

NAVAL POSTGRADUATE SCHOOL
Monterey, California



THESIS

**THE USE OF INTELLIGENT REAL TIME
TECHNOLOGIES TO IMPLEMENT, MONITOR AND
FORECAST MEDICAL READINESS**

by

Michael Whitecar

June 2001

Thesis Advisor:
Associate Advisor:

John Osmundson
Gary Porter

Approved for public release; distribution is unlimited

20010807 022

REPORT DOCUMENTATION PAGE

Form Approved OMB No. 0704-0188

Public reporting burden for this collection of information is estimated to average 1 hour per response, including the time for reviewing instruction, searching existing data sources, gathering and maintaining the data needed, and completing and reviewing the collection of information. Send comments regarding this burden estimate or any other aspect of this collection of information, including suggestions for reducing this burden, to Washington headquarters Services, Directorate for Information Operations and Reports, 1215 Jefferson Davis Highway, Suite 1204, Arlington, VA 22202-4302, and to the Office of Management and Budget, Paperwork Reduction Project (0704-0188) Washington DC 20503.

1. AGENCY USE ONLY (Leave blank)		2. REPORT DATE June 2001	3. REPORT TYPE AND DATES COVERED Master's Thesis	
4. TITLE AND SUBTITLE: Title (Mix case letters) The Use of Intelligent Real Time Technologies to Implement, Monitor and Forecast Medical Readiness			5. FUNDING NUMBERS	
6. AUTHOR(S) Michael Whitecar				
7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES) Naval Postgraduate School Monterey, CA 93943-5000			8. PERFORMING ORGANIZATION REPORT NUMBER	
9. SPONSORING / MONITORING AGENCY NAME(S) AND ADDRESS(ES) N/A			10. SPONSORING / MONITORING AGENCY REPORT NUMBER	
11. SUPPLEMENTARY NOTES The views expressed in this thesis are those of the author and do not reflect the official policy or position of the Department of Defense or the U.S. Government.				
12a. DISTRIBUTION / AVAILABILITY STATEMENT Approved for public release; distribution is unlimited.			12b. DISTRIBUTION CODE	
<p>13. ABSTRACT (maximum 200 words)</p> <p>One of the various essentials of military readiness is the administering and continual execution of medical immunizing agents. Service members and civilian personnel working under contract are required to maintain a satisfactory level of medical readiness pursuant to the environment or platform they are assigned to. However, the guidance that oversees and classifies this "satisfactory" level is numerous in volume, broad in terms of definition, and difficult to interpret in terms of the methodology used to assign a specific rating.</p> <p>Because of the many interpretations of medical readiness, receiving a designated rating of C2 is considered to be acceptable in wartime interactions. Therefore, annual immunization requirements may be neglected throughout one's tenure while still engaging in a wartime environment with the risk of becoming exposed to influenza infection.</p> <p>The collection, tracking, and analyzing of medical readiness data is interpreted differently thereby prohibiting the facilitation and compilation of adequate information. Medical systems are developed using many different technologies that omit the ability to interface with one another, are in adaptive to change, and do not provide an interactive gateway to permit service members to become proactive and responsible for their own medical readiness state.</p>				
14. SUBJECT TERMS Medical, Readiness, Intelligent, Technologies, Web Application			15. NUMBER OF PAGES 308	
			16. PRICE CODE	
17. SECURITY CLASSIFICATION OF REPORT Unclassified	18. SECURITY CLASSIFICATION OF THIS PAGE Unclassified	19. SECURITY CLASSIFICATION OF ABSTRACT Unclassified	20. LIMITATION OF ABSTRACT UL	

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. 239-18

THIS PAGE INTENTIONALLY LEFT BLANK

Approved for public release; distribution is unlimited

**THE USE OF INTELLIGENT REAL TIME
TECHNOLOGIES TO IMPLEMENT, MONITOR AND
FORECAST MEDICAL READINESS**

Michael Whitecar
Lieutenant, Medical Service Corps, United States Navy
B.S., Park University, 1991

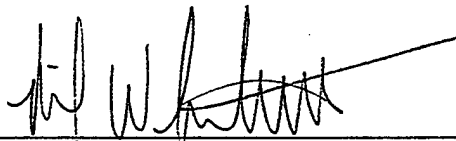
Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN INFORMATION TECHNOLOGY MANAGEMENT

from the

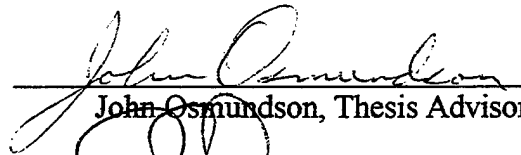
**NAVAL POSTGRADUATE SCHOOL
June 2001**

Author:

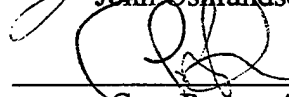


Michael Whitecar

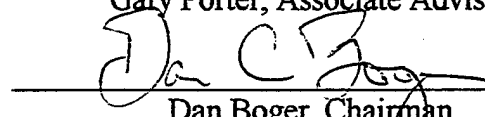
Approved by:



John Osmundson, Thesis Advisor



Gary Porter, Associate Advisor



Dan Boger, Chairman
Information Systems Academic Group

THIS PAGE INTENTIONALLY LEFT BLANK

ABSTRACT

One of the various essentials of military readiness is the administering and continual execution of medical immunizing agents. Service members and civilian personnel working under contract are required to maintain a satisfactory level of medical readiness pursuant to the environment or platform they are assigned to. However, the guidance that oversees and classifies this "satisfactory" level is numerous in volume, broad in terms of definition, and difficult to interpret in terms of the methodology used to assign a specific rating.

Because of the many interpretations of medical readiness, receiving a designated rating of C2 is considered to be acceptable in wartime interactions. Therefore, annual immunization requirements may be neglected throughout one's tenure while still engaging in a wartime environment with the risk of becoming exposed to influenza infection.

The collection, tracking, and analyzing of medical readiness data is interpreted differently thereby prohibiting the facilitation and compilation of adequate information. Medical systems are developed using many different technologies that omit the ability to interface with one another, are in adaptive to change, and do not provide an interactive gateway to permit service members to become proactive and responsible for their own medical readiness state.

THIS PAGE INTENTIONALLY LEFT BLANK

TABLE OF CONTENTS

I.	INTRODUCTION	1
	A. PURPOSE.....	1
	B. RESEARCH QUESTIONS.....	1
	C. THESIS OUTLINE	2
	D. EXPECTED BENEFITS OF THIS THESIS	3
II.	MEDICAL READINESS INFORMATION SYSTEMS.....	5
	A. A REVIEW OF MEDICAL READINESS INFORMATION SYSTEMS WITH THE DEPARTMENT OF THE NAVY (DON)	5
	1. Composite HealthCare System (CHCS)	5
	2. Standard Personnel Management System II (SPMS)	5
	3. Defense Medical Human Resource System (DMHRS).....	6
	4. Dental Management Information System (DENMIS).....	6
	5. Defense Occupational Health Readiness System (DOHRS)	6
	6. Immunization Tracking System (ITS).....	7
	7. Preventive Health Care Application (PHCA).....	7
	B. INTERACTION AMONG THE SYSTEMS.....	7
	C. THE EFFECTIVENESS OF REPORTING MEDICAL READINESS INFORMATION.....	9
III.	INTELLIGENT TECHNOLOGIES.....	13
	A. WHAT ARE INTELLIGENT TECHNOLOGIES?.....	13
	B. OVERVIEW OF AVAILABLE TECHNOLOGIES.....	15
	C. DEVELOPING INTELLIGENT SOFTWARE.....	19
	1. Rule-based Approach	19
	2. Knowledge Base Approach	19
	3. Machine Learning Approach.....	19
	D. INTEGRATING THE USE OF INTELLIGENT TECHNOLOGIES AND MEDICAL READINESS	20
IV.	DEVELOPING A MEDICAL READINESS WEB APPLICATION	23
	A. THE UNDERLYING ARCHITECTURE.....	23
	1. The Fundamentals	23
	a. Data Interfaces.....	24
	b. System Administration Layer	25
	c. System Layout	26
	d. Body Portal.....	26
	e. Desktop Interface.....	27
	f. User Interface	28
	2. Open Architecture	29
	3. Customization	30
	4. Navigation.....	31
	5. Personalization	32
	B. CREATING MEDICAL READINESS REQUIREMENTS.....	33
	C. THE PARADIGMS OF REPORTING TO MANAGEMENT	38
	D. FORECASTING MEDICAL READINESS.....	42

E.	PERSONALIZATION	44
F.	INTERFACING WITH OTHER MEDICAL SYSTEMS	46
G.	INCORPORATING INTELLIGENT TECHNOLOGIES IN THE TASKS OF SYSTEM ADMINISTRATION.....	47
	1. User Accounts.....	49
	2. Security Policies	49
	3. Error Control.....	50
H.	PROTECTING THE PRIVACY OF DATA.....	50
I.	NETWORK RESOURCES AND A DEPLOYMENT SCENARIO	52
	1. Network Resources	52
	2. Deployment Scenario.....	56
V.	CONCLUSIONS AND RECOMMENDATIONS	59
A.	CONCLUSIONS	59
B.	RECOMMENDATIONS	60
APPENDIX A.	SAMPLE MEDICAL SYSTEM SURVEY	63
APPENDIX B.	CONFIGURATION DATABASE DATA DICTIONARY	65
APPENDIX C.	USER DATABASE DATA DICTIONARY	73
APPENDIX D.	FILE STRUCTURE	79
APPENDIX E.	SOURCE CODE	85
	1. ADD-IN FILES	85
	2. OPEN-ARCHITECTURE DESIGN FILES	92
	3. SYSTEM ADMINISTRATION FILES.....	101
	4. PANEL COMPONENT FILES.....	157
	5. JAVASCRIPT FILES	161
	6. GET INFORMATION FILES.....	173
	7. USER FILES	188
	8. TABBED PAGES FILES	233
	9. SET INFORMATION FILES	247
	10. SUBROUTINE FILES	247
	11. INFRASTRUCTURE FILES	267
	LIST OF REFERENCES.....	285
	BIBLIOGRAPHY.....	287
	INITIAL DISTRIBUTION LIST	289

LIST OF FIGURES

Figure 1. Readiness Explorer System Application Layers	24
Figure 2. Example of HTTP and Pull Technology within Readiness Explorer.....	27
Figure 3. Sample User Interface Option.....	28
Figure 4. Open Architecture Design Process.....	30
Figure 5. Personalization Page Listing User Options.....	33
Figure 6. An example of Tabbed Profiles.....	34
Figure 7. Form used for creating profile field requirements	36
Figure 8. Designing Readiness Business Rules.....	37
Figure 9. SQL Report Form.....	40
Figure 10. Snapshot of Information Using Desktop Applications	41
Figure 11. Email Snapshot of Advanced Immunization Notification	42
Figure 12. Data Capture Strategies.....	45
Figure 13. Screen shot of creating an ODBC Interface.....	47

THIS PAGE INTENTIONALLY LEFT BLANK

LIST OF TABLES

Table 1. Medical System Interfaces.....	9
Table 2. Medical Readiness System Options	12
Table 3. List of Readiness Explorer Pre-Defined Reports	44
Table 4. Non-Readiness Explorer Traffic Utilization.....	53
Table 5. User Interaction and Readiness Explorer Response.....	54
Table 6. Search and Report Session and Readiness Explorer Response.....	55
Table 7. Performance Results.....	56

THIS PAGE INTENTIONALLY LEFT BLANK

ACKNOWLEDGMENTS

This thesis would not have been completed without the unconditional support of my wife and two daughters. It is the three of them that gave me the energy, inspiration, and dedication to my topic. I would also like to thank my primary and secondary readers for their time and effort, Professor John Osmundson and Professor Gary Porter, respectfully. The name of the web application that was developed within this thesis, *Readiness Explorer*, is a direct result of LCDR Richard Makarski's creativity. The success of *Readiness Explorer* would not be evident without the express support from LT John Kendricks and his staff at the Navy Administration Medical Unit, Monterey, California, and support from LT Tom Piner, US Naval Hospital Lemoore.

THIS PAGE INTENTIONALLY LEFT BLANK

I. INTRODUCTION

A. PURPOSE

The purpose of this research is to capitalize and integrate the use of intelligent real time technologies to effectively implement a system to monitor medical readiness resources, provide and forecast long-term readiness states, and promote the well being of our service members through proactive and interactive services.

B. RESEARCH QUESTIONS

This research thesis will address the following questions:

- How effective are the current medical readiness collection and reporting systems?
- Is leadership receiving the right data at the right time empowering them to make sound strategic decisions?
- Is there a sufficient and effective medical readiness model in place and can it adapt to change?
- Can the use of real time information technologies provide a stable and predictable medical readiness environment?
- What intelligent technologies are available to support these medical readiness requirements?
- What intelligent technologies are available to reduce the requirements of system administration and human interaction?
- Using intelligent technologies, can medical readiness actually be more effective?

C. THESIS OUTLINE

This thesis begins with an introduction to the medical systems within the Department of the Navy (DoN) that partially or in whole supports medical readiness. The mission of each system is reviewed; their various system functionalities; and interactions between each system are discussed followed by a short description of how medical readiness data is reported to senior management.

Next, a discussion of intelligent technologies and what is available to use today relative to medical readiness is presented, followed by brief summary on how to develop intelligent software. Finally, it shall be determined if using intelligent technologies integrated within the requirements of medical readiness is feasible.

With an underlying baseline of what medical systems are available and what intelligent technologies can do for medical readiness, the development of a web-based application is proposed to better meet the requirements by launching the fundamental architecture and how it will be designed. One of the key concepts of the web application design will reveal the paradigms of reporting data to management. A distinguishable feature of the web-application will be its ability to forecast medical readiness for entire commands or areas of operations in a real-time situation. In today's technological advances of taking advantage of the Internet, personalization will be another ingredient that can cater to successful medical readiness data capturing and reporting. System administration will also be discussed as it too plays an important role in developing a successful web application. Prior to the conclusion of this thesis, the privacy of medical data will be reviewed with options of integration presented.

The last section of this thesis will present conclusions by addressing the questions initially presented in this research. Recommendations will be made to provide Navy Medicine and possibly the Military Health System (MHS) with a cost-effective solution to ensuring that medical readiness is captured, reported, and forecasted properly.

D. EXPECTED BENEFITS OF THIS THESIS

Using intelligent technologies, the benefit of this thesis study will enable the development of a comprehensive, open architecture, and smart web-based application that will provide real time medical information and forecasting tools. Discovering, illuminating, and benefiting from these intelligent technologies will be combined into one application that because of its open architecture, will be able to communicate with other medical systems, thus creating an environment of system ingenuity and collaboration. The use of smart processes will learn the patterns of its users and adapt to their desires and requirements. In addition, automation and the censoring of events and processes will allow the removal of the mundane tasks of system administration by monitoring itself using well-defined business rules. Thus the research and the development of a smart web-based application will provide not only a model for perfected medical readiness, but also a standard in developing real time technical solutions using a practical graphical and multimedia interface, automated system learning capabilities, and compilation of information delivered in a multitude of desired and prescribed formats.

THIS PAGE INTENTIONALLY LEFT BLANK

II. MEDICAL READINESS INFORMATION SYSTEMS

A. A REVIEW OF MEDICAL READINESS INFORMATION SYSTEMS WITH THE DEPARTMENT OF THE NAVY (DON)

Currently there are over seven different varieties of medical or dental systems that either partially or in whole support some type of medical readiness. That is to say the systems may fulfill a subset of medical readiness requirements although not be in the original design. However, the systems have been able to provide just-in-time solutions to time critical instances of the Anthrax requirements and flu immunizations.

1. Composite HealthCare System (CHCS)

The baseline of these medical systems is the Composite Healthcare System (CHCS). Through the CHCS, medical data processing capabilities have been, and are continuing to be, developed, acquired, and deployed to assist health care providers and administrators with the management and delivery of quality care to all Department of Defense (DoD) health care recipients. The CHCS provides flexible medical data processing capabilities for DoD medical treatment facilities (MTFs). The integrated hardware and software architecture is fully scaleable to support the wide range of DoD medical facility sizes, from small, standalone facilities to large, regional medical centers. Currently in development and testing, CHCS II focuses on population health, force health protection and readiness reporting.

2. Standard Personnel Management System II (SPMS)

SPMS provides support to the functions of manpower, personnel, education and training, mobilization planning, and expense distribution for the Bureau of Medicine and Surgery (BUMED). SPMS II is an on-line menu-driven program that provides headquarters, support and field users with an interactive, automated capability to manage

operation, planning, programming, and budgeting in the functional areas of personnel, education and training, mobilization, and expense distribution. The Defense Medical Human Resources System (DMHRS) is expected to replace SPMS II the near future.

3. Defense Medical Human Resource System (DMHRS)

A program under development, is designed to provide a single solution to the tasks of scheduling, training, managing readiness, matching personnel with tasks, labor cost analysis and other human resources issues. Depending on user access levels, queries on skills, training and performance analysis of personnel trends will be available. DMHRS will interface with well over a dozen different systems without duplication of existing data. New efforts target a fully web-based application which should significantly reduce maintenance costs and complications.

4. Dental Management Information System (DENMIS)

DENMIS is an integrated solution for Navy dental healthcare facilities, providing comprehensive capability to manage dental-specific tasks. It generates reports, collects, and uploads dental workload data; manages patient recall and readiness; interacts with external personnel systems through a standard interface; calculates provider, department, clinic and command productivity as a unit-cost measure; and provides source data for the Medical Executive Information System (EIS) and other systems.

5. Defense Occupational Health Readiness System (DOHRS)

DOHRS is a target system to replace the Occupational Health Management Information System (OHMIS). DOHRS will provide a data processing system that will support Occupational Medicine, Industrial Hygiene, Hearing Conservation and Vision Conservation Programs. It will expand upon OHMIS by providing a dedicated communications resources with the MHSS infrastructure, support plug-and-play

replacement of components, support a wider variety of database applications, and provide increased security protection.

6. Immunization Tracking System (ITS)

A central repository located at the Naval Medical Information Management Center (NMIMC) that receives all SNAP Automated Medical System (SAMS) Anthrax and other immunization data. All MTFs, branch clinics and operational units functioning with SAMS, will transfer any immunization data to NMIMC. After NMIMC receives the data, the data will then be transmitted to DEERS via ITS.

7. Preventive Health Care Application (PHCA)

A tri-service, integrated computer system designed to automate the collection, documentation, and analysis of clinical preventive services, immunization tracking, and personnel and unit readiness information.

B. INTERACTION AMONG THE SYSTEMS

Each of the medical systems discussed above interface with one another using different types of connections, processes and workarounds.

The PHCA's immunization module interfaces with the Defense Enrollment Eligibility Reporting System (DEERS) in batch mode. It will download personnel immunization files available in DEERS and then update those files after new immunizations are documented. The interface connection is provided via dial-up from the PHCA server to DEERS. Although the Navy still requires Anthrax immunizations to be inputted to NMIMC through SAMS, all other immunizations can be documented in PHCA and directly uploaded to DEERS. The ITS is fed data from SAMS input. Furthermore, PHCA has no ability to link directly with SAMS or ITS. The owner of

PHCA, Space and Naval Warfare Systems (SPAWARS), has not displayed interest in creating a connection. Neither SPAWARS nor Navy Medicine have programmed for any such connections. A work around of a potential connection between ITS and PHCA is through DEERS. The planned two-way connectivity between DEERS and ITS will eventually create a window where PHCA to DEERS data will reach ITS via download from DEERS.

PHCA can also download basic demographic and treatment, allergy, prescription data from the local CHCS server. The download is triggered either by one at a time requests from a medical provider using PHCA, or by the overnight batch process of next day CHCS scheduled appointments. The future deployment of CHCS II at a Medical Treatment Facility (MTF) will include the migration of PHCA resident information into CHCS II. At that point PHCA will be turned off. This assumes that CHCS II will be approved by the MHS for worldwide deployment.

DOHRS is currently designing a connection with CHCS that may accommodate some instances of retrieving real time data while other non-immediate data will be retrieved in batch mode. It too has established an interface via batch mode with DEERS. However, DOHRS will actually retrieve the data from the Defense Manpower Data Center (DMDC) of which DEERS is a subset.

This year's effort for DHMRS is being referred to as personnel accounting support and labor accounting support application sets. There are many planned interfaces for the personnel accounting and labor accounting prototype is being worked.

In summary, the majority of these medical readiness-reporting systems support only a batch interface with other systems. In some instances where direct interfaces

cannot be established, workarounds are provided. Also it was noted that many of the interfaces require some form of human interaction to initiate a trigger of events such as uploading data. In the case of SAMS, a File Transfer Protocol (FTP) session must be established to send transaction data uploads to NMIMC for ITS updating. Depending on what data is required and by whom, a medical provider or management, retrieving such information can be cumbersome and be very time consuming. The following table lists the medical readiness systems in both the row and column headings with an "X" indicating that an interface is provided whether in real time or batch mode.

	PHCA	DEERS	CHCS	SAMS	ITS	DOHRS	DMHRS	SPMS
PHCA		X	X					
DEERS	X				X	X		
CHCS	X					X		
SAMS					X			
ITS		X		X				
DOHRS		X	X					
DMHRS								
SPMS								

Table 1. Medical System Interfaces

C. THE EFFECTIVENESS OF REPORTING MEDICAL READINESS INFORMATION

PHCA reporting capability has not worked as required. For example, there are canned reports that don't work well, or at all. The ad hoc reporting capability has never materialized. Software fixes and updates are in development to allow predefined canned reports to be accurately generated. Information within PHCA is not directly available to

service members in order to help them maintain their own medical readiness status nor is there an ability to modify PHCA for customizing readiness platforms. PHCA does not currently process information and pass to higher authority such as BUMED. Only until all fixes and updates are completed will such reporting commence. Once accomplished, PHCA will generate the reports required by BUMED-241 related to the Put Prevention into Practice (PPIP) Program. In relation to this, the Health Evaluation and Assessment Review (HEAR 2.0) tool imbedded in PHCA can provide a great deal of information about the health and wellness status of the MTF's population.

Parts of DOHRS reporting capabilities require trained users, i.e. Business Objects. Using Query-By-Examples (QBE) and the Structure Query Language (SQL) also requires trained personnel. DOHRS could possibly provide information to services members as all access is predicated on role-based security. DOHRS can be configured and maintained via a web-based front end. Additionally, web support is provided to higher authority to retrieve their required information. Again, this is also provided via protected role-based security.

DHMRS is another program that requires trained users for reporting. Many of the users of these capabilities are fairly junior personnel. While Business Objects is a robust tool and the MHS standard, it may be rather complicated for the target audience. Implementing SQL capabilities has not yet been determined. User-defined reports are not very likely to be included in this program. DHMRS expects its data to be pulled in from other interfaces. The local transactional data should be easily entered in real-time. New solution sets for DHMRS are targeting web-based development. As other programs are adapting, using appropriate access authorization to enter, view, and query data

regardless of location via a web application has the potential to be included. However, it is possible that some reports that are currently produced, printed and mailed may not be necessary once the full integration of web technologies are completed.

In spite of the many methods of reporting medical readiness, there is no supported standard for sending, receiving, storing or producing data. Too many workarounds are in place to make the systems interact with one another. Except for SPMS II, these systems do not provide its end users with modern web-based, graphical, and easy-to-use interfaces. Although, SPMS II supports this objective, the System Administrator must run the reports in advanced and then provide them to management in batch mode. Thus, the data is inherently delayed and perhaps little or no use.

The following table depicts the interaction and reporting capabilities among the systems. However, though the box may be marked "Yes" this doesn't mean that the method used is totally efficient. The Row Header consists of the system's ability to produce reports (REPORTS), establish a web interface (WEB), allow service member interaction (MBRS), provide the ability to customize the interface (CUSTOMIZE), provide reports to higher authority (HIGHER), produce adhoc reports (ADHOC), and provide a SQL interface (SQL).

	REPORTS	WEB	MBRS	CUSTOMIZE	HIGHER	ADHOC	SQL
PHCA	Yes	No	No	No	Yes	No	Yes
CHCS	Yes	No	No	No	Yes	Yes	No
SAMS	Yes	No	No	No	No	No	No
DHMRS	Yes	Yes	No	No	Yes	Yes	Yes
ITS	Yes	No	No	No	Yes	Yes	Yes
DOHRS	Yes	Yes	Yes	No	Yes	No	Yes
SPMS	Yes	Yes	No	No	Yes	Yes	No

Table 2. Medical Readiness System Options

III. INTELLIGENT TECHNOLOGIES

A. WHAT ARE INTELLIGENT TECHNOLOGIES?

Intelligent technologies have used since the industrial age and are prolific in the Information Age. Whether intelligent technologies are mechanical devices built into machines or computer software programs based on Artificial Intelligence (AI) the intelligence is explainable. Thus in this endower the emphasis in this thesis is on presenting intelligent technologies in the Information Age; more commonly referred to as the Internet or Digital Age.

Within this context such concepts, ideas, and technologies are commonly referred as intelligent agents. There are definitions of an intelligent agent. In its broadest sense, an agent is a merely a representation of a body of people or processes. At one of the spectrum, agents carry out fundamental tasks that may be repetitive or acquire little or no real knowledge. At the other end, agents can be as complex as technology permits. An agent can be anything from a thermostat to a human being! Taking this basic definition and applying it towards a more technical frame of reference an agent can be “anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors [Ref 1].” With continuing focus on technical terms relative to software, intelligent agents may be defined as “software entities that carry out some set of operations on behalf of a user or another program with some degree of independence or autonomy, and in so doing, employ some knowledge or representation of the user’s goals or desires [Ref 2].” With autonomy, the agent can take control of their own actions and initiate events independent of users or other agents. Thus an autonomous agent can react and perform tasks aligned with a decision process. This

concept may be refined as “a system situated within and a part of an environment that senses that environment and acts on it, over time, in pursuit of its own agenda and so as to effect what it senses in the future [Ref 3].” However, others would argue, “the program may be quite complex and be able to react to many different events, but the key is that the computer simply reacts, it does not act on its own [Ref 4].”

To place this into context and sum up an adequate definition of intelligent agents it can be determined that these programs carry out tasks on behalf of users. The unique feature, which distinguishes software agents from other programs, is the ability to automatically adapt their behavior to the conditions they encounter and to make decisions based on a set of business rules and criteria, without specific instruction from the user. Agents may also communicate with one another and with other programs or people to obtain information or enlist help [Ref 5].

Taking this definition and applying it to today's realm of capitalizing on information and the Internet, searching agents may be utilized with their autonomy to add another dimension to traditional search facilities. There are a variety of ways software agents may be used to our advantage. Such tasks may include searching the Internet for specific bits of information, online shopping and e-commerce entities, and booking traveling information relative to the user's preferences. Thus software agents are designed to enrich the user's net browsing experience by suggesting additional hyperlinks, based on previous user behavior; engage in transactions and negotiations on behalf of the user; monitor databases on the World Wide Web (WWW) and provide a regular updating service; monitor web site changes; compile a daily personalized newspaper; discover other people with common interests; check and prioritize email; or

make decisions based on content and act on them. This is already happening transparently as we surf the Internet.

Characteristics of agents may be separated into technological and social implications [Ref 6]. Beginning with the positive aspects of intelligent agents, one of the biggest benefits is that they have the potential to free humans from the tedious work of searching for information on the Internet and in databases. The intelligent agent is supposed to aid in the searching by filtering out the information and data, which is of little or no value with little human intervention. The concept is based on computers contributing substantially to the freedom of the human mind from tedious chores to enable higher levels of creativity. Confining oneself to an artificial world created by some human programmer(s) does limit human potential [Ref 7].

B. OVERVIEW OF AVAILABLE TECHNOLOGIES

The technologies available today to develop intelligent agents have evolved from a micro level of programming to a transparent and open-architecture development environment that is virtually accomplished by "point and click". Intelligent software is not new. Interests have been formulated for the past few centuries with a concentration in Artificial Intelligence (AI) focusing more on human thought processes and representing those processes via machines, such as computers or robots [Ref 8]. Some of the more common programming languages that focused on AI were LISP, Smalltalk, or Prolog.

Software agents that claim to be intelligent can practically be developed in a language that can accommodate the existence of sensors by capturing user interaction events, and providing the memory infrastructure to store user information and

preferences. Thus from this memory allocation, patterns may be established and business rules applied to enable the intelligent agent to react, trigger another set of events, or communicate with other agents.

Many of the languages today used for intelligent agents are geared towards the Internet, as the information is more acceptable to dynamic changes and modifications. Traditional client-server technologies use programming languages that compile code into machine language. Thus compilation increases the execution speed. Many programs developed for the Internet fall under the interpretive rule where each line of code is executed at run time. This leaves room for more error but quicker development and provides an interface to enable user interaction.

Programming languages that meet Internet potential include the popular Hypertext Markup Language (HTML), which is the core programming technology. More recently the development of Dynamic Hypertext Markup Language (DHTML) that includes the use of Cascading Style Sheets (CSS) has evolved. The uses of CSS allow the HTML author to separate presentation definitions from content in HTML documents. To provide more interaction with the end-user, an interpreted, object-oriented programming language was developed by the Netscape Corporation called JavaScript. It was developed mainly to add power and interactivity to web pages in order to provide better interaction with the user and cut down on network and server traffic [Ref 9]. Other varieties of this scripting language have been developed by Microsoft called VBScript and are a subset of the Visual Basic programming language. The introduction of these scripting languages provides increased programming flexibility and aids in network efficiencies. Another object-oriented programming language was developed by Sun

Microsystems called Java. Originally, Java was developed for real-time embedded software for consumer electronics. With the popularity of the Internet, the efforts of Java were redirected. The most notable outcomes of Java is the ability to be portable across many different platforms as its code is compiled as just-in-time byte code via systems that implement the Java Virtual Machine (JVM). Any one of the Internet based languages can behave as an intelligent agent as long as it has access to some type of storage media to receive user preferences and determine patterns. However, the use of these languages would require a great deal of programming.

Interaction with a web server and common programming languages such as Visual Basic and C++ have been provided by what is called CGI capabilities, or Common Gateway Interface. Thus before a web server would deliver the web page to a browser, an executable program would run externally via CGI that would process data calls or data manipulations. CGIs became a popular mechanism to run programs on the server but became very resource intensive. For every request by a web browser, the program would have to run in memory.

Another popular scripting language was developed called the Practical Extraction and Report Language or Perl. Borrowing heavily from the C programming language, Perl has become the language of choice for many input/output, file processing and management, process management, and system administration tasks.

In an attempt to minimize the use of computer resources on the client side via the web browser and bypass the CGI, other specific languages have been developed that can process information transparently and simultaneously before the web information is

delivered to the client's web browser. Two common languages are ColdFusion by Allaire and Active Server Pages (ASPs) by Microsoft.

ColdFusion is a rapid application development system for developers who want to create dynamic web applications and interactive Web sites. It provides an efficient way to integrate browser, server, and database technologies into powerful web applications [Ref 10]. ColdFusion applications are developed combining traditional HTML and a server-side programming language known as the ColdFusion Markup Language (CFML). ColdFusion provides many interfaces to popular Internet technologies such as the Simple Mail Transfer Protocol (SMTP), Post-Office Protocol Version 3 (POP3) for email services, the Hypertext Transport Protocol (HTTP) and File Transfer Protocol (FTP) for handling data transactions.

Microsoft's Active Server Page (ASP) is an open, compile-free application environment that combines HTML, scripts, and reusable ActiveX server components to create dynamic and powerful Web-based business solutions. ASPs enables server-side scripting for Microsoft's Internet Information Server (IIS) with native support for both VBScript and their Java-like Jscript, thus there is an extreme emphasis on using proprietary software.

To accommodate the transfer and sharing of information the Extensible Markup Language (XML) was developed as a recommended standard. XML is the universal format for structured documents and data on the Web. Along with the use of XML is a protocol called Web Distributed Data eXchange (WDDX), which is an XML-based technology that enables the exchange of complex data between Web programming languages. WDDX consists of a language-independent representation of data based on an

XML 1.0 Document Type Definition (DTD), and a set of modules for a wide variety of languages that use WDDX. WDDX can be used with HTTP, SMTP, POP3, FTP, and other Internet protocols that support transferring textual data.

More specifically developed for intelligent agents is software known as an Agent Communication Language (ACL) consisting of its vocabulary, an inner language called KIF (Knowledge Interchange Format) and an outer language called KQML (Knowledge Query and Manipulation Language). An ACL message is a KQML expression whose arguments are sentences in KIF formed from ACL's vocabulary [Ref 3]. Due to the nature and objectives of this thesis, ACL and KIF will not be furthered studied.

C. DEVELOPING INTELLIGENT SOFTWARE

There are many different methods to designing intelligent software. Whether it is in a formal or informal sense, a programmer with some basic business rules can apply them. Some of the more formal concepts are discussed below [Ref 3].

1. Rule-based Approach

The rule-based approach features a collection of user-programmed rules or processing information related to a particular task.

2. Knowledge Base Approach

This interface agent has domain-specific background with sufficient knowledge about the application and user to recognize plans and contribute to users' tasks.

3. Machine Learning Approach

This approach addresses problems encountered by the rule-based and knowledge-engineered approaches. This approach requires less initial work, and adapts over time. The agent acts as a personal assistant to cooperate with a user on a task, but makes allowances for user override.

Integrating these agents to exchange information and services with other programs independent of individual agents' internal data structures and algorithms requires a universal communication language to eliminate inconsistencies and arbitrary notational variations.

D. INTEGRATING THE USE OF INTELLIGENT TECHNOLOGIES AND MEDICAL READINESS

Integrating the use of intelligent technologies can provide an excellent opportunity to reengineer immunization processes, better inform Commanding Officers (CO) of their command's readiness status, and provide a proactive service to its service members.

An example is provided by an overseas command where military readiness tracking was a rigorous process that had continually reported a 98% deployable status. A deployable status is defined as total command personnel with a C1 or C2 readiness rating. A C2 rating is assigned when a requirement such as an annual HIV test is past due or does not exist at all. However, accepting a C2 rating as a deployable status allowed staff members to go without an annual HIV test far past the due date.

The method of reporting readiness status by the overseas MTF provided many inconsistencies revealing high levels of inaccuracy in the data collection process. Additionally, it is the responsibility of the service member to report the most recent immunization or test received to the Contingency department. This alone prohibited the ability to receive real time information.

The MTF was perplexed as to what really makes up a C1 readiness rating. What is medical readiness? What is a standard methodology of tracking the command and all

active duty members? What is a standard and convenient method of reporting the information to management?

To answer these questions, the command was able to turn to intelligent technologies with an applied baseline of medical requirements, and the ability to customize requirements based on their location. The command developed objectives and business rules that would support and determine the functionality of a web-based application. The objectives and business rules follow:

- In conjunction with the combination of higher authoritative directives, the command would develop a C1 requirement model.
- The responsibility to comply with the C1 model needs to be placed back into the hands of active duty members.
- The responsibility of tracking C1 requirements needs to be placed back into the department and division heads hands.
- The responsibility of updating the data needs to be placed back into the hands of the originators.
- Command leadership deserves accurate information reporting.

While aligned with the use of intelligent technologies, the MTF desired a continued utilization of their "available technology" via Internet email. This new addition could automatically notify the service members when their next HIV or Anthrax immunization is due and where to receive it. Additionally, the requirement allows other (COs) to retrieve their own information so they too can take a more proactive approach to becoming ready.

THIS PAGE INTENTIONALLY LEFT BLANK

IV. DEVELOPING A MEDICAL READINESS WEB APPLICATION

A. THE UNDERLYING ARCHITECTURE

The first essential requirement to developing a medical readiness web application was to determine its name. Through the collaboration and ideas from many potential users, Readiness Explorer was chosen as the application name. It will be referred to as such throughout the remainder of this study. ColdFusion by Allaire was chosen as the programming language because of its ease of development. Writing code in ColdFusion is very easy. Plus it easily integrates into HTML pages. The basic programming principles are very easy to grasp. Speed of development was also a key decision point and ColdFusion meets this requirement. Additionally, with ColdFusion, scalability is an issue that is virtually risk free. If there is ever a requirement to move a ColdFusion application to another operating system, it will migrate perfectly.

1. The Fundamentals

Readiness Explorer was designed with seven fundamental layers and a security reference monitor encompassing all. This reference monitor will ensure that users have access to only the web pages, data records and queries, and objects within Readiness Explorer that they are authorized to access. The Reference Monitor is located within the "application.cfm" page that is processed before any single ColdFusion Markup (CFM) is served to the user. The following figure shows each layer surrounded by the Reference Monitor with the interfaces listed at the bottom:

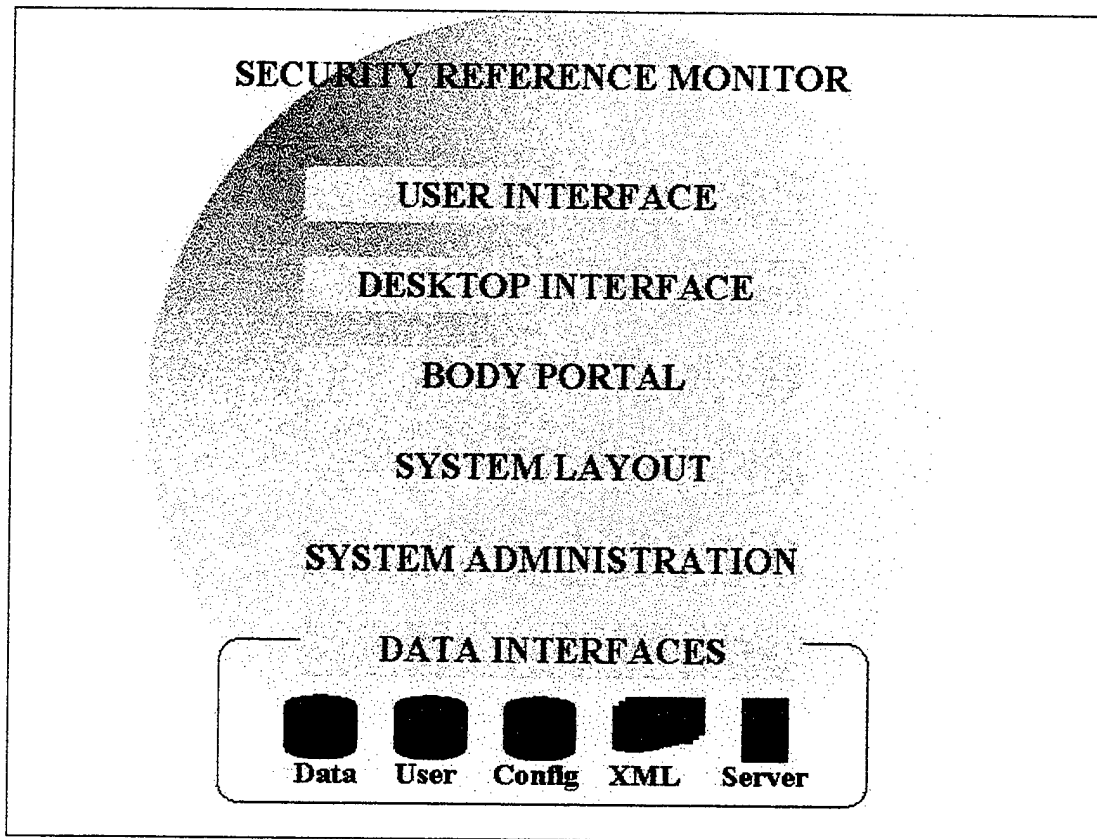


Figure 1. Readiness Explorer System Application Layers

a. Data Interfaces

The first layer consists of the Data Interfaces that stores, manages, and provides the necessary data to the user and application. There are three separate Microsoft Access databases designed to store and retrieve data using the Structure Query Language (SQL). One database called reCONFIG (see Appendix B) stores all of the information relative to the system itself that is configurable by the administrator or delegated user. This database also includes information relative to some of the underlying support for Readiness Explorer such as help summaries and link options within each page. The second database called reUSERS (see Appendix C) consists of all user information. In addition to basic user demographics, data such as the user's personal address books, calendar functions, and email is stored in numerous relational tables. The

last database called reDATA consists of command-defined tables that actually establish the business rules, medical readiness profiles, and reporting criteria. Because the reDATA database dynamically creates tables relative to a profile determination, a data dictionary is not provided.

The skeleton of Readiness Explorer is stored in special files on the server using XML format. The advantage of storing the infrastructure in XML was its portability. The three databases mentioned above could be stored in Microsoft's SQL Server or other larger scaled database systems. However, using XML in combination with ColdFusion allows the application to be stored on other operating systems such as Unix or Linux.

ColdFusion manages the final area of data storage on the server that holds Readiness Explorer. This data is never sent across the network. Thus ColdFusion and Readiness Explorer only use it specifically relative to the user. As each user logs into Readiness Explorer, ColdFusion establishes Session variables with an expiration that track the user's transactions and maintains demographic data as it may be needed during the use of Readiness Explorer. Using this methodology saves time on the server and reduces overload.

b. System Administration Layer

With the authorized access provided by the Reference Monitor, system administration is the next layer. It plays a critical role in using Readiness Explorer. When a system administrator logs into Readiness Explorer, authentication is immediately determined then the administrator is presented with additional options. Because Readiness Explorer can be customized to the environment of the command,

administrators can create their own pages and add page-specific help summaries. Therefore, if an administrator toggles “edit mode” most of the objects within Readiness Explorer will appear as form items so that they may be updated immediately. System Administrators have complete control of Readiness Explorer from any web browser.

c. System Layout

Once the Reference Monitor determines the users capabilities, authorizations, and preferences, the actual layout of Readiness Explorer will be displayed. This may include what “tabs” are provided and what “pull down” menus and other services they may have access to.

d. Body Portal

The body portal is where Readiness Explorer displays all of its information. This is an important feature of Readiness Explorer because of how the data is displayed. Readiness Explorer allows data to come from any device connected to the Internet. In order to maintain the “look and feel” of Readiness Explorer, ColdFusion uses the HTTP to retrieve the desired information and then pulls it into Readiness Explorer as if it were part of the native program.

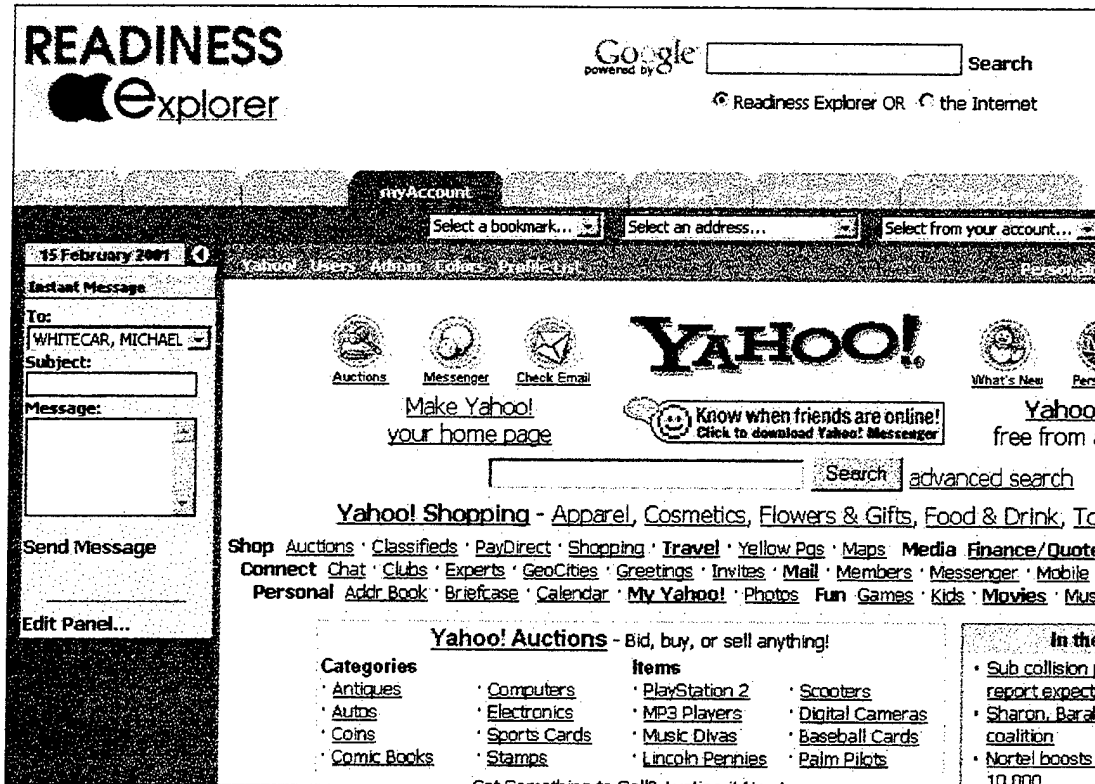


Figure 2. Example of HTTP and Pull Technology within Readiness Explorer

e. Desktop Interface

Everything that surrounds the “body portal” is the desktop interface itself that consists of seven separate parts. The first two are the left and right invisible frames at the top of Readiness Explorer. The left frame holds the official Readiness Explorer logo, and the right holds a search utility. The third frame consists of the tabs that support each page within Readiness Explorer, followed by the toolbar and panel areas. The footer is below the body portal. The desktop interface is completely customizable by the authoring command. However, some frames may be locked as desired to protect the initial or customized functionality of the program.

f. User Interface

The User Interface presents the functionality of Readiness Explorer to the user. Much emphasis has been placed on this interface. While using Readiness Explorer, the objective of the User Interface layer is to integrate all of the user's other web-based tasks into one universal and presentable application. With the ability to provide a portal of information, adding additional user functionality to make their experience more worthwhile is easily obtainable. Giving the user a sense of personal preferences and customization will ensure that the Readiness Explorer product is used properly. Some examples of providing users with additional functionality include receiving news updates, calendar operations, and email notification.

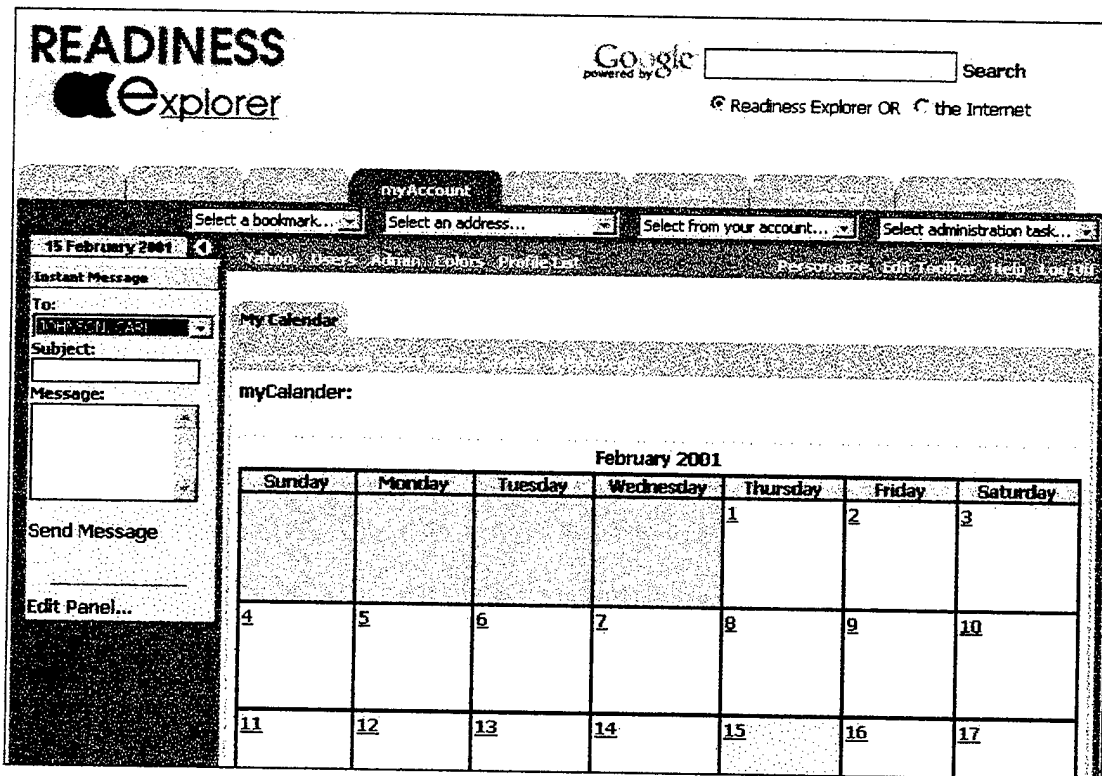


Figure 3. Sample User Interface Option

2. Open Architecture

In order to provide the command with an application that can be extended beyond its initial capabilities, developing Readiness Explorer under an open architecture was critical. Thus this architecture provides the command with the ability to add or remove any type of functionality. Adding additional functionality does not require the use of or knowing ColdFusion. In fact, any language may be used whether it is Perl or Microsoft's ASPs. The file is simply dropped into their respective frames. For example, the right frame at the top of each page, which initially contains a search utility, written in ColdFusion could become an intelligent search engine "bot" written in Perl.

The entire web application is designed using a specific protected administration page that provides options to use generated files, inline HTML code, or retrieve information directly from another web site via HTTP. This provides a web application that acts on behalf of a web portal in the sense that it can retrieve additional information, analyze it and display it to the user on one simple web page.

During the stage of designing the web application tabs, which contain each functional category in Readiness Explorer, the user is provided a "home" tab followed by Search, Tools, myAccount, Reports, Profiles, Resources, and Administration. Using the Security Reference Monitor, access of any of the tabs and the components within the page that the tab actually displays can be controlled. Once the tabs are created, clicking on them allows easy access to more information such as pointing to a file, web address, or by typing inline HTML code.

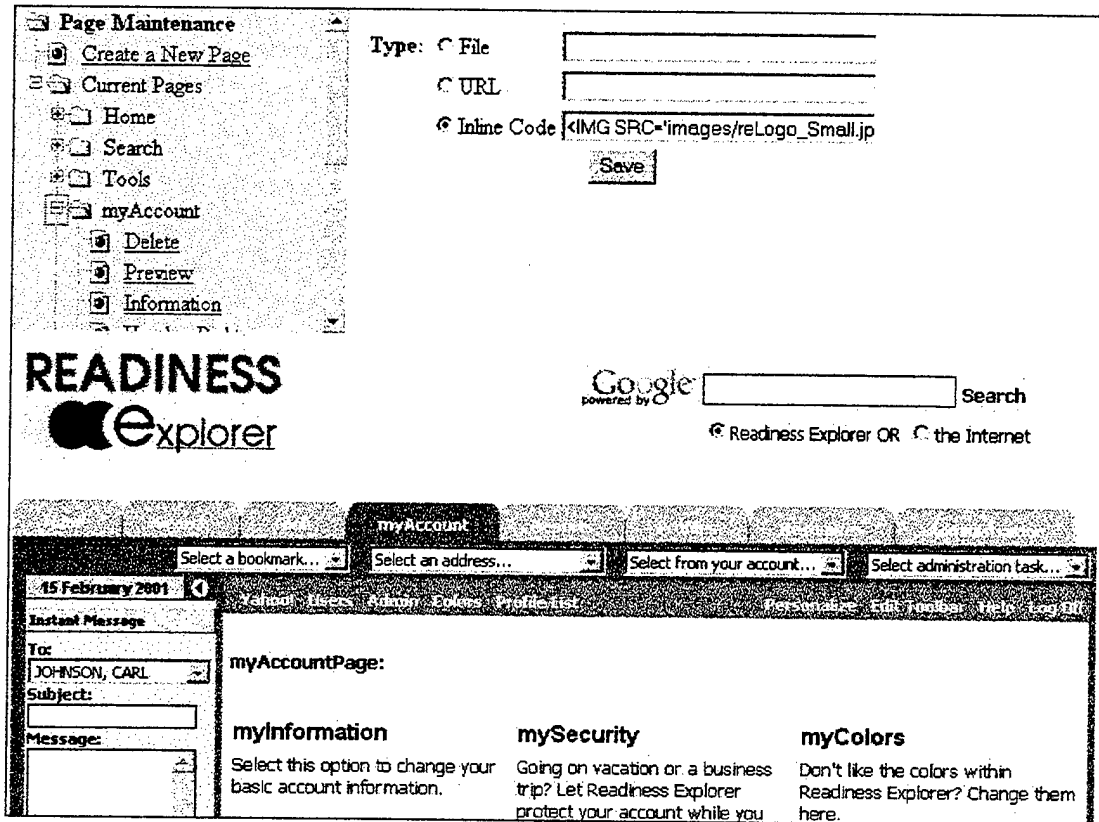


Figure 4. Open Architecture Design Process

3. Customization

To guarantee a successful web application that supports personalization, Readiness Explorer was developed with numerous user options. Users are immediately presented with the ability to personalize their desktop beginning with changing the colors. These colors are stored in XML format so that other users can share their colors schemes as well. Additional features such as customizing the toolbar and adding pull down menus to quickly get to information from anywhere within Readiness Explorer makes the application quick and efficient. A side panel is provided in Readiness Explorer specifically for user interaction. It consists of small-scale email notification and news updates. Users can easily choose what they want on their panel or they may hide it

completely. Panel components may also be placed in the Security Reference Monitor so that the System Administrator may only access them.

4. Navigation

A major design goal was to allow Readiness Explorer users access to information and services quickly and easily. For example, navigation within a web application plays a critical role in providing easy access to information. Because of the ease of HTML programming, adding this ability did not take much time. Adding additional navigation features is easy and inexpensive.

Within Readiness Explorer the “tabbed” interface provides an overall view of the application without having to drill down through layers of menus. Users are also able to create their own pull down menus below the tabs. These pull down menus serve as placeholders for common tasks such as bookmarks and routine transactions. This too is more efficient as users do not have to drill down through pages to reach common tasks. One of the pull down menus automatically gathers information relative to the tasks the user may use most often. This is accomplished by establishing the intelligent agents to sense the user’s common tasks and look for patterns. Below the user-defined pull down menu lies a toolbar that is split into two sections. One section includes application specific options such as getting help and logging off, while the other section is for the user’s use. Again, this gives the user the ability to quickly retrieve any type of information desired quickly. The “tabbed” names are also displayed as footer links at the bottom of each page.

Inside the body portal many of the tasks are grouped by another set of “tabbed” pages. The groups are also used in this fashion to prevent drill down or reloading of

pages for efficiency. Each set of tabs also has a toolbar with built in functions relative to the task being performed.

5. Personalization

Personalization is discussed in detail later in this paper. However, it is something that Internet technology requires. Personalization is the key to the successful implementation of any web application. As Readiness Explorer was developed considerable emphasis was placed on personalization to ensure that the user had a central familiar portal of information. Many options were added to include reminders that may help a user track a PPD test, email notification that could provide immediate subject notification, command news portals, quick access to folders of resourceful information, address books and bookmarks so that the user could quickly refer patients who may have inquiries relative to a particular immunization.

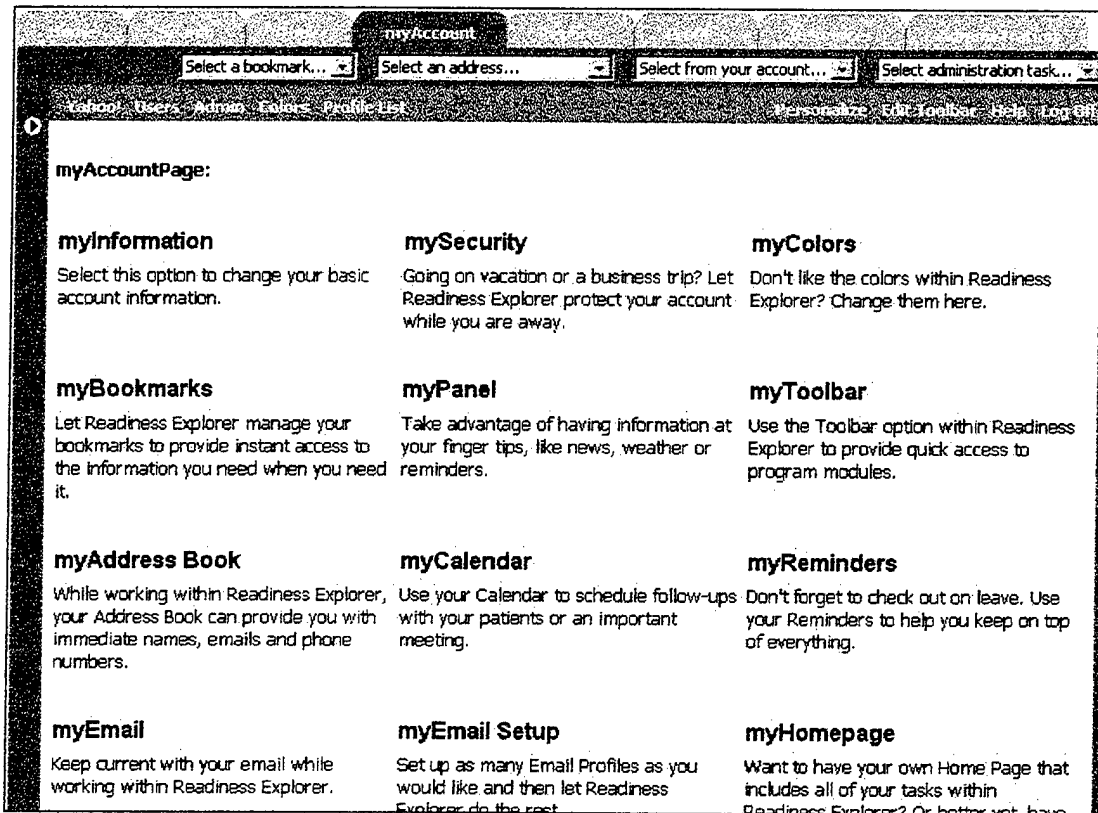


Figure 5. Personalization Page Listing User Options

B. CREATING MEDICAL READINESS REQUIREMENTS

Creating the medical readiness requirements is the foundation on which Readiness Explorer stands. A requirement is the basis for each platform and provides the information needed to present an accurate readiness status state. By providing an interface for establishing user defined readiness requirements, commands can add to a required baseline, established by higher authority, to meet their surrounding environment and cultures. For example, an overseas command may have readiness requirements that differ from a command within the United States. Therefore, the overseas command can simply add a new requirement and make it a part of their readiness needs. Separate platforms may be set up in order to prepare for special needs, for example a platform

customized for a Surgical Response Team (SRT) for an up coming deployment that may require additional medical readiness preparations readily accomplished.

