2013 International Conference on Electronic Engineering and Computer Science

Digital Graffiti – a Smart Information and Collaboration System

Wolfgang Narzt, Gustav Pomberger*

Department of Business Informatics – Software Engineering, Johannes Kepler University of Linz
Altenberger Street 69, A-4040 Linz, Austria.

Abstract

This paper illustrates the development process and the results of a research business informatics project. Inspired by an innovative idea and the objectives for new communications technology, we present the design process with the logical and technical architecture of a software framework “Digital Graffiti” as the technological basis for the development of software-intensive, mobile, location-based systems. As a result, we describe a prototype implementation of a mobile, smart, information- and collaboration system essential from the perspective of business informatics.

Keywords: Digital Graffiti; Location-Based Service; Information and Collaboration System

1. Classification

In the course of an earlier research and development project named “INSTAR – Information and Navigation Systems through Augmented Reality”, an innovative augmented reality car navigation system has been designed and implemented as a prototype (see Narzt et al. 2003, 2004 [1][2][3][4]) and accredited for patent (EU patent EP1415128B1 and US patent US7039521B2). The published visualization and implementation paradigms that evolved from this project have been picked up in nearly identical form by navigation system producers for implementation of their current and future product versions (see, e.g., [5][6][7]).

E-mail address: wolfgang.narzt@jku.at
In the course of this research project, the problem arose that in the live image of a camera, objects had to be identified, highlighted, and provided with additional information (e.g., a filling station and its current fuel prices when the fuel level sensor drops below a given threshold; see Fig. 1a). It soon became clear that this problem could not be solved with known methods and thus required a new solution. Exploring a solution to this problem led to the idea of virtual sensors to be implemented with an innovative technology that first had to be developed—we call it Digital Graffiti technology, elucidated in the following sections.

The name Digital Graffiti results from an initial thought where users were supposed to perceive hands-free drawn virtual information through an augmented reality display. Although it would have been technically manageable to implement this idea as illustrated in Fig. 1b and 1c utilizing tilt sensors and compasses, we reduced a Digital Graffito to information without expansion but restricted to a geographical point carrying a visibility radius for perception.

2. Idea and Goals

Inspired by the problems identified in the INSTAR project, the idea arose to define a technological platform that enables linking any position in three-dimensional space with one or more information elements of any kind (e.g., text, image, sound, video, hyperlink), providing this pair (geo-position and information element) with a visibility space and a set of recipients, and, when one of the recipients moves into the visibility space, transferring to it in suitable form the information elements associated with the geo-position at the center of the visibility space (for related developments see e.g., Liu et al., 2011 [8] and Ait-Cheik-Bihi et al., 2011 [9], Indulska and Sutton, 2003 [9]). Thus the filling station and the car in the example given above serve both abstractly and in reality as (virtual) sensors that, depending on their relative positions and their states, trigger a (sequence of) action(s). Such a technology platform seemed to be viable for various applications.

So, the major goal was to design a technology platform (in the form of a framework), modeled and implemented as a prototype in order to open previously unavailable and innovative information, navigation, and collaboration options that can be used in industry, business, and administration to improve business processes, as well as in private domains. In particular, the technology platform and the underlying architecture are to enable depositing information in the form of Digital Graffiti on mobile devices (notebooks, PDAs, mobile phones) at any location in public and private space and consuming such (i.e., linked to the geo-position of the device). Furthermore, the platform is to enable automatic control of electronic actions (e.g., opening a gate, starting or stopping a machine, triggering a measurement or transaction), i.e., without any additional (manual) action, when a given device is in the vicinity of a specially equipped digital graffito (a virtual sensor). A digital graffito is to be provided with access (more precisely, visibility) authorization that relates to a person, a device, a software system, etc. or that results indirectly through the settings of a certain interest profile.
Digital Graffiti technology is also to enable real-time determination of the locations of a selected group of persons or devices (requires mutual, revocable consent of system users) as well as adequate authorization management (who can log into the system with what authorizations, who can determine whose position, who can provide which information/graftito to whom or consume such, etc.).

3. Architecture

3.1. Logical Architecture (Excerpt)

Due to space limitations, we must omit details of the ca. 450,000 lines of code of the software system representing the new technology. Additional publications can be found in Narzt et al. 2007 [11], and Schmitzberger and Narzt 2009, 2010 [12][13]. Generally the definition of goals lead to the choice of the client/server model as the basic architectural model. Because on the one hand the geo-coded information elements can be spread across the entire globe and on the other hand system benefit rises with the number of user groups and therefore huge quantities of data must be managed, the system cannot be implemented on the basis of a simple client/server model; instead, this requires a design and mechanisms that enable unrestricted worldwide operation; i.e., the design must also accommodate scalability (and the associated performance control options).

