

---

**Centre for Environment and Development for the Arab Region & Europe**  
Electronic Conference on Sustainable Development Information Systems (SDIS)  
19-28 June, 2000

---

**Key Factors in Building Successful Environmental Information Systems in Developing Countries. Case Study: A Relational/Spatial Application for Environmental Monitoring Data.**

Carol Murray and Colin Daniel, ESSA Technologies Ltd.<sup>1</sup>

**ABSTRACT**

*ESSA Technologies Ltd. has been developing environmental information systems (EIS) in developing countries for the past decade. We focus on building custom desktop applications that meet each client's specific needs for managing and analysing their environmental data. The goal of all of our EIS projects is to help agencies develop their capacity to organize, analyze and present their data in a manner that allows them to make better decisions regarding environmental and resource management.*

*Our experience has shown us that three factors are key to building useful information system applications in developing countries: simplicity, flexibility, and technology transfer. The most successful EIS applications are those that are simple and user-friendly, allowing all users to operate them with confidence (including people who are "computer illiterate"). However, they must also be powerful enough to accommodate the more sophisticated user. That is one aspect of flexibility; another is the ability of the application to grow over time as the needs of the users evolve. The third factor pertains to the transfer of ownership of the application to the client after it is developed, so they have full control over application administration, maintenance, and evolution. This is a crucial aspect of capacity building, which is an integral part of our development philosophy.*

*This paper illustrates each of these three factors, using as an example an application we developed for managing environmental monitoring data in the Caribbean.*

---

**INTRODUCTION**

Sustainable development requires a good understanding of environmental condition; both the 'baseline' and how this condition changes as development progresses. Environmental information systems (EIS) can provide tremendous assistance in managing, and understanding, environmental data. Over the past 10 years we have developed a wide range of computerized EIS applications for clients in developing countries, ranging from large decision support systems with comprehensive environmental knowledge bases, to smaller applications for tracking environmental management information and analysing environmental data. Common challenges we encounter include:

---

<sup>1</sup> For more information, contact Carol Murray, R.P.Bio., M.Sc. ([cmurray@essa.com](mailto:cmurray@essa.com)) or Colin Daniel, M.Sc., P.Eng. ([cdaniel@essa.com](mailto:cdaniel@essa.com)). Head office: ESSA Technologies Ltd., 1765 W 8<sup>th</sup> Avenue, Suite 300, Vancouver, British Columbia, Canada, V6J 5C6. Website: [www.essa.com](http://www.essa.com).

- wide differences in the computer literacy of clients and prospective users;
- limited knowledge/experience with available software packages (e.g. using relational databases, but not taking full advantage of normalization and relational features; or knowing the mechanics of programming, but not employing the best programming design principles);
- hardware and software that may be several years out of date;
- slow (or in some cases non-existent) access to the internet; and
- low computer-to-staff ratios.

In addressing these challenges, we have come to recognize three factors that are crucial to building information system applications that truly benefit clients in developing countries: simplicity, flexibility, and technology transfer. (These are important factors for clients no matter where they are, but we focus here on issues that are particularly relevant to developing countries.)

These factors are each further explained below. The explanations are illustrated using an application that we developed for the Town and Country Planning Division, Ministry of Housing and Settlements, Government of the Republic of Trinidad and Tobago. This was a desktop application that combined relational database capability with mapping capability to allow clients to enter, store, display and query environmental monitoring data.

