SYSTEM AND METHOD FOR AUTOMATING A COMPLEX DOWNLOAD PROCESS WITH TERRITORIAL RESTRICTIONS

Abstract: A method and system to automate the process of transmission and storage of computer files and/or computer programs from one computing device to another computing device.
Published:
— without international search report and to be republished upon receipt of that report

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.
SYSTEM AND METHOD FOR AUTOMATING A COMPLEX
DOWNLOAD PROCESS WITH TERRITORIAL RESTRICTIONS

FIELD OF THE INVENTION

The present invention is related to a method and system to automate the purchase and/or electronic transmission of computer files and/or computer programs from one computing device to another computing device, and permitting and/or restricting the purchase and/or electronic transmission on a country-by-country basis.

BACKGROUND OF THE INVENTION

Important aspects of selling movies electronically via the Internet include transaction simplicity, reliability and territorial control. First, the simplification of, or automated assistance with, the process of selling and/or electronically distributing movies (or other computer files and/or computer programs) via communication means like the Internet, has increased in importance with the proliferation of the Internet for electronic distribution and file transfer. Secondly, the reliability of the communications means used by Internet customers is still a concern as the Internet grows in importance for electronic distribution and sale of movies. A popular computerized video format is the Microsoft Advanced Streaming Format, also known as ASF, and during the download process of an ASF, the communications connection could disconnect for a variety of reasons. If the communications means used by an Internet customer is disconnected, the resulting ASF that customer received could only be composed of that part of the entire ASF that was downloaded up to the point of the disconnection. Therefore, the customer could unknowingly end up with a portion of the
entire ASF they intended to download, and paid for. Due to the nature of the ASF, a partial ASF will start and playback normally and stop playing at the point when the communications means was disrupted, possibly resulting in an adverse customer service situation. Third, without specific transmission controls, movies can be electronically distributed and sold globally via the Internet without regard to geographic location. Today, the Motion Picture Industry operates under pre-existing business practices, which control where (e.g. what country) and when their movies may be sold. Use of the Internet as an electronic distribution means of movies, without use of territorial controls, may immediately cause the owner of such movies to be in violation of multiple existing movie distribution contracts on a country-by-country basis. For example, the owner of a movie may sell the distribution rights within Spain to one movie distribution company, sell the distribution rights within Sweden to another movie distribution company, and retain the movie distribution rights within the United States. If that movie owner commenced selling the movie from their website via the Internet, customers in Spain and Sweden could access the website, obtain the movie, and the movie owner could be in violation of existing movie distribution contracts. Due to the global nature of the Internet, a territorial identification and control system would be necessary to prevent certain geographic violations of existing movie distribution contracts. The sale or download of a movie to a geographic location in violation of existing movie distribution contracts can have a detrimental effect on the owner of the movie, as well as, to the business climate of the motion picture industry itself. It is the intention of the present invention to address these three issues.
SUMMARY OF THE INVENTION

The present invention automates the purchase and/or electronic transmission of movies (or other computer files and/or computer programs), provides means to re-establish a disconnected purchase and/or electronic transmission process, and supports the global distribution of movies on a country-by-country controlled basis.

Because distribution rights to a particular movie sometime permit and/or restrict the sale of that movie to certain countries, the present invention utilizes combination of methods and systems to effectively determine a user's geographic location and grant or deny a download or sale based on those results. Controlling the sale and/or download of a movie on a country-by-country basis by means of the present invention, certain contract violations of movie distribution contracts may be avoided. Additionally, the Office of Foreign Assets Control of the U.S. Department of the Treasury administers and enforces economic and trade sanctions against targeted foreign countries, terrorism sponsoring organizations and international narcotics traffickers based on U.S. foreign policy and national security goals. Within the country code database of the present invention, a table listing embargoed countries can be modified from time-to-time to automatically deny all purchase and/or electronic transmission requests initiated from such embargoed countries, with the intention to permit the user of the present invention to maintain compliance under Title 31 of the U.S. Code of Federal Regulations as countries are added or removed from the embargo list. Furthermore, in an effort to the user of the present invention to comply with laws and regulations of foreign nations, within the country
code database of the present invention, is a table listing restrictions on the sale of movies of a certain rating, on a movie-by-movie basis, and on a country-by-country basis.

The country detection structure of the present invention consists of a variety of layers. When a potential customer initially attempts to purchase and/or electronically acquire (e.g. download a movie), the present invention determines the geographic location of the potential customer's computing device based on its Internet Protocol address ("IP address") and then the present invention either programmatically grants or denies access to that movie based upon the programmatic results. The present invention first determines the IP address of the user's computing device through standard measures (e.g. through use of Microsoft Internet Information Server). The present invention then programmatically queries a local database for the country associated with the IP address and, if found, the present invention records that country for subsequent use by the present invention during the purchase and/or electronic transmission process.

Otherwise, the present invention then performs a reverse look-up to determine what Internet domain the user is coming from (e.g. "www.example.com"). Many Internet domains include the two letter country code as a prefix (e.g. "de.example.com") or a suffix (e.g. "www.example.de"). If this is the case, the present invention then writes an entry into a local database to record their IP address and the related country code. In addition, such suffixes as ".gov" and ".mil" are unique to the United States and the present invention records them as if the were ".us" suffixes. If the present invention is not able to determine geographic information based on these results (e.g. a ".com" generic suffix is the result of the reverse DNS look-up process), the
present invention proceeds with a WhoIs query (e.g. the present invention utilizes a WhoIs server such as, but not limited to, the American Registry Internet Network (ARIN) WhoIs server, which processes queries pertaining to domain names registered to computing devices in North & South America; the European Regional Internet Registry (RIPE) WhoIs server, which processes queries pertaining to domain names registered to computing devices in Europe; and the Asia Pacific Network Information Center (APNIC) WhoIs server, which processes queries pertaining to domain names registered to computing devices in Asia and along the Pacific Rim) to find the registered information for the IP Address of the computing device of the potential customer and its related Internet domain. A WhoIs service supplies the present invention with contact and address information for an IP address or an Internet domain name, and the present invention can then programmatically extract the related country code or country name. The present invention then records the IP Address of the computing device of the potential customer and its related country code into the local database for future use by the present invention.

If the present invention is not able to determine the geographic location related to the IP address of the potential customer's computing device, the present invention informs the potential customer that their geographic location cannot determined, therefore, their request to purchase and/or electronically acquire that particular movie (or other computer files and/or computer programs) has been denied. If, however, the present invention is able to determine their geographic location, the present invention programmatically compares their country code against the restrictions for the particular movie they are trying to purchase and/or
electronically acquire and, based on those results, the present invention grants or denies their request to purchase and/or electronically acquire that particular movie (or other computer files and/or computer programs).

The present invention offers a new and improved method and system to automate the process of transmission and storage of computer files and/or computer programs from one computing device to another computing device. The present invention is a software program residing on both a serving computer and a client computer permitting both to automatically interact during the download process. The present invention enables the automation of certain steps, which would otherwise be performed by the operator of the client computer. When a customer clicks on a download button requesting the download of a computer files and/or computer programs, a request is transmitted from the client device to the serving device, which then activates the functionality of the serving side of the present invention. The serving side of the present invention replies with a web page containing programming code, which communicates with the client side of the present invention and requests systems information pertaining to the client device. The client side of the present invention transmits the systems information to the serving side of the present invention. If the systems information of the client device meets the minimum requirements established by the user of the serving side of the present invention, the download process will continue.

