TIN: TerraScope Image Navigator Providing Effective Ubiquitous Access to Distributed Multimedia Earth Science Data

By

Amaury Cabarcas Álvarez

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

MASTER OF SCIENCE
In
COMPUTER ENGINEERING

UNIVERSITY OF PUERTO RICO
MAYAGÜEZ CAMPUS
2003

Approved by:

_____   _____
Manuel Rodríguez Martínez, Ph.D.   Date
Member, Graduate Committee

_____   _____
Pedro I. Rivera Vega, Ph.D.   Date
Member, Graduate Committee

_____   _____
Bienvenido Velez, Ph.D.   Date
President, Graduate Committee

_____   _____
Dorial Castellanos, Ph.D.   Date
Representative of Graduate Studies

_____   _____
Jose L. Cruz Rivera, Ph.D.   Date
Chairperson of the Department

Abstract

In this thesis we describe the design of the TerraScope **I**mage **N**avigator (**TIN**), the graphical user interface module of the TerraScope system. TerraScope is an earth science data middleware system that was designed to facilitate collaboration among a set of providers (peers) who wish to provide their own distributed geospatial data. The base of the system is a distributed database that collects heterogeneous spatial data taken from satellite ground stations. **TIN** as a component of TerraScope, retrieves images and related information from the distributed database and deals with the effective presentation of this kind of data to the user. The **TIN** prototype was implemented using an interactive movie authoring environment (Flash MX) and XML (e**X**tensible **M**arkup **L**anguage) to communicate queries and retrieve data and metadata from the server. As a result, we expect to port **TIN** to multiple platforms, including portable devices, with relative ease.

The idea behind **TIN** is allow to the users post queries and geo-represent automatically the results from the database. The user makes queries easily selecting the area of the earth he/she is interested in, date intervals and the sensor or sensors, then TIN represents the location of the images that match the user's search. Each image serves as a hyperlink that can be used to navigate into smaller geographical area where the user may recursively search the database for sub-images contained within the geospatial region covered by this base image.

Resumen

En esta tesis presentamos el diseño del **N**avegador de **I**mágenes de TerraScope (**TIN**), la interfaz gráfica del sistema TerraScope. TerraScope es un sistema intermedio para manejar datos terrestres que fue diseñado para facilitar la colaboración entre un conjunto de proveedores (peers), quienes desean compartir una imagen o un conjunto de datos. El soporte del sistema TerraScope es una base de datos distribuida que almacena información espacia heterogénea de estaciones satelitales en tierra. **TIN,** como un componente de TerraScope, obtiene imágenes e información relacionada con las imágenes, de la base de datos y busca presentar este tipo de información de una manera efectiva a los usuarios(as). TIN fue desarrollado usando una herramienta para desarrollo web (Flash MX) y usando Java Servlets y XML para la comunicación entre el cliente y los servlets. Como resultado esperamos poder usar **TIN** en múltiples plataformas, incluyendo dispositivos portátiles, con relativa facilidad.

TIN crea geo-representaciones de los resultados a las búsquedas planteadas por los usuarios(as), que pueden ser creadas fácilmente seleccionando el área en la que se esta interesado, un intervalo de fechas y el sensor o los sensores que tomaron las imágenes. Cada imagen funciona como un hipervínculo que puede ser usado para navegar dentro de regiones geográficas más pequeñas donde los usuarios(as) pueden buscar recursivamente por imágenes contenidas en la imagen base.

Acknowledgements

I wish to thank my advisor Dr. Bienvenido Vélez for his kind guidance and support during my Master thesis.

I also wish to thank the members of my committee Dr. Dorial Castellanos, Dr. Manuel Martinez and Dr. Pedro I. Rivera Vega.

I would like to acknowledge the academic contributions of Lizvette Malavé regarding this work.

Table of Contents

Chapter One

Introduction

Chapter Two

TerraScope System Architecture

Chapter Three

The Graphical User Interface

Chapter Four

Assessment Methodology and Experiments

List of Tables

List of Figures

Chapter One

Introduction

1.1 The Need for TerraScope

The emergence of multimedia technologies and the possibility of sharing scientific information through broadband computer networks have exacerbated the need for geographical information technologies. In the typical scenario, geospatial data is periodically collected and stored by entities that often differ in technical, administrative and even political domains. Multiple satellites orbits the earth to collect spatial data (Figure 1a), these data is send periodically to ground stations, Figure 1b. The research centers and universities take the images collected by the ground stations and store them in databases. These research centers and universities have been making theses databases available over the Internet providing effective access to the graphical/textual information from images that are stored in these databases.
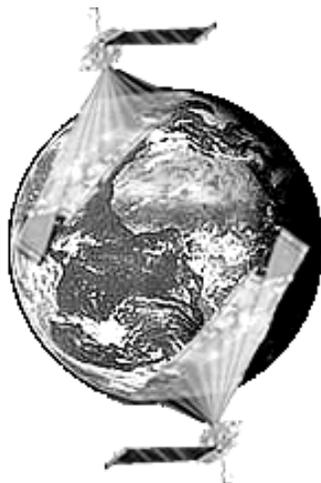
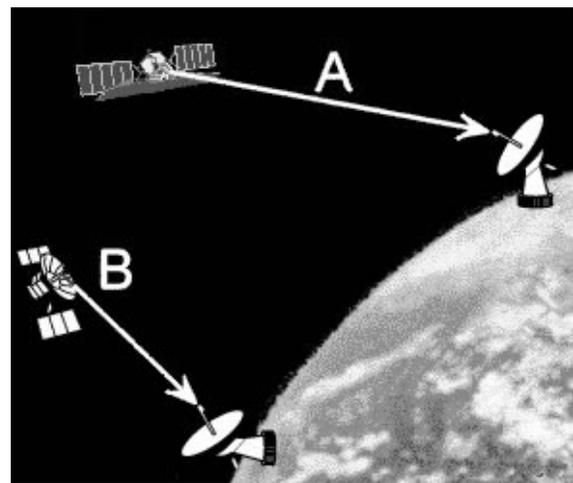Figure 1a.How Multiple Satellite Collected Data   Figure 1b. Transmitting data to Ground Stations

Because ground stations differ in technical, administrative and even political domains, the nature of geospatial data is inherently distributed. This distribution should be of no concern to scientists. Solutions attempting to offer integrated access to this distributed data by relying on centralization of data repositories are not often feasible. An information system that attempt to offer integrated access to the information should enable and facilitate collaboration with minimal loss of autonomy among a set of distributed data providers who wish to collaborate providing their own data repository.

The use of distributed database systems combined with the new technologies for web development facilitates sharing and accessing information stored on local or remote databases in an easy and transparent way for the user. This research emerges from the need to effectively access these large spatial databases and to know how to recover and visualize graphical/textual information from images that are stored in these sources.

Current applications such as Geographic Information Systems (GIS), digital libraries, and search engines impose some demands on the efficiency of query processing and the way how the data and metadata must be shown to the user. Based on this principle we are designing and developing the **TerraScope** Earth Science distributed peer-to-peer database middleware system conformed by a Search and Retrieval Engine (**SRE**) [1] and a Graphical User Interface (GUI) called **TerraScope I**mage **N**avigator **(TIN)**. The purpose of this thesis is to create the image navigator as an integral part of TerraScope Earth Science System.

The remainder of this chapter is organized as follows:

Section 1.2 presents our research objectives.

Section 1.3 presents the approach for the image navigator

Section 1.4 reviews related work.