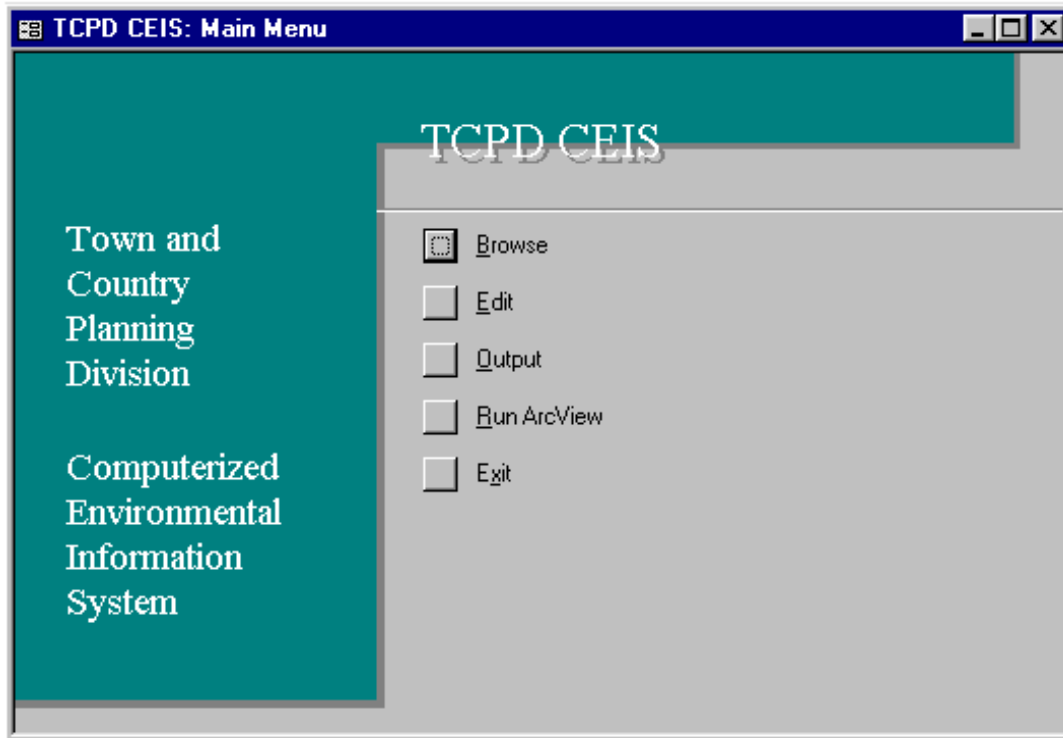
## **THREE KEY FACTORS**

### **1. Simplicity**

We have found that the most successful EIS applications in developing countries are those that are simple and user-friendly, allowing all users to operate them with confidence - including people who are 'computer illiterate'. We often find that most people in decision-making positions in client agencies are in the older demographic; and these are also the people who tend to be least comfortable using computers. This is not surprising, given the relatively recent invasion of computers into our daily lives and workplaces, but it does present a challenge: if the ultimate goal of an EIS is to improve decision-making, then the people making the decisions need access to it. That means that applications need to be simple enough for users who may not have had extensive formal computer training, and in fact may not be using computers much at all.

We have addressed this challenge in two ways. First, we add simple menu-driven user interfaces to our applications. We make sure that users always have a very simple and clear set of choices, so that it is very easy to navigate through the application.

In our example application, we designed a very simple Main Menu with five choices: browse the data (i.e. view only), edit the data, produce reports, work with maps, or exit. This menu is shown below. Each sub-menu also has a clear set of choices, providing a very simple 'road map' for users to follow, to help them navigate their way among the functions the application provides. It is almost impossible to get 'lost', and even if users know nothing about software or how the system actually works, they can follow this simple menu structure and find what they want.



Second, we try to streamline the application to best suit what users need. When programming customizations ourselves, this means providing the required functionality without a lot of ‘nifty’ but unnecessary bells and whistles. When selecting commercial applications to build on, this means choosing software that is of the appropriate size, scope and complexity for the users and their needs. For example, if we want to provide relational database functionality we will not choose Oracle if Microsoft® Access will suffice, and if we want to provide spatial capability we will not choose ARC/INFO when ArcView will provide what they need. This bias towards simplicity in application environment not only benefits users and administrators, but also results in lower-cost solutions.

## 2. Flexibility

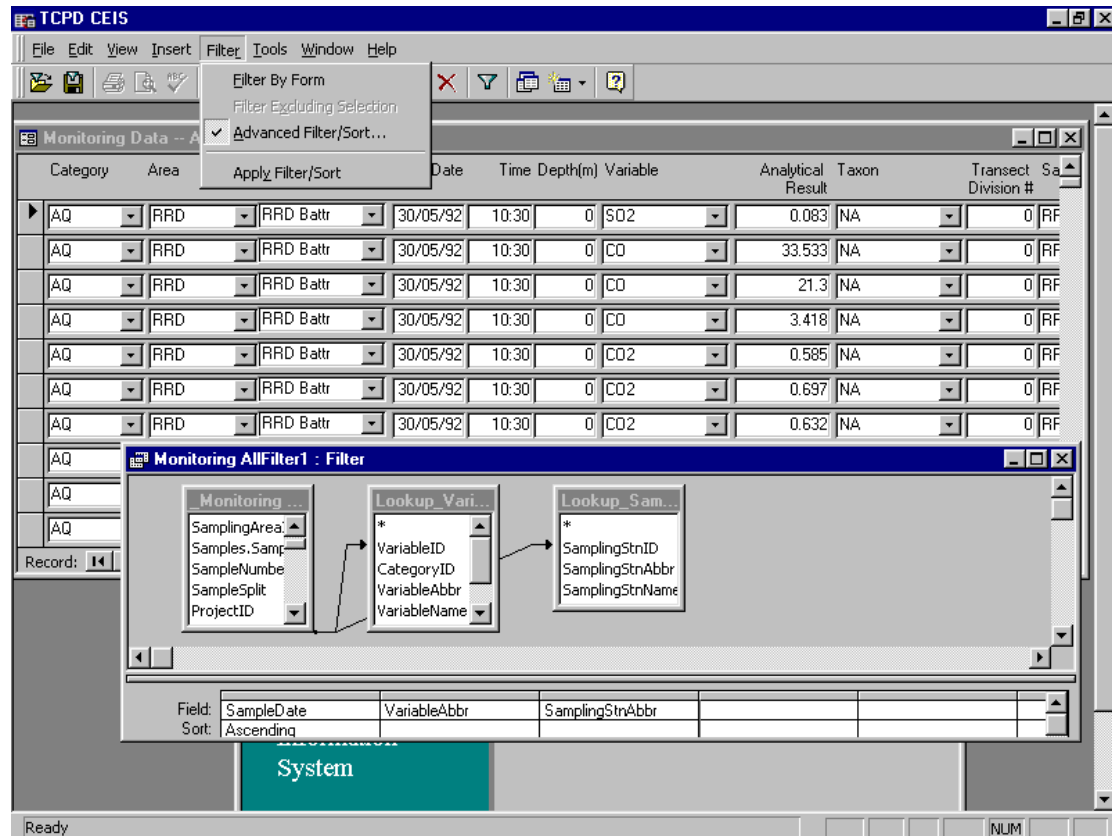
There are two aspects to flexibility: making sure the application accommodates users with a wide range of computer literacy, and making sure the application is designed to accommodate future upgrades/expansion.

