SmartCard Prototype

Neal Altman, Michael K. Martin,
Dawn Robertson and Kathleen M. Carley
September 2009
CMU-ISR-09-122

Institute for Software Research
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

CASOS
Center for the Computational Analysis of Social and Organizational Systems
CASOS technical report.

Abstract

The SmartCard is as a software tool intended to allow users to conveniently access the results of the 300 Cites virtual experiment using a graphical interface. This report discusses the evolution of the SmartCard concept through a series of experimental prototypes, the initiation of a releasable version, the design and execution of the SmartCard to date and future development plans.

Table of Contents

List of Figures

1   Introduction

The SmartCard is envisioned as a tool for users to conveniently access the results of the 300 Cites virtual experiment. In that study, the effectiveness of various strategies for modifying taxpaying behavior will be examined by means of simulation, using demographically varied population sets representative of specific urban areas. The study will produce a variety of metrics that describe the effect of IRS services on taxpayer compliance in each of the cities examined. See [1] for a more detailed description of the 300 cities study.

The SmartCard application is conceived as a means for delivering the results of the 300 cities virtual experiment to IRS decision-makers. As a decision-support tool, the SmartCard is intended to:

- Allow non-experts to access 300 Cities data in a convenient, task-relevant format.

- Serve as a model and prototype for presenting simulation results.

To achieve these goals, the SmartCard and in particular the SmartCard prototypes were designed with the following principles in mind:

- **Immediate Usability** – the SmartCard should be immediately usable by the intended users. Thus a person familiar with the subject area or 300 cities data in particular should be able to immediately use the SmartCard to access information and comprehend it. This implies that the interface provide simple controls that benefit by analogy to known computer use models. (However, the SmartCard is not intended to teach novice a user about the problem space, by the same logic that a calculator does not have to explain the nature and purpose of logarithms in order to be a useful tool for their calculation.)

- **Utility** – The SmartCard should provide information in ways that the user finds useful to their needs. For example:

    o Common queries should be answered clearly.

    o User should be able to control what data they view and be able to combine or compare the viewed data.

    o Users should be able to employ other tools in concert with the SmartCard. This includes export of raw data to other tools and inclusion of data views created in the SmartCard in other media.

- **Customization** - the SmartCard is oriented to a specific body of data. The utility of the SmartCard in large part derives from the fact that the user does not have to spend time establishing a scaffolding to make use of the data. But the user should not be straight jacketed by the choices made by the software designers. Instead, users should have the option of modifying views, creating their own views and extracting data and views of the data from the interface for other uses.

- **Portability** – the SmartCard should be widely deployable, without requiring special hardware, support software or network connection. This does not preclude the provision of a net-enabled version, but for security and simplicity the interface should not make large demands on the system or be difficult to install.

- **Extensibility** – the SmartCard should be able to incorporate new data and new ways of exploring data. This implies that it should be simple to extend the SmartCard tool programmatically.

These goals are interdependent and can be in opposition. For instance, immediate utility might be supported, in part, by decreasing the number of interface controls, limiting options and providing step-by-step prompting. On the other hand, to support customization, the user needs to be able to modify the selection and display of data, implying provision of additional controls and non-linear access to the data. Successful implementation of the SmartCard will require striking a proper balance between ease of use and flexibility.

This report describes progress to date towards achieving a working and potentially useful SmartCard. As work continues towards understanding and simulating taxpayer behavior the SmartCard will continue to evolve to incorporate the latest algorithms and outcome data.
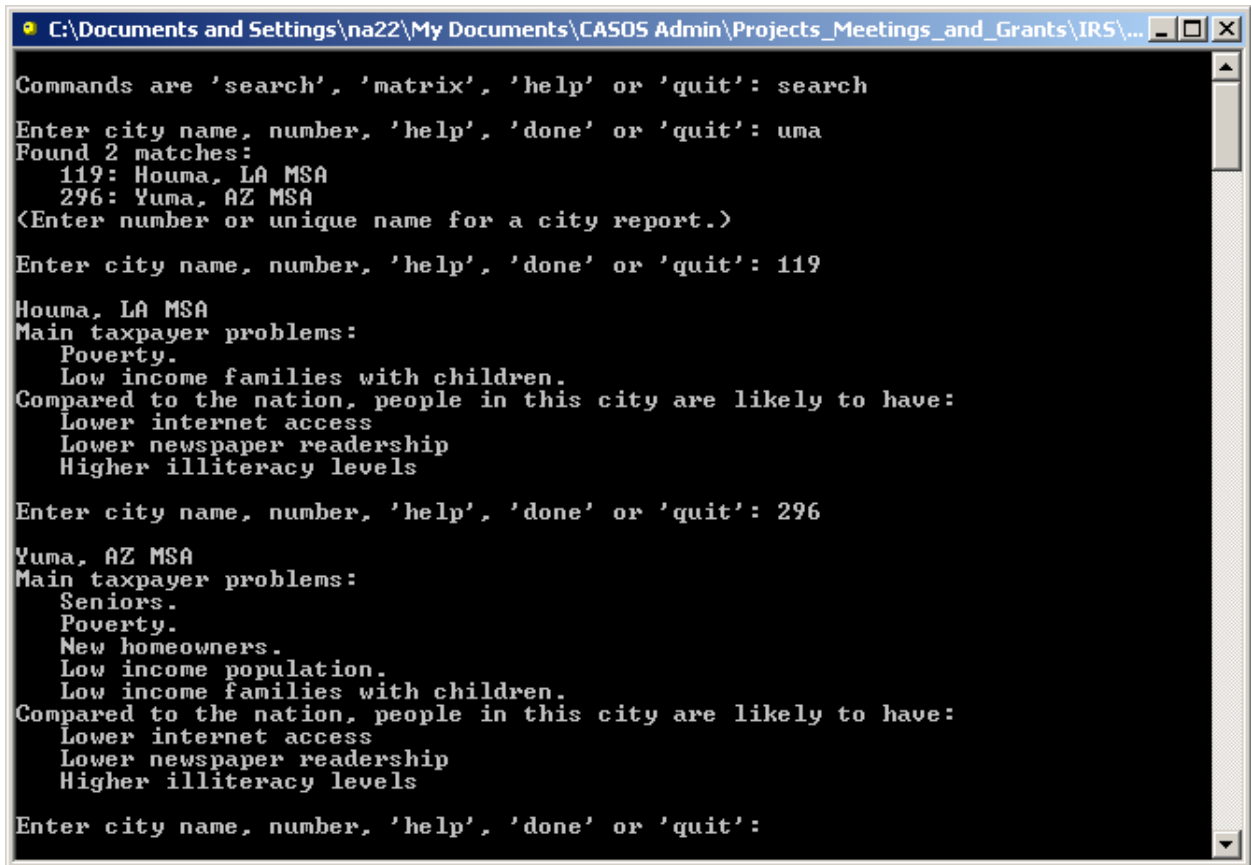
2 Exploratory Prototypes

