University of Crete
Computer Science Department

AN INTEGRATED PLATFORM FOR LOCATION–AWARE INFORMATION SYSTEMS:

SPATIAL CONTENT ADMINISTRATOR

MASTER’S THESIS

by
Nikolaos Kazepis

Heraklion, October 2008
To my father Ioannis, my sister Francesca

And the memory of my beloved mother Despoina
AN INTEGRATED PLATFORM FOR LOCATION – AWARE INFORMATION SYSTEMS:
SPATIAL CONTENT ADMINISTRATOR

By
NIKOLAOS KAZEPIS

A thesis submitted in partial fulfillment of the requirements for the degree of
Master of Science

Author:
Nikolaos Kazepis, Department of Computer Science

Supervisor:
Anthony Savidis, Associate Professor

Member:
Constantine Stephanidis, Professor

Member:
Antonios Argyros, Associate Professor

Chairman:
Panos Trahanias, Professor
Chairman of the Graduate Studies Committee

Heraklion, October 2008
University of Crete
Computer Science Department

AN INTEGRATED PLATFORM FOR
LOCATION – AWARE INFORMATION SYSTEMS:
SPATIAL CONTENT ADMINISTRATOR

MASTER’S THESIS

Nikolaos Kazepis

Abstract

Mobile location-aware information systems are capable to deliver position-dependent information over a portable device for users that are primarily on the move. Their support requires a combination of core or User Interface subsystems, ranging from authoring, database, middleware, monitoring, visualization, network and navigation.

In this Thesis we report the design and implementation of the Spatial content administrator facilitating the authoring of information content for exhibits visually, over area maps in a direct manipulation manner. Through the administrator, physical exhibits are associated to information content defined as information points, or information areas enclosing multiple information points. In summary, the following features are supported:

- Interactive definition of navigation scenarios
- Interactive addition, removal or displacement of information points and areas
- Detailed form-based content editing for all information categories
- Spatial supervision of information entry status over area maps

We proposed a paradigm shift from content-driven authoring to context-driven direct content manipulation as more appropriate for the administration of location-aware information systems. Some of the additional features included are: mobile database publishing, remote content administration and multimedia data editing and playback (video and audio).
Πανεπιστήμιο Κρήτης
Τμήμα Επιστημών Υπολογιστών

ΕΝΙΑΙΑ ΠΛΑΤFORMΑ ΓΙΑ
ΠΛΗΡΟΦΟΡΙΑΚΑ ΣΥΣΤΗΜΑΤΑ ΜΕ ΕΠΙΓΝΩΣΗ ΘΕΣΗΣ:
ΔΙΑΧΕΙΡΙΣΤΗΣ ΧΩΡΙΚΟΥ ΠΕΡΙΕΧΟΜΕΝΟΥ

ΜΕΤΑΠΤΥΧΙΑΚΗ ΕΡΓΑΣΙΑ

Νικόλαος Καζέπης

Περίληψη

Τα φορητά πληροφοριακά συστήματα με επίγνωση θέσης μπορούν να προσφέρουν πληροφορίες που εξαρτώνται από τη θέση, σε χρήστες που βρίσκονται κυρίως εν κινήσει, χρησιμοποιώντας φορητές συσκευές. Η υποστήριξη τους απαιτεί ένα συνδυασμό από βασικά υποσυστήματα ή διεπαφές χρήστη, που περιλαμβάνουν υποσυστήματα δικτύου, βάσεων δεδομένων, συγγραφής περιεχομένου, παρακολούθησης συστήματος, πλοήγησης, οπτικής αναπαράστασης δεδομένων και διαμεσολαβητικά υποσυστήματα.

Στην παρούσα εργασία περιγράφουμε το σχεδιασμό και την υλοποίηση του Διαχειριστή χωρικού περιεχομένου, ο οποίος διευκολύνει τη συγγραφή περιεχομένου για εκθέματα ενώ παρέχει απευθείας χειρισμό, αναπαραστώντας οπτικά τα δεδομένα πάνω σε χάρτες. Η διαδικασία της διαχείρισης αντιστοιχίζει εκθέματα με πληροφορίες περιεχομένου που ορίζονται είτε σαν σημεία ενδιαφέροντος, είτε σαν περιοχές ενδιαφέροντος οι οποίες περικλείουν πολλαπλά σημεία ενδιαφέροντος. Περιληπτικά υποστηρίζονται τα παρακάτω χαρακτηριστικά:

- Διαδραστικός καθορισμός σεναρίων πλοήγησης
- Διαδραστική προσοχή, διαγραφή και μετακίνηση περιοχών και σημείων ενδιαφέροντος
- Αναλυτική επεξεργασία περιεχομένου, για κάθε κατηγορία, βασισμένη σε φόρμες
- Χωρική επίβλεψη καταχωρήσεων περιεχομένου με χρήση χαρτών
Προτείνουμε τη μετάβαση από τη συγγραφή βάσει περιεχομένου στην απευθείας διαχείριση του περιεχομένου βάσει τοποθεσίας, ως καταλληλότερο πλαίσιο για τη διαχείριση πληροφοριακών συστημάτων με επίγνωση θέσης. Μερικά από τα πρόσθετα χαρακτηριστικά συμπεριλαμβάνουν: τη δημοσίευση της βάσης δεδομένων ως φορητό στιγμιότυπο στις φορητές συσκευές, την απομακρυσμένη διαχείριση περιεχομένου και την επεξεργασία και αναπαραγωγή πολυμεσικού περιεχομένου (βίντεο και ήχος).
Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επόπτη της μεταπτυχιακής μου εργασίας Αντώνιο Σαββίδη για την συνεχή καθοδήγηση και υποστήριξη του τα τελευταία τρία χρόνια στο πλαίσιο της συνεργασίας μας στο Εργαστήριο Επικοινωνίας Ανθρώπου-Μηχανής, του Ινστιτούτου Πληροφορικής του Ιδρύματος Τεχνολογίας και Έρευνας και στο πλαίσιο της εκπόνησης της μεταπτυχιακής μου εργασίας.

Επίσης, θα ήθελα να ευχαριστήσω το Δημήτρη Γραμμένο, τη Μαργαρίτα Αντώνα, την Ήλια Αδάμη και τους υπόλοιπους συνεργάτες μου στο Εργαστήριο Επικοινωνίας Ανθρώπου-Μηχανής, του Ινστιτούτου Πληροφορικής του Ιδρύματος Τεχνολογίας και Έρευνας για την σημαντική συμβολή τους στη διαμόρφωση του παρόντος κειμένου.

Ευχαριστώ, τέλος, την οικογένειά μου, τους φίλους μου και την Άννα που με υπέφεραν, με στήριξαν και με βοήθησαν όλα αυτά τα χρόνια.
# Table of Contents

Abstract ................................................................................................................................. v

Περίληψη ................................................................................................................................ vii

Ευχαριστίες .......................................................................................................................... x

Table of Contents .............................................................................................................. xi

List of Figures ....................................................................................................................... xiii

List of Tables ....................................................................................................................... xiv

1. Introduction .................................................................................................................... 15
   1.1. Contributions ........................................................................................................... 18
   1.2. Platform Architecture ............................................................................................ 19
   1.3. Thesis Outline ......................................................................................................... 21

2. Related Work ................................................................................................................ 23
   2.1. Hard-encoded and form-based input ...................................................................... 23
   2.2. Spatial Graffiti ...................................................................................................... 24
   2.3. Discussion ............................................................................................................... 25

3. System Overview ......................................................................................................... 27
   3.1. Spatial Content Administrator ............................................................................. 28
   3.2. Spatial Content Administrator Architecture ....................................................... 35
   3.3. Multimedia Database Schema ............................................................................. 36

4. On Map Direct Content Manipulation ...................................................................... 39
   4.1. Manipulating Information Areas ........................................................................... 39
       4.1.1. Creating new Information Areas ................................................................. 41
       4.1.2. Moving Information Areas ......................................................................... 41
       4.1.3. Deleting Information Areas ....................................................................... 42
   4.2. Manipulating Information Items ........................................................................... 43
       4.2.1. Creating Information Items .......................................................................... 44
       4.2.2. Moving Information Items .......................................................................... 45
       4.2.3. Deleting Information Items .......................................................................... 46
   4.3. Assigning Location Tags to Information Areas ..................................................... 46
   4.4. Defining Physical Map Boundaries ...................................................................... 48
   4.5. Producing the Mobile Database ........................................................................... 50
       4.5.1. Remote Synchronization .............................................................................. 50
       4.5.2. Metadata Generation .................................................................................... 51
4.5.3. Uploading to the Synchronization Server .............................................. 53
4.6. Requesting Mobile Devices to Synchronize ............................................. 54
5. Data Editing and Administration .................................................................. 56
  5.1. Data Language Management ................................................................... 56
    5.1.1. Adding new language entries ......................................................... 57
    5.1.2. Editing language entries ............................................................... 57
    5.1.3. Removing language entries ......................................................... 58
    5.1.4. Taking an existing language online / offline ............................... 59
  5.2. Maps ........................................................................................................ 60
    5.2.1. Adding new maps ....................................................................... 61
    5.2.2. Editing map details ................................................................... 62
    5.2.3. Deleting maps ........................................................................... 63
    5.2.4. Managing map geographical information .................................. 64
    5.2.5. Moving and scaling while preserving ratio ............................... 65
    5.2.6. Skewing information areas ......................................................... 66
  5.3. Information Content ................................................................................ 67
    5.3.1. Home Page ................................................................................ 68
    5.3.2. Description and Multimedia ....................................................... 69
    5.3.3. Adding auditory content ............................................................ 76
  5.4. Authoring of Navigation Scenarios .......................................................... 77
    5.4.1. Over viewing ............................................................................. 78
    5.4.2. Creating new navigation scenarios ............................................ 79
    5.4.3. Editing navigation scenarios ....................................................... 79
    5.4.4. Clearing navigation scenario items ............................................ 81
    5.4.5. Deleting navigation scenarios ..................................................... 81
  6. Application in a Museum ............................................................................ 82
    6.1. Central Management ......................................................................... 82
    6.2. Connecting to the Museum’s Database .......................................... 83
    6.3. Uploading changed multimedia files to the remote FTP content server........ 85
  7. Summary, Conclusions and Future Work ..................................................... 86
  8. Acknowledgements ................................................................................... 89
  9. References ................................................................................................ 90
## List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fig. 1</td>
<td>Spatial Content Administrator application</td>
<td>16</td>
</tr>
<tr>
<td>Fig. 2</td>
<td>Overall platform architecture</td>
<td>19</td>
</tr>
<tr>
<td>Fig. 3</td>
<td>Split of the information plane to maps, areas and information points</td>
<td>20</td>
</tr>
<tr>
<td>Fig. 4</td>
<td>A generic mobile location-aware information system</td>
<td>27</td>
</tr>
<tr>
<td>Fig. 5</td>
<td>Map representation in the Spatial Content Administrator</td>
<td>29</td>
</tr>
<tr>
<td>Fig. 6</td>
<td>Spatial Content Administrator: Menu Functions</td>
<td>30</td>
</tr>
<tr>
<td>Fig. 7</td>
<td>The map, the home page of an item and the</td>
<td>32</td>
</tr>
<tr>
<td>Fig. 8</td>
<td>Micro architecture of the Spatial Content Administrator</td>
<td>36</td>
</tr>
<tr>
<td>Fig. 9</td>
<td>The database schema.</td>
<td>38</td>
</tr>
<tr>
<td>Fig. 10</td>
<td>Information areas created clockwise (left) and counterclockwise (right)</td>
<td>39</td>
</tr>
<tr>
<td>Fig. 11</td>
<td>Information Areas Sidepanel</td>
<td>40</td>
</tr>
<tr>
<td>Fig. 12</td>
<td>Creating new Information Areas</td>
<td>41</td>
</tr>
<tr>
<td>Fig. 13</td>
<td>Direct area manipulation with drag &amp; drop facilities</td>
<td>42</td>
</tr>
<tr>
<td>Fig. 14</td>
<td>Legend for item colors as displayed on the map</td>
<td>43</td>
</tr>
<tr>
<td>Fig. 15</td>
<td>Information Items Sidepanel</td>
<td>44</td>
</tr>
<tr>
<td>Fig. 16</td>
<td>Set area tag id button</td>
<td>47</td>
</tr>
<tr>
<td>Fig. 17</td>
<td>Tag id button acting as a label</td>
<td>47</td>
</tr>
<tr>
<td>Fig. 18</td>
<td>Setting the area tag id</td>
<td>48</td>
</tr>
<tr>
<td>Fig. 19</td>
<td>Setting map boundaries</td>
<td>49</td>
</tr>
<tr>
<td>Fig. 20</td>
<td>The XML schema used for data synchronization</td>
<td>52</td>
</tr>
<tr>
<td>Fig. 21</td>
<td>The Content Synchronization process</td>
<td>55</td>
</tr>
<tr>
<td>Fig. 22</td>
<td>Language Manager</td>
<td>57</td>
</tr>
<tr>
<td>Fig. 23</td>
<td>Language Editor: Editing the language information</td>
<td>58</td>
</tr>
<tr>
<td>Fig. 24</td>
<td>Map Manager</td>
<td>61</td>
</tr>
<tr>
<td>Fig. 25</td>
<td>Map Manager: Editing map details</td>
<td>63</td>
</tr>
<tr>
<td>Fig. 26</td>
<td>Editing map geographical information</td>
<td>65</td>
</tr>
<tr>
<td>Fig. 27</td>
<td>Moving and scaling information areas (including contained information items)</td>
<td>66</td>
</tr>
<tr>
<td>Fig. 28</td>
<td>Information Item’s home page</td>
<td>68</td>
</tr>
<tr>
<td>Fig. 29</td>
<td>Text description page (information item’s short description)</td>
<td>71</td>
</tr>
<tr>
<td>Fig. 30</td>
<td>Photograph description page (information item’s short description)</td>
<td>72</td>
</tr>
<tr>
<td>Fig. 31</td>
<td>Video description page (information item’s detailed description)</td>
<td>73</td>
</tr>
<tr>
<td>Fig. 32</td>
<td>Multi paged description text (information item’s multimedia content)</td>
<td>75</td>
</tr>
<tr>
<td>Fig. 33</td>
<td>Auditory content tab in the information item content editor</td>
<td>77</td>
</tr>
<tr>
<td>Fig. 34</td>
<td>Overview of the selected navigation scenario</td>
<td>79</td>
</tr>
<tr>
<td>Fig. 35</td>
<td>Editing the information items that belong to the selected navigation scenario</td>
<td>80</td>
</tr>
<tr>
<td>Fig. 36</td>
<td>Connecting to the remote FTP server to fetch needed content files</td>
<td>83</td>
</tr>
<tr>
<td>Fig. 37</td>
<td>Connecting to the museum’s database which is running on a remote server</td>
<td>84</td>
</tr>
<tr>
<td>Fig. 38</td>
<td>Connection progress while trying to connect to the remote database</td>
<td>85</td>
</tr>
</tbody>
</table>
List of Tables

