TABLE OF CONTENTS

1. INTRODUCTION

1.1. PROBLEM STATEMENT

This project is the continuation of the Temperature Control System (TCS) done in previous semester (Course: ENPM641). The goals for this semester (Course: ENPM642) are:

1. Define quantitative requirements for the project at a system level and a component level.
2. Ensure that the components selected can communicate among each other by evaluating its input/output characteristics and constrains.
3. Perform optimization and tradeoff analysis so the best combination of components can be selected for a given set of decision criteria.
4. Perform appropriate traceability of requirements to ensure the components and configuration selected are able to fulfill the expected performance and behavior parameters.

Please refer to http://www.wam.umd.edu/~fgallo/ENPM641/Index.htm for the Final Report from ENPM641. The file name (Word version) of this Final Report is [UMD_ENPM641_VN.n.doc]. It is also available in PDF and in HTML versions. It is highly recommended to revise this document in order to have a clear understanding of the steps developed in present semester for the course ENPM642.

For simplicity purposes, all references to the Final Report developed for course ENPM641 will be done by using its file name: [UMD_ENPM641]. Note how the version number and type of file have been omitted since new versions of it may be generated in the future as iterations produce more mature models of the problem/solution set.

1.2. CLASS RELEVANCE

This project is related to concepts learned in class because it is an exercise in techniques such as system decomposition, system optimization, analysis of system structure and system behavior, system modeling, and system cost analysis. This project puts into practice the fundamental systems engineering precept which calls the engineer to envision the entire system beforehand, while cost is relatively low, rather than later where cost is relatively high.

Also the principle of optimizing at the system level architecture rather than at the detailed system design will be applied here.

2. RESTRICTIONS AND ASSUMPTIONS MADE FOR THIS PROJECT

Not all the functionality described in [UMD_ENPM641] will be included in this second stage of the project (Course ENPM642). The reason for this is that an attempt to simplify the project has been made to allow focusing more in systems engineering rather than in detail engineering.

Following is the list of the restrictions and assumptions made for this part of the project:

1. In ENPM641 the TCS was conceived as a generic system that could be installed into any room or space, this is, no specific restrictions were done on this regard. Now, for ENPM642 the system will be instantiated and optimized for a specific space. This space will be a warehouse with the following dimensions: Height= 8 feet. Width=100 feet. Length= 100 feet. The warehouse will be considered as a space poorly insulated.

2. A single sensor will be used to measure the temperature. Single temperature control actuator will be utilized to regulate the environment.
3. Several air entrances will be considered. This means, there will be only one heating unit, one cooling unit but there will be several entrances where the regulated air will be pumped into the warehouse.
4. The project will be focused on the following components:
    a. **Thermometer**
    b. **Controlling device** → for example, a PID or other.
    c. **Programmable parameters controlling device (PPCD)** → for example a PC where the user can program parameters such as desired temperature.
    d. **Fan**
    e. **Heater**
    f. **Chiller**

   No special attention will be paid to how the user interface works (i.e. the kind of display to be used and the input device – keypad or other).
5. Some simulation tools available on the Internet have been used to generate estimates (e.g. BTUs and similar computations). These simulation tools make certain assumptions based on the expected average temperature and general weather conditions of the place where the system to control de temperature is going to be installed. In order to provide a context for these suppositions, all the calculations done for this project were done assuming that the warehouse is located in Virginia.

3. EQUIVALENCES OF ANALYSIS CLASSES TO COMPONENTS

Table 3.1 provides the mapping of the Analysis Classes that represented the components in [UMD_ENPM641]. Note how the classes defined in such document belong to the Analysis workflow (according to the naming convention of the Rational Unified Process) of the life cycle of this project. In current document, these classes are linked to actual components (called “Realization Components”) that will realize their functionality. Note however that in Table 3.1 each of the Realization Components are general entities based on commercially available concepts though no particular vendor specification is included yet. The Realization Components will be considered as entities belonging to the Design workflow.

It is important to bear in mind that since a clearer definition of each component is being attained at this point, the appropriate behaviors will also have to be refined in order to ensure proper boundaries and to accurately set the responsibilities of each Realization Component and how the requirements are mapped into them.

In Table 3.1 note how some Analysis Classes are mapped into more than one Realization Component. The reason for this is that an Analysis Class may be implemented in more than one Realization Component. For instance, a PPCD can have a display; also, a Controlling device may have it as well. This circumstance may have an impact in the models presented in [UMD_ENPM641], however iterating back into such document is beyond the scope of this project. Under real circumstances that would be a must.

Analysis Class [from ENPM641 Final Report]→ See Fig. 3.4 and 3.5.	**Realization Component**
1. Control software	Controlling Device
2. Processing unit	Controlling Device
3. Storage unit	Controlling Device
4. Display	Controlling Device
5. Keypad	Controlling Device
1. Fan	Fan

Analysis Class [from ENPM641 Final Report]→ See Fig. 3.4 and 3.5.	**Realization Component**
1. Storage unit	PPCD
2. Validation software	PPCD
3. Control software	PPCD
4. Display	PPCD
5. Keypad	PPCD
6. Processing unit	PPCD
1. Cooler	Chiller
1. Heater	Heater
1. Thermometer	Thermometer

Table 3.1

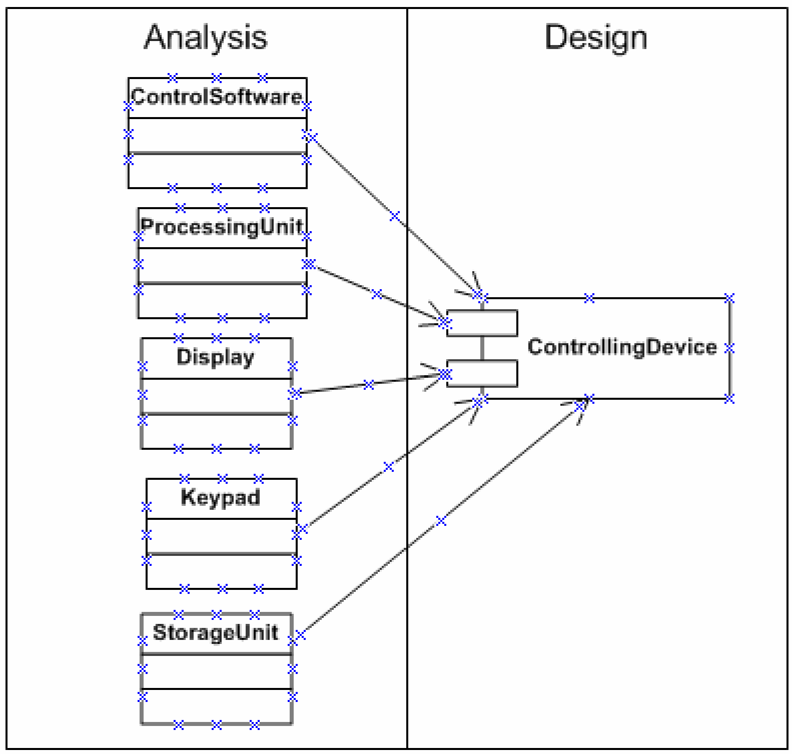


Figure 3.1

Figure 3.1and Figure 3.2 show the links between some of the entities in the Analysis workflow to those in the Design Workflow. These figures map directly to the equivalences shown in Table 3.1.

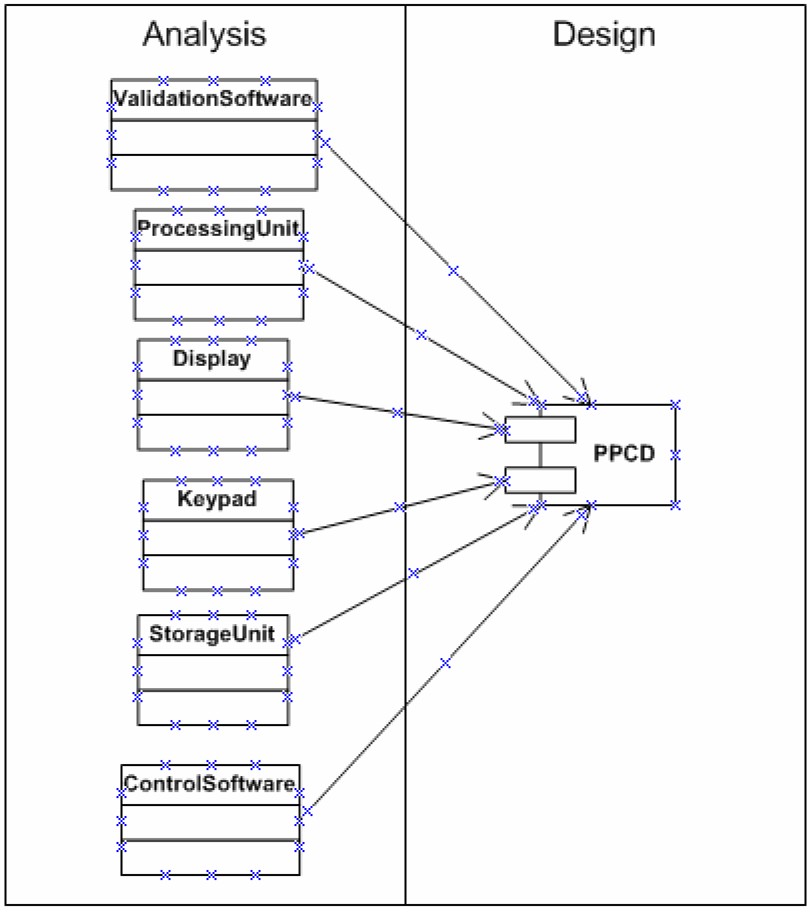


Figure 3.2

4. FORMULATION OF SYSTEM LEVEL REQUIREMENTS

These requirements are variations of those defined for this project for course ENPM641. The goal here is to refine these requirements so they are defined in terms of objective (quantitative) parameters.

A series of conventions will be used in Table 4.1. These conventions follow the naming rules and numbering sequences provided in document [UMD_ENPM641]. These conventions are explained below.

4.1. CONVENTIONS DESCRIPTION:

Several levels of requirements are defined. Please note that Levels 1 and 2 correspond to the requirements defined in [UMD_ENPM641], even though in such document they were not referred to by using Levels. Following, a description of each level of requirements is provided. Figure 4.1 summarizes each level of requirements.

1) **Level 1:** Corresponds to requirements defined in [UMD_ENPM641], section 2.1.
2) **Level 2:** Corresponds to requirements defined in [UMD_ENPM641], section 2.6.
3) **Level 3:** Corresponds to the re-definition of the requirements based on the assumptions and restrictions from Section 2 in this document. These Level 3 requirements are based

on the Level 2 ones and are mapped to them based on how similar they are. Level 3 requirements incorporate changes in the system; the mapping is a guide for traceability purposes but a 1 to 1 exact equivalence should not be expected. Note also that one requirement from Level 2 may be split into several Level 3 requirements or vice versa. See Table 4.1 for the mapping of Level 2 to Level 3 requirements.

4) **Level 4:** Corresponds to the requirements-specifications (req-specs) where objective (quantitative) parameters have been defined. These req-specs map to the Level 3 requirements. Note that the quantitative parameters defined for Level 4 req-specs have been determined by using various estimation tools based components available in the market. Most of the appropriate references have been included in the table where the Level 4 req-specs are shown (Table 4.3).

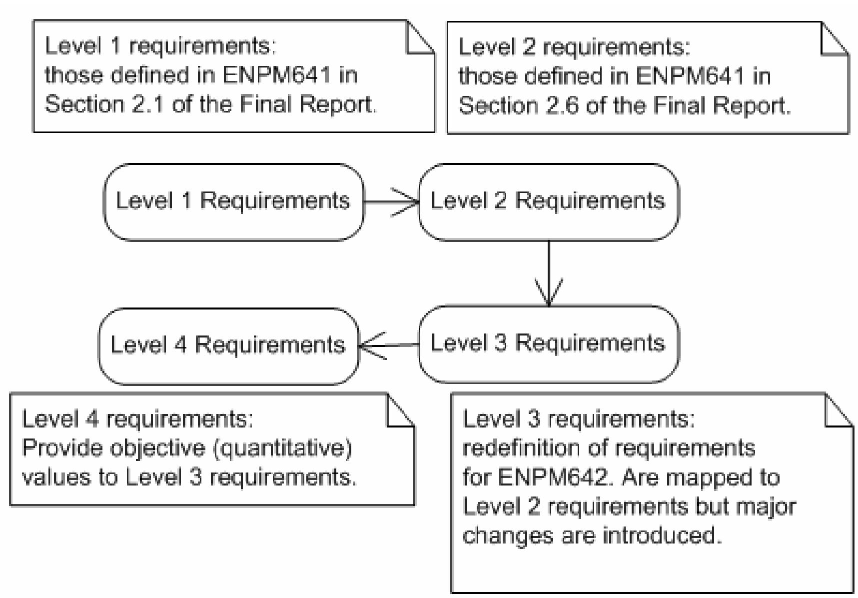


Figure 4.1

4.2. MAPPING OF LEVEL 2 TO LEVEL 3 REQUIREMENTS

Level 2 Req #	Level 2 Requirement Description	Level 3 Req #	Level 3 Requirement Description	Notes
d.1	The System requires logic for calculating the optimum solution for a given height, length, and width.	L03_d.1	The TCS shall have the appropriate power capabilities to control de temperature in the warehouse.	The system conceived in course ENPM641 was a general conceptual approach to the problem of controlling the temperature of a certain space. In ENPM642 the system has been elaborated into a more realistic one where a specific warehouse will be heated/cooled.
d.2	The System	L03_d.1	The TCS shall have the	Same as for item above.

Level 2 Req #	Level 2 Requirement Description	Level 3 Req #	Level 3 Requirement Description	Notes
	requires logic for validating the maximum and minimum sizes it is capable of heating/cooling within an acceptable time range		appropriate power capabilities to control de temperature in the warehouse.	
d.3	The System requires n sensors depending on the size of the room	L03_d.3	The system shall use a single sensor strategically located in the space to be controlled. This sensor will be a thermometer. This sensor shall have an acceptable level of accuracy. This sensor shall also be able to measure an appropriate range of temperatures. If this sensor uses power to operate, this power shall be minimum.	This requirement has been simplified. The reason for this simplification is provide a simpler set of scenarios facilitating the assessment of different alternatives.
d.4	The System requires logic to process multiple sensors in order to find the correct temperature to pump into the room			Because of requirement L03_d.3, requirement d.4 is no longer applicable.
e.1	The System requires a time constraint in which the set temperature is to be met	L03_e.1	The system shall be able to heat or cool a certain flux (volume) of air so the desired temperature can be reached within a certain time.	No major changes to this requirement. The statement has been modified to make it more explicit, but in essence it remains the same.
e.2	The System actuators should be activated depending on the time constraint	L03_e.1	The system shall be able to heat or cool a certain flux (volume) of air so the desired temperature can be reached within a certain time.	Same as for item above.
e.3	The System requires a method to heat	L03_e.2	The system shall be able to heat and cool the air inside the room while using the	Note that since the power delivered by the system is a function of

Level 2 Req #	Level 2 Requirement Description	Level 3 Req #	Level 3 Requirement Description	Notes
	and cool air outside the room		least possible amount of energy.	the space to be controlled and its physical characteristics, the key parameter to consider in this requirement will be the efficiency of the system. This is, the ability of the system to transfer as much power as possible to the room while minimizing the waste of energy (e.g. thermal losses in undesired parts).
e.4	The System requires a ventilation method to pump air into the room	L03_e.3	The system shall be able to pump air into the room while using the least possible amount of energy given a certain pumping capacity.	This is, the efficiency of the fan(s) used to pump air into the space should be as high as possible. This is similar to the efficiency described for requirement L03_e.1.
e.5	The System requires a ventilation method to remove air from the room	L03_e.4	The system shall be able to extract air from the room while using the least possible amount of energy given a certain pumping capacity.	This is, the efficiency of the fan(s) used to extract air from the space should be as high as possible. This is similar to the efficiency described for requirement L03_e.1.

Table 4.1

Requirements that were not included in Table 4.1 are those that will remain unchanged. This is, the declaration of each of them remains the same as in Table 3.4 in [UMD_ENPM641]. Note that these remaining requirements are focused on logical behavior and are basically related to the interaction of the system with the user. Table 4.2 maps these requirements.