A series of exploratory SmartCard prototypes have been developed as proofs of concept. While the prototypes were developed in sequence and worked towards common goals, they did not necessarily attempt to reuse underlying code. Instead useful concepts were carried forward into successive prototypes.

2.1 Initial Proofs of Concept

The initial demonstration versions of the SmartCard featured a simple text interface written in Perl. This approach was primarily intended to expedite development, but also allowed focus on certain key features.

2.1.1 Pure Text Version


```
C:\Documents and Settings\na22\My Documents\CASOS Admin\Projects_Meetings_and_Grants\IRS\...

Commands are 'search', 'matrix', 'help' or 'quit': search

Enter city name, number, 'help', 'done' or 'quit': uma
Found 2 matches:
    119: Houma, LA MSA
    296: Yuma, AZ MSA
<Enter number or unique name for a city report.>

Enter city name, number, 'help', 'done' or 'quit': 119

Houma, LA MSA
Main taxpayer problems:
    Poverty.
    Low income families with children.
Compared to the nation, people in this city are likely to have:
    Lower internet access
    Lower newspaper readership
    Higher illiteracy levels

Enter city name, number, 'help', 'done' or 'quit': 296

Yuma, AZ MSA
Main taxpayer problems:
    Seniors.
    Poverty.
    New homeowners.
    Low income population.
    Low income families with children.
Compared to the nation, people in this city are likely to have:
    Lower internet access
    Lower newspaper readership
    Higher illiteracy levels

Enter city name, number, 'help', 'done' or 'quit':
```

Figure 1 Pure text version.

This initial version of the SmartCard emphasized:

- Search.

- Comparison:

  o Single cities to national averages, in the form of a taxpayer report.

  o Cross-city comparison, with multiple city reports output and held simultaneously in the text window.

In addition, raw data, in the form of a distance matrix, could be output for use by other tools. The interface included a help command as a nod to usability, but was modal – for instance the command set changed when searching – which tends to reduce usability.

2.1.2 Graphical Text

These versions added a graphical wrapper to give the user a somewhat more intuitive interface. The graphical component was implemented by combining Perl/Tk with the original Perl code.

