



(19) **United States**
(12) **Patent Application Publication**
Geldenbott et al.

(10) **Pub. No.: US 2009/0077077 A1**
(43) **Pub. Date: Mar. 19, 2009**

(54) **OPTIMAL SELECTION OF MSAG ADDRESS FOR VALID CIVIC/POSTAL ADDRESS**

Publication Classification

(76) Inventors: **Gerhard Geldenbott**, Seattle, WA (US); **Gordon John Hines**, Seattle, WA (US); **Jeffrey Thomas Martin**, Seattle, WA (US); **Abe Backus**, Shelton, WA (US)

(51) **Int. Cl.** *G06F 17/30* (2006.01)
(52) **U.S. Cl.** 707/6; 707/E17.001

Correspondence Address:
MANELLI DENISON & SELTER PLLC
7th Floor, 2000 M Street, N.W.
Washington, DC 20036-3307 (US)

(57) **ABSTRACT**

A technique and apparatus to find an optimal MSAG-valid address in real time given an E9-1-1 caller's civic/postal address. Arrival at an accurate MSAG-valid address selection moves through as many as three well defined MSAG matching steps which may be performed sequentially, or in tandem. The three disclosed MSAG matching techniques include a Simple Match of an MSAG address, a historical data match for an MSAG address, and a pinned MSAG match performed using a commercial location engine.

(21) Appl. No.: **12/232,507**
(22) Filed: **Sep. 18, 2008**

Related U.S. Application Data

(60) Provisional application No. 60/960,148, filed on Sep. 18, 2007, provisional application No. 60/960,459, filed on Oct. 1, 2007.

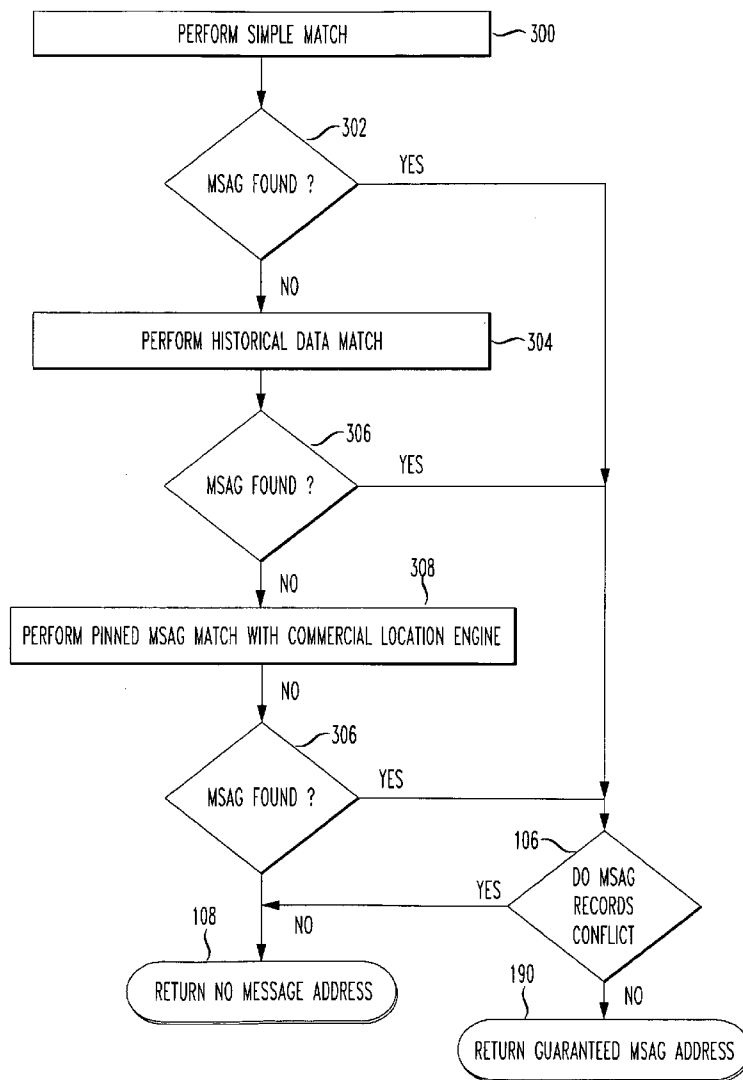
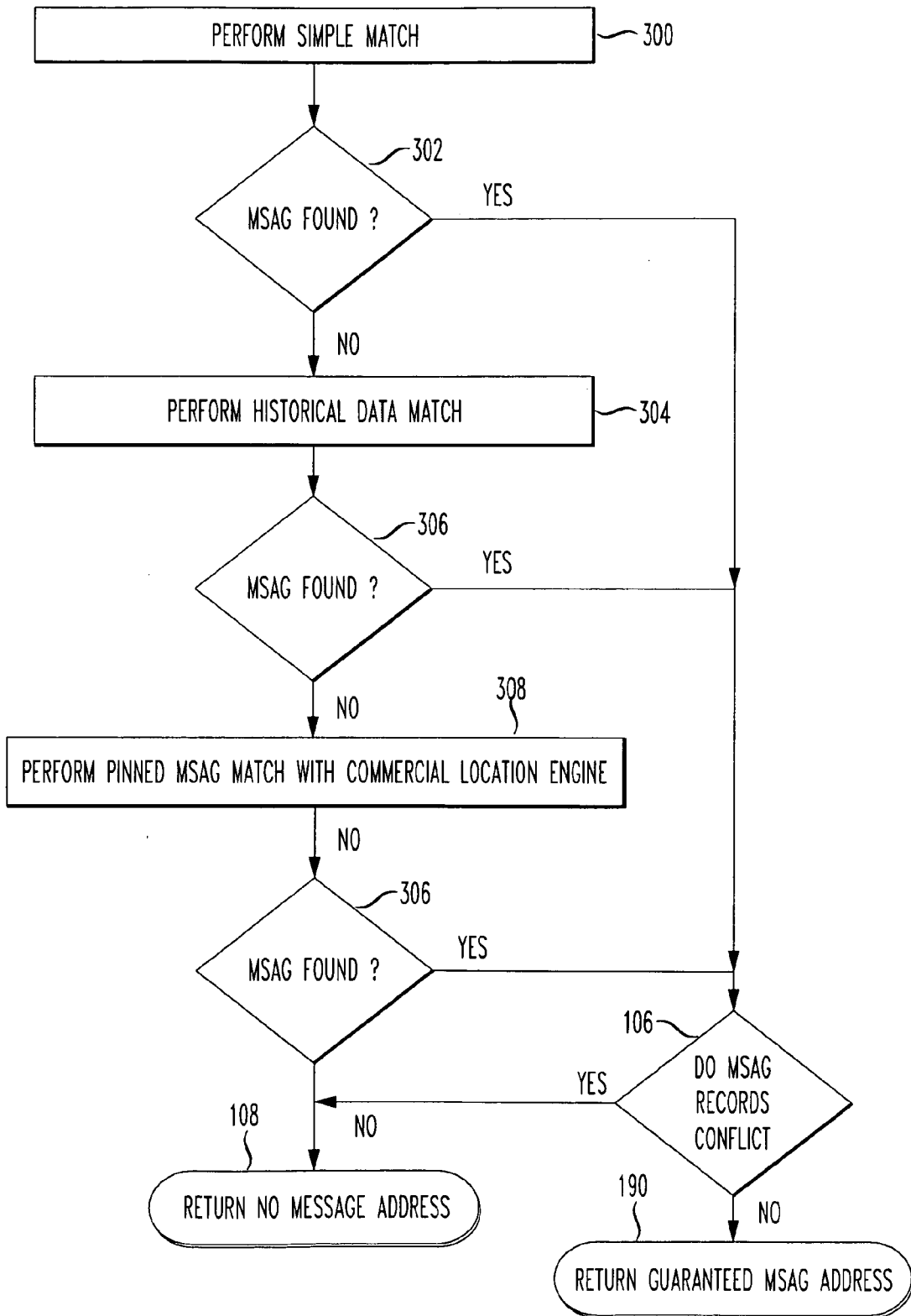


