US 20190342297A1

(19) **United States**

(12) **Patent Application Publication**

Adjaoute

(10) **Pub. No.: US 2019/0342297 A1**

(43) **Pub. Date: Nov. 7, 2019**

(54) **SECURING INTERNET-OF-THINGS WITH SMART-AGENT TECHNOLOGY**

(71) Applicant: **Brighterion, Inc.**, San Francisco, CA (US)

(72) Inventor: **Akli Adjaoute**, Mill Valley, CA (US)

(21) Appl. No.: **15/968,568**

(22) Filed: **May 1, 2018**

Publication Classification

(51) **Int. Cl.**

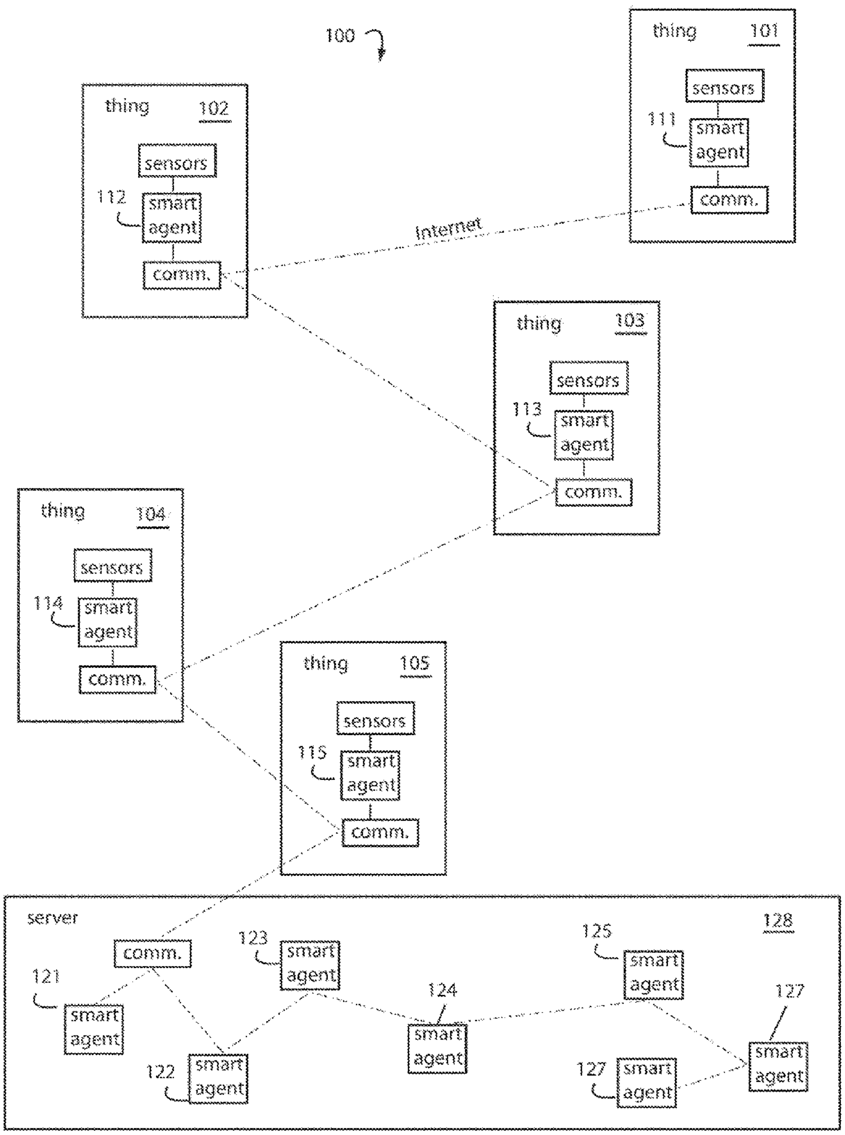
H04L 29/06	(2006.01)
G06N 99/00	(2006.01)

(52) **U.S. Cl.**

CPC ***H04L 63/102*** (2013.01); ***G06N 99/005*** (2013.01)

(57) **ABSTRACT**

An Internet-of-things (IoT) mechanizes, computerizes, automates, instruments, includes, and connects a broadly dispersed and extensively diverse universe of unrelated "things" to the Internet, e.g., credit cards, home appliances, industrial machinery, airplanes, cars, municipal water pumps, mobile devices, rain gauges, etc. Each thing is assigned a resident local "smart agent". Or an entity, manifesting remotely only as transaction records and reports, is assigned a virtual smart agent in a network server. These data structures follow, track, record, chart, monitor, characterize, describe, render, and otherwise provide a label and handle on independent things and entities.



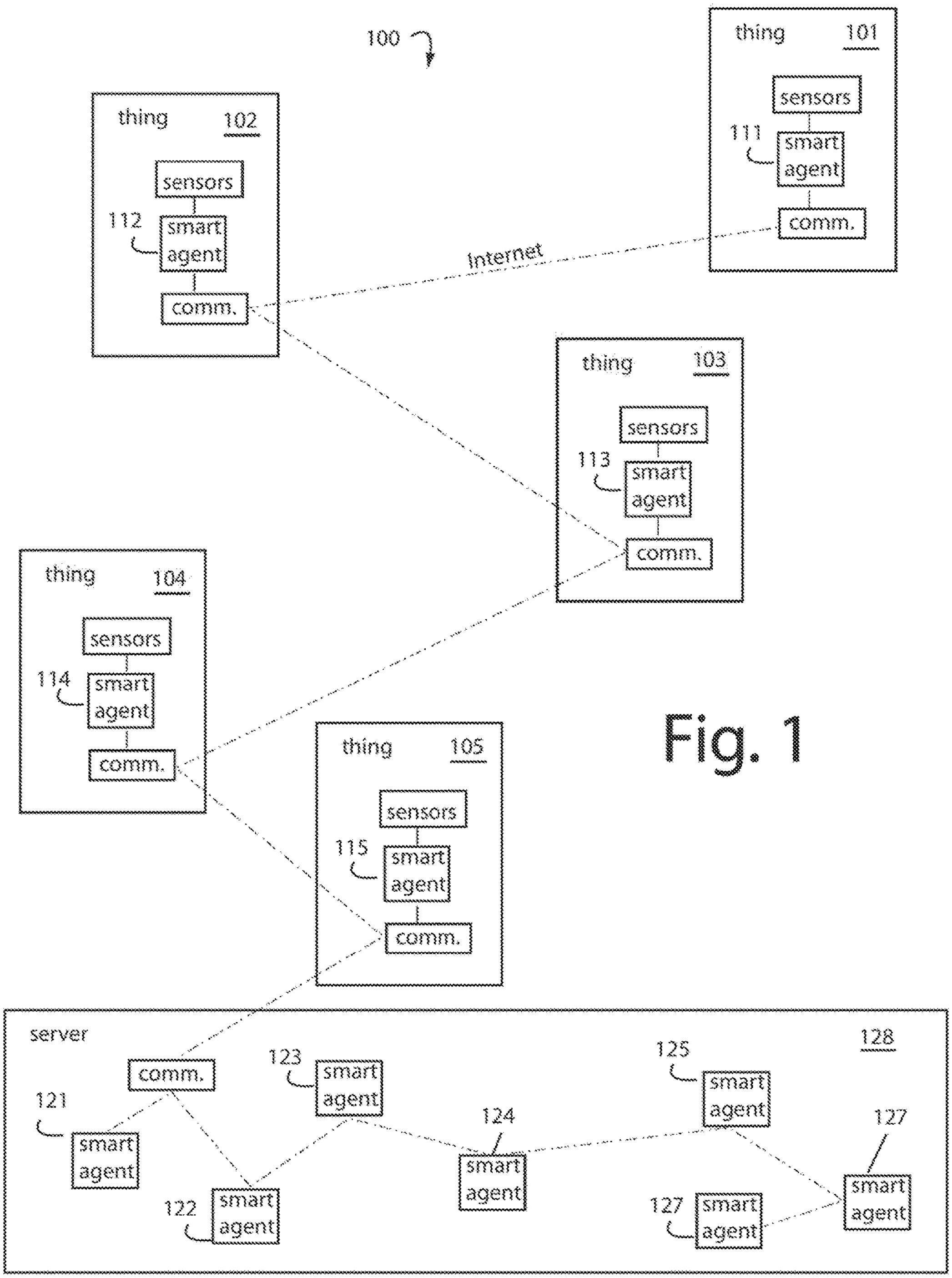
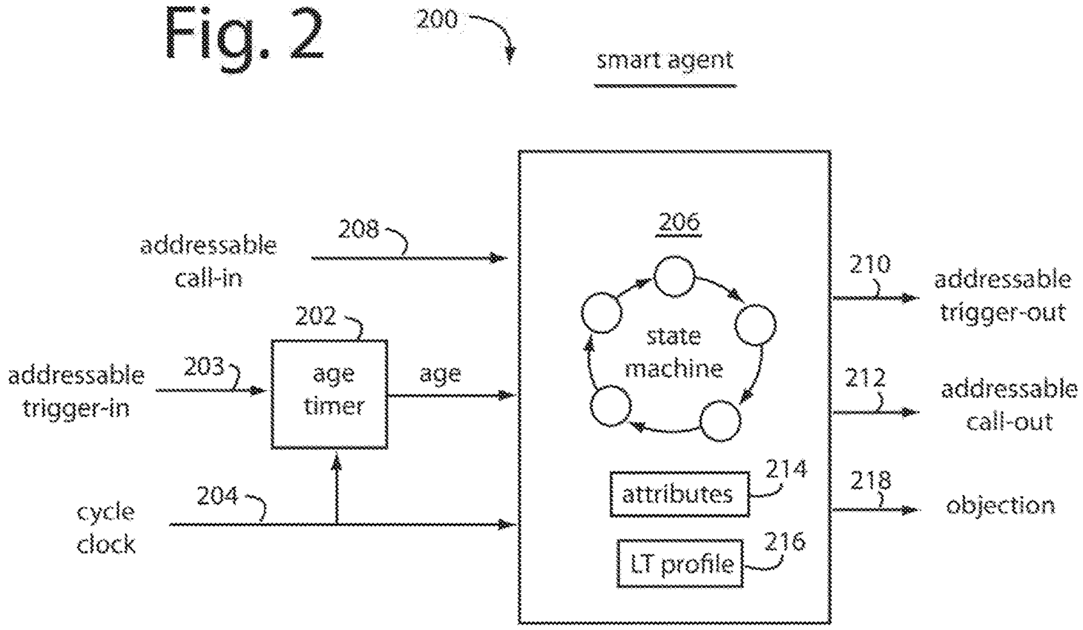

100
thing 101
sensors
111
smart agent
comm.
thing 102
sensors
112
smart agent
comm.
Internet
thing 103
sensors
113
smart agent
comm.
thing 104
sensors
114
smart agent
comm.
thing 105
sensors
115
smart agent
comm.
server 128
comm.
121
smart agent
122
smart agent
123
smart agent
124
smart agent
125
smart agent
127
smart agent
127
smart agent

Fig. 1

Fig. 2


200
smart agent
206
state
machine
208
addressable
call-in
202
age
timer
203
addressable
trigger-in
age
204
cycle
clock
210
addressable
trigger-out
212
addressable
call-out
214
attributes
216
LT profile
218
objection