2.1.2.1 National Comparison
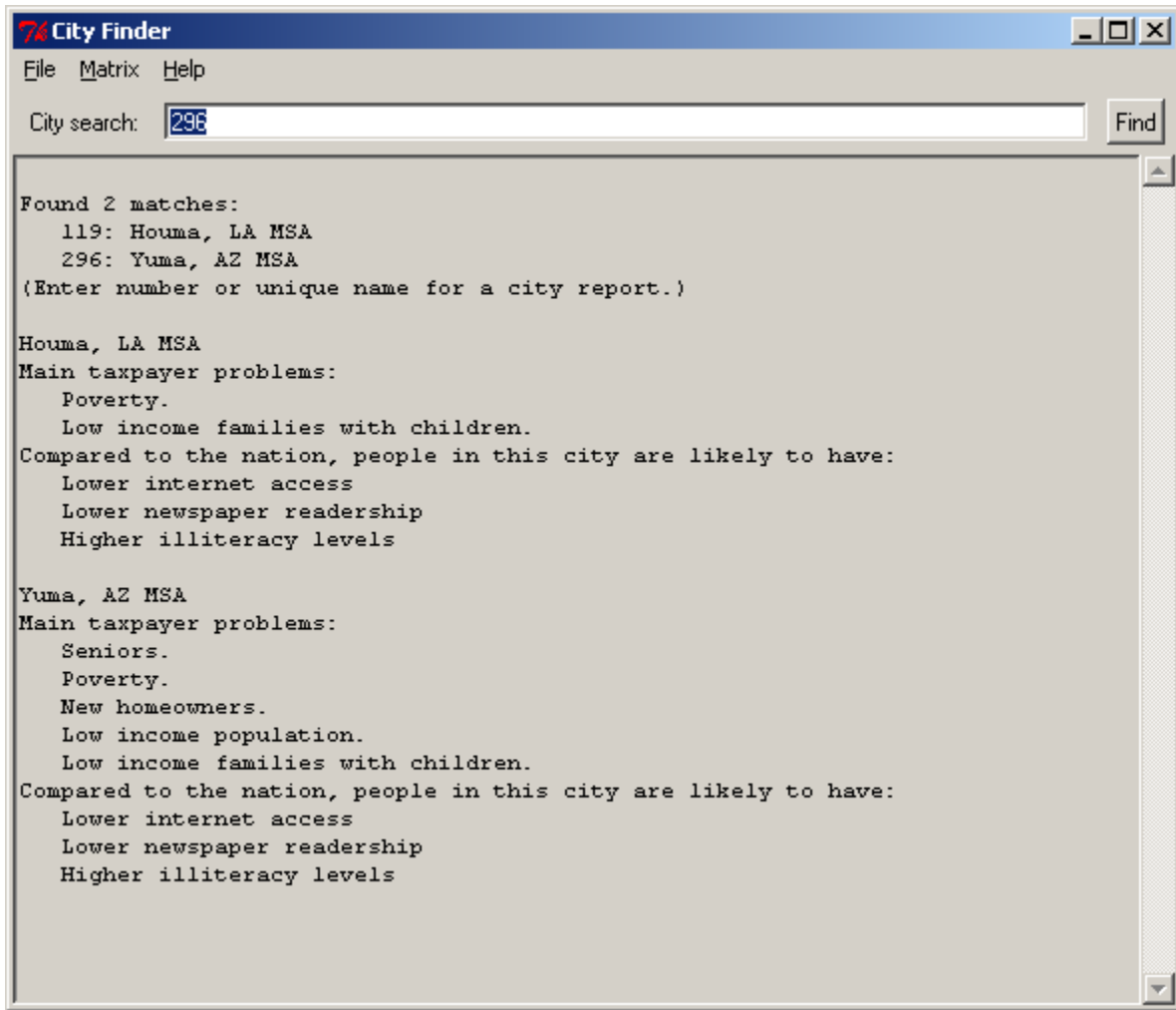
Figure 2 Initial graphical text version.

The initial version of the graphical interface simply encapsulated the functionality of the previous text version (national comparison), while enhancing usability. Benefits of this approach included enabling command exploration for users, where any command can be "discovered" by working with the visible commands and icons. Users also benefited from the use of familiar idioms (e.g., menu bar, pull down menus, buttons) and by avoiding modal operation, where the active command set varies based on a prior command.

2.1.2.2 Multiple Comparison

```
╔═══════════════════════════════════════════════════════════════════╗
║ 7k City Finder                                          _ □ ×      ║
╟───────────────────────────────────────────────────────────────────╢
║ File   Matrix   Compare   Help                                    ║
║                                                                   ║
║ City search:  [296                                    ]    Find   ║
╟───────────────────────────────────────────────────────────────────╢
║ Found 2 matches:                                                ▲ ║
║    119: Houma, LA MSA                                             ║
║    296: Yuma, AZ MSA                                              ║
║ (Enter number or unique name for a city report.)                 ║
║                                                                   ║
║ ==========================                                       ║
║                                                                   ║
║ Houma, LA MSA                                                     ║
║                                                                   ║
║ --------------------------:-----------:-----------:-----------:   ║
║ Tax item usage:          :           :    Region :      Size :   ║
║             COMPARED TO : Nation :   South :  Group 01 :          ║
║ --------------------------:-----------:-----------:-----------:   ║
║               Tip income :    Lower :    Lower :     Lower :      ║
║    Self employment income :  Average :  Average :   Average :     ║
║           Use of the EIC :   Higher :   Higher :    Higher :      ║
║          Student loan use :   Lower :    Lower :     Lower :      ║
║               Farm income :  Average :  Average :   Average :     ║
║   Social Security income :  Average :  Average :   Average :      ║
║           Capital gains :   Average :    Lower :   Average :      ║
║ --------------------------:-----------:-----------:-----------:   ║
║                                                                   ║
║ Information access, compared to the nation:                      ║
║    Lower internet access                                         ║
║    Lower newspaper readership                                    ║
║    Higher illiteracy levels                                      ║
║                                                                   ║
║ ==========================                                       ║
║                                                                   ║
║ Yuma, AZ MSA                                                      ║
║                                                                   ║
║ --------------------------:-----------:-----------:-----------:   ║
║                                                                 ▼ ║
╚═══════════════════════════════════════════════════════════════════╝
```
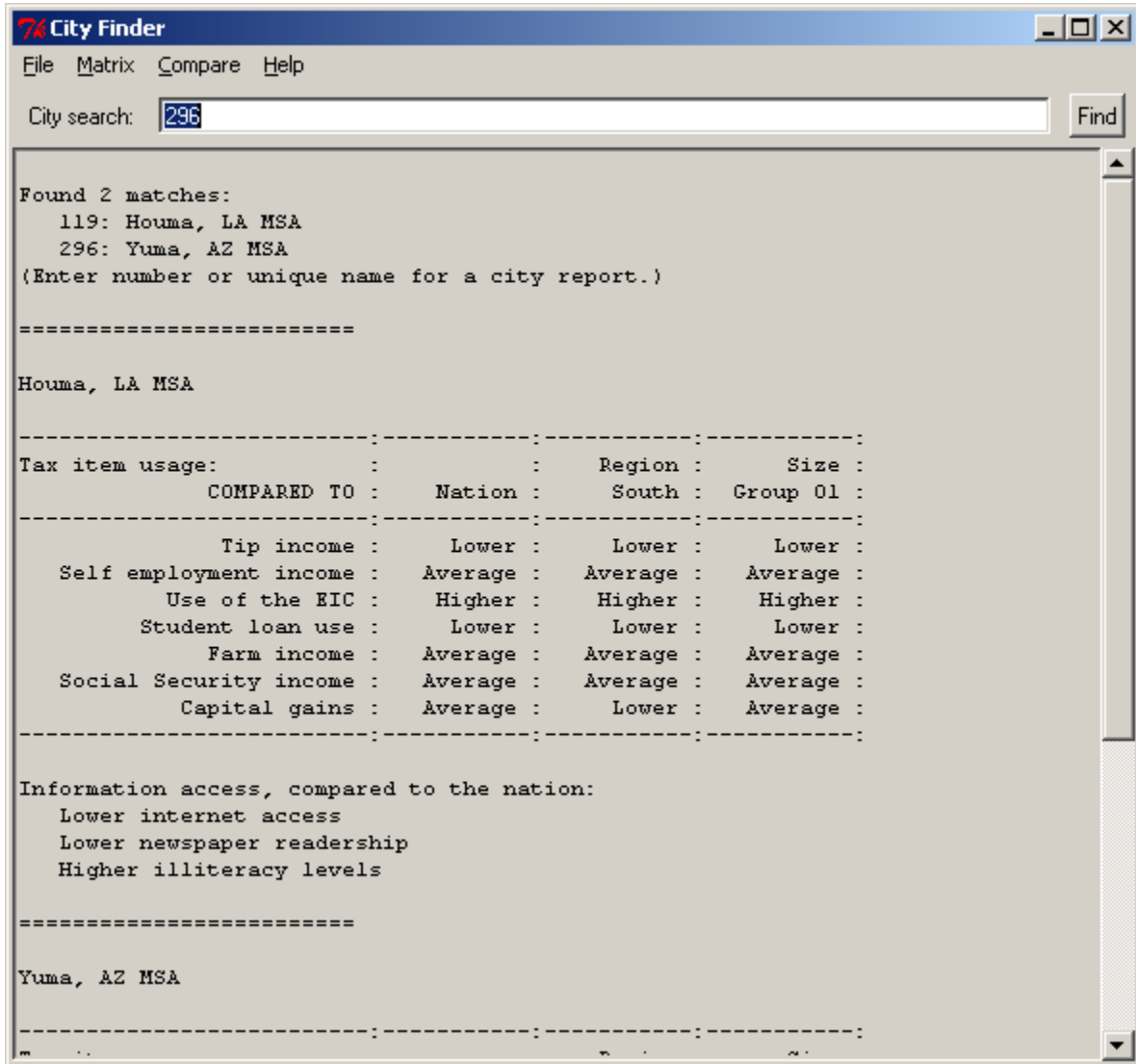
Figure 3 Multiple comparison graphical text version.