Section 1.5 summarizes our most important contributions.

Section 1.6 presents the structure of this writing.

1.2 Research Objectives

In this dissertation we introduce **TerraScope I**mage **N**avigator **(TIN)**: an interactive image navigator designed to perform spatial exploration in conjunction with a **S**earch and **R**etrieval **E**ngine **(SRE)**. SRE supports the execution of different queries that make possible to gather information stored among several data centers, with different formats or characteristics. SRE is currently under development by another member of the Advanced Data Management Group.

The main objective is to develop an interactive image navigator **(TerraScope Client)** designed to perform spatial exploration in conjunction with a Search and Retrieval Engine **(TerraScope Server)**.

The design of **TIN** is driven by the following goals:

- Build a prototype image navigator for TerraScope on top of standard technologies for building interactive movies, for example Flash MX[1], for the World Wide Web.

- Offer ubiquitous access to satellite data and metadata available at several locations through the web.

- Support operations such as panning and zooming over the image and display specific metadata for the image.

- Support recursive navigation and automatically georeferenced images.

- Support time-varying image databases.

- Support multiple distributed sources.

1.3 Approach

TIN was developed using Flash MX, XML [2] and ActionScript [3], this image navigator delivers satellite images with their corresponding metadata, GIS characteristics, and other information. Once an image of interest is located, users may easily search the

---

[1] Macromedia Flash MX is the fastest way to create rich Internet content and applications with a better return on investment.

database for images contained within the seed image (sub-images) or for images overlapping the seed image. In other words, each browsed image provides a geospatial context from which future exploration may proceed.

1.4 Related Work

In an effort to amortize the high costs associated with the deployment of the expensive equipment and communications infrastructure required to collect satellite imagery many scientist, research centers and universities have been developing systems to share their image data over the Internet. In this section we compare some of these systems emphasizing on the features that make TerraScope and **TIN** unique.

Actually in the Internet the main need is to obtain information in a quick, precise and instantaneous way, independently of the geographical localization of the given information; this reason has motivated many research centers and universities to make their databases available over the Internet. The **Quicklook Swath Browser** is an image browser developed by the Canada Centre for Remote Sensing and tries to bring a solution to this critical issue (**http://quicklook.ccrs.nrcan.gc.ca/ql2/en**) [4]. This research has some similarities with TIN, both create geo-representation of the images, and use the same mechanism to show the data and metadata to the users, combined text and graphics and use a graphical user interface accessible through web. Also both systems provide automatically hyperlinked images. The main difference between TerraScope and Quicklook Swath Browser is the distributed nature of the TerraScope database.

Quicklook Swath Browser is Java based while TIN GUI was developed using Macromedia Flash MX. Also, Quicklook Swath Browser does not support the type of recursive image navigation supported by **TIN.**

The USGS Global Visualization Viewer (**GloVis**) [5] is a quick and easy online search and order tool for selected satellite data. GloVis allows user-friendly access to all available browse images from the Advanced Spaceborn Thermal Emission and Reflection Radiometer ASTER TIR[2], ASTER VNIR[3], Landsat 7 (ETM+), and Landsat 4/5 (TM) sensors. Through a graphical user interface, the user can select any area of interest and quickly view all available browse images within the United States Geological Survey (USGS) inventory for the specified location. The principle difference between TerraScope and GloVis is the distributed nature of the TerraScope database. GloVis GUI is Java based while TIN GUI was developed using Macromedia Flash MX. Also, GloVis does not support the type of recursive image navigation supported by **TIN.**

NASA's Visible Earth System [6] is an image browser that provides a consistently updated, central point of access to the superset of NASA's Earth science-related images, animations, and data visualizations. The Visible Earth System is a web based image browser that does not have a graphical user interface and does not bring support for recursive image navigation. Another difference between TerraScope and Visible Earth is the distributed nature of the TerraScope database.

---

[2] Surface leaving radiance product that provides an estimate of the total radiance leaving the surface including any reflected sky component and an estimate of the sky irradiance for each of the five ASTER TIR bands.

[3] Surface leaving radiance and reflectance product that provides an estimate of the total radiance leaving the surface including both the reflected solar and sky components for ASTER bands 1-9.

The **Microsoft**® **Terraserver** [7] is one the largest public repositories of high resolution aerial, satellite, and topographic imagery. Terraserver stores its data in a relational database system and makes it available via the Internet from virtually any graphical web browser. Users can zoom and pan across a mosaic of tiles. TerraServer contains 3.3 tera-bytes of high resolution United States Geological Survey (USGS) aerial imagery and USGS topographic maps. Users can locate imagery by clicking on a map, entering a city or town name in the "Search TerraServer" form, or entering a U.S. street address. This research has some similarities with **TIN**. For instance, both Terraserver and TIN, allow similar user actions including submitting a query, and zooming and panning a particular image. One important difference between TerraScope and Terraserver is the distributed nature of the TerraScope database. TerraScope supports a set of distributed data sources while in Terraserver the data source is a centralized data warehouse. While Terraserver was designed for a static set of images, TerraScope was designed to support continuous collection of image data; also Terraserver does not support the type of recursive image navigation supported by **TIN.**

The **SAND** (Spatial and Non-spatial Data) system is a spatial database system developed at the University of Maryland [8]. This project shows an interactive image navigator for accessing spatial online databases. The users can interactively and visually manipulate spatial data remotely. This manipulation has the disadvantage of making the interaction very slow because this data generally is in the order of Gigabytes. SAND and **TIN** are distributed systems and both systems allow users to manipulate spatial data remotely. Both systems have a Graphical User Interface (GUI). SAND GUI is Java based

while **TIN** GUI was developed using Macromedia Flash MX. Also, **SAND** does not support the type of recursive image navigation supported by **TIN**.

A prototype system, similar to a GIS (Geographic Information System) where several types of data are integrated into one form was developed in [9]. The areas explored in that project include visualization, mapping, 3D object modeling and layering of different types of information.

G-Portal is a web portal providing digital library services over geospatial and geo-referenced content found on the World Wide Web (WWW) [10]. **G-Portal** adopts a map-based user interface to visualize and manipulate the distributed geospatial and geo-referenced content. The principal aim of this project is the identification, classification and organization of geospatial and geo-referenced resources on the web, and the provision of digital library services (e.g. searching, visualization) for these types of resources. G-Portal and **TIN** provide basic search and retrieval services of geospatial data over the Internet; both systems make a geographical representation for each geospatial object. However, **G-Portal** does not provide automatically hyperlinked images the way **TIN** does.

GeoWorlds system [11] integrates GIS, spatial digital libraries and other information analysis, retrieval and collaboration tools. **GeoWorlds** demonstrates how carefully integrating three key technologies, the WWW, Digital Libraries and GIS, can

provide teams of users with a sense of shared regional vision. It retrieves, organizes and displays available information about a particular region.

The prototype system presented in **Remotely-Sensed Geophysical Databases** [12] describes a prototype that integrates geophysical data and its metadata from both satellite and in situ sources, using a relational general-purpose Database Management System to manage the catalog and observational data, and a video optical disk to archive images, this prototype and TIN provide basic search and retrieval services of geospatial data over the Internet; TIN has multiple and distributed DataBase Management System (DBMS) and **Remotely-Sensed Geophysical Databases** use one DBMS. **Remotely-Sensed Geophysical Databases** does not provide automatically hyperlinked images the way **TIN** does and does not use recursive queries.