The first aspect follows from our earlier discussion. While simplicity is important for the reasons given above, applications must also be powerful enough to accommodate the more sophisticated user. If it is too simple, knowledgeable computer users will become frustrated by the limits on what the application can do. We have found a way to make both types of users happy, by allowing users access to the software behind the simple, menu-driven interface.

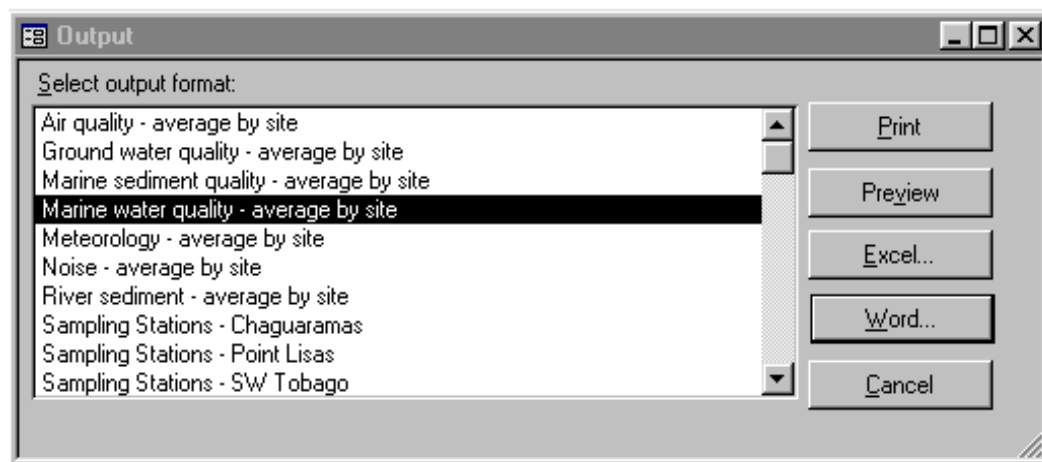
In our example, we designed the main interface within Microsoft® Access, because we wanted to employ a relational database for data management and querying. Users have the option of either following the simple menu structure we created, or going straight to built-in features within Access. Users knowledgeable in Access can employ additional functions of that software if they wish to work with the data more independently<sup>2</sup>. Our added menu structure is still there for users who want more guidance.

---

<sup>2</sup> Security features can be added to restrict this, if necessary to protect the data.

