

Crime Open Data Aggregation and Management for the Design of Safer Spaces in Urban Environments

Antonios Bonatsos, Lee Middleton, Panos Melas, and Zoheir Sabeur

University of Southampton IT Innovation Centre,
Faculty of Physical Sciences and Engineering, United Kingdom
{zas,ljm,pm}@it-innovation.soton.ac.uk

Abstract. This paper describes the major research and development activities which have been achieved so far since the launch of the DESURBS project (www.desurbs.eu) in 2011. The project focuses on the development of a Decision-Support System Portal (DSSP) which integrates information, data and software modules representing city assets, hazards and processing models that simulate exposures to risks and potential compromise to safety and security. The use of the DSSP will aid the design of safer and more resilient urban spaces. Specifically, it provides security related scenarios with contextual information to support various types of users who specialise in urban spatial design and planning. The DSSP is a web enabled system which is also adapted to mobile devices usage. It is supported with geographic maps and visualised aggregated data from a number of heterogeneous sources. A responsive web design which adapts to the resolution of smart mobile devices has also been achieved. That is, low powered mobiles can still provide map oriented data in a responsive fashion, while using multiple platforms (Android and iOS currently). The first DSSP prototype employs the United Kingdom crime statistics feed of year 2012 and analyses crime trends in 13 English Cities (including Greater London) which are distributed into four major-regions. The DSSP displays raw crime data via a marker on a map, while they are aggregated under specific crime type threads and visualised as “heat maps”. The specific visualisations are aligned to the various administrative regions such as neighbourhoods, catchments and postcodes. It also allows users to explore historical crime trends for a region over time, where crime statistics are contrasted. The scalability of the DSSP was also tested under increasingly large datasets and numbers of users, with tested loads on the map server and the main Django user application. The difference in speed between the mobile and desktop interfaces for a defined set of tasks using the application shall also be performed and presented in the near future.

Keywords: Smart Cities, urban safety, urban crime, data visualisation, decision-support, web-enablement, mobile applications.

1 Introduction

The traditional view of cities is that they bring together “hard” infrastructure to facilitate the lives, activities, security and communication between large and permanent settlement of people in urban zones. This view is becoming less valid since the emergence of digital and mobile technologies in the last few decades and the most recent ICT driven Future Internet concepts of “Smart cities” [5], [2]. For example, security in the smart cities context involves now the management of public spaces with the use of the internet to access to geo-information, overlaid with processed data and information for decision-support [4, 6]. From this technology perspective alone, a key driver which will aid smart cities to be managed at localised levels involves the deployment and use of large scale sensor networks for monitoring and recording crime incidents and beyond [1]. With the pervasive nature of smart mobile devices and their capability to be adopted as sensor platforms [3], sensors can be considered for exploitation as key Future Internet enablers in smart cities [7]. In particular, the generated big sensor data from mobile devices shall become the prime Citizens observatories for aiding local authorities achieve detailed and real-time measurement of crimes at catchment levels in cities. Furthermore, the new generation of decision-support systems for managing crimes in smart cities shall require access to data generated by citizens together with generic big data processing and analytics tools. These shall provide advanced real-time crime situation awareness at localised scales for citizens. The following sections of this paper provide the pertinent aspects of the new generation decision-support system design for monitoring crimes in cities. This is performed in a top down approach to provide the status of the current prototype system development.

2 Decision-Support System Portal (DSSP) Design

The DSSP is a web enabled information system environment which gives access to integrated data and information for the management of hazards and potential crimes incidents in cities. It allows users perform scenarios for the design of safer spaces. The DSSP is split into two main parts, the server side and the user or client side.

Fig. 1 illustrates a high level view of the overall system. The server-side is comprised of 3 distinct layers. The top layer concerns the various data sources. The middle layer is the analytics layer which incorporates local cache for data processing. Finally, the lower layer specialises in data visualisation. The client-side at the user end, is simple and relies on an HTML5 capable browser. The User Interface (UI) is platform agnostic, and user-friendly for tablets and mobile devices. In addition, a mobile application has been developed in order to facilitate end users to ingest data (reports, surveys) into the system.