Table 1: Information Item Content snippet for a museum ........................................ 67
1. Introduction

Managing the contents of spatial databases has always been a burden for the administrators of such systems. Despite the considerable value of currently implemented location-aware information systems, such as the ones mentioned in Abowd et al. 1997, Burrell et al. 2002, Bickmore et al. 1997, Cheverst et al. 2000, Espinoza et al. 2001, and Griswold et al. 2003, one aspect in which they can be improved is the provision of more effective methods for adding and managing context specific content.

Current location-aware information systems lack an effective method of updating and maintaining dynamic, location-specific content. Very few tools and applications are available to facilitate the data management and maintenance process. System administrators often have to memorize the entire layout of the particular implementation and then manually fill in the databases with the appropriate data either by directly editing the database tables or through graphical form-based interfaces. This process has been proven to be inefficient and sometimes the steps of the task are very difficult to remember. Furthermore, the administration and maintenance of the data and multimedia information which has been stored in the system is a very complex process. Without an effective and simple way for authors to provide new content and update the currently existing one, information stored in such systems can soon become outdated and useless. Without well-structured, up-to-date, comprehensive content, location-aware information systems can very quickly become outdated and eventually obsolete.

As mentioned in Griswold et al. 2003, when it comes to large systems and spatial location – specific context, maintenance and update can be very expensive and time consuming. Without the right tools and applications, maintenance can even be impossible because any existing unstructured data can’t be updated and hence soon become “stale”.

Towards providing a solution to this problem, this thesis proposes the utilization of spatially-aware content management editing administrative systems to
allow end-users with very little or no technical skills to easily add, modify, structure, update and delete any type of digital multimedia information that relates to real-world places (e.g., related to an exhibit of a museum or a particular store of a shopping center), in a short period of time. This can be easily done through the implemented Spatial Content Administrator application, assisting the authors in every step of the process. The supported functions are very straightforward to use, easy to master, remember and follow, while the content author is assisted throughout the process.

![Spatial Content Administrator application](image)

**Fig. 1: Spatial Content Administrator application**

This Thesis presents the key components of an integrated platform for the authoring and uptake of location-aware mobile information systems, putting
emphasis on specific novel features reflecting requirements that emerged in the course of large-scale real-life deployment. The development of the overall platform required three years (2004-2007). Within the integrated platform data are offered in multiple representations as different types of multimedia files in order to meet the individual needs of different venues.

The developed content management tool, named Spatial Data Administrator, was implemented using Microsoft’s Direct3D technology (see Fig. 1). The main underlying objective was to balance flexibility and ease of use for representing location-specific content. In this context, this Thesis provides a description of how the proposed design allows end-users to easily add, edit and structure location-related data. The Spatial Data Administrator application has been designed and implemented to facilitate the data administration process when it comes to spatial databases in large scale information systems. The system provides an easy-to-use interface to bind any type of multimedia content – such as text, pictures, audio, or video – to a specific location.

In summary, the primary objectives of this Thesis are:

- To facilitate the spatial data content administration of large-scale spatial location-aware information systems.
- To provide a straightforward and easy-to-use data manipulation tool that enables content administrators to add, edit, update, delete and bind multimedia information (e.g., related to an exhibit of a museum) to a related location on a map.
- To enable unskilled users to use this administration tool without the need to remember complicated processes.
- To give system administrators a visual overview of multimedia content that is associated with an actual geographic location.
- To associate complex tasks and processes with single button presses in order to simplify the maintenance process of large-scale information systems.
As denoted by the title of the current Thesis, the Spatial Content Administrator is part of a bigger project targeted to the development of an integrated platform for location-aware mobile information systems, integrating several additional interoperating modules:

- Spatial location-oriented data editing with a direct-manipulation editor, offering mixed-mode administration: (a) mobile, on-site, mainly to administer location data; and (b) non-mobile, off-site, mainly to administer typical semantic content.
- System-initiated location-triggered information delivery combined with user-initiated on-demand content exploration.
- Unified infrastructure applicable both to indoor and outdoor setups.
- Very efficient and intuitive device renting facilities through barcode readers.
- Multi-channel statically-prioritized location sensing, currently implemented to deploy WLAN positioning, GPS, and infrared beacons.

More details about the integrated project can be found in Savidis et al. 2008, as well as in Vandikas 2007, Zidianakis 2008 and Dubulakis 2008.

1.1. Contributions

The key technical contributions of the reported work, concerning the support for content administration, user navigation, and runtime management of user sessions, are:

- Spatial location-oriented data editing with a direct-manipulation editor, offering mixed-mode administration: (a) mobile, on-site, mainly to administer location data (mobile device application); and (b) non-mobile, off-site, mainly to administer typical semantic content.
- Unified infrastructure applicable both to indoor and outdoor setups.
1.2. Platform Architecture

The overall system architecture is depicted in Fig. 2 at two levels of decomposition detail: (i) primary applications and tools (shaded rectangles); and (ii) their respective key constituent components (white rectangles and buckets). Solid arrows indicate intra-process method invocations (e.g. deployment of the Map Viewer component), while dashed arrows denote inter-process Remote Procedure Calls (e.g. DB Access Layer). Underlined labels mark different versions of a component, such as: mobile / server DB Access Layer, mobile / server Map Viewer, client / server API categories, and mobile / master DBs. This thesis discusses the Central Content Management System which comprises the upper right part of the image. The micro-architecture of the Spatial Content Administrator will be discussed in detail in Chapter 3.

Some important sub-systems, not to be discussed in this thesis, are also incorporated in the architectural diagram of Fig. 2:
- On-demand synchronization of the master DB to the mobile DB on the PDAs (i.e. DB Sync Client / Server). While one would expect this facility to be offered by the DBMS, in the conducted work it had to be implemented from scratch, as the mobile edition of the MS SQL Server restricts the size of the mobile DB to at most 100 MB.

- Recording data for navigation sessions (e.g. time spent on an information item, information items reviewed, elapsed time of use, etc.) supporting various queries (e.g. most popular information item in a selected period, total time of use for all visitors, average exploration time for visitors, information items with simultaneous visitors up to a threshold, etc). The latter concern the Navigation Statistics UI.

- Security policy relying on the definition (during content administration) of the legal navigation boundaries over area maps, out of which specific alert procedures are triggered (e.g. PDA locking with message to the user, explicit alert pop-ups on the monitoring console, visitor identification from the PDA and notification from the museum audio system, etc.). The latter concerns the Visitor Monitoring System.

**Fig. 3:** Split of the information plane to maps, areas and information points

The proposed platform relies on a generic location-sensing interface enabling sensing APIs to be loaded dynamically (as Dynamic Link Libraries). The analyzed method splits the global information plane into independent maps, where maps
encompass polygonal information areas, which en-close the actual information points, i.e. the real items of interest such as exhibits (see Fig. 3). In this context, the location sensing interface allows distinct technologies to return either a point (higher precision) within the currently active map plane, or alternatively the identifier of a polygonal area (lower precision) that is associated to a particular physical area (e.g. room, hall, corridor, corner, stares, etc.). Following the proposed approach, the adopted technologies need not be merely point-based, but alternative techniques like infrared beacons or radio tags can be deployed as carriers of area identification information (e.g. an infrared beacon can be programmed to simply emit the logical identifier of an area denoting a specific room).

Finally it is very important to mention that the system described is part of an integrated platform that can be used wherever mobile location-aware information systems may apply. For example it could be used in museums as an electronic guide for the different types of exhibits, in multi-storey shopping centers as a means for shoppers to find their way around the place, as an electronic city guide (see the GUIDE project, Cheverst et al. 2000) etc. All the examples used in the current Thesis are based on the deployment of the overall integrated platform in a museum and namely the Natural History Museum of Heraklion, Crete, so that the reader can be given a complete picture of the design, the implementation and the deployment of the, as well as of its use in real settings. All the examples used are drawn from the experiences and use-case scenarios that arose from this deployment but do not imply that the overall system is suitable only for such kind of deployment.

1.3. Thesis Outline

This Thesis is organized as follows. Chapter 2 discusses related work on this subject. Chapter 3 describes the Spatial Content Administrator in general. Chapter 4 discusses direct on-map content manipulation. Chapter 5 refers to graphical form-based content administration including Maps and their role as a cornerstone to the whole project. Chapter 6 analyzes the navigation scenarios concept and their maintenance through the Spatial Content Editor. Finally, Chapter 7 addresses issues
regarding the deployment of the application in a museum and how they were solved, while Chapter 8 discusses the conclusions and suggests future work.
2. Related Work

Much research work has been conducted on spatially-aware content management systems in the wider area of location-aware information systems. Several implemented systems aiming at updating existing database data and global content management were studied. The design of the proposed system was based on the outcomes of this study.

