A Horizontal Platform for Creating M2M Applications

Overview

The world is moving toward an environment where there will be billions of devices interconnected via the internet. In a bold statement Ericcson president Hans Vestberg has predicted that the number of connected devices will increase from 5 billion today to 50 billion by 2020.[1] According to Ericcson, the current fragmented market, which is full of one-off deployments and specialized solutions, will expand by a factor of ten to include a diverse set of new use cases such as smart meters, point-of sale devices, special sensors on people's clothing, universal tracking of vehicles, supplies, and material. M2M sensors will track the position and movement of people, measure levels in tanks, monitor temperature, track inventory levels, and report back on all sorts of information that can be used to increase business efficiency. A series of technical challenges will need to be addressed to achieve this ambitious vision.

The technical challenge for the device makers is to produce a massive number of new and varied sensors. The challenge for carriers is to provide conductivity and device management for 50 billion connected devices. The challenge for software application developers is to write a massive number of new cloud-based applications that accept wireless data streams and convert this information into business value. The current approach for writing M2M applications, where each solution is custom-built as a one-off, will not scale to meet the business needs of a world with 50 billion interconnected devices.

To produce these new applications, the software community needs standardized development platforms with reusable modules that can be combined to produce high-quality applications. The development environment needed looks something like the following:

- Real-time spatial database to track the current and historical locations of mobile assets.
- Rules engine to trigger real-time alerts, for example when drivers break a geo-fence or leave the engine idling. These are but the simplest rules, advanced applications will undoubtedly require sophisticated analyses and alerts.
- Visualizations to enable users to make sense of the information collected by these devices.
- Statistical Analytics to reduce the complexity in the data and thereby make the information actionable.
- Dashboard authoring environments to enable developers, power users, and customers to create new M2M applications.
- Optimizations to produce efficiencies and enable actions that create value.
- Flexible tools to access and integrate in external data sources from external data providers.

[1] http://gigaom.com/2010/04/14/ericsson-sees-the-internet-of-things-by-2020/

- Real-time analytics to enable real-time actions and historical analytics to identify underlying business trends.
- Software Agents that run on mobile devices.
- APIs to enable custom applications to be built.

At VisTracks we have built an M2M application development platform. Our platform is particularly strong in Visualization and Analytics. We offer a compelling dashboard and authoring environment which makes it easy to create new M2M applications. Using our platform enables solution providers to use high quality implementations of pre-built software components, which improves software quality, lowers development costs, and decreases time to market.

M2M Service Layers

As represented in Figure 1, four layers are emerging within the M2M technology stack:

