Pittsburgh Green Map: A Study in “Civic Computing”

Peter Lucas
MAYA Design Group, Inc.
2100 Wharton Street, Suite 702
Pittsburgh, PA 15203
412 488 2900
lucas@maya.com

Ron Gdovic
3 Rivers Connect
425 6th Avenue, Suite 1310
Pittsburgh, PA 15219
412 392 1006
rgdovic@3rc.org

ABSTRACT

Pittsburgh Green Map is an on-line interactive service for locating information concerning environmental, recreational, and other "green" assets in Western Pennsylvania. The service is innovative along two dimensions: Its user interface introduces a unique navigation/query paradigm that combines information-centric map displays, dynamic query, and zoom-driven aggregation. The result is a highly approachable interface optimized to emphasize "locality" in information display. Secondly, the system is based upon a distributed data entry/maintenance model designed to support a single information space maintained by hundreds of independent organizations and individuals, each in charge of a large or small amount of data and minimal by a central authority. The result serves as a prototype for a genuinely public "civic information space".

Keywords

Civic Computing, GIS, Information Centric Computing, Dynamic Query

BACKGROUND

The Green Map System is a framework for a worldwide network of locally-maintained maps of geographic sites of environmental significance (see http://www.greenmap.org). There are currently over 100 published or announced Green Maps in over 35 countries. These range from conventional paper-based maps to interactive, web-based Geographic Information Systems (GIS). Although a number of the on-line Green Map sites are quite powerful and comprehensive, nearly all of them are based on conventional GIS technologies involving centrally-administered databases and relatively inflexible "layer" or "query/response" based models of user interaction.

3 Rivers Connect (3RC, see http://www.3rc.org) is a non-profit organization located in Pittsburgh, PA and dedicated to the support of "civic computing," i.e., the application of information and communications technologies in support of civic activities at the local and regional level. More specifically, 3RC has set as its goal the creation in the Pittsburgh Region of a vast "Information Commons" comprising a single, public information space designed in such a way that geographic locality plays prominently in its operation. The Commons is conceived as a vast, single, distributed collection of publicly-available information of all sorts. This information will be indexed and displayed primarily geographically, using map-based, and other "place-centric" displays. It will be designed in the spirit of the town commons, where ordinary citizens not only access, but also routinely contribute information. As such, the goal is to dramatically raise the bar on the ease at which information can be contributed to the system by non-computer oriented citizens.

When work was begun on a Green Map project for the Pittsburgh/South Western Pennsylvania region, there was a strong consensus in the community to use this opportunity as a "first essay" in designing and building the Information Commons. Under the auspices of 3RC, funding was obtained from the R. K. Mellon Foundation and the Heinz Endowments for a development project under the supervision of the newly organized Greenpittsburgh.net. This group consists of over 45 member organizations, all of which have pledged to provide and maintain relevant information for the system. After a broad RFP process, MAYA Design Group was selected to design and develop the Pittsburgh Green Map interface.

DESIGN GOALS

The goals of the project were ambitious. The Pittsburgh Green Map is intended to represent a discontinuity in the evolution of GIS systems intended for casual use by ordinary people. Traditional GIS systems are hugely powerful in skilled hands but generally have a very steep learning curve for untrained individuals. Although there exist many web-based front-ends to these systems that are far more approachable, this approachability is typically purchased at the expense of power and flexibility in information access. If the goal of a true Information Commons is to be achieved, new styles of user interaction must be developed that are both very approachable and also have sufficient power to support navigation through a huge, highly-varied information space. The Pittsburgh Green Map was intended to be a first exploration in the development of such an interaction style.

Moreover, the goal of increasing the symmetry between information access and information contribution implied a basic rethinking of the GIS data-management architecture. Traditional GIS systems are organized in well-defined thematic "layers". For instance, there might be a layer for "roads", one for "parks", and so on. Each layer is typically stored in a separate homogenous database, which forms the basic unit of data administration. Although this works well for many traditional GIS applications, it has the effect of making the addition of a new "kind" of item a major project, one generally requiring the active support of a central administrator. The pragmatics of this process are incompatible with the goal of easy-even casual-contribution of highly diverse information by ordinary citizens. Thus the project came to entail a basic rethinking of the traditional GIS data model.