Level 2 Req #	Level 2 Requirement Name	Level 3 Req #	Level 3 Requirement Name
a.1	The Administrator requires a password	L03_a.1	The Administrator requires a password
a.2	The System requires logic to process a password	L03_a.2	The System requires logic to process a password
b.1	The System requires logic to process time range parameters	L03_b.1	The System requires logic to process time range parameters
b.2	The System requires logic to tell time	L03_b.2	The System requires logic to tell time
b.3	The System requires logic to process two temperature parameters (non-	L03_b.3	The System requires logic to process two temperature parameters (non-

Level 2 Req #	Level 2 Requirement Name	Level 3 Req #	Level 3 Requirement Name
	simultaneously)		simultaneously)
c.1	The System requires temperature display modes: current temperature, interval 1, and interval 2	L03_c.1	The System requires temperature display modes: current temperature, interval 1, and interval 2
c.2	The System requires a method for transferring from one display mode to another	L03_c.2	The System requires a method for transferring from one display mode to another
c.3	The System requires input display mode: determine parameters	L03_c.3	The System requires input display mode: determine parameters
c.4	The System requires a method for transferring from one input mode to another	L03_c.4	The System requires a method for transferring from one input mode to another

Table 4.2

Since, as mentioned before, requirements in Table 4.2 are mainly of a logical nature, section identification of performance evaluation parameters (below) will only focus on requirements mapped in Table 4.1.

4.3. IDENTIFICATION OF PERFORMANCE EVALUATION PARAMETERS

Based on the information included in Table 4.1 the following performance evaluation parameters have been identified:

Major Criteria for Component Selection:

1. Initial acquisition cost.
2. Durability of system.
3. Reliability of system.

Criteria for Performance Evaluation:

1. Efficiency of energy usage of the system (i.e. minimum waste, maximum power transfer to actual load).
2. Accuracy of components.
3. Speed of system to reach desired conditions.
4. Power output (i.e. capability of the system to perform given the magnitude of the element to be controlled)→ e.g. BTUs of heating unit.

4.4. MAPPING OF LEVEL 3 TO LEVEL 4 REQUIREMENTS/SPECIFICATIONS

Table 4.3 lists the equivalences of Level 3 to Level 4 requirements. Note how the last column of this table includes the component each of these requirements maps to.

Level 3 Req #	Level 3 Requirement Description	Level 4 Req #	Level 4 Requirement Description	Notes	Maps to
L03_d.1	The TCS shall have the appropriate	L04_d.1.1	The BTUs of the TCS for heating	The BTUs defined for the	Heater

Level 3 Req #	Level 3 Requirement Description	Level 4 Req #	Level 4 Requirement Description	Notes	Maps to
	power capabilities to control de temperature in the warehouse.		purposes shall not exceed 320,000 units. The BTUs shall not be less than 160,000 units.	heater were calculated by using the tool available at [http://hearth.com/calc/btucalc.html].	
L03_d.1	The TCS shall have the appropriate power capabilities to control de temperature in the warehouse.	L04_d.1.2	The BTUs of the TCS for cooling purposes shall not exceed 350,000 units. The BTUs shall not be less than 200,000 units.	The BTUs defined for the cooling system were estimated by using the tool available at [http://www.a-wall.com/ACcalc.php].	Chiller
L03_d.3	The system shall use a single sensor strategically located in the space to be controlled. This sensor will be a thermometer. This sensor shall have an acceptable level of accuracy. This sensor shall also be able to measure an appropriate range of temperatures. If this sensor uses power to operate, this power shall be minimum.	L04_d.3.1	The error of the thermometer shall not be more than +/-5^{O} Celsius.	No additional notes	Thermo meter
L03_d.3	The system shall use a single sensor strategically located in the space to be controlled. This sensor will be a thermometer. This sensor shall have an acceptable level of accuracy. This sensor shall also be able to measure an appropriate range of temperatures. If	L04_d.3.2	The range that the thermometer shall be capable of measuring is: Tmin = -20^{O} C Tmax = 60^{O} C	No additional notes	Thermo meter

Level 3 Req #	Level 3 Requirement Description	Level 4 Req #	Level 4 Requirement Description	Notes	Maps to
	this sensor uses power to operate, this power shall be minimum.				
L03_d.3	The system shall use a single sensor strategically located in the space to be controlled. This sensor will be a thermometer. This sensor shall have an acceptable level of accuracy. This sensor shall also be able to measure an appropriate range of temperatures. If this sensor uses power to operate, this power shall be minimum.	L04_d.3.3	The voltage that the thermometer requires to operate shall not exceed 15V	No additional notes	Thermo meter
L03_e.1	The system shall be able to heat or cool a certain flux (volume) of air so the desired temperature can be reached within a certain time.	L04_e.1.1	The heater shall provide airflow of at least 3,000 CFM. The airflow provided by the heater shall not exceed 4,000CFM CFM = Cubic Feet per Minute.	No additional notes	Heater
L03_e.1	The system shall be able to heat or cool a certain flux (volume) of air so the desired temperature can be reached within a certain time.	L04_e.1.2	The chiller shall provide airflow of at least 3,000 CFM. The airflow provided by the chiller shall not exceed 4,000CFM CFM = Cubic Feet per Minute.	No additional notes	Chiller
L03_e.1	The system shall be able to heat or cool a certain flux (volume) of air so the desired temperature can be reached within a certain time.	L04_e.1.3	The fan shall provide airflow of at least 3,000 CFM. The airflow provided by the fan shall not exceed 4,000CFM	No additional notes	Fan

Level 3 Req #	Level 3 Requirement Description	Level 4 Req #	Level 4 Requirement Description	Notes	Maps to
			CFM = Cubic Feet per Minute.		
L03_e.2	The system shall be able to heat and cool the air inside the room while using the least possible amount of energy.	L04_e.2.1	The efficiency of the Heater shall not be less than 75%.	No additional notes	Heater
L03_e.2	The system shall be able to heat and cool the air inside the room while using the least possible amount of energy.	L04_e.2.2	The efficiency of the Chiller shall not be less than 68%.	No additional notes	Chiller
L03_e.3	The system shall be able to pump air into the room while using the least possible amount of energy given a certain pumping capacity.	L04_e.3.1	The efficiency of the fan shall not be less than 80%	No additional notes	Fan
L03_e.4	The system shall be able to extract air from the room while using the least possible amount of energy given a certain pumping capacity.	L04_e.3.1, L04_e.1.3	See note to the right.	This L03 requirement maps to the same L04 requirements as L03_e.3 and L03_e.1.	Fan

Table 4.3

4.5. CONSOLIDATION OF REQUIREMENTS THAT EACH COMPONENT SHOULD FULFILL

Table 4.4 lists the requirements/specifications defined in Table 4.3 sorted by the component they belong to. This view of the system is useful because it helps to visualize in a consolidated fashion, what the responsibilities of each component are.

Component	Level 4 Requirements	Other performance evaluation parameters
Thermometer	1. L04_d.3.1 2. L04_d.3.2 3. L04_d.3.3	1. Minimize: Initial acquisition cost 2. Maximize: Reliability 3. Maximize: Durability
Controlling device	No level 4 requirements	1. Minimize: Initial acquisition cost 2. Maximize: Reliability 3. Maximize: Durability
PPCD	No level 4 requirements	1. Minimize: Initial acquisition

		cost 2. Maximize: Reliability 3. Maximize: Durability
Fan	1. L04_e.1.3 2. L04_e.3.1	1. Minimize: Initial acquisition cost 2. Maximize: Reliability 3. Maximize: Durability
Heater	1. L04_d.1.1 2. L04_e.1.1 3. L04_e.2.1	1. Minimize: Initial acquisition cost 2. Maximize: Reliability 3. Maximize: Durability
Chiller	1. L04_d.1.2 2. L04_e.1.2 3. L04_e.2.2	1. Minimize: Initial acquisition cost 2. Maximize: Reliability 3. Maximize: Durability

Table 4.4

5. INTERFACE CONSTRAINER PROGRAM

For the system to operate, the alternatives that will be considered for its implementation should be made of components that can communicate among each other. To ensure this, a list of the parameters that each link should evaluate for compatibility has been provided in the matrix shown in Table 5.1. Data in columns represent the outputs from each of the components at the top of the table. Data in rows represent the inputs that each of the components in the leftmost column expect.

Three different concepts are used in this project to ensure compatibility:

1. Ranges of a certain kind of signal → Value_min and Value_max
2. Type of signal → Analog or Digital
3. Type of link → Serial or Parallel

For instance, for two components to be able to communicate, the ranges (e.g. V_{min} to V_{max}) should be tolerant to each other. Also, certain parameters such as the type of signal (e.g. Digital or Analog) should also be consistent.

Note that these compatibility concepts are just a small subset of the actual ones that should be considered in a real implementation. These concepts have been simplified here in order to allow a greater focus on the methodology rather than in the complexity of the alternatives.

The components in the system are connected according to Figure 5.1. The arrows indicate which component is the “inputter” and which component is the ”outputter.”

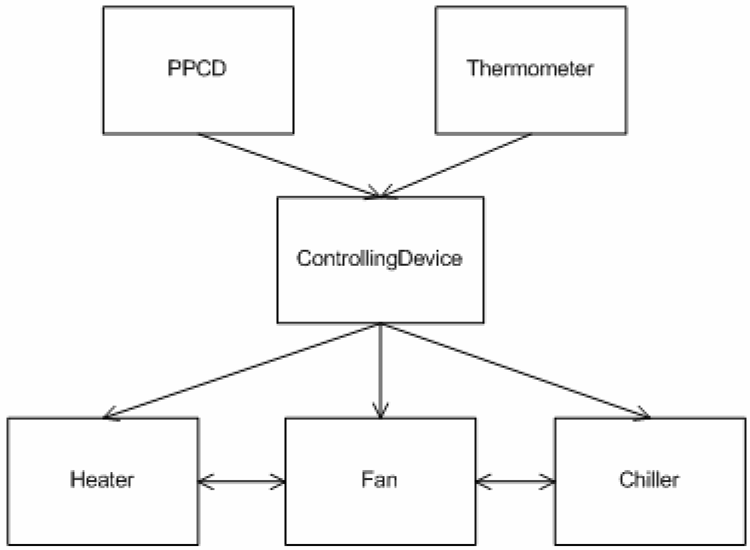


Figure 5.1

Most of the concepts in Table 5.1 are based on those provided in commercially available components that were found by doing Internet research.

Outputs(→) **Inputs(↓)**	**Heater**	**Chiller**	**Fan**	**Controlling Device**	**PPCD**	**Thermometer**
Heater	NA	NA	NA	*Voltage control signal* (V_{min} to V_{max}) *Impedance* *Type of signal: Analog or Digital* *Type of Link: Series or Parallel*	NA	NA
Chiller	NA	NA	NA	*Voltage control signal* (V_{min} to V_{max}) *Impedance* *Type of signal: Analog or Digital*	NA	NA

Outputs(→) Inputs(↓)	Heater	Chiller	Fan	Controlling Device	PPCD	Thermometer
				Type of Link: Series or Parallel		
Fan	*Air Flow* (CFM_{min} to CFM_{max})	*Air Flow* (CFM_{min} to CFM_{max})	NA	*Current control signal* (I_{min} to I_{max}) *Impedance* *Type of signal: Analog or Digital* *Type of Link: Series or Parallel*	NA	NA
Controller	NA	NA	NA	NA	*Voltage control signal* (V_{min} to V_{max}) *Type of signal: Analog or Digital* *Type of link: Serial or Parallel*	*Measuring signal* (V_{min} to V_{max}) *Impedance* (Z_{min} to Z_{max}) *Type of signal: Analog or Digital* *Type of link: Serial or Parallel*
PPCD	NA	NA	NA	NA	NA	NA
Thermometer	NA	NA	NA	NA	NA	NA

Table 5.1

Table 5.2 below shows a mapping between the components researched in the market that are in the Temperature Control Component Database XML file and the components which are optimized in the Multi-objective Optimization Excel Spreadsheet. The optimizable values in the database are the same values used in the spreadsheet. The values were researched at the sites included in the References section, though some values have been changed in order to make the interface constraint as well as the optimization more interesting.

Component Name: MultiobjectiveOptimization.xsl	Component Name: TemperatureControlComponentDatabase.xml
Controller 1	CALController 3300
Controller 2	CALController 9300

Component Name: MultiobjectiveOptimization.xsl	**Component Name: TemperatureControlComponentDatabase.xml**
Controller 3	CALController 9500
(Rejected by Interface Constrainer Program)	CALController 1200
Heater 1	Sterling 250 TF
Heater 2	Sterling 300 TF
Heater 3	Sterling 450 TF
(Rejected by Interface Constrainer Program)	Sterling 150 TF
Chiller 1	Sterling 090 AKV
Chiller 2	Sterling 100 AKV
Chiller 3	Sterling 120 AKV
(Rejected by Interface Constrainer Program)	Sterling 070 AKV
Fan 1	Sterling SHAT 2010
Fan 2	Sterling SHAT 2020
Fan 3	Sterling SHAT 2030
(Rejected by Interface Constrainer Program)	Sterling SHAT 1900
PPCD 1	Microsoft TempQiz 3.0
PPCD 2	Microsoft TempQiz 3.3
PPCD 3	Microsoft TempQiz 3.5
(Rejected by Interface Constrainer Program)	Microsoft TempQiz 2.0
Thermometer 1	Analog Devices TMP 17
Thermometer 2	Analog Devices TMP 18
Thermometer 3	Analog Devices TMP 19
(Rejected by Interface Constrainer Program)	Analog Devices TMP 10

Table 5.2

In past projects, students have optimized components from a component library without taking into account the complex interfaces and limitations inherent in connecting components together. The Interface Constrainer Program (ICF) program directly addresses this issue by constraining components specifically based on the interface (input and output) specifications they have in relation to other objects. By constraining components, it allows the user of the program (a systems engineer) to enter the interface requirements he or she may have from a component they desire to use in order to find only the components which are compatible with that component.

The library of TCS components is stored in an XML file according to a schema that organizes a component by the following hierarchy (See Appendix A for an HTML output of the component library used in the program.):

```
Component Library
        Component1
                Type
                Name
                General Characteristics
                        Characteristic 1
                                Name
                                Unit
                                Value
                                Description Text
                        Characteristic 2…
                Optimizable Characteristics
                        Characteristic 1
                                Name
                                Unit
```

        Value
        Description Text
    Characteristic 2…
Input Specification
    Possible Types
        Type 1
        Type 2
    Connected to Component 1
        Spec 1
            Range Flag
            Name
            Unit
            Value (Max and Min)
            Description Text
        Spec 2…
    Connected to Component 2…
Output Specification
    Possible Types
        Type 1
        Type 2
    Connected to Component 1
        Spec 1
            Range Flag
            Name
            Unit
            Value (Max and Min)
            Description Text
        Spec 2…
    Connected to Component 2…
Component Vendor
    Name
    Location
        Street Name
        Street Number
        City
        State
        Country
        Zip
        Zip Extension
        Apt Number
        Description Text
    Contact Information
        Telephone
            Prefix
            Suffix
            Area Code
            Extension
            Type Code
            Foreign City Code
            Foreign Country Code
            Description Text
        Electronic Information
            Email
            URL
            Description Text

Not all fields are mandatory. The component type, component name, component optimizable characteristics and input or output specification is mandatory for the scope of this project in order to constrain a component by its specification and return the optimizable characteristics to the optimizing component See Appendix A.

The algorithm implemented by the Interface Constrainer Program is as follows:
For 1 to n constrains,
    Take each component,
        Keep it in the feasible set if,
        It is the component type specified by the constraint
        It's input or output specification (also specified by the constraint) is acceptable.

Take the compiled results and convert them to html using one of the stylesheets specified by the user at runtime (See Appendix F and G), the user can choose to display only the optimizable characteristics in a table or both the optimizable characteristics and the interface specifications that belong to each feasible component.

The user also has an option at runtime to bypass the program's intended functionality and simply display all the components in the database without providing any constraint. Then if the user wishes, (s)he may re-run the program after viewing the components currently inside the database (See Appendix D for the properties file and Appendix C for the Interface Constraint Query in XML).

The program is written using Java Standard Edition version 1.4.2. It uses the JDOM application programmer interface to handle XML documents as Java objects. JDOM also allows stylesheet transformations to occur in the Java environment with the use of an available XML parser provided in the third party lib folder of the program directory. The program uses the Log4j application programmer interface to output log statements which are configurable by the user at runtime using Log4j's tiered output feature and nested statements (debug, info, error statements).

6. OPTIMIZATION AND TRADEOFF ANALYSIS

In section 5 (Interface Constrainer Program), a set of 3 options for each of the 6 components has been identified. In such section, it has been ensured that these components can communicate among each other (i.e. they are compatible). Now the next step is to choose which option is the best for each component.

The process of choosing the optimal set of components was developed by using Excel and three routines in Visual Basic for Applications (VBA). The file where this process was implemented is [MultiobjectiveOptimization.xls].