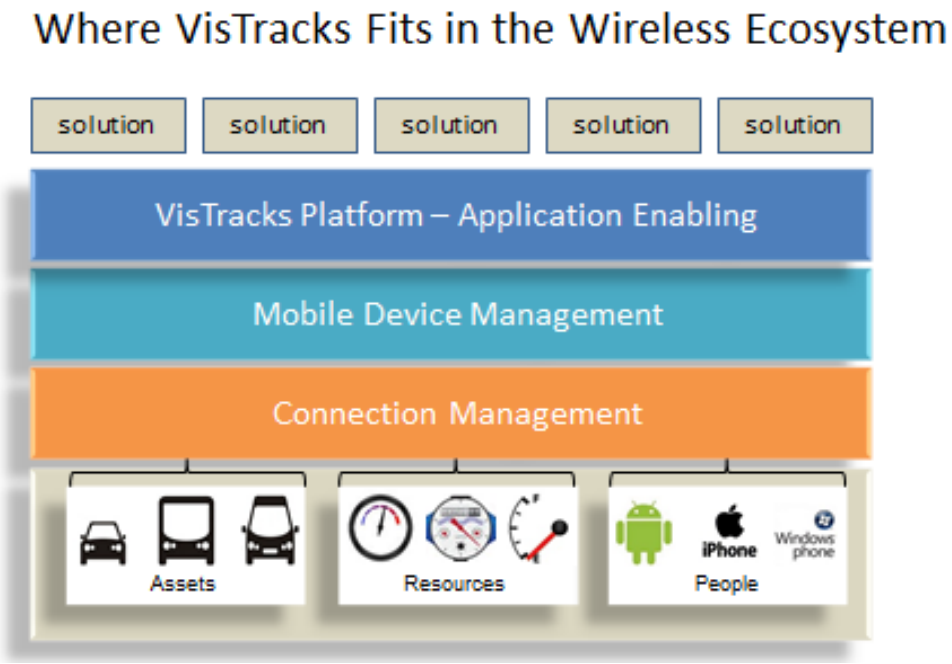


Figure 1. Technology Stack for M2M Applications

1. Devices – Vendors are creating all sorts of new sensors for a variety of protocols, e.g. ZigBee, Wi-Fi, etc. The sensors are becoming specialized with longer lasting and better power management. The new devices are smaller, more sophisticated, and less expensive.

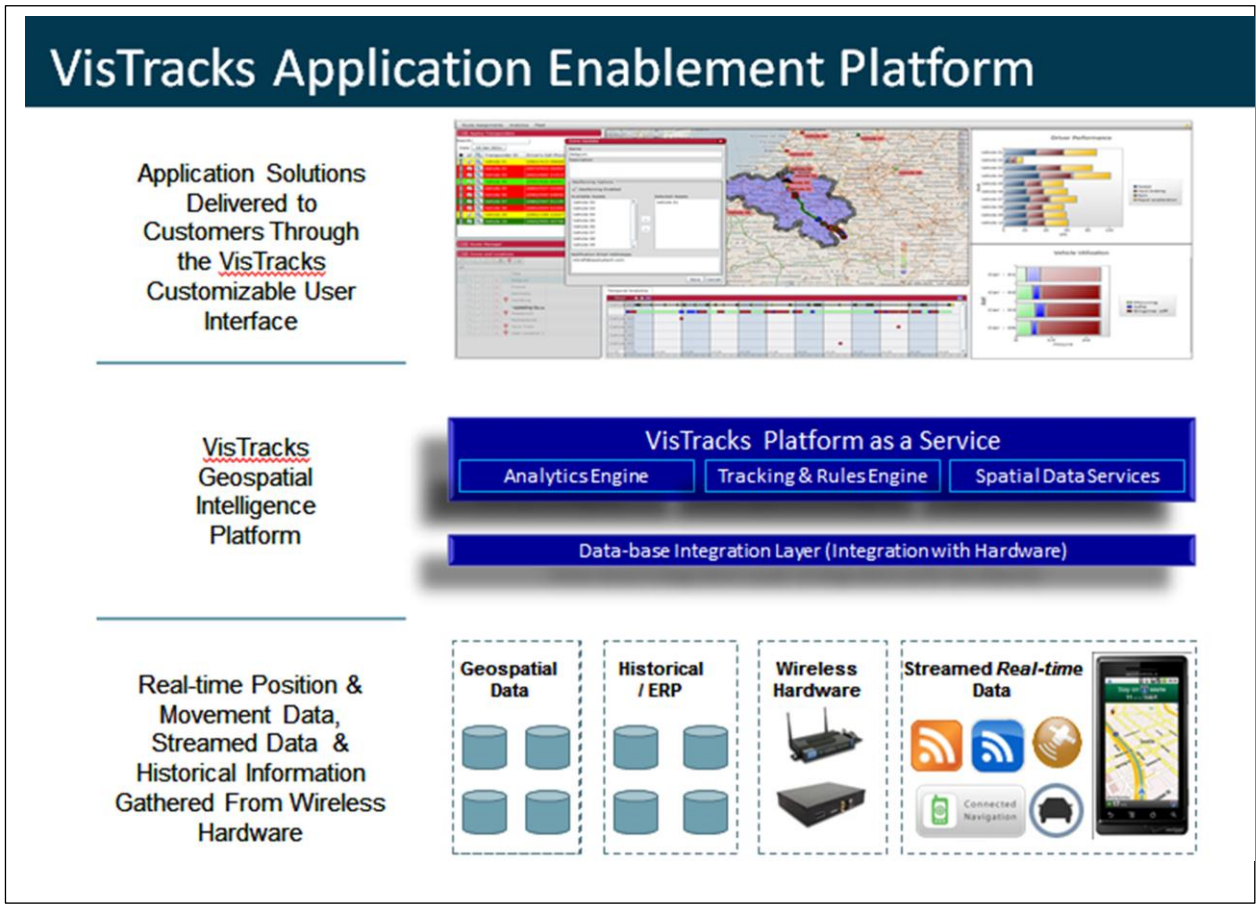
2. Connection Management – Airtime, SIM activation, protocol support, billing, IP address management, globalization, etc. are provided by the carriers and MVNOs.