2.1. Hard-encoded and form-based input

The first project that was studied was the “GUIDE” project (Cheverst et al. 2000). It was developed by the DMR group which is the distributed multimedia research group of the University of Lancaster. This project was about a location-aware tour guide of the homonym city. It could deliver useful information to the user about nearby points of interest such as restaurants, museums, movies, plays etc. The project because of its nature required to be associated with dynamic information which was updated on a very frequent basis to provide up to date information. This dynamic content was added and maintained by a full-time employee who worked on that contract. The content that needed to be updated contained different kinds of information such as the menus of associated restaurants and coffee houses, and thus the maintenance task was quite hard to accomplish. Moreover, many organizations might be uninterested of taking part in that kind of projects because of financial or practical reasons to hire an employee only for that job. Furthermore, when we think of very dynamic environments such as multi-storey shopping centers or universities the cost of maintenance would be forbidding. Even worse, the service providers may find it extremely inconvenient to have to update the whole content through the administrator tool every time they wanted to make a minor change to their content. These barriers could be avoided by designing fill-in forms that simplify content management for a large number of typical end-users.

Cyberguide (Abowd et al. 1997) was developed by the FCE group (Future Computing Environments) at the Georgia Institute of Technology. It was a mobile, context-aware tour guide that would be used as a dynamic mobile e-book.
Cyberguide’s information content was initially hard-coded, and then stored on mobile handhelds in special form such as local HTML pages. However, the FCE Group concluded that, in a dynamic environment, a “local and static database is only slightly more useful than a book.” As part of its future work, the FCE Group cited the need to develop a modifiable information base, in which tourists could add their own content making it something like a dynamic e-book with reviews from people who would have “been there”.

Google Local (Google’s Local Business search on Google Maps electronically available at: http://www.google.com/local) is a useful location-aware information system that allows mobile users to find nearby businesses and services. However, Google Local does have some limitations, such as requiring location-specific information to be in the form of a website, with an associated postal address. There may also be a delay of one or more days upon submission of the URL to Google. In addition, Google Local has a commercial orientation that may provide irrelevant or undesirable results for universities or other such organizations. These organizations may wish to have a structured, contained information system that they are able to personally and immediately manage.

2.2. Spatial Graffiti

The UC San Diego’s ActiveCampus Project (Griswold et al. 2003) implemented an array of context-aware application services in a university setting. Mobile users in this system were given features such as knowledge of their own location and the location of other mobile users, messaging among mobile users, and the capability of generating spatial graffiti. Spatial graffiti is based on the idea of allowing users to convey their thoughts in the form of electronic notes that are associated with particular physical locations (Burrell et al. 2002). Other mobile users can then view these notes when they are at these locations. The ActiveCampus Project utilized spatial graffiti as a principal method for users to generate new content. Over a thirteen-month period, there was one month where over fifty people posted graffiti; however, there were fewer than ten people (and in several months, only one or two
people) posting graffiti in most months. Considering the fact that over 500 users were involved in using the system, the number of spatial graffiti posters is relatively low. One reason for this could be the inconvenience of data entry on mobile handhelds. Additionally, as discovered by the E-graffiti project (Burrell et al. 2002), because spatial graffiti is “too openended” and “provides too many options,” users may not know what to write because they don’t “have the time or energy to think extensively about location and the information that might be relevant to their current context”.

The GeoNotes project (Espinoza et al. 2001) also points out that the unstructured nature of spatial graffiti can quickly create a cluttered information space in which users are unable to quickly find relevant information. For these reasons, in the system described in this thesis, location-specific content is organized through a cognitive structure that reduces end-users’ thinking time for the most common content additions and information requests.

2.3. Discussion

Although all the above studied projects made a notable attempt to design, implement and deploy location-aware platforms supporting large scale information systems, none of them took the final leap that was needed in order to catch users’ attention, make them get interested in using the system and making the actual use of the system easy enough for its users. None of the above projects focuses on the management of the content of the system which became the inspiration for the design and implementation of the Spatial Content Administrator authoring tool.

The Spatial Content Administrator tool was designed and implemented to facilitate the data maintenance process of large-scale information systems. The main objective is to help the users input, edit, structure or delete quickly and easily any piece of information stored. All administrative tasks are easily carried out through the simple yet very powerful user interface which helps the content authors through
the process. Users with little or no administrative skills can now easily maintain all
the information stored to such spatial information systems.
3. System Overview

Mobile location-aware information systems are capable to deliver position-dependent information over a portable device for users that are primarily on the move. Their support requires a combination of core or User Interface subsystems, ranging from authoring, database, middleware, monitoring, visualization, network and navigation (see Fig. 4)

Content management systems consist of two components, the content management application component and the content delivery application component (Universal Glossary of Web Publishing Terms.). The content management application component allows for the creation, modification, and removal of material in the system. The content delivery application component compiles and presents the information that the user sees on the screen. Such systems are commonly used to manage all but the simplest of web sites with dynamic content.
The current integrated platform for mobile location-aware information systems consists of three main subsystems. The Spatial Content Administrator application component as the content management application component, the Mobile Multimedia Navigator application component (Zidianakis, 2008) as the content delivery application component and a third supervising component that ensures the correct use of the mobile devices (Dubulakis, 2008). The current thesis discusses the Spatial Content Administrator.

3.1. Spatial Content Administrator

The Spatial Content Administrator, also referred to as Spatial Content Editor or just Content Editor, is the authoring tool of the entire platform. It is the content editing tool used to add, modify, delete and structure any data stored in the database of the system. In few words, it is the gateway through which the content of the system can be managed. Any data presented by the content delivery application of the system has been stored in the database through the Spatial Content Administrator.

In order to understand how the Spatial Content Administration works and how the content author uses it, its various functions must be matched against the way the content is illustrated on the content delivery application that runs on the mobile devices. The deployment of the platform in a museum will be used as the source of the examples used in this thesis, as mentioned in Chapter 1.

A visitor in a museum is given a mobile handheld device with wireless capabilities that runs the content delivery application. The mobile application shows the ground plan of the current floor where the visitor is located and it is fully automated. This means that it automatically informs the visitor about the various museum exhibits as he wanders around the place, without requiring any action at all from the user’s side. The museum is semantically divided into different areas of interest, namely information areas, where an area of interest could be for example a room containing renaissance paintings or a hall full of Egyptian Antiquities. Each
information area can contain one or more items of interest, namely information items, where an information item could be Sandro Botticelli's painting named Birth of Venus or a sculptural work (Michelangelo's David) or an illustrated Egyptian manuscript. As the visitor wanders around the different exhibits, the mobile device is able to calculate in real time the user’s position inside the museum, inform him that he has entered an information area, display the different information items inside this area or automatically show information about the exhibit situated in front of the user.

All of the above information is correspondingly editable one-to-one in the Spatial Content Administrator tool (see Fig. 5).

![Ground Plan of the current floor](image)

**Fig. 5: Map representation in the Spatial Content Administrator**

The main screen of the Spatial Content Administrator displays the map including all contained information areas and information items (see Fig. 5). The content author can select and edit any map from the menu “Maps” (see Fig. 6). However, the mobile device user can only see the map he is standing on. All the information areas and items are clickable. This means that the content author can either select them for editing or relocate them around the map. Furthermore, new information areas and items can be created to address the needs of any particular
use of the platform. Functions for creating, moving and editing information areas and information items are discussed in detail in Chapter 4.

The main menu bar of the Spatial Content Administrator contains the entries “File”, “Maps” and “Help”. The supported functions are displayed in Fig. 6.

The supported functions as depicted Fig. 6 include the program “Exit” under the menu “File”, the selection of the map the content author wants to work on under the “Maps” menu and the “Show Legend” function under the “Help” menu.

The sidepanel of the application contains buttons and controls that perform a variety of tasks. There are controls for the manipulation of information areas (further discussed in Chapter 4, section 4.1), controls used to manipulate information items (further discussed in Chapter 4, section 4.2) and buttons that invoke tools responsible for the manipulation of the data (further discussed in Chapter 5).
The Spatial Content Administrator includes numerous buttons that invoke graphical forms to facilitate the authors to edit all types of data. There are forms that administer:

- The maps stored in the database through the Map Administrator tool (also called Map Editor) invoked by the “Map Management” button (see Chapter 5, section 5.2).
- The boundaries of the stored maps through the “Map Boundaries Management” button (see Chapter 4, section 4.4).
- The different languages supported (i.e., languages in which data is input which is totally different from the Graphical User Interface supported languages) through the Language Manager invoked by the “Languages Management” button (see Chapter 5, section 5.1).
- The navigation scenarios through the Navigation Scenario Manager Interface invoked by the “Manage Navigation Scenarios” button (see Chapter 5, section 5.4).
- The production of the data for the mobile devices invoked by the “Create PDA Data” button and the Call for Synchronization invoked by the “Synchronize PDAs” button (see Chapter 4, sections 4.5 and 4.6).
- And any multimedia information bound to an information item through the Item Content Administrator (also called Item Content Editor) invoked by the “Edit Information Item” button (see Chapter 5, section 5.3).

When the visitor starts his tour around the place, the mobile device displays the current map the user is on (see Fig. 7). As the user wanders around the place, when he comes close to an exhibit, the mobile device automatically recalculates its position and changes its screen to display information (the “home page”) about the current exhibit (see Fig. 7).
Fig. 7: The map, the home page of an item and the Multimedia data types bound to an item as presented on the mobile device

All the information displayed can be edited through the Spatial Content Administrator. Maps are administered through the Map Administrator discussed in Chapter 5, section 5.2. The information contained in the Home Page of every item, as well as all the information bound to it, can be edited through the Item Content Editor discussed in Chapter 5, section 5.3.

In order to understand the use of the Spatial Content Administrator, it is important to document all the steps needed from the moment a content author uses the tool for the first time (i.e., the database is empty) until the point where the database contains enough data for the mobile device application to function properly.

The very first time the Spatial Content Administrator application is launched, the content author is presented with a blank screen (i.e., no maps have been registered yet) and the application sidepanel on the right. All the functions supported by the buttons of the sidepanel are disabled, and only the Map Manager button is enabled. The author needs to register one or more new maps before carrying out other tasks, such as defining new information areas or creating new information points. Using the Map Manager, the author adds the desired maps and clicks save. The application informs the author that the Spatial Content Manager has to be restarted for the new changes to take place and when the author clicks the “OK” button, the application restarts.
After the maps have been inserted and saved in the database, when the Spatial Content Administrator starts it displays the previously saved map. All the function buttons on the sidepanel on the right side of the screen are enabled and available to the content author. Usually, the next step for the content author after inserting the maps is to define the map limits for every map stored in the database. When all the map boundaries have been set, the author needs to launch the Language Manager, which is a graphical interface that can be used to manage the different languages saved in the database. These languages are used for the description of the different information items of the system. When the author adds a new “data language” this language becomes available for content entry unless otherwise specified by the author. When inserting location-specific data, the author can insert the information in any of the “registered” languages of the database that is not marked as offline (see Chapter 5, section 5.1.4). The data language management is further discussed in Chapter 5, section 5.1.

When the desired languages have been inserted, the author opens the Map Manager one more time to fill in the different maps’ information in the newly registered languages. This step is necessary because the mobile application displays this piece of information along with the visiting order of the exhibits when displaying the map to a visitor.

When all the desired languages are added to the database, the author can add the desired information areas by clicking on the map. The Information Area manipulation is discussed in detail in Chapter 4, section 4.1. The author can either add information points to a created information area before creating the next one, or create all the needed information areas and then populate them with information points. The order in which this is done is irrelevant. The author needs to bind to every added information item any data corresponding to it, in as many of the registered languages (that are not marked as offline) as desired.

After adding all the information areas and all the information items that are needed, the author specifies the navigation scenarios. When the visitor is given the handheld device, he is asked to choose among the available navigation scenarios.
These are created according to how much time the visitor wants to spend on site taking into consideration the visitor’s age as well as his interests. The navigation scenarios are created through the Navigation Scenario Manager which is discussed in detail in Chapter 5, section 5.4.