The approach followed was the implementation of a discrete multi-objective optimization analysis incorporating the performance parameters defined in section 4.3 (identification of performance evaluation parameters). Those parameters identified as Major Criteria for Component Selection will become the design objectives:

1. Initial acquisition cost.
2. Durability of system.
3. Reliability of system.

Parameters identified as Criteria for Performance Evaluation were allocated to specific variables in Table 4.3. Now, Table 4.4 consolidates all design objectives and variables and relates each of them to the appropriate component. Note that variables in Table 4.4 have been represented as Level 4 requirements. Level 4 requirements will become the constraints for optimization process.

Figure 6.1 shows a screenshot from the file [MultiobjectiveOptimization.xls]. It contains the information for the component Heater, including each of the parameters (i.e. the concepts that correspond to Level 4 requirements) and design objectives.

Component:	**Heater**					
Parameters of interest:						
Parameter	**Units**	**Value Min**	**Value Max**	**Heater1**	**Heater2**	**Heater3**
Power - Output	*BTU*	*160,000.00*	*320,000.00*	*310,000.00*	*120,000.00*	*290,000.00*
Airflow	*CFM*	*3,000.00*	*4,000.00*	*3,500.00*	*3,100.00*	*3,750.00*
Efficiency	*%*	*75.00*	*100.00*	*94.00*	*99.00*	*92.00*
Cost	*USD*	*0.00*	*35,000.00*	*32,000.00*	*27,000.00*	*31,500.00*
Durability	*Hours*	*100,000.00*	*1,000,000.00*	*220,000.00*	*350,000.00*	*350,000.00*
Reliability	*%*	*98.00*	*100.00*	*99.50*	*99.70*	*99.15*
List validation ends here				[EndComponent]	[EndComponent]	[EndComponent]

Figure 6.1

Figure 6.2 to Figure 6.6 also show screenshots for each different component. They are all similar to Figure 6.1. Note that the tool implemented in Excel allows to play with most of the parameters dynamically reflecting any changes in most of the results of the analyses.

Changes that are supported include: changes in the magnitude of one or more parameters; the situation one for one or more components less than three alternatives are present (in such case, the corresponding values should be entered as zeroes); with relatively minor modifications in the VBA code and in the references and formulas in Excel, changes such as including more components or options for each one can also be adapted.

Component:	**Chiller**					
Parameters of interest:						
Parameter	**Units**	**Value Min**	**Value Max**	**Chiller1**	**Chiller2**	**Chiller3**
Power - Output	*BTU*	*200,000.00*	*350,000.00*	*335,000.00*	*340,000.00*	*300,000.00*
Airflow	*CFM*	*3,000.00*	*4,000.00*	*3,950.00*	*3,750.00*	*3,900.00*
Efficiency	*%*	*68.00*	*100.00*	*97.00*	*85.00*	*94.00*
Cost	*USD*	*0.00*	*28,000.00*	*25,000.00*	*27,500.00*	*18,000.00*
Durability	*Hours*	*100,000.00*	*1,000,000.00*	*540,000.00*	*670,000.00*	*550,000.00*
Reliability	*%*	*97.00*	*100.00*	*99.99*	*99.97*	*99.97*
List validation ends here				[EndComponent]	[EndComponent]	[EndComponent]

Figure 6.2

Component:	**Fan**					
Parameters of interest:						
Parameter	**Units**	**Value Min**	**Value Max**	**Fan1**	**Fan2**	**Fan3**
Airflow	*CFM*	*3,000.00*	*4,000.00*	*3,100.00*	*3,800.00*	*3,950.00*
Efficiency	*%*	*80.00*	*100.00*	*95.00*	*93.00*	*91.00*
Reliability	*%*	*98.00*	*100.00*	*99.78*	*99.98*	*99.85*
Cost	*USD*	*0.00*	*1,500.00*	*1,350.00*	*1,200.00*	*1,400.00*
Durability	*Hours*	*100,000.00*	*1,000,000.00*	*350,000.00*	*320,000.00*	*420,000.00*
List validation ends here				[EndCompone	[EndComponer	[EndComponent]

Figure 6.3

Component:	**PPCD**					
Parameters of interest:						
Parameter	**Units**	**Value Min**	**Value Max**	**PPCD1**	**PPCD2**	**PPCD3**
Reliability	*%*	*99.00*	*100.00*	*99.95*	*99.75*	*99.99*
Cost	*USD*	*0.00*	*5,000.00*	*4,500.00*	*4,240.00*	*4,700.00*
Durability	*Hours*	*100,000.00*	*1,000,000.00*	*500,000.00*	*478,000.00*	*650,000.00*
List validation ends here				[EndComponent]	[EndComponent]	[EndComponent]

Figure 6.4

Component:	**Controlling Device**					
Parameters of interest:						
Parameter	**Units**	**Value Min**	**Value Max**	**ControllingDevice1**	**ControllingDevice2**	**ControllingDevice3**
Reliability	*%*	*99.00*	*100.00*	*98.99*	*99.15*	*99.85*
Cost	*USD*	*0.00*	*25,000.00*	*23,000.00*	*25,300.00*	*23,850.00*
Durability	*Hours*	*100,000.00*	*1,000,000.00*	*450,000.00*	*800,000.00*	*510,000.00*
List validation ends here				[EndComponent]	[EndComponent]	[EndComponent]

Figure 6.5

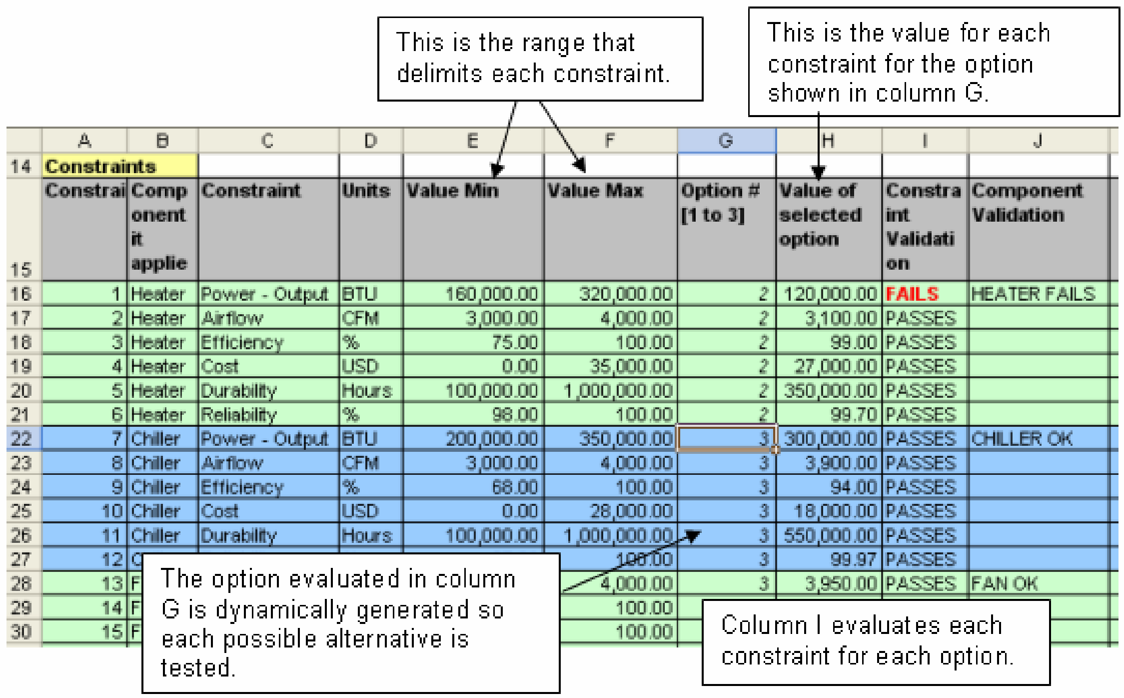
Component:	**Thermometer**					
Parameters of interest:						
Parameter	**Units**	**Value Min**	**Value Max**	**Thermometer1**	**Thermometer2**	**Thermometer3**
Temperature – Measure error	*Celsius*	*0.00*	*5.00*	*4.00*	*2.00*	*5.00*
Temperature – Measure range [Upp	*Celsius*	*60.00*	*70.00*	*65.00*	*67.00*	*58.00*
Temperature – Measure range [Low	*Celsius*	*-25.00*	*-20.00*	*-25.00*	*-25.00*	*-23.00*
Power – Supply required	*Volts*	*0.00*	*15.00*	*8.00*	*12.00*	*12.00*
Reliability	*%*	*98.00*	*100.00*	*99.95*	*99.98*	*99.55*
Cost	*USD*	*0.00*	*1,500.00*	*850.00*	*1,350.00*	*750.00*
Durability	*Hours*	*100,000.00*	*1,000,000.00*	*750,000.00*	*650,000.00*	*650,000.00*
List validation ends here				[EndComponent]	[EndComponent]	[EndComponent]

Figure 6.6

Figure 6.7 shows the contents of tab "OptimizationGrlFormulation" from the Excel file. This tab uses references to the tables containing the parameters for each component and consolidates all of them organized in such a manner that the three design objectives are put next to each other. Note however that each design objective shown here contains the value for each component individually. The ultimate goal is to select the best possible (non-inferior solution) combination of components.

	A	B	C	D	E	F	G	H
1	**Optimization General Formulation**							
2								
3	**Design Objectives**							
4	**Initial Acquisition Cost [USD]**				**Reliability [%]**			**Durability [Hours]**
5	**Component**	**Option1**	**Option2**	**Option3**	**Option1**	**Option2**	**Option3**	**Option1**
6	**Heater**	32,000.00	27,000.00	31,500.00	99.50	99.70	99.15	220,000.00
7	**Chiller**	25,000.00	27,500.00	18,000.00	99.99	99.97	99.97	540,000.00
8	**Fan**	1,350.00	1,200.00	1,400.00	99.78	99.98	99.85	350,000.00
9	**PCCD**	4,500.00	4,240.00	4,700.00	99.95	99.75	99.99	500,000.00
10	**ControllingDev**	23,000.00	25,300.00	23,850.00	98.99	99.15	99.85	450,000.00
11	**Thermometer**	850.00	1,350.00	750.00	99.95	99.98	99.55	750,000.00
12	**End of list of components**							

Figure 6.7

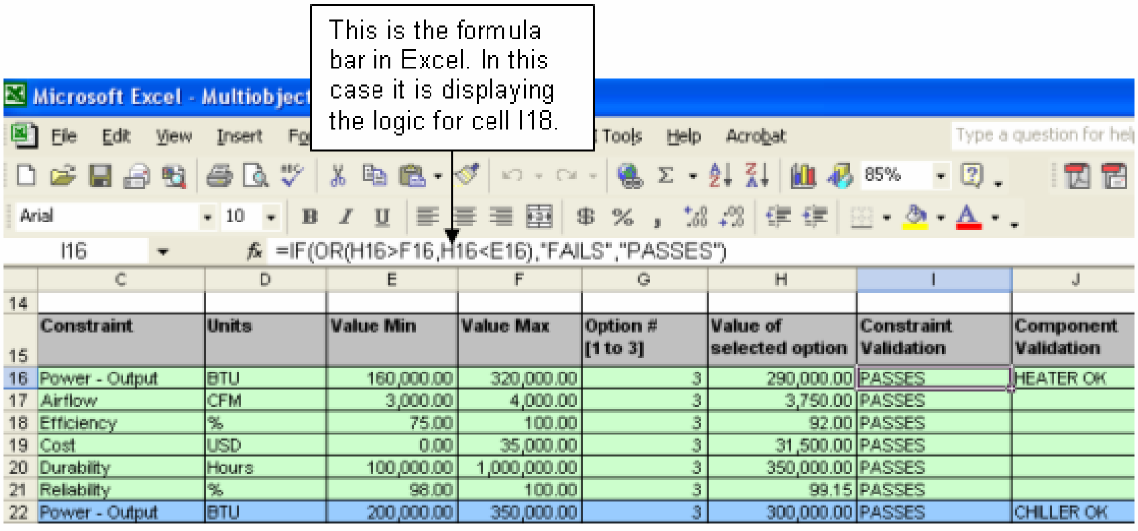


	A	B	C	D	E	F	G	H	I	J
14	Constraints									
15	Constrai	Component it applie	Constraint	Units	Value Min	Value Max	Option # [1 to 3]	Value of selected option	Constraint Validation	Component Validation
16	1	Heater	Power - Output	BTU	160,000.00	320,000.00	2	120,000.00	FAILS	HEATER FAILS
17	2	Heater	Airflow	CFM	3,000.00	4,000.00	2	3,100.00	PASSES	
18	3	Heater	Efficiency	%	75.00	100.00	2	99.00	PASSES	
19	4	Heater	Cost	USD	0.00	35,000.00	2	27,000.00	PASSES	
20	5	Heater	Durability	Hours	100,000.00	1,000,000.00	2	350,000.00	PASSES	
21	6	Heater	Reliability	%	98.00	100.00	2	99.70	PASSES	
22	7	Chiller	Power - Output	BTU	200,000.00	350,000.00	3	300,000.00	PASSES	CHILLER OK
23	8	Chiller	Airflow	CFM	3,000.00	4,000.00	3	3,900.00	PASSES	
24	9	Chiller	Efficiency	%	68.00	100.00	3	94.00	PASSES	
25	10	Chiller	Cost	USD	0.00	28,000.00	3	18,000.00	PASSES	
26	11	Chiller	Durability	Hours	100,000.00	1,000,000.00	3	550,000.00	PASSES	
27	12	C				100.00	3	99.97	PASSES	
28	13	F				4,000.00	3	3,950.00	PASSES	FAN OK
29	14	F				100.00				
30	15	F				100.00				

Figure 6.8

Figure 6.8 belongs to the same tab as Figure 6.7. This figure shows the list of constraints based on the Level 4 requirements. Each option for each component is evaluated against these constraints to determine if it fits inside or outside the feasible domain of solutions. In this case it can be observed how the Heater option 2 has a power output that falls below the minimum defined in column E, therefore it is rejected. Note that column I indicates if a specific constraint was violated and column J indicates the component that has to be rejected because one or more of its parameters do not pass the constraints associated to it.

The logic behind any cell in the Excel file can be appreciated by clicking on the appropriate cell. The formula or commands behind it will be displayed in the Formula Bar. Figure 6.9 shows an example of this.

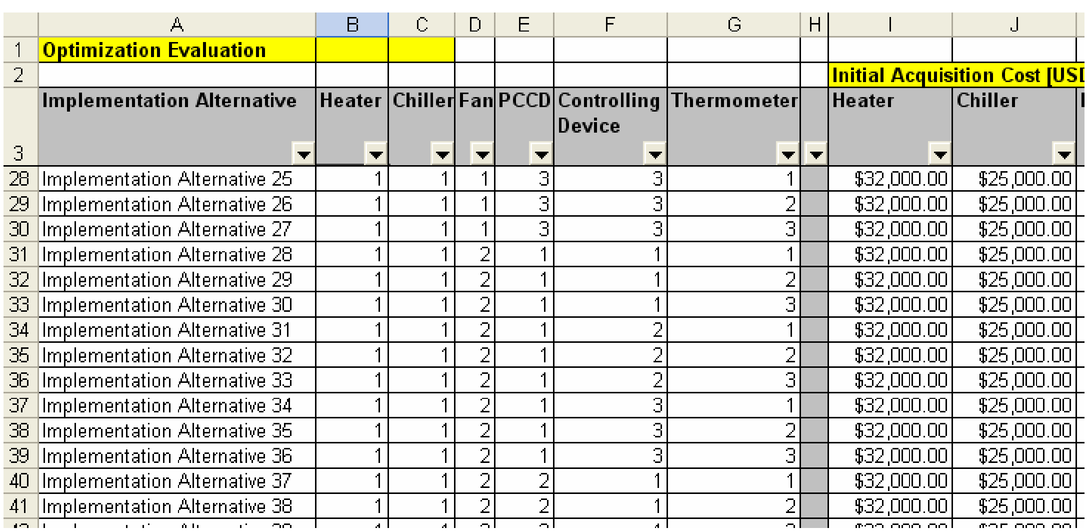


	Constraint	Units	Value Min	Value Max	Option # [1 to 3]	Value of selected option	Constraint Validation	Component Validation
16	Power - Output	BTU	160,000.00	320,000.00	3	290,000.00	PASSES	HEATER OK
17	Airflow	CFM	3,000.00	4,000.00	3	3,750.00	PASSES	
18	Efficiency	%	75.00	100.00	3	92.00	PASSES	
19	Cost	USD	0.00	35,000.00	3	31,500.00	PASSES	
20	Durability	Hours	100,000.00	1,000,000.00	3	350,000.00	PASSES	
21	Reliability	%	98.00	100.00	3	99.15	PASSES	
22	Power - Output	BTU	200,000.00	350,000.00	3	300,000.00	PASSES	CHILLER OK

Figure 6.9

In tab "OptimizationEvaluation" in the Excel file the tool will compute all possible implementation alternatives (since in this case there are 6 components and 3 options for each one, the total number of alternatives is 3^6 = 729). If this tab contains data and it is needed to reset it to perform a new calculation, hit "Ctrl + Shift + D" and a macro implemented in VBA will run and will clear everything.