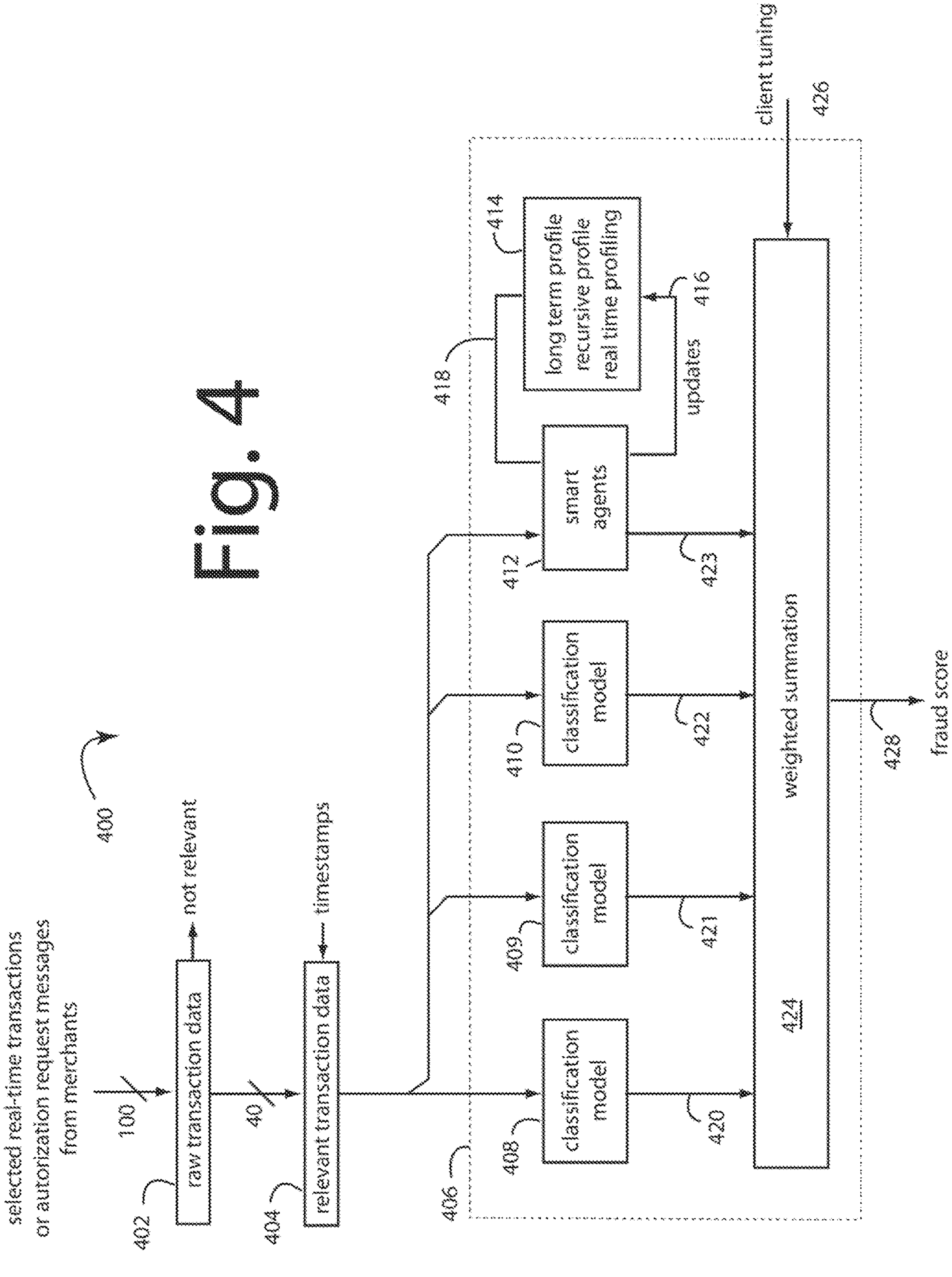

selected real-time transactions
or autorization request messages
from merchants
100
402
raw transaction data
not relevant
40
404
relevant transaction data
timestamps
400
406
408
classification
model
409
classification
model
410
classification
model
412
smart
agents
418
414
long term profile
recursive profile
real time profiling
416
updates
420
421
422
423
424
weighted summation
426
client tuning
428
fraud score

Fig. 4

Fig. 5

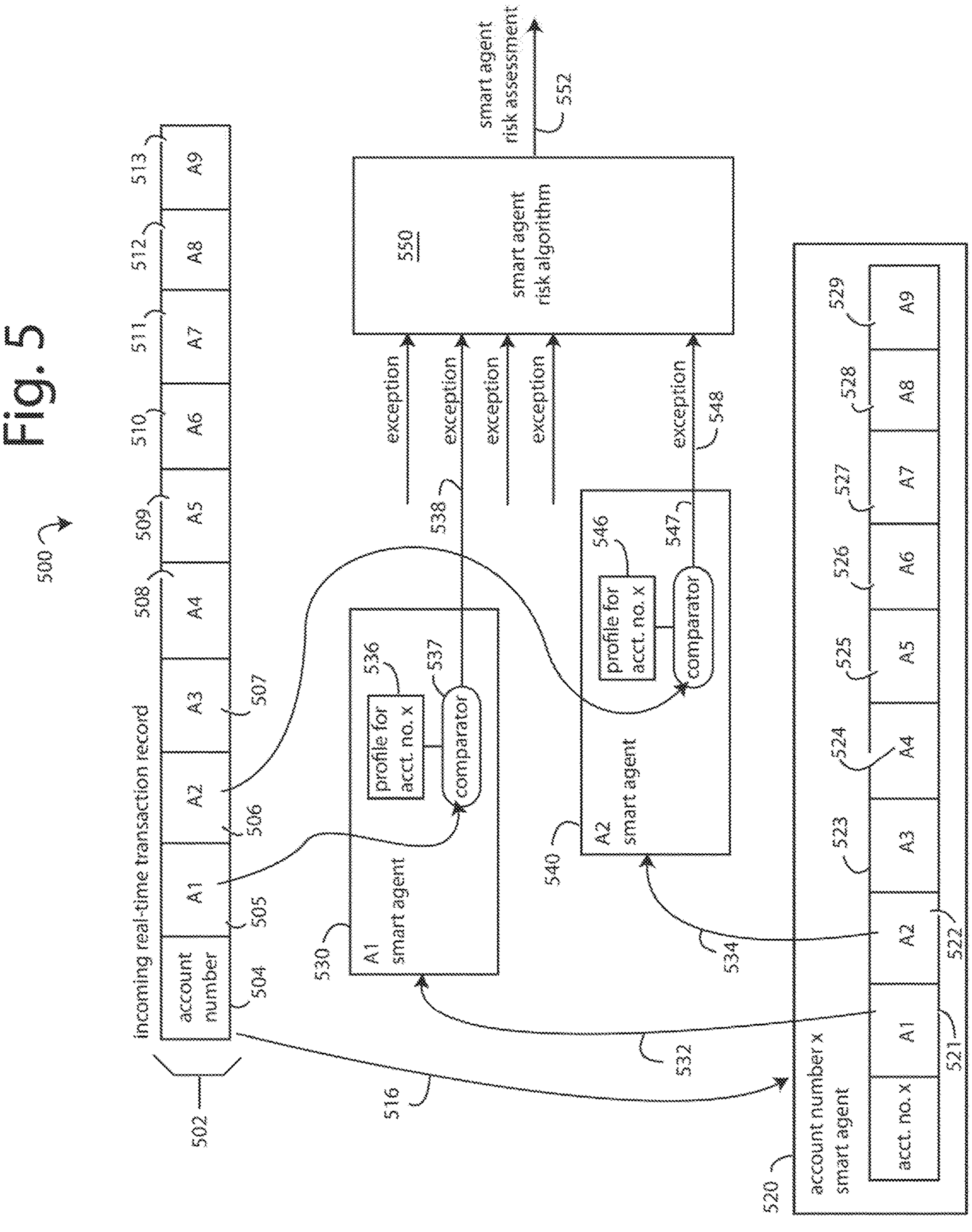
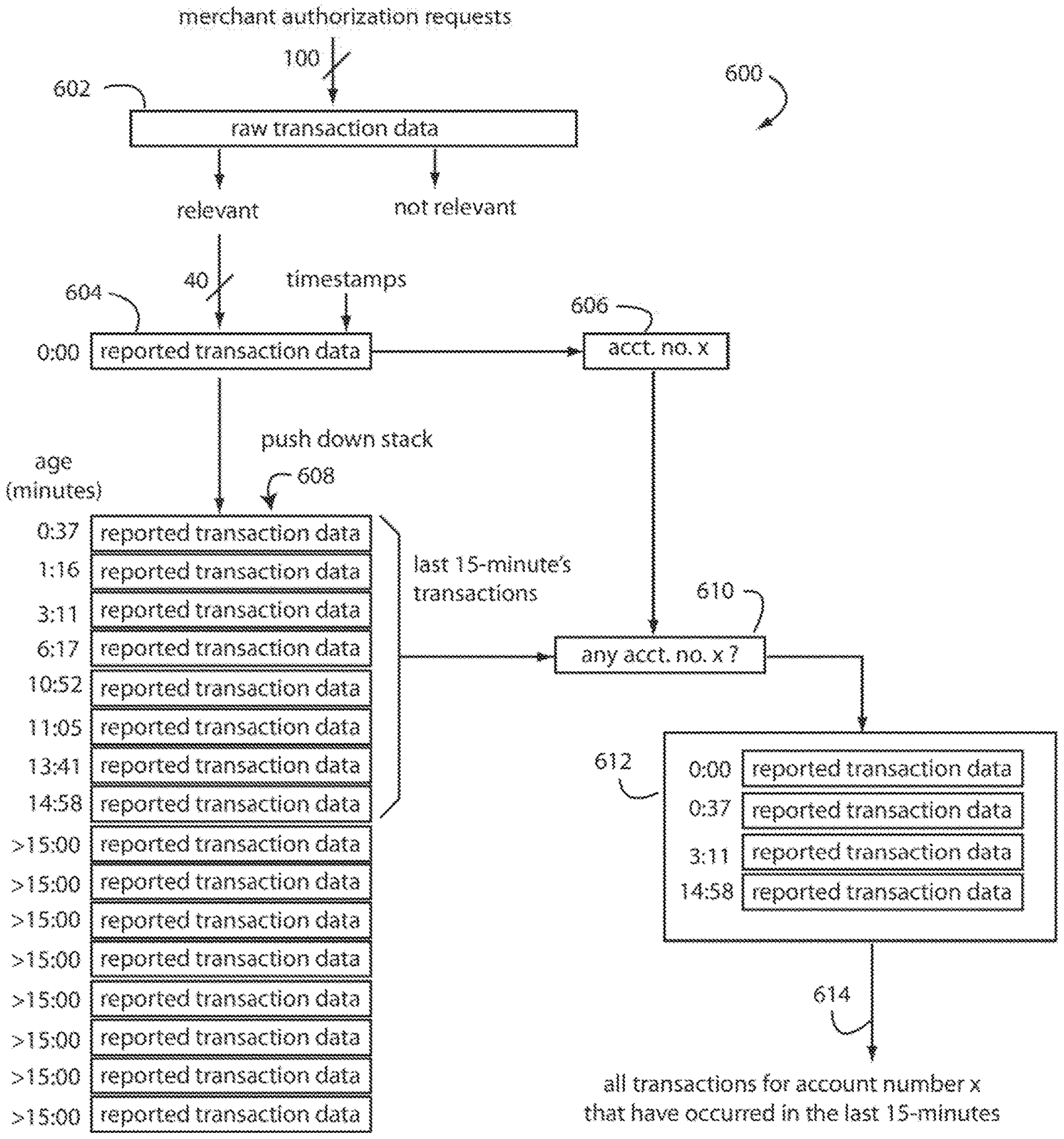

500
incoming real-time transaction record
502
account number
504
A1
505
A2
506
A3
507
A4
508
A5
509
A6
510
A7
511
A8
512
A9
513
516
A1 smart agent
530
profile for acct. no. x
536
comparator
537
538
exception
exception
exception
exception
exception
548
A2 smart agent
540
profile for acct. no. x
546
comparator
547
550
smart agent risk algorithm
552
smart agent risk assessment
532
534
account number x smart agent
520
acct. no. x
A1
521
A2
522
A3
523
A4
524
A5
525
A6
526
A7
527
A8
528
A9
529

Fig. 6


merchant authorization requests
100
602
600
raw transaction data
relevant
not relevant
40
timestamps
604
606
0:00
reported transaction data
acct. no. x
push down stack
age (minutes)
608
0:37
1:16
3:11
6:17
10:52
11:05
13:41
14:58
>15:00
reported transaction data
last 15-minute's transactions
610
any acct. no. x ?
612
0:00
0:37
3:11
14:58
reported transaction data
614
all transactions for account number x that have occurred in the last 15-minutes