After the navigation scenarios have been created, the content author presses the “Generate PDA Data” button to produce the database snapshot that the mobile devices will use and upload all available data to the FTP synchronization server (see Chapter 4, sections 4.5.2 and 4.5.3). Furthermore, when this process is completed, a synchronization call is sent to the synchronization console server by pressing the “Call PDAs to Synchronize” button (see Chapter 4, section 4.6). When the Spatial Content Administrator informs the author that the synchronization call has been successfully sent, the content author can close the application after pressing “YES” at the prompt which asks him if he would like to send all the new and changed multimedia files to the content synchronization FTP server (see Chapter 6, section 6.3).

The current thesis is organized into chapters according to the functions available in the Spatial Content Administrator. Chapter 4 described all direct content manipulation on the Map. It includes all performed functions that do not require the use of any other graphical form. All the described functions can be performed through direct interaction with the map interface of the application. These functions include among others the manipulation of information areas (Chapter 4, section 4.1), the manipulation of information items (Chapter 4, section 4.2), the definition of the physical boundaries of the maps used (Chapter 4, section 4.4) and the mobile data production and synchronization process (Chapter 4, sections 4.5 and 4.6). Section 4.3 refers to assigning beacon IDs to existing information areas in order to increase the accuracy of the positioning module.

Chapter 5 discusses functions that require graphical forms to be invoked in order for the content author to be able to add the necessary data. The processes described include the management of data languages (Chapter 5.1), the management of system maps (Chapter 5.2), the management of location-specific
content according to the information items (Chapter 5.3) and the authoring of navigation scenarios (Chapter 5.4).

Chapter 6 discusses some more issues that have been addressed while installing and deploying the platform in the National History Museum of Heraklion, Crete. Finally, Chapter 7 contains a summary along with conclusions and a discussion of future work.

3.2. **Spatial Content Administrator Architecture**

The architecture of the Spatial Content Administrator is provided in Fig. 8 at two levels of decomposition detail: (i) primary applications and tools (shaded rectangles and buckets); and (ii) their respective key constituent components (white rectangles with dashed border). Solid arrows indicate intra-process method invocations (e.g., deployment of the Map Viewer component). Different versions of a component are marked with underlined labels, for example mobile / server DB Access Layer, mobile / server Map Viewer, client / server API categories, and mobile / master DBs.

As shown in Fig. 8, the Spatial Content Administrator data and user management UI consists of several subcomponents. These include the Map Manager for the management of the Maps (Chapter 5, section 5.2), the subcomponent for the management of the languages in which the data will be available (Chapter 5, section 5.1), the subcomponent for the management of Navigation Scenarios (Chapter 5, section 5.4), the subcomponent for the management of the Content of Information Items (Chapter 5, section 5.3) and the subcomponent for the production of the mobile devices’ data (Chapter 4, section 4.5). As shown in Fig. 8, the Spatial Content Administrator connects to the Synchronization Console server and uploads data to the Synchronization DB FTP Server during the synchronization process (Chapter 4, subsection 4.5.3), and uploads multimedia content to the multimedia content server during the content synchronization process (Chapter 6, section 6.3).
Furthermore, the Spatial Content Administrator connects to the Multimedia Master Database through the DB Access Layer Component. The display of the data on the computer screen is achieved by the Map Viewer component that uses Microsoft’s Direct3D technology. Finally, the Mobile Device Data Producer communicates with the Ekahau Positioning server to acquire the Ekahau–assigned map IDs to be sent to the mobile devices.

### 3.3. Multimedia Database Schema

The schema of the multimedia master database is shown in Fig. 9. The database schema describes thoroughly the needs of the platform while staying as abstract and as high level as possible as described below. The approach that was followed was based on the convention to create a generic database scheme in terms
of abstraction. In order to accomplish that, the actual data was distinguished from the description of the data for all available languages. As shown in Fig. 9, different tables are used for storing actual data and for storing the multilingual data description. For example, a table is used for storing all the available maps of the system (table Maps), and a different one is used for storing the description of each map in all the available languages (table MapsInfo). This way, every map is saved inside the Maps table which contains all the necessary information to univocally identify each map, such as MapID and MapEkahauID, while the actual multilingual information about the map’s name and the map’s description resides inside the MapsInfo table. The same applies to all data saved in the database, such as Information Items, Information Areas, Data (as an abstract representation of any data type), DataTypes (as an auto incremented integer to distinguish the different types of multimedia content) and navigation schemes.
Fig. 9: The database schema.
4. On Map Direct Content Manipulation

The manipulation of data can be achieved either by interacting directly with the map, or through a series of window forms invoked by application buttons. The interesting part of authoring comes when directly interacting with the map. The user can create new information items, move the existing ones or delete possibly unused or outdated data.

The same applies to the informational areas. They can be created, relocated and deleted through a single mouse click or button push. All these and even more tasks that have conducted by directly interacting with the map are further analysed below.

4.1. Manipulating Information Areas

Information Areas are parts of the map which are of special interest to the users because they contain one or more information items. A map can hold multiple information areas which in turn can hold multiple information items. Areas are represented as convex polygons, and therefore are comprised of three or more vertices. They are stored as a series of geometrical coordinates of points \((x, y)\) starting from an arbitrary vertex, and following necessarily the clockwise direction. This is a limitation of the Direct-3D framework.

![Information areas created clockwise (left) and counterclockwise (right)](image)

Fig. 10: Information areas created clockwise (left) and counterclockwise (right)
Polygons are represented as poly-lines which are vectors of vertices also known as vertex buffers. The renderer of the Direct-3D engine starts by connecting the vertices one by one by drawing lines from the first vertex to the second, then from the second to the third and so on, until the last vertex is connected to the first one. The user needs to input the points following the clockwise direction; otherwise the interior of the polygon can’t be filled correctly (see Fig. 10). When the polygon outline is complete, the renderer paints the interior.

The names of all already created information areas are presented inside a listbox which resides in the sidepanel of the application (see Fig. 11). The user can click on an area name in the listbox to select that particular area on the map. The selected information area is visually distinguished by a green border highlight and green square points representing its vertices.

**Fig. 11:** Information Areas Sidepanel: (a) Create new Area button; (b) Area name inside the listbox; (c) Area name editing textbox; (d) Cancel button

When clicking on an area name in the listbox in the sidepanel, the respective area on the map becomes selected and its name appears inside the name editing textbox just below the listbox. The user then can simply type in the name they wish to give to the area and press the “Enter” key to save the changes.
4.1.1. Creating new Information Areas

The creation of a new Information Area takes just a few mouse clicks. The user has to press the “Create new Area” button, and then click on the corresponding points on the map where the vertices of the new Area will lie (see Fig. 12). When all the vertices are put in place, the creation of the new area can be completed by right-clicking anywhere on the map.

![Fig. 12: Creating new Information Areas](image)

The completed area stays selected and painted in a transparent shade of pink until the user deselects it or selects another object on the map. Every new Area is identified by a default name consisting of the literal “NEW AREA” followed by an auto incremented integer. This name can be changed later through the name editing textbox. When clicking on the “New Area” button, the new area’s name is inserted into the area names listbox in the side, panel so that the area can be selected at user will. The creation of an information area can be canceled at any time by clicking the “Cancel” button just below the area name editing textbox, or by clicking the right mouse button before specifying three or more vertices. A pop-up dialog box informs the user that less than three vertices were specified, and as a result the creation of the current information area will be cancelled.

4.1.2. Moving Information Areas

Information areas can also be relocated by dragging and dropping them using the mouse. When an area on the map is selected, it is automatically ready for editing
such as renaming or deleting, but not for moving. In order to move an area on the map, the user has to press the Ctrl-Key while dragging the area. This was done to avoid accidental relocation of information items and information areas. While holding down the control key, the user can click on an area and then drag and drop it on another location on the map. The area is selected for editing automatically like before. All the contained information points are automatically moved along with the area. Any changes are automatically saved when the user releases the mouse button dropping the area to its new location.

**Fig. 13:** Direct area manipulation with drag & drop facilities

### 4.1.3. Deleting Information Areas

The deletion of an information area is achieved by the “Delete Area” button. This button resides next to the “Create Area” button, and is disabled if the area contains one or more information items. This is a security feature to prevent undesired area deletions that would lead to data loss and user frustration. When a selected area does not contain any information items, the “Delete Area” button is automatically enabled, giving the user the possibility to delete it. Accidentally deleting an empty area is not major inconvenience, since the user can create a new one just in place with a few mouse clicks as described above. When deleting an area, its name is removed from the area names listbox.
4.2. Manipulating Information Items

Information items are items of interest lying inside a particular information area on the map. When referring to museums, an information item could be a particular exhibit somewhere inside the museum. Information items are represented as circles at the location specified by the geometrical coordinates of their center \((x, y)\). These coordinates are used to store the items in the database. If the center of an item lies inside the boundaries of an information area, the item is automatically associated with the area and it is considered that the information item belongs to that area. Moving an item around can result to the association of the item with another area or the complete disassociation of the item, thus affecting the management of navigation scenarios that contain the disassociated items.

Item colours depict the state of the item. A legend describing the various colors is placed on the main menu of the application under the menu item “Help” to help the user identify the state of each item (see Fig. 14). Selected items are painted in green. Items that are not selected but are associated with an information area are painted in blue. All items not associated with any area are painted in yellow (see Fig. 15). Newly created items that are associated with an information area but do not contain any information yet are painted in red colour. Moreover, all items added during an administration session – i.e., during any editing that takes place between two program start-ups – are displayed in double the size of all the other items, so that the administrator can distinguish them easily.

![Legend for item colors as displayed on the map](image)
The names of all the existing information items are held inside a listbox which resides in the side panel of the application (see Fig. 15). The contents of the listbox are clickable, so that clicking on an item name results in its selection on the map. The selected information items are visually distinguished by their green color highlight as mentioned before.

When clicking on an item name using the listbox in the side panel or by clicking on it on the map, its name appears inside the name editing textbox just below the listbox. The author simply types in the desired name and any changes are automatically saved when hitting the return button on the keyboard.

![Fig. 15: Information Items Sidepanel: (a) Item Name List; (b) Item Name Editbox; (c) Create Item Button; (d) Edit Item Information Button; (e) Delete Item Button](image)

### 4.2.1. Creating Information Items

An information item can be created by clicking the “Create New Information Item” button. The new item is created, initialized, placed in the center of the currently selected map and temporarily stored in memory. Newly created items are not associated with any information areas. The user must drag and drop the new item to associate it with an existing information area as soon as it is created. This is very important because “dangling” (i.e., not associated) items cannot exist and hence they are deleted on program exit. The association of a new item with an
existing area commits the item’s information automatically to the database by saving it. New items are painted in yellow because they are not associated with any information area and right after their association they are painted in tomato-red because they are associated but do not contain any information yet. Moreover, new items are double in size compared to the existing ones in order to be easily distinguished by the user.

Every new item is identified by its name which is “NEW ITEM” followed by an auto incremented integer. This name can be changed later through the name editing textbox. When clicking the “New Item” button, the new item’s name is inserted into the item names listbox in the side panel so that the item can be selected by the user. After a new item is created, it can be selected and its name and information edited in the item name editbox and the “Edit Item Info” button respectively.

4.2.2. Moving Information Items

Moving an information item is not different from moving an information area. In order for the relocation to take place, the user must hold the Ctrl-key on the keyboard while dragging the item around the map. This prevents unintentional relocation of information items and areas. It is a level of security for integrity reasons. The item or area doesn’t move if they are accidently dragged with the mouse without holding the Ctrl-key. Even though moving an item outside an information area marks the item as disassociated, the item still remains associated with that area. The reason is that disassociated items are not allowed, and if the item was actually disassociated and then “forgotten” by the user outside the area, it would be deleted on program exit (although there are warning dialog boxes to prevent that). All the information bound to it would be lost causing major inconvenience in many cases. This is the reason why a formerly associated item remains associated with its last stored “residence” area even if it is dragged and dropped outside. Relocating an information item from one area to another causes the item to associate with the new area. This must be done with extra caution, because moving items around can mess up the navigation scenarios. Although
discussed later, the user must bare in mind that if an item that belongs to a navigation scenario is relocated to a different area, it is automatically excluded from the navigation scheme it belonged to. This will be further discussed in Chapter 6 referring to navigation scenarios.