Now, hit "Ctrl + Shift + O" and a macro called "AlternativesEval" will run. This macro generates all possible implementation alternatives and displays the results in tab "OptimizationEvaluation" for each of the design objectives for the whole system. Figure 6.10 shows a screenshot after running such macro. In this case the Initial Acquisition Cost is the only design objective visible, but the three of them are calculated and can be seen by scrolling to the right in Excel.



	Implementation Alternative	Heater	Chiller	Fan	PCCD	Controlling Device	Thermometer		Heater	Chiller
28	Implementation Alternative 25	1	1	1	3	3	1		$32,000.00	$25,000.00
29	Implementation Alternative 26	1	1	1	3	3	2		$32,000.00	$25,000.00
30	Implementation Alternative 27	1	1	1	3	3	3		$32,000.00	$25,000.00
31	Implementation Alternative 28	1	1	2	1	1	1		$32,000.00	$25,000.00
32	Implementation Alternative 29	1	1	2	1	1	2		$32,000.00	$25,000.00
33	Implementation Alternative 30	1	1	2	1	1	3		$32,000.00	$25,000.00
34	Implementation Alternative 31	1	1	2	1	2	1		$32,000.00	$25,000.00
35	Implementation Alternative 32	1	1	2	1	2	2		$32,000.00	$25,000.00
36	Implementation Alternative 33	1	1	2	1	2	3		$32,000.00	$25,000.00
37	Implementation Alternative 34	1	1	2	1	3	1		$32,000.00	$25,000.00
38	Implementation Alternative 35	1	1	2	1	3	2		$32,000.00	$25,000.00
39	Implementation Alternative 36	1	1	2	1	3	3		$32,000.00	$25,000.00
40	Implementation Alternative 37	1	1	2	2	1	1		$32,000.00	$25,000.00
41	Implementation Alternative 38	1	1	2	2	1	2		$32,000.00	$25,000.00

Figure 6.10

The formulas used to calculate each of the design objectives are explained in the following subsections.

Formula used to calculate the system cost (Initial Acquisition Cost):

InitialAcquisitionCost = CostHeater(option-i) + CostChiller(option-j) + CostFan(option-k) + CostControllingDevice(option-l) + CostPPCD(option-m) + CostThermometer(option-n)

Formula used to calculate the Reliability of the system:

The principle used for the reliability was: components that when fail prevent the whole system from operating have been modeled in series. Components that may not critically affect the performance of the system when a failure is present have been modeled as parallel components.

The reliability model used for the TCS is shown in Figure 6.11. The reason why the Heater and the Chiller were modeled in parallel is because about half of the time the weather requires the temperature to be reduced (summer, part of fall and spring) and about half of the time the weather required the temperature to be increased (winter and part of fall and spring). Therefore a failure in any of these two components do not necessarily prevents the system from working.

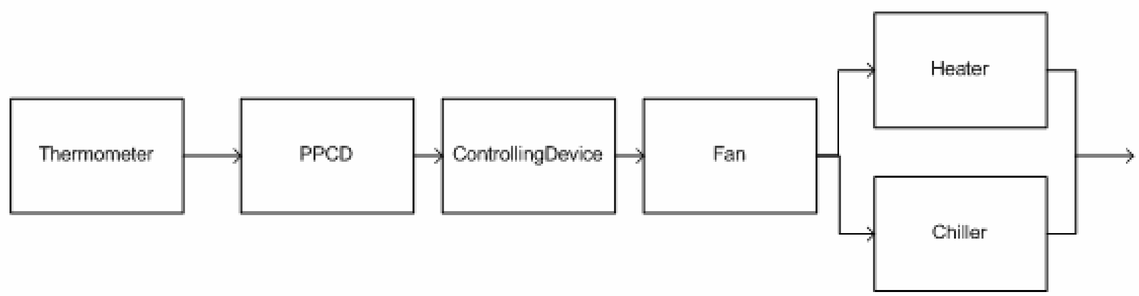


Figure 6.11

Based on this reliability model, the equation describing the reliability of the system based on the reliability of each individual component is:

*Rel = RelTher * RelPPCD * RelContDevice * RelFan * **{**1-(1-RelHeater)(1-RelChiller)**}***

Formula used to calculate the Durability of the system:

The system durability was calculated based on the durability of each individual component and its cost. The assumption is that if an expensive component fails, the overall durability of the system is reduced more dramatically than if a cheaper component fails and needs to be repaired or replaced. Therefore the formula is:

Durability = **{**(DurHeater x CostHeater) + (DurChiller x CostChiller) + (DurFan x CostFan) + (DurContrDevice x CostContDevice) + (DurPPCD x CostPPCD) + (DurTherm x CostTherm)**}** / TotCostofSystem

The Initial Acquisition Cost, Reliability and Durability for the total system after the macro “AlternativesEval” has run are in columns O, W and AE respectively. Note that none of these values has considered the constraints yet.

The charts showing the results in the performance space are in tabs “Chart - Cost Vs Reliability”, “Chart Cost Vs Durability” and “Chart-Reliability Vs Durability”.

Figure 6.12 shows one set of non-inferior solutions for the unconstrained chart of Cost Vs. Durability. Note how in this case the goal is to minimize system cost while maximizing system durability. For the purpose of this analysis, the cost will have a higher weight than the durability, therefore the leftmost set is selected (there are other non-inferior sets at the top of the chart, but in those the dominant objective is the durability).

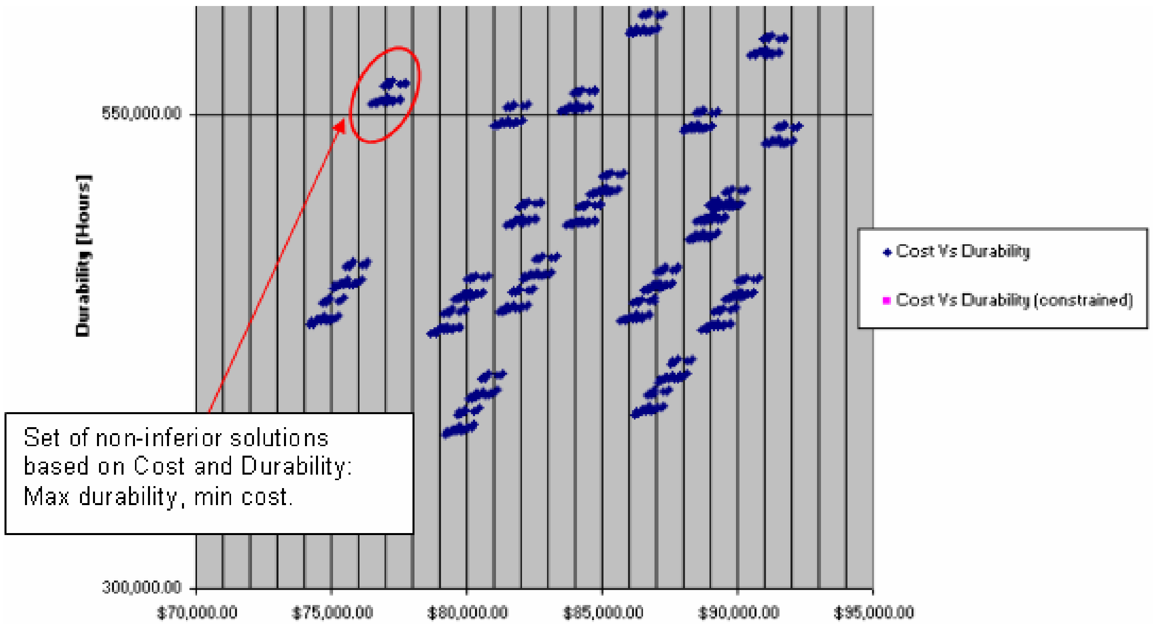
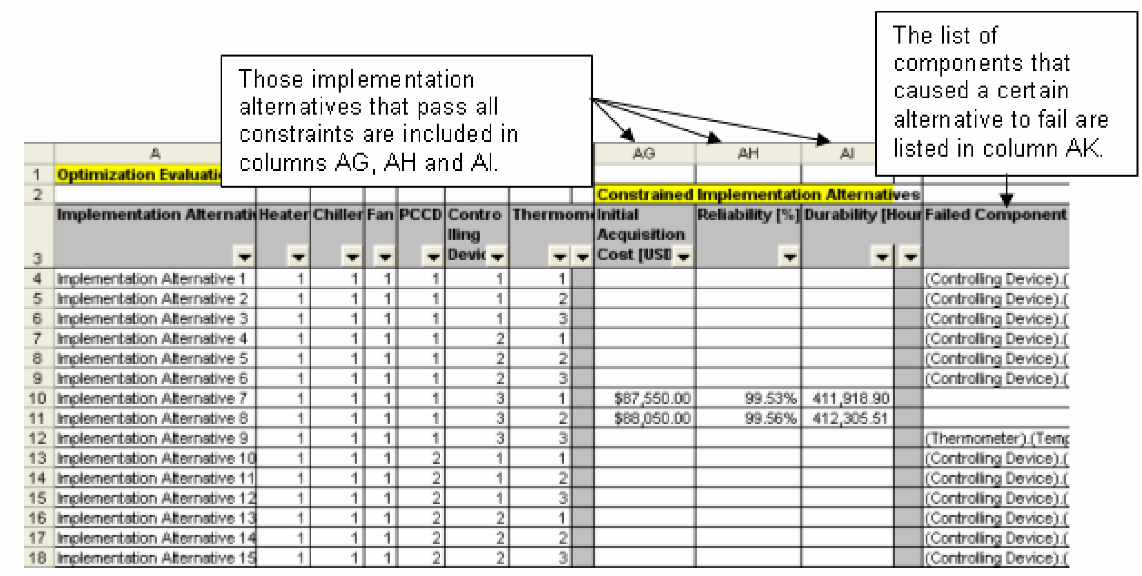


Figure 6.12

A similar analysis can be done for the other two charts showing the performance spaces for the remaining combinations of design objectives.

Now, the constraints will be incorporated into the analysis. In order to do this, the macro "ConstraintsEval" is run by hitting "Ctrl + Shift + C". Figure 6.13 shows the results from this macro and explains each of the columns that have been populated as a consequence of the routine run. Figure 6.14 shows the set of non-inferior solutions after the constrains have been considered. The points in pink represent those implementation alternatives that fall within the feasible domain. Note that this chart is the same one shown in Figure 6.12 (tab "Chart Cost Vs Durability").

After macro "ConstraintsEval" has run, tabs "Chart-Reliability Vs Durability" and "Chart - Cost Vs Reliability" will also include the constrained (pink) implementation alternatives.



	A							AG	AH	AI	
3	Implementation Alternati	Heater	Chiller	Fan	PCCD	Controlling Devic	Thermom	Initial Acquisition Cost [USD	Reliability [%]	Durability [Hour	Failed Component
4	Implementation Alternative 1	1	1	1	1	1	1				(Controlling Device).(
5	Implementation Alternative 2	1	1	1	1	1	2				(Controlling Device).(
6	Implementation Alternative 3	1	1	1	1	1	3				(Controlling Device).(
7	Implementation Alternative 4	1	1	1	1	2	1				(Controlling Device).(
8	Implementation Alternative 5	1	1	1	1	2	2				(Controlling Device).(
9	Implementation Alternative 6	1	1	1	1	2	3				(Controlling Device).(
10	Implementation Alternative 7	1	1	1	1	3	1	$87,550.00	99.53%	411,918.90	
11	Implementation Alternative 8	1	1	1	1	3	2	$88,050.00	99.56%	412,305.51	
12	Implementation Alternative 9	1	1	1	1	3	3				(Thermometer).(Temp
13	Implementation Alternative 10	1	1	1	2	1	1				(Controlling Device).(
14	Implementation Alternative 11	1	1	1	2	1	2				(Controlling Device).(
15	Implementation Alternative 12	1	1	1	2	1	3				(Controlling Device).(
16	Implementation Alternative 13	1	1	1	2	2	1				(Controlling Device).(
17	Implementation Alternative 14	1	1	1	2	2	2				(Controlling Device).(
18	Implementation Alternative 15	1	1	1	2	2	3				(Controlling Device).(

Figure 6.13

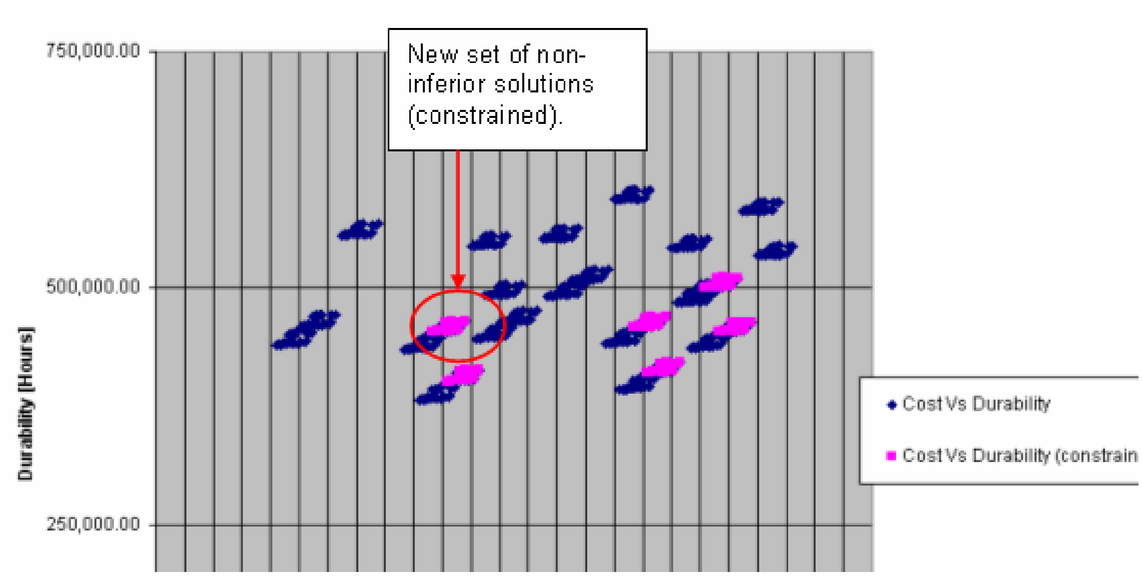


Figure 6.14

Figure 6.15 and Figure 6.16 show the sets of non-inferior solutions for the remaining charts. Each set of non-inferior solutions have been encircled.