Within this broad context, a number of specific design goals were identified. These include the following:

Design a Map, not an "Application"

Although Pittsburgh Green Map was to be realized as a highly interactive Web-based applet, we wished to honor the basic spirit of the Green Map movement by preserving the character of the user experience as being one of interacting with a *map*, not a computer application. This goal was pursued in the context of the "information-centric" design philosophy developed at MAYA in the context of the Visage information-visualization system [1, 2, 3]. The information-centric approach strives to present all information to users as visualizations comprising organized arrangements of well-defined information objects, designed to be interacted with via direct-manipulation and/or dynamic query techniques. The approach eschews dialog boxes and other indirect UI techniques in favor of "letting users get their hands on the data." The goal is to craft a user experience of dealing directly with collections of data, rather than accessing information indirectly through an "application." In the case of Pittsburgh Green Map, we wanted to end up with a "magical map," not an application for generating maps.

One Display, Many Data Sets

Although our goal was to support a large number of highly diverse data types, we wished to minimize or eliminate any fundamental distinctions between them at retrieval time. That is, we wished to have a single map on which any subset of the available data objects could commingle, regardless of type, depending only on user need. Using dynamic query techniques developed in the domain of information visualization [4] we wished to give the user direct control of what appears on the map at any given time. In effect, we wished to break down the "layers" that tend to isolate heterogeneous data in traditional GIS systems.

Clutter Control

Although the use of map-based displays is an obvious choice if one is interested in emphasizing geographic locality, issues of clutter threaten the scalability of this approach. For example, displaying the location of parks and other green spaces as icons on a map might work very well at the scale of a neighborhood, but it would become unwieldy at the scale of a large city. At the scale of a region, or the entire country it would be essentially useless. It was a goal of the project to preserve the value of map based retrieval techniques even in the presence of great item density.

Scalability

Because Pittsburgh Green Map was conceived as a "rapid prototype" of the Information Commons, significant effort was applied to establishing the architectural underpinnings of a future system that could scale arbitrarily. This scalability requirement can be analyzed along three dimensions. First and most obviously, it must scale to arbitrarily large number of items, both on the "back end" and at the user interface (the "clutter control" issue is an example of a concern for UI scalability). Second, there was a requirement for *category* scalability. That is, although the present project is focussed on "green" topics, it was an explicit goal that nothing in the basic design was to compromise the ability to broaden the coverage to include content of great diversity. Third, the architecture needed to scale administratively. The Commons is envisioned as a nearly completely distributed system, where only narrow "syntactic" standards will be centrally administered. Although a single, centralized server was deemed acceptable for the immediate project, care was taken to avoid making this a deep assumption of the architecture.

Aesthetic Quality

One of the great appeals of good paper maps is that they tend to be *nicer* than their computer-screen counterparts. There is a definite aesthetic to mapmaking that contributes mightily to the "user experience" when interacting with these artifacts. In choosing an on-line format for the Pittsburgh Green Map, we were determined to compromise this aspect of the user experience as little as possible. Consequently, much attention was paid to setting the technical foundations for the generation of map displays that delight as well as inform.

Distributed Data Maintenance

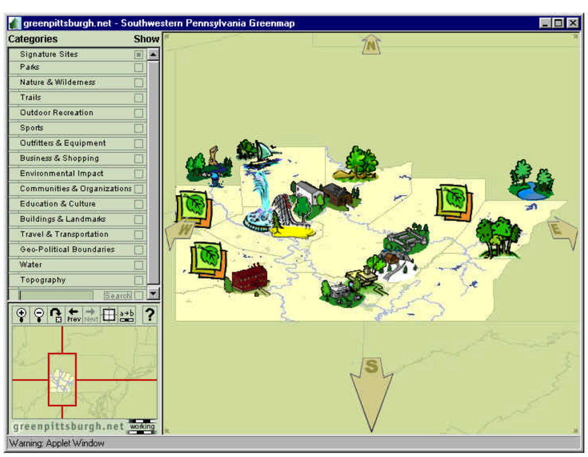
One of the basic innovations of Pittsburgh Green Map is its model for distributed data maintenance by multiple, autonomous content partners. Although the *display* of information was to be seamless, the task of *maintaining* content was to be easily partitionable across a potentially very large number of "owners" of various content. Thus, for example, the City of Pittsburgh might be responsible for entering and maintaining data concerning all city parks, whereas the State of Pennsylvania might maintain information about State Parks. Some providers might maintain large collections of items, while others (such as a stand-alone museum) might maintain only one. The requirement had deep architectural implications for the underlying database technology used in the system.

