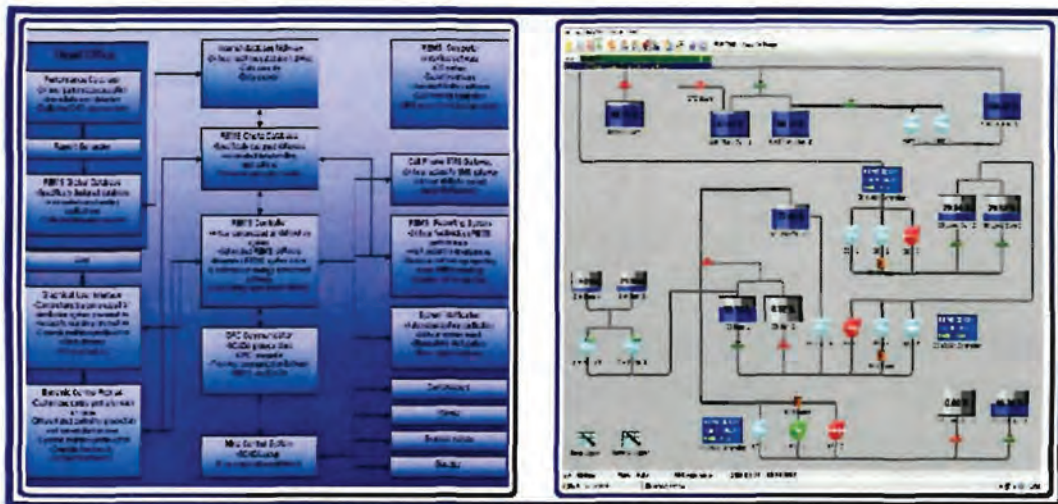


3. Building the novel solution as a feasible system



The building blocks developed in the previous chapter can now be combined. This results in a fully automated simulation and control system, for load shifting and optimising running costs, of a mine pumping system.

3.1. *Prelude*

Chapter 2 of this thesis describes the development of a solution in the form of a control algorithm. This algorithm will answer the need and problem statement as set out in chapter one. This developed control philosophy and algorithm does not by itself achieve the set goal.

This chapter describes how the control philosophy and algorithm was developed into a practical automated control product focussed on controlling any given water pump system in the industrial sector. This developed product incorporates the control philosophy that enables it to achieve the goals set for this research.

3.2. *Discussion - the development procedure*

3.2.1. *Understanding the operational environment*

An important step before developing a system is to first decide upon and understand the intended operation environment [96]. The environment is expected to provide the system with the resources it needs to fulfill its function.

A classic example of such an environment is when a system has to be small and robust enough to be taken anywhere by hand. These control systems are therefore usually developed to run inside the environment supplied by a programmable chip, such as those found in cell phones, calculators, GPS units, etc.

In other cases the environment is expected to provide more calculating power with a wide range of user interface devices. These control systems are usually developed for computer base environments where the system can take advantage of the benefits hosted in the computer environment.

It was decided that REMS should be run and hosted on a normal server PC (Personal Computer)- a PC adapted for operation in the industrial environment. An example of such a server is shown in the next figure.

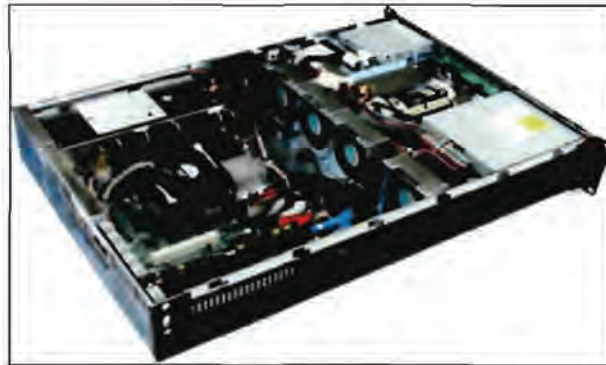


Figure 3-1 Computer server [100]

The following reasons drove the decision to develop REMS as a PC based application:

1. REMS will need to communicate to other SCADA (Supervisory Control and Data Acquisition) packages, in order to control pumps and exchange data. These packages are also run on computer based systems. Developing REMS as a PC based application will ease the communication process between these packages and REMS.
2. REMS will also need high level processing power to make continued calculations. This processing capacity is needed to power the control and simulation algorithms built into REMS, that steer the control philosophy and make simulation possible. This is also available in the environment, hosted in a computer.
3. REMS also needs massive data storage capabilities to store performance and reporting data. This functionality is also provided by a computer based environment. REMS saves status and system data in a database. This data is then used for hindsight optimisation and overall success calculation and reporting.

3.2.2. Drawing up the development specification

The development specification of an application or system defines the planned capability of the system. It also contains information about the recourses that will be

needed. It also contains specifications on how the system will access data required, and in what format the outputs of the system will be [89].

A complete development specification must contain the following information and descriptions.

1. **System features:** This section shortly describes the main system features. It also answers questions about the reason for the development of the systems.
2. **Architecture of the system:** This describes the medium in which the system will be presented and will function.
3. **Data connections:** This is a description of the portals and methods the system will use to acquire and present data, if applicable.
4. **System component layout:** Bigger systems are usually broken up into sections or sub-menus to aid understanding functionality and usage. This section usually describes the layout of these different sections and how they relate to one another.

The development specification can, and usually is, used as patent specification when systems are being patented. Since the development specification contains all the intended use and functionality of the proposed systems, it is adequate for this purpose.

3.2.3. Coding the application

The third step of the development procedure is to set the development specification to code. Best practice is to have the development specification finalised at this point. The choice of the coding language depends mostly upon the requirements set to the final product.

Delphi 6 was chosen as the coding language for REMS. One of the reasons for this choice was because Delphi is a language focussed on rapid application development for a broad spectrum of Microsoft Windows-based application. A second reason is

because there is a broad spectrum of API's (Application Program Interface) developed for Delphi, available on the internet, that aids in the incorporation of functionalities such as OPC and E-Mail capabilities.

The new application consists of 150 thousand lines of code. That is a 100 times more than the lines of text in this thesis.

An in depth look at the Delphi code of REMS is presented in Appendix 7.3. This appendix gives a layout of all the coding units and the application interfaces.

3.3. Development specification

3.3.1. System characteristics

SCADA communication

REMS must be able to control and monitor, and therefore communicate to the water pump system elements. REMS was therefore developed to communicate directly to SCADA systems. A SCADA system is software that is used on most mines and industrial plants to monitor and control all electrical and industrial components and processes from one control room. See section 3.3.4 on illustrations and discussion on how this was achieved.

Usually the SCADA consists of more than one interface computer giving many users simultaneous access to the controlled systems.

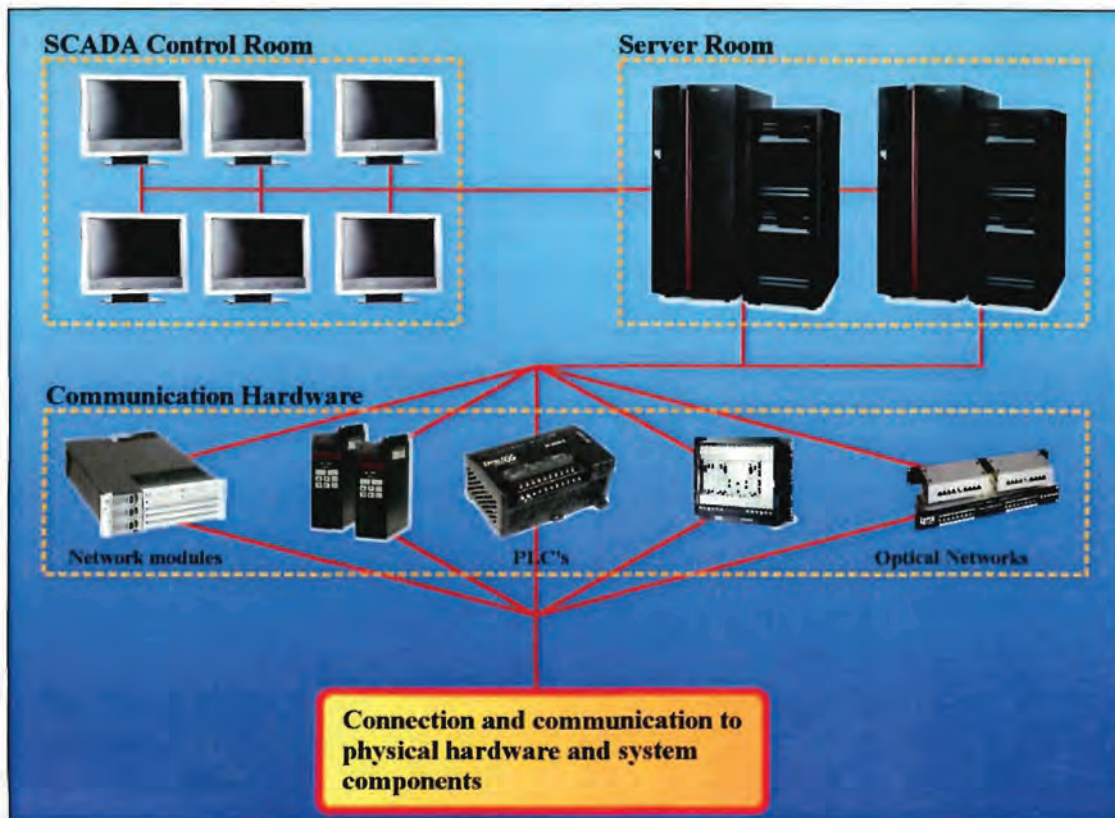


Figure 3-2 Typical SCADA layout

Since the working of the mine is dependent on the SCADA, it cannot be removed or replaced. REMS was therefore engineered with the capability to transact data via OPC (Object Linking and Embedding for Process Control) protocol. OPC is a standard protocol in the SCADA industry and most SCADA systems are OPC enabled. This is further discussed in section 3.3.4. The OPC connection enables REMS to work in conjunction with the SCADA and to gain control of all the system elements.

