activity. At **2670**, if there is another event in the queue, the processing for the event may begin at **2654**. If there are no other events in the queue, the process may end at **2672**.

Usage Payment Collection and Apportionment Component (UPCAC)

[0181] FIG. **27** is a logic flow diagram illustrating an example usage payment collection and apportionment component (UPCAC) in some embodiments of the CLAP. The process may start at **1702** by receiving a royalty report request at **2704**. In one implementation, the report request may include reporting criteria and/or categories. Examples of reporting criteria and/or categories may include, for example, a statement for a selected period of time (e.g., weekly, monthly, from/to, etc.), report by track, by artist, by song, by album, by territory, partner name, activity category, etc. Yet other examples of reporting criteria may include, but are not limited to: total number of end users, number of end users by device type (e.g., users with more than one device may count more than once), number of new end users, number of new end users by device type, number of active users (e.g., users who have one or more plays or downloads during a period of time), number of active users by device type, number of active downloaders (e.g., users who have at least one download during a period of time), number of active downloaders by device type, market share of downloads by device type for a partner, total downloads of all label content by device type, total digital downloads by device type for a partner (e.g., SME, EMI, WMG, etc.), total number of active listeners, market share of plays by device type for a partner, total plays for all label content, total number of playlists created, usage code (e.g., streaming, interactive radio, tethered plays, juke box, portable, etc.) and/or the like.

[0182] At **2706**, if no reporting criteria and/or categories are provided, default criteria may be selected at **2720**. An example default criteria may be last statement available. However, if reporting criteria and/or categories are provided, for each provided criteria and/or category, the UPCAC may, at **2708**, query a reporting database using the provided criteria and/or category to obtain matching tracks. In one implementation, no tracks matching the reporting criteria and/or categories may be obtained from the query at **2710**. In this case, at **2722**, the UPCAC may notify the requestor that no royalties are due for the specified criteria and the logic flow may conclude at **2724**. In an alternate implementation, one or more tracks matching the provided criteria/categories may be obtained at **2710**. At **2712**, the play count data for each of the identified tracks may be retrieved. At **2714**, a royalty database may be queried to obtain rates associated with tracks and/or partners. At **2716**, royalty payments may be calculated based on the play count data and the obtained rates. Further at **2718**, the requested royally report may be generated and provided. The report may include, in one implementation, a listing of

tracks matching the provided or default criteria and/or categories as well as the royalty amounts due per track. In one implementation, the play count data may be segmented according to territory, and royalty rates for each territory may be retrieved to determine the total royalties owed.

CLAP Controller

[0183] FIG. **28** shows a block diagram illustrating embodiments of a CLAP controller. In this embodiment, the CLAP controller **2801** may serve to aggregate, process, store, search, serve, identify, instruct, generate, match, and/or facilitate interactions with a computer through various technologies, and/or other related data.

[0184] Typically, users, which may be people and/or other systems, may engage information technology systems (e.g., computers) to facilitate information processing. In turn, computers employ processors to process information; such processors **2803** may be referred to as central processing units (CPU). One form of processor is referred to as a microprocessor. CPUs use communicative circuits to pass binary encoded signals acting as instructions to enable various operations. These instructions may be operational and/or data instructions containing and/or referencing other instructions and data in various processor accessible and operable areas of memory **2829** (e.g., registers, cache memory, random access memory, etc.). Such communicative instructions may be stored and/or transmitted in batches (e.g., batches of instructions) as programs and/or data components to facilitate desired operations. These stored instruction codes, e.g., programs, may engage the CPU circuit components and other motherboard and/or system components to perform desired operations. One type of program is a computer operating system, which, may be executed by CPU on a computer; the operating system enables and facilitates users to access and operate computer information technology and resources. Some resources that may be employed in information technology systems include: input and output mechanisms through which data may pass into and out of a computer; memory storage into which data may be saved; and processors by which information may be processed. These information technology systems may be used to collect data for later retrieval, analysis, and manipulation, which may be facilitated through a database program. These information technology systems provide interfaces that allow users to access and operate various system components.

[0185] In one embodiment, the CLAP controller **2801** may be connected to and/or communicate with entities such as, but not limited to: one or more users from user input devices **2811**; peripheral devices **2812**; an optional cryptographic processor device **2828**; and/or a communications network **2813**.

[0186] Networks are commonly thought to comprise the interconnection and interoperation of clients, servers, and intermediary nodes in a graph topology. It should be noted that the term "server" as used throughout this application refers generally to a computer, other device, program, or combination thereof that processes and responds to the requests of remote users across a communications network. Servers serve their information to requesting "clients." The term "client" as used herein refers generally to a computer, program, other device, user and/or combination thereof that is capable of processing and making requests and obtaining and processing any responses from servers across a communications network. A computer, other device, program, or combination thereof that facilitates, processes information and requests, and/or furthers the passage of information from a source user to a destination user is commonly referred to as a "node." Networks are generally thought to facilitate the transfer of information from source points to destinations. A node specifically tasked with furthering the passage of information from a source to a destination is commonly called a "router." There are many forms of networks such as Local Area Networks (LANs), Pico networks, Wide Area Networks (WANs), Wireless Networks (WLANs), etc. For example, the Internet is generally accepted as being an interconnection of a multitude of networks whereby remote clients and servers may access and interoperate with one another.

[0187] The CLAP controller **2801** may be based on computer systems that may comprise, but are not limited to, components such as: a computer systemization **2802** connected to memory **2829**.

Computer Systemization

[0188] A computer systemization **2802** may comprise a clock **2830**, central processing unit ("CPU(s)" and/or "processor(s)" (these terms are used interchangeable throughout the disclosure unless noted to the contrary)) **2803**, a memory **2829** (e.g., a read only memory (ROM) **2806**, a random access memory (RAM) **2805**, etc.), and/or an interface bus **2807**, and most frequently, although not necessarily, are all interconnected and/or communicating through a system bus **2804** on one or more (mother)board(s) **2802** having conductive and/or otherwise transportive circuit pathways through which instructions (e.g., binary encoded signals) may travel to effectuate communications, operations, storage, etc. The computer systemization may be connected to a power source **2886**; e.g., optionally the power source may be internal. Optionally, a cryptographic processor **2826** and/or transceivers (e.g., ICs) **2874** may be connected to the system bus. In another embodiment, the cryptographic processor and/or transceivers may be connected as either internal and/or external peripheral devices **2812** via the interface bus I/O. In turn, the transceivers may be connected to antenna(s) **2875**, thereby effectuating wireless transmission and reception of various communication and/or sensor protocols; for example the antenna(s) may connect to: a Texas Instruments WiLink WL1283 transceiver chip (e.g., providing 802.11n, Bluetooth 3.0, FM, global positioning system (GPS) (thereby allowing CLAP controller to determine its location)); Broadcom BCM4329FKUBG transceiver chip (e.g., providing 802.11n, Bluetooth 2.1+EDR, FM, etc.); a Broadcom BCM4750UB8 receiver chip (e.g., GPS); an Infineon Technologies X-Gold 618-PMB9800 (e.g., providing 2G/3G HSDPA/HSUPA communications); and/or the like. The system clock typically has a crystal oscillator and generates a base signal through the computer systemization's circuit pathways. The clock is typically coupled to the system bus and various clock multipliers that will increase or decrease the base operating frequency for other components interconnected in the computer systemization. The clock and various components in a computer systemization drive signals embodying information throughout the system. Such transmission and reception of instructions embodying information throughout a computer systemization may be commonly referred to as communications. These communicative instructions may further be transmitted, received, and the cause of return and/or reply communications beyond the instant computer systemization to: communications networks, input devices, other computer systemiza-

tions, peripheral devices, and/or the like. It should be understood that in alternative embodiments, any of the above components may be connected directly to one another, connected to the CPU, and/or organized in numerous variations employed as exemplified by various computer systems.

[0189] The CPU comprises at least one high-speed data processor adequate to execute program components for executing user and/or system-generated requests. Often, the processors themselves will incorporate various specialized processing units, such as, but not limited to: integrated system (bus) controllers, memory management control units, floating point units, and even specialized processing sub-units like graphics processing units, digital signal processing units, and/or the like. Additionally, processors may include internal fast access addressable memory, and be capable of mapping and addressing memory **2829** beyond the processor itself; internal memory may include, but is not limited to: fast registers, various levels of cache memory (e.g., level 1, 2, 3, etc.), RAM, etc. The processor may access this memory through the use of a memory address space that is accessible via instruction address, which the processor can construct and decode allowing it to access a circuit path to a specific memory address space having a memory state. The CPU may be a microprocessor such as: AMD's Athlon, Duron and/or Opteron; ARM's application, embedded and secure processors; IBM and/or Motorola's DragonBall and PowerPC; IBM's and Sony's Cell processor; Intel's Celeron, Core (2) Duo, Itanium, Pentium, Xeon, and/or XScale; and/or the like processor(s). The CPU interacts with memory through instruction passing through conductive and/or transportive conduits (e.g., (printed) electronic and/or optic circuits) to execute stored instructions (i.e., program code) according to conventional data processing techniques. Such instruction passing facilitates communication within the CLAP controller and beyond through various interfaces. Should processing requirements dictate a greater amount speed and/or capacity, distributed processors (e.g., Distributed CLAP), mainframe, multi-core, parallel, and/or super-computer architectures may similarly be employed. Alternatively, should deployment requirements dictate greater portability, smaller Personal Digital Assistants (PDAs) may be employed.

[0190] Depending on the particular implementation, features of the CLAP may be achieved by implementing a microcontroller such as CAST's R8051XC2 microcontroller; Intel's MCS 51 (i.e., 8051 microcontroller); and/or the like. Also, to implement certain features of the CLAP, some feature implementations may rely on embedded components, such as: Application-Specific Integrated Circuit ("ASIC"), Digital Signal Processing ("DSP"), Field Programmable Gate Array ("FPGA"), and/or the like embedded technology. For example, any of the CLAP component collection (distributed or otherwise) and/or features may be implemented via the microprocessor and/or via embedded components; e.g., via ASIC, coprocessor, DSP, FPGA, and/or the like. Alternately, some implementations of the CLAP may be implemented with embedded components that are configured and used to achieve a variety of features or signal processing.

[0191] Depending on the particular implementation, the embedded components may include software solutions, hardware solutions, and/or some combination of both hardware/software solutions. For example, CLAP features discussed herein may be achieved through implementing FPGAs, which are a semiconductor devices containing programmable logic components called "logic blocks", and programmable interconnects, such as the high performance FPGA Virtex series and/or the low cost Spartan series manufactured by Xilinx. Logic blocks and interconnects can be programmed by the customer or designer, after the FPGA is manufactured, to implement any of the CLAP features. A hierarchy of programmable interconnects allow logic blocks to be interconnected as needed by the CLAP system designer/administrator, somewhat like a one-chip programmable breadboard. An FPGA's logic blocks can be programmed to perform the operation of basic logic gates such as AND, and XOR, or more complex combinational operators such as decoders or mathematical operations. In most FPGAs, the logic blocks also include memory elements, which may be circuit flip-flops or more complete blocks of memory. In some circumstances, the CLAP may be developed on regular FPGAs and then migrated into a fixed version that more resembles ASIC implementations. Alternate or coordinating implementations may migrate CLAP controller features to a final ASIC instead of or in addition to FPGAs. Depending on the implementation all of the aforementioned embedded components and microprocessors may be considered the "CPU" and/or "processor" for the CLAP.

Power Source

[0192] The power source **2886** may be of any standard form for powering small electronic circuit board devices such as the following power cells: alkaline, lithium hydride, lithium ion, lithium polymer, nickel cadmium, solar cells, and/or the like. Other types of AC or DC power sources may be used as well. In the case of solar cells, in one embodiment, the case provides an aperture through which the solar cell may capture photonic energy. The power cell **2886** is connected to at least one of the interconnected subsequent components of the CLAP thereby providing an electric current to all subsequent components. In one example, the power source **2886** is connected to the system bus component **2804**. In an alternative embodiment, an outside power source **2886** is provided through a connection across the I/O **2808** interface. For example, a USB and/or IEEE 1394 connection carries both data and power across the connection and is therefore a suitable source of power.

Interface Adapters

[0193] Interface bus(ses) **2807** may accept, connect, and/or communicate to a number of interface adapters, conventionally although not necessarily in the form of adapter cards, such as but not limited to: input output interfaces (I/O) **2808**, storage interfaces **2809**, network interfaces **2810**, and/or the like. Optionally, cryptographic processor interfaces **2827** similarly may be connected to the interface bus. The interface bus provides for the communications of interface adapters with one another as well as with other components of the computer systemization. Interface adapters are adapted for a compatible interface bus. Interface adapters conventionally connect to the interface bus via a slot architecture. Conventional slot architectures may be employed, such as, but not limited to: Accelerated Graphics Port (AGP), Card Bus, (Extended) Industry Standard Architecture ((E)ISA), Micro Channel Architecture (MCA), NuBus, Peripheral Component Interconnect (Extended) (PCI(X)), PCI Express, Personal 19 Computer Memory Card International Association (PCMCIA), and/or the like.

[0194] Storage interfaces **2809** may accept, communicate, and/or connect to a number of storage devices such as, but not limited to: storage devices **2814**, removable disc devices, and/or the like. Storage interfaces may employ connection protocols such as, but not limited to: (Ultra) (Serial) Advanced Technology Attachment (Packet Interface) ((Ultra) (Serial) ATA(PI)), (Enhanced) Integrated Drive Electronics ((E)IDE), Institute of Electrical and Electronics Engineers (IEEE) 1394, fiber channel, Small Computer Systems Interface (SCSI), Universal Serial Bus (USB), and/or the like.

[0195] Network interfaces **2810** may accept, communicate, and/or connect to a communications network **2813**. Through a communications network **2813**, the CLAP controller is accessible through remote clients **2833***b* (e.g., computers with web browsers) by users **2833***a*. Network interfaces may employ connection protocols such as, but not limited to: direct connect, Ethernet (thick, thin, twisted pair 10/100/1000 Base T, and/or the like), Token Ring, wireless connection such as IEEE 802.11a-x, and/or the like. Should processing requirements dictate a greater amount speed and/or capacity, distributed network controllers (e.g., Distributed CLAP), architectures may similarly be employed to pool, load balance, and/or otherwise increase the communicative bandwidth required by the CLAP controller. A communications network may be any one and/or the combination of the following: a direct interconnection; the Internet; a Local Area Network (LAN); a Metropolitan Area Network (MAN); an Operating Missions as Nodes on the Internet (OMNI); a secured custom connection; a Wide Area Network (WAN); a wireless network (e.g., employing protocols such as, but not limited to a Wireless Application Protocol (WAP), I-mode, and/or the like); and/or the like. A network interface may be regarded as a specialized form of an input output interface. Further, multiple network interfaces **2810** may be used to engage with various communications network types **2813**. For example, multiple network interfaces may be employed to allow for the communication over broadcast, multicast, and/or unicast networks.

[0196] Input Output interfaces (I/O) **2808** may accept, communicate, and/or connect to user input devices **2811**, peripheral devices **2812**, cryptographic processor devices **2828**, and/or the like. I/O may employ connection protocols such as, but not limited to: audio: analog, digital, monaural, RCA, stereo, and/or the like; data: Apple Desktop Bus (ADB), IEEE 1394a-b, serial, universal serial bus (USB); infrared; joystick; keyboard; midi; optical; PC AT; PS/2; parallel; radio; video interface: Apple Desktop Connector (ADC), BNC, coaxial, component, composite, digital, Digital Visual Interface (DVI), high-definition multimedia interface (HDMI), RCA, RF antennae, S-Video, VGA, and/or the like; wireless transceivers: 802.11a/b/g/n/x; Bluetooth; cellular (e.g., code division multiple access (CDMA), high speed packet access (HSPA(+)), high-speed downlink packet access (HSDPA), global system for mobile communications (GSM), long term evolution (LTE), WiMax, etc.); and/or the like. One typical output device may include a video display, which typically comprises a Cathode Ray Tube (CRT) or Liquid Crystal Display (LCD) based monitor with an interface (e.g., DVI circuitry and cable) that accepts signals from a video interface, may be used. The video interface composites information generated by a computer systemization and generates video signals based on the composited information in a video memory frame. Another output device is a television set, which accepts signals from a video interface. Typically, the video interface provides the composited video information through a video connection interface that accepts a video display interface (e.g., an RCA composite video connector accepting an RCA composite video cable; a DVI connector accepting a DVI display cable, etc.).

[0197] User input devices **2811** often are a type of peripheral device **512** (see below) and may include: card readers, dongles, finger print readers, gloves, graphics tablets, joysticks, keyboards, microphones, mouse (mice), remote controls, retina readers, touch screens (e.g., capacitive, resistive, etc.), trackballs, trackpads, sensors (e.g., accelerometers, ambient light, GPS, gyroscopes, proximity, etc.), styluses, and/or the like.

[0198] Peripheral devices **2812** may be connected and/or communicate to I/O and/or other facilities of the like such as network interfaces, storage interfaces, directly to the interface bus, system bus, the CPU, and/or the like. Peripheral devices may be external, internal and/or part of the CLAP controller. Peripheral devices may include: antenna, audio devices (e.g., line-in, line-out, microphone input, speakers, etc.), cameras (e.g., still, video, webcam, etc.), dongles (e.g., for copy protection, ensuring secure transactions with a digital signature, and/or the like), external processors (for added capabilities; e.g., crypto devices **528**), force-feedback devices (e.g., vibrating motors), network interfaces, printers, scanners, storage devices, transceivers (e.g., cellular, GPS, etc.), video devices (e.g., goggles, monitors, etc.), video sources, visors, and/or the like. Peripheral devices often include types of input devices (e.g., cameras).

[0199] It should be noted that although user input devices and peripheral devices may be employed, the CLAP controller may be embodied as an embedded, dedicated, and/or monitor-less (i.e., headless) device, wherein access would be provided over a network interface connection.

[0200] Cryptographic units such as, but not limited to, microcontrollers, processors **2826**, interfaces **2827**, and/or devices **2828** may be attached, and/or communicate with the CLAP controller. A MC68HC16 microcontroller, manufactured by Motorola Inc., may be used for and/or within cryptographic units. The MC68HC16 microcontroller utilizes a 16-bit multiply-and-accumulate instruction in the 16 MHz configuration and requires less than one second to perform a 512-bit RSA private key operation. Cryptographic units support the authentication of communications from interacting agents, as well as allowing for anonymous transactions. Cryptographic units may also be configured as part of the CPU. Equivalent microcontrollers and/or processors may also be used. Other commercially available specialized cryptographic processors include: Broadcom's CryptoNetX and other Security Processors; nCipher's nShield; SafeNet's Luna PCI (e.g., 7100) series; Semaphore Communications' 40 MHz Roadrunner 184; Sun's Cryptographic Accelerators (e.g., Accelerator 6000 PCIe Board, Accelerator 500 Daughtercard); Via Nano Processor (e.g., L2100, L2200, U2400) line, which is capable of performing 500+MB/s of cryptographic instructions; VLSI Technology's 33 MHz 6868; and/or the like.

Memory

[0201] Generally, any mechanization and/or embodiment allowing a processor to affect the storage and/or retrieval of information is regarded as memory **2829**. However, memory is a fungible technology and resource, thus, any number of memory embodiments may be employed in lieu of or in

concert with one another. It is to be understood that the CLAP controller and/or a computer systemization may employ various forms of memory **2829**. For example, a computer systemization may be configured wherein the operation of on-chip CPU memory (e.g., registers), RAM, ROM, and any other storage devices are provided by a paper punch tape or paper punch card mechanism; however, such an embodiment would result in an extremely slow rate of operation. In a typical configuration, memory **2829** will include ROM **2806**, RAM **2805**, and a storage device **2814**. A storage device **2814** may be any conventional computer system storage. Storage devices may include a drum; a (fixed and/or removable) magnetic disk drive; a magneto-optical drive; an optical drive (i.e., Blueray, CD ROM/RAM/Recordable (R)/ReWritable (RW), DVD R/RW, HD DVD R/RW etc.); an array of devices (e.g., Redundant Array of Independent Disks (RAID)); solid state memory devices (USB memory, solid state drives (SSD), etc.); other processor-readable storage mediums; and/or other devices of the like. Thus, a computer systemization generally requires and makes use of memory.

Component Collection

[0202] The memory **2829** may contain a collection of program and/or database components and/or data such as, but not limited to: operating system component(s) **2815** (operating system); information server component(s) **2816** (information server); user interface component(s) **2817** (user interface); Web browser component(s) **2818** (Web browser); database(s) **2819**; mail server component(s) **2821**; mail client component (s) **2822**; cryptographic server component(s) **2820** (cryptographic server); the CLAP component(s) **2835**; shared discovery component **2852**, discover components **2851**, licensing & license acquisition component **2850**, royalty calculation/reporting component **2849**, usage reporting component **2848**, play count reporting component **2847**, crowd sourcing component **2846**, guru rewarding component **2845**, smart caching component **2844**, search component **2843**, non-local content cache component **2842**, magic playlist generation component **2841**, and/or the like (i.e., collectively a component collection). These components may be stored and accessed from the storage devices and/or from storage devices accessible through an interface bus. Although non-conventional program components such as those in the component collection, typically, are stored in a local storage device **2814**, they may also be loaded and/or stored in memory such as: peripheral devices, RAM, remote storage facilities through a communications network, ROM, various forms of memory, and/or the like.

Operating System

[0203] The operating system component **2815** is an executable program component facilitating the operation of the CLAP controller. Typically, the operating system facilitates access of I/O, network interfaces, peripheral devices, storage devices, and/or the like. The operating system may be a highly fault tolerant, scalable, and secure system such as: Apple Macintosh OS X (Server); AT&T Nan 9; Be OS; Unix and Unix-like system distributions (such as AT&T's UNIX; Berkley Software Distribution (BSD) variations such as FreeBSD, NetBSD, OpenBSD, and/or the like; Linux distributions such as Red Hat, Ubuntu, and/or the like); and/or the like operating systems. However, more limited and/or less secure operating systems also may be employed such as Apple Macintosh OS, IBM OS/2, Microsoft DOS, Microsoft Windows 2000/2003/3.1/95/98/CE/Millenium/NT/Vista/XP (Server), Palm OS, and/or the like. An operating system may communicate to and/or with other components in a component collection, including itself, and/or the like. Most frequently, the operating system communicates with other program components, user interfaces, and/or the like. For example, the operating system may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses. The operating system, once executed by the CPU, may enable the interaction with communications networks, data, I/O, peripheral devices, program components, memory, user input devices, and/or the like. The operating system may provide communications protocols that allow the CLAP controller to communicate with other entities through a communications network **2813**. Various communication protocols may be used by the CLAP controller as a subcarrier transport mechanism for interaction, such as, but not limited to: multicast, TCP/IP, UDP, unicast, and/or the like.

Information Server

[0204] An information server component **2816** is a stored program component that is executed by a CPU. The information server may be a conventional Internet information server such as, but not limited to Apache Software Foundation's Apache, Microsoft's Internet Information Server, and/or the like. The information server may allow for the execution of program components through facilities such as Active Server Page (ASP), ActiveX, (ANSI) (Objective-) C (++), C# and/or .NET, Common Gateway Interface (CGI) scripts, dynamic (D) hypertext markup language (HTML), FLASH, Java, JavaScript, Practical Extraction Report Language (PERL), Hypertext Pre-Processor (PHP), pipes, Python, wireless application protocol (WAP), WebObjects, and/or the like. The information server may support secure communications protocols such as, but not limited to, File Transfer Protocol (FTP); HyperText Transfer Protocol (HTTP); Secure Hypertext Transfer Protocol (HTTPS), Secure Socket Layer (SSL), messaging protocols (e.g., America Online (AOL) Instant Messenger (AIM), Application Exchange (APEX), ICQ, Internet Relay Chat (IRC), Microsoft Network (MSN) Messenger Service, Presence and Instant Messaging Protocol (PRIM), Internet Engineering Task Force's (IETF's) Session Initiation Protocol (SIP), SIP for Instant Messaging and Presence Leveraging Extensions (SIMPLE), open XML-based Extensible Messaging and Presence Protocol (XMPP) (i.e., Jabber or Open Mobile Alliance's (OMA's) Instant Messaging and Presence Service (IMPS)), Yahoo! Instant Messenger Service, and/or the like. The information server provides results in the form of Web pages to Web browsers, and allows for the manipulated generation of the Web pages through interaction with other program components. After a Domain Name System (DNS) resolution portion of an HTTP request is resolved to a particular information server, the information server resolves requests for information at specified locations on the CLAP controller based on the remainder of the HTTP request. For example, a request such as http://123.124.125.126/myInformation.html might have the IP portion of the request "123.124.125.126" resolved by a DNS server to an information server at that IP address; that information server might in turn further parse the http request for the "/myInformation.html" portion of the request and resolve it to a location in memory containing the information "myInformation.

html." Additionally, other information serving protocols may be employed across various ports, e.g., FTP communications across port **21**, and/or the like. An information server may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the information server communicates with the CLAP database **2819**, operating systems, other program components, user interfaces, Web browsers, and/or the like.

[0205] Access to the CLAP database may be achieved through a number of database bridge mechanisms such as through scripting languages as enumerated below (e.g., CGI) and through inter-application communication channels as enumerated below (e.g., CORBA, WebObjects, etc.). Any data requests through a Web browser are parsed through the bridge mechanism into appropriate grammars as required by the CLAP. In one embodiment, the information server would provide a Web form accessible by a Web browser. Entries made into supplied fields in the Web form are tagged as having been entered into the particular fields, and parsed as such. The entered terms are then passed along with the field tags, which act to instruct the parser to generate queries directed to appropriate tables and/or fields. In one embodiment, the parser may generate queries in standard SQL by instantiating a search string with the proper join/select commands based on the tagged text entries, wherein the resulting command is provided over the bridge mechanism to the CLAP as a query. Upon generating query results from the query, the results are passed over the bridge mechanism, and may be parsed for formatting and generation of a new results Web page by the bridge mechanism. Such a new results Web page is then provided to the information server, which may supply it to the requesting Web browser.

[0206] Also, an information server may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses.

User Interface

[0207] Computer interfaces in some respects are similar to automobile operation interfaces. Automobile operation interface elements such as steering wheels, gearshifts, and speedometers facilitate the access, operation, and display of automobile resources, and status. Computer interaction interface elements such as check boxes, cursors, menus, scrollers, and windows (collectively and commonly referred to as widgets) similarly facilitate the access, capabilities, operation, and display of data and computer hardware and operating system resources, and status. Operation interfaces are commonly called user interfaces. Graphical user interfaces (GUIs) such as the Apple Macintosh Operating System's Aqua, IBM's OS/2, Microsoft's Windows 2000/2003/3.1/95/98/CE/Millenium/NT/XP/Vista/7 (i.e., Aero), Unix's X-Windows (e.g., which may include additional Unix graphic interface libraries and layers such as K Desktop Environment (KDE), mythTV and GNU Network Object Model Environment (GNOME)), web interface libraries (e.g., ActiveX, AJAX, (D)HTML, FLASH, Java, JavaScript, etc. interface libraries such as, but not limited to, Dojo, jQuery(UI), MooTools, Prototype, script.aculo.us, SWFObject, Yahoo! User Interface, any of which may be used and) provide a baseline and means of accessing and displaying information graphically to users.

[0208] A user interface component **2817** is a stored program component that is executed by a CPU. The user interface may be a conventional graphic user interface as provided by, with, and/or atop operating systems and/or operating environments such as already discussed. The user interface may allow for the display, execution, interaction, manipulation, and/or operation of program components and/or system facilities through textual and/or graphical facilities. The user interface provides a facility through which users may affect, interact, and/or operate a computer system. A user interface may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the user interface communicates with operating systems, other program components, and/or the like. The user interface may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses.

Web Browser

[0209] A Web browser component **2818** is a stored program component that is executed by a CPU. The Web browser may be a conventional hypertext viewing application such as Microsoft Internet Explorer or Netscape Navigator. Secure Web browsing may be supplied with 128 bit (or greater) encryption by way of HTTPS, SSL, and/or the like. Web browsers allowing for the execution of program components through facilities such as ActiveX, AJAX, (D)HTML, FLASH, Java, JavaScript, web browser plug-in APIs (e.g., FireFox, Safari Plug-in, and/or the like APIs), and/or the like. Web browsers and like information access tools may be integrated into PDAs, cellular telephones, and/or other mobile devices. A Web browser may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the Web browser communicates with information servers, operating systems, integrated program components (e.g., plug-ins), and/or the like; e.g., it may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses. Also, in place of a Web browser and information server, a combined application may be developed to perform similar operations of both. The combined application would similarly affect the obtaining and the provision of information to users, user agents, and/or the like from the CLAP enabled nodes. The combined application may be nugatory on systems employing standard Web browsers.

Mail Server

[0210] A mail server component **2821** is a stored program component that is executed by a CPU **2803**. The mail server may be a conventional Internet mail server such as, but not limited to sendmail, Microsoft Exchange, and/or the like. The mail server may allow for the execution of program components through facilities such as ASP, ActiveX, (ANSI) (Objective-) C (++), C# and/or .NET, CGI scripts, Java, JavaScript, PERL, PHP, pipes, Python, WebObjects, and/or the like. The mail server may support communications protocols such as, but not limited to: Internet message access protocol (IMAP), Messaging Application Programming Interface (MAPI)/Microsoft Exchange, post office protocol (POP3), simple mail transfer protocol (SMTP), and/or the like. The mail server can route, forward, and process incoming and outgoing mail messages that have been sent, relayed and/or otherwise traversing through and/or to the CLAP.

[0211] Access to the CLAP mail may be achieved through a number of APIs offered by the individual Web server components and/or the operating system.

[0212] Also, a mail server may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, information, and/or responses.

Mail Client

[0213] A mail client component **2822** is a stored program component that is executed by a CPU **2803**. The mail client may be a conventional mail viewing application such as Apple Mail, Microsoft Entourage, Microsoft Outlook, Microsoft Outlook Express, Mozilla, Thunderbird, and/or the like. Mail clients may support a number of transfer protocols, such as: IMAP, Microsoft Exchange, POP3, SMTP, and/or the like. A mail client may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the mail client communicates with mail servers, operating systems, other mail clients, and/or the like; e.g., it may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, information, and/or responses. Generally, the mail client provides a facility to compose and transmit electronic mail messages.

Cryptographic Server

[0214] A cryptographic server component **2820** is a stored program component that is executed by a CPU **2803**, cryptographic processor **2826**, cryptographic processor interface **2827**, cryptographic processor device **2828**, and/or the like. Cryptographic processor interfaces will allow for expedition of encryption and/or decryption requests by the cryptographic component; however, the cryptographic component, alternatively, may run on a conventional CPU. The cryptographic component allows for the encryption and/or decryption of provided data. The cryptographic component allows for both symmetric and asymmetric (e.g., Pretty Good Protection (PGP)) encryption and/or decryption. The cryptographic component may employ cryptographic techniques such as, but not limited to: digital certificates (e.g., X.509 authentication framework), digital signatures, dual signatures, enveloping, password access protection, public key management, and/or the like. The cryptographic component will facilitate numerous (encryption and/or decryption) security protocols such as, but not limited to: checksum, Data Encryption Standard (DES), Elliptical Curve Encryption (ECC), International Data Encryption Algorithm (IDEA), Message Digest 5 (MD5, which is a one way hash operation), passwords, Rivest Cipher (RC5), Rijndael, RSA (which is an Internet encryption and authentication system that uses an algorithm developed in 1977 by Ron Rivest, Adi Shamir, and Leonard Adleman), Secure Hash Algorithm (SHA), Secure Socket Layer (SSL), Secure Hypertext Transfer Protocol (HTTPS), and/or the like. Employing such encryption security protocols, the CLAP may encrypt all incoming and/or outgoing communications and may serve as node within a virtual private network (VPN) with a wider communications network. The cryptographic component facilitates the process of "security authorization" whereby access to a resource is inhibited by a security protocol wherein the cryptographic component effects authorized access to the secured resource. In addition, the cryptographic component may provide unique identifiers of content, e.g., employing and MD5 hash to obtain a unique signature for an digital audio file. A cryptographic component may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. The cryptographic component supports encryption schemes allowing for the secure transmission of information across a communications network to enable the CLAP component to engage in secure transactions if so desired. The cryptographic component facilitates the secure accessing of resources on the CLAP and facilitates the access of secured resources on remote systems; i.e., it may act as a client and/or server of secured resources. Most frequently, the cryptographic component communicates with information servers, operating systems, other program components, and/or the like. The cryptographic component may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses.