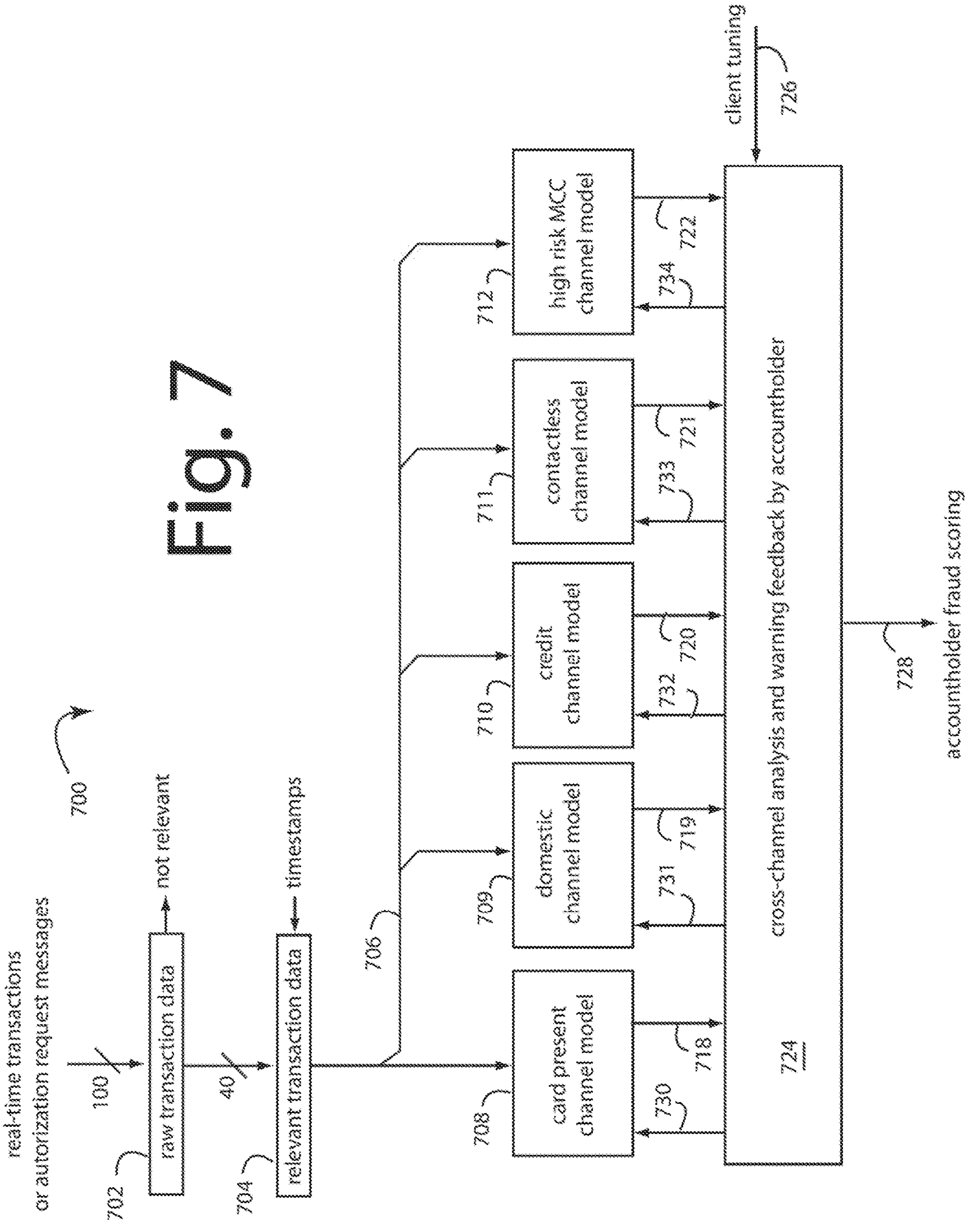


Fig. 7

700
real-time transactions
or autorization request messages
100
702
raw transaction data
not relevant
40
704
relevant transaction data
timestamps
706
708
card present channel model
709
domestic channel model
710
credit channel model
711
contactless channel model
712
high risk MCC channel model
718
719
720
721
722
730
731
732
733
734
cross-channel analysis and warning feedback by accountholder
724
client tuning
726
728
accountholder fraud scoring

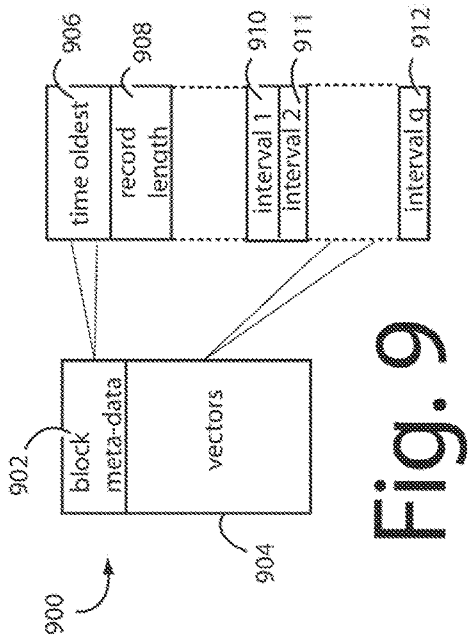


Fig. 9

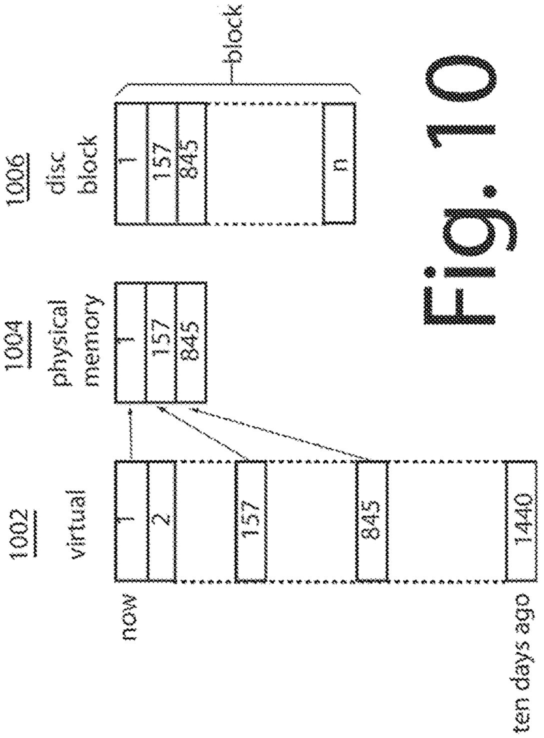


Fig. 10

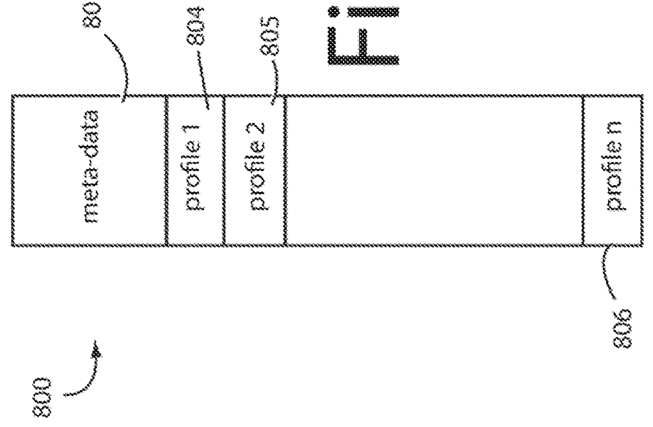


Fig. 8

Fig. 11

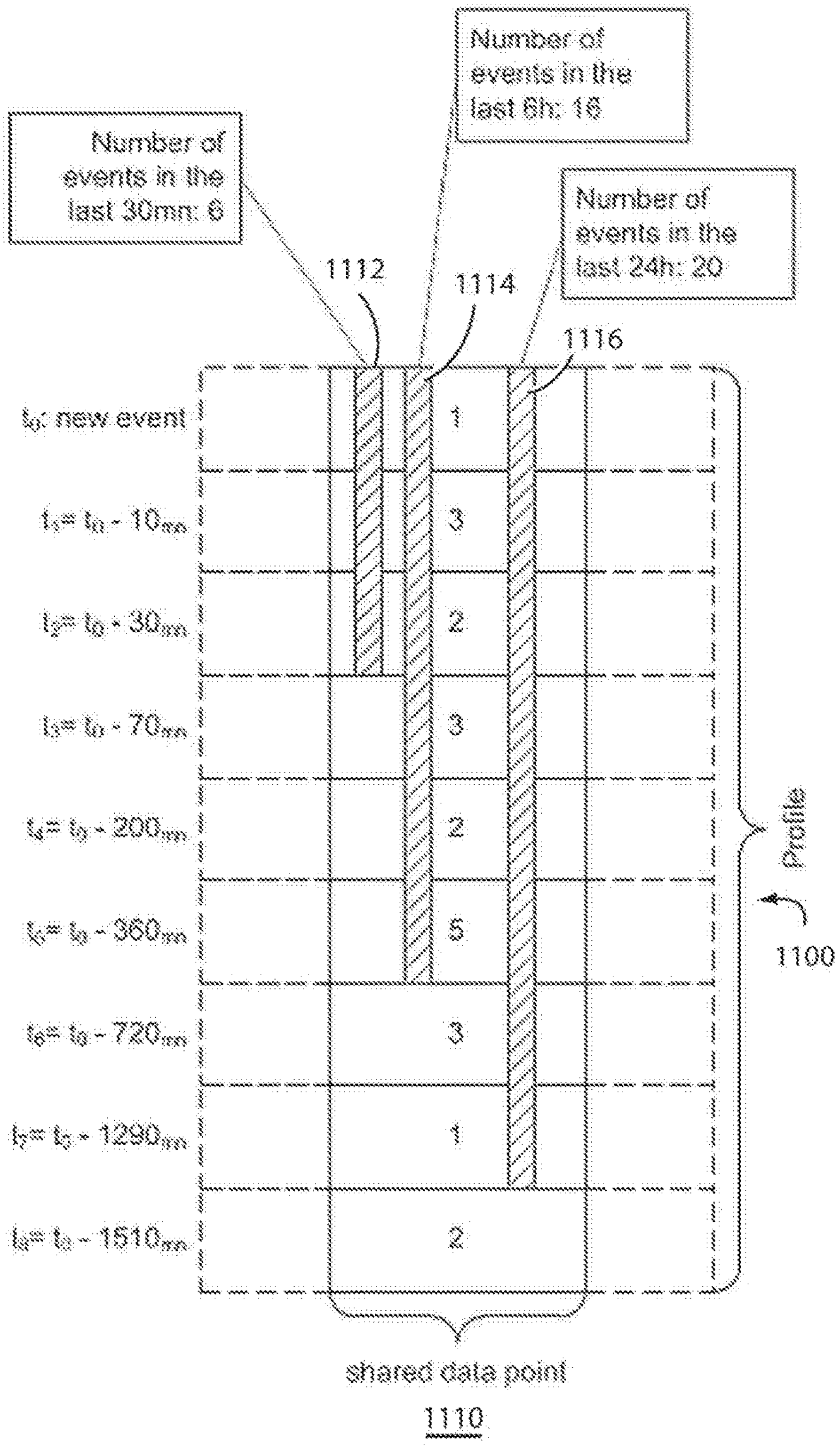

Number of events in the last 30mn: 6
Number of events in the last 6h: 16
Number of events in the last 24h: 20
1112
1114
1116
t_0: new event
t_1= t_0 - 10_{mn}
t_2= t_0 - 30_{mn}
t_3= t_0 - 70_{mn}
t_4= t_0 - 200_{mn}
t_5= t_0 - 360_{mn}
t_6= t_0 - 720_{mn}
t_7= t_0 - 1290_{mn}
t_8= t_0 - 1510_{mn}
1
3
2
3
2
5
3
1
2
Profile
1100
shared data point
1110