Data loggings

REMS was engineered to log any useful data in a manageable format. This data is later used for various purposes such as success measurement, verification and reporting. This is important to determine the success of the system on a given project where REMS was implemented.

This data is logged and stored in an easy accessible format, easing the use thereof. The data is logged with reference to time and component name, making this data valuable to not only REMS, but also the mine operators and system technicians.

Alarm system and alarm formats

REMS is involved in controlling components and systems of the mine which are directly linked to production of the mine. Owing to this, REMS was engineered with a broad spectrum of alarm systems. The alarms raised are visual and audio prompts normally associated with computer software.

REMS also extends alarms to E-Mails and SMS formats sent via the national cell phone network when need be. This wide range of available alarm formats can reach a wider medium than the standard audiovisual alarms associated with computer software and SCADAs.

Automated control

REMS introduces automated control. In all of the case studies where REMS was implemented, control was previously done by control room operators. The operator had the SCADA as his tool, enabling him to control all the pump stations easily from one central point, but the SCADAs did not offer solutions in terms of simulation and automation.

REMS automates this control by cutting out the necessity for full-time human assistance. This improves control sustainability. REMS incorporated alarm systems, alerting operators when and if human interference is needed.

3.3.2. Architecture of the new system

REMS was designed as a software application that will run in a normal computer environment. REMS contains different sections, as do most software applications. Usually, as also the case with REMS, each component has its own specialised function.

REMS will consist of the following components, each developed to fulfil the function as described:

1. Platform: The platform will act as a link between all the other components, putting in place routines and function for data communication and handling. The platform's main goal is to manage and activate the different components of the system as they are needed. This is necessary as REMS is designed to be fully automated.
2. GUI (Graphical User Interface): This is the part of the system that the user will see and interact with.
3. Data logging and management system.
4. Data acquisition interface: REMS, as a control system, needs to monitor certain system statuses. This interface enables the system to acquire this data.
5. Pump group controller: This unit contains the mathematical algorithms used to control elements, such as pumps. These mathematical algorithms are focussed on optimising the control to realise electrical cost reductions and load shifting.
6. Control interface: This unit enables REMS to send commands via certain command chains to the components and elements, such as pumps.
7. Alarm system: REMS utilises this unit to raise user-defined alarms. The alarms can be raised as audiovisual alarms and/or sent over the national cell phone network to any cell phone and/or sent as an e-mail.

3.3.3. Requirements of a high-quality computer system

A system engineered to run in a computer environment must be developed to meet certain standards. These standards are set to ensure that the software is reliable and compatible with normal standard computer hardware. These standards must be included into the development specification of any software and contains the following [88].

1. The system must be coded in such a way that it can mend and manage the computer memory being used. Whenever the software application is started or “opened” it is loaded into the memory of the computer. The software then uses additional memory to complete calculations and so forth. If the application is not written to standard, “memory leaks” could occur. This will result in eventual computer failure and must be avoided at all cost.
2. The software GUI (Graphical User Interface) must be easy to use and understandable. The GUI is the part of the software that the user sees. It is therefore imperative that the GUI is easy to use and understand.
3. The software should have functional data-save structures. The data save structures of any software application, enables the user to save settings and progress. In simpler terms, it’s the procedures and structures found under the “Save” button.

3.3.4. Data communication

The data communication network of a system describes the data input and output of the system under normal operational conditions. REMS’ data communication network is illustrated and described by the next figure.

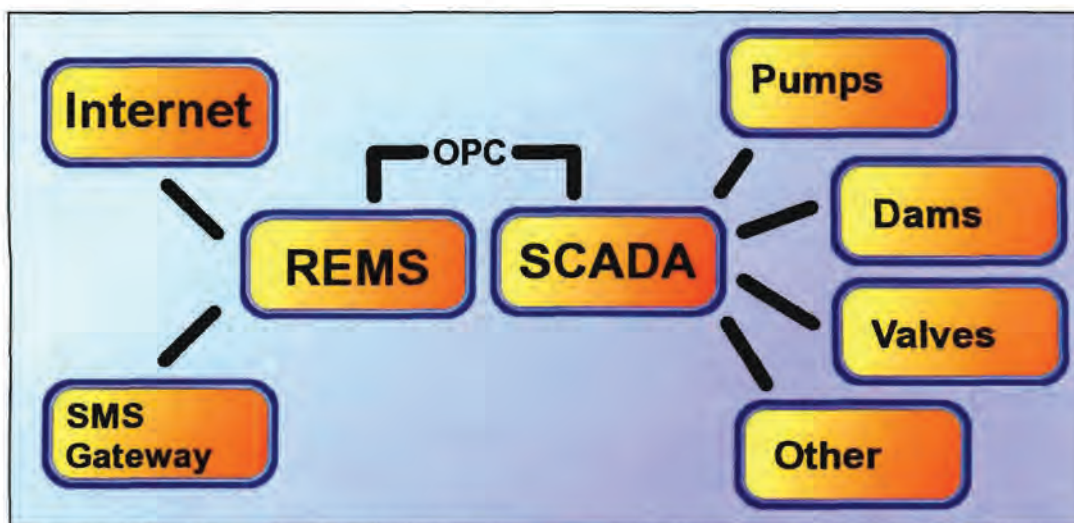


Figure 3-3 REMS data communication network

To allow REMS to fully interact with the required information and electrical components, the following channels, as illustrated in Figure 3-3, have been engineered into the system:

1. Connection to the Internet: REMS uses this connection to access varying electricity prices if applicable and to send e-mails.
2. SMS Gateway: This gateway is used to contact specific persons in prescribed events.
3. SCADA OPC Connection: This connection is needed to exchange data between REMS and the SCADA.

3.4. Input tools

Input tools are GUI interfaces developed, that will broaden the functionality of the final product. These tools and principles will end up as extra functionalities in the final application.

The composed value was developed to provide the user with a much wider input-possibility than normal single value. Consider the following example: Using a control system, the user is asked to provide the maximum number of pumps that is permitted to simultaneously run on a given pump station.

Following normal convention, the user would have been able to enter a single integer value. Composed values designed for this novel solution will act the same way as the integer, used to specify some value of criteria that will influence the system.

What makes the composed value different from the single integer is this: The value thereof can be set to be only a single integer. This is when the composed value is set to 'Fixed Value' mode, as selected in the left hand setting bar, as seen in Figure 3-4.

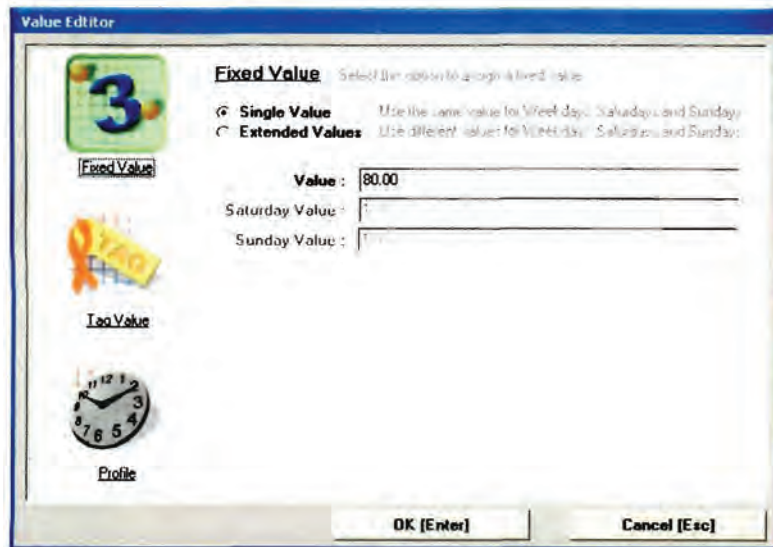


Figure 3-4 Composed value – ‘Fixed value’ mode

The value of a composed value can also be set to differ on Saturdays and Sundays with the provided settings as seen in Figure 3-4.

Another function of the composed value is setting it to always represent or take the value of a given OPC tag. By selecting the ‘Tag Value’ mode on the left hand of the composed value editor, the user is given the option to select an OPC tag for this purpose (see Figure 3-5). In this mode the composed value will always assume the value provided by the appropriate OPC tag.

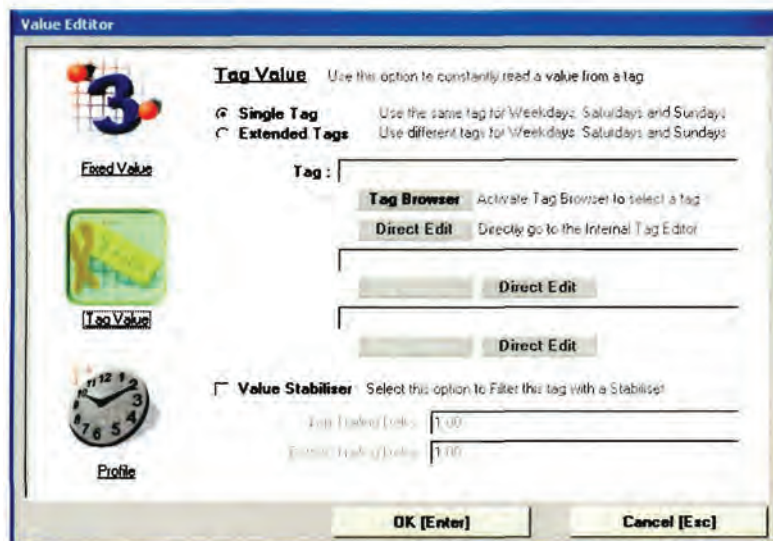


Figure 3-5 Composed value – ‘Tag value’ mode

In 'tag value' mode the user is also able to select different OPC tags that dictate the value of the composed value for Saturday and Sunday.

