Scientific Sensor Data Manager

Version <1.4>

3/31/2013

Document Control

Approval

The Guidance Team and the Customer shall approve this document.

Document Change Control

|  |  |
| --- | --- |
| Initial Release: | 1.0 |
| Current Release: | 1.4 |
| Indicator of Last Page in Document: | ∑ |
| Date of Last Review: | 3/27/2013 |
| Date of Next Review: | 3/28/2013 |
| Target Date for Next Update: | 3/28/2013 |

Distribution List

This following list of people shall receive a copy of this document every time a new version of this document becomes available:

Guidance Team Members:

Dr. Ann Gates

Omar Ochoa

Luis Garcia

Customers: Dr. Deana Pennington

Dr. Salamah Salamah

Software Team Members:

Joseph Aguirre

Juan Rodriquez

Andre De La Cruz

Jesus Tabares

Arturo Delgado

Change Summary

The following table details changes made between versions of this document

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Modifier | Description |
| 1.0 | 3/7/2013 | Juan Rodriguez | Introduction added |
| 1.0 | 3/7/2013 | Juan Rodriguez | Types of Databases |
| 1.0 | 3/7/2013 | Joseph Aguirre | Programming languages |
| 1.0 | 3/7/2013 | Andre De La Cruz | Mobile Phones |
| 1.0 | 3/7/2013 | Jesus Tabares | Operating Systems |
| 1.0 | 3/7/2013 | Arturo Delgado | Alert Systems |
| 1.1 | 3/15/2013 | Juan Rodriguez | Modified the Introduction |
| 1.2 | 3/19/2013 | Andre De la Cruz | Modified Mobile Phone Section |
| 1.2 | 3/19/2013 | Joseph Aguirre | Modified Programing Language |
| 1.3 | 3/26/2013 | Andre De La Cruz | Added Solution 2 and 4, Modified Phone Section |
|  |  |  |  |

Table of Contents

[Document Control ii](#_Toc352172351)

[Approval ii](#_Toc352172352)

[Document Change Control ii](#_Toc352172353)

[Distribution List ii](#_Toc352172354)

[Change Summary ii](#_Toc352172355)

[1. Introduction 5](#_Toc352172356)

[1.1. Purpose of the Feasibility Report 5](#_Toc352172357)

[1.2. Justification for the Proposed System 5](#_Toc352172358)

[1.3. Requirements Definition 5](#_Toc352172359)

[1.4. Use Cases 6](#_Toc352172360)

[1.4.1. Actors (descriptions) 6](#_Toc352172361)

[1.4.2. Use Case Descriptions 7](#_Toc352172362)

[2. Considerations 7](#_Toc352172363)

[2.1. Programming Languages 8](#_Toc352172364)

[2.1.1. Java 8](#_Toc352172365)

[2.1.2. ANSI C & C++ 8](#_Toc352172366)

[2.1.3. Python 9](#_Toc352172367)

[2.1.4. Javascript 9](#_Toc352172368)

[2.1.5. PHP 10](#_Toc352172369)

[2.2. Types of Databases 10](#_Toc352172370)

[2.2.1. Hierarchical Databases 10](#_Toc352172371)

[2.2.2. Network Databases 10](#_Toc352172372)

[2.2.3. Relational Databases 10](#_Toc352172373)

[2.2.4. Object-oriented Databases 11](#_Toc352172374)

[2.3. Operating Systems 11](#_Toc352172375)

[2.3.1. Microsoft Windows 11](#_Toc352172376)

[2.3.2. Mac OS 12](#_Toc352172377)

[2.3.3. Linux 12](#_Toc352172378)

[2.4. Mobile Phones 12](#_Toc352172379)

[2.4.1. Mobile Operating System (OS) 13](#_Toc352172380)

[2.4.2. Programming languages 13](#_Toc352172381)

[2.4.3. All Smartphones listed have web capabilities 13](#_Toc352172382)

[2.5. Alert Systems 14](#_Toc352172383)

[2.5.1. AMG Alerts 14](#_Toc352172384)

[2.5.2. CYDNE 14](#_Toc352172385)

[3. Solutions 15](#_Toc352172386)

[3.1. Solution 1 15](#_Toc352172387)

[3.1.1. Resources Needed 15](#_Toc352172388)

[3.1.2. Limitations 15](#_Toc352172389)

[3.2. Solution 2 15](#_Toc352172390)

[3.2.1. Description 15](#_Toc352172391)

[3.2.2. Resources Needed 16](#_Toc352172392)

[3.2.3. Limitations 16](#_Toc352172393)

[3.3. Solution 4 16](#_Toc352172394)

[3.3.1. Description 16](#_Toc352172395)

[3.3.2. Resources Needed 16](#_Toc352172396)

[3.3.3. Limitations 16](#_Toc352172397)

[3.4. Mobile Phone Solution 17](#_Toc352172398)

[3.4.1. Description 17](#_Toc352172399)

[3.4.2. Mobile Web Application 17](#_Toc352172400)

[3.4.3. Native Application 17](#_Toc352172401)

[3.4.4. Limitations 17](#_Toc352172402)

[4. Comparison of Solutions 17](#_Toc352172403)

[5. Conclusions 17](#_Toc352172404)

[6. References 17](#_Toc352172405)

# 1. Introduction

This is a report to evaluate the feasibility of the project. It is composed of the introduction, use-case diagram, considerations, solutions, comparisons of the solutions, and conclusions. The introduction will give a generalize history of the project. The use-case diagram will represent how the team currently visualizes how the system will work. The considerations will evaluate different tools that may be used to build the functionalities of the system. Solutions will look at possible combinations of technologies to fulfill the requested functionalities. Comparisons will be a matrix representation of the strength and weakness of each solution. Conclusions will discuss recommendations by the team and the reason for the recommendation.