The serving side of the present invention replies to the client side of the present invention with a web page embedded with programming code requesting Internet Protocol ("IP") address information of the client device. The
Internet uses a domain name system ("DNS") to locate and translate domain names into Internet Protocol ("IP") addresses. A domain name is a user-friendly name for a numeric IP address. The Internet utilizes a de-centralized method to maintain and update the listing and sharing of domain names and their associated IP address. This method utilizes a plurality of computers running the domain name system, or "DNS servers" to maintain and update the shared listing of domain names and IP addresses. These DNS servers are geographically distributed throughout the Internet in a hierarchy of authority. The serving side of the present invention leverages domain name information and IP address information of the Internet to determine the country of origin of a client device. Country information is embedded in one of the two types of Top Level Domains (TLDs), the country code type. Country code domains were created for use by each individual country, as they deemed necessary. Top Level Domains assigned to individual countries are maintained by an agency of the United Nations and are identifiable by their unique codes, which are listed in ISO-3166-1. These Top Level Domains are called country-code Top Level Domains, or ccTLDs.

Then the client side of the present invention replies to the serving side of the present invention with IP address of the client computer, the serving side of the present invention programatically cross references the IP address of the client device with known IP addresses in a database within the serving side of the present invention to identify the country code corresponding to the IP address of the client device. If the database within the serving side of the present invention includes the IP address of the client device, then the country code listed in that database
is cross referenced to determine if the particular computer file and/or computer program being requested is permitted to be downloaded into the country where the client device is located. However, if that database does not include the IP address of the client device, then the serving side of the present invention utilizes DNS servers to perform a reverse DNS lookup on IP addresses. The serving side of the present invention transmits the IP address of the client device to a DNS server to perform a reverse DNS lookup. If the DNS Server replies to the serving side of the present invention with the domain name information relating to the IP address of the client device, the serving side of the present invention programmatically extracts the country code portion of the domain name information and programmatically cross references that country code information in a database within the serving side of the present invention to determine if the particular computer file and/or computer program being requested is permitted to be downloaded into the country where the client device is located. However, if the DNS server utilized lacks the domain name information related to the IP address, the serving side of the present invention also can perform a similar lookup via a third-party domain name lookup service (e.g. WhoIs.net), which provides domain name information on requested IP addresses. The serving side of the present invention transmits the IP address of the client device to a third party DNS checking service for cross-referencing. If the third party DNS checking service replies to the serving side of the present invention with the domain name information relating to the IP address of the client device, the serving side of the present invention programmatically extracts the country code portion of the domain name information and programmatically cross references that country code information in a database within the
serving side of the present invention to determine if the particular computer file and/or computer program being requested is permitted to be downloaded into the country where the client device is located. However, if the third party DNS checking service does not respond to the serving side of the present invention with domain name information relating to the IP address of the client device, then the serving side of the present invention will terminate the download process.

If the client device passes the country code check based on domain name and/or IP address information, the serving side of the present invention will transmit a request to the client side of the present invention to programmatically query the system registry of the operating system of the client device to determine if a telephone modem is operational. If the modem is operational, the client side of the present invention will programmatically query the system registry of the operating system of the client device to determine the telephone number associated with the modem, then transmit the telephone number to the serving side of the present invention. The serving side of the present invention will extract that portion of the telephone number that identifies the country of origin and cross-reference it with a database within the serving side of the present invention to identify the telephone country code of the client device. The serving side of the present invention will programmatically cross reference the telephone country code of the client device to determine if it matches the IP country code of the client device. If the telephone country code and IP country code of the client device match, then the serving side of the present invention will process the download. If the telephone country code and IP country code
of the client device do not match, then the serving side of the present invention will terminate the download.

The present invention pertains to a system to automate the process of transmitting and storing computer files and/or computer programs from one computing device to another computing device. The system comprises a serving device having access to a computer file and/or computer program and transfer it. The system comprises a connector connected to the serving device on which a computer file and/or computer program travels and to which the serving device transfers the computer file and/or computer program.

The system comprises a client device, which receives the computer file and/or computer program. The serving device is separate, apart and distinct from the client device. The system comprises a serving device, which assigns the countries and/or geographic location where the computer file and/or computer program may be downloaded. The system comprises a serving device and a client device, which interact to determine if the software running on the client device and the hardware, which comprises the client device are adequate to permit the computer file and/or computer program to function properly on the client device. The system comprises a serving device, which determines the country and/or geographic location of the client device and determines if the computer file and/or computer program may be downloaded into that country and/or geographic location, and if permitted, downloads the computer file and/or computer program to the client device through the connector. The client device receives and saves the computer file and/or computer program to storage.
The present invention pertains to a method to automate the process of transmitting and storing computer files and/or computer programs from one computing device to another computing device. The method comprises the step of assigning the countries and/or geographic locations where the computer file and/or computer program may be downloaded. The method comprises the step of selecting the computer file and/or computer program by the user of the client device through means of a client interface. The method comprises the step of programmatically querying to determine if the system configuration of the client device is adequate to permit the computer file and/or computer program to function properly on the client device. The method comprises the step of programmatically querying the operating system of a client device for the country and/or geographic location of the client device and programmatically cross referencing a database to determine if the computer file and/or computer program may be downloaded into that country and/or geographic location. The method comprises the step, if permitted, of downloading the computer file and/or computer program from the storage of the serving device to storage of the client device through the connector. The method comprises the step of transmitting a message to the user of the client device for display on the client interface that the download of the computer file and/or computer program has been completed. The method comprises the step of terminating that portion of the programmed functionality of the invention running on the serving device and the client device respectively.
BRIEF DESCRIPTION OF THE DRAWINGS

In the accompanying drawings, the preferred embodiment of the invention and preferred methods of practicing the invention are illustrated in which:

Fig. 1 is a schematic diagram of the present invention.

Fig. 2 is a computer programming flowchart of the present invention.

Fig. 3 is a computer programming flowchart of the present invention.

Fig. 4 is a computer programming flowchart of the present invention.

Fig. 5 is a computer programming flowchart of the present invention.

Fig. 6 is a computer programming flowchart of the present invention.

Fig. 7 is a computer programming flowchart of the present invention.

DETAILED DESCRIPTION

Referring now to the drawings wherein like reference numerals refer to similar or identical parts
throughout the several views, and more specifically to figure 1 thereof, there is shown a system for processing a complex
download of a File 110. The system comprises a Serving
Device 10 having access to a File 110 and possessing the
means or mechanism to transfer or download the File 110. The
system comprises a connector connected to the Serving Device
10 on which the File 110 travels and to which the Serving
Device 10 transfers the File 110. The connector can be
Communication Means 120. The system comprises a Client
Device 11, which receives the File 110. The Serving Device
10 is separate, apart and distinct from the Client Device 11.

Preferably, a Serving Module 30 is connected to a
Serving Interface 20 which enables the user of the Serving
Device 10 to input information with respect to which
countries and/or geographic location the File 110 may be
downloaded. The Serving Module 30 and the Client Module 31
preferably interact to determine if the software running on
the Client Device 11 and the hardware, which comprises the
Client Device 11, are adequate to permit the File 110 to
function properly on the Client Device 11. The Serving
Device 10 preferably determines the country and/or geographic
location of the Client Device 11 and determines if the File
110 may be downloaded into that country and/or geographic
location, and if permitted, downloads the File 110 to the
Client Device 11 through the Communications Means 120. The
Client Device 11 receives and saves the File 110 to Client Storage 101.

The present invention pertains to a method to automate a complex download of a File 110. The method comprises the steps of assigning the countries and/or geographic locations where the File 110 may be downloaded. Then there is preferably the step of selecting the File 110 by the user of the Client Device 11 through means of a Client Interface 21. Then there is preferably the step of programatically querying the Client OS 61 to determine if the system configuration of the Client Device 11 is adequate to permit the File 110 to function properly on the Client Device 11. Then there is preferably the step of programatically querying the Client OS 61 to determine the country and/or geographic location of the Client Device 11 by the Serving Module 30 and programatically cross referencing the geographic location of the Client Device 11 with the information listed in the Country Code Database 130 to determine if the File 110 may be downloaded into that country and/or geographic location. Then there is preferably the step, if permitted, of downloading the File 110 from the Serving Storage 100 of the Serving Device 10 to the Client Storage 101 of the Client Device 11 through the Communications Means 120. Then there is preferably the step of transmitting a message to the user of the Client Device 11 for display on the Client Interface 21 that the download of the File 110 has been completed. Then there is preferably
the step of terminating the programmed functionality of the Serving Module 30 and the Client Module 31.