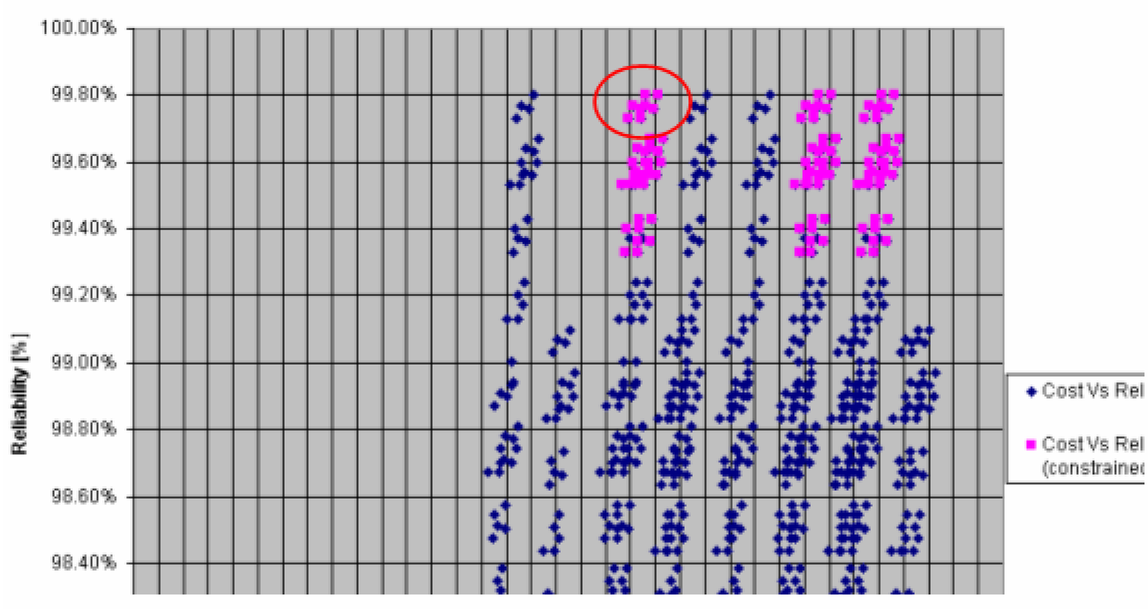


Figure 6.15

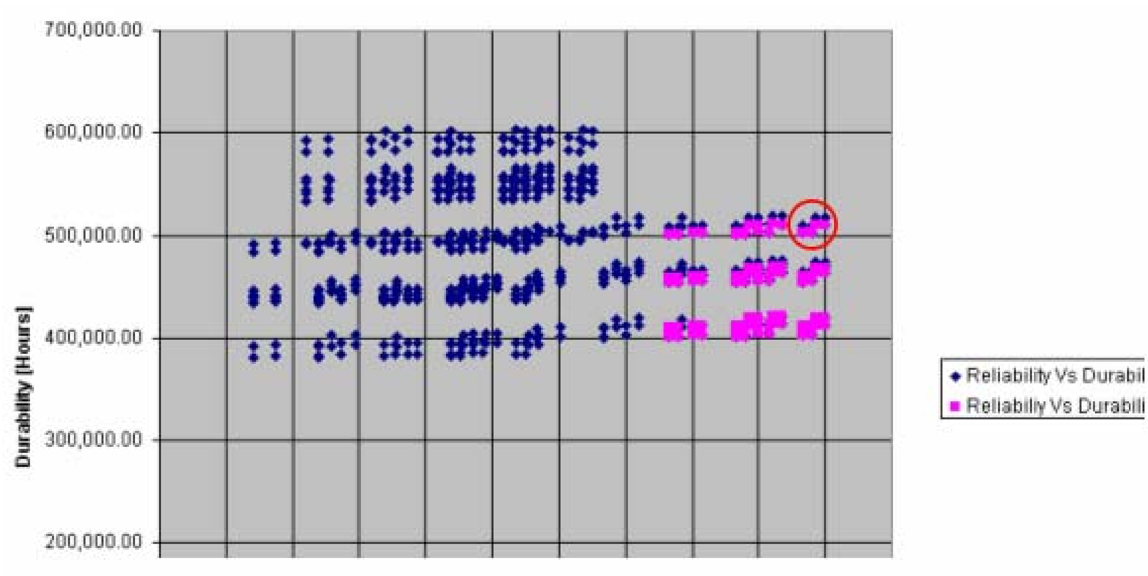


Figure 6.16

Based on the results observed in each chart, the best solutions from each one were compared in a separate table. The final goal is to find which one is the best option when considering the three design objectives: cost, durability and reliability. Figure 6.17 shows the final process of finding the optimal solution (tab "FinalResults" in the Excel file). Note how the weight for the cost is the highest one.

From this figure, the optimal solution is the one labeled as 700, this is:

Heater3, Chiller3, Fan2, PPCD3, ControllingDevice3 and Thermometer1

The design objectives for this implementation alternative are:

- Cost = $80,100.00
- Reliability = 99.77%
- Durability = 463,982.52 Hours.

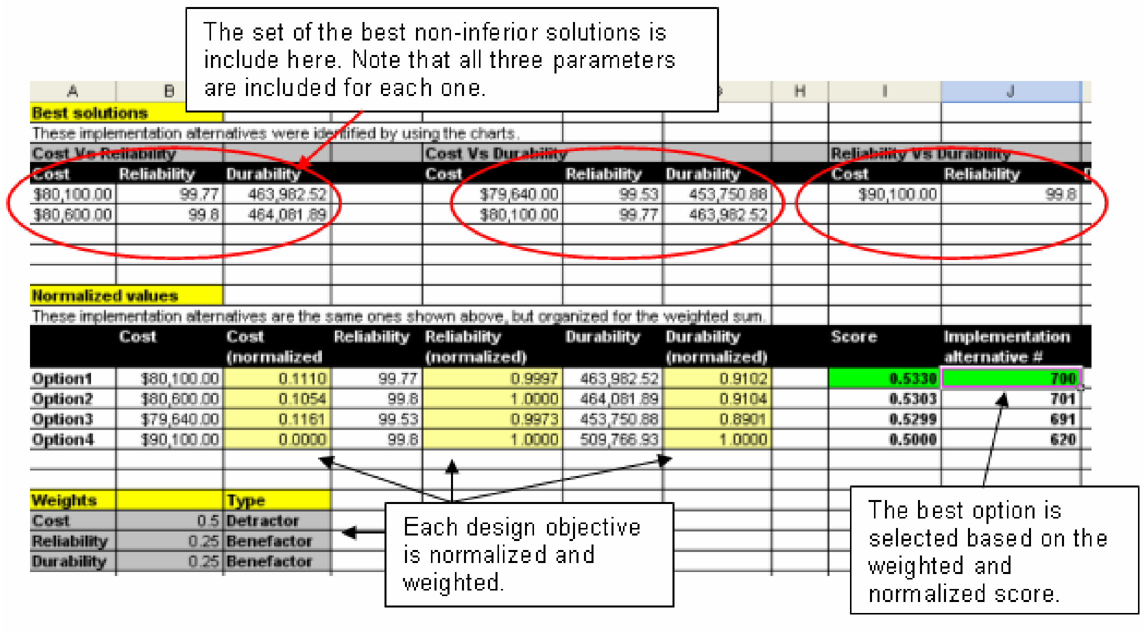


Figure 6.17

7. VBA CODE

The code behind each of the macros can be accessed by opening the Excel file [MultiobjectiveOptimization_VN.xls] and going to the menu “Tools→Macro→Macros” (this sequence of menus assumes Excel XP). This will display a window with the list of macros. Click on the one of interest and then click on “Edit”. Figure 7.1 shows this window.

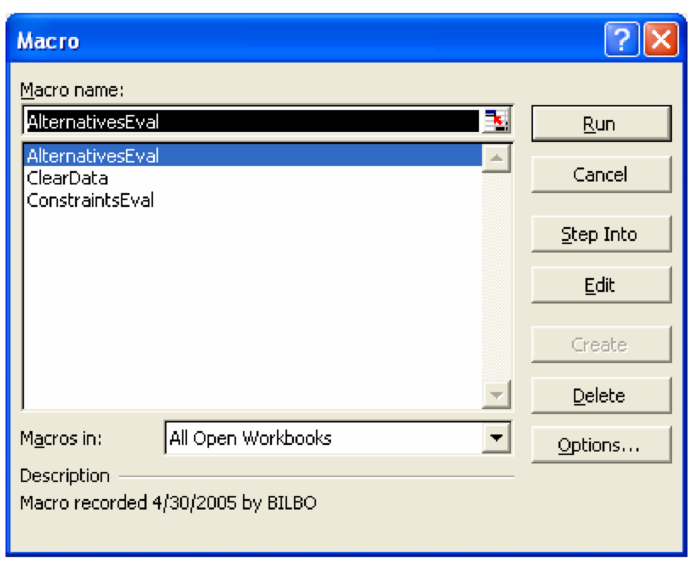


Figure 7.1

The code for each macro has been designed to be flexible and easy to understand and modify. Complete comments have been included. Figure 7.2 shows a screenshot of the VBA code.

```
(General)                                    AlternativesEval

Sub AlternativesEval()
'
' AlternativesEval Macro
' Macro developed by: Francisco Gallo, Josemaria Dolores
'
' Keyboard Shortcut: Ctrl+Shift+O
'

'****************************************
'DEFINITION OF PARAMETERS AND VARIABLES
'****************************************

'Number of components to be optimized
intNumComponents = 6
'Number of alternatives per component
intNumAlternatives = 3

'Sheet where the information for each component and constraints are
strSheetComponentsInfo = "OptimizationGrlFormulation"
'Sheet where the alternatives will be populated
strSheetAlternatives = "OptimizationEvaluation"

'++++++++++++++++++++++++++++++++++++++++++++++++++++
'Parameters for sheet assigned to strSheetAlternatives
```

Figure 7.2

8. REFERENCES

1. Austin, Mark. Lecture Notes for ENSE621: Systems Modeling and Analysis. Fall/2004 – University of Maryland.

2. de Neufville, Richard. Applied Systems Analysis. Massachusetts Institute of Technology.

3. Marsh, Adrian. System Level Design of a Combat Marking System. Spring 2003 – University of Maryland.

Controllers
http://www.cal-controls.com/products/tempover.html

Thermometers
http://www.analog.com/en/prod/0,,766_811_AD590%2C00.html
http://www.analog.com/en/subCat/0%2C2879%2C766%255F811%255F0%255F%255F0%255F%2C00.Html

Heaters Chillers and Fans used in XML Database
http://www.sterlinghvac.com/index.asp

Heaters
http://www.bush-nelson.com/products/eexdc.php
http://www.kennspenns.com/heating/wil%7EchmbrIn%7Edv.html
http://www.solarflo.com/

http://www.noblelight.net/ixir.html

Ventilation
http://www.pmengineer.com/CDA/ArticleInformation/coverstory/BNPCoverStoryItem/0,2730,143861,00.html

Global Spec Search Engine
http://www.globalspec.com/specifications/spechelpall?name=chillers_industrial&comp=1539

BTU Calculator
http://hearth.com/calc/btucalc.html
http://www.a-wall.com/ACcalc.php

9. APPENDIX A: ENTIRE XML DATABASE

TEMPERATURE CONTROL COMPONENT DATABASE OPTIMIZABLE SPECIFICATIONS

Controller: CALController 1200		
Attribute	**Value**	**Text Description**
Acquisition Cost(USD):	600000.00	The cost may vary on market prices.
Reliability(%):	75.00	Reliability is measured by internal testing departments.
Durability(hrs):	120000.00	Durability is a market specification provided by the Vendor.

Controller: CALController 3300		
Attribute	**Value**	**Text Description**
Acquisition Cost(USD):	23000.00	The cost may vary on market prices.
Reliability(%):	98.99	Reliability is measured by internal testing departments.
Durability(hrs):	450000.00	Durability is a market specification provided by the Vendor.

Controller: CALController 9300		
Attribute	**Value**	**Text Description**
Acquisition Cost(USD):	25300.00	The cost may vary on market prices.
Reliability(%):	99.15	Reliability is measured by internal testing departments.
Durability(hrs):	800000.00	Durability is a market specification provided by the Vendor.

Controller: CALController 9500		
Attribute	**Value**	**Text Description**
Acquisition Cost(USD):	23850.00	The cost may vary on market prices.
Reliability(%):	99.85	Reliability is measured by internal testing departments.
Durability(hrs):	510000.00	Durability is a market specification provided by the Vendor.

Heater: Sterling 250 TF		
Attribute	**Value**	**Text Description**
Power Output(BTU):	310000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3500	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	94.00	Efficiency Index is a market specification provided by the

		vendor.
Acquisition Cost(USD):	3200.00	The cost may vary on market prices.
Durability(hrs):	120000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.50	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	261	This Product is heavy.
Motor Strength(HP):	0.33	This product is strong.

Heater: Sterling 300 TF

Attribute	Value	Text Description
Power Output(BTU):	120000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3100	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	99.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	27000.00	The cost may vary on market prices.
Durability(hrs):	350000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.70	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	261	This Product is heavy.
Motor Strength(HP):	0.66	This product is strong.

Heater: Sterling 450 TF

Attribute	Value	Text Description
Power Output(BTU):	290000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3750	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	92.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	31500.00	The cost may vary on market prices.
Durability(hrs):	350000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.15	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	300	This Product is heavy.
Motor Strength(HP):	1	This product is strong.

Heater: Sterling 150 TF

Attribute	Value	Text Description
Power Output(BTU):	290000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3750	Cubic Feet per Minute is a standard used in industry.

Efficiency(%):	92.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	31500.00	The cost may vary on market prices.
Durability(hrs):	350000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.15	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	300	This Product is heavy.
Motor Strength(HP):	1	This product is strong.

Chiller: Sterling 090 AKV

Attribute	Value	Text Description
Power Output(BTU):	335000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3950	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	97.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	25000.00	The cost may vary on market prices.
Durability(hrs):	540000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.99	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	300	This Product is heavy.
Motor Strength(HP):	0.33	This product is strong.

Chiller: Sterling 100 AKV

Attribute	Value	Text Description
Power Output(BTU):	340000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3750	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	85.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	27500.00	The cost may vary on market prices.
Durability(hrs):	670000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.97	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	320	This Product is heavy.
Motor Strength(HP):	0.45	This product is strong.

Chiller: Sterling 120 AKV

Attribute	Value	Text Description
Power Output(BTU):	300000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air	3900	Cubic Feet per Minute is a standard used in industry.

Delivery(CFM):		
Efficiency(%):	94.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	18000.00	The cost may vary on market prices.
Durability(hrs):	550000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.97	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	400	This Product is heavy.
Motor Strength(HP):	1.5	This product is strong.

Chiller: Sterling 070 AKV

Attribute	Value	Text Description
Power Output(BTU):	300000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3900	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	94.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	18000.00	The cost may vary on market prices.
Durability(hrs):	550000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.97	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	400	This Product is heavy.
Motor Strength(HP):	1.5	This product is strong.

Fan: Sterling SHAT 2010

Attribute	Value	Text Description
Free Air Delivery(CFM):	3100	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	95.00	Efficiency Index is a market specification provided by the vendor.
Reliability(%):	99.78	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	1350.00	The cost may vary on market prices.
Durability(hrs):	350000.00	Durability is a market specification provided by the Vendor.

Fan: Sterling SHAT 2020

Attribute	Value	Text Description
Free Air Delivery(CFM):	3800	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	93.00	Efficiency Index is a market specification provided by the vendor.
Reliability(%):	99.98	Reliability Index is measured by internal testing departments.

Acquisition Cost(USD):	1200.00	The cost may vary on market prices.
Durability(hrs):	320000.00	Durability is a market specification provided by the Vendor.

Fan: Sterling SHAT 2030

Attribute	Value	Text Description
Free Air Delivery(CFM):	3950	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	91.00	Efficiency Index is a market specification provided by the vendor.
Reliability(%):	99.85	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	1400.00	The cost may vary on market prices.
Durability(hrs):	420000.00	Durability is a market specification provided by the Vendor.

Fan: Sterling SHAT 1900

Attribute	Value	Text Description
Free Air Delivery(CFM):	3950	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	91.00	Efficiency Index is a market specification provided by the vendor.
Reliability(%):	99.85	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	1400.00	The cost may vary on market prices.
Durability(hrs):	420000.00	Durability is a market specification provided by the Vendor.

Thermometer: Analog Devices TMP 17

Attribute	Value	Text Description
Temperature Measure Error(C):	4	N/A
Possible Temperature Sensed Maximum(C):	65	N/A
Possible Temperature Sensed Minimum(C):	-25	N/A
Power Consumption(V):	8	N/A
Reliability(%):	99.95	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	850.00	The cost may vary on market prices.
Durability(hrs):	750000.00	Durability is a market specification provided by the Vendor.

Thermometer: Analog Devices TMP 18

Attribute	Value	Text Description
Temperature Measure Error(C):	2	N/A
Possible Temperature Sensed Maximum(C):	67	N/A

Possible Temperature Sensed Minimum(C):	-25	N/A
Power Consumption(V):	12	N/A
Reliability(%):	99.98	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	1350.00	The cost may vary on market prices.
Durability(hrs):	650000.00	Durability is a market specification provided by the Vendor.

Thermometer: Analog Devices TMP 19		
Attribute	**Value**	**Text Description**
Temperature Measure Error(C):	5	N/A
Possible Temperature Sensed Maximum(C):	58	N/A
Possible Temperature Sensed Minimum(C):	-23	N/A
Power Consumption(V):	12	N/A
Reliability(%):	99.55	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	750.00	The cost may vary on market prices.
Durability(hrs):	650000.00	Durability is a market specification provided by the Vendor.

Thermometer: Analog Devices TMP 10		
Attribute	**Value**	**Text Description**
Temperature Measure Error(C):	4	N/A
Possible Temperature Sensed Maximum(C):	65	N/A
Possible Temperature Sensed Minimum(C):	-25	N/A
Power Consumption(V):	8	N/A
Reliability(%):	99.95	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	850.00	The cost may vary on market prices.
Durability(hrs):	750000.00	Durability is a market specification provided by the Vendor.

PPCD: Microsoft TempWiz 3.0		
Attribute	**Value**	**Text Description**
Reliability(%):	99.95	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	4500.00	The cost may vary on market prices.
Durability(hrs):	500000.00	Durability is a market specification provided by the Vendor.

PPCD: Microsoft TempWiz 3.3		
Attribute	**Value**	**Text Description**

Reliability(%):	99.75	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	4240.00	The cost may vary on market prices.
Durability(hrs):	478000.00	Durability is a market specification provided by the Vendor.