Fig. 12

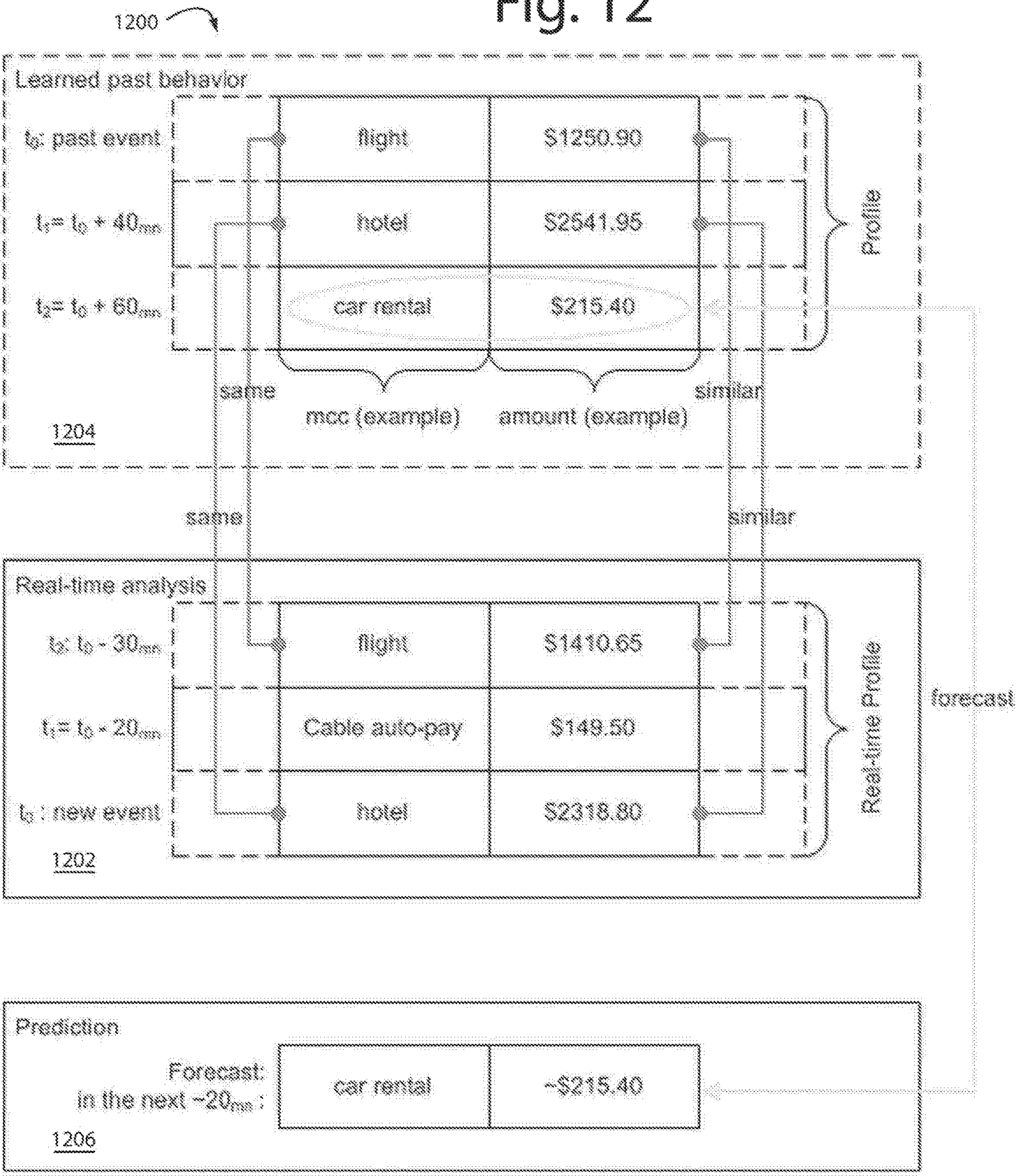
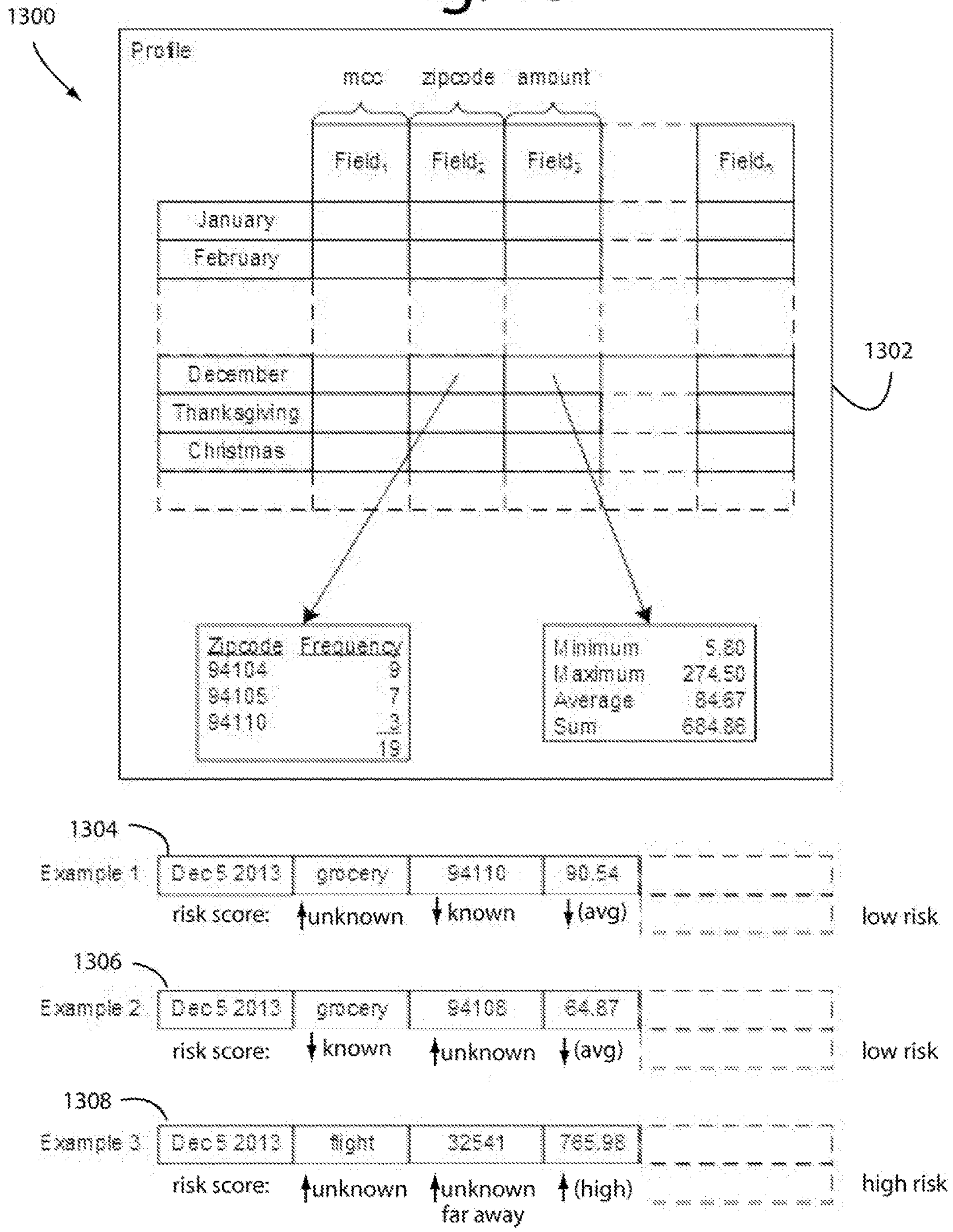

1200
Learned past behavior
t_0: past event
flight
$1250.90
$t_1 = t_0 + 40_{mn}$
hotel
$2541.95
$t_2 = t_0 + 60_{mn}$
car rental
$215.40
Profile
same
mcc (example)
amount (example)
similar
1204
same
similar
Real-time analysis
t_2: t_0 - 30_{mn}
flight
$1410.65
$t_1 = t_0 - 20_{mn}$
Cable auto-pay
$149.50
t_0 : new event
hotel
$2318.80
Real-time Profile
forecast
1202
Prediction
Forecast:
in the next ~20_{mn} :
car rental
~$215.40
1206

Fig. 13


1300
Profile
mcc
zipcode
amount
Field$_1$
Field$_2$
Field$_3$
Field$_n$
January
February
December
Thanksgiving
Christmas
1302
Zipcode Frequency
94104 9
94105 7
94110 3
19
Minimum 5.80
Maximum 274.50
Average 84.67
Sum 684.86
1304
Example 1
Dec 5 2013
grocery
94110
90.54
risk score:
unknown
known
(avg)
low risk
1306
Example 2
Dec 5 2013
grocery
94108
64.87
risk score:
known
unknown
(avg)
low risk
1308
Example 3
Dec 5 2013
flight
32541
765.98
risk score:
unknown
unknown far away
(high)
high risk

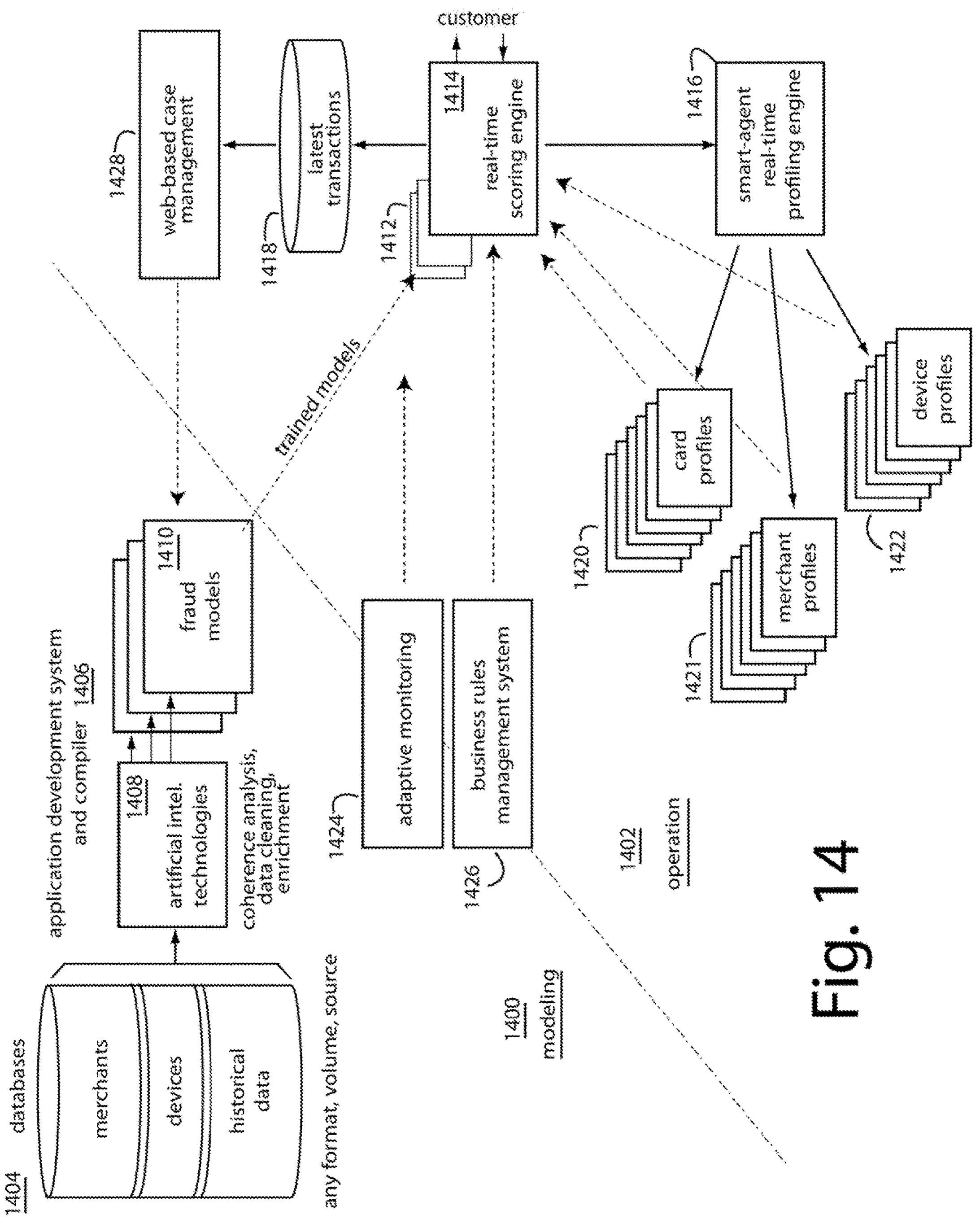

1404
databases
merchants
devices
historical data
any format, volume, source
application development system and compiler 1406
1408
artificial intel. technologies
coherence analysis, data cleaning, enrichment
1410
fraud models
trained models
1400
modeling
1424
adaptive monitoring
1426
business rules management system
1402
operation
1428
web-based case management
1418
latest transactions
1412
1414
real-time scoring engine
customer
1416
smart-agent real-time profiling engine
1420
card profiles
1421
merchant profiles
1422
device profiles

Fig. 14

SECURING INTERNET-OF-THINGS WITH SMART-AGENT TECHNOLOGY

BACKGROUND OF THE INVENTION

Field of the Invention

[0001] The present invention relates to artificial intelligence (AI), and more particularly to securing business insider threats detectable by automated system administrator behavior analysis.

Background Art

[0002] While AI seems to have only recently captured the attention of humanity, the reality is that AI has generally been around for over 60 years as a technological discipline. In the late 1950's, Arthur Samuel wrote a checkers playing program that could learn from its mistakes and thus, over time, became better at playing the game. MYCIN, the first rule-based expert system, was developed in the early 1970's and was capable of diagnosing blood infections based on the results of various medical tests. The MYCIN system was able to perform better than non-specialist doctors. Thus, in a general sense, while AI may be used to mimic what best humans minds can accomplish, AI is not a patent ineligible mental process as some have contended.

[0003] The Internet of Things (IoT) is a new idea for a network of machinery, appliances, and other physical objects or "things" embedded with electronics, software, sensors, and the network connectivity needed to enable automatic data exchanges with other connected devices. How to do that securely and not allow criminal to highjack these things for their own purposes is a major issue and challenge facing the developing IoT.

[0004] The Internet of Things adds sensors to everyday items, and is becoming more of a reality every day. IoT enables smart devices like alarms, heating systems and coffee machines, to store data and communicate with one another. Some manufacturers are trying to develop products, from washing machines to water filters, that can sense when a consumer is running low on supplies and automatically place an order without needing so much as to press a button. IBM announced a $3 billion investment to create an Internet of Things unit that will track all the data cast off by devices. Until now, most reports and forecasts of the Internet of Things has seemed a fantasy. That is starting to change.

[0005] In many organizations, information travels along well-worn routes. Proprietary information is secured in databases and analyzed in reports before it makes its way up the management chain. Information originating externally is gathered from public sources, collected from the Internet, or acquired from information suppliers.