Referring now to the drawings wherein like reference numerals refer to similar or identical parts throughout the several views, and more specifically to FIG. 1 through FIG. 7 thereof, there are shown serving module/client module 30 and 31 for invoking functionality of the Serving OS 60 and the Client OS 61 of computing devices Serving Device 10 and Client Device 11, respectively. The apparatuses 30 and 31 are connected to the Serving OS 60 and the Client OS 61 of computing devices Serving Device 10 and Client Device 11, respectively. The apparatus 30 comprises means or mechanism for invoking functionality of a Serving OS 60 of a Serving Device 10 to coordinate with the apparatus 31 to: determine if the software running on the Client Device 11 and the hardware which composes the Client Device 11 are adequate for the requested File 110 to function properly on the Client Device 11; determine if the Client Device 11 is located in a country or territory in which the File 110 may be downloaded; offer the user of the Client device the ability to: 1) select the name of the Folder 160, and which Client Storage 101, where the File 110 will be saved; or 2) accept the default name and location of the Folder 160 as determined by the user of the Serving Device 10 and as determined by the calculations of the Client Module 31 respectively; transmit a File 110 to a Client Device 11 through use of a Transceiver 70 connected to a Communication
Means 120 for storage in the Folder 160 on the Client Storage 101; and upon completion of the download of File 110, transmit a message to the user of the Client Device 11 indicating the completion of the download, 2) update the Download Database 170, and 3) terminate the functionality of the Serving Module 30 and the Client Module 31. The apparatus 31 comprises means for invoking functionality of a Client OS 61 of a Client Device 11 to coordinate with the apparatus 30 to execute the above described process.

Fig. 1 is a schematic diagram which may be used in carrying out the teachings of this invention for the purpose of automating a complex process to: enable the user of the Serving Interface 20 to input information into a plurality of database tables within a Country Code Database 130; transmit a File 110 into only those countries as selected by the user of the Serving Device 10 to a Client Device 11 through use of a Serving Transceiver 70 connected to a Communication Means 120. Fig. 1 is a schematic diagram which may also be used in carrying out the teachings of this invention for the purpose of automatically invoking functionality of the Client OS 61 of the Client Device 11 to: receive from transmission from a Serving Device 10 through use of a Client Transceiver 71 connected to a Communication Means 120 and store an electronic copy thereof in Client RAM 81; save the File 110 from Client RAM 81 to Client Storage 101; and
Fig. 2 is a computer programming flowchart which may be used in carrying out the teachings of this invention for the purpose of enabling the user of the Serving Interface 20 to input information into a plurality of database tables within a Country Code Database 130. Additionally, Fig. 2 is a computer programming flowchart which may be used in carrying out the teachings of this invention for the purpose of automatically invoking functionality of the Client Module 31 of the Client Device 11 and the Serving Module 30 of the Serving Device 10 to: programmatically query the system registry of the Client OS 61 to determine the system configuration of the Client Device 11; programmatically cross reference the system configuration of the Client Device 11 with the minimum systems requirements listed in the Minimum Requirements Database 135 to determine if the File 110 will properly function on the Client Device 11; and if the Serving Module 30 determines that the File 110 will properly function on the Client Device 11, then the Serving Module 30 proceeds with the download process as depicted on Fig. 3.; if the Serving Module 30 determines that the File 110 will not properly function on the Client Device 11, then the Serving Module 30 transmits a message to the Client Interface 21 indicating that the File 110 will not function properly on the Client Device 11 and that the File 110 will not be downloaded to the Client Device 11, then the Serving Module 30 transmits an instruction to the Client Module 31 to terminate the download process, then the Serving Module 30 terminates the download process; and
Fig. 3 is a computer programming flowchart which may be used in carrying out the teachings of this invention for the purpose of automatically invoking functionality of the Client Module 31 of the Client Device 11 and the Serving Module 30 of the Serving Device 10 to: transmit a request to the Client Module 31 to programmatically query the registry of the Client OS 61 for the IP address of the Client Device 31 and transmit back the information to the Serving Module 30; the Serving Module 30 then programmatically cross references the IP address of the Client Device 31 with known IP addresses associated with ccTLD information in the Known IP ccTLD Table 132 located in the Country Code Database 130; if the Serving Module 30 does not find ccTLD information on the IP address of the Client Device 11 in the Known IP ccTLD Table 132, then the IP address of the Client Device 11 is transmitted to a DNS Server 180 for cross referencing in a listing of domain names and their associated IP addresses and a reverse DNS lookup is performed by the DNS Server 180 and if domain name information is resolved from the input of the IP address of the Client Device 11, then that domain name information is transmitted by the DNS Server 180 to the Serving Module 30; if the DNS Server 180 does not identify domain name information on the IP address of the Client Device 11 through the reverse DNS lookup process, then the IP address of the Client Device 11 is transmitted by the Serving Module 30 to the WhoIs Server 190 for cross referencing in a database of IP addresses associated with domain name information and if a match is found, the domain name
information is transmitted by the WhoIs Server 190 to the
Serving Module 30; if the Serving Module 30 does not identify
domain name information or ccTLD information associated with
the IP address of the Client Device 11: 1) by use of the
Known IP ccTLD Table 132, 2) by use of a reverse DNS lookup
process utilizing a DNS Server 180, or 3) by use of the WhoIs
Server 190, then the Serving Module 30 transmits a message to
the Client Interface 21 indicating their country of origin
cannot be resolved and that the File 110 will not be
downloaded to the Client Device 11, then the Serving Module
30 transmits an instruction to the Client Module 31 to
terminate the download process, then the Serving Module 30
terminates the download process. If the Serving Module 30
identifies domain name information or ccTLD information
associated with the IP address of the Client Device 11: 1) by
use of the Known IP ccTLD Table 132, 2) by use of a reverse
DNS lookup process utilizing a DNS Server 180, or 3) by use
of the WhoIs Server 190, then the Serving Module 30
programmatically extracts country code information from the
country code Top Level Domain (the "ccTLD") portion of the
domain name associated with the IP address of the Client
Device 11, then the Serving Module 30 programmatically cross
references the ccTLD information within the domain name
related to the Client Device 11 with the ccTLD information
associated with the File 110 located in the File ccTLD Table
133 to determine if the File 110 may be downloaded into the
country where the Client Device 11 is located; if the Serving
Module 30 determines that the File 110 may be downloaded into
the country where the Client Device 11 is located, then the Serving Module 30 proceeds with the download process as depicted on Fig. 4.; if, however, the Serving Module 30 determines that the File 110 may not be downloaded into the country where the Client Device 11 is located, then the Serving Module 30 transmits a message to the Client Interface 21 indicating that the File 110 may not be downloaded into the country where the Client Device 11 is located, then the Serving Module 30 transmits an instruction to the Client Module 31 to terminate the download process, then the Serving Module 30 terminates the download process; and