PPCD: Microsoft TempWiz 3.5		
Attribute	**Value**	**Text Description**
Reliability(%):	99.99	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	4700.00	The cost may vary on market prices.
Durability(hrs):	650000.00	Durability is a market specification provided by the Vendor.

PPCD: Microsoft TempWiz 2.0		
Attribute	**Value**	**Text Description**
Reliability(%):	89.95	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	3000.00	The cost may vary on market prices.
Durability(hrs):	400000.00	Durability is a market specification provided by the Vendor.

10. APPENDIX B: FEASIBLE SET AFTER QUERY CONSTRAINS THE DATABASE

FEASIBLE COMPONENT OPTIMIZABLE SPECIFICATIONS

Controller: CALController 3300		
Attribute	**Value**	**Text Description**
Acquisition Cost(USD):	23000.00	The cost may vary on market prices.
Reliability(%):	98.99	Reliability is measured by internal testing departments.
Durability(hrs):	450000.00	Durability is a market specification provided by the Vendor.

Controller: CALController 9300		
Attribute	**Value**	**Text Description**
Acquisition Cost(USD):	25300.00	The cost may vary on market prices.
Reliability(%):	99.15	Reliability is measured by internal testing departments.
Durability(hrs):	800000.00	Durability is a market specification provided by the Vendor.

Controller: CALController 9500		
Attribute	**Value**	**Text Description**
Acquisition Cost(USD):	23850.00	The cost may vary on market prices.
Reliability(%):	99.85	Reliability is measured by internal testing departments.
Durability(hrs):	510000.00	Durability is a market specification provided by the Vendor.

Heater: Sterling 250 TF		
Attribute	**Value**	**Text Description**

Power Output(BTU):	310000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3500	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	94.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	3200.00	The cost may vary on market prices.
Durability(hrs):	120000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.50	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	261	This Product is heavy.
Motor Strength(HP):	0.33	This product is strong.

Heater: Sterling 300 TF

Attribute	Value	Text Description
Power Output(BTU):	120000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3100	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	99.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	27000.00	The cost may vary on market prices.
Durability(hrs):	350000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.70	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	261	This Product is heavy.
Motor Strength(HP):	0.66	This product is strong.

Heater: Sterling 450 TF

Attribute	Value	Text Description
Power Output(BTU):	290000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3750	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	92.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	31500.00	The cost may vary on market prices.
Durability(hrs):	350000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.15	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	300	This Product is heavy.
Motor Strength(HP):	1	This product is strong.

Chiller: Sterling 090 AKV

Attribute	Value	Text Description
Power Output(BTU):	335000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3950	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	97.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	25000.00	The cost may vary on market prices.
Durability(hrs):	540000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.99	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	300	This Product is heavy.
Motor Strength(HP):	0.33	This product is strong.

Chiller: Sterling 100 AKV

Attribute	Value	Text Description
Power Output(BTU):	340000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3750	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	85.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	27500.00	The cost may vary on market prices.
Durability(hrs):	670000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.97	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	320	This Product is heavy.
Motor Strength(HP):	0.45	This product is strong.

Chiller: Sterling 120 AKV

Attribute	Value	Text Description
Power Output(BTU):	300000.00	This is the main index of measurement which differentiates between heaters in industry.
Free Air Delivery(CFM):	3900	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	94.00	Efficiency Index is a market specification provided by the vendor.
Acquisition Cost(USD):	18000.00	The cost may vary on market prices.
Durability(hrs):	550000.00	Durability is a market specification provided by the Vendor.
Reliability(%):	99.97	Reliability Index is measured by internal testing departments.
Shipping Weight(lbs):	400	This Product is heavy.
Motor Strength(HP):	1.5	This product is strong.

Fan: Sterling SHAT 2010

Attribute	Value	Text Description
Free Air Delivery(CFM):	3100	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	95.00	Efficiency Index is a market specification provided by the vendor.
Reliability(%):	99.78	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	1350.00	The cost may vary on market prices.
Durability(hrs):	350000.00	Durability is a market specification provided by the Vendor.

Fan: Sterling SHAT 2020

Attribute	Value	Text Description
Free Air Delivery(CFM):	3800	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	93.00	Efficiency Index is a market specification provided by the vendor.
Reliability(%):	99.98	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	1200.00	The cost may vary on market prices.
Durability(hrs):	320000.00	Durability is a market specification provided by the Vendor.

Fan: Sterling SHAT 2030

Attribute	Value	Text Description
Free Air Delivery(CFM):	3950	Cubic Feet per Minute is a standard used in industry.
Efficiency(%):	91.00	Efficiency Index is a market specification provided by the vendor.
Reliability(%):	99.85	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	1400.00	The cost may vary on market prices.
Durability(hrs):	420000.00	Durability is a market specification provided by the Vendor.

Thermometer: Analog Devices TMP 17

Attribute	Value	Text Description
Temperature Measure Error(C):	4	N/A
Possible Temperature Sensed Maximum(C):	65	N/A
Possible Temperature Sensed Minimum(C):	-25	N/A
Power Consumption(V):	8	N/A
Reliability(%):	99.95	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	850.00	The cost may vary on market prices.

Durability(hrs):	750000.00	Durability is a market specification provided by the Vendor.

Thermometer: Analog Devices TMP 18		
Attribute	**Value**	**Text Description**
Temperature Measure Error(C):	2	N/A
Possible Temperature Sensed Maximum(C):	67	N/A
Possible Temperature Sensed Minimum(C):	-25	N/A
Power Consumption(V):	12	N/A
Reliability(%):	99.98	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	1350.00	The cost may vary on market prices.
Durability(hrs):	650000.00	Durability is a market specification provided by the Vendor.

Thermometer: Analog Devices TMP 19		
Attribute	**Value**	**Text Description**
Temperature Measure Error(C):	5	N/A
Possible Temperature Sensed Maximum(C):	58	N/A
Possible Temperature Sensed Minimum(C):	-23	N/A
Power Consumption(V):	12	N/A
Reliability(%):	99.55	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	750.00	The cost may vary on market prices.
Durability(hrs):	650000.00	Durability is a market specification provided by the Vendor.

PPCD: Microsoft TempWiz 3.0		
Attribute	**Value**	**Text Description**
Reliability(%):	99.95	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	4500.00	The cost may vary on market prices.
Durability(hrs):	500000.00	Durability is a market specification provided by the Vendor.

PPCD: Microsoft TempWiz 3.3		
Attribute	**Value**	**Text Description**
Reliability(%):	99.75	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	4240.00	The cost may vary on market prices.
Durability(hrs):	478000.00	Durability is a market specification provided by the Vendor.

PPCD: Microsoft TempWiz 3.5

Attribute	Value	Text Description
Reliability(%):	99.99	Reliability Index is measured by internal testing departments.
Acquisition Cost(USD):	4700.00	The cost may vary on market prices.
Durability(hrs):	650000.00	Durability is a market specification provided by the Vendor.

11. APPENDIX C: INTERFACE CONSTRAINT QUERY


```
- <InterfaceConstraints>
- <InterfaceConstraint>
  <ComponentType>Controller</ComponentType>
  <ConstraintType>Input</ConstraintType>
  <ConnectingComponent>PPCD</ConnectingComponent>
- <Spec>
  <SpecName>Input Signal</SpecName>
  <SpecUnit abbreviation="V">Voltage</SpecUnit>
  <SpecValue>7</SpecValue>
  <SpecOperator>=</SpecOperator>
    </Spec>
    </InterfaceConstraint>
- <InterfaceConstraint>
  <ComponentType>Heater</ComponentType>
  <ConstraintType>Input</ConstraintType>
  <ConnectingComponent>Controller</ConnectingComponent>
- <Spec>
  <SpecName>Input Signal</SpecName>
  <SpecUnit abbreviation="V">Voltage</SpecUnit>
  <SpecValue>7</SpecValue>
  <SpecOperator>=</SpecOperator>
    </Spec>
    </InterfaceConstraint>
- <InterfaceConstraint>
  <ComponentType>Chiller</ComponentType>
  <ConstraintType>Input</ConstraintType>
  <ConnectingComponent>Controller</ConnectingComponent>
- <Spec>
  <SpecName>Input Signal</SpecName>
  <SpecUnit abbreviation="V">Voltage</SpecUnit>
  <SpecValue>7</SpecValue>
  <SpecOperator>=</SpecOperator>
    </Spec>
    </InterfaceConstraint>
- <InterfaceConstraint>
  <ComponentType>Fan</ComponentType>
  <ConstraintType>Input</ConstraintType>
```

```
  <ConnectingComponent>Chiller</ConnectingComponent>
- <Spec>
  <SpecName>Input Air Flow</SpecName>
  <SpecUnit abbreviation="CFM">Cubic Feet Per Minute</SpecUnit>
  <SpecValue>3500</SpecValue>
  <SpecOperator>></SpecOperator>
    </Spec>
    </InterfaceConstraint>
- <InterfaceConstraint>
  <ComponentType>Thermometer</ComponentType>
  <ConstraintType>Output</ConstraintType>
  <ConnectingComponent>Controller</ConnectingComponent>
- <Spec>
  <SpecName>Output Signal</SpecName>
  <SpecUnit abbreviation="V">Voltage</SpecUnit>
  <SpecValue>2</SpecValue>
  <SpecOperator>></SpecOperator>
    </Spec>
    </InterfaceConstraint>
- <InterfaceConstraint>
  <ComponentType>PPCD</ComponentType>
  <ConstraintType>Output</ConstraintType>
  <ConnectingComponent>Controller</ConnectingComponent>
- <Spec>
  <SpecName>Output Signal</SpecName>
  <SpecUnit abbreviation="V">Voltage</SpecUnit>
  <SpecValue>-6</SpecValue>
  <SpecOperator><</SpecOperator>
    </Spec>
    </InterfaceConstraint>
    </InterfaceConstraints>
```


12. APPENDIX D: CONFIGURABLE PROPERTIES FILE

```
# COPYRIGHT INFORMATION
#
# This material may not be copied, used, or viewed without the written
or verbal
# consent of Josemaria L. Dolores or an authorized proxy.
#
#This is a properties file for the TCS Component Interface Constrainer
class.
#It holds string literals and element names.
#By having a properties file, the system is in general, more
configurable.
#----------------------------------------------------------------------

#File Path Names
databseFilepathName=../xml/TemperatureControlComponentDatabase.xml
constraintQueryFilepathName=../xml/InterfaceConstraintQuery.xml
```

```
#------------Changeable values----------------------------------------
#--------------------------------------------------------------------
outputFilepathName=../html/FeasibleComponents.html
#outputFilepathName=../html/TemperatureControlComponentDatabase.html

#Use only one of the available rendering stylesheets.
#renderingStylesheetFilepathName=../xsl/DisplayInterfaceCharacteristics
.xsl
renderingStylesheetFilepathName=../xsl/DisplayOptimizableCharacteristic
s.xsl

#Which components to output: choose from the list of components below
and separate with a comma and no space
#Controller,PPCD,Heater,Chiller,Fan,Thermometer,All
whichComponentsToOutput=All

#Flag used to output entire database
wholeDatabaseFlag=false
#--------------------------------------------------------------------
#--------------------------------------------------------------------

#Namespace Information
namespacePrefix=tcs
namespaceQualifiedName=http://www.wam.umd.edu/~fgallo/ENPM642/Project/T
CSNamespace

#Component Types
Controller=Controller
Heater=Heater
PPCD=PPCD
Chiller=Chiller
Fan=Fan
Thermometer=Thermometer
All=All

#Element Names
ComponentType=ComponentType
ComponentLibrary=ComponentLibrary
ConstraintType=ConstraintType
ConnectingComponent=ConnectingComponent
Spec=Spec
SpecName=SpecName
SpecValue=SpecValue
SpecOperator=SpecOperator
SpecUnit=SpecUnit
abbreviation=abbreviation
ComponentName=ComponentName
Input=Input
ComponentInputSpecification=ComponentInputSpecification
Output=Output
ComponentOutputSpecification=ComponentOutputSpecification
ConnectedToComponent=ConnectedToComponent
PossibleComponentTypes=PossibleComponentTypes
Range=Range
```