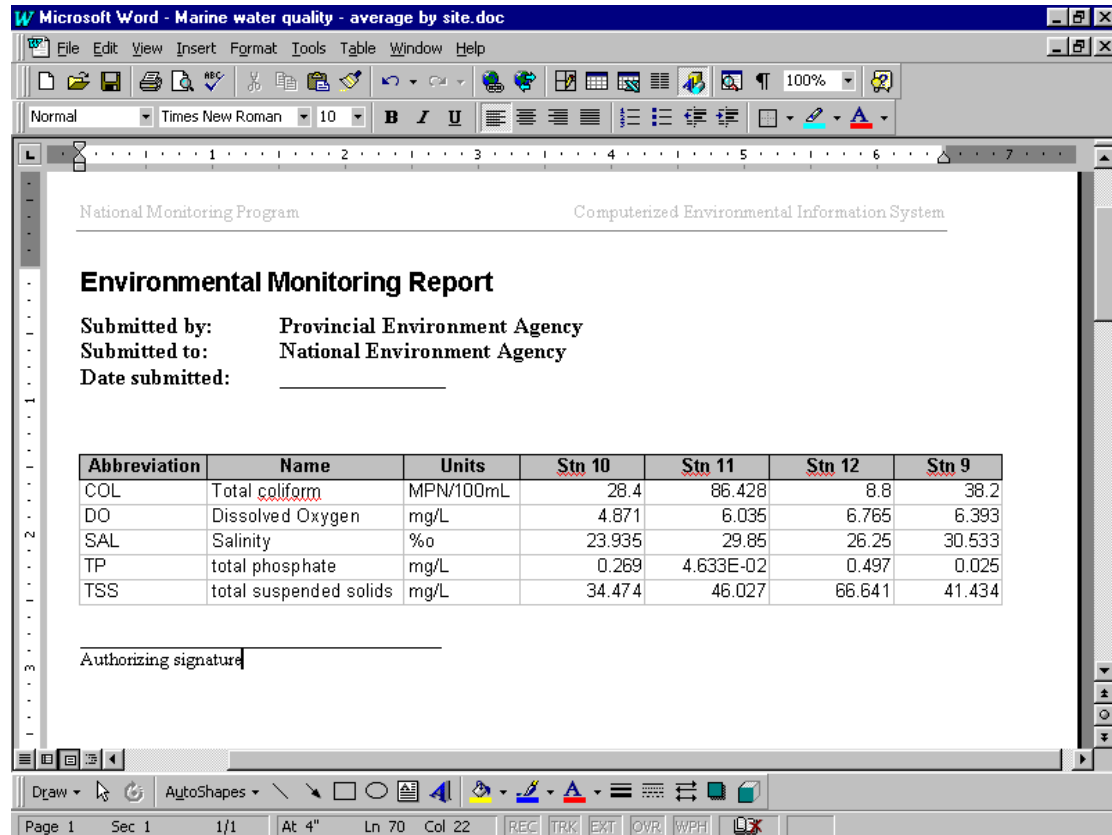


Beyond the optional menu structure, we have also provided flexibility in how users can get information out of the database. Using our example, we developed the application within the Microsoft® Office suite of software products. The Main Menu, Browse and Edit functions, and queries for synthesizing data into reports, all occur within Access. This was prudent because Access already has the capability to publish data to Word or Excel, affording users greater flexibility in how they report on the information in their database. We have expanded this capability to further allow users to send reports to customized Word templates (e.g. pre-formatted data reports). The screen below shows how users can select from a list of reports, and send the results to a Word file.

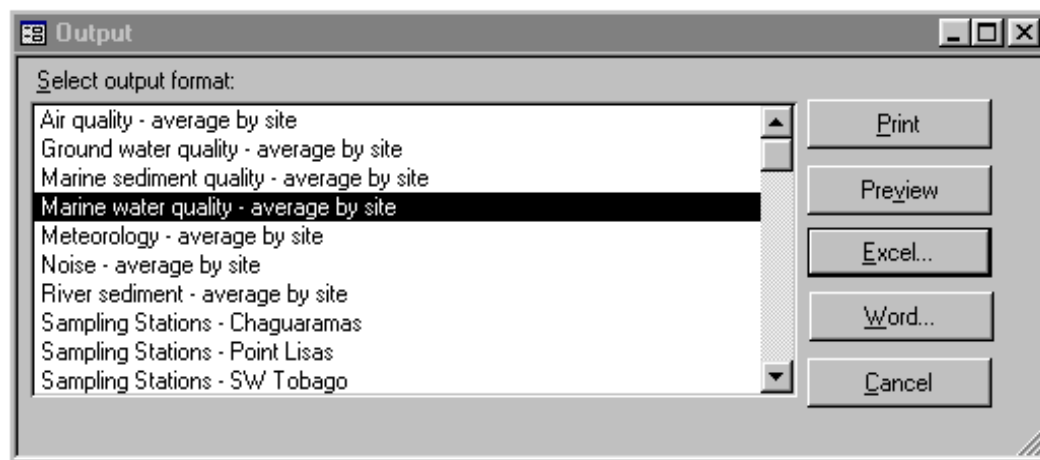


In addition to sending the results to a Word file, it also places these results automatically into a customized file template that could have headers, footers, headings, text, signature blocks; anything the user wants (an example is shown below). This automation can be a tremendous

timesaving tool for users who are required to produce regular data reports. This also provides the advantage of allowing users the flexibility to use a word processing environment they are likely familiar with to further format the file, or add more text, as they wish.