Fig. 4 is a computer programming flowchart which may be used in carrying out the teachings of this invention for the purpose of automatically invoking functionality of the Client Module 31 of the Client Device 11 and the Serving Module 30 of the Serving Device 10 to: transmit a request to the Client Module 31 to programmatically query the registry of the Client OS 61 to determine if the Client Transceiver 71 is a telephone modem; and if the Client Transceiver 71 is a telephone modem, the Client Module 31 transmits the telephone number of the Client Transceiver 71 to the Serving Module 30; if the Client Transceiver 71 is a telephone modem, and if the Client Module 31 transmits the telephone number of the Client Transceiver 71 to the Serving Module 30, then the Serving Module 30 programmatically extracts country code information from the telephone number of the Client Transceiver 71; if the Client Transceiver 71 is a telephone modem, and if the
Client Module 31 is not able to transmit the telephone number of the Client Transceiver 71 to the Serving Module 30, then the Serving Module 30 transmits a message to the Client Interface 21 indicating that their country of origin cannot be resolved and that the File 110 will not be downloaded to the Client Device 11, then the Serving Module 30 transmits an instruction to the Client Module 31 to terminate the download process, then the Serving Module 30 terminates the download process. If the telephone number of the Client Transceiver 71 is transmitted to the Serving Module 30 by the Client Module 31, then the Serving Module 30 programmatically extracts the numeric country code portion from the telephone number of the Client Transceiver 71, and programmatically cross references the numeric country code portion of the telephone number of the Client Transceiver 71 of the Client Device 31 with known numeric country code portions of telephone numbers associated with ccTLD information in the Telephone Country Code Table 134 located in the Country Code Database 130; if the Serving Module 30 does not identify ccTLD information on the telephone number of the Client Transceiver 71 of the Client Device 11 by use of the Telephone Country Code Table 134, then the Serving Module 30 transmits a message to the Client Interface 21 indicating that their country of origin cannot be resolved and that the File 110 will not be downloaded to the Client Device 11, then the Serving Module 30 transmits an instruction to the Client Module 31 to terminate the download process, then the Serving Module 30 terminates the download process. If the Serving
Module 30 identifies ccTLD information on the telephone number of the Client Transceiver 71 of the Client Device 11 by use of the Telephone Country Code Table 134, then the Serving Module 30 programatically cross references the ccTLD information related to the telephone number of the Client Transceiver 71 of the Client Device 11 with the ccTLD information associated with the File 110 located in the File ccTLD Table 133 to determine if the File 110 may be downloaded to the Client Device 11; if the Serving Module 30 determines that the File 110 may be downloaded to the Client Device 11, then the Serving Module 30 transmits information to the Client Module 31 necessary to continue the download process (e.g. byte size of File 110, Default Folder Name 150, etc.), and the Serving Module 30 creates the Cookie File 140 on the Client Storage 101; then the Client Module 31 proceeds with the download process as depicted on Fig. 5.; if, however, the Serving Module 30 determines that the File 110 may not be downloaded into the country where the Client Device 11 is located, then the Serving Module 30 transmits a message to the Client Interface 21 indicating that the File 110 may not be downloaded into the country where the Client Device 11 is located, then the Serving Module 30 transmits an instruction to the Client Module 31 to terminate the download process, then the Serving Module 30 terminates the download process; and

Fig. 5 is a computer programming flowchart which may be used in carrying out the teachings of this invention
for the purpose of automatically invoking functionality of
the Client Module 31 of the Client Device 11 and the Serving
Module 30 of the Serving Device 10 to: transmit a message to
the user of the Client Device 11 through means of the Client
Interface 21 inquiring if the user desires to select the name
of the file folder where the File 110 is to be downloaded and
saved; if the user desires to select the name of the file
folder where the File 110 is to be downloaded and saved, then
the Client Module 31 proceeds with the download process as
depicted on Fig. 6.; if the user does not desire to select
the name of the file folder where the File 110 is to be
downloaded and saved, then the Client Module 31: 1)
programmatically queries the registry of the Client OS 61 for
available space on all system storage devices (e.g. Client
Storage 101), 2) calculates which of the storage devices has
the most available storage space, 3) instructs the Client OS
61 to create Folder 160 on the Client Storage 101 with the
most available space and name it the same as Default Folder
Name 150, 4) instructs the Client OS 61 to update Cookie File
140 with name of Folder 160 the same as the name of the
Default Folder Name 150, and resave Cookie File 140 to
Storage Device 101, 5) requests and processes the download of
File 110 from Serving Module 30 for subsequent storage in
Folder 160 on Client Storage 101; then the Client Module 31
proceeds with the download process as depicted on Fig. 7; and

Fig. 6 is a computer programming flowchart which
may be used in carrying out the teachings of this invention
for the purpose of automatically invoking functionality of
the Client Module 31 of the Client Device 11 and the Serving
Module 30 of the Serving Device 10 to: transmit a message to
the user of the Client Device 11 through means of the Client
Interface 21 requesting of the user of the Client Device 11
the name of the Folder 160 where the File 110 is to be
downloaded and saved; instruct the Client OS 61 to update the
name of the Folder 160 in the Cookie File 140 and resave the
Cookie File 140 on the Storage Device 101; request and
process the download of File 110 from Serving Module 30 for
subsequent storage in Folder 160 on Client Storage 101; then
the Client Module 31 proceeds with the download process as
depicted on Fig. 7; and

Fig. 7 is a computer programming flowchart which
may be used in carrying out the teachings of this invention
for the purpose of automatically invoking functionality of
the Client Module 31 of the Client Device 11 and the Serving
Module 30 of the Serving Device 10 to: monitor the progress
of the download of the File 110 to Folder 160 in Client
Storage 101 of the Client Device 11; upon completion of the
download process, the Client Module 31 transmits message to
the Serving Module 30 that the download has been completed;
the Serving Module 30 transmits update information to the
Download Database 170 with information that the download of
File 110 has been completed then the Serving Module 30
discontinues communication with the Client Module 31 and ends
the program; also upon download complete, the Client Module
transmits message Client Interface 21 that download has successfully been completed then the Client Module 31 discontinues communication with the Serving Module 30 and ends the program.

Referring now to FIG. 1 through FIG. 7, a preferred embodiment of the invention is comprised of the following:

- Serving Device
- Client Device
- Serving Interface
- Client Interface
- Serving Module
- Client Module
- Serving OS
- Client OS
- Serving Transceiver
- Client Transceiver
- Serving RAM
- Client RAM
- Serving Processor
- Client Processor
- Serving Storage
- Client Storage
- File
- Communication Means
- ccTLD Database
- IP ccTLD Table
132 Known IP ccTLD Table
133 File ccTLD Table
134 Telephone Country Code Table
135 Software Requirements Database
136 Minimum Requirements Table
140 Cookie File
150 Default Folder Name
151 Target Folder Name
152 Custom Folder Name
160 Folder
170 Download Database
180 DNS Server
190 WhoIs Server

In FIG. 1 through FIG. 7, the following components are already commercially available: the Serving Device 10, the Client Device 11, the Serving Interface 20, the Client Interface 21, the Serving OS 60, the Client OS 61, the Serving Transceiver 70, the Client Transceiver 71, the Serving RAM 80, the Client RAM 81, the Serving Processor 90, the Client Processor 91, the Serving Storage 100, the Client Storage 101, the File 110, the Communication Means 120, the Cookie File 140, the Folder 160, the DNS Server 180 capable of a reverse DNS lookup process, and the WhoIs Server 190. The Serving Module 30, Client Module 31, the ccTLD Database 130, the IP ccTLD Table 131, the Known IP ccTLD Table 132, the File ccTLD Table 133, the Telephone Country Code Table
134, the Default Folder Name 150, the Target Folder Name 150, the Custom Folder Name 152, and the Download Database 170 are new teachings of this invention.

The Serving Device 10 is means, which can perform computational functions (such as, but not limited to, a Web Server, PC, Mac, PalmPC, Laptop, etc.). The Serving Device 10 is also means, which can electronically communicate with other computing devices. The Serving Device 10 is also means b to transmit computer files (e.g. the File 110) to other computing devices. The Serving Device 10 is also means to transmit information and/or requests to other computing devices. The Serving Device 10 may contain a video display, audio speakers, and other computing peripherals.