## Purpose of the Feasibility Report

The purpose of feasibility report is to evaluate the Scientific Sensor Data Manager project requirements against current technologies, team member skills, and possible solutions. The report will evaluate if there are technologies that can fulfill the propose requirements. It will also evaluate the technologies required to develop the system. A use case diagram will depict how the team currently understands the functionality of the requested system. The use case does not represent the final product but an abstract for understanding purposes.

The report will evaluate programing languages that are being considered, different types of database technologies, popular operating systems, popular cell phone technologies that are currently available, alert systems that are currently offered and how each may or may not fit into the requirements definition. A copy of the requirement definition is located in **~~Appendix C~~**. A summary of the requested requirements is located below in section 1.3.

## Justification for the Proposed System

The current system that research scientists are using to gather environmental data from various sensors across different geographical locations has been a problem for the scientist because the interface is difficult to use and the different proprietary formats that the sensor manufactures use. There has been a request by Dr. Pennington to improve the current system which evaluates, manages, and detects anomalies. The following is an excerpt from the taped Interview of Dr. Pennington conducted by Dr. Gates and the Software Engineering Students on February 19, 2013. Dr. Gates ask Dr. Pennington about the limitations of the current system and Dr. Pennington’s response was:

*“The biggest one is that it is written and design and formulated for a computer scientist. It got a language of computer science, it got the interface that is relevant to computer science, talks about queries and codes and rules that's not the language of science. So it does what it needs to do but it is not useable by a scientist.”*

## Requirements Definition

Part of the problem with the interface, according to Dr. Pennington during the Febuary 19 interview, is that the interface is “relevant to computer science, talks about queries and codes and rules that's not the language of science.” Another problem is that there are different data formats for different sensors, so the scientist needs to be able to describe the file format so that the system can “take those proprietary formats and turn them into something more standard.” (Dr. Pennington Febuary 19 interview) This has been an ongoing problem for the scientists. And they are requesting for a new system that will enhance the previous system by fulfilling the following functionalities as per the requirement definition handout in ~~Appendix C.~~

* Improve the user interface for the data property specification tool;
* Provide scientists with the ability to create, retrieve, update, or delete sensor data properties using a mobile device;

1. Provide the ability to describe the file formats for different types of sensors to allow the system to read a file and check for anomalies using the appropriate properties; and
2. Provide the ability to graph the processed data and show where data readings are not satisfying specified data properties.

## Use Cases

The use-case models the actors of the system and the interactions between actors and the system. Each stick figure represents an actor which is someone or something that will use the system for a purpose. The circles in the diagram represent functionalities of the system. The lines represent the interactions. A line connected to an actor and functionality indicates that the actor uses the system for such functionality.

Use Case Diagram (Level 1 abstraction)

### Actors (descriptions)

#### Scientist

The scientists are the individuals that will use the system to retrieve data to be analyzed. They are capable of displaying datasets which can be graphs, data from files, anomalies, or properties. However their access to information of the data may be limited to the property and data they created or were given access to by other scientist. The scientist will also need to describe the file format for external files that they want to analyze on the system. The scientist will also modify properties by creating pattern and scopes that the system will use. The scientist will also be able to set conditions for when an alert is triggered. The scientist will determine if the collected data should be accessible by the public.

#### Database

The database actor is the data store where all the information will be stored. It will be used to store information about the sensors and patterns such as the creator of the item as well as results from the analysis. It will be used by the system for scientist to store their properties, privileges to datasets, and alarm. It is also where saved information will be retrieved from.

#### General Public

The general public actor is representative of the general public that will have some limited access to the system as a part of the current trend of citizen science. The general public will only be able to view only datasets that the client will deem available to the general public, these users will not have the same privileges as a scientist and will not be able to describe a data property, describe a file format, parse a raw data file into a universal format, or verify a dataset.

#### Alert System

The alert system actor is the third party system that will be used to send alert messages to the scientist’s cell phones if they have configured the system to alert them via a cell phone text message. Dr. Pennington requested this feature during the first interview with the client.

#### Datacenter

The data center actor is used to represent a location where the data files from the sensors will be coming to the system. This actor will provide a raw data file to the system that is from a sensor and the system will then transform the file from the hardware specific file to a universal file that the system can analyze.

.

### Use Case Descriptions

#### Display Dataset

The display dataset use case is used to allow either the scientist or general public to view a raw dataset, which would be numeric values. The actors of this use case can also choose to view the dataset in a graphical form such as a pie chart, a line graph, a bar graph, etc. The scientist actor will be given more information to view and more access than the general public actor.

#### Describe Properties

Design Property is where a scientist will be able to create a new data property, edit an existing data property, or delete an existing data property. Once the user has specified a data property, the data property is self is stored in the database to be used or referenced later by the creating scientist or another scientist interested in the created data property.