Separate Data from Display

Finally, it was considered essential to maintain a very crisp line between the underlying data objects and the user interfaces through which they were viewed and manipulated. This requirement stemmed from our vision of the Information Commons as an abstract data space that will be marshaled for many purposes. Thus, for example, although the Green Map data is entered with the goal of supporting an on-line map applet, it should also be easy to use the same data to support a virtual-reality tour of the region, an interactive educational application, or a planning tool for government plan agencies.

THE CLIENT INTERFACE

The Green Map client is implemented as a Java applet running under a standard browser. No plugins or other client-side installation is necessary. In keeping with the spirit of our "map, not application" design philosophy, approximately 75% of the total screen space of the applet window is by default allocated to a dynamic map, with the rest divided between two distinct control areas. The basic design of the interface is extremely simple: Always show the user a map containing a reasonable number of items likely to be of interest. Items are represented as hand-rendered icons, positioned on the map according to the latitude and longitude of the corresponding items. Thus, for example, in Figure 1, which represents the initial map view at the start of a session, Pittsburgh's Point State Park is represented as a rendering of its signature fountain, and Frank Lloyd Wright's Fallingwater can be seen in caricature to the southeast of the city.



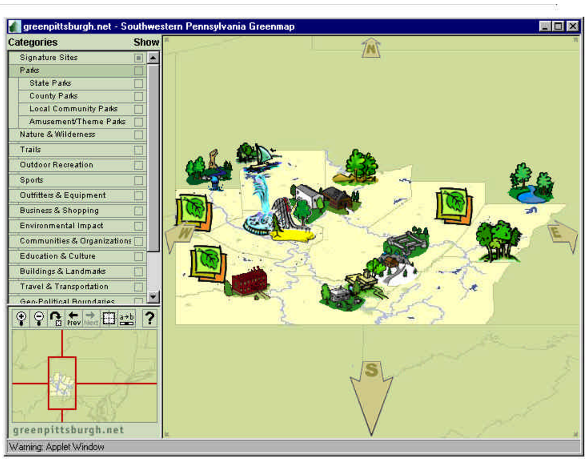
Clicking on one of these items pops up a "details" page in a separate window, as shown in Figure 2. This, in turn, may have clickable links to other items, or to external web pages.

In essence, the Green Map applet implements a user interface for navigating through an arbitrarily large data space of discrete items organized primarily geographically. In designing this interface we relied on an interaction model that was first developed in the context of the Visage information visualization environment [3]. This model, known as "Scope/Focus/Detail" defines three distinct dimensions of user navigation and control. "Scope" refers to the range of items ("how much of the world") is currently of interest. "Focus" answers the question of what the user wants to know about the current scope. "Detail" controls how fine-grained the data display--do we want a concise overview, or all the gory details? The Green Map interface provides to the user clear mechanisms for each of these dimensions.

Query Interface

The map view in Figure 1 is of the entire 11 county region covered by Greenpittsburgh.net. There are over 30 thousand data items in the system that are potentially visible on this map. How, then, did the system choose the 20 or so items that are in fact visible? The answer lies in the list of categories seen at the upper left of the applet window. This list is essentially a categorical dynamic query tool that controls the visibility of items based on their membership in various collections of items. This mechanism will be described from an information architecture perspective below. From the user's perspective the model is simply that all members of each selected category are displayed on the current map view. Initially, only the category "Signature Sites" is selected, so only the (few) items in that category are visible. The user may add or remove items at will simply by checking in the appropriate boxes. Note that the "query" thus described is scoped to the currently visible map view. Since, in Figure 1 the entire 11 county region is displayed, it is guaranteed that all items in the selected categories will be represented (although, as we will see below, not necessarily as a discrete icon). If, however, we were "zoomed" into a smaller region, it is possible that items in the selected categories may be "off the map". In this way, the "scope" and the "focus" of the current map display can be independently controlled--the former using the map pan/zoom controls (described below) and the latter via the category dynamic query control.