The Client Device 11 is means, which can perform computational functions (such as, but not limited to, a Web Server, PC, Mac, PalmPC, Laptop, etc.). The Client Device 11 is also means is means, which can electronically communicate with other computing devices. The Client Device 11 is also means to receive computer files (e.g. the File 110) from other computing devices. The Client Device 11 is also means to transmit information and/or requests to other computing devices. The Client Device 11 may contain a video display, audio speakers, and other computing peripherals.

The Serving Interface 20 (such as, but not limited to, web server software such as the Microsoft Internet Information Server 4.0 or 5.0) is means, which can receive
and execute requests transmitted from another computing device.

The Client Interface 21 (such as, but not limited to, web browser software such as the Microsoft Internet Explorer) is means, which can transmit requests to another computing device and can display the contents of the Serving Interface 20. The Client Interface 21 is also means, which is a client program that uses the hypertext transfer protocol ("HTTP") to make requests of a plurality of serving devices (e.g. Serving Device 10) throughout the Internet on behalf of the user of the Client Device 11.

The Serving Module 30 is means to instruct the Serving OS 60, or a communication program thereof, to communicate with a Client Device 11 via Communication Means 120. The Serving Module 30 is also means to instruct the Client Module 31 to transmit queries or instructions to the Client OS 61 of the Client Device 11 during the execution of the functionality of the Serving Module 30 and the Client Module 31. The Serving Module 30 is also means to receive instructions from a Client Module 31 via Communication Means 120. The Serving Module 30 is also means to enable the user of the Serving Device 10 to establish territorial rights and/or restrictions to be associated with a File 110 thereby limiting the geographic locations where the File 110 may be downloaded by the Client Device 11. The Serving Module 30 is also means to automatically transmit a File 110 to a Client Device 11 via Communication Means 120. The Serving Module 30
is also means working in conjunction with a Client Module 31 to automatically resume the transmission of a File 110 from the Serving Device 10 to a Client Device 11 via Communication Means 120, in the event that the transmission between the Serving Device 10 and the Client Device 11 via Communication Means 120 was disconnected. The Serving Module 30 may be embodied in computer coding software (such as, but not limited to, a program authored in the computer language c++, Active Server Pages, Visual Basic, Java Script, etc.) to execute the described functions. The Serving Module 30, if embodied as computer coding software, may be embodied or operating on a computing device (e.g. web server). The Serving Module 30 residing on a computing device can electronically communicate via the Communications Means 120 with another Serving Module 30, residing on another computing device and share and/or exchange information. For example, information in the ccTLD Database 130 of one Serving Module 30 can be shared and/or exchanged through the replication functions of Microsoft SQL Server software with the ccTLD Database 130 of another Serving Module 30, thus permitting an operation of the present invention running on multiple computing devices, in multiple physical locations, which operate collectively as the Serving Module 30. Therefore, multiple Serving Modules 30, can be separate, apart and distinct from other Serving Module 30, collectively operating as the Serving Module 30 of the present invention.

The Client Module 31 is means to automatically instruct the Client OS 61, or a communication program
thereof, to communicate with a Serving Device 10 via Communication Means 120. The Client Module 31 is also means to receive instructions from a Serving Module 30 via Communication Means 120. The Client Module 31 is also means to transmit queries and/or instructions to the Client OS 61 during the execution of the functionality of the Serving Module 30 and the Client Module 31. The Client Module 31 is also means to automatically receive from transmission, a File 110 transmitted from a Serving Device 10 via a Communication Means 120 and place an electronic copy thereof in Client RAM 81. The Client Module 31 is also means working in conjunction with a Serving Module 30 to automatically resume the transmission of a File 110 from the Serving Device 10 to a Client Device 11 via Communication Means 120, in the event that the transmission between the Serving Device 10 and the Client Device 11 via Communication Means 120 was disconnected. The Client Module 31 is also means to automatically instruct the Client OS 61 of a Client Device 11 to recall a File 110 from Client RAM 81 and save an electronic copy thereof to Client Storage 101. The Client Module 31 is means to transmit a message to the Client Interface 21 of the Client Device 11 upon completion of the execution of the functionality of the Serving Module 30 and Client Module 31. The Client Module 31 may be embodied in computer coding software (such as, but not limited to, a program authored in the computer language c++, Visual Basic, Java Script, Active X Controls, etc.) to execute the functions described herein above. The Client Module 31 may
be obtained by the user of a Client Device 11 in conjunction with the user's initial request to purchase and/or electronically acquire a File 110 from the Serving Device 11, and if embodied as computer coding software, the Client Module 31 may be automatically transmitted via Communications Means 120 from Serving Module 30 to the Client Device 11 and automatically installed on the Client Device 11. If the user of the Client Module 31 initiated a request to purchase and/or electronically acquire a File 110, from a website business operation separate, apart and distinct (the "referring website") from the business operation of the Serving Module 30, the Client Module 31 can record the URL information (e.g. through use of Microsoft Internet Information Server) of the referring website, which the user of the Client Module 31 can use to associate a specific purchase and/or electronic transmission transaction in order to subsequently compensate the user of the referring website for their participation in the transaction.

The Serving OS 60 (such as, but not limited to, the Microsoft Windows 2000 Server) is means to permit computing functionality of a Serving Device 10.

The Client OS 61 (such as, but not limited to, the Microsoft Windows 2000 Professional) is means to permit computing functionality of a Client Device 11.

The Serving Transceiver 70 (such as, but not limited to, a telephone modem, cable modem, network interface card, etc.) is means to electronically send and receive
communication signals via a Communication Means 120. The Serving Transceiver 70 is means used by software and/or firmware of, or connected to, the Serving Device 10 and/or the Serving OS 60, to electronically communicate via a Communication Means 120. The Serving Transceiver 70 is connected to the Serving Device 10 and is connected to the Communication Means 120.

The Client Transceiver 71 (such as, but not limited to, a telephone modem, cable modem, network interface card, etc.) is means to electronically send and receive communication signals via a Communication Means 120. The Client Transceiver 71 is means used by software and/or firmware of, or connected to, the Client Device 11 and/or the Client OS 61, to electronically communicate via a Communication Means 120. The Client Transceiver 71 is connected to the Client Device 11 and is connected to the Communication Means 120.

The Serving RAM 80 is means used by the Serving OS 60 of a Serving Device 10 to temporarily store computer files, computer programs or other computer information (such as, but not limited to, the File 110) for use by the Serving OS 60, computer programs running on the Serving OS 60 or other computer peripheral devices of the Serving Device 10.

The Client RAM 81 is means used by the Client OS 61 of a Client Device 11 to temporarily store computer files,
computer programs or other computer information (such as, but not limited to, the File 110) for use by the Client OS 61, computer programs running on the Client OS 61 or other computer peripheral devices of the Client Device 11.

The Serving Processor 90 is means of a Serving Device 10 to electronically process instructions of the Serving OS 60, other computer programs running on the Serving OS 60 or other computer peripheral devices of the Serving Device 10. The Serving Processor 90 is also means of a Serving Device 10 to electronically process instructions of other peripheral software and/or firmware devices of the Serving Device 10.

The Client Processor 91 is means of a Client Device 11 to electronically process instructions of the Client OS 61, other computer programs running on the Client OS 61 or other computer peripheral devices of the Client Device 11. The Client Processor 91 is also means of a Client Device 11 to electronically process instructions of other peripheral software and/or firmware devices of the Client Device 11.