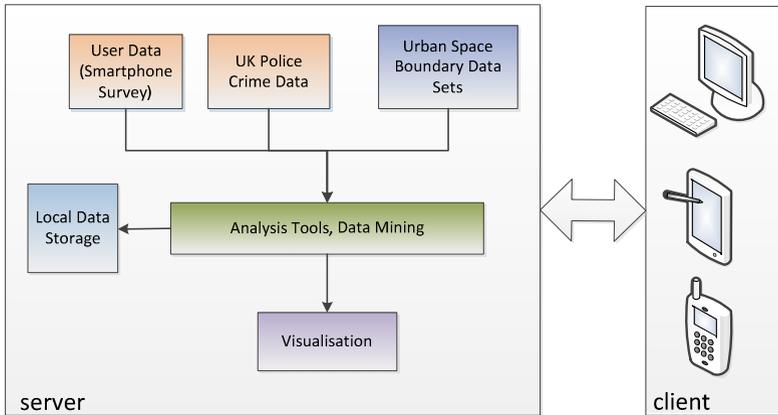


Fig. 1. System Top Level Design for the DSSP

2.1 Urban Space Boundary Sets

An important feature in DSSP is how the abstract idea of an urban space is defined. Having defined an urban space then data analysis on crime or on safety perception from other data resources can be easily applied. Urban spaces in DSSP are defined as boundaries in the form of geospatial data polygons. These polygons are used as our base boundary layer which allow us to filter and group the crime incidents and on top of which we do all the visualisations and analysis. Although a city can be defined in geographical terms by a single, but complex, polygon, smaller divisions of a city such as boroughs, districts, neighbourhoods, should also be defined in similar terms i.e. geospatial data polygons. These additional polygons will form the basis of the different zoom levels in our analysis tools. The transitions between different and with existing zoom levels should be consistent. For instance, polygons at the same zoom level should not overlap with neighbouring polygons. Similarly peripheral polygon boundaries should match higher zoom level polygon boundaries. For our analysis we examine 13 UK cities in which we define three distinct resolution levels. Data from the UK Census¹ and UK Ordnance Survey² was used to produce the required zoom level boundary sets. The different levels employed in the application are now defined.

- **Cities:** a boundary dataset containing polygons for each one of our cities. The census Districts boundary set is used for the council level.
- **Neighbourhoods:** This level offers a more in-depth view on each city's crime rates. The census Middle Layer Super Output Arias (MSOAs) boundary data set was used. MSOA polygons are population-wise similar (~7400), this provides a level of normalisation for our data analysis.

¹ <http://www.ons.gov.uk/ons/guide-method/census/census-2001/data-and-products/output-geography/output-areas/index.html>

² <http://www.ordnancesurvey.co.uk/oswebsite/products/boundary-line/index.html>

- **Postcodes:** The highest level of resolution. It allows a very detailed view of the crime rates in every part of the city.

Although boundary polygons for council and neighbourhood levels match, postcode boundary polygons do not match with either neighbourhood or council boundary polygons. This means that a postcode polygon can intersect two or more neighbourhood polygons. For this reason work was performed to make the different polygons consistent. The data sets are extracted from ESRI shapefiles and converted to use latitude and longitude coordinates within the WGS84 datum. The OS data for instance uses its own grid coordinate scheme as illustrated in Fig. 2a. Additionally, all necessary associations between neighbourhoods, councils and convex hulls were also created. An example of the city and neighbourhood boundaries is illustrated in Fig. 2b.