The CLAP Database

[0215] The CLAP database component **2819** may be embodied in a database and its stored data. The database is a stored program component, which is executed by the CPU; the stored program component portion configuring the CPU to process the stored data. The database may be a conventional, fault tolerant, relational, scalable, secure database such as Oracle or Sybase. Relational databases are an extension of a flat file. Relational databases consist of a series of related tables. The tables are interconnected via a key field. Use of the key field allows the combination of the tables by indexing against the key field; i.e., the key fields act as dimensional pivot points for combining information from various tables. Relationships generally identify links maintained between tables by matching primary keys. Primary keys represent fields that uniquely identify the rows of a table in a relational database. More precisely, they uniquely identify rows of a table on the "one" side of a one-to-many relationship.

[0216] Alternatively, the CLAP database may be implemented using various standard data-structures, such as an array, hash, (linked) list, struct, structured text file (e.g., XML), table, and/or the like. Such data-structures may be stored in memory and/or in (structured) files. In another alternative, an object-oriented database may be used, such as Frontier, ObjectStore, Poet, Zope, and/or the like. Object databases can include a number of object collections that are grouped and/or linked together by common attributes; they may be related to other object collections by some common attributes. Object-oriented databases perform similarly to relational databases with the exception that objects are not just pieces of data but may have other types of capabilities encapsulated within a given object. If the CLAP database is implemented as a data-structure, the use of the CLAP database **2819** may be integrated into another component such as the CLAP component **2835**. Also, the database may be implemented as a mix of data structures, objects, and relational structures. Databases may be consolidated and/or distributed in countless variations through standard data processing techniques. Portions of databases, e.g., tables, may be exported and/or imported and thus decentralized and/or integrated.

[0217] In one embodiment, the database component **2819** includes several tables **2819***a-k*. A User Accounts table **2219***a* may include fields such as, but not limited to: user_ID, user_password, user_device, user_IP, user_entity, user_media,

user_search, user_socialConnections, user_following, user_followed, and/or the like. The User table may support and/or track multiple entity accounts on a CLAP. A metadata table **2219***b* may include fields such as, but not limited to: track_ID, media_type, media_name, media_size, media_genre, media_album, media_artist, media_user, media_length, media_ranking, media_year, and/or the like. A search table **2219***c* may include fields such as, but not limited to: search_ID, search_userID, search_content, search_time, search_socialConnection, search_result, and/or the like. A social table **2219***d* may include fields such as, but not limited to: social_ID, social_name, social connection, social_searchHistory, social_medialData, and/or the like. A media table **2219***e* may include fields such as, but not limited to: user_ID, track_ID, media_type, and/or the like. A reporting table **2219***f* may include fields such as, but not limited to: track_ID, track_playcount, track_royalty, track added date, statement_ID, and/or the like. A playlist table **2219***g* may include fields such as, but not limited to: user_ID, track_ID, playlist_pubdate, playlist share, and/or the like. The core may include additional databases and/or tables. A log table **2219***h* may include fields such as, but not limited to: log_ID, log_user_ID, log_deviceID, log_date, log_type, and/or the like. A system model layout table **2219***i* may include fields such as, but not limited to: layout_ID, bandwidth, and/or the like. A service table **2219***j* may include fields such as, but not limited to: notification_rule, threshold, log source, resource requirements, priority, and/or the like. A Client Account table **2219***k* may include fields such as, but not limited to: client_ID, client_account, client_name, client_password, client_permissions, and/or the like.

[0218] In one embodiment, the CLAP database may interact with other database systems. For example, employing a distributed database system, queries and data access by search CLAP component may treat the combination of the CLAP database, an integrated data security layer database as a single database entity.

[0219] In one embodiment, user programs may contain various user interface primitives, which may serve to update the CLAP. Also, various accounts may require custom database tables depending upon the environments and the types of clients the CLAP may need to serve. It should be noted that any unique fields may be designated as a key field throughout. In an alternative embodiment, these tables have been decentralized into their own databases and their respective database controllers (i.e., individual database controllers for each of the above tables). Employing standard data processing techniques, one may further distribute the databases over several computer systemizations and/or storage devices. Similarly, configurations of the decentralized database controllers may be varied by consolidating and/or distributing the various database components **2819***a-k*. The CLAP may be configured to keep track of various settings, inputs, and parameters via database controllers.

[0220] The CLAP database may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the CLAP database communicates with the CLAP component, other program components, and/or the like. The database may contain, retain, and provide information regarding other nodes and data.

The CLAPs

[0221] The CLAP component **2835** is a stored program component that is executed by a CPU. In one embodiment, the CLAP component incorporates any and/or all combinations of the aspects of the CLAP that was discussed in the previous figures. As such, the CLAP affects accessing, obtaining and the provision of information, services, transactions, and/or the like across various communications networks.

[0222] The CLAP may transform inputs via CLAP components into outputs and/or the like and use of the CLAP. In one embodiment, the CLAP component **2235** takes inputs (e.g., content seed, play count data, event data, triggers, and/or the like) etc., and transforms the inputs via various components (e.g., discovery component, play count reporting component, license verification component, and/or the like), into outputs (e.g., search results, royalties, license verification, and/or the like).

[0223] The CLAP component enabling access of information between nodes may be developed by employing standard development tools and languages such as, but not limited to: Apache components, Assembly, ActiveX, binary executables, (ANSI) (Objective-) C (++), C# and/or .NET, database adapters, CGI scripts, Java, JavaScript, mapping tools, procedural and object oriented development tools, PERL, PHP, Python, shell scripts, SQL commands, web application server extensions, web development environments and libraries (e.g., Microsoft's ActiveX; Adobe AIR, FLEX & FLASH; AJAX; (D)HTML; Dojo, Java; JavaScript; jQuery(UI); MooTools; Prototype; script.aculo.us; Simple Object Access Protocol (SOAP); SWFObject; Yahoo! User Interface; and/or the like), WebObjects, and/or the like. In one embodiment, the CLAP server employs a cryptographic server to encrypt and decrypt communications. The CLAP component may communicate to and/or with other components in a component collection, including itself, and/or facilities of the like. Most frequently, the CLAP component communicates with the CLAP database, operating systems, other program components, and/or the like. The CLAP may contain, communicate, generate, obtain, and/or provide program component, system, user, and/or data communications, requests, and/or responses.

Distributed CLAPs

[0224] The structure and/or operation of any of the CLAP node controller components may be combined, consolidated, and/or distributed in any number of ways to facilitate development and/or deployment. Similarly, the component collection may be combined in any number of ways to facilitate deployment and/or development. To accomplish this, one may integrate the components into a common code base or in a facility that can dynamically load the components on demand in an integrated fashion.

[0225] The component collection may be consolidated and/or distributed in countless variations through standard data processing and/or development techniques. Multiple instances of any one of the program components in the program component collection may be instantiated on a single node, and/or across numerous nodes to improve performance through load-balancing and/or data-processing techniques. Furthermore, single instances may also be distributed across multiple controllers and/or storage devices; e.g., databases. All program component instances and controllers working in concert may do so through standard data processing communication techniques.

[0226] The configuration of the CLAP controller will depend on the context of system deployment. Factors such as, but not limited to, the budget, capacity, location, and/or use of

the underlying hardware resources may affect deployment requirements and configuration. Regardless of if the configuration results in more consolidated and/or integrated program components, results in a more distributed series of program components, and/or results in some combination between a consolidated and distributed configuration, data may be communicated, obtained, and/or provided. Instances of components consolidated into a common code base from the program component collection may communicate, obtain, and/or provide data. This may be accomplished through intra-application data processing communication techniques such as, but not limited to: data referencing (e.g., pointers), internal messaging, object instance variable communication, shared memory space, variable passing, and/or the like.

[0227] If component collection components are discrete, separate, and/or external to one another, then communicating, obtaining, and/or providing data with and/or to other component components may be accomplished through inter-application data processing communication techniques such as, but not limited to: Application Program Interfaces (API) information passage; (distributed) Component Object Model ((D)COM), (Distributed) Object Linking and Embedding ((D)OLE), and/or the like), Common Object Request Broker Architecture (CORBA), Jini local and remote application program interfaces, JavaScript Object Notation (JSON), Remote Method Invocation (RMI), SOAP, process pipes, shared files, and/or the like. Messages sent between discrete component components for inter-application communication or within memory spaces of a singular component for intra-application communication may be facilitated through the creation and parsing of a grammar. A grammar may be developed by using development tools such as lex, yacc, XML, and/or the like, which allow for grammar generation and parsing capabilities, which in turn may form the basis of communication messages within and between components.

[0228] For example, a grammar may be arranged to recognize the tokens of an HTTP post command, e.g.:

[0229] w3c -post http:// . . . Value1

[0230] where Value1 is discerned as being a parameter because "http://" is part of the grammar syntax, and what follows is considered part of the post value. Similarly, with such a grammar, a variable "Value1" may be inserted into an "http://" post command and then sent. The grammar syntax itself may be presented as structured data that is interpreted and/or otherwise used to generate the parsing mechanism (e.g., a syntax description text file as processed by lex, yacc, etc.). Also, once the parsing mechanism is generated and/or instantiated, it itself may process and/or parse structured data such as, but not limited to: character (e.g., tab) delineated text, HTML, structured text streams, XML, and/or the like structured data. In another embodiment, inter-application data processing protocols themselves may have integrated and/or readily available parsers (e.g., JSON, SOAP, and/or like parsers) that may be employed to parse (e.g., communications) data. Further, the parsing grammar may be used beyond message parsing, but may also be used to parse: databases, data collections, data stores, structured data, and/or the like. Again, the desired configuration will depend upon the context, environment, and requirements of system deployment.

[0231] For example, in some implementations, the CLAP controller may be executing a PHP script implementing a Secure Sockets Layer ("SSL") socket server via the information sherver, which listens to incoming communications on a server port to which a client may send data, e.g., data encoded in JSON format. Upon identifying an incoming communication, the PHP script may read the incoming message from the client device, parse the received JSON-encoded text data to extract information from the JSON-encoded text data into PHP script variables, and store the data (e.g., client identifying information, etc.) and/or extracted information in a relational database accessible using the Structured Query Language ("SQL"). An exemplary listing, written substantially in the form of PHP/SQL commands, to accept JSON-encoded input data from a client device via a SSL connection, parse the data to extract variables, and store the data to a database, is provided below:

```
<?PHP
header('Content-Type: text/plain');
// set ip address and port to listen to for incoming data
$address = '192.168.0.100';
$port = 255;
// create a server-side SSL socket, listen for/accept incoming
communication
$sock = socket_create(AF_INET, SOCK_STREAM, 0);
socket_bind($sock, $address, $port) or die('Could not bind to address');
socket_listen($sock);
$client = socket_accept($sock);
// read input data from client device in 1024 byte blocks until end of
message
do {
        $input = "";
        $input = socket_read($client, 1024);
        $data .= $input;
} while($input != "");
// parse data to extract variables
$obj = json_decode($data, true);
// store input data in a database
mysql_connect("201.408.185.132",$DBserver,$password); // access
database server
mysql_select("CLIENT_DB.SQL"); // select database to append
mysql_query("INSERT INTO UserTable (transmission)
VALUES ($data)"); // add data to UserTable table in  a CLIENT database
mysql_close("CLIENT_DB.SQL"); // close connection to database
?>
```

[0232] Also, the following resources may be used to provide example embodiments regarding SOAP parser implementation:

```
http://www.xav.com/perl/site/lib/SOAP/Parser.html
http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=
/com.ibm.IBMDI.doc/referenceguide295.htm
```

[0233] and other parser implementations:

```
http://publib.boulder.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=
/com.ibm.IBMDI.doc/referenceguide259.htm
```

[0234] all of which are hereby expressly incorporated by reference.

[0235] Additional example embodiments of the CLAP include:

[0236] 1. A non-local content caching processor-implemented method, comprising:

[0237] obtaining a universally resolvable list of content items on a local client;

[0238] identifying a non-local item from the universally resolvable list of content items that is absent on the local client;

[0239] generating a local cache request for the identified non-local item having an associated universally resolvable content identifier;

[0240] transmitting the generated local cache request to a universally resolvable content server;

[0241] receiving, in response to the transmitted request, a universally resolvable content item corresponding to the local cache request; and

[0242] marking the requested item as temporary and locally available upon receiving the content item.

[0243] 2. The method of embodiment 1, wherein the server queries a universally resolvable content database to retrieve the universally resolvable content item.

[0244] 3. The method of embodiment 1, wherein identifying the non-local item includes conducting a search for the non-local item on the local client.

[0245] 4. The method of embodiment 1, wherein the locally available content item is engageable as it is partially downloaded.

[0246] 5. The method of embodiment 1, wherein the locally available content item is engageable as it is fully downloaded.

[0247] 6. The method of embodiment 1, wherein the universally resolvable content identifier is a track identifier (ID).

[0248] 7. The method of embodiment 6, wherein identifying the non-local item includes comparing track identifiers associated local items in the local client to the obtained list of track identifiers.

[0249] 8. The method of embodiment 1, wherein the local cache request includes at least a universally resolvable content service user identifier and a universally resolvable content identifier associated with the identified non-local item.

[0250] 9. The method of embodiment 1, further comprising:

[0251] storing the received universally resolvable content item corresponding to the local cache request in a cache in the local client.

[0252] 10. The method of embodiment 1, further comprising:

[0253] deleting one or more content items in the cache local client prior to the storing.

[0254] 11. The method of embodiment 10, wherein the content items are deleted based on last hit time.

[0255] 12. The method of embodiment 10, wherein the content items are deleted based on content item size.

[0256] 13. The method of embodiment 10, wherein the content items are deleted based on priority.

[0257] 14. The method of embodiment 13, wherein the priority is determined based on user preference.

[0258] 15. The method of embodiment 10, wherein the content items are deleted based on at least one of play count and creation time.

[0259] 16. The method of embodiment 1, wherein the universally resolvable list of content items includes at least one of: (i) a magic playlist, (ii) a dynamically created interest list, (iii) a shared playlist, (iv) a smart cache list, and (v) a shared library.

[0260] 17. The method of embodiment 1, wherein the content items include at least one of music, books, videos, applications, user's media and user's media denoted by gurus, social, friends or favorites.

[0261] 18. A non-local content item caching system, comprising:

[0262] a memory;

[0263] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0264] obtain a universally resolvable list of content items on a local client;

[0265] identify a non-local item from the universally resolvable list of content items that is absent on the local client;

[0266] generate a local cache request for the identified non-local item having an associated universally resolvable content identifier;

[0267] transmit the generated local cache request to a universally resolvable content server;

[0268] receive, in response to the transmitted request, a universally resolvable content item corresponding to the local cache request; and

[0269] mark the requested item as temporary and locally available upon receiving the content item.

[0270] 19. The system of embodiment 18, wherein the server queries a universally resolvable content database to retrieve the universally resolvable content item.

[0271] 20. The system of embodiment 18, wherein identifying the non-local item includes conducting a search for the non-local item on the local client.

[0272] 21. The system of embodiment 18, wherein the locally available content item is engageable as it is partially downloaded.

[0273] 22. The system of embodiment 18, wherein the locally available content item is engageable as it is fully downloaded.

[0274] 23. The system of embodiment 18, wherein the universally resolvable content identifier is a track identifier (ID).

[0275] 24. The system of embodiment 23, wherein identifying the non-local item includes comparing track identifiers associated local items in the local client to the obtained list of track identifiers.

[0276] 25. The system of embodiment 18, wherein the local cache request includes at least a universally resolvable content service user identifier and a universally resolvable content identifier associated with the identified non-local item.

[0277] 26. The system of embodiment 18, wherein the processor issues further instructions to:

[0278] store the received universally resolvable content item corresponding to the local cache request in a cache in the local client.

[0279] 27. The system of embodiment 18, wherein the processor issues further instructions to:

[0280] delete one or more content items in the cache local client prior to the storing.

[0281] 28. The system of embodiment 27, wherein the content items are deleted based on last hit time.

[0282] 29. The system of embodiment 27, wherein the content items are deleted based on content item size.

[0283] 30. The system of embodiment 27, wherein the content items are deleted based on priority.

[0284] 31. The system of embodiment 27, wherein the priority is determined based on user preference.

[0285] 32. The system of embodiment 27 wherein the content items are deleted based on at least one of play count and creation time.

[0286] 33. The system of embodiment 18, wherein the universally resolvable list of content items includes at least one of: (i) a magic playlist, (ii) a dynamically created interest list, (iii) a shared playlist, (iv) a smart cache list, and (v) a shared library.

[0287] 34. The system of embodiment 18, wherein the content items include at least one of music, books, videos, applications, user's media and user's media denoted by gurus, social, friends or favorites.

[0288] 35. A non-local content caching processor-readable medium storing processor-issuable instructions, comprising:

[0289] obtain a universally resolvable list of content items on a local client;

[0290] identify a non-local item from the universally resolvable list of content items that is absent on the local client;

[0291] generate a local cache request for the identified non-local item having an associated universally resolvable content identifier;

[0292] transmit the generated local cache request to a universally resolvable content server;

[0293] receive, in response to the transmitted request, a universally resolvable content item corresponding to the local cache request; and

[0294] mark the requested item as temporary and locally available upon receiving the content item.

[0295] 36. The medium of embodiment 35, wherein the server queries a universally resolvable content database to retrieve the universally resolvable content item.

[0296] 37. The medium of embodiment 35, wherein identifying the non-local item includes conducting a search for the non-local item on the local client.

[0297] 38. The medium of embodiment 35, wherein the locally available content item is engageable as it is partially downloaded.

[0298] 39. The medium of embodiment 35, wherein the locally available content item is engageable as it is fully downloaded.

[0299] 40. The medium of embodiment 35, wherein the universally resolvable content identifier is a track identifier (ID).

[0300] 41. The medium of embodiment 35, wherein identifying the non-local item includes comparing track identifiers associated local items in the local client to the obtained list of track identifiers.

[0301] 42. The medium of embodiment 35, wherein the local cache request includes at least a universally resolvable content service user identifier and a universally resolvable content identifier associated with the identified non-local item.

[0302] 43. The medium of embodiment 35, wherein the processor issues further instructions to:

[0303] store the received universally resolvable content item corresponding to the local cache request in a cache in the local client.

[0304] 44. The medium of embodiment 35, wherein the processor issues further instructions to:

[0305] delete one or more content items in the cache local client prior to the storing.

[0306] 45. The medium of embodiment 44, wherein the content items are deleted based on last hit time.

[0307] 46. The medium of embodiment 44, wherein the content items are deleted based on content item size.

[0308] 47. The medium of embodiment 44, wherein the content items are deleted based on priority.

[0309] 48. The medium of embodiment 47, wherein the priority is determined based on user preference.

[0310] 49. The medium of embodiment 44 wherein the content items are deleted based on at least one of play count and creation time.

[0311] 50. The medium of embodiment 35, wherein the universally resolvable list of content items includes at least one of: (i) a magic playlist, (ii) a dynamically created interest list, (iii) a shared playlist, (iv) a smart cache list, and (v) a shared library.

[0312] 51. The medium of embodiment 35, wherein the content items include at least one of music, books, videos, applications, user's media and user's media denoted by gurus, social, friends or favorites.

[0313] 52. A non-local content item caching apparatus, comprising:

[0314] a memory:

[0315] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0316] obtain a universally resolvable list of content items on a local client;

[0317] identify a non-local item from the universally resolvable list of content items that is absent on the local client;

[0318] generate a local cache request for the identified non-local item having an associated universally resolvable content identifier;

[0319] transmit the generated local cache request to a universally resolvable content server;

[0320] receive, in response to the transmitted request, a universally resolvable content item corresponding to the local cache request; and

[0321] mark the requested item as temporary and locally available upon receiving the content item.

[0322] 53. The apparatus of embodiment 52, wherein the server queries a universally resolvable content database to retrieve the universally resolvable content item.

[0323] 54. The apparatus of embodiment 52, wherein identifying the non-local item includes conducting a search for the non-local item on the local client.

[0324] 55. The apparatus of embodiment 52, wherein the locally available content item is engageable as it is partially downloaded.

[0325] 56. The apparatus of embodiment 52, wherein the locally available content item is engageable as it is fully downloaded.

[0326] 57. The apparatus of embodiment 52, wherein the universally resolvable content identifier is a track identifier (ID).

[0327] 58. The apparatus of embodiment 52, wherein identifying the non-local item includes comparing track identifiers associated local items in the local client to the obtained list of track identifiers.

[0328] 59. The apparatus of embodiment 52, wherein the local cache request includes at least a universally resolvable content service user identifier and a universally resolvable content identifier associated with the identified non-local item.

[0329] 60. The apparatus of embodiment 52, wherein the processor issues further instructions to:

[0330] store the received universally resolvable content item corresponding to the local cache request in a cache in the local client.

[0331] 61. The apparatus of embodiment 52, wherein the processor issues further instructions to:

[0332] delete one or more content items in the cache local client prior to the storing.

[0333] 62. The apparatus of embodiment 61, wherein the content items are deleted based on last hit time.

[0334] 63. The apparatus of embodiment 61, wherein the content items are deleted based on content item size.

[0335] 64. The apparatus of embodiment 61, wherein the content items are deleted based on priority.

[0336] 65. The apparatus of embodiment 64, wherein the priority is determined based on user preference.

[0337] 66. The apparatus of embodiment 61 wherein the content items are deleted based on at least one of play count and creation time.

[0338] 67. The system of embodiment 52, wherein the universally resolvable list of content items includes at least one of: (i) a magic playlist, (ii) a dynamically created interest list, (iii) a shared playlist, (iv) a smart cache list, and (v) a shared library.

[0339] 68. The system of embodiment 52, wherein the content items include at least one of music, books, videos, applications, user's media and user's media denoted by gurus, social, friends or favorites.

[0340] 69. A non-local content caching processor-implemented method, comprising:

[0341] providing a universally resolvable list of content items to a client;

[0342] obtaining an identification of a non-local item from the universally resolvable list of content items that is absent from the client;

[0343] obtaining a cache request for the identified non-local item having an associated universally resolvable content identifier;

[0344] providing, in response to the obtained request, a universally resolvable content item corresponding to the cache request; and

[0345] updating the requested item as temporary and locally available.

[0346] 70. The method of embodiment 69, further comprising querying a universally resolvable content database to retrieve the universally resolvable content item.

[0347] 71. The method of embodiment 69, wherein obtaining the identification of the non-local item includes conducting a search for the non-local item on the local client.

[0348] 72. The method of embodiment 69, wherein the locally available content item is engageable as it is partially downloaded.

[0349] 73. The method of embodiment 69, wherein the locally available content item is engageable as it is fully downloaded.

[0350] 74. The method of embodiment 69, wherein the universally resolvable content identifier is a track identifier (ID).

[0351] 75. The method of embodiment 74, wherein obtaining the identification of the non-local item includes comparing track identifiers associated local items in the local client to the obtained list of track identifiers.

[0352] 76. The method of embodiment 69, wherein the cache request includes at least a universally resolvable content service user identifier and a universally resolvable content identifier associated with the identified non-local item.

[0353] 77. The method of embodiment 69, further comprising:

[0354] providing the universally resolvable content item corresponding to the cache request for storage in a cache in the client.

[0355] 78. The method of embodiment 77, further comprising:

[0356] obtaining an indication of deletion of one or more content items in the cache of the client prior to providing the content item for storage.

[0357] 79. The method of embodiment 78 wherein the content items are deleted based on last hit time.

[0358] 80. The method of embodiment 78, wherein the content items are deleted based on content item size.

[0359] 81. The method of embodiment 78, wherein the content items are deleted based on priority.

[0360] 82. The method of embodiment 81, wherein the priority is determined based on user preference.

[0361] 83. The method of embodiment 78, wherein the content items are deleted based on at least one of play count and creation time.

[0362] 84. The method of embodiment 69, wherein the universally resolvable list of content items includes at least one of: (i) a magic playlist, (ii) a dynamically created interest list, (iii) a shared playlist, (iv) a smart cache list, and (v) a shared library.

[0363] 85. The method of embodiment 69, wherein the content items include at least one of music, books, videos, applications, user's media and user's media denoted by gurus, social, friends or favorites.

[0364] 86. A non-local content item caching system, comprising:

[0365] a memory;

[0366] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0367] provide a universally resolvable list of content items to a client;

[0368] obtain an identification of a non-local item from the universally resolvable list of content items that is absent from the client;

[0369] obtain a cache request for the identified non-local item having an associated universally resolvable content identifier;

[0370] provide, in response to the obtained request, a universally resolvable content item corresponding to the cache request; and

[0371] update the requested item as temporary and locally available.

[0372] 87. The system of embodiment 86, wherein the processor issues further instructions to query a universally resolvable content database to retrieve the universally resolvable content item.

[0373] 88. The system of embodiment 86, wherein the processor issues further instructions to obtain the identification of the non-local item includes conducting a search for the non-local item on the local client.

[0374] 89. The system of embodiment 86, wherein the locally available content item is engageable as it is partially downloaded.

[0375] 90. The system of embodiment 86, wherein the locally available content item is engageable as it is fully downloaded.

[0376] 91. The system of embodiment 86, wherein the universally resolvable content identifier is a track identifier (ID).

[0377] 92. The system of embodiment 91, wherein the instructions to obtain the identification of the non-local item includes instructions to compare track identifiers associated local items in the local client to the obtained list of track identifiers.

[0378] 93. The system of embodiment 86, wherein the cache request includes at least a universally resolvable content service user identifier and a universally resolvable content identifier associated with the identified non-local item.

[0379] 94. The system of embodiment 86, wherein the processor issues further instructions to:

[0380] provide the universally resolvable content item corresponding to the cache request for storage in a cache in the client.

[0381] 95. The system of embodiment 94, wherein the processor issues further instructions to:

[0382] obtaining an indication of deletion of one or more content items in the cache of the client prior to providing the content item for storage.

[0383] 96. The system of embodiment 95 wherein the content items are deleted based on last hit time.

[0384] 97. The system of embodiment 95, wherein the content items are deleted based on content item size.

[0385] 98. The system of embodiment 95, wherein the content items are deleted based on priority.

[0386] 99. The system of embodiment 81, wherein the priority is determined based on user preference.

[0387] 100. The system of embodiment 95, wherein the content items are deleted based on at least one of play count and creation time.

[0388] 101. The system of embodiment 86, wherein the universally resolvable list of content items includes at least one of: (i) a magic playlist, (ii) a dynamically created interest list, (iii) a shared playlist, (iv) a smart cache list, and (v) a shared library.

[0389] 102. The system of embodiment 86, wherein the content items include at least one of music, books, videos, applications, user's media and user's media denoted by gurus, social, friends or favorites.

[0390] 103. A non-local content caching processor-readable medium storing processor-issuable instructions, comprising:

[0391] provide a universally resolvable list of content items to a client;

[0392] obtain an identification of a non-local item from the universally resolvable list of content items that is absent from the client;

[0393] obtain a cache request for the identified non-local item having an associated universally resolvable content identifier;

[0394] provide, in response to the obtained request, a universally resolvable content item corresponding to the cache request; and

[0395] update the requested item as temporary and locally available.

[0396] 104. The medium of embodiment 103, wherein the processor issues further instructions to query a universally resolvable content database to retrieve the universally resolvable content item.

[0397] 105. The medium of embodiment 103, wherein the processor issues further instructions to obtain the identification of the non-local item includes conducting a search for the non-local item on the local client.

[0398] 106. The medium of embodiment 103, wherein the locally available content item is engageable as it is partially downloaded.

[0399] 107. The medium of embodiment 103, wherein the locally available content item is engageable as it is fully downloaded.

[0400] 108. The medium of embodiment 103, wherein the universally resolvable content identifier is a track identifier (ID).

[0401] 109. The medium of embodiment 108, wherein the instructions to obtain the identification of the non-local item includes instructions to compare track identifiers associated local items in the local client to the obtained list of track identifiers.

[0402] 110. The medium of embodiment 103, wherein the cache request includes at least a universally resolvable content service user identifier and a universally resolvable content identifier associated with the identified non-local item.

[0403] 111. The medium of embodiment 103, wherein the processor issues further instructions to:

[0404] provide the universally resolvable content item corresponding to the cache request for storage in a cache in the client.

[0405] 112. The medium of embodiment in, wherein the processor issues further instructions to:

[0406] obtaining an indication of deletion of one or more content items in the cache of the client prior to providing the content item for storage.

[0407] 113. The medium of embodiment 112 wherein the content items are deleted based on last hit time.

[0408] 114. The medium of embodiment 112, wherein the content items are deleted based on content item size.

[0409] 115. The medium of embodiment 112, wherein the content items are deleted based on priority.

[0410] 116. The medium of embodiment 115, wherein the priority is determined based on user preference.

[0411] 117. The medium of embodiment 112, wherein the content items are deleted based on at least one of play count and creation time.

[0412] 118. The medium of embodiment 103, wherein the universally resolvable list of content items includes at least one of: (i) a magic playlist, (ii) a dynamically created interest list, (iii) a shared playlist, (iv) a smart cache list, and (v) a shared library.

[0413] 119. The medium of embodiment 103, wherein the content items include at least one of music, books, videos, applications, user's media and user's media denoted by gurus, social, friends or favorites.

[0414] 120. An apparatus, comprising:

[0415] a memory;

[0416] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0417] provide a universally resolvable list of content items to a client;

[0418] obtain an identification of a non-local item from the universally resolvable list of content items that is absent from the client;

[0419] obtain a cache request for the identified non-local item having an associated universally resolvable content identifier;

[0420] provide, in response to the obtained request, a universally resolvable content item corresponding to the cache request; and

[0421] update the requested item as temporary and locally available.

[0422] 121. The apparatus of embodiment 120, wherein the processor issues further instructions to query a universally resolvable content database to retrieve the universally resolvable content item.

[0423] 122. The apparatus of embodiment 120, wherein the processor issues further instructions to obtain the identification of the non-local item includes conducting a search for the non-local item on the local client.

[0424] 123. The apparatus of embodiment 120, wherein the locally available content item is engageable as it is partially downloaded.

[0425] 124. The apparatus of embodiment 120, wherein the locally available content item is engageable as it is fully downloaded.

[0426] 125. The apparatus of embodiment 120, wherein the universally resolvable content identifier is a track identifier (ID).

[0427] 126. The apparatus of embodiment 125, wherein the instructions to obtain the identification of the non-local item includes instructions to compare track identifiers associated local items in the local client to the obtained list of track identifiers.

[0428] 127. The apparatus of embodiment 120, wherein the cache request includes at least a universally resolvable content service user identifier and a universally resolvable content identifier associated with the identified non-local item.

[0429] 128. The apparatus of embodiment 120, wherein the processor issues further instructions to:

[0430] provide the universally resolvable content item corresponding to the cache request for storage in a cache in the client.

[0431] 129. The apparatus of embodiment 128, wherein the processor issues further instructions to:

[0432] obtain an indication of deletion of one or more content items in the cache of the client prior to providing the content item for storage.

[0433] 130. The apparatus of embodiment 129 wherein the content items are deleted based on last hit time.

[0434] 131. The apparatus of embodiment 129, wherein the content items are deleted based on content item size.

[0435] 132. The apparatus of embodiment 129, wherein the content items are deleted based on priority.

[0436] 133. The apparatus of embodiment 132, wherein the priority is determined based on user preference.

[0437] 134. The apparatus of embodiment 129, wherein the content items are deleted based on at least one of play count and creation time.

[0438] 135. The apparatus of embodiment 120, wherein the universally resolvable list of content items includes at least one of: (i) a magic playlist, (ii) a dynamically created interest list, (iii) a shared playlist, (iv) a smart cache list, and (v) a shared library.

[0439] 136. The apparatus of embodiment 120, wherein the content items include at least one of music, books, videos, applications, user's media and user's media denoted by gurus, social, friends or favorites.

[0440] 137. An apportionment heuristics based caching processor-implemented method, comprising:

[0441] obtaining content discovery supportive information for a universally resolvable user;

[0442] determining apportionment heuristics among the obtained information for the user;

[0443] identifying a first set of universally resolvable content items based on the determined apportionment heuristics;

[0444] creating a caching queue that includes the identified first set of universally resolvable content items; and

[0445] providing the first set of universally resolvable content items in the caching queue to the user.

[0446] 138. The method of embodiment 137, wherein providing the first set of universally resolvable content items is in response to a request for transmission that is triggered when a client device bandwidth usage is below a pre-determined threshold.

[0447] 139. The method of embodiment 137, wherein providing the first set of universally resolvable content items is in response to a request for transmission that is triggered in accordance with user specified caching criteria.

[0448] 140. The method of embodiment 137, wherein the first set of universally resolvable content items are arranged in a predefined download order in the caching queue.

[0449] 141. The method of embodiment 137, wherein the apportionment heuristics includes the user's entity graph.

[0450] 142. The method of embodiment 141, wherein the user's entity graph includes at least one of a social graph and an interest graph.

[0451] 143. The method of embodiment 137, wherein the apportionment heuristics includes user-specific usage.