#### Access Database

Access database use case is used to handle the interactions between the different use cases of the system and the database itself. When a user goes to display a dataset the system will need to retrieve some data from the database and will go though the access database use case. When a scientist describes a data property the system will need to either store or retrieve a property from the database or perform some operations on the data properties. Finally when a scientist describes a file format it is stored in the database or when the scientist needs to transform a new data file, the existing data file description is retrieved from the database. This use case as shown handles the many operations the system needs to carry out on the database.

#### Verify Dataset

The verify dataset use case is used by the scientist when they wish to analyze a dataset, here the user can choose to transform a new data file into a universal format that the system can analyze or they may choose to look at the list of anomalies and determine if they are true anomalies or are generated by a hardware malfunction.

#### Analyze Anomaly

The analyze anomaly is used when a scientist chooses to analyze a dataset with a property, when this analysis is done an anomaly or set of anomalies is generated and presented to the user to analyze if the generated anomaly is a true anomaly and should be documented or if it is a hardware malfunction.

#### Parse Raw Data files

Parse raw data files as the name implies is used to transform a new incoming data file from a data center into a universal format that the system can analyze, this is done by either taking the file format that the user has just created or by retrieving the file format from the database. This is needed as many different sensors may output their data in different file formats that the system will need to analyze, once this is done the system will be able to analyze the data from the file and generate the anomalies that fall out of the user specified data property.

# Considerations

This section will evaluate the technologies that can be used to develop the system. The programming language is the tool use to develop the system. The importance of this is that certain languages may limit the architecture the system can run on. Also, some languages may offer higher performance but with limited capabilities. We wish to explore the languages and evaluate some of the trade-offs that may occur. The types of databases will be evaluated to determine if one offers more functionality to fulfill the requirements. Operating system will represent the host that will hold the system.

## Programming Languages

The following is a list of proposed programming languages that could be used to implement the proposed project, each language description discusses their pros and cons along with the development team’s experience in each.

### Java

The first language under consideration for a possible solution will be the Java programming language; the Java language offers a few advantages as well as a few disadvantages, both of which will be touched on here. The first and probably most significant advantage is that Java is compatible with any machine that supports the Java Virtual Machine, when a Java program is compiled it is compiled into an intermediary language that the Java Virtual Machine can interpret and run, this advantage contributes to working out Dr. Pennington’s need of having the system execute on multiple operating systems (i.e. OSX, Windows, Linux). The next advantage that will be touched on next is in that Java is object-oriented, this feature aids developers to design the implementation of a system in a modular way that separates the different concerns of the overall system. The next significant advantage of Java that will be touched on here is that Java can create multiple threads of execution without the need of many system calls to the operating system, this feature allows the program to perform several tasks at once thereby addressing Dr. Pennington’s requirement that the proposed system is to perform in near real time. In the context of the programming team’s (SDAT) familiarity with the Java programming language, each member is proficient in the language and have the knowledge necessary implement the proposed system in the Java language.

To bring light to the main disadvantage of the Java programming language we look at what is considered one of its strongest advantages the Java programming language has. Since Java compiles its code into an intermediary code that can be interpreted on any system it is an interpreted language, that means that every time the program executes the computer must first interpret the code and then execute it. In the context of the proposed system this disadvantage conflicts with the client’s needs of having the system perform in real time.

### ANSI C & C++

The next Programming language for consideration is the ANSI C Language. The first advantage of using C is that unlike Java, which we previously discussed, is that once a C program is compiled it is compiled specific to the computer’s specific hardware architecture allowing it to run on the system itself. Which makes the C programming language very powerful in the sense that the programmers are able to manipulate the lower levels of the computer system such as memory. In light of the fact that C is able to work with the hardware directly this gives the C programming language a speed advantage that Java does not have, in Java the programs must be executed on a virtual machine where in C it runs on the computer’s hardware, this satisfies the client’s need of having the system perform in near real time. The development team (SDAT) has a slight familiarity and little experience with programming in the ANSI C language, that is not to say that the team does not know how to program in the C language but rather there will need to be some time taken to deepen the understanding of the C language.

Since C++ and ANSI C are generally the same, we will also be talking about the advantages of C++ here. In comparison to ANSI C the only difference between the two languages is that C++ has the features of object oriented programming which as stated previously aids developers to implement their code in small modules. Like ANSI C when a C program is compiled into the computers specific hardware architecture it can run on the computer system itself, it is because of this the program can manipulate the computer’s lower level systems such as memory. It is also because the program can execute on the computer directly that allows the program to execute faster than languages like Java.

The largest disadvantage of the ANSI C programming language is that because it has the ability to work with the computer’s hardware directly, this is the language’s biggest advantage and disadvantage. The programmer(s) must take extra care when they write their code because even if the C compiler does not detect anything wrong with the code, it still may be harmful to the computer and could have disastrous consequences. The programming team’s (SDAT) need for a deeper familiarity with the C language, The development team (SDAT) has some knowledge and experience with the C language however some time will be needed in order to gain a more deeper knowledge of the language for the context of the proposed system.

To continue the earlier discussion of C++ we will discuss the disadvantages of using C++. To start, a major drawback of the C++ language is that even if it is done unintentionally the use of some libraries in the language can restrict the programs’ execution or limit it to certain computing operating systems. Another major disadvantage of C++ that is shared with ANSI C is that since the programs work directly on the computer system, great care must be taken when developing a program because if a program fails it may fail catastrophically and may cause harm to the computer system. Fundamentally ANSI C and C++ are basically the same, in this the SDAT development team has some basic knowledge of the language, but will need some time to gain a deeper knowledge of the language and become more proficient in it.