FIG. 1



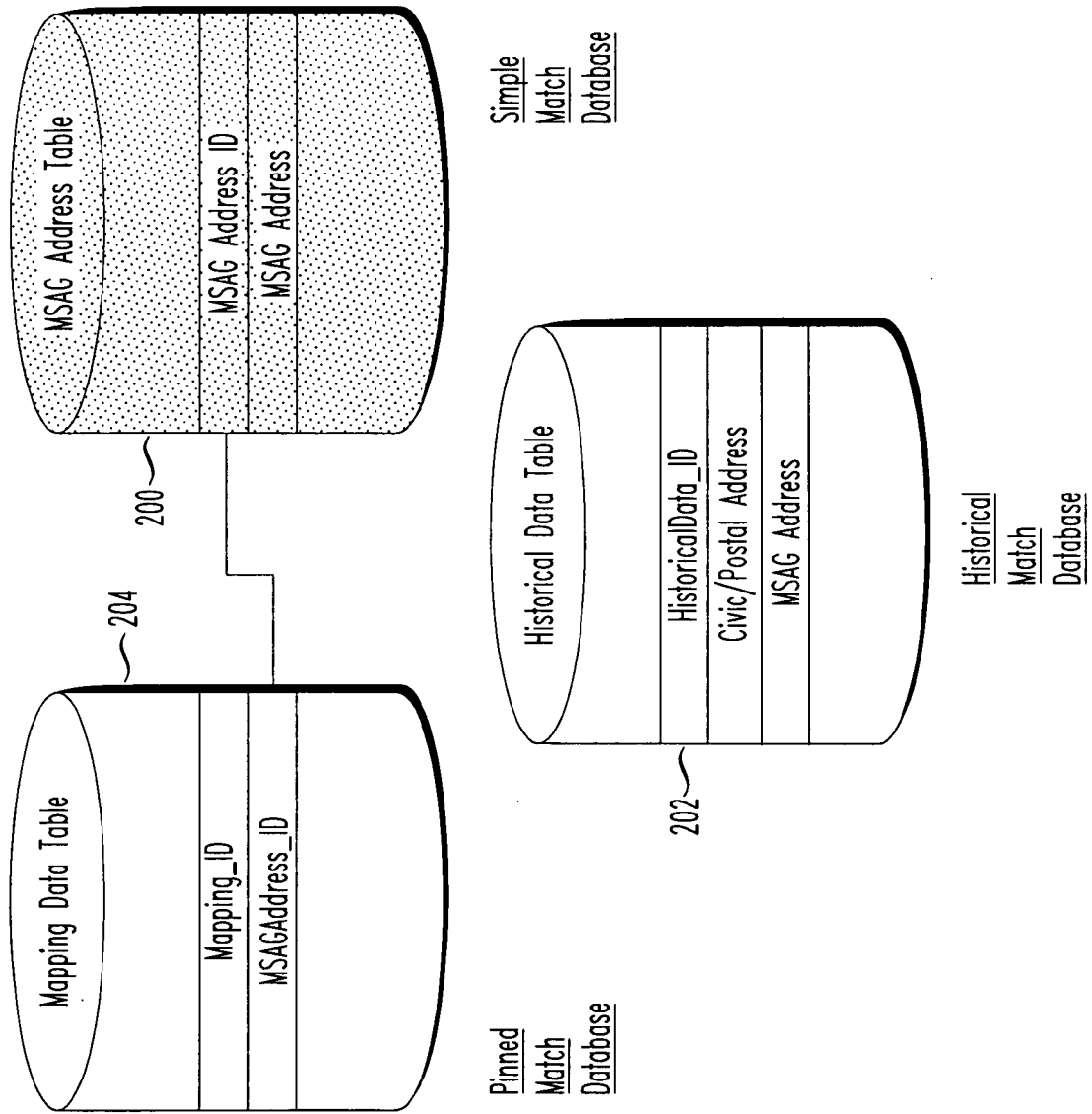


FIG. 2

FIG. 3

