# Safe Step: A Real-time GPS Tracking and Analysis System for Criminal Activities using Ankle Bracelets

Mohammed Daubal[1], Olajumoke Fajinmi[1], Lars Jangaard[1],
Niko Simonson[1], Brett Yasutake[1], Joe Newell[2], Mohamed Ali[3]

[1]University of Washington, {daubalm, ofajinmi, larsjang, nikouw, brett}@uw.edu
[2]BI Incorporated, Joe.Newell@bi.com
[3]Microsoft, mali@microsoft.com

## ABSTRACT
Court orders may require supervising agencies (probation, parole, pre-trial or law enforcement) to monitor a specific set of offenders using ankle bracelets. These ankle bracelets transmit the offender's GPS coordinates at regular intervals. The analysis of the offender's GPS data help: (1) detect unauthorized activities in real time and provide alerts to a community corrections officer, law enforcement dispatcher or a control center, and (2) mine for the offenders' suspicious behavior and predict probable future threats beforehand. Unauthorized activities include protecting geographically defined regions (e.g., school zones) in which the offender is not allowed to be present. Suspicious behaviors include the meeting of offenders with each other on a regular basis, possibly near restricted zones.

This demo presents "*Safe Step*", a real time tracking and analysis system for GPS data streamed out of ankle bracelets. The system's back end is a data stream processing system that collects data from multiple ankle bracelets, correlates the past and the current GPS locations together to analyze the offender's individual behavior as well as the interaction between offenders. The system's front end utilizes a web application interface that displays the current locations of offenders on a map. It also pushes to the user various alert notifications that have been generated by the system's backend. The system's back end is built using Microsoft StreamInsight and Microsoft SQL Server Spatial Libraries while the visualization in the system's front end utilizes Bing Maps.

## Categories and Subject Descriptors
H.2.4 [**DATABASE MANAGEMENT**]: Systems – Query Processing, Relational databases.

## General Terms
Algorithms, Design, Theory.

## Keywords
Data Streaming, SQL Server, StreamInsight, Spatio-temporal, Complex event processing, Analytics, GPS, Ankle bracelets.



## 1. INTRODUCTION
With the popularization of GPS technologies, the criminal justice system uses GPS systems to continuously track specific types of offenders. While they are placed on house arrests, under probation or awaiting trial, offenders are sometimes required to wear a mobile tracking device in the form of an ankle bracelet. GPS tracking of offenders has proven effective in reducing the population of prison inmates, cutting down the cost of building new prisons and lowering the overall expenses spent on the prison system. In contrast to imprisoning the offender, GPS tracking systems give the offender a degree of freedom in moving around the community. This degree of freedom, if not managed carefully, can result in a new offense taking place before the supervising agencies are able to take an action.

According to the decision of the criminal justice system, each offender with a tracking device is assigned a designated curfew. This curfew typically consists of *confinement zones* to which the offender is detained to, and a set of *restricted zones* to which he is obliged to stay away from. For example, an offender may be required to stay home at night during a court ordered curfew. Also, a sex offender would be restricted from visiting school zones. Offenders are free to move around without the monitoring agencies being alerted as long as they remain within the designated confinement regions and as long as they do not enter restricted zones.

While monitoring the offender's location against confinement and restricted zones is crucial, an offender may commit an offense within the boundaries of the confinement zone. An offender may also enter a restricted zone before a monitoring agency is able to dispatch a law enforcement vehicle or take an action (even if the agency was notified instantaneously). Hence, the prediction of an offender's intent is interesting to prevent an offense and raise a warning beforehand. Analyzing the offender's behavior would help supervising agencies focus on a smaller subset of offenders. For example, an offender who meets other offenders (who are also tracked by the system) on a regular basis can be an indication of the offender's intent to plan and commit a future crime. Also, an offender who loiters around a restricted zone (or a particular area) multiple times may give an indication that the offender is going to violate the terms of his supervision.

This paper presents the *Safe Step* system, a GPS monitoring system that consists of mobile tracking devices. These devices continuously stream the offender's location in the form of (timestamp, id, latitude, longitude) to the system's back end. At the system's backend, real time spatiotemporal data is processed by a set of continuous queries that are posed against the predefined confinement/restricted zones of the offenders. The location data streams across all offenders are also joined and correlated with each