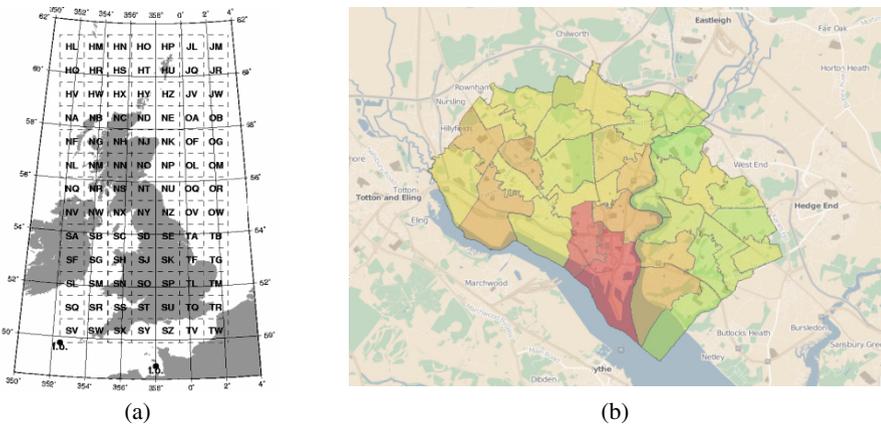


Fig. 2. (a) Ordinance Survey UK National Grid (b) The city and neighbourhood boundaries for Southampton

2.2 DSSP Data Sources

Having defined urban spaces in terms of boundary data with clear polygon sets, and zoom levels for all 13 UK cities, the next step is to aggregate data related to those cities. In addition as the data sources all have different coordinate systems we need to project them all onto the same system. We employed the WGS84 coordinate system for all our work.

Crime Data

The UK Police Crime database is based on data about reported crime incidents. In order to protect the privacy of individuals who are involved in the incidents, the police provide approximate and not completely accurate coordinates. For example crimes are not included unless they are from a street with 8 or more postal addresses. The raw data is publicly available via a police API which provides information on the type of crime, location (in Easting and Northing), and the date of the crime.

The DSSP system periodically checks for updates and downloads newly released crime data to keep its database up to date. UK crime data from January 2011 to April

2012 period were initially gathered and processed and cached locally. Gathered data is pre-processed to use a common grid reference system. Reported coordinates for each crime incident are provided in the form of Eastings and Northings 2-D Cartesian coordinates using the Transverse Mercator projection and the OSGB36 datum. The DSSP converts crime incidents coordinates into longitude and latitude within the WGS84 datum. Finally crime data are filtered and stored. We use a modified ray casting algorithm to decide if a crime lies within a specific polygon at a particular zoom level.

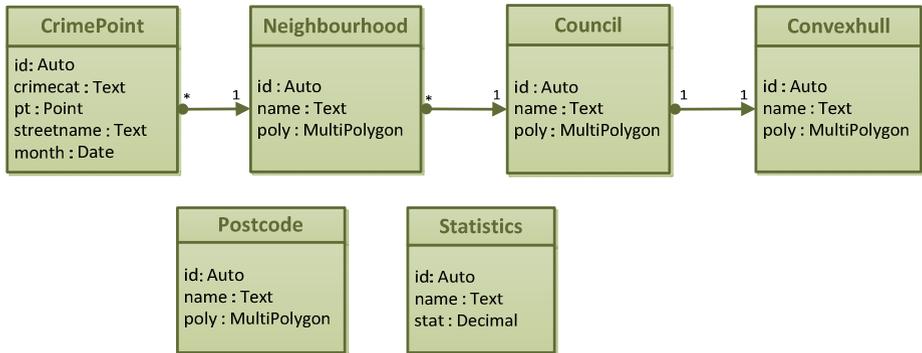


Fig. 3. Crime database schema

Fig. 3 shows the schema for the database. Notice that each crime is associated with exactly one neighbourhood, each neighbourhood is associated with exactly one council, and each council has exactly one convex hull. These associations allow us to group the data into areas, so that they can be easily retrieved later, by minimising the number of spatial queries. This is necessary as while our database provides fast spatial queries they are still slower than simple lookup queries. Postcodes are not associated with the other geographical entities, because their boundaries are not matching. Currently we only use the postcode information for visualisation so this is not a problem.

Participative Users Data

The system allows individual user to submit their own reports about their personal perception of safety in urban spaces. Perception of safety can have the form of a report or a survey that can describe an incident or opinion from the individual’s point of view. Individuals therefore can submit reports directly to the DSSP portal over a browser or via the portal API. The API is REST based and is fully self-describing and web browseable. This simplifies client side development. Submitted surveys and reports are stored in system user data database. A user data report includes name of report, description, submission date (including time), geospatial information, a user identifier (which is hashed to preserve privacy), and optional attachments such as images.