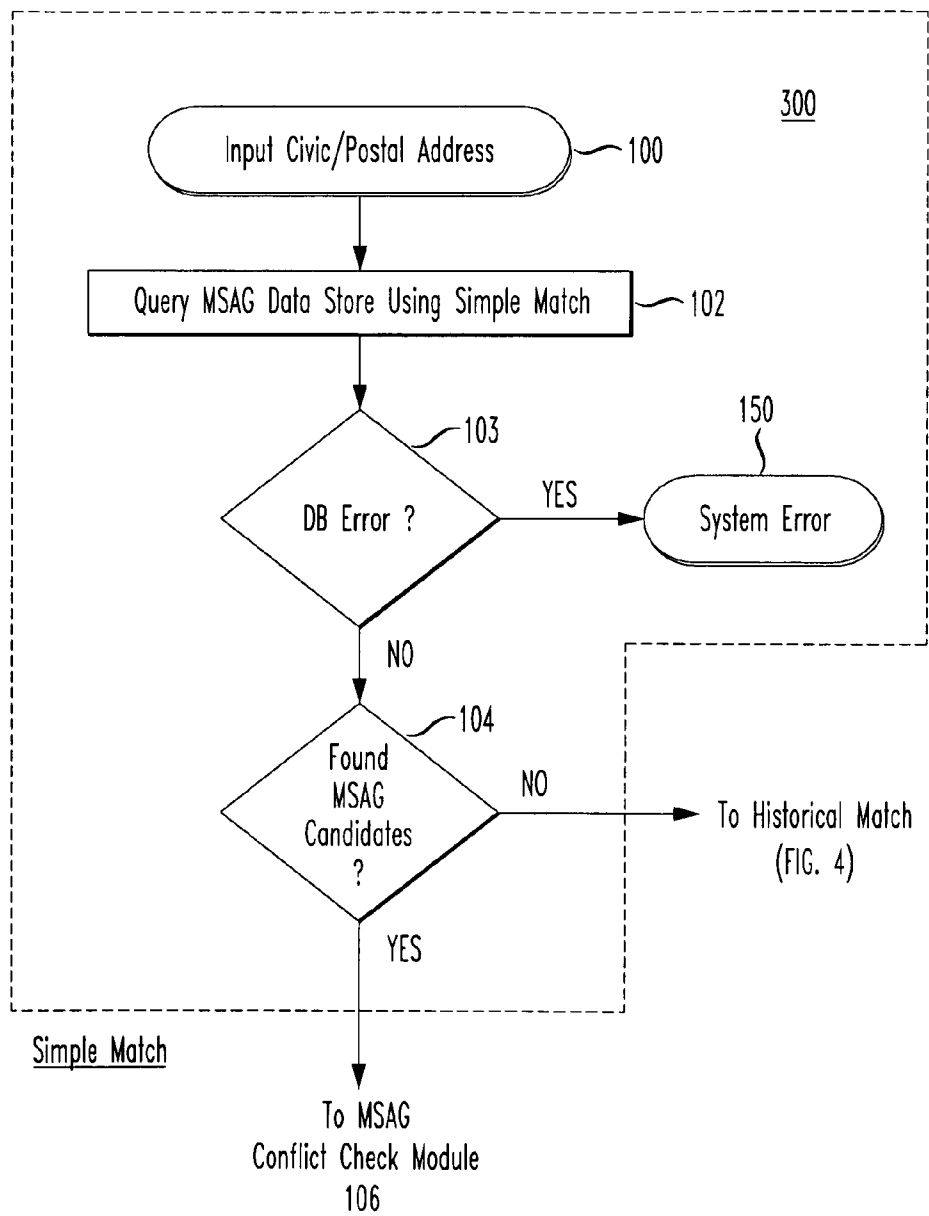


FIG. 4