[0452] 144. The method of embodiment 137, wherein the apportionment heuristics includes aggregate usage.

[0453] 145. The method of embodiment 137, wherein the apportionment heuristics includes preference profile.

[0454] 146. The method of embodiment 145, wherein the preference profile is associated with at least one of a user or a group of users, and indicative of preference for at least one of: (i) genres, (ii) artists, (iii) albums, (iv) tracks, (v) music attributes, (vi) location based preferences, and (vii) time based preferences.

[0455] 147. The method of embodiment 137, wherein the apportionment heuristics includes social recommendation.

[0456] 148. The method of embodiment 137, wherein the content discovery supportive information is updated based on an activity associated with one or more users.

[0457] 149. The method of embodiment 148, further comprising:

[0458] obtaining the updated content discovery supportive information for the universally resolvable content user;

[0459] determining updated apportionment heuristics among the obtained updated information for the user;

[0460] identifying a second set of universally resolvable content items based on the determined updated apportionment heuristics;

[0461] updating the caching queue that includes the identified second set of universally resolvable content items; and

[0462] providing the second set of universally resolvable content items in the updated caching queue to the user.

[0463] 150. The method of embodiment 149, wherein the second set of universally resolvable content items includes at least one content item from the first set of universally resolvable content items.

[0464] 151. The method of embodiment 137, wherein the content discovery supportive information includes at least one of: most frequently played content item, content item rated high, content item rated low, content item shared and content item bookmarked.

[0465] 152. An apportionment heuristics based caching system, comprising:

[0466] a memory;

[0467] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0468] obtain content discovery supportive information for a universally resolvable user;

[0469] determine apportionment heuristics among the obtained information for the user;

[0470] identify a first set of universally resolvable content items based on the determined apportionment heuristics;

[0471] create a caching queue that includes the identified first set of universally resolvable content items; and

[0472] provide the first set of universally resolvable content items in the caching queue to the user.

[0473] 153. The system of embodiment 152, wherein providing the first set of universally resolvable content items is in response to a request for transmission that is triggered when a client device bandwidth usage is below a pre-determined threshold.

[0474] 154. The system of embodiment 152, wherein the instructions to provide the first set of universally resolvable content items is in response to a request for transmission that is triggered in accordance with user specified caching criteria.

[0475] 155. The system of embodiment 152, wherein the first set of universally resolvable content items are arranged in a predefined download order in the caching queue.

[0476] 156. The system of embodiment 152, wherein the apportionment heuristics includes the user's entity graph.

[0477] 157. The system of embodiment 156, wherein the user's entity graph includes at least one of a social graph and an interest graph.

[0478] 158. The system of embodiment 152, wherein the apportionment heuristics includes user-specific usage.

[0479] 159. The system of embodiment 152, wherein the apportionment heuristics includes aggregate usage.

[0480] 160. The system of embodiment 152, wherein the apportionment heuristics includes preference profile.

[0481] 161. The system of embodiment 160, wherein the preference profile is associated with at least one of a user or a group of users, and indicative of preference for at least one of: (i) genres, (ii) artists, (iii) albums, (iv) tracks, (v) music attributes, (vi) location based preferences, and (vii) time based preferences.

[0482] 162. The system of embodiment 152, wherein the apportionment heuristics includes social recommendation.

[0483] 163. The system of embodiment 152, wherein the content discovery supportive information is updated based on an activity associated with one or more users.

[0484] 164. The system of embodiment 163, wherein the processor issues further instructions to:

[0485] obtain the updated content discovery supportive information for the universally resolvable content user;

[0486] determine updated apportionment heuristics among the obtained updated information for the user;

[0487] identify a second set of universally resolvable content items based on the determined updated apportionment heuristics;

[0488] update the caching queue that includes the identified second set of universally resolvable content items; and

[0489] provide the second set of universally resolvable content items in the updated caching queue to the user.

[0490] 165. The system of embodiment 164, wherein the second set of universally resolvable content items includes at least one content item from the first set of universally resolvable content items.

[0491] 166. The system of embodiment 152, wherein the content discovery supportive information includes at least one of: most frequently played content item, content item rated high, content item rated low, content item shared and content item bookmarked.

[0492] 167. An apportionment heuristics based caching processor-readable medium storing processor-issuable instructions, comprising:

[0493] obtain content discovery supportive information for a universally resolvable user;

[0494] determine apportionment heuristics among the obtained information for the user;

[0495] identify a first set of universally resolvable content items based on the determined apportionment heuristics;

[0496] create a caching queue that includes the identified first set of universally resolvable content items; and

[0497] provide the first set of universally resolvable content items in the caching queue to the user.

[0498] 168. The medium of embodiment 167, wherein providing the first set of universally resolvable content items is in response to a request for transmission that is triggered when a client device bandwidth usage is below a pre-determined threshold.

[0499] 169. The medium of embodiment 167, wherein the instructions to provide the first set of universally resolvable content items is in response to a request for transmission that is triggered in accordance with user specified caching criteria.

[0500] 170. The medium of embodiment 167 wherein the first set of universally resolvable content items are arranged in a predefined download order in the caching queue.

[0501] 171. The medium of embodiment 167, wherein the apportionment heuristics includes the user's entity graph.

[0502] 172. The medium of embodiment 171, wherein the user's entity graph includes at least one of a social graph and an interest graph.

[0503] 173. The medium of embodiment 167, wherein the apportionment heuristics includes user-specific usage.

[0504] 174. The medium of embodiment 167, wherein the apportionment heuristics includes aggregate usage.

[0505] 175. The medium of embodiment 167, wherein the apportionment heuristics includes preference profile.

[0506] 176. The medium of embodiment 175, wherein the preference profile is associated with at least one of a user or a group of users, and indicative of preference for at least one of:

(i) genres, (ii) artists, (iii) albums, (iv) tracks, (v) music attributes, (vi) location based preferences, and (vii) time based preferences.

[0507] 177. The medium of embodiment 167, wherein the apportionment heuristics includes social recommendation.

[0508] 178. The medium of embodiment 167, wherein the content discovery supportive information is updated based on an activity associated with one or more users.

[0509] 179. The medium of embodiment 178, wherein the processor issues further instructions to:

[0510] obtain the updated content discovery supportive information for the universally resolvable content user;

[0511] determine updated apportionment heuristics among the obtained updated information for the user;

[0512] identify a second set of universally resolvable content items based on the determined updated apportionment heuristics;

[0513] update the caching queue that includes the identified second set of universally resolvable content items; and

[0514] provide the second set of universally resolvable content items in the updated caching queue to the user.

[0515] 180. The medium of embodiment 179, wherein the second set of universally resolvable content items includes at least one content item from the first set of universally resolvable content items.

[0516] 181. The medium of embodiment 167, wherein the content discovery supportive information includes at least one of: most frequently played content item, content item rated high, content item rated low, content item shared and content item bookmarked.

[0517] 182. An apportionment heuristics based caching apparatus, comprising:

[0518] a memory;

[0519] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0520] obtain content discovery supportive information for a universally resolvable user;

[0521] determine apportionment heuristics among the obtained information for the user;

[0522] identify a first set of universally resolvable content items based on the determined apportionment heuristics;

[0523] create a caching queue that includes the identified first set of universally resolvable content items; and

[0524] provide the first set of universally resolvable content items in the caching queue to the user.

[0525] 183. The apparatus of embodiment 182, wherein providing the first set of universally resolvable content items is in response to a request for transmission that is triggered when a client device bandwidth usage is below a pre-determined threshold.

[0526] 184. The apparatus of embodiment 182, wherein the instructions to provide the first set of universally resolvable content items is in response to a request for transmission that is triggered in accordance with user specified caching criteria.

[0527] 185. The apparatus of embodiment 182, wherein the first set of universally resolvable content items are arranged in a predefined download order in the caching queue.

[0528] 186. The apparatus of embodiment 182, wherein the apportionment heuristics includes the user's entity graph.

[0529] 187. The apparatus of embodiment 186, wherein the user's entity graph includes at least one of a social graph and an interest graph.

[0530] 188. The apparatus of embodiment 182, wherein the apportionment heuristics includes user-specific usage.

[0531] 189. The apparatus of embodiment 182, wherein the apportionment heuristics includes aggregate usage.

[0532] 190. The apparatus of embodiment 182, wherein the apportionment heuristics includes preference profile.

[0533] 191. The apparatus of embodiment 160, wherein the preference profile is associated with at least one of a user or a group of users, and indicative of preference for at least one of: (i) genres, (ii) artists, (iii) albums, (iv) tracks, (v) music attributes, (vi) location based preferences, and (vii) time based preferences.

[0534] 192. The apparatus of embodiment 182, wherein the apportionment heuristics includes social recommendation.

[0535] 193. The apparatus of embodiment 182, wherein the content discovery supportive information is updated based on an activity associated with one or more users.

[0536] 194. The apparatus of embodiment 193, wherein the processor issues further instructions to:

[0537] obtain the updated content discovery supportive information for the universally resolvable content user;

[0538] determine updated apportionment heuristics among the obtained updated information for the user;

[0539] identify a second set of universally resolvable content items based on the determined updated apportionment heuristics;

[0540] update the caching queue that includes the identified second set of universally resolvable content items; and

[0541] provide the second set of universally resolvable content items in the updated caching queue to the user.

[0542] 195. The apparatus of embodiment 194, wherein the second set of universally resolvable content items includes at least one content item from the first set of universally resolvable content items.

[0543] 196. The apparatus of embodiment 182, wherein the content discovery supportive information includes at least one of: most frequently played content item, content item rated high, content item rated low, content item shared and content item bookmarked.

[0544] 197. An apportionment heuristics based caching processor-implemented method, comprising:

[0545] providing content discovery supportive information for a universally resolvable user;

[0546] providing an indication of apportionment heuristics among the obtained information for the user;

[0547] obtaining an identification of a first set of universally resolvable content items based on the determined apportionment heuristics;

[0548] obtaining an indication of creation of a caching queue that includes the identified first set of universally resolvable content items; and

[0549] obtaining the first set of universally resolvable content items in the caching queue to the user.

[0550] 198. The method of embodiment 197, wherein obtaining the first set of universally resolvable content items is in response to a request for transmission that is triggered when a client device bandwidth usage is below a pre-determined threshold.

[0551] 199. The method of embodiment 197, wherein obtaining the first set of universally resolvable content items

is in response to a request for transmission that is triggered in accordance with user specified caching criteria.

[0552] 200. The method of embodiment 197, wherein the first set of universally resolvable content items are arranged in a predefined download order in the caching queue.

[0553] 201. The method of embodiment 197, wherein the apportionment heuristics includes the user's entity graph.

[0554] 202. The method of embodiment 201, wherein the user's entity graph includes at least one of a social graph and an interest graph.

[0555] 203. The method of embodiment 197, wherein the apportionment heuristics includes user-specific usage.

[0556] 204. The method of embodiment 197, wherein the apportionment heuristics includes aggregate usage.

[0557] 205. The method of embodiment 197, wherein the apportionment heuristics includes preference profile.

[0558] 206. The method of embodiment 205, wherein the preference profile is associated with at least one of a user or a group of users, and indicative of preference for at least one of: (i) genres, (ii) artists, (iii) albums, (iv) tracks, (v) music attributes, (vi) location based preferences, and (vii) time based preferences.

[0559] 207. The method of embodiment 197, wherein the apportionment heuristics includes social recommendation.

[0560] 208. The method of embodiment 197, wherein the content discovery supportive information is updated based on an activity associated with one or more users.

[0561] 209. The method of embodiment 208, further comprising:

[0562] providing the updated content discovery supportive information for the universally resolvable content user;

[0563] obtaining an indication of determination of updated apportionment heuristics among the obtained updated information for the user;

[0564] obtaining an indication of identification of a second set of universally resolvable content items based on the determined updated apportionment heuristics;

[0565] obtaining the caching queue that includes the identified second set of universally resolvable content items; and

[0566] obtaining the second set of universally resolvable content items in the updated caching queue to the user.

[0567] 210. The method of embodiment 209, wherein the second set of universally resolvable content items includes at least one content item from the first set of universally resolvable content items.

[0568] 211. The method of embodiment 197, wherein the content discovery supportive information includes at least one of: most frequently played content item, content item rated high, content item rated low, content item shared and content item bookmarked.

[0569] 212. An apportionment heuristics based caching system, comprising:

[0570] a memory;

[0571] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0572] provide content discovery supportive information for a universally resolvable user;

[0573] obtain an indication of determination of apportionment heuristics among the obtained information for the user;

[0574] obtain an indication of identification of a first set of universally resolvable content items based on the determined apportionment heuristics;

[0575] obtain a caching queue that includes the identified first set of universally resolvable content items; and

[0576] obtain the first set of universally resolvable content items in the caching queue to the user.

[0577] 213. The system of embodiment 212, wherein obtaining the first set of universally resolvable content items is in response to a request for transmission that is triggered when a client device bandwidth usage is below a pre-determined threshold.

[0578] 214. The system of embodiment 212, wherein the instructions to obtain the first set of universally resolvable content items is in response to a request for transmission that is triggered in accordance with user specified caching criteria.

[0579] 215. The system of embodiment 212, wherein the first set of universally resolvable content items are arranged in a predefined download order in the caching queue.

[0580] 216. The system of embodiment 212, wherein the apportionment heuristics includes the user's entity graph.

[0581] 217. The system of embodiment 216, wherein the user's entity graph includes at least one of a social graph and an interest graph.

[0582] 218. The system of embodiment 212, wherein the apportionment heuristics includes user-specific usage.

[0583] 219. The system of embodiment 212, wherein the apportionment heuristics includes aggregate usage.

[0584] 220. The system of embodiment 212, wherein the apportionment heuristics includes preference profile.

[0585] 221. The system of embodiment 220, wherein the preference profile is associated with at least one of a user or a group of users, and indicative of preference for at least one of: (i) genres, (ii) artists, (iii) albums, (iv) tracks, (v) music attributes, (vi) location based preferences, and (vii) time based preferences.

[0586] 222. The system of embodiment 212, wherein the apportionment heuristics includes social recommendation.

[0587] 223. The system of embodiment 212, wherein the content discovery supportive information is updated based on an activity associated with one or more users.

[0588] 224. The system of embodiment 223, wherein the processor issues further instructions to:

[0589] provide the updated content discovery supportive information for the universally resolvable content user;

[0590] obtain an indication of determination of updated apportionment heuristics among the obtained updated information for the user;

[0591] obtain an indication of identification of a second set of universally resolvable content items based on the determined updated apportionment heuristics;

[0592] obtaining an indication of update of the caching queue that includes the identified second set of universally resolvable content items; and

[0593] obtaining the second set of universally resolvable content items in the updated caching queue to the user.

[0594] 225. The system of embodiment 224, wherein the second set of universally resolvable content items includes at least one content item from the first set of universally resolvable content items.

[0595] 226. The system of embodiment 212, wherein the content discovery supportive information includes at least

one of: most frequently played content item, content item rated high, content item rated low, content item shared and content item bookmarked.

[0596] 227. An apportionment heuristics based caching processor-readable medium storing processor-issuable instructions, comprising:

[0597] provide content discovery supportive information for a universally resolvable user;

[0598] obtain apportionment heuristics among the obtained information for the user;

[0599] obtain a first set of universally resolvable content items based on the determined apportionment heuristics;

[0600] obtain a caching queue that includes the identified first set of universally resolvable content items; and

[0601] obtain the first set of universally resolvable content items in the caching queue to the user.

[0602] 228. The medium of embodiment 227, wherein the instructions to obtain the first set of universally resolvable content items is in response to a request for transmission that is triggered when a client device bandwidth usage is below a pre-determined threshold.

[0603] 229. The medium of embodiment 227, wherein the instructions to provide the first set of universally resolvable content items is in response to a request for transmission that is triggered in accordance with user specified caching criteria.

[0604] 230. The medium of embodiment 227 wherein the first set of universally resolvable content items are arranged in a predefined download order in the caching queue.

[0605] 231. The medium of embodiment 227, wherein the apportionment heuristics includes the user's entity graph.

[0606] 232. The medium of embodiment 231, wherein the user's entity graph includes at least one of a social graph and an interest graph.

[0607] 233. The medium of embodiment 227, wherein the apportionment heuristics includes user-specific usage.

[0608] 234. The medium of embodiment 227, wherein the apportionment heuristics includes aggregate usage.

[0609] 235. The medium of embodiment 227, wherein the apportionment heuristics includes preference profile.

[0610] 236. The medium of embodiment 235, wherein the preference profile is associated with at least one of a user or a group of users, and indicative of preference for at least one of: (i) genres, (ii) artists, (iii) albums, (iv) tracks, (v) music attributes, (vi) location based preferences, and (vii) time based preferences.

[0611] 237. The medium of embodiment 227, wherein the apportionment heuristics includes social recommendation.

[0612] 238. The medium of embodiment 227, wherein the content discovery supportive information is updated based on an activity associated with one or more users.

[0613] 239. The medium of embodiment 238, wherein the processor issues further instructions to:

[0614] provide the updated content discovery supportive information for the universally resolvable content user;

[0615] obtain an indication of determination of updated apportionment heuristics among the obtained updated information for the user;

[0616] obtain an identification of a second set of universally resolvable content items based on the determined updated apportionment heuristics;

[0617] obtain an indication to update the caching queue that includes the identified second set of universally resolvable content items; and

[0618] obtain the second set of universally resolvable content items in the updated caching queue to the user.

[0619] 240. The medium of embodiment 239, wherein the second set of universally resolvable content items includes at least one content item from the first set of universally resolvable content items.

[0620] 241. The medium of embodiment 227, wherein the content discovery supportive information includes at least one of: most frequently played content item, content item rated high, content item rated low, content item shared and content item bookmarked.

[0621] 242. An apportionment heuristics based caching apparatus, comprising:

[0622] a memory;

[0623] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0624] provide content discovery supportive information for a universally resolvable user;

[0625] obtain an indication of determination of apportionment heuristics among the obtained information for the user;

[0626] obtain an indication of identification of a first set of universally resolvable content items based on the determined apportionment heuristics;

[0627] obtain a caching queue that includes the identified first set of universally resolvable content items; and

[0628] obtain the first set of universally resolvable content items in the caching queue to the user.

[0629] 243. The apparatus of embodiment 242, wherein providing the first set of universally resolvable content items is in response to a request for transmission that is triggered when a client device bandwidth usage is below a pre-determined threshold.

[0630] 244. The apparatus of embodiment 242, wherein the instructions to provide the first set of universally resolvable content items is in response to a request for transmission that is triggered in accordance with user specified caching criteria.

[0631] 245. The apparatus of embodiment 242, wherein the first set of universally resolvable content items are arranged in a predefined download order in the caching queue.

[0632] 246. The apparatus of embodiment 242, wherein the apportionment heuristics includes the user's entity graph.

[0633] 247. The apparatus of embodiment 246, wherein the user's entity graph includes at least one of a social graph and an interest graph.

[0634] 248. The apparatus of embodiment 242, wherein the apportionment heuristics includes user-specific usage.

[0635] 249. The apparatus of embodiment 242, wherein the apportionment heuristics includes aggregate usage.

[0636] 250. The apparatus of embodiment 242, wherein the apportionment heuristics includes preference profile.

[0637] 251. The apparatus of embodiment 250, wherein the preference profile is associated with at least one of a user or a group of users, and indicative of preference for at least one of: (i) genres, (ii) artists, (iii) albums, (iv) tracks, (v) music attributes, (vi) location based preferences, and (vii) time based preferences.

[0638] 252. The apparatus of embodiment 242, wherein the apportionment heuristics includes social recommendation.

[0639] 253. The apparatus of embodiment 242, wherein the content discovery supportive information is updated based on an activity associated with one or more users.

[0640] 254. The apparatus of embodiment 253, wherein the processor issues further instructions to:

[0641] provide the updated content discovery supportive information for the universally resolvable content user;

[0642] obtain an indication of determination of updated apportionment heuristics among the obtained updated information for the user;

[0643] obtain an indication of identification of a second set of universally resolvable content items based on the determined updated apportionment heuristics;

[0644] obtain an indication of update of the caching queue that includes the identified second set of universally resolvable content items; and

[0645] obtain the second set of universally resolvable content items in the updated caching queue to the user.

[0646] 255. The apparatus of embodiment 254, wherein the second set of universally resolvable content items includes at least one content item from the first set of universally resolvable content items.

[0647] 256. The apparatus of embodiment 242, wherein the content discovery supportive information includes at least one of: most frequently played content item, content item rated high, content item rated low, content item shared and content item bookmarked.

[0648] 257. A processor-implemented method for providing shared access to a media content collection, comprising:

[0649] obtaining from a first universally resolvable media content service user a request to share the user's universally resolvable media content collection;

[0650] obtaining from the first user a selection of at least one second universally resolvable user;

[0651] configuring the first user's media content collection with restricted shared access controls for the second user; and

[0652] providing the second user restricted shared access to the shared media content collection based on the configured restricted shared access controls.

[0653] 258. The method of embodiment 257, further comprising:

[0654] receiving from one of a plurality of shared users having access to the shared media content collection a request to perform an action on the shared media content collection;

[0655] performing said action on the shared media content collection to obtain a modified shared media content collection; and

[0656] providing the plurality of shared users access to the modified shared media content collection.

[0657] 259. The method of embodiment 257, further comprising:

[0658] downloading non-local content items of the shared media content collection to client devices of a plurality of shared users having access to the shared media content collection, wherein the plurality of shared users includes at least the first user and the second user.

[0659] 260. The method of embodiment 259, wherein the non-local content items are downloaded in the cache.

[0660] 261. The method of embodiment 257, wherein the shared media collection is stored in a universally resolvable content database in a server.

[0661] 262. The method of embodiment 258, wherein said action is selected from any of: (i) add a content item, (ii) remove a content item, (iii) re-order content items, (iv) modify privacy settings, (v) share with another universally resolvable user, (vi) delete the shared media collection, (vii) create a playlist, (viii) add an entity graph member, (ix) publish media collection, and (x) change meta data for a content item.

[0662] 263. The method of embodiment 257, wherein prior to receiving from the first user the request to share the first user's media content collection:

[0663] receiving from the first user a request to create the media content collection; and

[0664] receiving from the first user at least one content item for including in the created media content collection.

[0665] 264. The method of embodiment 257, wherein the media collection any one of: (i) a universally resolvable media content service user created media content collection, and (ii) a system generated media content collection.

[0666] 265. The method of embodiment 257, further comprising:

[0667] configuring access by other universally resolvable media content service users to the shared media collection based on the first user selected access constraints.

[0668] 266. The method of embodiment 265, wherein the access constraints define read only access to the shared media content collection to any one of: (i) friends having a predetermined degree of separation, (ii) Gurus, (iii) social network members having a predetermined degree of separation, (iv) relationship categories, and (v) everyone.

[0669] 267. The method of embodiment 257, further comprising:

[0670] receiving a request from the second user to perform an action on the shared media content collection; and

[0671] obtaining from the first user an acceptance or a denial to the received request to perform the action on the shared media content collection.

[0672] 268. The method of embodiment 267, wherein when said acceptance is obtained,

[0673] performing said action on the shared media content collection to obtain a modified shared media content collection; and

[0674] providing the modified shared media content collection to the first user and the second user; and

[0675] when said denial is obtained,

[0676] providing the second user a request declined message.

[0677] 269. The method of embodiment 257, wherein the media content collection includes: (i) a playlist, (ii) at least one content item, (iii) a media library in the cloud, and (iv) a media library in a local client.

[0678] 270. The method of embodiment 257, wherein said configuring includes obtaining the first user specified content parameters and access control parameters for the content.

[0679] 271. The method of embodiment 257, further comprising:

[0680] determining a subset of the first user's media content collection that is restricted by the first user's access control parameters for access by the second user.

[0681] 272. A system for providing shared access to a media content collection, comprising:

[0682] a memory;

[0683] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0684] obtain from a first universally resolvable media content service user a request to share the user's universally resolvable media content collection;

[0685] obtain from the first user a selection of at least one second universally resolvable user;

[0686] configure the first user's media content collection for shared access with the second user; and

[0687] provide the second user access to the shared media content collection.

[0688] 273. The system of embodiment 272, wherein the processor issues further instructions to:

[0689] receive from one of a plurality of shared users having access to the shared media content collection a request to perform an action on the shared media content collection;

[0690] perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0691] provide the plurality of shared users access to the modified shared media content collection.

[0692] 274. The system of embodiment 273, wherein said action is selected from any of: (i) add a content item, (ii) remove a content item, (iii) re-order content items, (iv) modify privacy settings, (v) share with another universally resolvable user, (vi) delete the shared media collection, (vii) create a playlist, (viii) add an entity graph member, (ix) publish media collection, and (x) change meta data for a content item.

[0693] 275. The system of embodiment 272, wherein the processor issues further instructions to:

[0694] download non-local content items of the shared media content collection to client devices of a plurality of shared users having access to the shared media content collection, wherein the plurality of shared users includes at least the first user and the second user.

[0695] 276. The system of embodiment 275, wherein the non-local content items are downloaded in the cache.

[0696] 277. The system of embodiment 272, wherein the shared media collection is stored in a universally resolvable content database in a server.

[0697] 278. The system of embodiment 272, wherein prior to issuing instructions to receive from the first user the request to share the first user's media content collection, the processor issues instruction to;

[0698] receive from the first user a request to create the media content collection; and

[0699] receive from the first user at least one content item for including in the created media content collection.

[0700] 279. The system of embodiment 272, wherein the media collection any one of: (i) a universally resolvable media content service user created media content collection, and (ii) a system generated media content collection.

[0701] 280. The system of embodiment 272, wherein the processor issues further instructions to:

[0702] configure access by other universally resolvable media content service users to the shared media collection based on the first user selected access constraints.

[0703] 281. The system of embodiment 280, wherein the access constraints define read only access to the shared media content collection to any one of: (i) friends having a predetermined degree of separation, (ii) Gurus, (iii) social network members having a predetermined degree of separation, (iv) relationship categories, and (v) everyone.

[0704] 282. The system of embodiment 272, wherein the processor issues further instructions to:

[0705] receive a request from the second user to perform an action on the shared media content collection; and

[0706] obtain from the first user an acceptance or a denial to the received request to perform the action on the shared media content collection.

[0707] 283. The system of embodiment 282, wherein when said acceptance is obtained, the processor issues instructions to:

[0708] perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0709] provide the modified shared media content collection to the first user and the second user; and

[0710] when said denial is obtained, the processor issues instructions to:

[0711] provide the second user a request declined message.

[0712] 284. The system of embodiment 272, wherein the media content collection includes: (i) a playlist, (ii) at least one content item, (iii) a media library in the cloud, and (iv) a media library in a local client.

[0713] 285. The system of embodiment 272, wherein said instructions to configure includes instructions to obtain the first user specified content parameters and access control parameters for the content.

[0714] 286. The system of embodiment 272, wherein said processor issues further instructions to:

[0715] determine a subset of the first user's media content collection that is restricted by the first user's access control parameters for access by the second user.

[0716] 287. A processor-readable medium storing processor-issuable instructions for providing shared access to a media content collection, wherein the processor issues instructions to:

[0717] obtain from a first universally resolvable media content service user a request to share the user's universally resolvable media content collection;

[0718] obtain from the first user a selection of at least one second universally resolvable user;

[0719] configure the first user's media content collection for shared access with the second user; and

[0720] provide the second user access to the shared media content collection.

[0721] 288. The medium of embodiment 287, wherein the processor issues further instructions to:

[0722] receive from one of a plurality of shared users having access to the shared media content collection a request to perform an action on the shared media content collection;

[0723] perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0724] provide the plurality of shared users access to the modified shared media content collection.

[0725] 289. The medium term of embodiment 288, wherein said action is selected from any of: (i) add a content item, (ii) remove a content item, (iii) re-order content items, (iv) modify privacy settings, (v) share with another universally

resolvable user, (vi) delete the shared media collection, (vii) create a playlist, (viii) add an entity graph member, (ix) publish media collection, and (x) change meta data for a content item.

[0726] 290. The medium of embodiment 287, wherein the processor issues further instructions to:

[0727] download non-local content items of the shared media content collection to client devices of a plurality of shared users having access to the shared media content collection, wherein the plurality of shared users includes at least the first user and the second user.

[0728] 291. The medium of embodiment 290, wherein the non-local content items are downloaded in the cache.

[0729] 292. The medium of embodiment 287, wherein the shared media collection is stored in a universally resolvable content database in a server.

[0730] 293. The medium of embodiment 287, wherein prior to issuing instructions to receive from the first user the request to share the first user's media content collection, the processor issues instruction to;

[0731] receive from the first user a request to create the media content collection; and

[0732] receive from the first user at least one content item for including in the created media content collection.

[0733] 294. The medium of embodiment 287, wherein the media collection any one of: (i) a universally resolvable media content service user created media content collection, and (ii) a system generated media content collection.

[0734] 295. The medium of embodiment 287, wherein the processor issues further instructions to:

[0735] configure access by other universally resolvable media content service users to the shared media collection based on the first user selected access constraints.

[0736] 296. The medium of embodiment 295, wherein the access constraints define read only access to the shared media content collection to any one of: (i) friends having a predetermined degree of separation, (ii) Gurus, (iii) social network members having a predetermined degree of separation, (iv) relationship categories, and (v) everyone.

[0737] 297. The medium of embodiment 287, wherein the processor issues further instructions to:

[0738] receive a request from the second user to perform an action on the shared media content collection; and

[0739] obtain from the first user an acceptance or a denial to the received request to perform the action on the shared media content collection.

[0740] 298. The medium of embodiment 297, wherein when said acceptance is obtained, the processor issues instructions to:

[0741] perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0742] provide the modified shared media content collection to the first user and the second user; and

[0743] when said denial is obtained, the processor issues instructions to:

[0744] provide the second user a request declined message.

[0745] 299. The medium of embodiment 287, wherein the media content collection includes: (i) a playlist, (ii) at least one content item, (iii) a media library in the cloud, and (iv) a media library in a local client.

[0746] 300. The medium of embodiment 287, wherein said instructions to configure includes instructions to obtain the first user specified content parameters and access control parameters for the content.

[0747] 301. The medium of embodiment 287, wherein said processor issues further instructions to:

[0748] determine a subset of the first user's media content collection that is restricted by the first user's access control parameters for access by the second user.

[0749] 302. An apparatus for providing shared access to a media content collection, comprising:

[0750] a memory;

[0751] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0752] obtain from a first universally resolvable media content service user a request to share the user's universally resolvable media content collection;

[0753] obtain from the first user a selection of at least one second universally resolvable user;

[0754] configure the first user's media content collection for shared access with the second user; and

[0755] provide the second user access to the shared media content collection.

[0756] 303. The apparatus of embodiment 302, wherein the processor issues further instructions to:

[0757] receive from one of a plurality of shared users having access to the shared media content collection a request to perform an action on the shared media content collection;

[0758] perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0759] provide the plurality of shared users access to the modified shared media content collection.

[0760] 304. The apparatus of embodiment 303, wherein said action is selected from any of: (i) add a content item, (ii) remove a content item, (iii) re-order content items, (iv) modify privacy settings, (v) share with another universally resolvable user, (vi) delete the shared media collection, (vii) create a playlist, (viii) add an entity graph member, (ix) publish media collection, and (x) change meta data for a content item.

[0761] 305. The apparatus of embodiment 302, wherein the processor issues further instructions to:

[0762] download non-local content items of the shared media content collection to client devices of a plurality of shared users having access to the shared media content collection, wherein the plurality of shared users includes at least the first user and the second user.

[0763] 306. The apparatus of embodiment 305, wherein the non-local content items are downloaded in the cache.

[0764] 307. The apparatus of embodiment 302, wherein the shared media collection is stored in a universally resolvable content database in a server.

[0765] 308. The apparatus of embodiment 302, wherein prior to issuing instructions to receive from the first user the request to share the first user's media content collection, the processor issues instruction to;

[0766] receive from the first user a request to create the media content collection; and

[0767] receive from the first user at least one content item for including in the created media content collection.

[0768] 309. The apparatus of embodiment 302, wherein the media collection any one of: (i) a universally resolvable media content service user created media content collection, and (ii) a system generated media content collection.

[0769] 310. The apparatus of embodiment 302, wherein the processor issues further instructions to:

[0770] configure access by other universally resolvable media content service users to the shared media collection based on the first user selected access constraints.

[0771] 311. The apparatus of embodiment 310, wherein the access constraints define read only access to the shared media content collection to any one of: (i) friends having a predetermined degree of separation, (ii) Gurus, (iii) social network members having a predetermined degree of separation, (iv) relationship categories, and (v) everyone.

[0772] 312. The apparatus of embodiment 302, wherein the processor issues further instructions to:

[0773] receive a request from the second user to perform an action on the shared media content collection; and

[0774] obtain from the first user an acceptance or a denial to the received request to perform the action on the shared media content collection.