### Python

The Python language offers a couple of key advantages. The first of which is that Python is supported on multiple systems; this is made possible because the language is written in a portable ANSI C. Another feature of Python is its object-oriented capability, which, previously stated in the other object-oriented languages aids the developers in breaking down the larger components of the system into a smaller, more manageable modules.

Python contains a number of drawbacks, one of which is that it is not a good choice for programs that will be CPU intensive. This in the context of the proposed system would not be a good choice as the program will be doing analysis work to generate graphs and forecasting models, with this drawback in mind this language would not be suitable for this system at this time. The next disadvantage is that Python programs execute slower than programs written in ANSI C or C++, this is because Python is interpretive language. Which previously discussed, means that each time the program is to be executed the byte code that the program is compiled into must be interpreted before it starts its execution. Finally, since the Python language is still being developed and expanded there is a large amount of documentation available for current and past releases. The members of the development team (SDAT) have little to no knowledge programming in Python, because of this the team will need some time to become familiar with the language and its features.

### Javascript

Javascript offers several advantages and is under consideration to solve Dr. Pennington’s requirement of having the proposed system available through a web based solution. The first advantage is in Javascript it can create alerts to the user, this is advantageous because this meets Dr. Pennington’s request of having the system be able to alert the user of an anomaly. In web development something called a ‘Cookie’ might be saved to the user’s web browser, a ‘Cookie’ is a piece of data that relates to a web site that when retrieved by a website alerts the web site of the previous user’s activity on the web page. ‘Cookies’ were designed to remember the state or previous activities on a web page, they also contain data on what buttons were pressed, the log in information, ect. This brings us to the next advantage of using Javascript. Javascript has the ability to set and read ‘Cookies’ this can be helpful to the development of the web based solution that Dr. Pennington requested, at this time it is uncertain how this will be helpful; further research will need to be done. Another advantage of Javascript is that it is cross platform; this means that it is available to the different browsing applications such as Firefox, Safari, explorer, ect. This helps to meet Dr. Pennington’s request of the system to be available through a web-based solution because of its portability of web browsers.

Javascript has a few disadvantages that should be considered, the most key disadvantage deals with security. Javascript code executes on a user’s computer instead of executing on the server, this speeds up the execution and download of the code however this presents a great security issue. Since the code executes on the user’s computer, this feature can be exploited to allow malicious code to execute on user computers. Another issue is that there remains an element of unpredictability. In server side languages typically the program is interpreted and or run in one way and is distributed to the various client web browsers. Javascript however is interpreted and run on the client side, this brings about a level of unpredictability as to how the program will run on a particular client web browser. Lastly the SDAT development team has little knowledge and experience programming in Javascript, the team will need some time to gain a deeper understanding of the language.

### PHP

PHP will allow us to fulfill the request to have a web-based solution. An advantage PHP has is that it is platform dependent, meaning that it can run on different operating systems. Another advantage of PHP is that it can interface with databases; this is helpful since the system will access a database, we are looking at this language because the development team has some knowledge and experience programming in PHP to access and present information from a database. A major flaw in the PHP language is that it is primarily a web programming language, this in some ways limits what can the program can do. In order for the PHP language to be an effective solution to the system, the development team needs some time to become more familiar with the language.

These are the proposed languages that are under consideration to implement the proposed system, in most cases the SDAT development team needs some time to become familiar with some of the languages proposed here. These are the objective pros and cons of each language previously discussed along with the development team’s needs of each.

## Types of Databases

The database will store the properties scientists create. It will also store user privileges owner, read, write.

### Hierarchical Databases

A hierarchical database is organized in a pyramid fashion. The parent record at the top of the pyramid is known as the root record, and any record descending from the parent record is known as the child record. A record search is conducted by starting at the top of the pyramid and working down through the tree from parent to child until the appropriate child record is found. Advantage of this database is that information can be accessed and updated quickly. However, a disadvantage is that each child can only have one parent, and relationships between children are not permitted. This result in a very rigid design that requires the entire database be redefined when adding a new field or record. This database requires developers to predict all possible access patterns in advance and design the database accordingly. A pattern that is not included in the design becomes very difficult and inefficient.

Examples: IBM Information Management System (IMS), RDM Mobile

### Network Databases

A network database is similar to the hierarchal database however the design is more of a cobweb of interconnected records. The children or members of the network database can have more than one parent or owner. Since more connections can be made between different types of data, network databases are considered more flexible. However, two limitations must be considered when using this kind of database. Similar to hierarchical databases, network databases must be defined in advance. There is also a limit to the number of connections that can be made between records.

Examples: Integrated Data Store (IDS), IDMS, RDM, TurboIMAGE, Univac DMS-1100.

### Relational Databases

Relational databases have relations between fields that are not hierarchical. The relations are established by a common key field shared between tables. The data is stored in tables usually with a unique identifier for each row. Relational databases work on the principle that each table has a key field that uniquely identifies each row, and these key fields can be used to connect one table to another.

An advantage of relational databases is that entries can be modified without redefining the entire structure. The downside of using a relational database is that searching for data can take more time compared to other databases.