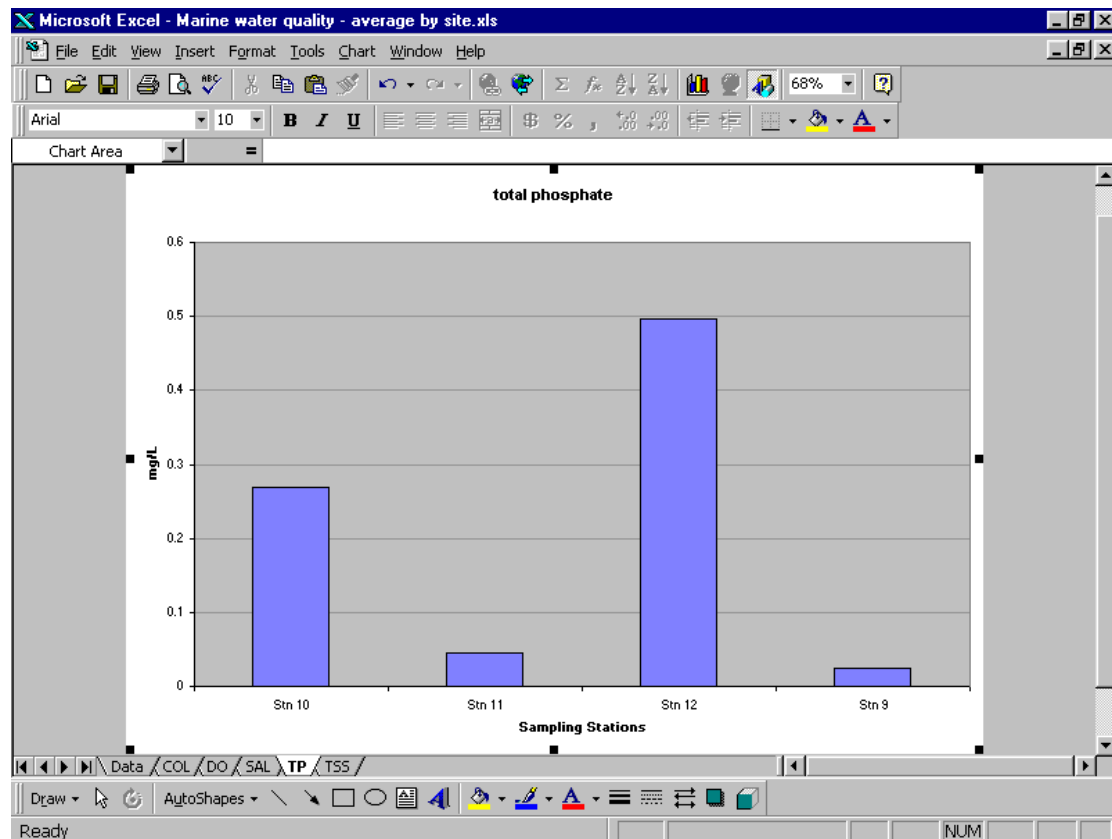


Similarly, users can select a report and send it directly to an Excel file. Not only will the data appear in an Excel spreadsheet, but we have also designed the application to automatically graph the data. This is shown in the following three screens. The advantage of this is that users who do not know Excel can still produce graphs from our application. Users knowledgeable in Excel are free to employ additional built-in features of Excel, if they wish to further analyse or graph the data.