The **WebSeek** [13] image retrieval system integrates textual and visual features for categorizing images and videos. **IPURE** [14] and **ZUI** [15] provide new methodology for formulating queries and new ideas to show users the images and provide new features in the way users can browse and navigate the images. None of these systems support recursive navigation of the image database and do not provide automatically hyperlinked images the way **TIN** does.

Remotely Image Navigator of Virtually Hawaii Project [16] is a web based tool for finding remote sensing images of Hawaii with a range of spatial resolutions. These images come from a variety of sources, including instruments carried on aircraft,

satellites and the Space Shuttle. The Remotely Image Navigator and TIN has several similarities, both support the recursive image navigation into smaller geographical area where the user may recursively search the database for sub-images contained within the geospatial region covered by this base image, also provide automatically hyperlinked images. The main difference consist in Remotely Image Navigator restrict the search area to the archipelago of Hawaii. TIN initially was created to cover the Caribbean region but in a future we want extend it to cover all earth surface. The Remotely Image Navigator does not provide basic search and retrieval services of geospatial data over the Internet; TIN has multiple and distributed databases and Remotely Image Navigator use one and local database.

Three features distinguish **TIN** from others proposed interactive navigators for accessing online geographical image databases. First, **TIN** has the capability to dynamically and recursively generate hyperlinks by embedding automatically geo-referenced sub-images within the images being inspected. Second **TIN** allows a type of navigation that we call *"recursive",* once the users select an image they may recursively search the database for sub-images contained within the geospatial region covered by this base image. Third, **TIN** was completely developed on an interactive movie authoring environment (Flash MX) making it accessible via any web browser with a freely available Flash player installed. Such players are available for virtually every major computing platform including portable devices (e.g. PDAs).

1.5 Summary of Contributions

In summary, this research makes the following contributions:

- Proposes a feasible solution to the critical issue of providing effective access to large spatial dynamic databases and allowing researchers to visualize graphical/textual information from images that are stored at distributed sources.

- Demonstrates the feasibility of implementing interactive image retrieval/browsing systems being an interactive movie authoring environment. This makes the application portable, easy to use and ready for deployment over the Web. All communications with the TerraScope server engine occurs via a client API that we have built on top of the standard Web technologies

- Developed an image navigator that enables users to pose spatial queries by simply pressing buttons and drawing boxes around a specific region or image section. Most importantly, **TIN** is designed with the capability to specify *hyperlinks* within the images being inspected and allows users to recursively browse the image database for sub-images contained within the geospatial region covered by any retrieved image.

- Provided a system prototype to the researchers of the **T**ropical **C**enter for **E**arth and **S**pace **S**tudies (**TCESS**) and the **NSF Cen**ter for **S**ubsurface **S**ensing and

Imaging **S**ystems (**CenSSIS**) that allows integrate and access federate heterogeneous data collections stored at geographically distributed data centers.

1.6 Structure of the Thesis

The reminder of this thesis is organized as follows: Chapter 2 discusses TIN's architecture, the graphical user interface and the image retrieval process in more detail. Chapter 3 shows the methodology used and the obtained results in the usability study. And finally, Chapter 4 shows our conclusions and suggests some areas for future work.

Chapter Two

TerraScope System Architecture

2.1 System Architecture

As shown in Figure 2, the TerraScope system follows the classical client-server architecture. The client consists of a Flash MX interactive movie, running inside a web page that can be accessible using any web browser (item 1 in Figure 2). The server consists of a set of Java servlets running inside a web server (item 2 in Figure 2). These modules implement an abstraction of a single data repository. However, the servlets implement this abstraction by communicating with multiple TerraScope SRE's peers. The servlets also provide a mechanism for interfacing TerraScope with virtually any available database system. Our current prototype stores data in a POSTGRES database.
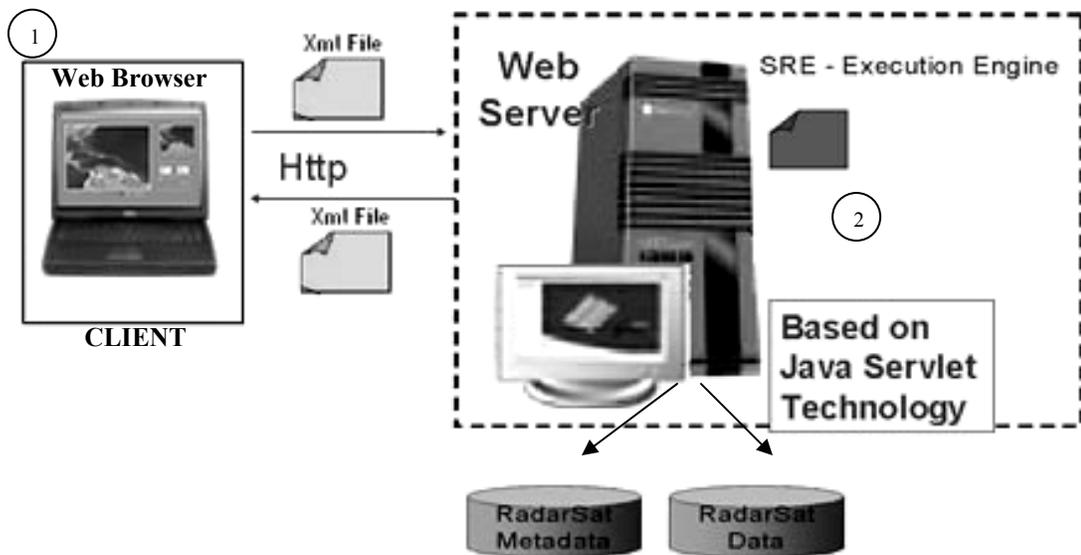

Figure 2   TerraScope Front-End Architecture

Using Macromedia Flash MX as a development platform, we have designed the first version of the **TerraScope I**mage **N**avigator (**TIN)** prototype. Users specify queries by manipulating familiar GUI elements TIN (e.g. textboxes, pull-down menus, checkboxes, etc.). TIN translates these queries into XML format before sending them to the SRE (Figure 3). The SRE computes the set of results by potentially forwarding queries to other SRE's believed to hold data pertaining to the query. The SRE then packages the results into an XML message similar to the one shown in Figure 4 and returns it to the client. The response includes a list of images and, for each image, its URL, condition, date, time, coordinates, source and any other image specific data stored in the database.

```
<? xml version="1.0"?>
<TerraScope>
<query>
        <querytype>3</querytype>
<source>
        <source1>CENSSIS</source1>
        <source2 />
</source>
<coordenada>
        <latitud1>95</latitud1>
        <longitud1>90</longitud1>
        <latitud2>3</latitud2>
        <longitud2>52</longitud2>
</coordenada>
<start_date>10/1/2003</start_date>
<end_date>10/5/2003</end_date>
</query>
</TerraScope>
```
Figure 3 Example TerraScope response in XML

The client parses the XML message and automatically geo-references the retrieved images into the previous base image using the geographical metadata associated with all the images.

```
<?xml version="1.0"?>
<TerraScope>
<CENSSIS>
    <results>
        <id_swath27967_1>
            <bound>
                <latitud0>28.035</latitud0>
                <latitud1>27.91527778</latitud1>
                <latitud2>24.82472222</latitud2>
                <latitud3>24.70472222</latitud3>
                <longitud0>71.79055556</longitud0>
                <longitud1>70.76722222</longitud1>
                <longitud2>72.47111111</longitud2>
                <longitud3>71.47555556</longitud3>
            </bound>
            <condition>Descending</condition>
            <duration_swath>00:54.30</duration_swath>
            <start_date>2001-03-14</start_date>
            <start_time>10:44:47</start_time>
            <url>http://icarus.ece.uprm.edu/~ecoronado/imagescens/27967_1.jpg</url>
        </id_swath27967_1>
        <numregs>1</numregs>
    </results>
</CENSSIS>
</TerraScope>
```

Figure 4 Example TerraScope response in XML

2.2 Summary

This chapter discusses the architecture and implementation of TIN; in this section we presented TIN's classical client-server architecture. TIN consists of a Flash MX interactive movie, which runs inside a web page that can be accessed using any web browser. The server consists of a set of Java servlets running inside a web server.