As shown in the figure, each set or group of requirements is known as a profile and thus is laid out in a tabbed form that easily distinguishes each type of profile. Not all service members may require a certain profile based on their assigned templates. Each profile is made up of fields that have been preinstalled as a baseline plus those dynamically added by a command. Any command, depending on their mission, may easily add additional requirements that will not affect other commands.

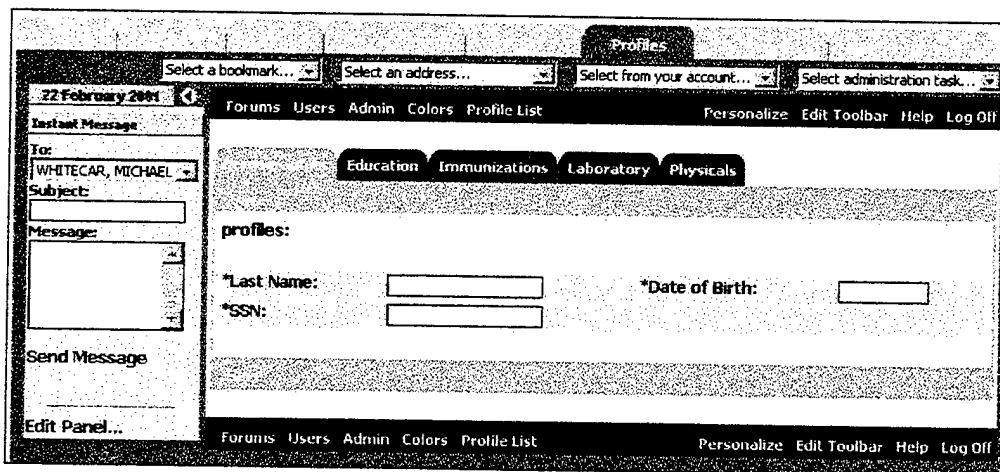


Figure 6. An example of Tabbed Profiles

Creating the profiles is accomplished first by identifying a group of requirements such as demographics, education, or immunizations. Completing a set of steps and determining the layout of the fields that will be contained within the profile is performed next. The demographic fields may be laid out so that only one record is displayed, whereas the immunization layout may be displayed in tabular format representing multiple encounters.

Once the new profile has been saved adding the required fields is the next step. As the fields are added, an actual database is being created and field elements added. Creating fields consists of assigning names, data types, and validation rules. Every strong data validation rules are used within Readiness Explorer to ensure that bad data is corrected before entry into the database. After creating the initial fields, identifying the readiness requirement follows. If the contents of the field fail a certain requirement, a C Status rating must be applied. This can be indicated from one through five options or none. For example, if an HIV field is created, failure to meet the requirements for an annual HIV test will result in a C2 rating followed by an indicator of color that determines how to visually show that the field has expired. Following the set of initial readiness requirements are few user selectable properties that provide help to the user. The design goal was to not only make it easy to add readiness requirements, but ensure that the end users are able to quickly enter data for each service member.

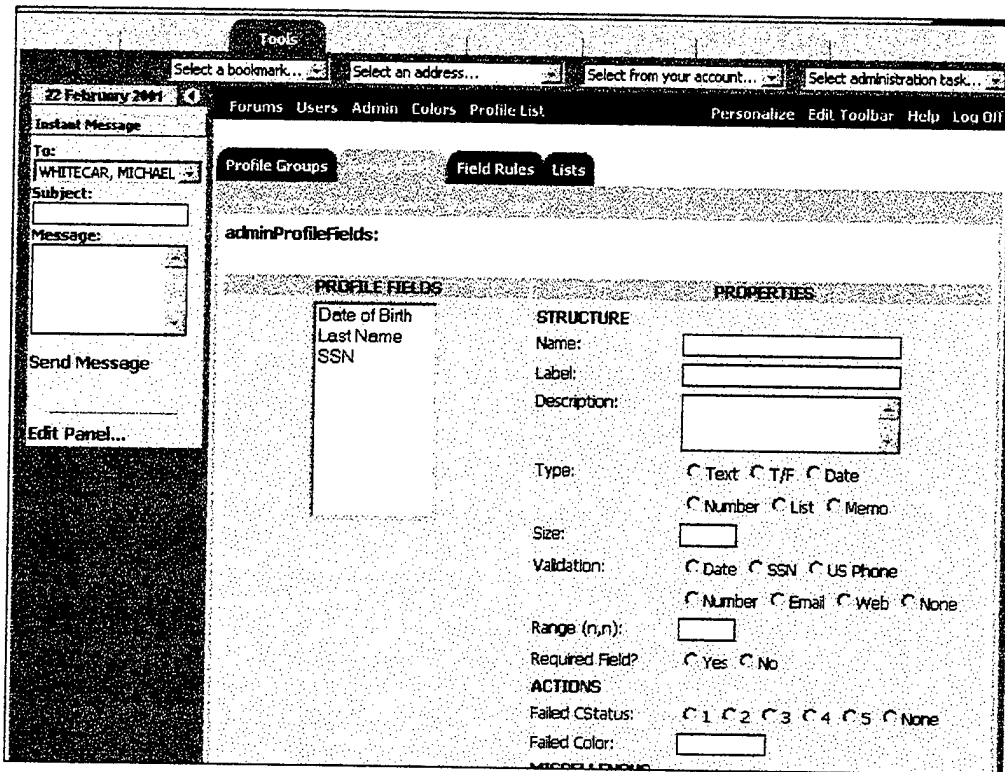


Figure 7. Form used for creating profile field requirements

During the next and most important phase Readiness Explorer now actually determines the medical readiness requirement by querying the database that was automatically created by user inputs. Similarly, the user need not know any formal query language to query the database and this can easily identify medical readiness requirements that can automatically be interpreted into code and SQL commands to ensure recognition by the database. The user simply adds rules by choosing from a list of fields and applying operators. Thus for an annual PPD, the user creates a new rule and calls it "Annual PPD", followed by selecting the PPD field. Then they select the "IS NOT EMPTY" operator and click on "Add Rule." This will add the first rule set and is transcribed by Readiness Explorer behind the scenes to an SQL instruction which reads (ppd IS NULL). This will trigger a failing event. The user may now continue to add

another rule definition by clicking on the “OR” condition followed by the “HAS LAPSED” operator and entering 365 in the Value field. Readiness Explorer will now interpret this to read (ppd IS NULL) OR (DateDiff(‘d’,ppd,Now()) >= 365) which means that if the PPD field is empty or if the date within the field has lapsed the past 365 days, then trigger an event that will assign the applicable C status rating, in this case C2. These business rules are stored in a database that will be used to apply to every profile that is displayed so a real time snapshot of the service member’s profile is available. These same rules will be applied during nightly batch processing to ensure that each record is accurately checked for the needed medical readiness requirements.

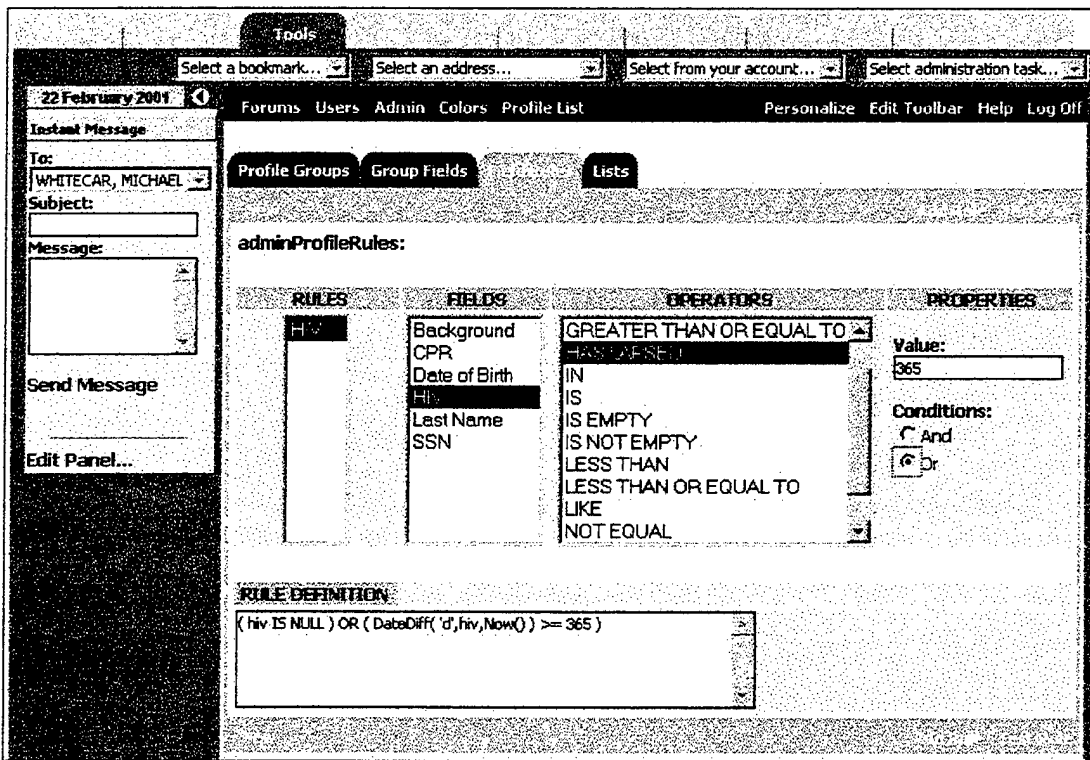


Figure 8. Designing Readiness Business Rules

C. THE PARADIGMS OF REPORTING TO MANAGEMENT

Information today is transferred, shuffled, managed, and interpreted all over the world using many different types of methodologies. Scenario A: When a senior manager or commanding officer in the case of the military needs information, too often the request is delegated through a long chain of individuals before the request is actually received by of the owner of the information. Typically by that time, the information request is either overdue or requires crisis management to responds quickly as described next. Scenario B: The information gathers, who often times are junior, receive the request late and rush to provide what they perceive to be proper information using the desired format. Many methods of producing information may include easy tasks as running a menu option or even a script, while at other times tasks may be more difficult such as running a programming script or even writing code to obtain the data. Once the data is collected it in turn is rushed back up through the chain of command to the senior member who requested it. The CO receives the information only to find out that from his perspective the information is incomplete and not in the desired format. "This wasn't exactly what I was looking for." And once again the request for the data goes back down the line and the cycle starts over again.

Advancements in technology have enabled us to better retrieve information. But the central issue is who owns the data. Who owns the process of retrieving the data? There are many products on the market that allow easier access to data and easier methods to create the reports than ever before. However, this still requires someone to be designated as that person who owns the data and knows hot to retrieve it. Many times requires specialized programming knowledge that costs a lot to training and maintaining

a certain level of staff expertise on board. There may be circumstances where the ability for users to pass on the information to others through On-The-Job Training (OJT), but this is not always the case.

Continued advancements have been made to add functionality to simple choices such as menu items that any user can simply select to activate a report. But this too requires a great deal of time from the administrator who is going to have to first ensure that the script is working and be able to add to the menu items. As shown from the initial scenario, once the information is provided, it may not be in desired format. Thus the administrator is now required to change the code for that individual and rerun the script. This is time consuming and costly.

Users at all levels of management and business processes must have easy, reliable access to information that aids them in their decision-making. Senior management cannot be expected to know SQL or understand how to use Business Objects, but they must be able to quickly retrieve information through a web browser, word processing file or spreadsheet. Thus using intelligent technologies, Readiness Explorer is able to accomplish this. Like other medical systems, Readiness Explorer provides the ability for adhoc reporting through the use of SQL as the figure shows below:

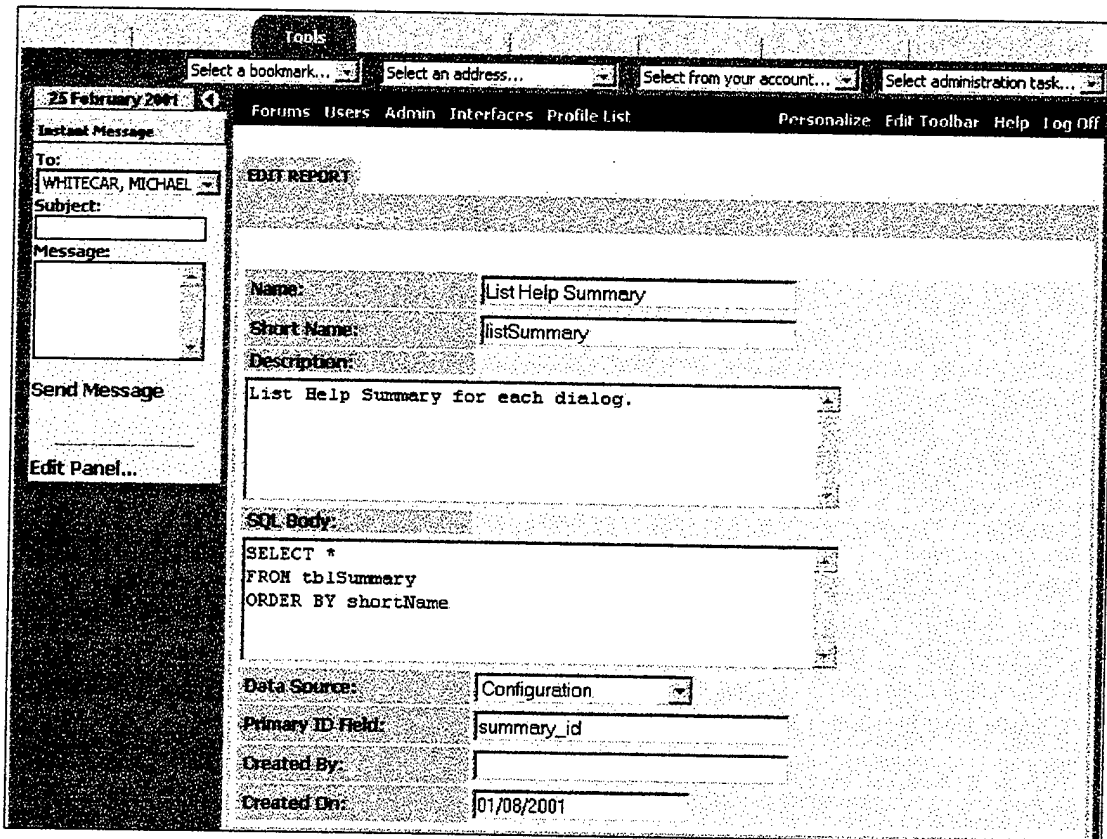


Figure 9. SQL Report Form

There are many other features that Readiness Explorer can provide. For example, reports may be created and saved for future retrieval. COs can have their reports emailed to them automatically on a weekly or monthly basis. The use of intelligent technologies today warrants the ability to send information quickly to other electrical devices such as pagers and cellular phones. If a CO deems necessary to be notified if the command's overall readiness status reaches 80%, then Readiness Explorer can automatically forward this information as soon as this threshold requirement is met. The figure below demonstrates the ability to bring up information within any desktop application that transparently retrieves information via the Internet from any web application into one simple document or spreadsheet.

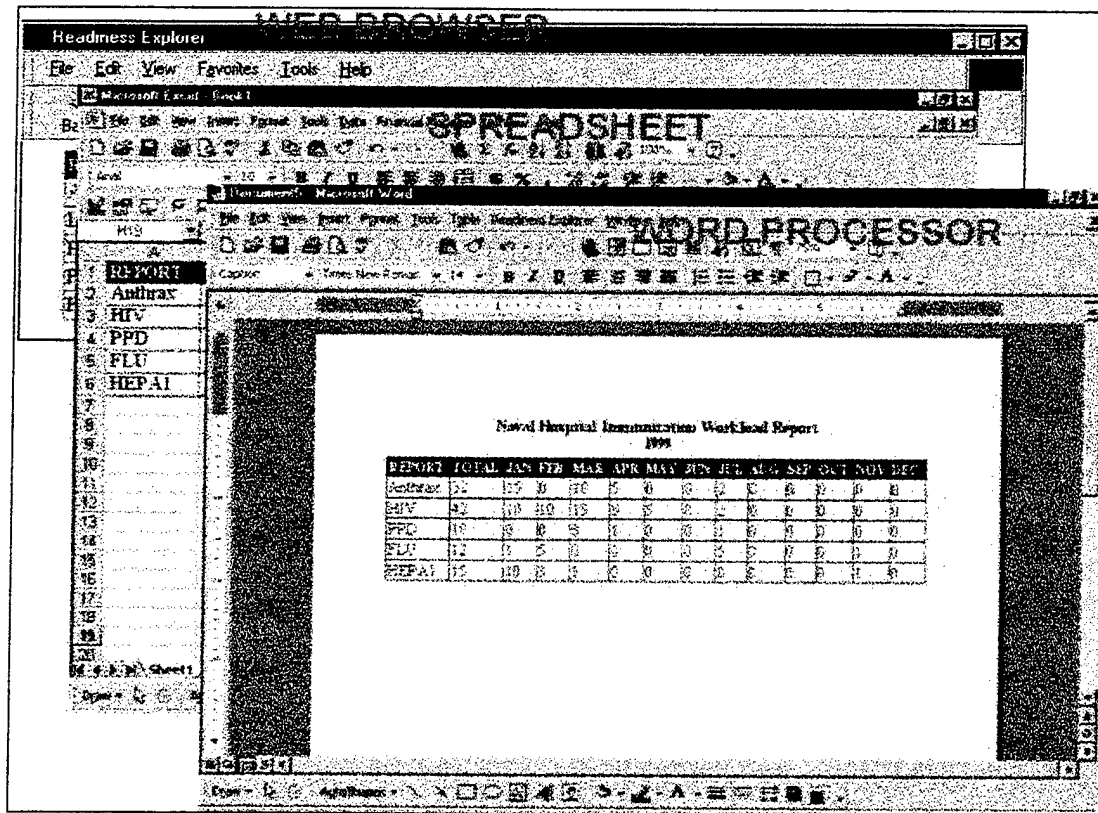


Figure 10. Snapshot of Information Using Desktop Applications

Along with sending information to management and process owners, Readiness Explorer continues to take advantage of intelligent agents by applying the command's business rules against service member data and reviews for compliance. Not only can Readiness Explorer check for compliance, but also review for proactive determination is possible. Thus, if a business rule fails, Readiness Explorer can automatically send an email message to the service member notifying the expiration of their current or future immunization. The following figures provides an example of the email message:

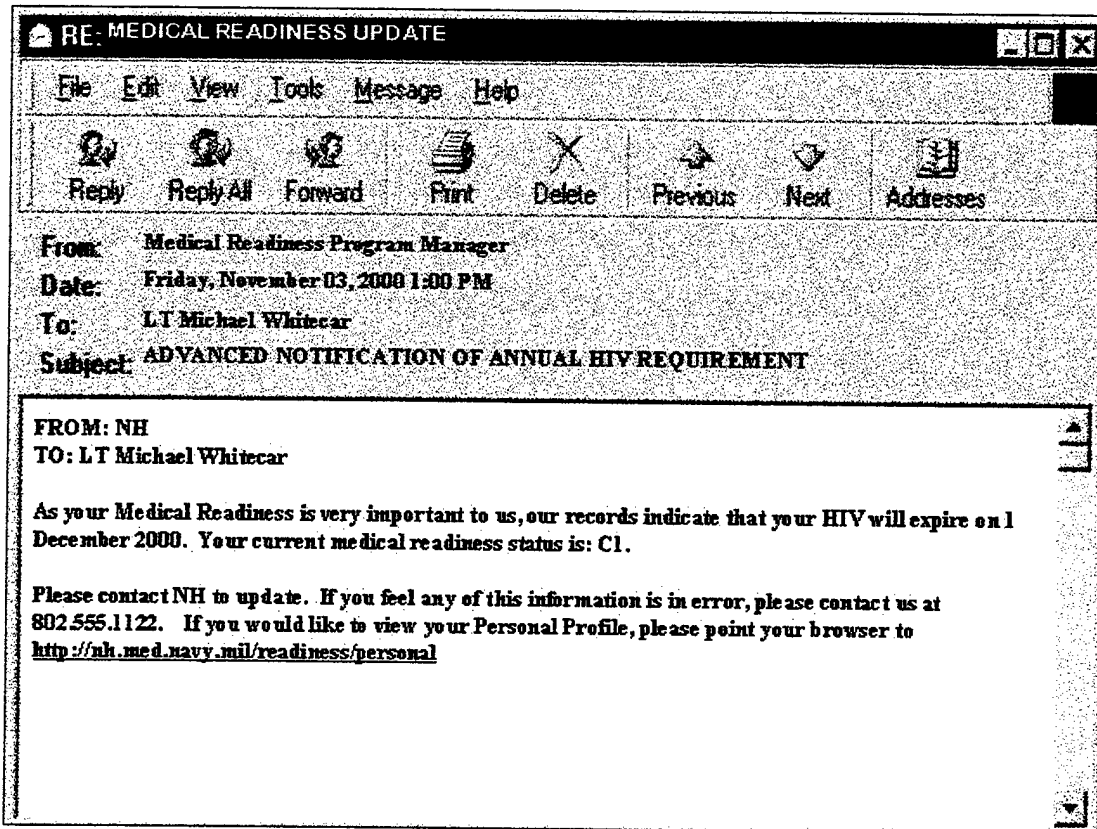


Figure 11. Email Snapshot of Advanced Immunization Notification

D. FORECASTING MEDICAL READINESS

Forecasting medical readiness through Readiness Explorer may be accomplished by continually analyzing the service member's data. The following reports are provided within Readiness Explorer and may be modified to better suit the user's needs:

REPORT	DEFINITION
Personnel Summary Report	A Personnel Summary Report provides a complete detailed summary of each profile within command and department selections made. Those requirements that are required to be C1 and are overdue are marked in the failed triggered color and outlined in brackets for easy printing. Those requirements that do not make a profile C1 but are helpful are marked in yellow/gold including outlining brackets.

Personnel Summary Forecast Report	The Personnel Summary Forecast Report provides the same information as the Personnel Summary Report except it allows a forecast between one and twelve months.
Anthrax Summary Report	The Anthrax Summary Reports provides a complete detailed summary where those personnel who are required the Anthrax Immunization. The report shows when the individual started the immunization and when the next one is due. In addition, the report outlines all future dates as well.
Data Count by Percentage	Use the Data Count by Percentage report to get an actual percentage of each C status category (1,2,3, or 4). If selections are merged together using the Merge Selections option and the Show View/Edit Options on Report option is selected, an additional column is added for viewing the actual profile that fall into each category. This method is an easy to use tool to evaluate who is C2, C3, or C4.
Overdue Requirements	The Overdue Requirements report provides a complete list of those profiles that are over due for a specific requirement. For example, selecting HIV in the pull down list will provide all profiles within the above selection that have not met the HIV requirements.
Requirements Forecast Report	The Requirements Forecast Report allows you to forecast up to twelve months ahead what immunizations or tests will be required. This report is the complement of the Overdue Requirements report. However, this report serves as an excellent tool to determine future requirements and enable cost savings from over ordering of immunizations for example.
Health Record Review Report by Birth Date	In addition to being able to provide annual Health Record Reviews (HRR) via the Overdue and Forecast reports above, use this report to provide the same verification but via member's Birth Date. For example, the Patient Administration department may verify all health records for the month of June.
Blood Type Roster	Use the Blood Type Roster to provide a convenient list of personnel with a specific blood type. This is very helpful in situations where blood donors are required within a certain type.

Platform Assignment Roster	The Platform Assignment Report provides a complete list of all platforms within the command. This report identifies those personnel who are not in a C1 compliant status. The information provided ensures that those personnel on one of the platforms are in compliance as they will be the first to possibly be deployed.
----------------------------	------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Table 3. List of Readiness Explorer Pre-Defined Reports

E. PERSONALIZATION

The use of intelligent agents not only provides the ability to create a robust web-based application that can refine the medical readiness process, but it also caters to the concept of personalization.

Personalization is not a type of technology but a process that is part of this overall web strategy. Using intelligent agents, monitoring user behaviors and transactions may be collected, analyzed and returned to the user in a customizable form. Personalization can be defined as a technique used to better understand the users, and as a process of ongoing modification to underlying systems to better address the preference of the end users [Ref 11]. Personalization may offer the end users more convenience, greater value and higher levels of customer service and satisfaction. By combining web technology and intelligent agents with the fundamentals of personalization, commands can gain a better understanding of their user preferences.

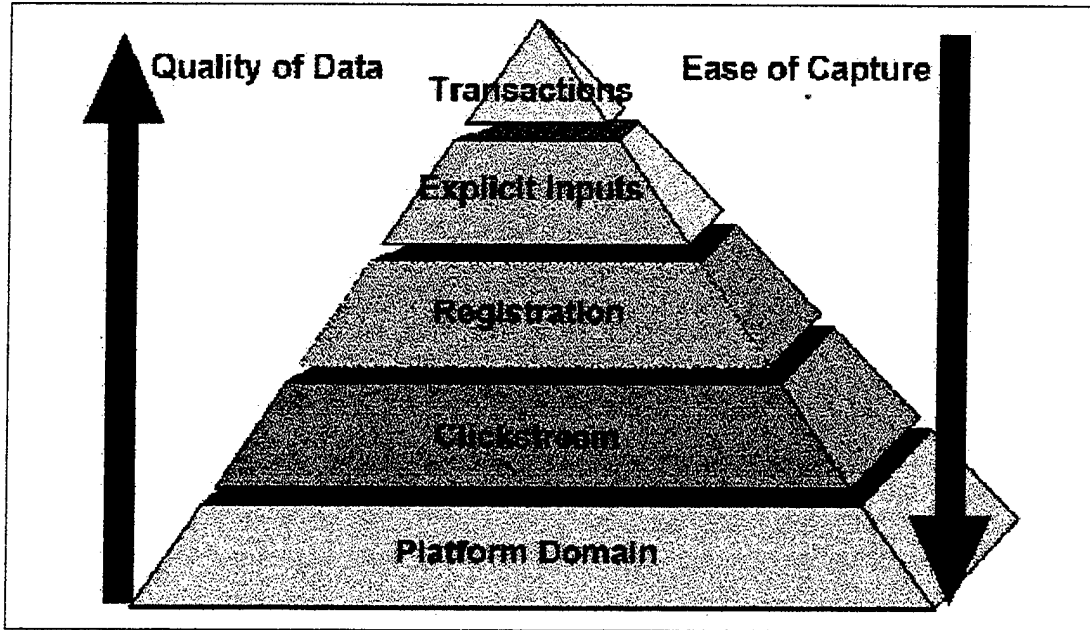


Figure 12. Data Capture Strategies

Data capturing strategies are a relevant and significant aspect of personalization. Jupiter Communications has developed a pyramid that depicts levels of data collection and how the data is collected [Ref 12].

The information at the bottom of the pyramid is considered the broadest variety. It is not very useful but it is very easy to capture. Moving up the pyramid, each of these types of information requires a more intensive method of personalization.

At the Platform Domain layer informs the command things like what browser their users are using, or what domain the user is coming from. This information can be obtained using relatively unsophisticated client-side sniffing technologies such as JavaScript or ColdFusion.

The Clickstream layer gives a command insight into a user's behavior within the web application. It provides information about where the user came from and where they

go within the application. This information can be used to evaluate the navigation strategy and the paths used to reach certain transactions. The quality of clickstream data will vary depending on the quality of the application analysis system used.

Registration can be very basic such as collecting the user's name and email address or more complex. In the case of Readiness Explorer, information about the user's command, Project Rotation Date (PRD), and how to contact them are requested during registration.

The layer of Explicit Inputs include when a user inputs words into a search query, response to an online survey, or questionnaire. If a user selects preferences within the application such as color schemes, is considered an explicit input. This information allows the command to connect one user with their personal type of data or services.

At the top of the pyramid is the Transactions layer. Transactions yield rich user data, enabling the command to capture specific patterns and trends

F. INTERFACING WITH OTHER MEDICAL SYSTEMS

A concept for interfacing with other medical systems is known as the Component Object Model (COM) and Distributed Component Object Model (DCOM) by Microsoft. Each small piece of DCOM code, usually written in Visual Basic or C++, is placed on the individual server. Data manipulation that is requested by Readiness Explorer will take place using the protocol Open Database Connectivity (ODBC). Data may also be provided by the means of flat files. The methods used to transfer the data may include HTTP, Telnet, or FTP. If business practices permit, data may be written back to the providing server as well as read only.

Each interface is created dynamically by the site that will begin by determining the servers, the data dictionary, and transfer protocol by an easy to use interface. In its simplest form, when a user retrieves a profile, the underlying fields may transparently retrieve data via an interface from another system. The following figure demonstrates creating an ODBC interface:

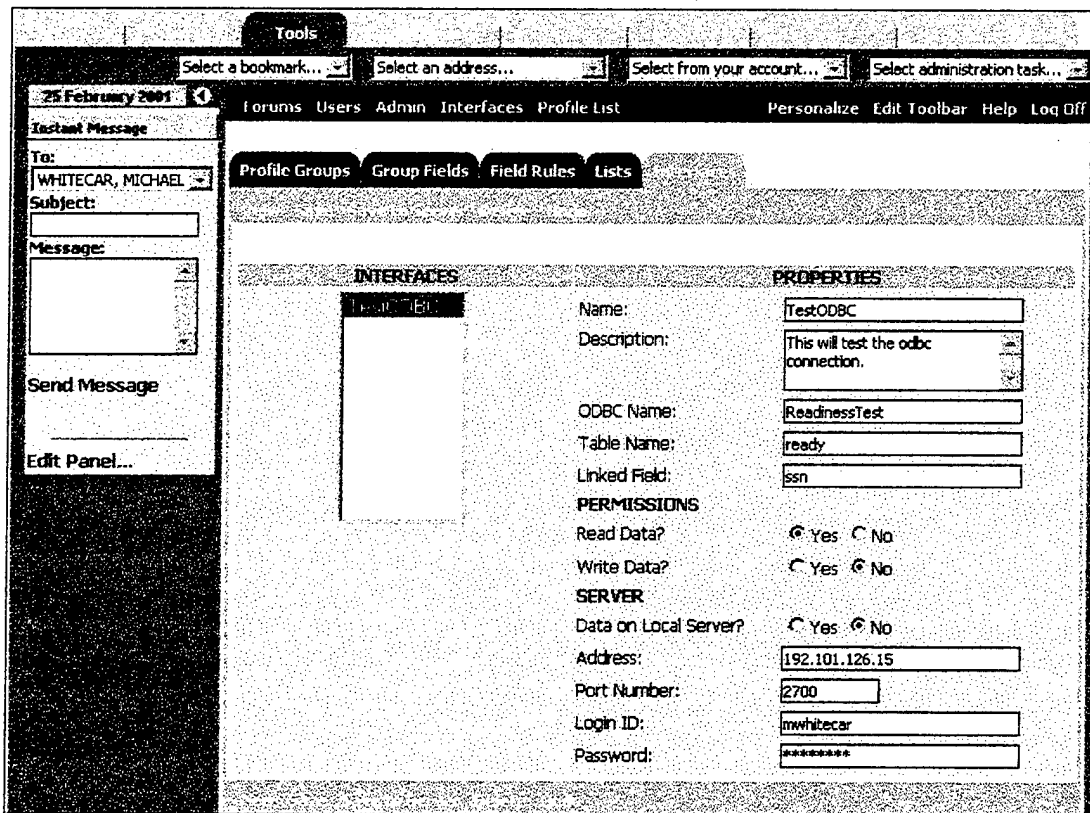


Figure 13. Screen shot of creating an ODBC Interface

G. INCORPORATING INTELLIGENT TECHNOLOGIES IN THE TASKS OF SYSTEM ADMINISTRATION

System Administration has always become a burden to the beholder of the title. Many of the responsibilities are placed on an individual who has the position as a collateral duty. This individual is then required to attend just-in-time training if it is available from the “outgoing” administrator, or attend training at an institution. Such

laborious learning techniques include database administration, running a typical type of server, and from the software program's perspective, run a myriad of individual tasks. Furthermore, to complicate the issue, administrators are required to run batch processes that can accommodate reporting capabilities and forward information to other external sites using such technologies as FTP or sending media on disk. As an administrator to the system, they have to create user accounts, maintain tables of information, and understand the process of the system. Security checks must be accommodated to ensure the protection and privacy of data stored on the system. From a collateral duty perspective, this is all too time consuming, as this may be one of many administrator duties this person may have.

To accommodate the administrator, intelligent technologies may be integrated to ease the burden and responsibilities. Intelligent agents may be used to "sniff" the audit logs to look for inappropriate patterns of connectivity and usage. Automation techniques may be imposed to further streamline the administration process.

If the system is decentralized with respect to its usage among many facilities, appointing administrators at each port may be difficult and not warranted for security purposes. User accounts cannot be managed individually because providing access to these accounts may provide access to other areas of administration that should not be delegated.

Some of the techniques used within Readiness Explorer are discussed individually below:

1. User Accounts

Maintenance of user accounts probably makes up the majority of the necessary time to run the system. This includes tasks as adding new users, ensuring the new users are trained in their new environment, providing new users with login IDs and passwords. When a user departs, it is crucial that the accounts are removed from the system. If the administrator is unavailable when the individual is checking onboard, then they have to wait for him/her to return. Therefore, Readiness Explorer will provide a quick and easy method for the administrator to create a new user and assign the appropriate security policy. Once the account is created, email may automatically be forwarded to the new user with their new login ID and password followed by a set of instructions on how to connect to the system. To make things easier, one of the requirements when adding a new user is to let the system know when the user will be departing. This way the system can automatically remove the user when their time has elapsed. This is a great way to ensure temporary accounts are in fact removed from the system without the concern for security. Along these lines, the tasks of adding users may be delegated to department administrators that have access to the system with additional privileges to add, edit and remove users. Thus, the task is delegated and the primary administrator is not tied down to one area of the system. The user account is now part of an intelligent process that will be monitored by the system, removed when appropriate or report any odd patterns.

2. Security Policies

It may be necessary to monitor user accounts while they are on leave or a business trip. In some cases if it appears to be a problem, accounts should be secured while the

user is in a vacant status. Again, using intelligent agents the system can scan the logs for patterns that do not appear normal. Furthermore, users can inform the system when they expect to be vacant and how long with the option to secure their accounts during this period. With these provisions and precautionary measures, users can comfortably be away from the system without the assumptions that someone is using their account due to knowing their login IDs and passwords. If the account was used during this vacant time frame, the system can identify who used the system and what exactly was performed during these transactions.

3. Error Control

Readiness Explorer provides an Error Monitor similar to the Security Reference Monitor, to eliminate unnecessary tasks by the System Administrator. This concept uses intelligent agents to react to an error within the application and try to fix the problem based on a preset of business rules that determine cause and effect. If the error can be corrected, the end user will not have any knowledge of the correction, however, the error and the measures taken to correct it will be recorded in a log file and forwarded to the System Administrator. If the error cannot be corrected, the user will be presented with the error information and an opportunity to provide additional information via a form.

H. PROTECTING THE PRIVACY OF DATA

Protecting the privacy and integrity of data, whether it is the recording of immunization dates, or information relative to user accounts, is a significant development issue. To address security within a web application it must first be understood what the application will do, what data it will transmit, and who its target audience is. Within the design framework of this scheme, there will be many levels of security. The first layer shall begin with the application server. The primary crucial requirement will be the

ability to place the database files outside of any web-enabled directories. References to the databases will be made by names other than the name itself. Within this database framework, specific data that may be of significant value to a user or transaction shall be encrypted. Second, storage of runtime information shall be stored on server variables that are up against a timer or expiration. Information may be stored in variables that are specific to the entire application, session or user. Only the server will have access to these variables, therefore, using them only to dynamically produce desired web pages without exposing the variables over the communication line. In the case of Readiness Explorer, session variables are used extensively not only for security reasons, but also for the effectiveness of providing just-in-time information relevant to the user's experience.

The next level is a common and basic nature, which is the standard login identification (ID), and password. Using a feature called Autopilot, Readiness Explorer can inform the user of password expiration via an email message or when they initially log in. Once the user logs in, the session is then forced through a set of constant conditions such as the time of the day or day of the week that access is allowed. Additional features include blocking by Internet Protocol (IP) addresses and domain names. Every single page that is executed is run through this scenario. Another level entered into this realm is the audit trail. Every single transaction is recorded. Not only is this used to track individual usage, but also the historical data is used for pattern recommendation to assist the user with future support and making recommendations to others users. Along the audit trail capabilities comes the idea of "logging" certain users or certain records if modified to cause an alert. During the maintenance phase that is performed on a daily basis, the audit trail is scanned for patterns that may appear out of

the ordinary. Furthermore, any security violations are automatically forwarded to the System Administrator via email.

Before each page is served to a user their identity is ran through a "reference monitor" concept and a rule based (user based) scenario. The System Administrator via security policies develops this reference monitor. Policies are created by establishing a set of business rules that may prevent an option or enable a component. In this case, the reference monitor consists of pages, components, and actual links that are protected. This becomes very important when a particular user may have the need to know the C Status results of its department but not necessarily know a complete history. Therefore, their ability to view an entire medical readiness profile is disabled.

Around this entire security realm shall exists a layer of protection enforced by a technology known as Secured Sockets Layer (SSL). The SSL protocol is intended to provide a practical, application-layer, widely applicable connection-oriented mechanism for Internet client/server communications security.

I. NETWORK RESOURCES AND A DEPLOYMENT SCENARIO

1. Network Resources

Tests performed by the Naval Medical Information Management Center (NMIMC) have been conducted to determine the effect of adding the traffic generated by Readiness Explorer to the existing traffic at the Bethesda National Naval Hospital and its remote sites.

Readiness Explorer was added to the Commercial Network (COMNET) model of Bethesda's communications backbone. Readiness Explorer traffic was generated both at

Bethesda and at 28 Bethesda remote sites. The existing traffic at Bethesda was modeled as background traffic.

It was assumed that there would be 1340 Readiness Explorer users at Bethesda and a total of 900 users at the remote sites. The assumption was made that the Bethesda users will be evenly distributed over 13 Local Area Network (LAN) segments, which work out to a total of 102 users per segment. In addition, each of the LAN segments has traffic from applications other than Readiness Explorer. The background utilization of each LAN segment from this traffic has been determined from an EDS simulation. The following table lists the addresses of each of the LAN segments, gives its bandwidth, and shows the utilization of each segment allocated to non-Readiness Explorer traffic in the simulation:

LAN Segment	Bandwidth	Utilization Due to Background Traffic
131.158.80	10BaseT	11.09 %
131.158.81	10BaseT	9.76 %
131.158.82	10BaseT	7.75 %
131.158.83	10BaseT	0.91 %
131.158.84	10BaseT	5.90 %
131.158.85	10BaseT	3.10 %
131.158.86	10BaseT	5.70 %
131.158.87	10BaseT	0.46 %
131.158.88	100BaseT	9.66%
131.158.89	10BaseT	0.66 %
131.158.160	10BaseT	3.94 %
131.158.161	100BaseT	2.86 %
131.158.163	100BaseT	1.86 %

Table 4. Non-Readiness Explorer Traffic Utilization

All the LAN segments used their current bandwidth. The 900 remote users were assumed to be evenly distributed across the 28 remote sites. This works out to approximately 32 users per site, regardless of the line speed of its access circuit.

It was assumed that every Readiness Explorer user would access the application at the same rate. This rate was assumed to be once every eight hours for each user accessing the system.

Based upon the demo Readiness Explorer application running on the web site, there are three basic types of activities users can access:

- Adding the profile of a new patient into the database;
- Searching the database for a patient;
- Viewing a report.

It was assumed that users would conduct two types of Readiness Explorer sessions. In the first type, a user would log in, input the profile of a new patient, and then log out. In the second type, a user would log in, search for a patient, view two reports (presumably of that patient), and then log out. It was further assumed that Search and Report session (the second type) would occur twice as often as the New Profile session (the first type). The following tables the user interactions and the response from Readiness Explorer:

User's Request	Host's Response
1. URL	2. Login Screen
3. Login Data	4. Home Screen
5. Profile Request	6. New Profile Screen
7. New Profile Data	8. Profile Listing
9. Logout	

Table 5. User Interaction and Readiness Explorer Response

User's Request	Host's Response
1. URL	2. Login Screen
3. Login Data	4. Home Screen
5. Search Request	6. Search Screen
7. Search Data	8. Search Results
9. Report Request	10. Report List
11. Specify Report	12. Report
13. Report Request	14. Report List
15. Specify Report	16. Report
17. Logout	

Table 6. Search and Report Session and Readiness Explorer Response

Whenever possible, message sizes were obtained by determining the size of the Readiness Explorer screens from the web site. Report sizes were established from a previous government communication system. Other message sizes were estimated. The smallest message size assumed was 1000 bytes.

The following table lists the performance results for the 13 Bethesda LAN segments, which contained Readiness Explorer users in the simulation. For each segment, the table shows its utilization, both overall and due to Readiness Explorer traffic, its collision percentage, and its average and maximum frame transmission delay.

Segment	Utilization	
	Overall	Due to Cold Fusion
131.158.80	11.2 %	.08 %
131.158.81	9.8 %	.07 %
131.158.82	7.8%	.08 %
131.158.83	1.0%	.07 %
131.158.84	7.6 %	.08 %
131.158.85	3.2%	.07%
131.158.86	5.8 %	.07 %
131.158.87	0.5%	.07 %
131.158.88	13.0%	3.4 %
131.158.89	0.7%	.07 %
131.158.160	4.0%	.08 %
131.158.161	2.9%	.01 %
131.158.163	1.9 %	.01%

Table 7. Performance Results

All of the segments have low utilization, with very little of the overall utilization coming from the Readiness Explorer application. Segment .88, which includes the host, has the highest Readiness Explorer utilization at about three and a half percent. The collision percentages are low for all segments, except segment .88 where it reaches 17%. As for the Wide Area Network (WAN) links, none of their utilization is more than 35%.

The addition of the Readiness Explorer application would have minimum effects on the performance of the LAN segments at Bethesda National Naval Hospital. The simulation shows that only the segment connected to the application server has a utilization of 13%, where Readiness Explorer contributed only 3.4% of the traffic. However, this segment does have around 17% collision rate. WAN links all have adequate capacity in support of Readiness Explorer.

2. Deployment Scenario

Using an overseas MTF as an example, the first step to deploy Readiness Explorer was to understand the technical solution. Because the command was already heavily

involved in building an Intranet, and the majority knew how to operate a web browser, implementing and using Readiness Explorer would not be a challenge. The command understood that using a web browser eliminates tedious hours of training, eliminates high bandwidth requirements for their remote clinics, and it is virtually available anywhere, anytime.

To precede the launch of Readiness Explorer, importing the older data was required. Because this data came from many sources such as Excel spreadsheets, Access databases, and paper rosters, it was determined that the best method was to start from scratch. Therefore the command utilized the Management Information Systems (MIS) department classroom, accommodated with eight Personal Computers (PCs), to review and enter data from each medical record. Less than an hour of training was provided to the Hospital Corpsmen (HMs) from Patient Administration, prior to the mass review and update.

Even this first step provided very useful indicators to the accuracy of the command's corpsmen knowing how to read a health record. After the completion of the data entry process, a model consisting of the requirements to satisfy a C1 rating was applied. This model, known as the "filtering process", occurs on a daily basis, registered the overall command with a low 32% C1 rating.

The next step was to establish the Port of Entries or "portals." The portals were defined as Personnel, Immunizations, Laboratory, Physical Examinations and Contingency. Each portal was equipped with a 486 PC or above and a web browser. This establishment and understanding of the portals is a key concept to the process. Each portal must understand their significance in ensuring that immediately after a service is

provided that it is entered into Readiness Explorer. Additionally, it is the responsibility of the service member to ensure this data is entered as well.

To place the responsibility back into the hands of the service member and to ensure their readiness status is in compliance, utilization of the forecast reports was needed and a portal for accessibility. The service member is required to provide their SSN, DOB, UIC and Last Name. Because each profile was visually appealing the service member could easily identify those items that rated a C2 status.

In alignment with taking medicine to the deckplate, mass immunizations were arranged at each external command. A team of Corpsmen would arrive at the command, provide the immunization, and immediately record it in the service member's profile via their web browser. This was the first time that not only was medicine taken to the deckplate, but data entry was done on location. This process took minimal time for the service member thus reducing their time away from the workforce. If a web browser was not available, laptops were provided with the capability of batch processing.

V. CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

In the development of Readiness Explorer, a prototype was implemented at Naval Hospital Lemoore with an overwhelming sense of success. The paradigm changes and the added functionality were well received. The recommendation is not to deploy Readiness Explorer, but to expose the concepts behind it, to take a look at some of the most advanced usage of advanced technologies, and to consider the benefits of changes and accepting the paradigms of turning data into accurate real-time information. The ideas and methodologies presented in this thesis may be easily transferred to existing programs and future innovations.

An observation is immediately noticeable that can craft Readiness Explorer into a portal of information or a thin client. Leaving systems already in place, Readiness Explorer can retrieve data from any one system and provide the compiled data to the user in one simple page. Furthermore, the data will take advantage of Readiness Explorer's ability to filter through the data and provide accurate real-time C status information. Readiness Explorer can also review the data nightly and provide an accurate C status rating, and it can look for those records overdue according to the business rules set and enforced by the command.

By providing a dynamic and open structure that can support a multitude of platforms, scenarios and environments, and medical readiness requirement may be easily recorded and tracked. For example, if a particular platform needs to be readily deployed to areas within the African borders, a platform could be created that would accommodate all of the required immunizations. Therefore, an accurate list of personnel can be

continuously tracked throughout the deployment. Other readiness requirements may also be added such as “wills”, financial elements, and educational elements.

In another scenario, if once deployed the platform or response team may not have a real time connection or perhaps only a connection at predetermined times that may not allow for real time reporting or updating capabilities. Therefore, batch processing is allowed that will allow a set of records to be downloaded to a PC or laptop for viewing and updating as necessary. Once a connection is established, the batch process may be uploaded and processed by the server. At the same time, a new set of records may be downloaded.

With respect to reporting, because all of the functionality within Readiness Explorer may be delegated, reporting tasks can now become the responsibility of an appointee at an external command such that the external command can now retrieve their own reports and furthermore, have their information automatically emailed to them on a regular basis.

B. RECOMMENDATIONS

The following recommendations emphasize the concepts within Readiness Explorer that may be integrated into other existing applications or used as a model to develop new:

- Develop Internet platforms that are portable, dynamic, and customizable.
- Use preexisting and standardized technologies such as XML to store, retrieve, and forward information.
- Provide an interface that is customizable by its users.

- Provide components within the application that can interface with other components, systems, and applications. This may include incorporating email interfaces, calendar functions, and portals to a myriad of information.
- Delegate reporting tasks and administration tasks to local and external system managers.
- Integrate multimedia technologies to aid the user with help and application tours.
- Enable pattern matching through intensive recording that can dynamically present information to its users in the order of usage and make recommendations to others.
- Take advantage of electronic communication technologies that can forward information such as alert thresholds to hand held devices such as cell phones and pagers, and Personal Digital Assistants (PDAs).
- Integrate features such as automatic date expirations with deletion to alleviate the need to routinely monitor user accounts, and other date sensitive tasks.
- Continuously interact with the user throughout their usage of the application. Error control shall be in place to monitor the user's action and correct any mistakes transparently and record such errors and forward them to the system administrator.
- Provide resources such as documents, white papers, brochures, and links to customers via printing on site, or forwarding via email.

- When running reports, users should be able to create their own reports and share them with others. Based off of pattern matching, reports by other users should be recommended.
- Intensify the use of portals to integrate systems, whether legacy or new, into one simple interface.
- Provide an open architecture that allows commands to develop their own code and share this code or components with others. This allows the ability to extend the baseline of the application to accommodate internal business processes and needs. Components may be developed in any type of language including HTML, DHTML, Perl, ASP, or ColdFusion.
- Enable a filtering process as records are retrieved and reports produced that dynamically and immediately ensure that the data is rated accordingly to readiness standards according to established business rules.

APPENDIX A. SAMPLE MEDICAL SYSTEM SURVEY

TO: _____

My name is LT Michael Whitecar and I am currently a student at the Naval Postgraduate School (NPS) in Monterey, CA., with a concentration in Information Systems Management and desire your assistance in the development of my thesis titled *“The Use of Intelligent Real Time Technologies to Implement, Monitor and Forecast Medical Readiness.”*

The purpose of this research is to capitalize and integrate the use of intelligent real time technologies to effectively implement a system to monitor medical readiness resources, provide and forecast long-term readiness states, and promote the well being of our service members through proactive and interactive services. I realize there may be similar systems already developed or in development, thus I would to analyze their characteristics.

It is my understanding, according to the Naval Medical Information Management Center (NMIMC) Homepage, that you are the Project Manager or contributing member to the _____ project. Therefore, I would like to take this opportunity to present a set of questions relative to _____ and what it contributes to medical readiness. If your time permits, I would truly be grateful for your reply.

Simply reply to this message by filling in the information below:

Please choose the system(s) listed below that interface with _____ and how the interface is conducted (i.e. real-time or batch processing)?

- SPMS II

- DMHRS
- CHCS
- DEERS
- DOHRS
- ITS
- PHCA
- Other _____

Please choose all levels of reporting capabilities:

- Dependent on trained user, i.e. the use of Business Objects
- Query By Example (QBE)
- Structure Query Language (SQL)
- Menu driven reports
- User-Defined reports with programming required
- User-Defined reports without programming required

Can service members retrieve information directly from _____? Yes or No

Is data collected at the site of service or entered in batch mode? Yes or No

Can _____ be customized to accommodate different Areas of Operations or platforms without contractual support? Yes or No

Does _____ provide a web-based front end? Yes or No

How is information passed to higher authority, i.e. BUMED?

- Email from site
- Email from program manager
- Higher authority is able to retrieve their own data when they need it
- Reports are produced, printed and mailed
- Reports are produced and submitted by electronic transfer (i.e. FTP)

Thinking in terms of System Administration, what advantages does _____ offer to accommodate ease of maintaining?

Thanks very much for your time.

/r Mike Whitecar
 LT Michael Whitecar
 Naval Postgraduate School
 Monterey, CA

APPENDIX B. CONFIGURATION DATABASE DATA DICTIONARY

TABLE: tblColorElements

DEFINITION: Color elements are used in part to define the display of Cascade Style Sheet.

NAME	TYPE	SIZE
colorElements_ID	Long Integer	4
ceLabel	Text	50
ceID	Text	50
ceValueName	Text	50
ceCSSName	Text	50
ceDefaultValue	Text	6

TABLE: tblColorGroups

DEFINITION: Used to identify specific components and their related color scheme.

NAME	TYPE	SIZE
colorGroup_ID	Long Integer	4
cgTitle	Text	30

TABLE: tblColor

DEFINITION: A complete list of colors and their appropriate HEX code.

NAME	TYPE	SIZE
color_id	Long Integer	4
color	Text	30
color_code	Text	10

TABLE: tblErrors

DEFINITION: Tracks each error and specific details.

NAME	TYPE	SIZE
error_id	Long Integer	4
detail	Memo	-
message	Memo	-
NativeErrorCode	Text	10
SQLState	Text	10
Type	Text	30
dateTimeofError	Date/Time	8
error_user	Text	20
error_ip	Text	15

TABLE: tblFeedback

DEFINITION: Tracks user feedback when errors occur.

NAME	TYPE	SIZE
feedback_id	Long Integer	4
feedback	Memo	-
type	Text	10
comments_by_admin	Memo	-
dateSubmitted	Date/Time	8
ip	Text	15
user	Text	20
cargo	Text	30

TABLE: tblForumBody

DEFINITION: Tracks messages of forum discussions.

NAME	TYPE	SIZE
forumBody_ID	Long Integer	4
forumHeader_ID	Long Integer	4
body	Memo	-

TABLE: tblForumHeaders

DEFINITION: Tracks specifics of forum discussions.

NAME	TYPE	SIZE
forumHeader_id	Long Integer	4
forum_id	Long Integer	4
datein	Date/Time	8
thread	Long Integer	4
parent	Long Integer	4
author	Text	64
subject	Text	128
email	Text	64
host	Text	64
email_reply	Text	10

TABLE: tblForums

DEFINITION: Initiates forum discussions.

NAME	TYPE	SIZE
forum_id	Long Integer	4
name	Text	60
collapse	Text	1
createdBy	Text	15
dateCreated	Date/Time	8

TABLE: tblInterface_ODBC

DEFINITION: Create an ODBC interface

NAME	TYPE	SIZE
interface_odbc_id	Long Integer	4
io_name	Text	30
io_description	Memo	-
io_odbc	Text	40
io_table	Text	30
io_linkedField	Text	30
io_read	Yes/No	1
io_write	Yes/No	1
io_isLocal	Yes/No	1
io_server	Text	50
io_server_port	Text	5
io_server_login	Text	20
io_server_password	Text	20

TABLE: tblInterfaceFieldMaps

DEFINITION: Tracks fields from interfaces to actual tables within application.

NAME	TYPE	SIZE
fieldMap_ID	Long Integer	4
pf_id	Long Integer	4
interface_id	Long Integer	4
fieldMap_Name	Text	50

TABLE: tblLinkColorToGroup

DEFINITION: Links color elements to application group

NAME	TYPE	SIZE
ColorElementsToGroups_ID	Long Integer	4
ColorGroupID	Long Integer	4
ColorID	Long Integer	4

TABLE: tblLinkFieldToProfile

DEFINITION: Links dynamic fields to profiles

NAME	TYPE	SIZE
linkFieldToProfile_ID	Long Integer	4
pg_id	Long Integer	4
pf_id	Long Integer	4

TABLE: tblLinkOptionsToPage

DEFINITION: Links page options to specific web pages.

NAME	TYPE	SIZE
linkOptionsToPage_ID	Long Integer	4
poi_id	Long Integer	4
po_id	Long Integer	4

TABLE: tblOperators

DEFINITION: A list of operators used for creating business rules.

NAME	TYPE	SIZE
operator_id	Long Integer	4
operator_symbol	Text	10
operator_text	Text	30
operator_type	Text	10

TABLE: tblPageOptionItems

DEFINITION: Links specific options to pages.

NAME	TYPE	SIZE
poi_id	Long Integer	4
poi_URL	Text	100
poi_title	Text	75
poi_text	Memo	-

TABLE: tblPageOptions

DEFINITION: A list of page options to assign.

NAME	TYPE	SIZE
po_id	Long Integer	4
po_name	Text	20
po_columns	Byte	1
po_showText	Yes/No	1

TABLE: tblPanelComponents

DEFINITION: A list of panel components.

NAME	TYPE	SIZE
panelComponent_ID	Long Integer	4
pcTitle	Text	30
pcOption	Byte	1
pcFileName	Text	30
pcLink	Text	75
pcInline	Memo	-

TABLE: tblProfileFields

DEFINITION: Tracks fields for each profile.