2.3 Analysis Tools and Data Mining

The DSSP provides all the necessary tools to analyse and harvest information from the different data sources. Processed data is stored in a local database. Analysis of data can take the form of normalisation, processing, and conversion between coordinate systems. Upon data ingest the data is filtered and aligned to the boundary polygons for use in creation of the crime data reports. For each crime type we also generate a heat-map. The data is interpolated to create a continuous function over the polygon. The heat-maps are generated at neighbourhood and postcode boundaries. Data mining takes place on user reported surveys to correlate with actual incidents from other data sources. This provides a measure of truthfulness to the user generated content. For example a police reported crime can be chosen, as the actual incident, and the data mining operations, e.g. correlation, can be applied on user reported data pool. In this way an incident can be enriched with user submitted set of data, and presented in the form of a timeline arrangement as shown in Fig. 4. Similarly the perception of safety for a particular location at a particular time can be detected, measured, and analysed.

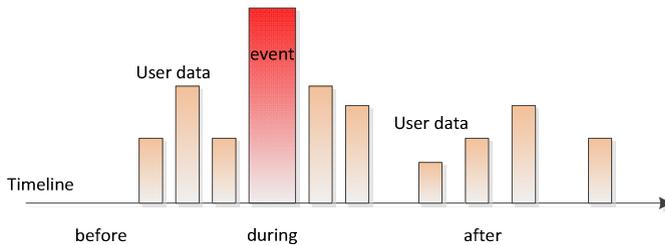


Fig. 4. Timeline of an incident

2.4 Visualisation

The DSSP provides a rich set of data visualisations including interactive maps with different levels of information for different resolution levels. Also, area statistics with pie and line charts and comparisons and correlations between two different areas of interest. Finally, it also provides heatmaps for each crime type and crime point projections. The DSSP mapping service is comprised of the following parts, the base map data, the javascript interaction library, and the map server. Since the Google Map API could not be used due to its licence constrains, the base map data is based on OpenStreetMap which is free and not bound by usage limits. For the Javascript interaction library we used Leaflet which provides all the necessary functionality. Additionally it is lightweight and designed to be mobile friendly. The map server is based on Tilestache server, a python-based tile server that can serve and cache map tiles based on geographic data. Tilestache takes requests in the form of tile URLs such as:

```
http://{address}/{layer name}/{zoom}/{column}/{row}.{extension}
```

This request describes the address of a map tile using the Google XYZ tile addressing format. The format defines that tiles are referenced by their zoom level and X and Y

values. It also takes a layer name as parameter. This allows many layers to be created each illustrating a different set of pertinent data. The DSSP currently has the base OpenStreetMap layer and 11 additional layers for different types of heat-maps. The charts are created with jqPlot which is a javascript based plotting library. It plots data transmitted from the server via JSON objects. The basis for the user interface is Twitter Bootstrap. This employs a 960 pixel grid approach which allows a large degree in versatility in layout and design. The consequence of this is the responsive design which allows the DSSP to support a number of different devices, both mobile and desktop, via a simple code base.

2.5 Client Applications

The DSSP offers its client functionality over a web interface, the UI is HTML5 based and therefore platform agnostic. Additionally we make our interface responsive (adapt to client device resolution). Special considerations were taken into account in order to provide a UI that is convenient for tablets and mobile devices too. As an example an interface based on mouse clicking items to bring up information is not directly useable on a touch based device. Smartphones with their unique characteristics such as wide spread adoption, permanently carried, always on, interactivity; provide a powerful complementary and disruptive mass media. DSSP considers and targets smartphones as the major source of the User Data, has developed a customised mobile application to submit reports and surveys. DInR is a simple mobile app, which is under development at the moment, in order to facilitate individuals submitting reports into the DSSP directly using their mobile phones. Currently DInR is running on Android 4.1 phones or tablets and provides most of required report fields. DInR can successfully create and submit reports to the current version of DSSP system. DInR can provide automatically 3/5 data fields in the user report, consequently smartphone users can submit to DSSP user data with a minimal effort.