Microsoft Excel - Marine water quality - average by site.xls

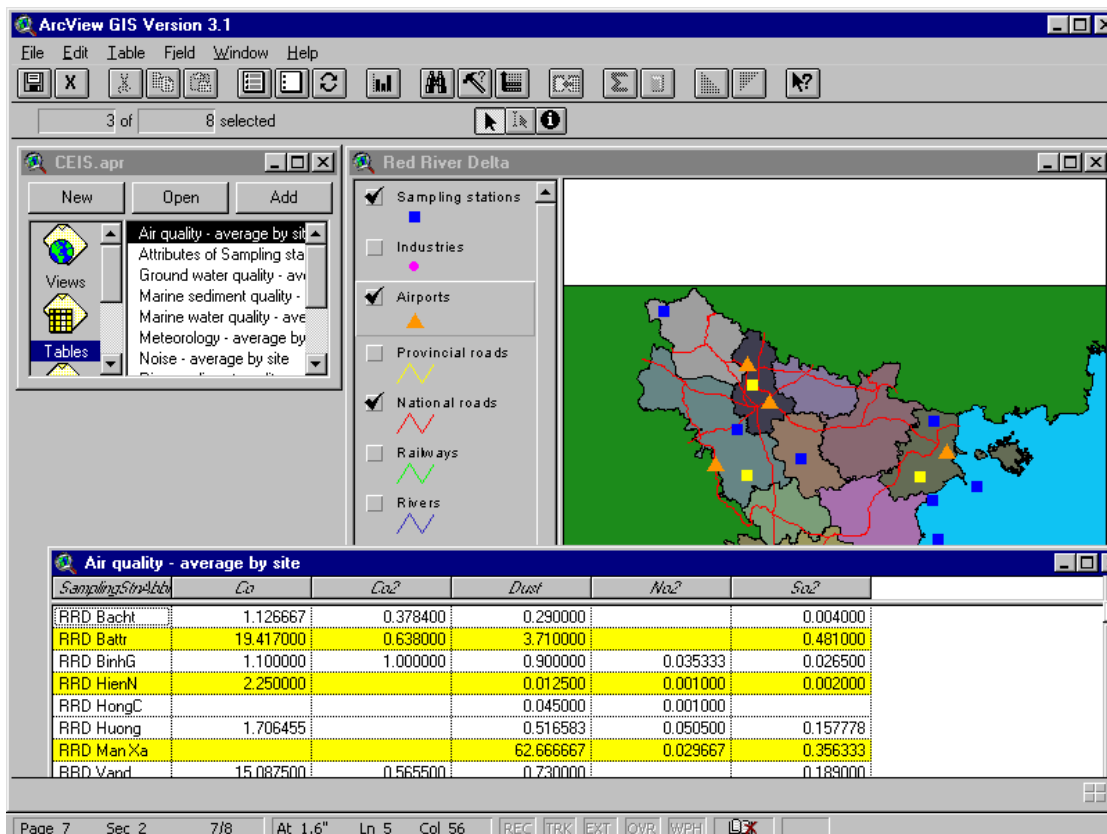
1	Abbreviation	Name	Units	Stn 10	Stn 11	Stn 12	Stn 9
2	COL	Total coliform	MPN/100mL	28.4	86.428	8.8	38.2
3	DO	Dissolved Oxygen	mg/L	4.871	6.035	6.765	6.393
4	SAL	Salinity	‰	23.935	29.85	26.25	30.533
5	TP	total phosphate	mg/L	0.269	0.046	0.497	0.025
6	TSS	total suspended solids	mg/L	34.474	46.027	66.641	41.434
7							
8							
9							
10							
11							
12							
13							
14							
15							
16							
17							
18							
19							
20							
21							
22							
23							
24							



The second aspect of flexibility is the ability of the application to grow over time as the needs of the users evolve. We accomplish this through an ‘open systems’ approach. The idea behind open systems is that while smaller systems must be built to solve localized problems, these

systems must also be built in such a way that they can be linked together to form a larger, enterprise-wide system. In this way present needs and capacity do not constrain the potential for the information system to expand and evolve in the future.

As our example has just illustrated, we select software environments that make it very easy to move data from one format to another (e.g. between Access and ArcView, or between Access and Excel). We employ standards such as ODBC, a programming interface that enables applications to access data within database management systems that use SQL (structured query language) as a data access standard. This also allows us to seamlessly interact with tables and queries from Access in ArcView. Our example application takes the reports previously seen in Access, links them in ArcView, and allows users to see specific monitoring sites on a map (shown below).