In this prototype the Perl/Tk interface was then rewritten to support a fresh set of city data and to allow simultaneous comparison of a city to a number of different reference groups. As can be seen in Figure 3, this prototype allowed users to compare cities to other cities in the region and nation. The interface was further modified to allowed user customization of the output. For example, it allowed users to specify cities and/or taxpayer issues to be compared as shown in Figure 4. Yet the amount of effort required of users to implement changes to the output was relatively high and the output remained plain text.
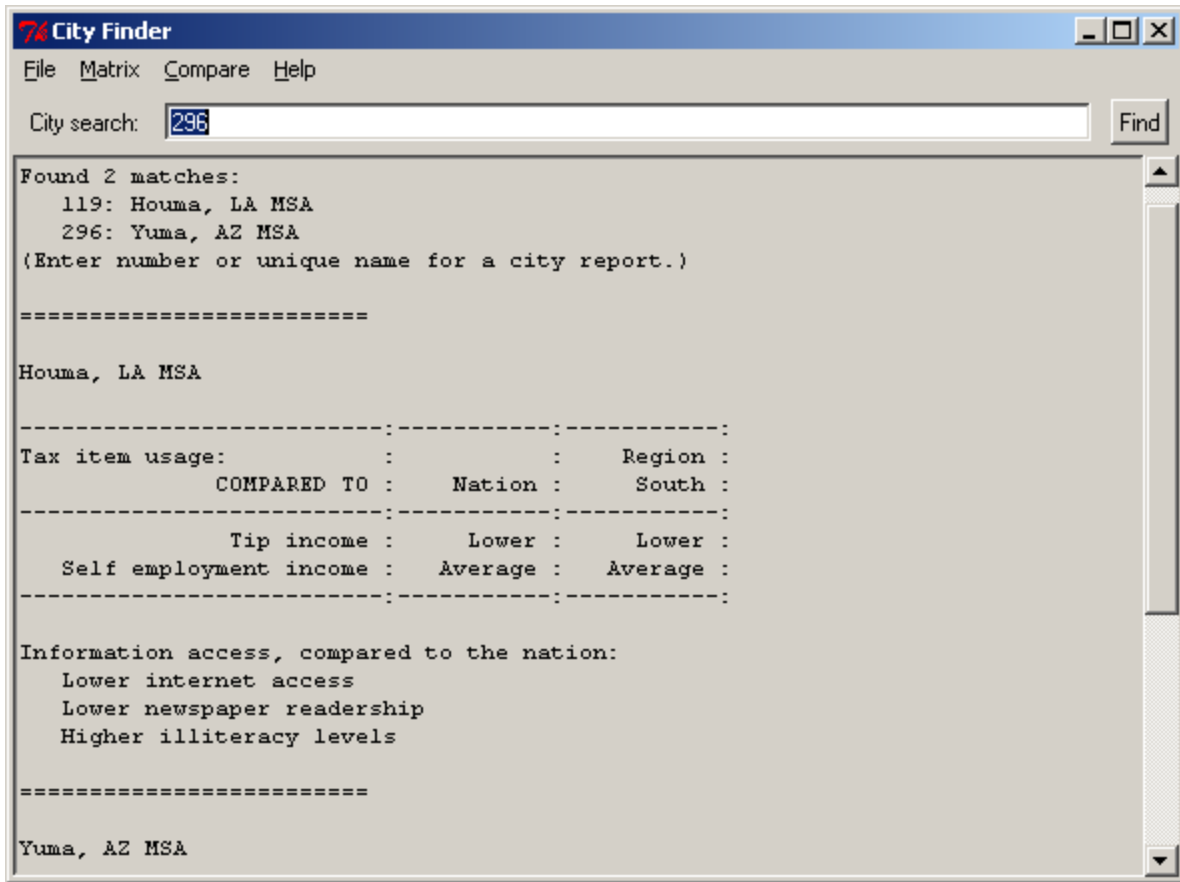
```
City Finder                                                    _ □ ×

File  Matrix  Compare  Help

City search:  296                                          Find

Found 2 matches:                                              ▲
   119: Houma, LA MSA
   296: Yuma, AZ MSA
(Enter number or unique name for a city report.)

==========================

Houma, LA MSA


---------------------------:-----------:-----------:
Tax item usage:           :           :   Region :
            COMPARED TO :    Nation :    South :
---------------------------:-----------:-----------:
             Tip income :    Lower :     Lower :
   Self employment income :  Average :   Average :
---------------------------:-----------:-----------:


Information access, compared to the nation:
   Lower internet access
   Lower newspaper readership
   Higher illiteracy levels

==========================

Yuma, AZ MSA                                                  ▼
```