Appendix E: Output Logger File

```
INFO - ____________________NEW CONSTRAINT______________________
INFO - Constraint number 1: The Controller component's Input Signal
must be = 7V when it is connected to the PPCD.
INFO - Evaluating the component: CALController 1200
INFO - Looking at the input specs.
INFO - Looking at the spec when this Controller is connected to a PPCD.
INFO - Looking at the Controller's Input Signal specification.
INFO - The max value for this component's Input Signal is: 5.0.
INFO - The min value for this component's Input Signal is: -5.0.
INFO - Discard this component.
INFO - Evaluating the component: CALController 3300
INFO - Looking at the input specs.
INFO - Looking at the spec when this Controller is connected to a PPCD.
INFO - Looking at the Controller's Input Signal specification.
INFO - The max value for this component's Input Signal is: 25.0.
INFO - The min value for this component's Input Signal is: -25.0.
INFO - Keep this component.
INFO - Evaluating the component: CALController 9300
INFO - Looking at the input specs.
INFO - Looking at the spec when this Controller is connected to a PPCD.
INFO - Looking at the Controller's Input Signal specification.
INFO - The max value for this component's Input Signal is: 25.0.
INFO - The min value for this component's Input Signal is: -25.0.
INFO - Keep this component.
INFO - Evaluating the component: CALController 9500
INFO - Looking at the input specs.
INFO - Looking at the spec when this Controller is connected to a PPCD.
INFO - Looking at the Controller's Input Signal specification.
INFO - The max value for this component's Input Signal is: 25.0.
INFO - The min value for this component's Input Signal is: -25.0.
INFO - Keep this component.
INFO - ____________________NEW CONSTRAINT______________________
INFO - Constraint number 2: The Heater component's Input Signal must be
= 7V when it is connected to the Controller.
INFO - Evaluating the component: Sterling 250 TF
INFO - Looking at the input specs.
INFO - Looking at the spec when this Heater is connected to a
Controller.
INFO - Looking at the Heater's Input Signal specification.
INFO - The max value for this component's Input Signal is: 25.0.
INFO - The min value for this component's Input Signal is: -25.0.
INFO - Keep this component.
INFO - Evaluating the component: Sterling 300 TF
INFO - Looking at the input specs.
INFO - Looking at the spec when this Heater is connected to a
Controller.
INFO - Looking at the Heater's Input Signal specification.
INFO - The max value for this component's Input Signal is: 10.0.
INFO - The min value for this component's Input Signal is: -10.0.
INFO - Keep this component.
INFO - Evaluating the component: Sterling 450 TF
INFO - Looking at the input specs.
INFO - Looking at the spec when this Heater is connected to a
Controller.
INFO - Looking at the Heater's Input Signal specification.
INFO - The max value for this component's Input Signal is: 35.0.
INFO - The min value for this component's Input Signal is: -35.0.
```

```
INFO - Keep this component.
INFO - Evaluating the component: Sterling 150 TF
INFO - Looking at the input specs.
INFO - Looking at the spec when this Heater is connected to a
Controller.
INFO - Looking at the Heater's Input Signal specification.
INFO - The max value for this component's Input Signal is: 5.0.
INFO - The min value for this component's Input Signal is: -5.0.
INFO - Discard this component.
INFO - ____________________NEW CONSTRAINT______________________
INFO - Constraint number 3: The Chiller component's Input Signal must
be = 7V when it is connected to the Controller.
INFO - Evaluating the component: Sterling 090 AKV
INFO - Looking at the input specs.
INFO - Looking at the spec when this Chiller is connected to a
Controller.
INFO - Looking at the Chiller's Input Signal specification.
INFO - The max value for this component's Input Signal is: 25.0.
INFO - The min value for this component's Input Signal is: -25.0.
INFO - Keep this component.
INFO - Evaluating the component: Sterling 100 AKV
INFO - Looking at the input specs.
INFO - Looking at the spec when this Chiller is connected to a
Controller.
INFO - Looking at the Chiller's Input Signal specification.
INFO - The max value for this component's Input Signal is: 10.0.
INFO - The min value for this component's Input Signal is: -10.0.
INFO - Keep this component.
INFO - Evaluating the component: Sterling 120 AKV
INFO - Looking at the input specs.
INFO - Looking at the spec when this Chiller is connected to a
Controller.
INFO - Looking at the Chiller's Input Signal specification.
INFO - The max value for this component's Input Signal is: 35.0.
INFO - The min value for this component's Input Signal is: -35.0.
INFO - Keep this component.
INFO - Evaluating the component: Sterling 070 AKV
INFO - Looking at the input specs.
INFO - Looking at the spec when this Chiller is connected to a
Controller.
INFO - Looking at the Chiller's Input Signal specification.
INFO - The max value for this component's Input Signal is: 4.0.
INFO - The min value for this component's Input Signal is: -4.0.
INFO - Discard this component.
INFO - ____________________NEW CONSTRAINT______________________
INFO - Constraint number 4: The Fan component's Input Air Flow must be
> 3500CFM when it is connected to the Chiller.
INFO - Evaluating the component: Sterling SHAT 2010
INFO - Looking at the input specs.
INFO - Looking at the spec when this Fan is connected to a Chiller.
INFO - Looking at the Fan's Input Air Flow specification.
INFO - The max value for this component's Input Air Flow is: 4000.0.
INFO - The min value for this component's Input Air Flow is: 0.0.
INFO - Keep this component.
INFO - Evaluating the component: Sterling SHAT 2020
INFO - Looking at the input specs.
INFO - Looking at the spec when this Fan is connected to a Chiller.
```

```
INFO - Looking at the Fan's Input Air Flow specification.
INFO - The max value for this component's Input Air Flow is: 4200.0.
INFO - The min value for this component's Input Air Flow is: 0.0.
INFO - Keep this component.
INFO - Evaluating the component: Sterling SHAT 2030
INFO - Looking at the input specs.
INFO - Looking at the spec when this Fan is connected to a Chiller.
INFO - Looking at the Fan's Input Air Flow specification.
INFO - The max value for this component's Input Air Flow is: 4300.0.
INFO - The min value for this component's Input Air Flow is: 0.0.
INFO - Keep this component.
INFO - Evaluating the component: Sterling SHAT 1900
INFO - Looking at the input specs.
INFO - Looking at the spec when this Fan is connected to a Chiller.
INFO - Looking at the Fan's Input Air Flow specification.
INFO - The max value for this component's Input Air Flow is: 3000.0.
INFO - The min value for this component's Input Air Flow is: 0.0.
INFO - Discard this component.
INFO - ____________________NEW CONSTRAINT______________________
INFO - Constraint number 5: The Thermometer component's Output Signal
must be > 2V when it is connected to the Controller.
INFO - Evaluating the component: Analog Devices TMP 17
INFO - Looking at the output specs.
INFO - This Thermometer can output to the following components:
Controller.
INFO - Looking at the spec when this Thermometer is connected to a
Controller.
INFO - Looking at the Thermometer's Output Signal specification.
INFO - The max value for this component's Output Signal is: 5.0.
INFO - The min value for this component's Output Signal is: -5.0.
INFO - Keep this component.
INFO - Evaluating the component: Analog Devices TMP 18
INFO - Looking at the output specs.
INFO - This Thermometer can output to the following components:
Controller.
INFO - Looking at the spec when this Thermometer is connected to a
Controller.
INFO - Looking at the Thermometer's Output Signal specification.
INFO - The max value for this component's Output Signal is: 7.0.
INFO - The min value for this component's Output Signal is: -7.0.
INFO - Keep this component.
INFO - Evaluating the component: Analog Devices TMP 19
INFO - Looking at the output specs.
INFO - This Thermometer can output to the following components:
Controller.
INFO - Looking at the spec when this Thermometer is connected to a
Controller.
INFO - Looking at the Thermometer's Output Signal specification.
INFO - The max value for this component's Output Signal is: 5.0.
INFO - The min value for this component's Output Signal is: -5.0.
INFO - Keep this component.
INFO - Evaluating the component: Analog Devices TMP 10
INFO - Looking at the output specs.
INFO - This Thermometer can output to the following components:
Controller.
INFO - Looking at the spec when this Thermometer is connected to a
Controller.
```

```
INFO - Looking at the Thermometer's Output Signal specification.
INFO - The max value for this component's Output Signal is: 1.0.
INFO - The min value for this component's Output Signal is: -1.0.
INFO - Discard this component.
INFO - ____________________NEW CONSTRAINT______________________
INFO - Constraint number 6: The PPCD component's Output Signal must be
< -6V when it is connected to the Controller.
INFO - Evaluating the component: Microsoft TempWiz 3.0
INFO - Looking at the output specs.
INFO - This PPCD can output to the following components: Controller.
INFO - Looking at the spec when this PPCD is connected to a Controller.
INFO - Looking at the PPCD's Output Signal specification.
INFO - The max value for this component's Output Signal is: -25.0.
INFO - The min value for this component's Output Signal is: -25.0.
INFO - Keep this component.
INFO - Evaluating the component: Microsoft TempWiz 3.3
INFO - Looking at the output specs.
INFO - This PPCD can output to the following components: Controller.
INFO - Looking at the spec when this PPCD is connected to a Controller.
INFO - Looking at the PPCD's Output Signal specification.
INFO - The max value for this component's Output Signal is: -35.0.
INFO - The min value for this component's Output Signal is: -35.0.
INFO - Keep this component.
INFO - Evaluating the component: Microsoft TempWiz 3.5
INFO - Looking at the output specs.
INFO - This PPCD can output to the following components: Controller.
INFO - Looking at the spec when this PPCD is connected to a Controller.
INFO - Looking at the PPCD's Output Signal specification.
INFO - The max value for this component's Output Signal is: -45.0.
INFO - The min value for this component's Output Signal is: -45.0.
INFO - Keep this component.
INFO - Evaluating the component: Microsoft TempWiz 2.0
INFO - Looking at the output specs.
INFO - This PPCD can output to the following components: Controller.
INFO - Looking at the spec when this PPCD is connected to a Controller.
INFO - Looking at the PPCD's Output Signal specification.
INFO - The max value for this component's Output Signal is: 5.0.
INFO - The min value for this component's Output Signal is: -5.0.
INFO - Discard this component.
INFO - ____________________See the HTML file.______________________
INFO - Display All components.
```

13. APPENDIX F: RENDERING STYLESHEET WHICH SHOWS BOTH INTERFACE SPECIFICATION AND OPTIMIZABLE CHARACTERISTICS

```
<?xml version="1.0" encoding="ISO-8859-1" ?>
- <xsl:stylesheet version="1.1"
xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
xmlns:tcs="http://www.wam.umd.edu/~fgallo/ENPM642/Project/TCSName
space">
- <!--
The purpose of this stylesheet is to take the feasible set of XML nodes
and display them neatly in an HTML table with the interface
characteristics included.
```

```
    -->
  - <!--
   Set the output type.
    -->
  <xsl:output method="html" />
  - <!--
   Match the root node.
    -->
- <xsl:template match="/">
- <xsl:element name="html">
- <xsl:element name="body">
  - <!--
   Create a header in the output document with the same text as the title.
    -->
- <xsl:element name="h1">
   <xsl:value-of select="'Feasible Component Optimizable Specifications with
    Interface Specifications'" />
    </xsl:element>
  <xsl:apply-templates select="tcs:ComponentLibrary" />
    </xsl:element>
    </xsl:element>
    </xsl:template>
  - <!--
   Match the components in order of their type so the output document is
    sorted in order of type.
    -->
- <xsl:template match="tcs:ComponentLibrary">
   <xsl:apply-templates select="tcs:Component[tcs:ComponentType='Controller']"
    />
  <xsl:apply-templates select="tcs:Component[tcs:ComponentType='Heater']" />
  <xsl:apply-templates select="tcs:Component[tcs:ComponentType='Chiller']" />
  <xsl:apply-templates select="tcs:Component[tcs:ComponentType='Fan']" />
                                                          <xsl:apply-templates
    select="tcs:Component[tcs:ComponentType='Thermometer']" />
  <xsl:apply-templates select="tcs:Component[tcs:ComponentType='PPCD']" />
    </xsl:template>
  - <!--
   Create a table for each component with a Caption holding the type and
    name of the component.
    -->
- <xsl:template match="tcs:Component">
- <xsl:element name="table">
  <xsl:attribute name="border">3</xsl:attribute>
- <xsl:element name="caption">
- <xsl:element name="b">
  <xsl:value-of select="tcs:ComponentType" />
  <xsl:text>:</xsl:text>
  <xsl:value-of select="tcs:ComponentName" />
    </xsl:element>
    </xsl:element>
- <xsl:element name="tr">
```

```
- <xsl:element name="th">
  <xsl:text>Attribute</xsl:text>
  </xsl:element>
- <xsl:element name="th">
  <xsl:text>Value</xsl:text>
  </xsl:element>
- <xsl:element name="th">
  <xsl:text>Text Description</xsl:text>
  </xsl:element>
  </xsl:element>
  <xsl:apply-templates select="tcs:ComponentOptimizableCharacteristics" />
  </xsl:element>
- <!--
  Separate each table with a line. The <br /> tag does not seem to work.
  -->
  <xsl:element name="p" />
- <xsl:element name="table">
  <xsl:attribute name="border">3</xsl:attribute>
- <xsl:element name="tr">
- <xsl:element name="th">
- <xsl:element name="u">
  <xsl:text>Input Specification</xsl:text>
  </xsl:element>
  </xsl:element>
- <xsl:element name="th">
- <xsl:element name="u">
  <xsl:text>Value</xsl:text>
  </xsl:element>
  </xsl:element>
- <xsl:element name="th">
- <xsl:element name="u">
  <xsl:text>Text Description</xsl:text>
  </xsl:element>
  </xsl:element>
  </xsl:element>
  <xsl:apply-templates select="tcs:ComponentInputSpecification" />
- <xsl:element name="tr">
- <xsl:element name="th">
- <xsl:element name="u">
  <xsl:text>Output Specification</xsl:text>
  </xsl:element>
  </xsl:element>
- <xsl:element name="th">
- <xsl:element name="u">
  <xsl:text>Value</xsl:text>
  </xsl:element>
  </xsl:element>
- <xsl:element name="th">
- <xsl:element name="u">
```

```
<xsl:text>Text Description</xsl:text>
  </xsl:element>
  </xsl:element>
  </xsl:element>
<xsl:apply-templates select="tcs:ComponentOutputSpecification" />
  </xsl:element>
- <!--
 Separate each table with a line. The <br /> tag does not seem to work.
  -->
<xsl:element name="p" />
  </xsl:template>
- <!--
 Populate the table with the Characteristics outlined.
  -->
- <xsl:template match="tcs:ComponentOptimizableCharacteristics">
<xsl:apply-templates select="tcs:Characteristic" />
  </xsl:template>
- <!--
 Populate the table with the Specifications of each connected input
  component.
  -->
- <xsl:template match="tcs:ComponentInputSpecification">
<xsl:apply-templates select="tcs:ConnectedToComponent" />
  </xsl:template>
- <!--
 Populate the table with the Specifications of each connected output
  component.
  -->
- <xsl:template match="tcs:ComponentOutputSpecification">
<xsl:apply-templates select="tcs:ConnectedToComponent" />
  </xsl:template>
- <!--
 Populate the table with the Specifications of each connected component.
  -->
- <xsl:template match="tcs:ConnectedToComponent">
- <xsl:element name="tr">
- <xsl:element name="th">
<xsl:text>When Connected to:</xsl:text>
<xsl:value-of select="tcs:ComponentType" />
  </xsl:element>
  </xsl:element>
<xsl:apply-templates select="tcs:Spec" />
  </xsl:template>
- <!--
 Use the Specification Name, Unit Value, and Text Description to populate
  the table.
  -->
- <xsl:template match="tcs:Spec">
- <xsl:element name="tr">
- <xsl:element name="td">
<xsl:value-of select="tcs:SpecName" />
```

```
- <xsl:if test="tcs:SpecUnit/@tcs:abbreviation!='null'">
  <xsl:text>(</xsl:text>
  <xsl:value-of select="tcs:SpecUnit/@tcs:abbreviation" />
  <xsl:text>)</xsl:text>
    </xsl:if>
  <xsl:text>:</xsl:text>
    </xsl:element>
- <xsl:choose>
- <xsl:when test="tcs:Range='Y'">
- <xsl:element name="td">
  <xsl:value-of select="tcs:SpecMinValue" />
  <xsl:text>to</xsl:text>
  <xsl:value-of select="tcs:SpecMaxValue" />
    </xsl:element>
    </xsl:when>
- <xsl:otherwise>
- <xsl:element name="td">
  <xsl:value-of select="tcs:SpecValue" />
    </xsl:element>
    </xsl:otherwise>
    </xsl:choose>
- <xsl:choose>
- <xsl:when test="tcs:SpecDescriptionText!='null'">
- <xsl:element name="td">
  <xsl:value-of select="tcs:SpecDescriptionText" />
    </xsl:element>
    </xsl:when>
- <xsl:otherwise>
- <xsl:element name="td">
  <xsl:text>N/A</xsl:text>
    </xsl:element>
    </xsl:otherwise>
    </xsl:choose>
    </xsl:element>
    </xsl:template>
  - <!--
   Use the Characteristic Name, Unit Value, and Text Description to
    populate the table.
    -->
- <xsl:template match="tcs:Characteristic">
- <xsl:element name="tr">
- <xsl:element name="th">
  <xsl:value-of select="tcs:CharacteristicName" />
- <xsl:if test="tcs:CharacteristicUnit/@tcs:abbreviation!='null'">
  <xsl:text>(</xsl:text>
  <xsl:value-of select="tcs:CharacteristicUnit/@tcs:abbreviation" />
  <xsl:text>)</xsl:text>
    </xsl:if>
  <xsl:text>:</xsl:text>
```

```
    </xsl:element>
- <xsl:element name="td">
  <xsl:value-of select="tcs:CharacteristicValue" />
    </xsl:element>
- <xsl:choose>
- <xsl:when test="tcs:CharacteristicDescriptionText!='null'">
- <xsl:element name="td">
  <xsl:value-of select="tcs:CharacteristicDescriptionText" />
    </xsl:element>
    </xsl:when>
- <xsl:otherwise>
- <xsl:element name="td">
  <xsl:text>N/A</xsl:text>
    </xsl:element>
    </xsl:otherwise>
    </xsl:choose>
    </xsl:element>
    </xsl:template>
    </xsl:stylesheet>
```

14. APPENDIX G: RENDERING STYLESHEET WHICH SHOWS ONLY OPTIMIZABLE CHARACTERISTICS

```
  <?xml version="1.0" encoding="ISO-8859-1" ?>
- <xsl:stylesheet version="1.1"
  xmlns:xsl="http://www.w3.org/1999/XSL/Transform"
  xmlns:tcs="http://www.wam.umd.edu/~fgallo/ENPM642/Project/TCSName
  space">
- <!--
 The purpose of this stylesheet is to take the feasible set of XML nodes
  and display them neatly in an HTML table.
  -->
- <!--
 Set the output type.
  -->
  <xsl:output method="html" />
- <!--
 Match the root node.
  -->
- <xsl:template match="/">
- <xsl:element name="html">
- <xsl:element name="body">
- <!--
 Create a header in the output document with the same text as the title.
  -->
- <xsl:element name="h1">
  <xsl:value-of select="'Feasible Component Optimizable Specifications'" />
    </xsl:element>
  <xsl:apply-templates select="tcs:ComponentLibrary" />
```

```
    </xsl:element>
    </xsl:element>
    </xsl:template>
  - <!--
   Match the components in order of their type so the output document is
   sorted in order of type.
   -->
- <xsl:template match="tcs:ComponentLibrary">
   <xsl:apply-templates  select="tcs:Component[tcs:ComponentType='Controller']"
   />
  <xsl:apply-templates select="tcs:Component[tcs:ComponentType='Heater']" />
  <xsl:apply-templates select="tcs:Component[tcs:ComponentType='Chiller']" />
  <xsl:apply-templates select="tcs:Component[tcs:ComponentType='Fan']" />
                                                          <xsl:apply-templates
   select="tcs:Component[tcs:ComponentType='Thermometer']" />
  <xsl:apply-templates select="tcs:Component[tcs:ComponentType='PPCD']" />
    </xsl:template>
  - <!--
   Create a table for each component with a Caption holding the type and
   name of the component.
   -->
- <xsl:template match="tcs:Component">
- <xsl:element name="table">
  <xsl:attribute name="border">1</xsl:attribute>
- <xsl:element name="caption">
- <xsl:element name="b">
  <xsl:value-of select="tcs:ComponentType" />
  <xsl:text>:</xsl:text>
  <xsl:value-of select="tcs:ComponentName" />
    </xsl:element>
    </xsl:element>
- <xsl:element name="tr">
- <xsl:element name="th">
  <xsl:text>Attribute</xsl:text>
    </xsl:element>
- <xsl:element name="th">
  <xsl:text>Value</xsl:text>
    </xsl:element>
- <xsl:element name="th">
  <xsl:text>Text Description</xsl:text>
    </xsl:element>
    </xsl:element>
  <xsl:apply-templates select="tcs:ComponentOptimizableCharacteristics" />
    </xsl:element>
  - <!--
   Separate each table with a line. The <br /> tag does not seem to work.
   -->
  <xsl:element name="p" />
    </xsl:template>
  - <!--
```

```
 Populate the table with the Characteristics outlined.
  -->
- <xsl:template match="tcs:ComponentOptimizableCharacteristics">
  <xsl:apply-templates select="tcs:Characteristic" />
  </xsl:template>
- <!--
 Use the Characteristic Name, Unit Value, and Text Description to
 populate the table.
  -->
- <xsl:template match="tcs:Characteristic">
- <xsl:element name="tr">
- <xsl:element name="th">
  <xsl:value-of select="tcs:CharacteristicName" />
- <xsl:if test="tcs:CharacteristicUnit/@tcs:abbreviation!='null'">
  <xsl:text>(</xsl:text>
  <xsl:value-of select="tcs:CharacteristicUnit/@tcs:abbreviation" />
  <xsl:text>)</xsl:text>
  </xsl:if>
  <xsl:text>:</xsl:text>
  </xsl:element>
- <xsl:element name="td">
  <xsl:value-of select="tcs:CharacteristicValue" />
  </xsl:element>
- <xsl:choose>
- <xsl:when test="tcs:CharacteristicDescriptionText!='null'">
- <xsl:element name="td">
  <xsl:value-of select="tcs:CharacteristicDescriptionText" />
  </xsl:element>
  </xsl:when>
- <xsl:otherwise>
- <xsl:element name="td">
  <xsl:text>N/A</xsl:text>
  </xsl:element>
  </xsl:otherwise>
  </xsl:choose>
  </xsl:element>
  </xsl:template>
  </xsl:stylesheet>
```