3. Device Management – Managing 50 billion devices is going to be a serious technical challenge. Device management systems will provide provisioning, remote monitoring, remote diagnostics, access control, over the air updates firmware updates, etc.

4. Application Enablement – All the work done on devices, connection management and device management will not have business payback until an environment and software infrastructure to enable M2M applications is built. The VisTracks platform is an environment for building M2M applications efficiently and cost effectively.

VisTracks M2M Application Development Platform

The VisTracks platform provides the following capabilities as a common software substrate for building M2M applications:



1. Multi-tenant Real-time Data Warehouse

As new wireless data transactions arrive from the sensors, which at VisTracks we call Position Facts, they are processed and organized into a multi-tenant cloud-based data warehouse. The database schema in the VisTracks warehouse is specifically engineered for manipulating position facts. The position fact data warehouse must be scalable and might include hundreds of millions of position facts collected from sensor devices.

2. Data Structures for M2M Applications

The VisTracks platform extends the traditional spatial database by creating new data types for position and movement data. In a spatial database, the base types are points, lines, polygons, shapes, boundaries, etc. The VisTracks platform extends a spatial database by adding new base types for trajectories, stop points, trips, and signatures. These new types are first class database objects, offering a complete set of methods, properties, and algorithms for manipulating the types, extracting information, making comparisons, etc.

Position Facts are sent from the devices to the VisTracks platform. As new position facts arrive, the platform maintains and updates internal data structures that are specifically tailored to this class of data. The position facts records may contain an arbitrary number of name-value pairs which are called attributes.

Position Facts Attributes are optional fields that are associated with the position facts produced by a sensor, smart phone, or other type of sensor that is provisioned on the platform. The attributes might record the complete state of a sensor device through time and are available to application developers through a set of APIs, along with all associated attribute values. These APIs make it easy for developers to create real-time tracking solutions, and identify if the sensor or mobile phone is in a location where it should receive a pre-programmed message.

Geospatial Zones are commonly used to set geo-fences. The zones may be circular, square, or arbitrary polygons. The platform enables users to create, edit, delete and manipulate the zones. Users may use zones and the rules engine to set alerts.

Trajectories are a data structure that captures the historical movements of an asset. A trajectory begins at a *Stop Point*, a location where an object has been stationary for a fixed time interval and movement distance threshold. A trajectory represents a continuous movement through a sequence of positions, and then ends at another stop point. The platform automatically calculates trajectories and stop points as each new Position Fact arrives. The attributes from the Position Facts are associated with each position along the trajectory. Also, the platform calculates various statistics on the trajectories such as average, minimum, and maximum speeds, distance travelled, etc. The platform provides APIs for calculating various statistics along a trajectory for attributes. For example, the statistics might include the average humidity, condition, state, temperature, pressure, weight, and shrinkage through time.