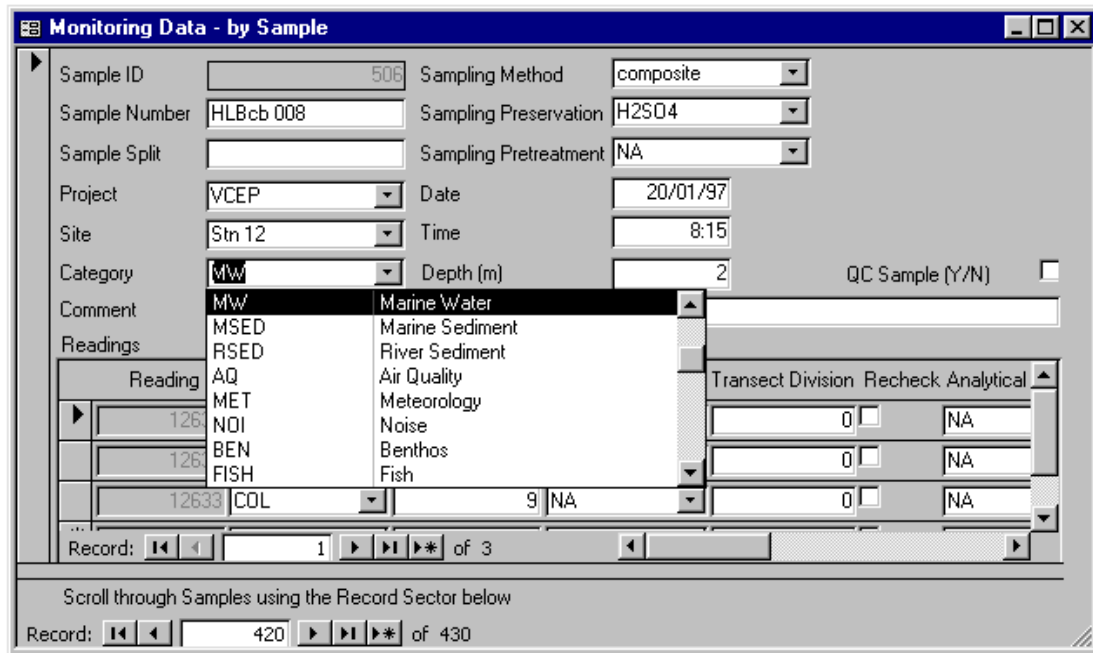


### 3. Technology Transfer

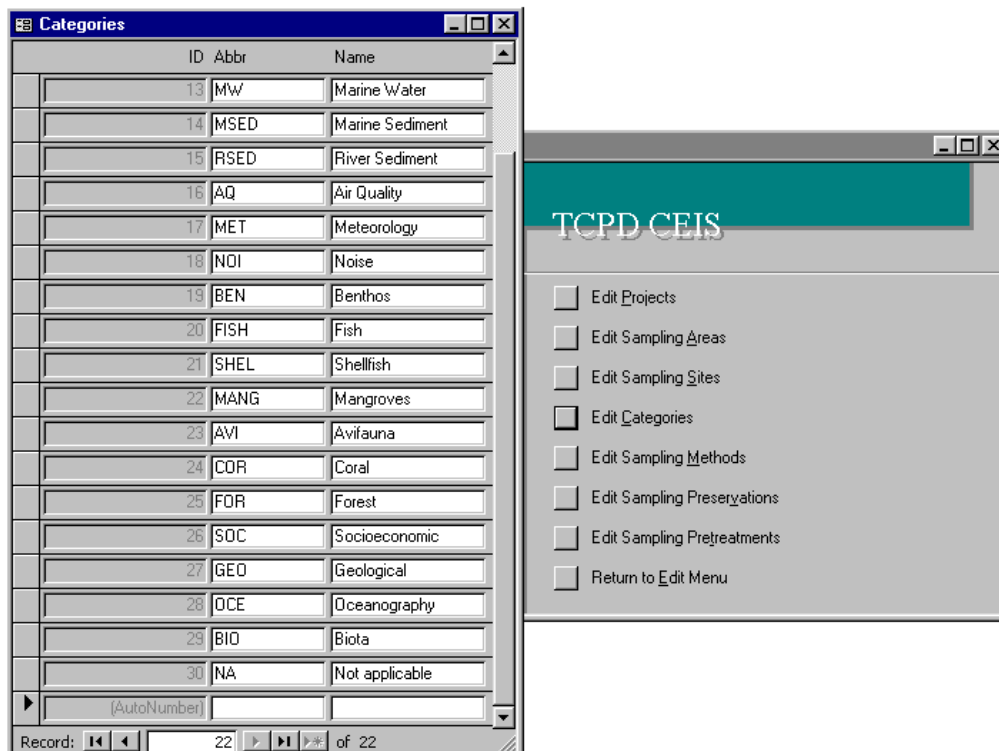
We follow a ‘capacity-building’ paradigm for all of our international projects. It is therefore important that we not only provide clients with useful computer applications, but also important that we empower the client to be self-sufficient in terms of management of the application once the project is complete. This is a crucial aspect of capacity development. It is very inefficient to maintain computer support for clients in distant parts of the world, so we make sure that clients do not need to rely on us for application administration, maintenance and expansion. We do this in two ways.