### 4.2.3. Deleting Information Items

In order to delete a particular information item, the user has to select it and then click the “Delete Item” button. When an item is selected, its name appears inside the name editing textbox and the area it belongs to is highlighted on the map. When the “delete” button is clicked a pop-up warning box asks if the user confirms the deletion, because the deletion will remove from the database all the information bound to that item. Deleting an item also removes it from each navigation scenario it belongs to, hence the user must be very careful. If the deleted item was the last item belonging to an information area, the “Delete Area” button gets automatically enabled so that the empty area can be deleted if no longer needed. In addition, when deleting an item, its name is removed from the item names listbox.

### 4.3. Assigning Location Tags to Information Areas

When it comes to the deployment of the system in a museum, the visitor of the museum is provided with a personal digital assistant device (PDA) which has Wi-Fi capabilities. Using the multiple access points that reside inside the museum or archaeological place, the device connects to a server (the EKAHAU positioning system is used) and queries information regarding its relative position in the 3D space. Using the reply it calculates its position and then queries the mobile SQL database snapshot for the appropriate data to present to the user according to its current location. The accuracy of the positioning system is measured in meters and the proximity is a couple of meters each time. This is barely enough because some museums are too crowded with exhibits for the mobile positioning system to work properly providing the necessary accuracy.
To solve this problem, alternative methods for positioning had to be implemented and used to increase accuracy. One of them that can be implemented both indoors and outdoors is the use of Infra Red beacons that emit an id which is unique for each individual exhibit. In this case, every exhibit is equipped with an IR beacon emitting an ID that helps the mobile device to determine its position with increased accuracy. On the administration part certain things were implemented, as described below, to enable the correct correlation of the information items with infra red beacons.

An additional button was added to the user interface on the sidepanel of the application. This button, namely the “Beacon ID” button, can invoke a window form that can be used to assign a beacon id to an information area (see Fig. 16). When selecting an area either by clicking on its name on the area names list or by directly clicking on it on the map, the “Beacon ID” label-button is automatically enabled. This is both a label and a button. When enabled, if a beacon id has already been assigned to the selected area, the “Beacon ID” button acts as a label displaying this beacon id while preserving its functionality as a button (see Fig. 17). For example the button text would say: “Beacon ID: 18” signifying that the beacon id no. 18 has been
assigned to this area. The default value “0” means that no beacon id has been assigned yet. In the case where no beacon id has been set or the already assigned id value needs to be changed, the user can click on the button invoking a small dialog box (see Fig. 18). Using the up and down arrows on the keyboard or typing a new value changes the assigned beacon id that related to the selected information area. The “Save” button commits the changes and closes the dialog box. The buttons label changes to display the new beacon id assigned to that area.

![](image)

Fig. 18: Setting the area tag id

The associations between beacons and areas are saved in the database, so that when the mobile device detects an emitted id it can look up the corresponding area and display the information concerning the information items it contains.

4.4. **Defining Physical Map Boundaries**

For every map that is inserted into the spatial content administrator, the author must specify the map’s boundaries. The map boundaries have to be determined in order for the monitoring application to work. The monitoring application supervises the use of the PDA devices inside the museum physical limits. Having specified the boundaries of every map, the monitoring application can determine if a device has been moved outside the museum’s territory and hence it can send an administrative alert to the employees of the museum while locking down the “fugitive” device preventing further use. The device’s screen is locked and
a message is displayed informing the user that he is moving off the limits. The visitor is kindly asked to return inside the physical space of the museum.

---

**Fig. 19: Setting map boundaries**

Defining the map boundaries is achieved by using the content editor. The whole process is very similar to creating a new information area defining its vertices. To create the limits of the current map, the user must first click on the “Edit Map Limits” button and then use the mouse to specify as many points as he wishes to form a convex polygon that delimits the part of the map that can be explored by the visitor (see Fig. 19). The direction followed by the user when specifying the points can be either clockwise or counter-clockwise, as the interior of the map limits polygon is not filled with color as opposed to creating information area polygons. A single right mouse click completes the polygon which is painted red. If the user wants to make any corrections, he has to press the “Edit Map Limits” button and then delete the edges of the polygon – one by one – by pressing the Delete button on the keyboard as many times as necessary. New edges can also be added by defining new vertices using the mouse. A single right click completes the processing of the limits just like before. The map limits polygon can be moved by dragging and dropping as if it was an ordinary information area, thus giving to the author the possibility for fine tuning.
4.5. Producing the Mobile Database

The correct order of the different tasks that are carried out is that first the content author adds all available data to the SQL database server and then the mobile devices query this server in order to synchronize with it and get the latest updates. The SQL server is located on a server rack at the server room, the content author runs the content administrator application on his own desktop computer and the PDA devices communicate with the servers over a wireless network. When synchronizing with the SQL server, the mobile devices acquire a mobile snapshot of the database, save it locally on some storage card and then query it only when needing information. This is done for integrity and robustness reasons, in order to avoid generating network traffic every time a device needs to query the server. Furthermore this design covers the scenarios where the wireless reception is bad or there is no wireless network at all because all the needed data resides on the local storage of the mobile device.

The production of the mobile database is accomplished by clicking on the “Generate PDA data” button on the side panel of the application. This button does all the necessary functions required for the synchronization of the content management application with the content FTP server. Along with the mobile database snapshot all the necessary multimedia files are uploaded during synchronization according to the mechanisms described below.

4.5.1. Remote Synchronization

Remote synchronization is the process during which the content managing application connects to the SQL database server, downloads the data-tables and creates a compact mobile SQL database snapshot.

Using a mobile database snapshot implies some limitation concerning database capacity. The storage capacity on the mobile devices is already limited to the maximum capacity of the compact flash memory card they can use. The mobile database capacity limit is even stricter. The database snapshot on the mobile device can’t exceed the size of 100MB. That is the reason why multimedia files such as
pictures, photographs, sketches and videos are not stored in the database as binary files. A file descriptor link (URI) describing their location on the local file system is saved instead. This reduces dramatically the size of the mobile database snapshot. Moreover, during the downloading of the data tables, the synchronization mechanism downloads only the tables that have been specified by the user at configuration time, thus saving both bandwidth and storage space.

The remote synchronization is accomplished by utilizing the database publication and subscription options that are offered by the SQL database server through the use of C# coding. The SQL database is marked for publication and the content manager application is registered as a subscriber for that particular database. As a result, when it comes to synchronizing, the content manager is able to connect to the SQL server and create the mobile snapshot database.

4.5.2. Metadata Generation

Because of the limited capacity of the mobile database snapshot, as mentioned before, the multimedia files are not stored inside the database. For each multimedia file the author adds, a file descriptor (URI) describing the location of the file in the local file system is stored in the database. The actual file is copied inside the content directory. When the database mobile snapshot is created, it is uploaded along with all the actual multimedia files on the synchronization FTP server. The mobile devices connect to this FTP server and synchronize the contents stored in their internal storage card with the server’s contents.

Multimedia files tend to consume a lot of storage space because the information stored is usually high definition pictures, video and sound. When the contents are being synchronized with the FTP synchronization server, the network traffic must be kept to a minimum in order to reduce bandwidth usage. This is accomplished by selective uploading and photograph resolution reduction.

Selective uploading means that only the files on the synchronization server that are new or modified are uploaded. Each file has the actual creation, last access
and modification timestamp. This timestamp must be logged so that when it comes to uploading the content, the system knows which files have been changed and therefore need to be uploaded.

Having that in mind, right after the mobile database snapshot has been created for each multimedia file descriptor it contains, its last modification time is logged. This parameter is read from the actual file residing on the local file system. As a result, a metadata XML file is created containing the last modification timestamps for all the multimedia files contained in the database (see Fig. 20).

Reducing the resolution of some photographs such as the Item Home Images and Maps helps avoiding excessive network traffic. This helps both when uploading – synchronizing – the content files and when the items are displayed on the mobile devices. The PDAs suffer from limited capacity of RAM memory. As a result, the displayed maps and the home page images of items have to be resized before sending them to the synchronization FTP server.