NAME	TYPE	SIZE
pf_id	Long Integer	4
pfName	Text	30
pfLabel	Text	30
pfSize	Byte	1
pfDescription	Memo	-
pfCstatus	Byte	1
pfType	Byte	1
pfFailColor	Text	10
pfEncrypt	Yes/No	1
pfIncludeInSearch	Yes/No	1
pfrequiredField	Yes/No	1
pfRange	Text	10
pfValidationType	Long Integer	4
pfHelpMessage	Text	75
pfHelpSummary	Memo	-
pfhasInterface	Yes/No	1
pfinterfaceName	Text	30

TABLE: tblProfileGroups

DEFINITION: Tracks specific information relative to each profile.

NAME	TYPE	SIZE
pg_id	Long Integer	4
pgName	Text	20
pgDescription	Memo	-
pgLabel	Text	30
pgLayoutType	Byte	1
pgLayoutCols	Byte	1
pgOrderBy	Text	10

TABLE: tblProfileRules

DEFINITION: Tracks specific information relative to each business rule within a profile.

NAME	TYPE	SIZE
pr_id	Long Integer	4
pr_name	Text	50
pr_description	Memo	-
pr_rule	Memo	-

TABLE: tblReportOptions

DEFINITION: Tracks specific information relative to options within each report.

NAME	TYPE	SIZE
reportOption_ID	Long Integer	4
report_ID	Long Integer	4
optionsText	Text	15
optionsURL	Text	150
optionsAccess	Text	2
optionsImage	Text	50

TABLE: tblReports

DEFINITION: Tracks specific information relative each report.

NAME	TYPE	SIZE
report_ID	Long Integer	4
reportName	Text	75
reportDescription	Memo	-
shortName	Text	20
includeToDate	Yes/No	1
includeFromDate	Yes/No	1
includeFieldList	Yes/No	1
includeMonthAsString	Yes/No	1
includeMonthCount	Yes/No	1
includeQuery	Yes/No	1
sqlQuery	Memo	-
sqlBody	Memo	-
accessLevel	Text	2
whoCreated	Text	15
dateCreated	Date/Time	8
queryDisplay	Text	30
queryValue	Text	20
includeList	Yes/No	1
listDisplay	Text	30
listValue	Text	20
idField	Text	20
isGlobal	Yes/No	1
queryField	Text	20
orderField	Text	20
usedataSource	Text	10

TABLE: tblSummary

DEFINITION: Contains list of help summaries.

NAME	TYPE	SIZE
summary_ID	Long Integer	4
shortName	Text	20
summary	Memo	-

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX C. USER DATABASE DATA DICTIONARY

TABLE: cData

DEFINITION: Specifically used by ColdFusion.

NAME	TYPE	SIZE
cfid	Text	20
app	Text	64
dat	Memo	-

TABLE: cgGlobal

DEFINITION: Specifically used for ColdFusion.

NAME	TYPE	SIZE
cfid	Text	20
data	Memo	-
lvisit	Date/Time	8

TABLE: tblGroups

DEFINITION: User groups

NAME	TYPE	SIZE
usergroup_id	Long Integer	4
ug_name	Text	20
ug_description	Text	50
ug_access	Text	15

TABLE: tblMyAddressBook

DEFINITION: User's personal Address Book

NAME	TYPE	SIZE
myAddressBook_ID	Long Integer	4
user_id	Long Integer	4
cFname	Text	30
cLname	Text	30
cRank	Text	10
cTitle	Text	50
cOrganization	Text	40
cHomepage	Text	75
cEmailAddress	Text	75
pPhone	Text	30
pFax	Text	20
pCellPhone	Text	20
pPager	Text	30
dNotes	Memo	-

TABLE: tblMyBookMarks

DEFINITION: User's personal bookmarks.

NAME	TYPE	SIZE
myBookmark_id	Long Integer	4
user_id	Long Integer	4
bmLabel	Text	20
bmMessage	Text	50
bmURL	Text	75
newWindow	Byte	1

TABLE: tblMyDesktop

DEFINITION: User's desktop preferences.

NAME	TYPE	SIZE
myDesktop_ID	Long Integer	4
user_id	Long Integer	4
panelStatus	Yes/No	1
homepage	Text	75
panelDisplay	Text	50
dateDisplay	Text	50
deskTop	Text	30

TABLE: tblMyEmail

DEFINITION: User's personal email.

NAME	TYPE	SIZE
myEmail_ID	Long Integer	4
user_id	Long Integer	4
myEmailSetup_id	Long Integer	4
emailTo	Text	75
emailFrom	Text	75
emailSubject	Text	75
emailReplyTo	Text	75
emailNumber	Long Integer	4
emailHeader	Memo	-
emailDate	Text	30
emailCC	Memo	-
emailAttachments	Memo	-
emailAttachFiles	Memo	-
emailMessage	Memo	-

TABLE: tblMyEmailSetup

DEFINITION: User's personal email configuration.

NAME	TYPE	SIZE
myEmailSetup_ID	Long Integer	4
user_id	Long Integer	4
emailTitle	Text	30
userName	Text	50
userOrganization	Text	50
userEmailAddress	Text	75
userReplyEmailAddress	Text	75
popAddress	Text	50
popAddressPort	Byte	1
smtpAddress	Text	50
smtpAddressPort	Byte	1
accountName	Text	30
accountPassword	Text	20
rememberPassword	Yes/No	1
deleteOnServer	Yes/No	1
timeOut	Integer	2
setAsDefault	Yes/No	1

TABLE: tblMyEmailTrash

DEFINITION: User's personal email trash.

NAME	TYPE	SIZE
myEmailTrash_ID	Long Integer	4
user_id	Long Integer	4
myEmailSetup_id	Long Integer	4
emailTo	Text	75
emailFrom	Text	75
emailSubject	Text	75
emailReplyTo	Text	75
emailNumber	Long Integer	4
emailHeader	Memo	-
emailDate	Text	30
emailCC	Memo	-
emailAttachments	Memo	-
emailAttachFiles	Memo	-
emailMessage	Memo	-

TABLE: tblMyPanelComponents

DEFINITION: User's personal panel components.

NAME	TYPE	SIZE
myPanelComponents_ID	Long Integer	4
panelComponent_ID	Long Integer	4
user_id	Long Integer	4

TABLE: tblMySecurity

DEFINITION: User's personal security preferences.

NAME	TYPE	SIZE
mySecurity_id	Long Integer	4
user_id	Long Integer	4
rememberLogin	Yes/No	1
autoPassword	Yes/No	1
absentFrom	Date/Time	8
absentTo	Date/Time	8
absentPreventLogon	Yes/No	1
emailAuditTrail	Yes/No	1
emailAuditTrailWhen	Byte	1
emailAdminChanges	Yes/No	1

TABLE: tblMyToolbar

DEFINITION: User's personal toolbar settings.

NAME	TYPE	SIZE
mytoolBar_id	Long Integer	4
user_id	Long Integer	4
tbLabel1	Text	20
tbMessage1	Text	50
tbURL1	Text	75
newWindow1	Byte	1
tbLabel2	Text	20
tbMessage2	Text	50
tbURL2	Text	75
newWindow2	Byte	1
tbLabel3	Text	20
tbMessage3	Text	50
tbURL3	Text	75
newWindow3	Byte	1
tbLabel4	Text	20
tbMessage4	Text	50
tbURL4	Text	75
newWindow4	Byte	1
tbLabel5	Text	20

tbMessage5	Text	50
tbURL5	Text	75
newWindow5	Byte	1

TABLE: tblUserAccounts

DEFINITION: User Accounts.

NAME	TYPE	SIZE
user_id	Long Integer	4
lname	Text	30
fname	Text	30
rank	Text	10
ssn	Text	4
phone	Text	15
email	Text	75
command	Text	30
user_loginid	Text	15
user_password	Text	10
password_change	Date/Time	8
access	Text	15
expire	Date/Time	8
template	Text	20
last_logon	Date/Time	8
login_attempts	Byte	1
force_password	Yes/No	1
locked	Yes/No	1
panelStatus	Yes/No	1
homepage	Text	75
panelDisplay	Text	50
dateDisplay	Text	50
deskTop	Text	30

THIS PAGE INTENTIONALLY LEFT BLANK

APPENDIX D. FILE STRUCTURE

The following tables list each of the required files with a definition relative to their role within Readiness Explorer. ColdFusion processes files with a “CFM” extension, “JS” contains JavaScript Code, “CFG” contain configuration items, and “HTM” are standard HTML files.

CATEGORY/FILENAME	DEFINITION
ADD-INS	Using the open architecture concept, these files are developed using ColdFusion and are simple “add-ins” that support the underlying infrastructure.
ADDINPANEL.CFM	Add-in that supports the User’s side panel.
ADDINSEARCH.CFM	Support search engine that is displayed at the top left of every page.
ADDINTABS.CFM	Controls and manages the display of tabs.
ADDINTOOLBAR.CFM	Controls and manages the display of the toolbar.
OPEN-ARCHITECTURE DESIGN	These set of files support the actual development of the application using scripts to create how such objects as tabs are set up.
ADMIN.CFM	Main file for developing the application and dynamically changing components. This file acts as a frame set that calls other files.
ADMINADDNEWPAGE.CFM	Adds a new page (i.e. tabbed page)
ADMINHELPPBODY.CFM	Displays help information relative to the body function of designing a page.
ADMINHELPPFOOTER.CFM	Displays help information relative to the footer function of designing a page.
ADMINHELPPHEADERLEFT.CFM	Displays help information relative to the

ADMINHELPHEADERRIGHT.CFM	left side of the header function of designing a page.
ADMINHELPPANEL.CFM	Displays help information relative to the right side of the header function of designing a page.
ADMINHELPTABS.CFM	Displays help information relative to the Panel function of designing a page.
ADMINHELPTOOLBAR.CFM	Displays help information relative to the tabs function of designing a page.
ADMINHELPTOOLBAR.CFM	Displays help information relative to the body function of designing a page.
ADMINPAGEOPTIONS.CFM	Displays the available options when creating a page.
ADMINUPDATEPAGE.CFM	Updates the XML configuration file for each change or addition.
ADMINWRITECONFIG.CFM	Write the actual configuration to disk.
BLANK.HTM	Simply a placeholder.

<p>SYSTEM ADMINISTRATION</p>	<p>The System Administration group includes the files that make up tasks including some of the underlying dynamic development processes such as creating readiness profiles.</p> <p>NOTE: Many of the files contain the "list" key word. Although these file allow interaction, the default action is to display a record set of the configurable items.</p>
-------------------------------------	---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

ADMINLISTINTERFACES.CFM	Provides the ability to create interfaces such as ODBC or FTP transactions.
ADMINLISTPANELCOMPONENTS.CFM	Manages Panel Components.
ADMINLISTPROFILEFIELDS.CFM	Manages the fields that are dynamically created and attached to groups.
ADMINLISTPROFILEFIELDSDRULES.CFM	Manages the business rules for each dynamically created field.
ADMINLISTPROFILES.CFM	Manages the profiles.
ADMINLISTREPORTOPTIONS.CFM	Manages report options such as edit and deleting reports.
ADMINLISTREPORTS.CFM	Manages reports.

ADMINLISTSUMMARY.CFM	Manages help summary for each page.
ADMINLISTUSERGROUPS.CFM	Manages user groups.
ADMINLISTUSERS.CFM	Manages users.
ADMINSECURITYCONFIGURATION.CFM	Manages security information and provides an interface to the creation of security profiles.
ADMINCONFIGURATION.CFM	Provides initial set of requirements for Readiness Explorer to execute.
ADMINWRITELOGON.CFM	Writes the logon screen as information changes relative to the administration of the application.

PANEL COMPONENTS

The panel component files are simply a collection of user-defined files that may be used within the user's panel.

COMPSENDINSTANTEMAIL.CFM	Send instant email via panel.
COMPSTOCKGRABBER.CFM	Retrieve and organize stock quotes within panel.

JAVASCRIPT FILES

These files hold JavaScript code that can be shared among multiple files.

COOKIES.JS	Contains functions for cookie operations.
WDDX.JS	Contains functions for WDDX/XML functions. This file mainly converts XML data to JavaScript objects.
RE2001.JS	Contains basic simple functions such as setStatus that display information in the status bar.

GET INFORMATION

Get Information files are small routines that can quickly retrieve information in order to support other running processes.

GETCONTENT.CFM	Get application content information.
GETGREETING.CFM	Get user-greeting information, like "Good Afternoon."
GETINFORMATION.CFM	Get general application information.
GETMESSAGE.CFM	Get a message from email.
GETPAGECOMMENTS.CFM	Get page comments.

GETPROGRAMDEFAULTS.CFM	Get program defaults such as directories, command name and system administrator information.
GETSECURITY.CFM	Get security information relative to both user and application.
GETSTRUCTURE.CFM	Get application structure.
GETSTYLE.CFM	Get CSS information.
GETUSERINFORMATION.CFM	Get user information.

USER FILES	User files are used to address their personalization options within Readiness Explorer.
-------------------	-----------------------------------------------------------------------------------------

MYACCOUNT.CFM	Interface for user to be able to modify some of their account information such as password and demographics.
MYADDRESSBOOK.CFM	Personal address book.
MYBOOKMARKS.CFM	Personal bookmarks.
MYCALENDAR.CFM	Personal calendar.
MYCOLORS.CFM	Provides an interface for user to be able to change the colors of Readiness Explorer and share with other users.
MYEMAIL.CFM	Personal email account management.
MYEMAILSETUP.CFM	Personal email account set up.
MYINFORMATION.CFM	Displays a list of user personalization options.
MYPANEL.CFM	Personal panel options.
MYREMINDERS.HTM	Personal reminders.
MYSECURITY.CFM	Provides an interface for the user to enter specific security requirements about their account while they may be in absence.
MYTOOLBAR.CFM	Personal toolbar.

TABBED PAGES	Tabbed pages are simply easy methods of organizing data. These pages are created dynamically and simply point to specific pages relative to the tab.
---------------------	------------------------------------------------------------------------------------------------------------------------------------------------------

PAGEADMIN.CFM	Administration tab.
PAGEERROR.CFM	Error page that is displayed under the

PAGEFEEDBACK.CFM	help tab. Also, provides an option for the user to send feedback to the system administrator.
PAGEFORUMS.CFM	Provides application feedback under the Help tab.
PAGEHELP.CFM	Provides an interface to the forums under the Resource tab.
PAGEHOME.CFM	Help tab.
PAGELOGOFF.CFM	Home tab.
PAGEPERSONALIZE.CFM	Logoff page that is not actually displayed but deletes all Session variables and logs user off of application.
PAGEPROFILES.CFM	User personalization page under the myAccounts tab.
PAGETOOLS.CFM	Profiles tab.
	Tools tab.

SET INFORMATION FILES	These files are designed to quickly and efficiently update information.
SETUSERINFORMATION.CFM	Sets user information.

SUBROUTINES	These files are subroutines that are called by other files for simple processes.
--------------------	----------------------------------------------------------------------------------

SRADDBODYOPTION.CFM	Supports adding body link elements to a page.
SRCREATEBODYOPTION.CFM	Creates links in a page.
SRDIALOGWINDOW.CFM	Creates a dialog window.
SRERRORMONITOR.CFM	Supports error management.
SRFEEDBACK.CFM	Supports providing feedback to user.
SRLINKFINDER.CFM	Turns links in email messages to active links.
SRLISTFORUMTHREADS.CFM	Lists forum threads.
SRMESSAGECONVERT.CFM	Converts forum message to HTML formatting.
SRPAGEOPTIONS.CFM	Supports adding pages.
SRREPORTENGINE.CFM	Supports the creation and production of reports.

SRSTOCKGRABBER.CFM	Supports the Stock Grabber panel component.
SRWINTAB.CFM	Creates a dialog, tabbed window.
SRHELPSUMMARY.CFM	Supports the display and updating of help summaries.

INFRASTRUCTURE FILES	Infrastructure files support the underlying layers of the application.
-----------------------------	------------------------------------------------------------------------

SUBHEADER.CFM	Provides framework support for the Header section.
SUBPANEL.CFM	Provides framework support for the Panel section.
SUBTABS.CFM	Provides framework support for the Tabbed section.
SUBTOOLBAR.CFM	Provides framework support for the Toolbar section.
SUBFOOTER.CFM	Provides framework support for the Footer section.
SUBBODY.CFM	Provides framework support for the Body section.
COMPONENT_DECLARE.CFM	Used to declare XML/WDDX packets and create the structure that supports each section of the application.
COMPONENT_PREPARE.CFM	Prepares and initializes the supporting structure.
APPLICATION.CFM	A ColdFusion specific file is ran each time another CFM file is called. This is where the Security Reference Monitor is placed.
COMPONENT.CFG	Used to store configuration information.
DESKTOP.CFG	Used to store configuration information relative to user desktop preferences.
INDEX.HTM	The first file user enters which is the logon screen. This has to be an HTML file as ColdFusion cannot yet process because it will try to call the application.cfm file, which will request information from the logon process.
MAIN.CFM	The actual main file that acts as a framework for all of the parts and sections.

APPENDIX E. SOURCE CODE

This appendix contains all of the source code of Readiness Explorer that has been written in ColdFusion using the development environment ColdFusion Studio. Sharing ColdFusion programs and procedures is common practice among programmers, therefore, code that has been written by someone else is stated as so.

1. ADD-IN FILES

ADDINPANEL.CFM

```
<STYLE TYPE="text/css">
A.editPanel:link
{
color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}

A.editPanel:active
{
color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}

A.editPanel:visited
{
color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}

A.editPanel:hover
{
color: blue;
text-decoration: underline;
font-size: 10pt;
}

</STYLE>

<CFQUERY NAME="getUserPanelContent"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      panelComponent_ID
FROM    tblMyPanelComponents
```

```

WHERE user_id = #SESSION.user_id#

</CFQUERY>
<CFQUERY NAME="getPanelContent"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      *
FROM    tblPanelComponents
WHERE 0=0 AND
<CFIF #getUserPanelContent.RecordCount# NEQ 0>
panelComponent_ID IN ( #ValueList(
getUserPanelContent.panelComponent_ID )# )
<CFELSE>
pcTitle = 'Add Components'
</CFIF>
ORDER BY pcTitle

</CFQUERY>

<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="0">

<!--- LEFT PANEL TITLE --->
<TR>
<CFOUTPUT>
<TD CLASS="panelTitleRaised" ALIGN="center">#DateFormat( Now(), 'd mmmmm
yyyy' )#</TD>
<TD ALIGN="center" VALIGN="middle"><A HREF="#CGI.path_info?#Replace(
CGI.query_string, 'showPanel=true', 'showPanel=false' )##Iif(
CGI.query_string DOES NOT CONTAIN 'showPanel', De( '&showPanel=false'
), De( '' ) )##Iif( CGI.query_string DOES NOT CONTAIN
'updateUser=panel', De( '&updateUser=panel' ), De( '' ) )#"><IMG
SRC="#VARIABLES.gvIMAGE_DIRECTORY#hidePanel.gif" BORDER=0 ALT="Click to
Hide Panel"></A></TD>
</CFOUTPUT>
</TR>

<!--- PREPARE TO DISPLAY LEFT PANEL SUB-TITLES AND CONTENTS --->
<TR>
<TD CLASS="panelBackground" ALIGN="center" VALIGN="top" COLSPAN=2>

<!--- ACTUAL SUB-TITLES AND CONTENTS --->
<TABLE WIDTH="125" BORDER="0" CELLPADDING="2" CELLSPACING="0">

<!--- SUB-TITLE DIVIDER --->
<TR>
<TD COLSPAN="2" HEIGHT="5"></TD>
</TR>

<CFOUTPUT QUERY="getPanelContent">
<!--- SUB-TITLE TWO --->
<TR>
<!--- title --->
<TD CLASS="panelComponentRaised" COLSPAN=2>#pcTitle#</TD>
</TR>
<TR>
<TD COLSPAN="2" HEIGHT="3"></TD>
</TR>

```

```

<TR>
<TD CLASS="panelComponentBody" COLSPAN="2">

<!-- body of component -->
<CF_getContent
TYPE="#Iif( pcOption EQ 1,De( 'file' ),De( Iif( pcOption EQ 2,De( 'url'
),De( 'inline' ) ) ) )#"
FILE="#pcFileName#"
URL="#pcLink#"
INLINE="#pcInline#"
>
<!-- end body of component -->

</TD>
</TR>
</CFOUTPUT>
<CFIF #getUserPanelContent.RecordCount# NEQ 0>

<TR>
<TD CLASS="panelComponentBody" COLSPAN="2" ALIGN="center">
<HR SIZE=1 WIDTH=75%>
<A CLASS="editPanel" HREF="main.cfm?a=myAccount&body=myPanel.cfm">Edit
Panel...</A>
</TD>
</TR>

</CFIF>

<TR>
<TD CLASS="panelComponentBody" COLSPAN="2" ALIGN="center">
<HR SIZE=1 WIDTH=75%>
<A CLASS="editPanel"
HREF="main.cfm?a=tools&body=adminListPanelComponents.cfm">Edit
Components...</A>
</TD>
</TR>

<!-- END ACTUAL SUB-TITLES AND CONTENTS -->
</TABLE>
</TD>
</TR>

</TABLE>

```

ADDINSEARCH.CFM

```

<STYLE TYPE="text/css">
TD.searchBody
{
font: 12px Tahoma, Verdana, Arial;
}
.searchEdit
{
background-color: #FFFFFF;
font: 10pt Tahoma, Verdana, Arial;
border: 1px solid blue;
padding: 3px;
}

```

```
width: 180px;
height: 22px;
}
```

```
A.searchLink:link
{
color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}
```

```
A.searchLink:active
{
color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}
```

```
A.searchLink:visited
{
color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}
```

```
A.searchLink:hover
{
color: blue;
text-decoration: underline;
font-size: 10pt;
}
```

```
</STYLE>
```

```
<CFFORM ACTION="addInSearch.cfm">
```

```
<TABLE BORDER=0 WIDTH=100%>
```

```
<TR>
```

```
<TD VALIGN="bottom" ALIGN="right"><IMG SRC="images/google.gif" BORDER=0
ALT="Google Search"></TD>
```

```
<TD VALIGN="middle"><INPUT CLASS="searchEdit" TYPE="text"
NAME="googleSearch">&nbsp;&nbsp;&nbsp;<A CLASS="searchLink"
HREF="addInSearch.cfm">Search</A></TD>
```

```
</TR>
```

```
<TR>
```

```
<TD>&nbsp;&nbsp;&nbsp;</TD>
```

```
<TD CLASS="searchBody"VALIGN="bottom"><CFINPUT TYPE="radio"
NAME="searchWhat" VALUE="re" CHECKED="true">Readiness
Explorer&nbsp;&nbsp;&nbsp;<CFINPUT TYPE="radio" NAME="searchWhat"
VALUE="internet">the Internet</TD>
```

```
</TR>
```

```
</TABLE>
```



```

<!--- TOOLBAR--->
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="0">

<TR VALIGN="middle">

<TD ALIGN="left">

<TABLE BORDER=0 CELLPADDING="0" CELLSPACING="0">
<TR>
<CFLOOP INDEX="listCounter" FROM="1" TO="#ListLen( VARIABLES.myToolBar
)#">
<TD ALIGN="center" VALIGN="middle">
<CFOUTPUT>
&nbsp;&nbsp;&nbsp;<A CLASS="toolBar" HREF="#ListGetAt(
VARIABLES.myToolBarLinks,listCounter )#" TARGET="#ListGetAt(
VARIABLES.myToolBarTarget,listCounter )#">#ListGetAt(
VARIABLES.myToolBar,listCounter )#</A>
</CFOUTPUT>
</TD>
</CFLOOP>
</TR>
</TABLE>

</TD>

<!--- system toolbar --->
<TD ALIGN="right">

<TABLE BORDER=0 CELLPADDING="0" CELLSPACING="0">
<TR>
<CFOUTPUT>
<TD ALIGN="center" VALIGN="middle">
</TD>
<TD ALIGN="center" VALIGN="middle">
&nbsp;&nbsp;&nbsp;<A CLASS="toolBar"
HREF="main.cfm?a=myAccount&body=pagePersonalize.cfm">Personalize</A>
</TD>
<TD ALIGN="center" VALIGN="middle">
&nbsp;&nbsp;&nbsp;<A CLASS="toolBar"
HREF="main.cfm?a=myAccount&body=myToolbar.cfm">Edit Toolbar</A>
</TD>
<TD ALIGN="center" VALIGN="middle">
&nbsp;&nbsp;&nbsp;<A CLASS="toolBar"
HREF="main.cfm?a=resources&body=pageHelp.cfm">Help</A>
</TD>
<TD ALIGN="center" VALIGN="middle">
&nbsp;&nbsp;&nbsp;<A CLASS="toolBar" HREF="pageLogoff.cfm" >Log Off</A>
</TD>
</CFOUTPUT>
</TR>
</TABLE>
</TD>

</TR>

</TABLE>

```

2. OPEN-ARCHITECTURE DESIGN FILES

ADMIN.CFM

```
<!-- frames -->
<FRAMESET ROWS="40%,*">
<FRAMESET COLS="30%,*">
<FRAME NAME="options" SRC="adminPageOptions.cfm" MARGINWIDTH="10"
MARGINHEIGHT="10" SCROLLING="auto" FRAMEBORDER="0">
<FRAME NAME="properties" SRC="blank.htm" MARGINWIDTH="10"
MARGINHEIGHT="10" SCROLLING="auto" FRAMEBORDER="0">
</FRAMESET>
<FRAME NAME="preview" SRC="main.cfm" MARGINWIDTH="10" MARGINHEIGHT="10"
SCROLLING="auto" FRAMEBORDER="0">
</FRAMESET>
```

ADMINADDNEWPAGE.CFM

```
<HTML>
<HEAD>
<TITLE>Untitled</TITLE>
</HEAD>

<CFIF IsDefined( "FORM.NewPageName" )>

<CFINCLUDE TEMPLATE="component_declare.cfm">
<CFINCLUDE TEMPLATE="component_prepare.cfm">

<CFSCRIPT>

temp = ArrayAppend( aComponent,ArrayNew(2) );
ElementNumber = ArrayLen( aComponent );

ComponentNames =
"HEADER_RIGHT,HEADER_LEFT,TABS,TOOLBAR,PANEL,BODY,FOOTER";
ComponentFiles =
"addInSearch.cfm,adminHelpHeaderLeft.cfm,addInTabs.cfm,addInPanel.cfm,addInToolbar.cfm,adminHelpBody.cfm,adminHelpFooter.cfm";

for ( counter = 1; counter LTE ListLen( ComponentNames ); counter =
counter + 1 )
{
aComponent[ElementNumber][counter][cCOMPONENT_NAME] = ListGetAt(
ComponentNames,counter );
aComponent[ElementNumber][counter][cCOMPONENT_TYPE] = "file";
aComponent[ElementNumber][counter][cCOMPONENT_INLINECODE] = "";
aComponent[ElementNumber][counter][cCOMPONENT_FILE] = ListGetAt(
ComponentFiles,counter );
aComponent[ElementNumber][counter][cCOMPONENT_URL] = "";
aComponent[ElementNumber][counter][cCOMPONENT_LOCKED] = "no";
aComponent[ElementNumber][counter][cCOMPONENT_LOCKEDBY] = "";
}
}
```

```

aComponent[ElementNumber][cAPPLICATION][cAPPLICATION_NAME] =
"#FORM.NewPageName#";
aComponent[ElementNumber][cAPPLICATION][cAPPLICATION_DESCRIPTION] =
"Readiness Explorer";
aComponent[ElementNumber][cAPPLICATION][cAPPLICATION_DATECREATED] =
"#Now()#";
aComponent[ElementNumber][cAPPLICATION][cAPPLICATION_CREATEDBY] = "";
aComponent[ElementNumber][cAPPLICATION][cAPPLICATION_LASTUPDATED] = "";
aComponent[ElementNumber][cAPPLICATION][cAPPLICATION_UPDATEDBY] = "";
aComponent[ElementNumber][cAPPLICATION][cAPPLICATION_CARGO1] = "";

</CFSCRIPT>

<CFWDDX
INPUT="#aComponent#"
OUTPUT="NewPacket"
ACTION="CFML2WDDX"
>

<CFFILE
ACTION="write"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#component.cfg"
OUTPUT="#NewPacket#"
>

<CFELSEIF IsDefined( "FORM.DeletePage" )>

<CFINCLUDE TEMPLATE="component_declare.cfm">
<CFINCLUDE TEMPLATE="component_prepare.cfm">

<CFSCRIPT>
temp = ArrayDeleteAt( aComponent,URL.aID );
</CFSCRIPT>

<CFWDDX
INPUT="#aComponent#"
OUTPUT="NewPacket"
ACTION="CFML2WDDX"
>

<CFFILE
ACTION="write"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#component.cfg"
OUTPUT="#NewPacket#"
>

</CFIF>

<BODY>

<CFIF IsDefined( "URL.aID" )>

<CFIF NOT IsDefined( "FORM.DeletePage" )>

<CFFORM NAME="DeletePage" ACTION="adminAddNewPage.cfm?aID=#URL.aID#">
<B>Are you sure you want to delete this page? </B><INPUT TYPE="submit"
NAME="DeletePage" VALUE="Yes">

```

```

</CFFORM>

<CFELSE>

<B>Page deleted.</B>

</CFIF>

<CFELSE>

<CFFORM NAME="AddNewPage" ACTION="adminAddNewPage.cfm">
<B>Page Name: </B><CFINPUT TYPE="text" NAME="NewPageName" SIZE="20"
MAXLENGTH="20" REQUIRED="yes" MESSAGE="Required Field">
<CENTER><INPUT TYPE="submit" VALUE="Add New Page"></CENTER>
</CFFORM>

</CFIF>

</BODY>
</HTML>

```

ADMINHELPPBODY.CFM

The Body section is where the actual page contents is displayed.

ADMINHELPPFOOTER.CFM

The Footer section is where copyrights, quick links, or date last displayed is provided.

ADMINHEADERLEFT.CFM

The Left Header is used to display the application logo. This is usually a GIF or JPEG file.

ADMINHEADERRIGHT.CFM

The Right Header is used to display the application logo. This is usually a GIF or JPEG file.

ADMINHELPPANEL.CFM

The Panel section, which may be displayed on the right, left, or hidden is provided for page and user specific information.

ADMINHELPTABS.CFM

The Tab section is where each page is designated by a tab.

ADMINHELPTOOLBAR.CFM

The Toolbar section is provided for application and user specific instant links.

ADMINPAGEOPTIONS.CFM

```

<CFINCLUDE TEMPLATE="component_declare.cfm">

<CFSCRIPT>

if ( NOT FileExists( cCONFIG_FILE ) )
{

aComponent = ArrayNew( 3 );

ComponentNames =
"HEADER_RIGHT,HEADER_LEFT,TABS,TOOLBAR,PANEL,BODY,FOOTER";
ComponentFiles =
"adminHelpHeaderRight.cfm,adminHelpHeaderLeft.cfm,adminHelpTabs.cfm,adm
inHelpToolbar.cfm,adminHelpPanel.cfm,adminHelpBody.cfm,adminHelpFooter.
cfm";

for ( counter = 1; counter LTE ListLen( ComponentNames ); counter =
counter + 1 )
{
aComponent[1][counter][cCOMPONENT_NAME] = ListGetAt(
ComponentNames,counter );
aComponent[1][counter][cCOMPONENT_TYPE] = "file";
aComponent[1][counter][cCOMPONENT_INLINECODE] = "";
aComponent[1][counter][cCOMPONENT_FILE] = ListGetAt(
ComponentFiles,counter );
aComponent[1][counter][cCOMPONENT_URL] = "";
aComponent[1][counter][cCOMPONENT_LOCKED] = "no";
aComponent[1][counter][cCOMPONENT_LOCKEDBY] = "";
}

aComponent[1][cAPPLICATION][cAPPLICATION_NAME] = "New";
aComponent[1][cAPPLICATION][cAPPLICATION_DESCRIPTION] = "";
aComponent[1][cAPPLICATION][cAPPLICATION_DATECREATED] = "";
aComponent[1][cAPPLICATION][cAPPLICATION_CREATEDBY] = "";
aComponent[1][cAPPLICATION][cAPPLICATION_LASTUPDATED] = "";
aComponent[1][cAPPLICATION][cAPPLICATION_UPDATEDBY] = "";
aComponent[1][cAPPLICATION][cAPPLICATION_CARGO1] = "";

}

</CFSCRIPT>

<CFIF NOT FileExists( cCONFIG_FILE )>

<CFWDDX
INPUT="#aComponent#"
OUTPUT="NewPacket"
ACTION="CFML2WDDX"
>

<CFFILE
ACTION="write"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#component.cfg"
OUTPUT="#NewPacket#"
>

<CFELSE>

```

```

<CFINCLUDE TEMPLATE="component_prepare.cfm">

</CFIF>

<html>
<head>
<title>Untitled</title>
</head>

<body>

<SCRIPT LANGUAGE="JavaScript">

imageDirectory = "images/";
<CFINCLUDE TEMPLATE="ftiens4.js">

foldersTree = gFld("<B>Page Maintenance</B>", "")
insDoc(foldersTree, gLnk(4, "Create a New Page",
"adminAddNewPage.cfm", "", "properties"))

aux1 = insFld(foldersTree, gFld("Current Pages", ""))
<CFLOOP INDEX="counter" FROM="1" TO="#ArrayLen( aComponent )#">

<CFSET VARIABLES.auxLocation = "aux#Evaluate(counter+1)#">
<CFOUTPUT>
#VARIABLES.auxLocation# = insFld(aux1,
gFld("#aComponent[counter][cAPPLICATION][cAPPLICATION_NAME]#", ""))
insDoc(#VARIABLES.auxLocation#, gLnk(4, "Delete",
"adminAddNewPage.cfm?aID=#counter#", "", "properties" ))
insDoc(#VARIABLES.auxLocation#, gLnk(4, "Preview",
"main.cfm?aID=#counter#", "", "preview" ))
insDoc(#VARIABLES.auxLocation#, gLnk(4, "Information",
"adminUpdatePage.cfm?aID=#counter#&comID=#cAPPLICATION#", "", "properties
" ))
insDoc(#VARIABLES.auxLocation#, gLnk(4, "Header: Right",
"adminUpdatePage.cfm?aID=#counter#&comID=#cHEADER_RIGHT#", "", "propertie
s" ))
insDoc(#VARIABLES.auxLocation#, gLnk(4, "Header: Left",
"adminUpdatePage.cfm?aID=#counter#&comID=#cHEADER_LEFT#", "", "properties
" ))
insDoc(#VARIABLES.auxLocation#, gLnk(4, "Tabs",
"adminUpdatePage.cfm?aID=#counter#&comID=#cTABS#", "", "properties" ))
insDoc(#VARIABLES.auxLocation#, gLnk(4, "Toolbar",
"adminUpdatePage.cfm?aID=#counter#&comID=#cTOOLBAR#", "", "properties" ))
insDoc(#VARIABLES.auxLocation#, gLnk(4, "Panel",
"adminUpdatePage.cfm?aID=#counter#&comID=#cPANEL#", "", "properties" ))
insDoc(#VARIABLES.auxLocation#, gLnk(4, "Body",
"adminUpdatePage.cfm?aID=#counter#&comID=#cBODY#", "", "properties" ))
insDoc(#VARIABLES.auxLocation#, gLnk(4, "Footer",
"adminUpdatePage.cfm?aID=#counter#&comID=#cFOOTER#", "", "properties" ))
</CFOUTPUT>

</CFLOOP>

</SCRIPT>
<SCRIPT LANGUAGE="javascript">

```

```
initializeDocument()
</script>

</head>
<BODY BGCOLOR="beige">
</body>
</html>
```

ADMINUPDATEPAGE.CFM

```
<HTML>
<HEAD>
<TITLE>Configuration</TITLE>
</HEAD>

<CFINCLUDE TEMPLATE="component_declare.cfm">

<CFIF ISDEFINED( "FORM.Packet" )>

<CFWDDX
INPUT="#FORM.Packet#"
OUTPUT="aComponent"
ACTION="WDDX2CFML"
>

<CFIF URL.comID EQ 8>

<CFSET aComponent[URL.aID][URL.comID][cAPPLICATION_NAME] =
"#FORM.appName#">
<CFSET aComponent[URL.aID][URL.comID][cAPPLICATION_DESCRIPTION] =
"#FORM.appDescription#">
<CFSET aComponent[URL.aID][URL.comID][cAPPLICATION_DATECREATED] =
"#DateFormat( Now(), 'dd-mmm-yyyy' )#">
<CFSET aComponent[URL.aID][URL.comID][cAPPLICATION_CREATEDBY] =
"mwhitecar">
<CFSET aComponent[URL.aID][URL.comID][cAPPLICATION_LASTUPDATED] = "">
<CFSET aComponent[URL.aID][URL.comID][cAPPLICATION_UPDATEDBY] = "">
<CFSET aComponent[URL.aID][URL.comID][cAPPLICATION_CARGO1] =
"#FORM.appCargo1#">

<CFELSE>

<CFSET aComponent[URL.aID][URL.comID][cCOMPONENT_TYPE] =
"#FORM.comType#">
<CFSET aComponent[URL.aID][URL.comID][cCOMPONENT_INLINECODE] =
"#FORM.comInlineCode#">
<CFSET aComponent[URL.aID][URL.comID][cCOMPONENT_FILE] =
"#FORM.comFileLocation#">
<CFSET aComponent[URL.aID][URL.comID][cCOMPONENT_URL] =
"#FORM.comURL#">
<CFSET aComponent[URL.aID][URL.comID][cCOMPONENT_LOCKED] = "no">
<CFSET aComponent[URL.aID][URL.comID][cCOMPONENT_LOCKEDBY] = "">

</CFIF>

<CFWDDX
```

```

INPUT="#aComponent#"
OUTPUT="NewPacket"
ACTION="CFML2WDDX"
>

<CFFILE
ACTION="write"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#component.cfg"
OUTPUT="#NewPacket#"
>

</CFIF>

<CFINCLUDE TEMPLATE="component_prepare.cfm">

<BODY>
<CFFORM ACTION="adminUpdatePage.cfm?aID=#URL.aID#&comID=#URL.comID#"
NAME="ActionForm">
<TABLE BORDER=0>
<CFOUTPUT>
<INPUT TYPE="hidden" NAME="Packet" VALUE="#HTMLFormat( DataPacket
) #">
</CFOUTPUT>

<CFIF URL.comID EQ 8>

<TR>
<TD><B>Name:</B></TD>
<TD><CFINPUT TYPE="Text" NAME="appName" REQUIRED="Yes" SIZE="30"
MAXLENGTH="15"
VALUE="#aComponent[URL.aID][cAPPLICATION][cAPPLICATION_NAME]#"></TD>
</TR>
<TR>
<TD><B>Description:</B></TD>
<TD><CFINPUT TYPE="Text" NAME="appDescription" REQUIRED="No" SIZE="30"
MAXLENGTH="60"
VALUE="#aComponent[URL.aID][cAPPLICATION][cAPPLICATION_DESCRIPTION]#"><
/TD>
</TR>
<TR>
<TD><B>Notes:</B></TD>
<TD><CFINPUT TYPE="Text" NAME="appCargo1" REQUIRED="No" SIZE="30"
MAXLENGTH="60"
VALUE="#aComponent[URL.aID][cAPPLICATION][cAPPLICATION_CARGO1]#"></TD>
</TR>

<CFELSE>

<TR>
<TD><B>Type:</B></TD>
<TD><CFINPUT TYPE="Radio" NAME="comType" VALUE="file" CHECKED="#Iif(
aComponent[URL.aID][URL.comID][cCOMPONENT_TYPE] EQ 'file',true,false
) #">File</TD>
<TD><CFINPUT TYPE="Text" NAME="comFileLocation" REQUIRED="No" SIZE="30"
MAXLENGTH="200"
VALUE="#aComponent[URL.aID][URL.comID][cCOMPONENT_FILE]#"
onFOCUS="document.ActionForm.comType[0].checked = true;"></TD>

```

```

</TR>
<TR>
<TD>&nbsp;</TD>
<TD><CFINPUT TYPE="Radio" NAME="comType" VALUE="URL" CHECKED="#Iif(
aComponent[URL.aID][URL.comID][cCOMPONENT_TYPE] EQ 'URL',true,false
)#">URL</TD>
<TD><CFINPUT TYPE="Text" NAME="comURL" REQUIRED="No" SIZE="30"
MAXLENGTH="200"
VALUE="#aComponent[URL.aID][URL.comID][cCOMPONENT_URL]#"
onFOCUS="document.ActionForm.comType[1].checked = true;"></TD>
</TR>
<TR>
<TD>&nbsp;</TD>
<TD><CFINPUT TYPE="Radio" NAME="comType" VALUE="inline" CHECKED="#Iif(
aComponent[URL.aID][URL.comID][cCOMPONENT_TYPE] EQ 'inline',true,false
)#">Inline Code</TD>
<TD><CFINPUT TYPE="Text" NAME="comInlineCode" REQUIRED="No" SIZE="30"
MAXLENGTH="200"
VALUE="#aComponent[URL.aID][URL.comID][cCOMPONENT_INLINECODE]#"
onFOCUS="document.ActionForm.comType[2].checked = true;"></TD>
</TR>

</CFIF>

<TR>
<TD COLSPAN=3 ALIGN="center"><INPUT TYPE="submit" VALUE="Save"></TD>
</TR>
</TABLE>
</CFFORM>

</BODY>
</HTML>

```

ADMINWRITECONFIG.CFM

```

<!-- CREATE VARIABLE TO HOLD FILE TO WRITE TO. NOTE: THIS WILL
OVERWRITE! --->
<CFSET filetowrite = #GetDirectoryFromPath( GetTemplatePath() )# &
"getProgramDefaults.cfm">

<!-- BEGIN WRITING. NOTE: DATA WRITTEN DETERMINE WHAT URL VARIABLES
WAS USED. --->
<CFLOCK TIMEOUT="60">
<CFFILE ACTION="WRITE"
FILE="#filetowrite#"
OUTPUT="
<!-- PROGRAM SPECIFIC --->
<CFSET gvHOMEPAGE_NAME = '#VARIABLES.gvHOMEPAGE_NAME#'>

<!-- DATASOURCE DEFAULTS --->
<CFSET gvDATASOURCE_USER = 'reUSERS'>
<CFSET gvDATASOURCE_DATA = 'reDATA'>
<CFSET gvDATASOURCE_CONFIG = 'reCONFIG'>
<CFSET gvDATASOURCE_ARCHIVE = 'reARCHIVE'>

<!-- ADMINISTRATION --->

```

```

<CFSET gvAdminName                = '#VARIABLES.gvADMINNAME#'>
<CFSET gvAdminPhone                = '#VARIABLES.gvADMINPHONE#'>
<CFSET gvAdminEmail                = '#VARIABLES.gvADMINEMAIL#'>
<CFSET gvAdminName_Sec             = '#VARIABLES.gvADMINNAME_SEC#'>
<CFSET gvAdminPhone_Sec           = '#VARIABLES.gvADMINPHONE_SEC#'>
<CFSET gvAdminEmail_Sec           = '#VARIABLES.gvADMINEMAIL_SEC#'>

<!--- DIRECTORIES DEFAULTS --->
<CFSET gvFILEPATH                  = '#GetDirectoryFromPath(
GetTemplatePath() )#'>
<CFSET gvIMAGE_DIRECTORY           = 'images/'>
<CFSET gvJS_DIRECTORY              = 'js/'>
<CFSET gvUSER_DIRECTORY            = 'users/'>
<CFSET gvRESOURCE_DIRECTORY        = '#VARIABLES.gvRESOURCE_DIRECTORY#'>

<!--- RESOURCES --->
<CFSET gvMS_RESOURCE_CONVERTERS    =
'#VARIABLES.gvMS_RESOURCE_CONVERTERS#'>

<!--- SECURITY ACCESS DEFAULTS --->
<CFSET gvLEVEL_ADMIN               = '10'>

<!--- ORGANIZATION --->
<CFSET gvAPP_HOME                  = '#VARIABLES.gvAPP_HOME#'>
<CFSET gvAPP_COMMAND               = '#VARIABLES.gvAPP_COMMAND#'>
<CFSET gvHOME_PAGE                 = '#VARIABLES.gvHOME_PAGE#'>
<CFSET gvSMTP_SERVER               = '#VARIABLES.gvSMTP_SERVER#'>
<CFSET gvFromEmail                 = '#VARIABLES.gvFROMEMAIL#'>

<!--- SECURITY --->
<CFSET gvTIMEOUT_MINS               = '#VARIABLES.gvTIMEOUT_MINS#'>
<CFSET gvTIMEOUT_SECS              = '#Evaluate( ( VARIABLES.gvTIMEOUT_MINS *
60 ) - 60 )#'>
<CFSET gvSECURITYEMAIL             = '#VARIABLES.gvSECURITYEMAIL#'>
<CFSET gvLOGON_ATTEMPTS            = '#VARIABLES.gvLOGON_ATTEMPTS#'>
<CFSET gvIP_BLOCKING               = '#VARIABLES.gvIP_BLOCKING#'>
<CFSET gvDOW_BLOCKING              = '#VARIABLES.gvDOW_BLOCKING#'>
<CFSET gvDOMAIN_BLOCKING           = '#VARIABLES.gvDOMAIN_BLOCKING#'>
<CFSET gvDOMAIN_BLOCKINGLIST       = '#VARIABLES.gvDOMAIN_BLOCKINGLIST#'>
<CFSET gvDOW_DAYSTOBLOCK           = '#VARIABLES.gvDOW_DAYSTOBLOCK#'>
<CFSET gvTIME_BLOCKING             = '#VARIABLES.gvTIME_BLOCKING#'>
<CFSET gvTIME_FROMBLOCK            = '#VARIABLES.gvTIME_FROMBLOCK#'>
<CFSET gvTIME_TOBLOCK              = '#VARIABLES.gvTIME_TOBLOCK#'>
<CFSET gvDEFAULTPSWDEXP            = '#VARIABLES.gvDEFAULTPSWDEXP#'>
<CFSET gvMIN_PSWD_SIZE             = '#VARIABLES.gvMIN_PSWD_SIZE#'>
<CFSET gvPSWD_REUSE                = '#VARIABLES.gvPSWD_REUSE#'>
<CFSET gvPSWD_FORGET               = '#VARIABLES.gvPSWD_FORGET#'>
<CFSET gvPSWD_REMEMBER             = '#VARIABLES.gvPSWD_REMEMBER#'>
<CFSET gvPSWD_FORGET_CHG           = '#VARIABLES.gvPSWD_FORGET_CHG#'>
<CFSET gvPSWD_LOCKIFFAIL           = '#VARIABLES.gvPSWD_LOCKIFFAIL#'>
<CFSET gvPSWD_SENDEMAIL            = '#VARIABLES.gvPSWD_SENDEMAIL#'>
<CFSET gvPSWD_ISCASE               = '#VARIABLES.gvPSWD_ISCASE#'>
<CFSET gvPSWD_INCLNUM              = '#VARIABLES.gvPSWD_INCLNUM#'>
<CFSET gvMSG_FAILEDLOGON           = '#VARIABLES.gvMSG_FAILEDLOGON#'>
<CFSET gvMSG_USEREXPIRE            = '#VARIABLES.gvMSG_USEREXPIRE#'>
<CFSET gvMSG_RECORDLOCKED          = '#VARIABLES.gvMSG_RECORDLOCKED#'>
<CFSET gvMSG_IPBLOCKING            = '#VARIABLES.gvMSG_IPBLOCKING#'>

```

```

<CFSET gvMSG_DOMAINBLOCKING = '#VARIABLES.gvMSG_DOMAINBLOCKING#'>
<CFSET gvMSG_DOWBLOCKING = '#VARIABLES.gvMSG_DOWBLOCKING#'>
<CFSET gvMSG_TIMEBLOCKING = '#VARIABLES.gvMSG_TIMEBLOCKING#'>

<!-- SECURITY ACCESS DEFAULTS -->
<CFSET gvLEVEL_ADMIN = '10'>

">
</CFLOCK>

```

3. SYSTEM ADMINISTRATION FILES

ADMINLISTINTERFACES.CFM

```

<CFPARAM NAME="URL.action" DEFAULT="edit">
<CFSET mainFile = "main.cfm?a=tools&body=adminListInterfaces.cfm">
<CFSET mainTable = "tblInterface_ODBC">
<CFSET idField = "interface_odbc_id">
<CFSET editTitle = "Interfaces">
<CFSET useSource = "#VARIABLES.gvDATASOURCE_CONFIG#">

<CFSET packet = "Interfaces">
<CFSET packetList = "InterfaceList">

<CFIF NOT #CompareNoCase( URL.action, "View" )#>

<CF_srReportEngine
ACTION="run"
NAME="listInterfaces"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="New"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=add"
>

<CFELSEIF NOT #CompareNoCase( URL.action, "Edit" )#>

<CFSET formAction = "#VARIABLES.mainFile#&action=save">
<CFSET formName="formEdit">

<CFELSEIF NOT #CompareNoCase( URL.action, "Save" )#>

<CFWDDX
INPUT="#FORM.WDDXContent#"
OUTPUT="Packet"
ACTION="WDDX2CFML"
>
<CFSET insertCount = 0>
<CFSET updateCount = 0>
<CFSET deleteCount = 0>

<CFLOOP QUERY="Packet">

<CFIF NOT #CompareNoCase( Evaluate( "Packet." & VARIABLES.idField
), "new" )#>

```

```

<CFTRANSACTION>

<!-- insert fields into field table --->
<CFQUERY DATASOURCE="#VARIABLES.useSource#">

INSERT INTO #VARIABLES.mainTable# (
io_name,io_odbc,io_table,io_read,io_write,io_isLocal,io_server,io_server_port,io_server_login,io_server_password,io_description,io_linkedField
)
VALUES (
'#io_name#','#io_odbc#','#io_table#','#io_read#','#io_write#','#io_isLocal#',
'#io_server#','#io_server_port#','#io_server_login#','#io_server_password#',
'#io_description#','#io_linkedField#' )

</CFQUERY>

</CFTRANSACTION>

<CFSET insertCount = insertCount + 1>

<CFELSEIF NOT #CompareNoCase( Packet.wasEdited,"Delete" )#>

<CFQUERY DATASOURCE="#VARIABLES.useSource#">

DELETE FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = #Evaluate( VARIABLES.idField )#

</CFQUERY>

<CFSET deleteCount = deleteCount + 1>

<CFELSEIF NOT #CompareNoCase( Packet.wasEdited,"Yes" )#>

<CFQUERY DATASOURCE="#VARIABLES.useSource#">

UPDATE #VARIABLES.mainTable#
SET io_name = '#io_name#',
io_odbc = '#io_odbc#',
io_table = '#io_table#',
io_read = #io_read#,
io_write = #io_write#,
io_isLocal = #io_isLocal#,
io_server = '#io_server#',
io_server_port = '#io_server_port#',
io_server_login = '#io_server_login#',
io_server_password = '#io_server_password#',
io_description = '#io_description#',
io_linkedfield = '#io_linkedfield#'

WHERE #VARIABLES.idField# = #Evaluate( VARIABLES.idField )#

</CFQUERY>

<CFSET updateCount = updateCount + 1>

</CFIF>

</CFLOOP>

```

```

<CFLOCATION URL="#VARIABLES.mainFile#">

</CFIF>

<CFIF #CompareNoCase( URL.action,"view" )#>

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

SELECT      *
FROM #VARIABLES.mainTable#
ORDER BY io_name;

</CFQUERY>

<SCRIPT LANGUAGE="javascript" SRC="wddx.js"></SCRIPT>
<SCRIPT LANGUAGE="JAVASCRIPT">

<CFOUTPUT>
function onSubmit( destination )
{
document.#VARIABLES.formName#.action = destination;
document.#VARIABLES.formName#.submit();
}

<!-- Convert query to Javascript object --->
<CFWDDX
ACTION="CFML2JS"
INPUT="#getRecordInfo#"
TOPLEVELVARIABLE="#VARIABLES.packet#"
>

// Add a column called "wasedited" to the recordset
// A "Yes" in this column means the row was "touched"
#VARIABLES.packet#.addColumn("wasedited");

////////////////////////////////////
// This function fills the SELECT list with users
function initControls()
{
with (document.#VARIABLES.formName#)
{
// Clear any current OPTIONS from the SELECT
#VARIABLES.packetList#.options.length = 0;

// For each record...
for (var i = 0; i < #VARIABLES.packet#.getRowCount(); i++)
{

// Create a new OPTION object
NewOpt = new Option;
NewOpt.value = #VARIABLES.packet#.interface_odbc_id[i];
NewOpt.text = #VARIABLES.packet#.io_name[i];

// Add the new object to the SELECT list
#VARIABLES.packetList#.options[#VARIABLES.packetList#.options.length] =
NewOpt;

```

```

}
//UserList.selectedIndex = 0;
//fillControls();
}
}

function fillControls()
{
with (document.#VARIABLES.formName#)
{
// Add one to the OPTION number to get the data row number
var RowNum = #VARIABLES.packetList#.selectedIndex;

// Populate textboxes with data in that row
io_name.value = #VARIABLES.packet#.io_name[RowNum];
io_description.value = #VARIABLES.packet#.io_description[RowNum];
io_linkedfield.value = #VARIABLES.packet#.io_linkedfield[RowNum];
io_odbc.value = #VARIABLES.packet#.io_odbc[RowNum];
io_table.value = #VARIABLES.packet#.io_table[RowNum];
io_server.value = #VARIABLES.packet#.io_server[RowNum];
io_server_port.value = #VARIABLES.packet#.io_server_port[RowNum];
io_server_login.value = #VARIABLES.packet#.io_server_login[RowNum];
io_server_password.value =
#VARIABLES.packet#.io_server_password[RowNum];

if ( #VARIABLES.packet#.io_read[RowNum] != 0 )
io_Read[0].checked = true;
else
io_Read[1].checked = true;
if ( #VARIABLES.packet#.io_write[RowNum] != 0 )
io_Write[0].checked = true;
else
io_Write[1].checked = true;
if ( #VARIABLES.packet#.io_islocal[RowNum] != 0 )
io_isLocal[0].checked = true;
else
io_isLocal[1].checked = true;
}
}

function newRecord()
{
with (document.#VARIABLES.formName#)
{
// Add a new row to the recordset
#VARIABLES.packet#.addRows(1);
NewRow = #VARIABLES.packet#.getRowCount()-1;

#VARIABLES.packet#.setField(NewRow, "interface_odbc_id", "new");
#VARIABLES.packet#.setField(NewRow, "io_name", "(New Field)");
#VARIABLES.packet#.setField(NewRow, "io_odbc", "");
#VARIABLES.packet#.setField(NewRow, "io_linkedfield", "");
#VARIABLES.packet#.setField(NewRow, "io_description", "");
#VARIABLES.packet#.setField(NewRow, "io_table", "");
#VARIABLES.packet#.setField(NewRow, "io_server", "");
#VARIABLES.packet#.setField(NewRow, "io_server_port", "");

```

```

#VARIABLES.packet#.setField(NewRow, "io_server_login", "");
#VARIABLES.packet#.setField(NewRow, "io_server_password", "");
#VARIABLES.packet#.setField(NewRow, "io_read", "1");
#VARIABLES.packet#.setField(NewRow, "io_write", "0");
#VARIABLES.packet#.setField(NewRow, "io_islocal", "1");

// Re-initialize the SELECT list
initControls();

// Re-select the book that was selected before
#VARIABLES.packetList#.selectedIndex = NewRow;
fillControls();
}
}

function savetoServer()
{
with (document.#VARIABLES.formName#)
{
mySerializer = new WddxSerializer();
#VARIABLES.packet#AsWDDX = mySerializer.serialize( #VARIABLES.packet#
);
WDDXContent.value = #VARIABLES.packet#AsWDDX;
submit();
}
}

function saveChanges()
{
with (document.#VARIABLES.formName#)
{
var selectedProfile = #VARIABLES.packetList#.selectedIndex;
var RowNum = selectedProfile;

// Populate textboxes with data in that row
#VARIABLES.packet#.io_name[RowNum] = io_name.value;
#VARIABLES.packet#.io_odbc[RowNum] = io_odbc.value;
#VARIABLES.packet#.io_linkedfield[RowNum] = io_linkedfield.value;
#VARIABLES.packet#.io_description[RowNum] = io_description.value;
#VARIABLES.packet#.io_table[RowNum] = io_table.value;
#VARIABLES.packet#.io_server[RowNum] = io_server.value;
#VARIABLES.packet#.io_server_port[RowNum] = io_server_port.value;
#VARIABLES.packet#.io_server_login[RowNum] = io_server_login.value;
#VARIABLES.packet#.io_server_password[RowNum] =
io_server_password.value;

if ( io_Read[0].checked )
#VARIABLES.packet#.io_read[RowNum] = io_Read[0].value;
else
#VARIABLES.packet#.io_read[RowNum] = io_Read[1].value;
if ( io_Write[0].checked )
#VARIABLES.packet#.io_write[RowNum] = io_Write[0].value;
else
#VARIABLES.packet#.io_write[RowNum] = io_Write[1].value;
if ( io_isLocal[0].checked )
#VARIABLES.packet#.io_islocal[RowNum] = io_isLocal[0].value;
else