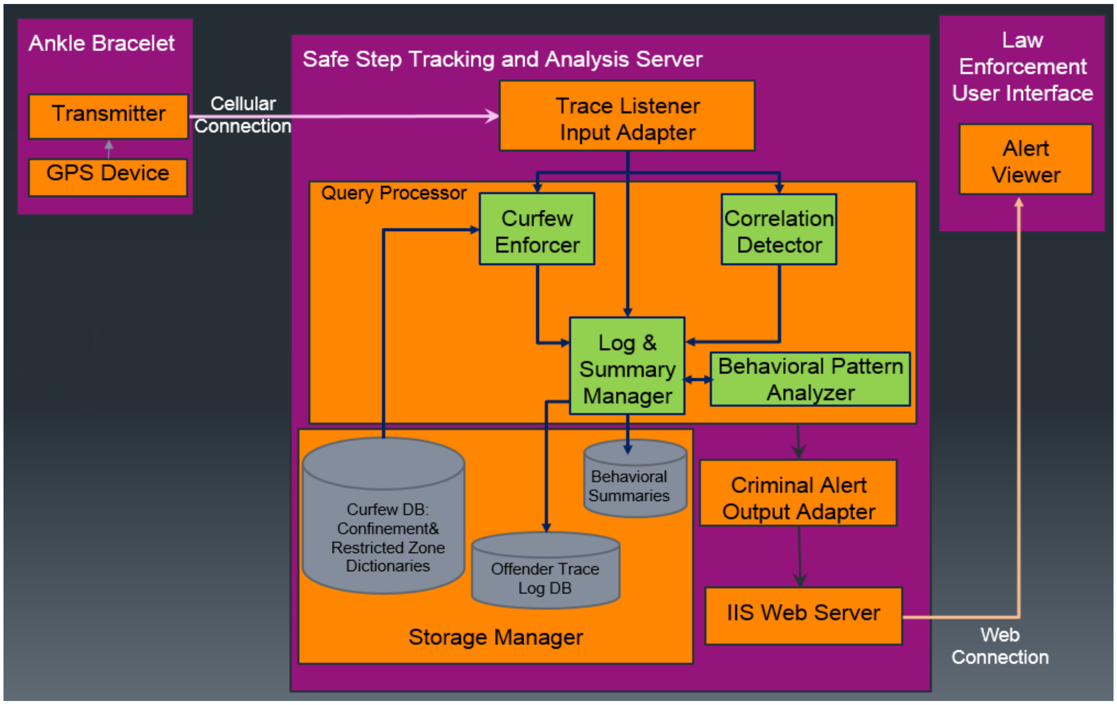


**Figure 1. An architectural overview of the Safe Step tracking system.**

other to determine if offenders meet together and the duration of the meeting. The system's front end is a user interface for monitoring and alerting supervising agencies. The system's back end is powered by *Microsoft StreamInsight* (a real time data stream management system) and *Microsoft SQL Server Spatial Libraries* (a high performance environment for persisting and analyzing spatial data). The system's front end visualizes the result using *Bing Maps*.

The rest of the paper is organized as follows. Section 2 overviews the system's architecture. Section 3 describes the demo scenario. The paper is concluded in Section 4.

## 2. ARCHITECTURE

Figure 1 gives an overview of the system's architecture. An ankle bracelet continuously streams the offender's GPS location over the cellular network to the system's input adapters. The system's input adapters, or the *Trace Listeners*, push the incoming GPS readings to the system's query processor. The system's query processor sends its output to an *Alert Viewer* on the supervising agencies' side using various web protocols.

The system query processor is built using Microsoft StreamInsight[1][2], a real time data stream management system. Microsoft StreamInsight is a platform that monitors stream data and extracts meaningful patterns and trends. StreamInsight analyzes and correlates data from multiple sources incrementally to yield low latency response times. The applicability of StreamInsight in the spatial domain has been studied in [3] and [4]. While StreamInsight is a general purpose data stream management system, its extensibility framework [5] has been explored to implement various real time spatiotemporal operators (e.g., INSIDE, KNN [6], spatial filters and aggregators [7]). The Safe Step Tracking query processor leverages the StreamInsight capabilities to execute a set of spatiotemporal continuous queries against the incoming stream.

The query processor consists of four main components. First, the *Curfew Enforcer* executes a spatial range filter that detects whether the offender's location abides to the confinement and restricted zone regulations. Note that the confinement and restricted zone regulations are specified on a per offender basis and are stored in the *CurfewDB* database that is maintained by the storage manager. CurfewDB is a SQL Server database and the SQL Server Spatial Libraries are invoked to manipulate and load curfew information into the query processor operators [8]. The *Correlation Detector* spatially joins an offender's location with the locations of other offenders over a window of time at real time to detect the possibility of an offenders' meet-up. The raw GPS traces and the output of both the Curfew Enforcer and the Correlation Detector are fed into a *Log and Summary Manager*. The Log and Summary Manager stores the raw GPS traces along with a set of interesting summaries in the *Offender Trace Log DB* and in the *Behavioral Summaries DB*, respectively. Thanks to the high performance dynamic pattern matching operator of StreamInsight [9], current offender locations, play-backs of offender trace logs and behavioral summaries are blended together in the *Behavioral Pattern Analyzer*. The behavioral pattern analyzer carries over the responsibility of monitoring recurrent and frequent places/people that are visited/met by the offender over time. It also mines for outliers in the offender's trajectory (relative to the daily observed trajectories of the same offender). Hence, the behavioral Pattern Analyzer acts as a machine learning module that warns the supervising agencies of changes in the offender's lifestyle, probable future offenses and suspicious behaviors.