The architectural model that we developed for a worldwide, scalable system is oriented to the basic model of cellular telephony: A network of Digital Graffiti providers (DGPs) distributes the load ensuing from (asynchronously) communicating users. Each DGP serves a manageable number of users and is responsible for communication with affected users that are assigned to other DGPs. Each user is identified unambiguously worldwide via a data tuple consisting of provider ID and a sequential number. Information elements (graftiti) that have been stored in the databases of DGPs (by content providers) are stored according to the same schema. An additional type id avoids possible confusion between graffiti IDs and user IDs. Fig. 2a illustrates this detail in the logical layer architecture.

![Fig. 2. (a) Scalable network of Digital Graffiti providers; (b) Lookup mechanism for regional servers](image-url)
If a user creates a graffito, then it is stored in the database of his assigned DGP. If this graffito is addressed to a user served by another DGP, then via a special propagation mechanism the other DGP and thereby the recipients are notified of the existence of the graffito. Due to the unambiguous ID of a graffito, it is transparent for the recipient which DGP manages the graffito.

If we wish to accommodate that content providers often offer only regional information and do not want to store their information on the server of a DGP, especially if security-related or confidential information is involved, then the architecture model must be enhanced. Here we designed a component that enables a content provider to administrate its data on its own servers and still integrate it in the system while decoupled from DGPs. We call this component a regional server. It enables any content provider to establish its own range of operations including the associated servers. These regional servers are integrated into the data management network via a mechanism that functions similarly to Domain Name Services (DNS) on the Web, where a server name is assigned to an unambiguous IP address. The component that provides this mechanism is called Position Lookup Service (PLS). Fig. 2b illustrates this design detail in the logical architecture layer.

This architecture model implies that each user is always connected only with his DGP. When the user moves into a (geographic) area where one (or more – overlapping of regions is possible and permitted) regional server is installed, the DGP connects to these regional servers and transfers the information elements provided by the content provider. This design of the logical architecture layer ensures that the content providers can manage and provide their information independently and decoupled from the DGPs.

3.2. Technical Architecture (Excerpt)

The components modeled in the logical architecture layer along with their collaboration mechanisms and their interfaces are systematically represented in the technical architecture layer. Now each abstractly modeled module is classified for its technical implementation in hardware and software units or in their interfaces (operating system specific components, virtual machines) and middleware components. This architecture design enables implementation as a division of labor by different developer groups. Both from a software engineering perspective (see Sommerville 2011 [14]) – a modular, decoupled and thus testable and extensible system architecture – as well as from an operational viewpoint, this is a significant attribute of this research and development project.

The refinements of the provider and regional server concepts modeled on the logical architecture layer (and the associated client components not described here due to space limitations) merge in the technical architecture, as represented in the overview in Fig. 3. The technical architecture description also includes determination of the programming languages, techniques, libraries and frameworks employed in the implementation.

For the server components (DGPs and regional servers), for pragmatic reasons (performance, utilization of platform-specific functions, direct storage operations, etc.) C++ was chosen as the implementation language, with an operating system kernel wrapper layer as its basis that encapsulates operating system functions (e.g., sockets, threads) in a uniform interface so that all layers above it (except the final application itself) can be implemented independently of the operating system. The server components in the present version have been implemented using Microsoft Foundation Class Library (MFC) for Windows systems. Various theses have already impressively proven the quality of the architecture: By porting only this layer, the remaining framework components could be used for Apple iPhone and the iPad (see Lakew 2010 [15]).
On top of this OS kernel wrapper, we placed a basic framework layer with components for functions such as network communication (peers, general communication channels, proxies, etc.). Around this is a layer with components for data exchange, i.e., for the protocols used by the system. The code for this layer can be generated automatically using a specially developed protocol generator component. Above this Generated Client framework layer, the next layer contains the server framework functions required by both the DGP components and the regional server components (e.g., the graffiti manager for administrating information elements in databases; the components and algorithms for efficient geo-based searching in databases with R-trees, for managing users and forwarding location information to authorized parties). The top framework layer contains exclusively components to handle task-specific functions that affect only the DGP or the regional server.

The technical architecture of the client framework components is designed analogously. Digital Graffiti systems are to be available for implementation on as many as possible of the end devices today and in the future and thus on various platforms; therefore the decision to implement the client components in the programming language Java meant that the wrapper layer for encapsulating functions near the operating system level could be omitted. The Java Virtual Machine (JVM) guarantees the required platform-independence and is available on many of today's platforms.