```

```

#VARIABLES.packet#.io_islocal[RowNum] = io_isLocal[1].value;

#VARIABLES.packet#.wasedited[RowNum] = 'Yes';
initControls();

#VARIABLES.packetList#.selectedIndex = -1;
#VARIABLES.packetList#.selectedIndex = selectedProfile;
}
}

function deleteRecord()
{
with (document.#VARIABLES.formName#)
{
var selectedRecord = #VARIABLES.packetList#.selectedIndex;
var RowNum = selectedRecord;

// Populate textboxes with data in that row
#VARIABLES.packet#.io_name[RowNum] = "MARKED FOR DELETION";
io_name.value = #VARIABLES.packet#.io_name[RowNum];

#VARIABLES.packet#.wasedited[RowNum] = 'Delete';

initControls();

#VARIABLES.packetList#.selectedIndex = -1;
#VARIABLES.packetList#.selectedIndex = selectedRecord;
}
}

window.onload=initControls;

</CFOUTPUT>
</SCRIPT>

<!-- display appropriate header --->
<!--#getProfileInfo.pgName#"-->
<CF_srWinTab
TABS="Profile Groups,Group Fields,Field Rules,Lists,Interfaces"
TABSELECTED="5"
TABURL="main.cfm?a=tools&body=adminListProfiles.cfm,main.cfm?a=tools&bo
dy=adminListProfileFields.cfm,main.cfm?a=tools&body=adminListProfileFie
ldsRules.cfm,main.cfm?a=tools&body=adminListProfileLists.cfm,main.cfm?a
=tools&body=adminListInterfaces.cfm"
SYSTEM_TOOLBAR="New,Save,Save to Server,Delete,Undo"
SYSTEM_TOOLBARLINKS="javascript:newRecord(),javascript:saveChanges(),ja
vascript:savetoServer(),javascript:deleteRecord(),javascript:fillContro
ls()"
>

<TABLE BORDER=0 WIDTH=100%>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<INPUT TYPE="hidden" NAME="WDDXContent" VALUE="">
<TR>
<TD>
<SPAN CLASS="winTab">

```



```

</TR>
<TR>
<TD CLASS="winTabField">Read Data?</TD>
<TD CLASS="winTabField">
<INPUT NAME="io_Read" TYPE="radio" VALUE="1">Yes
<INPUT NAME="io_Read" TYPE="radio" VALUE="0">No
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Write Data?</TD>
<TD CLASS="winTabField">
<INPUT NAME="io_Write" TYPE="radio" VALUE="1">Yes
<INPUT NAME="io_Write" TYPE="radio" VALUE="0">No
</TD>
</TR>
<TR>
<TD CLASS="winTabField" COLSPAN=2><B>SERVER</B></TD>
</TR>
<TR>
<TD CLASS="winTabField">Data on Local Server?</TD>
<TD CLASS="winTabField">
<INPUT NAME="io_isLocal" TYPE="radio" VALUE="1">Yes
<INPUT NAME="io_isLocal" TYPE="radio" VALUE="0">No
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Address:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="io_server" TYPE="text" SIZE="30"
MAXLENGTH="50">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Port Number:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="io_server_port" TYPE="text" SIZE="10"
MAXLENGTH="5">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Login ID:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="io_server_login" TYPE="text" SIZE="30"
MAXLENGTH="20">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Password:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="io_server_password" TYPE="password"
SIZE="30" MAXLENGTH="20">
</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>

```

```
</TD>
</TR>
</CFFORM>
</TABLE>

</CF_srWinTab>

</CFIF>
```

ADMINLISTPANELCOMPONENTS.CFM

```
<CFPARAM NAME="URL.action" DEFAULT="View">
<CFSET mainFile = "main.cfm?a=tools&body=adminListPanelComponents.cfm">
<CFSET mainTable = "tblPanelComponents">
<CFSET idField = "panelComponent_id">
<CFSET addTitle = "CREATE A NEW PANEL COMPONENT">
<CFSET editTitle = "EDIT PANEL COMPONENT">

<CFIF URL.action EQ "View">

<CF_srReportEngine
ACTION="run"
NAME="listPanelComponents"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="New"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=add"
>

<CFELSEIF URL.action EQ "Add">

<CFSET title = "#VARIABLES.addTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=saveNew">
<CFSET formName="formAdd">

<CFELSEIF URL.action EQ "Edit">

<CFSET title = "#VARIABLES.editTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=save">
<CFSET formName="formEdit">

<CFELSEIF URL.action EQ "saveNew">

<CFINSERT DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#"
TABLENAME="#VARIABLES.mainTable#">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "save">

<CFUPDATE DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#"
TABLENAME="#VARIABLES.mainTable#">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "delete">
```

```

<CFQUERY NAME="deleteInfo"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

DELETE
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>
<CFLOCATION URL="#VARIABLES.mainFile#">

</CFIF>

<CFIF URL.action NEQ "View">

<CFQUERY NAME="getRecordInfo"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      *
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = <CFIF URL.action EQ "Edit">#Evaluate(
'URL.' & VARIABLES.idField )#<CFELSE>-1</CFIF>

</CFQUERY>

<!-- display appropriate header --->
<CF_srWinTab
TABS="#VARIABLES.title#"
SYSTEM_TOOLBAR="Save,List Panel Components"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit(),
#VARIABLES.mainFile#"
>

<TABLE BORDER=0 WIDTH=100%>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<CFOUTPUT>
<INPUT TYPE="hidden" NAME="#VARIABLES.idField#" VALUE="#Evaluate(
'getRecordInfo.' & VARIABLES.idField )#">
</CFOUTPUT>
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="adminListPanelComponents">
</SPAN>
<P>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel">&nbsp;Title:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="pcTitle"
TYPE="text"
VALUE="#getRecordInfo.pcTitle#"
SIZE="30"
MAXLENGTH="30"
REQUIRED="yes"
MESSAGE="A Title Name must be entered"

```

```

>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Information Method:</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="pcOption" VALUE="1" CHECKED="#Iif(
getRecordInfo.pcOption EQ 1,De( 'true' ),De( 'false' ) )#">File
<CFINPUT TYPE="Radio" NAME="pcOption" VALUE="2" CHECKED="#Iif(
getRecordInfo.pcOption EQ 2,De( 'true' ),De( 'false' ) )#">Web Link
<CFINPUT TYPE="Radio" NAME="pcOption" VALUE="3" CHECKED="#Iif(
getRecordInfo.pcOption EQ 3,De( 'true' ),De( 'false' ) )#">Inline Code
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;File Name:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="pcFileName"
TYPE="text"
VALUE="#getRecordInfo.pcFileName#"
SIZE="30"
MAXLENGTH="30"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Web Link:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="pcLink"
TYPE="text"
VALUE="#getRecordInfo.pcLink#"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Inline Code:</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabField" COLSPAN=2>
<TEXTAREA COLS=50 ROWS=5 NAME="pcInline"
WRAP="soft"><CFOUTPUT>#getRecordInfo.pcInline#</CFOUTPUT></TEXTAREA>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</CFFORM>
</TABLE>

</CF_srWinTab>

```

</CFIF>

ADMINLISTPROFILEFIELDS.CFM

```
<CFPARAM NAME="URL.action" DEFAULT="edit">
<CFSET mainFile = "main.cfm?a=tools&body=adminListProfileFields.cfm">
<CFSET mainTable = "tblProfileFields">
<CFSET idField = "pf_id">
<CFSET editTitle = "Profile Fields">
<CFSET useSource = "#VARIABLES.gvDATASOURCE_CONFIG#">
<CFSET packet = "ProfileFields">
<CFSET packetList = "ProfileFieldList">

<CFIF URL.action EQ "View">

<CF_srReportEngine
ACTION="run"
NAME="listProfileFields"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="New"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=add"
>

<CFELSEIF URL.action EQ "Edit">

<CFSET formAction = "#VARIABLES.mainFile#&action=save">
<CFSET formName="formEdit">

<CFELSEIF URL.action EQ "save">

<CFWDDX
INPUT="#FORM.WDDXContent#"
OUTPUT="Packet"
ACTION="WDDX2CFML"
>
<CFSET insertCount = 0>
<CFSET updateCount = 0>
<CFSET deleteCount = 0>

<CFLOOP QUERY="Packet">

<CFIF Packet.pf_id EQ "new">

<CFTRANSACTION>

<!-- insert fields into field table --->
<CFQUERY DATASOURCE="#VARIABLES.useSource#">

INSERT INTO tblProfileFields (
pfName,pfLabel,pfSize,pfDescription,pfCstatus,pfType,pfValidationType,p
fFailColor,pfEncrypt,pfIncludeInSearch,pfrequiredField,pfRange,pfHelpMe
ssage,pfHelpSummary,pfhasInterface,pfInterfaceName )
VALUES (
'#pfName#','#pfLabel#','#pfSize#','#pfDescription#','#pfCstatus#','#pfType#,
```

```
#pfValidationType#,'#pfFailColor#','#pfEncrypt#','#pfIncludeInSearch#','#pfr  
equiredField#,'#pfRange#','#pfHelpMessage#','#pfHelpSummary#','#pfhasInt  
erface#,'#pfInterfaceName#' )
```

```
</CFQUERY>
```

```
<!--- get new id for field --->
```

```
<CFQUERY NAME="getNewID" DATASOURCE="#VARIABLES.useSource#">
```

```
SELECT      Max( pf_id ) as NewID  
FROM    tblProfileFields
```

```
</CFQUERY>
```

```
<!--- link group and new field --->
```

```
<CFQUERY DATASOURCE="#VARIABLES.useSource#">
```

```
INSERT INTO tblLinkFieldToProfile ( pg_id,pf_id )  
VALUES ( #FORM.pg_id#,#getNewID.NewID# )
```

```
</CFQUERY>
```

```
</CFTRANSACTION>
```

```
<CFTRANSACTION>
```

```
<!--- now alter group table by adding new field requirement --->
```

```
<CFSET tableName = "tbl" & #FORM.pgName#>
```

```
<CFQUERY DATASOURCE="#VARIABLES.gvDATASOURCE_DATA#">
```

```
ALTER TABLE #VARIABLES.tableName#
```

```
<CFSWITCH EXPRESSION="#FORM.pfType#">
```

```
<CFCASE VALUE="1">
```

```
ADD COLUMN #FORM.pfName# varchar (#FORM.pfSize#);
```

```
</CFCASE>
```

```
<CFCASE VALUE="2">
```

```
ADD COLUMN #FORM.pfName# BIT;
```

```
</CFCASE>
```

```
<CFCASE VALUE="3">
```

```
ADD COLUMN #FORM.pfName# DATETIME;
```

```
</CFCASE>
```

```
<CFCASE VALUE="4">
```

```
ADD COLUMN #FORM.pfName# SHORT;
```

```
</CFCASE>
```

```
<CFCASE VALUE="6">
```

```
ADD COLUMN #FORM.pfName# LONGTEXT;
```

```
</CFCASE>
```

```
</CFSWITCH>
```

```
</CFQUERY>
```

```
</CFTRANSACTION>
```

```
<CFSET insertCount = insertCount + 1>
```

```
<CFELSEIF Packet.wasEdited EQ "Delete">
```

```

<CFQUERY DATASOURCE="#VARIABLES.useSource#">

DELETE FROM tblProfileGroups
WHERE pg_id = #pg_id#

</CFQUERY>

<CFSET tableName = "tbl" & #pgName#>
<CFQUERY DATASOURCE="#VARIABLES.gvDATASOURCE_DATA#">

DROP TABLE #VARIABLES.tableName#

</CFQUERY>

<CFSET deleteCount = deleteCount + 1>

<CFELSEIF Packet.wasEdited EQ "Yes">

<CFQUERY DATASOURCE="#VARIABLES.useSource#">

UPDATE tblProfileFields
SET pfName = '#pfname#',
pfDescription = '#pfdescription#',
pfLabel = '#pflabel#',
pfSize = #pfsize#,
pfCstatus = #pfcstatus#,
pfType = #pftype#,
pfFailColor = '#pffailcolor#',
pfEncrypt = #pfencrypt#,
pfIncludeInSearch = #pfincludeinsearch#,
pfRequiredField = #pfrequiredfield#,
pfRange = '#pfrange#',
pfValidationType = #pfvalidationtype#,
pfHelpMessage = '#pfhelpmessage#',
pfHelpSummary = '#pfhelpsummary#',
pfhasInterface = #pfhasinterface#,
pfInterfaceName = '#pfinterfacename#'

WHERE pf_id = #pf_id#

</CFQUERY>
<CFSET updateCount = updateCount + 1>

</CFIF>

</CFLOOP>

<CFLOCATION URL="#VARIABLES.mainFile#&pg_id=#FORM.pg_id#">

</CFIF>

<CFIF URL.action NEQ "view">

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

SELECT
    tblProfileFields.pf_id,pfName,pfLabel,pfSize,pfDescription,pfCsta

```

```

tus, pfType, pfValidationType, pfFailColor, pfEncrypt, pfIncludeInSearch, pfRequiredField, pfRange, pfHelpMessage, pfHelpSummary, pfhasInterface, pfInterfaceName
FROM ( tblLinkFieldToProfile INNER JOIN tblProfileFields ON
tblLinkFieldToProfile.pf_id = tblProfileFields.pf_id) INNER JOIN
tblProfileGroups ON tblLinkFieldToProfile.pg_id =
tblProfileGroups.pg_id
WHERE      tblProfileGroups.pg_id = <CFIF IsDefined(
"FORM.ProfileList" )>#FORM.ProfileList#<CFELSE>#URL.pg_id#</CFIF>
ORDER BY  pfName;

```

```

</CFQUERY>
<CFQUERY NAME="getProfileInfo" DATASOURCE="#VARIABLES.useSource#">

```

```

SELECT      pg_id, pgName
FROM      tblProfileGroups
WHERE pg_id = <CFIF IsDefined( "FORM.ProfileList"
)>#FORM.ProfileList#<CFELSE>#URL.pg_id#</CFIF>

```

```

</CFQUERY>

```

```

<!---
<CFQUERY NAME="getInterfaces" DATASOURCE="#VARIABLES.useSource#">

```

```

SELECT      interface_odbc_id, io_name
FROM      tblInterface_ODBC
ORDER BY   io_name

```

```

</CFQUERY>

```

```

--->

```

```

<SCRIPT LANGUAGE="javascript" SRC="wddx.js"></SCRIPT>
<SCRIPT LANGUAGE="JAVASCRIPT">

```

```

<CFOUTPUT>
function onSubmit( destination )
{
document.#VARIABLES.formName#.action = destination;
document.#VARIABLES.formName#.submit();
}

```

```

<!--- Convert query to Javascript object --->

```

```

<CFWDDX
ACTION="CFML2JS"
INPUT="#getRecordInfo#"
TOPLEVELVARIABLE="#VARIABLES.packet#"
>

```

```

// Add a column called "wasedited" to the recordset
// A "Yes" in this column means the row was "touched"
#VARIABLES.packet#.addColumn("wasedited");

```

```

////////////////////////////////////

```

```

// This function fills the SELECT list with users

```

```

function initControls()
{
with (document.#VARIABLES.formName#)

```

```

{
// Clear any current OPTIONS from the SELECT
#VARIABLES.packetList#.options.length = 0;

// For each record...
for (var i = 0; i < #VARIABLES.packet#.getRowCount(); i++)
{

// Create a new OPTION object
NewOpt = new Option;
NewOpt.value = #VARIABLES.packet#.pf_id[i];
NewOpt.text = #VARIABLES.packet#.pflabel[i];

// Add the new object to the SELECT list
#VARIABLES.packetList#.options[#VARIABLES.packetList#.options.length] =
NewOpt;

}

//UserList.selectedIndex = 0;
//fillControls();
}
}

function fillControls()
{
with (document.#VARIABLES.formName#)
{
// Add one to the OPTION number to get the data row number
var RowNum = #VARIABLES.packetList#.selectedIndex;

// Populate textboxes with data in that row
pg_id.value = "#getProfileInfo.pg_id#";
pgName.value = "#getProfileInfo.pgName#";

pfName.value = #VARIABLES.packet#.pfname[RowNum];
pfLabel.value = #VARIABLES.packet#.pflabel[RowNum];
pfSize.value = #VARIABLES.packet#.pfsize[RowNum];
pfDescription.value = #VARIABLES.packet#.pfdescription[RowNum];
pfFailColor.value = #VARIABLES.packet#.pffailcolor[RowNum];
pfRange.value = #VARIABLES.packet#.pfrange[RowNum];
pfHelpMessage.value = #VARIABLES.packet#.pfhelpmessage[RowNum];
pfHelpSummary.value = #VARIABLES.packet#.pfhelpsummary[RowNum];
pfinterfacename.value = #VARIABLES.packet#.pfinterfacename[RowNum];

for ( n=0;n<=6;n++ )
if ( #VARIABLES.packet#.pfcstatus[RowNum] == n+1)
pfCStatus[n].checked = true;

for ( n=0;n<6;n++ )
if ( #VARIABLES.packet#.pftype[RowNum] == n+1)
pfType[n].checked = true;

for ( n=0;n<7;n++ )
if ( #VARIABLES.packet#.pfvalidationtype[RowNum] == n+1)
pfValidationType[n].checked = true;
}
}

```

```

if ( #VARIABLES.packet#.pfencrypt[RowNum] != 0 )
pfEncrypt[0].checked = true;
else
pfEncrypt[1].checked = true;
if ( #VARIABLES.packet#.pfincludeinsearch[RowNum] != 0 )
pfIncludeInSearch[0].checked = true;
else
pfIncludeInSearch[1].checked = true;
if ( #VARIABLES.packet#.pfrequiredfield[RowNum] != 0 )
pfRequiredField[0].checked = true;
else
pfRequiredField[1].checked = true;
if ( #VARIABLES.packet#.pfhasinterface[RowNum] != 0 )
pfhasinterface[0].checked = true;
else
pfhasinterface[1].checked = true;
}
}

function newRecord()
{
with (document.#VARIABLES.formName#)
{
// Add a new row to the recordset
#VARIABLES.packet#.addRows(1);
NewRow = #VARIABLES.packet#.getRowCount()-1;

#VARIABLES.packet#.setField(NewRow, "pf_id", "new");
#VARIABLES.packet#.setField(NewRow, "pfname", "(New Field)");
#VARIABLES.packet#.setField(NewRow, "pflabel", "(New Field)");
#VARIABLES.packet#.setField(NewRow, "pfsize", "10");
#VARIABLES.packet#.setField(NewRow, "pfdescription", "");
#VARIABLES.packet#.setField(NewRow, "pfcstatus", "6");
#VARIABLES.packet#.setField(NewRow, "pftype", "1");
#VARIABLES.packet#.setField(NewRow, "pfvalidationtype", "7");
#VARIABLES.packet#.setField(NewRow, "pffailcolor", "red");
#VARIABLES.packet#.setField(NewRow, "pfencrypt", "0");
#VARIABLES.packet#.setField(NewRow, "pfincludeinsearch", "0");
#VARIABLES.packet#.setField(NewRow, "pfrequiredfield", "0");
#VARIABLES.packet#.setField(NewRow, "pfrange", "0,0");
#VARIABLES.packet#.setField(NewRow, "pfhelpmessage", "");
#VARIABLES.packet#.setField(NewRow, "pfhelpsummary", "");
#VARIABLES.packet#.setField(NewRow, "pfhasinterface", "0");
#VARIABLES.packet#.setField(NewRow, "pfinterfacename", "");

// Re-initialize the SELECT list
initControls();

// Re-select the book that was selected before
#VARIABLES.packetList#.selectedIndex = NewRow;
fillControls();
}
}

function savetoServer()
{
with (document.#VARIABLES.formName#)

```

```

{
mySerializer = new WddxSerializer();
#VARIABLES.packet#AsWDDX = mySerializer.serialize( #VARIABLES.packet#
);
WDDXContent.value = #VARIABLES.packet#AsWDDX;
submit();
}
}

function saveChanges()
{
with (document.#VARIABLES.formName#)
{
var selectedProfile = #VARIABLES.packetList#.selectedIndex;
var RowNum = selectedProfile;

// Populate textboxes with data in that row
#VARIABLES.packet#.pfname[RowNum] = pfName.value;
#VARIABLES.packet#.pflabel[RowNum] = pfLabel.value;
#VARIABLES.packet#.pfsizel[RowNum] = pfSize.value;
#VARIABLES.packet#.pfdescription[RowNum] = pfDescription.value;
#VARIABLES.packet#.pffailcolor[RowNum] = pfFailColor.value;

#VARIABLES.packet#.pfrange[RowNum] = pfRange.value;
#VARIABLES.packet#.pfhelpmessage[RowNum] = pfHelpMessage.value;
#VARIABLES.packet#.pfhelpsummary[RowNum] = pfHelpSummary.value;
#VARIABLES.packet#.pfinterfacename[RowNum] = pfinterfacename.value;

for ( n=0;n<6;n++ )
if ( pfCStatus[n].checked )
#VARIABLES.packet#.pfcstatus[RowNum] = pfCStatus[n].value;

for ( n=0;n<6;n++ )
if ( pfType[n].checked )
#VARIABLES.packet#.pftype[RowNum] = pfType[n].value;

for ( n=0;n<7;n++ )
if ( pfValidationType[n].checked )
#VARIABLES.packet#.pfvalidationtype[RowNum] =
pfValidationType[n].value;

if ( pfEncrypt[0].checked )
#VARIABLES.packet#.pfencrypt[RowNum] = pfEncrypt[0].value;
else
#VARIABLES.packet#.pfencrypt[RowNum] = pfEncrypt[1].value;

if ( pfIncludeInSearch[0].checked )
#VARIABLES.packet#.pfincludeinsearch[RowNum] =
pfIncludeInSearch[0].value;
else
#VARIABLES.packet#.pfincludeinsearch[RowNum] =
pfIncludeInSearch[1].value;

if ( pfRequiredField[0].checked )
#VARIABLES.packet#.pfrequiredfield[RowNum] = pfRequiredField[0].value;
else
#VARIABLES.packet#.pfrequiredfield[RowNum] = pfRequiredField[1].value;

```

```

if ( pfhasinterface[0].checked )
#VARIABLES.packet#.pfhasinterface[RowNum] = pfhasinterface[0].value;
else
#VARIABLES.packet#.pfhasinterface[RowNum] = pfhasinterface[1].value;

#VARIABLES.packet#.wasedited[RowNum] = 'Yes';
initControls();

#VARIABLES.packetList#.selectedIndex = -1;
#VARIABLES.packetList#.selectedIndex = selectedProfile;
}
}

function deleteRecord()
{
with (document.#VARIABLES.formName#)
{
var selectedProfile = #VARIABLES.packetList#.selectedIndex;
var RowNum = selectedProfile;

// Populate textboxes with data in that row
#VARIABLES.packet#.pflabel[RowNum] = "MARKED FOR DELETION";
pfLabel.value = #VARIABLES.packet#.pflabel[RowNum];

#VARIABLES.packet#.wasedited[RowNum] = 'Delete';

initControls();

#VARIABLES.packetList#.selectedIndex = -1;
#VARIABLES.packetList#.selectedIndex = selectedProfile;
}
}

window.onload=initControls;

</CFOUTPUT>
</SCRIPT>

<!-- display appropriate header -->
<!--#getProfileInfo.pgName#"-->
<CF_srWinTab
TABS="Profile Groups,Group Fields: #getProfileInfo.pgName#,Field
Rules,Lists,Interfaces"
TABSELECTED="2"
TABURL="javascript:onSubmit(
'main.cfm?a=tools&body=adminListProfiles.cfm' ),javascript:onSubmit(
'main.cfm?a=tools&body=adminListProfileFields.cfm'
),javascript:onSubmit(
'main.cfm?a=tools&body=adminListProfileFieldsRules.cfm'
),javascript:onSubmit(
'main.cfm?a=tools&body=adminListProfileLists.cfm'
),javascript:onSubmit( 'main.cfm?a=tools&body=adminListInterfaces.cfm'
)"
SYSTEM_TOOLBAR="New,Save,Save to Server,Delete,Undo"

```



```

<TEXTAREA CLASS="stdTextBox" COLS=30 ROWS=3 NAME="pfDescription"
WRAP="soft"></TEXTAREA>
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Type:</TD>
<TD CLASS="winTabField">
<INPUT NAME="pfType" TYPE="radio" VALUE="1">Text
<INPUT NAME="pfType" TYPE="radio" VALUE="2">T/F
<INPUT NAME="pfType" TYPE="radio" VALUE="3">Date
</TD>
</TR>
<TR>
<TD CLASS="winTabField">&nbsp;</TD>
<TD CLASS="winTabField">
<INPUT NAME="pfType" TYPE="radio" VALUE="4">Number
<INPUT NAME="pfType" TYPE="radio" VALUE="5">List
<INPUT NAME="pfType" TYPE="radio" VALUE="6">Memo
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Size:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="pfSize" TYPE="text" SIZE="5"
MAXLENGTH="2">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Validation:</TD>
<TD CLASS="winTabField">
<INPUT NAME="pfValidationType" TYPE="radio" VALUE="1">Date
<INPUT NAME="pfValidationType" TYPE="radio" VALUE="2">SSN
<INPUT NAME="pfValidationType" TYPE="radio" VALUE="3">US Phone
</TD>
</TR>
<TR>
<TD CLASS="winTabField">&nbsp;</TD>
<TD CLASS="winTabField">
<INPUT NAME="pfValidationType" TYPE="radio" VALUE="4">Number
<INPUT NAME="pfValidationType" TYPE="radio" VALUE="5">Email
<INPUT NAME="pfValidationType" TYPE="radio" VALUE="6">Web
<INPUT NAME="pfValidationType" TYPE="radio" VALUE="7">None
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Range (n,n):</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="pfRange" TYPE="text" SIZE="5"
MAXLENGTH="10">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Required Field?</TD>
<TD CLASS="winTabField">
<INPUT NAME="pfRequiredField" TYPE="radio" VALUE="1">Yes
<INPUT NAME="pfRequiredField" TYPE="radio" VALUE="0">No
</TD>

```

```

</TR>
<TR>
<TD CLASS="winTabField" COLSPAN=2><B>ACTIONS</B></TD>
</TR>
<TR>
<TD CLASS="winTabField">Failed CStatus:</TD>
<TD CLASS="winTabField">
<INPUT NAME="pfCStatus" TYPE="radio" VALUE="1">1
<INPUT NAME="pfCStatus" TYPE="radio" VALUE="2">2
<INPUT NAME="pfCStatus" TYPE="radio" VALUE="3">3
<INPUT NAME="pfCStatus" TYPE="radio" VALUE="4">4
<INPUT NAME="pfCStatus" TYPE="radio" VALUE="5">5
<INPUT NAME="pfCStatus" TYPE="radio" VALUE="6">None
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Failed Color:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="pfFailColor" TYPE="text" SIZE="10"
MAXLENGTH="10">
</TD>
</TR>
<TR>
<TD CLASS="winTabField" COLSPAN=2><B>MISCELLENOUS</B></TD>
</TR>
<TR>
<TD CLASS="winTabField">Encrypt Data?</TD>
<TD CLASS="winTabField">
<INPUT NAME="pfEncrypt" TYPE="radio" VALUE="1">Yes
<INPUT NAME="pfEncrypt" TYPE="radio" VALUE="0">No
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Include in Search?</TD>
<TD CLASS="winTabField">
<INPUT NAME="pfIncludeInSearch" TYPE="radio" VALUE="1">Yes
<INPUT NAME="pfIncludeInSearch" TYPE="radio" VALUE="0">No
</TD>
</TR>
<TR>
<TD CLASS="winTabField" COLSPAN=2><B>INTERFACE</B></TD>
</TR>
<TR>
<TD CLASS="winTabField">Has an Interface?</TD>
<TD CLASS="winTabField">
<INPUT NAME="pfhasinterface" TYPE="radio" VALUE="1">Yes
<INPUT NAME="pfhasinterface" TYPE="radio" VALUE="0">No
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Interface Name:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="pfinterfacename" TYPE="text" SIZE="30"
MAXLENGTH="30">
</TD>
</TR>
<TR>

```

```

<TD CLASS="winTabField" COLSPAN=2><B>HELP SUPPORT</B></TD>
</TR>
<TR>
<TD CLASS="winTabField">Status Bar:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="pfHelpMessage" TYPE="text" SIZE="30"
MAXLENGTH="75">
</TD>
</TR>
<TR>
<TD CLASS="winTabField" VALIGN="top">Popup Window:</TD>
<TD CLASS="winTabField">
<TEXTAREA CLASS="stdTextBox" COLS=30 ROWS=3 NAME="pfHelpSummary"
WRAP="soft"></TEXTAREA>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>
</CFFORM>
</TD>
</TR>
</TABLE>

</CF_srWinTab>

</CFIF>

```

ADMINLISTPROFILES.CFM

```

<CFPARAM NAME="URL.action" DEFAULT="edit">
<CFSET mainFile = "main.cfm?a=tools&body=adminListProfiles.cfm">
<CFSET mainTable = "tblProfileGroups">
<CFSET idField = "pg_id">
<CFSET editTitle = "Profile Groups">
<CFSET useSource = "#VARIABLES.gvDATASOURCE_CONFIG#">

<CFIF NOT #CompareNoCase( URL.action, "View" )#>

<CF_srReportEngine
ACTION="run"
NAME="listProfileGroups"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="New"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=add"
>

<CFELSEIF NOT #CompareNoCase( URL.action, "Edit" )#>

<CFSET title = "#VARIABLES.editTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=save">
<CFSET formName="formEdit">

<CFELSEIF NOT #CompareNoCase( URL.action, "Save" )#>

```

```

<CFWDDX
INPUT="#FORM.WDDXContent#"
OUTPUT="EditedProfiles"
ACTION="WDDX2CFML"
>
<CFSET insertCount = 0>
<CFSET updateCount = 0>
<CFSET deleteCount = 0>

<CFLOOP QUERY="EditedProfiles">

<CFIF NOT #CompareNoCase( EditedProfiles.pg_id,"new" )#>

<CFQUERY DATASOURCE="#VARIABLES.useSource#">

INSERT INTO tblProfileGroups (
pgName,pgDescription,pgLabel,pgLayoutType,pgLayoutCols )
VALUES (
'#pgName#','#pgDescription#','#pgLabel#','#pgLayoutType#','#pgLayoutCols#'
)

</CFQUERY>

<CFSET tableName = "tbl" & #pgName#>
<CFSET primaryField = #pgName# & "_id">

<CFQUERY DATASOURCE="#VARIABLES.gvDATASOURCE_DATA#">

CREATE TABLE #VARIABLES.tableName#
(
#VARIABLES.primaryField# Counter,
CONSTRAINT pk_constraint PRIMARY KEY (#VARIABLES.primaryField#)
)

</CFQUERY>

<CFSET insertCount = insertCount + 1>

<CFELSEIF NOT #CompareNoCase( EditedProfiles.wasEdited,"Delete" )#>

<CFQUERY DATASOURCE="#VARIABLES.useSource#">

DELETE FROM tblProfileGroups
WHERE pg_id = #pg_id#

</CFQUERY>

<CFSET tableName = "tbl" & #pgName#>
<CFQUERY DATASOURCE="#VARIABLES.gvDATASOURCE_DATA#">

DROP TABLE #VARIABLES.tableName#

</CFQUERY>

<CFSET deleteCount = deleteCount + 1>

```

```

<CFELSEIF NOT #CompareNoCase( EditedProfiles.wasEdited,"Yes" )#>

<CFQUERY DATASOURCE="#VARIABLES.useSource#">

UPDATE tblProfileGroups
SET pgName = '#pgName#',
pgDescription = '#pgDescription#',
pgLabel = '#pgLabel#',
pgLayoutType = #pgLayoutType#,
pgLayoutCols = #pgLayoutCols#
WHERE pg_id = #pg_id#

</CFQUERY>
<CFSET updateCount = updateCount + 1>

</CFIF>

</CFLOOP>

<CFLOCATION URL="#VARIABLES.mainFile#">

</CFIF>

<CFIF #CompareNoCase( URL.action,"View" )#>

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

SELECT      pg_id, pgName, pgDescription, pgLabel, pgLayoutType,
pgLayoutCols
FROM #VARIABLES.mainTable#
ORDER BY pgName

</CFQUERY>

<SCRIPT LANGUAGE="javascript" SRC="wddx.js"></SCRIPT>
<SCRIPT LANGUAGE="JAVASCRIPT">

function onSubmit( destination )
{
<CFOUTPUT>
document.#VARIABLES.formName#.action = destination;
document.#VARIABLES.formName#.submit();
</CFOUTPUT>
}

<!-- Convert query to Javascript object --->
<CFWDDX
ACTION="CFML2JS"
INPUT="#getRecordInfo#"
TOPLEVELVARIABLE="Profiles"
>

// Add a column called "wasedited" to the recordset
// A "Yes" in this column means the row was "touched"
Profiles.addColumn("wasedited");

```

```

////////////////////////////////////

```

```

// This function fills the SELECT list with users
function initControls()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{
// Clear any current OPTIONS from the SELECT
ProfileList.options.length = 0;

// For each record...
for (var i = 0; i < Profiles.getRowCount(); i++)
{

// Create a new OPTION object
NewOpt = new Option;
NewOpt.value = Profiles.pg_id[i];
NewOpt.text = Profiles.pgname[i];

// Add the new object to the SELECT list
ProfileList.options[ProfileList.options.length] = NewOpt;

}
//UserList.selectedIndex = 0;
//fillControls();
}
}

function fillControls()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{
// Add one to the OPTION number to get the data row number
var RowNum = ProfileList.selectedIndex;

// Populate textboxes with data in that row
pgName.value = Profiles.pgname[RowNum];
pgDescription.value = Profiles.pgdescription[RowNum];
pgLabel.value = Profiles.pglabel[RowNum];

for ( n=0;n<2;n++ )
if ( Profiles.pglayouttype[RowNum] == n+1)
pgLayoutType[n].checked = true;

if ( Profiles.pglayoutcols[RowNum] == 2)
pgLayoutCols[0].checked = true;

if ( Profiles.pglayoutcols[RowNum] == 4)
pgLayoutCols[1].checked = true;

}
}

function newRecord()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{
// Add a new row to the recordset
Profiles.addRows(1);
}
}

```

```

NewRow = Profiles.getRowCount()-1;

Profiles.setField(NewRow, "pg_id", "new");
Profiles.setField(NewRow, "pgname", "(New Profile)");
Profiles.setField(NewRow, "pglabel", "");
Profiles.setField(NewRow, "pgdescription", "");
Profiles.setField(NewRow, "pglayouttype", "1");
Profiles.setField(NewRow, "pglayoutcols", "2");

// Re-initialize the SELECT list
initControls();

// Re-select the book that was selected before
ProfileList.selectedIndex = NewRow;
fillControls();
}
}

function savetoServer()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{
mySerializer = new WddxSerializer();
profilesAsWDDX = mySerializer.serialize( Profiles );
WDDXContent.value = profilesAsWDDX;
submit();
}
}

function saveChanges()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{
var selectedProfile = ProfileList.selectedIndex;
var RowNum = selectedProfile;

// Populate textboxes with data in that row
Profiles.pgname[RowNum] = pgName.value;
Profiles.pgdescription[RowNum] = pgDescription.value;
Profiles.pglabel[RowNum] = pgLabel.value;

for ( n=0;n<2;n++ )
if ( pgLayoutType[n].checked )
Profiles.pglayouttype[RowNum] = pgLayoutType[n].value;

for ( n=0;n<2;n++ )
if ( pgLayoutCols[n].checked )
Profiles.pglayoutcols[RowNum] = pgLayoutCols[n].value;

Profiles.wasedited[RowNum] = 'Yes';

initControls();

ProfileList.selectedIndex = -1;
ProfileList.selectedIndex = selectedProfile;
}
}

```

```

function deleteRecord()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{
var selectedProfile = ProfileList.selectedIndex;
var RowNum = selectedProfile;

// Populate textboxes with data in that row
Profiles.pglabel[RowNum] = "MARKED FOR DELETION";
pgLabel.value = Profiles.pglabel[RowNum];

Profiles.wasedited[RowNum] = 'Delete';

initControls();

ProfileList.selectedIndex = -1;
ProfileList.selectedIndex = selectedProfile;
}
}

window.onload=initControls;

</SCRIPT>

<!-- display appropriate header -->
<CF_srWinTab
TABS="Profile Groups,Group Fields,Field Rules,Lists,Interfaces"
TABSELECTED="1"
TABURL="javascript:onSubmit(
'main.cfm?a=tools&body=adminListProfiles.cfm' ),javascript:onSubmit(
'main.cfm?a=tools&body=adminListProfileFields.cfm'
),javascript:onSubmit(
'main.cfm?a=tools&body=adminListProfileFieldsRules.cfm'
),javascript:onSubmit(
'main.cfm?a=tools&body=adminListProfileLists.cfm'
),javascript:onSubmit( 'main.cfm?a=tools&body=adminListInterfaces.cfm'
)"
SYSTEM_TOOLBAR="New,Save,Save to Server,Delete,Undo"
SYSTEM_TOOLBARLINKS="javascript:newRecord(),javascript:saveChanges(),ja
vascript:savetoServer(),javascript:deleteRecord(),javascript:fillContro
ls()"
>

<TABLE BORDER=0 WIDTH=100%>

<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="adminProfiles">
</SPAN>
<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<INPUT TYPE="hidden" NAME="WDDXContent" VALUE="">
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel" ALIGN="center" WIDTH=40%>&nbsp;   PROFILE
GROUPS</TD>

```

```

<TD CLASS="winTabLabel" ALIGN="center">&nbsp;   PROPERTIES</TD>
</TR>
<TR>
<TD CLASS="winTabField" WIDTH=40% ALIGN="center" VALIGN="top">
<SELECT NAME="ProfileList" SIZE="10" onCHANGE="fillControls()">
<OPTION>===== (loading...) =====
</SELECT>
</TD>
<TD CLASS="winTabField" VALIGN="top">

<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabField">Name:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="pgName" TYPE="text" SIZE="30"
MAXLENGTH="20">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Tab Label:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="pgLabel" TYPE="text" SIZE="30"
MAXLENGTH="30">
</TD>
</TR>
<TR VALIGN="top">
<TD CLASS="winTabField">Description:</TD>
<TD CLASS="winTabField">
<TEXTAREA CLASS="stdTextBox" COLS=30 ROWS=3 NAME="pgDescription"
WRAP="soft"></TEXTAREA>
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Layout Type:</TD>
<TD CLASS="winTabField">
<INPUT NAME="pgLayoutType" TYPE="radio" VALUE="1">Single Page
<INPUT NAME="pgLayoutType" TYPE="radio" VALUE="2">Datasheet
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Layout Columns:</TD>
<TD CLASS="winTabField">
<INPUT NAME="pgLayoutCols" TYPE="radio" VALUE="2">Single
<INPUT NAME="pgLayoutCols" TYPE="radio" VALUE="4">Double
</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>
</CFFORM>
</TD>
</TR>
</TABLE>

</CF_srWinTab>

```

</CFIF>

ADMINLISTSUMMARY.CFM

```
<CFPARAM NAME="URL.action" DEFAULT="View">
<CFSET mainFile = "main.cfm?a=tools&body=adminListSummary.cfm">
<CFSET mainTable = "tblSummary">
<CFSET idField = "summary_id">
<CFSET addTitle = "CREATE A NEW SUMMARY">
<CFSET editTitle = "EDIT SUMMARY">

<CFIF URL.action EQ "View">

<CF_srReportEngine
ACTION="run"
NAME="listSummary"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="New"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=add"
>

<CFELSEIF URL.action EQ "Add">

<CFSET title = "#VARIABLES.addTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=saveNew">
<CFSET formName="formAdd">

<CFELSEIF URL.action EQ "Edit">

<CFSET title = "#VARIABLES.editTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=save">
<CFSET formName="formEdit">

<CFELSEIF URL.action EQ "saveNew">

<CFINSERT DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#"
TABLENAME="#VARIABLES.mainTable#">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "save">

<CFUPDATE DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#"
TABLENAME="#VARIABLES.mainTable#">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "delete">

<CFQUERY NAME="deleteInfo"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

DELETE
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>
```

```

<CFLOCATION URL="#VARIABLES.mainFile#">

</CFIF>

<CFIF URL.action NEQ "View">

<CFQUERY NAME="getRecordInfo"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      *
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = <CFIF URL.action EQ "Edit">#Evaluate(
'URL.' & VARIABLES.idField )#<CFELSE>-1</CFIF>

</CFQUERY>

<!--- display appropriate header --->
<CF_srWinTab
TABS="#VARIABLES.title#"
SYSTEM_TOOLBAR="Save,List Summary"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit(),
main.cfm?a=tools&body=adminListSummary.cfm"
>

<TABLE BORDER=0 WIDTH=100%>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<CFOUTPUT>
<INPUT TYPE="hidden" NAME="#VARIABLES.idField#" VALUE="#Evaluate(
'getRecordInfo.' & VARIABLES.idField )#">
</CFOUTPUT>
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="adminListSummary">
</SPAN>
<P>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel">&nbsp;Short Name:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="shortName"
TYPE="text"
VALUE="#getRecordInfo.shortName#"
SIZE="30"
MAXLENGTH="20"
REQUIRED="yes"
MESSAGE="A Summary Short Name must be entered"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Summary:</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>

```

```

<TD CLASS="winTabField" COLSPAN=2>
<TEXTAREA COLS=50 ROWS=5 NAME="summary"
WRAP="soft"><CFOUTPUT>#getRecordInfo.summary#</CFOUTPUT></TEXTAREA>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</CFFORM>
</TABLE>

</CF_srWinTab>

</CFIF>

```

ADMINLISTUSERGROUPS.CFM

```

<CFPARAM NAME="URL.action" DEFAULT="view">
<CFSET mainFile = "main.cfm?a=tools&body=adminListUserGroups.cfm">
<CFSET mainTable = "tblGroups">
<CFSET idField = "usergroup_id">
<CFSET addTitle = "CREATE A NEW GROUP">
<CFSET editTitle = "EDIT GROUP">
<CFSET useSource = "#VARIABLES.gvDATASOURCE_USER#">

<CFIF URL.action EQ "View">

<CF_srReportEngine
ACTION="run"
NAME="listUserGroups"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="New"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=add"
>

<CFELSEIF URL.action EQ "Add">

<CFSET title = "#VARIABLES.addTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=savenew">
<CFSET formName="formAdd">

<CFELSEIF URL.action EQ "Edit">

<CFSET title = "#VARIABLES.editTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=save">
<CFSET formName="formEdit">

<CFELSEIF URL.action EQ "saveNew">

<CFINSERT DATASOURCE="#VARIABLES.useSource#"
TABLENAME="#VARIABLES.mainTable#">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "save">

```

```

<CFUPDATE DATASOURCE="#VARIABLES.useSource#"
TABLENAME="#VARIABLES.mainTable#">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "delete">

<CFQUERY NAME="deleteInfo" DATASOURCE="#VARIABLES.useSource#">

DELETE
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>
<CFLOCATION URL="#VARIABLES.mainFile#">

</CFIF>

<CFIF URL.action NEQ "view">

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

SELECT      *
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = <CFIF URL.action EQ "Edit">#Evaluate(
'URL.' & VARIABLES.idField )#<CFELSE>-1</CFIF>

</CFQUERY>

<!--- display appropriate header --->
<CF_srWinTab
TABS="#VARIABLES.title#"
SYSTEM_TOOLBAR="Save,List Groups"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit()",
#VARIABLES.mainFile#"
>

<TABLE BORDER=0 WIDTH=100%>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<CFOUTPUT>
<INPUT TYPE="hidden" NAME="#VARIABLES.idField#" VALUE="#Evaluate(
'getRecordInfo.' & VARIABLES.idField )#">
</CFOUTPUT>
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="adminUserGroups">
</SPAN>
<P>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel">&nbsp;Name:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="ug_name"
TYPE="text"
VALUE="#getRecordInfo.ug_name#"

```



```

<CFELSEIF URL.action EQ "save">

<CFWDDX
INPUT="#FORM.WDDXContent#"
OUTPUT="EditedUsers"
ACTION="WDDX2CFML"
>
<CFSET insertCount = 0>
<CFSET updateCount = 0>
<CFSET deleteCount = 0>

<CFLOOP QUERY="EditedUsers">

<CFIF EditedUsers.user_id EQ "new">

<CFQUERY DATASOURCE="#VARIABLES.useSource#">
INSERT INTO tblUserAccounts
(lname, fname, rank, ssn, phone, email, command, user_loginid, user_password, ex
pire, force_password, locked )
VALUES (
'#lname#', '#fname#', '#rank#', '#ssn#', '#phone#', '#email#', '#command#', '#
user_loginid#', '#user_password#', #CreateODBCDate( dateexpire
) #, #force_password#, #locked# )
</CFQUERY>
<CFSET insertCount = insertCount + 1>

<CFELSEIF EditedUsers.wasEdited EQ "Delete">

<CFQUERY DATASOURCE="#VARIABLES.useSource#">
DELETE FROM tblUserAccounts
WHERE user_id = #user_id#
</CFQUERY>
<CFSET deleteCount = deleteCount + 1>

<CFELSEIF EditedUsers.wasEdited EQ "Yes">

<CFQUERY DATASOURCE="#VARIABLES.useSource#">
UPDATE tblUserAccounts
SET lname = '#lname#',
fname = '#fname#',
rank = '#rank#',
ssn = '#ssn#',
phone = '#phone#',
email = '#email#',
command = '#command#',
user_loginid = '#user_loginid#',
user_password = '#user_password#',
expire = #CreateODBCDate( dateexpire ) #,
force_password = #force_password#,
locked = #locked#
WHERE user_id = #user_id#
</CFQUERY>
<CFSET updateCount = updateCount + 1>

</CFIF>

```

```

</CFLOOP>

<CFLOCATION URL="#VARIABLES.mainFile#">

</CFIF>

<CFIF URL.action NEQ "view">

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

SELECT      lname & ', ' & fname AS fullname, user_id, lname, fname,
rank, ssn, phone, email, command, user_loginid, format( expire, 'mm/dd/yyyy' )
AS dateExpire, force_password,
locked, template, last_logon, login_attempts, user_password
FROM #VARIABLES.mainTable#
ORDER BY lname

</CFQUERY>

<SCRIPT LANGUAGE="javascript" SRC="wddx.js"></SCRIPT>
<SCRIPT LANGUAGE="JAVASCRIPT">

<!-- Convert query to Javascript object -->
<CFWDDX
ACTION="CFML2JS"
INPUT="#getRecordInfo#"
TOPLEVELVARIABLE="Users"
>

// Add a column called "wasedited" to the recordset
// A "Yes" in this column means the row was "touched"
Users.addColumn("wasedited");

////////////////////////////////////
// This function fills the SELECT list with users
function initControls()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{
// Clear any current OPTIONS from the SELECT
UserList.options.length = 0;

// For each record...
for (var i = 0; i < Users.getRowCount(); i++)
{

// Create a new OPTION object
NewOpt = new Option;
NewOpt.value = Users.user_id[i];
NewOpt.text = Users.fullname[i];

// Add the new object to the SELECT list
UserList.options[UserList.options.length] = NewOpt;

}
//UserList.selectedIndex = 0;
//fillControls();

```

```

}
}

function fillControls()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{
// Add one to the OPTION number to get the data row number
var RowNum = UserList.selectedIndex;

// Populate textboxes with data in that row
lname.value = Users.lname[RowNum];
fname.value = Users.fname[RowNum];
rank.value = Users.rank[RowNum];
ssn.value = Users.ssn[RowNum];
command.value = Users.command[RowNum];
phone.value = Users.phone[RowNum];
email.value = Users.email[RowNum];
expire.value = Users.dateexpire[RowNum];
user_loginid.value = Users.user_loginid[RowNum];
user_password.value = Users.user_password[RowNum];
if ( Users.force_password[RowNum] != 0 )
force_password[0].checked = true;
else
force_password[1].checked = true;
if ( Users.locked[RowNum] != 0 )
locked[0].checked = true;
else
locked[1].checked = true;
}
}

function newRecord()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{
// Add a new row to the recordset
Users.addRows(1);
NewRow = Users.getRowCount()-1;

Users.setField(NewRow, "user_id", "new");
Users.setField(NewRow, "fullname", "(New Record)");
Users.setField(NewRow, "lname", "Last Name");
Users.setField(NewRow, "fname", "First Name");
Users.setField(NewRow, "rank", "");
Users.setField(NewRow, "ssn", "");
Users.setField(NewRow, "command", "");
Users.setField(NewRow, "phone", "");
Users.setField(NewRow, "email", "");
Users.setField(NewRow, "dateexpire", "");
Users.setField(NewRow, "user_loginid", "");
Users.setField(NewRow, "user_password", "");
Users.setField(NewRow, "force_password", "");
Users.setField(NewRow, "locked", "");

// Re-initialize the SELECT list
initControls();

```

```

// Re-select the book that was selected before
UserList.selectedIndex = NewRow;
fillControls();
}
}

function savetoServer()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{
mySerializer = new WddxSerializer();
usersAsWDDX = mySerializer.serialize( Users );
WDDXContent.value = usersAsWDDX;
submit();
}
}

function saveChanges()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{
var selectedUser = UserList.selectedIndex;
var RowNum = selectedUser;

// Populate textboxes with data in that row
Users.lname[RowNum] = lname.value;
Users.fname[RowNum] = fname.value;
Users.rank[RowNum] = rank.value;
Users.ssn[RowNum] = ssn.value;
Users.command[RowNum] = command.value;
Users.phone[RowNum] = phone.value;
Users.email[RowNum] = email.value;
Users.dateexpire[RowNum] = expire.value;
Users.user_loginid[RowNum] = user_loginid.value;
Users.user_password[RowNum] = user_password.value;
if ( force_password[0].checked )
Users.force_password[RowNum] = force_password[0].value;
else
Users.force_password[RowNum] = force_password[1].value;
if ( locked[0].checked )
Users.locked[RowNum] = locked[0].value;
else
Users.locked[RowNum] = locked[1].value;
Users.wasedited[RowNum] = 'Yes';

initControls();

UserList.selectedIndex = -1;
UserList.selectedIndex = selectedUser;
}
}

function deleteRecord()
{
with (document.<CFOUTPUT>#VARIABLES.formName#</CFOUTPUT>)
{

```

```

var selectedUser = userList.selectedIndex;
var rowNum = selectedUser;

// Populate textboxes with data in that row
Users.fullName[rowNum] = "(D)" + Users.lname[rowNum] + ", " +
Users.fname[rowNum];
Users.lname[rowNum] = "MARKED FOR";
Users.fname[rowNum] = "DELETION";
lname.value = Users.lname[rowNum];
fname.value = Users.fname[rowNum];
Users.wasEdited[rowNum] = 'Delete';

initControls();

UserList.selectedIndex = -1;
UserList.selectedIndex = selectedUser;
}
}

window.onload=initControls;

</SCRIPT>

<!-- display appropriate header -->
<CF_sWinTab
TABS="#VARIABLES.title#"
SYSTEM_TOOLBAR="New,Save,Save to Server,Delete,Undo"
SYSTEM_TOOLBARLINKS="javascript:newRecord(),javascript:saveChanges(),ja
vascript:savetoServer(),javascript:deleteRecord(),javascript:fillContro
ls()"
>

<TABLE BORDER=0 WIDTH=100%>

<CFQUERY NAME="GetCommands"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT          *
FROM            tblCommands
ORDER BY       cmd_description

</CFQUERY>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<INPUT TYPE="hidden" NAME="WDDXContent" VALUE="">
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_sHELPsummary shortName="adminUsers">
</SPAN>
<P>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel" ALIGN="center" WIDTH=40%>&nbsp;USERS</TD>
<TD CLASS="winTabLabel" ALIGN="center">&nbsp;PROPERTIES</TD>
</TR>
<TR>

```

```

<TD CLASS="winTabField" WIDTH=40% ALIGN="center" VALIGN="top">
<SELECT NAME="UserList" SIZE="10" onCHANGE="fillControls()">
<OPTION>===== (loading...) =====
</SELECT>
</TD>
<TD CLASS="winTabField" VALIGN="top">

<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabField" COLSPAN=2><B>DEMOGRAPHICS</B></TD>
</TR>
<TR>
<TD CLASS="winTabField">Name:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="lname" TYPE="text" SIZE="20"
MAXLENGTH="30">
 
<INPUT CLASS="stdTextBox" NAME="fname" TYPE="text" SIZE="15"
MAXLENGTH="30">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Rank:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="rank" TYPE="text" SIZE="20"
MAXLENGTH="10">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">SSN (last 4):</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="ssn" TYPE="text" SIZE="5"
MAXLENGTH="4">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Organization:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="command" TYPE="text" SIZE="30"
MAXLENGTH="30">&nbsp;&nbsp;<A HREF="">List</A>
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Phone:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="phone" TYPE="text" SIZE="20"
MAXLENGTH="15">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Email:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="email" TYPE="text" SIZE="40"
MAXLENGTH="75">
</TD>
</TR>
<TR>

```

```

<TD CLASS="winTabField" COLSPAN=2><B>SECURITY</B></TD>
</TR>
<TR>
<TD CLASS="winTabField">Account expires:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="expire" TYPE="text" SIZE="10"
MAXLENGTH="10">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Login ID:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="user_loginid" TYPE="text" SIZE="20"
MAXLENGTH="15">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Password:</TD>
<TD CLASS="winTabField">
<INPUT CLASS="stdTextBox" NAME="user_password" TYPE="password"
SIZE="20" MAXLENGTH="10">
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Force Password?</TD>
<TD CLASS="winTabField">
<INPUT NAME="force_password" TYPE="radio" VALUE="1">Yes
<INPUT NAME="force_password" TYPE="radio" VALUE="0">No
</TD>
</TR>
<TR>
<TD CLASS="winTabField">Lock Account?</TD>
<TD CLASS="winTabField">
<INPUT NAME="locked" TYPE="radio" VALUE="1">Yes
<INPUT NAME="locked" TYPE="radio" VALUE="0">No
</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</CFFORM>
</TABLE>

</CF_srWinTab>

</CFIF>

```

ADMINSECURITYCONFIGURATION.CFM

```

<CFSET mainFile =
"main.cfm?a=administration&body=adminSecurityConfiguration.cfm">

<CFPARAM NAME="URL.action" DEFAULT="view">

```

```

<CFIF #URL.action# IS "save">

<CFSET gvDEFAULTPSWDEXP = "#FORM.DEFAULTPSWDEXP#">
<CFSET gvMIN_PSWD_SIZE = "#FORM.MIN_PSWD_SIZE#">
<CFSET gvPSWD_REUSE = "#FORM.PSWD_REUSE#">
<CFSET gvPSWD_FORGET = "#FORM.PSWD_FORGET#">
<CFSET gvPSWD_FORGET_CHG = "#FORM.PSWD_FORGET_CHG#">
<CFSET gvPSWD_REMEMBER = "#FORM.PSWD_REMEMBER#">
<CFSET gvPSWD_LOCKIFFAIL = "#FORM.PSWD_LOCKIFFAIL#">
<CFSET gvPSWD_SENDEMAIL = "#FORM.PSWD_SENDEMAIL#">
<CFSET gvPSWD_ISCASE = "#FORM.PSWD_ISCASE#">
<CFSET gvPSWD_INCLNUM = "#FORM.PSWD_INCLNUM#">
<CFSET gvTIMEOUT_MINS = "#FORM.TIMEOUT_MINS#">
<CFSET gvLOGON_ATTEMPTS = "#FORM.LOGON_ATTEMPTS#">
<CFSET gvSECURITYEMAIL = "#FORM.SECURITYEMAIL#">
<CFSET gvIP_BLOCKING = "#FORM.IP_BLOCKING#">
<CFSET gvDOMAIN_BLOCKING = "#FORM.DOMAIN_BLOCKING#">
<CFSET gvDOMAIN_BLOCKINGLIST = "#FORM.DOMAIN_BLOCKINGLIST#">
<CFSET gvDOW_BLOCKING = "#FORM.DOW_BLOCKING#">
<CFSET gvDOW_DAYSTOBLOCK = "#FORM.DOW_DAYSTOBLOCK#">
<CFSET gvTIME_BLOCKING = "#FORM.TIME_BLOCKING#">
<CFSET gvTIME_FROMBLOCK = "#FORM.TIME_FROMBLOCK#">
<CFSET gvTIME_TOBLOCK = "#FORM.TIME_TOBLOCK#">
<CFSET gvMSG_FAILEDLOGON = "#FORM.MSG_FAILEDLOGON#">
<CFSET gvMSG_USEREXPIRE = "#FORM.MSG_USEREXPIRE#">
<CFSET gvMSG_RECORDLOCKED = "#FORM.MSG_RECORDLOCKED#">
<CFSET gvMSG_IPBLOCKING = "#FORM.MSG_IPBLOCKING#">
<CFSET gvMSG_DOMAINBLOCKING = "#FORM.MSG_DOMAINBLOCKING#">
<CFSET gvMSG_DOWBLOCKING = "#FORM.MSG_DOWBLOCKING#">
<CFSET gvMSG_TIMEBLOCKING = "#FORM.MSG_TIMEBLOCKING#">

<CFINCLUDE TEMPLATE="adminWriteConfig.cfm">
<CFINCLUDE TEMPLATE="adminWriteLogon.cfm">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "view">

<CFSET title = "SECURITY POLICY">
<CFSET formAction = "#VARIABLES.mainFile#&action=save">
<CFSET formName="formPolicy">

<!-- display appropriate header --->
<CF_srWinTab
TABS="Security Policy,Program Configuration"
TABSELECTED="1"
TABURL="main.cfm?a=administration&body=adminSecurityConfiguration.cfm,main.cfm?a=administration&body=adminConfiguration.cfm"
SYSTEM_TOOLBAR="Save As...,Save,Policies,View IPs,Add New IP"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=savesas,javascript:document.#VARIABLES.formName#.submit(),#VARIABLES.mainFile#&action=policies,listIPblock.cfm,listIPblock.cfm?action=add"
BASEURL="#VARIABLES.mainFile#"
>

<TABLE BORDER=0 WIDTH=100%>