## 3. DEMO SCENARIO

The demo scenario is based on GPS traces recorded by volunteer students at the University of Washington, Bothell who agreed to act as offenders for the demo purpose. Given the sensitivity of the actual offenders' GPS data and due to privacy concerns mandated by the government, the recorded GPS traces by our volunteer students are used to mimic the actual offenders' traces. The

University of Washington campus has been declared as a confinement zone where our volunteer students are required to be detained to for the simulation period. Selected buildings in the campus have been declared as restricted zones where our volunteer students are required to stay away from. Absolutely, the volunteer students have been told “not” to fully abide by the declared curfew regulations so the system can be demonstrated under various violations of offenders.

In Figure 2, there are three restricted zones visualized on the map. Restricted zones are represented as red polygons. Once an offender crosses the boundary of a restricted zone, an alert is raised by the system’s backend and pushed to the front end, where it gets displayed on the UI (right pane of Figure 2) and is sent as a message to the control center.

Figure 3 shows another type of alert, the proximity alert, which is triggered when two criminals get too close to each other, e.g., within one meter of each other. As displayed in the right pane, the query can be enriched with more spatiotemporal predicates, aggregation operators and windowing semantics. The alert says that “OffenderID 10 and OffenderID 20 have been within 5 meters of each other within the last 5 seconds”. Thanks to the temporal semantics of the underlying Microsoft StreamInsight server, queries can be declaratively expressed in LINQ [10][11] and executed by the data streaming query processor. The demo will give the audience the opportunity to experience a set of analysis queries in LINQ that reveal similarities among offenders’ trajectories over periods of time.

## 4. CONCLUSION

The criminal justice system uses GPS systems to continuously track specific types of offenders. These offenders are forced by law to wear ankle bracelets that stream their locations to the monitoring agencies. This demo presented a real time data stream processing system that collects the location data of offenders. The system