[0006] Nevertheless, the foreseeable pathways of information are changing. The physical world is becoming a type of information system itself. In the Internet of Things, sensors and actuators embedded in physical objects—from roadways to pacemakers—are linked by wired and wireless networks, using the Internet Protocol (IP) that connects the rest of the Internet. These networks spew out enormous volumes of data for analysis.

[0007] When objects can both sense the environment and communicate with others, they become agents for understanding and quick response. Physical information systems are now being deployed that go to work without needing human intervention. Automated farming equipment with wireless data links can adjust the way each individual field is farmed. Billboards in Japan now peer back at passersby, assessing how they fit consumer profiles, and adapt the messages displayed based on ad hoc automated calculations.

[0008] A widespread adoption of the Internet of Things is expected to take time. Advances in standardized wireless networking technology and communications protocols makes it now possible to collect data from sensors almost anywhere any time. Integrated circuit devices are gaining new capabilities, costing less, and being ever more widely adopted and understood. Surprising and continuing increases in storage and computing power make very large-scale number crunching possible and at next to no cost.

[0009] There are a number of issued patents that relate to IoT. The following issued patents have turned up in a search for art that may or may not be relevant to the technologies claimed below: U.S. Pat. Nos. 9,342,779; 9,222,837; and 9,117,949. However, none of these patents disclose or describe the use of artificial intelligence and machine learning based technologies as claimed below.

[0010] In any case, there are opportunities in the art to provide an improved component that helps secure IoT through use of smart agents, artificial intelligence, and machine learning.

SUMMARY OF THE INVENTION

[0011] Briefly, an Internet-of-things (IoT) embodiment of the present invention mechanizes, computerizes, automates, instruments, includes, and connects a broadly dispersed and extensively diverse universe of unrelated "things" to the Internet, e.g., credit cards, home appliances, industrial machinery, airplanes, cars, municipal water pumps, mobile devices, rain gauges, etc. Each thing is assigned a resident local "smart agent". Or an entity, manifesting remotely only as transaction records and reports, is assigned a virtual smart agent in a network server. These follow, track, record, chart, monitor, characterize, describe, render, and otherwise provide a label and handle on independent things and entities.

[0012] Thus, provided is an Internet-of-things (IoT) component, comprising: an electronic smart-agent data structure resident in a computer memory, and accessible through a communications device connectable to the Internet; a means for associating data related to a particular thing or entity to the smart-agent data structure; and a means for collecting data and attributes descriptive of or related to said thing or entity and packing such into the smart-agent data structure. Also provided are a means for rendering a descriptive representation or behavioral analysis of the thing or entity from the data kept within the smart-agent data structure; and a profile for storing in an array any historical data related to said particular thing or entity.

[0013] In some embodiments, the component may further include a means for operating alone while disposed within a single said particular thing or entity; and means for communicating said profile, data and attributes with other smart-agent data structures resident in different locations.

[0014] Other and still further objects, features, and advantages of the present invention will become apparent upon consideration of the following detailed description of specific embodiments thereof, especially when taken in conjunction with the accompanying drawings.

BRIEF DESCRIPTION OF THE DRAWINGS

[0015] FIG. **1** is functional block diagram of an Internet-of-Things (IoT) embodiment of the present invention which assigned and attaches smart agents to the things and entities of interest;

[0016] FIG. **2** is a functional block diagram of a smart-agent embodiment of the present invention could be useful in the IoT of FIG. **1**;

[0017] FIG. **4** is functional block diagram of a real-time payment fraud management system like that illustrated in FIG. **1** as applied payment fraud model;

[0018] FIG. **5** is functional block diagram of a smart agent process embodiment of the present invention;

[0019] FIG. **6** is functional block diagram of a most recent fifteen-minute transaction velocity counter;

[0020] FIG. **7** is functional block diagram of a cross-channel payment fraud management embodiment of the present invention;

[0021] FIG. **8** is a diagram of a group of smart agent profiles stored in a custom binary file;

[0022] FIG. **9** is a diagram of the file contents of an exemplary smart agent profile;

[0023] FIG. **10** is a diagram of a virtual addressing scheme used to access transactions in atomic time intervals by their smart agent profile vectors;

[0024] FIG. **11** is a diagram of a small part of an exemplary smart agent profile that spans several time intervals;

[0025] FIG. **12** is a diagram of a behavioral forecasting aspect of the present invention;

[0026] FIG. **13** is a diagram representing a simplified smart agent profile and how individual constituent datapoints are compared to running norms and are accumulated into an overall risk score; and

[0027] FIG. **14** is a functional block diagram of a modeling and operational environment in which an application development system is used initially to generate, launch, and run millions of smart agents and their profiles.

DETAILED DESCRIPTION OF THE INVENTION

Definitions and Overview

[0028] Before describing the invention in detail, it is to be understood that the invention is not generally limited to specific electronic platforms or types of computing systems, as such may vary. It is also to be understood that the terminology used herein is intended to describe particular embodiments only, and is not intended to be limiting.

[0029] Furthermore, as used in this specification and the appended claims, the singular article forms "a," "an," and "the" include both singular and plural referents unless the context clearly dictates otherwise. Thus, for example, reference to "a smart agent" includes a plurality of smart agents as well as a single smart agent, reference to "a component" includes a single component as well as a collection of component, and the like.

[0030] In addition, the appended claims are to be interpreted as reciting subject matter that may take the form of a new and useful process machine, manufacture, and/or composition of matter, and/or any new and useful improvement thereof instead of an abstract idea.

[0031] In this specification and in the claims that follow, reference is made to a number of terms that are defined to have the following meanings, unless the context in which they are employed clearly indicates otherwise:

[0032] The terms "electronic," "electronically," and the like are used in their ordinary sense and relate to structures, e.g., semiconductor microstructures, that provide controlled conduction of electrons or other charge carriers, e.g., microstructures that allow for the controlled movement of holes or electrons in electron clouds.

[0033] The term "entity" is used herein in its ordinary sense and may refer to a construct or thing with distinct and independent existence, such as a human individual, a business, a corporation, a partnership, etc.

[0034] The term "internet" is used herein in its ordinary sense and refers to an interconnected system of networks that connects computers around the world via the TCP/IP and/or other protocols. Unless the context of its usage clearly indicates otherwise, the term "web" is generally used in a synonymous manner with the term "internet."

[0035] The term "method" is used herein in a synonymous manner as the term "process" is used in 35 U.S.C. 101. Thus, both "methods" and "processes" described and claimed herein are patent eligible per 35 U.S.C. 101.

[0036] The term "smart agent" is used herein as a term of art to refer to specialized technology that differs from prior art technologies relating to bots or agents, e.g., used in searching information or used by social medial to keep track of birthday's systems or order pizzas. A "smart agent" described herein is an entity that is capable of having an effect on itself and its environment. It disposes of a partial representation of this environment. Its behavior is the outcome of its observations, knowledge and interactions with other smart agents. The smart agent technology described herein, rather than being pre-programmed to try to anticipate every possible scenario or relying on pre-trained models, tracks and adaptively learns the specific behavior of every entity of interest over time. Thus, continuous one-to-one electronic behavioral analysis provides real-time actionable insights and/or warnings. In addition, smart agent technology described herein engages in adaptive learning that continually updates models to provide new intelligence. Furthermore, the smart agent technology solves technical problems associated with massive databases and/or data processing. Experimental data show about a one-millisecond response on entry-level computer servers. Such a speed is not achievable with prior art technologies. Additional differences between the smart agent technology claimed and prior so-called "smart agent" technology will be apparent upon review of the disclosure contained herein.

[0037] The term "thing" is used in its ordinary sense and refers to a physical object, e.g., embedded with electronics, software, sensors, etc.

[0038] The terms "substantial" and "substantially" are used in their ordinary sense and are the antithesis of terms such as "trivial" and "inconsequential." For example, when the term "substantially" is used to refer to behavior that deviates from a reference normal behavior profile, the difference cannot constitute a mere trivial degree of deviation. The terms "substantial" and "substantially" are used analogously in other contexts involve an analogous definition.

EXEMPLARY EMBODIMENTS OF THE INVENTION

[0039] FIG. **1** represents an Internet-of-things (IoT) embodiment of the present invention and is referred to herein by the general reference numeral **100**. IoT **100** mechanizes, computerizes, automates, instruments, includes, and connects a broadly dispersed and extensively diverse universe of unrelated component "things" **101-105** to the Internet **108**, e.g., credit cards, home appliances, industrial machinery, airplanes, cars, municipal water pumps, mobile devices, rain gauges, etc. Each thing **101-105** is assigned a resident local "smart agent" **111-115**. Or an entity, manifesting remotely only as transaction records and reports, is assigned a virtual smart agent **121-127** in a server **128**. These follow, track, record, chart, monitor, characterize, describe, render, and otherwise provide a label and handle on independent things and entities. Heretofore such things and entities ranged from unknown to unseen, to inaccessible, to forgotten, to unreachable, to uncontrollable, to inscrutable, to unreadable, to incomprehensible.

[0040] A "smart agent" by our definition resembles a data structure prearranged within a computer memory to epitomize an entity or thing in all its essentials by the entity's attributes, history, and behaviors that are manifest, e.g., from specifications, programming, sensors, and transaction data. Each smart agent allows an electronic rendering of the entity or thing it is consigned to in several different and selectable dimensions. Each smart agent **111-115** and **121-127** can be equipped with "action items" or a "job description" that boils down into clocking a series of steps in a finite state machine (FSM), given various conditions being present in its data, sensors, programming, or other data solicited from other smart agents. Such finite state machine can issue output signals and triggers in each step to other smart agents. The computer processor resources necessary to support such functionality are specially adapted to read/write the data structures in memory, run the finite state machines, provide the inputs, and generate the outputs. In the field, such computer processor resources can be a shared mobile device, an embedded microcomputer, or batch processor. A typical smartphone today represents all the sensor, processing, and communications hardware a typical smart agent **111-115** would need.

[0041] A smart agent can exist at some central location pooled with hundreds, thousands, or even millions of others that receive transaction records reporting the remote activities of the corresponding participants each is assigned to follow. For example, inside a network server they can interoperate and intercommunicate fairly freely and efficiently.

[0042] In the Internet-of-Things, the hundreds, thousands, or even millions of participants can be widely dispersed and are each assigned and equipped with a smart agent that is able to communicate with the others. Nowadays, that communication would rely on a wireless technology like WiFi, Bluetooth, NFC, GSM, 4G, etc. Some wireless technologies can breach long distances, others have the advantage of needing to be proximate or very close. That may help secure access to authorized (local) users in a very simple way hard to subvert.

[0043] Participants and entities in general are describable by their attributes. Even in widely diverse groups. In one sense, such participants and entities are nothing more than the sum of their attributes. Groups too. And attributes too are describable by their attributes. For example, if one attribute of an entity with a smart agent attached was the color blue, then a smart agent for "blue" could have as its attributes all the users who have the attribute blue. Another attribute could be a paint manufacturer's paint formula code for the blue. Accessing the blue smart agent would get you links immediately to every other smart agent describing itself as blue.

[0044] Attributes can be independently variable, fixed, or programmable. The operational status (on/off) of a device can be an independent variable reportable as an attribute. As are ambient temperature, noise, vibration, load, voltage, fuel level, service age. The model and serial number of a device can be a fixed attribute, as are assigned location, color, weight, specifications. A programmable attribute can be like start/stop, accelerate/decelerate, inflate/deflate, heat/cool.

[0045] Not all attributes need to be communicated. It may be safe or reasonable to assume or interpolate. For example, to limit communications bandwidth demands.

[0046] A piece of machinery can be a "thing" in the Internet-of-Things. Such could be equipped with appropriate sensors to measure ambient noise, temperature, load, output, energy consumption, vibration, etc. Measured and logged over time these attributes will usually fall into some routine or normal pattern of behavior. Smart agent profiles can be used to store and characterize what is normal for its "thing". Deviations from such normalcy can spell trouble, warn of impending failure, call for maintenance, signal intrusion, etc. The smart agent itself can be designed to take some kind of action, e.g., by communicating a "check machine" warning.

[0047] For a car, a location attribute would be an independent variable requiring a sensor like a GPS receiver. But for a utility power transformer, the location would normally be fixed. It could however be programmable, as in the case where a new utility power transformer is located in a warehouse inventory, then assigned to be installed on a utility pole, or has been removed to a hazardous waste site to have its toxic oil reserves removed. In this later case, the smart agent could be implemented within the utility power transformer, but would be better in a virtual location that always had power and communications to stay alive. Like an on-line dossier file.

[0048] The "thing" smart agents **111-115** can comprise attributes that have corresponding "attribute" smart agents among **121-127**. Discovering or broadcasting to all "thing" smart agents **111-115** that share a particular attribute would be immediately and directly accessible and quantifiable by the particular "attribute" smart agent **121-127** within server **128**. For example, if the attribute were "MasterCard", then all credit card entities or things **101-105** with that attribute would be immediately accessible. In another example, if the attribute were "California AKW 887", then any automobile "thing" **101-105** with that license plate attribute would be immediately accessible and available to be queried to report its GPS location.