Figure 4 Multiple comparison graphical text version, with modified comparison.

2.1   Database-Enabled Proof of Concept

The database enabled prototype, which encapsulated the same data set used in the multiple comparison Perl/Tk interface, was created using a database oriented development environment (Microsoft Access, with Visual Basic). The major benefits of this approach included:

- Integral database support, facilitating search and retrieval of data and integration of new data.

- Data display support (e.g. charts and graphs), simplifying data presentation.

Development of this prototype beyond proof of concept, however, would have been problematic, because it requires use of a licensed and proprietary platform.
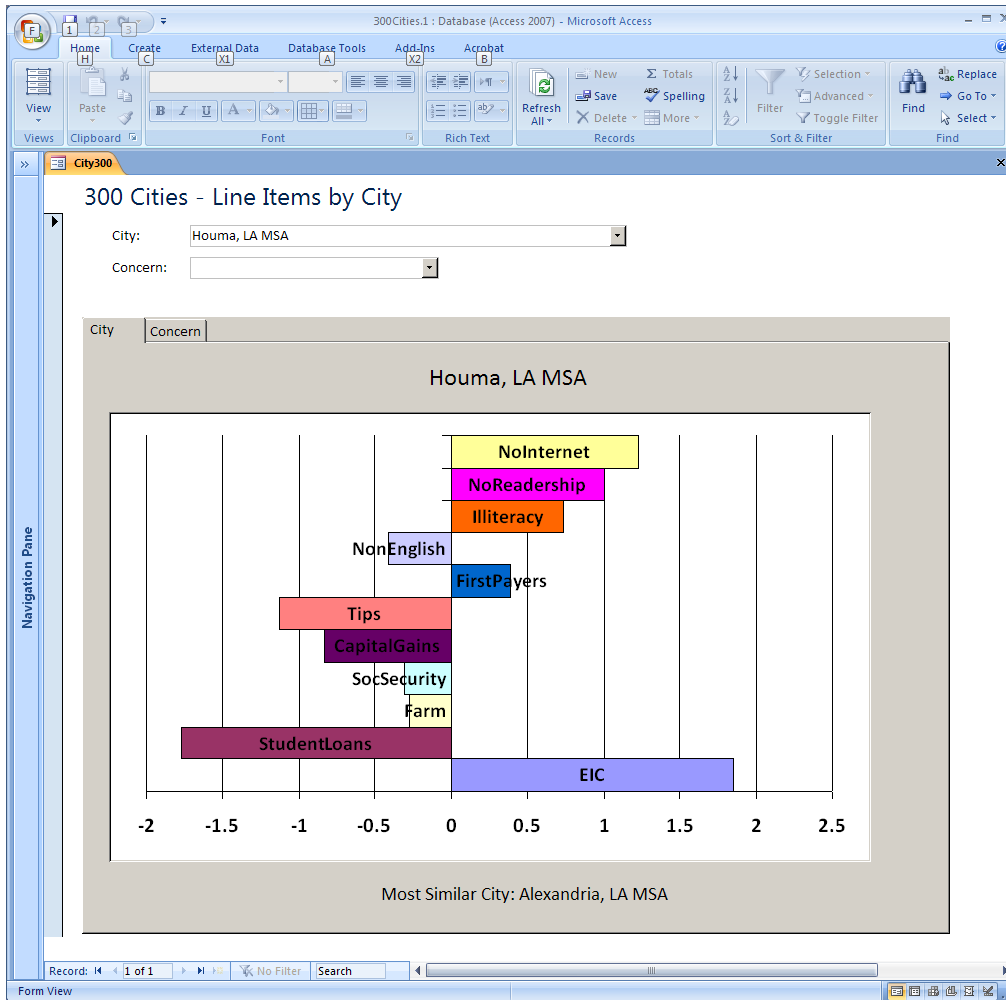
6

Figure 5 Database enabled prototype.