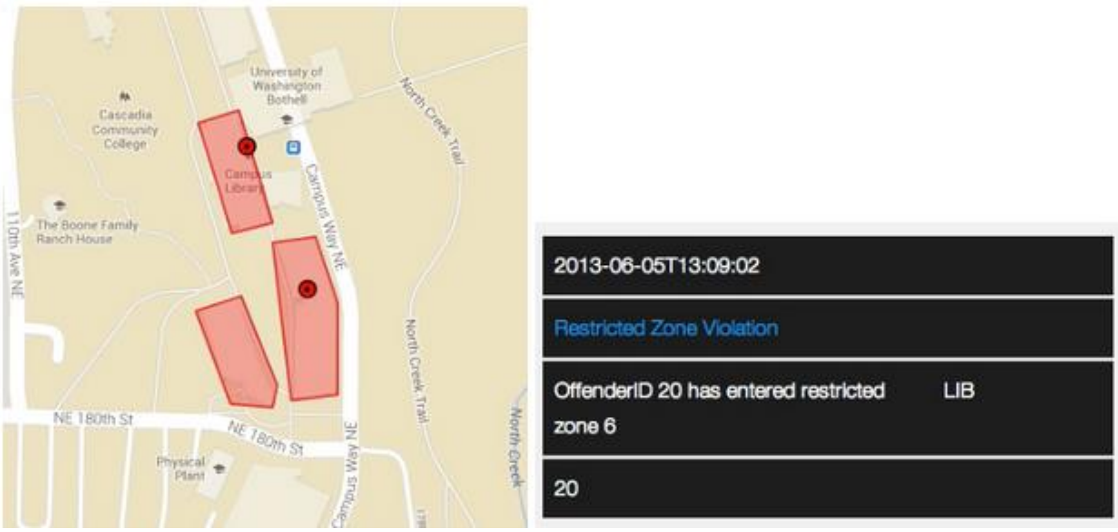


**Figure 2. Restricted Zone Alert, triggered when a criminal enters a restricted zone.**

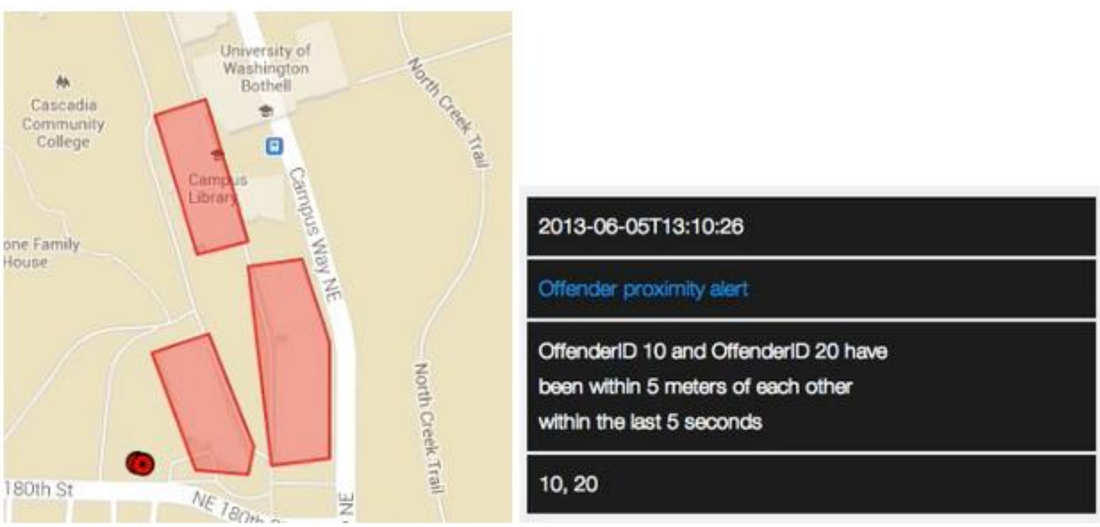


**Figure 3. Proximity Alert, triggered when two or more criminals get close to each other.**

highlights the ability to monitoring and analyze, in real-time, the offenders' GPS traces to detect when offenders violate the court order to stay within the confinement zone or when they enter a restricted zone, e.g. school zone. Moreover, the system analyzes the behavior of an offender and correlates it with the behaviors of other offenders being monitored by the system. For example, the system alerts for a probable future offense in the case that an offender frequently meets with other offenders. The analysis of the offenders' past trajectories reveals a wealth of information that can be utilized to predict and prevent future offenses and, hence, protect the community and reduce costs incurred by supervising agencies.

## REFERENCES


[1] Roger S. Barga, Jonathan Goldstein, Mohamed H. Ali, and Mingsheng Hong. Consistent Streaming Through Time: A Vision for Event Stream Processing. In Proceedings of CIDR, 412-422, 2007.

[2] Jonathan Goldstein, Mingsheng Hong, Mohamed Ali, and Roger Barga. Consistency Sensitive Streaming Operators in CEDR. Technical Report, MSR-TR-2007-158, Microsoft Research, Dec 2007.

[3] Mohamed H. Ali, Badrish Chandramouli, Balan Raman, and Ed Katibah, Spatio-Temporal Stream Processing in Microsoft StreamInsight. IEEE Data Eng. Bull. 33(2): 69-74 (2010).

[4] Mohamed Ali, Badrish Chandramouli, Balan S. Raman, and Ed Katibah, Real-Time Spatio-Temporal Analytics using Microsoft StreamInsight. In Proceedings of the ACM SIGSPATIAL International Conference on Advances in Geographic Information Systems (ACM SIGSPATIAL GIS), November 2010.

[5] Mohamed Ali, Badrish Chandramouli, Jonathan Goldstein, and Roman Schindlauer. The Extensibility Framework in Microsoft StreamInsight. In Proceedings of the International Conference on Data Engineering (ICDE), April 2011.

[6] Jeremiah Miller, Miles Raymond, Josh Archer, Seid Adem, Leo Hansel, Sushma Konda, Malik Luti, Yao Zhao, Ankur Teredesai, and Mohamed Ali, An Extensibility Approach for Spatio-temporal Stream Processing using Microsoft StreamInsight. In Proceedings of the International Symposium on Spatial and Temporal Databases (SSTD), August 2011.

[7] Seyed Jalal Kazemitabar, Ugur Demiryurek, Mohamed H. Ali, Afsin Akdogan, and Cyrus Shahabi, Geospatial Stream Query Processing using Microsoft SQL Server StreamInsight. In Proceedings of the International Conference on Very Large Data Bases (VLDB), Sept. 2010.

[8] Mark Simms. Using SQL server for reference data in a StreamInsight query. Retrieved, 2013, from http://code.msdn.microsoft.com/windowsdesktop/Using-SQL-Server-for-4b4f9c79.

[9] B. Chandramouli, J. Goldstein, and D. Maier. High-Performance Dynamic Pattern Matching over Disordered Streams. In Proceedings of the International Conference on Very Large Data Bases VLDB, 220-231, 2010.

[10] Paolo Pialorsi, Marco Russo. Programming Microsoft LINQ, Microsoft Press, May 2008.

[11] Microsoft LINQ. http://tinyurl.com/42egdn.