To sum up, the following are created: the generated mobile database snapshot, the actual multimedia files inside the content folder on the local hard drive (some reduced in size and some in their original state) and the metadata XML file

```xml
<?xml version="1.0" standalone="yes"?>
<DocumentElement>
  <Remote>
    <Date>2006-06-15T15:30:18+03:00</Date>
    <FileName>1221F03.jpg</FileName>
  </Remote>
  <Remote>
    <Date>2006-07-07T15:53:19+03:00</Date>
    <FileName>mini_ground_floor.bmp</FileName>
  </Remote>
  <Remote>
    <Date>2006-07-07T15:53:29+03:00</Date>
    <FileName>sdfFile.sdf</FileName>
  </Remote>
</DocumentElement>

Fig. 20: The XML schema used for data synchronization
containing information on which files have been modified and therefore need to be uploaded to the synchronization server. As a result, the necessary bandwidth usage is minimized, and then the above are uploaded to the synchronization FTP server.

4.5.3. Uploading to the Synchronization Server

Synchronization could pose the problem of introducing bandwidth bottlenecks in the case scenario that a lot of files are modified constantly and therefore need to be uploaded to some FTP server. However, this is usually not the case in most venues. Additionally, synchronization can take place when no presentation of the content is ongoing.

The content synchronization FTP server is an FTP server running on a server rack at the museum’s server room hosting all the content the mobile devices need to work properly. As mentioned before, a synchronization process is used in order to avoid having to upload all content, which could waste bandwidth and speed. However, usually very few multimedia files change each time the content author does minor changes to the museum’s content. This is the reason why the last modification timestamps XML metadata file was created. This file is compared against the XML file that resides on the FTP server side to decide which files have been updated and thus must be uploaded.

To synchronize the content with the FTP server, a connection should be made to the server to acquire the XML metadata file. Then for each file mentioned in the newly created XML file its previous timestamp must be looked up in the XML file the server holds and if the two timestamps are not equal, that means that the file has been updated and that has to be uploaded. If a file that exists in the new XML document does not exist in the server’s XML document, that means that the file is a new one and has to be uploaded too. Moreover, if a file exists in the server’s XML document but does not exist in the new XML file, it has to be deleted from the content FTP server as no longer used.
Uploading and downloading of files is accomplished by an FTP client written in C# – for the needs of the current project – that has embedded support for uploading and downloading whole directories along with the files they may contain.

Having uploaded all the multimedia files, the mobile database snapshot (referred to as the sdf file named after the file extension used by the SQL server) and the XML metadata file holding the latest modification timestamps, the synchronization is complete. The last remaining step is to request the synchronization of the mobile devices.

### 4.6. Requesting Mobile Devices to Synchronize

Requesting the mobile devices to synchronize is a very easy task. A lightweight server has been coded in C# called the Synchronization Server for the synchronization process. This server detects the connections from both the content manager and the PDAs. The content manager notifies the synchronization server whenever a new update for the mobile devices is available at the FTP synchronization server. When the mobile devices return to their docking station for charging, they connect to the synchronization server asking if there is an update to download (see Fig. 21).

When the content author wants to make the changes he has made available to the mobile devices he has to generate the PDA data using the “Generate PDA data” button on the side panel of the application as described above, and then click the “Call PDAs to synchronize” button which is located right below the previous one. When clicking this button, the content management application connects to the Synchronization Console Server and informs it that a new update is available. The Synchronization Console Server connects to the database and sets the synchronization state fields indicating that new updates are available. Doing so will result in a positive reply the next time any mobile device asks the Synchronization Console Server if there are available updates. When the mobile device gets a positive answer, it connects to the content FTP synchronization server, retrieves all the new...
updates using the timestamps XML metadata file and then inform the Synchronization Console Server to update the field in the database which indicates that the device is up-to-date (see Fig. 21). Storing the devices’ MAC address along with their state (updated or outdated) helps the synchronization server to distinguish the multiple mobile devices and update them all one by one on request.

Fig. 21: The Content Synchronization process
5. Data Editing and Administration

There is a number of data editing and administration tasks that can be accessed from the sidepanel of the user interface in the developed system. These tasks include the management of the languages in which the database content will be available, the administration of the different kinds of maps, and the editing of the information items’ data.

5.1. Data Language Management

The content management system supports multiple languages, so that any multimedia information bound to an information item can be presented in many languages. The content administrator first selects an information item and then presses the “Edit Item” button (an information item must have been previously selected for the button to be enabled) and he is guided through a series of window forms that allow adding, editing or removing data concerning the specified item. For each piece of information that belongs to a particular information-item, the content administrator is given one form for each available language in the database. The whole process is dynamic, so that if a new language entry is added, the corresponding window forms are generated automatically to permit data entry in that particular language. On the other side, removing a language from the database will hide the forms referring to the deleted language. However, the effects of language removal are far more complicated than just the suppression of the editing forms and will be discussed in more detail later on. The manipulation of the available languages includes adding a new language making it available for data entry, editing the information of an existing language, removing a language from the system and taking a language offline or bringing it back online. Each of these tasks will be addressed separately.

The Language Manager form (see Fig. 22) can be invoked by the “Manage Languages” button on the side panel of the main user interface. It contains a list of available languages, a flag preview image box that displays an image of the flag of
the selected language from the list and several buttons for adding, editing, deleting or taking a language online / offline.

![Language Manager](image)

**Fig. 22:** Language Manager; Buttons from left to right: New, Edit, Delete, Take Offline, Close

### 5.1.1. Adding new language entries

Adding a new language entry to the database automatically makes it available for data entry in the Information Item Content Editor. This means that the language entry will cause several window forms to be generated to permit to the content author to add information about any information item in that particular language. The addition of new languages can be done through the “Add” new language button on the Language Manager Form. A new window form appears, asking for the new language name and prompting the user to browse the local file system to find an image representing the corresponding flag. Any changes are saved when pressing the “Save & Close” button of the form. The languages is successfully saved and becomes available in the Item Content Editor discussed in Chapter 5, section 5.3.

### 5.1.2. Editing language entries

Editing the information of an existing language is very similar to entering information during the creation of a new one. The editing form is invoked by the “Edit Language Information” button on the Language Manager. The window form that appears (see Fig. 23) is exactly the same as the creation form, apart from the fact
that it is already filled with all the information concerning the selected language. The content administrator can edit any of the available information, such as the language name and/or the language flag image, and then save or discard the changes by pressing the “Save” or “Cancel” button. Any changes are automatically reflected wherever the data entry supports multilingual information.

Fig. 23: Language Editor: Editing the language information

5.1.3. Removing language entries

The removal of an existing language must be done with extra caution, as it results in the cascade deletion of all the information of every information area, item, map or navigation scenario in that particular language. The deletion is as simple as selecting the language by clicking on it from the language list of the Language Manager and then pressing the “Delete Language” button located below the list. A pop up dialog box appears asking the administrator to confirm the deletion, also mentioning that all information encoded in the language to be deleted will also be deleted. If the user presses the “OK” button, the changes take place instantly.
5.1.4. Taking an existing language online / offline

The functionality for taking a language online or offline was added because of the catastrophic effects of the inadvertent deletion of an existing language. The original idea was to add some functionality that would have the same results as deleting an existing language, but without the actual deletion of data taking place. It is like an undo mechanism that can revert any deletion and bring the database to the state it was before the deletion.

The implementation of this idea was achieved by the addition of an extra field in the languages table in the SQL database, marking each language as either active (online) or inactive (offline). All the selection queries were altered to return only the active languages list instead of all the languages in the table, except for the query that populates the languages list of the Language Manager. That was obviously done for preserving the ability to reactivate an existing language taking it back to an online state.

Taking an existing language online or offline is done by selecting it from the Language Manager’s list and then pressing the Online or Offline button. The online and offline button is labeled according to the current state of the selected language. Whenever a language is selected from the Language Manager’s list, its state is queried in the SQL database. If its state is online, then the Online/Offline button gets enabled and its label changes into “Take Offline”, whereas if the language’s state is already offline, the label of the button becomes “Take Online”. In both cases, the button gets enabled displaying the correct label. Furthermore, when pressing the button, the changes are instantly applied and its label changes to display the function supported.

When the content author takes a language offline, all the data associated with this language are not deleted, but they are hidden instead. The data still exists in the database but are not presented to the user because of the retrieval queries. These queries return only the data associated with the online languages omitting any data associated with any offline language.
5.2. Maps

Maps are the cornerstone of the whole system. They contain all the information areas and the information items of the venue. The content author can select any map to work on from the list of all the available maps under the “Maps” menu in the menu bar of the application. When the content author selects a map from the list, the entry becomes “checked” by a checkmark and the selected map is displayed in the main window application.

The very first time the Content Administrator application is run, the author is presented with the user interface but the main window is blank and all the buttons are disabled except for the “Map Administration” button. The first task that has to be completed is the insertion of the maps of the venue. Once the maps are inserted, the application automatically restarts in order to be reinitialized using the newly added maps. Once the application restarts, the content manager is ready to be used for data entry.

The maps used in the Content Administrator are the same maps used and presented to the visitors by the mobile devices. When a device is trying to pinpoint its current location it communicates with the Ekahau positioning engine and queries it for the device’s coordinates. The Ekahau server responds with the coordinates of a point \((x, y)\) in the 2D space (the maps are 2D), as well as with a map ID so that the device will display the information points of the correct map. Moreover, during the mobile database snapshot creation and just before the beginning of the downloading of the information tables from the SQL database, the Content Administrator queries the Ekahau about the IDs assigned to each map and fills in any missing or changed information storing it to the DB. As a result, when the mobile device gets a reply from the Ekahau server, it will know to which map it refers.

Having all the above in mind, one can conclude that the maps used on the Content Administrator have to be the same maps used by the Ekahau positioning system or the mobile device location calculation will fail. The maps used by the mobile devices are input by the Content administrator. As a result, if all the corresponding maps used are equivalent and come from the same source, the
mobile device will be able to position itself properly. This is a very important parameter that should not be neglected.

The management of the different maps used is achieved through the Map Manager (see Fig. 24). The Map Manager is a window form that contains a list with all the available maps in the application, a preview panel that shows a preview of the selected map in the list, and several buttons that are used to add, edit, delete or perform other functions on the selected map.

![Map Manager](image)

**Fig. 24:** Map Manager; Buttons from left to right: New, edit, delete, Scale, GeoInfo, Close

### 5.2.1. Adding new maps

Adding a new map to the Content Administrator is a very easy task. The content author presses the “Maps Management” button, which resides in the sidepanel of the application, to invoke the Map Manager window-form. When the Map Manager opens, there is an “Add” button right below the maps list. When pressed, a new window form opens and the content author is asked to fill in the new map’s information. There are both necessary fields as well as optional ones. The necessary fields are the Map, URI which is a URI describing the location of the map in the local file system, the map, ID which is a unique identifier used to distinguish the map, the map scale and a check box indicating whether this is an indoor or outdoor map.
The map URI can be specified by clicking the “Browse” button and searching for the map image inside the local tree structure of the local file system. When clicking “OK”, the image is automatically copied inside the content folder to be later uploaded wherever used.

The map identifier is a word or phrase that must be unique for each map in the system. Moreover, it must be the same identifier used when adding the map to the Ekahau positioning server or the map ID lookup during the mobile database creation process mentioned earlier will fail. The map ID is the keywords appearing under the menu “Maps” in the Content Manager menu-bar.

Along with the necessary fields without which the saving of the new map will not be allowed, there are some optional fields that can be left blank at will. These fields concern the multilingual information describing the new map. Every map can have a title and a description string for each language registered in the system which is not marked as offline. This piece of information is not mandatory, because the creation of the map can be completed without it. However, if a mobile device has to be rented for a tour using any of the languages that the information of the map was left blank, the renting will fail.

After specifying all the necessary information, the content author may save the new map by pressing the “Save & Close” button. The window form closes and the user is presented with the previous form of the Map Manager. The new map is added to the list of maps.

5.2.2. Editing map details

Editing the information of a map is similar to creating a new one. The editing window form (see Fig. 25) can be invoked by selecting a map from the Map Manager’s list and then pressing the “Edit” button. The window form that opens is the same as when creating a map, except for the fact that any information previously stored in the database describing this map is already loaded in the appropriate fields. The author can do any desired changes and then press the “Save & Close”
button to commit them to the database. Any changes affecting the image of the map have as a result the restart of the application in order for the Content Manager to be reinitialized with the new data. The user is previously informed that no data will be lost during the restart of the application.

5.2.3. Deleting maps

There is a restriction that applies to the deletion of a map. No map can be deleted if it contains one or more information areas. Furthermore, no information area can be deleted if it is associated with one or more information items. Therefore, no map with information items can be deleted. This is a security feature to prevent accidental deletion of useful data that would cause great inconvenience. As a result, the delete button remains disabled for as long as the map is not “empty”. If a map is empty or if the last remaining information area on the map is deleted, the button gets automatically enabled. This is a quick test to see which maps contain information areas. When on the Map Manager form, the content author can determine whether a map contains any information areas by clicking on it inside the map listbox and seeing whether the delete button gets enabled. If it does, it means that the map doesn’t contain any information areas.