As mentioned, the category control is hierarchical. By clicking on the name of a category, the user can "drill down" in to subcategories. Thus, as shown in Figure 3, clicking on "Parks" reveals the subcategories of "National", "State" and "Local" parks, each of which can be independently selected or deselected. Selecting the parent category automatically forces the selection of its subcategories. If only some subcategories of a category are selected, the parent shows as "partially selected". In this case, the user can toggle among three states (unselected, selected, partially selected). The partial selection set is "remembered" across such toggles.



The applet also supports a text search feature on item names and keywords. Search is presented to the user in the form of a special "search" category, which can be seen at the bottom of the category list in Figure 1. The user simply enters a text string and executes the search. The result of each search is presented to the user as a dynamically generated category (actually a subcategory of the Search "category"). It then behaves exactly like the pre-defined categories, and can be selected or deselected at will. Search result categories persist for the duration of the user's session.

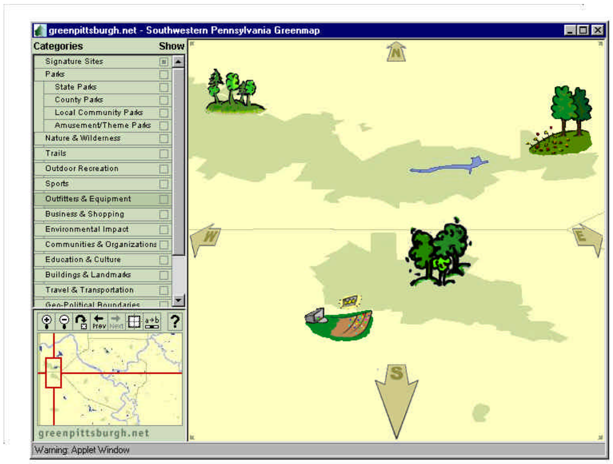
Map Controls

The lower left portion of the applet window is dedicated to miscellaneous controls, primarily having to do with map navigation. Most important are the map pan and zoom controls. Panning can be accomplished in two ways. A single click on the map (other than on an item icon) will center that point on the map display. In addition, the control area contains a thumbnail "context map" with crosshairs and a box showing the current view. This box is draggable within the context map, thus supporting "long distance panning." Zooming in can also be done in two ways. Conventional "In/out" buttons are available in the control area. In addition, dragging out an area in the main map (similar to a selection box in a conventional WIMP interface) causes a zoom in to the selected region. Finally, a "home" button returns the map to its fully-zoomed-out 11 county view.

Several miscellaneous controls should be mentioned. A "ruler" is a modal control that permits the user to drag out a line on the map that serves as a kind of ruler, measuring the approximate distance between any two visible points. This control works properly at all scales, from feet to miles. "Previous" and "Next" buttons permit the user to navigate through a history of previous states of the interface (much as in a web browser), thus affording the ability to experiment with the interface without danger. Finally, there is a "Projection" control that toggles between two map modes. By default, the map is rendered as an oblique projection, with the effect that vertical and horizontal distances are not in the same scale. This mode also selects icons drawn in perspective, intended to introduce a sense of depth and interest in the display. Clicking the "Projection" button toggles the display between this mode and a conventional "top down" orthogonal map projection. This mode also replaces the perspective icons with standard map symbols as specified by the international Green Map organization.

Automatic Aggregation

We have already described how the interface provides for control of "scope" and "focus"--two of the three dimensions of our navigation model. Control of "detail" is addressed in two ways. The first is via the trivial mechanism by which the user clicks on icons to obtain detail pages. The second--which addresses the above-mentioned clutter-control problem--is much more subtle and interesting. The number of items that may appear on the map at any given time is governed by a sophisticated algorithm for the automatic aggregation of items that would otherwise be densely packed in screen space. The details of this algorithm, which significantly extend previous work in this area [5, 6], are beyond the scope of this paper. However, the general idea is that whenever the interaction between the category selection and map zoom level yields item sets that would overlap excessively, the offending items are not displayed, but rather represented by a special "aggregate" icon seen as a small "pile. " Clicking on the aggregate icon causes an automatic "zoom-in" operation to a scale at which the individual items become directly visible (see Figure 4). In effect, this approach binds the control of "detail" to "zoom", since "zooming-out" generally tends to produce a higher screen density of items, which in turn will tend to be automatically aggregated. The decluttering algorithm has a number of parameters (e.g., permissible degree of item overlap) that could in principle be exposed to users. However, for our target user community it was adjudged that the tight coupling of zoom and detail control represents a good balance between power and simplicity.