A further functionality of this mode is the value stabiliser. When activated, this function will stabilise the data received from the given OPC tag. In some cases this is needed when the data provided by the OPC tag fluctuates. The data or value given by the OPC tag will be filtered before the value is passed over by the composed value.

The composed value has a third mode called 'Profile' mode. This mode lets the user set up a profile that dictates the value of the composed value. This is shown in Figure 3-6.

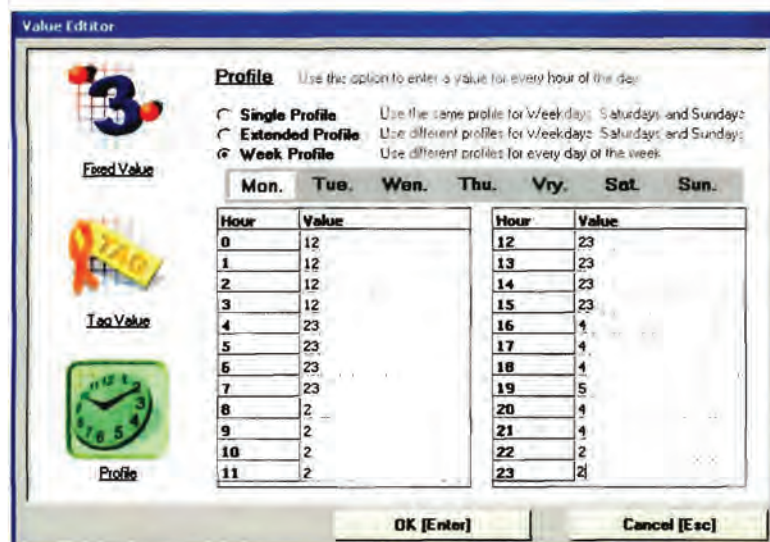


Figure 3-6 Composed value – 'Profile' mode

As seen in Figure 3-6, the profile is comprised of 24 values, each value representing an hour of the day. The user can use only one profile, or set a profile for every day of the week.

The composed value will be used throughout the intended solution, always providing the given functionalities to any setting that has to be set by the user.

3.5. System features

3.5.1. Main interface

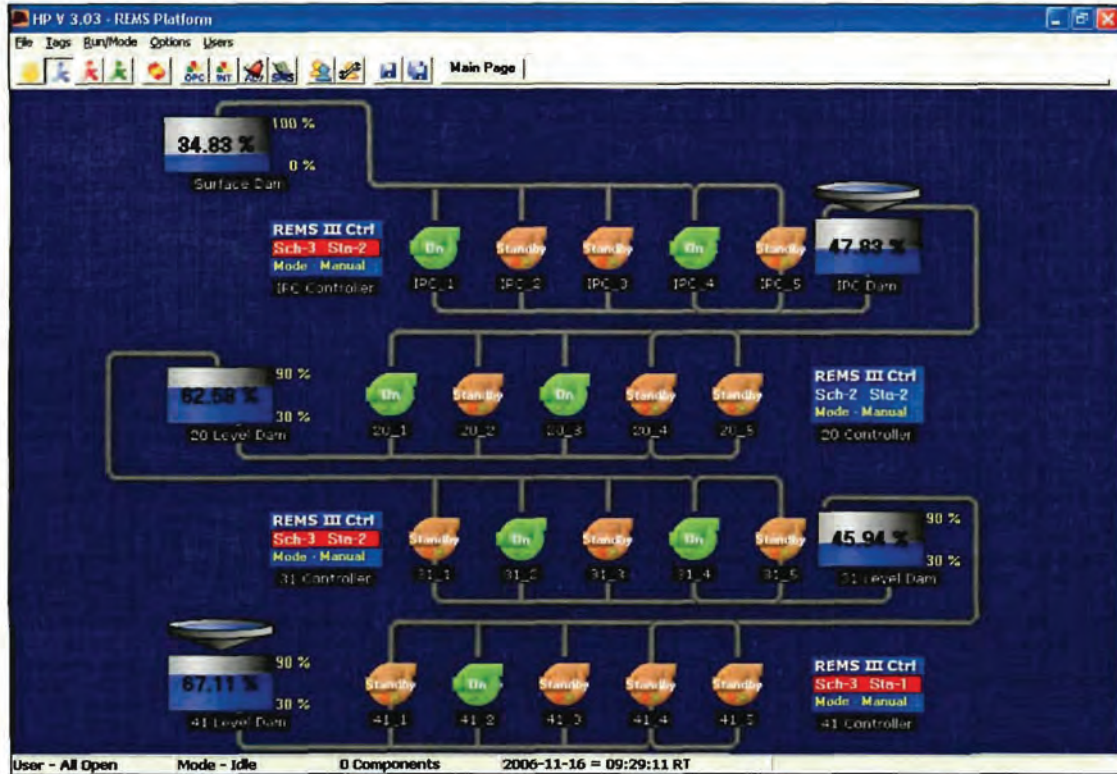


Figure 3-7 REMS platform interface - main interface

The main interface shows the system build up of the water pumping system that is being controlled by REMS. All the components on the interface are interactive, making access to the different tools and properties quick and easy. Clicking on any of the components will access the tools associated with that component. The interface displays real time data regarding the components for immediate on-glance comprehension.

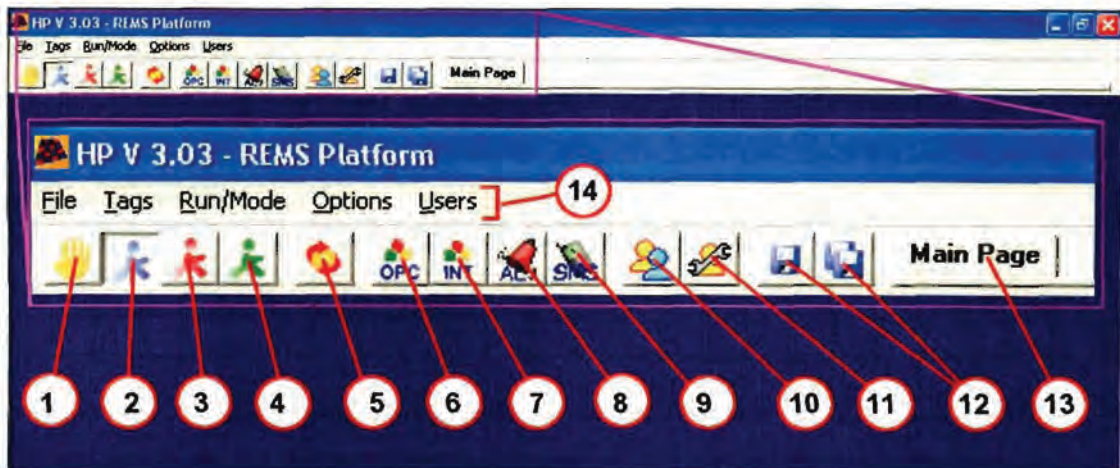


Figure 3-8 REMS platform interface - menu

Figure 3-8 shows the main menu of REMS. From here the user is given access to all the different components and features built into the system. The buttons and functionalities of the main interface are as follows:

(1),(2),(3),(4). These four buttons switch REMS between its 4 different operating modes. See section 3.5.2.

(5). Access the Run and Pulse option. See section 3.5.3.

(6). Access the OPC settings. See section 3.5.4.

(7). Access the internal tag manager. See section 3.5.5. The internal tags enable the user to filter or manipulate data that is fed into or sent out of REMS.

(8). Access the alarms manager and alarm options. See section 3.5.6.

(9). Open the SMS options and contact manager. Here the user specifies how and when SMS' should be send. The cellular phone numbers of the applicable people are also specified and listed here.

(10). User log on. REMS restricts certain users from accessing sensitive data and settings. This button allows each user to 'log on' so that REMS can know which settings and functionalities to make available.

(11). User manager. This tool lets the system administrator decide which users have access to what functionality and settings.

(12). Tools to save and make back-ups of the system control set-up.

(13). Tabbed controller to flip through the different defined system layouts. Each layout can be made to show different data.

(14). REMS system main menu.

3.5.2. Platform modes

REMS platform can be switched between 4 different modes to activate and deactivate certain functionalities as needed. The four modes are:

1. **Edit Mode:** In this mode the user is able to change and edit the system build-up of the water pump system. The user can add, or delete pumps, dams, level controllers, piping etc. The user also connects the components to one another, dictating how they relate.
2. **Idle mode:** In this mode REMS is idle and does nothing. The control system is deactivated and no control or scheduling is being done.
3. **Manual mode:** In this mode REMS will calculate the optimised control schedules for the components of the water pump system, but no control is being performed. The optimised schedules are displayed to the user and the pumps can be manually controlled in accordance to these schedules if the operator wishes.
4. **Auto (Automated) mode:** In this mode REMS will calculate the optimised schedules for the components in the water pump system and control these components in accordance with the schedules.