```

```

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="adminSecurity">
</SPAN>
<P>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;BLOCKING OPTIONS</TD>
<TD CLASS="winTabField">&nbsp;&nbsp;&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Allow certain IPs or Groups?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="IP_BLOCKING" VALUE="No" CHECKED="#( NOT
VARIABLES.gvIP_BLOCKING )#">No
<CFINPUT TYPE="Radio" NAME="IP_BLOCKING" VALUE="Yes"
CHECKED="#VARIABLES.gvIP_BLOCKING#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;Blocked IP message:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="MSG_IPBLOCKING"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
VALUE="#VARIABLES.gvMSG_IPBLOCKING#"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Allow only certain domains?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="DOMAIN_BLOCKING" VALUE="No" CHECKED="#( NOT
VARIABLES.gvDOMAIN_BLOCKING )#">No
<CFINPUT TYPE="Radio" NAME="DOMAIN_BLOCKING" VALUE="Yes"
CHECKED="#VARIABLES.gvDOMAIN_BLOCKING#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;List domains separated by
commas:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="DOMAIN_BLOCKINGLIST"
SIZE="30"
MAXLENGTH="30"
REQUIRED="NO"
VALUE="#VARIABLES.gvDOMAIN_BLOCKINGLIST#"
>
</TD>
</TR>

```



```

REQUIRED="no"
VALUE="#VARIABLES.gvMSG_DOWBLOCKING#"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Allow only certain times?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="TIME_BLOCKING" VALUE="No" CHECKED="#( NOT
VARIABLES.gvTIME_BLOCKING )#">No
<CFINPUT TYPE="Radio" NAME="TIME_BLOCKING" VALUE="Yes"
CHECKED="#VARIABLES.gvTIME_BLOCKING#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Enter the times to block:</TD>
<TD CLASS="winTabField">
From:
<CFINPUT
TYPE="TEXT"
NAME="TIME_FROMBLOCK"
SIZE="6"
MAXLENGTH="5"
REQUIRED="NO"
VALIDATE="time"
VALUE="#VARIABLES.gvTIME_FROMBLOCK#"
>
&nbsp;&nbsp;&nbsp;To:
<CFINPUT
TYPE="TEXT"
NAME="TIME_TOBLOCK"
SIZE="6"
MAXLENGTH="5"
REQUIRED="NO"
VALIDATE="time"
VALUE="#VARIABLES.gvTIME_TOBLOCK#"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Blocked by time message:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="MSG_TIMEBLOCKING"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
VALUE="#VARIABLES.gvMSG_TIMEBLOCKING#"
>
</TD>
</TR>
<TR>
<TD>&nbsp;&nbsp;&nbsp;</TD>
<TD>&nbsp;&nbsp;&nbsp;</TD>
</TR>
<TR>

```

```

<TD CLASS="winTabLabel">&nbsp;PASSWORD REQUIREMENTS</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Passwords expire in:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="DEFAULTPSWDEXP"
TYPE="text"
VALUE="#VARIABLES.gvDEFAULTPSWDEXP#"
SIZE="5"
MAXLENGTH="4"
>
&nbsp;Days
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Minimum password length:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="MIN_PSWD_SIZE"
TYPE="text"
VALUE="#VARIABLES.gvMIN_PSWD_SIZE#"
SIZE="5"
MAXLENGTH="4"
>
&nbsp;Characters
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Can users reuse their old password?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="PSWD_REUSE" VALUE="No" CHECKED="#( NOT
VARIABLES.gvPSWD_REUSE )#">No
<CFINPUT TYPE="Radio" NAME="PSWD_REUSE" VALUE="Yes"
CHECKED="#VARIABLES.gvPSWD_REUSE#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Passwords must include number(s)?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="PSWD_INCLNUM" VALUE="No" CHECKED="#( NOT
VARIABLES.gvPSWD_INCLNUM )#">No
<CFINPUT TYPE="Radio" NAME="PSWD_INCLNUM" VALUE="Yes"
CHECKED="#VARIABLES.gvPSWD_INCLNUM#">Yes
</TD>
</TR>
<TR>
<TD>&nbsp;</TD>
<TD>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;PROTECTION FEATURES</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>

```

```

<TD CLASS="winTabLabel">&nbsp;If users fail to change password, lock
accounts?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="PSWD_LOCKIFFAIL" VALUE="No" CHECKED="#( NOT
VARIABLES.gvPSWD_LOCKIFFAIL )#">No
<CFINPUT TYPE="Radio" NAME="PSWD_LOCKIFFAIL" VALUE="Yes"
CHECKED="#VARIABLES.gvPSWD_LOCKIFFAIL#">Yes
</TD>
</TR>
<TR>
<TD>&nbsp;</TD>
<TD>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;LOGON FEATURES</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Passwords are case-sensative?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="PSWD_ISCASE" VALUE="No" CHECKED="#( NOT
VARIABLES.gvPSWD_ISCASE )#">No
<CFINPUT TYPE="Radio" NAME="PSWD_ISCASE" VALUE="Yes"
CHECKED="#VARIABLES.gvPSWD_ISCASE#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Enable "Forget Password?" on login?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="PSWD_FORGET" VALUE="No" CHECKED="#( NOT
VARIABLES.gvPSWD_FORGET )#">No
<CFINPUT TYPE="Radio" NAME="PSWD_FORGET" VALUE="Yes"
CHECKED="#VARIABLES.gvPSWD_FORGET#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Change password if user
forgot?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="PSWD_FORGET_CHG" VALUE="No" CHECKED="#( NOT
VARIABLES.gvPSWD_FORGET_CHG )#">No
<CFINPUT TYPE="Radio" NAME="PSWD_FORGET_CHG" VALUE="Yes"
CHECKED="#VARIABLES.gvPSWD_FORGET_CHG#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Enable "Remember Login?" on login?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="PSWD_REMEMBER" VALUE="No" CHECKED="#( NOT
VARIABLES.gvPSWD_REMEMBER )#">No
<CFINPUT TYPE="Radio" NAME="PSWD_REMEMBER" VALUE="Yes"
CHECKED="#VARIABLES.gvPSWD_REMEMBER#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Lock account after how many unsuccessful
attempts:</TD>

```

```

<TD CLASS="winTabField">
<CFINPUT
NAME="LOGON_ATTEMPTS"
TYPE="text"
VALUE="#VARIABLES.gvLOGON_ATTEMPTS#"
VALIDATE="integer"
RANGE="0,10"
SIZE="5"
MAXLENGTH="4"
>
 Tries
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel"> Failed to logon message:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="MSG_FAILEDLOGON"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
VALUE="#VARIABLES.gvMSG_FAILEDLOGON#"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel"> Expired account message:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="MSG_USEREXPIRE"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
VALUE="#VARIABLES.gvMSG_USEREXPIRE#"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel"> User record locked message:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="MSG_RECORDLOCKED"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
VALUE="#VARIABLES.gvMSG_RECORDLOCKED#"
>
</TD>
</TR>
<TR>
<TD> </TD>
<TD> </TD>
</TR>
<TR>

```

```

<TD CLASS="winTabLabel">&nbsp;AUTO PILOT</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Send email reminders to change
password?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="PSWD_SENDEMAIL" VALUE="No" CHECKED="#( NOT
VARIABLES.gvPSWD_SENDEMAIL )#">No
<CFINPUT TYPE="Radio" NAME="PSWD_SENDEMAIL" VALUE="Yes"
CHECKED="#VARIABLES.gvPSWD_SENDEMAIL#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Email Security Violations to
Administrator?</TD>
<TD CLASS="winTabField">
<CFINPUT TYPE="Radio" NAME="SecurityEmail" VALUE="No" CHECKED="#Iif(
NOT VARIABLES.gvSECURITYEMAIL,De( 'true' ),De( 'false' ) )#">No
<CFINPUT TYPE="Radio" NAME="SecurityEmail" VALUE="Yes" CHECKED="#Iif(
VARIABLES.gvSECURITYEMAIL,De( 'true' ),De( 'false' ) )#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Automatically log off user after:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="TIMEOUT_MINS"
TYPE="text"
VALIDATE="integer"
VALUE="#VARIABLES.gvTIMEOUT_MINS#"
SIZE="5"
MAXLENGTH="4"
>
&nbsp;Minutes
</TD>
</TR>
</TABLE>
</TD>
</TR>
</CFFORM>
</TABLE>
</CF_srWinTab>

</CFIF>

```

ADMINCONFIGURATION.CFM

```

<CFSET mainFile =
"main.cfm?a=administration&body=adminConfiguration.cfm">

<CFPARAM NAME="URL.action" DEFAULT="view">

<CFIF #URL.action# IS "save">

<CFSET gvAdminEmail      = "#FORM.ADMINEMAIL#">

```

```

<CFSET gvAdminName      = "#FORM.ADMINNAME#">
<CFSET gvAdminPhone     = "#FORM.ADMINPHONE#">
<CFSET gvAdminEmail_Sec = "#FORM.ADMINEMAIL_SEC#">
<CFSET gvAdminName_Sec  = "#FORM.ADMINNAME_SEC#">
<CFSET gvAdminPhone_Sec = "#FORM.ADMINPHONE_SEC#">
<CFSET gvAPP_HOME       = "#FORM.APP_HOME#">
<CFSET gvAPP_COMMAND    = "#FORM.APP_COMMAND#">
<CFSET gvHOME_PAGE      = "#FORM.HOME_PAGE#">
<CFSET gvSMTP_SERVER    = "#FORM.SMTP_SERVER#">
<CFSET gvFromEmail      = "#FORM.FROMEMAIL#">

<CFINCLUDE TEMPLATE="adminWriteConfig.cfm">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "view">

<CFSET title = "PROGRAM CONFIGURATION">
<CFSET formAction = "#VARIABLES.mainFile#&action=save">
<CFSET formName="formProgConfig">

<!--- display appropriate header --->
<CF_srWinTab
TABS="Security Policy,Program Configuration"
TABSELECTED="2"
TABURL="main.cfm?a=administration&body=adminSecurityConfiguration.cfm,main.cfm?a=administration&body=adminConfiguration.cfm"
SYSTEM_TOOLBAR="Save"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit()"
BASEURL="#VARIABLES.mainFile#"
>

<TABLE BORDER=0 WIDTH=100%>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="adminConfig">
</SPAN>
<P>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel">&nbsp;ORGANIZATION INFORMATION</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Organization:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="APP_COMMAND"
SIZE="30"
MAXLENGTH="60"
REQUIRED="YES"
VALUE="#VARIABLES.gvAPP_COMMAND#"
MESSAGE="An Organization Name is required."
>

```

```

</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Homepage Address:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="HOME_PAGE"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
VALUE="#VARIABLES.gvHOME_PAGE#"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Readiness Explorer Address:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="APP_HOME"
SIZE="30"
MAXLENGTH="75"
REQUIRED="YES"
VALUE="#VARIABLES.gvAPP_HOME#"
MESSAGE="A Program Web Address is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;SMTP Mail Server (IP Address):</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="SMTP_SERVER"
SIZE="30"
MAXLENGTH="40"
REQUIRED="YES"
VALUE="#VARIABLES.gvSMTP_SERVER#"
MESSAGE="A SMTP Mail Server is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Offical Email Address:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="FROMEMAIL"
SIZE="30"
MAXLENGTH="75"
REQUIRED="YES"
VALUE="#VARIABLES.gvFROMEMAIL#"
MESSAGE="An Offical Email Address is required."
>
</TD>
</TR>

```

```

<TR>
<TD>&nbsp;&nbsp;&nbsp;</TD>
<TD>&nbsp;&nbsp;&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;PRIMARY ADMINISTRATOR</TD>
<TD CLASS="winTabField">&nbsp;&nbsp;&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Name:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="ADMINNAME"
SIZE="30"
MAXLENGTH="30"
REQUIRED="YES"
VALUE="#VARIABLES.gvADMINNAME#"
MESSAGE="A Primary Administrator NAME is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Phone:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="ADMINPHONE"
SIZE="30"
MAXLENGTH="20"
REQUIRED="NO"
VALUE="#VARIABLES.gvADMINPHONE#"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Email:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="ADMINEMAIL"
SIZE="30"
MAXLENGTH="60"
REQUIRED="YES"
VALUE="#VARIABLES.gvADMINEMAIL#"
MESSAGE="A Primary Administrator EMAIL ADDRESS is required."
>
</TD>
</TR>
<TR>
<TD>&nbsp;&nbsp;&nbsp;</TD>
<TD>&nbsp;&nbsp;&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;SECONDARY ADMINISTRATOR</TD>
<TD CLASS="winTabField">&nbsp;&nbsp;&nbsp;</TD>
</TR>

```

```

<TR>
<TD CLASS="winTabLabel">&nbsp;Name:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="ADMINNAME_SEC"
SIZE="30"
MAXLENGTH="30"
REQUIRED="NO"
VALUE="#VARIABLES.gvADMINNAME_SEC#"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Phone:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="ADMINPHONE_SEC"
SIZE="30"
MAXLENGTH="20"
REQUIRED="NO"
VALUE="#VARIABLES.gvADMINPHONE_SEC#"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Email:</TD>
<TD CLASS="winTabField">
<CFINPUT
TYPE="TEXT"
NAME="ADMINEMAIL_SEC"
SIZE="30"
MAXLENGTH="60"
REQUIRED="NO"
VALUE="#VARIABLES.gvADMINEMAIL_SEC#"
>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</CFFORM>
</TABLE>
</CF_srWinTab>

</CFIF>

```

ADMINWRITELOGON.CFM

```

<CFSET filetowrite = #GetDirectoryFromPath( GetTemplatePath() )# &
#VARIABLES.gvHOMEPAGE_NAME#>

<CFSET Beginning = "
<HTML>
<HEAD>

```

```

<TITLE>Welcome to Readiness Explorer</TITLE>
<STYLE TYPE='text/css'>
/*****
LOG-IN PROPERTIES
*****/
TD.Login
{
font: 12px tahoma,verdana,arial;
color: black;
font-weight: bold;
}
TD.LoginFooter
{
font: 12px tahoma,verdana,arial;
color: black;
}
A.LogIn:link
{
color: black;
text-decoration: none;
font: 10pt Tahoma, Verdana, Arial;
font-weight: bold;
}

A.LogIn:active
{
color: black;
text-decoration: none;
font: 10pt Tahoma, Verdana, Arial;
font-weight: bold;
}

A.LogIn:visited
{
color: black;
text-decoration: none;
font: 10pt Tahoma, Verdana, Arial;
font-weight: bold;
}

A.LogIn:hover
{
color: blue;
text-decoration: underline;
font: 10pt Tahoma, Verdana, Arial;
font-weight: bold;
}
SPAN.LogIn
{
color: black;
text-decoration: none;
font: 10pt Tahoma, Verdana, Arial;
font-weight: bold;
}
/*****
END LOG-IN PROPERTIES
*****/

```

```

</STYLE>
">

<CFIF #VARIABLES.gvPSWD_REMEMBER#>

<CFSET rememberPswd = "
<SCRIPT LANGUAGE='JavaScript' SRC='cookies.js'></SCRIPT>
<SCRIPT LANGUAGE='javascript'>
function init()
{
var xL = readCookie('cookieLogin');
var xP = readCookie('cookiePassword');
var xR = readCookie('cookieRemember');
if( xL != null )
document.logonform.xLogInID.value = xL;
if( xP != null )
document.logonform.xPassword.value = xP;
if( xR != null )
document.logonform.xRememberLogin.checked = xR;

document.logonform.xLogInID.focus();
}
</SCRIPT>
">

<CFELSE>

<CFSET rememberPswd = "
<SCRIPT LANGUAGE='javascript'>
function init()
{
document.logonform.xLogInID.focus();
}
</SCRIPT>
">

</CFIF>

<CFSET Middle = "
</HEAD>

<BODY onLOAD='init()'>

<BR>
<BR>
<BR>
<BR>

<FORM ACTION='main.cfm' METHOD=POST NAME='logonform'>

<CENTER>
<TABLE BORDER=0 CELLPADDING=0 CELLSPACING=0>
<TR>
<TD ALIGN='left'>

<TABLE BORDER=0 CELLSPACING=0 CELLPADDING=0 BORDERCOLOR='##A8B0D8'>
<TR BGCOLOR='##A8B0D8'>

```

```

<TD ALIGN='left' WIDTH='23'><IMG
SRC='#VARIABLES.gvIMAGE_DIRECTORY#top_left_corner.gif' WIDTH='23'
HEIGHT='23' BORDER='0'></TD>
<TD ALIGN='left' CLASS='Login'>WELECOME TO READINESS EXPLORER</TD>
<TD ALIGN='right' WIDTH='23'><IMG
SRC='#VARIABLES.gvIMAGE_DIRECTORY#top_right_corner.gif' WIDTH='23'
HEIGHT='23' BORDER='0'></TD>
</TR>
</TABLE>

</TD>
</TR>

<TR>
<TD ALIGN='left'>
<TABLE BORDER='1' CELLSPACING='0' CELLPADDING='0'
BORDERCOLOR='##A8B0D8'>
<TR>
<TD>
<TABLE BORDER='0' WIDTH='100%'>
<TR>
<TD VALIGN='middle' ALIGN='left'>
<IMG SRC='#VARIABLES.gvIMAGE_DIRECTORY#reLogo_Small.jpg' BORDER=0
ALT='Readiness Explorer'>
</TD>
<TD VALIGN='top' ALIGN='left'>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS='Login'>Login ID:</TD><TD><INPUT type='password'
name='xLogInID' size='20' value=''></TD>
</TR>
<TR>
<TD CLASS='Login'>Password:</TD><TD><INPUT type='password'
name='xPassword' size='20' value=''></TD>
</TR>
<TR>
<TD>&nbsp;&nbsp;&nbsp;</TD>
<TD ALIGN='center'>
<TABLE BORDER='0'>
<TR>
<TD NOWRAP ALIGN='center' VALIGN='middle'><A CLASS='LogIn'
HREF='javascript:document.logonform.submit()'>Log In</A></TD>
</TR>
">

<CFIF #VARIABLES.gvPSWD_REMEMBER#>
<CFSET rememberPswd_Block = "
<TR>
<TD NOWRAP ALIGN='center' VALIGN='middle'><INPUT TYPE='checkbox'
VALUE='yes' NAME='xRememberLogin'><SPAN CLASS='LogIn'>Remember
Login</DIV></TD>
</TR>
">
<CFELSE>
<CFSET rememberPswd_Block = "">
</CFIF>

```

```

<CFIF #VARIABLES.gvPSWD_FORGET#>
<CFSET forgetPswd_Block = "
<TR>
<TD NOWRAP ALIGN='center' VALIGN='middle'><A CLASS='LogIn'
HREF='javascript:document.logonform.submit()'>Forgot Password?</A></TD>
</TR>
">
<CFELSE>
<CFSET forgetPswd_Block = "">
</CFIF>

<CFSET Ending = "
</TABLE>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>

</TD>
</TR>

<TR>
<TD ALIGN='center' CLASS='LoginFooter'>
<BR>
You have entered a Department of Defense (DoD) web site<BR>
maintained by #VARIABLES.gvAPP_COMMAND#. This site is monitored and
records all activity.
</TD>
</TR>
</TABLE>
</CENTER>
</FORM>
</BODY>
</HTML>
">

<CFLOCK TIMEOUT="60">
<CFFILE
ACTION="WRITE"
FILE="#filetowrite#"
OUTPUT="#VARIABLES.beginning# #VARIABLES.rememberPswd#
#VARIABLES.Middle# #VARIABLES.rememberPswd_Block#
#VARIABLES.forgetPswd_Block# #VARIABLES.Ending#"
>
</CFLOCK>

```

4. PANEL COMPONENT FILES

COMPSENDINSTANTEMAIL.CFM

```

<CFIF IsDefined( "FORM.msgSubject" )>

<CFQUERY NAME="getUserEmail"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      email, smtpAddress, smtpAddressPort
FROM        tblUserAccounts, tblMyEmailSetup
WHERE       tblUserAccounts.user_id = #SESSION.user_id#
AND         myEmailSetup_id = 1

</CFQUERY>

<CFMAIL
FROM="#getUserEmail.email#"
SUBJECT="#FORM.msgSubject#"
TO="#FORM.msgTo#"
TYPE="HTML"
SERVER="#getUserEmail.smtpAddress#"
PORT="#getUserEmail.smtpAddressPort#"
>
#FORM.msgBody#
</CFMAIL>

</CFIF>

<STYLE TYPE="text/css">
.msgEdit
{
font: 8pt Tahoma, Verdana, Arial;
background-color: #FFFFFF;
border: 1px solid black;
}

.toList
{
background-color: FFFFFFFF;
font: 8pt Tahoma, Verdana, Arial;
}

SPAN.msgText
{
font: 8pt Tahoma, Verdana, Arial;
font-weight: bold;
color: blue;
text-decoration: none;
}

A.sendMsg:link
{
color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}

A.sendMsg:active
{

```

```

color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}

A.sendMsg:visited
{
color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}

A.sendMsg:hover
{
color: blue;
text-decoration: underline;
font-size: 10pt;
}

</STYLE>

<CFQUERY NAME="getAddressBook"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      cLname & ', ' & cFname AS fullName,cEmailAddress
FROM      tblMyAddressBook
WHERE     user_id = #SESSION.user_id#

</CFQUERY>

<CFFORM NAME="instantEmail" ACTION="main.cfm?a=myAccount">
<SPAN CLASS="msgText">To:</SPAN><BR>
<SELECT
CLASS="toList"
NAME="msgTo"
SIZE="1"
>
<CFLOOP QUERY="getAddressBook">
<CFOUTPUT>
<OPTION VALUE="#cEmailAddress#">#fullName#
</CFOUTPUT>
</CFLOOP>
</SELECT>
<SPAN CLASS="msgText">Subject:</SPAN><BR><INPUT CLASS="msgEdit"
TYPE="text" NAME="msgSubject" SIZE="20" VALUE="" MAXLENGTH="100"
REQUIRED="no"><BR>
<SPAN CLASS="msgText">Message:</SPAN><BR><TEXTAREA CLASS="msgEdit"
COLS=20 ROWS=5 NAME="msgBody" WRAP="soft"></TEXTAREA>
<BR>
<BR>
<A CLASS="sendMsg"
HREF="javascript:document.instantEmail.submit();">Send Message</A>
</CFFORM>

```

COMPSTOCKGRABBER.CFM

```
<STYLE TYPE="text/css">
.msgEdit
{
font: 8pt Tahoma, Verdana, Arial;
background-color: #FFFFFF;
border: 1px solid black;
}
.toList
{
background-color: FFFF;
font: 8pt Tahoma, Verdana, Arial;
}

SPAN.msgText
{
font: 8pt Tahoma, Verdana, Arial;
font-weight: bold;
color: blue;
text-decoration: none;
}

A.sendMsg:link
{
color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}

A.sendMsg:active
{
color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}

A.sendMsg:visited
{
color: blue;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}

A.sendMsg:hover
{
color: blue;
text-decoration: underline;
font-size: 10pt;
}

</STYLE>
<!--
```

```

<CF_srStockGrabber
TickerSymbols="smmix"
QueryName="GetQuotes"
ErrorCheck="Yes"
>
---->
<CFOUTPUT>
<B>My Stocks:</B>
<BR>
<CFIF IsDefined( "QUERY.GetQuotes" )>
#GetQuotes.Symbol#<BR>
#GetQuotes.Exchange#<BR>
#GetQuotes.Last_Traded_Price#<BR>
#GetQuotes.Change#<BR>
#GetQuotes.Last_Traded_Time#<BR>
#GetQuotes.Last_Traded_Date#<BR>
#GetQuotes.Opening_Price#<BR>
#GetQuotes.Days_High#<BR>
#GetQuotes.Days_Low#<BR>
#GetQuotes.Volume#<BR>
<CFELSE>
Not Available
</CFIF>
</CFOUTPUT>

```

5. JAVASCRIPT FILES

COOKIES.JS

```

// Cookie Functions
// save/read/delete cookie functions for storing small chunks of data
in the browser
// 19990326

// Copyright (C) 1999 Dan Steinman
// Distributed under the terms of the GNU Library General Public
License
// Available at http://www.dansteinman.com/dynapi/

// thanks to: Jese Chisholm <JCHISHOLM@SENSORMATIC-VPD.com>

function saveCookie(name,value,days) {
if (days) {
var date = new Date();
date.setTime(date.getTime()+(days*24*60*60*1000))
var expires = "; expires="+date.toGMTString()
}
else expires = ""
document.cookie = name+"="+value+expires+"; path=/"
}
function readCookie(name) {
var nameEQ = name + "="
var ca = document.cookie.split(';')
for(var i=0;i<ca.length;i++) {
var c = ca[i];
while (c.charAt(0)==' ') c = c.substring(1,c.length)

```

```

if (c.indexOf(nameEQ) == 0) return c.substring(nameEQ.length,c.length)
}
return null
}
function deleteCookie(name) {
saveCookie(name,"",-1)
}

```

WDDX.JS

```

////////////////////////////////////
////
//  serializeValue() serializes any value that can be serialized
//  returns true/false
function wddxSerializer_serializeValue(obj)
{
var bSuccess = true;

if (typeof(obj) == "string")
{
// String value
this.serializeString(obj);
}
else if (typeof(obj) == "number")
{
// Number value
this.write("<number>" + obj + "</number>");
}
else if (typeof(obj) == "boolean")
{
// Boolean value
this.write("<boolean value='" + obj + "'/>");
}
else if (typeof(obj) == "object")
{
if (obj == null)
{
// Null values become empty strings
this.write("<string></string>");
}
else if (typeof(obj.wddxSerialize) == "function")
{
// Object knows how to serialize itself
bSuccess = obj.wddxSerialize(this);
}
else if (
typeof(obj.join) == "function" &&
typeof(obj.reverse) == "function" &&
typeof(obj.sort) == "function" &&
typeof(obj.length) == "number")
{
this.write("<array length='" + obj.length + "'>");
for (var i = 0; bSuccess && i < obj.length; ++i)
{
bSuccess = this.serializeValue(obj[i]);
}
}
}
}

```

```

this.write("</array>");
}
else if (
typeof(obj.getTimezoneOffset) == "function" &&
typeof(obj.toGMTString) == "function")
{
// Possible Date
this.write( "<dateTime>" +
obj.getYear() + "-" + (obj.getMonth() + 1) + "-" + obj.getDate() +
"T" + obj.getHours() + ":" + obj.getMinutes() + ":" +
obj.getSeconds());
if (this.useTimezoneInfo)
{
this.write(this.timezoneString);
}
this.write("</dateTime>");
}
else
{
// Some generic object; treat it as a structure
this.write("<struct>");
for (var prop in obj)
{
bSuccess = this.serializeVariable(prop, obj[prop]);
if (! bSuccess)
{
break;
}
}
this.write("</struct>");
}
else
{
// Error: undefined values or functions
bSuccess = false;
}

// Successful serialization
return bSuccess;
}

```

```

////////////////////////////////////
////
//   serializeString() serializes a string using JavaScript
functionality
//   available in NS 3.0 and above
function wddxSerializer_serializeString(s)
{
this.write("<string>");
for (var i = 0; i < s.length; ++i)
{
this.write(this.et[s.charAt(i)]);
}
this.write("</string>");
}

```

```

////////////////////////////////////
////
//    serializeStringOld() serializes a string using JavaScript
//    functionality
//    available in IE 3.0
function wddxSerializer_serializeStringOld(s)
{
this.write("<string><![CDATA[");

pos = s.indexOf("]]>");
if (pos != -1)
{
startPos = 0;
while (pos != -1)
{
this.write(s.substring(startPos, pos) + "]]>]]&gt;<![CDATA[");

startPos = pos + 3;
if (startPos < s.length)
{
pos = s.indexOf("]]>", startPos);
}
else
{
// Work around bug in indexOf()
// "" will be returned instead of -1 if startPos > length
pos = -1;
}
}
this.write(s.substring(startPos, s.length));
}
else
{
this.write(s);
}

this.write("]]></string>");
}

////////////////////////////////////
////
//    serializeVariable() serializes a property of a structure
//    returns true/false
function wddxSerializer_serializeVariable(name, obj)
{
var bSuccess = true;

if (typeof(obj) != "function")
{
this.write("<var name='" + name.toUpperCase() + "'>");
bSuccess = this.serializeValue(obj);
this.write("</var>");
}
}

```

```
return bSuccess;
}
```

```
////////////////////////////////////
////
// write() appends text to the wddxPacket buffer
function wddxSerializer_write(str)
{
this.wddxPacket += str;
}
```

```
////////////////////////////////////
////
// serialize() creates a WDDX packet for a given object
// returns the packet on success or null on failure
function wddxSerializer_serialize(rootObj)
{
this.wddxPacket = "";

this.write("<wddxPacket version='0.9'><header/><data>");
var bSuccess = this.serializeValue(rootObj);
this.write("</data></wddxPacket>");

if (bSuccess)
{
return this.wddxPacket;
}
else
{
return null;
}
}
```

```
////////////////////////////////////
////
// WddxSerializer() binds the function properties of the object
function WddxSerializer()
{
// Compatibility section
if (navigator.appVersion != "" && navigator.appVersion.indexOf("MSIE
3.") == -1)
{
// Character encoding table

// Encoding table
var et = new Array();

// Numbers to characters table and
// characters to numbers table
var n2c = new Array();
var c2n = new Array();

for (var i = 0; i < 256; ++i)
{
```

```

// Build a character from octal code
var d1 = Math.floor(i/64);
var d2 = Math.floor((i%64)/8);
var d3 = i%8;
var c = eval("\"\\\"\\\" + d1.toString(10) + d2.toString(10) +
d3.toString(10) + \"\\\"");

// Modify character-code conversion tables
n2c[i] = c;
c2n[c] = i;

// Modify encoding table
if (i < 32 && i != 9 && i != 10 && i != 13)
{
// Control characters that are not tabs, newlines, and carriage returns

// Create a two-character hex code representation
var hex = i.toString(16);
if (hex.length == 1)
{
hex = "0" + hex;
}

et[n2c[i]] = "<char code='" + hex + "'/>";
}
else if (i < 128)
{
// Low characters that are not special control characters
et[n2c[i]] = n2c[i];
}
else
{
// High characters
et[n2c[i]] = "&#x" + i.toString(16) + ";";
}
}

// Special escapes
et["<"] = "&lt;";
et[">"] = "&gt;";
et["&"] = "&amp;";

// Store tables
this.n2c = n2c;
this.c2n = c2n;
this.et = et;

// The browser is not MSIE 3.x
this.serializeString = wddxSerializer_serializeString;
}
else
{
// The browser is most likely MSIE 3.x, it is NS 2.0 compatible
this.serializeString = wddxSerializer_serializeStringOld;
}

// Setup timezone information

```

```

var tzOffset = (new Date()).getTimezoneOffset();

// Invert timezone offset to convert local time to UTC time
if (tzOffset >= 0)
{
this.timezoneString = '-';
}
else
{
this.timezoneString = '+';
}
this.timezoneString += Math.floor(Math.abs(tzOffset) / 60) + ":" +
(Math.abs(tzOffset) % 60);

// Common properties
this.useTimezoneInfo = true;

// Common functions
this.serialize = wddxSerializer_serialize;
this.serializeValue = wddxSerializer_serializeValue;
this.serializeVariable = wddxSerializer_serializeVariable;
this.write = wddxSerializer_write;
}

////////////////////////////////////
////
//
//      WddxRecordset
//
////////////////////////////////////
////

////////////////////////////////////
////
//      getRowCount() returns the number of rows in the recordset
function wddxRecordset_getRowCount()
{
var nRowCount = 0;
for (var col in this)
{
if (typeof(this[col]) == "object")
{
nRowCount = this[col].length;
break;
}
}
return nRowCount;
}

////////////////////////////////////
////
//      addColumn(name) adds a column with that name and length ==
getRowCount()

```

```

function wddxRecordset_addColumn(name)
{
var nLen = this.getRowCount();
var colValue = new Array(nLen);
for (var i = 0; i < nLen; ++i)
{
colValue[i] = null;
}
this[name] = colValue;
}

////////////////////////////////////
////
//   addRows() adds n rows to all columns of the recordset
function wddxRecordset_addRows(n)
{
for (var col in this)
{
var nLen = this[col].length;
for (var i = nLen; i < nLen + n; ++i)
{
this[col][i] = null;
}
}
}

////////////////////////////////////
////
//   getField() returns the element in a given (row, col) position
function wddxRecordset_getField(row, col)
{
return this[col][row];
}

////////////////////////////////////
////
//   setField() sets the element in a given (row, col) position to
//   value
function wddxRecordset_setField(row, col, value)
{
this[col][row] = value;
}

////////////////////////////////////
////
//   wddxSerialize() serializes a recordset
//   returns true/false
function wddxRecordset_wddxSerialize(serializer)
{
// Create an array and a list of column names
var colNamesList = "";
var colNames = new Array();
var i = 0;

```

```

for (var col in this)
{
if (typeof(this[col]) == "object")
{
colNames[i++] = col;

if (colNamesList.length > 0)
{
colNamesList += ",";
}
colNamesList += col;
}
}

var nRows = this.getRowCount();

serializer.write("<recordset rowCount='" + nRows + "' fieldNames='" +
colNamesList + "'>");

var bSuccess = true;
for (i = 0; bSuccess && i < colNames.length; i++)
{
var name = colNames[i];
serializer.write("<field name='" + name + "'>");

for (var row = 0; bSuccess && row < nRows; row++)
{
bSuccess = serializer.serializeValue(this[name][row]);
}

serializer.write("</field>");
}

serializer.write("</recordset>");

return bSuccess;
}

////////////////////////////////////
////
// dump(escapeStrings) returns an HTML table with the recordset data
// It is a convenient routine for debugging and testing recordsets
// The boolean parameter escapeStrings determines whether the <>&
// characters in string values are escaped as &lt;&gt;&amp;
function wddxRecordset_dump(escapeStrings)
{
// Get row count
var nRows = this.getRowCount();

// Determine column names
var colNames = new Array();
var i = 0;
for (var col in this)
{
if (typeof(this[col]) == "object")
{

```

```

colNames[i++] = col;
}
}

// Build table headers
var o = "<table border=1><tr><td><b>RowNumber</b></td>";
for (i = 0; i < colNames.length; ++i)
{
o += "<td><b>" + colNames[i] + "</b></td>";
}
o += "</tr>";

// Build data cells
for (var row = 0; row < nRows; ++row)
{
o += "<tr><td>" + row + "</td>";
for (i = 0; i < colNames.length; ++i)
{
var elem = this.getField(row, colNames[i]);
if (escapeStrings && typeof(elem) == "string")
{
var str = "";
for (var j = 0; j < elem.length; ++j)
{
var ch = elem.charAt(j);
if (ch == '<')
{
str += "&lt;";
}
else if (ch == '>')
{
str += "&gt;";
}
else if (ch == '&')
{
str += "&amp;";
}
else
{
str += ch;
}
}
o += ("<td>" + str + "</td>");
}
else
{
o += ("<td>" + elem + "</td>");
}
}
o += "</tr>";
}

// Close table
o += "</table>";

// Return HTML recordset dump
return o;

```

```
}
```

```
////////////////////////////////////  
////
```

```
// WddxRecordset() creates an empty recordset  
// WddxRecordset(columns) creates a recordset with these columns  
// WddxRecordset(columns, rows) creates a recordset with these columns  
and some number of rows
```

```
function WddxRecordset()  
{  
  this.getRowCount = wddxRecordset_getRowCount;  
  this.addColumn = wddxRecordset_addColumn;  
  this.addRows = wddxRecordset_addRows;  
  this.getField = wddxRecordset_getField;  
  this.setField = wddxRecordset_setField;  
  this.wddxSerialize = wddxRecordset_wddxSerialize;  
  this.dump = wddxRecordset_dump;
```

```
  if (WddxRecordset.arguments.length > 0)  
  {  
    var cols = WddxRecordset.arguments[0];  
    var nLen = WddxRecordset.arguments.length > 1 ?  
    WddxRecordset.arguments[1] : 0;
```

```
    for (var i = 0; i < cols.length; ++i)  
    {  
      var colValue = new Array(nLen);  
      for (var j = 0; j < nLen; ++j)  
      {  
        colValue[j] = null;  
      }  
    }  
  }  
  this[cols[i]] = colValue;  
}
```

```
  }  
}
```

RE2001.JS

```
<!--
```

```
function setStatus( strStatusLine )  
{  
  window.status = strStatusLine;  
}
```

```
// make sure user really wants to delete record  
function EnsureDeleteRecord()  
{  
  if ( confirm( "Are you sure you want to delete this record?" ) )  
  {  
    return( true );  
  }  
  else  
  return( false );  
}
```

```

}

function quickList( element )
{
if ( element.value != "" )
location = element.value;
}

<!-- Original: Nannette Thacker (nannette@shiningstar.net) -->
<!-- Web Site: http://www.shiningstar.net -->

<!-- This script and many more are available free online at -->
<!-- The JavaScript Source!! http://javascript.internet.com -->

<!-- Begin
var version4 = (navigator.appVersion.charAt(0) == "4");
var popupHandle;
function closePopup() {
if(popupHandle != null && !popupHandle.closed) popupHandle.close();
}
function displayPopup(position,url,name,height,width,evt) {
// position=1 POPUP: makes screen display up and/or left, down and/or
right
// depending on where cursor falls and size of window to open
// position=2 CENTER: makes screen fall in center
var properties = "toolbar = 0, location = 0, height = " + height;
properties = properties + ", width=" + width;
var leftprop, topprop, screenX, screenY, cursorX, cursorY, padAmt;
if(navigator.appName == "Microsoft Internet Explorer") {
screenY = document.body.offsetHeight;
screenX = window.screen.availWidth;
}
else {
screenY = window.outerHeight
screenX = window.outerWidth
}
if(position == 1) { // if POPUP not CENTER
cursorX = evt.screenX;
cursorY = evt.screenY;
padAmtX = 10;
padAmtY = 10;
if((cursorY + height + padAmtY) > screenY) {
// make sizes a negative number to move left/up
padAmtY = (-30) + (height * -1);
// if up or to left, make 30 as padding amount
}
if((cursorX + width + padAmtX) > screenX) {
padAmtX = (-30) + (width * -1);
// if up or to left, make 30 as padding amount
}
if(navigator.appName == "Microsoft Internet Explorer") {
leftprop = cursorX + padAmtX;
topprop = cursorY + padAmtY;
}
else {
leftprop = (cursorX - pageXOffset + padAmtX);
topprop = (cursorY - pageYOffset + padAmtY);
}
}
}

```

```

}
}
else{
leftvar = (screenX - width) / 2;
rightvar = (screenY - height) / 2;
if(navigator.appName == "Microsoft Internet Explorer") {
leftprop = leftvar;
topprop = rightvar;
}
else {
leftprop = (leftvar - pageXOffset);
topprop = (rightvar - pageYOffset);
}
}
if(evnt != null) {
properties = properties + ", left = " + leftprop;
properties = properties + ", top = " + topprop;
}
closePopup();
popupHandle = open(url,name,properties);
}

//-->

```

6. GET INFORMATION FILES

GETCONTENT.CFM

```

<!-- INCLUDE DEFAULTS JUST IN CASE USER DEFINED CONFIG FILE NOT FOUND
--->
<CFINCLUDE TEMPLATE="getProgramDefaults.cfm">

<CFSWITCH EXPRESSION="#ATTRIBUTES.Type#">
<CFCASE VALUE="file">
<CFIF #ATTRIBUTES.File# NEQ "">
<CFIF #FileExists( GetDirectoryFromPath( GetTemplatePath() ) &
ATTRIBUTES.File )#>
<CFINCLUDE TEMPLATE="#ATTRIBUTES.File#">
<CFELSE>
File Not Available.
</CFIF>
<CFELSE>
File Not Available.
</CFIF>
</CFCASE>
<CFCASE VALUE="url">
<CFHTTP URL="#ATTRIBUTES.URL#" RESOLVEURL="yes">
<CFOUTPUT>
#CFHTTP.FileContent#
</CFOUTPUT>
</CFCASE>
<CFCASE VALUE="inline">
<CFOUTPUT>
#ATTRIBUTES.InLine#
</CFOUTPUT>
</CFCASE>

```

</CFSWITCH>

GETGREETING.CFM

```
<STYLE TYPE="text/css">
SPAN.getGreeting
{
color: black;
text-decoration: none;
font: 10pt Tahoma, Verdana, Arial;
font-weight: bold;
}
</STYLE>
<CFINCLUDE TEMPLATE="getProgramDefaults.cfm">

<!-- get user's record -->
<CFQUERY NAME="GetUserRecord"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

        SELECT      *
        FROM          tblUserAccounts
        WHERE         user_loginid = '#SESSION.loginid#'

</CFQUERY>

<CFOUTPUT QUERY="GetUserRecord">

<CFSET username = #rank# & " " & #fname# & " " & #lname#>

<CFIF ( #Hour( Now() )# GTE 0 ) AND ( #Hour( Now() )# LT 12 )>
        <SPAN CLASS="getGreeting">Good Morning
#VARIABLES.username#</SPAN>
<CFELSEIF ( #Hour( Now() )# GTE 12 ) AND ( #Hour( Now() )# LT 18 )>
        <SPAN CLASS="getGreeting">Good Afternoon
#VARIABLES.username#</SPAN>
<CFELSEIF ( #Hour( Now() )# GTE 18 ) AND ( #Hour( Now() )# LTE 23 )>
        <SPAN CLASS="getGreeting">Good Evening
#VARIABLES.username#</SPAN>
</CFIF>

</CFOUTPUT>
```

GETINFORMATION.CFM

```
<CFQUERY NAME="getInformation"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      *
FROM        tblWhatsNew
ORDER BY    infoDateCreated DESC

</CFQUERY>

<TABLE BORDER=0 WIDTH=100%>
<CFOUTPUT QUERY="getInformation">
```

```

<TR>
<TD CLASS="adWindowBody">
<B>#infoTitle#</B><BR>
#infoBody#
</TD>
</TR>
</CFOUTPUT>
<TR>
<TD CLASS="adWindowBody" ALIGN="center">
<HR SIZE=1 WIDTH=75%>
<A CLASS="basic"
href="main.cfm?a=administration&body=adminWhatsNew.cfm">Add/Edit What's
New Information</A>
</TD>
</TR>
</TABLE>

```

GETMESSAGE.CFM

```

<CFPOP
SERVER="saipan.nps.navy.mil"
USERNAME="mwhiteca"
PASSWORD="mjma2B01"
ACTION="GetAll"
MESSAGENUMBER="#URL.msgid#"
NAME="GetMessage"
>

<CFOUTPUT QUERY="GetMessage">
Message Number: #HTMLFormat( MessageNumber )#<BR>
To: #HTMLFormat( To )#<BR>
From: #HTMLFormat( From )#<BR>
Subject: #HTMLFormat( Subject )#<BR>
Date: #HTMLFormat( Date )#<BR>
From: #HTMLFormat( From )#<BR>
CC: #HTMLFormat( cc )#<BR>
Reply To: #HTMLFormat( replyto )#<BR>
Header: #HTMLFormat( header )#<BR><BR>
Body: #HTMLFormat( body )#<BR>

</CFOUTPUT>

```

GETPAGECOMMENTS.CFM

```

<CFOUTPUT>
<!--
*****
*****
*** PAGE NAME:
    #aComponent[URL.aID][cAPPLICATION][cAPPLICATION_NAME]#
*** DESCRIPTION:
    #aComponent[URL.aID][cAPPLICATION][cAPPLICATION_DESCRIPTION]#
*** DATE CREATED:
    #aComponent[URL.aID][cAPPLICATION][cAPPLICATION_DATECREATED]#

```

```

*** CREATED BY:
    #aComponent[URL.aID][cAPPLICATION][cAPPLICATION_CREATEDBY]#
*** LAST UPDATED:
    #aComponent[URL.aID][cAPPLICATION][cAPPLICATION_LASTUPDATED]#
*** UPDATED BY:
    #aComponent[URL.aID][cAPPLICATION][cAPPLICATION_UPDATEDBY]#
*** COMMENTS:
    #aComponent[URL.aID][cAPPLICATION][cAPPLICATION_CARGO1]#
*****
*****
-->
</CFOUTPUT>

```

GETPROGRAMDEFAULTS.CFM

```

<!-- PROGRAM SPECIFIC -->
<CFSET gvHOMEPAGE_NAME = 'index.htm'>

<!-- DATASOURCE DEFAULTS -->
<CFSET gvDATASOURCE_USER = 'reUSERS'>
<CFSET gvDATASOURCE_DATA = 'reDATA'>
<CFSET gvDATASOURCE_CONFIG = 'reCONFIG'>
<CFSET gvDATASOURCE_ARCHIVE = 'reARCHIVE'>

<!-- ADMINISTRATION -->
<CFSET gvAdminName = 'Michael Whitecar'>
<CFSET gvAdminPhone = '831.393.1087'>
<CFSET gvAdminEmail = 'mwhitecar@med.navy.mil'>
<CFSET gvAdminName_Sec = ''>
<CFSET gvAdminPhone_Sec = ''>
<CFSET gvAdminEmail_Sec = ''>

<!-- DIRECTORIES DEFAULTS -->
<CFSET gvFILEPATH = 'C:\WINDOWS\Profiles\Michael
Whitecar\My Documents\WebApplications\ReadinessExplorer\2000\'>
<CFSET gvIMAGE_DIRECTORY = 'images/'>
<CFSET gvJS_DIRECTORY = 'js/'>
<CFSET gvUSER_DIRECTORY = 'users/'>
<CFSET gvRESOURCE_DIRECTORY = 'Resources'>

<!-- RESOURCES -->
<CFSET gvMS_RESOURCE_CONVERTERS =
'http://www.microsoft.com/office/000/viewers.htm'>

<!-- SECURITY ACCESS DEFAULTS -->
<CFSET gvLEVEL_ADMIN = '10'>

<!-- ORGANIZATION -->
<CFSET gvAPP_HOME = 'http://127.0.0.1/re2001/'>
<CFSET gvAPP_COMMAND = 'Michael Whitecars Test Site'>
<CFSET gvHOME_PAGE =
'http://www.geocities.com/queuelifedevelopment/'>
<CFSET gvSMTP_SERVER = '127.0.0.1'>
<CFSET gvFromEmail = 'mikewhitecar@cs.com'>

```

```

<!-- SECURITY --->
<CFSET gvTIMEOUT_MINS           = '90'>
<CFSET gvTIMEOUT_SECS          = 5340>
<CFSET gvSECURITYEMAIL         = 'No'>
<CFSET gvLOGON_ATTEMPTS        = '3'>
<CFSET gvIP_BLOCKING           = 'No'>
<CFSET gvDOW_BLOCKING          = 'No'>
<CFSET gvDOMAIN_BLOCKING       = 'Yes'>
<CFSET gvDOMAIN_BLOCKINGLIST   = 'mil'>
<CFSET gvDOW_DAYSTOBLOCK       = ''>
<CFSET gvTIME_BLOCKING         = 'No'>
<CFSET gvTIME_FROMBLOCK        = '00:00'>
<CFSET gvTIME_TOBLOCK          = '00:00'>
<CFSET gvDEFAULTPSWDEXP        = '60'>
<CFSET gvMIN_PSWD_SIZE         = '8'>
<CFSET gvPSWD_REUSE            = 'No'>
<CFSET gvPSWD_FORGET           = 'Yes'>
<CFSET gvPSWD_REMEMBER         = 'Yes'>
<CFSET gvPSWD_FORGET_CHG       = 'No'>
<CFSET gvPSWD_LOCKIFFAIL       = 'No'>
<CFSET gvPSWD_SENDEMAIL        = 'Yes'>
<CFSET gvPSWD_ISCASE           = 'Yes'>
<CFSET gvPSWD_INCLNUM          = 'No'>
<CFSET gvMSG_FAILEDLOGON       = 'Unauthorized logon!'>
<CFSET gvMSG_USEREXPIRE        = 'Account has expired!'>
<CFSET gvMSG_RECORDLOCKED      = 'Your record has been locked! Please
contact the administrator.'>
<CFSET gvMSG_IPBLOCKING        = 'Not authorized from your location!'>
<CFSET gvMSG_DOMAINBLOCKING    = 'Not authorized from your domain!'>
<CFSET gvMSG_DOWBLOCKING       = 'Not authorized during this day!'>
<CFSET gvMSG_TIMEBLOCKING      = 'Not authorized during this time!'>

<!-- SECURITY ACCESS DEFAULTS --->
<CFSET gvLEVEL_ADMIN           = '10'>

```

GETSECURITY.CFM

```

<CFIF NOT IsDefined( "URL.sID" )>

<CFOUTPUT>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="adWindowBodyBold">Last Visit:</TD>
<TD CLASS="adWindowBody">#DateFormat( SESSION.LastLogOn, 'dd-mmm-yyyy'
)# at #TimeFormat( SESSION.LastLogOn, 'H:mm' )#</TD>
</TR>
<TR>
<TD CLASS="adWindowBodyBold">Today's Date:</TD>
<TD CLASS="adWindowBody">#DateFormat( Now(), 'dd-mmm-yyyy' )# at
#TimeFormat( Now(), 'H:mm' )#</TD>
</TR>
<TR>
<TD CLASS="adWindowBodyBold">Last Login Address:</TD>
<TD CLASS="adWindowBody">#CLIENT.LastIP#</TD>
</TR>
<TR>

```

```

<TD CLASS="adWindowBodyBold">Curent Login Address :</TD>
<TD CLASS="adWindowBody">#CLIENT.CurrIP#</TD>
</TR>
<TR>
<TD CLASS="adWindowBodyBold">Account Expires:</TD>
<TD CLASS="adWindowBody">#DateFormat( SESSION.accountExpires, 'dd-mmm-
yyyy' )#</TD>
</TR>
<TR>
<TD CLASS="adWindowBodyBold">Password Expires:</TD>
<TD CLASS="adWindowBody">Working</TD>
</TR>
<TR>
<TD CLASS="adWindowBody" ALIGN="center" COLSPAN=2>
<HR SIZE=1 WIDTH=75%>
<A CLASS="basic" HREF="main.cfm?a=myAccount&body=mySecurity.cfm">Goto
mySecurity Page</A>
</TD>
</TR>
</TABLE>
</CFOUTPUT>

</CFIF>

```

GETSTRUCTURE.CFM

```

<!-- get XML configuration --->
<CFINCLUDE TEMPLATE="component_declare.cfm">

<CFFILE
ACTION="read"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#component.cfg"
VARIABLE="DataPacket"
>

<CFWDDX
INPUT="#DataPacket#"
OUTPUT="aComponent"
ACTION="WDDX2CFML"
>

```

GETSTYLE.CFM

```

<!-- convert default SESSION.colors to XML and write to disk --->
<CFIF NOT #FileExists( GetDirectoryFromPath( GetTemplatePath() ) &
"desktop.cfg" )#>

<CFQUERY NAME="getColorElements"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT ceDefaultValue,ceCSSName
FROM tblColorElements

</CFQUERY>

```

```

<!-- declare array to hold color structures --->
<CFSET myColors = ArrayNew(2)>

<!-- declare color structure --->
<CFSET SESSION.colors = StructNew(>

<!-- populate structure with color data --->
<CFLOOP QUERY="getColorElements">
<CFSET temp = #StructInsert( SESSION.colors, ceCSSName, ceDefaultValue
) #>
</CFLOOP>

<CFSET myColors[1][1] = "Default">
<CFSET myColors[1][2] = SESSION.colors>

<CFWDDX
INPUT="#myColors#"
OUTPUT="NewPacket"
ACTION="CFML2WDDX"
>
<CFLOCK TIMEOUT="60">
<CFFILE
ACTION="write"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#desktop.cfg"
OUTPUT="#NewPacket#"
>
</CFLOCK>

</CFIF>

<CFIF NOT IsDefined( "SESSION.colors" )>

<!-- open desktop.cfg file and convert back to CFM --->
<CFLOCK TIMEOUT="60">
<CFFILE
ACTION="read"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#desktop.cfg"
VARIABLE="DataPacket"
>
</CFLOCK>

<CFWDDX
INPUT="#DataPacket#"
OUTPUT="myColors"
ACTION="WDDX2CFML"
>

<!-- get structure of colors according to user preference --->
<CFSET colorFlag = False>
<CFIF #SESSION.desktop# EQ "" OR #SESSION.desktop# EQ "Default">

<CFSET SESSION.colors = myColors[1][2]>

<CFELSE>

<CFLOOP INDEX="counter" FROM="1" TO="#ArrayLen( myColors )#">

```

```

<CFIF myColors[#counter#][1] EQ "#SESSION.deskTop#">

<CFSET SESSION.colors = myColors[#counter#][2]>
<CFSET colorFlag = True>
<CFBREAK>

</CFIF>

</CFLOOP>

<CFIF NOT colorFlag>

<CFSET SESSION.colors = myColors[1][2]>

</CFIF>

</CFIF>

</CFIF>

<CFOUTPUT>
/* Main Home Page */
Body.Home
{
background-color: #Iif( StructKeyExists( SESSION.colors,"Body.Home"
),De( StructFind( SESSION.colors,"Body.Home" ) ),De( 'ffffff' ) )#;
font: 10px Tahoma, Verdana, Arial;
margin-top: 3;
margin-left: 5;

}

HR.standard
{
color: #Iif( StructKeyExists( SESSION.colors,"HR.standard" ),De(
StructFind( SESSION.colors,"HR.standard" ) ),De( '666699' ) )#;
}

/*****
QUICKLINK PROPERTIES
*****/
A.quickLink:link
{
color: #Iif( StructKeyExists( SESSION.colors,"A.quickLink:link" ),De(
StructFind( SESSION.colors,"A.quickLink:link" ) ),De( 'ffffff' ) )#;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.quickLink:active
{
color: #Iif( StructKeyExists( SESSION.colors,"A.quickLink:active" ),De(
StructFind( SESSION.colors,"A.quickLink:active" ) ),De( 'ffffff' ) )#;
text-decoration: none;
font-size: 8pt;

```

```

font-weight: bold;
}

A.quickLink:visited
{
color: #Iif( StructKeyExists( SESSION.colors,"A.quickLink:visited"
),De( StructFind( SESSION.colors,"A.quickLink:visited" ) ),De( 'ffffff'
) )#;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.quickLink:hover
{
color: #Iif( StructKeyExists( SESSION.colors,"A.quickLink:hover" ),De(
StructFind( SESSION.colors,"A.quickLink:hover" ) ),De( 'ffffff' ) )#;
text-decoration: none;
font-size: 8pt;
}
/*****
END-QUICKLINK PROPERTIES
*****/

/* body of work area */
TD.workAreaBody
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.workAreaBody" ),De( StructFind(
SESSION.colors,"TD.workAreaBody" ) ),De( 'ffffff' ) )#;
}

/* bottom footer of each page */
TD.footer
{
font: 11px Tahoma, Verdana, Arial;
}

/* Toolbar for each page */
A.toolBar:link
{
color: #Iif( StructKeyExists( SESSION.colors,"A.toolBar:link" ),De(
StructFind( SESSION.colors,"A.toolBar:link" ) ),De( 'ffffff' ) )#;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.toolBar:active
{
color: #Iif( StructKeyExists( SESSION.colors,"A.toolBar:active" ),De(
StructFind( SESSION.colors,"A.toolBar:active" ) ),De( 'ffffff' ) )#;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