[0049] In one embodiment of the present invention we described in United States patent application, SYSTEM ADMINISTRATOR BEHAVIOR ANALYSIS, Ser. No. 14/634,786, filed, Feb. 28, 2015, a smart-agent with case-based-reasoning (CBR) is virtually "attached" and assigned to every individual system admin, job, and task. Storage room for their respective profiles are maintained in secure memory inaccessible to the trusted system administrators being watched.

[0050] Referring now to FIG. **2**, each smart-agent **200** is addressable and has a timer **202** can be triggered into life with an addressable trigger-in **203** and begin aging tick-by-tick with a cycle clock **204**. A state machine **206** can be addressably called into action like a “call” to a subroutine with an addressable call-in **208**. An addressable trigger-out **210** can trigger into life other smart-agents. An addressable call-out **212** can call into action other smart-agents as if they were addressable subroutines. A list of attributes **214** describes, in an exemplary instance here, the particular tasks employed by this particular job, or the tasks that a particular system administrator can employ. A long term (LT) profile **216** is a memory log of the past activities that this smart-agent was involved in, and is able to develop a behavior profile of what is “normal” behavior for this entity.

[0051] An objection **218** can issue by the state machine **206** if the instant behavior for this entity seems abnormal, or if an age timeout **220** occurs before the state machine has run or finished in response to an addressable call-in **208**.

[0052] Activity reports **220** are cleaned up, filtered for the particular smart-agent **200**, and used to build LT profile **216**. As each report comes in its information is inspected by state machine **206** to see if the activity was expected, normal, timely, respected priorities, etc. For example, if the activity was the running of a task.

[0053] Once an addressable call-in **208** is received, the state machine **206** will typically consult the attributes **214** to see what other addressable triggers-out **210** and addressable calls-out **212** should issue and in which clock cycles. For example, if a Job-A requires tasks t**1**-t**7** to be run, then the Job-A smart-agent will trigger all seven of the T**1**-T**7** smart-agents. If they timeout (age is too old) without having been employed in a call by the system admin, then the ones who weren’t called into action will issue objections.

[0054] Here, an individual smart-agent **200** is spawned and attached to every identifiable system admin, job, and task. Each such smart-agent has its own characteristic attributes, e.g., a job smart-agent will have task attributes corresponding to every task that this particular job has called, should call, or should not call. The tasks it calls can have a priority order, and that would be another attribute and another smart-agent. The various smart-agents are interconnected, interrelated and each can be randomly accessed and consulted.

[0055] For example, any job smart-agent can have its LT profile **216** accessed to see who has called it, triggered it, it has called, it has triggered, etc. It can further be queried as to its attributes **214**. It is therefore as easy to query what jobs have been done by which system administrators as it is to query which system administrators have done which jobs.

[0056] A CBR case consists of a problem, a previous solution that worked, and remarks about how the solution was derived. Case-based reasoning can be formalized as a four-step process:

Retrieve	For each target problem, cases are retrieved from memory relevant to solving it.
Reuse	The solution is mapped from the previous case to the target problem and may involve adapting the solution to fit the new situation.
Revise	The new solution is tested and, if necessary, revised.
Retain	After a solution has been used successfully on the target problem, the resulting experience is stored as a new case in memory.

[0057] Herein, a case comprises a system administrator’s job task and the solutions comprise what particular system administrators did to do that job task. (There being many ways to solve a problem or do a job that will express the personalities involved.)

[0058] FIG. **4** represents a real-time payment fraud management system **400** like that illustrated in FIG. **1** as applied payment fraud model **114**. A raw transaction separator **402** filters through the forty or so data items that are relevant to the computing of a fraud score. A process **404** adds time-stamps to these relevant data points and passes them in parallel to a selected applied fraud model **406**. This is equivalent to a selected one of applied fraud models **316-323** in FIG. **3** and applied payment fraud model **114** in FIG. **1**.

[0059] During a session in which the time-stamped relevant transaction data flows in, a set of classification models **408-410** operate independently according to their respective natures. A population of smart agents **412** and profilers **414** also operate on the time-stamped relevant transaction data inflows. Each new line of time-stamped relevant transaction data will trigger an update **416** of the respective profilers **414**. Their attributes **418** are provided to the population of smart agents **412**.

[0060] The classification models **408-410** and population of smart agents **412** and profilers **414** all each produce an independent and separate vote or fraud score **420-423** on the same line of time-stamped relevant transaction data. A weighted summation processor **424** responds to client tunings **426** to output a final fraud score **428**.

[0061] The profiles associated with our Smart Agents enable them to automatically adapt without human intervention and without going off-line for retraining of the classification algorithms. And, neural networks are only one type of fraud classification used in parallel with Smart Agents to provide a vote that can be balanced and tuned by the user banks. Weighted Summation **424** FIG. **4**. Our Smart Agents do not themselves include a neural network classification model. We also discovered that a single classification model is not enough, several different kinds in parallel are better at controlling fraud.

[0062] FIG. **5** represents a smart agent process **500** in an embodiment of the present invention. For example, these would include the smart agent population build **334** and profiles **336** in FIG. **3** and smart agents **412** and profiles **414** in FIG. **4**. A series of payment card transactions arriving in real-time in an authorization request message is represented here by a random instantaneous incoming real-time transaction record **502**.

[0063] Such record **502** begins with an account number **504**. It includes attributes A**1**-A**9** numbered **505-513** here. These attributes, in the context of a payment card fraud application would include data points for card type, transaction type, merchant name, merchant category code (MCC), transaction amount, time of transaction, time of processing, etc.

[0064] Account number **504** in record **502** will issue a trigger **516** to a corresponding smart agent **520** to present itself for action. Smart agent **520** is simply a constitution of its attributes, again A**1**-A**9** and numbered **521-529** in FIG. **5**. These attributes A**1**-A**9** **521-529** are merely pointers to attribute smart agents. Two of these, one for A**1** and one for A**2**, are represented in FIG. **5**. Here, an A**1** smart agent **530** and an A**2** smart agent **540**. These are respectively called into action by triggers **532** and **542**.

[0065] A**1** smart agent **530** and A**2** smart agent **540** will respectively fetch correspondent attributes **505** and **506** from incoming real-time transaction record **502**. Smart agents for A**3**-A**9** make similar fetches to themselves in parallel. They are not shown here to reduce the clutter for FIG. **5** that would otherwise result.

[0066] Each attribute smart agent like **530** and **540** will include or access a corresponding profile data point **536** and **546**. This is actually a simplification of the three kinds of profiles **336** (FIG. **3**) that were originally built during training and updated in update **416** (FIG. **4**). These profiles are used to track what is "normal" behavior for the particular account number for the particular single attribute.

[0067] For example, if one of the attributes reports the MCC's of the merchants and another reports the transaction amounts, then if the long-term, recursive, and real time profiles for a particular account number x shows a pattern of purchases at the local Home Depot and Costco that average $100-$300, then an instantaneous incoming real-time transaction record **502** that reports another $200 purchase at the local Costco will raise no alarms. But a sudden, unique, inexplicable purchase for $1250 at a New York Jeweler will and should throw more than one exception.

[0068] Each attribute smart agent like **530** and **540** will further include a comparator **537** and **547** that will be able to compare the corresponding attribute in the instantaneous incoming real-time transaction record **502** for account number x with the same attributes held by the profiles for the same account. Comparators **537** and **547** should accept some slack, but not too much. Each can throw an exception **538** and **548**, as can the comparators in all the other attribute smart agents. It may be useful for the exceptions to be a fuzzy value, e.g., an analog signal 0.0 to 1.0. Or it could be a simple binary one or zero. What sort of excursions should trigger an exception is preferably adjustable, for example with client tunings **426** in FIG. **4**.

[0069] These exceptions are collected by a smart agent risk algorithm **550**. One deviation or exception thrown on any one attribute being "abnormal" can be tolerated if not too egregious. But two or more should be weighted more than just the simple sum, e.g., $(1+1)^n=2^n$ instead of simply 1+1=2. The product is output as a smart agent risk assessment **552**. This output is the equivalent of independent and separate vote or fraud score **423** in FIG. **4**.

[0070] FIG. **6** represents a most recent 15-minute transaction velocity counter **600**, in an embodiment of the present invention. It receives the same kind of real-time transaction data inputs as were described in connection with FIG. **4** as raw transaction data **402** and FIG. **5** as records **502**. A raw transaction record **602** includes a hundred or so data points. About forty of those data points are relevant to fraud detection an identified in FIG. **6** as reported transaction data **604**.

[0071] The reported transaction data **604** arrive in a time series and randomly involve a variety of active account numbers. But, let's say the most current reported transaction data **604** with a time age of 0:00 concerns a particular account number x. That fills a register **606**.

[0072] Earlier arriving reported transaction data **604** build a transaction time-series stack **608**. FIG. **6** arbitrarily identifies the respective ages of members of transaction time-series stack **608** with example ages 0:73, 1:16, 3:11, 6:17, 10:52, 11:05, 13:41, and 14:58. Those aged more than 15-minutes are simply identified with ages ">15:00". This embodiment of the present invention is concerned with only the last 15-minutes worth of transactions. As time passes transaction time-series stack **608** pushes down.

[0073] The key concern is whether account number x has been involved in any other transactions in the last 15-minutes. A search process **610** accepts a search key from register **606** and reports any matches in the most 15-minute window with an account activity velocity counter **612**. Too much very recent activity can hint there is a fraudster at work, or it may be normal behavior. A trigger **614** is issued that can be fed to an additional attribute smart agent that is included with attributes smart agents **530** and **540** and the others in parallel. Exception from this new account activity velocity counter smart agent is input to smart agent risk algorithm **550** in FIG. **5**.

[0074] FIG. **7** represents a cross-channel payment fraud management embodiment of the present invention, and is referred to herein by general reference numeral **700**.

[0075] Real-time cross-channel monitoring uses track cross channel and cross product patterns to cross pollinate information for more accurate decisions. Such track not only the channel where the fraud ends but also the initiating channel to deliver a holistic fraud monitoring. A standalone internet banking fraud solution will allow a transaction if it is within its limits, however if core banking is in picture, then it will stop this transaction, as we additionally know the source of funding of this account (which mostly in missing in internet banking).

[0076] In FIG. **3**, a variety of selected applied fraud models **316-323** represent the applied fraud models **114** that result with different settings of filter switch **306**. A real-time cross-channel monitoring payment network server can be constructed by running several of these selected applied fraud models **316-323** in parallel.

[0077] FIG. **7** represents a real-time cross-channel monitoring payment network server **700**, in an embodiment of the present invention. Each customer or accountholder of a financial institution can have several very different kinds of accounts and use them in very different transactional channels. For example, card-present, domestic, credit card, contactless, and high risk MCC channels. So in order for a cross-channel fraud detection system to work at its best, all the transaction data from all the channels is funneled into one pipe for analysis.

[0078] Real-time transactions and authorization requests data is input and stripped of irrelevant datapoints by a process **702**. The resulting relevant data is time-stamped in a process **704**. The 15-minute vector process of FIG. **6** may be engaged at this point in background. A bus **706** feeds the data in parallel line-by-line, e.g., to a selected applied fraud channel model for card present **708**, domestic **709**, credit **710**, contactless **711**, and high risk MCC **712**. Each can pop an exception to the current line input data with an evaluation flag or score **718-722**. The involved accountholder is understood.

[0079] These exceptions are collected and analyzed by a process **724** that can issue warning feedback for the profiles maintained for each accountholder. Each selected applied fraud channel model **708-712** shares risk information about particular accountholders with the other selected applied fraud models **708-712**. A suspicious or outright fraudulent transaction detected by a first selected applied fraud channel model **708-712** for a particular customer in one channel is cause for a risk adjustment for that same customer in all the other applied fraud models for the other channels.

[0080] Exceptions **718-722** to an instant transactions on bus **706** trigger an automated examination of the customer or accountholder involved in a profiling process **724**, especially with respect to the 15-minute vectors and activity in the other channels for the instant accountholder. A client tuning input **726** will affect an ultimate accountholder fraud scoring output **728**, e.g., by changing the respective risk thresholds for genuine-suspicious-fraudulent.

[0081] A corresponding set of warning triggers **73-734** is fed back to all the applied fraud channel models **708-712**. The compromised accountholder result **728** can be expected to be a highly accurate and early protection warning.