3   Developmental Prototype

Using the concepts from the exploratory prototypes, we have initiated development of a more general and deployable version of the SmartCard. While this version employs the database approach, it is designed to avoid commitment to proprietary software platforms. In addition, wherever practical, the deployable SmartCard will use language, libraries and development environments that are consistent with common CASOS practice.

Development of the prototype is proceeding in an incremental fashion, where a series of prototype versions are rapidly developed, evaluated and then used to define the next incremental development step. Development is occurring in parallel with the statistical and simulation studies, in order to minimize development time.

3.1 Functional Requirements

The functional requirements of the prototype are expected to evolve as the prototype versions generate feedback and new data becomes available. The initial elements implemented included:

- City summary reports.
- Custom graph builder via a data explorer.
- Intervention costs
- Database with a single instance (i.e. one data set).
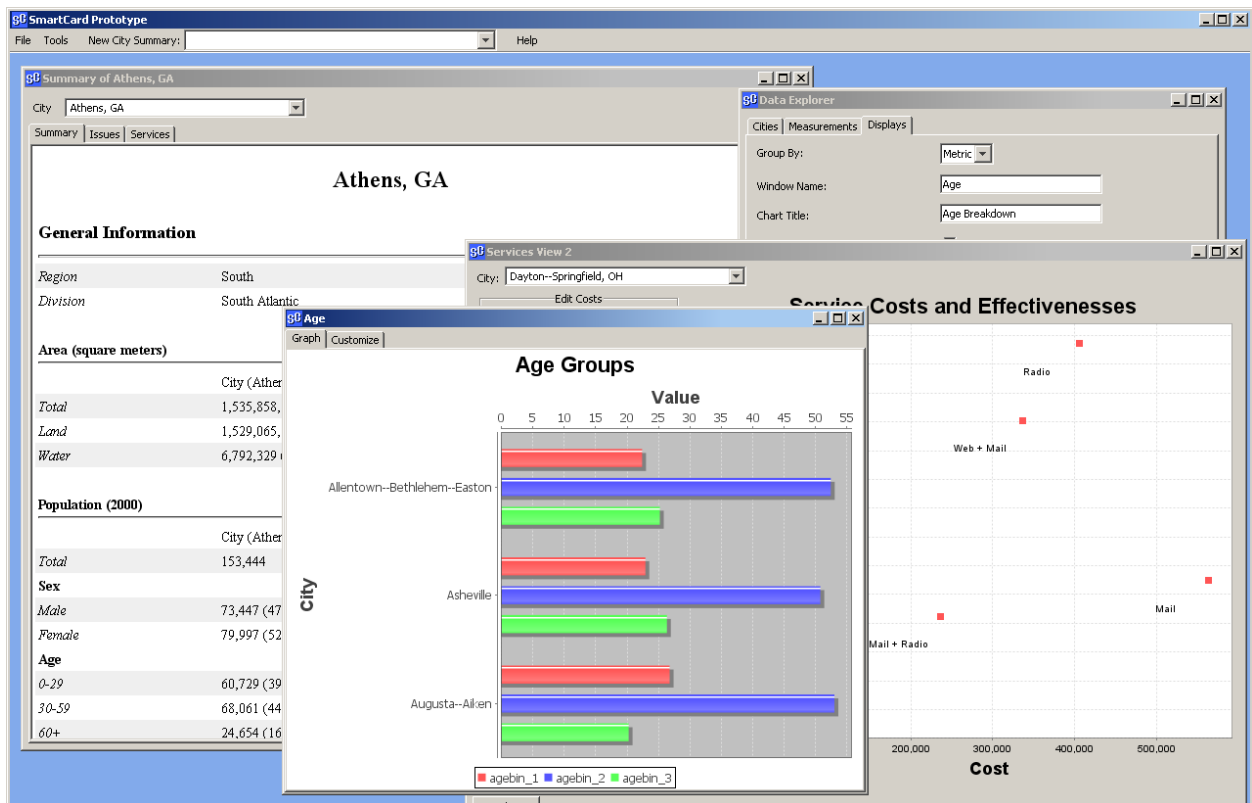- User help framework

Figure 6 Smart Card Prototype main window.

Development of this initial set of features allowed verification of the concepts and tools selected for the over development process. The next increment will focus on:

- Data management, consistency and documentation.

    o Fully implement the data hierarchy (schema) in the system.

    o Add data tagging, including support for user tagging.

- Usability

- Additional tools for data display and manipulation.

8

3.2 Database

The SmartCard employs a relational database to store and retrieve data from the U.S. Census Bureau and the 300 cities virtual experiment. From a user perspective, the data of greatest interest is contained in a rectangular table structured with city rows against columns of variables (an idiom familiar to many users from their use of spreadsheets). Ancillary tables hold additional data: comparison data (e.g. for geographic regions and the nation) and non-city data (for example a list of walk-in service centers, which need not be located in a city).

The city data resides in a logical hierarchy (which is currently partially implemented):

- **Schema**: A complete data set definition (with unique label). Initially only one schema will be provided with the SmartCard but additional schemas can be added as new studies are performed. SmartCard tools may be limited to working with a specific schema.