3.5.3. Run options

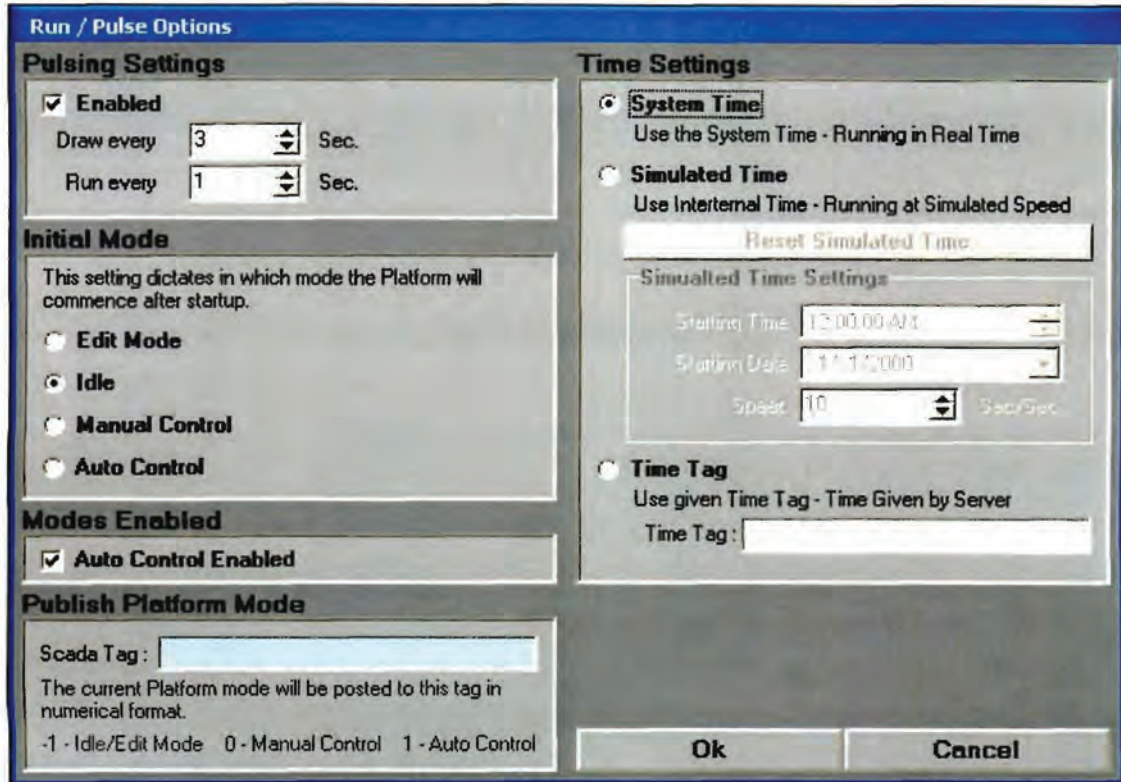


Figure 3-9 REMS run options

Figure 3-9 show the REMS run options editor. The tools and functionality provided by this editor are as follows:

Pulsing settings

The ‘Draw every’ setting dictates how often the interface that display live system information should be updated.

The ‘Run every’ setting dictates how often the control schedules must be revisited and updated.

Initial mode

This setting dictates in which mode REMS will commence after a system restart or start-up.

Publish platform mode

On some projects it is necessary for the SCADA to know in which mode REMS is operating. This functionality, when activated, publishes the current mode of REMS in a given OPC tag. As the SCADA has access to the OPC tags, it can read the REMS mode from this tag.

Time settings

REMS can be run in any of three time-modes. Note that this is different from the normal REMS modes. They are:

1. System time: REMS will take the computer clock time as the relevant time. When controlling a real world water pump system, REMS is run in this time mode.
2. Simulated time: When REMS is used to do simulations, see sections 2.4.4 and 3.5.7, REMS is set to the simulated time mode. The settings here also dictate how fast the simulated time will run.
3. Time tag: This time mode is used when the internal clock of REMS is synchronised with the SCADA clock. This helps to synchronise data logging, control commands etc. with that of the SCADA.

3.5.4. OPC settings

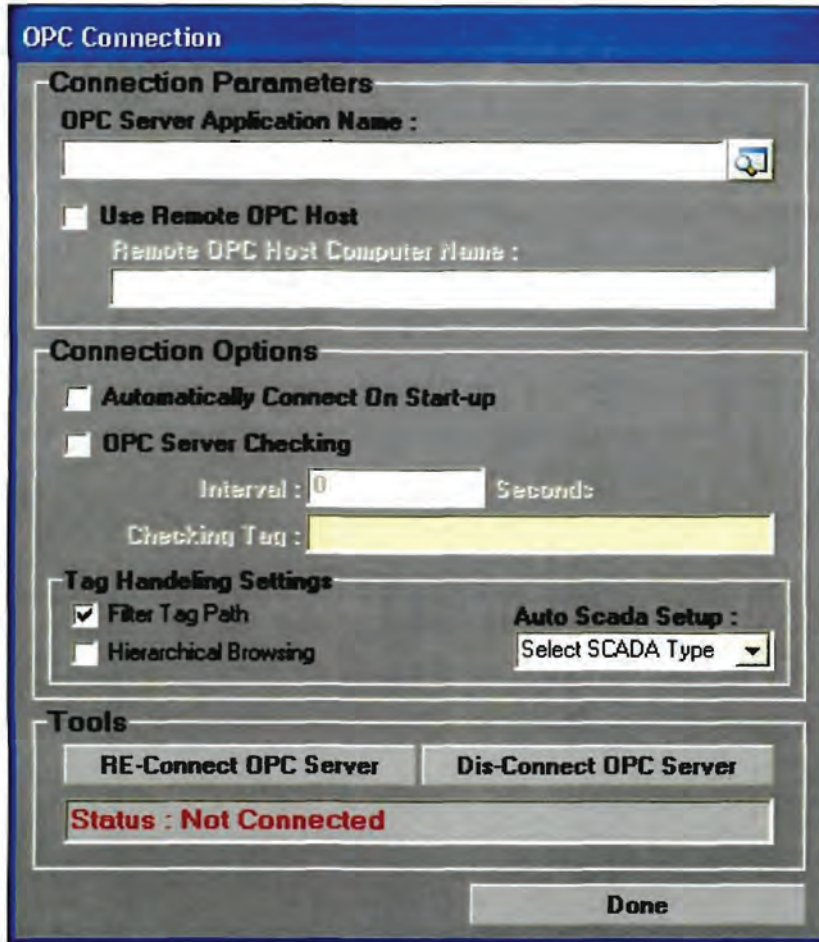


Figure 3-10 REMS OPC options

This interface allows the user to set up the OPC communication to the SCADA.

OPC server application name

Here the user must specify the SCADA application name that hosts the OPC server.

Use remote OPC host

This information is set if the SCADA application is running on a different/remote computer server. This setting is a network address allowing REMS access to the remote server.

Connection options

These settings dictate how the OPC connection must be handled in the event of a server restart or start-up. Also displayed here is the OPC server checking tools. These tools are used to minimise the communication errors between the SCADA and REMS. See section 3.6 that handles system reliability.

Tag handling settings

These tools enable the user to filter the tags made available by the SCADA. Most SCADAs host many tags used for internal communication purposes. Filtering these tags to make them invisible to REMS, increase the safety and reliability of the system.

3.5.5. Internal tags

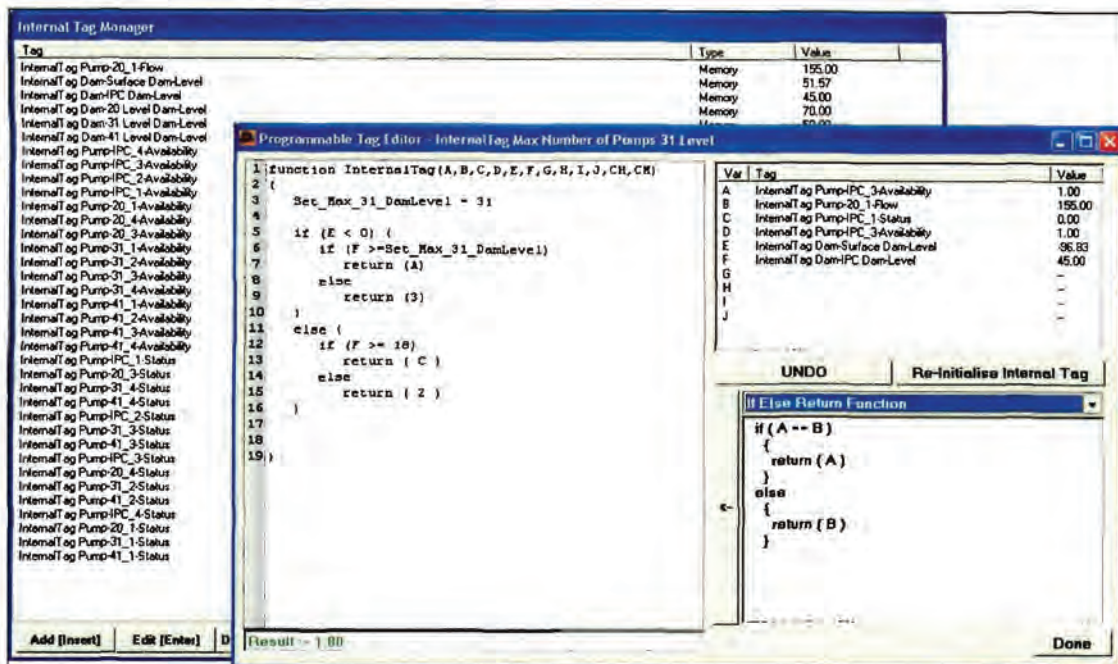


Figure 3-11 REMS internal tag manager and editor

The internal tags, as mentioned, give the user the ability to manipulate any data going into or out of REMS. This tool is not needed to shift load or to realise electrical running cost reduction, but gives the high-level user much more functionality in terms of pro-active control.