[0775] 313. The apparatus of embodiment 282, wherein when said acceptance is obtained, the processor issues instructions to:

[0776] perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0777] provide the modified shared media content collection to the first user and the second user; and

[0778] when said denial is obtained, the processor issues instructions to:

[0779] provide the second user a request declined message.

[0780] 314. The apparatus of embodiment 302, wherein the media content collection includes: (i) a playlist, (ii) at least one content item, (iii) a media library in the cloud, and (iv) a media library in a local client.

[0781] 315. The apparatus of embodiment 302, wherein said instructions to configure includes instructions to obtain the first user specified content parameters and access control parameters for the content.

[0782] 316. The apparatus of embodiment 302, wherein said processor issues further instructions to:

[0783] determine a subset of the first user's media content collection that is restricted by the first user's access control parameters for access by the second user.

[0784] 317. A processor-implemented method, comprising:

[0785] providing from a first universally resolvable media content service user a request to share the user's universally resolvable media content collection;

[0786] providing from the first user a selection of at least one second universally resolvable user;

[0787] providing authorization to configuring the first user's media content collection with restricted shared access controls for the second user; and

[0788] providing authorization to provide the second user restricted shared access to the shared media content collection based on the configured restricted shared access controls.

[0789] 318. The method of embodiment 317, further comprising:

[0790] providing to one of a plurality of shared users having access to the shared media content collection a request to perform an action on the shared media content collection;

[0791] providing authorization to perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0792] obtaining for the plurality of shared users access to the modified shared media content collection.

[0793] 319. The method of embodiment 317, further comprising:

[0794] providing authorization to download non-local content items of the shared media content collection to client devices of a plurality of shared users having access to the shared media content collection, wherein the plurality of shared users includes at least the first user and the second user.

[0795] 320. The method of embodiment 319, wherein the non-local content items are downloaded in the cache.

[0796] 321. The method of embodiment 317, wherein the shared media collection is stored in a universally resolvable content database in a server.

[0797] 322. The method of embodiment 318, wherein said action is selected from any of: (i) add a content item, (ii) remove a content item, (iii) re-order content items, (iv) modify privacy settings, (v) share with another universally resolvable user, (vi) delete the shared media collection, (vii) create a playlist, (viii) add an entity graph member, (ix) publish media collection, and (x) change meta data for a content item.

[0798] 323. The method of embodiment 317, wherein prior to providing from the first user the request to share the first user's media content collection:

[0799] providing from the first user a request to create the media content collection; and

[0800] providing from the first user at least one content item for including in the created media content collection.

[0801] 324. The method of embodiment 317, wherein the media collection any one of: (i) a universally resolvable media content service user created media content collection, and (ii) a system generated media content collection.

[0802] 325. The method of embodiment 317, further comprising:

[0803] providing authorization to configure access by other universally resolvable media content service users to the shared media collection based on the first user selected access constraints.

[0804] 326. The method of embodiment 325, wherein the access constraints define read only access to the shared media content collection to any one of: (i) friends having a predetermined degree of separation, (ii) Gurus, (iii) social network members having a predetermined degree of separation, (iv) relationship categories, and (v) everyone.

[0805] 327. The method of embodiment 317, further comprising:

[0806] providing a request from the second user to perform an action on the shared media content collection; and

[0807] providing from the first user an acceptance or a denial to the received request to perform the action on the shared media content collection.

[0808] 328. The method of embodiment 327, wherein when said acceptance is obtained,

[0809] providing authorization to perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0810] obtaining the modified shared media content collection to the first user and the second user; and

[0811] when said denial is obtained,

[0812] obtaining the second user a request declined message.

[0813] 329. The method of embodiment 317, wherein the media content collection includes: (i) a playlist, (ii) at least one content item, (iii) a media library in the cloud, and (iv) a media library in a local client.

[0814] 330. The method of embodiment 317, wherein said configuring includes obtaining the first user specified content parameters and access control parameters for the content.

[0815] 331. The method of embodiment 317, further comprising:

[0816] obtaining an indication of determination of a subset of the first user's media content collection that is restricted by the first user's access control parameters for access by the second user.

[0817] 332. A system for providing shared access to a media content collection, comprising:

[0818] a memory;

[0819] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0820] provide from a first universally resolvable media content service user a request to share the user's universally resolvable media content collection;

[0821] provide from the first user a selection of at least one second universally resolvable user;

[0822] provide authorization to configure the first user's media content collection for shared access with the second user; and

[0823] provide authorization to the second user access to the shared media content collection.

[0824] 333. The system of embodiment 332, wherein the processor issues further instructions to:

[0825] provide from one of a plurality of shared users having access to the shared media content collection a request to perform an action on the shared media content collection;

[0826] provide authorization to perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0827] obtain the plurality of shared users access to the modified shared media content collection.

[0828] 334. The system of embodiment 333, wherein said action is selected from any of: (i) add a content item, (ii) remove a content item, (iii) re-order content items, (iv) modify privacy settings, (v) share with another universally resolvable user, (vi) delete the shared media collection, (vii) create a playlist, (viii) add an entity graph member, (ix) publish media collection, and (x) change meta data for a content item.

[0829] 335. The system of embodiment 332, wherein the processor issues further instructions to:

[0830] provide authorization to download non-local content items of the shared media content collection to client devices of a plurality of shared users having access to the shared media content collection, wherein the plurality of shared users includes at least the first user and the second user.

[0831] 336. The system of embodiment 335, wherein the non-local content items are downloaded in the cache.

[0832] 337. The system of embodiment 332, wherein the shared media collection is stored in a universally resolvable content database in a server.

[0833] 338. The system of embodiment 332, wherein prior to issuing instructions to receive from the first user the request to share the first user's media content collection, the processor issues instruction to:

[0834] receive from the first user a request to create the media content collection; and

[0835] receive from the first user at least one content item for including in the created media content collection.

[0836] 339. The system of embodiment 332, wherein the media collection any one of: (i) a universally resolvable media content service user created media content collection, and (ii) a system generated media content collection.

[0837] 340. The system of embodiment 332, wherein the processor issues further instructions to:

[0838] configure access by other universally resolvable media content service users to the shared media collection based on the first user selected access constraints.

[0839] 341. The system of embodiment 330, wherein the access constraints define read only access to the shared media content collection to any one of: (i) friends having a predetermined degree of separation, (ii) Gurus, (iii) social network members having a predetermined degree of separation, (iv) relationship categories, and (v) everyone.

[0840] 342. The system of embodiment 332, wherein the processor issues further instructions to:

[0841] receive a request from the second user to perform an action on the shared media content collection; and

[0842] obtain from the first user an acceptance or a denial to the received request to perform the action on the shared media content collection.

[0843] 343. The system of embodiment 342, wherein when said acceptance is obtained, the processor issues instructions to:

[0844] perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0845] provide the modified shared media content collection to the first user and the second user; and

[0846] when said denial is obtained, the processor issues instructions to:

[0847] provide the second user a request declined message.

[0848] 344. The system of embodiment 332, wherein the media content collection includes: (i) a playlist, (ii) at least one content item, (iii) a media library in the cloud, and (iv) a media library in a local client.

[0849] 345. The system of embodiment 332, wherein said instructions to configure includes instructions to obtain the first user specified content parameters and access control parameters for the content.

[0850] 346. The system of embodiment 332, wherein said processor issues further instructions to:

[0851] determine a subset of the first user's media content collection that is restricted by the first user's access control parameters for access by the second user.

[0852] 347. A processor-readable medium storing processor-issuable instructions for providing shared access to a media content collection, wherein the processor issues instructions to:

[0853] provide from a first universally resolvable media content service user a request to share the user's universally resolvable media content collection;

[0854] provide from the first user a selection of at least one second universally resolvable user;

[0855] provide authorization to configure the first user's media content collection for shared access with the second user; and

[0856] provide authorization to provide the second user access to the shared media content collection.

[0857] 348. The medium of embodiment 347, wherein the processor issues further instructions to:

[0858] provide from one of a plurality of shared users having access to the shared media content collection a request to perform an action on the shared media content collection;

[0859] provide authorization to perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0860] obtain the plurality of shared users access to the modified shared media content collection.

[0861] 349. The medium of embodiment 348, wherein said action is selected from any of: (i) add a content item, (ii) remove a content item, (iii) re-order content items, (iv) modify privacy settings, (v) share with another universally resolvable user, (vi) delete the shared media collection, (vii) create a playlist, (viii) add an entity graph member, (ix) publish media collection, and (x) change meta data for a content item.

[0862] 350. The medium of embodiment 347, wherein the processor issues further instructions to:

[0863] provide authorization to download non-local content items of the shared media content collection to client devices of a plurality of shared users having access to the shared media content collection, wherein the plurality of shared users includes at least the first user and the second user.

[0864] 351. The medium of embodiment 350, wherein the non-local content items are downloaded in the cache.

[0865] 352. The medium of embodiment 347, wherein the shared media collection is stored in a universally resolvable content database in a server.

[0866] 353. The medium of embodiment 347, wherein prior to issuing instructions to receive from the first user the request to share the first user's media content collection, the processor issues instruction to;

[0867] provide from the first user a request to create the media content collection; and

[0868] provide from the first user at least one content item for including in the created media content collection.

[0869] 354. The medium of embodiment 347, wherein the media collection any one of: (i) a universally resolvable media content service user created media content collection, and (ii) a system generated media content collection.

[0870] 355. The medium of embodiment 347, wherein the processor issues further instructions to:

[0871] provide authorization to configure access by other universally resolvable media content service users to the shared media collection based on the first user selected access constraints.

[0872] 356. The medium of embodiment 355, wherein the access constraints define read only access to the shared media content collection to any one of: (i) friends having a predetermined degree of separation, (ii) Gurus, (iii) social network members having a predetermined degree of separation, (iv) relationship categories, and (v) everyone.

[0873] 357. The medium of embodiment 347, wherein the processor issues further instructions to:

[0874] provide a request from the second user to perform an action on the shared media content collection; and

[0875] provide from the first user an acceptance or a denial to the received request to perform the action on the shared media content collection.

[0876] 358. The medium of embodiment 357, wherein when said acceptance is obtained, the processor issues instructions to:

[0877] provide authorization to perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0878] obtain the modified shared media content collection to the first user and the second user; and

[0879] when said denial is obtained, the processor issues instructions to:

[0880] provide authorization to provide the second user a request declined message.

[0881] 359. The medium of embodiment 347, wherein the media content collection includes: (i) a playlist, (ii) at least one content item, (iii) a media library in the cloud, and (iv) a media library in a local client.

[0882] 360. The medium of embodiment 347, wherein said instructions to configure includes instructions to obtain the first user specified content parameters and access control parameters for the content.

[0883] 361. The medium of embodiment 347, wherein said processor issues further instructions to:

[0884] provide authorization to determine a subset of the first user's media content collection that is restricted by the first user's access control parameters for access by the second user.

[0885] 362. An apparatus for providing shared access to a media content collection, comprising:

[0886] a memory;

[0887] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0888] provide from a first universally resolvable media content service user a request to share the user's universally resolvable media content collection;

[0889] provide from the first user a selection of at least one second universally resolvable user;

[0890] provide authorization to configure the first user's media content collection for shared access with the second user; and

[0891] provide authorization to provide the second user access to the shared media content collection.

[0892] 363. The apparatus of embodiment 362, wherein the processor issues further instructions to:

[0893] obtain from one of a plurality of shared users having access to the shared media content collection a request to perform an action on the shared media content collection;

[0894] provide authorization to perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0895] obtain the plurality of shared users access to the modified shared media content collection.

[0896] 364. The apparatus of embodiment 363, wherein said action is selected from any of: (i) add a content item, (ii) remove a content item, (iii) re-order content items, (iv) modify privacy settings, (v) share with another universally resolvable user, (vi) delete the shared media collection, (vii) create a playlist, (viii) add an entity graph member, (ix) publish media collection, and (x) change meta data for a content item.

[0897] 365. The apparatus of embodiment 362, wherein the processor issues further instructions to:

[0898] provide authorization to download non-local content items of the shared media content collection to client devices of a plurality of shared users having access to the shared media content collection, wherein the plurality of shared users includes at least the first user and the second user.

[0899] 366. The apparatus of embodiment 365, wherein the non-local content items are downloaded in the cache.

[0900] 367. The apparatus of embodiment 362, wherein the shared media collection is stored in a universally resolvable content database in a server.

[0901] 368. The apparatus of embodiment 362, wherein prior to issuing instructions to receive from the first user the request to share the first user's media content collection, the processor issues instruction to:

[0902] provide from the first user a request to create the media content collection; and

[0903] provide from the first user at least one content item for including in the created media content collection.

[0904] 369. The apparatus of embodiment 362, wherein the media collection any one of: (i) a universally resolvable media content service user created media content collection, and (ii) a system generated media content collection.

[0905] 370. The apparatus of embodiment 362, wherein the processor issues further instructions to:

[0906] provide authorization to configure access by other universally resolvable media content service users to the shared media collection based on the first user selected access constraints.

[0907] 371. The apparatus of embodiment 370, wherein the access constraints define read only access to the shared media content collection to any one of: (i) friends having a predetermined degree of separation, (ii) Gurus, (iii) social network members having a predetermined degree of separation, (iv) relationship categories, and (v) everyone.

[0908] 372. The apparatus of embodiment 362, wherein the processor issues further instructions to:

[0909] provide a request from the second user to perform an action on the shared media content collection; and

[0910] provide from the first user an acceptance or a denial to the received request to perform the action on the shared media content collection.

[0911] 373. The apparatus of embodiment 372, wherein when said acceptance is obtained, the processor issues instructions to:

[0912] provide authorization to perform said action on the shared media content collection to obtain a modified shared media content collection; and

[0913] obtain the modified shared media content collection to the first user and the second user; and

[0914] when said denial is obtained, the processor issues instructions to:

[0915] obtain the second user a request declined message.

[0916] 374. The apparatus of embodiment 362, wherein the media content collection includes: (i) a playlist, (ii) at least one content item, (iii) a media library in the cloud, and (iv) a media library in a local client.

[0917] 375. The apparatus of embodiment 362, wherein said instructions to configure includes instructions to obtain the first user specified content parameters and access control parameters for the content.

[0918] 376. The apparatus of embodiment 362, wherein said processor issues further instructions to:

[0919] provide authorization to determine a subset of the first user's media content collection that is restricted by the first user's access control parameters for access by the second user.

[0920] 377. A processor-implemented method for providing access to a portion of a media library, comprising:

[0921] receiving from a first universally resolvable user a request to access a media library of a second universally resolvable user;

[0922] retrieving the second user specified privacy controls;

[0923] applying the second user specified privacy controls to determine a portion of the media library permitted for shared access by the first user; and

[0924] allowing the first user access to the determined portion of the media library.

[0925] 378. The method of embodiment 377, further comprising:

[0926] determining a degree of separation between the first user and the second user; and

[0927] applying the degree of separation as an access constraint to the media library of the second user.

[0928] 379. The method of embodiment 377, further comprising:

[0929] receiving a request from the first user to download at least one content item located in the determined portion of the media library; and

[0930] in response to the received request to download the at least one content item, providing the at least one content item in the first user's media library.

[0931] 380. The method of embodiment 377, wherein access by the first user includes modification of the second user's media library.

[0932] 381. The method of embodiment 380, wherein the modification of the second user's media library includes at least one of: (i) adding a local content item, (ii) removing a content item, (iii) adding a social graph member, and (iv) creating a playlist.

[0933] 382. The method of embodiment 381, wherein the content item includes: (i) a playlist, (ii) entity graph, (iii) music, (iv) movie, (v) book, and (vi) video.

[0934] 383. The method of embodiment 380, wherein the playlist is a shared playlist modifiable by the first user and the second user.

[0935] 384. The method of embodiment 377, wherein the portion of the media library permitted for access by the first user includes at least one of: (i) albums, (ii) genre, (iii) artists, (iv) purchased, (v) playlist, and (vii) all content.

[0936] 385. A system for providing access to a portion of a media library, comprising:

[0937] a memory;

[0938] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0939] receive from a first universally resolvable user a request to access a media library of a second universally resolvable user;

[0940] retrieve the second user specified privacy controls;

[0941] apply the second user specified privacy controls to determine a portion of the media library permitted for access by the first user; and

[0942] allow the first user access to the determined portion of the media library.

[0943] 386. The system of embodiment 385, wherein the processor issues further instructions to:

[0944] determine a degree of separation between the first user and the second user; and

[0945] apply the degree of separation as an access constraint to the media library of the second user.

[0946] 387. The system of embodiment 385, wherein the processor issues further instructions to:

[0947] receive a request from the first user to download at least one content item located in the determined portion of the media library; and

[0948] in response to the received request to download the at least one content item, provide the at least one content item in the first user's media library.

[0949] 388. The system of embodiment 385, wherein access by the first user includes modification of the second user's media library.

[0950] 389. The system of embodiment 388, wherein the modification of the second user's media library includes at least one of: (i) adding a local content item, (ii) removing a content item, (iii) adding a social graph member, and (iv) creating a playlist.

[0951] 390. The system of embodiment 389, wherein the content item includes: (i) a playlist, (ii) entity graph, (iii) music, (iv) movie, (v) book, and (vi) video.

[0952] 391. The system of embodiment 390, wherein the playlist is a shared playlist modifiable by the first user and the second user.

[0953] 392. The system of embodiment 385, wherein the portion of the media library permitted for access by the first user includes at least one of: (i) albums, (ii) genre, (iii) artists, (iv) purchased, (v) playlist, and (vii) all content.

[0954] 393. A processor-readable medium storing processor-issuable instructions for providing access to a portion of a media library wherein the processor issues instructions to:

[0955] receive from a first universally resolvable user a request to access a media library of a second universally resolvable user;

[0956] retrieve the second user specified privacy controls;

[0957] apply the second user specified privacy controls to determine a portion of the media library permitted for access by the first user; and

[0958] allow the first user access to the determined portion of the media library.

[0959] 394. The medium of embodiment 393, wherein the processor issues further instructions to:

[0960] determine a degree of separation between the first user and the second user; and

[0961] apply the degree of separation as an access constraint to the media library of the second user.

[0962] 395. The medium of embodiment 393, wherein the processor issues further instructions to:

[0963] receive a request from the first user to download at least one content item located in the determined portion of the media library; and

[0964] in response to the received request to download the at least one content item, provide the at least one content item in the first user's media library.

[0965] 396. The medium of embodiment 393, wherein access by the first user includes modification of the second user's media library.

[0966] 397. The medium of embodiment 396, wherein the modification of the second user's media library includes at least one of: (i) adding a local content item, (ii) removing a content item, (iii) adding a social graph member, and (iv) creating a playlist.

[0967] 398. The medium of embodiment 397, wherein the content item includes: (i) a playlist, (ii) entity graph, (iii) music, (iv) movie, (v) book, and (vi) video.

[0968] 399. The medium of embodiment 398, wherein the playlist is a shared playlist modifiable by the first user and the second user.

[0969] 400. The medium of embodiment 393, wherein the portion of the media library permitted for access by the first user includes at least one of: (i) albums, (ii) genre, (iii) artists, (iv) purchased, (v) playlist, and (vii) all content.

[0970] 401. An apparatus for providing access to a portion of a media library wherein the processor issues instructions to:

[0971] a memory;

[0972] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[0973] receive from a first universally resolvable user a request to access a media library of a second universally resolvable user;

[0974] retrieve the second user specified privacy controls;

[0975] apply the second user specified privacy controls to determine a portion of the media library permitted for access by the first user; and

[0976] allow the first user access to the determined portion of the media library.

[0977] 402. The apparatus of embodiment 401, wherein the processor issues further instructions to:

[0978] determine a degree of separation between the first user and the second user; and

[0979] apply the degree of separation as an access constraint to the media library of the second user

[0980] 403. The apparatus of embodiment 401, wherein the processor issues further instructions to:

[0981] receive a request from the first user to download at least one content item located in the determined portion of the media library; and

[0982] in response to the received request to download the at least one content item, provide the at least one content item in the first user's media library.

[0983] 404. The apparatus of embodiment 401, wherein access by the first user includes modification of the second user's media library.

[0984] 405. The apparatus of embodiment 404, wherein the modification of the second user's media library includes at least one of: (i) adding a local content item, (ii) removing a content item, (iii) adding a social graph member, and (iv) creating a playlist.

[0985] 406. The apparatus of embodiment 405, wherein the content item includes: (i) a playlist, (ii) entity graph, (iii) music, (iv) movie, (v) book, and (vi) video.

[0986] 407. The apparatus of embodiment 404, wherein the playlist is a shared playlist modifiable by the first user and the second user.

[0987] 408. The apparatus of embodiment 401, wherein the portion of the media library permitted for access by the first user includes at least one of: (i) albums, (ii) genre, (iii) artists, (iv) purchased, (v) playlist, and (vii) all content.

[0988] 409. A processor-implemented method for sharing a universally resolvable media content collection, comprising:

[0989] obtaining privacy settings for sharing a universally resolvable media content collection, said privacy settings identifying:

[0990] a selection of a universally resolvable media content collection; and

[0991] a selection of at least one universally resolvable user authorized to access the universally resolvable media content collection;

[0992] generating and transmitting a sharing request including the privacy settings;

[0993] receiving a confirmation that the universally resolvable media content collection is configured for sharing with the at least one universally resolvable user.

[0994] 410. The method of embodiment 409, wherein the at least one universally resolvable user includes at least one of: (i) a friend having a specified degree of separation; (ii) a social network member having a specified degree of separation; (iii) a friend; (iv) a family; (v) an acquaintance; and (vi) a work friend.

[0995] 411. The method of embodiment 409, wherein the at least one universally resolvable content collection includes at least one of: (i) all content items in the collection; (ii) most recently played; (iii) playlists; (iv) selected; (v) highest rated; (vi) purchased; and (vii) most frequently played.

[0996] 412. The method of embodiment 409, further comprising:

[0997] receiving a request from a universally resolvable user to access a universally resolvable content collection.

[0998] 413. The method of embodiment 412, further comprising:

[0999] in response to the request, allowing the universally resolvable user access the universally resolvable content collection.

[1000] 414. The method of embodiment 412, further comprising:

[1001] in response to the request, denying the universally resolvable user access the at least one universally resolvable content collection.

[1002] 415. A system for sharing a universally resolvable media content collection, comprising:

[1003] a memory;

[1004] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1005] obtain privacy settings for sharing a universally resolvable media content collection, said privacy settings identifying:

[1006] a select of a universally resolvable media content collection; and

[1007] a select of at least one universally resolvable user authorized to access the universally resolvable media content collection;

[1008] generate and transmit a sharing request including the privacy settings;

[1009] receive a confirmation that the universally resolvable media content collection is configured for sharing with the at least one universally resolvable user.

[1010] 416. The system of embodiment 415, wherein the at least one universally resolvable user includes at least one of: (i) a friend having a specified degree of separation; (ii) a social network member having a specified degree of separation; (iii) a friend; (iv) a family; (v) an acquaintance; and (vi) a work friend.

[1011] 417. The system of embodiment 415, wherein the at least one universally resolvable content collection includes at least one of: (i) all content items in the collection; (ii) most recently played; (iii) playlists; (iv) selected; (v) highest rated; (vi) purchased; and (vii) most frequently played.

[1012] 418. The system of embodiment 415, wherein the processor issues further instructions to:

[1013] receive a request from a universally resolvable user to access a universally resolvable content collection.

[1014] 419. The system of embodiment 418, wherein in response to the request, the processor issues further instructions to allow the universally resolvable user access the universally resolvable content collection.

[1015] 420. The system of embodiment 418, wherein in response to the request, the processor issues further instructions to deny the universally resolvable user access the at least one universally resolvable content collection.

[1016] 421. A processor-readable medium storing processor-issuable instructions for providing access to a portion of a media library wherein the processor issues instructions to:

[1017] obtain privacy settings for sharing a universally resolvable media content collection, said privacy settings identifying:

[1018] a select of a universally resolvable media content collection; and

[1019] a select of at least one universally resolvable user authorized to access the universally resolvable media content collection;

[1020] generate and transmit a sharing request including the privacy settings;

[1021] receive a confirmation that the universally resolvable media content collection is configured for sharing with the at least one universally resolvable user.

[1022] 422. The medium of embodiment 421, wherein the at least one universally resolvable user includes at least one of: (i) a friend having a specified degree of separation; (ii) a social network member having a specified degree of separation; (iii) a friend; (iv) a family; (v) an acquaintance; and (vi) a work friend.

[1023] 423. The medium of embodiment 421, wherein the at least one universally resolvable content collection includes at least one of: (i) all content items in the collection; (ii) most recently played; (iii) playlists; (iv) selected; (v) highest rated; (vi) purchased; and (vii) most frequently played.

[1024] 424. The medium of embodiment 421, wherein the processor issues further instructions to:

[1025] receive a request from a universally resolvable user to access a universally resolvable content collection.

[1026] 425. The medium of embodiment 424, wherein in response to the request, the processor issues further instructions to allow the universally resolvable user access the universally resolvable content collection.

[1027] 426. The medium of embodiment 424, wherein in response to the request, the processor issues further instructions to deny the universally resolvable user access the at least one universally resolvable content collection.

[1028] 427. An apparatus for sharing a universally resolvable media content collection, comprising:

[1029] a memory;

[1030] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1031] obtain privacy settings for sharing a universally resolvable media content collection, said privacy settings identifying:

[1032] a select of a universally resolvable media content collection; and

[1033] a select of at least one universally resolvable user authorized to access the universally resolvable media content collection;

[1034] generate and transmit a sharing request including the privacy settings;

[1035] receive a confirmation that the universally resolvable media content collection is configured for sharing with the at least one universally resolvable user.

[1036] 428. The apparatus of embodiment 427, wherein the at least one universally resolvable user includes at least one of: (i) a friend having a specified degree of separation; (ii) a social network member having a specified degree of separation; (iii) a friend; (iv) a family; (v) an acquaintance; and (vi) a work friend.

[1037] 429. The apparatus of embodiment 427, wherein the at least one universally resolvable content collection includes at least one of: (i) all content items in the collection; (ii) most recently played; (iii) playlists; (iv) selected; (v) highest rated; (vi) purchased; and (vii) most frequently played.

[1038] 430. The apparatus of embodiment 427, wherein the processor issues further instructions to:

[1039] receive a request from a universally resolvable user to access a universally resolvable content collection.

[1040] 431. The apparatus of embodiment 430, wherein in response to the request, the processor issues further instructions to allow the universally resolvable user access the universally resolvable content collection.

[1041] 432. The apparatus of embodiment 430, wherein in response to the request, the processor issues further instructions to deny the universally resolvable user access the at least one universally resolvable content collection.

[1042] 433. A processor-implemented method, comprising:

[1043] generating from a first universally resolvable user a request to access a media library of a second universally resolvable user;

[1044] retrieving the second user specified privacy controls;

[1045] supplying the second user specified privacy controls to determine a portion of the media library permitted for shared access by the first user; and

[1046] providing authorization to allow the first user access to the determined portion of the media library.

[1047] 434. The method of embodiment 433, further comprising:

[1048] obtaining indication of determination of a degree of separation between the first user and the second user; and

[1049] providing authorization to apply the degree of separation as an access constraint to the media library of the second user.

[1050] 435. The method of embodiment 433, further comprising:

[1051] providing a request from the first user to download at least one content item located in the determined portion of the media library; and

[1052] in response to the provided request to download the at least one content item, obtaining the at least one content item in the first user's media library.

[1053] 436. The method of embodiment 433, wherein access by the first user includes modification of the second user's media library.

[1054] 437. The method of embodiment 436, wherein the modification of the second user's media library includes at least one of: (i) adding a local content item, (ii) removing a content item, (iii) adding a social graph member, and (iv) creating a playlist.

[1055] 438. The method of embodiment 437, wherein the content item includes: (i) a playlist, (ii) entity graph, (iii) music, (iv) movie, (v) book, and (vi) video.

[1056] 439. The method of embodiment 436, wherein the playlist is a shared playlist modifiable by the first user and the second user.

[1057] 440. The method of embodiment 433, wherein the portion of the media library permitted for access by the first user includes at least one of: (i) albums, (ii) genre, (iii) artists, (iv) purchased, (v) playlist, and (vii) all content.

[1058] 441. A system for providing access to a portion of a media library, comprising:

[1059] a memory;

[1060] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1061] generate from a first universally resolvable user a request to access a media library of a second universally resolvable user;

[1062] retrieve the second user specified privacy controls;

[1063] supply the second user specified privacy controls to determine a portion of the media library permitted for access by the first user; and

[1064] provide authorization to the first user access to the determined portion of the media library.

[1065] 442. The system of embodiment 441, wherein the processor issues further instructions to:

[1066] obtain indication of determination of a degree of separation between the first user and the second user; and

[1067] supply the degree of separation as an access constraint to the media library of the second user.

[1068] 443. The system of embodiment 441, wherein the processor issues further instructions to:

[1069] provide a request from the first user to download at least one content item located in the determined portion of the media library; and

[1070] in response to the request to download the at least one content item, obtain the at least one content item in the first user's media library.

[1071] 444. The system of embodiment 441, wherein access by the first user includes modification of the second user's media library.

[1072] 445. The system of embodiment 444, wherein the modification of the second user's media library includes at least one of: (i) adding a local content item, (ii) removing a content item, (iii) adding a social graph member, and (iv) creating a playlist.

[1073] 446. The system of embodiment 445, wherein the content item includes: (i) a playlist, (ii) entity graph, (iii) music, (iv) movie, (v) book, and (vi) video.

[1074] 447. The system of embodiment 446, wherein the playlist is a shared playlist modifiable by the first user and the second user.

[1075] 448. The system of embodiment 441, wherein the portion of the media library permitted for access by the first user includes at least one of: (i) albums, (ii) genre, (iii) artists, (iv) purchased, (v) playlist, and (vii) all content.

[1076] 449. A processor-readable medium storing processor-issuable instructions for providing access to a portion of a media library wherein the processor issues instructions to:

[1077] generate from a first universally resolvable user a request to access a media library of a second universally resolvable user;

[1078] retrieve the second user specified privacy controls;

[1079] supply the second user specified privacy controls to determine a portion of the media library permitted for access by the first user; and

[1080] providing authorization to the first user access to the determined portion of the media library.

[1081] 450. The medium of embodiment 449, wherein the processor issues further instructions to:

[1082] obtain indication of determination of a degree of separation between the first user and the second user; and

[1083] apply the degree of separation as an access constraint to the media library of the second user.

[1084] 451. The medium of embodiment 449, wherein the processor issues further instructions to:

[1085] provide a request from the first user to download at least one content item located in the determined portion of the media library; and

[1086] in response to the request to download the at least one content item, provide the at least one content item in the first user's media library.

[1087] 452. The medium of embodiment 449, wherein access by the first user includes modification of the second user's media library.

[1088] 453. The medium of embodiment 452, wherein the modification of the second user's media library includes at least one of: (i) adding a local content item, (ii) removing a content item, (iii) adding a social graph member, and (iv) creating a playlist.

[1089] 454. The medium of embodiment 453, wherein the content item includes: (i) a playlist, (ii) entity graph, (iii) music, (iv) movie, (v) book, and (vi) video.

[1090] 455. The medium of embodiment 454, wherein the playlist is a shared playlist modifiable by the first user and the second user.

[1091] 456. The medium of embodiment 449, wherein the portion of the media library permitted for access by the first user includes at least one of: (i) albums, (ii) genre, (iii) artists, (iv) purchased, (v) playlist, and (vii) all content.

[1092] 457. An apparatus for providing access to a portion of a media library wherein the processor issues instructions to:

[1093] a memory;

[1094] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1095] provide from a first universally resolvable user a request to access a media library of a second universally resolvable user;

[1096] retrieve the second user specified privacy controls;

[1097] supply the second user specified privacy controls to determine a portion of the media library permitted for access by the first user; and

[1098] provide authorization to the first user access to the determined portion of the media library.

[1099] 458. The apparatus of embodiment 457, wherein the processor issues further instructions to:

[1100] obtain an indication of determination of a degree of separation between the first user and the second user; and

[1101] supply the degree of separation as an access constraint to the media library of the second user.

[1102] 459. The apparatus of embodiment 457, wherein the processor issues further instructions to:

[1103] provide a request from the first user to download at least one content item located in the determined portion of the media library; and

[1104] in response to the request to download the at least one content item, obtain the at least one content item in the first user's media library.

[1105] 460. The apparatus of embodiment 457, wherein access by the first user includes modification of the second user's media library.

[1106] 461. The apparatus of embodiment 460, wherein the modification of the second user's media library includes at

least one of: (i) adding a local content item, (ii) removing a content item, (iii) adding a social graph member, and (iv) creating a playlist.