[0082] In general, a process for cross-channel financial fraud protection comprises training a variety of real-time, risk-scoring fraud models with training data selected for each from a common transaction history to specialize each member in the monitoring of a selected channel. Then arranging the variety of real-time, risk-scoring fraud models after the training into a parallel arrangement so that all receive a mixed channel flow of real-time transaction data or authorization requests. The parallel arrangement of diversity trained real-time, risk-scoring fraud models is hosted on a network server platform for real-time risk scoring of the mixed channel flow of real-time transaction data or authorization requests. Risk thresholds are immediately updated for particular accountholders in every member of the parallel arrangement of diversity trained real-time, risk-scoring fraud models when any one of them detects a suspicious or outright fraudulent transaction data or authorization request for the accountholder. So, a compromise, takeover, or suspicious activity of the accountholder's account in any one channel is thereafter prevented from being employed to perpetrate a fraud in any of the other channels.

[0083] Such process for cross-channel financial fraud protection can further comprise steps for building a population of real-time and a long-term and a recursive profile for each the accountholder in each the real-time, risk-scoring fraud models. Then during real-time use, maintaining and updating the real-time, long-term, and recursive profiles for each accountholder in each and all of the real-time, risk-scoring fraud models with newly arriving data. If during real-time use a compromise, takeover, or suspicious activity of the accountholder's account in any one channel is detected, then updating the real-time, long-term, and recursive profiles for each accountholder in each and all of the other real-time, risk-scoring fraud models to further include an elevated risk flag. The elevated risk flags are included in a final risk score calculation **728** for the current transaction or authorization request.

[0084] The 15-minute vectors described in FIG. **6** are a way to cross pollenate risks calculated in one channel with the others. The 15-minute vectors can represent an amalgamation of transactions in all channels, or channel-by channel. Once a 15-minute vector has aged, it can be shifted into a 30-minute vector, a one-hour vector, and a whole day vector by a simple shift register means. These vectors represent velocity counts that can be very effective in catching fraud as it is occurring in real time.

[0085] In every case, embodiments of the present invention include adaptive learning that combines three learning techniques to evolve the artificial intelligence classifiers, e.g., **408-414**. First is the automatic creation of profiles, or smart-agents, from historical data, e.g., long-term profiling. See FIG. **3**. The second is real-time learning, e.g., enrichment of the smart-agents based on real-time activities. See FIG. **4**. The third is adaptive learning carried by incremental learning algorithms. See FIG. **7**.

[0086] For example, two years of historical credit card transactions data needed over twenty seven terabytes of database storage. A smart-agent is created for each individual card in that data in a first learning step, e.g., long-term profiling. Each profile is created from the card's activities and transactions that took place over the two year period. Each profile for each smart-agent comprises knowledge extracted field-by-field, such as merchant category code (MCC), time, amount for an mcc over a period of time, recursive profiling, zip codes, type of merchant, monthly aggregation, activity during the week, weekend, holidays, Card not present (CNP) versus card present (CP), domestic versus cross-border, etc. this profile will highlights all the normal activities of the smart-agent (specific card).

[0087] Smart-agent technology has been observed to outperform conventional artificial and machine learning technologies. For example, data mining technology creates a decision tree from historical data. When historical data is applied to data mining algorithms, the result is a decision tree. Decision tree logic can be used to detect fraud in credit card transactions. But, there are limits to data mining technology. The first is data mining can only learn from historical data and it generates decision tree logic that applies to all the cardholders as a group. The same logic is applied to all cardholders even though each merchant may have a unique activity pattern and each cardholder may have a unique spending pattern.

[0088] A second limitation is decision trees become immediately outdated. Fraud schemes continue to evolve, but the decision tree was fixed with examples that do not contain new fraud schemes. So stagnant non-adapting decision trees will fail to detect new types of fraud, and do not have the ability to respond to the highly volatile nature of fraud.

[0089] Another technology widely used is "business rules" which requires actual business experts to write the rules, e.g., if-then-else logic. The most important limitations here are that the business rules require writing rules that are supposed to work for whole categories of customers. This requires the population to be sliced into many categories (students, seniors, zip codes, etc.) and asks the experts to provide rules that apply to all the cardholders of a category.

[0090] How could the US population be sliced? Even worse, why would all the cardholders in a category all have the same behavior? It is plain that business rules logic has built-in limits, and poor detection rates with high false positives. What should also be obvious is the rules are outdated as soon as they are written because conventionally they don't adapt at all to new fraud schemes or data shifts.

[0091] Neural network technology also limits, it uses historical data to create a matrix weights for future data classification. The Neural network will use as input (first layer) the historical transactions and the classification for fraud or not as an output). Neural Networks only learn from past transactions and cannot detect any new fraud schemes (that arise daily) if the neural network was not re-trained with this type of fraud. Same as data mining and business rules the classification logic learned from the historical data will be applied to all the cardholders even though each merchant has a unique activity pattern and each cardholder has a unique spending pattern.

[0092] Another limit is the classification logic learned from historical data is outdated the same day of its use because the fraud schemes changes but since the neural network did not learn with examples that contain this new type of fraud schemes, it will fail to detect this new type of fraud it lacks the ability to adapt to new fraud schemes and do not have the ability to respond to the highly volatile nature of fraud.

[0093] Contrary to previous technologies, smart-agent technology learns the specific behaviors of each cardholder and create a smart-agent that follow the behavior of each cardholder. Because it learns from each activity of a cardholder, the smart-agent updates the profiles and makes effective changes at runtime. It is the only technology with an ability to identify and stop, in real-time, previously unknown fraud schemes. It has the highest detection rate and lowest false positives because it separately follows and learns the behaviors of each cardholder.

[0094] Smart-agents have a further advantage in data size reduction. Once, say twenty-seven terabytes of historical data is transformed into smart-agents, only 200-gigabytes is needed to represent twenty-seven million distinct smart-agents corresponding to all the distinct cardholders.

[0095] Incremental learning technologies are embedded in the machine algorithms and smart-agent technology to continually re-train from any false positives and negatives that occur along the way. Each corrects itself to avoid repeating the same classification errors. Data mining logic incrementally changes the decision trees by creating a new link or updating the existing links and weights. Neural networks update the weight matrix, and case based reasoning logic updates generic cases or creates new ones. Smart-agents update their profiles by adjusting the normal/abnormal thresholds, or by creating exceptions.

[0096] In real-time behavioral profiling by the smart-agents, both the real-time and long-term engines require high speed transfers and lots of processor attention. Conventional database systems cannot provide the transfer speeds necessary, and the processing burdens cannot be tolerated.

[0097] Embodiments of the present invention include a fast, low overhead, custom file format and storage engine designed to retrieve profiles in real-time with a constant low load and save time. For example, the profiles **336** built in FIG. **3**, and long-term, recursive, and real-time profiles **414** in FIG. **4**.

[0098] Referring now to FIG. **8**, a group of smart agent profiles is stored in a custom binary file **800** which starts with a meta-data section **802** containing a profile definition, and a number of fixed size profile blocks, e.g., **804**, **805**, . . . **806** each containing the respective profiles. Such profiles are individually reserved to and used by a corresponding smart agent, e.g., profile **536** and smart agent **530** in FIG. **5**. Fast file access to the profiles is needed on the arrival of every transaction **502**. In FIG. **5**, account number **504** signals the particular smart agents and profiles to access and that are required to provide a smart agent risk assessment **552** in real-time. For example, an approval or a denial in response to an authorization request message.

[0099] FIG. **9** represents what's inside each such profile, e.g., a profile **900** includes a meta-data **902** and a rolling list of vectors **904**. The meta-data **902** comprises the oldest one's time field **906**, and a record length field **908**. Transaction events are timestamped, recorded, and indexed by a specified atomic interval, e.g., ten minute intervals are typical, which is six hundred seconds. Each vector points to a run of profile datapoints that all share the same time interval, e.g., intervals **910-912**. Some intervals will have no events, and therefor no vectors **904**. Here, all the time intervals less than ninety days old are considered by the real-time (RT) profiles. Ones older than that are amalgamated into the respective long-term (LT) profiles.

[0100] What was purchased and how long ago a transaction for a particular accountholder occurred, and when their other recent transactions occurred can provide valuable insights into whether the transactions the accountholder is presently engaging in are normal and in character, or deviating. Forcing a fraud management and protection system to hunt a conventional database for every transaction a particular random accountholder engaged in is not practical. The accountholders' transactions must be pre-organized into their respective profiles so they are always randomly available for instant calculations. How that is made possible in embodiments of the present invention is illustrated here in FIGS. **5**, **6**, and **8-10**.

[0101] FIG. **10** illustrates a virtual memory system **1000** in which a virtual address representation **1002** is translated into a physical memory address **1004**, and/or a disk block address **1006**.

[0102] Profiling herein looks at events that occurred over a specific span of time. Any vectors that were assigned to events older than that are retired and made available for re-assignment to new events as they are added to the beginning of the list.

[0103] The following pseudo-code examples represent how smart agents (e.g., **412**, **550**) lookup profiles and make behavior deviation computations. A first step when a new transaction (e.g., **502**) arrives is to find the one profile it should be directed to in the memory or filing system.

```
find_profile ( T: transaction, PT : Profile's Type )
Begin
    Extract the value from T for each key used in the routing logic for PT
    Combine the values from each key into PK
    Search for PK in the in-memory index
    If found, load the profile in the file of type PT based on the indexed
    position.
    Else, this is a new element without a profile of type PT yet.
End
```

[0104] If the profile is not a new one, then it can be updated, otherwise a new one has to be created.

```
update_profile ( T: transaction, PT : Profile's Type )
Begin
    find_profile of type PT P associated to T
    Deduce the timestamp t associated to T
    If P is empty, then add a new record based on the atomic interval for
    t
    Else locate the record to update based on t
        If there is no record associated to t yet,
        Then add a new record based on the atomic interval for t
    For each datapoint in the profile, update the record with the values in
    T (by increasing a count, sum, deducing a new minimum, maximum
    ...).
    Save the update to disk
End
compute_profile ( T: transaction, PT : Profile's Type )
Begin
    update_profile P of type PT with T
```

-continued

```
    Deduce the timestamp t associated to T
    For each datapoint DP in the profile,
        Initialize the counter C
        For each record R in the profile P
            If the timestamp t associated to R belongs to the span of time for DR
            Then update C with the value of DB in the record R (by increasing a count, sum,
                deducing a new minimum, maximum ...)
        End For
    End For
    Return the values for each counter C
End
compute_profile ( T: transaction, PT : Profile's Type )
Begin
    update_profile P of type PT with T
    Deduce the timestamp t associated to T
    For each datapoint DP in the profile,
        Initialize the counter C
        For each record R in the profile P
            If the timestamp t associated to R belongs to the span of time for DR
            Then update C with the value of DB in the record R (by increasing a count, sum,
                deducing a new minimum, maximum ...)
        End For
    End For
    Return the values for each counter C
End
```

[0105] The entity's behavior in the instant transaction is then analyzed to determine if the real-time (RT) behavior is out of the norm defined in the corresponding long-term (LT) profile. If a threshold (T) is exceeded, the transaction risk score is incremented.

```
analyze_entity_behavior ( T: transaction )
Begin
    Get the real-time profile RT by calling compute_profile( T, real-time )
    Get the long-term profile LT by calling compute_profile( T, long-term )
    Analyze the behavior of the entity by comparing its current behavior RT to its past behavior LT:
    For each datapoint DP in the profile,
        Compare the current value in RT to the one in LT (by computing the ratio or distance between the values).
            If the ratio or distance is greater than the pre-defined threshold,
            Then increase the risk associated to the transaction T
            Else decrease the risk associated to the transaction T
    End For
    Return the global risk associated to the transaction T
End
```

[0106] The entity's behavior in the instant transaction can further be analyzed to determine if its real-time (RT) behavior is out of the norm compared to its peer groups. defined in the corresponding long-term (LT) profile. If a threshold (T) is exceeded, the transaction risk score is incremented.

[0107] Recursive profiling compares the transaction (T) to the entity's peers one at a time.

```
compare_entity_to_peers ( T: transaction )
Begin
    Get the real-time profile RTe by calling compute_profile( T, real-time )
    Get the long-term profile LTe by calling compute_profile( T, long-term )
    Analyze the behavior of the entity by comparing it to its peer groups:
```

-continued

```
    For each peer group associated to the entity
        Get the real-time profile RTp of the peer: compute_profile( T, real-time )
        Get the long-term profile LTp of the peer: compute_profile( T, long-term )
        For each datapoint DP in the profile,
        Compare the current value in RTe and LTe to the ones in RTp and LTp (by computing the ratio or distance between the values).
            If the ratio or distance is greater than the pre-defined threshold,
            Then increase the risk associated to the transaction T
            Else decrease the risk associated to the transaction T
        End For
    End For
    Return the global risk associated to the transaction T
End
```

[0108] Each attribute inspection will either increase or decrease the associated overall transaction risk. For example, a transaction with a zipcode that is highly represented in the long term profile would reduce risk. A transaction amount in line with prior experiences would also be a reason to reduce risk. But an MCC datapoint that has never been seen before for this entity represents a high risk. (Unless it could be forecast or otherwise predicted.)