Examples: Oracle Database, MS SQL Server, MySQL, IBM DB2, IBM Informix, Sybase ASE, Sybase IQ, Teradata

### Object-oriented Databases

An object-oriented database can be used to store data from a variety of media sources, such as photographs and text, and produce work, as output, in a multimedia format. Object-oriented databases use small, reusable chunks of software called objects. The objects themselves are stored in the object-oriented database. Each object consists of two elements: a piece of data like graphs, and the instructions for what to do with the data. However object-oriented are more costly to develop, speed of access may be reduced by late binding. Another disadvantage is the lack of standards including the lack of a common query language such as SQL

Example: ObjectDB

## Operating Systems

We decided to focus on the consideration of the following operating systems:

### Microsoft Windows

Developed, marketed and sold by Microsoft, Windows is a series of graphical interface operating systems. Programmed in C, C++, and Assembly language. The design of our system will focus on the version Windows Vista and above, that is, Windows 7 and Windows 8 because earlier versions of Windows are no longer being supported.

Versions: We will focus on the following three versions of this operating system:

***1. Windows Vista:*** This operating system designed for use on personal computers. Windows Vista primary stated objective was to improve the state of security in the Windows operating system.

**Specifications**

**Release Date:** January 30, 2007.

**Source Model:** Closed model/Shared Source.

**Kernel Type:** hybrid.

**Platform Support:** IA-32, x86-64.

***2. Windows 7:*** This version of the Microsoft Windows operating system was designed for personal computers. Among this new version’s features we include improved performance on multi-processors, and improved boot performance compared to previous versions.

**Specifications**

**Release Date:** October 22, 2009.

**Source Model:** Closed model/Shared Source.

**Kernel Type:** hybrid.

**Platform Support:** IA-32, x86-64.

***3. Windows 8:*** The latest version of the Microsoft Windows operating system, it was designed for personal computers. The main new features include support for ARM microprocessors in addition to the x86 microprocessors from Intel, ADM and VIA Technologies.

**Specifications**

**Release Date:** October 22, 2012.

**Source Model:** Closed model/Shared Source.

**Kernel Type:** hybrid.

**Platform Support:** IA-32, x86-64, and ARM.

### Mac OS

It can be defined as a series of graphical user interface-based operating systems for Macintosh computer systems developed by Apple Inc. Programmed in C, C++, and Objective-C, This operating system is closed source.

**Versions:**

**OS X:** Introduced as Mac OS X, this UNIX operating system is based on the NeXSTEP operating system and the Mach kernel.

**Specifications**

**Release Date:** March 24, 2001.

**Source Model:** Closed model/With Open source components.

**Kernel Type:** hybrid.

**Platform Support:** IA-32, x86-64, and PowerPC.

### Linux

This operating system is Unix-like, free and open source for development and distribution. This system’s defining component is Linux kernel. The designed purpose includes computer architecture support, embedded systems, stability, and security. Other advantages include localization to a specific region or language, targeting specific user groups, support for real-time applications.

**Specifications**

**Release Date:** 1991.

**Source Model:** Free and open source software.

**Kernel Type:** monolithic.

**Platform Support:** [Alpha](http://en.wikipedia.org/wiki/DEC_Alpha), [ARM](http://en.wikipedia.org/wiki/ARM_architecture), [AVR32](http://en.wikipedia.org/wiki/AVR32), [Blackfin](http://en.wikipedia.org/wiki/Blackfin), [C6x](http://en.wikipedia.org/wiki/C6x), [ETRAX CRIS](http://en.wikipedia.org/wiki/ETRAX_CRIS), [FR-V](http://en.wikipedia.org/wiki/FR-V), [H8/300](http://en.wikipedia.org/wiki/H8/300), [Hexagon](http://en.wikipedia.org/wiki/Qualcomm_Hexagon), [Itanium](http://en.wikipedia.org/wiki/Itanium), [M32R](http://en.wikipedia.org/wiki/M32R), [m68k](http://en.wikipedia.org/wiki/M68k), [Microblaze](http://en.wikipedia.org/wiki/Microblaze), [MIPS](http://en.wikipedia.org/wiki/MIPS_architecture), [MN103](http://en.wikipedia.org/wiki/MN103), [OpenRISC](http://en.wikipedia.org/wiki/OpenRISC), [PA-RISC](http://en.wikipedia.org/wiki/PA-RISC), [PowerPC](http://en.wikipedia.org/wiki/PowerPC), [s390](http://en.wikipedia.org/wiki/S390), [S+core](http://en.wikipedia.org/wiki/S%2Bcore), [SuperH](http://en.wikipedia.org/wiki/SuperH), [SPARC](http://en.wikipedia.org/wiki/SPARC), [TILE64](http://en.wikipedia.org/wiki/TILE64), [Unicore32](http://en.wikipedia.org/wiki/Unicore32), [x86](http://en.wikipedia.org/wiki/X86), and [Xtensa](http://en.wikipedia.org/wiki/Xtensa).

## Mobile Phones

The project has also been required to display and functioned correctly on multiple mobile devices. According to the project description, a field scientist needs to be able to create, retrieve, update or delete data properties. Our development team has identified four different platforms we believed will be used. The products we have identified are Apple, Android, Blackberry, and Windows Mobile.

All platforms have different Operating Systems (OS) and their associated programming languages. For example Android application can be program in Java while Apple applications are programmed in Objective C. Writing code for each platform can be difficult especially since they have their own set of APIs. In order for us to approach this in a practical way we need to find a solution that can be done once but works will all platforms without dealing with all different programming languages. Another approach to this is by looking at the web as common platform since all Smartphones have web capabilities. The client will have the final determination in whether to develop a native application or mobile web application. Keep in mind there are other useful tools out there like RhoMobile (Motorola) and Particlecode which uses Java. Our team felt the ones listed are strong contenders to build an application that meets the requirements. It’s imperative our development team become familiar with HTLM, JavaScript and CSS since most of these tools utilize these languages.