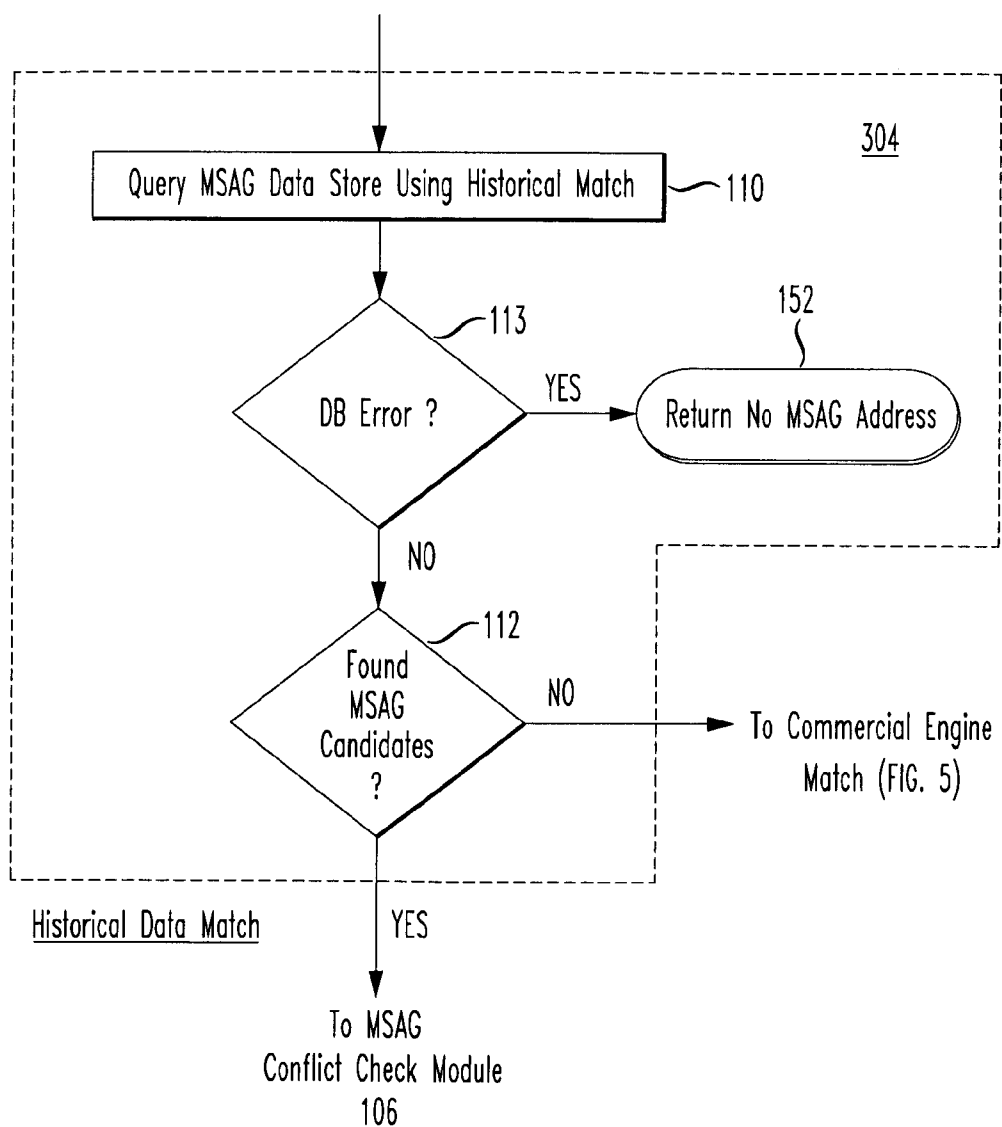
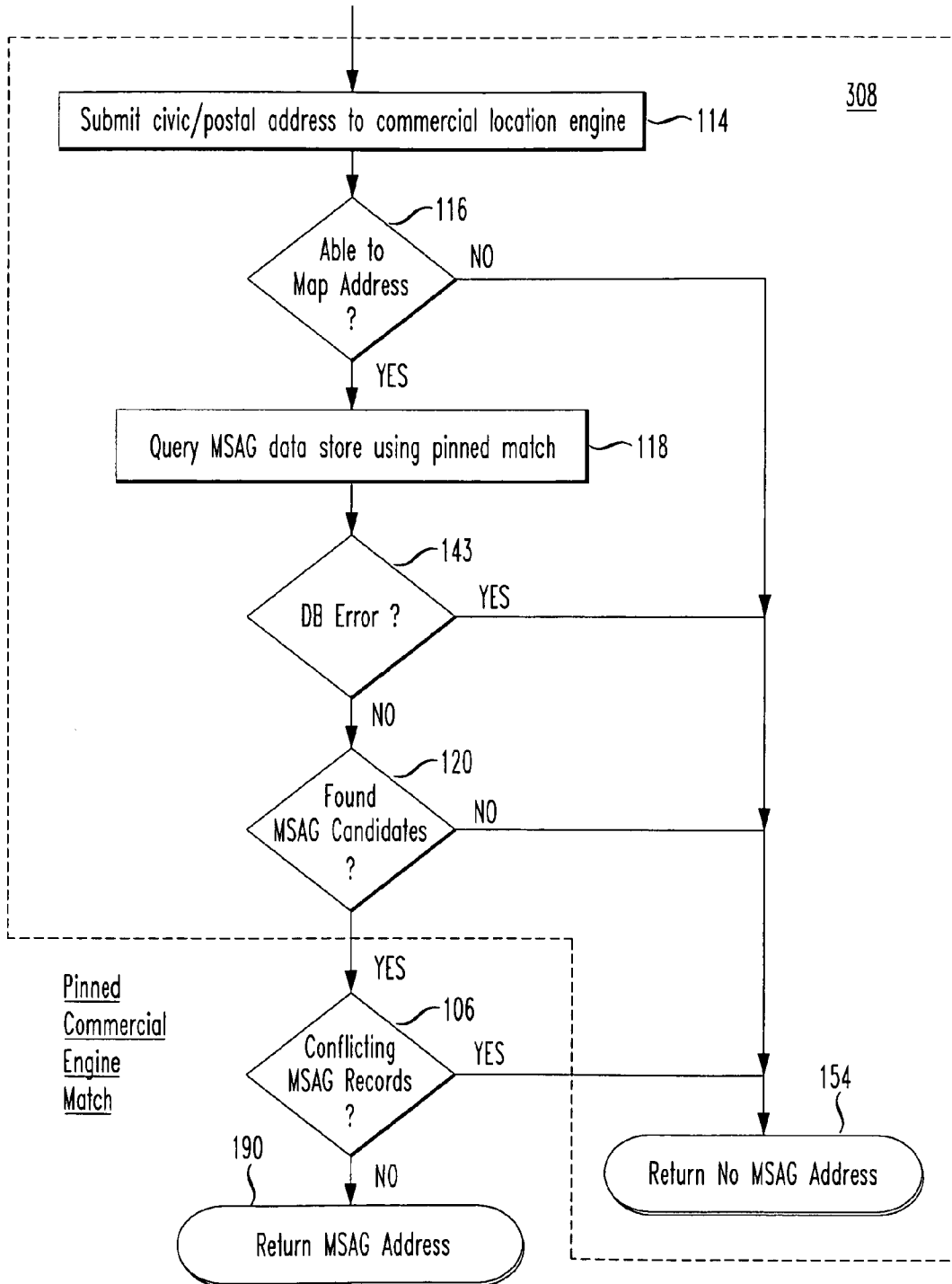