The Serving Storage 100 is means in, or connected to, a Serving Device 10, which can be used to electronically save an electronic copy of the digital code of a computer program or computer file (such as, but not limited to, the File 110) from Serving RAM 80 of the Serving Device 10.
The Client Storage 101 is means in, or connected to, a Client Device 11, which can be used to electronically save an electronic copy of the digital code of a computer program or computer file (such as, but not limited to, the File 110) from Client RAM 81 of the Client Device 11.

The File 110 is a computer file or a computer program (such as, but not limited to, a word processing document (such as, but not limited to, MS Word, Lotus WordPro), a spreadsheet file (such as, but not limited to, Lotus 1-2-3, MS Excel), an audio file (such as, but not limited to, ASF, WMA, MP3, WAV, AUI), a video file (such as, but not limited to, ASF, WMV, AVI, MPEG), an executable program (such as, but not limited to, EXE), etc.). The File 110 (e.g. a movie formatted in ASF) may be zipped for transmission (e.g. using WinZip software) to facilitate a resume of transmission in the event of a communications disruption during the transmission process.

The Communication Means 120 (such as, but not limited to, telephone lines, cable TV lines, coax cable, fiber optics, radio, cellular, satellite, serial cables, parallel cables, infrared communication, laser communication, universal serial bus (USB) cables, the Internet, LAN, Ethernet, network generally, etc.) is means by which computing devices connected thereto can electronically communicate. The Communication Means 120 is also means by which computing devices connected thereto can invoke standard communication protocols (such as, but not limited to, hypertext transfer protocol (HTTP); file transfer protocol (FTP); etc.) to transmit and receive signals and/or computer
programs or computer files (such as, but not limited to, the File 110). The Communication Means 120 is also means by which computing devices connected thereto can invoke encrypted communication protocols (such as, but not limited to, secure sockets layer (SSL), transport layer security (TLS), virtual private network (VPN), etc.) to transmit and receive encrypted signals. The Communication Means 120 is also means which can include a worldwide system of computer networks, or a network of networks, known as the "Internet" in which users at any one computing device can get information from any other computer device. The Communication Means 120 is connected to the Transceiver 70 and 71 of a Serving Device 10 and a Client Device 11, respectively.

The ccTLD Database 130 is a computer program, which records information in a matrix of tables. The ccTLD Database 130 may be a relational database. The ccTLD Database 130 records information such as, but not limited to, ccTLD information; telephone country code information; countries permitted/restricted for each File 110; etc. The ccTLD Database 130 is executed on the Serving Device 10.

The IP ccTLD Table 131 is a database table within the ccTLD Database 130 and contains a listing of the top-level domain country codes of uniform resource locators ("URL") by country. The IP ccTLD Table 131 can be updated by a Serving Module 30 (either directly or through information sharing and/or exchanging with other Serving Modules 30) with
information automatically or manually inputted by a user of the Serving Module 30, etc.

The Known IP ccTLD Table 132 is a database table within the ccTLD Database 130 and contains a listing of known IP addresses by country. The Known IP ccTLD Table 132 can be updated by a Serving Module 30 (either directly or through information sharing and/or exchanging with other Serving Modules 30) with information: obtained through use of the DNS Server 180; obtained through use of the WhoIs Server 190; manually inputted by a user of the Serving Module 30, etc.

The File ccTLD Table 133 is a database table within the ccTLD Database 130 and contains a listing of the countries, which a File 110 may, or may not, be downloaded. The File ccTLD Table 133 can be updated by a Serving Module 30 (either directly or through information sharing and/or exchanging with other Serving Modules 30) with information automatically or manually inputted by a user of the Serving Module 30, etc.

The Telephone Country Code Table 134 is a database table within the ccTLD Database 130 and contains a listing of the telephone country codes by country. The Telephone Country Code Table 134 can be updated by a Serving Module 30 (either directly or through information sharing and/or exchanging with other Serving Modules 30) with information automatically or manually inputted by a user of the Serving Module 30, etc.
The Minimum Requirements Database 135 is a computer program, which records information in a matrix of tables. The Minimum Requirements Database 135 may be a relational database. The Minimum Requirements Database 135 records minimum software, hardware, etc., requirements information such as, but not limited to, web browser, media player, operating system, processor, etc., necessary for a File 110 to function properly on a computing device. The Minimum Requirements Database 135 is executed on the Serving Device 10.

The Minimum Requirements Table 136 is a database table within the Minimum Requirements Database 135 and contains a listing of known hardware, software, etc. (e.g. web browser, media player, operating system, processor, etc.), necessary for a File 110 to function properly on a computing device. The Minimum Requirements Table 136 can be updated by a Serving Module 30 (either directly or through information sharing and/or exchanging with other Serving Modules 30) with information automatically or manually inputted by a user of the Serving Module 30, etc.

The Cookie File 140 is a computer file, which contains information that the client 31 creates on the Client Storage 101. The Cookie File 140 is a computer file, which the Serving Module 30 may access to input, update, and/or retrieve information. The Cookie File 140 can include information unique to each Client Module 31, thus permitting
the Serving Module 30 to uniquely identify an individual purchase and/or electronic transmission of a File 110.

The Default Folder Name 150 is the name of the Folder 160 initially specified by the user of the Serving Device 10.

The Target Folder Name 151 is the name of the Folder 160 as listed in the Cookie File 140 and may be changed by the Serving Module 30 and/or the Client Module 31 from time to time.

The Custom Folder Name 152 is the name of the Folder 160 specifically requested by the user of the Client Device 11.

The Folder 160 is the folder location on the Client Device 11 where the File 110 is to be downloaded and saved.

The Folder 160 is the folder location on the Client Device 11 where a zipped version of the File 110 can be downloaded and temporarily saved prior to being un-zipped by the Client Module 11, then subsequently saved as the File 110.

The Download Database 170 is a computer program, which records information in a matrix of tables. The Download Database 170 may be a relational database. The Download Database 170 records information such as, but not limited to, number of downloads completed for each File 110;
etc. The Download Database 170 is executed on the Serving Device 10.

The DNS Server 180, is means used to translate Internet domain names into Internet Protocol ("IP") addresses. There is a plurality of DNS Servers 180 located throughout the Internet. The DNS Server 180 is capable of performing a reverse DNS lookup to translate IP addresses into domain names.

The WhoIs Server 190 is a server running a program that provides information, via the Internet, pertaining to the owner of registered second-level domain names and can include the geographic location of the domain name. Examples of WhoIs Servers 190 include: the American Registry Internet Network (ARIN), which processes queries pertaining to domain names registered to computing devices in North & South America; the European Regional Internet Registry (RIPE), which processes queries pertaining to domain names registered to computing devices in Europe; and the Asia Pacific Network Information Center (APNIC), which processes queries pertaining to domain names registered to computing devices in Asia and along the Pacific Rim. The WhoIs Server 190 is independent of the Serving Device 10.

By means of the Serving Interface 20, the user of the Serving Device 10 saves a File 110 to Serving Storage 100 within, or connected to, the Serving Device 10. By means of the Serving Interface 20, the user of the Serving Device 10 establishes a link, connection or other form of electronic
association (such as, but not limited to, a hyperlink using hypertext transfer protocols) (a "hyperlink") between the Serving Interface 20 and the location of the File 110 in Serving Storage 100. The Transceiver 70 of the Serving Device 10 is connected to a Communication Means 120. By means of the Serving Interface 20, the user of the Serving Device 10 inputs (such as, but not limited to, inputs via a keyboard, mouse, etc.) the countries or geographic locations where the File 110 may, or may not, be downloaded through the Serving Module 30. The Serving Interface 20 residing on a computing device can electronically communicate via the Communications Means 120 with another Serving Interface 20, residing on another computing device and share and/or exchange information (e.g. information in the ccTLD Database 130, the Minimum Requirements Table 136, etc.) permitting an operation of the present invention running on multiple computing devices, in multiple physical locations, which in total, operate as the Serving Interface 20.