- **Version**: A specific version of the data set. A new version is created each time the schema definition is changed (e.g. variables are added or removed). Users do not normally need to take note the version, but SmartCard tools are verified to work at the version level.

- **Instance**: A data set or plane within a schema. A given schema/version may contain multiple instances and each instance can (and does) hold completely separate data. For example, if data from two censuses were employed to create two chronologically distinct city sets for a simulation study, the results could be stored in two instances (e.g. the "City_2000" and "City_2010" instances). In addition, while current SmartCard tools generally work with a single instance at a time, tools for cross-comparison of instances are feasible and will be constructed as such data sets are created.

- **Table**: A rectangular table of data in the form entity x variables (e.g. Pittsburgh: population, size, etc.). The city table is the key data table in the initial schema as it encapsulates the results of the 300 cities virtual experiment.

- **Bin** (optional): A group of associated variables that have a logical connection with each other. The user may want to manipulate such variables as a group. Bins typically provide a breakdown of another (continuous) variable or allied concept. For instance the age bin divides the overall population into age categories (i.e., young, middle-aged, and senior), with each person counted into a single bin.

- **Variable**: A data definition along with associated datum. This is the lowest level item in the current schema's hierarchy. In addition to having basic characteristics such as a name, type and size, SmartCard variables include:
  - **Tags**: Additional descriptive information which can be associated to a variable at will. Bin definitions, data logical hierarchies and other information are stored as predefined tags. In addition the user will be able to add tag information. This capability allows users to tag a set of variables as their "favorites" or as being relevant to a particular research issue such as "EITC",

for example. It further allows users to share user-generated tags with other users to support cooperative work.

- o **Description**: A definition of the variable's meaning (i.e. what kind of information the variable holds). Description information supplements the variable name to give the user a better sense of the variable's meaning.

- o **Source**: Information about the variable source or how the variable was computed or derived. Like the data description, this information allows users to make informed use of the variable.

3.3   Software Development Environment: Java and Eclipse

The SmartCard development effort uses the Eclipse integrated development environment (IDE) [2] and code written using the Java programming language [3]. This approach is consistent with several other major tools developed by CASOS (particularly ORA). Java applications are portable across multiple system types and Java includes a graphical user interface toolkit (Swing [4]). Java application performance can be lower than native application code, but this was not considered a critical weakness for the SmartCard, where the major need is allow a user to access and display data. However, the final decision with respect to the software development environment was contingent on finding and testing suitable third-party packages for database and data display (see Support Libraries).

Code is regularly archived in a Subversion repository, which is also consistent with current CASOS practice.

3.4   Support Libraries

Development of the exploratory prototypes emphasized the potential of employing third-party support libraries and helped define the suite of services that would best support building the developmental prototype. Functionality (features and ease of use) was the key requirement for selection but the selection process also required that the library not impose a license restriction that would require release of the SmartCard source code or data for public use. In addition, preference was given to open source and freeware libraries.

3.4.1   GUI Toolkit: Java Swing

A GUI toolkit provides the building blocks employed by the programmer to implement the interface design. For instance, the toolkit provides controls, dialogs and window components that the programmer customizes and combines to create the interface.

Java provides a graphical user interface toolkit, named Swing, which simplifies the creation of interfaces and can be adjusted to closely resemble the native look and feel used several of the most popular computer operating systems [4]. While the past Java offering in this area (AWT) was difficult to use, Swing is much easier to use and has been successfully applied to other CASOS projects. This selection was made in concert with the selection of data display libraries, due to the close relationship between the two tools. Additional packages extending the toolkit are available; the SmartCard prototype currently uses the Glazed Lists package [5].

3.4.2   Database: Apache Derby

The database enabled prototype demonstrated the utility of encapsulating the 300 cities data in a relational database. A programmer can define searches and sorts as simple commands, reducing the effort required to find and reorder data. A database also provides considerable potential for moving interface information out of the program source code to a common (and easily updated) location. Other benefits of a database include convenient mechanisms for storing system state and user preferences.

A database for the SmartCard does not need high performance, as the size of the database tables employed will be modest. The database does have to support independent operation on stand-alone systems which are not web enabled and should allow use of standard query formals (SQL). In addition, the database candidate had to be compatible with Java and preferably compatible with the Eclipse IDE.

Apache Derby proved to have the required features and a non-restrictive license [6]. Java code using Derby can be packaged to deploy as a single package or as a client in a client-server relationship as required. This means that while the SmartCard is will initially support a stand-alone database, it can subsequently be enhanced to use data on shared or centralized repositories.

3.4.3   Data Display

In the absence of a single package able to support the full range of data presentation envisioned multiple support libraries are employed. Use of multiple support libraries for data display in the SmartCard reflects both the diversity of outputs and the lack of any single suitable candidate package.

3.4.3.1   Graphs and Charts: JFreeChart

For numeric measures, various chart and graphs are required. JFreeChart [7] provides a wide range of customizable two-dimensional and three-dimensional chart and integrates easily with the Java Swing graphical environment. JFreeChart requires use of the JCommon package [8]. JFreeChart is also employed in other CASOS tools (although this was discovered post-facto).

3.4.3.2   Geospatial Visualization: TBD

Because much of the data is recorded by or derived from geographically distinct urban areas, geographical visualization of data may provide insight to relationships in the data. Support for displaying geographic data is currently under study.

3.4.3.3   Textual Data: TBD

Certain types of data will be presented in textual form or in mixed text and graphical reports. Textual data is already included in the prototype. The prototype employs the iText package [9], but additional tool support for rapid report creation and output is under consideration.