Chapter Three

The Graphical User Interface

3.1 Introduction

TIN was developed using Flash MX, XML and ActionScript. This image navigator delivers satellite images with their corresponding metadata, GIS characteristics, and other information. TIN is a graphical user interface accessible through any web browser that has the ability to manipulate images and provides basic search and retrieval services of geospatial data over the Internet; make a geographical representation for each geospatial object Once an image of interest is located, users may easily search the database for sub-images contained within the base image or for images overlapping the base image. In other words, each browsed image provides a geospatial context from which future exploration may proceed.

Using familiar GUI controls TIN users can restrict the scope of their search to a specific data repository, geographical region, and type of satellite sensor (e.g. MODIS, RADARSAT and Landsat 7) or collection date. By default the scope of the search is delimited by the geographical region covered by the entire image. TIN connects to an assigned TerraScope SRE and retrieves a result set from it. However, in order to satisfy the user request, this SRE may contact other peer SREs holding data matching the criteria specified by the user. Thus, SREs act as servers to TIN clients, and also as clients of other SREs. This type of architecture is often termed peer-to-peer. When TIN receives

the result set from the SRE it dynamically geo-represent it. The current prototype contains data collected by the Tropical Center for Earth and Space Studies at the University of Puerto Rico Mayagüez Campus. The ground station collects imagery from the Caribbean region.

3.2 Graphical User Interface

Using Macromedia Flash MX as a development platform, we have designed the first version of the **TerraScope I**mage **N**avigator (**TIN)** prototype. As shown in Figure 5 the GUI consists of two image windows and a number of input controls. The main window on the left side (item 2 in Figure 5) displays the currently selected section of the image being browsed. By default the main window shows an image of the area covered by the satellites scope reducing the search region to the Caribbean. This section can be manipulated using TIN's zooming and panning controls (item 3 in Figure 5). Initially TIN displays a map showing the geographical region covered by the satellite ground stations contributing data to the distributed repository. The smaller window on the upper right hand side of the screen (item 4 in Figure 5) continuously outlines the region in the image that the user is examining. The item 1 in the Figure 5 shows the selected section by the user, this section can be selected using TIN's rubber banding controls.

The current prototype includes data collected by the Tropical Center for Earth and Space Studies (TCESS) at the University of Puerto Rico Mayagüez Campus. The ground station collects imagery from the Caribbean region.

Using familiar GUI controls TIN's for example textboxes, pull-down menus, checkboxes, (item 5 Figure 5) users can restrict the scope of their search to a specific data repository, geographical region, type of satellite sensor (e.g. NOAA, RADARSAT and Landsat 7) or collection date. Figure 5 show that a user has selected the region that shaded is the center of the image using a rubber band tool.
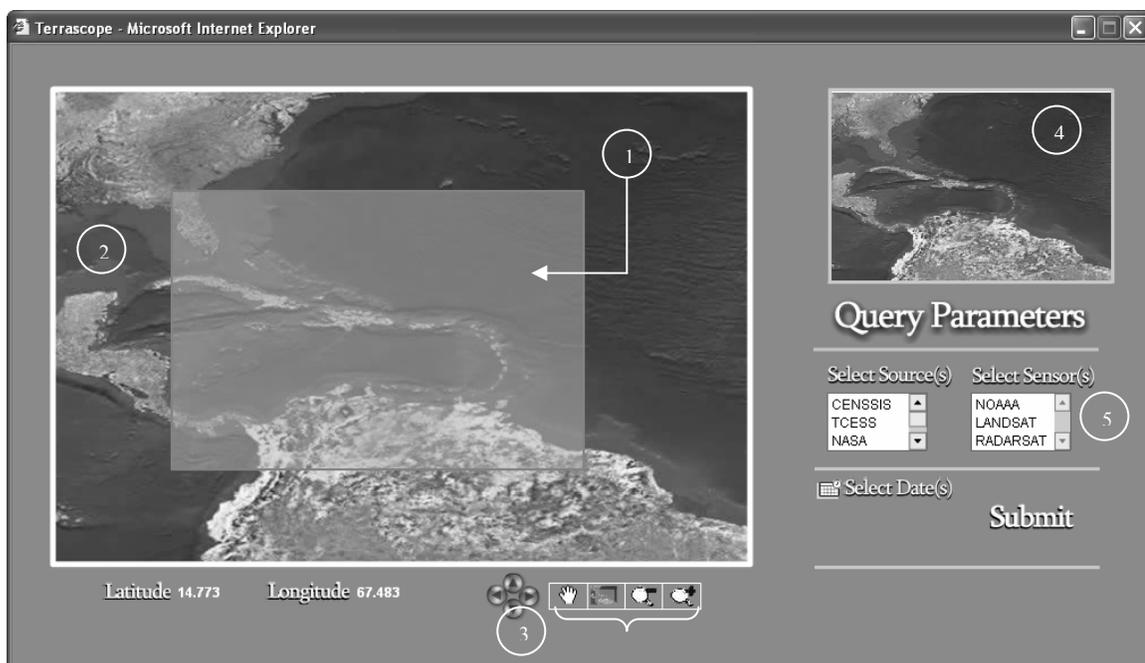

Figure 5 Snapshot of TIN prototype

Users specify queries by manipulating familiar GUI elements TIN; TIN translates these queries into XML format before sending them to the SRE (Figure 3). The SRE computes the set of results by potentially forwarding queries to other SRE's believed to hold data pertaining to the query. The SRE then packages the results into an XML message similar to the one shown in Figure 4 and returns it to the client. TIN parses the

XML message and automatically geo-references the retrieved images into the previous base image using the geographical metadata associated with all the images.