First, we build self-sufficiency into the application as much as possible. We do this by providing users with mechanisms through which they can alter much of the application’s content. For example, we use drop-down boxes in fields where content is finite and likely to be repeated. This is helpful for data entry, because users can select from a list of pre-defined items instead of having to repeatedly re-type the same text (which not only wastes time, but is vulnerable to error). The screen below shows an example of a drop-down box listing all the

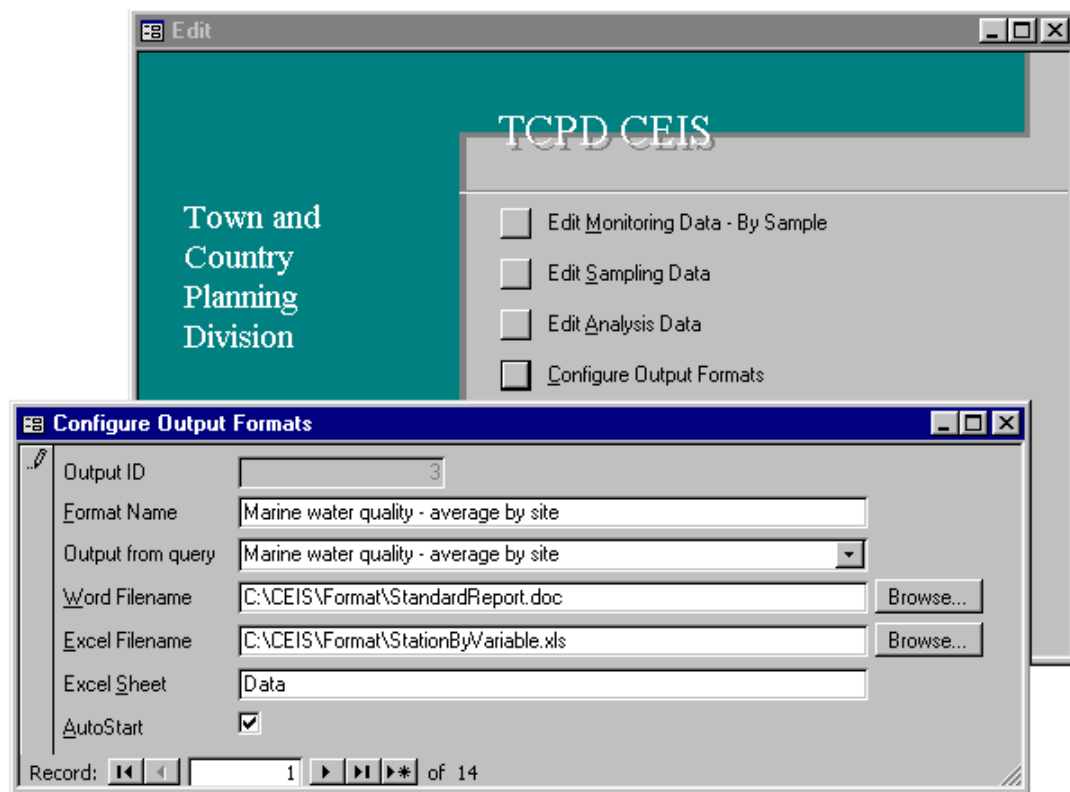
possible categories that the client might monitor. Users entering monitoring data need only select the appropriate category from the list when creating a record for a new sample.



However, it is quite possible that this list might need to be changed in the future (e.g. if the organization decides to monitor something new) so we add a provision for users to be able to edit the contents of this drop-down list. This is done through the Edit Menu, and the contents of all drop-down boxes can be revised or expanded. The following screen shows how the list of Categories can be edited.



Another element of self-sufficiency is providing users with a simple way of creating or modifying reports. While we can anticipate many of the types of syntheses that users might want from their data, in time there will likely be new types of reports that users will need. Our application includes a feature that allows users to create and add new reports, through the Edit Menu, as shown below. Users can define the new report, link it to a new query, and also define new Word and Excel templates. This removes their dependence on us after the project is over, if they decide they want new ways to report their data.



Finally, we not only train users in how to operate the application, but we also train the technical staff within the client organization as to how the application was constructed. This provides them with the knowledge necessary to make whatever revisions or expansions the users may require in the future as their information management needs change and grow. They can request our participation and assistance if they wish, but there are no technical or licensing restrictions requiring them do to so.

## SUMMARY

Understanding environmental condition is an important element of sustainable development. This usually requires a great deal of data, and environmental information systems can provide tremendous assistance in organizing, managing, understanding, and reporting environmental data.

We believe that simplicity, flexibility and technology transfer are key ingredients for developing successful environmental information systems in developing countries. We use two strategies to achieve simplicity in the information systems we develop: adding a simple menu-driven user interface to make it easily navigable by users unfamiliar with computers, and streamlining applications to provide only the necessary features and capabilities. We achieve flexibility by making sure the application also accommodates more sophisticated

computer users by allowing them greater independence and access to more functions, and by adopting an 'open systems approach' to ensuring the application can grow over time as the needs of the users evolve. Technology transfer, which is essential for capacity building, occurs through designing self-sufficiency into our applications, and through training of client staff on the technical aspects of the application. At the end of an EIS project, we leave the client not only with an application that meets their needs, but one over which they can take full management and future development control.