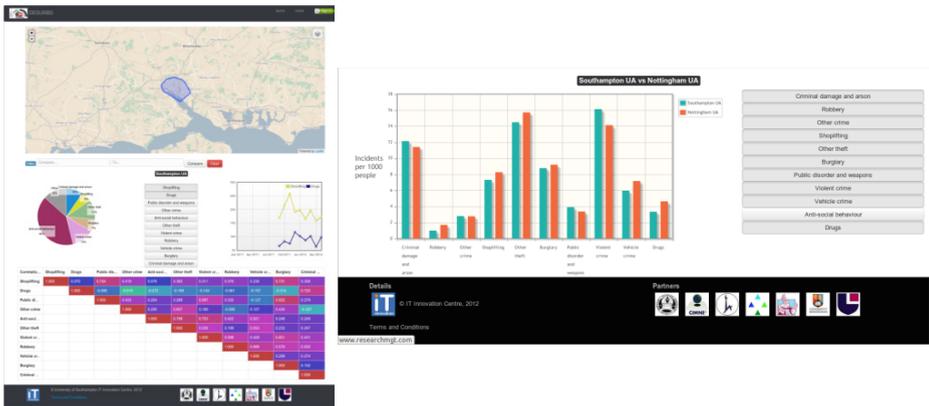
3 DSSP Prototype

3.1 Functionality

The current implementation of the system prototype runs on two distinct machines. The first machine hosts the main application while the second on the Tileserver. This separation of machine allows for independent optimisation and deployment of the two main components. The main application runs on a machine with an 2.83 GHz CPU and 4GB of RAM. The Tileserver has an 2.83 GHz CPU, 4GB of RAM, and 2 TB of fast storage. Currently a demonstration version of the system application is hosted externally³. The application allows the user to respond to a number of questions of the raw crime data. It can also perform them at a variety of zoom levels. These levels include Cities, Neighbourhoods, and Postal Codes which are in accord with the standard Ordinance Survey (OS) classifications in the UK. An example of the browse mode is shown in Figure5(a). The top part of the figure is a GIS view that shows a zoomed version of the region of interest (South of England) with a blue polygon hig-

³ <http://desurbs.it-innovation.soton.ac.uk/police>

highlighting the city under investigation. The lower part of the figure shows a series of graphs concerning crime data analytics. The pie chart illustrates the relative proportion of the specific crime types. The line chart shows the historical rates of crime types, where the specific crimes illustrated in the graph are selectable. Further, a colour coded matrix shows the cross-correlations of the different crime types in the City of interest. This allows the user to cluster crimes of various types in the investigation. As illustrated in Figure 5(b) “Drugs” related crimes statistically correlate well with “Criminal Damage and Arson” based crimes in the same City. This feature in the DSSP enables crime knowledge experts to demonstrate crime status in specific areas of a City with greater emphasis to a collective audience of decision-makers in urban design of safer spaces.



(a)

(b)

Fig. 5. (a) Illustration of the application showing crime data for a specific city (b) Comparison of two cities

The comparison between various cities crime trends can also be processed by the DSSP as illustrated in **Chyba! Nenalezen zdroj odkazů.b**. The bar chart shows the number of incidents of a specifically selected crime type in Cities. The specific crime types are also user selected. Furthermore, the crime data is normalised in accord with the respective population densities in each city. With this feature, user queries about the relative numbers of crimes in different urban regions of a given country can be investigated. In Fig. 6 a Heat-Map is visualised and aligned to the OS administrative boundaries. In addition to the above, the locations of all specific crime types in a City can be added as new layers above the Heat-Map for further analysis and comparative studies. These are visualised with representative points that can be removed by the user if needed. The visualisation of the crimes provides the respective crime-specific locations in a city of interest together with the intuitive discovery of urban areas that are more exposed to specific crimes than others.