Stop points are locations where the moving entity was stationary for a fixed period. There is a settable threshold, e.g., 15 minutes, which is used by the platform to determine whether a short stop is really a temporary pause in the movement, such as might be caused by traffic, or a true stop point. Stop points are useful to understand pauses in movement, which often represent friction points. A pattern of excessively long stop points at a delivery location may indicate a loading and unloading inefficiency that needs to be addressed.

3. Statistical Analysis

VisTracks integrates with the R statistical language to provide deep statistical analysis. Analysts can program statistical algorithms using R, and thereby correlate position and movement data with other external data types.

Real-time analytics are needed to support real-time actions. For M2M applications that data arrives as a stream and statistical operations must operate on the stream to enable real-time actions. The analytics might calculate a statistical score or generate a new variable which may be passed to a rules engine for immediate action.

Historical analytics are useful for identifying underlying trends in data and spotting outliers. The VisTracks platform supports both flavors of analysis.

Position and Movement Signatures are calculated for each moving entity. The movement signatures are probabilistic representations of the movement patterns and can be used to classify the different types of movement. For example, a movement signature that shows a week day pattern of movement from the Chicago suburbs downtown and back is characteristic of a commuter.

The concept of signatures is more valuable than traditional threshold-based analysis because a signature incorporates many dimensions of movement and there allows more flexibility in categorizing and predicting movement patterns. Conversely, a threshold-based classification is inherently one dimensional and therefore quite limited. For example, one use case for movement signatures is to identify patterns of excessively aggressive driving, which might indicate an insurance risk. Aggressive driving can only be identified in context of the situation when considering multiple dimensions including the driving patterns of the other vehicles, maximum speed, acceleration, fast lane changes, hard braking events, high speeds during turns, fast decelerations, etc. A simple threshold-based system cannot accurately identify aggressive driving because it does not include the full context of the activity.

Pattern libraries are useful to organize and segment movement patterns. For example, one type of pattern library might describe movement patterns for daily commuters. Another type might describe the movement patterns of college students or perhaps young stay-at-home mothers.

4. Route Optimization and Logistics

The VisTracks platform provides a logistics engine for vehicle routing optimization. Vehicle routing logistics is extremely complicated. Organizations such as UPS and FedEx have built massively sophisticated and expensive vehicle routing and optimization systems. These systems schedule pickups, plan routes, manage exceptions, handle emergency pickups, minimize driver mileage, include optimizations for traffic and road construction, and calculate the best tradeoffs between driver and vehicle expenses. Sophisticated logistics systems, although useful, are expensive and complicated to implement and run. By using the latest technology, VisTracks provides much of this capability with much less effort and at far lower cost.

5. Programmable Rules Engine for Complex Event Processing

The platform provides a programmable rules engine with complex logic that enables developers to create guided or arbitrary rules to create actions. For example, the actions might be to stand up an alert, send a mobile subscriber a SMS, trigger an alarm, send an email message, generate a list, etc.

In contrast with the rules engine for a simple tracking solution or location-based services, what is unique about the VisTracks rules engine is that it supports complex event processing on the full history of movement of an entity. A complex rule, for example, might involve which POIs the trajectory of an entity passed, its speed when it passed the POIs, time of day, etc.

6. Real-time Visual Analytics Dashboard

VisTracks provides a configurable visual analytics dashboard as a dynamic presentation interface to do quick visual analysis of position and movement data. The dashboard is fully configurable, real-time, and enables customers to quickly access statistics associated with the most recent information streaming into the system. The dashboard includes maps, timelines, bar charts, scatterplots, and other interactive tools to support ad hoc analysis.

The authoring environment for the VisTracks dashboard makes it easy for developers, power users, and customers to create new applications. Creating a new application involves dropping widgets onto a layout grid, creating data sources, and binding the data sources to the widgets. The VisTracks dashboard is live. The data sources and widgets automatically update when new data arrives.