FIG. 5



OPTIMAL SELECTION OF MSAG ADDRESS FOR VALID CIVIC/POSTAL ADDRESS

[0001] This application claims priority from U.S. Provisional Application No. 60/960,148, entitled “Optimal Selection of MSAG Address for Valid Civic/Postal Address”, to Goldenbott, filed Sep. 18, 2007; and also from U.S. Provisional Application No. 60/960,459, entitled “MSAG Simple Matching a Civic/Postal Address Using Unique Normalized House Number Fields”, to Goldenbott, Hines, Martin and Backus, filed Oct. 1, 2007, the entirety of both of which are expressly incorporated herein by reference.

BACKGROUND OF THE INVENTION

[0002] 1. Field of the Invention

[0003] This invention relates generally to long distance carriers, Internet Service Providers (ISPs), and information content delivery services/providers and long distance carriers. More particularly, it relates to emergency call systems (e.g., E9-1-1) including wireless and Internet Protocol (IP) based Voice Over Internet Protocol (VoIP) emergency call systems.

[0004] 2. Background of Related Art

[0005] 9-1-1 is a phone number widely recognized in North America as an emergency phone number that is used to contact emergency dispatch personnel. Enhanced 9-1-1 (E9-1-1) is defined by an emergency call being selectively routed to an appropriate PSAP and enhanced information (callback number, name and location) is provided to the PSAP. This is accomplished through the use of the ANI. The ANI may be the real phone number of the caller (in landline E911) or a pseudo-ANI called an ESRK (in cellular E911) or ESQK (in VoIP E911). Regardless of the network type, a 9-1-1 service becomes E-9-1-1 when automatic number identification and automatic location information related to the call is provided to the 9-1-1 operator at the PSAP.

[0006] This identifier allows the PSAP to retrieve location information such as the Master Street Address Guide (MSAG) valid address of the E9-1-1 caller’s Civic/Postal Address. The Master Street Address Guide (MSAG) represents a community provided local address master guide that permits the most accurate dispatch of emergency personnel to a “correct address.” The “correct address” originates as the civic/postal address of an E9-1-1 emergency caller over a wireless and/or Internet Protocol (IP) based Voice Over Internet Protocol (VoIP) emergency call system. The civic/postal address represents the caller’s location at the time when the emergency call is placed. This civic/postal address needs to be associated with an appropriate corresponding MSAG address, which is in most cases is required by the public safety answering point (PSAP).