However, the decision to use Java-based client framework components excludes the increasingly widespread Apple iPhone because no JVM is available for it. Nonetheless, the careful design of the system architecture, which ensures maximum flexibility, enables Digital Graffiti technology to be used on the iPhone: For conventional mobile phones with Symbian as their operating system, corresponding JVMs are available for which a Java application framework enables the development of graffiti clients. For other mobile phones that are equipped with no JVM or a JVM with only limited functionality, the system architecture provides for Digital Graffiti systems to be operated via a Web browser (here the new W3C standard) enables access to GPS data from a Web application: A Digital Graffiti proxy that can also be developed in Java on the basis of the underlying framework (although without the user interface) runs on a Web server and via Ajax it can dynami-
cally transfer data to a Web browser for display. Thus the framework architecture also supports devices that were not on the market when the architecture was designed.

To implement extensive, large-format Windows-based desktop clients, the technical architecture provides for the same framework kernel as for other client applications. However, to permit a modern user interface design based on technologies such as C# and Windows Presentation Foundation Classes (WPF), the framework design provides for compiling the Java framework to a J# assembly and using it in C# with corresponding adaptors. These aspects of the technical architecture help to ensure that Digital Graffiti clients can be implemented elegantly and simply for various end devices (see examples in Fig. 4).

Fig. 4. Digital Graffiti systems on various platforms (a) iOS, (b) Windows, (c) Android, (d) Symbian

4. Research prototype

To illustrate typical application systems that build on Digital Graffiti technology, in the following we present the Smart Information Campus System (SICS) of the University of Linz, which provides anyone associated with the university (teaching staff, students, administrative personnel) a new, mobile campus information, collaboration and navigation platform that is unique worldwide in its design and functionality. Acquisition and installation of the system is simple and device-specific via a specially installed website (dg.jku.at). No personal contact with a service location is required in order to install and employ the system. From a business informatics viewpoint, this is an important prerequisite for a new ICT system.

SICS offers the university community the possibility of publishing location-based information elements such as texts, images, sounds and videos or to procure these via (mobile) devices on location. Thus a mobile phone display can conveniently show where you are on the campus (Fig. 5a), where a certain seminar room or conference room is (e.g., MT 132/1, Fig. 5b and 5c), how far away it is (Fig. 5d) and which course is being held there (Fig. 5e).
SICS also enables leaving a information-element anywhere in public or private space (i.e., at any geo-position) in the form of a digital graffito. The publisher of the graffito must either be at that location physically or move virtually via a cursor to the desired position (geo-coordinates). The system provides a graffito editor that lets the publisher specify the recipient(s), the subject line, the content, the visibility radius, and the attachments (text, images, sound, videos, etc.).
SICS includes a Friendfinder component for social networking that permits a user to network with other users (friends, fellow students, professors, etc.) so that they can exchange and see their momentary locations (geo-positions). To network with someone, a user sends the respective person a networking request. Only if this person affirms the networking request, can both partners (called friends) immediately and mutually see their respective locations practically in real time. To protect privacy, the system permits location information only with mutual consent, which can be revoked at any. One’s own visibility can be disabled anytime either
generally or individually for a person, temporarily or permanently. The basic principle is that persons can see their respective locations only on a mutual basis. Fig. 6 illustrates details of the Friendfinder component.

5. Summary and outlook

The project described here is a new technology created to improve information-, collaboration- and navigation systems; although it pursues and achieves primarily business administration goals, likewise economic, social, socio-political and individual goals have been addressed. On the one hand, this technology enables a redesign of business processes and applies new potential for optimization of business processes (see Narzt 2012 [16][17], Graf et al. 2012 [18][19]), while on the other hand it facilitates valuable serves, e.g., catastrophe management, organization of emergency services, support for handicapped people, etc. and enables new dimensions of networking, communication and collaboration. This is a technology for implementing human/task/technology systems that enhance the possibilities for managing information as a production factor in business, industry, administration, and on a personal level.

The research methodology followed chiefly an engineering science approach in which above all prototyping and agile system development played a central role, but likewise social and applied economics methods were also employed, especially in the systematic evaluation of the usability of prototypes, the assessment of market potential and the planning of sales and exploitation models. Available knowledge was combined in new ways, innovative solutions were created, and new knowledge was generated.

The Smart Information Campus System, which represents one of the first large applications of the new technology, is now available as a platform for new research projects of all three colleges of the University of Linz. Hence, e.g., the university administration has suggested that the College of Law pursue the legal questions of ownership of public space in location-based services, that sociologists and psychologists explore the effects of such systems on communication and interaction behavior and their possible consequences, that economists develop cost/benefit models and evaluate the marketability, and that engineers, building on the now available base technology, design and implement new location- and context-based systems.

References

cation-Based Services”, Procedia Computer Science, Volume 5, 2011, pp. 697-704.