By means of the Client Interface 21, the user of the Client Device 11 communicates with the Serving Device 10 via the Communication Means 120. The Transceiver 71 of the Client Device 11 is connected to a Communication Means 120. By means of the Client Interface 21, the user of the Client Device 11 views the Serving Interface 20 on the video display of the Client Device 11 and identifies the hyperlink associated with the desired File 110. The user of the Client Device 11 uses the Client Interface 21 to initiate a hyperlink transmission request (such as, but not limited to,
a double mouse click on the hyperlink associated with the File 110) via the Communication Means 120 to purchase and/or electronically acquire the File 110 from the Serving Device 10. This request initiates a series of automated actions by the Serving Module 30 then by the Client Module 31.

First, the user of the Client Device 11 inputs a request through use of the Client Interface 21 for the download of a File 110, then the Client Interface 21 transmits the download request to the Serving Interface 20, then the Serving Interface 20 activates the functionality of the Serving Module 30, then the Serving Interface 20 relays the download request to the Serving Module 30.

Next, the Serving Module 30 transmits a request to the Client Module 31 requesting information on the system configuration (e.g. software type and version, hardware, etc.) of the Client Device 11, then the Client Module 31 programatically queries, through use of computer programming scripts, the registry of the Client OS 61 for the information requested by the Serving Module 30, then the Client Module 31 transmits the information relating to the system configuration of the Client Device 11 to the Serving Module 30.

Next, the Serving Module 30 programatically cross references, through use of computer programming scripts, the system configuration (e.g. software type and version, hardware, etc.) of the Client Device 11 with the listing of
minimum systems requirements located in the Minimum Requirements Table 136 in the Minimum Requirements Database 135 on the Serving Device 10 to determine if the File 110 will function properly on the Client Device 11. If the
5 Serving Module 30 successfully determines, through the steps mentioned in the above sentence, that the Client Device 11 is configured (e.g. is running the proper software type and version, is physically constructed with the necessary hardware, etc.) in a manner which will not permit proper
10 function of the File 110 on the Client Device 11, then the Serving Module 30 transmits a message to the Client Interface 21 indicating that the File 110 will not function properly on their computing device (e.g. Client Device 11) and that the File 110 will not be downloaded to their computing device
15 (e.g. Client Device 11), then the Serving Module 30 transmits an instruction to the Client Module 31 to terminate the download process, then the Serving Module 30 terminates the download process.

Next, the Serving Module 30 transmits a request to
20 the Client Module 31 requesting the IP address of the Client Device 11, then the Client Module 31 programatically queries the registry of the Client OS 61 for the information requested by the Serving Module 30, then the Client Module 31 transmits the IP address of the Client Device 11 to the
25 Serving Module 30.

Internet protocols associate IP addresses with domain names. This is known as the Domain Name Service
("DNS"). Domain name information contains "top-level" domain information. There are two types of top-level domains, generic (the "Generic TLD") and country code (the "ccTLD"). The Generic TLD is composed of a three letter generic code (e.g. ".com," ".org," ".net," ".edu," ".gov," ".mil," and ".int") and ".mil" and ".gov" are specifically reserved for use by devices physically located within the United States only. The ccTLD is composed of a two letter country code (e.g. ".de" for Germany, ".ca" for Canada).

Next, the Serving Module 30 programatically cross references the IP address of the Client Device 11 with known IP addresses in the Known IP ccTLD Table 132 located in the Country Code Database 130 to determine the ccTLD related to the IP address of the Client Device 11; if the Serving Module 30 does not find ccTLD information on the IP address of the Client Device 11 in the Known IP ccTLD Table 132, then the IP address of the Client Device 11 is transmitted to a DNS Server 180 and a reverse DNS lookup is performed by the DNS Server 180 to resolve the domain name associated with the IP address of the Client Device 11, and if domain name information is resolved, then that domain name information is transmitted by the DNS Server 180 to the Serving Module 30; if the DNS Server 180 does not resolve domain name information on the IP address of the Client Device 11 through the reverse DNS lookup process, then the IP address of the Client Device 11 is transmitted by the Serving Module 30 to the WhoIs Server 190 for cross referencing in a database of IP addresses associated with domain name information and if
a match is found, the domain name information is transmitted by the WhoIs Server 190 to the Serving Module 30; if the Serving Module 30 does not identify domain name information or ccTLD information associated with the IP address of the Client Device 11: 1) by use of the Known IP ccTLD Table 132, or 3) by use of the WhoIs Server 190, then the Serving Module 30 transmits a message to the Client Interface 21 indicating that the File 110 may not be downloaded to the country where the Client Device 11 is located, then the Serving Module 30 transmits an instruction to the Client Module 31 to terminate the download process, then the Serving Module 30 terminates the download process.

If the Serving Module 30 is successful in matching the IP address of the Client Device 11 with known IP addresses listed in the Known IP ccTLD Table 132, thus identifying the related ccTLD, then the Serving Module 30 programmatically cross references the ccTLD information related to the Client Device 11 with the ccTLD information associated with the File 110 located in the File ccTLD Table 133 to determine if the File 110 may be downloaded to the Client Device 11, then the Serving Module 30 continues with the steps listed in the next paragraph below. Otherwise, if the reverse DNS lookup process utilized by the Serving Module 30 in conjunction with the DNS Server 180, is successful in identifying ccTLD information or ".mil" or ".gov" Generic TLD information associated with the IP address of the Client Device 11, then the Serving Module 30 programmatically cross references the ccTLD information related to the Client Device
11 with the ccTLD information associated with the File 110 located in the File ccTLD Table 133 to determine if the File 110 may be downloaded to the Client Device 11, the Serving Module 30 continues with the steps listed in the next paragraph below. Otherwise, if the WhoIs process utilized by the Serving Module 30 in conjunction with the WhoIs Server 190 is successful in identifying ccTLD information or ".mil" or ".gov" Generic TLD information associated with the IP address of the Client Device 11, then the Serving Module 30 programmatically cross references the ccTLD information related to the Client Device 11 with the ccTLD information associated with the File 110 located in the File ccTLD Table 133 to determine if the File 110 may be downloaded to the Client Device 11, the Serving Module 30 continues with the steps listed in the next paragraph below.

If the Serving Module 30 successfully determines, through the steps mentioned in the above paragraph, that the File 110 may be downloaded to the Client Device 11 based on domain name information and IP address information, then the Serving Module 30 transmits a request to the Client Module 31 to programmatically query the registry of the Client OS 61 to determine if the Client Transceiver 71 is a telephone modem. If the Client Transceiver 71 is a telephone modem, the Client Module 31 transmits the telephone number of the Client Transceiver 71 to the Serving Module 30.

If the Client Module 31 successfully determines, through the steps mentioned in the above paragraph, that the
Client Transceiver 71 is a telephone modem, and if the Client Module 31 is not able to transmit the telephone number of the Client Transceiver 71 to the Serving Module 30, then the Serving Module 30 transmits a message to the Client Interface 21 indicating that their country of origin cannot be determined and that the File 110 will not be downloaded to the Client Device 11, then the Serving Module 30 transmits an instruction to the Client Module 31 to terminate the download process, then the Serving Module 30 terminates the download process.

If the telephone number of the Client Transceiver 71 is transmitted to the Serving Module 30 by the Client Module 31, then the Serving Module 30 programmatically extracts the numeric country code portion from the telephone number of the Client Transceiver 71, and programmatically cross references the numeric country code portion of the telephone number of the Client Transceiver 71 of the Client Device 31 with known numeric country code portions of telephone numbers associated with ccTLD information in the Telephone Country Code Table 134 located in the Country Code Database 130. If the Serving Module 30 does not identify numeric country code information on the telephone number of the Client Transceiver 71 of the Client Device 11 by use of the Telephone Country Code Table 134, then the Serving Module 30 transmits a message to the Client Interface 21 indicating that their country of origin cannot be determined and that the File 110 will not be downloaded to the country where the
Client Device 11 is located, then the Serving Module 30 transmits an instruction to the Client Module 31 to terminate the download process, then the Serving Module 30 terminates the download process.