[0007] A Public Service Answering Point (PSAP) is a dispatch office that receives 9-1-1 calls from the public. A PSAP may be a local, fire or police department, an ambulance service or a regional office covering all services. As used herein, the term “PSAP” refers to either a public safety access point (PSAP), or to an Emergency Call Center (ECC), a VoIP term.

[0008] Distributed emergency call systems in telecommunications are in general very complex computing systems. Emergency calls that originate from a VoIP network use well proven routing paradigms already used for cellular 911 calls, or for traditional landline 911 calls. These paradigms usually work well, because VoIP customers can usually be grouped

into two categories: a mobile VoIP caller that resembles a cellular user, and a stationary VoIP user resembling landline usage.

[0009] Traditional landline systems use pre-provisioned, generally static subscriber addresses, where the landline automatic location identification (ALI) provisioning process insures a match to a master street address guide (MSAG) record, which contains an emergency service number (ESN) used to route emergency calls to a PSAP.

[0010] But determination of the location of a mobile device proves much, much more challenging. To determine location of a mobile device, some conventional cellular systems use separate triangulation technologies (or any of a number of other techniques) to find a latitude & longitude of an emergency caller. These systems then use a geographic information system (GIS) system to query for a pre-defined region (e.g., a PSAP polygon) that contains this location.

[0011] Of course, errors may occur in conventional systems when determining the location of a mobile user. But even though it’s very possible that these queried PSAP polygons can lead to a different (i.e., wrong) neighboring PSAP than an equivalent address provisioned in a landline ALI, this discrepancy is conventionally accepted by PSAPs because the location itself is likely to be imprecise due to measurement errors—sometimes the location is off by hundreds of feet.

[0012] Conventional VoIP systems use proprietary technologies, usually based on GIS polygons, or based on pre-provisioning of the caller in the traditional landline ALI long before the need for an emergency call.

[0013] Thus, traditional landline paradigms provide the most accurate location for its static users, but require the caller’s address to be pre-provisioned into a landline automatic location identifier (ALI). This pre-provisioning (often referred to as service order interface (SOI) loading) usually takes a few days between the caller notifying their service provider of their address change, and this change being reflected in the landline ALI. But during this window a 911 call might be made, and if so it would be routed using the “old” data still in the landline ALI. Even the fastest possible conventional landline ALI provisioning takes at least several hours.

[0014] Existing solutions include the NENA VoIP architecture for enhanced 9-1-1 services standard NENA 08-001. However, such conventional technologies are too complicated and not always practical. Moreover, conventional systems are disadvantageous because they are unable to handle the embedded geographic location to precisely route the caller to the correct PSAP using the “just-in-time” paradigm.

[0015] The MSAG-valid address is required by most PSAPs, as it represents a community provided local address that allows accurate dispatch of emergency personnel to the correct address. But a challenge remains to provide this MSAG-valid address for a real-time Voice Over Internet Protocol (VoIP) emergency call, given that the source address is usually civic/postal.

SUMMARY OF THE INVENTION

[0016] In accordance with the principles of the present invention, a method of determining an optimal MSAG-valid match in real time, comprises performing a master street address guide (MSAG) simple match to find a set of MSAG address candidates, performing an MSAG historical data match to find a set of MSAG address candidates, and performing a pinned MSAG data match using a commercial

location engine to find a set of MSAG address candidates. The first non-empty set of candidates is used, and the MSAG addresses in this set are checked for conflicts. An MSAG-valid address is returned when no conflicts are found.

[0017] In accordance with other aspects of the invention, apparatus for returning an optimal MSAG-valid match in real time comprises a master street address guide (MSAG) simple match module to determine a set of MSAG address candidates. An MSAG historical data match module determines a set of MSAG address candidates. A pinned MSAG data match module uses a commercial location engine to determine a set of MSAG address candidates. A comparison module compares the MSAG addresses in the first non-empty set of candidates, and returns an MSAG-valid address when no conflicts are found.

BRIEF DESCRIPTION OF THE DRAWINGS

[0018] Features and advantages of the present invention will become apparent to those skilled in the art from the following description with reference to the drawings, in which:

[0019] FIG. 1 depicts operation of three different MSAG matching steps which together, if consistent, allow determination of an optimal MSAG-valid address, in accordance with the principles of the present invention.

[0020] FIG. 2 shows an exemplary MSAG data store including a database used for a simple MSAG match, a database used for a historical data match, and a database used for a pinned match, in accordance with the principles of the present invention.

[0021] FIG. 3 shows the process within the simple MSAG match module in an exemplary MSAG address selection technique shown in FIG. 1, in more detail.

[0022] FIG. 4 shows the historical data MSAG match in an exemplary MSAG address selection module shown in FIG. 1 in more detail.

[0023] FIG. 5 shows the pinned commercial engine MSAG match in an exemplary MSAG address selection module shown in FIG. 1, in more detail.

DETAILED DESCRIPTION OF ILLUSTRATIVE EMBODIMENTS

[0024] The present invention finds the optimal Master Street Address Guide (MSAG) address for a given civic/postal address (such as the one assigned to an E9-1-1 caller).