3.4.4   User Help: JavaHelp

As with other CASOS tools, help information will be included as part of the SmartCard (Figure 7). The JavaHelp package [10] is used to organize and present help information. JavaHelp allows the user to structure and display help topic pages created using HTML, where the user can create both a viewing hierarchy and an index. In addition, the user can both search and bookmark the help topics.
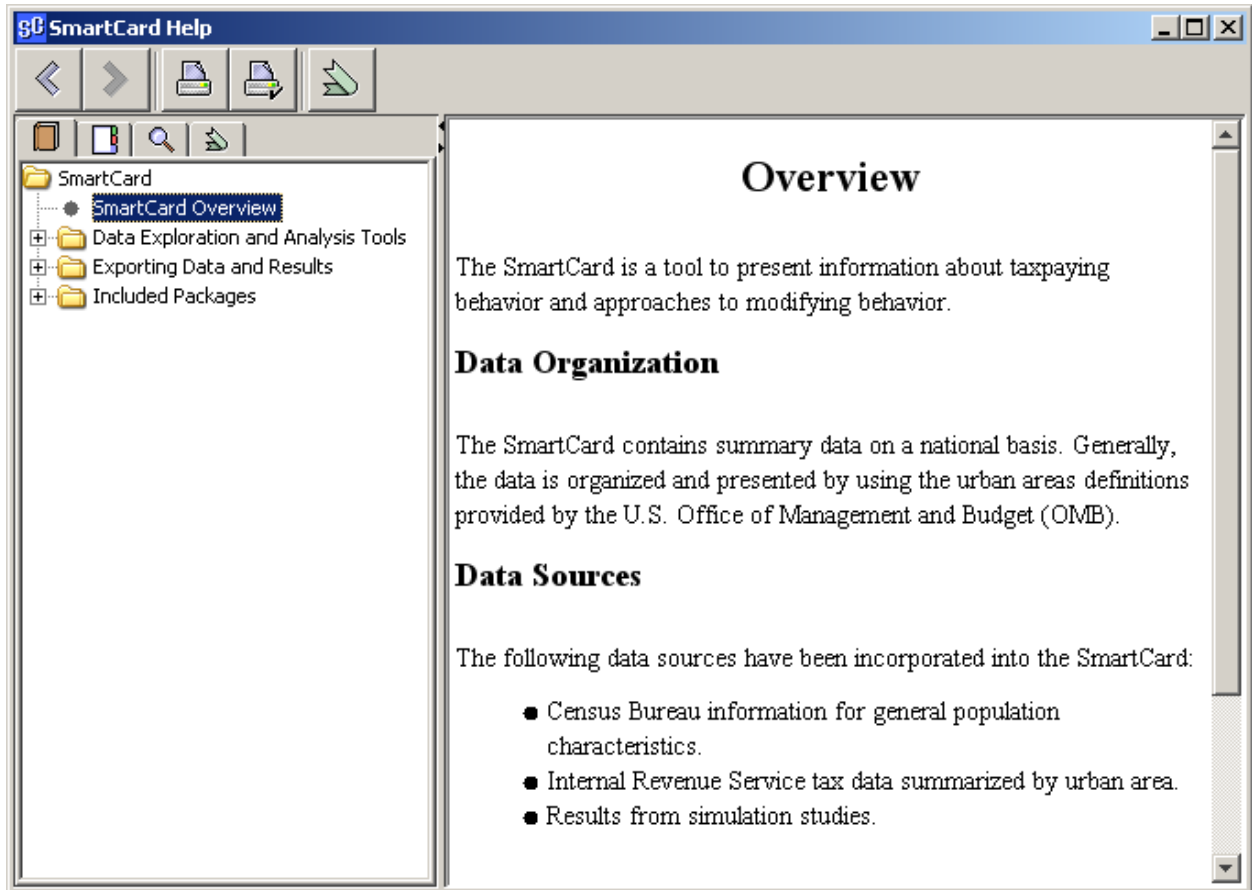

Figure 7 SmartCard Prototype help window.

4   Summary and Future Directions

The SmartCard is envisioned as a tool for users to conveniently access the results of the 300 Cites virtual experiment, which is intended to support decisions regarding the selection and deployment of IRS services. This report describes the initial technical approach, as defined through a series of exploratory prototypes, the current developmental prototype's development environment and the tool's functionality.

In the immediate future, the prototype will evolve to include a more general database and flexible tagging scheme. As final data and analysis results from the 300 cities study are produced, the SmartCard will incorporate the results, refine existing tools and add new tools to meet user needs and specifications.

5   References

[1]   McGillen, J., Martin, M.K., Robertson, D., Altman, N., & Carley, K.M. (under review). "300 Cities Virtual Experiment," Carnegie Mellon University, School of Computer Science, Institute for Software Research, Technical Report CMU-ISR-[xx]-[xx].

[2]   Eclipse: http://www.eclipse.org/

[3]   Java: http://www.java.com/en/

[4]   Swing: http://java.sun.com/javase/technologies/desktop/

[5]   Glazed Lists: http://publicobject.com/glazedlists/

[6]   Apache Derby: http://db.apache.org/derby/

[7]   JFreeChart: http://www.jfree.org/jfreechart/

[8]   JCommon: http://www.jfree.org/jcommon/

[9]   iText: http://www.lowagie.com/iText/

[10] JavaHelp System: http://java.sun.com/javase/technologies/desktop/javahelp/