After parsing the XML result message file, converts all geographical coordinates (latitude and longitude) to pixel coordinates and displays a superimposed polygonal outline of each image in the result set (item 1 in Figure 6). In addition, TIN displays a pop-up window with the metadata in a text box in the low right corner of the screen (item 2 in Figure 6). Using this pop-up window the user can examine any non-graphical data associated with each image; this pop-up window is associated with the mouse rollover event in each image.
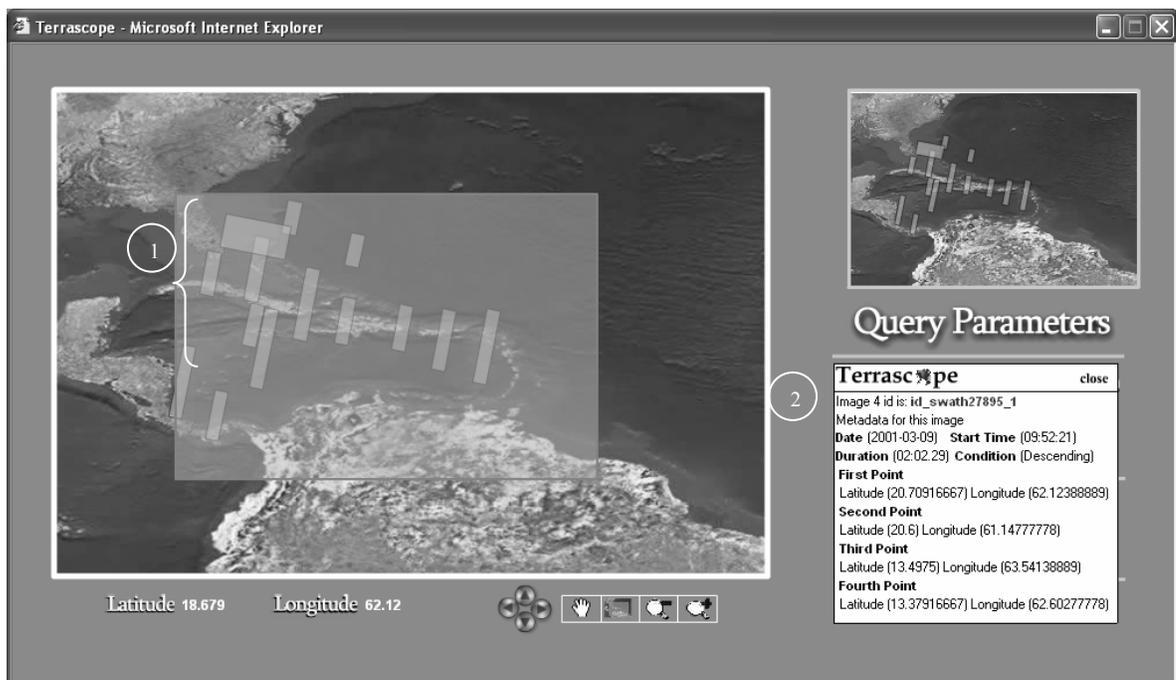

Figure 6 Sample output with Window Metadata

Each polygonal outline serves as a hyperlink that the user can click in order to transport the search scope into a smaller region. This facilitates exploration of smaller

regions with higher resolution images. For example in Figure 6, if the user wants to inspect other image, he/she only needs to click on the geo-representation of the image.
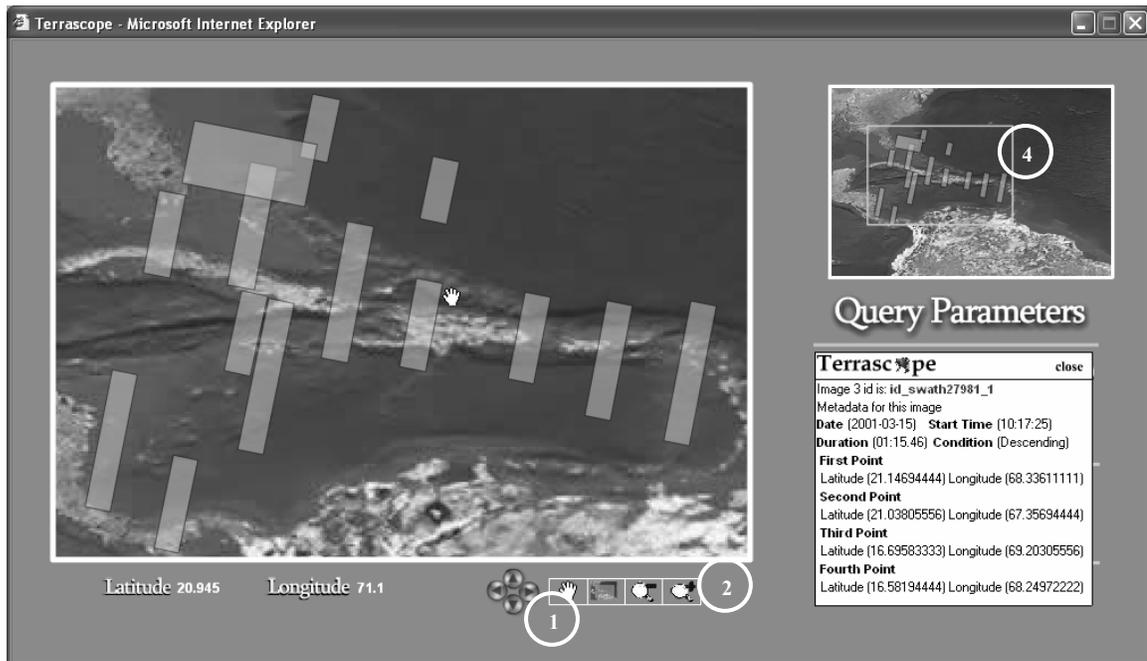

Figure 7. The TIN periscope (item 4) outlines the area that the user is examining in the main window.

If a user chooses to explore the current image, she can use panning and zooming controls (items 1 and 2 in Figure 7) to move around the image or get closer to specific areas. The small window on the upper right corner continuously displays an outline (item 4 in Figure 7) indicating the area in the image currently displayed in the main window. The importance of this periscope window is greater when one explores images without landmarks and other aids to maintaining the user's geographically orientation.

Figure 8 shows the resulting screen after clicking on one of the polygonal outlines from Figure 6.

Figure 8 Sample output from the TerraScope Client when a user retrieves an image.

At this time, the newly selected image becomes the new base image. Any further navigation is geographically constrained to the region spanned by this new base image. In other words, each browsed image provided a geospatial context from which further exploration may proceed. The user can either manipulate the base image (e.g. zooming and panning) or conduct a new search for images contained within this base image. In the current prototype the initial image of the Caribbean region is simply a loaded by default by TIN during its initialization. Figure 9 shows the result for a recursive search constrained by the image shown in Figure 8.

While users perform successive queries, a history mechanism maintains a record of previous queries. In case users forget the search terms used to retrieve the corresponding set of images, he/she just need to click the back button and the search terms and the

respective images to the query will appear at the bottom of the history section. This feature in the current prototype allows users go backward to the first result set after post multiples queries. For example, if users are in the Figure 9 and wants see the result for the initial query in the Figure 6, they can use the back button and see all the intermediate results obtained.
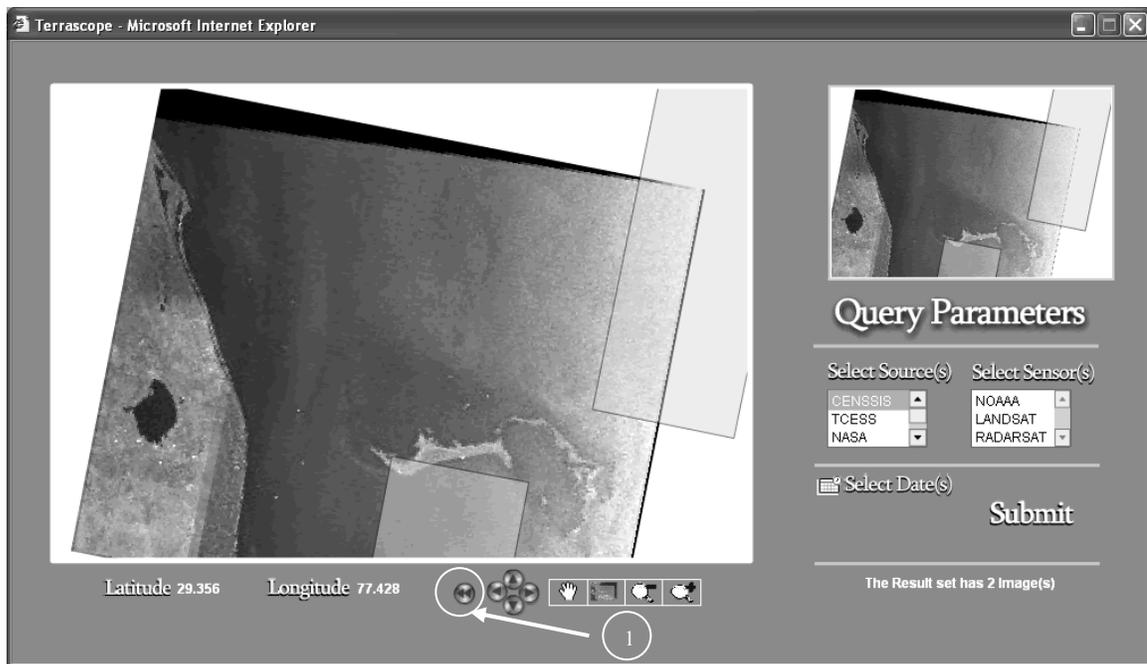