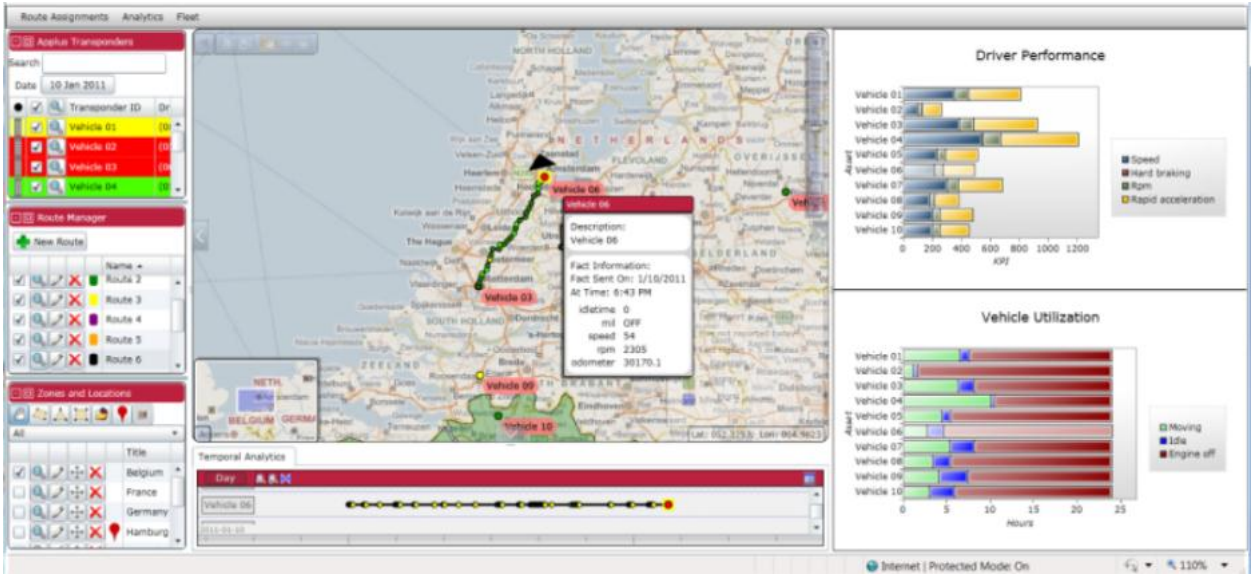


Figure 2. VisTracks real-time visual analytics dashboard.

The VisTracks dashboard is extensible in two ways. Developers can create new data sources, using either query data sources created with SQL queries, R code, or generic data read from Internet sources. The second way the dashboard can be extended is by adding new widgets. Using the VisTracks APIs developers can author their own widgets and include them into the VisTracks platform.

7. Software Agents for Mobile Devices

VisTracks also provides Android and iPhone agents that run on a mobile device and communicates back to our servers. The agent can be remotely configured to send geo-fence alerts, detect acceleration, and report back with other attributes. The interplay between the agents on the mobile device and our platform enables applications to be built quickly.

8. Platform APIs

The VisTracks platform APIs are the mechanism that developers use to create applications using the VisTracks platform. There are several classes of APIs

1. **Simple APIs** – these are commodity APIs that all platforms must provide to support basic operations including real-time tracking applications.

2. **Statistical Analytics –** this set of APIs provides analytical functions including statistical analysis and correlation.

3. **Routing and Optimization –** this set of APIs is for calculating routes and optimizing movement patterns.

4. **Rules Engine –** this set of APIs enables a developer to manipulate the VisTracks rules engine

5. **Visual Analytics Dashboard –** these APIs enable the developer to generate custom dashboard visualizations and APIs.

Conclusion

VisTracks has created a next generation cloud-based PaaS system for solution providers to create real-time M2M applications. The platform extends spatial databases by providing new data structures for position and movement data, provides analysis and optimization, and includes a flexible authoring environment for creating dashboard application. The platform reduces the time needed to develop and increases the quality of M2M applications.