5.2.4. Managing map geographical information

The location sensing regarding the mobile devices is accomplished by querying the Ekahau positioning server about the device’s location which is calculated by the triangulation of the signal strength according to the various access points installed at the venue. This has a variable accuracy of about a couple of meters. Because of the non-standard deviation, the positioning system can’t be held reliable and responsible to blindly guide the mobile application. For that reason, alternative ways of determining the position of the user were introduced. Infra Red beacons emitting a unique beacon id for each exhibit were installed as mentioned in previous chapters. The mobile devices receive the emitted id and looks up the area where it belongs.

However, the Infra Red beacons are too expensive and sometimes useless when it comes to archaeological sites that are outdoors. Sometimes the exhibits are not approachable to the visitor and can only be looked at them at a distance. For these cases, a GPS approach has been implemented. The mobile devices used are GPS capable. Furthermore, the GPS reception is free and there are available APIs to drive the devices’ GPS cards. Therefore, the mobile application can receive the GPS signal from numerous satellites and calculate the position of the visitor holding the device. In order for this to work the map has to be calibrated. This means that the application has to know a priori the geographical coordinates of the top right, the bottom right and the bottom left corners of the map. This can be measured once for each archaeological place using on site calculation or using satellite photographs as maps whose coordinates are known. Having the necessary coordinates calculated, the content author has to input them in the system.

For each map in the Map Manager’s list of maps, there is a button called “Geographical Information”. This button invokes a window form which has a schematic representation of the map, having the three geographical points whose coordinates are needed marked in red. There are six input text boxes (see Fig. 26), two for the two coordinates of each of the three points. The author has to fill all of them and press the “Save” button. The changes are saved in the database and are
passed to the mobile devices on the next synchronization after the mobile database snapshot has been built.

![Fig. 26: Editing map geographical information](image)

**5.2.5. Moving and scaling while preserving ratio**

Sometimes the content administrator may want to change an existing map, replacing it with a bigger one either in resolution or in actual dimensions. That would mean that all the information areas and all the information items would have to be moved to their new locations. In cases the new location is some pixels on the left, right, up, or down, a simple translocation is sufficient. This applies to scenarios where the new map is bigger than the old one to include an additional part of the archaeological site or the museum.

This functionality has been included in the Map Administrator through the button “Scale”. A new window form pops up (Fig. 27) asking the author about the magnitude of the translocation. The distance between the old and the new location is given in pixels for both the horizontal and the vertical axis. When pressing the
“Perform Scaling” button the changes are immediate. All information areas and information points are moved according to the dimensions given.

![Image](image.png)

**Fig. 27: Moving and scaling information areas (including contained information items)**

### 5.2.6. Skewing information areas

The previous sub-section addressed the cases where the resolution of the map was constant, but the image size was increased to include more parts of the museum or the archaeological space. There are scenarios in which the displayed area dimensions are preserved while increasing the image resolution. This results in the dislocation of all information areas and information points. In these cases, a simple translocation would not solve the problem, because the information areas would not only have to be moved but they would also have to be represented as polygons with a bigger area than before. A translocation would move the information areas to the correct location but wouldn’t make them bigger. The functionality for making the areas bigger was included in the scaling functionality for the maps.

When selecting a map and pressing the “Scale Map” button, the author is can resize all the information areas of the selected map making them bigger or smaller according to the percentage of the transformation. Therefore, the content author has the ability to make information areas bigger or smaller to match the map size. The resizing percentage has to be determined for both the horizontal and the vertical axis separately. Giving the same percentage for both axes would result in the resizing of the information area preserving proportions. However, if the two percentages differ, the result will be an information area skewed along the axis with
the greater percentage value, thus allowing for asymmetrical changes on the dimensions of the maps.

After specifying the two percentages, the author needs to press the “Perform Scaling” button. Any changes take place instantly. All the information areas are transformed according to the percentages and all their information items are moved inside the area to adjust to the new area size.

5.3. Information Content

Each exhibit or point of interest of a venue is represented as an informational item inside an informational area on the map displayed by the Content Administrator application. An exhibit can be described by different types of multimedia content such as texts containing short and detailed descriptions, photographs, sketches or videos (see Table 1). Any of these types of multimedia content can be assigned to an exhibit using the Information Item Content Editor, invoked by the press of the “Edit” button under the category Information Items on the side panel of the application. In order to understand how to use the Item Content Editor, it would be useful to first see how all this information is displayed on the mobile device when a visitor is exploring an information item.

<table>
<thead>
<tr>
<th>Text</th>
<th>Dear visitors, welcome to the Archaeological Museum of Mycenae, a modern local museum that opened up its gates in 2003. It is hidden in the northern side of the acropolis. It was build within the archaeological site providing a shelter for the archaeological artifacts found within. We hope you enjoy your stay.</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Title</td>
<td>Mycenae Archaeological Museum</td>
</tr>
<tr>
<td>Image File</td>
<td>12M01F01b.jpg</td>
</tr>
<tr>
<td>Image Sketch</td>
<td>Acropolis hill</td>
</tr>
<tr>
<td>File</td>
<td>12M01F02.jpg</td>
</tr>
</tbody>
</table>

Table 1: Information Item Content snippet for a museum
When the mobile device senses that the visitor is in front of a particular exhibit, it displays the exhibit’s home page. It contains a picture of the information item, a title, and a sound description of the title. The visitor can click on the buttons of the device to view a short description of the item, a more detailed description and some more multimedia content containing texts, photographs, sketches and videos. All this information is displayed as a series of pages. This is why they are referred to as text pages, photograph pages, sketch pages and video pages. This section describes how the Item Content Editor is used to input all this information.

![Fig. 28: Information Item’s home page](image)

### 5.3.1. Home Page

The Information Item Content Editor window form consists of a series of tabbed panels and sub-panels. The main tab selected is the one referring to the information displayed on the home page of the item (see Fig. 28), whereas there are more tabs on the top of the window to manage the short and detailed description of
the item as well as any additional multimedia content. The last tab is dedicated to auditory information and will be further explained later.

The home page tab consists of two main panels. The first one is on the left of the window and contains an image box-holder where the content author is asked to input the image displayed on the home page of the item. The image can be browsed in the local filesystem by pressing the “Browse” button.

The remaining information on the item’s home page is multilingual. This means that the system needs to be able to display the title of the home page as well as playback the sound description of the title in the visitor’s preferred language. The right panel is multi tabbed.

The author can enter the data that describes an information item in the languages that are supported. The languages supported are presented on the right side panel of the graphical interface in the form of tabs. There is one tab for each language registered in the database that is not marked as offline. The language of every tab has an icon of the flag of the language. These tabs are automatically generated according to the results of the corresponding database query. Each language tab contains the title textbox that will hold the title string of the page and a textbox that will hold the sound file with the recording of the title in the specified language. Once again the sound URI, which is actually the filename of the sound file, is stored in the database whereas the actual multimedia file is copied inside the content directory of the computer the Content Administrator is running on. Any changes are saved by pressing the “Commit” button at the bottom of the form. A progress bar informs the user about the progress, although saving doesn’t take much time.

5.3.2. Description and Multimedia

Beside the Home Page tab, there are the “Short Description”, the “Detailed Description” and the “Multimedia Content” tabs corresponding to the short and long description of the item and to additional multimedia content respectively. All of
these tabs hold multimedia content such as text, photographs, sketches and videos describing the information item being edited.

The short description tab stores all the information displayed when the short description button of the mobile application running on the mobile devices is pressed. If the visitor is in a hurry and wants to see only the most important multimedia content describing the item, that is what he gets. If he wants to receive further details, he can press the detailed description button to see content committed under the detailed description tab and the multimedia content button to see any information committed under the multimedia content tab. The contents of these tabs can be text pages, photographs, sketches or videos.

The layout of all three tabs is an empty panel, with capability to hold multiple sub-tabs. On the top of the panel there is a drop down list containing four entries. These entries correspond to the text, photograph, sketch and video sub-tabs respectively. The content author may select the appropriate content type from the drop down list and then click the “Add” button which lies next to the list. There are also buttons for deleting a sub-tab, or moving it to the left or to the right. The ordering of the sub-tabs is very important, as the contained information is saved in that order and is also displayed in the same order on the mobile devices.

Although the layout of the application is the same under all three sub-tabs concerning the photo, sketch and video sub-tabs, the text layout is different in the multimedia content sub-tab than in the other two. All of the different types of content pages will be discussed in detail.

5.3.2.1. Text Description
Both short and long description text pages have the same layout. The text page consists of the actual text and a sound file corresponding to the text. They both exist in as many copies as the languages registered in the database and marked as online.
The text page panel layout consists of a multi tabbed panel containing one tab for each language. Each tab has a multi line textbox that holds the actual text and a sound path textbox holding the URI to the sound description file (see Fig. 29). The content author can type in any text and then click on the browse button to search for the sound file on the local file system. The sound file is copied inside the content directory of the computer on which the Content Administrator application is running, and the path of the file is stored in the database. Any changes can be saved by pressing the “Commit” button at the bottom of the form. If any tab-pages are left blank, they are not saved at the database at commit time. This means that there isn’t any information saved concerning these particular languages and not that empty information is saved.

Fig. 29: Text description page (information item’s short description)
5.3.2.2. Photographs and Sketches

Photograph and Sketch sub-tabs are similar to the home page of the item. The content author may select the photo or sketch page multimedia type from the drop down list and then press the “Add” button.

The layout is the same as the layout of the item’s home page. The tab is divided into two panels. The left one contains the image holder where the corresponding photograph or sketch will be added. The content author may click the “Browse” button to search for the sketch or photograph on the local filesystem. When clicking on the “OK” button, the file is copied inside the content directory of the computer and the image is displayed inside the image holder.

Fig. 30: Photograph description page (information item’s short description)

The right panel contains the textbox that will hold the description of the photograph or sketch to the left and a path textbox that will hold the sound file
containing the audio description of the image (see Fig. 30). The content author may write anything inside the description text box and then click on the browse button to search for the audio description sound file on the local filesystem. The home page, as well as any tab or page that contains a sound path textbox, contains the browse button to facilitate the user searching for the file as well as two buttons for playing back and stopping the sound so that the content author is able to preview the selected sound before saving it to the database.

All the information contained in these types of pages is multilingual. This means that the photograph and sketch pages contain as many sub-tabs as the languages registered in the database and marked as online.
5.3.2.3. Videos

All of the information contained in the video pages is also multilingual. The video page contains a series of sub-tabs, one for each registered online language. The layout of all sub-tabs is the same.

Every sub-tab contains three textboxes, one to hold the path of the video file, one to hold the description of the video and one to hold the path of the audio description of the video title (see Fig. 31). The content author can click on the browse button next to the video path textbox to search the local filesystem for the video file. When pressing “OK”, the path is displayed in the path textbox and the user can preview the selected video by clicking on the “Playback” and “Stop” buttons located below the textbox. When the author fills in the description of the video file, he can click on the browse button beside the sound textbox to search for the audio description file. When selected, the sound file can be previewed using the “Start” (play) and “Stop” button below the textbox.
5.3.2.4. Multi Paged Text

The multi paged texts is a special type of multimedia content that can be stored under the Multimedia tab. The multimedia content described so far could be of text type but each text page was only a single page of text. The multi paged texts contain multiple text pages grouped under the same title.

The layout of multi paged text pages is a little more complicated than that of single paged texts. The layout contains buttons for adding, ordering and deleting multi-paged texts. Each multi-paged text page contains a multi-tabbed panel containing the title textbox and the audio description textbox (see Fig. 32). This is because the title should be filled in for each registered online language. The sound
path textbox will hold the audio description of the title in each of the registered languages.

Below the multilingual title information there is a multi-tabbed panel to hold the actual multilingual text pages. This panel contains the actual text pages. Each text page has a multi line textbox and a sound descriptionTextbox just like the simple text pages. Both textboxes are inside a multi-tabbed panel because as many copies are needed as the actual registered online languages. There are also buttons to order the text pages, add additional ones, or delete them.

The content author needs to fill in the title of the text and the title’s sound description in as many languages as he wants, and then add the text pages.

5.3.3. Adding auditory content

There is a special renting mode called the auditory mode. When the visitor is about to rent his electronic guide at the reception of the museum, he can choose between the multimedia mode and the auditory mode. The multimedia mode is the most commonly used mode and includes all the multimedia content such as photographs, sketches, text, and videos. The auditory mode is a sound only mode. The screen of the device goes black and only sound can be heard. This mode is used for visually disabled people or for people that are not interested in any visual multimedia content. The last tab of the Item Content Editor is the auditory tab. It is a multilingual tab that contains auditory information.

The auditory information is a text describing the information item and a sound file containing the audio description of the text (see Fig. 33). The text is used in case there is a speech synthesizer available, so that the sound file with the audio description will not be needed. The content author may input the appropriate text in the multi line textbox and then click on the browse button to browse for the audio file on the local filesystem. The audio file can be previewed using the “Play” and “Stop” buttons like before. The described layout is similar to the layout of text pages. The auditory tab can be described as a dedicated tab for special purpose text pages.
Changes are saved during the commit of the item when pressing the “Commit” button which is located at the bottom of the form.

![Fig. 33: Auditory content tab in the information item content editor](image)

5.4. Authoring of Navigation Scenarios