[1107] 462. The apparatus of embodiment 461, wherein the content item includes: (i) a playlist, (ii) entity graph, (iii) music, (iv) movie, (v) book, and (vi) video.

[1108] 463. The apparatus of embodiment 460, wherein the playlist is a shared playlist modifiable by the first user and the second user.

[1109] 464. The apparatus of embodiment 457, wherein the portion of the media library permitted for access by the first user includes at least one of: (i) albums, (ii) genre, (iii) artists, (iv) purchased, (v) playlist, and (vii) all content.

[1110] 465. A processor-implemented method for sharing a universally resolvable media content collection, comprising:

[1111] providing privacy settings for sharing a universally resolvable media content collection, said privacy settings identifying:

[1112] a selection of a universally resolvable media content collection; and

[1113] a selection of at least one universally resolvable user authorized to access the universally resolvable media content collection;

[1114] providing authorization to generate and transmit a sharing request including the privacy settings;

[1115] providing a confirmation that the universally resolvable media content collection is configured for sharing with the at least one universally resolvable user.

[1116] 466. The method of embodiment 465, wherein the at least one universally resolvable user includes at least one of: (i) a friend having a specified degree of separation; (ii) a social network member having a specified degree of separation; (iii) a friend; (iv) a family; (v) an acquaintance; and (vi) a work friend.

[1117] 467. The method of embodiment 465, wherein the at least one universally resolvable content collection includes at least one of: (i) all content items in the collection; (ii) most recently played; (iii) playlists; (iv) selected; (v) highest rated; (vi) purchased; and (vii) most frequently played.

[1118] 468. The method of embodiment 465, further comprising:

[1119] providing a request from a universally resolvable user to access a universally resolvable content collection.

[1120] 469. The method of embodiment 468, further comprising:

[1121] in response to the request, providing authorization to allow the universally resolvable user access the universally resolvable content collection.

[1122] 470. The method of embodiment 468, further comprising:

[1123] in response to the request, providing authorization to deny the universally resolvable user access the at least one universally resolvable content collection.

[1124] 471. A system for sharing a universally resolvable media content collection, comprising:

[1125] a memory;

[1126] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1127] provide privacy settings for sharing a universally resolvable media content collection, said privacy settings identifying:

[1128] a selection of a universally resolvable media content collection; and

[1129] a selection of at least one universally resolvable user authorized to access the universally resolvable media content collection;

[1130] providing authorization to generate and transmit a sharing request including the privacy settings;

[1131] provide a confirmation that the universally resolvable media content collection is configured for sharing with the at least one universally resolvable user.

[1132] 472. The system of embodiment 471, wherein the at least one universally resolvable user includes at least one of: (i) a friend having a specified degree of separation; (ii) a social network member having a specified degree of separation; (iii) a friend; (iv) a family; (v) an acquaintance; and (vi) a work friend.

[1133] 473. The system of embodiment 471, wherein the at least one universally resolvable content collection includes at least one of: (i) all content items in the collection; (ii) most recently played; (iii) playlists; (iv) selected; (v) highest rated; (vi) purchased; and (vii) most frequently played.

[1134] 474. The system of embodiment 471, wherein the processor issues further instructions to:

[1135] provide a request from a universally resolvable user to access a universally resolvable content collection.

[1136] 475. The system of embodiment 474, wherein in response to the request, the processor issues further instructions to provide authorization to allow the universally resolvable user access the universally resolvable content collection.

[1137] 476. The system of embodiment 474, wherein in response to the request, the processor issues further instructions to provide authorization to deny the universally resolvable user access the at least one universally resolvable content collection.

[1138] 477. A processor-readable medium storing processor-issuable instructions for providing access to a portion of a media library wherein the processor issues instructions to:

[1139] provide privacy settings for sharing a universally resolvable media content collection, said privacy settings identifying:

[1140] a selection of a universally resolvable media content collection; and

[1141] a selection of at least one universally resolvable user authorized to access the universally resolvable media content collection;

[1142] provide authorization to generate and transmit a sharing request including the privacy settings;

[1143] provide a confirmation that the universally resolvable media content collection is configured for sharing with the at least one universally resolvable user.

[1144] 478. The medium of embodiment 477, wherein the at least one universally resolvable user includes at least one of: (i) a friend having a specified degree of separation; (ii) a social network member having a specified degree of separation; (iii) a friend; (iv) a family; (v) an acquaintance; and (vi) a work friend.

[1145] 479. The medium of embodiment 477, wherein the at least one universally resolvable content collection includes at least one of: (i) all content items in the collection; (ii) most recently played; (iii) playlists; (iv) selected; (v) highest rated; (vi) purchased; and (vii) most frequently played.

[1146] 480. The medium of embodiment 477, wherein the processor issues further instructions to:

[1147] receive a request from a universally resolvable user to access a universally resolvable content collection.

[1148] 481. The medium of embodiment 480, wherein in response to the request, the processor issues further instructions to allow the universally resolvable user access the universally resolvable content collection.

[1149] 482. The medium of embodiment 480, wherein in response to the request, the processor issues further instructions to deny the universally resolvable user access the at least one universally resolvable content collection.

[1150] 483. An apparatus for sharing a universally resolvable media content collection, comprising:

[1151] a memory;

[1152] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1153] provide privacy settings for sharing a universally resolvable media content collection, said privacy settings identifying:

[1154] a selection of a universally resolvable media content collection; and

[1155] a selection of at least one universally resolvable user authorized to access the universally resolvable media content collection;

[1156] provide authorization to generate and transmit a sharing request including the privacy settings;

[1157] provide a confirmation that the universally resolvable media content collection is configured for sharing with the at least one universally resolvable user.

[1158] 484. The apparatus of embodiment 483, wherein the at least one universally resolvable user includes at least one of: (i) a friend having a specified degree of separation; (ii) a social network member having a specified degree of separation; (iii) a friend; (iv) a family; (v) an acquaintance; and (vi) a work friend.

[1159] 485. The apparatus of embodiment 483, wherein the at least one universally resolvable content collection includes at least one of: (i) all content items in the collection; (ii) most recently played; (iii) playlists; (iv) selected; (v) highest rated; (vi) purchased; and (vii) most frequently played.

[1160] 486. The apparatus of embodiment 483, wherein the processor issues further instructions to:

[1161] provide a request from a universally resolvable user to access a universally resolvable content collection.

[1162] 487. The apparatus of embodiment 486, wherein in response to the request, the processor issues further instructions to provide authorization to allow the universally resolvable user access the universally resolvable content collection.

[1163] 488. The apparatus of embodiment 486, wherein in response to the request, the processor issues further instructions to provide authorization to deny the universally resolvable user access the at least one universally resolvable content collection.

[1164] 489. A graphical user interface, comprising:

[1165] a media display view, said media display view including a plurality of universally resolvable attribute rows, wherein each attribute row is configured to display universally resolvable attribute row items including social graph, wherein the media display view is configured to recursively:

[1166] obtain a selected content attribute row item;

[1167] display the selected content attribute row item in a corresponding attribute row of a first selection column;

[1168] identify related content attribute row items; and

[1169] provide the identified related content attribute row items for selection; and

[1170] display the selected related content attribute row item in a corresponding attribute row of a subsequent selection column.

[1171] 490. The graphical user interface of embodiment 489, wherein the selection column establishes the sequence of selection of the content attribute row items.

[1172] 491. The graphical user interface of embodiment 489, further comprising a media selection view displaying a designated spotlight area that is configured to receive the selected content attribute row item.

[1173] 492. The graphical user interface of embodiment 491, wherein the selection of the content attribute row item is made by a user via one of (i) a single action, or (ii) a double action.

[1174] 493. The graphical user interface of embodiment 492, wherein rows and columns are orthogonal and not spatially perpendicular.

[1175] 494. The graphical user interface of embodiment 489, wherein the media display view is one of a plurality of media display views, wherein each media display view corresponds to a time period and is configured to:

[1176] obtain data corresponding to a plurality of content attribute row items spotlighted during the time period; and

[1177] display the obtained data in corresponding content attribute rows and selection columns.

[1178] 495. The graphical user interface of embodiment 489, wherein the content attribute row items are selected from a group including: (i) artists, (ii) tracks, (iii) albums, (iv) playlists, (v) gurus, and (vi) social graph.

[1179] 496. A processor-implemented method for displaying content items, comprising:

[1180] obtaining a user selection of a content attribute row item;

[1181] displaying the selected content attribute row item in a corresponding attribute row of a first selection column of a media display view;

[1182] identifying related content attribute row items; and

[1183] providing the identified related content attribute row items for selection; and

[1184] displaying the selected related content attribute row item in a corresponding attribute row of a subsequent selection column of the media display view.

[1185] 497. The method of embodiment 496, wherein the selection column establishes the sequence of selection of the content attribute row items.

[1186] 498. The method of embodiment 496, further comprising a media selection view displaying a designated spotlight area that is configured to receive the selected content attribute row item.

[1187] 499. The method of embodiment 497, wherein the selection of the content attribute row item is made by a user via one of (i) a single action, or (ii) a double action.

[1188] 500. The method of embodiment 489, wherein rows and columns are orthogonal and not spatially perpendicular.

[1189] 501. The method of embodiment 496, wherein the media display view corresponds to a time period, the method further comprising:

[1190] obtaining data corresponding to a plurality of content attribute row items spotlighted during the time period; and

[1191] displaying the obtained data in corresponding content attribute rows and selection columns.

[1192] 502. The method of embodiment 496, wherein the content attribute row items are selected from a group including: (i) artists, (ii) tracks, (iii) albums, (iv) playlists, (v) gurus, and (vi) social graph.

[1193] 503. A system for displaying content items, comprising:

[1194] a memory;

[1195] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1196] obtain a user selection of a content attribute row item;

[1197] display the selected content attribute row item in a corresponding attribute row of a first selection column of a media display view;

[1198] identify related content attribute row items; and

[1199] provide the identified related content attribute row items for selection; and

[1200] display the selected related content attribute row item in a corresponding attribute row of a subsequent selection column of the media display view.

[1201] 504. The system of embodiment 503, wherein the selection column establishes the sequence of selection of the content attribute row items.

[1202] 505. The system of embodiment 503, further comprising a media selection view wherein the processor issues instructions to display a designated spotlight area that is configured to receive the selected content attribute row item.

[1203] 506. The system of embodiment 504, wherein the selection of the content attribute row item is made by a user via one of (i) a single action, or (ii) a double action.

[1204] 507. The system of embodiment 503, wherein rows and columns are orthogonal and not spatially perpendicular.

[1205] 508. The system of embodiment 503, wherein the media display view corresponds to a time period, and wherein the processor issues further instructions to:

[1206] obtain data corresponding to a plurality of content attribute row items spotlighted during the time period; and

[1207] display the obtained data in corresponding content attribute rows and selection columns.

[1208] 509. The system of embodiment 503, wherein the content attribute row items are selected from a group including: (i) artists, (ii) tracks, (iii) albums, (iv) playlists, (v) gurus, and (vi) social graph.

[1209] 510. A processor-readable medium storing processor-issuable instructions for displaying content items wherein the processor issues instructions to:

[1210] obtain a user selection of a content attribute row item;

[1211] display the selected content attribute row item in a corresponding attribute row of a first selection column of a media display view;

[1212] identify related content attribute row items; and

[1213] provide the identified related content attribute row items for selection; and

[1214] display the selected related content attribute row item in a corresponding attribute row of a subsequent selection column in the media display view.

[1215] 511. The medium of embodiment 510, wherein the selection column establishes the sequence of selection of the content attribute row items.

[1216] 512. The medium of embodiment 510, further comprising a media selection view wherein the processor issues instructions to display a designated spotlight area that is configured to receive the selected content attribute row item.

[1217] 513. The medium of embodiment 511, wherein the selection of the content attribute row item is made by a user via one of (i) a single action, or (ii) a double action.

[1218] 514. The medium of embodiment 510, wherein rows and columns are orthogonal and not spatially perpendicular.

[1219] 515. The medium of embodiment 510, wherein each media display view corresponds to a time period, and wherein the processor issues further instructions to:

[1220] obtain data corresponding to a plurality of content attribute row items spotlighted during the time period; and

[1221] display the obtained data in corresponding content attribute rows and selection columns.

[1222] 516. The medium of embodiment 510, wherein the content attribute row items are selected from a group including: (i) artists, (ii) tracks, (iii) albums, (iv) playlists, (v) gurus, and (vi) social graph.

[1223] 517. A search and discovery interface, comprising:

[1224] an entry point representation surrounded by a plurality of discovery supportive heuristic representations,

[1225] wherein the entry point representation is configured to receive a first universally resolvable content item selected from one of the plurality of discovery supportive heuristic representations;

[1226] wherein the interface is configured to identify, for each of the plurality of discovery supportive heuristic representations, a universally resolvable content item related to the content item in the entry point representation; and

[1227] wherein each of the plurality of discovery supportive heuristic representations is configured to display the corresponding identified universally resolvable content item.

[1228] 518. The interface of embodiment 517, wherein the plurality of discovery supportive heuristic representations include (i) track, (ii) artist, (iii) album, (iv) gurus, and (v) playlist.

[1229] 519. The interface of embodiment 518, wherein the content item in each of the plurality of discovery supportive heuristic representations is the most related (i) track, (ii) artist, (iii) album, (iv) gurus, and (v) playlist.

[1230] 520. The interface of embodiment 517, wherein the universally resolvable content seed in the entry point repre-

sentation and in each one of the plurality of discovery supportive category representations is shareable with one or more users.

[1231] 521. The interface of embodiment 517, wherein the universally resolvable content seed in the entry point representation and in each one of the plurality of discovery supportive category representations is downloadable to one or more client devices.

[1232] 522. The interface of embodiment 517, wherein the universally resolvable content seed in the entry point representation and in each one of the plurality of discovery supportive category representations is includable in one or more playlists.

[1233] 523. A processor-implemented search and discovery method, comprising:

[1234] receiving at an entry point representation a first universally resolvable content item seed selected from one of a plurality of discovery supportive heuristic representations;

[1235] obtaining for each of the plurality of discovery supportive heuristic representations a universally resolvable content item related to the content item seed; and

[1236] displaying at each of the plurality of discovery supportive heuristic representations the corresponding related universally resolvable content item.

[1237] 524. The method of embodiment 523, wherein the plurality of discovery supportive heuristic representations include (i) track, (ii) artist, (iii) album, (iv) gurus, and (v) playlist.

[1238] 525. The method of embodiment 524, wherein the content item in each of the plurality of discovery supportive heuristic representations is the most related (i) track, (ii) artist, (iii) album, (iv) gurus, and (v) playlist.

[1239] 526. The method of embodiment 523, wherein the universally resolvable content seed in the entry point representation and in each one of the plurality of discovery supportive category representations is shareable with one or more users.

[1240] 527. The method of embodiment 523, wherein the universally resolvable content seed in the entry point representation and in each one of the plurality of discovery supportive category representations is downloadable to one or more client devices.

[1241] 528. The method of embodiment 523, wherein the universally resolvable content seed in the entry point representation and in each one of the plurality of discovery supportive category representations is includable in one or more playlists.

[1242] 529. A search and discovery system, comprising:

[1243] a memory;

[1244] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1245] receive at an entry point representation a first universally resolvable content item seed selected from one of a plurality of discovery supportive heuristic representations;

[1246] obtain for each of the plurality of discovery supportive heuristic representations a universally resolvable content item related to the content item seed; and

[1247] display at each of the plurality of discovery supportive heuristic representations the corresponding related universally resolvable content item.

[1248] 530. The system of embodiment 529, wherein the plurality of discovery supportive heuristic representations include (i) track, (ii) artist, (iii) album, (iv) gurus, and (v) playlist.

[1249] 531. The system of embodiment 530, wherein the content item in each of the plurality of discovery supportive heuristic representations is the most related (i) track, (ii) artist, (iii) album, (iv) gurus, and (v) playlist.

[1250] 532. The system of embodiment 529, wherein the universally resolvable content seed in the entry point representation and in each one of the plurality of discovery supportive category representations is shareable with one or more users.

[1251] 533. The system of embodiment 529, wherein the universally resolvable content seed in the entry point representation and in each one of the plurality of discovery supportive category representations is downloadable to one or more client devices.

[1252] 534. The system of embodiment 529, wherein the universally resolvable content seed in the entry point representation and in each one of the plurality of discovery supportive category representations is includable in one or more playlists.

[1253] 535. A processor-readable search and discovery medium storing processor-issuable instructions, wherein the processor issues instructions to:

[1254] receive at an entry point representation a first universally resolvable content item seed selected from one of a plurality of discovery supportive heuristic representations;

[1255] obtain for each of the plurality of discovery supportive heuristic representations a universally resolvable content item related to the content item seed; and

[1256] display at each of the plurality of discovery supportive heuristic representations the corresponding related universally resolvable content item.

[1257] 536. The medium of embodiment 535, wherein the plurality of discovery supportive heuristic representations include (i) track, (ii) artist, (iii) album, (iv) gurus, and (v) playlist.

[1258] 537. The medium of embodiment 536, wherein the content item in each of the plurality of discovery supportive heuristic representations is the most related (i) track, (ii) artist, (iii) album, (iv) gurus, and (v) playlist.

[1259] 538. The medium of embodiment 535, wherein the universally resolvable content seed in the entry point representation and in each one of the plurality of discovery supportive category representations is shareable with one or more users.

[1260] 539. The medium of embodiment 535, wherein the universally resolvable content seed in the entry point representation and in each one of the plurality of discovery supportive category representations is downloadable to one or more client devices.

[1261] 540. The medium of embodiment 535, wherein the universally resolvable content seed in the entry point representation and in each one of the plurality of discovery supportive category representations is includable in one or more playlists.

[1262] 541. A processor-implemented method for providing universally resolvable content items, comprising:

[1263] receiving user selection of a universally resolvable content seed;

[1264] determining via a processor socially influenced content attributes associated with the universally resolvable content seed;

[1265] querying a universally resolvable content database using the socially influenced content attributes;

[1266] obtaining a ranked list of universally resolvable content items from the querying;

[1267] providing the ranked list of universally resolvable content items to the user.

[1268] 542. The method of embodiment 541, further comprising:

[1269] creating a content query based on the socially influenced content attributes for the querying.

[1270] 543. The method of embodiment 542, further comprising:

[1271] creating socially influenced ranking categories and populating the ranking categories with results from the query to obtain the ranked list of universally resolvable content items.

[1272] 544. The method of embodiment 541, wherein determining the socially influenced content attributes associated with the universally resolvable content seed includes:

[1273] determining if the universally resolvable content seed is a universally resolvable content item seed.

[1274] 545. The method of embodiment 544, wherein when the universally resolvable content seed is a universally resolvable content item seed, determining the socially influenced content attributes includes:

[1275] retrieving user profile and social graph preferences;

[1276] utilizing the retrieved user profile and social graph preferences to create socially influenced ranking categories to obtain the ranked list of universally resolvable content items.

[1277] 546. The method of embodiment 545, wherein the social graph preferences are derived from the user's one or more social networks.

[1278] 547. The method of embodiment 544, wherein when the universally resolvable content seed is not a content item seed, determining the socially influenced content attributes further comprises:

[1279] selecting a socially influenced content item seed selection criterion; and

[1280] retrieving a universally resolvable content item seed associated with the universally resolvable content seed based on the selected criterion.

[1281] 548. The method of embodiment 547, further comprising:

[1282] retrieving user profile and social graph preferences; and

[1283] utilizing the retrieved user profile and social graph preferences to create socially influenced ranking categories for obtaining the ranked list of universally resolvable content items.

[1284] 549. The method of embodiment 547, wherein the socially influenced content item seed selection criterion is one of a: (i) a specified default, (ii) currently played track, (iii) most played track, (iv) favorite track, (v) favorite genre, and (vi) last purchase.

[1285] 550. The method of embodiment 543, wherein the results from the query are assigned a weight corresponding to relevance to the content query.

[1286] 551. The method of embodiment 550, wherein obtaining the ranked list of universally resolvable content item further comprises:

[1287] assigning a category weight to each one of the socially influenced ranking categories; and

[1288] ranking the socially influenced categorized results based on a calculated social influence score.

[1289] 552. The method of embodiment 551, further comprising:

[1290] for each one of the unique results, calculating the social influence score based on the assigned category weight and the weight corresponding to relevance to the content query.

[1291] 553. The method of embodiment 541, further comprising:

[1292] obtaining from the user's client device a local cache request for a non-local universally resolvable content item; and

[1293] providing the non-local universally resolvable content item to the user's client device.

[1294] 554. The method of embodiment 551, wherein the non-local universally resolvable content item in the user's client device is marked as temporary.

[1295] 555. The method of embodiment 541, further comprising:

[1296] determining via the processor interest graph influenced content attributes associated with the universally resolvable content seed;

[1297] querying the universally resolvable content database using the interest graph influenced content attributes;

[1298] obtaining a ranked list of universally resolvable content items from the querying;

[1299] providing the ranked list of universally resolvable content items to the user.

[1300] 556. A system for providing universally resolvable content items, comprising:

[1301] a memory;

[1302] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1303] receive user selection of a universally resolvable content seed;

[1304] determine via a processor socially influenced content attributes associated with the universally resolvable content seed;

[1305] query a universally resolvable content database using the socially influenced content attributes;

[1306] obtain a ranked list of universally resolvable content items from the querying; and

[1307] provide the ranked list of universally resolvable content items to the user.

[1308] 557. The system of embodiment 556, wherein the processor issues further instructions to:

[1309] create a content query based on the socially influenced content attributes for the querying.

[1310] 558. The system of embodiment 557, wherein the processor issues further instructions to:

[1311] create socially influenced ranking categories and populating the ranking categories with results from the query to obtain the ranked list of universally resolvable content items.

[1312] 559. The system of embodiment 556, wherein the instructions to determine the socially influenced content attributes associated with the universally resolvable content seed includes instructions to:

[1313] determine if the universally resolvable content seed is a universally resolvable content item seed.

[1314] 560. The system of embodiment 559, wherein when the universally resolvable content seed is a universally resolvable content item seed, the instructions to determine the socially influenced content attributes includes instructions to:

[1315] retrieve user profile and social graph preferences;

[1316] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories to obtain the ranked list of universally resolvable content items.

[1317] 561. The system of embodiment 560, wherein the social graph preferences are derived from the user's one or more social networks.

[1318] 562. The system of embodiment 559, wherein when the universally resolvable content seed is not a content item seed, the instructions to determine the socially influenced content attributes further comprises instructions to:

[1319] select a socially influenced content item seed selection criterion; and

[1320] retrieve a universally resolvable content item seed associated with the universally resolvable content seed based on the selected criterion.

[1321] 563. The system of embodiment 562, wherein the processor issues further instructions to:

[1322] retrieve user profile and social graph preferences; and

[1323] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories for obtaining the ranked list of universally resolvable content items.

[1324] 564. The system of embodiment 562, wherein the socially influenced content item seed selection criterion is one of a: (i) a specified default, (ii) currently played track, (iii) most played track, (iv) favorite track, (v) favorite genre, and (vi) last purchase.

[1325] 565. The system of embodiment 558, wherein the results from the query are assigned a weight corresponding to relevance to the content query.

[1326] 566. The system of embodiment 565, wherein the instructions to obtain the ranked list of universally resolvable content item further comprises instructions to:

[1327] assign a category weight to each one of the socially influenced ranking categories; and

[1328] rank the socially influenced categorized results based on a calculated social influence score.

[1329] 567. The system of embodiment 566, wherein for each one of the unique results, the processor issues further instructions to calculate the social influence score based on the assigned category weight and the weight corresponding to relevance to the content query.

[1330] 568. The system of embodiment 556, wherein the processor issues further instructions to:

[1331] obtain from the user's client device a local cache request for a non-local universally resolvable content item; and

[1332] provide the non-local universally resolvable content item to the user's client device.

[1333] 569. The system of embodiment 568, wherein the non-local universally resolvable content item in the user's client device is marked as temporary.

[1334] 570. The system of embodiment 556, wherein the processor issues further instructions to:

[1335] determine via the processor interest graph influenced content attributes associated with the universally resolvable content seed;

[1336] query the universally resolvable content database using the interest graph influenced content attributes;

[1337] obtain a ranked list of universally resolvable content items from the querying;

[1338] provide the ranked list of universally resolvable content items to the user.

[1339] 571. A processor-readable medium storing processor-issuable instructions for providing universally resolvable content items, wherein the processor issues instructions to:

[1340] receive user selection of a universally resolvable content seed;

[1341] determine via a processor socially influenced content attributes associated with the universally resolvable content seed;

[1342] query a universally resolvable content database using the socially influenced content attributes;

[1343] obtain a ranked list of universally resolvable content items from the querying; and

[1344] provide the ranked list of universally resolvable content items to the user.

[1345] 572. The medium of embodiment 571, wherein the processor issues further instructions to:

[1346] create a content query based on the socially influenced content attributes for the querying.

[1347] 573. The medium of embodiment 572, wherein the processor issues further instructions to:

[1348] create socially influenced ranking categories and populating the ranking categories with results from the query to obtain the ranked list of universally resolvable content items.

[1349] 574. The medium of embodiment 571, wherein the instructions to determine the socially influenced content attributes associated with the universally resolvable content seed includes instructions to:

[1350] determine if the universally resolvable content seed is a universally resolvable content item seed.

[1351] 575. The medium of embodiment 574, wherein when the universally resolvable content seed is a universally resolvable content item seed, the instructions to determine the socially influenced content attributes includes instructions to:

[1352] retrieve user profile and social graph preferences;

[1353] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories to obtain the ranked list of universally resolvable content items.

[1354] 576. The medium of embodiment 575, wherein the social graph preferences are derived from the user's one or more social networks.

[1355] 577. The medium of embodiment 574, wherein when the universally resolvable content seed is not a content item seed, the instructions to determine the socially influenced content attributes further comprises instructions to:

[1356] select a socially influenced content item seed selection criterion; and

[1357] retrieve a universally resolvable content item seed associated with the universally resolvable content seed based on the selected criterion.

[1358] 578. The medium of embodiment 577, wherein the processor issues further instructions to:

[1359] retrieve user profile and social graph preferences; and

[1360] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories for obtaining the ranked list of universally resolvable content items.

[1361] 579. The medium of embodiment 577, wherein the socially influenced content item seed selection criterion is one of a: (i) a specified default, (ii) currently played track, (iii) most played track, (iv) favorite track, (v) favorite genre, and (vi) last purchase.

[1362] 580. The medium of embodiment 573, wherein the results from the query are assigned a weight corresponding to relevance to the content query.

[1363] 581. The medium of embodiment 580, wherein the instructions to obtain the ranked list of universally resolvable content item further comprises instructions to:

[1364] assign a category weight to each one of the socially influenced ranking categories; and

[1365] rank the socially influenced categorized results based on a calculated social influence score.

[1366] 582. The medium of embodiment 581, wherein for each one of the unique results, the processor issues further instructions to calculate the social influence score based on the assigned category weight and the weight corresponding to relevance to the content query.

[1367] 583. The medium of embodiment 571, wherein the processor issues further instructions to:

[1368] obtain from the user's client device a local cache request for a non-local universally resolvable content item; and

[1369] provide the non-local universally resolvable content item to the user's client device.

[1370] 584. The medium of embodiment 583, wherein the non-local universally resolvable content item in the user's client device is marked as temporary.

[1371] 585. The medium of embodiment 571, wherein the processor issues further instructions to:

[1372] determine via the processor interest graph influenced content attributes associated with the universally resolvable content seed;

[1373] query the universally resolvable content database using the interest graph influenced content attributes;

[1374] obtain a ranked list of universally resolvable content items from the querying;

[1375] provide the ranked list of universally resolvable content items to the user.

[1376] 586. An apparatus for providing universally resolvable content items, comprising:

[1377] a memory;

[1378] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1379] receive user selection of a universally resolvable content seed;

[1380] determine via a processor socially influenced content attributes associated with the universally resolvable content seed;

[1381] query a universally resolvable content database using the socially influenced content attributes;

[1382] obtain a ranked list of universally resolvable content items from the querying; and

[1383] provide the ranked list of universally resolvable content items to the user.

[1384] 587. The apparatus of embodiment 586, wherein the processor issues further instructions to:

[1385] create a content query based on the socially influenced content attributes for the querying.

[1386] 588. The apparatus of embodiment 587, wherein the processor issues further instructions to:

[1387] create socially influenced ranking categories and populating the ranking categories with results from the query to obtain the ranked list of universally resolvable content items.

[1388] 589. The apparatus of embodiment 586, wherein the instructions to determine the socially influenced content attributes associated with the universally resolvable content seed includes instructions to:

[1389] determine if the universally resolvable content seed is a universally resolvable content item seed.

[1390] 590. The apparatus of embodiment 589, wherein when the universally resolvable content seed is a universally resolvable content item seed, the instructions to determine the socially influenced content attributes includes instructions to:

[1391] retrieve user profile and social graph preferences;

[1392] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories to obtain the ranked list of universally resolvable content items.

[1393] 591. The apparatus of embodiment 590, wherein the social graph preferences are derived from the user's one or more social networks.

[1394] 592. The apparatus of embodiment 589, wherein when the universally resolvable content seed is not a content item seed, the instructions to determine the socially influenced content attributes further comprises instructions to:

[1395] select a socially influenced content item seed selection criterion; and

[1396] retrieve a universally resolvable content item seed associated with the universally resolvable content seed based on the selected criterion.

[1397] 593. The apparatus of embodiment 592, wherein the processor issues further instructions to:

[1398] retrieve user profile and social graph preferences; and

[1399] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories for obtaining the ranked list of universally resolvable content items.

[1400] 594. The apparatus of embodiment 592, wherein the socially influenced content item seed selection criterion is one of a: (i) a specified default, (ii) currently played track, (iii) most played track, (iv) favorite track, (v) favorite genre, and (vi) last purchase.

[1401] 595. The apparatus of embodiment 588, wherein the results from the query are assigned a weight corresponding to relevance to the content query.

[1402] 596. The apparatus of embodiment 595, wherein the instructions to obtain the ranked list of universally resolvable content item further comprises instructions to:

[1403] assign a category weight to each one of the socially influenced ranking categories; and

[1404] rank the socially influenced categorized results based on a calculated social influence score.

[1405] 597. The apparatus of embodiment 596, wherein for each one of the unique results, the processor issues further instructions to calculate the social influence score based on the assigned category weight and the weight corresponding to relevance to the content query.

[1406] 598. The apparatus of embodiment 586, wherein the processor issues further instructions to:

[1407] obtain from the user's client device a local cache request for a non-local universally resolvable content item; and

[1408] provide the non-local universally resolvable content item to the user's client device.

[1409] 599. The apparatus of embodiment 598, wherein the non-local universally resolvable content item in the user's client device is marked as temporary.

[1410] 600. The apparatus of embodiment 586, wherein the processor issues further instructions to:

[1411] determine via the processor interest graph influenced content attributes associated with the universally resolvable content seed;

[1412] query the universally resolvable content database using the interest graph influenced content attributes;

[1413] obtain a ranked list of universally resolvable content items from the querying;

[1414] provide the ranked list of universally resolvable content items to the user.

[1415] 601. A processor-implemented method for providing universally resolvable content items, comprising:

[1416] providing user selection of a universally resolvable content seed;

[1417] providing a request to determine via a processor socially influenced content attributes associated with the universally resolvable content seed;

[1418] providing a request to querying a universally resolvable content database using the socially influenced content attributes;

[1419] providing a request obtaining a ranked list of universally resolvable content items from the querying;

[1420] receiving the ranked list of universally resolvable content items to the user.

[1421] 602. The method of embodiment 601, further comprising:

[1422] creating a content query based on the socially influenced content attributes for the querying.

[1423] 603. The method of embodiment 602, further comprising:

[1424] providing authorization to create socially influenced ranking categories and populating the ranking categories with results from the query to obtain the ranked list of universally resolvable content items.

[1425] 604. The method of embodiment 601, wherein determining the socially influenced content attributes associated with the universally resolvable content seed includes:

[1426] providing authorization to determine if the universally resolvable content seed is a universally resolvable content item seed.

[1427] 605. The method of embodiment 604, wherein when the universally resolvable content seed is a universally resolvable content item seed, determining the socially influenced content attributes includes:

[1428] retrieving user profile and social graph preferences;

[1429] utilizing the retrieved user profile and social graph preferences to create socially influenced ranking categories to obtain the ranked list of universally resolvable content items.

[1430] 606. The method of embodiment 605, wherein the social graph preferences are derived from the user's one or more social networks.

[1431] 607. The method of embodiment 604, wherein when the universally resolvable content seed is not a content item seed, determining the socially influenced content attributes further comprises:

[1432] selecting a socially influenced content item seed selection criterion; and

[1433] retrieving a universally resolvable content item seed associated with the universally resolvable content seed based on the selected criterion.