[0025] The present invention provides a technique and apparatus to find an optimal MSAG-valid address in real time following a well-defined path that determines the accurate, i.e. optimal, MSAG address for the E9-1-1 caller. Using the inventive multi-step technique, an MSAG-valid address is selected for a given civic/postal address in such a way as to ensure that the resulting MSAG address is the most optimal address.

[0026] In accordance with the principles of the present invention, arrival at an optimal MSAG address selection moves through as many as three well defined MSAG matching steps. In the disclosed embodiments, the three MSAG matching steps are preferably performed in tandem or in parallel to allow quick, real-time MSAG address determination, though sequential functionality is within the scope of the present invention.

[0027] In the preferred embodiments, a simple match of an MSAG address is performed, a historical data match for an

MSAG address is attempted, and a pinned MSAG match is performed using a commercial location engine.

[0028] FIG. 1 depicts operation of three different MSAG matching steps which together allow determination of an optimal MSAG-valid address, in accordance with the principles of the present invention.

[0029] In particular, in a first step **300** shown in FIG. 1, a civic/postal address is Simple Matched against the relevant data store (**200** shown in FIG. 2) containing MSAG Address data. Preferably the Simple Match database **200** uses normalized fields for the highest possible success rate. In this way, in many cases the Simple Match **300** finds MSAG address candidate(s).

[0030] Step **302** determines if the Simple Match module **300** found one or more MSAG address candidates. If so, then the process provides the MSAG address candidates to a conflict check module **106** (shown in FIGS. 1 and 5) to determine if the MSAG address candidates conflict with one another. If there is a conflict between the two or more MSAG address candidates, then the process proceeds to step **108** and returns no MSAG address. If there are no conflicts between the MSAG address candidates from Simple Match **300** then that MSAG address is returned as the MSAG-valid address in step **190**.

[0031] In accordance with the principles of the present invention, if step **302** determines that no MSAG address candidates were found, a second real time step of an historical data MSAG match **304** is also performed to find MSAG address candidates. In the historical data MSAG match module **304**, the civic/postal address is compared to historical matched data in the MSAG historical address table forming the historical match database **202**.

[0032] Step **306** determines if the Historical Data Match module **304** found one or more MSAG address candidates. If so, then the process provides the MSAG address candidates to a conflict check module **106** (shown in FIGS. 1 and 5) to determine if the MSAG address candidates conflict with one another. If there is a conflict between the two or more MSAG address candidates, then the process proceeds to step **108** and returns no MSAG address. If there are no conflicts between the MSAG address candidates from Historical Data Match **304** then that MSAG address is returned as the MSAG-valid address in step **190**.

[0033] In accordance with the principles of the present invention, if step **302** determines that no MSAG address candidates were found, a final real time Pinned MSAG Match using a commercial location engine is performed in step **308**. Step **310** determines if the Pinned MSAG Match module **308** found any MSAG address candidates. If none were found, then the process proceeds to step **108** and returns no MSAG address. If however one or more MSAG address candidates were found in the Pinned MSAG Match module **308**, then the process provides the MSAG Address Candidates to a conflict check module **106** to determine if the MSAG address candidates conflict with one another. If there is a conflict between the two or more MSAG address candidates, then the process proceeds to step **108** and returns no MSAG address. If there are no conflicts between the MSAG address candidates from Pinned MSAG Match **304**, then that MSAG is returned as the MSAG-valid address in step **190**.

[0034] FIG. 2 shows an exemplary MSAG data store including a database **200** used for a simple MSAG match, a database **202** used for a historical data match, and a database

204 used for a pinned match, in accordance with the principles of the present invention.

[0035] In particular, as shown in FIG. 2, an exemplary MSAG data store includes an MSAG address table **200**, an historical data table **202**, and a mapping data table **204**.

[0036] Entries in the MSAG address table **200** preferably include an MSAG Address_ID, and an MSAG Address.

[0037] Entries in the historical data table **202** include Historical Data_ID, the civic/postal address, and the MSAG address.

[0038] Entries in the mapping data table **204** include a Mapping_ID, and the MSAG Address_ID.

[0039] FIG. 3 shows the process within the Simple MSAG Match module **300** in an exemplary MSAG address selection technique shown in FIG. 1, in more detail.

[0040] In particular, as shown in FIG. 3, the input civic/postal address **100** is normalized for case, whitespace, & punctuation, then compared (i.e., Simple Matched) to similarly normalized MSAG addresses in the MSAG address table **200**. The normalization process allows representation of equivalent address data in consistent format allowing easier comparison and a higher success rate. See, e.g., co-pending and co-owned U.S. patent application Ser. No. 12/_____, co-filed herewith and claiming priority from 60/960,148 and 60/960,459, entitled "Optimal Selection of MSAG Address For Valid Civic/Postal Address" to Geldenbott et al., the entirety of which is expressly incorporated herein by reference.

[0041] In step **102**, the appropriate MSAG data store **200** (FIG. 2, Simple Match Database) is searched using the input civic/postal address against the MSAG addresses in the table.