The internal tags take data, and generate results according to lines of code entered by the user. The user has to provide lines of code that manipulate the data, giving the user the functionality provided by the coding environment.

Examples of when these tools are used are the following:

1. It is possible that measuring equipment portray a dam level in terms of feet. If the user wants the level of the dam in terms of percentage, an internal tag can be programmed to convert the data from feet to percentage.
2. If non-standard decisions have to be made, that dictates the control of the water pump system on a set of set rules. For example, if the maximum number of pumps running on a given pump station is dependent on many factors, an internal tag is coded to make this calculation based on the information provided.
3. Data logging of summarised data. Internal tags can be used to pre-process data. The processed results of the internal tags can then be logged, making the data more manageable and meaningful.
4. Stabilising data fluctuations and generating a running average of tag data. In some cases, especially when older measuring equipment is used to measure levels or flows, the point values of the data fluctuate making the data risky to use in complex equations such as found in the control algorithms of REMS. Internal tags can then be used to even out these data fluctuations by means of running averages etc.

3.5.6. Alarm manager

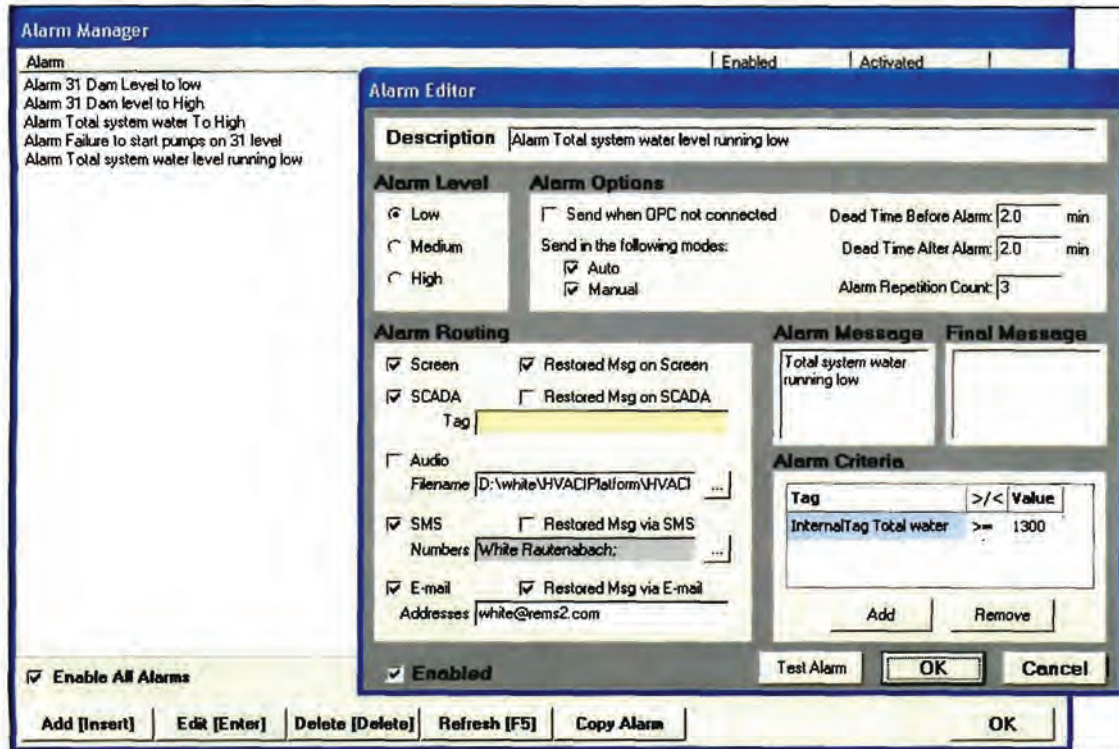


Figure 3-12 REMS alarm manager and editor

Any amount of alarms can be generated in the alarm manager. Each alarm, edited with the alarm editor, can be set to go off (raised) at any given criteria or set of criteria. Each alarm can be set in a series of formats including audiovisual prompts, e-mail and or SMS.

The settings available in the alarm editor are as follows:

Alarms description

A unique name is given to each alarm to discern between the different alarms.

Alarm level

All alarms in REMS will be given a level or priority depending on the severity of the condition. The levels are low, medium and high. If the alarm is raised, the level is displayed so that the operator can know the gravity of the situation.

Alarm options

The alarm options will let the user dictate when and how many times the alarm must be repeated when the set alarm criteria are violated. The user can also dictate in which REMS platform mode the alarm must be raised and/or ignored.

Alarm routing

These settings will let the user decide in which formats the alarm must be raised. The formats to choose from include audiovisual prompt, e-mail and or SMS. The settings regarding the raised formats are also set in this section. The user can also dictate if, and how, the alarm is to be communicated to the SCADA.

Alarm message

The user will set the alarm message here. The alarm message is displayed whenever the alarm is raised. This message is also sent in the e-mail and SMS if applicable.

Alarm criteria

These criteria dictate when the alarm is raised. The user is able to add any amount of criteria. These criteria have access to any of the internal tags, OPC tags, pump station conditions, dam level or pump status conditions.

By allowing the user to use one of the internal tags in the alarm criteria, any programmable condition can be monitored with the use of an alarm. For example: If the user has written an internal tag to calculate the total water in the system, he can link that result to an alarm. This enables the user to monitor the total water in the system and raise an alarm if that level is either too low or too high.

3.5.7. Simulation tools

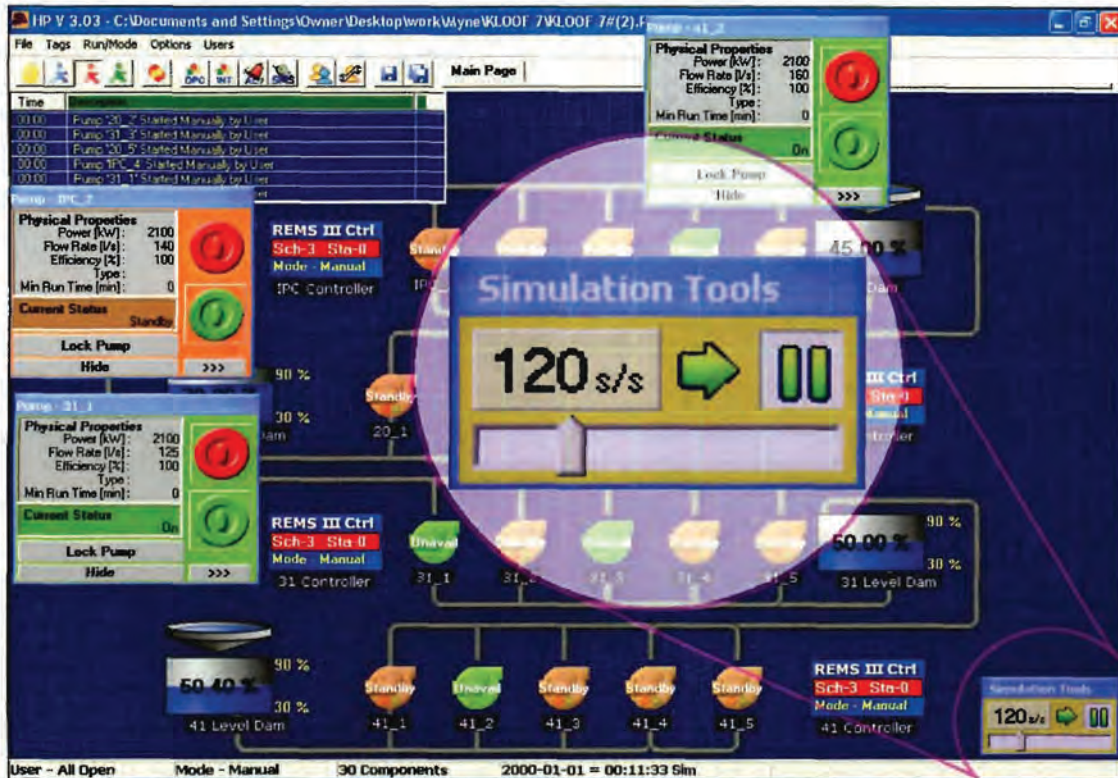


Figure 3-13 REMS simulation tools

REMS has been designed as a simulation tool as well. Figure 3-13 shows the simulation of a mine. This is done to provide the user with information on how the actual system will react when controlled by REMS.

Figure 3-13 show the build-up of a typical mine. During the simulation REMS does not need to be connected to the SCADA. This means that the simulation can be done anywhere and independent of the actual system that is being simulated. The user will set the control parameters as if controlling the actual mine system and then start the simulation.

The simulation can be run at any speed. In Figure 3-13 the simulation is run at 120 seconds per second. Thus the simulation is run at 120 times actual time. The speed can be altered during the simulation process. The simulation can also be paused and resumed at any time.

The statuses of the system are logged in data files for further in-depth investigations when the simulation is concluded. See section 4.2 on how this data is used and processed to predict the potential of a project.