[1434] 608. The method of embodiment 607, further comprising:

[1435] providing user profile and social graph preferences; and

[1436] providing authorization to utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories for obtaining the ranked list of universally resolvable content items.

[1437] 609. The method of embodiment 607, wherein the socially influenced content item seed selection criterion is one of a: (i) a specified default, (ii) currently played track, (iii) most played track, (iv) favorite track, (v) favorite genre, and (vi) last purchase.

[1438] 610. The method of embodiment 603, wherein the results from the query are assigned a weight corresponding to relevance to the content query.

[1439] 611. The method of embodiment 610, wherein obtaining the ranked list of universally resolvable content item further comprises:

[1440] assigning a category weight to each one of the socially influenced ranking categories; and

[1441] ranking the socially influenced categorized results based on a calculated social influence score.

[1442] 612. The method of embodiment 611, further comprising:

[1443] for each one of the unique results, calculating the social influence score based on the assigned category weight and the weight corresponding to relevance to the content query.

[1444] 613. The method of embodiment 601, further comprising:

[1445] obtaining from the user's client device a local cache request for a non-local universally resolvable content item; and

[1446] providing the non-local universally resolvable content item to the user's client device.

[1447] 614. The method of embodiment 611, wherein the non-local universally resolvable content item in the user's client device is marked as temporary.

[1448] 615. The method of embodiment 601, further comprising:

[1449] providing authorization to determine via the processor interest graph influenced content attributes associated with the universally resolvable content seed;

[1450] providing authorization to query the universally resolvable content database using the interest graph influenced content attributes;

[1451] providing a ranked list of universally resolvable content items from the querying;

[1452] obtaining the ranked list of universally resolvable content items to the user.

[1453] 616. A system for providing universally resolvable content items, comprising:

[1454] a memory;

[1455] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1456] provide user selection of a universally resolvable content seed;

[1457] provide a request to determine via a processor socially influenced content attributes associated with the universally resolvable content seed;

[1458] provide a request to query a universally resolvable content database using the socially influenced content attributes;

[1459] provide a request to obtain a ranked list of universally resolvable content items from the querying; and

[1460] obtain the ranked list of universally resolvable content items to the user.

[1461] 617. The system of embodiment 616, wherein the processor issues further instructions to:

[1462] create a content query based on the socially influenced content attributes for the querying.

[1463] 618. The system of embodiment 617, wherein the processor issues further instructions to:

[1464] create socially influenced ranking categories and populating the ranking categories with results from the query to obtain the ranked list of universally resolvable content items.

[1465] 619. The system of embodiment 616, wherein the instructions to determine the socially influenced content attributes associated with the universally resolvable content seed includes instructions to:

[1466] determine if the universally resolvable content seed is a universally resolvable content item seed.

[1467] 620. The system of embodiment 619, wherein when the universally resolvable content seed is a universally resolvable content item seed, the instructions to determine the socially influenced content attributes includes instructions to:

[1468] retrieve user profile and social graph preferences;

[1469] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories to obtain the ranked list of universally resolvable content items.

[1470] 621. The system of embodiment 620, wherein the social graph preferences are derived from the user's one or more social networks.

[1471] 622. The system of embodiment 619, wherein when the universally resolvable content seed is not a content item seed, the instructions to determine the socially influenced content attributes further comprises instructions to:

[1472] select a socially influenced content item seed selection criterion; and

[1473] retrieve a universally resolvable content item seed associated with the universally resolvable content seed based on the selected criterion.

[1474] 623. The system of embodiment 622, wherein the processor issues further instructions to:

[1475] retrieve user profile and social graph preferences; and

[1476] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories for obtaining the ranked list of universally resolvable content items.

[1477] 624. The system of embodiment 622, wherein the socially influenced content item seed selection criterion is one of a: (i) a specified default, (ii) currently played track, (iii) most played track, (iv) favorite track, (v) favorite genre, and (vi) last purchase.

[1478] 625. The system of embodiment 618, wherein the results from the query are assigned a weight corresponding to relevance to the content query.

[1479] 626. The system of embodiment 625, wherein the instructions to obtain the ranked list of universally resolvable content item further comprises instructions to:

[1480] assign a category weight to each one of the socially influenced ranking categories; and

[1481] rank the socially influenced categorized results based on a calculated social influence score.

[1482] 627. The system of embodiment 626, wherein for each one of the unique results, the processor issues further instructions to calculate the social influence score based on the assigned category weight and the weight corresponding to relevance to the content query.

[1483] 628. The system of embodiment 616, wherein the processor issues further instructions to:

[1484] obtain from the user's client device a local cache request for a non-local universally resolvable content item; and

[1485] provide the non-local universally resolvable content item to the user's client device.

[1486] 629. The system of embodiment 628, wherein the non-local universally resolvable content item in the user's client device is marked as temporary.

[1487] 630. The system of embodiment 616, wherein the processor issues further instructions to:

[1488] provide a request to determine via the processor interest graph influenced content attributes associated with the universally resolvable content seed;

[1489] provide a request to query the universally resolvable content database using the interest graph influenced content attributes;

[1490] provide a request to obtain a ranked list of universally resolvable content items from the querying;

[1491] obtain the ranked list of universally resolvable content items to the user.

[1492] 631. A processor-readable medium storing processor-issuable instructions for providing universally resolvable content items, wherein the processor issues instructions to:

[1493] provide user selection of a universally resolvable content seed;

[1494] provide a request to determine via a processor socially influenced content attributes associated with the universally resolvable content seed;

[1495] provide a request to query a universally resolvable content database using the socially influenced content attributes;

[1496] provide a request to obtain a ranked list of universally resolvable content items from the querying; and

[1497] obtain the ranked list of universally resolvable content items to the user.

[1498] 632. The medium of embodiment 631, wherein the processor issues further instructions to:

[1499] create a content query based on the socially influenced content attributes for the querying.

[1500] 633. The medium of embodiment 632, wherein the processor issues further instructions to:

[1501] create socially influenced ranking categories and populating the ranking categories with results from the query to obtain the ranked list of universally resolvable content items.

[1502] 634. The medium of embodiment 631, wherein the instructions to determine the socially influenced content attributes associated with the universally resolvable content seed includes instructions to:

[1503] determine if the universally resolvable content seed is a universally resolvable content item seed.

[1504] 635. The medium of embodiment 634, wherein when the universally resolvable content seed is a universally resolvable content item seed, the instructions to determine the socially influenced content attributes includes instructions to:

[1505] retrieve user profile and social graph preferences;

[1506] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories to obtain the ranked list of universally resolvable content items.

[1507] 636. The medium of embodiment 635, wherein the social graph preferences are derived from the user's one or more social networks.

[1508] 637. The medium of embodiment 634, wherein when the universally resolvable content seed is not a content item seed, the instructions to determine the socially influenced content attributes further comprises instructions to:

[1509] select a socially influenced content item seed selection criterion; and

[1510] retrieve a universally resolvable content item seed associated with the universally resolvable content seed based on the selected criterion.

[1511] 638. The medium of embodiment 637, wherein the processor issues further instructions to:

[1512] retrieve user profile and social graph preferences; and

[1513] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories for obtaining the ranked list of universally resolvable content items.

[1514] 639. The medium of embodiment 637, wherein the socially influenced content item seed selection criterion is one of a: (i) a specified default, (ii) currently played track, (iii) most played track, (iv) favorite track, (v) favorite genre, and (vi) last purchase.

[1515] 640. The medium of embodiment 633, wherein the results from the query are assigned a weight corresponding to relevance to the content query.

[1516] 641. The medium of embodiment 640, wherein the instructions to obtain the ranked list of universally resolvable content item further comprises instructions to:

[1517] assign a category weight to each one of the socially influenced ranking categories; and

[1518] rank the socially influenced categorized results based on a calculated social influence score.

[1519] 642. The medium of embodiment 641, wherein for each one of the unique results, the processor issues further instructions to calculate the social influence score based on the assigned category weight and the weight corresponding to relevance to the content query.

[1520] 643. The medium of embodiment 631, wherein the processor issues further instructions to:

[1521] obtain from the user's client device a local cache request for a non-local universally resolvable content item; and

[1522] provide the non-local universally resolvable content item to the user's client device.

[1523] 644. The medium of embodiment 643, wherein the non-local universally resolvable content item in the user's client device is marked as temporary.

[1524] 645. The medium of embodiment 631, wherein the processor issues further instructions to:

[1525] provide a request to determine via the processor interest graph influenced content attributes associated with the universally resolvable content seed;

[1526] provide a request to query the universally resolvable content database using the interest graph influenced content attributes;

[1527] provide a request to obtain a ranked list of universally resolvable content items from the querying;

[1528] obtain the ranked list of universally resolvable content items to the user.

[1529] 646. An apparatus for providing universally resolvable content items, comprising:

[1530] a memory;

[1531] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1532] provide user selection of a universally resolvable content seed;

[1533] provide a request to determine via a processor socially influenced content attributes associated with the universally resolvable content seed;

[1534] provide a request to query a universally resolvable content database using the socially influenced content attributes;

[1535] provide a request to obtain a ranked list of universally resolvable content items from the querying; and

[1536] obtain the ranked list of universally resolvable content items to the user.

[1537] 647. The apparatus of embodiment 646, wherein the processor issues further instructions to:

[1538] create a content query based on the socially influenced content attributes for the querying.

[1539] 648. The apparatus of embodiment 647, wherein the processor issues further instructions to:

[1540] create socially influenced ranking categories and populating the ranking categories with results from the query to obtain the ranked list of universally resolvable content items.

[1541] 649. The apparatus of embodiment 646, wherein the instructions to determine the socially influenced content attributes associated with the universally resolvable content seed includes instructions to:

[1542] determine if the universally resolvable content seed is a universally resolvable content item seed.

[1543] 650. The apparatus of embodiment 649, wherein when the universally resolvable content seed is a universally resolvable content item seed, the instructions to determine the socially influenced content attributes includes instructions to:

[1544] retrieve user profile and social graph preferences;

[1545] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories to obtain the ranked list of universally resolvable content items.

[1546] 651. The apparatus of embodiment 650, wherein the social graph preferences are derived from the user's one or more social networks.

[1547] 652. The apparatus of embodiment 649, wherein when the universally resolvable content seed is not a content item seed, the instructions to determine the socially influenced content attributes further comprises instructions to:

[1548] select a socially influenced content item seed selection criterion; and

[1549] retrieve a universally resolvable content item seed associated with the universally resolvable content seed based on the selected criterion.

[1550] 653. The apparatus of embodiment 652, wherein the processor issues further instructions to:

[1551] retrieve user profile and social graph preferences; and

[1552] utilize the retrieved user profile and social graph preferences to create socially influenced ranking categories for obtaining the ranked list of universally resolvable content items.

[1553] 654. The apparatus of embodiment 652, wherein the socially influenced content item seed selection criterion is one of a: (i) a specified default, (ii) currently played track, (iii) most played track, (iv) favorite track, (v) favorite genre, and (vi) last purchase.

[1554] 655. The apparatus of embodiment 648, wherein the results from the query are assigned a weight corresponding to relevance to the content query.

[1555] 656. The apparatus of embodiment 655, wherein the instructions to obtain the ranked list of universally resolvable content item further comprises instructions to:

[1556] assign a category weight to each one of the socially influenced ranking categories; and

[1557] rank the socially influenced categorized results based on a calculated social influence score.

[1558] 657. The apparatus of embodiment 656, wherein for each one of the unique results, the processor issues further instructions to calculate the social influence score based on the assigned category weight and the weight corresponding to relevance to the content query.

[1559] 658. The apparatus of embodiment 646, wherein the processor issues further instructions to:

[1560] obtain from the user's client device a local cache request for a non-local universally resolvable content item; and

[1561] provide the non-local universally resolvable content item to the user's client device.

[1562] 659. The apparatus of embodiment 658, wherein the non-local universally resolvable content item in the user's client device is marked as temporary.

[1563] 660. The apparatus of embodiment 646, wherein the processor issues further instructions to:

[1564] provide a request to determine via the processor interest graph influenced content attributes associated with the universally resolvable content seed;

[1565] provide a request to query the universally resolvable content database using the interest graph influenced content attributes;

[1566] provide a request to obtain a ranked list of universally resolvable content items from the querying;

[1567] obtain the ranked list of universally resolvable content items to the user.

[1568] 661. A processor-implemented method, comprising:

[1569] obtaining information relating to universally resolvable media content ("URMC") social influence weighted user engagement in at least one URMC socially influencing activity;

[1570] obtaining the user's social influence weight in a universally resolvable media content service;

[1571] obtaining a social influence weight associated with the activity; and

[1572] updating the user's social influence profile based on the activity.

[1573] 662. The method of embodiment 661, further comprising:

[1574] prior to the updating, determining whether the activity meets the social influence weight upgrade criteria.

[1575] 663. The method of embodiment 661, wherein updating the user's social influence profile based on the activity further comprises:

[1576] obtaining social influence points associated with the activity; and

[1577] determine the social influence weight corresponding to the social influence points.

[1578] 664. The method of embodiment 663, further comprising:

[1579] obtaining historical data corresponding to the user engagement in the activity; and

[1580] adjusting the social influence points based on the historical data.

[1581] 665. The method of embodiment 661, wherein the activity is at least one of: (i) playlist creation; (ii) playlist sharing; (iii) media content rating; (iv) messaging; (v) posting; and (vi) influencing a number of users to play, download or purchase one or more tracks.

[1582] 666. The method of embodiment 661, further comprising:

[1583] qualifying the user for at least one promotional incentive based on the updated social influence profile.

[1584] 667. The method of embodiment 666, wherein the at least one promotional incentive is one: (i) purchase discount; (ii) free merchandise; and (iii) admission to events.

[1585] 668. The method of embodiment 666, further comprising:

[1586] providing the user the at least one promotional incentive when the activity matches one or more activity criteria.

[1587] 669. The method of embodiment 661, wherein the socially influencing activity is related to at least one of: a social graph and an interest graph.

[1588] 670. The method of embodiment 665, further comprising:

[1589] determining whether the activity is a URMC socially influencing activity.

[1590] 671. The method of embodiment 670, wherein the determining is based on tracked frequency of user engagement with a playlist sharing activity.

[1591] 672. The method of embodiment 670, wherein the determining is based on tracked license purchase attributable to the user.

[1592] 673. The method of embodiment 670, wherein the determining is based on tracked number of users following and unfollowing the user.

[1593] 674. The method of embodiment 670, wherein the determining is based on tracked response to content sharing.

[1594] 675. The method of embodiment 670, wherein the determining is based on tracked usage volume of other users attributable to the user.

[1595] 676. A system, comprising:

[1596] a memory;

[1597] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1598] obtain information relating to universally resolvable media content ("URMC") social influence weighted user engagement in at least one URMC socially influencing activity;

[1599] obtain the user's social influence weight in a universally resolvable media content service;

[1600] obtain a social influence weight associated with the activity; and

[1601] update the user's social influence profile based on the activity.

[1602] 677. The system of embodiment 676, wherein the processor issues further instructions to:

[1603] prior to the update, determine whether the activity meets the social influence weight upgrade criteria.

[1604] 678. The system of embodiment 676, wherein the instructions to update the user's social influence profile based on the activity further comprises instructions to:

[1605] obtain social influence points associated with the activity; and

[1606] determine the social influence weight corresponding to the social influence points.

[1607] 679. The system of embodiment 678, wherein the processor issues further instructions to: obtain historical data corresponding to the user engagement in the activity; and

[1608] adjust the social influence points based on the historical data.

[1609] 680. The system of embodiment 676, wherein the activity is at least one of: (i) playlist creation; (ii) playlist sharing; (iii) media content rating; (iv) messaging; (v) posting; and (vi) influencing a number of users to play, download or purchase one or more tracks.

[1610] 681. The system of embodiment 676, wherein the processor issues further instructions to:

[1611] qualify the user for at least one promotional incentive based on the updated social influence profile.

[1612] 682. The system of embodiment 681, wherein the at least one promotional incentive is one: (i) purchase discount; (ii) free merchandise; and (iii) admission to events.

[1613] 683. The system of embodiment 681, wherein the processor issues further instructions to:

[1614] provide the user the at least one promotional incentive when the activity matches one or more activity criteria.

[1615] 684. The system of embodiment 676, wherein the socially influencing activity is related to at least one of: a social graph and an interest graph.

[1616] 685. The system of embodiment 680, wherein the processor issues further instructions to:

[1617] determine whether the activity is a URMC socially influencing activity.

[1618] 686. The system of embodiment 685, wherein the determination is based on tracked frequency of user engagement with a playlist sharing activity.

[1619] 687. The system of embodiment 685, wherein the determination is based on tracked license purchase attributable to the user.

[1620] 688. The system of embodiment 685, wherein the determination is based on tracked number of users following and unfollowing the user.

[1621] 689. The system of embodiment 685, wherein the determination is based on tracked response to content sharing.

[1622] 690. The system of embodiment 685, wherein the determination is based on tracked usage volume of other users attributable to the user.

[1623] 691. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[1624] obtain information relating to universally resolvable media content ("URMC") social influence weighted user engagement in at least one URMC socially influencing activity;

[1625] obtain the user's social influence weight in a universally resolvable media content service;

[1626] obtain a social influence weight associated with the activity; and

[1627] update the user's social influence profile based on the activity.

[1628] 692. The medium of embodiment 691, wherein the processor issues further instructions to:

[1629] prior to the update, determine whether the activity meets the social influence weight upgrade criteria.

[1630] 693. The medium of embodiment 691, wherein the instructions to update the user's social influence profile based on the activity further comprises instructions to:

[1631] obtain social influence points associated with the activity; and

[1632] determine the social influence weight corresponding to the social influence points.

[1633] 694. The medium of embodiment 693, wherein the processor issues further instructions to: obtain historical data corresponding to the user engagement in the activity; and

[1634] adjust the social influence points based on the historical data.

[1635] 695. The medium of embodiment 691, wherein the activity is at least one of: (i) playlist creation; (ii) playlist sharing; (iii) media content rating; (iv) messaging; (v) posting; and (vi) influencing a number of users to play, download or purchase one or more tracks.

[1636] 696. The medium of embodiment 691, wherein the processor issues further instructions to:

[1637] qualify the user for at least one promotional incentive based on the updated social influence profile.

[1638] 697. The medium of embodiment 696, wherein the at least one promotional incentive is one: (i) purchase discount; (ii) free merchandise; and (iii) admission to events.

[1639] 698. The medium of embodiment 696, wherein the processor issues further instructions to:

[1640] provide the user the at least one promotional incentive when the activity matches one or more activity criteria.

[1641] 699. The medium of embodiment 691, wherein the socially influencing activity is related to at least one of: a social graph and an interest graph.

[1642] 700. The medium of embodiment 695 wherein the processor issues further instructions to:

[1643] determine whether the activity is a URMC socially influencing activity.

[1644] 701. The medium of embodiment 700, wherein the determination is based on tracked frequency of user engagement with a playlist sharing activity.

[1645] 702. The medium of embodiment 700, wherein the determination is based on tracked license purchase attributable to the user.

[1646] 703. The medium of embodiment 700, wherein the determination is based on tracked number of users following and unfollowing the user.

[1647] 704. The medium of embodiment 700, wherein the determination is based on tracked response to content sharing.

[1648] 705. The medium of embodiment 700, wherein the determination is based on tracked usage volume of other users attributable to the user.

[1649] 706. An apparatus, comprising:

[1650] a memory;

[1651] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1652] obtain information relating to universally resolvable media content ("URMC") social influence weighted user engagement in at least one URMC socially influencing activity;

[1653] obtain the user's social influence weight in a universally resolvable media content service;

[1654] obtain a social influence weight associated with the activity; and

[1655] update the user's social influence profile based on the activity.

[1656] 707. The apparatus of embodiment 706, wherein the processor issues further instructions to:

[1657] prior to the update, determine whether the activity meets the social influence weight upgrade criteria.

[1658] 708. The apparatus of embodiment 706, wherein the instructions to update the user's social influence profile based on the activity further comprises instructions to:

[1659] obtain social influence points associated with the activity; and

[1660] determine the social influence weight corresponding to the social influence points.

[1661] 709. The apparatus of embodiment 708, wherein the processor issues further instructions to: obtain historical data corresponding to the user engagement in the activity; and

[1662] adjust the social influence points based on the historical data.

[1663] 710. The apparatus of embodiment 706, wherein the activity is at least one of: (i) playlist creation; (ii) playlist sharing; (iii) media content rating; (iv) messaging; (v) posting; and (vi) influencing a number of users to play, download or purchase one or more tracks.

[1664] 711. The apparatus of embodiment 706, wherein the processor issues further instructions to:

[1665] qualify the user for at least one promotional incentive based on the updated social influence profile.

[1666] 712. The apparatus of embodiment 711, wherein the at least one promotional incentive is one: (i) purchase discount; (ii) free merchandise; and (iii) admission to events.

[1667] 713. The apparatus of embodiment 711, wherein the processor issues further instructions to:

[1668] provide the user the at least one promotional incentive when the activity matches one or more activity criteria.

[1669] 714. The apparatus of embodiment 706, wherein the socially influencing activity is related to at least one of: a social graph and an interest graph.

[1670] 715. The apparatus of embodiment 710 wherein the processor issues further instructions to:

[1671] determine whether the activity is a URMC socially influencing activity.

[1672] 716. The apparatus of embodiment 715, wherein the determination is based on tracked frequency of user engagement with a playlist sharing activity.

[1673] 717. The apparatus of embodiment 715, wherein the determination is based on tracked license purchase attributable to the user.

[1674] 718. The apparatus of embodiment 715, wherein the determination is based on tracked number of users following and unfollowing the user.

[1675] 719. The apparatus of embodiment 715, wherein the determination is based on tracked response to content sharing.

[1676] 720. The apparatus of embodiment 715, wherein the determination is based on tracked usage volume of other users attributable to the user.

[1677] 721. A processor-implemented method, comprising:

[1678] providing information relating to universally resolvable media content ("URMC") social influence weighted user engagement in at least one URMC socially influencing activity;

[1679] providing the user's social influence weight in a universally resolvable media content service;

[1680] providing a social influence weight associated with the activity; and

[1681] providing a request to update the user's social influence profile based on the activity.

[1682] 722. The method of embodiment 721, further comprising:

[1683] prior to the updating, determining whether the activity meets the social influence weight upgrade criteria.

[1684] 723. The method of embodiment 721, wherein updating the user's social influence profile based on the activity further comprises:

[1685] obtaining social influence points associated with the activity; and

[1686] determine the social influence weight corresponding to the social influence points.

[1687] 724. The method of embodiment 723, further comprising:

[1688] obtaining historical data corresponding to the user engagement in the activity; and

[1689] adjusting the social influence points based on the historical data.

[1690] 725. The method of embodiment 721, wherein the activity is at least one of: (i) playlist creation; (ii) playlist sharing; (iii) media content rating; (iv) messaging; (v) posting; and (vi) influencing a number of users to play, download or purchase one or more tracks.

[1691] 726. The method of embodiment 721, further comprising:

[1692] qualifying the user for at least one promotional incentive based on the updated social influence profile.

[1693] 727. The method of embodiment 726, wherein the at least one promotional incentive is one: (i) purchase discount; (ii) free merchandise; and (iii) admission to events.

[1694] 728. The method of embodiment 726, further comprising:

[1695] providing the user the at least one promotional incentive when the activity matches one or more activity criteria.

[1696] 729. The method of embodiment 721, wherein the socially influencing activity is related to at least one of: a social graph and an interest graph.

[1697] 730. The method of embodiment 725, further comprising:

[1698] determining whether the activity is a URMC socially influencing activity.

[1699] 731. The method of embodiment 730, wherein the determining is based on tracked frequency of user engagement with a playlist sharing activity.

[1700] 732. The method of embodiment 730, wherein the determining is based on tracked license purchase attributable to the user.

[1701] 733. The method of embodiment 730, wherein the determining is based on tracked number of users following and unfollowing the user.

[1702] 734. The method of embodiment 730, wherein the determining is based on tracked response to content sharing.

[1703] 735. The method of embodiment 730, wherein the determining is based on tracked usage volume of other users attributable to the user.

[1704] 736. A system, comprising:

[1705] a memory;

[1706] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1707] provide information relating to universally resolvable media content ("URMC") social influence weighted user engagement in at least one URMC socially influencing activity;

[1708] provide the user's social influence weight in a universally resolvable media content service;

[1709] provide a social influence weight associated with the activity; and

[1710] providing a request to update the user's social influence profile based on the activity.

[1711] 737. The system of embodiment 736, wherein the processor issues further instructions to:

[1712] prior to the update, determine whether the activity meets the social influence weight upgrade criteria.

[1713] 738. The system of embodiment 736, wherein the instructions to update the user's social influence profile based on the activity further comprises instructions to:

[1714] obtain social influence points associated with the activity; and

[1715] determine the social influence weight corresponding to the social influence points.

[1716] 739. The system of embodiment 738, wherein the processor issues further instructions to: obtain historical data corresponding to the user engagement in the activity; and

[1717] adjust the social influence points based on the historical data.

[1718] 740. The system of embodiment 736, wherein the activity is at least one of: (i) playlist creation; (ii) playlist sharing; (iii) media content rating; (iv) messaging; (v) posting; and (vi) influencing a number of users to play, download or purchase one or more tracks.

[1719] 741. The system of embodiment 736, wherein the processor issues further instructions to:

[1720] qualify the user for at least one promotional incentive based on the updated social influence profile.

[1721] 742. The system of embodiment 741, wherein the at least one promotional incentive is one: (i) purchase discount; (ii) free merchandise; and (iii) admission to events.

[1722] 743. The system of embodiment 741, wherein the processor issues further instructions to:

[1723] provide the user the at least one promotional incentive when the activity matches one or more activity criteria.

[1724] 744. The system of embodiment 736, wherein the socially influencing activity is related to at least one of: a social graph and an interest graph.

[1725] 745. The system of embodiment 740, wherein the processor issues further instructions to:

[1726] determine whether the activity is a URMC socially influencing activity.

[1727] 746. The system of embodiment 745, wherein the determination is based on tracked frequency of user engagement with a playlist sharing activity.

[1728] 747. The system of embodiment 745, wherein the determination is based on tracked license purchase attributable to the user.

[1729] 748. The system of embodiment 745, wherein the determination is based on tracked number of users following and unfollowing the user.

[1730] 749. The system of embodiment 745, wherein the determination is based on tracked response to content sharing.

[1731] 750. The system of embodiment 745, wherein the determination is based on tracked usage volume of other users attributable to the user.

[1732] 751. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[1733] provide information relating to universally resolvable media content ("URMC") social influence weighted user engagement in at least one URMC socially influencing activity;

[1734] provide the user's social influence weight in a universally resolvable media content service;

[1735] provide a social influence weight associated with the activity; and

[1736] providing a request to update the user's social influence profile based on the activity.

[1737] 752. The medium of embodiment 751, wherein the processor issues further instructions to:

[1738] prior to the update, determine whether the activity meets the social influence weight upgrade criteria.

[1739] 753. The medium of embodiment 751, wherein the instructions to update the user's social influence profile based on the activity further comprises instructions to:

[1740] obtain social influence points associated with the activity; and

[1741] determine the social influence weight corresponding to the social influence points.

[1742] 754. The medium of embodiment 753, wherein the processor issues further instructions to: obtain historical data corresponding to the user engagement in the activity; and

[1743] adjust the social influence points based on the historical data.

[1744] 755. The medium of embodiment 751, wherein the activity is at least one of: (i) playlist creation; (ii) playlist sharing; (iii) media content rating; (iv) messaging; (v) posting; and (vi) influencing a number of users to play, download or purchase one or more tracks.

[1745] 756. The medium of embodiment 751, wherein the processor issues further instructions to:

[1746] qualify the user for at least one promotional incentive based on the updated social influence profile.

[1747] 757. The medium of embodiment 756, wherein the at least one promotional incentive is one: (i) purchase discount; (ii) free merchandise; and (iii) admission to events.

[1748] 758. The medium of embodiment 756, wherein the processor issues further instructions to:

[1749] provide the user the at least one promotional incentive when the activity matches one or more activity criteria.

[1750] 759. The medium of embodiment 751, wherein the socially influencing activity is related to at least one of: a social graph and an interest graph.

[1751] 760. The medium of embodiment 755 wherein the processor issues further instructions to:

[1752] determine whether the activity is a URMC socially influencing activity.

[1753] 761. The medium of embodiment 760, wherein the determination is based on tracked frequency of user engagement with a playlist sharing activity.

[1754] 762. The medium of embodiment 760, wherein the determination is based on tracked license purchase attributable to the user.

[1755] 763. The medium of embodiment 760, wherein the determination is based on tracked number of users following and unfollowing the user.

[1756] 764. The medium of embodiment 760, wherein the determination is based on tracked response to content sharing.

[1757] 765. The medium of embodiment 760, wherein the determination is based on tracked usage volume of other users attributable to the user.

[1758] 766. An apparatus, comprising:

[1759] a memory;

[1760] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1761] obtain information relating to universally resolvable media content ("URMC") social influence weighted user engagement in at least one URMC socially influencing activity;

[1762] provide the user's social influence weight in a universally resolvable media content service;

[1763] provide a social influence weight associated with the activity; and

[1764] provide a request to update the user's social influence profile based on the activity.

[1765] 767. The apparatus of embodiment 766, wherein the processor issues further instructions to:

[1766] prior to the update, determine whether the activity meets the social influence weight upgrade criteria.

[1767] 768. The apparatus of embodiment 766, wherein the instructions to update the user's social influence profile based on the activity further comprises instructions to:

[1768] obtain social influence points associated with the activity; and

[1769] determine the social influence weight corresponding to the social influence points.

[1770] 769. The apparatus of embodiment 768, wherein the processor issues further instructions to: obtain historical data corresponding to the user engagement in the activity; and

[1771] adjust the social influence points based on the historical data.

[1772] 770. The apparatus of embodiment 766, wherein the activity is at least one of: (i) playlist creation; (ii) playlist sharing; (iii) media content rating; (iv) messaging; (v) posting; and (vi) influencing a number of users to play, download or purchase one or more tracks.

[1773] 771. The apparatus of embodiment 766, wherein the processor issues further instructions to:

[1774] qualify the user for at least one promotional incentive based on the updated social influence profile.

[1775] 772. The apparatus of embodiment 771, wherein the at least one promotional incentive is one: (i) purchase discount; (ii) free merchandise; and (iii) admission to events.

[1776] 773. The apparatus of embodiment 771, wherein the processor issues further instructions to:

[1777] provide the user the at least one promotional incentive when the activity matches one or more activity criteria.

[1778] 774. The apparatus of embodiment 766, wherein the socially influencing activity is related to at least one of: a social graph and an interest graph.

[1779] 775. The apparatus of embodiment 770 wherein the processor issues further instructions to:

[1780] determine whether the activity is a URMC socially influencing activity.

[1781] 776. The apparatus of embodiment 775, wherein the determination is based on tracked frequency of user engagement with a playlist sharing activity.

[1782] 777. The apparatus of embodiment 775, wherein the determination is based on tracked license purchase attributable to the user.

[1783] 778. The apparatus of embodiment 775, wherein the determination is based on tracked number of users following and unfollowing the user.

[1784] 779. The apparatus of embodiment 775, wherein the determination is based on tracked response to content sharing.

[1785] 780. The apparatus of embodiment 775, wherein the determination is based on tracked usage volume of other users attributable to the user.

[1786] 781. A processor-implemented method, comprising:

[1787] detecting user initiation of a universally resolvable media content ("URMC") event in a client;

[1788] obtaining the URMC event identifying information;

[1789] recording the URMC event identifying information in association with the event in an event log in the client;

[1790] obtaining reporting frequency preference setting, wherein the preference setting includes at least one URMC user activity upload rule;

[1791] determining activation of a URMC upload threshold trigger by evaluating the at least one URMC user activity upload rule;

[1792] initiating reporting of the logged URMC event identifying information based on the trigger activation; and

[1793] updating the client upon successful acknowledgement of said reporting by a server.

[1794] 782. The method of embodiment 781, further comprising:

[1795] anonymizing the URMC event identifying information in the event log before initiating said reporting.

[1796] 783. The method of embodiment 782, wherein a decision for anonymizing the URMC event identifying information is based on user preference information.

[1797] 784. The method of embodiment 781, wherein when the client is disconnected from a communication network:

[1798] suspending said reporting until the client is connected to the communication network.

[1799] 785. The method of embodiment 784, wherein when said reporting is suspended and a user initiation of a URMC event is detected:

[1800] determining, using at least one URMC upload threshold rule, whether to disable contents of the URMC collection in the client.

[1801] 786. The method of embodiment 781, wherein the URMC event is one of: (i) content started; (ii) content paused; (iii) content stopped; (iv) content skipped; and (v) application closed.