[0042] In step **103**, a determination is made as to whether or not there has been a database error. If so, then the process proceeds to report a system error in step **150**, and no MSAG address is returned. If there were no system errors, then the process determines in step **104** if any MSAG address candidates have been found. If candidates have been found, the process provides them to module **106** that determines if the MSAG address candidates conflict with one another. Only if no conflicting MSAG address candidates are found is the optimal MSAG-valid address returned. If conflicting candidates are found, however, the method unsuccessfully comes to end and returns no MSAG address since at this point the MSAG data store **204** is considered ambiguous, at least for the given input civic/postal address.

[0043] If there were no errors, but no MSAG address candidates were found by Simple Match **102** at step **104**, then proceed to the next MSAG matching module, e.g., to the historical data MSAG match module **304** shown in FIG. 4.

[0044] FIG. 4 shows the historical data MSAG match in an exemplary MSAG address selection module **304** shown in FIG. 1 in more detail.

[0045] In particular, as shown in FIG. 4, the civic/postal address is input, and the appropriate MSAG historical data table **202** (FIG. 2) is queried using an historical match data technique, as depicted in step **110**. The historical MSAG data store **202** may be either human or machine generated data that associates a select set of adjacent civic/postal address to their correct MSAG addresses. The same normalizations are applied to the civic/postal address as described earlier for Simple Match, and then the Historical Data Table **202** is searched using the input civic/postal address against the civic/postal addresses in the table **202**.

[0046] In step **113**, a determination is made as to whether or not there has been a database error. If so, then the process proceeds to return no MSAG address in step **152**. If no error occurred, then the process determines in step **112** if any MSAG address candidates have been found. If MSAG candidate addresses were found, the process provides them to module **106** that determines if the MSAG address candidates conflict with one another. Only if no conflicting MSAG address candidates are found is the optimal MSAG-valid address returned. If conflicting candidates are found, the method unsuccessfully comes to end and returns no MSAG address since at this point the Historical data store **202** is considered ambiguous, at least for the given input civic/postal address.

[0047] If there were no errors, but no MSAG address candidates were found by Historical Match **110** at step **112**, then proceed to the next MSAG matching module, e.g., to the pinned data MSAG match module **308** shown in FIG. 5.

[0048] FIG. 5 shows the Pinned MSAG Match using a commercial engine in an exemplary MSAG address selection module **308** shown in FIG. 1, in more detail.

[0049] In particular, as shown in FIG. 5, the civic/postal address is input, and submitted to a commercial location engine, as depicted in step **114**. If the commercial location engine submission fails to result in a mapped address, then the process proceeds to step **154** and returns no MSAG address. If, on the other hand, the commercial location engine submission does result in a mapped address, the process proceeds to a module **118** that queries the Mapping data table in the pinned match database **204**. Commercial location engines use a Mapping_ID for each street segment known to the engine. The pinned match data in **204** is a collection of MSAG addresses associated with (pinned to) these Mapping_IDs. The query in module **118** uses the Mapping_ID found for the input civic/postal from **114**, to query the database **204** for the MSAG addresses associated with (pinned to) that same Mapping_ID.

[0050] In step **143**, a determination is made as to whether or not there has been a database error. If so, then the process proceeds to return no MSAG address in step **154**. If no errors occurred, then the process determines in step **120** if any MSAG address candidates have been found. If no MSAG address candidates were found, then the process proceeds to step **154** and returns no MSAG address. If MSAG address candidates were found, the process provides them to step **106** that determines if the MSAG address candidates conflict with one another. If conflicting candidates are found, the method unsuccessfully comes to end and returns no MSAG address since at this point the Pinned Match data store **204** is considered ambiguous, at least for the given input civic/postal address. If the MSAG address candidates don't conflict, the MSAG-valid address is returned at **170**.

[0051] While the invention has been described with reference to the exemplary embodiments thereof, those skilled in the art will be able to make various modifications to the described embodiments of the invention without departing from the true spirit and scope of the invention.

1. A method of determining an optimal MSAG-valid match in real time, comprising:

- performing a master street address guide (MSAG) simple match to find a set of MSAG simple match address candidates;
- performing an MSAG historical data match to find a set of MSAG historical data address candidates;

performing a pinned MSAG data match using a commercial location engine to find a set of MSAG pinned address candidates; and

comparing MSAG addresses in a first non-empty set of said MSAG address candidates to verify they do not conflict, and return an MSAG-valid address.

2. Apparatus for determining an optimal MSAG-valid match in real time, comprising:

a master street address guide (MSAG) simple match module to determine an MSAG simple match set of MSAG simple match address candidates;

an MSAG historical data match module to determine an MSAG historical data match set of MSAG historical data address candidates;

a pinned MSAG data match module using a commercial location engine to determine a pinned MSAG match set of MSAG pinned address candidates; and

a comparison module to compare MSAG addresses in a first non-empty set of said MSAG address candidates to verify they do not conflict, and return an MSAG-valid address.

3. Apparatus for determining an optimal MSAG-valid match in real time, comprising:

means for performing a master street address guide (MSAG) simple match to find a set of MSAG simple match address candidates;

means for performing an MSAG historical data match to find a set of MSAG historical data address candidates;

means for performing a pinned MSAG data match using a commercial location engine to find a set of MSAG pinned address candidates; and

means for comparing MSAG addresses in a first non-empty set of said MSAG address candidates to verify they do not conflict, and return an MSAG-valid address.

* * * * *