The statuses and system conditions are also displayed in real time on the system build-up. This enables the user immediate understanding on how the system will react in an actual control situation. The simulation can be replayed at any time and speed.

3.5.8. Component editors

All components on the interface are created in edit mode by dropping them from the available component list onto the interface where they are to be positioned. In edit mode, clicking on the component will open the component specific editor.

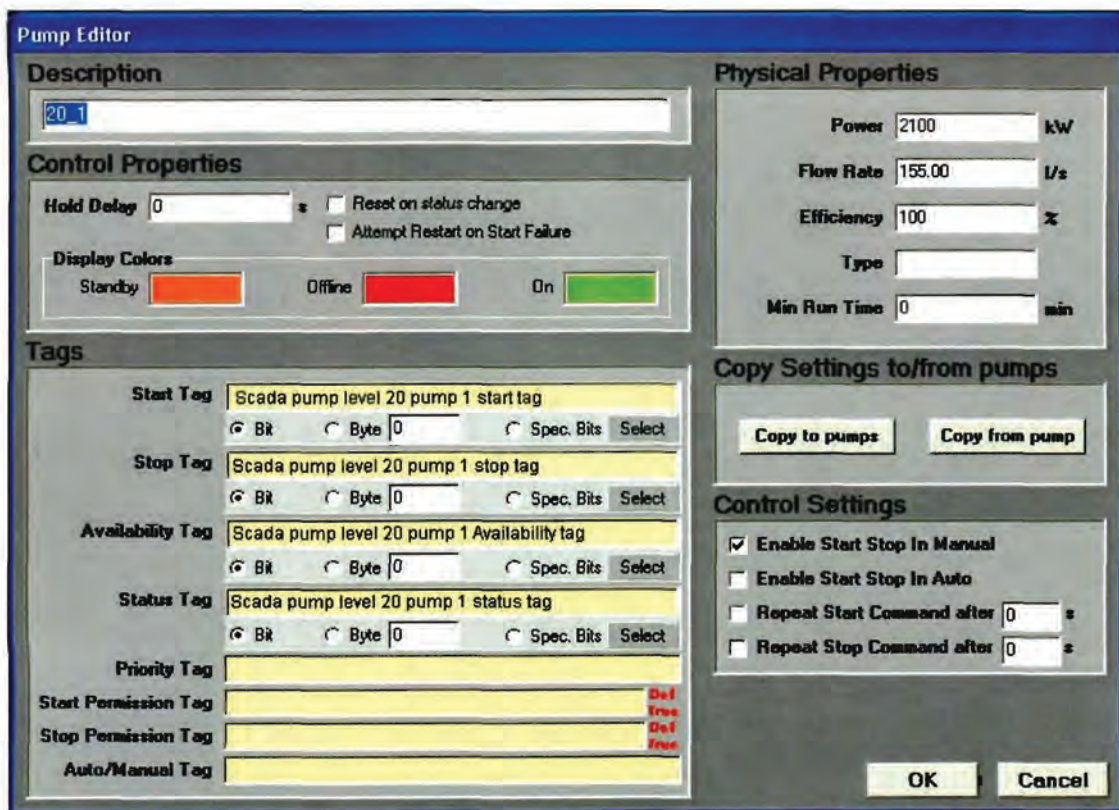


Figure 3-14 REMS pump editor

The pump editor, as shown in Figure 3-14, is used to set up a pump component that is dropped onto the interface. The settings on the pump editor are as follows:

Description

Here the user is allowed to set a descriptive name for each of the pumps on the interface. The descriptive names are necessary to discern the different pumps from one another.

Hold delay

This setting dictates the duration of a control instruction, such as “start pump”, or “stop pump”, that is sent to the SCADA.

Display colours

These settings dictate the colours that are associated with the different statuses of the pumps. The pumps are displayed in these set colours in accordance with their actual statuses.

Tags

The OPC tags that are used to control the pump via the SCADA are specified here. The start tag is used to start the pump etc. The availability tag is used to determine if the pump is available for operation. The start permission tag, if provided, tells REMS whether the pump may be started etc.

Physical properties

The physical properties of the given pump are set here.

Control settings

The operator uses this setting to adjust how the pump may be controlled from the main REMS interface.

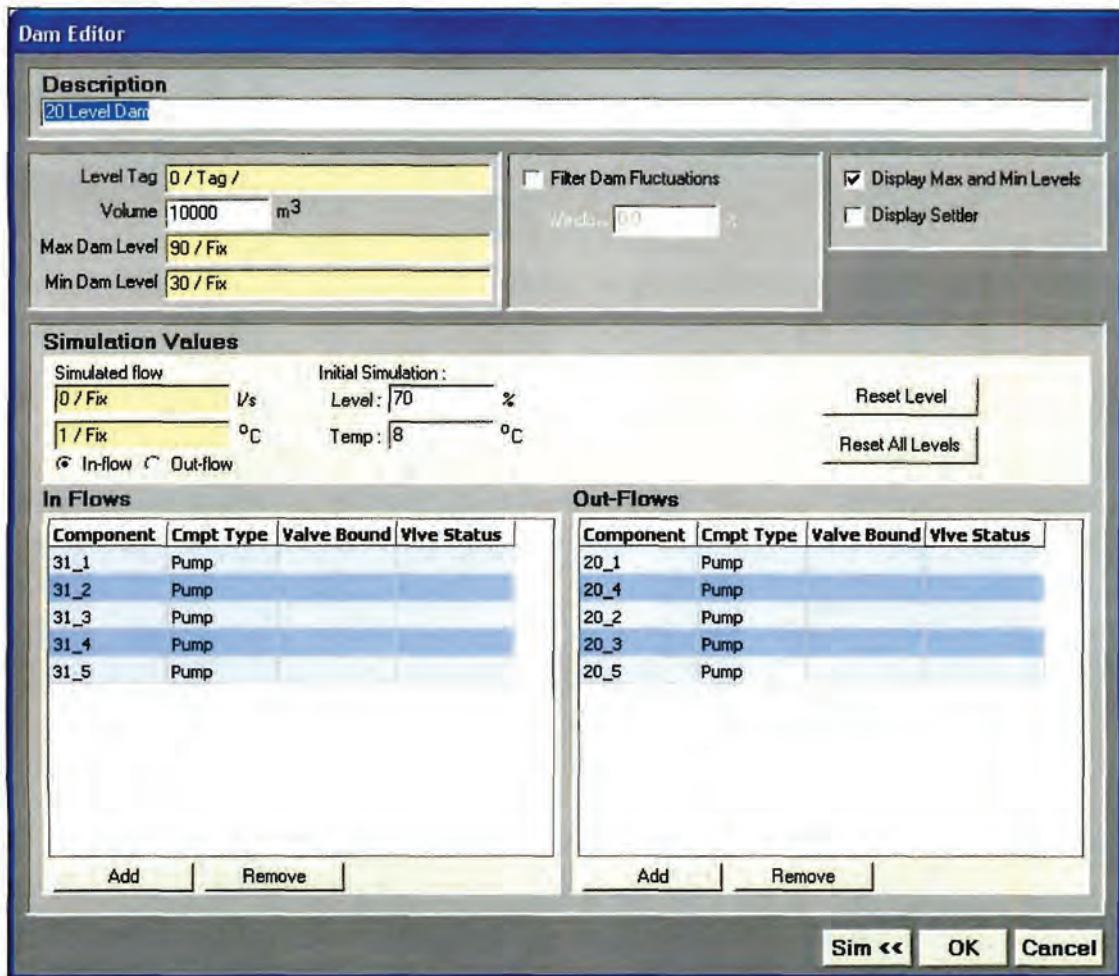


Figure 3-15 REMS dam editor

The dam editor, as seen in Figure 3-15, is used to set up the dam components. The functionalities on the dam component editor are as follows:

Description

This editor, similar to the pump component editor, provides the user with the functionality to give every dam component a unique descriptive name.

Level tag

The OPC tag linked here is used to ascertain the actual dam level.

Volume

The actual volume of the dam is specified here.

Maximum and Minimum dam level

These are the constraints that define the maximum and minimum levels that are allowed in the dam. REMS controls the pump stations to maintain the level of the dam between these two parameters.

Filter dam fluctuations

In some cases the data that is provided by the OPC tag, that portray the dam level, gives fluctuating values. This can cause complications in the control algorithms that REMS uses in the control engine. By activating this option the data is pre-processed evening out the fluctuations by the set percentage window.

Simulation values

These settings are only needed when REMS is used as a simulation tool. These settings are also hidden and can only be accessed when the simulation mode is activated. The simulated flow settings are used to specify if the dam has a constant external in- or outflow. The initial simulated dam level and temperature are also set here.

In flows and out flows

Specified here are the pumps pumping water into the dam and out of the dam.

3.5.9. Component control panels

The component control tools or panels are displayed when the user clicks on any of the interactive components in the main interface. These tools are used in normal control conditions and are focussed on giving the operator information about the current control conditions and statuses.