[1802] 787. The method of embodiment 781, wherein the URMC event identifying information includes play session information including: play session start time, play session end time, total play session time for each content item in at least one of user library section, community section, discovery section and magic playlist.

[1803] 788. The method of embodiment 781, wherein the URMC event identifying information includes at least one of: (i) content event name; (ii) event time stamp; (iii) universally resolvable content identifier; and (iv) user information.

[1804] 789. The method of embodiment 781, wherein the at least one URMC user activity upload rule specifies a time for initiating reporting.

[1805] 790. The method of embodiment 781, wherein the at least one URMC user activity upload rule specifies a number of URMC events for initiating reporting.

[1806] 791. The method of embodiment 781, wherein the at least one URMC user activity upload rule specifies a number of each type of URMC event for initiating reporting.

[1807] 792. The method of embodiment 781, further comprising categorizing the URMC event as at least one of: (i) play event; (ii) import event; (iii) share event; (iv) community event; and (v) search event.

[1808] 793. The method of embodiment 792, further comprising selecting one or more URMC event categories for reporting based on the user's enrollment status in guru program.

[1809] 794. The method of embodiment 792, wherein the reported URMC event identifying information is used for determining influence points attributable to one or more URMC service users.

[1810] 795. The method of embodiment 792, wherein the reported URMC event identifying information is used for determining recommendations for the user.

[1811] 796. The method of embodiment 792, wherein the reported URMC event identifying information is used for determining license payouts to URMC service partners.

[1812] 797. A system, comprising:

[1813] a memory;

[1814] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1815] detect user initiation of a universally resolvable media content ("URMC") event in a client;

[1816] obtain the URMC event identifying information;

[1817] record the URMC event identifying information in association with the event in an event log in the client;

[1818] obtain reporting frequency preference setting, wherein the preference setting includes at least one URMC user activity upload rule;

[1819] determine activation of a URMC upload threshold trigger by evaluating the at least one URMC user activity upload rule;

[1820] initiate reporting of the logged URMC event identifying information based on the trigger activation; and

[1821] update the client upon successful acknowledgement of said reporting by a server.

[1822] 798. The system of embodiment 797, further comprising instructions to:

[1823] anonymize the URMC event identifying information in the event log before initiating said reporting.

[1824] 799. The system of embodiment 798, wherein a decision for anonymizing the URMC event identifying information is based on user preference information.

[1825] 800. The system of embodiment 797, wherein when the client is disconnected from a communication network, providing instructions to:

[1826] suspend said reporting until the client is connected to the communication network.

[1827] 801. The system of embodiment 800, wherein when said reporting is suspended and a user initiation of a URMC event is detected, providing instructions to:

[1828] determine, using at least one URMC upload threshold rule, whether to disable contents of the URMC collection in the client.

[1829] 802. The system of embodiment 797, wherein the URMC event is one of: (i) content started; (ii) content paused; (iii) content stopped; (iv) content skipped; and (v) application closed.

[1830] 803. The system of embodiment 797, wherein the URMC event identifying information includes play session information including: play session start time, play session end time, total play session time for each content item in at

least one of user library section, community section, discovery section and magic playlist.

[1831] 804. The system of embodiment 797, wherein the URMC event identifying information includes at least one of: (i) content event name; (ii) event time stamp; (iii) universally resolvable content identifier; and (iv) user information.

[1832] 805. The system of embodiment 797, wherein the at least one URMC user activity upload rule specifies a time for initiating reporting.

[1833] 806. The system of embodiment 797, wherein the at least one URMC user activity upload rule specifies a number of URMC events for initiating reporting.

[1834] 807. The system of embodiment 797, wherein the at least one URMC user activity upload rule specifies a number of each type of URMC event for initiating reporting.

[1835] 808. The system of embodiment 797, further comprising categorizing the URMC event as at least one of: (i) play event; (ii) import event; (iii) share event; (iv) community event; and (v) search event.

[1836] 809. The system of embodiment 808, further comprising instructions to select one or more URMC event categories for reporting based on the user's enrollment status in guru program.

[1837] 810. The system of embodiment 808, wherein the reported URMC event identifying information is used for determining influence points attributable to one or more URMC service users.

[1838] 811. The system of embodiment 808, wherein the reported URMC event identifying information is used for determining recommendations for the user.

[1839] 812. The system of embodiment 808, wherein the reported URMC event identifying information is used for determining license payouts to URMC service partners.

[1840] 813. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[1841] detect user initiation of a universally resolvable media content ("URMC") event in a client;

[1842] obtain the URMC event identifying information;

[1843] record the URMC event identifying information in association with the event in an event log in the client;

[1844] obtain reporting frequency preference setting, wherein the preference setting includes at least one URMC user activity upload rule;

[1845] determine activation of a URMC upload threshold trigger by evaluating the at least one URMC user activity upload rule;

[1846] initiate reporting of the logged URMC event identifying information based on the trigger activation; and

[1847] update the client upon successful acknowledgement of said reporting by a server.

[1848] 814. The medium of embodiment 813, further comprising instructions to:

[1849] anonymize the URMC event identifying information in the event log before initiating said reporting.

[1850] 815. The medium of embodiment 814, wherein a decision for anonymizing the URMC event identifying information is based on user preference information.

[1851] 816. The medium of embodiment 813, wherein when the client is disconnected from a communication network, providing instructions to:

[1852] suspend said reporting until the client is connected to the communication network.

[1853] 817. The medium of embodiment 816, wherein when said reporting is suspended and a user initiation of a URMC event is detected, providing instructions to:

[1854] determine, using at least one URMC upload threshold rule, whether to disable contents of the URMC collection in the client.

[1855] 818. The medium of embodiment 813, wherein the URMC event is one of: (1) content started; (ii) content paused; (iii) content stopped; (iv) content skipped; and (v) application closed.

[1856] 819. The medium of embodiment 813, wherein the URMC event identifying information includes play session information including: play session start time, play session end time, total play session time for each content item in at least one of user library section, community section, discovery section and magic playlist.

[1857] 820. The medium of embodiment 813, wherein the URMC event identifying information includes at least one of: (i) content event name; (ii) event time stamp; (iii) universally resolvable content identifier; and (iv) user information.

[1858] 821. The medium of embodiment 813, wherein the at least one URMC user activity upload rule specifies a time for initiating reporting.

[1859] 822. The medium of embodiment 813, wherein the at least one URMC user activity upload rule specifies a number of URMC events for initiating reporting.

[1860] 823. The medium of embodiment 813, wherein the at least one URMC user activity upload rule specifies a number of each type of URMC event for initiating reporting.

[1861] 824. The medium of embodiment 813, further comprising categorizing the URMC event as at least one of: (i) play event; (ii) import event; (iii) share event; (iv) community event; and (v) search event.

[1862] 825. The medium of embodiment 824, further comprising instructions to select one or more URMC event categories for reporting based on the user's enrollment status in guru program.

[1863] 826. The medium of embodiment 824, wherein the reported URMC event identifying information is used for determining influence points attributable to one or more URMC service users.

[1864] 827. The medium of embodiment 824, wherein the reported URMC event identifying information is used for determining recommendations for the user.

[1865] 828. The medium of embodiment 824, wherein the reported URMC event identifying information is used for determining license payouts to URMC service partners.

[1866] 829. An apparatus, comprising:

[1867] a memory;

[1868] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1869] detect user initiation of a universally resolvable media content ("URMC") event in a client;

[1870] obtain the URMC event identifying information;

[1871] record the URMC event identifying information in association with the event in an event log in the client;

[1872] obtain reporting frequency preference setting, wherein the preference setting includes at least one URMC user activity upload rule;

[1873] determine activation of a URMC upload threshold trigger by evaluating the at least one URMC user activity upload rule;

[1874] initiate reporting of the logged URMC event identifying information based on the trigger activation; and

[1875] update the client upon successful acknowledgement of said reporting by a server.

[1876] 830. The apparatus of embodiment 829, further comprising instructions to:

[1877] anonymize the URMC event identifying information in the event log before initiating said reporting.

[1878] 831. The apparatus of embodiment 830, wherein a decision for anonymizing the URMC event identifying information is based on user preference information.

[1879] 832. The apparatus of embodiment 829, wherein when the client is disconnected from a communication network, providing instructions to:

[1880] suspend said reporting until the client is connected to the communication network.

[1881] 833. The apparatus of embodiment 832, wherein when said reporting is suspended and a user initiation of a URMC event is detected, providing instructions to:

[1882] determine, using at least one URMC upload threshold rule, whether to disable contents of the URMC collection in the client.

[1883] 834. The apparatus of embodiment 829, wherein the URMC event is one of: (1) content started; (ii) content paused; (iii) content stopped; (iv) content skipped; and (v) application closed.

[1884] 835. The apparatus of embodiment 829, wherein the URMC event identifying information includes play session information including: play session start time, play session end time, total play session time for each content item in at least one of user library section, community section, discovery section and magic playlist.

[1885] 836. The apparatus of embodiment 829, wherein the URMC event identifying information includes at least one of: (i) content event name; (ii) event time stamp; (iii) universally resolvable content identifier; and (iv) user information.

[1886] 837. The apparatus of embodiment 829, wherein the at least one URMC user activity upload rule specifies a time for initiating reporting.

[1887] 838. The apparatus of embodiment 829, wherein the at least one URMC user activity upload rule specifies a number of URMC events for initiating reporting.

[1888] 839. The apparatus of embodiment 829, wherein the at least one URMC user activity upload rule specifies a number of each type of URMC event for initiating reporting.

[1889] 840. The apparatus of embodiment 829, further comprising categorizing the URMC event as at least one of: (i) play event; (ii) import event; (iii) share event; (iv) community event; and (v) search event.

[1890] 841. The apparatus of embodiment 840, further comprising instructions to select one or more URMC event categories for reporting based on the user's enrollment status in guru program.

[1891] 842. The apparatus of embodiment 840, wherein the reported URMC event identifying information is used for determining influence points attributable to one or more URMC service users.

[1892] 843. The apparatus of embodiment 840, wherein the reported URMC event identifying information is used for determining recommendations for the user.

[1893] 844. The apparatus of embodiment 840, wherein the reported URMC event identifying information is used for determining license payouts to URMC service partners.

[1894] 845. A processor-implemented method, comprising:

[1895] obtaining notification of a user initiation of a universally resolvable media content ("URMC") event in a client;

[1896] obtaining an event log recording the URMC event identifying information in association with the event;

[1897] providing reporting frequency preference setting, wherein the preference setting includes at least one URMC user activity upload rule;

[1898] obtaining reporting of the logged URMC event identifying information that is triggered in accordance with at least one URMC user activity upload rule; and

[1899] providing an acknowledgment to the client upon obtaining said reporting.

[1900] 846. A system, comprising:

[1901] a memory;

[1902] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1903] obtain notification of a user initiation of a universally resolvable media content ("URMC") event in a client;

[1904] obtain an event log recording the URMC event identifying information in association with the event;

[1905] provide reporting frequency preference setting, wherein the preference setting includes at least one URMC user activity upload rule;

[1906] obtain reporting of the logged URMC event identifying information that is triggered in accordance with at least one URMC user activity upload rule; and

[1907] provide an acknowledgment to the client upon obtaining said reporting.

[1908] 847. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[1909] obtain notification of a user initiation of a universally resolvable media content ("URMC") event in a client;

[1910] obtain an event log recording the URMC event identifying information in association with the event;

[1911] provide reporting frequency preference setting, wherein the preference setting includes at least one URMC user activity upload rule;

[1912] obtain reporting of the logged URMC event identifying information that is triggered in accordance with at least one URMC user activity upload rule; and

[1913] provide an acknowledgment to the client upon obtaining said reporting.

[1914] 848. An apparatus, comprising:

[1915] a memory;

[1916] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1917] obtain notification of a user initiation of a universally resolvable media content ("URMC") event in a client;

[1918] obtain an event log recording the URMC event identifying information in association with the event;

[1919] provide reporting frequency preference setting, wherein the preference setting includes at least one URMC user activity upload rule;

[1920] obtain reporting of the logged URMC event identifying information that is triggered in accordance with at least one URMC user activity upload rule; and

[1921] provide an acknowledgment to the client upon obtaining said reporting.

[1922] 849. A processor-implemented method, comprising:

[1923] identifying an unlicensed content item and uniquely resolving it within a universally resolvable media content ("URMC") service;

[1924] obtaining aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;

[1925] obtaining an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;

[1926] evaluating the obtained aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;

[1927] identifying a target for a license request for the uniquely resolved content item; and

[1928] sending the license request to the identified target.

[1929] 850. The method of embodiment 849, further comprising:

[1930] obtaining, from the target, a license authorizing addition of the uniquely resolved content item to the URMC catalog.

[1931] 851. The method of embodiment 850, further comprising:

[1932] obtaining lossless original media file;

[1933] licensing, encrypting and encoding the obtained media file; and

[1934] making the encoded content media file available to users of the URMC service.

[1935] 852. The method of embodiment 851, wherein the encoding includes a standard quality encoding and a mobile quality encoding.

[1936] 853. The method of embodiment 849, wherein the license request includes at least the uniquely resolved content identifying information and a request to add the content item to the URMC collection.

[1937] 854. The method of embodiment 849, wherein the URMC service user engagement metric is track play count.

[1938] 855. The method of embodiment 849, wherein the URMC service user engagement metric is at least one of: (i) share count; (ii) download count; (iii) rating; and (iv) comment count.

[1939] 856. The method of embodiment 855, wherein the URMC service user engagement metric is associated with at least one of: (i) a track; (ii) a playlist; (iii) a smart playlist; (iv) a shared playlist; (v) a magic playlist; and (vi) a shared library.

[1940] 857. The method of embodiment 849, wherein the at least one URMC license request threshold rule specifies a trigger for the license request when the aggregate URMC service user engagement metric associated with the uniquely resolved content item is greater than a percent threshold of the aggregate URMC service user engagement metric associated with the plurality of URMC content items.

[1941] 858. The method of embodiment 849, wherein the at least one URMC license request threshold rule specifies a threshold for the aggregate URMC service user engagement metrics associated with the uniquely resolved content item.

[1942] 859. The method of embodiment 849, wherein identifying the unlicensed content item and uniquely resolving it within the URMC service further comprises acoustically matching the content item with URMC items in a URMC catalog.

[1943] 860. The method of embodiment 849, wherein identifying the unlicensed content item and uniquely resolving it within the URMC service further comprises querying a URMC metadata database using metadata associated with the content item.

[1944] 861. The method of embodiment 849, further comprising:

[1945] obtaining metadata associated with the uniquely resolved content item; and

[1946] querying a URMC license database using the obtained metadata for availability of license.

[1947] 862. The method of embodiment 849, wherein the target for the license request is identified based on at least one of an acoustical fingerprint and metadata associated with the uniquely resolved content item.

[1948] 863. The method of embodiment 849, wherein the identified target is a rights clearing agency.

[1949] 864. The method of embodiment 849, wherein the identified target is one of participating licensors of the URMC service.

[1950] 865. A system, comprising:

[1951] a memory;

[1952] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[1953] identify an unlicensed content item and uniquely resolving it within a universally resolvable media content ("URMC") service;

[1954] obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;

[1955] obtain an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;

[1956] evaluate the obtained aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;

[1957] identify a target for a license request for the uniquely resolved content item; and

[1958] send the license request to the identified target.

[1959] 866. The system of embodiment 865, further comprising instructions to:

[1960] obtain, from the target, a license authorizing addition of the uniquely resolved content item to the URMC catalog.

[1961] 867. The system of embodiment 866, further comprising instructions to:

[1962] obtain lossless original media file;

[1963] license, encrypting and encoding the obtained media file; and

[1964] make the encoded content media file available to users of the URMC service.

[1965] 868. The system of embodiment 867, wherein the encoding includes a standard quality encoding and a mobile quality encoding.

[1966] 869. The system of embodiment 865, wherein the license request includes at least the uniquely resolved content identifying information and a request to add the content item to the URMC collection.

[1967] 870. The system of embodiment 865, wherein the URMC service user engagement metric is track play count.

[1968] 871. The system of embodiment 865, wherein the URMC service user engagement metric is at least one of: (i) share count; (ii) download count; (iii) rating; and (iv) comment count.

[1969] 872. The system of embodiment 871, wherein the URMC service user engagement metric is associated with at least one of: (i) a track; (ii) a playlist; (iii) a smart playlist; (iv) a shared playlist; (v) a magic playlist; and (vi) a shared library.

[1970] 873. The system of embodiment 865, wherein the at least one URMC license request threshold rule specifies a trigger for the license request when the aggregate URMC service user engagement metric associated with the uniquely resolved content item is greater than a percent threshold of the aggregate URMC service user engagement metric associated with the plurality of URMC content items.

[1971] 874. The system of embodiment 865, wherein the at least one URMC license request threshold rule specifies a threshold for the aggregate URMC service user engagement metrics associated with the uniquely resolved content item.

[1972] 875. The system of embodiment 865, wherein identifying the unlicensed content item and uniquely resolving it within the URMC service further comprises acoustically matching the content item with URMC items in a URMC catalog.

[1973] 876. The system of embodiment 865, wherein identifying the unlicensed content item and uniquely resolving it within the URMC service further comprises querying a URMC metadata database using metadata associated with the content item.

[1974] 877. The system of embodiment 865, further comprising instructions to:

[1975] obtain metadata associated with the uniquely resolved content item; and

[1976] query a URMC license database using the obtained metadata for availability of license.

[1977] 878. The system of embodiment 865, wherein the target for the license request is identified based on at least one of an acoustical fingerprint and metadata associated with the uniquely resolved content item.

[1978] 879. The system of embodiment 865, wherein the identified target is a rights clearing agency.

[1979] 880. The system of embodiment 865, wherein the identified target is one of participating licensors of the URMC service.

[1980] 881. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[1981] identify an unlicensed content item and uniquely resolving it within a universally resolvable media content ("URMC") service;

[1982] obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;

[1983] obtain an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;

[1984] evaluate the obtained aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;

[1985] identify a target for a license request for the uniquely resolved content item; and

[1986] send the license request to the identified target.

[1987] 882. The medium of embodiment 881, further comprising instructions to:

[1988] obtain, from the target, a license authorizing addition of the uniquely resolved content item to the URMC catalog.

[1989] 883. The medium of embodiment 882, further comprising instructions to:

[1990] obtain lossless original media file;

[1991] license, encrypting and encoding the obtained media file; and

[1992] make the encoded content media file available to users of the URMC service.

[1993] 884. The medium of embodiment 883, wherein the encoding includes a standard quality encoding and a mobile quality encoding.

[1994] 885. The medium of embodiment 881, wherein the license request includes at least the uniquely resolved content identifying information and a request to add the content item to the URMC collection.

[1995] 886. The medium of embodiment 881, wherein the URMC service user engagement metric is track play count.

[1996] 887. The medium of embodiment 881, wherein the URMC service user engagement metric is at least one of: (i) share count; (ii) download count; (iii) rating; and (iv) comment count.

[1997] 888. The medium of embodiment 887, wherein the URMC service user engagement metric is associated with at least one of: (i) a track; (ii) a playlist; (iii) a smart playlist; (iv) a shared playlist; (v) a magic playlist; and (vi) a shared library.

[1998] 889. The medium of embodiment 881, wherein the at least one URMC license request threshold rule specifies a trigger for the license request when the aggregate URMC service user engagement metric associated with the uniquely resolved content item is greater than a percent threshold of the aggregate URMC service user engagement metric associated with the plurality of URMC content items.

[1999] 890. The medium of embodiment 881, wherein the at least one URMC license request threshold rule specifies a threshold for the aggregate URMC service user engagement metrics associated with the uniquely resolved content item.

[2000] 891. The medium of embodiment 881, wherein identifying the unlicensed content item and uniquely resolving it within the URMC service further comprises acoustically matching the content item with URMC items in a URMC catalog.

[2001] 892. The medium of embodiment 881, wherein identifying the unlicensed content item and uniquely resolving it within the URMC service further comprises querying a URMC metadata database using metadata associated with the content item.

[2002] 893. The medium of embodiment 881, further comprising instructions to:

[2003] obtain metadata associated with the uniquely resolved content item; and

[2004] query a URMC license database using the obtained metadata for availability of license.

[2005] 894. The medium of embodiment 881, wherein the target for the license request is identified based on at least one of an acoustical fingerprint and metadata associated with the uniquely resolved content item.

[2006] 895. The medium of embodiment 881, wherein the identified target is a rights clearing agency.

[2007] 896. The medium of embodiment 881, wherein the identified target is one of participating licensors of the URMC service.

[2008] 897. An apparatus, comprising:

[2009] a memory;

[2010] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2011] identify an unlicensed content item and uniquely resolving it within a universally resolvable media content ("URMC") service;

[2012] obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;

[2013] obtain an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;

[2014] evaluate the obtained aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;

[2015] identify a target for a license request for the uniquely resolved content item; and

[2016] send the license request to the identified target.

[2017] 898. The apparatus of embodiment 897, further comprising instructions to:

[2018] obtain, from the target, a license authorizing addition of the uniquely resolved content item to the URMC catalog.

[2019] 899. The apparatus of embodiment 898, further comprising instructions to:

[2020] obtain lossless original media file;

[2021] license, encrypting and encoding the obtained media file; and

[2022] make the encoded content media file available to users of the URMC service.

[2023] 900. The apparatus of embodiment 899, wherein the encoding includes a standard quality encoding and a mobile quality encoding.

[2024] 901. The apparatus of embodiment 897, wherein the license request includes at least the uniquely resolved content identifying information and a request to add the content item to the URMC collection.

[2025] 902. The apparatus of embodiment 897, wherein the URMC service user engagement metric is track play count.

[2026] 903. The apparatus of embodiment 897, wherein the URMC service user engagement metric is at least one of: (i) share count; (ii) download count; (iii) rating; and (iv) comment count.

[2027] 904. The apparatus of embodiment 903, wherein the URMC service user engagement metric is associated with at least one of: (i) a track; (ii) a playlist; (iii) a smart playlist; (iv) a shared playlist; (v) a magic playlist; and (vi) a shared library.

[2028] 905. The apparatus of embodiment 897, wherein the at least one URMC license request threshold rule specifies a trigger for the license request when the aggregate URMC service user engagement metric associated with the uniquely resolved content item is greater than a percent threshold of the aggregate URMC service user engagement metric associated with the plurality of URMC content items.

[2029] 906. The apparatus of embodiment 897, wherein the at least one URMC license request threshold rule specifies a threshold for the aggregate URMC service user engagement metrics associated with the uniquely resolved content item.

[2030] 907. The apparatus of embodiment 897, wherein identifying the unlicensed content item and uniquely resolving it within the URMC service further comprises acoustically matching the content item with URMC items in a URMC catalog.

[2031] 908. The apparatus of embodiment 897, wherein identifying the unlicensed content item and uniquely resolving it within the URMC service further comprises querying a URMC metadata database using metadata associated with the content item.

[2032] 909. The apparatus of embodiment 897, further comprising instructions to:

[2033] obtain metadata associated with the uniquely resolved content item; and

[2034] query a URMC license database using the obtained metadata for availability of license.

[2035] 910. The apparatus of embodiment 897, wherein the target for the license request is identified based on at least one of an acoustical fingerprint and metadata associated with the uniquely resolved content item.

[2036] 911. The apparatus of embodiment 897, wherein the identified target is a rights clearing agency.

[2037] 912. The apparatus of embodiment 897, wherein the identified target is one of participating licensors of the URMC service.

[2038] 913. A processor-implemented method, comprising:

[2039] providing an unlicensed content item for identification and uniquely resolving it within a universally resolvable media content ("URMC") service;

[2040] providing an indication to obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;

[2041] providing an indication to obtain an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;

[2042] obtaining an indication of an evaluation of the aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;

[2043] obtaining an identification of a target for a license request for the uniquely resolved content item; and

[2044] obtaining an indication of sending the license request to the identified target.

[2045] 914. A system, comprising:

[2046] a memory;

[2047] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2048] provide an unlicensed content item for identification and uniquely resolving it within a universally resolvable media content ("URMC") service;

[2049] provide an indication to obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;

[2050] provide an indication to obtain an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;

[2051] obtain an indication of an evaluation of the aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;

[2052] obtain an identification of a target for a license request for the uniquely resolved content item; and

[2053] obtain an indication of sending the license request to the identified target.

[2054] 915. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[2055] provide an unlicensed content item for identification and uniquely resolving it within a universally resolvable media content ("URMC") service;

[2056] provide an indication to obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;

[2057] provide an indication to obtain an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;

[2058] obtain an indication of an evaluation of the aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;

[2059] obtain an identification of a target for a license request for the uniquely resolved content item; and

[2060] obtain an indication of sending the license request to the identified target.

[2061] 916. An apparatus, comprising:

[2062] a memory;

[2063] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2064] provide an unlicensed content item for identification and uniquely resolving it within a universally resolvable media content ("URMC") service;

[2065] provide an indication to obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;

[2066] provide an indication to obtain an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;

[2067] obtain an indication of an evaluation of the aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;

[2068] obtain an identification of a target for a license request for the uniquely resolved content item; and

[2069] obtain an indication of sending the license request to the identified target.

[2070] 917. A processor-implemented method, comprising:

[2071] detecting a plurality of universally resolvable media content ("URMC") events initiated at a client;

[2072] obtaining URMC event identifying information for each detected URMC event;

[2073] determining if each detected URMC event is reportable based on the URMC event identifying information;

[2074] determining a reporting category for each reportable URMC event based on the URMC event identifying information;

[2075] recording each reportable URMC event and the associated reporting category in an event log in the client; and

[2076] initiating reporting of the logged URMC event identifying information.

[2077] 918. The method of embodiment 917, further comprising:

[2078] obtaining reporting frequency preference setting, wherein the preference setting includes at least one URMC event reporting rule; and

[2079] initiating reporting of the logged URMC event identifying information based on the obtained reporting frequency preference setting.

[2080] 919. The method of embodiment 918, further comprising updating the client upon successful acknowledgement of said reporting by a server.

[2081] 920. The method of embodiment 917, wherein reportable events include at least one of: (i) play session time; (ii) playlist feature use; (iii) importing; (iv) sharing; (v) community; (vi) search; (vii) website; (viii) client; (ix) email; (x) marketing; (xi) user; and (xii) license and device.

[2082] 921. The method of embodiment 920, wherein each reportable event is associated with a URMC service user identifier and a date and time stamp.

[2083] 922. The method of embodiment 921, wherein determining if each detected URMC event is reportable based on the URMC event identifying information.

[2084] 923. A system, comprising:

[2085] a memory;

[2086] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2087] detect a plurality of universally resolvable media content ("URMC") events initiated at a client;

[2088] obtain URMC event identifying information for each detected URMC event;

[2089] determine if each detected URMC event is reportable based on the URMC event identifying information;

[2090] determine a reporting category for each reportable URMC event based on the URMC event identifying information;

[2091] record each reportable URMC event and the associated reporting category in an event log in the client; and

[2092] initiate reporting of the logged URMC event identifying information.

[2093] 924. The system of embodiment 923, further comprising:

[2094] obtain reporting frequency preference setting, wherein the preference setting includes at least one URMC event reporting rule; and

[2095] initiate reporting of the logged URMC event identifying information based on the obtained reporting frequency preference setting.

[2096] 925. The system of embodiment 924, further comprising updating the client upon successful acknowledgement of said reporting by a server.

[2097] 926. The system of embodiment 923, wherein reportable events include at least one of: (i) play session time; (ii) playlist feature use; (iii) importing; (iv) sharing; (v) community; (vi) search; (vii) website; (viii) client; (ix) email; (x) marketing; (xi) user; and (xii) license and device.

[2098] 927. The system of embodiment 926, wherein each reportable event is associated with a URMC service user identifier and a date and time stamp.

[2099] 928. The system of embodiment 927, wherein determining if each detected URMC event is reportable based on the URMC event identifying information.

[2100] 929. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[2101] detect a plurality of universally resolvable media content ("URMC") events initiated at a client;

[2102] obtain URMC event identifying information for each detected URMC event;

[2103] determine if each detected URMC event is reportable based on the URMC event identifying information;

[2104] determine a reporting category for each reportable URMC event based on the URMC event identifying information;

[2105] record each reportable URMC event and the associated reporting category in an event log in the client; and

[2106] initiate reporting of the logged URMC event identifying information.

[2107] 930. The medium of embodiment 929, further comprising:

[2108] obtain reporting frequency preference setting, wherein the preference setting includes at least one URMC event reporting rule; and

[2109] initiate reporting of the logged URMC event identifying information based on the obtained reporting frequency preference setting.

[2110] 931. The medium of embodiment 930, further comprising updating the client upon successful acknowledgement of said reporting by a server.

[2111] 932. The medium of embodiment 929, wherein reportable events include at least one of: (i) play session time; (ii) playlist feature use; (iii) importing; (iv) sharing; (v) community; (vi) search; (vii) website; (viii) client; (ix) email; (x) marketing; (xi) user; and (xii) license and device.

[2112] 933. The medium of embodiment 932, wherein each reportable event is associated with a URMC service user identifier and a date and time stamp.

[2113] 934. The medium of embodiment 933, wherein determining if each detected URMC event is reportable based on the URMC event identifying information.

[2114] 935. An apparatus, comprising:

[2115] a memory;

[2116] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2117] detect a plurality of universally resolvable media content ("URMC") events initiated at a client;

[2118] obtain URMC event identifying information for each detected URMC event;

[2119] determine if each detected URMC event is reportable based on the URMC event identifying information;

[2120] determine a reporting category for each reportable URMC event based on the URMC event identifying information;

[2121] record each reportable URMC event and the associated reporting category in an event log in the client; and

[2122] initiate reporting of the logged URMC event identifying information.

[2123] 936. The apparatus of embodiment 935, further comprising:

[2124] obtain reporting frequency preference setting, wherein the preference setting includes at least one URMC event reporting rule; and

[2125] initiate reporting of the logged URMC event identifying information based on the obtained reporting frequency preference setting.

[2126] 937. The apparatus of embodiment 936, further comprising updating the client upon successful acknowledgement of said reporting by a server.

[2127] 938. The apparatus of embodiment 935, wherein reportable events include at least one of: (i) play session time; (ii) playlist feature use; (iii) importing; (iv) sharing; (v) community; (vi) search; (vii) website; (viii) client; (ix) email; (x) marketing; (xi) user; and (xii) license and device.

[2128] 939. The apparatus of embodiment 938, wherein each reportable event is associated with a URMC service user identifier and a date and time stamp.

[2129] 940. The apparatus of embodiment 939, wherein determining if each detected URMC event is reportable based on the URMC event identifying information.

[2130] 941. A processor-implemented method, comprising:

[2131] providing a request to detect a plurality of universally resolvable media content ("URMC") events initiated at a client;

[2132] providing URMC event identifying information for each detected URMC event;

[2133] providing a request to determine if each detected URMC event is reportable based on the URMC event identifying information;

[2134] providing a request to determine a reporting category for each reportable URMC event based on the URMC event identifying information;

[2135] providing a request to record each reportable URMC event and the associated reporting category in an event log in the client; and

[2136] providing a request to initiate reporting of the logged URMC event identifying information.

[2137] 942. A system, comprising:

[2138] a memory;

[2139] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2140] provide a request to detect a plurality of universally resolvable media content ("URMC") events initiated at a client;

[2141] provide URMC event identifying information for each detected URMC event;

[2142] provide a request to determine if each detected URMC event is reportable based on the URMC event identifying information;

[2143] provide a request to determine a reporting category for each reportable URMC event based on the URMC event identifying information;

[2144] provide a request to record each reportable URMC event and the associated reporting category in an event log in the client; and

[2145] provide a request to initiate reporting of the logged URMC event identifying information.

[2146] 943. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[2147] provide a request to detect a plurality of universally resolvable media content ("URMC") events initiated at a client;

[2148] provide URMC event identifying information for each detected URMC event;

[2149] provide a request to determine if each detected URMC event is reportable based on the URMC event identifying information;

[2150] provide a request to determine a reporting category for each reportable URMC event based on the URMC event identifying information;

[2151] provide a request to record each reportable URMC event and the associated reporting category in an event log in the client; and

[2152] provide a request to initiate reporting of the logged URMC event identifying information.

[2153] 944. An apparatus, comprising:

[2154] a memory;

[2155] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2156] provide a request to detect a plurality of universally resolvable media content ("URMC") events initiated at a client;

[2157] provide URMC event identifying information for each detected URMC event;

[2158] provide a request to determine if each detected URMC event is reportable based on the URMC event identifying information;

[2159] provide a request to determine a reporting category for each reportable URMC event based on the URMC event identifying information;

[2160] provide a request to record each reportable URMC event and the associated reporting category in an event log in the client; and

[2161] provide a request to initiate reporting of the logged URMC event identifying information.