### Mobile Web Application

WebKit is an open source web browser engine that works on all platforms except Windows Phone. WebKit was created by Apple for their Safari web browsers on their devices. However, it’s quickly becoming a standard for all mobile web browsing. WebKit utilizes HTML5 features and local database support. Our team needs to be knowledgeable in HTML and JavaScript to be able to use WebKit.

### Sencha Touch

Sencha Touch is a framework that builds web application that currently functions on Android, Apple, and Blackberry using HTML5. A feature that Sencha Touch has that meets one of our requirements is graphing data. Sencha Touch comes with an API that displays data sets graphically, for example a line chart. Unfortunately, Sencha Touch doesn’t work with Windows Mobile as of yet. Using Sencha Touch will require a purchase ranging from $995 to $18,000. This include IT support and free tutorials on their website.

### JQuery

The JQuery mobile framework allows the developer to use HTML 5 to create a mobile app that displays on all the Smartphones we listed. JQuery comes with UI builder which is a drag and drop software. This can also be extended to create application for tablets. They are also third-party plug-ins can be used to enhance some of the requirements such as google maps and multiview (viewing different datasets).

### Mobile Native Application

Creating a native application has its advantage over web mobile application. Native applications can access the mobile internal features such as GPS location or camera. Native applications run faster than a web browser. The software tools listed below to create a native application are actually Hybrid apps. This means they are cross-platform compatible using a web layout however they can access the mobile devices features. The hybrid applications are created using HTML, JavaScript and CSS.

### PhoneGap

PhoneGap is a free and open source framework to create mobile applications that can run on six different platforms including Windows Phone. PhoneGap has an extensive API collection that can meet the requirements set by the client. For instance, a feature of the application is that it can capture geolocations of the device and get notifications such as alerts. Our development team will also need training in order to use PhoneGap. There is a price for training and tutorials using PhoneGap depending on the size of the development team. For instance, if the development team has 5 members, the price start at $495a month.

### Appcelerator

Appcelerator is open source framework to develop native apps that run on Android, Apple, and Blackberry. Currently there is no support for Windows Phone. It uses XML as web a language and JavaScript. Just like PhoneGap, Appcelerator has a large API library (5000).

## Alert Systems

### AMG Alerts

The client wants the system to incorporate an alert system to send an alert for when an anomaly is detected. The alert is supposed to be determined by the user, whether it is by email, text or any other forms, the user has the ultimate decision. AMG Alerts will let us send a notification to the scientist by text, email, voice and web based notifications. This system can plug into any system with the given API. AMG Alerts also lets us have an administrator to keep track of all the subscribers, in this case scientists, and can also edit who receives alerts. This system however does have its drawbacks. There is an annual fee required in order to use the system.

### CYDNE

Another system that could be used which will allow us to send both SMS as well as phone notifications to scientists. This system comes with an API which will allow us to easily connect to any web based application developed in PHP, C#, Python etc. This system has a fee which can be both a pro and a con. It is a pro in which the fee is only a small amount per month, however a con because although small, the fee will add up to a good amount of money. Another con is that the alert options that Dr. Pennington wanted are not all there. There is only phone and text, there is no email or web based notifications.

# Solutions

## Solution 1

### Programming Languages

Solution 1 will use Java and C++ because these languages are compatible with the object oriented database system that this solution will employ. As stated in the programming languages section above the C++ language once compiled will work on the computer system directly, letting the program execute in the near real time that the client specified in the interview. The Java language will be used as an intermediary that will allow the development team to write code in a language they are familiar with and then later translate the Java code into C++.

### Database

Using an Object Oriented database all the functionality requirements will be met. Since an object oriented database treats the data as objects it makes it easier to work with object oriented languages such as python and java. However it is not limited to object oriented languages, other languages like C++ and XML are also supported. This will allow for flexibility on how the interface can be design between the user and the system. Another advantage of the object oriented database since it stores information as objects it is possible to store graphs or whole sets of data in the database. This would improve performance since data will not have to reanalyze every time it is retrieved. Another advantage is that object oriented databases can be represented as either a pure object oriented database management system or as a hybrid of a relational database. This means that the database can behave very similar to the popular relational databases. However there are some disadvantages. Since object oriented database is relative new compared to the other type of database the technical support is not as available as other databases. As all emerging technologies change it is very possible that the system may require constant maintenance in order to keep it stable and running. The number of tools available for object oriented databases is not as large as relative databases. Since the object oriented database is a new technology currently available the development team will need a week or two to research the database system and become more familiar with how it works.

## Solution 2

### Programming Languages

The programming languages chosen for this solution are Java, C, and C++. This solution will utilize these languages because these are some of the languages that are compatible with the Oracle database server. As mentioned in the above considerations section, the ANSI C and C++ languages are very similar to one another, and both languages once they are compiled into machine code can work very closely to the lower levels of the computer system. This is good, because since the system doesn’t need to interpret the instructions for the program, the program can then execute and perform at the near real time rate that Dr. Salamah mentioned in the interview with him. Because of this if care is not taken when developing the program, it has the possibility of failing catastrophically and potentially harming the computer system. Since the SDAT team does not have much experience with programing in C or C++ the Java language will be used as intermediary, where some code will be produced in but will then be translated into C.