Figure 9 Example of recursive navigation

This unique feature makes of TIN a real image navigator and brings the possibility of navigating between several results sets. In the Figure 9 we can see the back button marked with item 1.

3.3 Lessons from using Flash MX

Flash was originally conceived as an interactive movie authoring tool for web development. Some research projects have begun to use Flash and its scripting language, ActionScript, as an application development tool. For instance, Halo [17] implements a technique for visualization of off-screen locations; in this case they designed and implemented a GUI using Flash to test these techniques in a Compaq iPAQ Pocket PC.

Our experience developing TIN proves that feature rich and robust client applications can be developed using Flash with much less effort an in shorter time span than with more traditional Web development tools. We have learned that Flash MX is a vehicle for delivering non-traditional content and information, within a player that is small enough to fit into PDAs. Yet, Flash MX is more than a web development tool; it is also a unique form of content itself, leveraging the strengths and working around the limitations of other dynamic web content authoring tools. Clearly Flash MX can be used to developed robust front end multimedia web applications.

3.4 Summary

Also in this chapter we showed all the details of graphical user interface and the image retrieval process; after the user specify the query parameters, TIN translate the query into a XML file, sends the request to the web server and waits for the response. The response is also an XML file, TIN analyzes and parses this file, then creates a geo-representation of every image in the result set. After that TIN displays all the retrieved

images and waits for the next query. After we designed and developed TIN we proved that feature-rich and robust client applications can be developed using Flash with less effort and a shorter time span than with more traditional Web development tools.

Chapter Four

Assessment Methodology and Experiments

4.1 Introduction

We believe that the combination of visual and text information with zoom and panning tools is the best way to show the data and metadata to the users and make better use of screen space than scrolling text. We describe our experiment and give some practical guidelines for future image browsers. In order to get a basis for understanding the context from which our system was designed, we use the following definitions according [12]:

An Image Retrieval (IR) System is an application that returns one or more images given some descriptive information. This information can be in the form of an image, Keywords or phrases, or Natural language

An Image Browser is an application that allows users to select one or more images from multiple images. This browser has to be able to display multiple images at one time (possibly reduced resolution versions), and support inspection of original full resolution versions of an image.

Image Browser display the returned set of images (results) obtained from the Image Retrieval query for further refinement of the search by the user. Often it is the case that

the results of the query yield more images than the user desires, so he/she has to browse. It is unfortunate that many query systems ignore browsing and just give the results of the query perhaps in the form of a list. This makes it hard and sometimes impossible for a user to select exactly what he/she needs. After analyzing these features in many browsers we decided to contribute to the scientific community and make our own application, a recursive image navigator called TIN.

4.2 Purpose of the Experiment

We performed a user study to assess our system in comparison with others systems and try to show how useful is the recursive query. We adopted the hypothesis that there would be no statistically significant differences in the time it took users to locate the targeted images, or in the number of incorrect selections made on a particular browser.

Name of Image Browser	Web Tool
Tcess Image Browser. Tropical Center for Earth and Space Studies	Java – Html
Quick Look Swath Browser. Canada Centre for Remote Sensing	Java – Html
Visible Earth. NASA	Html
GloVis. USGS Global Visualization Viewer	Java – Html
Virtually Hawaii	Html

Table 1: List of systems analyzed

To assess our system comparison with other systems, we developed a test prototype that assembled the main characteristics of current systems (see Table 1 for the list of systems analyzed for this experiment) in order to have the same database and properly evaluate user tasks on the systems. The prototype created to the tests show the result set using a table and a textbox.

4.3 Training

Before beginning the experiment, each participant was educated in the use of image browsers in general. We wanted to be sure each participant had a clear understanding of the assignment they were about to perform. Participants completed some pre-tasks using the first image browser that they would use. There was no time limit to do the pre-tasks; participants were informed that they did not have to proceed with any further tasks until they felt comfortable with the browsing system. The goal of training was to verify that our subjects understood the navigation techniques for our prototype.

4.4 Subjects Profile

There were 20 participants involved in this experiment; most of them were students at the University of Puerto Rico at Mayagüez, with various backgrounds including Computer Engineering and Electrical Engineering. Approximately 63% of the subjects were male and 37% of the subjects were female. Participants' ages were recorded using ranges so they would not feel uncomfortable disclosing their ages. From the data we collected, 79% of the participants were between the ages of 20 and 24, 21% between 25 and 31. 47% of subjects reported they were experts on the WWW and 47% of subjects reported they were intermediate user on the WWW, with the average user browsing 20 hours per week. Users also reported using a personal computer (PC) an average of 37 hours per week.

4.5 Results

This section presents some measurements of the performance of TIN followed by results from a small pilot user study designed to evaluate its unique combination of features.

4.5.1 System Latency

The results presented below measure TIN's response time as the real time elapsed from the instant the user clicks the submit button to the instant when TIN finishes displaying the results set. This time can be divided in at least three segments: a client segment, a server segment and a communication segment. The client segment includes packing the GUI controls into an XML message, parsing the resulting XML message, and embedding the outlines into the base image. The server segment consist of translating the XML query into one or more SQL queries, submitting the SQL query to the peer Postgres databases, processing of the queries by Postgres, returning the result sets to the SRE, combining the result sets, and packaging the result set in a single XML message. Communication time is the time that the XML messages and SQL queries remain in transit in the network.

The database currently used by TIN includes two peer SRE's which collectively hold 80 images. This is a rather small database compared to what would be expected in real geo-spatial collections. Unfortunately, our experiments have been constrained by the

unavailability of a large repository of images. An effort is currently undergoing to assess the performance of TIN on much larger databases.