In summary, we believe that the Green Map applet represents a simple, elegant, approachable interface. Its parsimony belies a powerful set of capabilities for producing surprisingly sophisticated and useful information displays of a large, complex underlying data set. We believe that we have succeeded in our goal of introducing a sense of "place" into the Internet in a powerful and extensible way.

INFORMATION ARCHITECTURE AND DATA MANAGEMENT

At least as important to our long term goals as the specific user interface described above is the unique information architecture upon which the system is based. In the long run, we believe that the most significant aspects of this work will prove to be (a) the crisp separation of well-defined "data objects" from their rendering on a particular device in a particular way and (b) the use of this architecture to implement a radically distributed approach to data ownership and maintenance. Both of these issues are discussed in this section.

"Object-oriented Data"

If we are to support the emergence of truly open public data spaces, a crucial issue that must be addressed in the need to maintain the tractability of this space as it evolves. This problem is highly analogous to the problem of maintaining tractability in large computer programs in which the need for coordinating the work of many programmers has led to sophisticated methodologies for structuring and managing the "code space." Moreover, there is every reason to believe that the same basic techniques will apply, namely the need to maintain very well defined, hierarchically structured "objects" that interact only in rigorously controlled ways.

This is an obvious conclusion. However, most applications of it have taken the form of fairly literal attempts to directly apply OO coding models to problems of data representation. In contrast, our work starts with the assumption that the resulting blurring of the distinction between code and declarative data is fundamentally mistaken. On the contrary, we believe that a crisp separation between "information objects" and the "code objects" that operate on them is essential to the emergence of a true Cyberspace. The fewer assumptions that are made about the devices that will access our data objects, the more liquid (i.e., free-flowing) those data objects will be. Thus, for example, a web page described in XML is far less bound to the specifics of any given browser or device than the same page coded in HTML.

Although XML is an important step in the right direction, it suffers from the flaw of not being very "object oriented". Truly liquid data objects will require a simpler, sharper notion of identity and boundary than is enforced by XML in its current state. In the context of our DARPA-funded Visage project, we introduced the concept of the "u-form" to address this issue. A u-form is simply a bundle of attribute-value pairs identified by a universally unique identifier (UUID). U-forms are intended as a low-level, standardized data container upon which all higher-level semantics can be layered. In this regard they play a role similar to that of the byte: bytes do not have types, they come on only one flavor, and yet they suffice as the common basis for virtually all of computing. They are the glue that literally holds together all modern computing architecture--hardware and software.

In Green Map, as in Visage, literally all data are represented as u-forms. Each map item has behind it a single u-form. Each category is a u-form representing a collection of items. Each map feature and icon is stored in its own u-form. All relations among entities within the system are represented as values of attributes of u-forms that contain the UUIDs of other u-forms. In this way, hierarchical and other high-level structures are easily created and modified. For example, the way the applet renders the category selection list is by fetching a well-known u-form that contains an attribute MEMBERS (our convention for all "container" u-forms) whose value is a vector of UUIDs of the top-level categories. In turn, these category u-forms have a MEMBERS attribute that may point either to sub-categories or directly to map items. This structure may continue to any depth. In this way the category structure, and hence the "ontology" of the data space is completely data-driven. Categories can be created or rearranged via simple edits to the appropriate u-forms.

Since an item may belong to any number of collections, the same mechanism can be easily deployed to create other characterizations of data. For example, each registered content provider for Greenpittsburgh.net "owns" one or more collections of u-forms that they are responsible for maintaining. This mechanism is described in the following section. As can be seen, the workflow and data management issues involved in the system's unique distributed maintenance model reduces to extremely simple operations on collections of u-forms. Moreover, although the current implementation relies on centralized map and data servers, the rigorous use of UUIDs sets the foundation for future implementations involving data replication and a distributed server architecture.