```

```

A.toolBar:visited
{
color: #Iif( StructKeyExists( SESSION.colors,"A.toolBar:visited" ),De(
StructFind( SESSION.colors,"A.toolBar:visited" ) ),De( 'ffffff' ) )#;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.toolBar:hover
{
color: #Iif( StructKeyExists( SESSION.colors,"A.toolBar:hover" ),De(
StructFind( SESSION.colors,"A.toolBar:hover" ) ),De( 'ffffff' ) )#;
text-decoration: underline;
font-size: 8pt;
}

TD.toolBar
{
background-color: #Iif( StructKeyExists( SESSION.colors,"TD.toolBar"
),De( StructFind( SESSION.colors,"TD.toolBar" ) ),De( '000066' ) )#;
}

/* Main tabs for each page */
A.mainTab:link
{
color: #Iif( StructKeyExists( SESSION.colors,"A.mainTab:link" ),De(
StructFind( SESSION.colors,"A.mainTab:link" ) ),De( 'ffffff' ) )#;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.mainTab:active
{
color: #Iif( StructKeyExists( SESSION.colors,"A.mainTab:active" ),De(
StructFind( SESSION.colors,"A.mainTab:active" ) ),De( 'ffffff' ) )#;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.mainTab:visited
{
color: #Iif( StructKeyExists( SESSION.colors,"A.mainTab:visited" ),De(
StructFind( SESSION.colors,"A.mainTab:visited" ) ),De( 'ffffff' ) )#;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.mainTab:hover
{
color: #Iif( StructKeyExists( SESSION.colors,"A.mainTab:hover" ),De(
StructFind( SESSION.colors,"A.mainTab:hover" ) ),De( 'ffffff' ) )#;
text-decoration: underline;
}

```

```

font-size: 8pt;
}

TD.mainTabSelected
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.mainTabSelected" ),De( StructFind(
SESSION.colors,"TD.mainTabSelected" ) ),De( '666699' ) )#;
}

TD.mainTabUnSelected
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.mainTabUnSelected" ),De( StructFind(
SESSION.colors,"TD.mainTabUnSelected" ) ),De( 'ffcc66' ) )#;
}

/* panel settings for each page */
TD.panelTitleRaised
{
border-top: solid ##FFFFFF 1px;
border-left: solid ##FFFFFF 1px;
border-right: solid ##584300 1px;
border-bottom: solid ##584300 1px;
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.panelTitleRaised" ),De( StructFind(
SESSION.colors,"TD.panelTitleRaised" ) ),De( 'ffcc00' ) )#;
font-family: Arial,sans;
font-size: 11px;
font-weight: bold;
}

TD.panelBackground
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.panelBackground" ),De( StructFind(
SESSION.colors,"TD.panelBackground" ) ),De( 'ffffcc' ) )#;
}

TD.panelComponentRaised
{
border-top: solid ##FFFFFF 1px;
border-left: solid ##FFFFFF 1px;
border-right: solid ##584300 1px;
border-bottom: solid ##584300 1px;
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.panelComponentRaised" ),De( StructFind(
SESSION.colors,"TD.panelComponentRaised" ) ),De( 'ffcc66' ) )#;
font-family: Tahoma,Arial,sans;
font-size: 10px;
font-weight: bold;
}

TD.panelComponentBody
{
font-family: Tahoma,Arial,sans;
font-size: 10px;
}

```

```

}

TABLE.adWindow
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TABLE.adWindow" ),De( StructFind(
SESSION.colors,"TABLE.adWindow" ) ),De( '9999cc' ) )#;
}

TD.adWindowTitle
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.adWindowTitle" ),De( StructFind(
SESSION.colors,"TD.adWindowTitle" ) ),De( '9999cc' ) )#;
font-family: arial,Helvetica;
font-size: 14px;
font-weight: bold;
}

TD.adWindowBody
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.adWindowBody" ),De( StructFind(
SESSION.colors,"TD.adWindowBody" ) ),De( 'ffffff' ) )#;
font-family: arial,Helvetica;
font-size: 12px;
}

TD.adWindowBodyBold
{
font-weight: bold;
font-size: 13px;
}

/*****
REPORT PROPERTIES
*****/
TR.reportFieldLabel
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TR.reportFieldLabel" ),De( StructFind(
SESSION.colors,"TR.reportFieldLabel" ) ),De( 'A8B0D8' ) )#;
font: 10px tahoma,verdana,arial;
color: black;
font-weight: bold;
}

TR.reportFieldDataEven
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TR.reportFieldDataEven" ),De( StructFind(
SESSION.colors,"TR.reportFieldDataEven" ) ),De( 'F7F7F7' ) )#;
font: 8pt tahoma,verdana,arial;
color: black;
}

```

```

TR.reportFieldDataOdd
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TR.reportFieldDataOdd" ),De( StructFind(
SESSION.colors,"TR.reportFieldDataOdd" ) ),De( 'grey' ) )#;
font: 8pt tahoma,verdana,arial;
color: black;

}

TD.reportTitle
{
font: 12px tahoma,verdana,arial;
color: #Iif( StructKeyExists( SESSION.colors,"TD.reportTitle" ),De(
StructFind( SESSION.colors,"TD.reportTitle" ) ),De( '000000' ) )#;
font-weight: bold;
}

TD.reportRowHeader
{
font: 12px tahoma,verdana,arial;
color: #Iif( StructKeyExists( SESSION.colors,"TD.reportRowHeader" ),De(
StructFind( SESSION.colors,"TD.reportRowHeader" ) ),De( '000000' ) )#;
font-weight: bold;
}

TD.reportColumn
{
font: 12px tahoma,verdana,arial;
color: #Iif( StructKeyExists( SESSION.colors,"TD.reportColumn" ),De(
StructFind( SESSION.colors,"TD.reportColumn" ) ),De( '000000' ) )#;
}

A.reportOptions:link
{
color: #Iif( StructKeyExists( SESSION.colors,"A.reportOptions:link"
),De( StructFind( SESSION.colors,"A.reportOptions:link" ) ),De(
'000000' ) )#;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.reportOptions:active
{
color: #Iif( StructKeyExists( SESSION.colors,"A.reportOptions:active"
),De( StructFind( SESSION.colors,"A.reportOptions:active" ) ),De(
'000000' ) )#;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.reportOptions:visited
{

```

```

color: #Iif( StructKeyExists( SESSION.colors,"A.reportOptions:visited"
),De( StructFind( SESSION.colors,"A.reportOptions:visited" ) ),De(
'000000' ) )#;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.reportOptions:hover
{
color: #Iif( StructKeyExists( SESSION.colors,"A.reportOptions:hover"
),De( StructFind( SESSION.colors,"A.reportOptions:hover" ) ),De(
'000000' ) )#;
text-decoration: underline;
font-size: 8pt;
}

/*****
END REPORT PROPERTIES
*****/

/*****
WIN TAB PROPERTIES
*****/
TABLE.winTab
{
border-color: #Iif( StructKeyExists( SESSION.colors,"TABLE.winTab"
),De( StructFind( SESSION.colors,"TABLE.winTab" ) ),De( 'A8B0D8' ) )#;
border-width: thin;
border-style: solid;
}

TR.winTab
{
background-color: #Iif( StructKeyExists( SESSION.colors,"TR.winTab"
),De( StructFind( SESSION.colors,"TR.winTab" ) ),De( 'A8B0D8' ) )#;
font: 8pt tahoma,verdana,arial;
font-weight: bold;
}

TD.winTabSelected
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.winTabSelected" ),De( StructFind(
SESSION.colors,"TD.winTabSelected" ) ),De( 'A8B0D8' ) )#;
font: 8pt tahoma,verdana,arial;
font-weight: bold;
}

TD.winTabUnselected
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.winTabUnSelected" ),De( StructFind(
SESSION.colors,"TD.winTabUnSelected" ) ),De( 'A8B0D8' ) )#;
font: 8pt tahoma,verdana,arial;
font-weight: bold;
}

```

```

TD.winTabLabel
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.winTabLabel" ),De( StructFind(
SESSION.colors,"TD.winTabLabel" ) ),De( 'A8B0D8' ) )#;
font: 12px tahoma,verdana,arial;
color: black;
font-weight: bold;
}

```

```

TD.winTabField
{
background-color: #Iif( StructKeyExists(
SESSION.colors,"TD.winTabField" ),De( StructFind(
SESSION.colors,"TD.winTabField" ) ),De( 'F7F7F7' ) )#;
font: 12px tahoma,verdana,arial;
color: black;
}

```

```

SPAN.winTab
{
font: 10pt Tahoma, Verdana, Arial;
}

```

```

/*****
END WIN TAB PROPERTIES
*****/

```

```

.quickList
{
background-color: FFFFF;
font: 8pt Tahoma, Verdana, Arial;
margin: 4px;
}

```

```

.stdTextBox
{
font: 8pt Tahoma, Verdana, Arial;
background-color: FFFFFFF;
border: 1px solid black;
}

```

```

A.basic:link
{
color: blue;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

```

```

A.basic:active
{
color: blue;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

```

```

}

A.basic:visited
{
color: blue;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.basic:hover
{
color: blue;
text-decoration: underline;
font-size: 8pt;
}

</CFOUTPUT>

```

GETUSERINFORMATION.CFM

```

<CFQUERY NAME="getUserInformation"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      *
FROM    tblUserAccounts
WHERE user_loginid = '#SESSION.loginid#'

</CFQUERY>

```

7. USER FILES

MYACCOUNT.CFM

```

<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="myAccountPage">
</SPAN>

<CF_srPAGEOPTIONS NAME="myAccount">

```

MYADDRESSBOOK.CFM

```

<CFIF NOT IsDefined( "URL.user_id" )>

<CFSET URL.user_id = #SESSION.user_id#>

</CFIF>

<CFPARAM NAME="URL.action" DEFAULT="View">
<CFSET mainFile = "main.cfm?a=myAccount&body=myAddressBook.cfm">
<CFSET mainTable = "tblMyAddressBook">
<CFSET idField = "myAddressBook_id">
<CFSET addTitle = "CREATE A NEW ADDRESS">

```

```

<CFSET editTitle = "EDIT ADDRESS">
<CFSET useSource = "#VARIABLES.gvDATASOURCE_USER#">

<CFIF URL.action EQ "View">

<CF_srReportEngine
ACTION="run"
NAME="listMyAddressBook"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="New"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=add"
ADDWHERE="user_ID=#URL.user_ID#"
>

<CFELSEIF URL.action EQ "Add">

<CFSET title = "#VARIABLES.addTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=saveNew">
<CFSET formName="formAdd">

<CFELSEIF URL.action EQ "Edit">

<CFSET title = "#VARIABLES.editTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=save">
<CFSET formName="formEdit">

<CFELSEIF URL.action EQ "saveNew">

<CFINSERT DATASOURCE="#VARIABLES.useSource#"
TABLENAME="#VARIABLES.mainTable#"
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "save">

<CFUPDATE DATASOURCE="#VARIABLES.useSource#"
TABLENAME="#VARIABLES.mainTable#"
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "delete">

<CFQUERY NAME="deleteInfo" DATASOURCE="#VARIABLES.useSource#">

DELETE
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>
<CFLOCATION URL="#VARIABLES.mainFile#">

</CFIF>

<CFIF URL.action NEQ "View">

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

SELECT      *

```



```

>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Rank:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="cRank"
TYPE="text"
VALUE="#getRecordInfo.cRank#"
SIZE="10"
MAXLENGTH="10"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Title:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="cTitle"
TYPE="text"
VALUE="#getRecordInfo.cTitle#"
SIZE="30"
MAXLENGTH="50"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Organization:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="cOrganization"
TYPE="text"
VALUE="#getRecordInfo.cOrganization#"
SIZE="30"
MAXLENGTH="40"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Email Address:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="cEmailAddress"
TYPE="text"
VALUE="#getRecordInfo.cEmailAddress#"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Homepage:</TD>

```

```

<TD CLASS="winTabField">
<CFINPUT
NAME="cHomepage"
TYPE="text"
VALUE="#getRecordInfo.cHomepage#"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD>&nbsp;&nbsp;&nbsp;</TD>
<TD>&nbsp;&nbsp;&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;TELECOMMUNICATION</TD>
<TD CLASS="winTabField">&nbsp;&nbsp;&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Phone:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="pPhone"
TYPE="text"
VALUE="#getRecordInfo.pPhone#"
SIZE="30"
MAXLENGTH="30"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Fax:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="pFax"
TYPE="text"
VALUE="#getRecordInfo.pFax#"
SIZE="30"
MAXLENGTH="20"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Cell Phone:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="pCellPhone"
TYPE="text"
VALUE="#getRecordInfo.pCellPhone#"
SIZE="30"
MAXLENGTH="20"
REQUIRED="no"
>
</TD>

```

```

</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Pager:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="pPager"
TYPE="text"
VALUE="#getRecordInfo.pPager#"
SIZE="30"
MAXLENGTH="30"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD>&nbsp;</TD>
<TD>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;DETAILS</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Notes:</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabField" COLSPAN=2>
<CFOUTPUT>
<TEXTAREA COLS=40 ROWS=5 NAME="dNotes"
WRAP="soft">#getRecordInfo.dNotes#</TEXTAREA>
</CFOUTPUT>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</CFFORM>
</TABLE>

</CF_srWinTab>

</CFIF>

```

MYBOOKMARKS.CFM

```

<CFIF NOT IsDefined( "URL.user_id" )>
<CFSET URL.user_id = #SESSION.user_id#>
</CFIF>

<CFPARAM NAME="URL.action" DEFAULT="View">
<CFSET mainFile = "main.cfm?a=myAccount&body=myBookmarks.cfm">
<CFSET mainTable = "tblMyBookmarks">
<CFSET idField = "myBookmark_id">

```

```

<CFSET addTitle = "CREATE A NEW BOOKMARK">
<CFSET editTitle = "EDIT BOOKMARK">
<CFSET useSource = "#VARIABLES.gvDATASOURCE_USER#">

<CFIF Mid( SESSION.access,VARIABLES.gvLEVEL_ADMIN,1 ) EQ 'X'>

<CFQUERY NAME="getSummary"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      *
FROM    tblSummary
WHERE shortName = 'myBookmarks'

</CFQUERY>

</CFIF>

<CFIF URL.action EQ "View">

<CF_srReportEngine
ACTION="run"
NAME="listMyBookmarks"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="New#Iif( Mid( SESSION.access,VARIABLES.gvLEVEL_ADMIN,1
) EQ 'X' AND getSummary.RecordCount EQ 0,De( ',Add Summary' ),De(
',Edit Summary' ) )#"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=add#Iif( Mid(
SESSION.ACCESS,VARIABLES.gvLEVEL_ADMIN,1 ) EQ 'X' AND
getSummary.RecordCount EQ 0,De(
',main.cfm?a=tools&body=adminListSummary.cfm&action=add' ),De(
',main.cfm?a=tools&body=adminListSummary.cfm&action=edit&summary_id=#ge
tSummary.summary_id#' ) )#"
ADDWHERE="user_ID=#URL.user_ID#"
>

<CFELSEIF URL.action EQ "Add">

<CFSET title = "#VARIABLES.addTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=saveNew">
<CFSET formName="formAdd">

<CFELSEIF URL.action EQ "Edit">

<CFSET title = "#VARIABLES.editTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=save">
<CFSET formName="formEdit">

<CFELSEIF URL.action EQ "saveNew">

<CFINSERT DATASOURCE="#VARIABLES.useSource#"
TABLENAME="#VARIABLES.mainTable#">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "save">

```

```

<CFUPDATE DATASOURCE="#VARIABLES.useSource#"
TABLENAME="#VARIABLES.mainTable#"
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "delete">

<CFQUERY NAME="deleteInfo" DATASOURCE="#VARIABLES.useSource#">

DELETE
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>
<CFLOCATION URL="#VARIABLES.mainFile#">

</CFIF>

<CFIF URL.action NEQ "View">

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

SELECT      *
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = <CFIF URL.action EQ "Edit">#Evaluate(
'URL.' & VARIABLES.idField )#<CFELSE>-1</CFIF>

</CFQUERY>

<!-- display appropriate header -->
<CF_srWinTab
TABS="#VARIABLES.title#"
SYSTEM_TOOLBAR="Save"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit()"
TOOLBAR="My Bookmarks"
TOOLBARLINKS="main.cfm?a=myAccount&body=myBookmarks.cfm&user_id=#URL.us
er_id#"
>

<TABLE BORDER=0 WIDTH=100%>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<CFOUTPUT>
<INPUT TYPE="hidden" NAME="#VARIABLES.idField#" VALUE="#Evaluate(
'getRecordInfo.' & VARIABLES.idField )#">
<INPUT TYPE="hidden" NAME="user_id" VALUE="#URL.user_id#">
</CFOUTPUT>
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="myBookmarks">
</SPAN>
<P>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel">&nbsp;Label:</TD>
<TD CLASS="winTabField">
</CFINPUT

```

```

NAME="bmLabel"
TYPE="text"
VALUE="#getRecordInfo.bmLabel#"
SIZE="30"
MAXLENGTH="20"
REQUIRED="yes"
MESSAGE="A Bookmark Label is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Web Address:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="bmURL"
TYPE="text"
VALUE="#getRecordInfo.bmURL#"
SIZE="50"
MAXLENGTH="75"
REQUIRED="yes"
MESSAGE="A Bookmark Web Address is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Message:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="bmMessage"
TYPE="text"
VALUE="#getRecordInfo.bmMessage#"
SIZE="50"
MAXLENGTH="50"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Open link where?</TD>
<TD CLASS="winTabField">
<CFIF #getRecordInfo.newWindow# NEQ "">
<CFSET temp = #getRecordInfo.newWindow#>
<CFELSE>
<CFSET temp = 1>
</CFIF>
<CFINPUT TYPE="Radio" NAME="newWindow" VALUE="1" CHECKED="#Iif(
VARIABLES.temp EQ '1',De( 'true' ),De( 'false' ) )#">New Window
<CFINPUT TYPE="Radio" NAME="newWindow" VALUE="2" CHECKED="#Iif(
VARIABLES.temp EQ '2',De( 'true' ),De( 'false' ) )#">Current Window
<CFINPUT TYPE="Radio" NAME="newWindow" VALUE="3" CHECKED="#Iif(
VARIABLES.temp EQ '3',De( 'true' ),De( 'false' ) )#">Within Readiness
Explorer
</TD>
</TR>
</TABLE>
</TD>
</TR>

```

```

</CFFORM>
</TABLE>

</CF_srWinTab>

</CFIF>

```

MYCALENDAR.CFM

```

<CFPARAM NAME="URL.cYear"      DEFAULT="#DateFormat( Now(), 'yyyy' )#">
<CFPARAM NAME="URL.cMonth"    DEFAULT="#DateFormat( Now(), 'm' )#">
<CFPARAM NAME="URL.cDay"      DEFAULT="#DateFormat( Now(), 'd' )#">

<CFIF ( URL.cMonth EQ DatePart( 'm',Now() ) AND URL.cYear EQ DatePart(
'yyyy',Now() ) AND URL.cDay EQ 0 )>

<CFSET URL.cDay = #DateFormat( Now(), 'd' )#>

</CFIF>

<CFSET mainFile = "main.cfm?a=myAccount&body=myCalendar.cfm">

<!--Find the start day the month-->
<CFSET start_day = DayOfWeek( CreateDate(
"#URL.cYear#", "#URL.cMonth#", "1" ) )>

<!--Find the total no of days in the month-->
<CFSET total_days_in_month = DaysInMonth( CreateDate(
"#URL.cYear#", "#URL.cMonth#", "1" ) )>

<!--Find the end day of the month-->
<CFSET end_day = DayOfWeek( CreateDate(
"#URL.cYear#", "#URL.cMonth#", "#VARIABLES.total_days_in_month#" ) )>

<CF_srWinTab
TABS="My Calendar"
SYSTEM_TOOLBAR="Previous Month,Next Month,Today,Add Event"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&cDay=0&cMonth=#Iif(
URL.cMonth EQ 1,12,URL.cMonth-1 )#&cYear=#Iif( URL.cMonth EQ
1,URL.cYear-1,URL.cYear )#, #VARIABLES.mainFile#&cDay=0&cMonth=#Iif(
URL.cMonth EQ 12,1,URL.cMonth+1 )#&cYear=#Iif( URL.cMonth EQ
12,URL.cYear+1,URL.cYear )#, #VARIABLES.mainFile#,temp.cfm"
>

<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD>

<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="myCalander">
</SPAN>
<P>

<TABLE BORDER=0 WIDTH=100%>
<TR>

```

```

<TD CLASS="winTabField" ALIGN="center">

<TABLE CELLSPACING=0 CELLPADDING=0 WIDTH=600 BORDER=0>
<TR>
<TD ALIGN=LEFT BGCOLOR=#FFFFFF>
<TABLE CELLPADDING=2 WIDTH="100%" BORDER=0>
<TR>
<CFOUTPUT>
<TD CLASS="reportTitle" ALIGN="center" WIDTH="100%">#DateFormat(
CreateDate( URL.cYear,URL.cMonth,1 ),"mmmm yyyy" )#</TD>
</CFOUTPUT>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>

<TABLE CELLSPACING=0 CELLPADDING=0 WIDTH=600 BORDER=0>
<TR>
<TD>
<TABLE CELLSPACING=2 CELLPADDING=0 WIDTH="100%" BGCOLOR=#000000
BORDER=0>
<TR ALIGN=MIDDLE BGCOLOR=#CCCCFF>
<TD CLASS="reportTitle" WIDTH=100>Sunday</TD>
<TD CLASS="reportTitle" WIDTH=100>Monday</TD>
<TD CLASS="reportTitle" WIDTH=100>Tuesday</TD>
<TD CLASS="reportTitle" WIDTH=100>Wednesday</TD>
<TD CLASS="reportTitle" WIDTH=100>Thursday</TD>
<TD CLASS="reportTitle" WIDTH=100>Friday</TD>
<TD CLASS="reportTitle" WIDTH=100>Saturday</TD>
</TR>
<CFSET week_day = start_day>
<CFSET display_day = 1>
<CFLOOP CONDITION="#VARIABLES.display_day# LTE
#VARIABLES.total_days_in_month#">
<TR>
<CFLOOP CONDITION="#VARIABLES.week_day# LTE 7">

<CFLOOP CONDITION="#VARIABLES.start_day# NEQ 1">

<TD VALIGN=TOP WIDTH=100 BGCOLOR=#DDDDDD HEIGHT=75>
<TABLE CELLSPACING=0 CELLPADDING=2 WIDTH="100%" BORDER=0>
<TR>
<TD><FONT FACE=ARIAL,VERDANA,HELVETICA COLOR=#000000>&nbsp;</FONT></TD>
</TR>
<TR>
<TD CLASS="reportColumn" ALIGN="right">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="reportColumn" ALIGN="right">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="reportColumn" ALIGN="right">&nbsp;</TD>
</TR>
</TABLE>
</TD>

```

```

<CFSET VARIABLES.start_day = VARIABLES.start_day - 1>

</CFLOOP>
<CFIF VARIABLES.display_day LTE VARIABLES.total_days_in_month>

<CFOUTPUT>
<TD VALIGN=TOP WIDTH=100 BGCOLOR="#Iif( display_day EQ URL.cDay,De(
'FFFFCC' ),De( 'FFFFFF' ) )#" HEIGHT=75>
</CFOUTPUT>
<TABLE CELLSPACING=0 CELLPADDING=2 WIDTH="100%" BORDER=0>
<TR>
<CFOUTPUT>
<TD CLASS="reportTitle"><A HREF="">#VARIABLES.display_day#</A></TD>
</CFOUTPUT>
</TR>
<TR>
<TD CLASS="reportColumn" ALIGN="right">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="reportColumn" ALIGN="right">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="reportColumn" ALIGN="right">&nbsp;</TD>
</TR>
</TABLE>
</TD>

<CFELSE>

<CFLOOP CONDITION="#VARIABLES.end_day# NEQ 7">

<TD VALIGN=TOP WIDTH=100 BGCOLOR=#DDDDDD HEIGHT=75>
<TABLE CELLSPACING=0 CELLPADDING=2 WIDTH="100%" BORDER=0>
<TR>
<TD><FONT FACE=ARIAL,VERDANA,HELVETICA COLOR=#000000>&nbsp;</FONT></TD>
</TR>
<TR>
<TD CLASS="reportColumn" ALIGN="right">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="reportColumn" ALIGN="right">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="reportColumn" ALIGN="right">&nbsp;</TD>
</TR>
</TABLE>
</TD>

<CFSET VARIABLES.end_day = VARIABLES.end_day + 1>

</CFLOOP>

</CFIF>
<CFSET week_day = week_day + 1>
<CFSET display_day = display_day + 1>
</CFLOOP>
</TR>

```

```

<CFSET week_day = 1>
</CFLOOP>
</TABLE>
</TD>
</TR>
</TABLE>

</TD>
</TR>
</TABLE>

</TD>
</TR>
</TABLE>

</CF_srWinTab>

```

MYCOLORS.CFM

```

<CFIF NOT ISDEFINED( "URL.user_id" )>
<CFSET URL.USER_ID = #SESSION.USER_ID#>
</CFIF>

<CFPARAM NAME="URL.action" DEFAULT="View">
<CFPARAM NAME="URL.status" DEFAULT="preview">
<CFSET MAINFILE = "main.cfm?a=myAccount&body=myColors.cfm">
<CFSET TITLE="MY COLORS">
<CFSET USESOURCE = "#VARIABLES.gvDATASOURCE_USER#">

<SCRIPT LANGUAGE="javascript">

function onSubmit( actionType )
{
<CFOUTPUT>
if ( actionType == "saveas" )
{
var styleName = prompt( "Please provide a Color Style name:", "" );
if ( styleName != null && styleName != "" )
{
document.viewColors.action = "#VARIABLES.mainFile#&action=" +
actionType + "&styleName=" + styleName;
document.viewColors.submit();
}
}
else
{
document.viewColors.action = "#VARIABLES.mainFile#&action=" +
actionType;
document.viewColors.submit();
}
}
</CFOUTPUT>
}

</SCRIPT>

```

```

<CFIF URL.ACTION EQ "restore">

<CFIF ISDEFINED( "SESSION.currentColors" )>
<CFSET SESSION.COLORS = #STRUCTCOPY( SESSION.CURRENTCOLORS )#>
<CFSET STRUCTDELETE(SESSION, "currentColors")>
</CFIF>
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.ACTION EQ "preview">

<!-- protect current colors --->
<CFSET SESSION.CURRENTCOLORS = #STRUCTCOPY( SESSION.COLORS )#>

<CFQUERY NAME="getColorElements"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT ceValueName,ceCSSName,ceID
FROM tblColorElements

</CFQUERY>

<!-- declare temporary color structure --->
<CFSET TEMPCOLORS = STRUCTNEW()>

<!-- populate structure with color data --->
<CFLOOP QUERY="getColorElements">
<CFSET TEMP = #STRUCTINSERT( VARIABLES.TEMPCOLORS,CECSSNAME,EVALUATE (
"FORM." & ceID ) )#>
</CFLOOP>

<CFSET SESSION.COLORS = #STRUCTCOPY( VARIABLES.TEMPCOLORS )#>
<CFLOCATION URL="#VARIABLES.mainFile#&status=restore">

<CFELSEIF URL.ACTION EQ "selectNew">

<CFSET STRUCTDELETE(SESSION, "colors")>
<CFSET SESSION.DESKTOP = "#FORM.chooseStyle#">
<CF_SETUSERINFORMATION USER="#SESSION.loginID#" FIELDS="desktop"
VALUES="#FORM.chooseStyle#">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.ACTION EQ "saveas">

<CFQUERY NAME="getColorElements"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT ceValueName,ceCSSName,ceID
FROM tblColorElements

</CFQUERY>

<!-- declare temporary color structure --->
<CFSET TEMPCOLORS = STRUCTNEW()>

<!-- populate structure with color data --->

```

```

<CFLOOP QUERY="getColorElements">
<CFSET TEMP = #STRUCTINSERT( VARIABLES.TEMPCOLORS,CECSSNAME,EVALUATE(
"FORM." & ceID ) )#>
</CFLOOP>

<!-- open desktop.cfg file and convert back to CFM --->
<CFLOCK TIMEOUT="60">
<CFFILE
ACTION="read"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#desktop.cfg"
VARIABLE="DataPacket"
>
</CFLOCK>
<CFWDDX
INPUT="#DataPacket#"
OUTPUT="myColors"
ACTION="WDDX2CFML"
>

<CFSET TEMP = ARRAYAPPEND( MYCOLORS,ARRAYNEW(1) )>
<CFSET MYCOLORS[#ARRAYLEN(MYCOLORS)#][1] = "#URL.styleName#">
<CFSET MYCOLORS[#ARRAYLEN(MYCOLORS)#][2] = #TEMPCOLORS#>

<CFWDDX
INPUT="#myColors#"
OUTPUT="NewPacket"
ACTION="CFML2WDDX"
>
<CFLOCK TIMEOUT="60">
<CFFILE
ACTION="write"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#desktop.cfg"
OUTPUT="#NewPacket#"
>
</CFLOCK>

<CFSET SESSION.COLORS = #STRUCTCOPY( VARIABLES.TEMPCOLORS )#>
<CFSET SESSION.DESKTOP = "#URL.styleName#">
<CFSET STRUCTDELETE(SESSION, "currentColors")>

<CF_SETUSERINFORMATION USER="#SESSION.loginID#" FIELDS="desktop"
VALUES="#URL.styleName#">

<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.ACTION EQ "save">

<CFQUERY NAME="getColorElements"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT ceValueName,ceCSSName,ceID
FROM tblColorElements

</CFQUERY>

<!-- declare temporary color structure --->
<CFSET TEMPCOLORS = STRUCTNEW()>

```

```

<!--- populate structure with color data --->
<CFLOOP QUERY="getColorElements">
<CFSET TEMP = #STRUCTINSERT( VARIABLES.TEMPCOLORS,CECSSNAME,EVALUATE(
"FORM." & ceID) )#>
</CFLOOP>

<!--- open desktop.cfg file and convert back to CFM --->
<CFLOCK TIMEOUT="60">
<CFFILE
ACTION="read"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#desktop.cfg"
VARIABLE="DataPacket"
>
</CFLOCK>
<CFWDDX
INPUT="#DataPacket#"
OUTPUT="myColors"
ACTION="WDDX2CFML"
>

<CFLOOP INDEX="counter" FROM="1" TO="#ArrayLen( myColors )#">

<CFIF MYCOLORS[#COUNTER#][1] EQ "#FORM.chooseStyle#">

<CFSET MYCOLORS[#COUNTER#][2] = #STRUCTCOPY( VARIABLES.TEMPCOLORS )#>
<CFBREAK>

</CFIF>

</CFLOOP>

<CFWDDX
INPUT="#myColors#"
OUTPUT="NewPacket"
ACTION="CFML2WDDX"
>
<CFLOCK TIMEOUT="60">
<CFFILE
ACTION="write"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#desktop.cfg"
OUTPUT="#NewPacket#"
>
</CFLOCK>

<CFSET SESSION.COLORS = #STRUCTCOPY( VARIABLES.TEMPCOLORS )#>
<CFSET STRUCTDELETE(SESSION, "currentColors")>
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.ACTION EQ "delete">

<CFIF #FORM.CHOOSESTYLE# EQ "Default">
<CFLOCATION URL="#VARIABLES.mainFile#">
</CFIF>

<!--- open desktop.cfg file and convert back to CFM --->
<CFLOCK TIMEOUT="60">

```

```

<CFFILE
ACTION="read"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#desktop.cfg"
VARIABLE="DataPacket"
>
</CFLOCK>
<CFWDDX
INPUT="#DataPacket#"
OUTPUT="myColors"
ACTION="WDDX2CFML"
>

<CFLOOP INDEX="counter" FROM="1" TO="#ArrayLen( myColors )#">

<CFIF MYCOLORS[#COUNTER#][1] EQ "#FORM.chooseStyle#">

<CFSET TEMP = #ARRAYDELETEAT( MYCOLORS,COUNTER )#>
<CFBREAK>

</CFIF>

</CFLOOP>

<CFWDDX
INPUT="#myColors#"
OUTPUT="NewPacket"
ACTION="CFML2WDDX"
>
<CFLOCK TIMEOUT="60">
<CFFILE
ACTION="write"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#desktop.cfg"
OUTPUT="#NewPacket#"
>
</CFLOCK>

<CFSET STRUCTDELETE(SESSION, "colors")>
<CF_SETUSERINFORMATION USER="#SESSION.loginID#" FIELDS="desktop"
VALUES="default">
<CFSET SESSION.DESKTOP = "default">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.ACTION EQ "View">

<CFQUERY NAME="getColorElements"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT cgTitle,ceLabel,ceID,ceValueName,ceCSSName,ceDefaultValue
FROM tblColorGroups INNER JOIN (tblColorElements INNER JOIN
tblLinkColorToGroup ON tblColorElements.colorElements_ID =
tblLinkColorToGroup.ColorID) ON tblColorGroups.colorGroup_ID =
tblLinkColorToGroup.ColorGroupID
ORDER BY cgTitle,ceLabel

</CFQUERY>

<CFQUERY NAME="getColors" DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

```

```

SELECT *
FROM tblColors
ORDER BY color

</CFQUERY>

<CFSET FORMACTION = "#VARIABLES.mainFile#">
<CFSET FORMNAME = "viewColors">

<!-- open desktop.cfg file and convert back to CFM --->
<CFLOCK TIMEOUT="60">
<CFFILE
ACTION="read"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#desktop.cfg"
VARIABLE="DataPacket"
>
</CFLOCK>
<CFWDDX
INPUT="#DataPacket#"
OUTPUT="myColors"
ACTION="WDDX2CFML"
>

<STYLE TYPE="text/css">
SELECT.list
{
background-color: fffffff;
font: 8pt Tahoma, Verdana, Arial;
}

</STYLE>

<SCRIPT LANGUAGE="JAVASCRIPT">

var lastShown = null;

var elementWithFocus;

function updateElement( elementID )
{
elementWithFocus = elementID;
}

function updateColors( newColor )
{
if ( elementWithFocus != null )
{
tempColor = eval( elementWithFocus );
tempColor.style.backgroundColor = newColor;
tempValue = eval( "document.viewColors." + elementWithFocus );
tempValue.value = newColor.substr(1);
}
}

function showSelection( element )
{

```



```

<OPTION VALUE="#VARIABLES.myColors[counter][1]#" #IIF(
VARIABLES.MYCOLORS[COUNTER][1] EQ SESSION.DESKTOP,DE( 'SELECTED' ),DE(
' ' ) )#>#VARIABLES.myColors[counter][1]#
</CFOUTPUT>
</CFLOOP>
</SELECT>
</TD>
</TR>
<TR>
<TD ALIGN="center" WIDTH=40%>&nbsp;&nbsp;&nbsp;</TD>
<TD ALIGN="center">&nbsp;&nbsp;&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel" ALIGN="center" WIDTH=40%>&nbsp;&nbsp;&nbsp;COLOR
OBJECTS</TD>
<TD CLASS="winTabLabel" ALIGN="center">&nbsp;&nbsp;&nbsp;PROPERTIES</TD>
</TR>

<TR>
<TD CLASS="winTabField" WIDTH=40% ALIGN="center" VALIGN="top">
<SELECT NAME="ColorGroups" SIZE="10" ONCHANGE="showSelection(
this.value )">
</SELECT>
</TD>
<TD CLASS="winTabField" VALIGN="top" ALIGN="left">
<CENTER>
<BR>
<B>Color Picker:</B>&nbsp;&nbsp;&nbsp;
<SELECT CLASS="list" NAME="colorPicker" SIZE="1"
ONCHANGE="updateColors(this.value)">
<CFOUTPUT QUERY="getColors">
<OPTION STYLE="background: #color_code#" VALUE="#color_code#">#color#
</CFOUTPUT>
</SELECT>
<HR CLASS="standard" SIZE=1 WIDTH=75%>
</CENTER>

<CFOUTPUT QUERY="getColorElements" GROUP="cgTitle">

<SCRIPT LANGUAGE="javascript">

with (document.#VARIABLES.formName#)
{
NewOpt = new Option;
NewOpt.value = "#cgTitle#";
NewOpt.text = "#cgTitle#";

ColorGroups.options[ColorGroups.options.length] = NewOpt;
}

</SCRIPT>
<DIV ID="#cgTitle#" STYLE="display: none;">
<TABLE BORDER=0 WIDTH=100%> <!-- ID="#cgTitle#" STYLE="visibility:
hidden;"-->
<CFOUTPUT>

<TR>

```

```

<TD CLASS="winTabField" WIDTH=5% ALIGN="center"
VALIGN="middle">&nbsp;&nbsp;&nbsp;<CFINPUT TYPE="radio" NAME="elementName"
VALUE="#ceID#" ONCLICK="updateElement( this.value );"></TD>
<TD CLASS="winTabField" WIDTH=75%>&nbsp;&nbsp;&nbsp;#ceLabel#</TD>
<TD ID="#ceID#" WIDTH=20% STYLE="background-color: #Iif(
StructKeyExists( SESSION.colors,ceCSSName ),De( StructFind(
SESSION.colors,ceCSSName ) ),De( ceDefaultValue ) )#"></TD>
<INPUT TYPE="hidden" NAME="#ceID#" VALUE="#Iif( StructKeyExists(
SESSION.colors,ceCSSName ),De( StructFind( SESSION.colors,ceCSSName )
),De( ceDefaultValue ) )#">
</TR>

</CFOUTPUT>
</TABLE>
</DIV>
</CFOUTPUT>

</TD>
</TR>
</TABLE>
</TD>
</TR>
</CFFORM>
</TABLE>

</CF_srWinTab>

</CFIF>

```

MYEMAIL.CFM

```

<CFIF NOT IsDefined( "URL.user_id" )>
<CFSET URL.user_id = #SESSION.user_id#>
</CFIF>

<CFPARAM NAME="URL.action" DEFAULT="List">
<CFSET mainFile = "main.cfm?a=myAccount&body=myEmail.cfm">
<CFIF IsDefined( "URL.myEmailTrash_id" )>
<CFSET VARIABLES.idField = "myEmailTrash_id">
<CFSET VARIABLES.mainTable = "tblMyEmailTrash">
<CFELSE>
<CFSET mainTable = "tblMyEmail">
<CFSET idField = "myEmail_id">
</CFIF>
<CFSET useSource = "#VARIABLES.gvDATASOURCE_USER#">

<CFIF Mid( SESSION.access,VARIABLES.gvLEVEL_ADMIN,1 ) EQ 'X'>

<CFQUERY NAME="getSummary"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      *
FROM  tblSummary
WHERE  shortName = 'myEmail'

```

```

</CFQUERY>

</CFIF>

<CFIF URL.action EQ "List">

<CF_srReportEngine
ACTION="run"
NAME="listMyEmail"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="Get Mail,Compose Mail,Sent Mail,Deleted Mail"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=getMail,#VARIABLES.mai
nFile#&action=composeMail,#VARIABLES.mainFile#&action=sentMail,#VARIABLES.mai
nFile#&action=deletedMail"
ADDWHERE="user_ID=#URL.user_ID#"
>

<CFELSEIF URL.action EQ ,"deletedMail">

<CF_srReportEngine
ACTION="run"
NAME="deletedMail"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="inBox,Compose Mail,Sent Mail"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#,#VARIABLES.mainFile#&action=c
omposeMail,#VARIABLES.mainFile#&action=sentMail"
ADDWHERE="user_ID=#URL.user_ID#"
>

<CFELSEIF URL.action EQ "getMail">

<CFQUERY NAME="getEmailSetup"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      *
FROM      tblMyEmailSetup
WHERE user_id = #URL.user_id#
AND myEmailSetup_ID = 1

</CFQUERY>

<CFPOP
NAME="getMail"
ACTION="GETALL"
SERVER="#getEmailSetup.popAddress#"
PORT="#getEmailSetup.popAddressPort#"
USERNAME="#getEmailSetup.accountName#"
PASSWORD="#getEmailSetup.accountPassword#"
TIMEOUT="#getEmailSetup.timeOut#"
ATTACHMENTPATH="#GetDirectoryFromPath( GetTemplatePath() )#"
>

<CFLOOP QUERY="getMail">

```

```

<CFQUERY NAME="insertMail" DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

INSERT INTO tblMyEmail
(myEmailSetup_id,user_id,emailTo,emailFrom,emailSubject,emailReplyTo,em
ailNumber,emailDate,emailCC,emailAttachments,emailAttachFiles,emailMess
age)
VALUES
(1,#URL.user_id#,'#to#','#from#','#subject#','#replyTo#','#messageNumber
#','#date#','#cc#','#attachments#','#attachmentfiles#','#body#' )

</CFQUERY>

</CFLOOP>
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "delete">

<CFTRANSACTION>
<CFQUERY NAME="moveToTrash" DATASOURCE="#VARIABLES.useSource#">

INSERT INTO tblMyEmailTrash
(user_id,emailTo,emailFrom,emailSubject,emailReplyTo,emailNumber,emailD
ate,emailCC,emailAttachments,emailAttachFiles,emailMessage)
SELECT
user_id,emailTo,emailFrom,emailSubject,emailReplyTo,emailNumber,emailDa
te,emailCC,emailAttachments,emailAttachFiles,emailMessage
FROM tblMyEmail
WHERE #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>

<CFQUERY NAME="deleteInfo" DATASOURCE="#VARIABLES.useSource#">

DELETE
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>
</CFTRANSACTION>
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "remove">

<CFQUERY NAME="deleteInfo" DATASOURCE="#VARIABLES.useSource#">

DELETE
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>
<CFLOCATION URL="#VARIABLES.mainFile#&action=deletedMail">

<CFELSEIF URL.action EQ "View">

<CFSET title = "View Message">

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

```

```

SELECT      *
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>

<!-- display appropriate header --->
<CF_srWinTab
TABS="#VARIABLES.title#"
SYSTEM_TOOLBAR="inbox,#Iif( IsDefined( 'URL.myEmailTrash_id' ),De(
'Deleted Mail,' ),De( 'Reply,Forward,' ) )##Iif( IsDefined(
'URL.myEmailTrash_id' ),De( 'Remove' ),De( 'Delete' ) )#,Compose"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#,#Iif( IsDefined(
'URL.myEmailTrash_id' ),De( VARIABLES.mainFile & '&action=deletedMail,'
),De( VARIABLES.mainFile & '&action=reply,' & VARIABLES.mainFile &
'&action=forward,' ) )##VARIABLES.mainFile#&action=#Iif( IsDefined(
'URL.myEmailTrash_id' ),De( 'remove' ),De( 'delete' )
)#,#VARIABLES.mainFile#&action=composeMail"
SYSTEM_ADDID="#VARIABLES.idField##Evaluate( 'URL.' & VARIABLES.idField
)#"
TOOLBAR="Previous,Next"
TOOLBARLINKS="temp.cfm,temp.cfm"
>

<TABLE BORDER=0 WIDTH=100%>

<TR>
<TD>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabField" VALIGN="top" COLSPAN=2>

<CFOUTPUT>
<TABLE BORDER=0 WIDTH=100% CELLSPACING=0 CELLPADDING=2>
<TR VALIGN="top">
<TD CLASS="reportTitle" ALIGN="right">Subject:</TD>
<TD CLASS="winTabField">#HTMLFormat( getRecordInfo.emailSubject
)#</TD>
<TD ROWSPAN="2" ALIGN="right"><IMG
SRC="#VARIABLES.gvIMAGE_DIRECTORY#re_stamp.gif" BORDER=0></TD>
</TR>
<TR VALIGN="top">
<TD CLASS="reportTitle" ALIGN="right">Date:</TD>
<TD CLASS="winTabField">#getRecordInfo.emailDate#</TD>
</TR>
<TR VALIGN="top">
<TD CLASS="reportTitle" ALIGN="right">From:</TD>
<TD CLASS="winTabField" COLSPAN=2>#HTMLFormat(
getRecordInfo.emailFrom )#</TD>
</TR>
<TR VALIGN="top">
<TD CLASS="reportTitle" ALIGN="right">To:</TD>
<TD CLASS="winTabField" COLSPAN=2>#HTMLFormat(
getRecordInfo.emailTo )#</TD>
</TR>
<TR VALIGN="top">

```

```

<TD CLASS="reportTitle" ALIGN="right">&nbsp;&nbsp;&nbsp;</TD>
<TD CLASS="winTabField" COLSPAN=2><BR><CF_srLINKFINDER
DATA="#getRecordInfo.emailMessage#"></TD>
</TR>
</TABLE>
</CFOUTPUT>

</TD>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>

</CF_srWinTab>

</CFIF>

```

MYEMAILSETUP.CFM

```

<CFIF NOT IsDefined( "URL.user_id" )>
<CFSET URL.user_id = #SESSION.user_id#>
</CFIF>

<CFPARAM NAME="URL.action" DEFAULT="View">
<CFSET mainFile = "main.cfm?a=myAccount&body=myEmailsetup.cfm">
<CFSET mainTable = "tblmyEmailSetup">
<CFSET idField = "myEmailSetup_id">
<CFSET addTitle = "CREATE A NEW EMAIL ACCOUNT">
<CFSET editTitle = "EDIT EMAIL ACCOUNT">
<CFSET useSource = "#VARIABLES.gvDATASOURCE_USER#">

<CFIF URL.action EQ "View">

<CF_srReportEngine
ACTION="run"
NAME="listMyEmailAccounts"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="New"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=add"
ADDWHERE="user_ID=#URL.user_ID#"
>

<CFELSEIF URL.action EQ "Add">

<CFSET title = "#VARIABLES.addTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=saveNew">
<CFSET formName="formAdd">

<CFELSEIF URL.action EQ "Edit">

<CFSET title = "#VARIABLES.editTitle#">
<CFSET formAction = "#VARIABLES.mainFile#&action=save">

```

```

<CFSET formName="formEdit">

<CFELSEIF URL.action EQ "saveNew">

<CFINSERT DATASOURCE="#VARIABLES.useSource#"
TABLENAME="#VARIABLES.mainTable#"
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "save">

<CFUPDATE DATASOURCE="#VARIABLES.useSource#"
TABLENAME="#VARIABLES.mainTable#"
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "delete">

<CFQUERY NAME="deleteInfo" DATASOURCE="#VARIABLES.useSource#">

DELETE
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>
<CFLOCATION URL="#VARIABLES.mainFile#">

</CFIF>

<CFIF URL.action NEQ "View">

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

SELECT      *
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = <CFIF URL.action EQ "Edit">#Evaluate(
'URL.' & VARIABLES.idField )#<CFELSE>-1</CFIF>

</CFQUERY>

<!-- display appropriate header --->
<CF_srWinTab
TABS="#VARIABLES.title#"
SYSTEM_TOOLBAR="Save,List My Email Accounts"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit(),
main.cfm?a=myAccount&body=myEmailSetup.cfm&user_id=#URL.user_id#"
>

<TABLE BORDER=0 WIDTH=100%>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<CFOUTPUT>
<INPUT TYPE="hidden" NAME="#VARIABLES.idField#" VALUE="#Evaluate(
'getRecordInfo.' & VARIABLES.idField )#">
<INPUT TYPE="hidden" NAME="user_id" VALUE="#URL.user_id#">
</CFOUTPUT>
<TR>
<TD>
<SPAN CLASS="winTab">

```

```

<CF_srHELPSUMMARY shortName="myEmailSetup">
</SPAN>
<P>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;ACCOUNT INFORMATION</TD>
<TD CLASS="winTabField">&nbsp;&nbsp;&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Title of this account:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="emailTitle"
TYPE="text"
VALUE="#getRecordInfo.emailTitle#"
SIZE="30"
MAXLENGTH="30"
REQUIRED="yes"
MESSAGE="An Email Account Title is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Your Name:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="userName"
TYPE="text"
VALUE="#getRecordInfo.userName#"
SIZE="30"
MAXLENGTH="50"
REQUIRED="yes"
MESSAGE="Your Name is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Your Organization:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="userOrganization"
TYPE="text"
VALUE="#getRecordInfo.userOrganization#"
SIZE="30"
MAXLENGTH="50"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Your Email Address for this account:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="userEmailAddress"
TYPE="text"
VALUE="#getRecordInfo.userEmailAddress#"
SIZE="30"

```

```

MAXLENGTH="75"
REQUIRED="yes"
MESSAGE="An Email Address is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Reply Email Address for this
account:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="userReplyEmailAddress"
TYPE="text"
VALUE="#getRecordInfo.userReplyEmailAddress#"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD>&nbsp;</TD>
<TD>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;EMAIL SERVER INFORMATION</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Incoming (POP3) Server Address:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="popAddress"
TYPE="text"
VALUE="#getRecordInfo.popAddress#"
SIZE="30"
MAXLENGTH="50"
REQUIRED="yes"
MESSAGE="A POP Server Address is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Incoming (POP3) Server Port Number:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="popAddressPort"
TYPE="text"
VALUE="#Iif( getRecordInfo.popAddressPort EQ '',De( '110' ),De(
getRecordInfo.popAddressPort ) )#"
VALIDATE="integer"
SIZE="10"
MAXLENGTH="4"
REQUIRED="yes"
MESSAGE="A POP Server Port Number is required."
>
</TD>

```

```

</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Outgoing (SMTP) Server Address:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="smtpAddress"
TYPE="text"
VALUE="#getRecordInfo.smtpAddress#"
SIZE="30"
MAXLENGTH="50"
REQUIRED="yes"
MESSAGE="A SMTP Server Address is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Outgoing (SMTP) Server Port Number:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="smtpAddressPort"
TYPE="text"
VALUE="#Iif( getRecordInfo.smtpAddressPort EQ '',De( '25' ),De(
getRecordInfo.smtpAddressPort ) )#"
VALIDATE="integer"
SIZE="10"
MAXLENGTH="4"
REQUIRED="yes"
MESSAGE="A SMTP Server Port Number is required."
>
</TD>
</TR>
<TR>
<TD>&nbsp;&nbsp;&nbsp;</TD>
<TD>&nbsp;&nbsp;&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;SECURITY INFORMATION</TD>
<TD CLASS="winTabField">&nbsp;&nbsp;&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Account Login:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="accountName"
TYPE="text"
VALUE="#getRecordInfo.accountName#"
SIZE="30"
MAXLENGTH="30"
REQUIRED="yes"
MESSAGE="An account name is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Account Password:</TD>
<TD CLASS="winTabField">
<CFINPUT

```

```

NAME="accountPassword"
TYPE="password"
VALUE="#getRecordInfo.accountPassword#"
SIZE="30"
MAXLENGTH="20"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Remember password?</TD>
<TD CLASS="winTabField">
<CFIF #getRecordInfo.rememberPassword# NEQ "">
<CFSET temp = #getRecordInfo.rememberPassword#>
<CFELSE>
<CFSET temp = false>
</CFIF>
<CFINPUT TYPE="Radio" NAME="rememberPassword" VALUE=false
CHECKED="#Iif( NOT VARIABLES.temp,De( 'true' ),De( 'false' ) )#">No
Thanks
<CFINPUT TYPE="Radio" NAME="rememberPassword" VALUE=true CHECKED="#Iif(
VARIABLES.temp,De( 'true' ),De( 'false' ) )#">Yes
</TD>
</TR>
<TR>
<TD>&nbsp;</TD>
<TD>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;OPTIONS</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Delete email on server after
receiving?</TD>
<TD CLASS="winTabField">
<CFIF #getRecordInfo.deleteOnServer# NEQ "">
<CFSET temp = #getRecordInfo.deleteOnServer#>
<CFELSE>
<CFSET temp = false>
</CFIF>
<CFINPUT TYPE="Radio" NAME="deleteOnServer" VALUE=false CHECKED="#Iif(
NOT VARIABLES.temp,De( 'true' ),De( 'false' ) )#">No Thanks
<CFINPUT TYPE="Radio" NAME="deleteOnServer" VALUE=true CHECKED="#Iif(
VARIABLES.temp,De( 'true' ),De( 'false' ) )#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Seconds to wait before timing out:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="timeOut"
TYPE="text"
VALIDATE="integer"
RANGE="0,300"
VALUE="#getRecordInfo.timeOut#"
SIZE="10"

```

```

MAXLENGTH="3"
REQUIRED="no"
>
Seconds
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Set this account as the default?</TD>
<TD CLASS="winTabField">
<CFIF #getRecordInfo.setAsDefault# NEQ "">
<CFSET temp = #getRecordInfo.setAsDefault#>
<CFELSE>
<CFSET temp = false>
</CFIF>
<CFINPUT TYPE="Radio" NAME="setAsDefault" VALUE=false CHECKED="#Iif(
NOT VARIABLES.temp,De( 'true' ),De( 'false' ) )#">No Thanks
<CFINPUT TYPE="Radio" NAME="setAsDefault" VALUE=true CHECKED="#Iif(
VARIABLES.temp,De( 'true' ),De( 'false' ) )#">Yes
</TD>
</TR>
</TABLE>
</TD>
</TR>
</CFFORM>
</TABLE>

</CF_srWinTab>

</CFIF>

```

MYINFORMATION.CFM

```

<CFIF NOT IsDefined( "URL.user_id" )>
<CFSET URL.user_id = #SESSION.user_id#>
</CFIF>

<CFPARAM NAME="URL.action" DEFAULT="edit">
<CFSET mainFile =
"main.cfm?a=myAccount&body=myInformation.cfm&user_id=#URL.user_id#">
<CFSET mainTable = "tblUserAccounts">
<CFSET idField = "user_id">

<CFIF URL.action EQ "informationSave">

<!-- save new password if required -->
<SCRIPT LANGUAGE="javascript" SRC="cookies.js"></SCRIPT>
<SCRIPT LANGUAGE="javascript">
var xR = readCookie( 'cookieRemember' );
if ( xR != null )
<CFOUTPUT>
saveCookie( 'cookiePassword', '#FORM.user_password#', 30);
</CFOUTPUT>
</SCRIPT>

```

```

<CFUPDATE DATASOURCE="#VARIABLES.gvDATASOURCE_USER#"
TABLENAME="#VARIABLES.mainTable#"
FORMFIELDS="lname, fname, rank, ssn, phone, email, command, user_password, home
page">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "edit">

<CFSET title = "my Information">
<CFSET formAction = "#VARIABLES.mainFile#&action=informationSave">
<CFSET formName="formAccount">

<CFQUERY NAME="getRecordInfo"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      *
FROM #VARIABLES.mainTable#
WHERE user_id = #URL.user_id#

</CFQUERY>

<!--- display appropriate header --->
<CF_srWinTab
TABS="myInformation,mySecurity,myColors,myToolbar,myPanel"
TABSELECTED="1"
TABURL="main.cfm?a=myAccount&body=myInformation.cfm&user_id=#URL.user_id#,#main.cfm?a=myAccount&body=mySecurity.cfm&user_id=#URL.user_id#,#main.cfm?a=myAccount&body=myColors.cfm&user_id=#URL.user_id#,#main.cfm?a=myAccount&body=myToolbar.cfm&user_id=#URL.user_id#,#main.cfm?a=myAccount&body=myPanel.cfm&user_id=#URL.user_id#"
SYSTEM_TOOLBAR="Save"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit()"
>

<TABLE BORDER=0 WIDTH=100%>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<CFOUTPUT>
<INPUT TYPE="hidden" NAME="#VARIABLES.idField#" VALUE="#Evaluate(
'getRecordInfo.' & VARIABLES.idField )#">
</CFOUTPUT>
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="myInformation">
</SPAN>
<P>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel">&nbsp;Date you last logged in:</TD>
<TD CLASS="winTabField">
<CFIF getRecordInfo.last_logon NEQ "">
<CFOUTPUT>#DateFormat( getRecordInfo.last_logon, 'dd mmm yyyy' )# at
#TimeFormat( getRecordInfo.last_logon, 'H:mm' )#</CFOUTPUT>
</CFIF>
</TD>
</TR>