The graph in Figure 10 shows system latency versus number of images retrieved by the query. The time measurements are taken with a completely offloaded SRE. Result set size varies from ten (10) to eighty (80) images. Each plotted time is averaged over several trials.
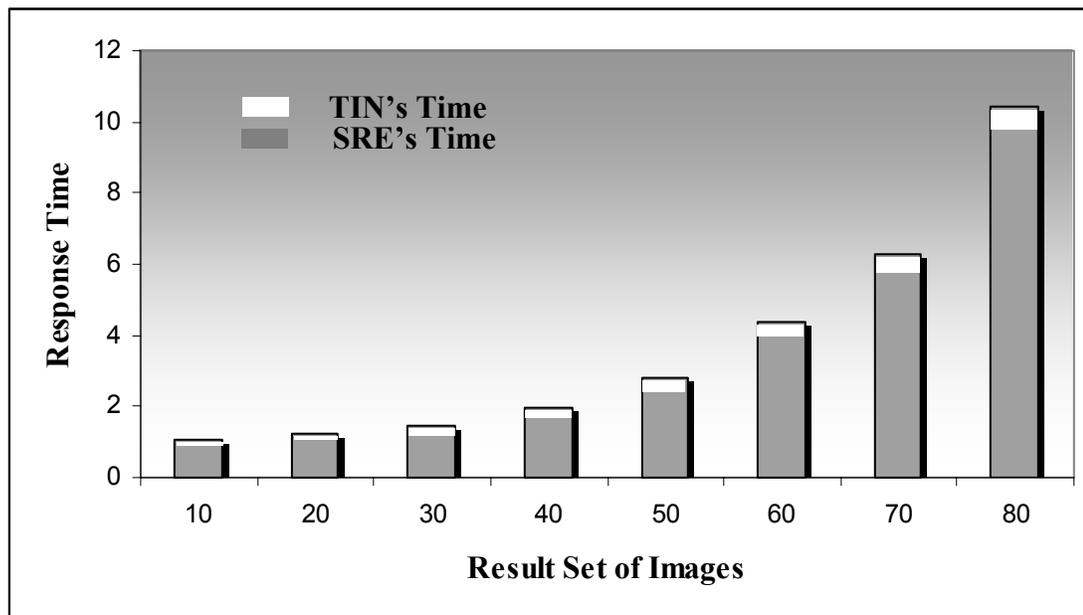

Figure 10. TIN's response time

As expected, response time increases with result set size at what appears to be a super-linear rate. We are currently consulting the Terrascope SRE team for an explanation of this growth rate. The response time for result sets with less of 40 images are always less than 2 seconds. For all the different result set sizes, TIN time remains in

the sub second range. Therefore, using a development tool like Flash we can create a client like TIN, which appears to run efficiently enough for practical use.

TIN uses Action Script's [3] built-in methods and properties of the XML object to build, send, load, parse, and otherwise manipulate XML documents. These methods appear to be highly optimized as the latency of the system is dominated by server time.

4.5.2 Pilot User Study

We performed a pilot user study to assess the browsing effectiveness of TIN relative to other existing image browsing/retrieval systems. We adopted the hypothesis that there would be not statistically significant differences in the time it takes to users to locate the targeted images, or in the number of incorrect selections made on a particular browser.

As a basis for comparison we developed a BASE prototype that included the main features of the list of currently available image retrieval systems shown in Table 1. Figure 11 (next page) shows a snapshot of the base system GUI. The main window now presents a list of images retrieved by the query. As the user scrolls this list, the corresponding image is highlighted in the small window on the upper right corner. Both TIN and the BASE access the same image database.

Figure 11. Snapshot of the BASE Prototype.

Each subject conducted two very similar tasks, one on each system. Figure 12, 13 and 14 summarizes the user satisfaction scores for the three main questions formulated in the exit survey. These questions asked users to rate each system in a scale from one to five with respect to a particular aspect. For each question we conducted an Excel ANOVA single factor test to determine statistical significance.
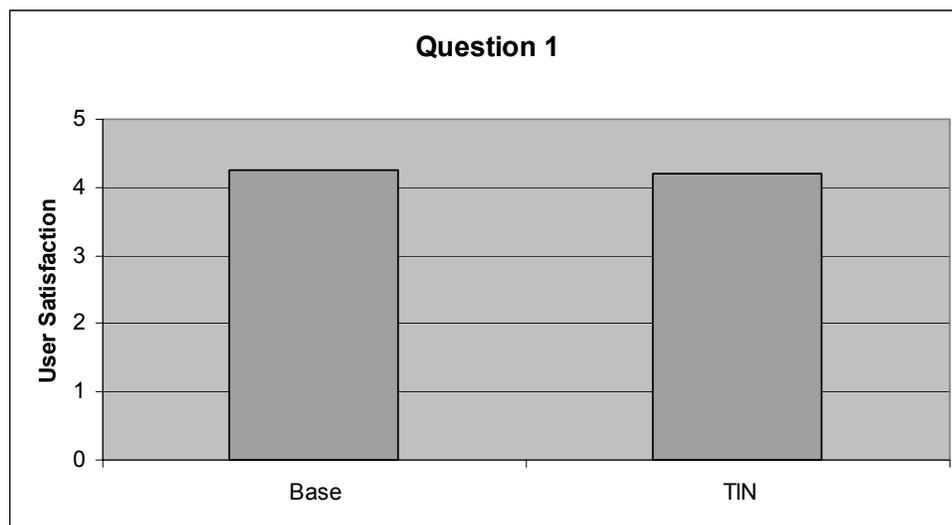

Figure 12. User satisfaction ratings the systems. (Question 1)

The Figure 12 plots user satisfaction with respect to the way that each system shows the image textual data and metadata. In this case BASE had a slightly higher score but the difference was not statistically significant (ANOVA single factor) ($p>.05$). With this results we can conclude that TIN does not appear affect metadata presentation.

Figure 13 represents user satisfaction with respect to the way that both systems allow access and manipulation of the retrieved images. This time TIN had the highest user satisfaction and this time the difference was statistically significant ($p<.05$). We can conclude that TIN improves image presentation and manipulation.
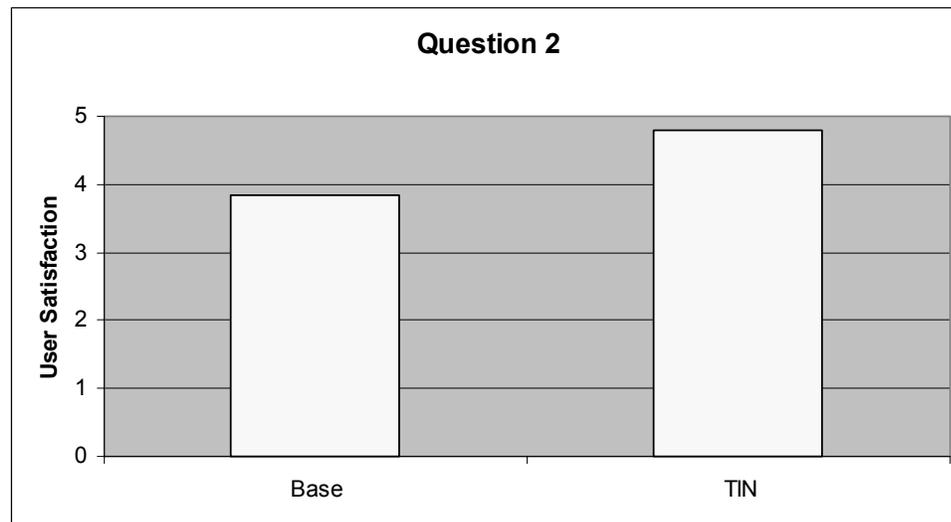
Question 2

Figure 13. User satisfaction ratings the systems. (Question 2)