System Architecture

The system is configured as a client/server architecture, with the applet serving as client to two distinct servers. The first server is a standard Visage Repository, which is a minimalist database engine that directly implements the u-form abstraction. The basic interface to a Repository consists of transactions involving a UUID, and attribute name and either a "set" or a "get" operation. Aside from a few simple metaoperations concerning authentication, etc, virtually all interactions with the server reduces to these operations.

The second server is a map server. Based loosely on BBN's OpenMap™ suite (see http://openmap.bbn.com), this server is responsible for creating on-demand background maps, which are used by the client as "wallpaper" behind the items. Essentially, everything that the user can directly manipulate comes from the Visage Repository, while static, non-manipulable background data come from the map server.

Note, however, that the map server itself is a client of the Visage Repository, since all of the feature information for the background maps, such as roads, rivers, political boundaries, etc, is itself stored in u-forms. In this way, the distinction between what is "background" and what is "item" is flexible and readily under the control of the interface designer.

Distributed Data Management

The distributed data management model is implemented as a separate server written in Python that serves up Web-based forms for the creation and editing of collections of u-forms. Each content provider can view the collections that they own, and edit, delete, or add to their contents. Provisions are made for automatically geocoding items using a simple point-and-click procedure on the map.

As a matter of policy, all changes are currently deferred until an editor has a chance to review and approve the changes. This is accomplished by making copies (with different UUIDs) of each modified u-form, entering pointers to them in a special "to be approved" collection, and notifying the editor. When the editor approves the changes, the *contents* of the edited u-forms are moved to the original u-forms, which immediately makes the changes "live." This example is helpful in understanding the "container" nature of u-forms.

Conclusions

Although in itself an interesting but bounded application, Greenpittsburgh.net represents a first exploration of an approach to "civic computing" whose full-blown implementation we believe will have major implications. A true Information Commons would represent the first large-scale "public spaces" in Cyberspace. The introduction of geographic locality to the mix of agendas that are the Internet is little explored and potentially sweeping in its implications. We hope that this modest essay will prove provocative in moving this agenda forward.

ACKNOWLEDGMENTS

This project represents a unique collaboration among community organizations, environmentalists, funders, and designers. Special thanks to George Calaba, 3 Rivers Connect; Court Gould, and Evan Verbanic, Sustainable Pittsburgh; the R. K. Mellon Foundation; the Heinz Endowments; Jeremiah Blatz, Joe Hughes, Anukul Kapoor, Bill Lucas, Heather McQuaid, Justin Petro, Stuart Roth, Jeff Senn, Skip Shelly, Rich Tribone and Leslie Wheeler, MAYA Design Group, Inc.

REFERENCES

1. Burks, Michale B., Dunmire, Carolyn, Gomberg, Christina C., Kolojejchick, John A., Lucas, Peter, Roth, Steven F., Senn, Jeffrey A., & Stroffolino, Phillip J. Visage: A User Interface Environment for Exploring Information. *Proceedings of Information Visualization*, IEEE, San Francisco, pp. 3-12.
2. Kolojejchick, John A., Roth, Steven F., & Lucas, Peter. Information Appliances and Tools in Visage. *IEEE Computer Graphics and Applications*, Volume 17, Number 4, July/August, 32-41.
3. Gomberg, Christina C., Lucas, Peter & Roth, Steven F., Visage: Dynamic Information Exploration. *Conference Companion of the Conference on Human Factors in Computing Systems* (SIG CHI '96), Vancouver, BC, Canada.
4. Ahlberg, C., & Shneiderman, B. (1994). Visual information seeking: tight coupling of dynamic query filters with starfield displays. *Proceedings of the CHI '94 Conference on Human Factors in Computing Systems*, 313-317. New York: ACM.
5. Rayson, J. Aggregate Towers: Scale Sensitive Visualization and Decluttering of Geospatial Data, *Proceedings of Information Visualization '99*.
6. SPAWAR Systems Center San Diego, Pop-up Declutter Tool, http://hsi.nosc.mil/java/PUDT/PUDT.html