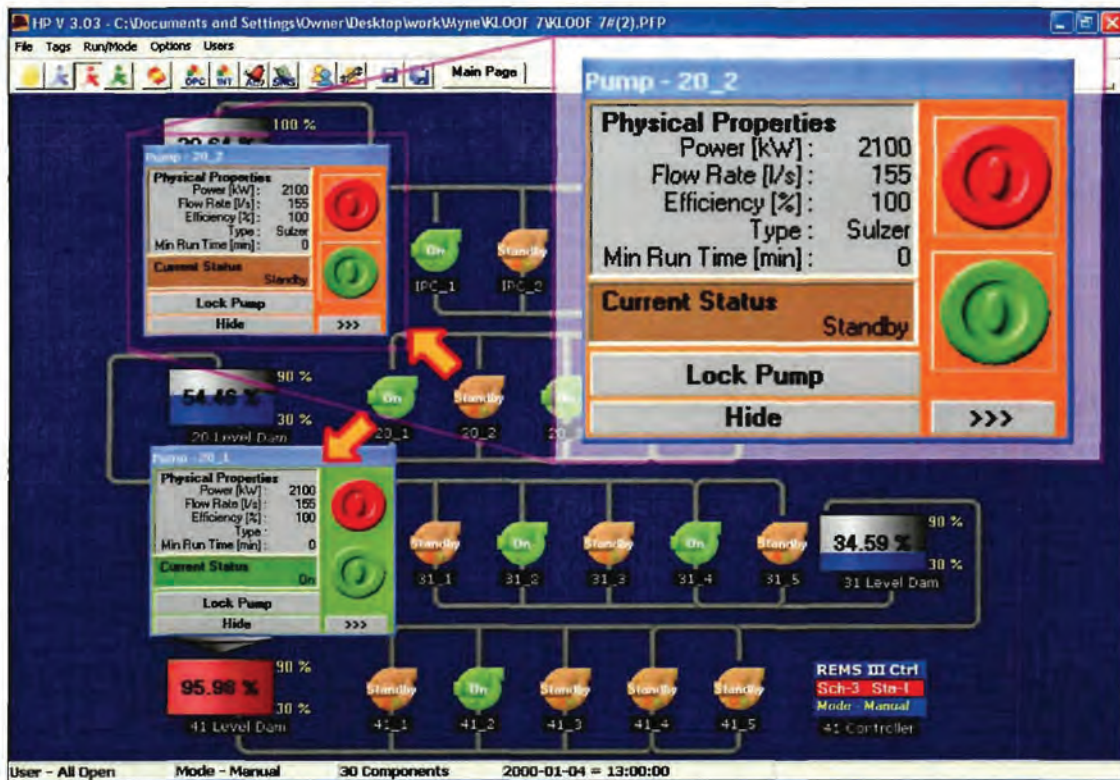


Figure 3-16 REMS pump control panel

The pump control panel for a specific pump, as seen in Figure 3-16, can be called up by clicking on that specific pump. More than one pump control panel can be called up during any time. The pump control panel consists of the following components:

Physical properties

This section gives the user information about the pump including the running power usage of the pump, the flow rate of the pump when operational, the efficiency, and the brand of the pump.

Current status

This portrays the current status of the pump. Note that the colour of the pump control panel also indicates the status of the pump.

Lock pump

This function is used to mark the pump as “locked”. This is done when the pump is not to be used or controlled by REMS. This function is usually used when the pump is undergoing maintenance.

On and off buttons

These red (stop) and green (start) buttons can be used to control the pump when the REMS platform is in manual mode. In auto mode these buttons are hidden as the pumps are controlled automatically.

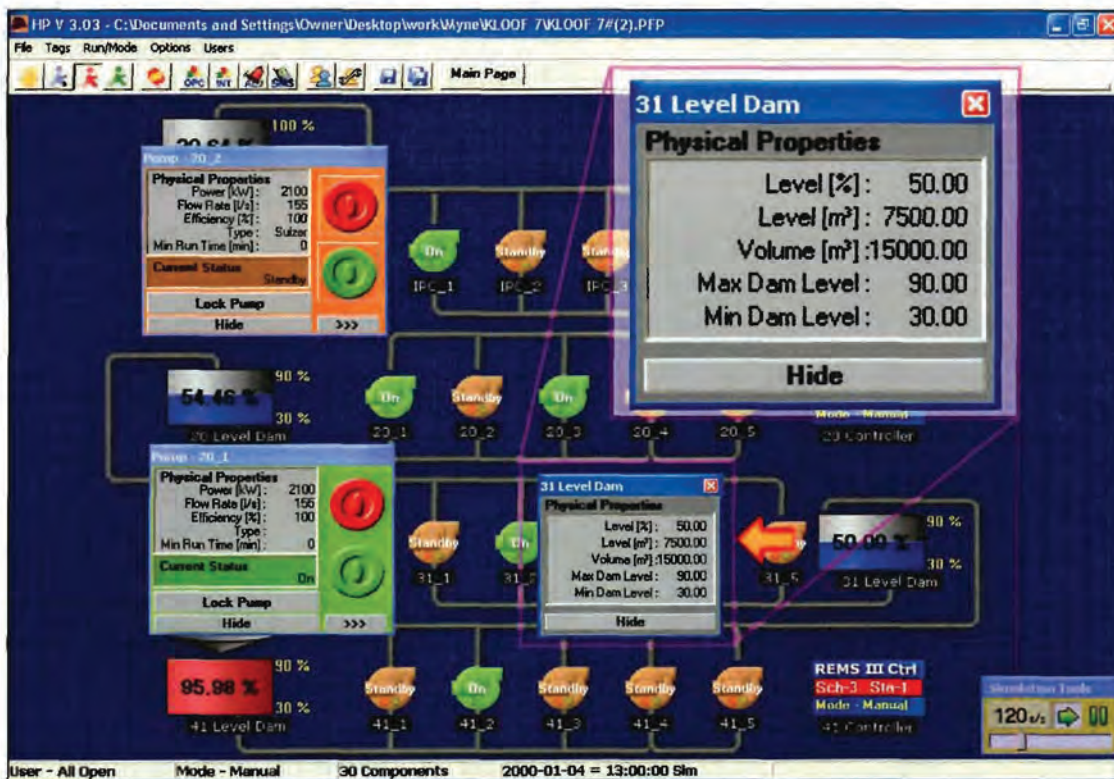


Figure 3-17 REMS dam control panel

The dam control panel as shown in Figure 3-17, does not offer any control over the dam as the name might suggest, but it does, like the pump control panel, provide information about the dam. It gives information such as the current level in terms of percentage and m^3 , the volume of the dam, and the permitted maximum and minimum levels.

3.5.10. Level controllers

A level controller component is created for every pump group in the system. Consider the next figure.

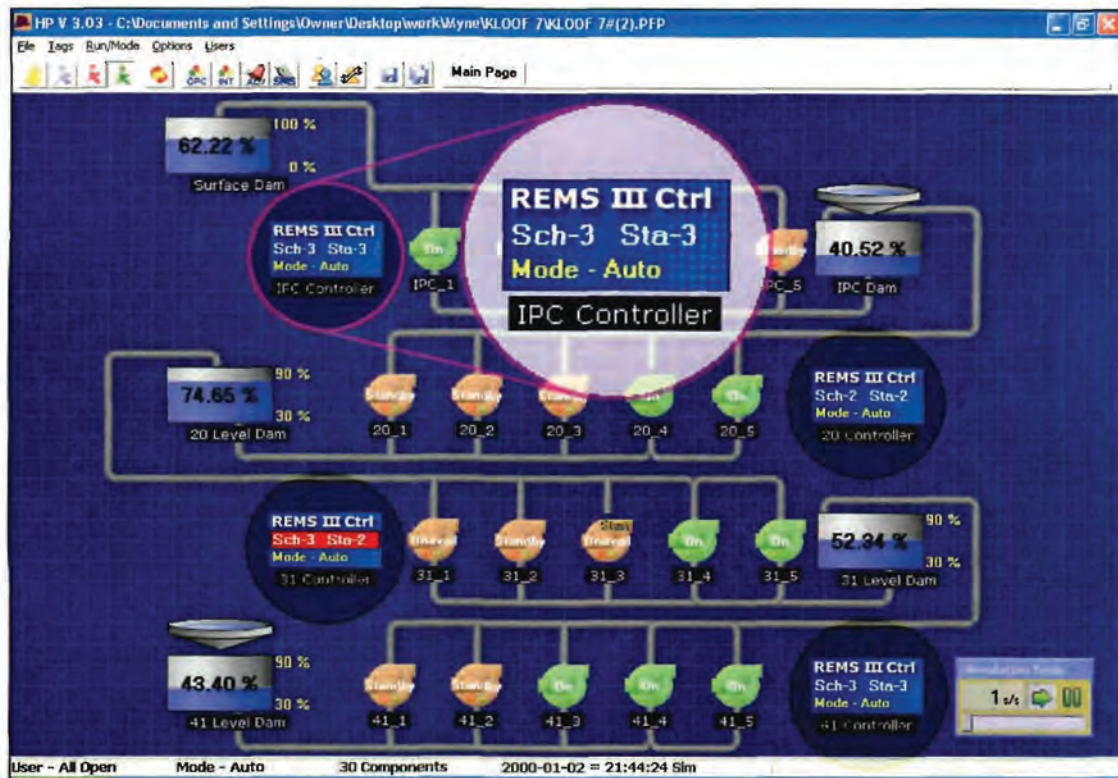


Figure 3-18 REMS level controllers

Figure 3-18 shows a system representation of a typical water pump system that has been built up in the REMS platform. The water pump system consists of four pump stations. There is a corresponding level controller for each of these pumping stations.

Each controller displays its descriptive name as set by the user. Also displayed within the blue rectangle is a schedule, indicated by the letters “Sch” and a status, indicated by the letters “Sta”. The status indicates the number of pumps currently running on the given pump group. The schedule is the amount of pumps that is supposed to be running according to the optimised schedule that has been calculated by REMS’ control algorithms.