Figure 14 represents the user satisfaction with respect to the way that systems combined techniques to show the images, image data and metadata to the users; in this case TIN had the highest user satisfaction. The ANOVA test showed a p-value of .08 meaning that the difference was not statistically significant. We can conclude that there is not evidence that TIN improves graphics/text combination.
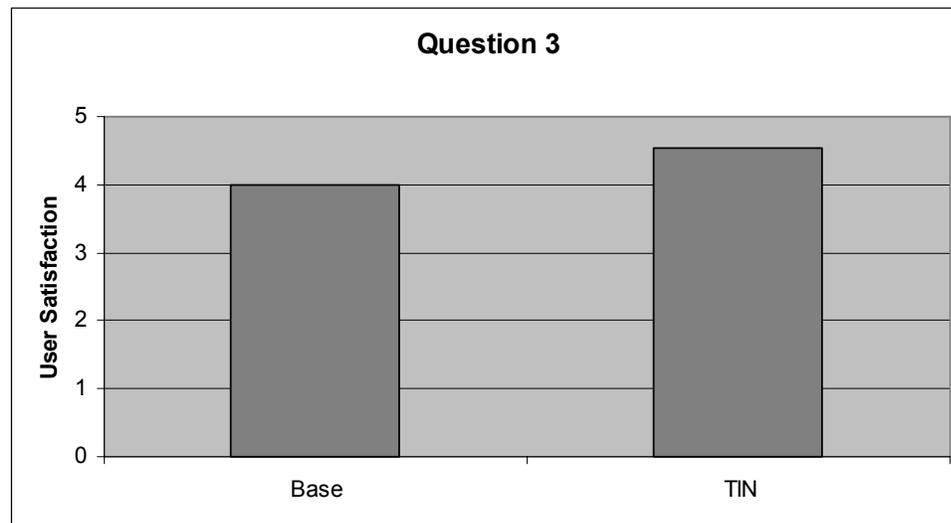
Figure 14. User satisfaction ratings the systems. (Question 3)

The users also evaluated the usability of the zooming and panning tools, the back button and the recursive queries used in the TIN prototype. All users considered panning and zooming very useful for inspecting and manipulating images with more detail. Also, all users expressed that the back button and the recursive query feature would be useful tools in image navigators to search, inspect and manipulate image of higher resolution.

4.6 Conclusions

The result of this experiment showed that TIN works best for images manipulation and image navigation; since provides the highest user satisfaction with respect to the way that allows access and manipulation of the retrieved images. In this case the users prefer TIN because the multiple tools provided by this navigator make of this system a good web-tool to manipulate and access the retrieved images.

After observed the result we can affirm that a system that mix multiple techniques to show the retrieved images and the respective data and metadata works best and bring highest user satisfaction.

Conclusions and Future Work

We have presented the design of TIN, the new image navigator front-end for the *TerraScope* peer-to-peer image data middleware system which is under development by the Advanced Data Management Research Group at the University of Puerto Rico Mayagüez. TIN is designed with the capability to dynamically generate *hyperlinks* connecting sub-images with the images being inspected with no human intervention. Users can geographically constrain their searches to a specific region using visual GUI tools. TIN presents query results graphically by automatically superimposing polygonal outlines of retried images onto any retrieved image serving as a base for the reach. A textual listing of the result set displays any non-graphical information associated with the retrieved images. This information may include: a URL for the actual satellite image, its geographical bounding box, date taken, radar sensor used to scan it, image size, among other types of metadata.

TIN supports recursive navigation. When the user selects an image the users may recursively search the database for sub-images contained within the geospatial region covered by this base image. Each retrieved sub-image is dynamically geo-referenced in the base image and becomes a hyperlink that can be used to navigate into smaller geographical regions with higher resolution. TIN was developed with Macromedia Flash MX, Action Script and uses XML to communicate queries and data to and from the SRE. Macromedia Flash MX has proven to be a fast way to create multimedia rich platform-independent Internet applications with a better return on investment.

Future efforts of the TerraScope TIN Team will be focused on the following goals:

- Develop alternative advanced query languages to support dynamic data composition in the spirit of the image cutters tools proposed in [18]

- Expand the diversity of our collection of radar images.

- Optimize the TIN client application.

- Implement a mechanism that allows prefetching to improve the performance of recursive queries.

- Implement classifications algorithms to create image clusters.

Bibliography

[1] M. Rodríguez and E. Coronado, SRE Search and Retrieval Engine of TerraScope Earth Science Information System, Proceedings of IASTED International Conference, Computer Science and Technology, Cancun, Mexico, 2003

[2] XML, From the Inside Out. **http://www.xml.com**

[3] C. Moock. ActionScript for Flash MX: The Definitive Guide, Second Edition. O'Reilly & Associates, Inc. December 2002.

[4] Quicklook Swath Browser: A web-based tool host by The Canada Centre for Remote Sensing. **http://quicklook.ccrs.nrcan.gc.ca/ql2/en?action=search**

[5] U.S. Geological Survey (USGS) Landsat 7 Image Viewer: **http://glovis.usgs.gov/**

[6] Visible Earth: a searchable directory of images, visualizations, and animations of the earth. **http://visibleearth.nasa.gov/.**

[7] T. Barclay, J. Gray, and D. Slutz, Microsoft TerraServer: A Spatial Data Warehouse, Proceedings of International Conference on Management of Data and Symposium on Principles of Database Systems, ACM SIGMOD, 2000.

[8] E. Tanin, F. Brabec, and H. Samet, Remote Access to large Spatial Databases, Proceedings of the tenth ACM international symposium on Advances in geographic information systems. McLean, Virginia, United States, 2002.

[9] C. Chan, T. Chan, and E. Prakash, A System for Geographical and Spatial Data Exploration on the Internet: Proceeding Series, ACM International Conference, Selected papers from the Pan-Sydney workshop on Visualization - Volume 2. Sydney, Australia 2002.

[10] E. Lim, Goh, D., et al., G-Portal: A Map-based Digital Library for Distributed Geospatial and Georeferenced Resources, Proceeding of the second ACM/IEEE-CS

joint conference, International Conference on Digital Libraries. Portland, Oregon, United States, 2002.

[11] V. Kumar, A. Bugacov, M. Coutinho, and R. Neches, Integrating Geographic Information Systems, Spatial Digital Libraries and Information Spaces for conducting Humanitarian Assistance and Disaster Relief Operations in Urban Environments, Proceedings of the seventh ACM international symposium on Advances in geographic information systems. Kansas City, Missouri, United States 1999.

[12] G. M. Lohman, et al., Remotely-Sensed Geophysical Databases: Experience and Implications for Generalized DBMS, Proceedings of the 1983 ACM SIGMOD International conference on Management of data, San Jose, California, United States, 1983.

[13] S. Chang, J.R. Smith, M. Beigi, and A. Benitez, Visual Information Retrieval from Large Distributed Online Repositories, Communications of the ACM Volume 40, Issue 12, 1997.

[14] G. Aggarwall, T. V. Ashwin, and S. Ghosal, An Image Retrieval System with Automatic Query Modification: Multimedia, IEEE Transactions Volume: 4, Iss.2 on page(s): 201-214. June 2002.

[15] T.A. Combs, and B. Bederson, Does Zooming Improve Image Browsing?, Proceedings of the fourth ACM Conference, International Conference on Digital Libraries, Berkeley, California, United States. 1999.

[16] P. Mouginis-Mark, L. Glaze, P. Flament. Virtually Hawaii, **http://satftp.soest.hawaii.edu/space/hawaii/**

[17] P. Baudisch, and R. Rosenholtz, Halo, a technique for Visualizing Off-Screen Locations, Proceedings of Conference on Human Factors and Computing Systems, Ft. Lauderdale, Florida, United States, 2003.

[18] M. Rodríguez, P. Rivera, A. Alvear, et al. Database Middleware System Supporting Remote Sensing Analysis over Distributed Sources, Proceedings of IASTED International Conference, Computer Science and Technology, Cancun, Mexico, 2003.