15. APPENDIX H: VBA CODE FOR MACRO “ALTERNATIVESEVAL”

```
Sub AlternativesEval()
'
' AlternativesEval Macro
' Macro developed by: Francisco Gallo, Josemaria Dolores
'
' Keyboard Shortcut: Ctrl+Shift+O
'

'***************************************
```

```
'DEFINITION OF PARAMETERS AND VARIABLES
'************************************

'Number of components to be optimized
intNumComponents = 6
'Number of alternatives per component
intNumAlternatives = 3

'Sheet where the information for each component and constraints are stored
strSheetComponentsInfo = "OptimizationGrlFormulation"
'Sheet where the alternatives will be populated
strSheetAlternatives = "OptimizationEvaluation"

'+++++++++++++++++++++++++++++++++++++++++++++++++++++
'Parameters for sheet assigned to strSheetAlternatives

'Row where the list of options starts
intFirstRow = 4
'Row where current alternative is being evaluated
intCurrentAlternative = 0

'+++++++++++++++++++++++++++++++++++++++++++++++++++++++
'Parameters for sheet assigned to strSheetComponentsInfo

'Row where the list of components starts
intFirstRowComponents = 6
'Row where component being read is located
intRowCurrComponent = 0

'+++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
'Declaration of arrays to store the information for the components

'Dimensions for the arrays
intArrNumComponents = intNumComponents - 1
intArrNumAlternatives = intNumAlternatives - 1

'Array where the cost for each component is stored.
Dim intComponentsCostArr(5, 2)
'Array where the reliability for each component is stored.
Dim intComponentsReliabArr(5, 2)
'Array where the durability for each component is stored.
Dim intComponentsDurArr(5, 2)
'Array with letters of alphabet to reference the positions of the values in the sheet
Dim strAlphabetArr As Variant

'************************************************
'EACH ARRAY IS POPULATED WITH THE APPR0PRIATE DATA
'************************************************

'The array for the alphabet is populated
strAlphabetArr = Array("A", "B", "C", "D", "E", "F", "G", "H", "I", "J", "K", "L", "M", "N", "O", "P", "Q",
"R", "S", "T", "U", "V", "W", "X", "Y", "Z")

Sheets(strSheetComponentsInfo).Select

'The array with the costs is populated
```

```
For intCount01 = 0 To intArrNumComponents

    intRowCurrComponent = intFirstRowComponents + intCount01

    intComponentsCostArr(intCount01, 0) = Range(strAlphabetArr(1) & intRowCurrComponent)
    intComponentsCostArr(intCount01, 1) = Range(strAlphabetArr(2) & intRowCurrComponent)
    intComponentsCostArr(intCount01, 2) = Range(strAlphabetArr(3) & intRowCurrComponent)

Next intCount01

'The array with the reliabilities is populated
For intCount01 = 0 To intArrNumComponents

    intRowCurrComponent = intFirstRowComponents + intCount01

    intComponentsReliabArr(intCount01, 0) = Range(strAlphabetArr(4) & intRowCurrComponent)
    intComponentsReliabArr(intCount01, 1) = Range(strAlphabetArr(5) & intRowCurrComponent)
    intComponentsReliabArr(intCount01, 2) = Range(strAlphabetArr(6) & intRowCurrComponent)

Next intCount01

'The array with the durabilities is populated
For intCount01 = 0 To intArrNumComponents

    intRowCurrComponent = intFirstRowComponents + intCount01

    intComponentsDurArr(intCount01, 0) = Range(strAlphabetArr(7) & intRowCurrComponent)
    intComponentsDurArr(intCount01, 1) = Range(strAlphabetArr(8) & intRowCurrComponent)
    intComponentsDurArr(intCount01, 2) = Range(strAlphabetArr(9) & intRowCurrComponent)

Next intCount01

'**************************
'GENERATION OF ALTERNATIVES
'**************************

'The appropriate sheet is selected
Sheets(strSheetAlternatives).Select

'This loop populates the 1st component
For intCount01 = 1 To intNumAlternatives

    'This loop populates the 2nd component
    For intCount02 = 1 To intNumAlternatives

        intRowToInsert02 = intFirstRow + intCurrentAlternative

        strTargetCell02 = "C" & intRowToInsert02

        'This loop populates the 3rd component
        For intCount03 = 1 To intNumAlternatives

            intRowToInsert03 = intFirstRow + intCurrentAlternative

            strTargetCell03 = "D" & intRowToInsert03
```

```
'This loop populates the 4th component
For intCount04 = 1 To intNumAlternatives

    intRowToInsert04 = intFirstRow + intCurrentAlternative

    strTargetCell04 = "E" & intRowToInsert04

    'This loop populates the 5th component
    For intCount05 = 1 To intNumAlternatives

        intRowToInsert05 = intFirstRow + intCurrentAlternative

        strTargetCell05 = "F" & intRowToInsert05

        'This loop populates the 6th component
        For intCount06 = 1 To intNumAlternatives

            'Row where option will be inserted is defined here
            intRowToInsert = intFirstRow + intCurrentAlternative

            '*********************************************
            'THIS PORTION OF CODE INSERTS THE ALTERNATIVES
            '*********************************************

            'The target cells are defined for each component
            strTargetCell01 = "B" & intRowToInsert
            strTargetCell02 = "C" & intRowToInsert
            strTargetCell03 = "D" & intRowToInsert
            strTargetCell04 = "E" & intRowToInsert
            strTargetCell05 = "F" & intRowToInsert
            strTargetCell06 = "G" & intRowToInsert

            'Component 1 option is inserted here
            Range(strTargetCell01) = intCount01
            'Component 2 option is inserted here
            Range(strTargetCell02) = intCount02
            'Component 3 option is inserted here
            Range(strTargetCell03) = intCount03
            'Component 4 option is inserted here
            Range(strTargetCell04) = intCount04
            'Component 5 option is inserted here
            Range(strTargetCell05) = intCount05
            'Component 6 option is inserted here
            Range(strTargetCell06) = intCount06

            '**************************************************************************
            'THIS PORTION OF CODE INSERTS THE VALUES FOR EACH IMPLEMENTATION ALTERNATIVE
            '**************************************************************************

            'The target cells are defined for each component
            strTargetCell07 = "I" & intRowToInsert
            strTargetCell08 = "J" & intRowToInsert
            strTargetCell09 = "K" & intRowToInsert
            strTargetCell10 = "L" & intRowToInsert
            strTargetCell11 = "M" & intRowToInsert
```

```
strTargetCell12 = "N" & intRowToInsert

'Component 1 cost is inserted here
Range(strTargetCell07) = intComponentsCostArr(0, intCount01 - 1)
'Component 2 cost is inserted here
Range(strTargetCell08) = intComponentsCostArr(1, intCount02 - 1)
'Component 3 cost is inserted here
Range(strTargetCell09) = intComponentsCostArr(2, intCount03 - 1)
'Component 4 cost is inserted here
Range(strTargetCell10) = intComponentsCostArr(3, intCount04 - 1)
'Component 5 cost is inserted here
Range(strTargetCell11) = intComponentsCostArr(4, intCount05 - 1)
'Component 6 cost is inserted here
Range(strTargetCell12) = intComponentsCostArr(5, intCount06 - 1)

'The target cells are defined for each component for the reliability
'Here the same ones used for Costs will be reused
strTargetCell07 = "Q" & intRowToInsert
strTargetCell08 = "R" & intRowToInsert
strTargetCell09 = "S" & intRowToInsert
strTargetCell10 = "T" & intRowToInsert
strTargetCell11 = "U" & intRowToInsert
strTargetCell12 = "V" & intRowToInsert

'Component 1 reliability is inserted here
Range(strTargetCell07) = intComponentsReliabArr(0, intCount01 - 1)
'Component 2 reliability is inserted here
Range(strTargetCell08) = intComponentsReliabArr(1, intCount02 - 1)
'Component 3 reliability is inserted here
Range(strTargetCell09) = intComponentsReliabArr(2, intCount03 - 1)
'Component 4 reliability is inserted here
Range(strTargetCell10) = intComponentsReliabArr(3, intCount04 - 1)
'Component 5 reliability is inserted here
Range(strTargetCell11) = intComponentsReliabArr(4, intCount05 - 1)
'Component 6 reliability is inserted here
Range(strTargetCell12) = intComponentsReliabArr(5, intCount06 - 1)

'The target cells are defined for each component for the durability
'Here the same ones used for Costs will be reused
strTargetCell07 = "Y" & intRowToInsert
strTargetCell08 = "Z" & intRowToInsert
strTargetCell09 = "AA" & intRowToInsert
strTargetCell10 = "AB" & intRowToInsert
strTargetCell11 = "AC" & intRowToInsert
strTargetCell12 = "AD" & intRowToInsert

'Component 1 durability is inserted here
Range(strTargetCell07) = intComponentsDurArr(0, intCount01 - 1)
'Component 2 durability is inserted here
Range(strTargetCell08) = intComponentsDurArr(1, intCount02 - 1)
'Component 3 durability is inserted here
Range(strTargetCell09) = intComponentsDurArr(2, intCount03 - 1)
'Component 4 durability is inserted here
Range(strTargetCell10) = intComponentsDurArr(3, intCount04 - 1)
'Component 5 durability is inserted here
Range(strTargetCell11) = intComponentsDurArr(4, intCount05 - 1)
```

```
            'Component 6 durability is inserted here
            Range(strTargetCell12) = intComponentsDurArr(5, intCount06 - 1)

            '++++++++++++++++++++++++++++++++++++++++++++
            'The counter to move to next row is increased
            intCurrentAlternative = intCurrentAlternative + 1

          Next intCount06

        Next intCount05

      Next intCount04

    Next intCount03

  Next intCount02

Next intCount01

End Sub
```

16. APPENDIX I: VBA CODE FOR MACRO “CONSTRAINTSEVAL”

```
Sub ConstraintsEval()
'
' ConstraintsEval Macro
' Macro developed on 5/5/2005 by Francisco Gallo
'
' Keyboard Shortcut: Ctrl+Shift+C
'

'*************************************
'DEFINITION OF PARAMETERS AND VARIBLES
'*************************************

'Number of components to be optimized
intNumComponents = 6
'Number of alternatives per component
intNumAlternatives = 3

'Row where the list of options starts
intFirstRow = 4

'String where the description of the components failing the constraints is
'stored to be displayed once the constraint analysis is finished.
strComponentFailedMessage = ""

'Sheet where the information for each component and constraints are stored
strSheetComponentsInfo = "OptimizationGrlFormulation"
'Sheet where the alternatives will be populated
strSheetAlternatives = "OptimizationEvaluation"

'++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++
'Declaration of parameters belonging to sheet stored in strSheetAlternatives
```

```
'Array that contains the columns where each of the component options is stored
'for each implementation alternative
Dim strImplemAlternatCols As Variant

'Array that contains each option for a specific implementation alternative
Dim intImplemAlternatVals(5)

'Array that contains the columns where the constrained values for each desing
'objective will be entered.
Dim strConstDesObjsCols As Variant

'Column where the list of components that failed one or more constraints are displayed
chrColFailedComps = "AK"

'++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++++
++++
'Declaration of parameters belonging to sheet stored in strSheetComponentsInfo

'Column where the name of each component is
chrColComponentResult = "B"
'Column where the criterai for a certain constrain is
chrColConstraintCriteria = "C"
'Column where each option is indicated to be tested against constraints
chrColOptionTest = "G"
'Column where each constraint is evaluated (Fails or Passes)
chrColConstraintMes = "I"

'Array containing the distribution of rows containing the contraints for each
'component. The order of the components is the same as in sheet stored in
'strSheetAlternatives. This is: 0-Heater, 1-chiller, 2-fan, 3-PPCD, 4-Controlling Device,
'and 5-thermometer.
Dim intComponentConstraints As Variant

'************************
'THE ARRAYS ARE POPULATED
'************************

strImplemAlternatCols = Array("B", "C", "D", "E", "F", "G")
strConstDesObjsCols = Array("AG", "AH", "AI")
intComponentConstraints = Array(16, 22, 28, 33, 36, 39, 46)

'*************************************************************************************
'EVALUATION OF EACH CONSTRAINT FOR EACH COMPONENT FOR EACH
IMPLEMENTATION ALTERNATIVE
'*************************************************************************************

'The total number of implementation alternatives is calculated
intTotImpAlternat = ((intNumAlternatives ^ intNumComponents) - 1)

Sheets(strSheetAlternatives).Select

'This loop goes through all implementation alternatives
For intCount01 = intFirstRow To (intTotImpAlternat + intFirstRow)

    'This flag is set to False when a certain combination of componets fails one or more constraints
```

```
    bPassFlag = True

    'This loop goes through each component option for a certain implementation alternative
    For intCount02 = 0 To (intNumComponents - 1)

        'Cell for each option of each component for each implementation alternative
        strCellImpAlternatComponent = strImplemAlternatCols(intCount02) & intCount01

        intImplemAlternatVals(intCount02) = Range(strCellImpAlternatComponent)

    Next intCount02

    Sheets(strSheetComponentsInfo).Select

    'This loop enters each option in the cells that evaluate the constraints
    For intCount02 = 0 To (intNumComponents - 1)

        'The appropriate starting row is determined here
        intRowStart = intComponentConstraints(intCount02)
        'The appropriate ending row is determined here
        intRowEnd = intComponentConstraints(intCount02 + 1)

        'Each row is populated with the appropriate option for the appropriate component
        For intCount03 = intRowStart To (intRowEnd - 1)

            Range(chrColOptionTest & intCount03) = intImplemAlternatVals(intCount02)

            'The option is evaluated to determine if current constraint fails
            If Range(chrColConstraintMes & intCount03) = "FAILS" Then

                bPassFlag = False

                'The component that failed is stored for later display
                If strComponentFailedMessage = "" Then

                    strComponentFailedMessage = "(" & Range(chrColComponentResult & intCount03)
& ").(" & Range(chrColConstraintCriteria & intCount03) & ")"

                Else

                    strComponentFailedMessage = strComponentFailedMessage & ", (" &
Range(chrColComponentResult & intCount03) & ").(" & Range(chrColConstraintCriteria &
intCount03) & ")"

                End If

            End If

        Next intCount03

    Next intCount02


    'Now the appropriate cell for the constrained implementation alternative
    'design objective is filled accordingly (constraint failed or passed)
    Sheets(strSheetAlternatives).Select
```

```
    'If one or more constraints failed, then a "fails" message is entered for that
    'implementation alternative
    If bPassFlag = False Then

        Range(strConstDesObjsCols(0) & intCount01) = ""
        Range(strConstDesObjsCols(1) & intCount01) = ""
        Range(strConstDesObjsCols(2) & intCount01) = ""

        'The list of components that failed (if any) is included here
        Range(chrColFailedComps & intCount01) = strComponentFailedMessage

    Else

        'If the implementation alternative passes all constraints, each
        'design objective is populated in the constrained implementation alternatives cells
        Range(strConstDesObjsCols(0) & intCount01) = Range("O" & intCount01)
        Range(strConstDesObjsCols(1) & intCount01) = Range("W" & intCount01)
        Range(strConstDesObjsCols(2) & intCount01) = Range("AE" & intCount01)

    End If

    'The variable containing the list of contraints failed is cleared
    strComponentFailedMessage = ""

Next intCount01


End Sub
```