### Database

The SDAT development team decided to explore the use of the Oracle system because today it is one of the more popular and widely used database servers. The Oracle database is a relational database, a relational database is a database in which the different fields have a relationship with one another. The data is stored in tables usually with a unique identifier for each row. Relational databases work on the principle that each table has a key field that uniquely identifies each row, and these key fields can be used to connect one table to another. The Oracle X3-2 database appliance is an integrated system of software, servers, storage, and networking. The X3-2 appliance is capable of holding up to 18 terabytes of data, which is ideal since the data being processed on a daily basis could range in the terabytes. Since we don’t know how much data will be processed on a daily basis or even a weekly basis it is difficult to assess how much storage space is needed for regular operation. The monetary cost of this piece of hardware is unknown because the Oracle Company would need to gain a better understanding of what the client needs out of the database server.

## Solution 3

### Programming Languages

Solution 3 will implement the use of the C and C++ programming languages. These programming languages are compatible with the RDM and since they are very similar to one another, and both languages once they are compiled into machine code can work very closely to the lower levels of the computer system. This is good, because since the system doesn’t need to interpret the instructions for the program, the program can then execute and perform at the near real time rate that Dr. Salamah mentioned in the interview with him. Using these parameters, the developing process of the program must be done carefully since the possibility of failing catastrophically and potentially harming the computer system exists Since the SDAT team does not have much experience with programing in C or C++ the Java language will be used as intermediary, where some code will be produced in but will then be translated into C. The incurred cost for the general programming is nothing since there is no software or development environment needed start the implementation of the proposed system, the main cost to be incurred would be on time as the development team would need about two to three weeks to gain a deeper understanding of the C/C++ language and become familiar enough with it to become confident programmers in the language.

### Database

The SDAT team decided to use the *Raima* database because it is cross-platform, widely use and popular. *Raima Database Manager (RDM)* is a database management system that is currently divided into several environments but we will focus on the usage of the following: *Desktop & Server*, *Embedded*, and *Mobile*. The *Desktop & Server* environment is used for the development of applications in environments such as backup systems and security applications. The *Embedded* environment is used when building solutions to real-time operating systems. The *Mobile* environment is used for applications for smart phones and tablet devices. These three environments will be used and incorporated in the system. *RDM* works under the principle of having tables where each has a special field called a key, which identifies each row as unique. The keys can be used to establish a relation or connection between tables. A relevant feature offered by this database management system is the ability to be ported to different embedded or real-time operating systems known as cross-platform. It is difficult to assess the storage space that will be required for a standard operation since the exact amount of processed data for either is unknown.

## Solution 4

### Description

In order to create a mobile application to meet the complexity of the system, our team believes using PhoneGap. The software is free and applications can run on the Smartphones we listed which include Android, Apple, Blackberry, and Windows Phone. A lot of the other tools did not support Windows Phone. Creating a native application or hybrid, we have access to mobile’s internal features such as Contacts and GPS locations that can enhance the requirements of the system.

### Resources Needed

PhoneGap can be use with Eclipse IDE which our team is familiar with. We need to have a basic understanding of mobile web development which includes using HTML5, JavaScript and CSS. Our team has some understanding of these languages and how they affect the behavioral of the web page. This solution will not required us to learn each platforms programming language.

### Limitations

A limitation on using this solution is the cost of tutorials and IT support. It will also take time to learn the API and collaboration of the web languages mentioned.

# Comparison of Solutions

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Cost** | **Needed Resources** | **Portability** | **Risk** |
| *Solution 1* | Cost of the object oriented database system, roughly about 3 weeks to learn and research what will be needed to make this solution viable and how to make it work  (1 week to understand the object oriented database system, and 2 weeks to gain a deeper understanding of C++) | The system will have a module to view datasets in graphical forms, with this in mind a graphing utility will be needed to accomplish this feature. Also the system will be able to visualize where a dataset is gathering its data on a map, with this feature in mind a mapping utility will also be needed to allow for this functionality | This solution will be available on operating systems that have the capability to run C programs | The risks that are associated with this solution are mainly errors associated with the development team’s lack of knowledge in C based languages and unfamiliarity with the object oriented database system. To mitigate this the team requests some time to learn the C based languages and to further learn more about the object oriented database system |
| *Solution 2* | the Oracle database system a quote is needed from the Oracle company to the client in order to put a specific monetary value There will need to be a two to three week timeframe for the development team to gain a deeper knowledge of the C and C++ languages | The system will have a module to view datasets in graphical forms, with this in mind a graphing utility will be needed to accomplish this feature. Also the system will be able to visualize where a dataset is gathering its data on a map, with this feature in mind a mapping utility will also be needed to allow for this functionality | This solution will be available on operating systems that have the capability to run C programs | The risks that are associated with this solution are mainly errors associated with the development team’s lack of knowledge in C based languages and unfamiliarity with the Rima database system. To mitigate this the team requests some time to learn the C based languages and to further learn more about the Rima database system |
| *Solution 3* | Cost of the Rima database system, roughly about 3 weeks to learn and research what will be needed to make this solution viable and how to make it work  (1 week to understand the Rima database system, and 2 weeks to gain a deeper understanding of C++) | The system will have a module to view datasets in graphical forms, with this in mind a graphing utility will be needed to accomplish this feature. Also the system will be able to visualize where a dataset is gathering its data on a map, with this feature in mind a mapping utility will also be needed to allow for this functionality | This solution will be available on operating systems that have the capability to run C programs |  |
| *Solution 4* | <<$$$ and time cost>> | The system will have a module to view datasets in graphical forms, with this in mind a graphing utility will be needed to accomplish this feature. Also the system will be able to visualize where a dataset is gathering its data on a map, with this feature in mind a mapping utility will also be needed to allow for this functionality | << OS System is viable on>> |  |