[0109] One or more datapoints in a transaction can be expanded with a velocity count of how-many or how-much of the corresponding attributes have occurred over at least one different span of time intervals. The velocity counts are included in a calculation of the transaction risk.

[0110] Transaction risk is calculated datapoint-by-datapoint and includes velocity count expansions. The datapoint values that exceed a normative point by a threshold value increment the transaction risk. Datapoint values that do not exceed the threshold value cause the transaction risk to be decremented. A positive or negative bias value can be added that effectively shifts the threshold values to sensitize or desensitize a particular datapoint for subsequent transactions related to the same entity. For example, when an airline expense is certain to be followed by a rental car or hotel expense in a far away city. The MCC's for rental car and hotel expenses are desensitized, as are datapoints for merchant locations in a corresponding far away city.

[0111] FIG. **11** illustrates an example of a profile **1100** that spans a number of time intervals t_0 to t_8. Transactions, and therefore profiles normally have dozens of datapoints that either come directly from each transaction or that are computed from transactions for a single entity over a series of time intervals. A typical datapoint **1110** velocity counts the number of events that have occurred in the last thirty minutes (count **1112**), the last six hours (count **1114**), and the last twenty-four hours (count **1116**). In this example, t_0 had one event, t_1 had 3 events, t_2 had 2 events, t_3 had 3 events, t_4 had 2 events, t_5 had 5 events, t_6 had 3 events, t_7 had one event, and t_8 had 2 events; therefore, t_2 count **1112**=6, t_5 count **1114**=16, and t_7 count **1116**=20. These three counts, **1112-1116** provide their velocity count computations in a simple and quick-to-fetch summation.

[0112] FIG. **12** illustrates a behavioral forecasting aspect of the present invention. A forecast model **1200** engages in a real-time analysis **1202**, consults a learned past behavior **1204**, and then makes a behavioral prediction **1206**. For example, the real-time analysis **1202** includes a flight purchase for $1410.65, an auto pay for cable for $149.50, and a hotel for $2318.80 in a most recent event. It makes sense that the booking and payment for a flight would be concomitant with a hotel expense, both represent travel. Consulting the learned past behavior **1204** reveals that transactions for flights and hotels has also been accompanied by a car rental. So an easy forecast for a car rental in the near future is and easy and reasonable assumption to make in behavioral prediction **1206**.

[0113] Normally, an out-of-character expense for a car rental would carry a certain base level of risk. But if it can be forecast one is coming, and it arrives, then the risk can reduced since it has been forecast and is expected. Embodiments of the present invention therefore temporarily reduce risk assessments in the future transactions whenever particular classes and categories of expenses can be predicted or forecast.

[0114] In another example, a transaction to pay tuition at a local college could be expected to result in related expenses. So forecasts for bookstore purchases and ATM cash withdrawals at the college are reasonable. The bottom-line is fewer false positives will result.

[0115] FIG. **13** illustrates a forecasting example 1300. A smart agent profile **1302** has several datapoint fields, field **1** through field n. Here we assume the first three datapoint fields are for the MCC, zipcode, and amount reported in a new transaction. Several transaction time intervals spanning the calendar year include the months of January . . . December, and the Thanksgiving and Christmas seasons. In forecasting example 1300 the occurrence of certain zip codes is nine for 94104, seven for 94105, and three for 94110. Transaction amounts range $5.80 to $274.50 with an average of $84.67 and a running total of $684.86.

[0116] A first transaction risk example 1304 is timestamped Dec. 5, 2013 and was for an unknown grocery store in a known zipcode and for the average amount. The risk score is thus plus, minus, minus for an overall low-risk.

[0117] A second transaction risk example 1306 is also timestamped Dec. 5, 2013 and was for a known grocery store in an unknown zipcode and for about the average amount. The risk score is thus minus, plus, minus for an overall low-risk.

[0118] A third transaction risk example 1306 is timestamped Dec. 5, 2013, and was for an airline flight in an unknown, far away zipcode and for almost three times the previous maximum amount. The risk score is thus triple plus for an overall high-risk. But before the transaction is flagged as suspicious or fraudulent, other datapoints can be scrutinized.

[0119] Each datapoint field can be given a different weight in the computation in an overall risk score.

[0120] In a forecasting embodiment of the present invention, each datapoint field can be loaded during an earlier time interval with a positive or negative bias to either sensitize or desensitize the category to transactions affecting particular datapoint fields in later time intervals. The bias can be permanent, temporary, or decaying to none.

[0121] For example, if a customer calls in and gives a heads up they are going to be traveling next month in France, then location datapoint fields that detect locations in France in next month's time intervals can be desensitized so that alone does not trigger a higher risk score. (And maybe a "declined" response.)

[0122] Some transactions alone herald other similar or related ones will follow in a time cluster, location cluster, and/or in an MCC category like travel, do-it-yourself, moving, and even maternity. Still other transactions that time cluster, location cluster, and/or share a category are likely to reoccur in the future. So a historical record can provide insights and comfort.

[0123] FIG. **14** represents the development, modeling, and operational aspects of a single-platform risk and compliance embodiment of the present invention that depends on millions of smart agents and their corresponding behavioral profiles. It represents an example of how user device identification (Device ID) and profiling is allied with accountholder profiling and merchant profiling to provide a three-dimensional examination of the behaviors in the penumbra of every transaction and authorization request. The development and modeling aspects are referred to herein by the general reference numeral **1400**. The operational aspects are referred to herein by the general reference numeral **1402**. In other words, compile-time and run-tine.

[0124] The intended customers of embodiments of the present invention are financial institutions who suffer attempts by fraudsters at payment transaction fraud and need fully automated real-time protection. Such customers provide the full database dossiers **1404** that they keep on their authorized merchants, the user devices employed by their accountholders, and historical transaction data. Such data is required to be accommodated in any format, volume, or source by an application development system and compiler (ADSC) **1406**. ADSC **1406** assists expert programmers to use a dozen artificial intelligence and classification technologies **1408** they incorporate into a variety of fraud models **1410**. This process is more fully described in U.S. patent application Ser. No. 14/514,381, filed Oct. 15, 2014 and titled, ARTIFICIAL INTELLIGENCE FRAUD MANAGEMENT SOLUTION. Such is fully incorporated herein by reference.

[0125] One or more trained fraud models **1412** are delivered as a commercial product or service to a single platform risk and compliance server with a real-time scoring engine **1414** for real-time multi-layered risk management. In one perspective, trained models **1412** can be viewed as efficient and compact distillations of databases **1404**, e.g., a **100**:**1** reduction. These distillations are easier to store, deploy, and afford.

[0126] During operation, real-time scoring engine **1414** provides device ID and clickstream analytics, real-time smart agent profiling, link analysis and peer comparison for merchant/internal fraud detection, real-time cross-channel fraud prevention, real-time data breach detection and identification device ID and clickstream profiling for network/device protection.

[0127] A real-time smart agent profiling engine **1416** receives behavioral digests of the latest transactions **1418** and uses them to update three populations of profiles **1420-1422**. Specifically, a population of card profiles **1420**, a population of merchant profiles **1421**, and a population of device profiles **1422** all originally generated by ADSC **1406** and included in the trained models **1412**. These are all randomly and individually consulted in real-time by smart agent profiling engine **1416** to understand what is "normal" for a particular card, merchant, and user device.

[0128] Real-time smart agent profiling engine **1416** accepts customer transaction data and scores each line. Such scores are in accordance with business rules provided by a business rules management system (BRMS) **1424** and any adaptive updates **1426** needed to the original set of models **1410** produced by artificial intelligence technologies and classifiers **1408**. A web-based case management system **1428** uses false positives and false negatives to tighten up models **1410**. These are periodically used to remotely update models **1412**.

[0129] In general smart agent process embodiments of the present invention generate a population of smart agent profiles by data mining of historical transaction data. A corresponding number of entities responsible for each transaction are sorted and each are paired with a newly minted smart agent profile. Each smart agent profile so generated is modelled to collect and list individual and expanded attributes of said transactions in one column dimension and by time interval series in another row dimension. Each smart agent profile is stored in a file access system of a network server platform.

[0130] Each newly arriving transaction record is compared and contrasted attribute-by-attribute with the time interval series of attributes archived in its paired smart agent profile, and each such comparison and contrast incrementally increases or decreases a computed fraud risk score. The computed fraud risk score is thereafter output as a determination of whether the newly arriving transaction record represents a genuine transaction, a suspicious transaction, or a fraudulent transaction. Or maybe just OK-bad, or a fuzzy score between 0 . . . 1.

[0131] Each time interval series can be partitioned or divided in its row dimension into a real-time part and a long-term part to separately pre-compute from the real-time part and the long-term part a velocity count and statistics of said individual and expanded attributes. The newly arriving transaction record is then compared item-by-item to relevant items in each said real-time part and long-term part, and thereby determines if each item represents known behavior or unknown behavior.

[0132] Each newly arriving transaction record is inspected to see if the entity it represents has not yet been paired to a smart agent profile, and if not then generating and pairing a newly minted smart agent profile for it.

[0133] In another embodiment, three populations of smart agent profiles are generated by data mining the historical transaction data. A corresponding number of cardholder, merchant, and identified device entities involved in each transaction are sorted and each are paired with a newly minted smart agent profile. Then, each newly arriving transaction record is compared and contrasted attribute-by-attribute with the time interval series of attributes archived in the smart agent profiles paired with the particular cardholder, and with the particular merchant, and with the particular identified device (Device ID), and each such comparison and contrast incrementally increases or decreases a computed overall fraud risk score. See our U.S. patent application Ser. No. 14/517,863, filed 19 Oct. 2014, and titled User Device Profiling In Transaction Authentications, for details on the Device ID technology we have in mind here.

[0134] The entity Smart Agents herein are data structures used to store attributes which describe the entity it follows with links to attribute Smart Agents whose attributes are the identities of the other Smart Agents that invoke them. For example, a single healthcare provider would be assigned an entity Smart Agent that lists that provider's attributes, e.g., medical qualifications, medical specialty, medical office facility, organization size, etc. Each of those attributes has its own Smart Agent accessible via a link. Some attributes do not change, some change slowly, some signal risk if changed, some change daily, and some, like short term and long term profiles are histories that can be compartmented into the last minute, the last hour, the last day, the last week, the last month, and the last year. The attribute Smart Agents link back to the entity Smart Agents that are described by them.

[0135] For example, the attribute Smart Agent for medical specialty will link back to all the entity Smart Agents with the medical specialty. This would help cross check one healthcare provider with another that shared the same medical specialty, medical office facility, organization size, etc. E.g., peer groups by selectable, single dimensions.

[0136] Variations of the present invention will be apparent to those of ordinary skill in the art in view of the disclosure contained herein. For example, specialized tools and modules, e.g., in the form of software, computer programs, or circuitry, may be developed to allow programmers and administrators to set up systems and processes or methods in accordance with the invention.

[0137] In any case, it should be noted that any particular embodiment of the invention may be modified to include or exclude features of other embodiments as appropriate without departing from the spirit of the invention. It is also believed that principles such as "economies of scale" and "network effects" are applicable to the invention and that synergies arising from the invention's novelty and non-obviousness increase when the invention is practiced with increasing numbers of individuals, entities, users, and/or institutions. Appropriate usage of computerized and/or communication means, e.g., web-based hardware and/or software, cellular and land-based telephonic equipment, and antenna-based, satellite and coaxial and/or ethernet cable/wire technologies, allow for further synergies, thereby rendering the invention more non-obvious that that described in the printed references that do not disclose the above-identified computerized and/or communication means.

[0138] It is to be understood that, while the invention has been described in conjunction with the preferred specific embodiments thereof, the foregoing description merely illustrates and does not limit the scope of the invention. Numerous alternatives and equivalents exist which do not depart from the invention set forth above. Other aspects, advantages, and modifications within the scope of the invention will be apparent to those skilled in the art to which the invention pertains.

[0139] All patents and publications mentioned herein are hereby incorporated by reference in their entireties to the fullest extent not inconsistent with the description of the invention set forth above.

What is claimed is:

1. An Internet-of-things (IoT) component, comprising:

an electronic smart-agent data structure resident in a computer memory, and accessible through a communications device connectable to the Internet;

a means for associating data related to a particular thing or entity to the smart-agent data structure;

a means for collecting data and attributes descriptive of or related to said thing or entity and packing such into the smart-agent data structure;

a means for rendering a descriptive representation or behavioral analysis of the thing or entity from the data kept within the smart-agent data structure; and

a profile for storing in an array any historical data related to said particular thing or entity.

2. The IoT component of claim **1**, further comprising:

means for operating alone while disposed within a single said particular thing or entity; and

means for communicating said profile, data and attributes with other smart-agent data structures resident in different locations.

* * * * *