Fig. 6. Heatmap based on the crime data with specific crime points overlaid

3.2 Optimisation

The DSSP was built to provide both functionality and performance on multiple platforms. Many optimisations carried out in order to improve the system performance and responsiveness where the goal was to present similar user experiences even with low powered mobile platforms. The general strategy to meet this goal is to perform most of the processing work on the server and send to the client only the minimal amount of data required. The layers for the boundary datasets at different resolutions are chosen carefully to match their borders. The vertices of the polygons are simplified wherever possible to increase performance at different zoom factors. The UK crime data is filtered, trimmed, and stored locally for the regions of interest only. In order to enforce consistency between the boundary datasets data normalisation at different resolutions was performed. The use of a unified coordinate scheme also makes visualisation and analysis simpler. Geospatial features of PostgreSQL are also not used unless there is a processing benefit from it. The mapping server (Tilestache) is lightweight and scalable and has the benefit of caching data. In order to reduce network traffic lightweight javascript libraries are used. Prior computation and caching takes place on the DSSP server extensively. Specifically, pre-calculation is used to compute spatial features. For example the Southampton convex hull was reduced from 2834 vertices to 53 points. When the visualisation is computationally expensive the imagery is cached on the tileserver. Statistical analysis of the data is also performed offline and displayed via a single lookup in a database. Finally, the clustering of the crime points is also performed offline at different zoom levels.

4 Conclusions and Future Work

An early DSSP prototype which specialises in the provision of integrated data and information for designing safer spaces in urban agglomerations has been described in this paper. The new system is enabled under the Future Internet context of 21st century smart cities; where citizens participate in providing collective observations on potential hazards, incidents and their perceptions of safety around their local areas. The supporting hardware infrastructure of the DSSP consists of a number of special purpose machines with specific roles. These include the front-end application server and the tile server for data and information mapping. A set of optimisations have been performed in order to make the system work efficiently, particularly on low-powered mobile clients. The DSSP is currently innovative for potentially collecting citizen's

perception of safety in local urban zones and contrasting it with the existing UK Police crime data. Nevertheless, the system prototype has still a “work in progress” status considering the number of visualisation features and functionalities which shall be added to it prior to the end of summer 2013. For example, the current tile-server which serves identical sized tiles for different devices will be optimised and improved to perform more efficiently across all type of platforms. Additional enhancements shall be made to the mobile interface too. Specifically, the device dimensions will be used to inform the layout of the user interface. The mobile survey tool will also be expanded with more generic features in order to apply to wider participatory observations under the smart cities context. This is pertinent to making the system useful for a variety of smart cities oriented applications.

References

1. Filipponi, L., Vitaletti, A., Landi, G., Memeo, V., Laura, G., Pucci, P.: Smart City: An Event Driven Architecture for Monitoring Public Spaces with Heterogeneous Sensors. In: Fourth International Conference on Sensor Technologies and Applications (SENSORCOMM), pp. 281–286 (2010)
2. Gibson, D.V., Kozmetsky, G., Smilor, R.W.: The Technopolis Phenomenon: Smart Cities, Fast Systems, Global Networks. Rowman & Littlefield, New York (1992)
3. Lane, N.D., Eisenman, S.B., Musolesi, M., Miluzzo, E., Campbell, A.T.: Urban Sensing Systems: Opportunistic or Participatory? In: Proceedings of the 9th Workshop on Mobile Computing Systems and Applications, pp. 11–16 (2008)
4. Nam, T., Pardo, T.A.: Smart City as Urban Innovation: Focusing on Management, Policy, and Context. In: Proceedings of the 5th International Conference on Theory and Practice of Electronic Governance, pp. 185–194 (2011)
5. Sabeur, Z.: Future Internet Smart Cities User Behaviour Driven Innovation Platforms. Presented at the 9th Edition of the Future Internet Assembly, Aalborg, Denmark (2012)
6. Schaffers, H., Komninos, N., Pallot, M., Trousse, B.: Smart Cities and the Future Internet: Towards Cooperation Frameworks for Open Innovation. In: The Future Internet, pp. 431–446 (2011)
7. Walravens, N.: The City as a Platform A Case-based Exploration of Mobile Service Platform Types in the Context of the City. In: International Conference on Intelligence in Next Generation Networks, pp. 283–288 (2011)