```

```

<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Date you last changed your password:</TD>
<TD CLASS="winTabField">
<CFIF getRecordInfo.last_logon NEQ "">
<CFOUTPUT>#DateFormat( getRecordInfo.password_change, 'dd mmm yyyy'
) #</CFOUTPUT>
</CFIF>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Date your account expires:</TD>
<TD CLASS="winTabField">
<CFOUTPUT>#DateFormat( getRecordInfo.expire, 'dd mmm yyyy' ) #</CFOUTPUT>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;User account template:</TD>
<TD CLASS="winTabField">
<CFOUTPUT>#getRecordInfo.template#</CFOUTPUT>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Login ID:</TD>
<TD CLASS="winTabField">
<CFOUTPUT>#getRecordInfo.user_loginid#</CFOUTPUT>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Last Name:</FONT></TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="lname"
TYPE="text"
VALUE="#getRecordInfo.lname#"
SIZE="30"
MAXLENGTH="30"
REQUIRED="yes"
MESSAGE="A Last Name must be entered"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;First Name:</FONT></TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="fname"
TYPE="text"
VALUE="#getRecordInfo.fname#"
SIZE="30"
MAXLENGTH="30"
REQUIRED="yes"
MESSAGE="A First Name must be entered"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;&nbsp;&nbsp;Rank:</FONT></TD>

```

```

<TD CLASS="winTabField">
<CFINPUT
NAME="rank"
TYPE="text"
VALUE="#getRecordInfo.rank#"
SIZE="20"
MAXLENGTH="10"
REQUIRED="yes"
MESSAGE="Rank must be entered"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;SSN (Last four):</FONT></TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="ssn"
TYPE="text"
VALUE="#getRecordInfo.ssn#"
SIZE="20"
MAXLENGTH="4"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Phone:</FONT></TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="phone"
TYPE="text"
VALUE="#getRecordInfo.phone#"
SIZE="20"
MAXLENGTH="15"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Email Address:</FONT></TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="email"
TYPE="text"
VALUE="#getRecordInfo.email#"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Homepage:</FONT></TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="homepage"
TYPE="text"

```

```

VALUE="#getRecordInfo.homepage#"
SIZE="30"
MAXLENGTH="75"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Command:</FONT></TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="command"
TYPE="text"
VALUE="#getRecordInfo.command#"
SIZE="30"
MAXLENGTH="75"
REQUIRED="yes"
MESSAGE="A Command name must be entered."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Password:</FONT></TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="user_password"
TYPE="password"
VALUE="#getRecordInfo.user_password#"
SIZE="30"
MAXLENGTH="10"
REQUIRED="yes"
MESSAGE="A Password must be entered."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Re-Enter Password:</FONT></TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="user_password_check"
TYPE="password"
VALUE="#getRecordInfo.user_password#"
SIZE="30"
MAXLENGTH="10"
REQUIRED="yes"
MESSAGE="You must re-enter your password."
>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</CFFORM>
</TABLE>

</CF_srWinTab>

```

</CFIF>

MYPANEL.CFM

<CFIF NOT IsDefined("URL.user_id")>

<CFSET URL.user_id = #SESSION.user_id#>

</CFIF>

<CFPARAM NAME="URL.action" DEFAULT="View">

<CFSET mainFile = "main.cfm?a=myAccount&body=myPanel.cfm">

<CFSET formAction =

"#VARIABLES.mainFile#&form=true&user_id=#SESSION.user_id#">

<CFSET formName="formPanelComponents">

<CFSET title = "My Panel Components">

<CFIF URL.action EQ "View">

<SCRIPT LANGUAGE="Javascript">

function onSubmit(actionType)

{

<CFOUTPUT>

document.#VARIABLES.formName#.action =

"#VARIABLES.formAction#&actionType=" + actionType;

document.#VARIABLES.formName#.submit();

</CFOUTPUT>

}

</SCRIPT>

<CFIF IsDefined("URL.form")>

<CFIF #URL.form#>

<CFIF #URL.actionType# EQ "add" >

<CFLOOP INDEX="Counter" FROM="1" TO="#ListLen(FORM.Components)#">

<CFQUERY NAME="AddNew" DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

INSERT INTO tblMyPanelComponents (user_id,panelComponent_ID)

VALUES (#URL.user_id#,#ListGetAt(FORM.Components,Counter)#)

</CFQUERY>

</CFLOOP>

<CFELSEIF #URL.actionType# EQ "remove" >

<CFQUERY NAME="Remove" DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

DELETE FROM tblMyPanelComponents

WHERE user_id = #URL.user_id#

AND panelComponent_ID IN (#FORM.UserComponents#)

```

</CFQUERY>

</CFIF>

<CFLOCATION URL="#VARIABLES.mainFile#">

</CFIF>

</CFIF>

<CFQUERY NAME="getUserComponentIDs"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      panelComponent_ID
FROM        tblMyPanelComponents
WHERE       user_id = #URL.user_id#

</CFQUERY>
<CFQUERY NAME="getComponents"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      *
FROM        tblPanelComponents
WHERE       0=0
<CFIF #getUserComponentIDs.RecordCount# NEQ 0>
AND panelComponent_ID NOT IN ( #ValueList(
getUserComponentIDs.panelComponent_ID )# )
</CFIF>
AND pcTitle <> 'Add Components'
ORDER BY   pcTitle

</CFQUERY>
<CFQUERY NAME="getUserComponents"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      *
FROM        tblPanelComponents
WHERE
<CFIF #getUserComponentIDs.RecordCount# NEQ 0>
panelComponent_ID IN ( #ValueList(
getUserComponentIDs.panelComponent_ID )# )
<CFELSE>
0=1
</CFIF>
ORDER BY   pcTitle

</CFQUERY>

<!-- display appropriate header --->
<CF_srWinTab
TABS="myInformation,mySecurity,myColors,myToolbar,myPanel"
TABSELECTED="5"
TABURL="main.cfm?a=myAccount&body=myInformation.cfm&user_id=#URL.user_id#,main.cfm?a=myAccount&body=mySecurity.cfm&user_id=#URL.user_id#,main.cfm?a=myAccount&body=myColors.cfm&user_id=#URL.user_id#,main.cfm?a=myAccount&body=myToolbar.cfm&user_id=#URL.user_id#,main.cfm?a=myAccount&body=myPanel.cfm&user_id=#URL.user_id#"

```

```

SYSTEM_TOOLBAR="#Iif( getComponents.RecordCount NEQ 0,De( 'Add' ),De(
'' ) )#, #Iif( getUserComponents.RecordCount NEQ 0,De( 'Remove' ),De( ''
) )#, #Iif( getUserComponents.RecordCount NEQ 0,De( 'Properties' ),De(
'' ) )#"
SYSTEM_TOOLBARLINKS="#Iif( getComponents.RecordCount NEQ 0,De(
'javascript: onSubmit( 'add' )' ),De( '' ) )#, #Iif(
getUserComponents.RecordCount NEQ 0,De( 'javascript: onSubmit(
'remove' )' ),De( '' ) )#, #Iif( getUserComponents.RecordCount NEQ
0,De( 'javascript: onSubmit( 'properties' )' ),De( '' ) )#"
>

```

```
<TABLE BORDER=0 WIDTH=100%>
```

```
<TR>
```

```
<TD>
```

```
<SPAN CLASS="winTab">
```

```
<CF_srHELPSUMMARY shortName="myPanel"
```

```
Page="#aComponent[URL.aID][cAPPLICATION][cAPPLICATION_NAME]#">
```

```
</SPAN>
```

```
<P>
```

```
<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
```

```
<TABLE BORDER=0 WIDTH=100%>
```

```
<TR>
```

```
<TD CLASS="winTabLabel" WIDTH="50%" ALIGN="center">Available  
Components</TD>
```

```
<TD CLASS="winTabLabel" WIDTH="50%" ALIGN="center">My Components</TD>
```

```
</TR>
```

```
<TR>
```

```
<TD CLASS="winTabField" WIDTH="50%" ALIGN="center">
```

```
<CFSELECT
```

```
NAME="Components"
```

```
QUERY="getComponents"
```

```
SIZE="8"
```

```
VALUE="panelComponent_ID"
```

```
DISPLAY="pcTitle"
```

```
MULTIPLE="YES"
```

```
>
```

```
</CFSELECT>
```

```
</TD>
```

```
<TD CLASS="winTabField" WIDTH="50%" ALIGN="center">
```

```
<CFSELECT
```

```
NAME="UserComponents"
```

```
QUERY="getUserComponents"
```

```
SIZE="8"
```

```
VALUE="panelComponent_ID"
```

```
DISPLAY="pcTitle"
```

```
MULTIPLE="YES"
```

```
>
```

```
</CFSELECT>
```

```
</TD>
```

```
</TR>
```

```
</TABLE>
```

```
</CFFORM>
```

```
</TD>
```

```
</TR>
```

```
</TABLE>
```

```

</CF_srWinTab>

</CFIF>

MYSECURITY.CFM

<CFPARAM NAME="URL.action" DEFAULT="edit">
<CFIF NOT IsDefined( "URL.user_id" )>

<CFSET URL.user_id = #SESSION.user_id#>

</CFIF>

<CFIF IsDefined( "URL.actionCookie" )>

<CFQUERY NAME="getUserInfo" DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      user_loginid,user_password
FROM      tblUserAccounts
WHERE user_loginid = '#SESSION.loginid#'

</CFQUERY>

<CFIF #URL.actionCookie# EQ "save">

<SCRIPT LANGUAGE="javascript" SRC="cookies.js"></SCRIPT>
<SCRIPT LANGUAGE="javascript">
<CFOUTPUT>
<!-- set expiration to 30 days for now, this will be an option for
ADMIN --->
saveCookie( 'cookieLogin','#getUserInfo.user_loginid#,30);
saveCookie( 'cookiePassword','#getUserInfo.user_password#,30);
saveCookie( 'cookieRemember','true',30);
</CFOUTPUT>
</SCRIPT>

<CFELSE>

<!-- delete cookies just in case --->
<SCRIPT LANGUAGE="javascript" SRC="cookies.js"></SCRIPT>
<SCRIPT LANGUAGE="javascript">
deleteCookie( 'cookieLogin' );
deleteCookie( 'cookiePassword' );
deleteCookie( 'cookieRemember' );
</SCRIPT>

</CFIF>

</CFIF>

<CFSET mainFile =
"main.cfm?a=myAccount&body=mySecurity.cfm&user_id=#URL.user_id#">
<CFSET mainTable = "tblMySecurity">
<CFSET idField = "mySecurity_id">

<CFIF URL.action EQ "securitySave">

```

```

<CFIF #FORM.mySecurity_id# NEQ "">
<CFUPDATE DATASOURCE="#VARIABLES.gvDATASOURCE_USER#"
TABLENAME="#VARIABLES.mainTable#">
<CFELSE>
<CFINSERT DATASOURCE="#VARIABLES.gvDATASOURCE_USER#"
TABLENAME="#VARIABLES.mainTable#">
</CFIF>

<CFIF #FORM.rememberLogin#>
<CFLOCATION URL="#VARIABLES.mainFile#&actionCookie=save">
<CFELSE>
<CFLOCATION URL="#VARIABLES.mainFile#&actionCookie=delete">
</CFIF>

<CFELSEIF URL.action EQ "edit">

<CFSET title = "my Security">
<CFSET formAction = "#VARIABLES.mainFile#&action=securitySave">
<CFSET formName="formSecurity">

<CFQUERY NAME="getRecordInfo"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      *
FROM        #VARIABLES.mainTable#
WHERE user_id = #URL.user_id#

</CFQUERY>

<!-- display appropriate header --->
<CF_srWinTab
TABS="myInformation,mySecurity,myColors,myToolbar,myPanel"
TABSELECTED="2"
TABURL="main.cfm?a=myAccount&body=myInformation.cfm&user_id=#URL.user_id#,&body=mySecurity.cfm&user_id=#URL.user_id#,&body=myColors.cfm&user_id=#URL.user_id#,&body=myToolbar.cfm&user_id=#URL.user_id#,&body=myPanel.cfm&user_id=#URL.user_id#"
SYSTEM_TOOLBAR="Save"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit()"
>

<TABLE BORDER=0 WIDTH=100%>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<CFOUTPUT>
<INPUT TYPE="hidden" NAME="#VARIABLES.idField#" VALUE="#Evaluate(
'getRecordInfo.' & VARIABLES.idField )#">
<INPUT TYPE="hidden" NAME="user_id" VALUE="#URL.user_id#">
</CFOUTPUT>
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="mySecurity">
</SPAN>

```

```

<P>

<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel">&nbsp;PASSWORD INFORMATION</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Remember login information?</TD>
<TD CLASS="winTabField">
<CFIF #getRecordInfo.rememberLogin# NEQ "">
<CFSET temp = #getRecordInfo.rememberLogin#>
<CFELSE>
<CFSET temp = false>
</CFIF>
<CFINPUT TYPE="Radio" NAME="rememberLogin" VALUE=false CHECKED="#Iif(
NOT VARIABLES.temp,De( 'true' ),De( 'false' ) )#">No Thanks
<CFINPUT TYPE="Radio" NAME="rememberLogin" VALUE=true CHECKED="#Iif(
VARIABLES.temp,De( 'true' ),De( 'false' ) )#">Yes
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Change password automatically and
email?</TD>
<TD CLASS="winTabField">
<CFIF #getRecordInfo.autoPassword# NEQ "">
<CFSET temp = #getRecordInfo.autoPassword#>
<CFELSE>
<CFSET temp = false>
</CFIF>
<CFINPUT TYPE="Radio" NAME="autoPassword" VALUE=false CHECKED="#Iif(
NOT VARIABLES.temp,De( 'true' ),De( 'false' ) )#">No Thanks
<CFINPUT TYPE="Radio" NAME="autoPassword" VALUE=true CHECKED="#Iif(
VARIABLES.temp,De( 'true' ),De( 'false' ) )#">Yes
</TD>
</TR>
<TR>
<TD>&nbsp;</TD>
<TD>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;VACATION/BUSINESS ABSENCE</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;When will you depart?</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="absentFrom"
TYPE="text"
VALIDATE="date"
VALUE="#DateFormat( getRecordInfo.absentFrom, 'mm/dd/yyyy' )#"
SIZE="10"
MAXLENGTH="10"
REQUIRED="no"
>
&nbsp;

```

```

<CF_srAddPopUpCalendar
FIELD="document.#VARIABLES.formName#.absentFrom">
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;When will you return?</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="absentTo"
TYPE="text"
VALIDATE="date"
VALUE="#DateFormat( getRecordInfo.absentTo,'mm/dd/yyyy' )#"
SIZE="10"
MAXLENGTH="10"
REQUIRED="no"
>
&nbsp;
<CF_srAddPopUpCalendar FIELD="document.#VARIABLES.formName#.absentTo">
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Prevent logins during this period?</TD>
<TD CLASS="winTabField">
<CFIF #getRecordInfo.absentPreventLogon# NEQ "">
<CFSET temp = #getRecordInfo.absentPreventLogon#>
<CFELSE>
<CFSET temp = false>
</CFIF>
<CFINPUT TYPE="Radio" NAME="absentPreventLogon" VALUE=false
CHECKED="#Iif( NOT VARIABLES.temp,De( 'true' ),De( 'false' ) )#">No
Thanks
<CFINPUT TYPE="Radio" NAME="absentPreventLogon" VALUE=true
CHECKED="#Iif( VARIABLES.temp,De( 'true' ),De( 'false' ) )#">Yes
</TD>
</TR>
<TR>
<TD>&nbsp;</TD>
<TD>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;ACCOUNT ACTIVITY</TD>
<TD CLASS="winTabField">&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Email audit trail reports?</TD>
<TD CLASS="winTabField">
<CFIF #getRecordInfo.emailAuditTrail# NEQ "">
<CFSET temp = #getRecordInfo.emailAuditTrail#>
<CFELSE>
<CFSET temp = false>
</CFIF>
<CFINPUT TYPE="Radio" NAME="emailAuditTrail" VALUE=false CHECKED="#Iif(
NOT VARIABLES.temp,De( 'true' ),De( 'false' ) )#">No Thanks
<CFINPUT TYPE="Radio" NAME="emailAuditTrail" VALUE=true CHECKED="#Iif(
VARIABLES.temp,De( 'true' ),De( 'false' ) )#">Yes
</TD>
</TR>

```



```

<CFPARAM NAME="URL.action" DEFAULT="edit">
<CFSET mainFile =
"main.cfm?a=myAccount&body=myToolbar.cfm&user_id=#URL.user_id#">
<CFSET mainTable = "tblMyToolbar">
<CFSET idField = "myToolbar_id">

<CFIF URL.action EQ "toolbarSave">

<CFIF #FORM.myToolbar_id# NEQ "">
<CFUPDATE DATASOURCE="#VARIABLES.gvDATASOURCE_USER#"
TABLENAME="#VARIABLES.mainTable#">
<CFELSE>
<CFINSERT DATASOURCE="#VARIABLES.gvDATASOURCE_USER#"
TABLENAME="#VARIABLES.mainTable#">
</CFIF>
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.action EQ "edit">

<CFSET title = "my Toolbar">
<CFSET formAction = "#VARIABLES.mainFile#&action=toolbarSave">
<CFSET formName="formToolbar">

<CFQUERY NAME="getRecordInfo"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      *
FROM #VARIABLES.mainTable#
WHERE user_id = #URL.user_id#

</CFQUERY>

<!--- display appropriate header --->
<CF_srWinTab
TABS="myInformation,mySecurity,myColors,myToolbar,myPanel"
TABSELECTED="4"
TABURL="main.cfm?a=myAccount&body=myInformation.cfm&user_id=#URL.user_id#,#main.cfm?a=myAccount&body=mySecurity.cfm&user_id=#URL.user_id#,#main.cfm?a=myAccount&body=myColors.cfm&user_id=#URL.user_id#,#main.cfm?a=myAccount&body=myToolbar.cfm&user_id=#URL.user_id#,#main.cfm?a=myAccount&body=myPanel.cfm&user_id=#URL.user_id#"
SYSTEM_TOOLBAR="Save"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit()"
>

<TABLE BORDER=0 WIDTH=100%>

<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="myToolbar">
</SPAN>
<P>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<CFOUTPUT>

```



```

<TR>
<TD CLASS="winTabLabel">&nbsp;Message:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="tbMessage#counter#"
TYPE="text"
VALUE="#Evaluate( 'getRecordInfo.tbMessage' & counter )#"
SIZE="50"
MAXLENGTH="50"
REQUIRED="no"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Open link where?</TD>
<TD CLASS="winTabField">
<CFIF #Evaluate( 'getRecordInfo.newWindow' & counter )# NEQ "">
<CFSET temp = #Evaluate( 'getRecordInfo.newWindow' & counter )#>
<CFELSE>
<CFSET temp = 1>
</CFIF>
<CFINPUT TYPE="Radio" NAME="newWindow#counter#" VALUE="1"
CHECKED="#Iif( VARIABLES.temp EQ '1',De( 'true' ),De( 'false' ) )#">New
Window
<CFINPUT TYPE="Radio" NAME="newWindow#counter#" VALUE="2"
CHECKED="#Iif( VARIABLES.temp EQ '2',De( 'true' ),De( 'false' )
)#">Current Window
<CFINPUT TYPE="Radio" NAME="newWindow#counter#" VALUE="3"
CHECKED="#Iif( VARIABLES.temp EQ '3',De( 'true' ),De( 'false' )
)#">Within Readiness Explorer
</TD>
</TR>
<CFIF counter NEQ 5>
<TR>
<TD>&nbsp;</TD>
<TD>&nbsp;</TD>
</TR>
</CFIF>

</CFLOOP>
</TABLE>
</CFFORM>

</TD>
</TR>
</TABLE>

</CF_srWinTab>

</CFIF>

```

8. TABBED PAGES FILES

PAGEADMIN.CFM

```
<SPAN CLASS="winTab">
```

```

<CF_srHELPSUMMARY shortName="AdminPage">
</SPAN>

<CF_srPAGEOPTIONS NAME="Admin">

PAGEERROR.CFM

<CFIF IsDefined( "FORM.cargo" )>

<CFINSERT DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#"
TABLENAME="tblFeedback">

<!-- send email to sys admin -->
<CF_srFEEDBACK TYPE="error">

</CFIF>

<CFSET mainFile =
"main.cfm?body=pageError.cfm&error_id=#URL.error_id#">
<CFSET title = "ERROR">
<CFSET formAction = "#VARIABLES.mainFile#&action=sendFeedBack">
<CFSET formName="sendFeedBack">

<CFQUERY NAME="getError" DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      type, message, detail
FROM      tblErrors
WHERE error_id = #URL.error_id#

</CFQUERY>

<!-- display appropriate header -->
<CF_srWinTab
TABS="#VARIABLES.title#"
SYSTEM_TOOLBAR="Send Feedback"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit()"
>

<TABLE BORDER=0 WIDTH=100%>

<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="Error">
</SPAN>
<P>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<CFOUTPUT>
<INPUT TYPE="hidden" NAME="cargo" VALUE="#URL.error_id#">
<INPUT TYPE="hidden" NAME="type" VALUE="ERROR">
<INPUT TYPE="hidden" NAME="ip" VALUE="#CGI.REMOTE_ADDR#">
<INPUT TYPE="hidden" NAME="dateSubmitted" VALUE="#Now()#">
<INPUT TYPE="hidden" NAME="user" VALUE="#SESSION.loginid#">
</CFOUTPUT>
<TABLE BORDER=0 WIDTH=100%>

```

```

<TR>
<TD CLASS="winTabField"><B>INFORMATION REPORTED BY READINESS
EXPLORER</B></TD>
</TR>
<TR>
<TD CLASS="winTabField"><CFOUTPUT>#getError.message#</CFOUTPUT></TD>
</TR>
<TR>
<TD>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabField"><B>INFORMATION REPORTED BY COLD FUSION</B></TD>
</TR>
<TR>
<TD CLASS="winTabField"><CFOUTPUT>#getError.detail#</CFOUTPUT></TD>
</TR>
<TR>
<TD>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabField"><B>ADDITIONAL FEEDBACK</B></TD>
</TR>
<TR>
<TD CLASS="winTabField">
<TEXTAREA CLASS="stdTextBox" COLS=70 ROWS=10 NAME="feedback"
WRAP="soft"></TEXTAREA>
</TD>
</TR>
</TABLE>
</CFFORM>

</TD>
</TR>
</TABLE>

</CF_srWinTab>

```

PAGEFEEDBACK.CFM

```

<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="#URL.type#Feedback">
</SPAN>

```

PAGEFORUMS.CFM

```

<CFPARAM NAME="URL.action" DEFAULT="View">
<CFSET MAINFILE = "main.cfm?a=resources&body=pageForums.cfm">
<CFSET MAINTABLE = "tblForums">
<CFSET IDFIELD = "forum_id">
<CFSET ADDTITLE = "CREATE A NEW FORUM">
<CFSET EDITTITLE = "EDIT FORUM">
<CFSET USESOURCE = "#VARIABLES.gvDATASOURCE_CONFIG#">

<CFIF URL.ACTION EQ "View">

```

```

<CF_SRREPORTENGINE
ACTION="run"
NAME="Forums"
INCLUDEOPTIONS="true"
HEADERTYPE="List"
SYSTEM_TOOLBAR="New"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#&action=add"
>

<CFELSEIF URL.ACTION EQ "Add">

<CFSET TITLE = "#VARIABLES.addTitle#">
<CFSET FORMACTION = "#VARIABLES.mainFile#&action=saveNew">
<CFSET FORMNAME="formAdd">

<CFELSEIF URL.ACTION EQ "saveNew">

<CFINSERT DATASOURCE="#VARIABLES.useSource#"
TABLENAME="#VARIABLES.mainTable#">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.ACTION EQ "Edit">

<CFSET TITLE = "#VARIABLES.editTitle#">
<CFSET FORMACTION = "#VARIABLES.mainFile#&action=save">
<CFSET FORMNAME="formEdit">

<CFELSEIF URL.ACTION EQ "save">

<CFUPDATE DATASOURCE="#VARIABLES.useSource#"
TABLENAME="#VARIABLES.mainTable#">
<CFLOCATION URL="#VARIABLES.mainFile#">

<CFELSEIF URL.ACTION EQ "delete">

<CFQUERY NAME="deleteInfo" DATASOURCE="#VARIABLES.useSource#">

DELETE
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>
<CFLOCATION URL="#VARIABLES.mainFile#">

</CFIF>

<CFIF ( NOT COMPARENOCASE( URL.ACTION,"Add" ) ) OR ( NOT COMPARENOCASE(
URL.ACTION,"Edit" ) )>

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

SELECT      *
FROM #VARIABLES.mainTable#
WHERE #VARIABLES.idField# = <CFIF URL.ACTION EQ "Edit">#Evaluate(
'URL.' & VARIABLES.idField )#<CFELSE>-1</CFIF>

```

```

</CFQUERY>

<!-- display appropriate header --->
<CF_SRWINTAB
TABS="#VARIABLES.title#"
SYSTEM_TOOLBAR="Save"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit()"
>

<TABLE BORDER=0 WIDTH=100%>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<CFOUTPUT>
<INPUT TYPE="hidden" NAME="#VARIABLES.idField#" VALUE="#Evaluate(
'getRecordInfo.' & VARIABLES.idField )#">
</CFOUTPUT>
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_SRHELPSUMMARY SHORTNAME="adminForums">
</SPAN>
<P>
<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD CLASS="winTabLabel">&nbsp;Name:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="name"
TYPE="text"
VALUE="#getRecordInfo.name#"
SIZE="30"
MAXLENGTH="60"
REQUIRED="yes"
MESSAGE="A Forum Name is required."
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Created By:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="createdBy"
TYPE="text"
VALUE="#Iif( URL.action EQ 'Add',De( SESSION.loginid ),De(
getRecordInfo.createdBy ) )#"
SIZE="30"
MAXLENGTH="15"
REQUIRED="yes"
MESSAGE="Who Created the report must be entered"
>
</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Created On:</TD>
<TD CLASS="winTabField">
<CFINPUT
NAME="dateCreated"

```

```

TYPE="text"
VALUE="#Iif( URL.action EQ 'Add',De( DateFormat( Now(), 'mm/dd/yyyy' )
),De( DateFormat( getRecordInfo.dateCreated, 'mm/dd/yyyy' ) ) )#"
SIZE="20"
VALIDATE="date"
MAXLENGTH="15"
REQUIRED="Yes"
MESSAGE="Date Created the report must be entered"
>
</TD>
</TR>
</TABLE>
</TD>
</TR>
</CFFORM>
</TABLE>

</CF_SRWINTAB>

<CFELSEIF NOT COMPARENOCASE( URL.ACTION, "Open" )>

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

SELECT      name, collapse
FROM        #VARIABLES.mainTable#
WHERE       #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>

<!-- Save collapse variable into Client.collapse --->
<CFPARAM NAME="URL.forum_id"   DEFAULT="0">
<CFPARAM NAME="CLIENT.collapse" DEFAULT="#getRecordInfo.collapse#">
<CFPARAM NAME="collapse"      DEFAULT="#Client.collapse#">

<!-- display appropriate header --->
<CF_SRWINTAB
TABS="#getRecordInfo.name#"
SYSTEM_TOOLBAR="List Forums, New Topic, #Iif( collapse EQ 0, De( 'Collapse
Threads' ), De( 'View Threads' ) )#, Search"
SYSTEM_TOOLBARLINKS="#VARIABLES.mainFile#, #VARIABLES.mainFile#&action=n
ewTopic&forum_id=#URL.forum_id#, #Iif( collapse EQ 0, De(
VARIABLES.mainFile & '&action=open&collapse=1&forum_id=' & URL.forum_id
), De( VARIABLES.mainFile & '&action=open&collapse=0&forum_id=' &
URL.forum_id )
)#, #VARIABLES.mainFile#&action=search&forum_id=#URL.forum_id#"
>

<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_SRHELPSUMMARY SHORTNAME="Forums">
</SPAN>
<P>

<CFIF #COLLAPSE# EQ 0>

```

```

<CFINCLUDE TEMPLATE="srListForumThreads.cfm">

<CFELSE>

<!--- Get all messages corresponding to selected forum --->
<CFQUERY DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#"
NAME="topic_query">
SELECT forumHeader_id, thread, parent, author, subject, email, host
FROM tblForumHeaders
WHERE forum_id = #URL.forum_id# and parent = 0
ORDER by thread desc, parent
</CFQUERY>

<!--- Get count of messages in threads and max date of thread --->
<CFQUERY DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#"
NAME="count_query">
SELECT thread, count(*) as tcount, max( datein ) as date_
FROM tblForumHeaders
WHERE forum_id = #URL.forum_id#
GROUP by thread
ORDER by thread desc
</CFQUERY>

<TABLE BORDER=0 WIDTH=100%>

<TR CLASS="reportFieldLabel" ALIGN="left">
<TH ALIGN="center" NOWRAP>Topics</TH>
<TH ALIGN="center" NOWRAP>Author</TH>
<TH ALIGN="center" NOWRAP>Date</TH>
</TR>

<CFLOOP QUERY="topic_query">
<CFOUTPUT>
<TR CLASS="reportFieldData#Iif( CurrentRow Mod 2,De( 'Even' ),De( 'Odd'
) )#" ALIGN="left" VALIGN="top">

<TD><A CLASS="reportOptions"
HREF="#VARIABLES.mainFile#&action=read&forum_id=#URL.forum_id#&id=#foru
mHeader_id#&thread=#thread#">#subject#
(#count_query.tcount[CurrentRow]#)</A></TD>
<TD>#author#</TD>
<TD>#DateFormat( count_query.date_[CurrentRow], 'mm-dd-yyyy' )#</TD>

</TR>
</CFOUTPUT>
</CFLOOP>

</TABLE>

</CFIF>
<CFSET CLIENT.COLLAPSE = COLLAPSE>

</TD>
</TR>
</TABLE>

</CF_SRWINTAB>

```

```

<CFELSEIF NOT COMPARENOCASE( URL.ACTION, "Read" )>

<SCRIPT LANGUAGE="javascript">

function quoteMessage()
{
document.forumForm.body.value = document.forumForm.body.value +
document.forumForm.hide.value;
document.forumForm.hide.value='';
document.forumForm.body.focus();
}

</SCRIPT>

<CFQUERY NAME="getRecordInfo" DATASOURCE="#VARIABLES.useSource#">

SELECT      name
FROM        #VARIABLES.mainTable#
WHERE       #VARIABLES.idField# = #Evaluate( 'URL.' & VARIABLES.idField )#

</CFQUERY>

<!-- select message --->
<CFQUERY NAME="getMessage" DATASOURCE="#VARIABLES.useSource#">

SELECT      author, subject, email, host, datein, body, parent,
forum_id, h.forumHeader_id, h.thread, email_reply
FROM        tblForumHeaders h, tblForumBody b
WHERE       h.forumHeader_id = b.forumHeader_ID
AND         h.forumHeader_ID = #URL.id#

</CFQUERY>

<CFSET text          = #getMessage.body#>
<CFSET convert_type  = "fromHTML">
<CFINCLUDE TEMPLATE = "srMessageConvert.cfm">
<CFSET body_quote    = #text#>

<!-- display appropriate header --->
<CF_srWINTAB
TABS="#getRecordInfo.name#"
TOOLBAR="List Forums,Go To Top,New Topic,Search"
TOOLBARLINKS="#VARIABLES.mainFile#, #VARIABLES.mainFile#&action=open&forum_id=#URL.forum_id#, #VARIABLES.mainFile#&action=newTopic&forum_id=#URL.forum_id#, #VARIABLES.mainFile#&action=search&forum_id=#URL.forum_id#"
SYSTEM_TOOLBAR="Quote Message,Post Reply"
SYSTEM_TOOLBARLINKS="javascript:quoteMessage(), javascript:document.forumForm.submit()"
>

<TABLE BORDER=0 WIDTH=100%>
<TR>
<TD>
<SPAN CLASS="winTab">
<CF_SRHELPSUMMARY SHORTNAME="Forums">
</SPAN>

```

```

<P>
<TABLE BORDER=0 WIDTH=100%>

<CFOUTPUT QUERY="getMessage">
<TR>
<TD CLASS="winTabLabel">&nbsp;Subject:</TD>
<TD CLASS="winTabField">&nbsp;#subject#</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Author:</TD>
<TD CLASS="winTabField">&nbsp;<A CLASS="reportOptions"
HREF="mailto:#RTrim( email )#?subject=#RTrim( subject )#">#RTrim(
author )#</A></TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Submitted:</TD>
<TD CLASS="winTabField">&nbsp;#DateFormat( datein, 'mm-dd-yyyy' )#</TD>
</TR>
<TR>
<TD COLSPAN=2>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Discussion:</TD>
<TD>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabField" COLSPAN=2>#body#</TD>
</TR>
<TR>
<TD COLSPAN=2>&nbsp;<CFINCLUDE TEMPLATE="srListForumThreads.cfm"></TD>
</TR>
<TR>
<TD COLSPAN=2>&nbsp;</TD>
</TR>
<TR>
<TD CLASS="winTabLabel">&nbsp;Reply:</TD>
<TD>&nbsp;</TD>
</TR>
<CFFORM ACTION="post.cfm" NAME="forumForm">
<INPUT NAME="forumHeader_id" TYPE="hidden" VALUE="#forumHeader_id#">
<INPUT NAME="thread" TYPE="hidden" VALUE="#thread#">
<INPUT NAME="forum_id" TYPE="hidden" VALUE="#forum_id#">
<INPUT NAME="parent" TYPE="hidden" VALUE="#parent#">
<INPUT NAME="date" TYPE="hidden"
VALUE="#datein#">
<INPUT NAME="subject" TYPE="hidden" VALUE="#Iif(
CompareNoCase( Left( subject,4 ), 'RE: ' ), De( 'RE: ' & subject ), De(
subject ) )#">

<INPUT NAME="hide" TYPE="hidden" VALUE="
#RTrim( author )# wrote:
-----
#body_quote#

">
<TR>
<TD CLASS="winTabField" COLSPAN=2>

```

```

<TEXTAREA CLASS="stdTextBox" COLS=100 ROWS=10 NAME="body"
WRAP="soft"></TEXTAREA>
</TD>
</TR>
</CFFORM>
</CFOUTPUT>
</TABLE>
</TD>
</TR>
</TABLE>

</CF_srWINTAB>

</CFIF>

```

PAGEHELP.CFM

```

<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="HelpPage">
</SPAN>

<CF_srPAGEOPTIONS NAME="Help">

```

PAGEHOME.CFM

```

<CFIF #SESSION.FirstOn#>

<CF_getGREETING><BR><BR>

</CFIF>

<TABLE BORDER=0 WIDTH=100%>

<TR VALIGN="top">
<TD WIDTH=50%>
<CF_srDIALOGWINDOW TITLE="Information">
<CFINCLUDE TEMPLATE="getInformation.cfm">
</CF_srDIALOGWINDOW>
</TD>
<TD WIDTH=50%>
<CF_srDIALOGWINDOW TITLE="My Security">
<CFINCLUDE TEMPLATE="getSecurity.cfm">
</CF_srDIALOGWINDOW>
</TD>
</TR>

<TR>
<TD COLSPAN=2 VALIGN="middle" ALIGN="center">
<BR>
<HR CLASS="standard" WIDTH=75% SIZE=1 NOSHADE>
</TD>
</TR>

<TR>
<TD COLSPAN=2 VALIGN="top">

```

```

<CF_srPAGEOPTIONS NAME="Home">
</TD>
</TR>

</TABLE>

```

PAGELOGOFF.CFM

```

<CFSET StructDelete(Session, "InitializeThis")>
<CFSET StructDelete(Session, "loginid")>
<CFSET StructDelete(Session, "LastLogOn")>
<CFSET StructDelete(Session, "FirstOn")>
<CFSET StructDelete(Session, "LogOnAttempts")>
<CFSET StructDelete(Session, "LogSequence")>
<CFSET StructDelete(Session, "showPanel")>
<CFSET StructDelete(Session, "deskTop")>
<CFSET StructDelete(Session, "colors")>

<CFLOCATION URL="index.htm">

```

PAGEPERSONALIZE.CFM

```

<SPAN CLASS="winTab">
<CF_srHELPSUMMARY shortName="Personalize"
Page="#aComponent[URL.aID] [cAPPLICATION] [cAPPLICATION_NAME]#">
</SPAN>

<CF_srPAGEOPTIONS NAME="Personalize">

```

PAGEPROFILES.CFM

```

<CFPARAM NAME="URL.action" DEFAULT="View">
<CFSET MAINFILE = "main.cfm?a=profiles&body=pageProfiles.cfm">
<CFSET USESOURCE = "#VARIABLES.gvDATASOURCE_DATA#">

<CFIF NOT #CompareNoCase( URL.action, "saveNew" )#>

<CFQUERY NAME="getFieldInfo"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      tblProfileFields.pfName
FROM        tblProfileFields INNER JOIN (tblProfileGroups INNER JOIN
tblLinkFieldToProfile ON tblProfileGroups.pg_id =
tblLinkFieldToProfile.pg_id) ON tblProfileFields.pf_id =
tblLinkFieldToProfile.pf_id
WHERE       tblProfileGroups.pg_id = #URL.pg_id#

</CFQUERY>

<CFINSERT DATASOURCE="#VARIABLES.gvDATASOURCE_DATA#"
TABLENAME="#FORM.tableName#" FORMFIELDS="#ValueList(
getFieldInfo.pfName )#">
<CFLOCATION URL="#VARIABLES.mainFile#&#URL.pg_id#">

```

```

</CFIF>

<CFQUERY NAME="getProfiles"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      pg_id,pgName,pgLabel,'#VARIABLES.mainFile#&pg_id=' & pg_id
AS idURL
FROM  tblProfileGroups
ORDER BY pgName

</CFQUERY>

<CFIF IsDefined( "URL.pg_id" )>
<CFSET var_pg_id = #URL.pg_id#>
<CFSET tabSelected = #ListFind( ValueList( getProfiles.pg_id ),
URL.pg_id )#>
<CFELSE>
<CFSET var_pg_id = #getProfiles.pg_id[1]#>
<CFSET tabSelected = 1>
</CFIF>

<CFSET FORMACTION =
"#VARIABLES.mainFile#&action=saveNew&pg_id=#VARIABLES.var_pg_id#">
<CFSET FORMNAME = "profiles">

<!-- display appropriate header --->
<CF_srWinTab
TABS="#ValueList( getProfiles.pgLabel )#"
TABURL="#ValueList( getProfiles.idURL )#"
TABSELECTED="#VARIABLES.tabSelected#"
SYSTEM_TOOLBAR="Save"
SYSTEM_TOOLBARLINKS="javascript:document.#VARIABLES.formName#.submit()"
BASEURL="#VARIABLES.mainFile#"
>

<TABLE BORDER=0 WIDTH=100%>

<TR>
<TD>
<SPAN CLASS="winTab">
<CF_srHELPSUMMARY SHORTNAME="profiles">
</SPAN>
<P>
<CFQUERY NAME="getProfileInfo"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      *
FROM  tblProfileGroups
WHERE pg_id = #VARIABLES.var_pg_id#

</CFQUERY>

<CFQUERY NAME="getFieldInfo"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      tblProfileFields.*

```

```

FROM tblProfileFields INNER JOIN (tblProfileGroups INNER JOIN
tblLinkFieldToProfile ON tblProfileGroups.pg_id =
tblLinkFieldToProfile.pg_id) ON tblProfileFields.pf_id =
tblLinkFieldToProfile.pf_id
WHERE      tblProfileGroups.pg_id = #getProfileInfo.pg_id#;

</CFQUERY>

<CFSET NumberOfItems = #getFieldInfo.RecordCount#>
<CFSET NumberOfRows = #Iif( getProfileInfo.pgLayoutCols EQ
2,NumberOfItems,Ceiling( NumberOfItems / 2 ) )#>
<CFSET ItemNumberToDisplay_hdr = 0>

<CFFORM ACTION="#VARIABLES.formAction#" NAME="#VARIABLES.formName#">
<TABLE BORDER=0 WIDTH=100%>
<CFOUTPUT>
<INPUT TYPE="hidden" NAME="tableName"
VALUE="tbl#getProfileInfo.pgName#">
</CFOUTPUT>

<!-- how many row sections to display? -->
<CFLOOP INDEX="RowCounter" FROM="1" TO="#VARIABLES.NumberOfRows#">

<TR VALIGN="top">

<CFLOOP INDEX="ColumnCounter" FROM="1"
TO="#getProfileInfo.pgLayoutCols#" STEP="2">

<CFSET VARIABLES.ItemNumberToDisplay_hdr = #Evaluate(
VARIABLES.ItemNumberToDisplay_hdr + 1 )#>
<CFOUTPUT>
<CFIF VARIABLES.NumberofItems - VARIABLES.ItemNumberToDisplay_hdr GTE
0>
<TD CLASS="winTabField"><CFIF
#getFieldInfo.pfRequiredField[VARIABLES.ItemNumberToDisplay_hdr]#>* </CF
IF><B>#getFieldInfo.pfLabel[VARIABLES.ItemNumberToDisplay_hdr]#: </B></T
D>
<TD CLASS="winTabField">&nbsp;
<CFSWITCH
EXPRESSION="#getFieldInfo.pfType[VARIABLES.ItemNumberToDisplay_hdr]#">
<!-- TEXT --->
<CFCASE VALUE="1">
<INPUT CLASS="stdTextBox" TYPE="text"
NAME="#getFieldInfo.pfName[VARIABLES.ItemNumberToDisplay_hdr]#"
SIZE="20"
MAXLENGTH="#getFieldInfo.pfSize[VARIABLES.ItemNumberToDisplay_hdr]#"
onFOCUS="setStatus(
'#getFieldInfo.pfHelpMessage[VARIABLES.ItemNumberToDisplay_hdr]#' )"
onBLUR="setStatus( ' ' )">
</CFCASE>
<!-- BOOLEAN --->
<CFCASE VALUE="2">
<INPUT NAME="#getFieldInfo.pfName[VARIABLES.ItemNumberToDisplay_hdr]#"
TYPE="radio" VALUE="1" onFOCUS="setStatus(
'#getFieldInfo.pfHelpMessage[VARIABLES.ItemNumberToDisplay_hdr]#' )"
onBLUR="setStatus( ' ' )">Yes

```



```
<CF_srPAGEOPTIONS NAME="Tools">
```

9. SET INFORMATION FILES

SETUSERINFORMATION.CFM

```
<CFINCLUDE TEMPLATE="getProgramDefaults.cfm">
```

```
<CFQUERY NAME="setUserInformation"  
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">
```

```
UPDATE      tblUserAccounts  
SET  
<CFLOOP INDEX="counter" FROM="1" TO="#ListLen( ATTRIBUTES.fields, ','  
)#">  
#ListGetAt( ATTRIBUTES.fields, counter )# = '#ListGetAt(  
ATTRIBUTES.values, counter )#' #Iif( counter NEQ ListLen(  
ATTRIBUTES.fields ), De( ',' ), De( '' ) )#  
</CFLOOP>  
WHERE user_loginid = '#ATTRIBUTES.user#'  
  
</CFQUERY>
```

10. SUBROUTINE FILES

SRADDBODYOPTION.CFM

```
<CFASSOCIATE BASETAG="cf_srCreateBodyOption"  
DATACOLLECTION="optionList">
```

SRCREATEBODYOPTION.CFM

```
<CFIF ThisTag.ExecutionMode IS "Start">
```

```
<STYLE TYPE="text/css">
```

```
A.optionLink:link  
{  
color: black;  
text-decoration: none;  
font-size: 10pt;  
font-weight: bold;  
}
```

```
A.optionLink:active  
{  
color: black;  
text-decoration: none;  
font-size: 10pt;  
font-weight: bold;  
}
```

```
A.optionLink:visited  
{
```

```

color: black;
text-decoration: none;
font-size: 10pt;
font-weight: bold;
}

A.optionLink:hover
{
color: black;
text-decoration: underline;
font-size: 10pt;
}

TD.optionBody
{
font: 10pt Tahoma, Verdana, Arial;
color: blue;
}

</STYLE>

<!-- only process when end tag is reached --->
<CFELSEIF ThisTag.ExecutionMode IS "End">

<!-- default parameters --->
<CFPARAM NAME="ATTRIBUTES.MaxItemsPerRow"          DEFAULT="2">

<!-- get total number of items created --->
<CFSET NumberOfItems = #ArrayLen( ThisTag.optionList )#>
<CFSET NumberOfRows = #Ceiling( VARIABLES.NumberOfItems /
ATTRIBUTES.MaxItemsPerRow )#>
<CFSET ItemNumberToDisplay_hdr = 0>
<CFSET ItemNumberToDisplay_ftr = 0>

<TABLE BORDER=0 WIDTH=100% CELLSPACING=4>

<!-- add some space if necessary --->
<TR>
<CFOUTPUT>
<TD COLSPAN="#Evaluate( VARIABLES.NumberofItems * 2 )#">&nbsp;  </TD>
</CFOUTPUT>
</TR>

<!-- how many row sections to display? --->
<CFLOOP INDEX="RowCounter" FROM="1" TO="#VARIABLES.NumberOfRows#">

<!-- add titles --->
<TR VALIGN="top">
<CFLOOP INDEX="ColumnCounter" FROM="1"
TO="#ATTRIBUTES.MaxItemsPerRow#">
<CFSET VARIABLES.ItemNumberToDisplay_hdr = #Evaluate(
VARIABLES.ItemNumberToDisplay_hdr + 1 )#>
<CFOUTPUT>
<CFIF VARIABLES.NumberofItems - VARIABLES.ItemNumberToDisplay_hdr GTE
0>
<TD VALIGN="top" ALIGN="left"
WIDTH="#Evaluate(100/ATTRIBUTES.MaxItemsPerRow)%"><A

```

```

CLASS="optionLink"
HREF="#ThisTag.optionList[VARIABLES.ItemNumberToDisplay_hdr].url#"><FONT
FACE="Arial,sans"
SIZE="3"><B>#ThisTag.optionList[VARIABLES.ItemNumberToDisplay_hdr].titl
e#</B></FONT></A></TD>
<CFELSE>
<TD WIDTH="#Evaluate(100/ATTRIBUTES.MaxItemsPerRow)%">&nbsp;</TD>
</CFIF>
</CFOUTPUT>
</CFLOOP>
</TR>

<!-- add option text --->
<TR>
<CFLOOP INDEX="ColumnCounter" FROM="1"
TO="#ATTRIBUTES.MaxItemsPerRow#">
<CFSET VARIABLES.ItemNumberToDisplay_ftr = #Evaluate(
VARIABLES.ItemNumberToDisplay_hdr + 1 )#>
<CFOUTPUT>
<CFIF VARIABLES.NumberofItems - VARIABLES.ItemNumberToDisplay_ftr GTE
0>
<TD CLASS="optionBody" VALIGN="top"
WIDTH="#Evaluate(100/ATTRIBUTES.MaxItemsPerRow)%">#ThisTag.optionList[
VARIABLES.ItemNumberToDisplay_ftr].text#</TD>
<CFELSE>
<TD WIDTH="#Evaluate(100/ATTRIBUTES.MaxItemsPerRow)%">&nbsp;</TD>
</CFIF>
</CFOUTPUT>
</CFLOOP>
</TR>

<TR>
<CFOUTPUT>
<TD COLSPAN="#Evaluate( VARIABLES.NumberofItems * 2 )#">&nbsp;</TD>
</CFOUTPUT>
</TR>

</CFLOOP>

</TABLE>

</CFIF>

```

SRDIALOGWINDOW.CFM

```
<cfsetting enablecfoutputonly="yes">
```

```
<!-- NAME: CF_ACTIVETABLE
```

```

Activefeedback.com 2000
http://www.activefeedback.com
Email: support@activefeedback.com
Phone: 408-230-4444

```

```
--->
```

```
<!-- Default width of the window --->
```

```

<cfparam name="attributes.width" default="100%">
<!-- Default height of the window --->
<cfparam name="attributes.height" default="">
<!-- Default table cellpadding --->
<cfparam name="attributes.cellpadding" default="0">
<!-- Default table cellspacing --->
<cfparam name="attributes.cellspacing" default="0">
<!-- Default table border --->
<cfparam name="attributes.border" default="0">
<!-- Default window header height --->
<cfparam name="attributes.headerheight" default="14">
<!-- Default header font --->
<cfparam name="attributes.headerfont" default="<font
face=verdana,arial,Helvetica color=ffffff size=1">
<!-- Background color of the header --->
<cfparam name="attributes.headerbgcolor" default="midnightblue">
<!-- Background color of the form itself --->
<cfparam name="attributes.bodybgcolor" default="buttonface">
<!-- Default Form title --->
<cfparam name="attributes.title" default="">
<!-- Allow scrolling of the internal layer --->
<cfparam name="attributes.scroll" default="false">
<!-- Default minimize parameter --->
<cfparam name="attributes.minimize" default="false">
<!-- Random container parameter necessary to control layers --->
<cfparam name="attributes.containerid"
default="#randrange(111111111,999999999)#">

<cfswitch expression="#thistag.executionmode#">
<!-- start tag processing --->
<cfcase value='start'>
<cfoutput>
<script language=jscript>
<!--
function change_d#attributes.containerid#() {
if (c#attributes.containerid#.style.display == "") {
c#attributes.containerid#.style.display = "none";
}
else{
c#attributes.containerid#.style.display= "";
}
}
-->
</script>
<div style="{ padding-left : 3; padding-right : 3; padding-top : 2;
padding-bottom : 5; background-color : buttonface; border-style :
outset; border-width : thin; width:#attributes.width#; <cfif
len(attributes.height)>height:#attributes.height#; </cfif}">

<table width="#attributes.width#"
cellpadding="#attributes.cellpadding#"
cellspacing="#attributes.cellspacing#" border="#attributes.border#">
<tr bgcolor="#attributes.headerbgcolor#">
<td CLASS="adWindowTitle" height="#attributes.headerheight#">
&nbsp;  <b>#attributes.title#</b>
</td>

```

```

<td CLASS="adWindowTitle" align="right"
height="#attributes.headerheight#" valign="middle" nowrap>
<button onclick="change_d#attributes.containerid#()" style=" height:
15px; width: 15px; cursor : hand; line-height : 3;"></button>
</td>
</tr>
</table>
<!-- switch default layer at the click of a button --->
<cfif attributes.minimize>
<div id="c#attributes.containerid#"
style="position:relative;top:0;left:0; display: none;">

<cfelse>
<div id="c#attributes.containerid#"
style="position:relative;top:0;left:0; display: ;" <cfif
attributes.scroll>overflow : scroll;</cfif>
</cfif>
<table width="#attributes.width#" cellpadding="5"
cellspacing="#attributes.cellspacing#" border="#attributes.border#">
<tr><td CLASS="adWindowBody">

</cfoutput>
</cfcase>
<cfcase value='end'>
<!-- end tag processing --->
<cfoutput>
</td></tr></table>
</div>
</div>
</cfoutput>
</cfcase>
</cfswitch>
<cfsetting enablecfoutputonly="no">

```

SRERRORMONITOR.CFM

```

<CFPARAM NAME="ATTRIBUTES.ERROR_CODE"          DEFAULT="None">
<CFPARAM NAME="ATTRIBUTES.SQLSTATE"            DEFAULT="None">

<CFINCLUDE TEMPLATE="getProgramDefaults.cfm">

<!-- if admin email, then email --->

<!-- if developer email, then email --->

<CFTRANSACTION>

<CFQUERY DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

INSERT INTO tblErrors (
detail,message,NavtiveErrorCode,SQLState,Type,dateTimeofError,error_use
r,error_ip )
VALUES (
'#ATTRIBUTES.detail#','#ATTRIBUTES.message#','#ATTRIBUTES.Error_Code#',
'#ATTRIBUTES.SQLState#','#ATTRIBUTES.Type#','#CreateODBCDate( Now()
)#','#ATTRIBUTES.user#','#ATTRIBUTES.ip#' )

```

```
</CFQUERY>
```

```
<CFQUERY NAME="getErrorID"  
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">
```

```
SELECT      Max( error_id ) AS errorid  
FROM      tblErrors
```

```
</CFQUERY>
```

```
</CFTRANSACTION>
```

```
<CFLOCATION  
URL="main.cfm?body=pageError.cfm&error_id=#getErrorID.errorid#">
```

SRFEEDBACK.CFM

```
<CFLOCATION  
URL="main.cfm?body=pageFeedback.cfm&type=#ATTRIBUTES.type#">
```

SRLINKFINDER.CFM

```
<!--
```

```
=====
```

Utility:	ColdFusion LinkFinder v1.0
Author:	Dain Anderson
Email:	dain_anderson@yahoo.com
Latest Version:	http://www.cfcomet.com/

```
=====
```

```
Contributor(s):  
Michael Dinowitz (mdinowit@houseoffusion.com) for his example of  
email RegEx usage in Fusion Authority Weekly News Alert #38.  
--->
```

```
<CFPARAM NAME="Attributes.Data" DEFAULT="">  
<CFIF NOT LEN(Attributes.Data)><B><FONT COLOR="FF0000">Error: No data  
to parse!</FONT></B><BR></CFIF>  
<CFSCRIPT>  
this = Attributes.Data;  
this = REReplaceNoCase(this, "([^.*)((ht|f)(tps?:\\\/\\\/) ([^  
|;|,|<|>|#|!|\"|:|[:cntrl:]]*))", "\1<A STYLE=""text-decoration:  
underline"" TARGET=""_blank"" HREF=""\2"">\2</A>", "ALL");  
this = REReplaceNoCase(this, "([[:alnum:]][-a-zA-Z0-  
9_%\.]*)?[:alnum:]]@[:alnum:]][-a-zA-Z0-9%\>.*\.[[:alpha:]]{2,})",  
"<A STYLE=""text-decoration: underline"" HREF=""mailto:\1"">\1</A>",  
"ALL");  
</CFSCRIPT>  
<CFOUTPUT>#ParagraphFormat(this)#</CFOUTPUT>
```

SRLISTFORUMTHREADS.CFM

```
<TABLE BORDER=0 WIDTH=100% cellspacing="0" cellpadding="0">
```

```
<CFPARAM NAME = "admin" DEFAULT="0">
```

```

<CFQUERY NAME="thread_query"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      forumHeader_id, thread, parent, author, subject, email,
host, datein
FROM      tblForumHeaders
WHERE      0=0
<CFIF IsDefined( "URL.thread" )>
AND thread = #URL.thread#
ORDER BY  parent, datein desc
<CFELSE>
AND forum_id = #URL.forum_id#
ORDER BY  thread desc, parent, datein desc
</CFIF>

</CFQUERY>

<!-- place topics data into "threads" struct --->
<CFSET threads = ArrayNew( 1 )>

<!-- form "threads" array of structure --->
<CFLOOP QUERY = "thread_query">

<CFSCRIPT>
threads[CurrentRow]=StructNew();
StructInsert(threads[CurrentRow], "id",
thread_query.forumHeader_id[CurrentRow]);
StructInsert(threads[CurrentRow], "thread",
thread_query.thread[CurrentRow]);
StructInsert(threads[CurrentRow], "parent",
thread_query.parent[CurrentRow]);
StructInsert(threads[CurrentRow], "author",
thread_query.author[CurrentRow]);
StructInsert(threads[CurrentRow], "subject",
thread_query.subject[CurrentRow]);
StructInsert(threads[CurrentRow], "email",
thread_query.email[CurrentRow]);
StructInsert(threads[CurrentRow], "host",
thread_query.host[CurrentRow]);
StructInsert(threads[CurrentRow], "date",
thread_query.datein[CurrentRow]);
</CFSCRIPT>

</CFLOOP>

<CFSET count = #thread_query.RecordCount#>

<!-- "lines" is array of branches in tree --->
<CFSET lines = "">

<TR CLASS="reportFieldLabel" ALIGN="left">
<TH ALIGN="center" NOWRAP>Topics</TH>
<TH ALIGN="center" NOWRAP>Author</TH>
<TH ALIGN="center" NOWRAP>Date</TH>
</TR>

```



```

url_pattern = ListGetAt(url_patterns, i);
/* get position of occurrence */
f = REFindNoCase(url_pattern, text, 1, "TRUE");
while (f.pos[1] neq 0) {
/* get url from founded construction */
url = Mid(text, f.pos[1], f.len[1]);
/* masking ":" */
url = REReplace(url, ":", "::<");
/* Convert urls */
if (Left(url,6) is "mailto") {
/* if "mailto:" - place url text without "mailto:" */
text = REReplace(text, url_pattern,
"<a href='#url#'>#Right(url, Len(url)-8)#</a>");
} else {
text = REReplace(text, url_pattern,
"<a href='#url#'>#url#</a>");
}
/* get next occurrence */
f = REFindNoCase(url_pattern, text, f.pos[1]+1, "TRUE");
}
}
/* Restore ":" in urls */
text = Replace(text, Chr(38) & "amp;", Chr(38), "ALL");
text = REReplace(text, "::<", ":", "ALL");
</cfscript>
</cfif>

<cfif convert_type is "fromHTML">
<cfscript>
p1 = "<i>,</i>,<i>,</i>,<b>,</b>,<b>,</b>";
text = REReplace(text, "<a href='mailto:[^>]+>", "mailto:", "ALL");
text = REReplace(text, "<a[^>]+>", "", "ALL");
text = REReplace(text, "</a>", "", "ALL");
text = Replace(text, "<br>", Chr(13) & Chr(10), "ALL");

</cfscript>
</cfif>