In automated control the schedule and status of each pump group must match. Whenever this is not the case it is clearly indicated, as seen on the left bottom level controller in Figure 3-18.



Figure 3-19 REMS level controller editor

The level controller editor, as shown in Figure 3-19, is used to set up each level controller’s control parameters. The functions and tools available in this editor are as follows:

Description

The description field is used to give a unique descriptive name to each level controller.

Upstream control

This block contains the control parameters used by the upstream control algorithm and dictates how this algorithm operates. For an extensive explanation on the control algorithms, their function and how they operate, see section 2.5.2 of this thesis.

Pumps

This list contains the names and links to the pumps that make up this specific pump group. The links are interactive and double-clicking on them will activate the pump editors as explained and displayed in section 3.5.8 and Figure 3-14.

Downstream control

This block contains the control parameters used by the downstream control algorithm and dictates how this algorithm operates. For an extensive explanation on the control algorithms, their function, and how they operate, see section 2.5.2 of this thesis.

Maximum number of pumps

This setting dictates how many pumps are allowed to be simultaneously operational at any given time.

Minimum number of pumps

This setting dictates the minimum number of operational pumps on the pump station at any given time.

Start up delay, stop delay and toggle delay

These settings dictate how commands are sent to the SCADA. These are commands regarding the control of the pumps.

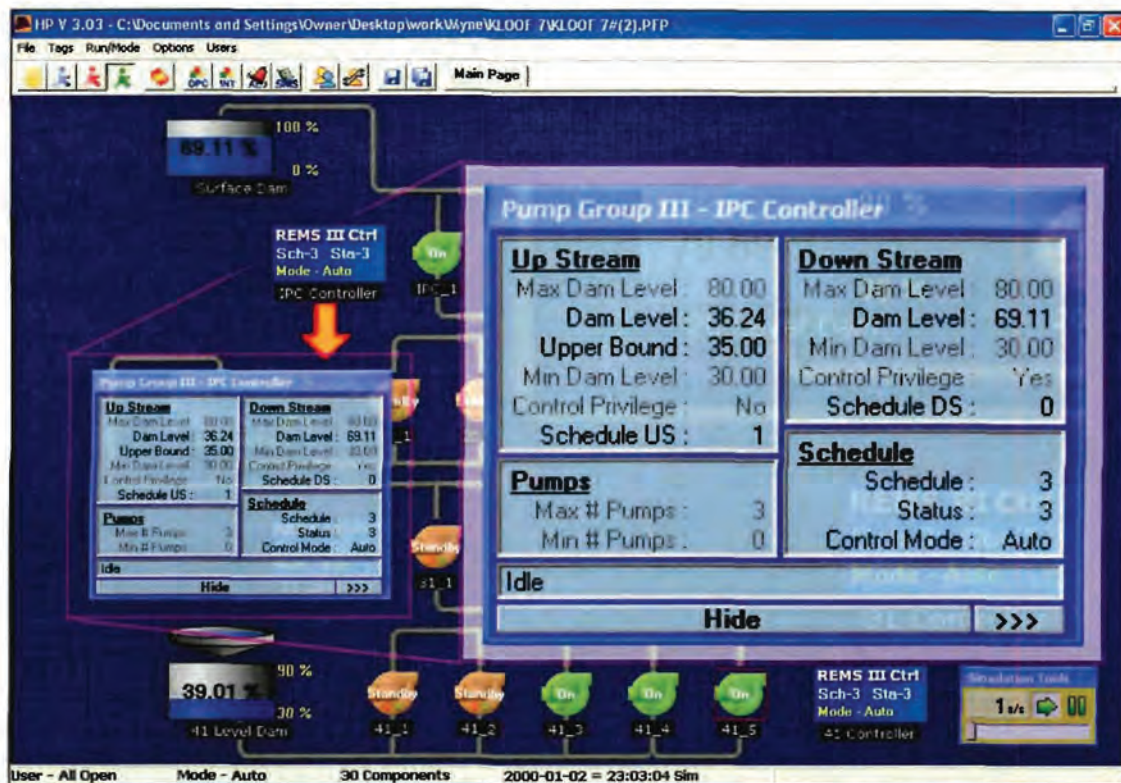


Figure 3-20 REMS level controller information panel

By clicking the level controller component on the platform, the level controller information panel is displayed. This information panel displays information regarding the control algorithms and the current control situation. Some of the more important constraints set on the level controller editor, are also displayed here.

3.6. System reliability and sustainability

3.6.1. A reliable structure

REMS is engineered for an environment where sustainability and reliability is of paramount concern. The user of such a system can suffer huge financial losses should the system fail to execute the required control. This can result, for example, in an underground mine dam flooding, or pumps running dry.

This problem was approached in the following way. The system was installed for a test period in a real-world application, and every failure that was experienced, which could conceivably have lead to a negative result, was documented. All such

documented events were thereafter seen as potential failures that could reoccur. The entire REMS system was subsequently improved in order to eliminate these potential failures.

This process does not end there. The identification of potential failures, and the actions taken to prevent them in future, is an ongoing process. Following are discussions of potential failures that have been identified thus far. Section 3.6.3 discusses how REMS was engineered and adapted to mend these potential failures.

3.6.2. Describing potential problem areas

i) System crash

The REMS system is an application developed in a programming environment and is hosted on a typical computer system. Therefore the possibility exists of it crashing or in other terms ‘stop responding’.

This kind of failure is highly dangerous, as in such an event no control or observation is executed on the relevant components, such as dams. This can result in pumps running dry or dams overflowing.

ii) Computer breakdown

As mentioned, REMS is a software application that runs on a computer. The risk of any of the components of the computer failing, leaving the computer inert, is very real. No computer designed today is without the risk of breaking down. Considering the intended operation environments sketches the problem and risk clearly.

iii) Control of equipment denied

As previously discussed, REMS controls the chosen electrical equipment via a SCADA system. In the event of maintenance to equipment, operators can mark equipment as being unavailable inside the SCADA application. This is called ‘lock out’ of equipment and is compulsory in industrial environments when hands-on attention is needed on equipment for inspection or maintenance. This is done to ensure the safety of the operators working directly with the equipment.

iv) OPC Connection to the SCADA lost

As discussed previously, REMS controls the chosen electrical equipment via a SCADA system. REMS communicates with this SCADA system using OPC protocol via a LAN hosted by the client or mine. For the REMS computer to have access to this LAN, it is assigned a password and is then connected to the LAN using a typical network adapter.

The REMS system loses its connection to this LAN as soon as the password that it has been given expires, or the network cables are broken or disconnected, or when the LAN itself undergoes maintenance.

3.6.3. Addressing problem areas

To eliminate the risk involved in system crashes or breakdown, the following operational practice has been developed. Situated at the HVACI headquarters is a system that constantly communicates with all the distributed REMS systems currently in operation. When this system is unable to establish communication with any one of these functional REMS systems, it alerts the designated person responsible for the operation of the system to take the appropriate action.

To further minimise downtime due to computer breakdown, the system was developed to run on dual redundant server based computers. This is a system where every component of the computer, such as processors and motherboards, operate in conjunction with an identical twin. In the event of a component failing, its twin will automatically take over its function until the failed unit is replaced.

REMS incorporates a series of alarm systems to alert operators or designated people in case control of equipment is denied or when communication to the SCADA is lost. These alarms include visual and audio prompts and even SMS's sent through the national cellular phone network.

3.7. Additional benefits of the new system

3.7.1. Automated control

The introduction of automated control on the water pumping systems was one of the huge contributions this research made. The introduction of automated control in many industrial systems increases the reliability of that system [90] [91]. Prior to the implementation of REMS the water levels of the different dams under, and above the ground were controlled manually. This process was only as reliable as the operators that were appointed to the job.

Another drawback of a system controlled by human operators is that optimised control is not always possible and sustainable. This could be because the operators did not have the necessary communication platform available to communicate vital information needed for optimum control, or that the operators did not have the necessary training and understanding to control a vast system in a correlated manner to produce optimum output.

By implementing REMS, the 24 hourly intervening of human operators was not necessary anymore. This resulted in a more reliable system that constantly controlled the dam levels for optimum output. REMS were designed with optimum control in mind, and therefore always aims to achieve optimum results.

Labour can amount to between 40% - 50% of the total expenses of a typical deep level mine [92]. By incorporating automated control this expense can be lowered.

3.7.2. Better safety and alarm systems

REMS was also designed with a multitude of different alarm systems. In the event of system component failure, even in components of the water pumping system, better and more diverse alarm systems were introduced.

A very good example of this was the introduction of the SMS alarming and reporting system. REMS can be set up to send any type of alarm to the cell phone of any given person in the form of an SMS. The benefits of this are self-evident.

3.7.3. Comprehensive data logging

Another benefit REMS contributed was the comprehensive data logging it provided. REMS was designed to log extensive data as it is needed to optimise the control philosophy. Expanding the functionality of the system in such a way that the data was available to the user, resulted in huge benefits. This data can be used to draw up sustainability curves of pumps or to do research on the performance of the system etc.

3.7.4. Maintenance

Normal maintenance on system components is also improved with the help of this new technology. Maintenance is critical in assuring sustainable performance and operation of equipment [93]. Traditional maintenance consists of scheduling workload between components and servicing or replacing crucial parts and components at set work-hour intervals.

This is eased by REMS, as it can be set to automatically distribute workload as dictated by the component technicians. REMS also reports on the work-hours and schedule history of every component in the system. This is then used to schedule the servicing and replacement of components. Predictive maintenance such as this contributes huge benefits to a mine [94].