When the museum visitor is given the mobile device to guide him around, he is asked to choose among the different navigation scenarios available. There are different scenarios predefined for each archaeological place to suit the needs of children, adults, visitors who are in a hurry and visitor who want to visit each and every one exhibit of the place. The management of the navigation scenarios is performed using the navigation scenario manager.

The main user interface of the Content Manager was built around the management of information items and information areas. The main concept is that the content author has an overview of the place while editing the different pieces of information. The “Manage Navigation Scenarios” button hides all the functions that
have to do with information areas and information items, thus transforming the interface into the Navigation Scenario Manager.

The navigation scenario manager side panel consists of the navigation scenario list, the navigation scenario name editing textbox and four buttons used for the creation, editing, deletion and clearing of the information items of a scenario. To exit the Navigation Scenario Editing mode, the content author must press the “Exit Scenario Editing Mode” button. The user interface changes back into the normal Content Administrator interface.

**5.4.1. Overviewing**

The content author is given the functionality to overview any navigation scenario stored in the database. Clicking on any navigation scenario from the scenario list selects it and draws it on the screen (see Fig. 34). While being in the navigation scenario editing mode, the user has no other functionality over the items and areas but from moving them around. While working on a navigation scenario, the author can move around any information area or item watching the scenario drawing on the screen update at real time. The first scenario item to be visited is depicted by a little green square and the last one by a red square. All other items belonging to a scenario are linked together using a thick green line in a “connect the dots” game style.

The author can also change the name of any selected navigation scenario just like changing the name of any selected information area or item. When selecting a scenario from the list, its name appears inside the scenario name editing textbox. The author can write the desired name and any changes are automatically committed to the database when pressing the return key on the keyboard.
5.4.2. Creating new navigation scenarios

To create a new navigation scenario, the author must press the “New” button. A new empty scenario is automatically created and added to the database. It is named with the string “NEW SCENARIO” followed by an auto incremented integer. Upon creation, it is added to the scenarios list and the scenario name editing textbox is changed to display the scenario’s name. The new scenario gets selected in the list but nothing is displayed on the screen because the scenario is empty. No information items belong to it yet. It can be renamed to anything the author wants by using the name editing textbox.

5.4.3. Editing navigation scenarios

When selecting a navigation scenario from the scenario list of the Scenario Manager, the “Edit” button gets enabled. The edit button provides a window form for editing the selected scenario. This form contains two lists of items. The list on the left is a list of selected items and the list on the right is a list of all the remaining items of the museum. For each information-item shown in each list, additional information is displayed such as the id of the item, the id of the information area it belongs to and its name. The name of the item is show in the language selected in the language drop-down box just above the two lists.
To include a navigation item to the selected scenario, the author must select it from the remaining items list and then press the move left button on the form. The item is added at the tail of the selected items list. Further rearrangements in this list can be made using the move up and down buttons of the form. When selecting one or more items from the selected information items list and then pressing the move up button they are all moved one position upwards in the list. The content author has the freedom to arrange the items in whatever order and then save the scenario by pressing the “Save” button. When creating a navigation scenario, it is important to have in mind that no scenario can be created in which the visitor will enter an information area twice. This is an implied limitation of the mobile device application in order to avoid circular paths that have no meaning. If the user does create such a scenario and tries to save it, he will be informed about the conflict and will be asked to make the necessary changes in the selected items list.

For each edited scenario, the user can specify a scenario image to be displayed next to its name on the mobile device. The image can be specified by
clicking the “Browse” button and then searching for an image on the local filesystem. Pressing the “Save” button saves the image to the scenario.

Each scenario has a description string in all the existing languages that are not marked as offline. A series of textboxes – one for each language – accommodate this functionality. The scenario descriptions are saved when pressing the “Save” button.

5.4.4. Clearing navigation scenario items
An extra button was added to the interface to allow the content author to “clear” a navigation scenario from any items it might include. The “Clear” button when pressed informs the user that the selected scenario will be cleared, and if the user wants to continue it clears the selected scenario without deleting it. The scenario preserves its name, stays selected in the scenario list but the drawing on the screen is erased as it does not contain any information items any more. This functionality is some kind of “shortcut” to the end user because he doesn’t have to edit the scenario and delete the information items it consists of, one by one.

5.4.5. Deleting navigation scenarios
The “Delete” button of the Scenario Manager deletes the selected scenario, which is removed from the scenario list and gets deleted from the database. The user is formerly informed about the deletion and asked to confirm the deletion. If the answer is positive, the changes take place instantly.
6. Application in a Museum

The Content Manager Application was developed in order to be deployed in several museums and archaeological sites throughout Greece. So was the Electronic Guide, the mobile devices’ application.

In the related set-up, all the required servers such as the Ekahau positioning server and the SQL database server run in the museums server room. The Content Administrator application runs on the desktop PC of the content author, supporting central administration. Registration and renting applications for the mobile devices are also provided, running on a desktop PC at the reception of the archaeological place. The visitor asks the employee of the museum for an electronic guide device which is rented in the visitor’s language and handed over to him. All these mobile devices synchronize with the SQL database and the multimedia content of the museum during the device synchronization process as described in previous chapters.

6.1. Central Management

Although the Content Administrator was designed to be run on the desktop computer of the content author, it was extended to support central management. This means, for example, that necessary changes can be performed centrally from the Hellenic Ministry of Culture by connecting to any of the museums deploying the system.

One more FTP server were needed to support the central management of the museums. This server is the museum’s content server, holding all the multimedia content included in the database and therefore needed for the Content Administrator application to run properly.

When the Content Administrator is started, it shows a dialog box asking the user if he would like to connect to a particular museum and download the content needed by the application in order to run properly (see Fig. 36). The user must select a museum from the predefined museum list and then press the “Start Downloading”
button. A progress bar informs the user about the status of the file downloading and the currently downloaded file is displayed in the dialog window. The whole process is asynchronous, so that the user can cancel the downloading at any time by pressing the “Cancel” button. When the downloading of the necessary files is completed, the application continues normally.

![Connecting to the remote FTP server to fetch needed content files](image)

Fig. 36: Connecting to the remote FTP server to fetch needed content files

When deploying the Content Administrator application in a museum, the available museums list is disabled to prevent the content author to have access to any other museum’s content but the one he is working for.

### 6.2. Connecting to the Museum’s Database

After fetching the content from the museums FTP content server, the application displays a dialog window asking for the user to input some connection information such as the server IP where the SQL server is running, the name of the database and the username and password of the SQL server’s user account.

All this information could be entered at the configuration file of the application so that the content author will not have to complete these fields every time the application is run. However, when it comes to central administration this info should be completed each time the administrator would like to connect to a
museum. Having completed the fields – all of which are mandatory – the user can try to connect to the database specified by clicking on the “Connect” button.

After the “Connect” button is pressed, the application informs the user that it is trying to connect to the remote database (see Fig. 38) and if the connection succeeds there is one more level of security before the administrator can have access to the map and do any changes to the museums database. An information table has been created containing the usernames and passwords of all the administrators that can have access to the database. This table is stored inside each database. When the administrator successfully connects to the remote SQL server, he is asked to authenticate himself using his username and password. The input is checked against the entries of the table stored inside the connected database, and if the administrator is authenticated he is allowed further actions. If the administrator is not authenticated, he can’t continue any further and the Content Manager user interface is never displayed.
6.3. Uploading changed multimedia files to the remote FTP multimedia content server

When the content author has performed all the maintenance tasks needed and tries to close the Content Administrator application, a pop up dialog window appears asking him if he wants to upload any changed multimedia content back to the museum’s content FTP server. If the user clicks on “Start Uploading”, the application starts uploading any new or changed multimedia files to the museum’s FTP server. The uploading mechanism is sophisticated enough to distinguish new and modified from unchanged files, minimizing the bandwidth usage.

The downloading and the uploading mechanism uses the same techniques described in the mobile devices synchronization section. The server holds an XML metadata file containing the timestamp of the last modification time of every file on the server. When the content files are downloaded from the server, only the ones that have a more recent timestamp than the copies on the author’s PC are actually transferred. On the other hand, when uploading files to the FTP server, both new files and modified ones are transferred so that the FTP server always remains up-to-date.
7. Summary, Conclusions and Future Work

This Thesis has presented the key components of an integrated platform for the authoring and uptake of location-aware mobile information systems, putting emphasis on specific novel features reflecting requirements that emerged in the course of large-scale real-life deployment. The development of the overall platform required three years (2004-2007), while authoring and installation for the 15 major museums and archaeological sites of Greece started in February 2007. Within the integrated platform, data are offered in multiple representations as different types of multimedia files in order to meet the individual needs of different museums.

The Spatial Content Administrator application reported in this thesis is the key component for authoring and organizing spatial data. It supports numerous functions for the direct – on map – manipulation of data as informational areas and informational items. It facilitates the spatial data content administration of large-scale spatial location-aware information systems by providing a straightforward and easy-to-use data manipulation tool that enables content administrators to add, edit, update, delete and bind multimedia information (e.g., related to an exhibit of a museum) to a related location on a map. Moreover, it enables unskilled users to use this administration tool without the need to remember complicated processes. It gives system administrators a visual overview of multimedia content that is associated with an actual geographic location. Finally, it associates complex tasks and processes with single button presses in order to simplify the maintenance process of large-scale information systems.

Although the described application has many features that facilitate the data management and the maintenance of the electronic guiding mobile devices, there are some more thoughts and ideas to be implemented as additions in the context of future work:

- To begin with, resizing support could be added on information areas and map boundaries. When an area is selected, its vertices are displayed as square points. These points could be dragged and dropped, allowing area resizing which would be more convenient than deleting the area and creating a new one. The same could apply to map boundaries polygon. Additionally, gravity
could be added to the cursor when creating new information areas or map limits polygons to facilitate the creation of orthogonal areas and boundaries.

- Navigation scenarios could be created by direct clicking on the information items that must be included in the scenario. There is research done for a scenario creation mode during which the selected scenario can be altered by adding, ordering and deleting information items. Moreover, direction arrows could be displayed while displaying a navigation scenario, in order to show the direction of the selected path on the map.

- One more addition to the Content Administrator could be the support of undo and redo over the different executed tasks. This would be of great convenience, as the user would be able to undo and/or redo all of the tasks that are carried out using the Content Administrator application.

As localization technologies improve and offer more accurate positioning information, mobile location-aware information systems become more and more popular and widespread. As a result, the entire platform may find numerous fields of use in the future. Apart from exhibition areas the platform may be used in the deployment of mobile electronic city guides offering information to tourists about nearby archaeological sites, restaurants, theaters, cafés etc. Furthermore, the platform may be used in the deployment of electronic guides for multi-storey shopping centers or as a framework able to provide information to robots acting as guides in order to be able to find their way around a place or provide different types of location-related information.

Moreover, the database schema could be altered to permit the storage of additional information describing the maps of the system. Additional information could be added for the description of the topology of the place presented by each map, including walls, areas that are out of reach and possible obstacles. All this information could be used for advanced path planning so that the navigation scenarios could include a proposed navigation path for users or robotic guides to follow, in order to move as efficiently as possible. Specifically for the robotic guides,
the stored topological information could describe valuable exhibits as obstacles or areas that are out of reach in order to prohibit the robotic guide from approaching them, hence reducing the possibility of accidental damage to sensitive exhibits.

In addition, the platform could be extended to support the storage of user information. Users could be permitted to store their experiences, knowledge and feelings when visiting a place that is covered by the information system. Filtering functionalities should be implemented for the users to be able to select, filter and view location-related information stored by other users. Without the proper filtering facilities, the information space could become cluttered because of the nature of the information stored by the users. Moreover the user data-entry process should be properly restricted by the user interface so that the users should be permitted to store information only under several pre-defined categories.

Besides user comments and experiences the platform could be extended to permit information editing by the users. If a user disagreed with the displayed information he/she could be able to send a notification to the content administrators indicating the part of the stored information that should be changed along with all the necessary changes that should be made. Finally, the content authors should be able to review all these comments and make the necessary changes when they conclude that the stored information should be changed.

Concluding, the Spatial Content Administrator is a very powerful application that is currently being deployed in museums all over Greece. It is a very convenient means of managing all the multimedia content of a museum as it is very simple and easy to use allowing both experienced and novice users to manage, administer and efficiently maintain the entire information system.
8. Acknowledgements

Part of this work was funded by the Hellenic Ministry of Culture, via a subcontract from SIEMENS to FORTH, for the installation and deployment of the platform at fifteen museums and archeological sites across Greece, including Acropolis, Olympia, Knossos, Delphi and Mycenae. The total project budget was 11 Million Euros, with installations and pilot deployment scheduled to be completed before summer 2008.

The platform is being installed and deployed at the Museum of Natural History on Crete Island under the terms of the “Panorama” project which is scheduled to be complete by the end of October 2008.
9. References