```

SRPAGEOPTIONS.CFM

```

<CFINCLUDE TEMPLATE="getProgramDefaults.cfm">

<CFQUERY NAME="getPageStructure"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      *
FROM        tblPageOptions
WHERE       po_name = '#ATTRIBUTES.Name#'

</CFQUERY>

<CFQUERY NAME="getPageOptions"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      tblPageOptionItems.*

```

```
FROM ( tblLinkOptionsToPage INNER JOIN tblPageOptionItems ON
tblLinkOptionsToPage.poi_id = tblPageOptionItems.poi_id ) INNER JOIN
tblPageOptions ON tblLinkOptionsToPage.po_id = tblPageOptions.po_id
WHERE      tblPageOptions.po_name = '#ATTRIBUTES.Name#'
```

```
</CFQUERY>
```

```
<CF_srCREATEBODYOPTION MAXITEMSPERROW="#getPageStructure.po_columns#">
```

```
<CFLOOP QUERY="getPageOptions">
```

```
<CF_srADDBODYOPTION
```

```
TITLE="#getPageOptions.poi_title#"

```

```
URL="#getPageOptions.poi_url#"

```

```
TEXT="#Iif( getPageStructure.po_showText,De( getPageOptions.poi_text
),De( ' ' ) )#"

```

```
>
```

```
</CFLOOP>
```

```
</CF_srCREATEBODYOPTION>
```

SRREPORTENGINE.CFM

```
<CFINCLUDE TEMPLATE="getProgramDefaults.cfm">
```

```
<CFIF CGI.queryString EQ "">
```

```
<CFPARAM NAME="ATTRIBUTES.includeOptions"           DEFAULT="False">
<CFPARAM NAME="ATTRIBUTES.OptionList"              DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.OptionLink"              DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.headerType"              DEFAULT="Basic">
<CFPARAM NAME="ATTRIBUTES.addWhere"                DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.addTitle"                DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.addOptionParam"          DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.toolBar"                 DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.toolBarLinks"            DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.system_toolBar"          DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.system_toolBarLinks"     DEFAULT="">
```

```
<CFSET rptName = "#ATTRIBUTES.name#">
```

```
<CFSET rptAction = "#ATTRIBUTES.action#">
```

```
<CFSET rptOptions = "#Iif( ATTRIBUTES.includeOptions,true,Iif(
ATTRIBUTES.optionList NEQ '',true,false ) )#">
```

```
<CFSET rptOptionList = "#ATTRIBUTES.optionList#">
```

```
<CFSET rptOptionLink = "#ATTRIBUTES.optionLink#">
```

```
<CFSET rptHeaderType = "#ATTRIBUTES.headerType#">
```

```
<CFSET rptImageDir = "#VARIABLES.gvIMAGE_DIRECTORY#">
```

```
<CFSET rptAddWhere = "#ATTRIBUTES.addWhere#">
```

```
<CFSET rptAddTitle = "#ATTRIBUTES.addTitle#">
```

```
<CFSET rptAddOptionParam = "#ATTRIBUTES.addOptionParam#">
```

```
<CFSET rptToolBar = "#ATTRIBUTES.toolBar#">
```

```
<CFSET rptToolBarLinks = "#ATTRIBUTES.toolBarLinks#">
```

```
<CFSET rptSystem_ToolBar = "#ATTRIBUTES.System_toolBar#">
```

```
<CFSET rptSystem_ToolBarLinks = "#ATTRIBUTES.System_toolBarLinks#">
```

```

<CFELSE>

<CFSET rptName = "#URL.name#">
<CFSET rptAction = "#URL.action#">
<CFSET rptOptions = "#URL.includeOptions#">

</CFIF>

<CFIF VARIABLES.rptAction EQ "List">

<CFELSEIF VARIABLES.rptAction EQ "Run">

<!--- get report information --->
<CFQUERY NAME="getReportInfo"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT
    report_ID,sqlBody,reportName,shortName,idField,useDataSource
FROM   tblReports
WHERE  shortName = '#VARIABLES.rptName#'

</CFQUERY>

<CFSET rptSqlBody = #PreserveSingleQuotes( getReportInfo.sqlBody )#>

<!--- check to see if additional information has been added to query --
-->
<CFIF VARIABLES.rptAddWhere NEQ "">

<CFSET otherClause = Find( "ORDER BY",VARIABLES.rptSqlBody )>
<CFIF otherClause EQ 0>
<CFSET otherClause = Find( "GROUP BY",VARIABLES.rptSqlBody )>
<CFIF otherClause EQ 0>
<CFSET otherClause = Find( "HAVING",VARIABLES.rptSqlBody )>
</CFIF>
</CFIF>

<!--- see if there is a WHERE clause --->
<CFIF VARIABLES.rptSqlBody CONTAINS "where">

<CFIF otherClause NEQ 0>

<CFSET rptSqlBody = Insert( " AND " & VARIABLES.rptAddWhere & "
",rptSqlBody,otherClause - 1 )>

<CFELSE>

<CFSET rptSqlBody = Insert( " AND " &
VARIABLES.rptAddWhere,rptSqlBody,Len( rptSqlBody ) )>

</CFIF>

<CFELSE>

<CFIF otherClause NEQ 0>

```

```

<CFSET rptSqlBody = Insert( " WHERE " & VARIABLES.rptAddWhere & "
",rptSqlBody,otherClause - 1 )>

<CFELSE>

<CFSET rptSqlBody = Insert( " WHERE " &
VARIABLES.rptAddWhere,rptSqlBody,Len( rptSqlBody ) )>

</CFIF>

</CFIF>

</CFIF>

<!-- run SQL to gather report data -->
<CFQUERY NAME="getReportData" DATASOURCE="#Evaluate(
'VARIABLES.gvDATASOURCE_' & getReportInfo.useDataSource )#">

#PreserveSingleQuotes( VARIABLES.rptSqlBody )#

</CFQUERY>

<CFIF NOT #CompareNoCase( ATTRIBUTES.headerType,"ListOnly" )#>

<CFINCLUDE TEMPLATE="srReportEngineBody.cfm">

<CFELSE>

<!-- display appropriate header -->
<CF_srWinTab
TABS="#Ucase( getReportInfo.reportName )#&nbsp;#VARIABLES.rptAddTitle#"
TABSELECTED="1"
TOOLBAR="#VARIABLES.rptToolBar#"
TOOLBARLINKS="#VARIABLES.rptToolBarLinks#"
SYSTEM_TOOLBAR="#VARIABLES.rptSystem_ToolBar#"
SYSTEM_TOOLBARLINKS="#VARIABLES.rptSystem_ToolBarLinks#"
>
<CFINCLUDE TEMPLATE="srReportEngineBody.cfm">
</CF_srWinTab>

</CFIF>

</CFIF>

```

SRSTOCKGRABBER.CFM

```

NAME:          CF_StockGrabber
FILE:          StockGrabber.cfm
CREATED:       09/20/1997
LAST MODIFIED: 06/02/1998
VERSION:       2.0
AUTHOR:        Rob Bilson (rbils@amkor.com)
COPYRIGHT:     Copyright (C) 1997-1998 by Rob Bilson, All Rights
Reserved

```

This program is free software; you can redistribute it and/or modify it under the terms of the GNU General Public License as published by the Free Software Foundation; either version 2 of the License, or any later version.

```
<cfif isdefined('attributes.ErrorCheck')>
<CFIF #Attributes.ErrorCheck# IS "Yes">
```

```
<CFIF NOT ISDEFINED('attributes.ErrorPage')>
<CFSET #Attributes.ErrorPage#="error.cfm">
</CFIF>
```

```
<CFIF NOT ISDEFINED('attributes.ErrorMailTo')>
<CFSET #Attributes.ErrorMailTo#="">
</CFIF>
```

```
<CFERROR type="request"
template="#Attributes.ErrorPage#"
mailto="#Attributes.ErrorMailTo#">
</CFIF>
</CFIF>
```

```
<!-------
BECAUSE OF A BUG IN THE WAY THAT CFHTTP INTERACTS WITH COMMA
SEPERATED TEXT FILES, IT IS NECESSARY TO INSERT AN EXTRA
RECORD IN THE BEGINNING THAT GETS OMITTED BY CFHTTP. THIS EXTRA
RECORD IS ADDED BY APPENDING A DUPLIUATE TICKER SYMBOL TO THE
LIST BEING PASSED TO YAHOO
----->
```

```
<CFSET #Symbol_List# = #ListFirst(Attributes.TickerSymbols)#>
```

```
<!-------
CHECKS TO SEE IF A LIST OF SYMBOLS WAS PASSED, AND IF SO,
STRIPS THE COMMA DELIMITERS AND ADDS THE + SIGN INSTEAD
SO THAT IT CAN BE PASSED TO YAHOO IN A URL. IF NO SYMBOLS
ARE PASSED, CF_STOCKGRABBER WILL STILL EXECUTE, USING
YAHOO'S SYMBOL :-)
----->
```

```
<CFIF #ParameterExists(Attributes.TickerSymbols)# IS "Yes">
<CFIF #Attributes.TickerSymbols# IS "">
<CFSET #TickerSymbols# ="yhoo">
<CFSET #Symbol_List# = ListAppend(#Symbol_List#, #TickerSymbols#)>
<CFELSE>
<CFSET #TickerSymbols# ="#Attributes.TickerSymbols#">
<CFSET #Symbol_List# = ListAppend(#Symbol_List#, #TickerSymbols#)>
</CFIF>
<CFELSE>
<CFSET #TickerSymbols# ="yhoo">
<CFSET #Symbol_List# = ListAppend(#Symbol_List#, #TickerSymbols#)>
</CFIF>
```

```
<!-------
THIS IS WHERE THE DELIMITERS ARE CHANGED TO + SIGNS
----->
<CFSET #Symbol_List# = ListChangeDelims(#Symbol_List#, "+")>
```

```
<!-------
BECAUSE AN EXTRA + SIGN IS ADDED TO THE LIST, IT
```

NEEDS TO BE REMOVED OR IT WILL CAUSE AN ERROR

```
----->
<CFSET #RemovePlus# = Len(#Symbol_List#)>
<CFSET #Symbol_List# = RemoveChars(#Symbol_List#, #RemovePlus#, 1)>

<CFIF #ParameterExists(Attributes.QueryName)# IS "Yes">
<CFIF #Attributes.QueryName# IS "">
<CFSET #QueryName# = "GetQuotes">
<CFSET #Caller.QueryName# = "GetQuotes">
<CFELSE>
<CFSET #QueryName# = "#Attributes.QueryName#">
</CFIF>
<CFELSE>
<CFSET #QueryName# = "GetQuotes">
<CFSET #Caller.QueryName# = "GetQuotes">
</CFIF>
```

```
<!-------
Using CFHttp, go out to Yahoo's site, query the
server for the desired quotes, returning the results
as a comma seperated list, parsing the list into
variables, and returning them to the user.
----->
```

```
<!--- Get the Us/Canada quotes --->
<CFHTTP METHOD="GET"
URL="http://quote.yahoo.com/download/quotes.csv?Symbols=#Symbol_List#&format=slldt1c1ohgv&ext=.csv"
NAME="#QueryName#"
COLUMNS="Symbol,Last_Traded_Price,Last_Traded_Date,Last_Traded_Time,Change,Opening_Price,Days_High,Days_Low,Volume"
DELIMITER=","
TEXTQUALIFIER="">
```

```
<!--- RECREATE QUERY --->
<CFSET MyArray = ArrayNew(1)>
<CFSET MyQuery = Evaluate("#QueryName#")>
<CFSET NewColumns = "#MyQuery.ColumnList#, EXCHANGE">
<CFSET NewQuery = QueryNew(NewColumns)>
```

```
<!--- ADD ROWNUMBER TO END OF EACH ROW'S VALUE --->
<CFOUTPUT QUERY="MyQuery">
<CFSET MyArray[CurrentRow] = NumberFormat(CurrentRow, "000009")>
<CFSET Temp = QueryAddRow(NewQuery)>
</CFOUTPUT>
```

```
<!--- POPULATE THE NEW QUERY WITH THE INFO FROM THE OLD ONE, BUT WITH
ALL QUOTES REMOVED --->
```

```
<CFLOOP FROM=1 TO=#MyQuery.RecordCount# INDEX="This">
<CFSET Row = Val(Right(MyArray[This], 6))>
<CFLOOP LIST="#MyQuery.ColumnList#" INDEX="Col">

<CFIF RIGHT(Evaluate("MyQuery.Symbol[Row]"),3) IS ".M"">
<CFSET Exch="Montreal">
<CFELSEIF RIGHT(Evaluate("MyQuery.Symbol[Row]"),3) IS ".V"">
```

```

<CFSET Exch="Vancouver">
<CFELSEIF RIGHT(Evaluate("MyQuery.Symbol[Row]"),4) IS ".TO"">
<CFSET Exch="Toronto">
<CFELSEIF RIGHT(Evaluate("MyQuery.Symbol[Row]"),4) IS ".AL"">
<CFSET Exch="Alberta">
<CFELSE>
<CFSET Exch="US">
</CFIF>

<CFSET Temp = QuerySetCell(NewQuery, Col,
Replace(Evaluate("MyQuery.#Col#[Row]"),"",",","All"), This)>
</CFLOOP>
<CFSET Temp = QuerySetCell(NewQuery, "EXCHANGE", Exch, this)>
</CFLOOP>

```

```

<!-- PASS QUERY WITH QUOTATION MARKS REMOVED BACK TO CALLING TEMPLATE
-->

```

```

<CFSET "Caller.#QueryName#" = NewQuery>

```

SRWINTAB.CFM

```

<CFIF ThisTag.ExecutionMode IS "Start">

```

```

<CFINCLUDE TEMPLATE="getProgramDefaults.cfm">

```

```

<CFPARAM NAME="ATTRIBUTES.tabs"                DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.tabSelected"         DEFAULT="1">
<CFPARAM NAME="ATTRIBUTES.tabURL"              DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.baseURL"            DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.system_toolBar"     DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.system_toolBarLinks" DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.system_toolBarAddid" DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.toolBar"            DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.toolBarLinks"       DEFAULT="">

```

```

<TABLE BORDER=0 CELSPACING=0 CELLPADDING=0>
<TR>

```

```

<CFSET counter = 0>
<CFLOOP LIST="#ATTRIBUTES.tabs#" INDEX="element">

```

```

<CFSET counter = counter + 1>
<!-- show beginning tab -->
<CFIF counter EQ 1>

```

```

<CFOUTPUT>
<TD CLASS="#Iif( counter EQ ATTRIBUTES.tabSelected,De( 'winTabSelected'
),De( 'winTabUnselected' ) )#" ALIGN="right" VALIGN="bottom"><IMG
SRC="#VARIABLES.gvIMAGE_DIRECTORY#tab_left_side.gif" BORDER="0"></TD>
<TD CLASS="#Iif( counter EQ ATTRIBUTES.tabSelected,De( 'winTabSelected'
),De( 'winTabUnselected' ) )#" ALIGN="right" VALIGN="middle">
<CFIF #ListLen( ATTRIBUTES.tabs )# NEQ 1>
<A CLASS="mainTab" HREF="#ListGetAt( ATTRIBUTES.tabURL,counter
)#">#element#</A>
<CFELSE>
#element#

```

```

</CFIF>
</TD>
<CFIF counter EQ #ListLen( ATTRIBUTES.tabs )#>
<TD CLASS="#Iif( counter EQ ATTRIBUTES.tabSelected,De( 'winTabSelected'
),De( 'winTabUnselected' ) )#" ALIGN="left" VALIGN="bottom"><IMG
SRC="#Iif( URL.aID EQ 1,De( VARIABLES.gvIMAGE_DIRECTORY &
'tab_right_select.gif' ),De( VARIABLES.gvIMAGE_DIRECTORY &
'tab_last_right_unselect.gif' ) )#" BORDER="0"></TD>
<CFELSE>
<TD CLASS="#Iif( counter EQ ATTRIBUTES.tabSelected,De( 'winTabSelected'
),De( 'winTabUnselected' ) )#" ALIGN="left" VALIGN="bottom"><IMG
SRC="#Iif( URL.aID EQ 1,De( VARIABLES.gvIMAGE_DIRECTORY &
'tab_right_select.gif' ),De( VARIABLES.gvIMAGE_DIRECTORY &
'tab_right_unselect.gif' ) )#" BORDER="0"></TD>
</CFIF>
</CFOUTPUT>

<!-- show ending tab --->
<CFELSEIF counter EQ #ListLen( ATTRIBUTES.tabs )#>

<CFOUTPUT>
<TD CLASS="#Iif( counter EQ ATTRIBUTES.tabSelected,De( 'winTabSelected'
),De( 'winTabUnselected' ) )#" ALIGN="right" VALIGN="bottom"><IMG
SRC="#VARIABLES.gvIMAGE_DIRECTORY#tab_left_side.gif" BORDER="0"></TD>
<TD CLASS="#Iif( counter EQ ATTRIBUTES.tabSelected,De( 'winTabSelected'
),De( 'winTabUnselected' ) )#" ALIGN="right" VALIGN="middle">
<CFIF #ListLen( ATTRIBUTES.tabs )# NEQ 1>
<A CLASS="mainTab" HREF="#ListGetAt( ATTRIBUTES.tabURL,counter
)#">#element#</A>
<CFELSE>
#element#
</CFIF>
</TD>
<TD CLASS="#Iif( counter EQ ATTRIBUTES.tabSelected,De( 'winTabSelected'
),De( 'winTabUnselected' ) )#" ALIGN="left" VALIGN="bottom"><IMG
SRC="#VARIABLES.gvIMAGE_DIRECTORY#tab_last_right_unselect.gif"
BORDER="0"></TD>
</CFOUTPUT>

<!-- show middle tab --->
<CFELSE>

<CFOUTPUT>
<TD CLASS="#Iif( counter EQ ATTRIBUTES.tabSelected,De( 'winTabSelected'
),De( 'winTabUnselected' ) )#" ALIGN="right" VALIGN="bottom"><IMG
SRC="#VARIABLES.gvIMAGE_DIRECTORY#tab_left_side.gif" BORDER="0"></TD>
<TD CLASS="#Iif( counter EQ ATTRIBUTES.tabSelected,De( 'winTabSelected'
),De( 'winTabUnselected' ) )#" ALIGN="right" VALIGN="middle">
<CFIF #ListLen( ATTRIBUTES.tabs )# NEQ 1>
<A CLASS="mainTab" HREF="#ListGetAt( ATTRIBUTES.tabURL,counter
)#">#element#</A>
<CFELSE>
#element#
</CFIF>
</TD>
<TD CLASS="#Iif( counter EQ ATTRIBUTES.tabSelected,De( 'winTabSelected'
),De( 'winTabUnselected' ) )#" ALIGN="left" VALIGN="bottom"><IMG

```

```

SRC="#Iif( URL.aID EQ counter,De( VARIABLES.gvIMAGE_DIRECTORY &
'tab_right_select.gif' ),De( VARIABLES.gvIMAGE_DIRECTORY &
'tab_right_unselect.gif' ) )#" BORDER="0"></TD>
</CFOUTPUT>

```

```

</CFIF>

```

```

</CFLOOP>

```

```

</TR>

```

```

</TABLE>

```

```

<CFOUTPUT>

```

```

<TABLE CLASS="winTab" WIDTH=100% CELLPACING=0 CELLPADDING=0>

```

```

<TR CLASS="winTab" >

```

```

<TD ALIGN="left">

```

```

<TABLE BORDER=0 CELLPADDING=1 CELLPACING=1>

```

```

<TR>

```

```

<TD>

```

```

<TABLE BORDER=0 CELLPADDING=0 CELLPACING=0>

```

```

<TR>

```

```

<CFLOOP INDEX="Counter" FROM="1" TO="#ListLen(
ATTRIBUTES.system_toolBar )#">

```

```

<TD ALIGN="center">

```

```

&nbsp;&nbsp;&nbsp;<A CLASS="toolBar" HREF="#ListGetAt(
ATTRIBUTES.system_toolBarLinks,Counter )##Iif( ATTRIBUTES.system_addid
NEQ '',De( '&' & ATTRIBUTES.system_addid ),De( '' ) )#">#ListGetAt(
ATTRIBUTES.system_toolBar,Counter )#</A>

```

```

</TD>

```

```

</CFLOOP>

```

```

</TR>

```

```

</TABLE>

```

```

</TD>

```

```

</TR>

```

```

</TABLE>

```

```

</TD>

```

```

<TD ALIGN="right">

```

```

<TABLE BORDER=0 CELLPADDING=1 CELLPACING=1>

```

```

<TR>

```

```

<TD>

```

```

<TABLE BORDER=0 CELLPADDING=0 CELLPACING=0>

```

```

<TR>

```

```

<CFLOOP INDEX="Counter" FROM="1" TO="#ListLen( ATTRIBUTES.toolBar )#">

```

```

<TD ALIGN="center">

```

```

&nbsp;&nbsp;&nbsp;<A CLASS="toolBar" HREF="#ListGetAt(
ATTRIBUTES.toolBarLinks,Counter )#">#ListGetAt(
ATTRIBUTES.toolBar,Counter )#</A>

```

```

</TD>

```

```

</CFLOOP>

```

```

</TR>

```

```

</TABLE>

```

```

</TD>

```

```

</TR>

```

```

</TABLE>

```

```

</TD>

```

```

</TR>

<TR>
<TD COLSPAN=2>
</CFOUTPUT>

<CFELSEIF ThisTag.ExecutionMode IS "End">

<CFOUTPUT>
</TD>
</TR>
<TR CLASS="winTab" >

<TD ALIGN="left">
<TABLE BORDER=0 CELLPADDING=1 CELLSPACING=1>
<TR>
<TD>
<TABLE BORDER=0 CELLPADDING=0 CELLSPACING=0>
<TR>
<CFLOOP INDEX="Counter" FROM="1" TO="#ListLen(
ATTRIBUTES.system_toolBar )#">
<TD ALIGN="center">
    <A CLASS="toolBar" HREF="#ListGetAt(
ATTRIBUTES.system_toolBarLinks,Counter )#">#ListGetAt(
ATTRIBUTES.system_toolBar,Counter )#</A>
</TD>
</CFLOOP>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>
</TD>

<TD ALIGN="right">
<TABLE BORDER=0 CELLPADDING=1 CELLSPACING=1>
<TR>
<TD>
<TABLE BORDER=0 CELLPADDING=0 CELLSPACING=0>
<TR>
<CFLOOP INDEX="Counter" FROM="1" TO="#ListLen( ATTRIBUTES.toolBar )#">
<TD ALIGN="center">
    <A CLASS="toolBar" HREF="#ListGetAt(
ATTRIBUTES.toolBarLinks,Counter )#">#ListGetAt(
ATTRIBUTES.toolBar,Counter )#</A>
</TD>
</CFLOOP>
</TR>
</TABLE>
</TD>
</TR>
</TABLE>
</TD>

</TR>
</TABLE>

```

</CFOUTPUT>

</CFIF>

SRHELPSUMMARY.CFM

```
CFPARAM NAME="ATTRIBUTES.id"           DEFAULT="-1">
<CFPARAM NAME="ATTRIBUTES.shortName"   DEFAULT="">
<CFPARAM NAME="ATTRIBUTES.Page"        DEFAULT="">
```

```
<CFINCLUDE TEMPLATE="getProgramDefaults.cfm">
```

```
<CFIF IsDefined( "FORM.summary_id" )>
```

```
<CFIF NOT CompareNoCase( FORM.summary_id, "" )>
```

```
<CFINSERT DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#"
TABLENAME="tblSummary">
```

```
<CFELSE>
```

```
<CFUPDATE DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#"
TABLENAME="tblSummary">
```

```
</CFIF>
```

```
</CFIF>
```

```
<CFQUERY NAME="getSummary"
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG">
```

```
SELECT      *
FROM        tblSummary
WHERE       0=0 AND
<CFIF #ATTRIBUTES.shortName# NEQ "">
shortName  = '#ATTRIBUTES.shortName#'
<CFELSEIF #ATTRIBUTES.id# NEQ -1>
summary_id = '#ATTRIBUTES.id#'
</CFIF>
```

```
</CFQUERY>
```

```
<CFSET textToDisplay = #getSummary.summary#>
```

```
<CFIF IsDefined( "SESSION.aem" )>
```

```
<STYLE TYPE="text/css">
.summaryEdit
{
font: 8pt Tahoma, Verdana, Arial;
background-color: #FFFFFF;
border: 1px solid black;
}
```

```
A.summaryOption:link
```

```

{
color: black;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.summaryOption:active
{
color: black;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.summaryOption:visited
{
color: black;
text-decoration: none;
font-size: 8pt;
font-weight: bold;
}

A.summaryOption:hover
{
color: black;
text-decoration: underline;
font-size: 8pt;
}
}

</STYLE>

<CFIF #SESSION.aem#>

<CFOUTPUT>
<FORM NAME="aem_helpSummary" ACTION="?#CGI.query_string#"
METHOD="post">
<INPUT TYPE="hidden" NAME="summary_id" VALUE="#getSummary.summary_id#">
<INPUT TYPE="hidden" NAME="shortName" VALUE="#ATTRIBUTES.shortName#">
<TABLE BORDER=0>
<TR>
<TD CLASS="winTabLabel">&nbsp;Help Summary (#Iif( ATTRIBUTES.Page NEQ
"",De( ATTRIBUTES.Page & '/' ),De( '' ) )##ATTRIBUTES.shortName#);</TD>
</TR>
<TR>
<TD CLASS="winTabField">
<TEXTAREA CLASS="summaryEdit" COLS=70 ROWS=5 NAME="summary"
WRAP="soft">#getSummary.summary#</TEXTAREA>
</TD>
</TR>
<TR>
<TD CLASS="winTabField" ALIGN="center">
<A CLASS="summaryOption"
HREF="javascript:document.aem_helpSummary.submit()">Save</A>&nbsp;&nbsp;&nbsp;
<A CLASS="summaryOption"
HREF="javascript:document.aem_helpSummary.reset()">Undo</A>
</TD>

```

```

</TR>
</TABLE>
</FORM>
</CFOUTPUT>

</CFIF>

<CFELSE>

<STYLE TYPE="text/css">

TD.summaryText
{
font: 10pt Tahoma, Verdana, Arial;
color: black;
}

</STYLE>

<TABLE BORDER=0>
<TR>
<TD CLASS="summaryText">
<CFOUTPUT>
#VARIABLES.textToDisplay#
</CFOUTPUT>
</TD>
</TR>
</TABLE>

</CFIF>

```

11. INFRASTRUCTURE FILES

SUBHEADER.CFM

```

<TABLE BORDER="0" WIDTH=100% CELLPADDING=0 CELLSPACING=0>

<!-- MAIN LOGO --->
<TR VALIGN="bottom">
<TD WIDTH=50%>
<CF_getContent
TYPE="#aComponent[URL.aID][cHEADER_LEFT][cCOMPONENT_TYPE]#"
FILE="#aComponent[URL.aID][cHEADER_LEFT][cCOMPONENT_FILE]#"
URL="#aComponent[URL.aID][cHEADER_LEFT][cCOMPONENT_URL]#"
INLINE="#aComponent[URL.aID][cHEADER_LEFT][cCOMPONENT_INLINECODE]#"
>
</TD>
<TD WIDTH=50%>
<CF_getContent
TYPE="#aComponent[URL.aID][cHEADER_RIGHT][cCOMPONENT_TYPE]#"
FILE="#aComponent[URL.aID][cHEADER_RIGHT][cCOMPONENT_FILE]#"
URL="#aComponent[URL.aID][cHEADER_RIGHT][cCOMPONENT_URL]#"
INLINE="#aComponent[URL.aID][cHEADER_RIGHT][cCOMPONENT_INLINECODE]#"
>
</TD>
</TR>

```



```
</TR>
```

```
</TABLE>
```

SUBFOOTER.CFM

```
<!--- FORMAT LEFT PANEL CONTENTS --->
<TABLE WIDTH="100%" BORDER="0" CELLPADDING="0" CELLSPACING="0">
<TR>
<TD CLASS="footer">
<CF_getContent
TYPE="#aComponent[URL.aID][cFOOTER][cCOMPONENT_TYPE]#"
FILE="#aComponent[URL.aID][cFOOTER][cCOMPONENT_FILE]#"
URL="#aComponent[URL.aID][cFOOTER][cCOMPONENT_URL]#"
INLINE="#aComponent[URL.aID][cFOOTER][cCOMPONENT_INLINECODE]#"
>
</TD>
</TR>
<!--- END FORMAT LEFT PANEL CONTENTS --->
</TABLE>
```

SUBBODY.CFM

```
<!--- ACTUAL WORK AREA --->
<TABLE BORDER="0" WIDTH="100%" CELLSPACING="3" CELLPADDING="0">
<TR>
<TD>
<CFIF IsDefined( "URL.body" )>

<CFIF NOT FileExists( "#GetDirectoryFromPath( GetTemplatePath()
)#\#URL.body#" )>
File Not Found!
<CFELSE>
<CFINCLUDE TEMPLATE="#URL.body#">
</CFIF>

<CFELSEIF IsDefined( "URL.www" )>

<CFHTTP URL="#URL.www#" RESOLVEURL="yes">
<CFOUTPUT>
#CFHTTP.FileContent#
</CFOUTPUT>

<CFELSE>

<CF_getContent
TYPE="#aComponent[URL.aID][cBODY][cCOMPONENT_TYPE]#"
FILE="#aComponent[URL.aID][cBODY][cCOMPONENT_FILE]#"
URL="#aComponent[URL.aID][cBODY][cCOMPONENT_URL]#"
INLINE="#aComponent[URL.aID][cBODY][cCOMPONENT_INLINECODE]#"
>

</CFIF>
</TD>
```

</TR>

<!-- END ACTUAL WORK AREA -->
</TABLE>

COMPONENT_DECLARE.CFM

<CFSCRIPT>

```
cCONFIG_FILE = GetDirectoryFromPath( GetTemplatePath() ) &
"component.cfg";
```

```
cHEADER_RIGHT = 1;
cHEADER_LEFT = 2;
cTABS = 3;
cPANEL = 4;
cTOOLBAR = 5;
cBODY = 6;
cFOOTER = 7;
cAPPLICATION = 8;
```

```
cCOMPONENT_NAME = 1;
cCOMPONENT_TYPE = 2;
cCOMPONENT_INLINECODE = 3;
cCOMPONENT_FILE = 4;
cCOMPONENT_URL = 5;
cCOMPONENT_LOCKED = 6;
cCOMPONENT_LOCKEDBY = 7;
```

```
cAPPLICATION_NAME = 1;
cAPPLICATION_DESCRIPTION = 2;
cAPPLICATION_DATECREATED = 3;
cAPPLICATION_CREATEDBY = 4;
cAPPLICATION_LASTUPDATED = 5;
cAPPLICATION_UPDATEDBY = 6;
cAPPLICATION_CARGO1 = 7;
```

</CFSCRIPT>

COMPONENT_PREPARE.CFM

```
<CFFILE
ACTION="read"
FILE="#GetDirectoryFromPath( GetTemplatePath() )#component.cfg"
VARIABLE="DataPacket"
>
```

```
<CFWDDX
INPUT="#DataPacket#"
OUTPUT="aComponent"
ACTION="WDDX2CFML"
>
```

APPLICATION.CFM

```
<!--- INCLUDE DEFAULTS JUST IN CASE USER DEFINED CONFIG FILE NOT FOUND
--->
<CFINCLUDE TEMPLATE="getProgramDefaults.cfm">

<!--- APPLICATION SPECIFIC INFORMATION --->
<CFAPPLICATION NAME="ReadinessExplorer" SESSIONMANAGEMENT="yes"
CLIENTMANAGEMENT="YES" CLIENTSTORAGE="reUSERS">

<CFIF IsDefined( "URL.aem" )>
<CFIF #URL.aem#>
<CFSET SESSION.aem = true>
<CFELSE>
<CFSET StructDelete(SESSION, "aem")>
</CFIF>
</CFIF>

<!--- look for these two variables in order start login process over --
-->
<CFIF ( Not( IsDefined( "SESSION.InitializeThis" ) ) OR IsDefined(
"FORM.xloginid" ) )>

<!--- did user come in from the login form?, If so, grab all data about
them --->
<CFIF IsDefined( "FORM.xLogInID" ) AND IsDefined( "FORM.xPassword" )>

<CFQUERY NAME="GetUserRecord"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      *
FROM        tblUserAccounts
WHERE       user_loginid = '#FORM.xLogInID#' AND
user_Password = '#FORM.xPassword#'

</CFQUERY>

<!--- if record was found and not locked, then get and set some data
about them --->
<CFSET flagUserOK = false>
<CFIF GetUserRecord.RecordCount NEQ 0>

<CFIF #VARIABLES.gvPSWD_ISCASE#>

<CFIF ( #Compare( GetUserRecord.user_loginid,FORM.xLogInID )# EQ 0 )
AND
( #Compare( GetUserRecord.user_password,FORM.xPassword )# EQ 0 )>

<CFSET flagUserOK = true>

</CFIF>

<CFELSE>

<CFSET flagUserOK = true>
```

```

</CFIF>

</CFIF>

<CFIF #VARIABLES.flagUserOK#>

<CFIF NOT #GetUserRecord.locked#>

<CFOUTPUT QUERY="GetUserRecord">

<CFSET SESSION.InitializeThis = Now()>
<CFSET SESSION.loginid = #user_loginid#>
<CFSET SESSION.access = #access#>
<CFSET SESSION.user_id = #user_id#>
<CFSET SESSION.LastLogOn = #Last_Logon#>
<CFSET SESSION.FirstOn = TRUE>
<CFSET SESSION.LogOnAttempts = 1>
<CFSET SESSION.LogSequence = 0>
<CFSET SESSION.showPanel = #panelStatus#>
<CFSET SESSION.deskTop = #deskTop#>

<CFSET SESSION.accountExpires = #expire#>

<!-- remember to add client variables for such things as desktop, etc.
--->
<CFIF IsDefined( "CLIENT.CurrIP" )>
<CFSET CLIENT.LastIP = #CLIENT.CurrIP#>
<CFELSE>
<CFSET CLIENT.LastIP = "First Time On">
</CFIF>
<CFSET CLIENT.CurrIP = #CGI.REMOTE_ADDR#>

</CFOUTPUT>

<!-- ***** NEED TO ADD SECTION FOR mySECURITY REMEMBER LOGIN
***** --->
<CFQUERY NAME="UpdateLogon" DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

UPDATE      tblUserAccounts
SET  last_logon = #Now()#, login_attempts = 0
WHERE      user_loginid = '#SESSION.loginid#'

</CFQUERY>

<CFQUERY NAME="UpdateSecurity"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

SELECT      Count(*) AS mySecurityOk
FROM  tblMySecurity
WHERE      user_id = #SESSION.user_id#

</CFQUERY>
<CFIF #UpdateSecurity.mySecurityOk# NEQ 0>

<CFQUERY NAME="UpdateSecurity"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

```

```

UPDATE      tblmySecurity
SET   rememberLogin = <CFIF IsDefined( "FORM.xRememberLogin"
)>true<CFELSE>>false</CFIF>
WHERE      user_id = #SESSION.user_id#

</CFQUERY>

</CFIF>

<!--- check for cookie requirements and update --->
<CFIF #VARIABLES.gvPSWD_REMEMBER#>

<CFIF IsDefined( "FORM.xRememberLogin" )>

<CFIF #FORM.xRememberLogin#>

<SCRIPT LANGUAGE="javascript" SRC="cookies.js"></SCRIPT>
<SCRIPT LANGUAGE="javascript">
<CFOUTPUT>
<!--- set expiration to 30 days for now, this will be an option for
ADMIN --->
saveCookie( 'cookieLogin', '#FORM.xLogInID#', 30);
saveCookie( 'cookiePassword', '#FORM.xPassword#', 30);
saveCookie( 'cookieRemember', '#FORM.xRememberLogin#', 30);
</CFOUTPUT>
</SCRIPT>

</CFIF>

<CFELSE>

<!--- delete cookies just in case --->
<SCRIPT LANGUAGE="javascript" SRC="cookies.js"></SCRIPT>
<SCRIPT LANGUAGE="javascript">
deleteCookie( 'cookieLogin' );
deleteCookie( 'cookiePassword' );
deleteCookie( 'cookieRemember' );
</SCRIPT>

</CFIF>

</CFIF>

<CFELSE>

<!--- sorry, record is locked, so don't let them in --->
<CFABORT SHOWERROR="Deny!">

</CFIF>

<!--- DONE WITH FIRST TIME LOG IN --->

<CFELSE>

<!--- CHECK TO SEE IF login ID IS REAL, AND IF SO ADJUST ATTEMPTS --->
<CFQUERY NAME="GetLogInRecord"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

```

```

SELECT      *
FROM        tblUserAccounts
WHERE       user_loginid = '#FORM.xLogInID#'

</CFQUERY>

<!-- IF LOGIN ID IS FOUND, UPDATE USER'S RECORD WITH AN ATTEMPT ---->
<CFIF GetLogInRecord.RecordCount IS NOT 0>

<CFQUERY NAME="UpdateAttempts"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

UPDATE      tblUserAccounts
SET  login_attempts = login_attempts + 1
WHERE       user_loginid = '#FORM.xLogInID#'

</CFQUERY>

<!-- IF ATTEMPTS REACHES NUMBER, LOCK RECORD IF AVAILABLE. NOTE: ADD 1
BECAUSE OF UPDATE ABOVE NOT IN QUERY ---->
<CFIF ( #GetLogInRecord.login_attempts# + 1 ) GTE
#VARIABLES.gvLOGON_ATTEMPTS#>

<!-- SORRY, REACHED THE LOGIN ATTEMPTS LIMIT, SO LOCK RECORD ---->
<CFQUERY NAME="UpdateAttempts"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

UPDATE      tblUserAccounts
SET  locked = true
WHERE       user_loginid = '#FORM.xLogInID#'

</CFQUERY>

<!-- sorry, record is locked, so don't let them in ---->
<CFABORT SHOWERROR="Deny!">

</CFIF>

</CFIF>

<!-- sorry, record is locked, so don't let them in ---->
<CFABORT SHOWERROR="Deny!">

</CFIF>

<CFELSE>

<CFLOCATION URL="index.htm">

</CFIF>

<CFELSE>

<CFSET SESSION.InitializeThis = Now()>
<CFSET SESSION.FirstOn = FALSE>

```

```
<CFSET SESSION.LogOnAttempts = 1>
```

```
</CFIF>
```

INDEX.CFG

```
<HTML>
```

```
<HEAD>
```

```
<TITLE>Welcome to Readiness Explorer</TITLE>
```

```
<STYLE TYPE='text/css'>
```

```
/******
```

```
LOG-IN PROPERTIES
```

```
******/
```

```
TD.Login
```

```
{
```

```
font: 12px tahoma,verdana,arial;
```

```
color: black;
```

```
font-weight: bold;
```

```
}
```

```
TD.LoginFooter
```

```
{
```

```
font: 12px tahoma,verdana,arial;
```

```
color: black;
```

```
}
```

```
A.LogIn:link
```

```
{
```

```
color: black;
```

```
text-decoration: none;
```

```
font: 10pt Tahoma, Verdana, Arial;
```

```
font-weight: bold;
```

```
}
```

```
A.LogIn:active
```

```
{
```

```
color: black;
```

```
text-decoration: none;
```

```
font: 10pt Tahoma, Verdana, Arial;
```

```
font-weight: bold;
```

```
}
```

```
A.LogIn:visited
```

```
{
```

```
color: black;
```

```
text-decoration: none;
```

```
font: 10pt Tahoma, Verdana, Arial;
```

```
font-weight: bold;
```

```
}
```

```
A.LogIn:hover
```

```
{
```

```
color: blue;
```

```
text-decoration: underline;
```

```
font: 10pt Tahoma, Verdana, Arial;
```

```
font-weight: bold;
```

```
}
```

```
SPAN.LogIn
```

```

{
color: black;
text-decoration: none;
font: 10pt Tahoma, Verdana, Arial;
font-weight: bold;
}
/*****
END LOG-IN PROPERTIES
*****/
</STYLE>

<SCRIPT LANGUAGE='JavaScript' SRC='cookies.js'></SCRIPT>
<SCRIPT LANGUAGE='javascript'>
function init()
{
var xL = readCookie('cookieLogin');
var xP = readCookie('cookiePassword');
var xR = readCookie('cookieRemember');
if( xL != null )
document.logonform.xLogInID.value = xL;
if( xP != null )
document.logonform.xPassword.value = xP;
if( xR != null )
document.logonform.xRememberLogin.checked = xR;

document.logonform.xLogInID.focus();
}
</SCRIPT>

</HEAD>

<BODY onLOAD='init()'>

<BR>
<BR>
<BR>
<BR>

<FORM ACTION='main.cfm' METHOD=POST NAME='logonform'>

<CENTER>
<TABLE BORDER=0 CELLPADDING=0 CELLSPACING=0>
<TR>
<TD ALIGN='left'>

<TABLE BORDER=0 CELLSPACING=0 CELLPADDING=0 BORDERCOLOR='#A8B0D8'>
<TR BGCOLOR='#A8B0D8'>
<TD ALIGN='left' WIDTH='23'><IMG SRC='images/top_left_corner.gif'
WIDTH='23' HEIGHT='23' BORDER='0'></TD>
<TD ALIGN='left' CLASS='Login'>WELECOME TO READINESS EXPLORER</TD>
<TD ALIGN='right' WIDTH='23'><IMG SRC='images/top_right_corner.gif'
WIDTH='23' HEIGHT='23' BORDER='0'></TD>
</TR>
</TABLE>

</TD>
</TR>

```



```

<TD ALIGN='center' CLASS='LoginFooter'>
<BR>
You have entered a Department of Defense (DoD) web site<BR>
maintained by Michael Whitecars Test Site. This site is monitored and
records all activity.
</TD>
</TR>
</TABLE>
</CENTER>
</FORM>
</BODY>
</HTML>

```

MAIN.CFM

```

<CFTRY>

<!-- get the XML structures in order to build pages -->
<CFINCLUDE TEMPLATE="getStructure.cfm">

<!-- define default variables -->
<CFPARAM NAME="URL.workOnLine" DEFAULT="False">

<CFIF IsDefined( "URL.a" )>
<CFLOOP INDEX="counter" FROM="1" TO="#ArrayLen( aComponent )#">
<CFIF aComponent[counter][cAPPLICATION][cAPPLICATION_NAME] EQ #URL.a#>
<CFSET URL.aID = counter>
<CFBREAK>
</CFIF>
</CFLOOP>
<CFELSE>
<CFPARAM NAME="URL.aID" DEFAULT="1">
</CFIF>

<!--
Store the status of show the panel:
If a URL is passed, i.e. when the appropriate arrows are clicked,
then set a SESSION variable to reflect URL variable, else check
to see if SESSION variable has been set and if not then make true.
-->
<CFIF IsDefined( "URL.showPanel" )>

<CFSET SESSION.showPanel = URL.showPanel>

<CFIF IsDefined( "URL.updateUser" )>

<CFIF URL.updateUser EQ "panel">

<CFQUERY NAME="UpdatePanelStatus"
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">

UPDATE      tblUserAccounts
SET  panelStatus = #URL.showPanel#
WHERE      user_loginid = '#SESSION.loginid#'

</CFQUERY>

```

```
</CFIF>
```

```
</CFIF>
```

```
</CFIF>
```

```
<CFSET ColSpan = Iif( NOT SESSION.showPanel,2,1 )>
```

```
<CFINCLUDE TEMPLATE="getPageComments.cfm">
```

```
<HTML>
```

```
<HEAD>
```

```
<TITLE><CFOUTPUT>#aComponent[URL.aID] [cAPPLICATION] [cAPPLICATION_DESCRIP  
TION]#</CFOUTPUT></TITLE>
```

```
<STYLE TYPE="text/css">
```

```
<CFINCLUDE TEMPLATE="getStyle.cfm">
```

```
</STYLE>
```

```
<script language="JavaScript" src="pupdate.js">
```

```
/*
```

```
Popup calendar script by Sev Kotchnev (webmaster@personal-  
connections.com)
```

```
For full source code and installation instructions to this script
```

```
Visit http://www.dynamicdrive.com
```

```
*/
```

```
</script>
```

```
<SCRIPT LANGUAGE="javascript">
```

```
<CFINCLUDE TEMPLATE="re2001.js">
```

```
<CFINCLUDE TEMPLATE="ftiens4.js">
```

```
</SCRIPT>
```

```
<CFQUERY NAME="getBookmarks"
```

```
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">
```

```
SELECT *
```

```
FROM tblMyBookmarks
```

```
WHERE user_id = #SESSION.user_id#
```

```
ORDER BY bmLabel
```

```
</CFQUERY>
```

```
<CFQUERY NAME="getAddressBook"
```

```
DATASOURCE="#VARIABLES.gvDATASOURCE_USER#">
```

```
SELECT *
```

```
FROM tblMyAddressBook
```

```
WHERE user_id = #SESSION.user_id#
```

```
ORDER BY cLname
```

```
</CFQUERY>
```

```
<CFQUERY NAME="getmyAccount"
```

```
DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">
```

```
SELECT poi_url,poi_title
```

```
FROM ( tblLinkOptionsToPage INNER JOIN tblPageOptionItems ON
```

```
tblLinkOptionsToPage.poi_id = tblPageOptionItems.poi_id ) INNER JOIN
```

```
tblPageOptions ON tblLinkOptionsToPage.po_id = tblPageOptions.po_id
```

```

WHERE      tblPageOptions.po_name = 'myAccount'

</CFQUERY>
<CFQUERY NAME="getAdmin" DATASOURCE="#VARIABLES.gvDATASOURCE_CONFIG#">

SELECT      poi_url,poi_title
FROM      ( tblLinkOptionsToPage INNER JOIN tblPageOptionItems ON
tblLinkOptionsToPage.poi_id = tblPageOptionItems.poi_id ) INNER JOIN
tblPageOptions ON tblLinkOptionsToPage.po_id = tblPageOptions.po_id
WHERE      tblPageOptions.po_name = 'Admin'

</CFQUERY>

</HEAD>

<BODY CLASS="Home">

<script language=JavaScript>
<!--

/*
Disable right mouse click Script (By Crash @ http://walk.to/crash)
Submitted to and permission granted to Dynamicdrive.com to feature
script in it's archive
For full source code to this script and 100's more, visit
http://dynamicdrive.com
*/

var message="Right Mouse button disabled.";
function click(e) {
if (document.all) {
if (event.button==2||event.button==3) {
alert(message);
return false;
}
}
else if (document.layers||document.getElementById) {
if (e.which == 3) {
alert(message);
return false;
}
}
}
if (document.layers) {
document.captureEvents(Event.MOUSEDOWN);
}
document.onmousedown=click;
// -->
</script>

<!-- show header --->
<CFINCLUDE TEMPLATE="subHeader.cfm">

<!-- show tabs with HOME active --->
<CFINCLUDE TEMPLATE="subTabs.cfm">

<TABLE BORDER=0 WIDTH=100% CELLPADDING=0 CELLSPACING=0>

```

```

<!-- DISPLAY AREA -->
<TR>
<TD COLSPAN=2>

<!-- SET UP DISPLAY AREA -->
<TABLE BORDER=0 WIDTH=100% CELLPADDING=0 CELLSPACING=0>

<!-- TOP EMPTY ROW -->
<FORM NAME="quickLinks" ACTION="">
<TR>
<TD CLASS="mainTabSelected" COLSPAN="5" WIDTH="100%" ALIGN="right">
<SELECT CLASS="quickList" NAME="myBookmarks" SIZE="1"
ONCHANGE="quickList( this )">
<OPTION VALUE="">Select a bookmark...
<CFOUTPUT QUERY="getBookmarks">
<CFIF #newWindow# EQ "1">
<OPTION VALUE="main.cfm?a=myAccount&www=#bmURL#">#bmLabel#
<CFELSEIF #newWindow# EQ "2">
<OPTION VALUE="main.cfm?a=myAccount&www=#bmURL#">#bmLabel#
<CFELSEIF #newWindow# EQ "3">
<OPTION VALUE="main.cfm?a=myAccount&body=#bmURL#">#bmLabel#
</CFIF>
</CFOUTPUT>
<OPTION VALUE="">
<OPTION
VALUE="main.cfm?a=myAccount&body=myBookmarks.cfm&action=add">New
Bookmark...
<OPTION VALUE="main.cfm?a=myAccount&body=myBookmarks.cfm">Edit
Bookmarks...
</SELECT>
<SELECT CLASS="quickList" NAME="myAddressBook" SIZE="1"
ONCHANGE="quickList( this )">
<OPTION VALUE="">Select an address...
<CFOUTPUT QUERY="getAddressBook">
<OPTION VALUE="mailto:#cEmailAddress#">#cLname#, #cFname# #cRank#
</CFOUTPUT>
<OPTION VALUE="">
<OPTION
VALUE="main.cfm?a=myAccount&body=myAddressBook.cfm&action=add">New
Address...
<OPTION VALUE="main.cfm?a=myAccount&body=myAddressBook.cfm">Edit
Address Book...
</SELECT>
<SELECT CLASS="quickList" NAME="myAccount" SIZE="1"
ONCHANGE="quickList( this )">
<OPTION VALUE="">Select from your account...
<CFOUTPUT QUERY="getmyAccount">
<OPTION VALUE="#poi_url#">#poi_title#
</CFOUTPUT>
</SELECT>
<SELECT CLASS="quickList" NAME="adminList" SIZE="1"
ONCHANGE="quickList( this )">
<OPTION VALUE="">Select administration task...
<CFOUTPUT QUERY="getAdmin">
<OPTION VALUE="#poi_url#">#poi_title#
</CFOUTPUT>

```

```

</SELECT>
</TD>
</TR>
</FORM>

<!-- SECOND ROW --->
<TR>

<!-- include panel if it is on --->
<CFIF SESSION.showPanel>

<!-- LEFT PAD --->
<TD CLASS="mainTabSelected" HEIGHT="30" WIDTH="1%">&nbsp;&nbsp;&nbsp;</TD>

<!-- LEFT PANEL (REMEMBER THIS WILL BE AN OPTION) --->
<TD CLASS="mainTabSelected" WIDTH="9%" ALIGN="center" ROWSPAN="2"
VALIGN="top">
<CFINCLUDE TEMPLATE="subPanel.cfm">
</TD>
<!-- DIVIDER BETWEEN LEFT SIDE PANEL AND WORK AREA --->
<TD CLASS="mainTabSelected" WIDTH="1%">&nbsp;&nbsp;&nbsp;</TD>

<CFELSE>

<!-- LEFT PAD --->
<CFOUTPUT>
<TD CLASS="mainTabSelected" HEIGHT="30" WIDTH="1%" VALIGN="top"
ALIGN="center"><BR>&nbsp;&nbsp;&nbsp;<A HREF="#CGI.path_info##Replace(
CGI.query_string,'showPanel=false','showPanel=true')##Iif(
CGI.query_string DOES NOT CONTAIN 'showPanel',De( '&showPanel=true'
),De( ' ' ) )##Iif( CGI.query_string DOES NOT CONTAIN
'updateUser=panel',De( '&updateUser=panel' ),De( ' ' ) )#"><IMG
SRC="#VARIABLES.gvIMAGE_DIRECTORY#showPanel.gif" BORDER=0 ALT="Click to
Show Panel"></A>&nbsp;&nbsp;&nbsp;</TD>
</CFOUTPUT>

</CFIF>

<!-- THIS IS THE ACTUAL WORK AREA COLUMN --->
<!-- WILL NEED TO ADJUST COLSPAN FOR NO OPTIONS TO 2 --->
<CFOUTPUT>
<TD CLASS="workAreaBody" COLSPAN="#VARIABLES.colSpan#" WIDTH="98%"
VALIGN="top" HEIGHT="1" ALIGN="left">
</CFOUTPUT>

<CFINCLUDE TEMPLATE="subToolBar.cfm">

&nbsp;&nbsp;&nbsp;<BR>

<CFINCLUDE TEMPLATE="subBody.cfm">

&nbsp;&nbsp;&nbsp;<BR>

<CFINCLUDE TEMPLATE="subToolBar.cfm">

</TD>

```



```

</TD>
</TR>

<TR>
<TD COLSPAN=2><CFINCLUDE TEMPLATE="subFooter.cfm"></TD>
</TR>
</TABLE>
<CFINCLUDE TEMPLATE="srPopUpCalendar.cfm">
</BODY>
</HTML>

<CFCATCH TYPE="Database">

<CF_srERRORMONITOR
DETAIL="#CFCATCH.Detail#"
MESSAGE="#CFCATCH.Message#"
ERROR_CODE="#CFCATCH.NativeErrorCode#"
SQLSTATE="#CFCATCH.SQLState#"
TYPE="#CFCATCH.type#"
IP="#CGI.REMOTE_ADDR#"
USER="#SESSION.loginid#"
>

</CFCATCH>
<CFCATCH TYPE="Any">

<CF_srERRORMONITOR
DETAIL="#CFCATCH.Detail#"
MESSAGE="#CFCATCH.Message#"
TYPE="#CFCATCH.type#"
IP="#CGI.REMOTE_ADDR#"
USER="#SESSION.loginid#"
>

</CFCATCH>

</CFTRY>

```

LIST OF REFERENCES

1. Stuart Russell and Peter Norvig (1995), Artificial Intelligence: A Modern Approach, Engelwood Cliffs, NJ: Prentice Hall, 33.
2. *Guide to Network Resource Tools*, "Intelligent Agents," URL: <http://www.grnet.gr/gnrt/specialist/agents.html/>, (24 May 1999).
3. "Criteria for Building Interface Agents," URL: <http://ksi.cpsc.ucalgary.ca/courses/547-95/thomsona/agents.html/>, (21 Jan 2001).
4. *Dick Stenmark's Cyber Home*, "Intelligent Software Agents," URL: <http://w3.informatik.gu.se/~dixi/agent/agent.htm/>, (22 Apr 1999).
5. *Holding On to Reality*, "Introduction: Information vs. Reality," URL: <http://www.press.uchicago.edu/Misc/Chicago/066258.html/>, (1999).
6. King, James A., "Intelligent Agents: Bringing Good Things to Life," URL: http://coqui.lce.org/cedu5100/Intelligent_Agents.htm/, (1995).
7. "A Critique of Intelligent Agents" White Paper, URL: http://www.caitud.org/pub_critique.html/, (19 Jan 2001).
8. Efraim Turban and Jay E. Aronson (1998), Decision Support Systems and Intelligent Systems Fifth Edition, Upper Saddle River, NJ: Prentice Hall, 1999.
9. Peter Kent and Kent Multer (1997), Netscape JavaScript 1.2, Programmer's Reference, Mountain View, CA: Netscape Publishing Relations, 1.
10. CFML Language Reference Manual, Allaire Corporation, 1.

11. Ivkov, Kimberly, *Personalization: The Key to Building One-to-One Relationships with Customers*. HIMSS Session 102, 4 Feb 2001.
12. McAlteer, Seamus, Allard, Ken, Graves, Lucia, Gluck, Marisa, and May, Michael, *Jupiter Communications*, Proactive Personalization: Learning to Swim, Not Drown, in Consumer Data Vision Report Volume 8, (19 Aug 1999).

BIBLIOGRAPHY

1. Bigus, Joseph P., and Bigus, Jennifer, *Constructing Intelligent Agents with Java*, Wiley Computer Publishing, 1998.
2. Champeon, Steven, and Fox, David S., *Building Dynamic HTML GUIs*, M&T Books, 1999.
3. Cottingham, Marion, *Excel 2000 Developer's Handbook*, Sybex, 1999.
4. Forta, Ben, and others, *Advanced ColdFusion 4.0 Application Development*, QUE, 1999.
5. Franklin, Carl, *Visual Basic 6.0 Internet Programming*, Wiley Computer Publishing, 1999.
6. Hart-Dave, Guy, *Word 2000 Developer's Handbook*, Sybex, 1999.
7. Kent, Peter and Multer, Kent, *Official Netscape JavaScript 1.2 Programmer's Reference*, Netscape Press, 1997.
8. North, Simon and Hermans, Paul, *Teach Yourself XML in 21 Days*, Sams, 1999.
9. *Visual Basic 6 Complete*, Sybex, 1999.

THIS PAGE INTENTIONALLY LEFT BLANK

INITIAL DISTRIBUTION LIST

1. Defense Technical Information Center2
8725 John J. Kingman Road, Suite 0944
Ft. Belvoir, VA 22060-6218
2. Dudley Knox Library2
Naval Postgraduate School
411 Dyer Road
Monterey, CA 93943-5101
3. LT Michael Whitecar3
3919 Plainsfield Way
Sacramento, CA 95821