If the Serving Module 30 successfully determines, through the steps mentioned in the above paragraph, numeric country code information on the telephone number of the Client Transceiver 71 of the Client Device 11 by use of the Telephone Country Code Table 134, then the Serving Module 30 programmatically cross references the country code information related to the telephone number of the Client Transceiver 71 of the Client Device 11 with the ccTLD information associated with the File 110 located in the File ccTLD Table 133 to determine if the File 110 is permitted to be downloaded to the Client Device 11. If the Serving Module 30 determines that the File 110 is permitted to be downloaded to the Client Device 11, then the Serving Module 30 transmits information to the Client Module 31 necessary to continue the download process (e.g. byte size of File 110, Default Folder Name 150, etc.), and the Serving Module 30 instructs the Client Module 31 to create the Cookie File 140 on the Client Storage 101. If the Serving Module 30 determines that the File 110 is not permitted to be downloaded to the Client Device 11, then the Serving Module 30 transmits a message to the Client Interface 21 indicating that the File 110 is not permitted to be downloaded into the country where the Client Device 11 is located, then the Serving Module 30 transmits an
instruction to the Client Module 31 to terminate the download process, then the Serving Module 30 terminates the download process.

Next, the Client Module 31 transmits a message to the user of the Client Device 11 through means of the Client Interface 21 inquiring if the user desires to select the name of the Folder 160 where the File 110 is to be downloaded and saved. If the user of the Client Device 11 does not desire to select the name of the file folder where the File 110 is to be downloaded and saved, then the Client Module 31: programmatically queries the registry of the Client OS 61 for available space on all system storage devices (e.g. Client Storage 101); then calculates which of the storage devices has the most available storage space; then instructs the Client OS 61 to create Folder 160 on the Client Storage 101 with the most available space and name it the same as Default Folder Name 150; creates or updates the Cookie File 140 and renames Folder 160 to match the name of the Default Folder Name 150 and indicates which of the storage devices the Client Module 31 selected, and resaves Cookie File 140 to Storage Device 101; then requests and processes the download of File 110 from Serving Module 30 for subsequent storage in Folder 160 on Client Storage 101. However, if the user of the Client Device 11 desires to select the name of the Folder 160 where the File 110 is to be downloaded and saved, then the Client Module 31: transmits a message to the user of the Client Device 11 through means of the Client Interface 21.
requesting of the user of the Client Device 11 to input the
name of the Folder 160 and which storage device Client
Storage 101 where the File 110 is to be downloaded and saved;
updates the Cookie File 140 with the name of Folder 160 as
inputted by the user of the of the Client Device 11 and
indicate which of the storage devices the user of the Client
Device 11 selected, and resaves Cookie File 140 to Storage
Device 101; and requests and processes the download of File
110 from Serving Module 30 for subsequent storage in Folder
160 on Client Storage 101.

Next, the Client Module 31 monitors the progress of
the download of the File 110 to Folder 160 in Client Storage
101 of the Client Device 11. If the File 110 (e.g. a movie
in ASF format) was zipped (e.g. using WinZip zipping
software) and downloaded to, and saved in, the Folder 160,
then Client Module 31 programmatically initiates a automatic
un-zip process (e.g. the automatic un-zip feature of the
WinZip software) to extract, and save, the File 110 to Folder
160. Upon completion of the download process, and un-zip
process if applicable, the Client Module 31 transmits a
message to the Serving Module 30 indicating that the download
has been completed. Then the Serving Module 30 transmits
update information to the Download Database 170 indicating
that the download of File 110 has been completed, and then
the Serving Module 30 discontinues communication with the
Client Module 31 and ends the program. Also upon completion
of the download process, the Client Module 31 transmits a
message to the Client Interface 21 that the download has successfully been completed, and then the Client Module 31 discontinues communication with the Serving Module 30 and ends the program.

5    If during the download process of the desired File 110 described above, the transmission between the Serving Device 10 and the Client Device 11 via Communication Means 120 was disconnected, the Client Module 31 transmits a message to the Client Interface 21 that the download process was interrupted. Then, the user of the Client Device 11 re-establishes communication (e.g. accesses the website related to the operation of the Serving Module 30) with the Serving Device 10 via the Communication Means 120 by means of the Client Interface 21. By means of the Client Interface 21, the user of the Client Device 11 views the Serving Interface 20 on the video display of the Client Device 11 and re-identifies the hyperlink associated with the desired File 110. The user of the Client Device 11 uses the Client Interface 21 to re-initiate a hyperlink transmission request (such as, but not limited to, a double mouse click on the hyperlink associated with the desired File 110) via the Communication Means 120 to resume the download of the desired File 110 from the Serving Device 10, then the download process automatically resumes. Then the Serving Module 30 transmits update information to the Download Database 170 indicating that the download of File 110 has been completed, and then the Serving Module 30 discontinues communication
with the Client Module 31 and ends the program. Then upon completion of the download process, the Client Module 31 transmits a message to the Client Interface 21 that the download has successfully been completed, and then the Client Module 31 discontinues communication with the Serving Module 30 and ends the program.

Although the invention has been described in detail in the foregoing embodiments for the purpose of illustration, it is to be understood that such detail is solely for that purpose and that variations can be made therein by those skilled in the art without departing from the spirit and scope of the invention except as it may be described by the following claims.
WHAT IS CLAIMED IS:

1. A system to automate the process of transmission and storage of computer files and/or programs comprising:

2. A method to automate the process of transmission and storage of computer files and/or programs comprising the steps of:

   determining if a client device requesting a video digital signal from a servicing device is in a desired country;

   sending the digital video signal from the servicing device to the client device if the client device is in the desired country;

   a client device connected to the communication means for receiving a digital video signal;

   a communication means connected to the server device; and

   a server device having the video digital signal and means for determining if the client device is in a desired country in order for the digital video signal to be sent by the server device to the client device.
Fig. 1

Minimum Requirements Database (135)
  Minimum Requirements Table (136)
  coTLD Database (130)
    IP coTLD Table (131)
    Known IP coTLD Table (134)
    File coTLD Table (133)
    Telephone Country Code Table (134)
  Download Database (170)

Serving Interface (20)
  Serving Module (30)
    Operating System (60)
      File (110)
  Transceiver (70)
    RAM (80)
    Processor (90)
    Storage (100)

Client Interface (21)
  Client Module (31)
    Operating System (81)
    Storage (101)
    File (110)
      Folder 160
        Default Folder Name (130)
        Target Folder Name (131)
        Custom Folder Name (139)
  Transceiver (71)
    RAM (81)
    Processor (91)
  Client Device (11)

Communication Means (120)

WinCIS Server (160)
  Cookie File (140)
Fig. 2

Serving Interface 20

input database information

- Initial Serving Module 20

Minimum Requirements Database 126

Minimum Requirements Table 126

- yes
  - meets requirements
  - no

- request system configuration information (OS, browser, media player, processor, etc.)
  - cross reference the software information Client Device 11 with the minimum requirements listed in the Minimum Software Table 126 to determine if the software of the Client Device 11 meets the minimum software requirements

- instruct Client Module to terminate program

- transmit message to Client Interface 21 that the Client Device 11 fails to meet the minimum system requirements

- continue at next line on Fig. 3

Client Interface 21

- display message

Client Device 11

- user requests the download of a file 110

- check registry for system configuration information

- reply with system configuration information

- terminate program

Client OS 31

- registry