# Conclusions

The SDAT feels the best possible solution for the system is <<SOLUTION X>> because of the following reasons

1. Cost is XXXX
2. Needed resources are compared to
3. Portability
4. Team skills
5. Functionality requirements met.

# References

"Topic: Database Fundamentals." *Lesson 5: Types of Database Management Systems*. N.p., n.d. Web. 08 Mar. 2013.

"Hierarchical Database." *Hierarchical Database*. N.p., n.d. Web. 08 Mar. 2013.

Dover, Sara. "Study: Number of smartphone users tops 1 billion." *CBS News*. CBS Interactive Inc, 12 10 2012. Web. 5 Mar 2013. <http://www.cbsnews.com/8301-205\_162-57534583/study-number-of-smartphone-users-tops-1-billion/>.

Wroblewski, Luke. *Mobile First*. New York: Jeffrey Zeldman, 2011. Print.

**Kuh, Nick. *Foundation iPhone App Development: Build An iPhone App in 5 Days with iOS 6 SDK:* Springer, 2012. Print**

Website: *Windows*

Link: <http://windows.microsoft.com/en-us/windows/history>

Website: *Wikipedia: The Free Encyclopedia*

Link: <http://en.wikipedia.org/wiki/Microsoft_Windows>

Website: *Wikipedia: The Free Encyclopedia*

Link: <http://en.wikipedia.org/wiki/Mac_OS#OS_X>

Website: *Wikipedia: The Free Encyclopedia*

Link: <http://en.wikipedia.org/wiki/Linux>

Website: AMG Alerts

Link: https://www.amgalerts.com/index.asp

Website: CYDNE Corporation

Link: http://www.cdyne.com/api/phone/sms/?gclid=CIvqhcKz7LUCFemiPAodxRUAqw

[**http://developer.blackberry.com/**](http://developer.blackberry.com/)

[**https://developer.apple.com/xcode/**](https://developer.apple.com/xcode/)

[**http://developer.android.com/index.html**](http://developer.android.com/index.html)

[**http://dev.windowsphone.com/en-us**](http://dev.windowsphone.com/en-us)

[**http://www.mobl-lang.org/**](http://www.mobl-lang.org/)

[**http://phonegap.com/**](http://phonegap.com/)

**Requirements Definition**

**CS 4310 Spring 2013**

**Introduction**

Current trends and fluctuations in the Earth’s climate have resulted in an increased focus by scientists to study changes in environmental conditions to better understand climate change and associated impacts. Scientists use advanced sensor technology such as meteorological towers, wireless sensor networks, and robotic trams equipped with sensors to collect data at remote research sites. As the amount of data collection instruments introduced in the field increases, so does the volume of environmental sensor data acquired by such instruments. The measurements taken by sensors are discrete samples of physical phenomena and are subject to review of their accuracy dependent on their location.

**The Problem**

For some scientific projects, instrumentation typically does not include mechanisms to detect *anomalies* as data are collected. In this context, an *anomaly* is a deviation from an expected sensor data value. An anomaly in data does not always represent errors: it may represent environmental variability requiring further analysis. A common practice for environmental scientists is to collect sensor data for extended periods of time, possibly creating numerous individual data files. During data collection, these files are typically not checked to ensure that they adhere to predefined data quality standards. Checking often occurs when the files are transferred to a database. A challenge with this practice is that a large amount of incorrect data can be collected due, for example, to a faulty sensor, and this can be undetected for extended periods of time. Furthermore, if data are identified as incorrect during analysis at a much later time, the data gathering process may have to be repeated.

Repeating data collection is expensive especially when the site is at a remote location. Sensor technology needs to be redeployed and possibly recalibrated; the amount of time required to gather the data can be significant. For time sensitive data that is required for policy decision-making, it might not be even possible to repeat the data gathering process. Because environmental sensor data can be non-reproducible entities, i.e. the observations at a given time and set of conditions cannot be repeated, the knowledge that could have been obtained from the correct data is not captured.

**Requirements Overview.** To assist scientists in locating anomalies in scientific sensor data, Cyber-ShARE Center created a tool to specify sensor data properties that can monitor data collected in near real time or data that has been stored in files. The overall purpose of the proposed system is to enhance support for specifying data properties and to apply these properties to monitor sensor data files that may have different formats.

There are two viewpoints in this approach. The notion of using patterns and scope to facilitate specification of commonly used data properties comes from software engineering. The other is the field scientist whose interest is in identifying anomalies in the data set. How can the system facilitate their work in the field?

Your task is to build a system that addresses the following:

* Improve the user interface for the data property specification tool;
* Provide scientists with the ability to create, retrieve, update, or delete sensor data properties using a mobile device;

1. Provide the ability to describe the file formats for different types of sensors to allow the system to read a file and check for anomalies using the appropriate properties; and
2. Provide the ability to graph the processed data and show where data readings are not satisfying specified data properties.