[2162] 945. A processor-implemented method, comprising:

[2163] receiving from a requestor a payment request for usage of a universally resolvable media content ("URMC") item;

[2164] querying a URMC usage database to obtain the URMC item usage metric;

[2165] determining, based on URMC usage payment obligation rules associated with the requestor and the URMC item usage metric, a payment amount owed for usage of the URMC item; and

[2166] providing the determined payment amount to the requestor.

[2167] 946. The method of embodiment 945, wherein the URMC usage payment obligation rules specify royally rate for a unit of the usage metric.

[2168] 947. The method of embodiment 946, wherein the royalty rate is specific to a geography.

[2169] 948. The method of embodiment 945, wherein the payment amount is device license fees based on prorated share of the URMC item usage.

[2170] 949. The method of embodiment 945, wherein the request specifies at least one criterion for selecting the URMC item.

[2171] 950. The method of embodiment 945, further comprising querying a URMC reporting database using the specified criterion to obtain the URMC item.

[2172] 951. The method of embodiment 945, wherein the URMC usage metric is play count.

[2173] 952. The method of embodiment 945, wherein the URMC usage metric is number of license activations.

[2174] 953. A system, comprising:

[2175] a memory;

[2176] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2177] receive from a requestor a payment request for usage of a universally resolvable media content ("URMC") item;

[2178] query a URMC usage database to obtain the URMC item usage metric;

[2179] determine, based on URMC usage payment obligation rules associated with the requestor and the URMC item usage metric, a payment amount owed for usage of the URMC item; and

[2180] provide the determined payment amount to the requestor.

[2181] 954. The system of embodiment 953, wherein the URMC usage payment obligation rules specify royally rate for a unit of the usage metric.

[2182] 955. The system of embodiment 954, wherein the royally rate is specific to a geography.

[2183] 956. The system of embodiment 953, wherein the payment amount is device license fees based on prorated share of the URMC item usage.

[2184] 957. The system of embodiment 953, wherein the request specifies at least one criterion for selecting the URMC item.

[2185] 958. The system of embodiment 953, further comprising instructions to query a URMC reporting database using the specified criterion to obtain the URMC item.

[2186] 959. The system of embodiment 953, wherein the URMC usage metric is play count.

[2187] 960. The system of embodiment 953, wherein the URMC usage metric is number of license activations.

[2188] 961. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[2189] receive from a requestor a payment request for usage of a universally resolvable media content ("URMC") item; query a URMC usage database to obtain the URMC item usage metric;

[2190] determine, based on URMC usage payment obligation rules associated with the requestor and the

URMC item usage metric, a payment amount owed for usage of the URMC item; and

[2191] provide the determined payment amount to the requestor.

[2192] 962. The medium of embodiment 961, wherein the URMC usage payment obligation rules specify royally rate for a unit of the usage metric.

[2193] 963. The medium of embodiment 962, wherein the royally rate is specific to a geography.

[2194] 964. The medium of embodiment 961, wherein the payment amount is device license fees based on prorated share of the URMC item usage.

[2195] 965. The medium of embodiment 961, wherein the request specifies at least one criterion for selecting the URMC item.

[2196] 966. The medium of embodiment 961, further comprising instructions to query a URMC reporting database using the specified criterion to obtain the URMC item.

[2197] 967. The medium of embodiment 961, wherein the URMC usage metric is play count.

[2198] 968. The medium of embodiment 961, wherein the URMC usage metric is number of license activations.

[2199] 969. An apparatus, comprising:

[2200] a memory;

[2201] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2202] receive from a requestor a payment request for usage of a universally resolvable media content ("URMC") item;

[2203] query a URMC usage database to obtain the URMC item usage metric;

[2204] determine, based on URMC usage payment obligation rules associated with the requestor and the URMC item usage metric, a payment amount owed for usage of the URMC item; and

[2205] provide the determined payment amount to the requestor.

[2206] 970. The apparatus of embodiment 969, wherein the URMC usage payment obligation rules specify royally rate for a unit of the usage metric.

[2207] 971. The apparatus of embodiment 970, wherein the royalty rate is specific to a geography.

[2208] 972. The apparatus of embodiment 969, wherein the payment amount is device license fees based on prorated share of the URMC item usage.

[2209] 973. The apparatus of embodiment 969, wherein the request specifies at least one criterion for selecting the URMC item.

[2210] 974. The apparatus of embodiment 969, further comprising instructions to query a URMC reporting database using the specified criterion to obtain the URMC item.

[2211] 975. The apparatus of embodiment 969, wherein the URMC usage metric is play count.

[2212] 976. The apparatus of embodiment 969, wherein the URMC usage metric is number of license activations.

[2213] 977. A processor-implemented method, comprising:

[2214] providing from a requestor a payment request for usage of a universally resolvable media content ("URMC") item;

[2215] providing a request to query a URMC usage database to obtain the URMC item usage metric;

[2216] providing a request to determine, based on URMC usage payment obligation rules associated with the requestor and the URMC item usage metric, a payment amount owed for usage of the URMC item; and

[2217] obtaining the determined payment amount to the requestor.

[2218] 978. A system, comprising:

[2219] a memory;

[2220] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2221] provide from a requestor a payment request for usage of a universally resolvable media content ("URMC") item;

[2222] provide a request to query a URMC usage database to obtain the URMC item usage metric;

[2223] provide a request to determine, based on URMC usage payment obligation rules associated with the requestor and the URMC item usage metric, a payment amount owed for usage of the URMC item; and

[2224] obtain the determined payment amount to the requestor.

[2225] 979. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[2226] provide from a requestor a payment request for usage of a universally resolvable media content ("URMC") item; provide a request to query a URMC usage database to obtain the URMC item usage metric;

[2227] provide a request to determine, based on URMC usage payment obligation rules associated with the requestor and the URMC item usage metric, a payment amount owed for usage of the URMC item; and

[2228] obtain the determined payment amount to the requestor.

[2229] 980. An apparatus, comprising:

[2230] a memory;

[2231] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2232] provide from a requestor a payment request for usage of a universally resolvable media content ("URMC") item;

[2233] provide a request to query a URMC usage database to obtain the URMC item usage metric;

[2234] provide a request to determine, based on URMC usage payment obligation rules associated with the requestor and the URMC item usage metric, a payment amount owed for usage of the URMC item; and

[2235] obtain the determined payment amount to the requestor.

[2236] 981. A processor-implemented method, comprising:

[2237] obtaining from a user of a universally resolvable media content ("URMC") service a request to purchase an unlocked URMC item;

[2238] obtaining the user's social influence metric in the service;

[2239] obtaining a purchase price associated with the URMC content item;

[2240] determining, based on the user's social influence metric, a discount;

[2241] providing the user an option to purchase the URMC content item at a purchase price reduced by the discount;

[2242] receiving from the user an indication and an authorization to purchase the URMC item;

[2243] charging an account associated with the user the purchase price reduced by the discount; and

[2244] providing the user a mechanism for unlocking the URMC item.

[2245] 982. The method of embodiment 981, wherein the unlocking mechanism is a digital rights management (DRM) free version of the URMC item provided to the client.

[2246] 983. The method of embodiment 981, wherein the unlocking mechanism is a decryption key configured to unlock the URMC item by the user's client software.

[2247] 984. The method of embodiment 981, wherein the unlocked URMC item has no rights management restrictions.

[2248] 985. The method of embodiment 981, wherein the unlocked URMC item has no encryption

[2249] 986. The method of embodiment 981, wherein the social influence metric is a number of points associated with a social activity.

[2250] 987. The method of embodiment 986, wherein the discount is triggered when the number of social activity points associated with the user exceeds a threshold.

[2251] 988. A system, comprising:

[2252] a memory;

[2253] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2254] obtain from a user of a universally resolvable media content ("URMC") service a request to purchase an unlocked URMC item;

[2255] obtain the user's social influence metric in the service;

[2256] obtain a purchase price associated with the URMC content item;

[2257] determine, based on the user's social influence metric, a discount;

[2258] provide the user an option to purchase the URMC content item at a purchase price reduced by the discount;

[2259] receive from the user an indication and an authorization to purchase the URMC item;

[2260] charge an account associated with the user the purchase price reduced by the discount; and

[2261] provide the user a mechanism for unlocking the URMC item.

[2262] 989. The system of embodiment 988, wherein the unlocking mechanism is a digital rights management (DRM) free version of the URMC item provided to the client.

[2263] 990. The system of embodiment 988, wherein the unlocking mechanism is a decryption key configured to unlock the URMC item by the user's client software.

[2264] 991. The system of embodiment 988, wherein the unlocked URMC item has no rights management restrictions.

[2265] 992. The system of embodiment 988, wherein the unlocked URMC item has no encryption

[2266] 993. The system of embodiment 988, wherein the social influence metric is a number of points associated with a social activity.

[2267] 994. The system of embodiment 993, wherein the discount is triggered when the number of social activity points associated with the user exceeds a threshold.

[2268] 995. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[2269] obtain from a user of a universally resolvable media content ("URMC") service a request to purchase an unlocked URMC item;

[2270] obtain the user's social influence metric in the service;

[2271] obtain a purchase price associated with the URMC content item;

[2272] determine, based on the user's social influence metric, a discount;

[2273] provide the user an option to purchase the URMC content item at a purchase price reduced by the discount;

[2274] receive from the user an indication and an authorization to purchase the URMC item;

[2275] charge an account associated with the user the purchase price reduced by the discount; and

[2276] provide the user a mechanism for unlocking the URMC item.

[2277] 996. The medium of embodiment 995, wherein the unlocking mechanism is a digital rights management (DRM) free version of the URMC item provided to the client.

[2278] 997. The medium of embodiment 995, wherein the unlocking mechanism is a decryption key configured to unlock the URMC item by the user's client software.

[2279] 998. The medium of embodiment 995, wherein the unlocked URMC item has no rights management restrictions.

[2280] 999. The medium of embodiment 995, wherein the unlocked URMC item has no encryption

[2281] 1000. The medium of embodiment 995, wherein the social influence metric is a number of points associated with a social activity.

[2282] 1001. The medium of embodiment 1000, wherein the discount is triggered when the number of social activity points associated with the user exceeds a threshold.

[2283] 1002. An apparatus, comprising:

[2284] a memory;

[2285] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2286] obtain from a user of a universally resolvable media content ("URMC") service a request to purchase an unlocked URMC item;

[2287] obtain the user's social influence metric in the service;

[2288] obtain a purchase price associated with the URMC content item;

[2289] determine, based on the user's social influence metric, a discount;

[2290] provide the user an option to purchase the URMC content item at a purchase price reduced by the discount;

[2291] receive from the user an indication and an authorization to purchase the URMC item;

[2292] charge an account associated with the user the purchase price reduced by the discount; and

[2293] provide the user a mechanism for unlocking the URMC item.

[2294] 1003. The apparatus of embodiment 1002, wherein the unlocking mechanism is a digital rights management (DRM) free version of the URMC item provided to the client.

[2295] 1004. The apparatus of embodiment 1002, wherein the unlocking mechanism is a decryption key configured to unlock the URMC item by the user's client software.

[2296] 1005. The apparatus of embodiment 1002, wherein the unlocked URMC item has no rights management restrictions.

[2297] 1006. The apparatus of embodiment 1002, wherein the unlocked URMC item has no encryption

[2298] 1007. The apparatus of embodiment 1002, wherein the social influence metric is a number of points associated with a social activity.

[2299] 1008. The apparatus of embodiment 1007, wherein the discount is triggered when the number of social activity points associated with the user exceeds a threshold.

[2300] 1009. A processor-implemented method, comprising:

[2301] obtaining a request to purchase an encryption free universally resolvable media content ("URMC") item from a user of the URMC service;

[2302] obtaining a purchase price associated with the content item;

[2303] determining whether the user meets a discount criteria for purchasing the encryption free content item;

[2304] calculating, based on the determining, a discounted purchase price;

[2305] providing the user an option to purchase the encryption free content item at the discounted purchase price;

[2306] receiving from the user an indication and an authorization to purchase the encryption free content item;

[2307] charging an account associated with the user the discounted purchase price; and

[2308] providing the user the encryption free content item.

[2309] 1010. The method of embodiment 1009, wherein the discount criteria includes reaching a threshold number of points via influencing activity.

[2310] 1011. The method of embodiment 1009, wherein the discount criteria includes having the content item in a playlist created by the user.

[2311] 1012. The method of embodiment 1011, wherein the purchase price is progressively discounted by an amount for every degree of separation users that add the content.

[2312] 1013. The method of embodiment 1009, wherein the discount criteria includes reaching a threshold number of plays of the content item discovered via the user's playlist or library.

[2313] 1014. The method of embodiment 1009, wherein the discount criteria includes reaching a threshold number of users playing the content item discovered via the user's playlist or library.

[2314] 1015. The method of embodiment 1009, wherein the discount criteria includes reaching a threshold number of encryption free purchases of content item.

[2315] 1016. A system, comprising:

[2316] a memory;

[2317] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2318] obtain a request to purchase an encryption free universally resolvable media content ("URMC") item from a user of the URMC service;

[2319] obtain a purchase price associated with the content item;

[2320] determine whether the user meets a discount criteria for purchasing the encryption free content item;

[2321] calculate, based on the determining, a discounted purchase price;

[2322] provide the user an option to purchase the encryption free content item at the discounted purchase price;

[2323] receive from the user an indication and an authorization to purchase the encryption free content item;

[2324] charge an account associated with the user the discounted purchase price; and

[2325] provide the user the encryption free content item.

[2326] 1017. The system of embodiment 1016, wherein the discount criteria includes reaching a threshold number of points via influencing activity.

[2327] 1018. The system of embodiment 1016, wherein the discount criteria includes having the content item in a playlist created by the user.

[2328] 1019. The system of embodiment 1018, wherein the purchase price is progressively discounted by an amount for every degree of separation users that add the content.

[2329] 1020. The system of embodiment 1016, wherein the discount criteria includes reaching a threshold number of plays of the content item discovered via the user's playlist or library.

[2330] 1021. The system of embodiment 1016, wherein the discount criteria includes reaching a threshold number of users playing the content item discovered via the user's playlist or library.

[2331] 1022. The system of embodiment 1016, wherein the discount criteria includes reaching a threshold number of encryption free purchases of content item.

[2332] 1023. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[2333] obtain a request to purchase an encryption free universally resolvable media content ("URMC") item from a user of the URMC service;

[2334] obtain a purchase price associated with the content item;

[2335] determine whether the user meets a discount criteria for purchasing the encryption free content item;

[2336] calculate, based on the determining, a discounted purchase price;

[2337] provide the user an option to purchase the encryption free content item at the discounted purchase price;

[2338] receive from the user an indication and an authorization to purchase the encryption free content item;

[2339] charge an account associated with the user the discounted purchase price; and

[2340] provide the user the encryption free content item.

[2341] 1024. The medium of embodiment 1023, wherein the discount criteria includes reaching a threshold number of points via influencing activity.

[2342] 1025. The medium of embodiment 1023, wherein the discount criteria includes having the content item in a playlist created by the user.

[2343] 1026. The medium of embodiment 1025, wherein the purchase price is progressively discounted by an amount for every degree of separation users that add the content.

[2344] 1027. The medium of embodiment 1023, wherein the discount criteria includes reaching a threshold number of plays of the content item discovered via the user's playlist or library.

[2345] 1028. The medium of embodiment 1023, wherein the discount criteria includes reaching a threshold number of users playing the content item discovered via the user's playlist or library.

[2346] 1029. The medium of embodiment 1023, wherein the discount criteria includes reaching a threshold number of encryption free purchases of content item.

[2347] 1030. An apparatus, comprising:

[2348] a memory;

[2349] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2350] obtain a request to purchase an encryption free universally resolvable media content ("URMC") item from a user of the URMC service;

[2351] obtain a purchase price associated with the content item;

[2352] determine whether the user meets a discount criteria for purchasing the encryption free content item;

[2353] calculate, based on the determining, a discounted purchase price;

[2354] provide the user an option to purchase the encryption free content item at the discounted purchase price;

[2355] receive from the user an indication and an authorization to purchase the encryption free content item;

[2356] charge an account associated with the user the discounted purchase price; and

[2357] provide the user the encryption free content item.

[2358] 1031. The apparatus of embodiment 1030, wherein the discount criteria includes reaching a threshold number of points via influencing activity.

[2359] 1032. The apparatus of embodiment 1030, wherein the discount criteria includes having the content item in a playlist created by the user.

[2360] 1033. The apparatus of embodiment 1032, wherein the purchase price is progressively discounted by an amount for every degree of separation users that add the content.

[2361] 1034. The apparatus of embodiment 1030, wherein the discount criteria includes reaching a threshold number of plays of the content item discovered via the user's playlist or library.

[2362] 1035. The apparatus of embodiment 1030, wherein the discount criteria includes reaching a threshold number of users playing the content item discovered via the user's playlist or library.

[2363] 1036. The apparatus of embodiment 1030, wherein the discount criteria includes reaching a threshold number of encryption free purchases of content item.

[2364] 1037. A processor-implemented method, comprising:

[2365] obtaining from a user of a universally resolvable media content ("URMC") service a request to purchase an unlocked URMC item;

[2366] obtaining a purchase price associated with the URMC item;

[2367] providing the user an option to purchase the URMC item at a purchase price;

[2368] receiving from the user an indication and an authorization to purchase the URMC item;

[2369] charging an account associated with the user the purchase price; and

[2370] providing the user a mechanism for unlocking the URMC item.

[2371] 1038. A system, comprising:

[2372] a memory;

[2373] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2374] obtain from a user of a universally resolvable media content ("URMC") service a request to purchase an unlocked URMC item;

[2375] obtain a purchase price associated with the URMC item;

[2376] provide the user an option to purchase the URMC item at a purchase price;

[2377] receive from the user an indication and an authorization to purchase the URMC item;

[2378] charge an account associated with the user the purchase price; and

[2379] provide the user a mechanism for unlocking the URMC item.

[2380] 1039. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[2381] obtain from a user of a universally resolvable media content ("URMC") service a request to purchase an unlocked URMC item;

[2382] obtain a purchase price associated with the URMC item;

[2383] provide the user an option to purchase the URMC item at a purchase price;

[2384] receive from the user an indication and an authorization to purchase the URMC item;

[2385] charge an account associated with the user the purchase price; and

[2386] provide the user a mechanism for unlocking the URMC item.

[2387] 1040. An apparatus, comprising:

[2388] a memory;

[2389] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2390] obtain from a user of a universally resolvable media content ("URMC") service a request to purchase an unlocked URMC item;

[2391] obtain a purchase price associated with the URMC item;

[2392] provide the user an option to purchase the URMC item at a purchase price;

[2393] receive from the user an indication and an authorization to purchase the URMC item;

[2394] charge an account associated with the user the purchase price; and

[2395] provide the user a mechanism for unlocking the URMC item.

[2396] 1041. A processor implemented method comprising:

[2397] detecting a request to engage a universally resolvable media content ("URMC") item;

[2398] obtaining an expiration date for a URMC license token associated with the URMC item;

[2399] determining, based on the obtained expiration date, whether the license token is expired;

[2400] discarding a license key associated with the expired license token;

[2401] denying the request to engage the URMC item with the associated expired license token;

[2402] providing a request for an updated token and requisite credentials for the updated token;

[2403] obtaining a response including an updated token; and

[2404] engaging the requested URMC item with an associated valid updated token.

[2405] 1042. The method of embodiment 1041, wherein the response includes a user node identifier and at least one subscription node identifier.

[2406] 1043. The method of embodiment 1042, wherein the response includes the user node to the at least one subscription node link.

[2407] 1044. The method of embodiment 1043, wherein the subscription node specifies a territory where a user's device is licensed for use.

[2408] 1045. The method of embodiment 1043, wherein the subscription node specifies a territory where the URMC item is licensed for use.

[2409] 1046. The method of embodiment 1041, wherein the response includes a user to device link having an associated expiration date.

[2410] 1047. The method of embodiment 1043, further comprising retrieving a key for territory specified by the subscription node for decrypting encryption for engaging the URMC item.

[2411] 1048. A system, comprising:

[2412] a memory;

[2413] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2414] detect a request to engage a universally resolvable media content ("URMC") item;

[2415] obtain an expiration date for a URMC license token associated with the URMC item;

[2416] determine, based on the obtained expiration date, whether the license token is expired;

[2417] discard a license key associated with the expired license token;

[2418] deny the request to engage the URMC item with the associated expired license token;

[2419] provide a request for an updated token and requisite credentials for the updated token;

[2420] obtain a response including an updated token; and

[2421] engage the requested URMC item with an associated valid updated token.

[2422] 1049. The system of embodiment 1048, wherein the response includes a user node identifier and at least one subscription node identifier.

[2423] 1050. The system of embodiment 1049, wherein the response includes the user node to the at least one subscription node link.

[2424] 1051. The system of embodiment 1050, wherein the subscription node specifies a territory where a user's device is licensed for use.

[2425] 1052. The system of embodiment 1050, wherein the subscription node specifies a territory where the URMC item is licensed for use.

[2426] 1053. The system of embodiment 1048, wherein the response includes a user to device link having an associated expiration date.

[2427] 1054. The system of embodiment 1050, further comprising instructions to retrieve a key for territory specified by the subscription node for decrypting encryption for engaging the URMC item.

[2428] 1055. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[2429] detect a request to engage a universally resolvable media content ("URMC") item;

[2430] obtain an expiration date for a URMC license token associated with the URMC item;

[2431] determine, based on the obtained expiration date, whether the license token is expired;

[2432] discard a license key associated with the expired license token;

[2433] deny the request to engage the URMC item with the associated expired license token;

[2434] provide a request for an updated token and requisite credentials for the updated token;

[2435] obtain a response including an updated token; and

[2436] engage the requested URMC item with an associated valid updated token.

[2437] 1056. The medium of embodiment 1055, wherein the response includes a user node identifier and at least one subscription node identifier.

[2438] 1057. The medium of embodiment 1056, wherein the response includes the user node to the at least one subscription node link.

[2439] 1058. The medium of embodiment 1057, wherein the subscription node specifies a territory where a user's device is licensed for use.

[2440] 1059. The medium of embodiment 1057, wherein the subscription node specifies a territory where the URMC item is licensed for use.

[2441] 1060. The medium of embodiment 1055, wherein the response includes a user to device link having an associated expiration date.

[2442] 1061. The medium of embodiment 1057, further comprising instructions to retrieve a key for territory specified by the subscription node for decrypting encryption for engaging the URMC item.

[2443] 1062. An apparatus, comprising:

[2444] a memory;

[2445] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2446] detect a request to engage a universally resolvable media content ("URMC") item;

[2447] obtain an expiration date for a URMC license token associated with the URMC item;

[2448] determine, based on the obtained expiration date, whether the license token is expired;

[2449] discard a license key associated with the expired license token;

[2450] deny the request to engage the URMC item with the associated expired license token;

[2451] provide a request for an updated token and requisite credentials for the updated token;

[2452] obtain a response including an updated token; and

[2453] engage the requested URMC item with an associated valid updated token.

[2454] 1063. The apparatus of embodiment 1062, wherein the response includes a user node identifier and at least one subscription node identifier.

[2455] 1064. The apparatus of embodiment 1063, wherein the response includes the user node to the at least one subscription node link.

[2456] 1065. The apparatus of embodiment 1064, wherein the subscription node specifies a territory where a user's device is licensed for use.

[2457] 1066. The apparatus of embodiment 1064, wherein the subscription node specifies a territory where the URMC item is licensed for use.

[2458] 1067. The apparatus of embodiment 1062, wherein the response includes a user to device link having an associated expiration date.

[2459] 1068. The apparatus of embodiment 1064, further comprising instructions to retrieve a key for territory specified by the subscription node for decrypting encryption for engaging the URMC item.

[2460] 1069. A processor-implemented method, comprising:

[2461] obtaining a request for a license token for decrypting a universally resolvable media content ("URMC") item for a user of the URMC service;

[2462] determining whether an issued link between the user and at least one device associated with the request is valid;

[2463] obtaining confirmation of at least one instance of content usage reporting for the issued link from the device;

[2464] determining, based on the valid issued link and the obtained confirmation,

[2465] an expiration date for a new link;

[2466] generating and sending the requested license token for the new link to the device; and

[2467] issuing and sending the new link to the device.

[2468] 1070. A system, comprising:

[2469] a memory;

[2470] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2471] obtain a request for a license token for decrypting a universally resolvable media content ("URMC") item for a user of the URMC service;

[2472] determine whether an issued link between the user and at least one device associated with the request is valid;

[2473] obtain confirmation of at least one instance of content usage reporting for the issued link from the device;

[2474] determine, based on the valid issued link and the obtained confirmation, an expiration date for a new link;

[2475] generate and send the requested license token for the new link to the device; and

[2476] issue and send the new link to the device.

[2477] 1071. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:

[2478] obtain a request for a license token for decrypting a universally resolvable media content ("URMC") item for a user of the URMC service;

[2479] determine whether an issued link between the user and at least one device associated with the request is valid;

[2480] obtain confirmation of at least one instance of content usage reporting for the issued link from the device;

[2481] determine, based on the valid issued link and the obtained confirmation, an expiration date for a new link;

[2482] generate and send the requested license token for the new link to the device; and

[2483] issue and send the new link to the device.

[2484] 1072. An apparatus, comprising:

[2485] a memory;

[2486] a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:

[2487] obtain a request for a license token for decrypting a universally resolvable media content ("URMC") item for a user of the URMC service;

[2488] determine whether an issued link between the user and at least one device associated with the request is valid;

[2489] obtain confirmation of at least one instance of content usage reporting for the issued link from the device;

[2490] determine, based on the valid issued link and the obtained confirmation, an expiration date for a new link;

[2491] generate and send the requested license token for the new link to the device; and

[2492] issue and send the new link to the device.

[2493] In order to address various issues and advance the art, the entirety of this application for CONTENT LICENSE ACQUISITION PLATFORM APPARATUSES, METHODS AND SYSTEMS (including the Cover Page, Title, Headings, Field, Background, Summary, Brief Description of the Drawings, Detailed Description, Claims, Abstract, FIGUREs, Appendices, and otherwise) shows, by way of illustration, various embodiments in which the claimed innovations may be practiced. The advantages and features of the application are of a representative sample of embodiments only, and are not exhaustive and/or exclusive. They are presented only to assist in understanding and teach the claimed principles. It should be understood that they are not representative of all claimed innovations. As such, certain aspects of the disclosure have not been discussed herein. That alternate embodiments may not have been presented for a specific portion of the innovations or that further undescribed alternate embodi-

ments may be available for a portion is not to be considered a disclaimer of those alternate embodiments. It will be appreciated that many of those undescribed embodiments incorporate the same principles of the innovations and others are equivalent. Thus, it is to be understood that other embodiments may be utilized and functional, logical, operational, organizational, structural and/or topological modifications may be made without departing from the scope and/or spirit of the disclosure. As such, all examples and/or embodiments are deemed to be non-limiting throughout this disclosure. Also, no inference should be drawn regarding those embodiments discussed herein relative to those not discussed herein other than it is as such for purposes of reducing space and repetition. For instance, it is to be understood that the logical and/or topological structure of any combination of any program components (a component collection), other components and/or any present feature sets as described in the figures and/or throughout are not limited to a fixed operating order and/or arrangement, but rather, any disclosed order is exemplary and all equivalents, regardless of order, are contemplated by the disclosure. Furthermore, it is to be understood that such features are not limited to serial execution, but rather, any number of threads, processes, services, servers, and/or the like that may execute asynchronously, concurrently, in parallel, simultaneously, synchronously, and/or the like are contemplated by the disclosure. As such, some of these features may be mutually contradictory, in that they cannot be simultaneously present in a single embodiment. Similarly, some features are applicable to one aspect of the innovations, and inapplicable to others. In addition, the disclosure includes other innovations not presently claimed. Applicant reserves all rights in those presently unclaimed innovations including the right to claim such innovations, file additional applications, continuations, continuations in part, divisions, and/or the like thereof. As such, it should be understood that advantages, embodiments, examples, functional, features, logical, operational, organizational, structural, topological, and/or other aspects of the disclosure are not to be considered limitations on the disclosure as defined by the claims or limitations on equivalents to the claims. It is to be understood that, depending on the particular needs and/or characteristics of a CLAP individual and/or enterprise user, database configuration and/or relational model, data type, data transmission and/or network framework, syntax structure, and/or the like, various embodiments of the CLAP, may be implemented that enable a great deal of flexibility and customization. For example, aspects of the CLAP may be adapted for p2p music discovery and delivery platform. While various embodiments and discussions of the CLAP have been directed to multimedia applications, however, it is to be understood that the embodiments described herein may be readily configured and/or customized for a wide variety of other applications and/or implementations.

What is claimed is:

1. A processor-implemented method, comprising:

identifying an unlicensed content item and uniquely resolving it within a universally resolvable media content ("URMC") service;

obtaining aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;

obtaining an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;

evaluating the obtained aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;

identifying a target for a license request for the uniquely resolved content item; and

sending the license request to the identified target.

2. The method of claim **1**, further comprising:

obtaining, from the target, a license authorizing addition of the uniquely resolved content item to the URMC catalog.

3. The method of claim **2**, further comprising:

obtaining lossless original media file;

licensing, encrypting and encoding the obtained media file; and

making the encoded content media file available to users of the URMC service.

4. The method of claim **3**, wherein the encoding includes a standard quality encoding and a mobile quality encoding.

5. The method of claim **1**, wherein the license request includes at least the uniquely resolved content identifying information and a request to add the content item to the URMC collection.

6. The method of claim **1**, wherein the URMC service user engagement metric is track play count.

7. The method of claim **1**, wherein the URMC service user engagement metric is at least one of: (i) share count; (ii) download count; (iii) rating; and (iv) comment count.

8. The method of claim **7**, wherein the URMC service user engagement metric is associated with at least one of: (i) a track; (ii) a playlist; (iii) a smart playlist; (iv) a shared playlist; (v) a magic playlist; and (vi) a shared library.

9. The method of claim **1**, wherein the at least one URMC license request threshold rule specifies a trigger for the license request when the aggregate URMC service user engagement metric associated with the uniquely resolved content item is greater than a percent threshold of the aggregate URMC service user engagement metric associated with the plurality of URMC content items.

10. The method of claim **1**, wherein the at least one URMC license request threshold rule specifies a threshold for the aggregate URMC service user engagement metrics associated with the uniquely resolved content item.

11. The method of claim **1**, wherein identifying the unlicensed content item and uniquely resolving it within the URMC service further comprises acoustically matching the content item with URMC items in a URMC catalog.

12. The method of claim **1**, wherein identifying the unlicensed content item and uniquely resolving it within the URMC service further comprises querying a URMC metadata database using metadata associated with the content item.

13. The method of claim **1**, further comprising:

obtaining metadata associated with the uniquely resolved content item; and

querying a URMC license database using the obtained metadata for availability of license.

14. The method of claim **1**, wherein the target for the license request is identified based on at least one of an acoustical fingerprint and metadata associated with the uniquely resolved content item.

15. The method of claim **1**, wherein the identified target is a rights clearing agency.

16. The method of claim **1**, wherein the identified target is one of participating licensors of the URMC service.

17. A system, comprising:
a memory;
a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:
identify an unlicensed content item and uniquely resolving it within a universally resolvable media content ("URMC") service;
obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;
obtain an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;
evaluate the obtained aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;
identify a target for a license request for the uniquely resolved content item; and
send the license request to the identified target.

18. A processor-readable medium storing processor-issuable instructions, wherein the processor issues instructions to:
identify an unlicensed content item and uniquely resolving it within a universally resolvable media content ("URMC") service;
obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;
obtain an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;
evaluate the obtained aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;
identify a target for a license request for the uniquely resolved content item; and
send the license request to the identified target.

19. An apparatus, comprising:
a memory;
a processor disposed in communication with said memory, and configured to issue a plurality of processing instructions stored in the memory, wherein the processor issues instructions to:
identify an unlicensed content item and uniquely resolving it within a universally resolvable media content ("URMC") service;
obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;
obtain an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;
evaluate the obtained aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;
identify a target for a license request for the uniquely resolved content item; and
send the license request to the identified target.

20. A processor-implemented method, comprising:
providing an unlicensed content item for identification and uniquely resolving it within a universally resolvable media content ("URMC") service;
providing an indication to obtain aggregate URMC service user engagement metric associated with the uniquely resolved content item during a predefined period of time;
providing an indication to obtain an aggregate URMC service user engagement metric associated with a plurality of URMC items during the predefined period of time;
obtaining an indication of an evaluation of the aggregate URMC service user engagement metrics using at least one URMC license request threshold rule;
obtaining an identification of a target for a license request for the uniquely resolved content item; and
obtaining an indication of sending the license request to the identified target.

* * * * *